

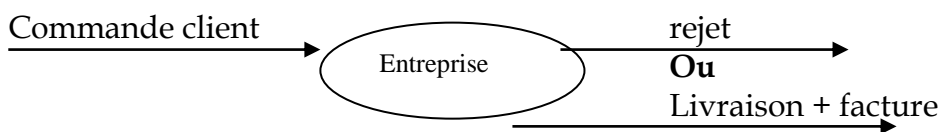
METHODE D'ANALYSE MERISE

1-introduction

C'est une **méthode systémique** qui s'appuie sur la théorie des systèmes en vue d'une **approche systémique et globale** d'un système d'information d'une organisation.

1-1 - Définition d'un système

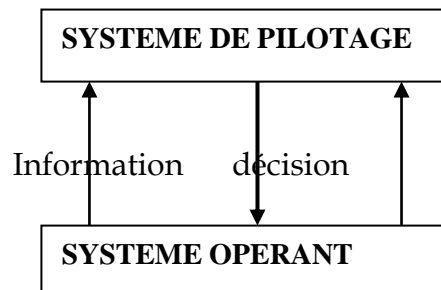
Un système est un ensemble d'éléments matériels ou immatériels (hommes, machine, méthodes, règles, etc. ..) en interaction et transformant, grâce à un processus, des éléments d'entrées en éléments de sorties.



Tout système d'entreprises est constitué par :

- **Système de pilotage** qui dirige l'entreprise et maintient le cap sur les objectifs choisis.

- **Système opérant** : assure la production en vue de la réalisation de certaines tâches



Les informations entre le système de pilotage et le système opérant sont véhiculées par un troisième système, appelé: **système d'information** qui distribue les informations nécessaires à tous les points de l'organisation.

1-2 définition (2)

On peut distinguer 3 sous-systèmes :

Le système de pilotage : direction, contrôle, décisions, définition des objectifs

Le système opérant : réalisation des tâches (des opérations)

Le système d'information : Interface entre les deux systèmes précédents.

Le SI est la mémoire de l'organisation.

Exemple : Une voiture

Système de pilotage : le chauffeur

Système opérant : le moteur (flux d'entrée - carburant, flux de sortie - énergie)

Système d'Information : le tableau de bord

Le système de pilotage : Contrôle et dirige l'ensemble

Finalité : Élaborer des règles de gestions et produire des décisions

Le système opérant : transforme les flux d'entrées en flux de sorties

Finalité : La transformation des flux

Le SI transmet les ordres au système de production et renvoie au système de pilotage les informations

Finalité : Assure une représentation d'un certain nombre d'éléments de l'entreprise

Gère la mémoire collective de l'entreprise

Offre des services d'accès rapide à l'information

1- 3 Le fonctionnement d'un système

L'approche systémique (relatif à un système dans son ensemble), appliqué au système de l'entreprise, offre un modèle proche de la réalité

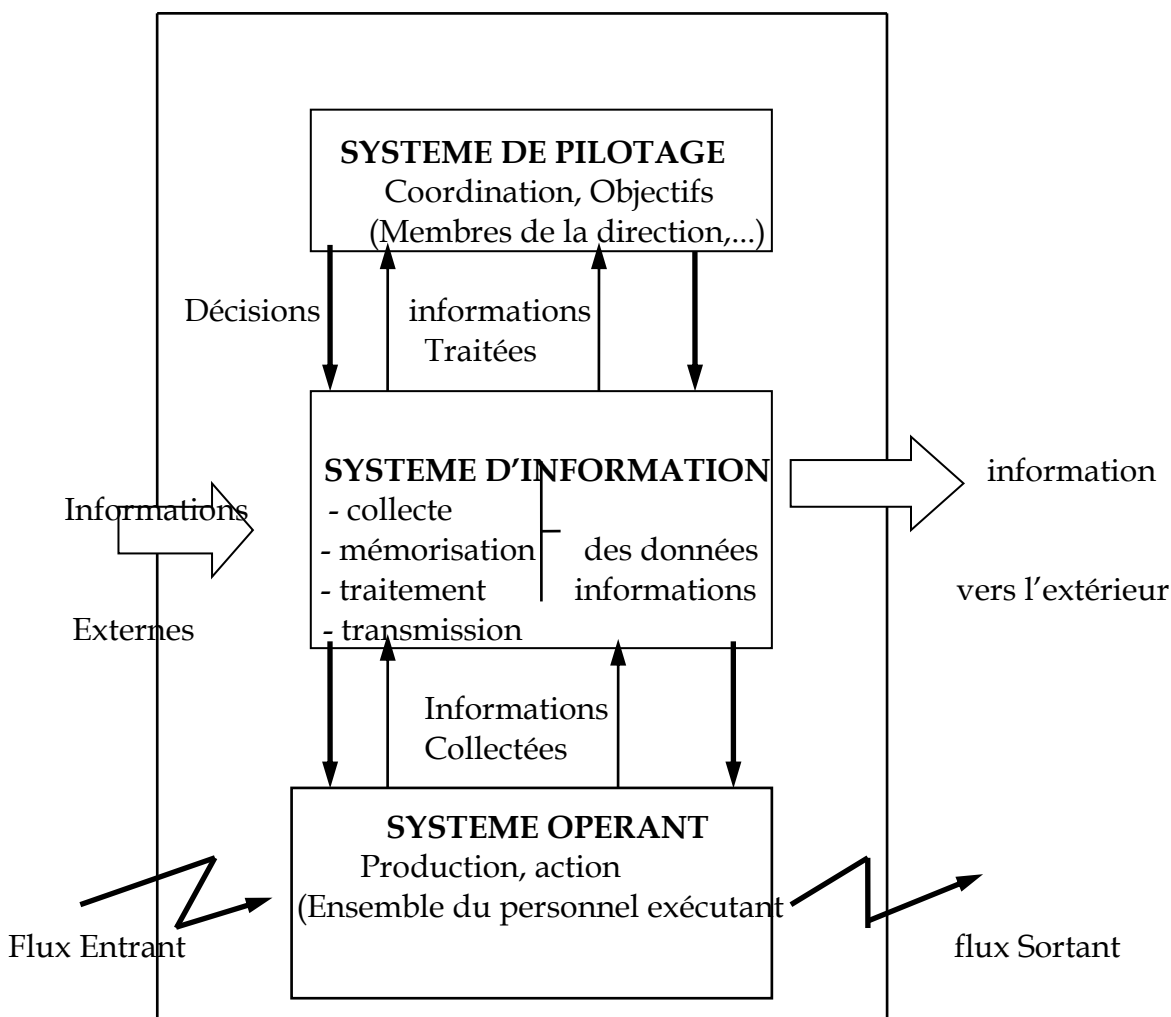


Schéma de l'approche systémique d'un système d'une entreprise

Le système d'information réalise alors quatre fonctions essentielles:

- ✓ **Collecter** les informations provenant des éléments du système ou de l'environnement externe au système → fonction dynamique
- ✓ **Mémoriser** les données manipulées par le système → fonction statique
- ✓ **Traiter** les données stockées; → fonction dynamique
- ✓ **Transmettre** des informations vers les composants du système ainsi que vers l'environnement externe au système → fonction dynamique

On a alors trois fonctions dynamiques et une fonction statique. Le système d'information est donc **dynamique**.

La conception du **SI** nécessite des méthodes permettant de mettre en place un modèle (un modèle consiste à créer une représentation virtuelle d'une réalité de telle façon à faire sortir les points auxquels on s'intéresse). Ce type de méthode est appelé **analyse**. Il existe plusieurs méthodes d'analyse, la méthode la plus utilisée étant la méthode **MERISE**.

La méthode MERISE est une méthode **de conception d'un système d'information**.

2- Présentation de la méthode merise

MERISE est une méthode de conception, de développement et de réalisation de projets informatiques. Le but de cette méthode est d'arriver à concevoir un système d'information. La méthode MERISE est basée sur la séparation des données et des traitements.

La séparation des données et des traitements assure une longévité du **SI**.

2-1- La vocation de MERISE

Sa vocation est double:

- méthode de développement de SI
- méthode de conception des SI

La démarche de développement d'un système d'information est conduite suivant 3 axes appelés cycles

Le cycle de vie : Se situe sur une échelle de temps du point de départ à l'exploitation du système.

Selon merise est découpé en 3 périodes : la conception du **SI**, la réalisation du programme, la maintenance

Le cycle de décision : représente l'ensemble des choix qui doivent être fait durant le déroulement du cycle de vie. L'entreprise s'assure que le système correspond aux objectifs, et prend différents types de résolutions.

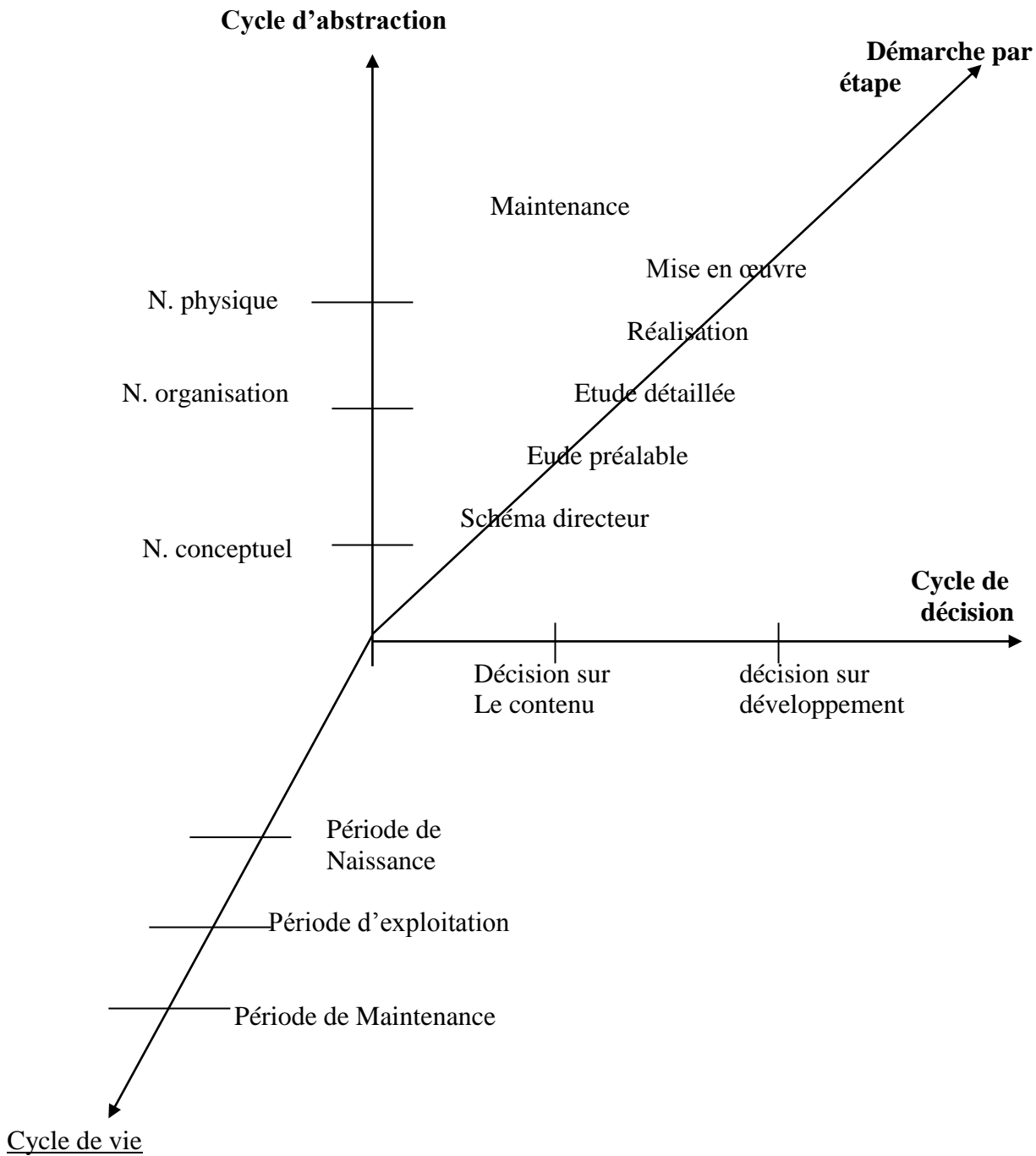
Le cycle d'abstraction : A pour but de découper le **SI** en niveaux :

Le niveau conceptuel, le niveau logique et le niveau technique.

A chaque niveau correspondent une préoccupation et un ensemble de **modèles** pour la représentation des données et de traitements.

MERISE est une méthode de réalisation de projets informatiques.

Suivant les 3 cycles on défini les étapes de la réalisation de projet informatique.



2-2- Approche par étapes

Schéma directeur: Pont entre stratégies et besoins d'information.

Etude préalable: Etudie par domaine le projet à mettre en œuvre et son interface.

Etude détaillée: Description fonctionnelle de la solution à réaliser.

Réalisation : Production du logiciel, Permet d'obtenir le logiciel testé sur un jeu d'essai.

Mise en oeuvre: Exécuter toutes les actions (formation, installation des matériels, initialisation des données, réception, ...) qui permettront d'aboutir au lancement du système auprès des utilisateurs.

Maintenance: Mise à niveau éventuel des applications. Prolonger la durée de vie.

Ses **atouts** en tant que méthode de conception:

2-3- Approche par niveaux d'abstraction

La méthode MERISE est basée sur la séparation des données et des traitements.

MERISE utilise une démarche de **modélisation à trois niveaux**. A chaque niveau correspondent à un modèle pour la représentation des données et à un modèle pour la représentation de traitements.

Un formalisme de représentation est associé à chaque modèle.

L'ensemble de ces trois niveaux constitue **le cycle d'abstraction** qui met en évidence les règles qui régissent le SI.

La méthode MERISE distingue trois niveaux d'analyse :

Le niveau conceptuel

- spécifie les choix de gestion: **quoi faire ?**

Permet de décrire:

- **pour les données**: la signification de chacune d'elle et les rapports qui existent entre **objet, propriété, relation**.
- **pour les traitements**: événement d'entrée, synchronisation, opération, événement de sortie (ou résultat), processus.

Le niveau logique et organisationnel

- spécifie les choix d'organisation: **qui fait? quand? où?**

Permet de décrire:

- **pour les données**: les structures logiques qui sont **Record, set, relation, tableau**
- **pour les traitements**: **poste de travail, la nature de traitement, durée**

Le niveau physique et opérationnel.

- spécifie les choix techniques: **comment faire?**

Permet de décrire:

- **pour les données**: l'organisation en **fichiers** (ou en base de données fichier).
- **pour les traitements**: le découpage en unités de traitement (UT) et la structure d'enchaînement des (UT) , la définition des , **maquette d'écran, programme**.

On peut résumer l'architecture des trois niveaux du cycle d'abstraction par le schéma suivant:

| NIVEAUX | DONNEES | TRAITEMENT |
|-------------------------------|---|---|
| CONCEPTUEL | MODELE CONCEPTUEL DES DONNEES (MCD) | MODELE CONCEPTUEL DES TRAITEMENTS (MCT) |
| ORGANISATIONNEL ET LOGIQUE | MODELE LOGIQUE DES DONNEES (MLD) | MODELE ORGANISATIONNEL DES TRAITEMENTS (MOT) |
| OPERATIONNEL ET PHYSIQUE | MODELE PHYSIQUE DES DONNEES (MPD) | MODELE OPERATIONNEL DES TRAITEMENTS (MOPT) |

Le parcours des différents niveaux peut alors être chronologiquement comme suivant

- 1) Etude e l'existant
- 2) (En parallèle, par deux équipes différentes, si cela est possible)

| | |
|-------------------------------|--------------------------------------|
| Modèle conceptuel des données | Modèle conceptuel des traitements |
| | Modèle organisationnel du traitement |

- 3) validation
- 4) Modèle logique des données
- 5) (Ensemble)

- Modèle physique des données

- Modèle opérationnel des traitements

Aux étapes précédentes s'ajoute deux étapes l'une en amont constituée de **l'étude de l'existant** ,et l'autre en cours de conception , ayant pour objet de s'assurer que les données du modèle conceptuel permettent les traitements du modèle organisationnel : **validation**

3- Etude de l'existant :

L'objectif est de prendre connaissance du domaine dont l'entreprise souhaite améliorer le fonctionnement et de recenser l'ensemble des objectifs que poursuit l'entreprise.

3-1- recueil de l'existant :

Le recueil de l'existant se fait auprès de la direction et des postes de travail.

Au niveau de la direction, il s'agit de prendre connaissance du problème posé, cerner les principaux postes de travail, décrire les interfaces avec d'autres projets et délimiter le champ d'étude.

Au niveau des postes de travail, recueillir toutes les opérations effectuées et toutes les informations (données) manipulées sur le poste de travail

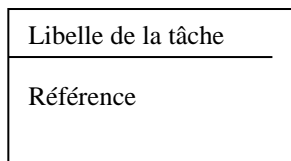
Au fil de l'interview on construira un diagramme tâche -documents

Avantage du diagramme : de guider l'interview et de poser des questions :

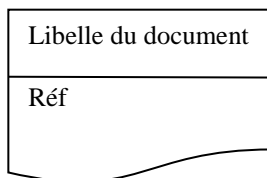
- pour chaque tâche quels résultats produit elle et par quoi est elle déclenché,
- pour chaque document à quoi sert il ?

Formalisme

Le symbole est le suivant



tâche



document



sens de circulation de l'information



transmission par tél ,fax ..

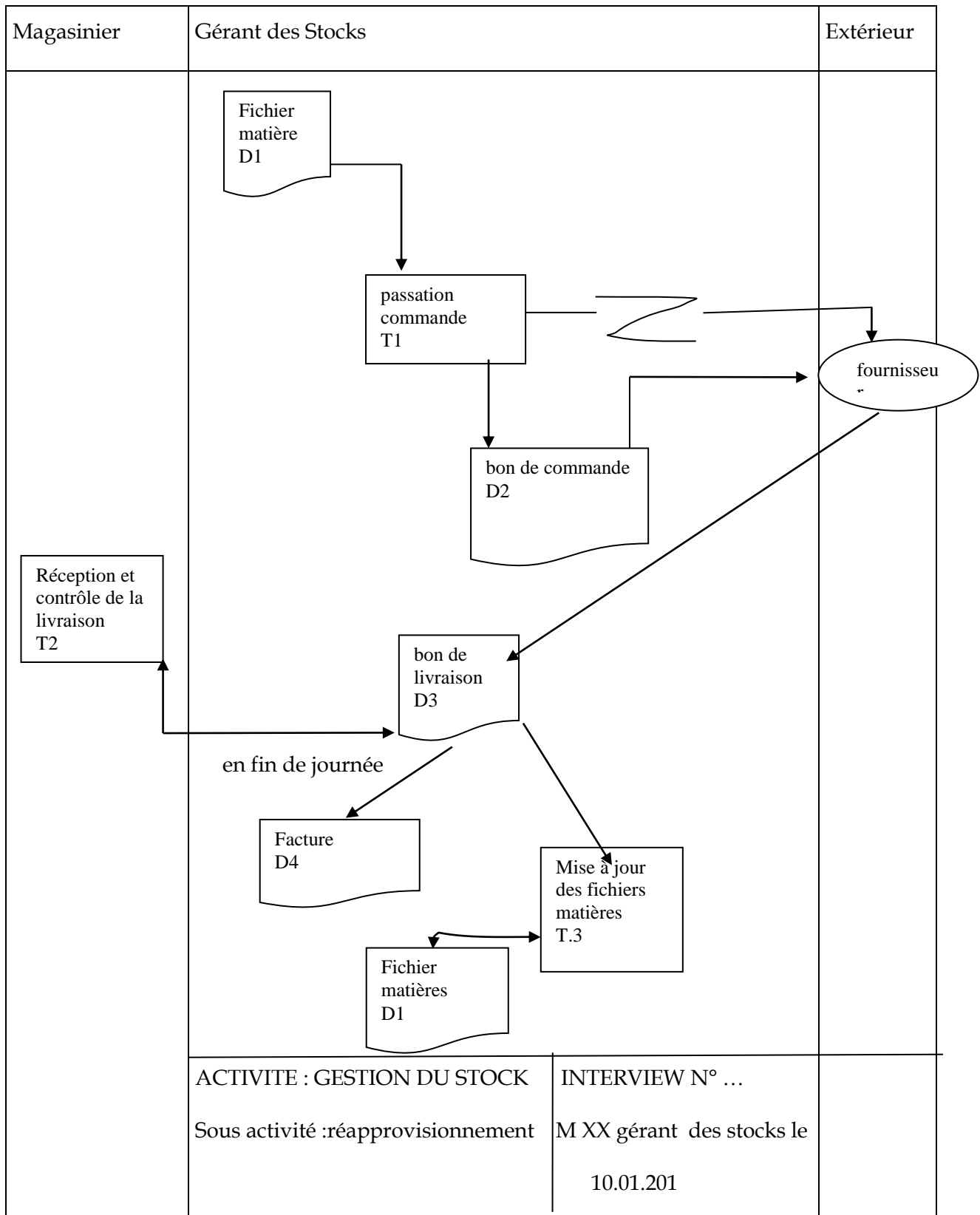


Mise à jour



Poste de travail exclu du champ de l'étude

EXP :le diagramme suivant extrait de l'interview du responsable de la gestion du stocks d'un restaurateur



La feuille d'accompagnement d'écrivant les documents

| N° de document | Libelle -rôle | N° de tâche |
|----------------|--|-------------|
| D1 | Fichier matière : Etat et valeur du stock | T1 |
| D2 | Bon de commande :Article et quantité à commander | T1 |
| D3 | Bon de livraison :il décrit la livraison effectuée | T3 , T2 |
| D4 | Facture : service comptabilité | |

Celle des tâches

| N° de tâche | La tâche | Poste de travail | Fréquence et volume | Document en entrée | Document en sortie |
|-------------|---|------------------|--------------------------|--------------------|--------------------|
| T1 | Etablissement d'une commande à partir d'un besoin | Gérant des stock | 4à10 par jour | D1 | D2 |
| T2 | Vérification du bon livraison contrôle de quantité et qualité | magasinier | 8 produits par livraison | D3 | D3 |
| T3 | Inscription de livraison sur le fichier matière | Gérant du stock | 10à 13 par jour | D3 | D4 |

(Notion de tâche :ensemble d'action)

3-2- Recensement des règles

La manipulation des données et l'exécution des tâches au sein d'un poste de travail sont toujours gouvernées par un certain nombre de règles traduisant des objectifs ou des contraintes. Ces règles sont formulées au fur et à mesure de l'avancement de la méthode (règles de gestion, règles d'organisation et règles techniques).

Les règles de gestion sont associées au niveau conceptuel et décrivent le « **quoi** » de l'entreprise. Elles l'expriment d'une façon dynamique les actions qui doivent être accomplies, mais aussi statique en détaillant la réglementation jointe à ces actions

Exemple

Règles d'action : décrira les actions que doit accomplir l'entreprise

- action dynamique : un inventaire doit être dressé périodiquement
- action statique : tout produit livré sera entré en stock

Les règles d'organisation : associées au niveau organisationnel, décrivant le « **où** », le « **qui** » et le « **quand** ». Traduisent l'organisation mise en place dans l'entreprise afin d'atteindre les objectifs fixés

Exemple :

- l'état du stock sera suivi par une gestion informatisée au magasinier
- l'enregistrement des livraisons sera fait en fin de journée
- les commandes à la centrale d'achats ne pourront être passées que le mardi

Les règles techniques liées au niveau opérationnel décrivant le « **comment** » en expriment la condition technique de mise en œuvre des tâches

Exemple :

- le système d'exploitation permettra un travail multiposte (plusieurs poste de travail pourront simultanément consulter l'état du stocks)

3-3- critique de l'existant

La critique de l'existant consiste, essentiellement, à faire un diagnostic de la gestion et de l'organisation de l'entreprise et de définir les orientations à suivre pour remédier aux anomalies détectées.

3-4 dictionnaires des données

Le dictionnaire des données contient l'ensemble des données recensées lors de l'étude de l'existant.

Les documents recueilles lors de l'étude de l'existant

D1 : fichier matière

- N° magasin
- Nom magasin
- Code produit
- Libellé produit
- Quantité en stock
- Date modification

D2 : bon de commande

- N° commande
- Date commande
- N° fournisseur
- Adresse fournisseur
- N° client
- Adresse client
- Libellé produit
- Quantité commandée
- P U

D3 : bon de livraison

- N° Livraison
- date de livraison
- N° fournisseur
- Adresse fournisseur
- libellé produit
- quantité livrée
- N° client
- Adresse client
- Mode de livraison
- Total HT
- Total TVA
- Total TTC

D4 : facture

- N° Facture
- Date facture
- N° de livraison
- Code produit
- Désignation
- Quantité
- P U
- Remise
- N° client
- Adresse client
- NET à payer
- Mode de payement
- Total HT
- Total TVA
- Total TTC

Afin d'éviter un certain nombre d'anomalies sur les données retenues, des considérations relatives à la structure et à la nature des données sont à prendre en compte.

Une propriété élémentaire correspond à une **donnée** qui ne résulte pas d'une concaténation de propriétés.

Exp : L'adresse, composée de la rue, de la ville et du code postal est une donnée non élémentaire. Donc il faut la décomposer en Rue, ville, code postale

Les propriétés élémentaires sont **de trois ordres** :

- **les identifiants** ou donnée synthétiques représentatives d'objets selon la loi : à tout objet existant est associé un identifiant et un seul.

- **les libellés et les données qualitatives** sont des propriétés alphanumériques qui ne participent à aucun calcul.

- **les montants et les données numériques** pouvant servir d'opérandes dans les calculs. Le taux de TVA, le pourcentage à appliquer sur une valeur sont des exemples de données numériques intervenant dans des calculs.

Les données calculées sont obtenues directement à partir d'un calcul. Au niveau conceptuel, généralement ne pas mémoriser ce type de données.

Une propriété (donnée) est appelée **signifié**, tandis que le symbole par lequel il est représenté a pour nom **variable ou code**

Nature des propriétés

- **Les données de situation** varient avec le temps ou suivant les périodes, par exemple un crédit d'un compte client au sein d'une institution financière.

C'est également le cas de la température constamment variable avec le temps.

- **Les données mouvement**, elles résultent de circonstances spécifiques, elles n'existent que parce qu'un événement a eu lieu.

Par exemple :

- la quantité d'un produit commandé suppose qu'une commande concernant ce produit a été effectuée.

- le cas du nombre de passagers dans un avion lors d'un vol.

- **Les données signalétiques** ou stables demeurent généralement inchangées.

Elles ne peuvent pas être modifiées quelles que soient les circonstances. C'est le cas des date et lieu de naissance. On considérera comme données signalétiques toutes les propriétés qui ne sont ni mouvement, ni de situation.

Les propriétés collectées sont rassemblées dans un Dictionnaire de données. Les colonnes du Dictionnaire indiquent respectivement pour chaque propriété élémentaire :

- **le variable ou code**,
- **la signification**,
- **le type** (alphabétique, alphanumérique, ou numérique),
- **la longueur** donnée en nombre de caractères,
- **la nature** (signalétique, mouvement ou situation)

EXP : Le dictionnaire de données

D'après les documents de données recueilles lors de l'étude de l'existant, on établit Le dictionnaire de données.

| Variable | Signification | Type (AN, A, N) | LG | Nature (SI, ST, M) | Remarques |
|------------|--------------------------|-----------------|----|--------------------|-----------|
| NumMag | Numéro magasin | N | 1 | SI | |
| NomMag | Nom Magasin | A | 10 | SI | |
| Codprd | Code du produit | AN | 5 | SI | |
| libelléprd | Libellé du produit | A | 10 | SI | |
| QTstock | Quantité en stock | N | 3 | ST | |
| DatMod | Date de modification | AN | 8 | M | JJ/MM/AA |
| Num Comd | Numéro commande | N | 5 | SI | |
| DatComd | Date commande | AN | 8 | M | JJ/MM/AA |
| NumFour | Numéro fournisseur | N | 5 | SI | |
| AdrFour | Adresse fournisseur | AN | 30 | SI | |
| NumCl | Numéro client | N | 5 | SI | |
| AdrCl | Adresse client | AN | 30 | SI | |
| QTComd | Quantité commandée | N | 3 | M | |
| Numliv | Numéro de livraison | N | 5 | SI | |
| Datliv | Date de livraison | AN | 8 | M | |
| QTliv | Quantité livrée | N | 3 | M | |
| Modliv | Mode de transport | A | 10 | M | |
| Numfat | Numéro de facture | N | 5 | SI | |
| Datfat | Date facture | AN | 8 | M | |
| Designat | Nom du produit | A | 10 | SI | |
| QTfat | Quantité facturée | N | 3 | M | |
| P U | Prix unitaire du produit | N | 5 | ST | |
| Remise | Remise sur le produit | N | 4 | M | |
| Taux TVA | Taux de TVA | N | 4 | M | |

Lexique:

| | | |
|----------------------|-----------------|-------------------|
| AN : Alpha Numérique | | SI : Signalétique |
| N : Numérique | CA : Calculé | ST : Situation |
| A: Alphabétique | E : Elémentaire | M : Mouvement |

3-5 Epurations du dictionnaire des données

Les données recueillies dans les diverses interviews de poste de travail fait généralement apparaître des incohérences qu'il convient d'éliminer.

Ces incohérences sont de deux types:

a) De synonymes

Des noms différents désignent la même réalité

Exp : numéro de commande et référence commande, agent et employé, Marchandise et produit, ...

On dit aussi qu'existent alors deux signifiants pour un même signifié.

b) Des polysèmes

Un même nom désigne deux réalités distinctes.

Exp :café désigne l'établissement et la boisson, prix désigne le prix d'achat et le prix de stockage,...

On dit que l'on a un signifiant pour deux signifiés.

On supprimera les synonymes et les polysèmes de façon à établir une bijection Entre l'ensemble des signifiants et celui des signifiés

4- LE MODELE CONCEPTUEL DES DONNEES : MCD

Le modèle conceptuel des données est une représentation statique du système d'information de l'entreprise. Mais pas figée, car un modèle est par principe évolutif. En effet, il vit en même temps que la réalité qu'il décrit.

4-1- Graphe des Dépendances Fonctionnelles (GDF)

Définition

Une Dépendance Fonctionnelle (**DF**), c'est le fait de relier de manière unique une propriété à d'autre propriété ou un ensemble de propriétés

Ce lien est représenté par une flèche: par exemple, $A \rightarrow B$ est une dépendance fonctionnelle de A vers B et se lit: à partir de A, on peut déterminer B.

Exp : Soit une liste des données recensées dans un établissement scolaire

- nom de l'élève
- prénom de l'élève
- adresse de l'élève
- matière enseignée
- nombre d'heure
- nom de la classe
- nom du professeur
- note
- numéro de salle
- numéro de l'élève
- numéro de la matière
- numéro de la classe

Les règles de gestion appliquées dans cet établissement

RG1 : A chaque classe est attribuée une et une seule salle de cours

RG2 : chaque matière n'est enseignée que par un seul professeur

RG3 : pour chaque classe et chaque matière est défini un nombre fixe d'heures de cours

RG4 : l'établissement gère les emplois du temps des professeurs et des élèves ainsi que le contrôle des connaissances

La lecture des données suggère des regroupements en sous-ensembles présentant une cohérence interne et une autonomie les uns vis-à-vis des autres

Par exemple, on parlera de l'objet « **Elève** » dont les propriétés sont

- nom de l'élève,
- prénom de l'élève,
- adresse de l'élève.

De l'objet « **classe** » les propriétés sont :

- nom de la classe
- numéro de salle

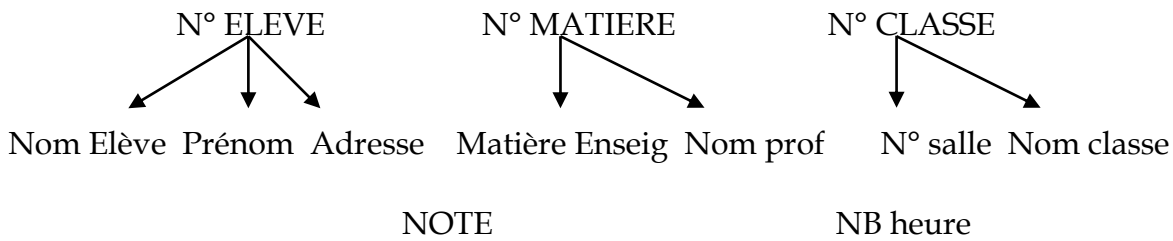
De l'objet « **Matière** » les propriétés sont :

- matière enseignée
- nom du professeur

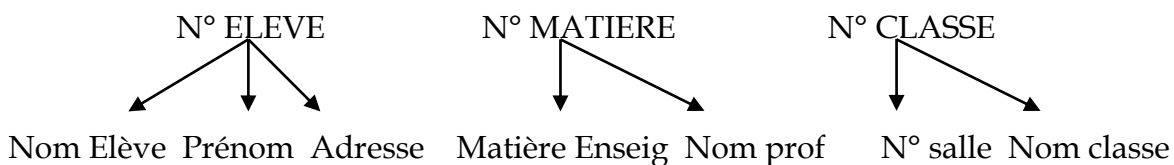
En revanche, la donnée « **Note** » ou la donnée « **Nombre d'heures** » n'a pas de signification seule. On doit préciser:

- pour la note, le nom de l'élève et la matière dans laquelle elle a été obtenue,
- pour le nombre d'heures, la classe et la matière enseignée.

Pour la **note** n'existe que par rapport aux objets « Elève » et « Matière », pour le **nombre d'heures** n'existe que par rapport aux objets « Classe » et « Matière ».



Chacun de ces trois sous-ensembles a un sens en lui-même, indépendamment des autres, et présente bien une cohérence interne. On dira que l'on a ainsi défini trois **objets** comportant des propriétés. La propriété de l'en tête de chaque sous-ensemble est l'identifiant de l'objet



On utilise le **DF** élémentaire pour définir les objets conceptuels du système

Après avoir défini les objets, on détermine les liaisons entre objet par l'intermédiaire de leurs identifiant. On a :

Identifiant de A \longrightarrow Identifiant de B

La dépendance fonctionnelle entre deux identifiants de deux objets doit être justifiée par 2 types de règles :

- Une règle de gestion ou une contrainte,
- Un lien stable.

Exp :

Soit la règle de gestion suivante

Un enseignant exerce à titre principal dans un seul établissement représente une contrainte qui a été fixée.

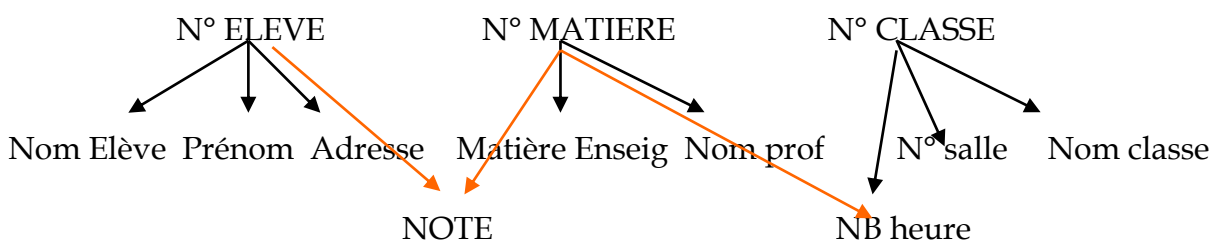
Numéro enseignant \longrightarrow numéro établissement

La dépendance fonctionnelle Toujours vérifiée: traduit un lien stable.

Numéro ville \longrightarrow numéro pays

G D F de l'application

Pour la **note** n'existe que par rapport aux objets « **Elève** » et « **Matière** », pour le **nombre d'heures** n'existe que par rapport aux objets « **Classe** » et « **Matière** ».



L'ensemble des dépendances fonctionnelles d'une application peut être représenté par une **arborescence** appelée **graphe des dépendances fonctionnelles (GDF)**.

4-1-1- Propriétés des dépendances fonctionnelles

Considérons les quatre ensembles d'attributs (propriétés et leurs valeurs) X, Y, Z, et W

Transitivité

Si $X \rightarrow Y$ et $Y \rightarrow Z$, on a : $X \rightarrow Z$.

Par exemple ville \rightarrow continent pourrait résulter de: ville \rightarrow pays et pays \rightarrow continent.

La transitivité doit être écartée du GDF

Réflexivité

Si X est un sous ensemble de Y , on a $X \rightarrow Y$

D'où l'on déduit :

Tout ensemble représentant une propriété est un sous ensemble de lui-même et est relié à lui-même par une dépendance fonctionnelle, ce qui se traduit par la DF réflexive $X \rightarrow X$

Toutefois dans les schémas qui suivent les DF réflexives ne sont pas représentés

Augmentation

A partir de $X \rightarrow Y$, on peut écrire : $XZ \rightarrow YZ$

Union

Si $X \rightarrow Y$ et $X \rightarrow Z$, on a : $X \rightarrow YZ$

Dépendance fonctionnelle élémentaire.DF

$X \rightarrow Y$ est une dépendance fonctionnelle élémentaire si et seulement si:

- X n'est pas un sous-ensemble de Y et si
- Il n'existe pas une partie X' de X telle que $X' \rightarrow Y$.

Exp : Code Produit alimentaire \rightarrow Code Produit

N'est pas une dépendance fonctionnelle élémentaire car le produit alimentaire est un sous-ensemble de L'ensemble des produits.

On utilise le **GF** élémentaire pour regrouper des sous -arbres les propriétés qui caractérisent logiquement le même objet

Construction d'un GDF

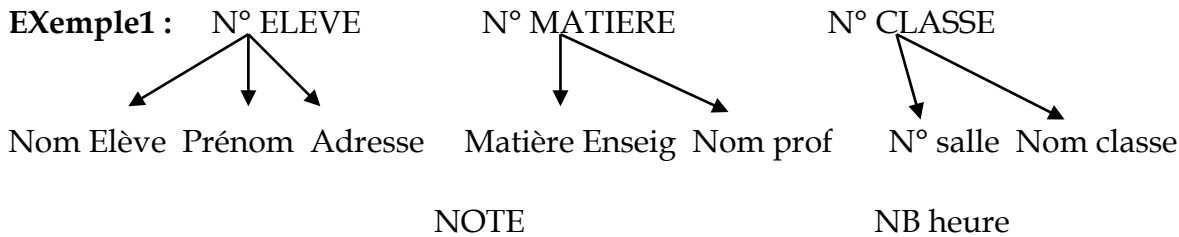
L'ensemble des dépendances fonctionnelles d'une application peut être représenté par une arborescence appelée graphe des dépendances fonctionnels **GDF**.

Lorsqu'on met en œuvre le modèle conceptuel des données, il peut être avantageux de chercher les dépendances fonctionnelles entre les propriétés. il y a une façon systématique de procéder qui consiste à considérer les propriétés deux à deux et de voir s'il peut exister entre elles une dépendance fonctionnelle : sur des cas simples, c'est envisageable , mais lorsqu'on dispose d'un nombre important de données ,le procédé s'avère fastidieux .on peut alors faire appel à une approche plus pratique : on regroupe à l'avance dans des sous- arbres les propriétés qui caractérisent logiquement la même entité : c'est à partir du sommet de chaque sous -arbre qu'on détermine les dépendances fonctionnelles propres à chaque entité. Les identifiants sont les sommets des sous - arbres parce qu'à partir d'un identifiant, on détermine les autres propriétés de l'entité.

4-2 Association

Définition

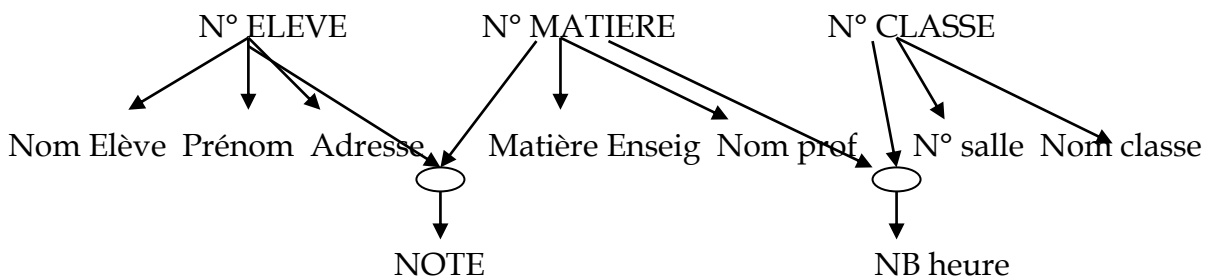
Dans une dépendance fonctionnelle, les liaisons entre deux objets Ex et Ey reposent sur le lien stable, les règles de gestion : (l'identifiant de l'objet Ex dépend de l'identifiant de l'objet Ey). le lien entre deux objets aboutit à un troisième objet, on parle d'association (ou **relation**). Elle peut être porteuse de propriétés reflétant l'interdépendance entre les objets associées.



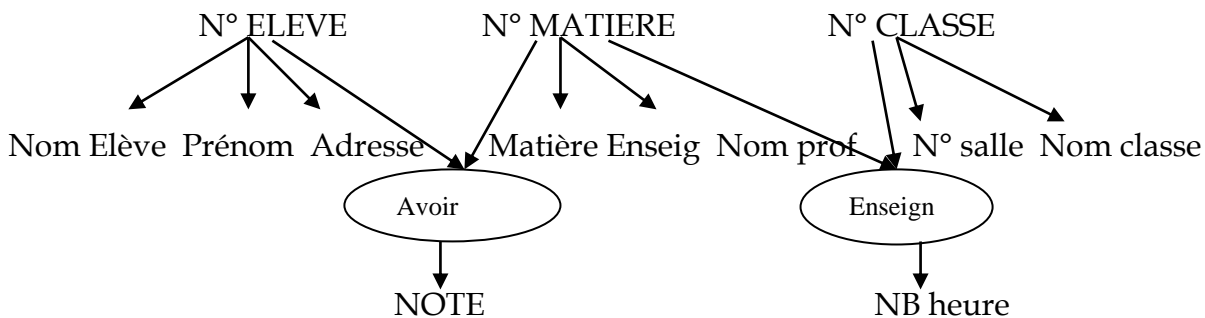
On dira que les données **note et nombre d'heures**, qui ne peuvent être portées par un seul objet, sont portées par des **relations** sur les objets. C'est ainsi que l'on définira, sur les objets « Elève » et « Matière », la relation « Avoir pour note » dont la propriété sera la note obtenue.

De même, sur les objets « Classe » et « Matière », on définira la relation « Enseigner durant »; sur laquelle figurera le nombre d'heures.

Le Graphe des dépendances fonctionnelles sera comme suivant



le lien entre deux objets aboutit à un troisième objet, on parle d'association (ou **relation**). Elle peut être porteuse de propriétés reflétant l'interdépendance entre les objets associées.



Exp2

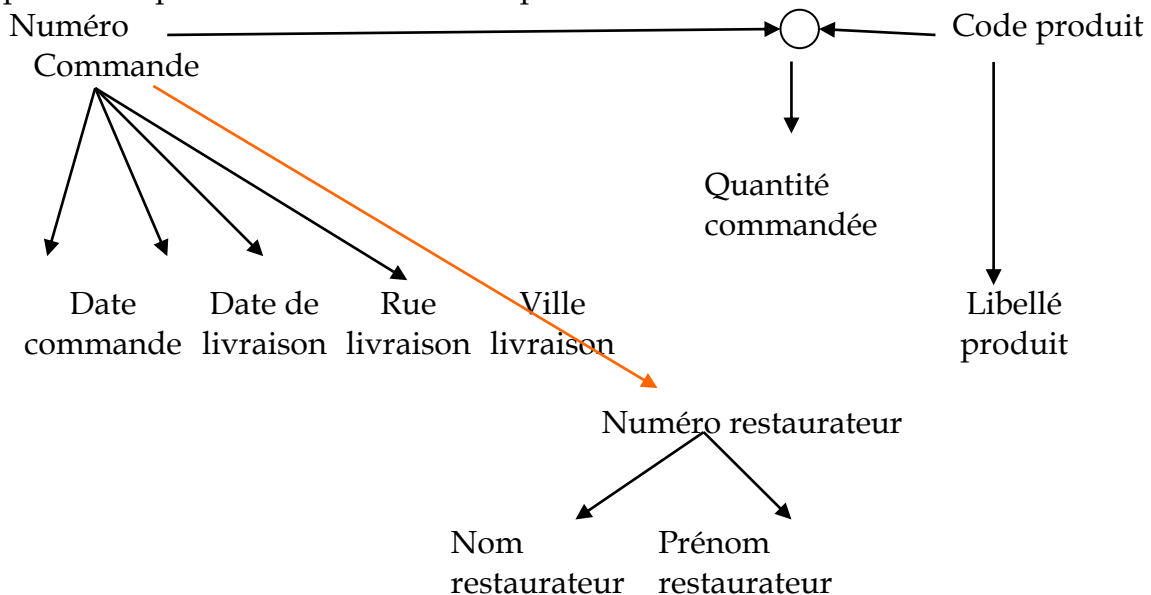
"Les restaurateurs achètent des produits alimentaires". En considérant que ces achats sont effectués dans le cadre de commandes, les règles de gestion suivantes ont été retenues :

Une commande est passée par un seul restaurateur et elle concerne des produits alimentaires. Un restaurateur peut passer plusieurs commandes et un produit alimentaire peut faire partie de plusieurs commandes.

En retenant la liste des propriétés ci-dessous:

- code produit alimentaire
- numéro restaurateur
- numéro commande
- libellé produit alimentaire
- date commande
- nom restaurateur
- prénom restaurateur
- quantité commandée
- date de livraison
- rue livraison
- ville livraison,

Le graphe des dépendances fonctionnelles prend l'allure suivante :



4-3. Le formalisme du modèle conceptuel individuel

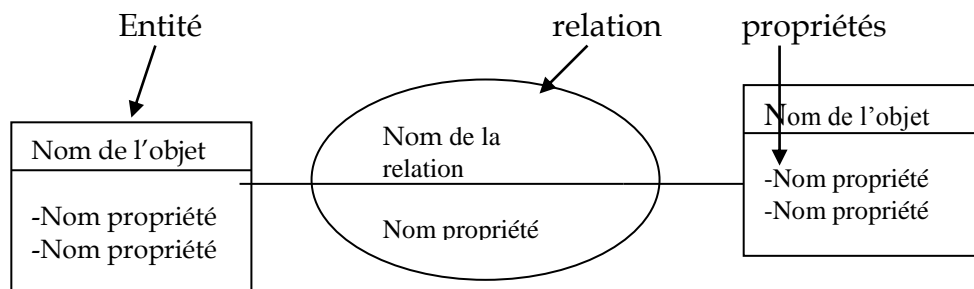
Les concepts de base sont **propriétés- Entité - Relation**

Les sous-ensembles obtenus à partir des DF entre les propriétés élémentaires définissent **les entités** ; et les origines de ces sous-ensembles sont les identifiants.

Donc l' **Entité** c'est L'**objet type** qui est une association de propriétés.

Graphisme

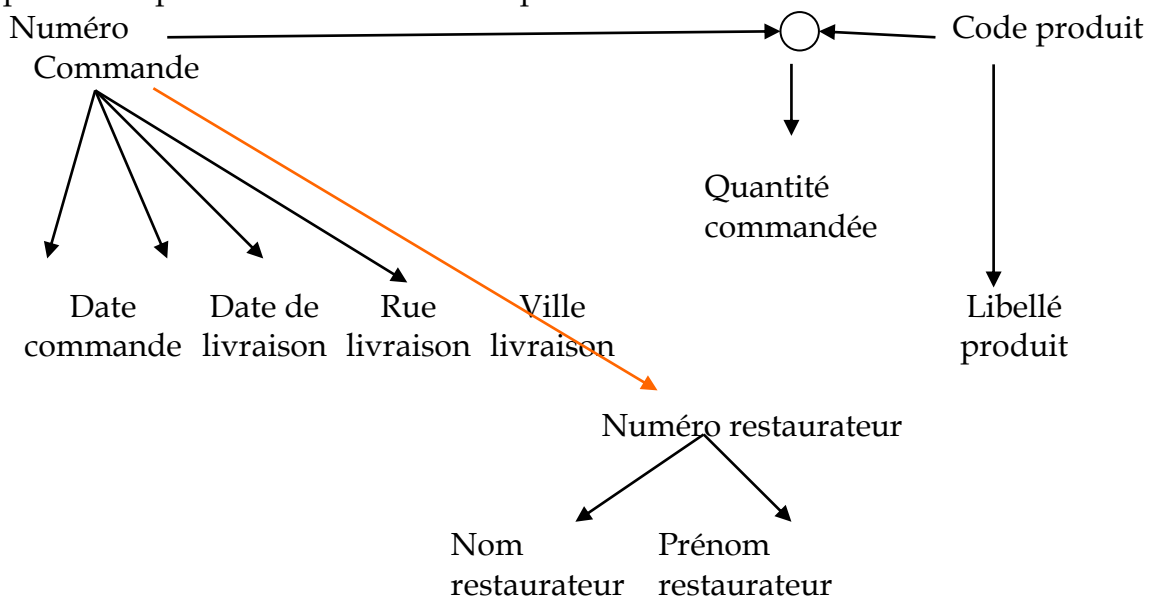
Une entité est représentée par un rectangle dont la partie supérieure reçoit l'intitulé de l'entité, tandis que les propriétés de l'entité sont logées dans la partie inférieure du rectangle. La représentation graphique d'une liaison entre entités à une forme circulaire (ovale, ellipse, cercle, rectangle avec des arrondies circulaires) dans la quelle on spécifie le verbe caractérisant au mieux la liaison dans la partie haute, la partie basse accueille les propriétés éventuelles portées par une liaison .le segment qui relie le symbole de l'entité et le symbole de la liaison est une patte.



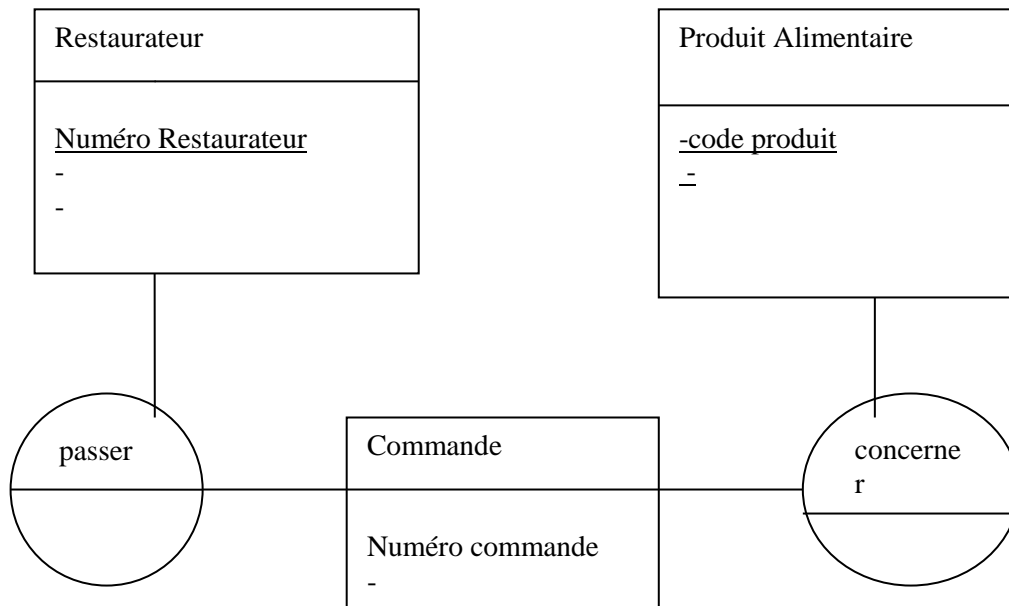
Représentation schématique

Exemple :

Le graphe des dépendances fonctionnelles prend l'allure suivante :

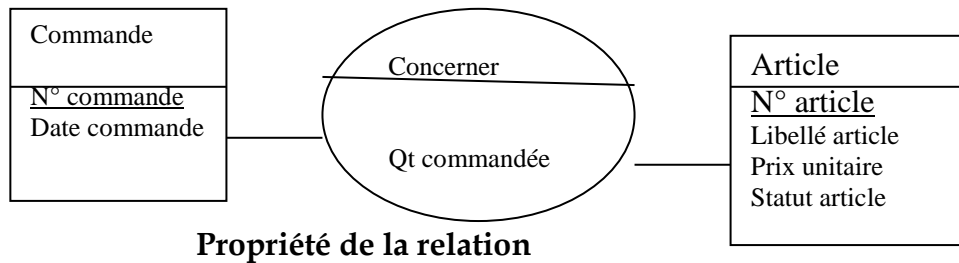


Le graphe MCD correspondant est :



4-4 - Les propriétés de la relation

Les propriétés de la liste devaient être rattachées aux identifiants des objets suivant la règle de la dépendance fonctionnelle élémentaire. Certaines de ces propriétés sont restées en attente, car en dépendance fonctionnelle de plusieurs identifiants d'objets ; elles sont tout simplement portées par des relations entre ces objets. C'est par exemple le cas de la propriété, « Quantité commandée » entre les objets « ARTICLE » et « COMMANDE ».

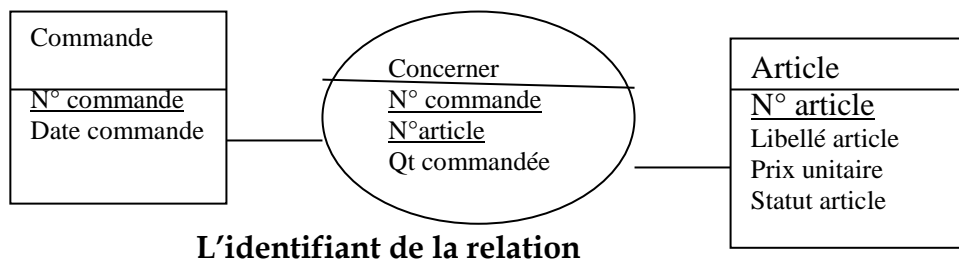


4-2-2- Identifiant de la relation

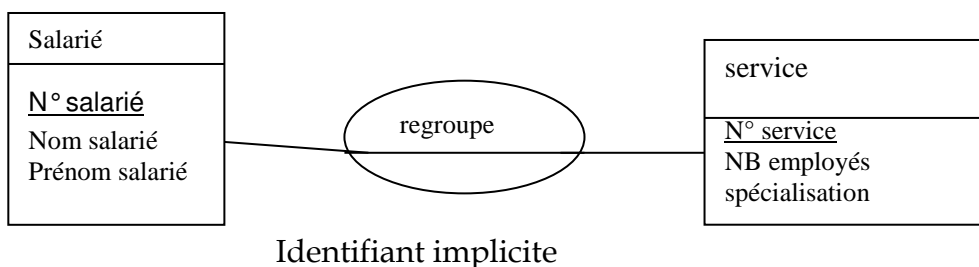
La relation, comme l'objet, possède un identifiant, qui est la combinaison, ou concaténation (enchaînement) des identifiants des objets de sa collection, mais n'est pas porté sur le modèle (il est implicite).

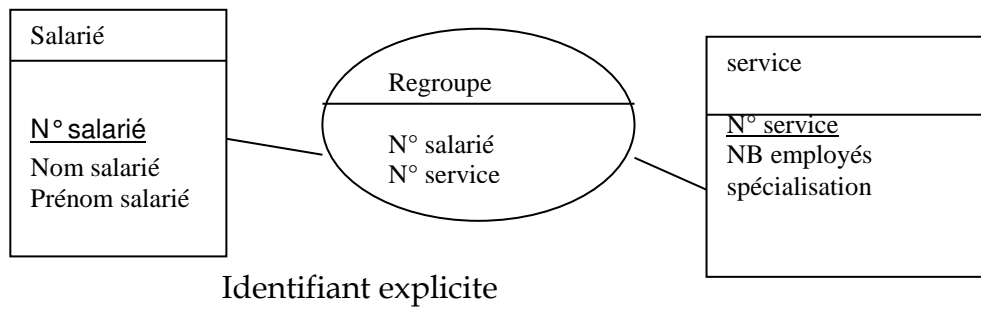
Exemple :

Un identifiant d'une relation « CONCERNER » entre les objets « COMMANDE » et « ARTICLE » est l'enchaînement des identifiants de ces deux objets, le modèle devant être lu en extension



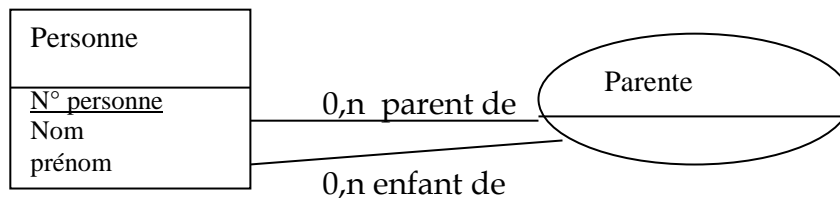
si la relation est une relation vide, elle sera tout de même identifiée, par les identifiants des objets associés dans la relation





4-3- La relation réflexive

La relation réflexive, est une association qui relie un objet à lui même; une occurrence de la relation associe une occurrence de l'objet à une autre occurrence de ce même objet



Relation réflexive

Pour enlever cet ambiguïté la relation se transforme en (**objet**)

Remarque : l'initiative reste toujours au concepteur de lever l'ambiguïté

4-5- Le diagramme d'occurrences

Construire un diagramme d'occurrences, c'est attribuer à chacun des objets et relations du modèle étudié un ensemble de valeur; plausible pour en vérifier la validité

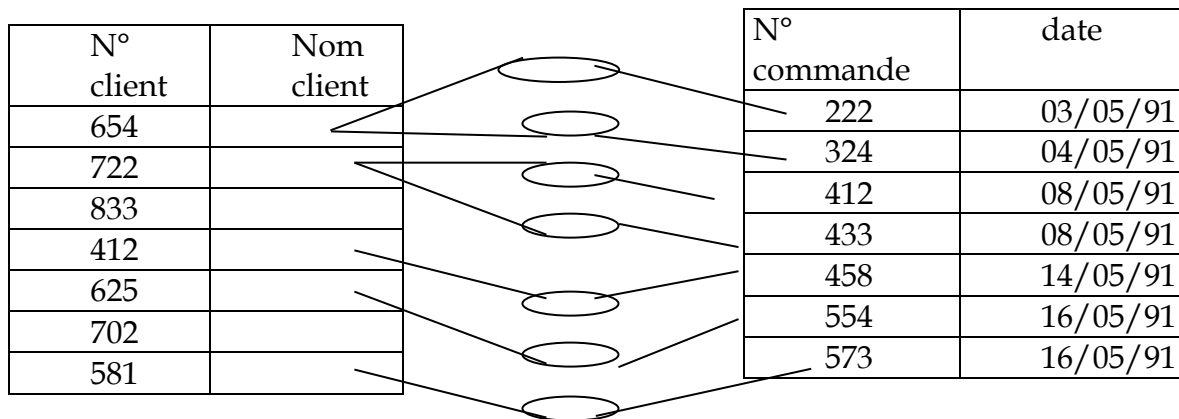
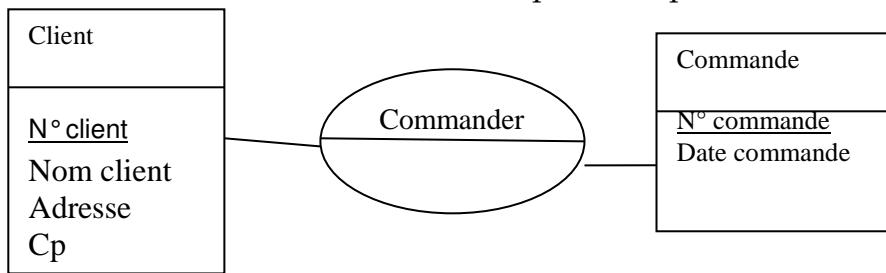
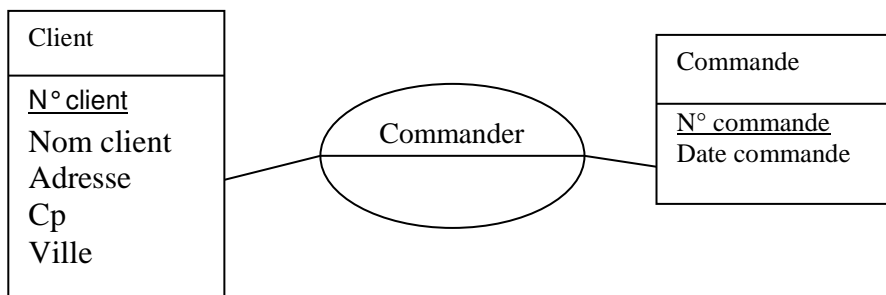


Diagramme d'occurrence

4-6- -Les cardinalités

Le diagramme d'occurrences permet de connaître avec exactitude le degré de participation de chaque objet à une relation.



L'étude des cardinalités suppose de se poser la question des bornes inférieure et supérieure à propos de chaque couple objet relation



Pour déterminer avec rigueur les cardinalités de ces objets dans la relation « COMMANDER », il faudrait connaître l'ensemble des occurrences possibles de chacun d'eux, ce qui n'est pas envisageable. Cependant, il n'est pas utile d'énoncer avec une telle précision la participation de toutes les occurrences, car seules les bornes supérieures et inférieures doivent être connues pour permettre le passage au niveau logique.

les cardinalités - couples objet relation.

Les cardinalités représentent pour chaque couple objet-relation les nombres minimum et maximum d'occurrences de la relation pouvant exister pour une occurrence de l'objet. Elles mesurent, lorsque l'on parcourt l'ensemble des occurrences de l'objet impliqué dans le couple, le minimum et le maximum de leur participation à la relation (borne inférieure et supérieure), et sont notées (0-1), (0-n), (1,1), (1,n). Ce sont les seules quantifications admises.

(0-1) : une occurrence de l'objet peut exister sans participer à la relation (0) et ne participe jamais plus d'une fois (1).

(0,n) : c'est la cardinalité la plus ouverte ; une occurrence de l'objet peut exister sans participer à la relation (0) et peut participer sans limitation (n).

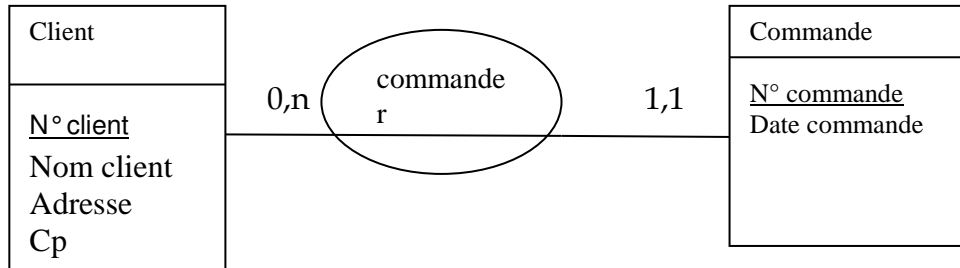
(1,1):une occurrence de l'objet participe au moins et au plus une fois à la relation.

(1,n) : une occurrence de l'objet participe au moins une fois à la relation (1)et peut participer sans limitation (n).

Exemple

L'objet **CLIENT** peut ne pas participer à la relation (0) et qu'il peut participer sans limitation (n), nous dirons alors que ses cardinalités sont (0,n) dans la relation **COMMANDER**

Un objet **COMMANDE** correspond à un client et un seul, les cardinalités dans cette relation sont (1,1)

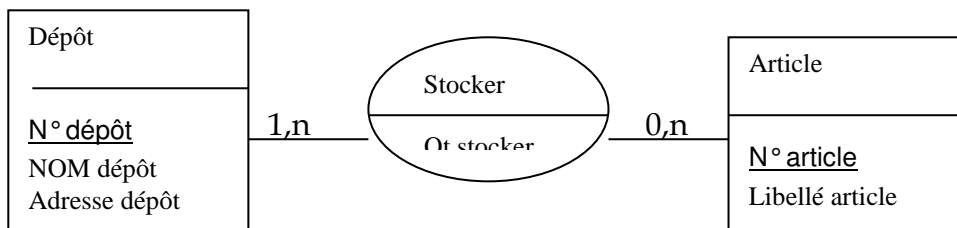
**Les cardinalités correspondant à chacun des couples objet-relation**

Les cardinalités, dont la connaissance est indispensable pour l'application des règles de passage au modèle logique, sont en outre la traduction des règles de gestion fondamentales

Exemple

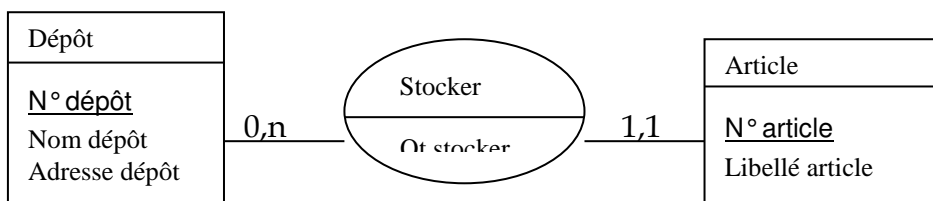
Première hypothèse

- Un article peut exister sans être stocké (catalogue, étude. etc.).
- Il peut être stocké en divers endroits.
- Un dépôt ne peut exister en tant que tel s'il ne stocke rien.

**Cardinalité et règle de gestion première hypothèse**

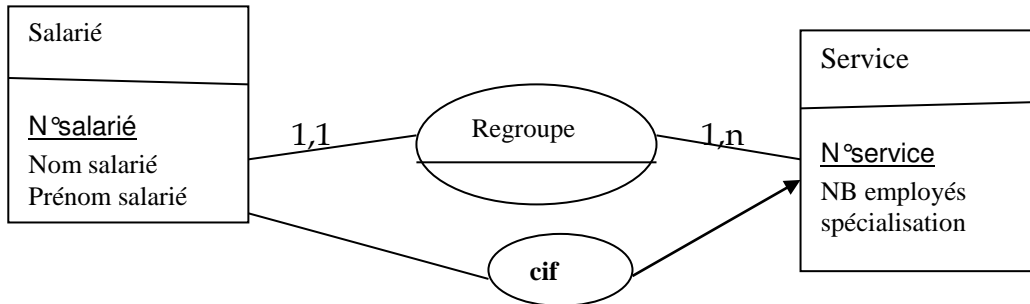
Deuxième hypothèse

- Un article ne peut exister sans être stocké (la quantité en stock pouvant prendre la valeur zéro en cas de rupture).
- Il ne peut être stocké qu'en un endroit.
- Un dépôt peut ne rien stocker.

**Cardinalité et règle de gestion deuxième hypothèse**

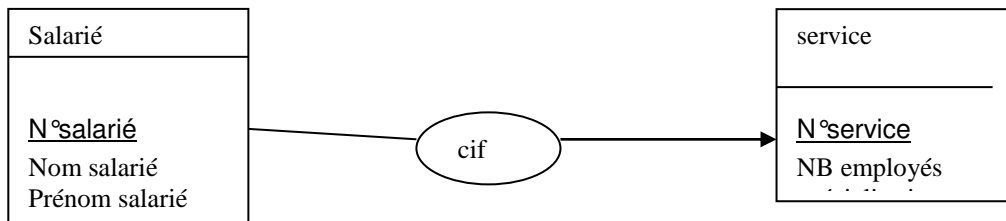
4-7-.Les contraintes d'intégrité fonctionnelle

La contrainte d'intégrité fonctionnelle (CIF) définie sur une relation représente le fait que l'un des objets de sa collection est identifié sans aucun doute par la connaissance d'un ou plusieurs autres. Une relation binaire ayant des cardinalités (0, 1) ou (1, 1) est une contrainte d'intégrité fonctionnelle. Elle représente une contrainte de gestion, et est représentée par une flèche pointant vers l'objet déterminé



Contrainte d'intégrité fonctionnelle

Si la relation binaire est vide et qu'il n'existe pas d'autre relation entre les deux objets, on pourra la remplacer par le symbole de la contrainte d'intégrité fonctionnelle

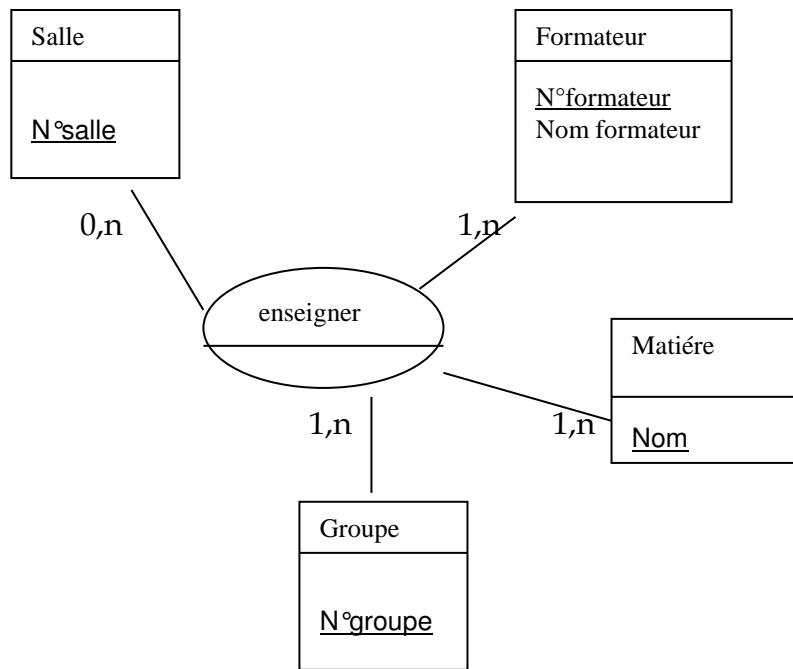


Contrainte d'intégrité fonctionnelle sur une relation binaire vide

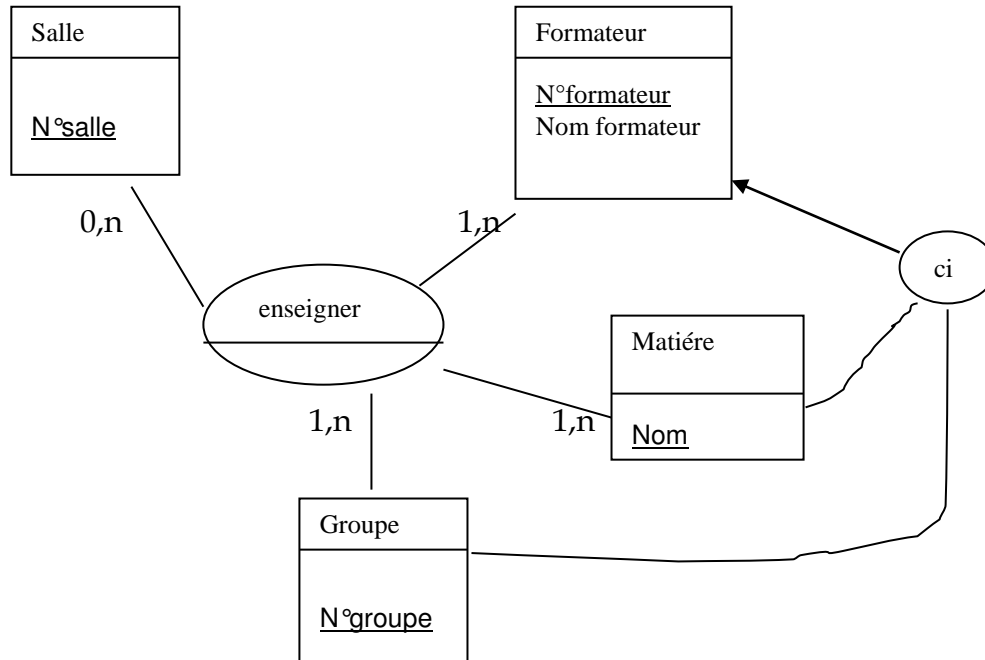
4-8-. Décomposition des relations

Les contraintes d'intégrité fonctionnelle permettent de simplifier les relations de dimension supérieure à 2. la CIF est surtout utile en présence de relations de plus grande dimension, peu opérationnelles, et qu'il est souhaitable de simplifier.

Soit par exemple, la relation « ENSEIGNER de dimension 4 traduisant la règle suivante: dans l'organisation étudiée, la gestion des formations demande de connaître l'occupation des salles par journée, sachant en outre qu'une matière est toujours enseignée par le même formateur, à un groupe de stagiaires donné

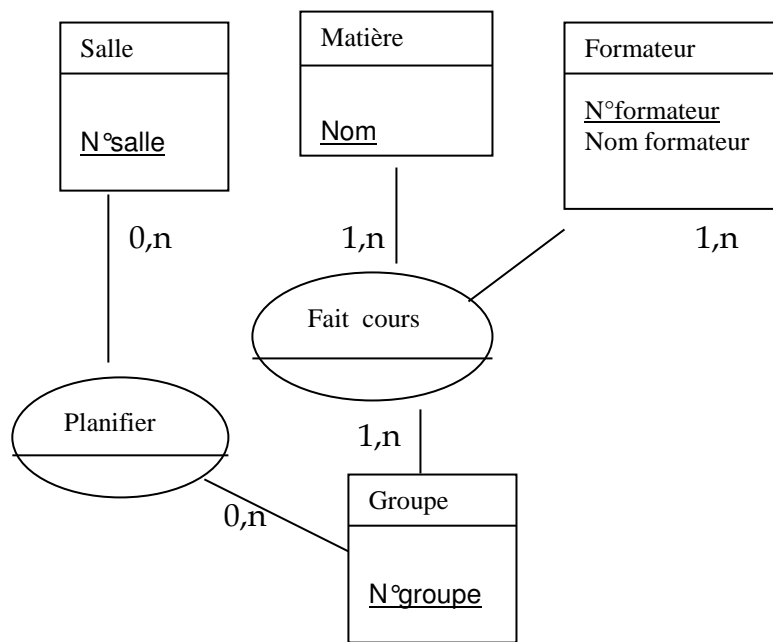


Ce choix ayant été fixé, on observe que la connaissance du groupe et de la matière détermine le formateur (cf dépendance fonctionnelle). Il y a donc présence d'une CIF entre « GROUPE », « MATIÈRE » et « FORMATEUR »



Cif sur la relation enseigner

Cette CIF nous permet de supprimer la relation « ENSEIGNER » et de la remplacer par deux relations ternaires beaucoup plus opérationnelles



Décomposition de la relation enseigner

4-9- La vérification du modèle

L'application systématique de chacune des règles de vérification sur les éléments du modèle conceptuel de données permet de s'assurer qu'il est conforme à ce que l'on attend, et donc apte à générer le modèle logique correspondant à la réalité à décrire.

La règle

Toutes les Propriétés doivent être élémentaires, C'est-à-dire non décomposables.

2eme règle

Chaque objet doit posséder un identifiant et un seul.

Certains objets réels ne sont pas identifiés au préalable. Il faut donc créer la donnée permettant de se mettre en conformité avec la règle 2. Rappelons qu'il ne peut y avoir deux occurrences identiques d'un même identifiant.

Il peut arriver que deux propriétés d'un même objet puissent remplir ce rôle d'identification. Dans ce cas, il faut étudier la participation de chacune d'elles à la vie du système pour choisir celle qu'il convient de retenir. La dépendance fonctionnelle entre ces deux propriétés est alors réciproque.

Exemple

Les propriétés N° national d'identification et « N° matricule » sont toutes les deux ont un rôle d'identifier l'objet « SALARIÉ » du système d'information, l'identifiant qu'il convient de retenir c'est N° matricule.

3eme règles

Les propriétés d'un objet autres que l'identifiant doivent être en dépendance fonctionnelle monovaluée (seule valeur) de l'identifiant de l'objet.

Pour une occurrence de l'identifiant de l'objet, chacune des propriétés ne peut prendre qu'une valeur. On ne peut pas donc avoir de valeur répétitive pour une même propriété.

Exemple :

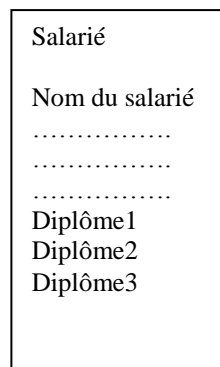
Soit l'objet « SALARIÉ » défini par les propriétés « N° Matricule », « Nom salarié », « Prénom salarié », « Date de naissance salarié », « Diplômes »

N° matricule $\xrightarrow{\text{détermine}}$ Nom salarié

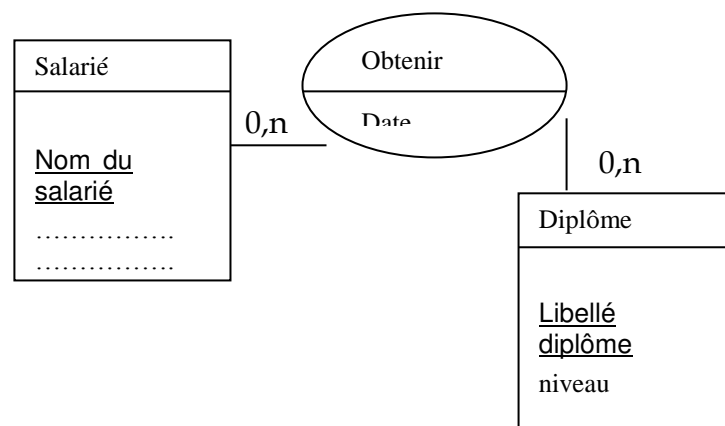
N° matricule $\xrightarrow{\text{multi détermine}}$ diplôme

Cet objet ne peut être défini ainsi que dans le cas bien improbable où chaque salarié de l'entreprise n'aurait au plus qu'un diplôme.

Si le salarié a plusieurs diplômes, L'une des solutions possible est de définir plusieurs propriétés destinées à prendre les différentes valeurs de diplômes détenus par un même salarié « le 1^{er} diplôme », « 2eme diplôme », etc.



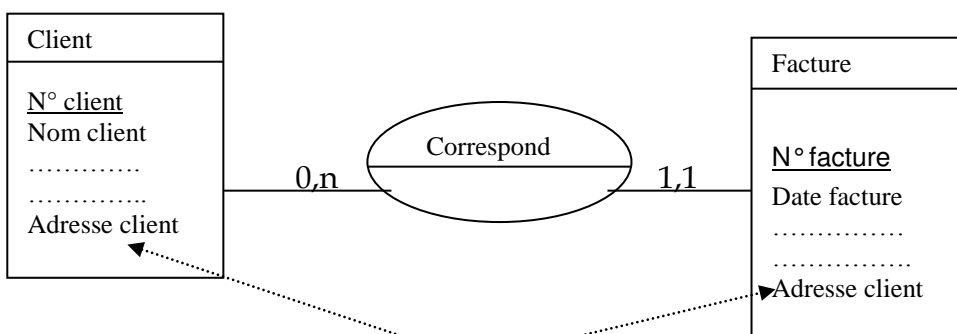
Une autre solution consiste à créer un objet « **diplôme** » et une **relation** l'associant à l'objet « **SALARIÉ** » ce qui est bien préférable. Si une propriété est en dépendance fonctionnelle de l'identifiant de plusieurs objets, alors elle est portée par une relation entre ces objets ; la troisième règle s'applique également à ces propriétés.



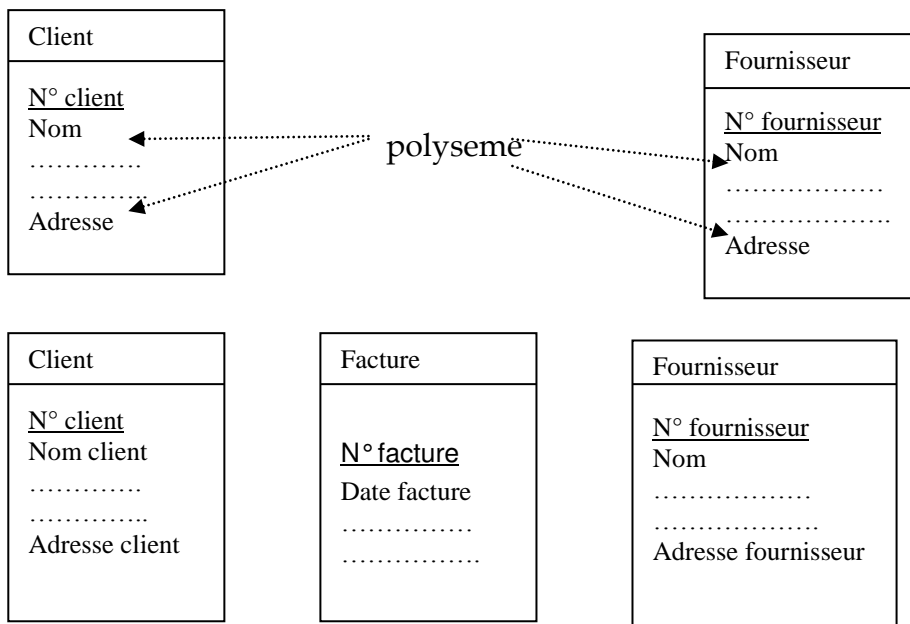
4eme règle

Une propriété ne peut qualifier qu'un seul objet ou qu'une seule relation.

La propriété « Nom du client », ne peut être présente à la fois dans l'objet « CLIENT » et dans l'objet «FACTURE », sera une redondance ;de même une propriété ne peut s'appeler « Adresse » dans l'objet « CLIENT » et «Adresse » dans l'objet « FOURNISSEUR » (polysème). Si tel est le cas, alors il faut appeler l'une «Adresse client», et l'autre « Adresse fournisseur



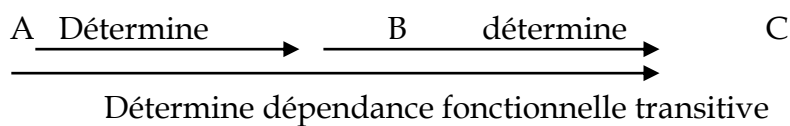
Redondance



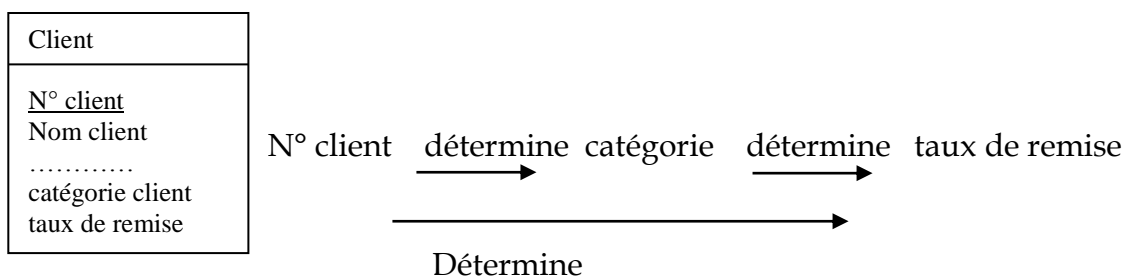
5eme règle

La dépendance fonctionnelle transitive doit être écartée.

Si une propriété est en dépendance fonctionnelle de l'identifiant, et d'une autre propriété de l'objet, elle même en dépendance fonctionnelle simple de cet identifiant, il y a un objet imbriqué dans celui que l'on étudie.



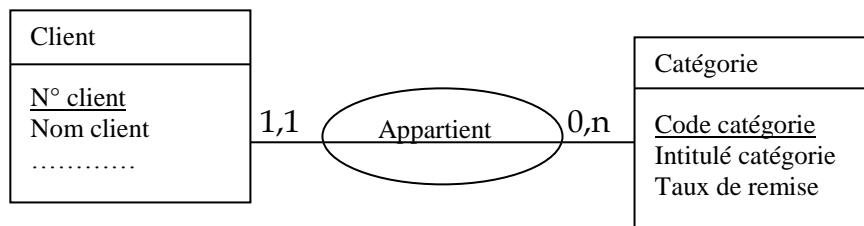
Chaque objet doit décrire un concept sémantique unique, et en conséquence, il faut éclater en deux objets celui qui contient une dépendance fonctionnelle transitive.



Dans notre exemple, considérons la règle suivante: le prix de vente au client est calculé sur le prix de vente public, diminué d'une remise dont le montant est fonction de la

catégorie à laquelle appartient le client (grossiste, semi-grossiste, détaillant ...). Ajoutons qu'un client est forcément rattaché à une catégorie, et au plus à une. Il n'est pas faux de considérer à priori les propriétés « Catégorie client » et « Taux de remise » comme définissant l'objet « CLIENT ». En effet, ces deux propriétés sont bien en dépendance fonctionnelle de l'identifiant « N° client (par hypothèse, à un client donné correspond une catégorie et un taux de remise).

Toutefois, la propriété « Taux de remise » dépend également, fonctionnellement, de la propriété « Catégorie client » ; elle est donc soumise à deux dépendances fonctionnelles simples. Il y a ici description de deux concepts différents : le client d'une part, et la catégorie d'autre part, donc deux objets distincts: « CLIENT » et « CATÉGORIE ».



Résolution de la dépendance fonctionnelle transitive

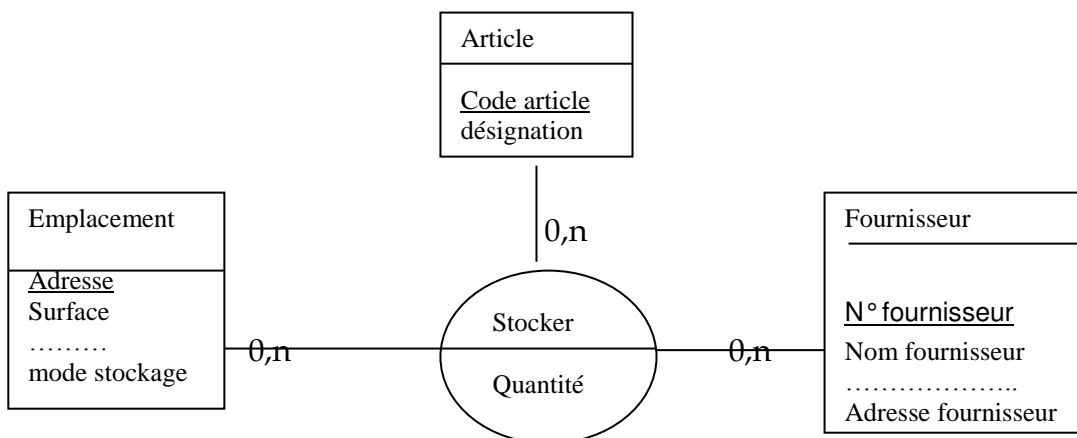
6eme règle

Les propriétés d'une relation doivent dépendre de la totalité de l'identifiant de cette relation.

Si ce n'est pas le cas, il faut éclater la relation en autant de relations que nécessaire.

Exemple :

Considérons la relation, « STOCKER » entre les objets « FOURNISSEUR », « ARTICLE » et « EMBLACEMENT »

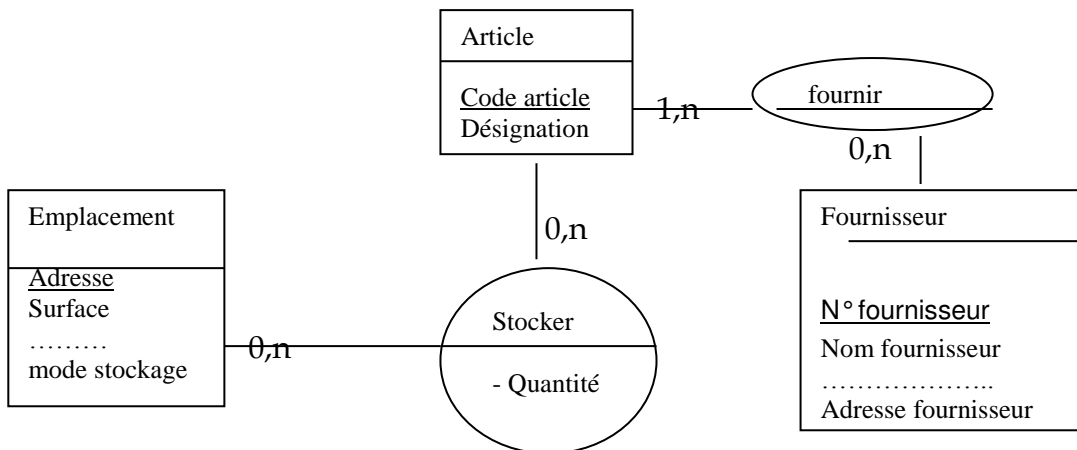


Si Certains articles proposés par les fournisseurs n'étant pas forcément présents en stock.

La propriété **quantité** ne dépend pas de la totalité de l'identifiant (certains quantité n'est pas en dépendance fonctionnelle de l'emplacement), donc la relation stocker qui est ternaire doit être éclater en 2 relations binaires

La relation stocker se décompose en fait en deux :

- D'une part, un article est fourni par un fournisseur;
- D'autre part, un article est stocké sur un emplacement.



4-10- Etablissement du MCD

- Établir une liste des données à partir des documents de l'entreprise, et plus généralement de tous les supports de l'information.
- Classer ces données par ordre alphabétique afin de balayer tout a priori sur les regroupements de propriétés.
- Procéder à l'épuration des polysèmes, des synonymes et des redondances.
- Repérer les identifiants existants pour dégager les objets naturels.
- Rattacher à ces objets les propriétés en dépendance fonctionnelle de leur identifiant.
- Placer les relations et leur rattacher si besoin est les propriétés en dépendance fonctionnelle de plusieurs identifiants.
- Considérer les propriétés restantes afin de les regrouper en objets pour lesquels on créera les identifiants non formalisés.

- Étudier les cardinalités de chaque couple objet-relation
- Simplifier le modèle à l'aide des contraintes d'intégrité fonctionnelle.
- Procéder à la vérification à l'aide des règles.

5- LE MODELE CONCEPTUELE DES TRAITEMENTS (MCT)

Le **M C T** presente la vue fonctionnelle de l'entreprise

Le niveau conceptuel exprimait le **quoi** (quoi faire ?)

5-1- définitions et formalisme

Lors de l'étude de l'existant, des opérations ont été recensées et les événements qui déclenchent chacune d'elles ont été spécifiés ainsi que les règles de gestion correspondantes et les résultats des opérations.

Les notions qui figurent dans le **MCT** : Evénement, règle de synchronisation, opération, règle d'émission, résultat

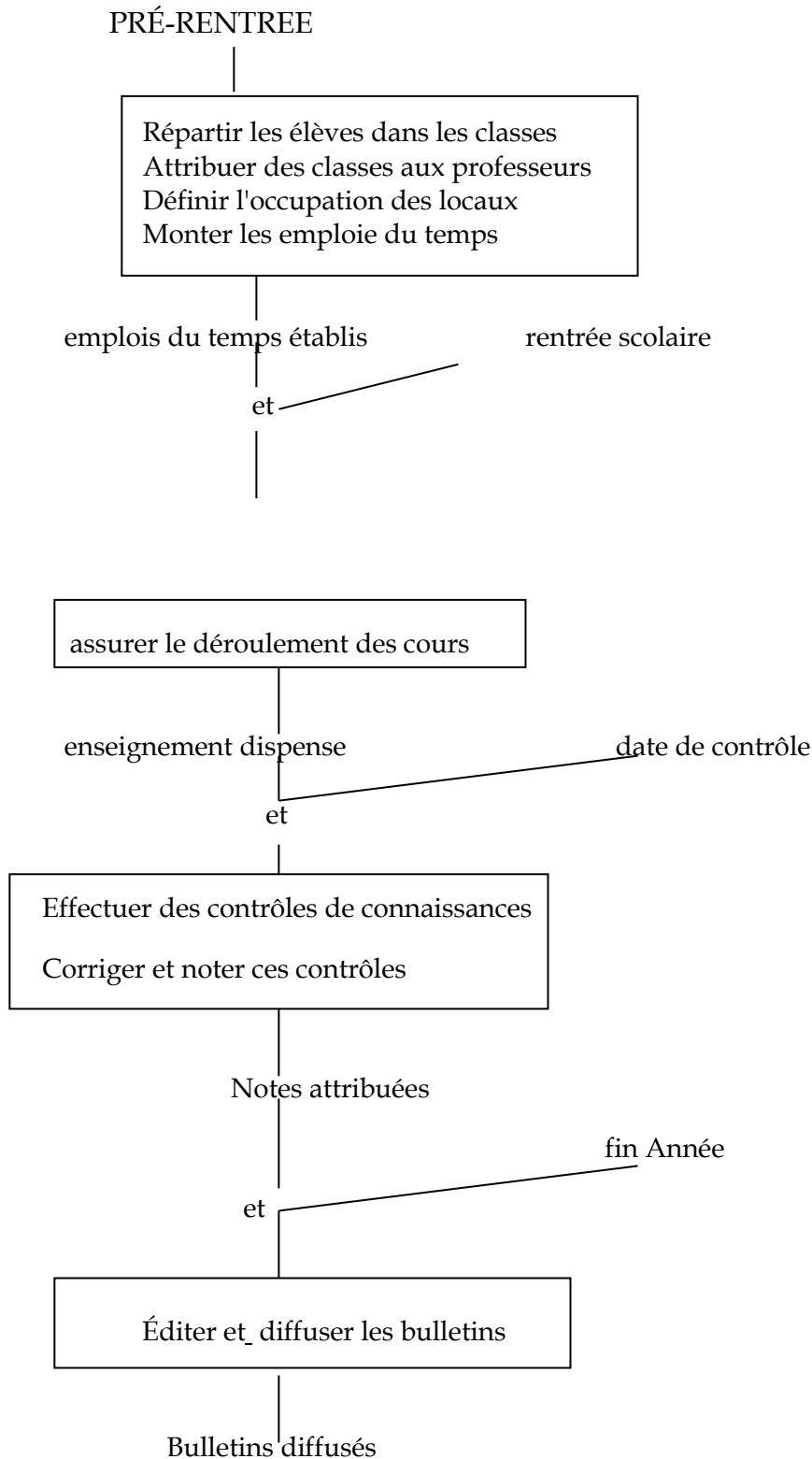
Exemple : approche intuitive :

L'établissement scolaire a pour objectif principal la formation des élèves. Pour atteindre celui-ci il est nécessaire de dérouler un certain nombre d'actions.

- répartir les élèves par classe,
- attribuer les classes aux professeurs,
- définir l'occupation des locaux,
- monter les emplois du temps,
- assurer le déroulement des cours,
- effectuer les contrôles de connaissances,
- corriger et noter les contrôles,
- éditer et diffuser les bulletins scolaires,
- etc.

Ces actions ne décrivent que le « **quoi** » de l'entreprise et l'on ne précise ni le « **qui** » (qui attribue les classes aux professeurs ?), ni le « **où** » (où sont édités les bulletins scolaire ni le « **comment** » (comment est définie l'occupation des locaux?

Pour le représenter on regroupe dans un même rectangle les actions du même domaine d'activité pour former des opérations puis on encadrera chaque opération par l'événement qui la déclenche et par le résultat qu'elle produit, lui même déclenchant l'opération suivante et ainsi de suite.



Evènement. : Fait réel dont la venue a pour effet de déclencher l'exécution d'une ou plusieurs actions.

Exemple : pre-rentree scolaire

Les événements (qu'ils soient déclencheurs ou résultats) sont identifiés par un nom et symbolisés par un ovale, relié à l'opération qu'ils déclenchent ou dont ils résultent.

Exemple

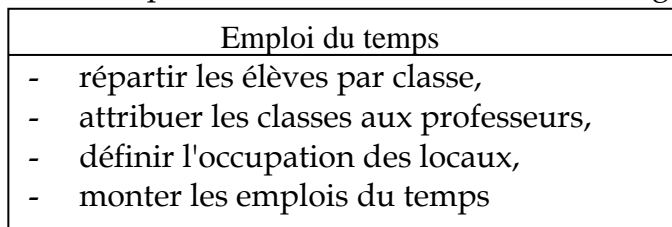
Pre-rentree scolaire

Opération : Ensemble d'actions dont l'enchaînement ininterrompible n'est conditionné par l'attente d'aucun événement autre que le déclencheur initial.

Exemple : L'opération « Emploi du temps » regroupe les actions ininterrompibles suivantes:

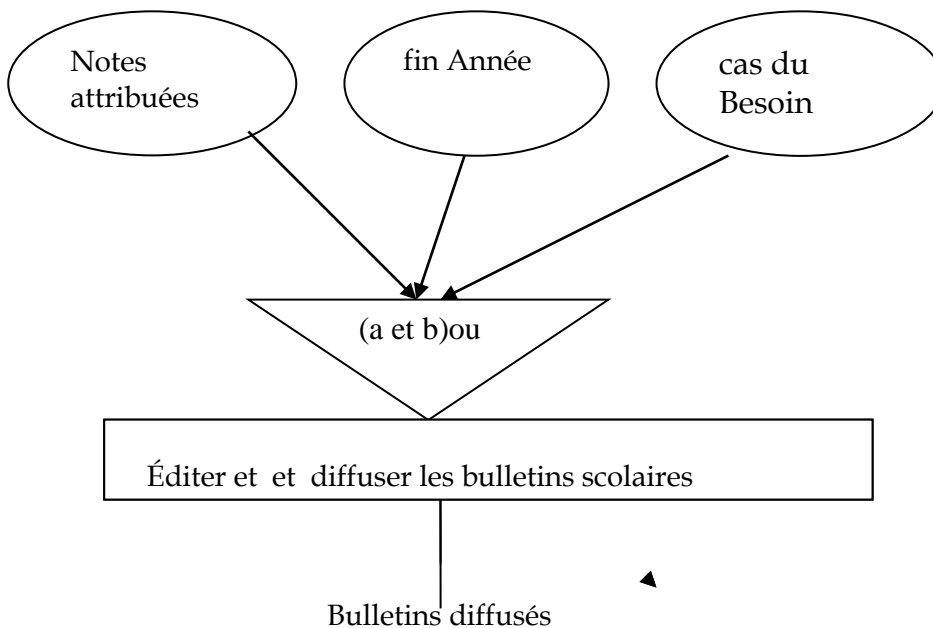
- répartir les élèves par classe,
- attribuer les classes aux professeurs,
- définir l'occupation des locaux,
- monter les emplois du temps,

On ne fera figurer sur le schéma que le nom de l'opération de façon à ne pas le surcharger. Ce nom identifiera l'opération et sera encadré d'un rectangle.



Synchronisation. Condition booléenne, traduisant les règles de gestion, que doivent vérifier les événements pour déclencher des opérations

Exemple : [(notes attribuées) **et** (fin Année)] **ou** (cas du Besoin)



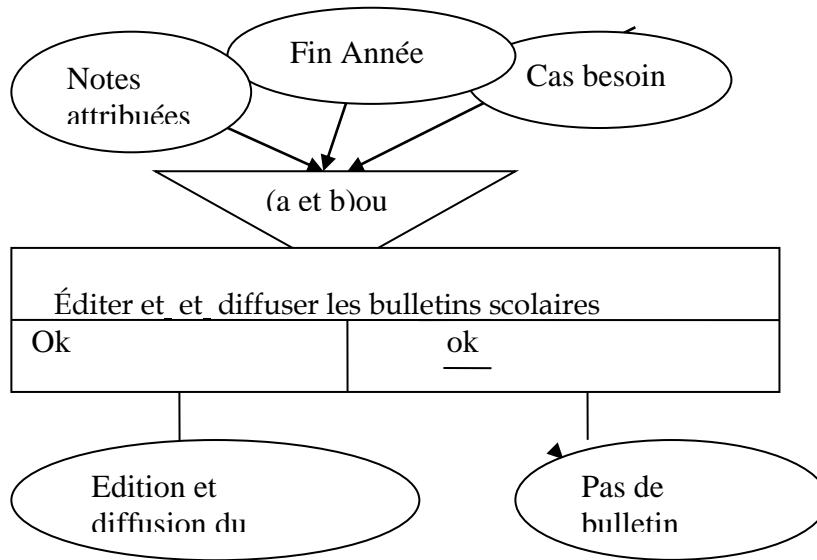
La proposition logique de déclenchement sur les événements est décrite à l'intérieur d'une synchronisation liée à l'opération.

On utilisera les symboles \wedge (pour « et ») et \vee (pour « ou ») et, si le nombre d'événements est important, on utilise des lettres symboliques pour les représenter.

Règle d'emssion. Condition, traduisant les règles de gestion, à laquelle est soumise l'émission des résultats d'une opération.

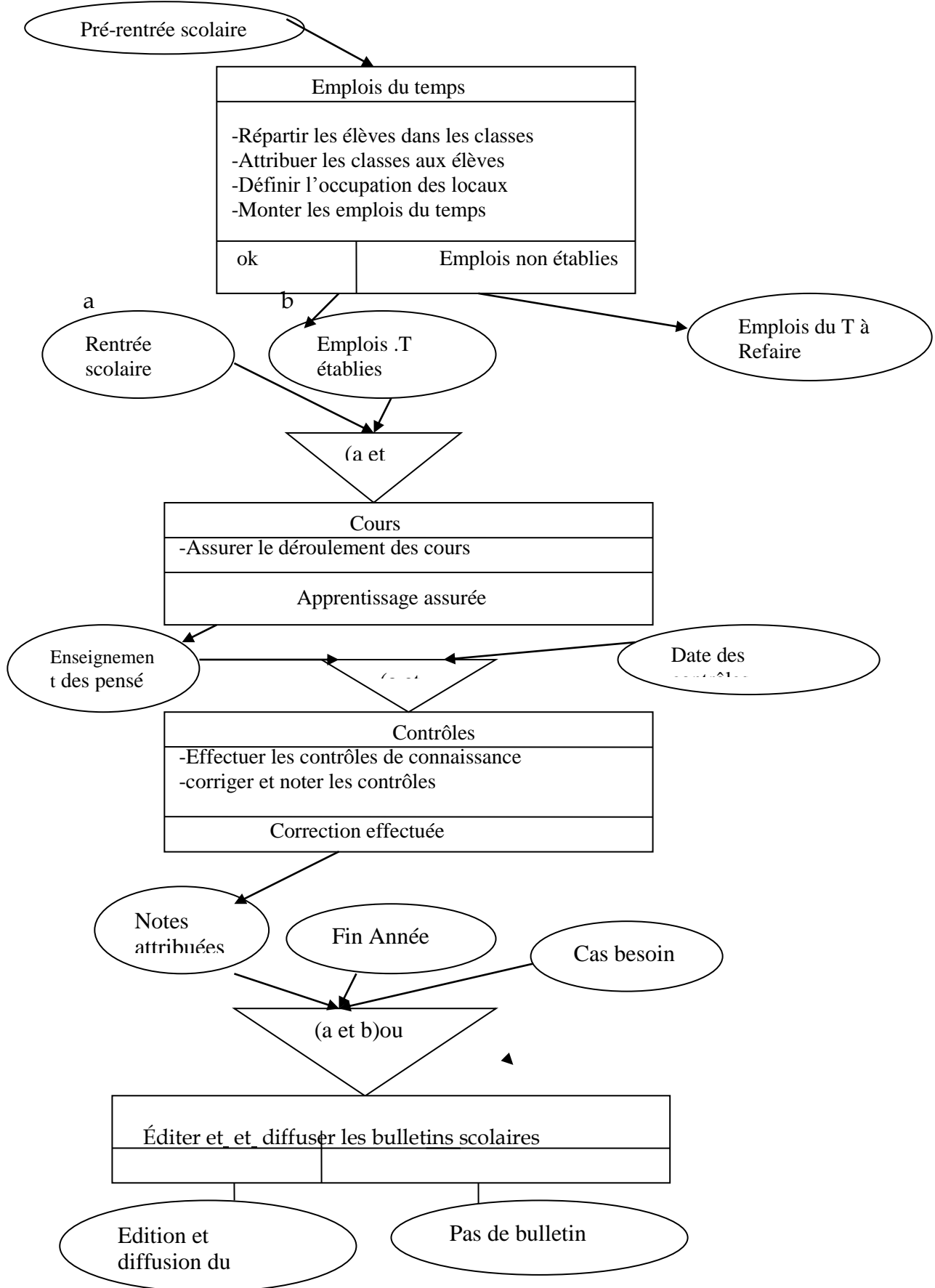
Exemple : Si c'est conforme alors...

La condition d'émission du résultat pourra être précisée dans le rectangle de l'opération. On trouvera essentiellement des messages tels que **OK, OK, SI, SINON, ...**



Résultat .: Produit de l'exécution d'une opération. Le résultat, fait réel de même nature que l'événement, pourra être le déclencheur d'une autre opération.

MCT de l'établissement scolaire

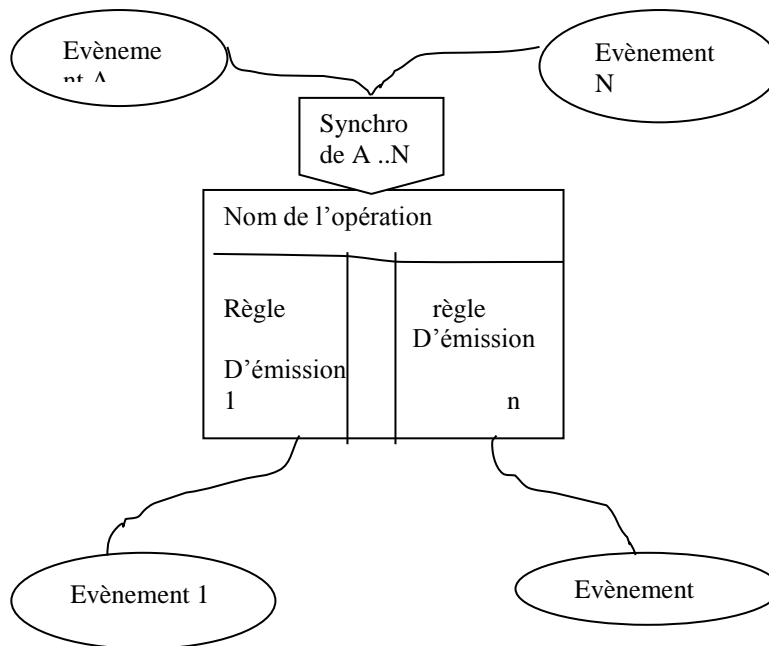


5-2 formalisme

On regroupe dans un même rectangle des actions du même domaine d'activité. un tel groupe d'actions est appelé opération

Il traduit les règles de gestion par l'utilisation des opérateurs logiques entre les événements déclencheurs (« **et** », « **ou** » et toute **combinaison** de ceux-ci) pour l'enchaînement des opérations, de ce qui les déclenche et de ce qu'elles produisent.

Le schéma qui les assemble sera le modèle conceptuel des traitements



5-3-Constitution du modèle M C T

5-3-1- création des opérations

Passage des actions aux opérations

La synthèse des traitements recueillis lors des interviews dégage un ensemble d'actions fondamentales pour l'entreprise, groupés en domaine d'activité.

On regroupe les actions dont l'enchaînement peut se dérouler sans attente d'événement extérieur. Un tel groupe est appelé **opération**

Remarque : les actions groupées au sein d'une opération s'enchaînent conformément aux règles de gestion

Notion de processus

Un processus est un enchaînement d'opération dont les actions sont incluses dans un même domaine d'activité

5-3-2-Evénements

Un tel événement traduit donc un pont entre deux opérations ou processus

5-3-2-1-Classification des événements :

- **Evénement externe à l'entreprise** : on appelle externe à l'entreprise un événement qui, sur l'ensemble des processus dégagés, est uniquement déclencheur d'une opération, soit uniquement résultat d'une opération

Exp : Rentrée scolaire

- **Evénement interne à l'entreprise et externe au processus** : on appelle interne à l'entreprise et externe au processus un événement qui est à la fois résultat d'une opération d'un processus et déclencheur d'une opération d'un autre processus

Exp ; Notes attribuées

- **Evénement interne au processus** : on appelle événement interne au processus un événement qui est à la fois résultat d'une opération et déclencheur d'une autre dans le même processus

Exp : Edition des notes déclenche le processus diffusion des bulletins

Remarque : le rôle des événements : les types 1 et 2 « cassent » les opérations alors que le type 3 enchaîne les produits « morceaux »

5-3-3-Règles de synchronisation. Règles d'émission

Elles sont la traduction des règles de gestion que l'on choisit de faire figurer sur le modèle et précisent pour :

5-3-4-Règles de synchronisation : la condition de déclenchement des opérations

5-3-5- Règles d'émission : la condition de production des événements résultats

6- MODELE ORGANISATIONNEL DES TRAITEMENTS (MOT)

Le modèle conceptuel des traitements traduit l'enchaînement des traitements. Le retour à la réalité conduit à mettre en place une organisation ; c'est l'objet du niveau suivant que traduira le modèle organisationnel des traitements.

Le niveau conceptuel exprimait le **quoi** ?

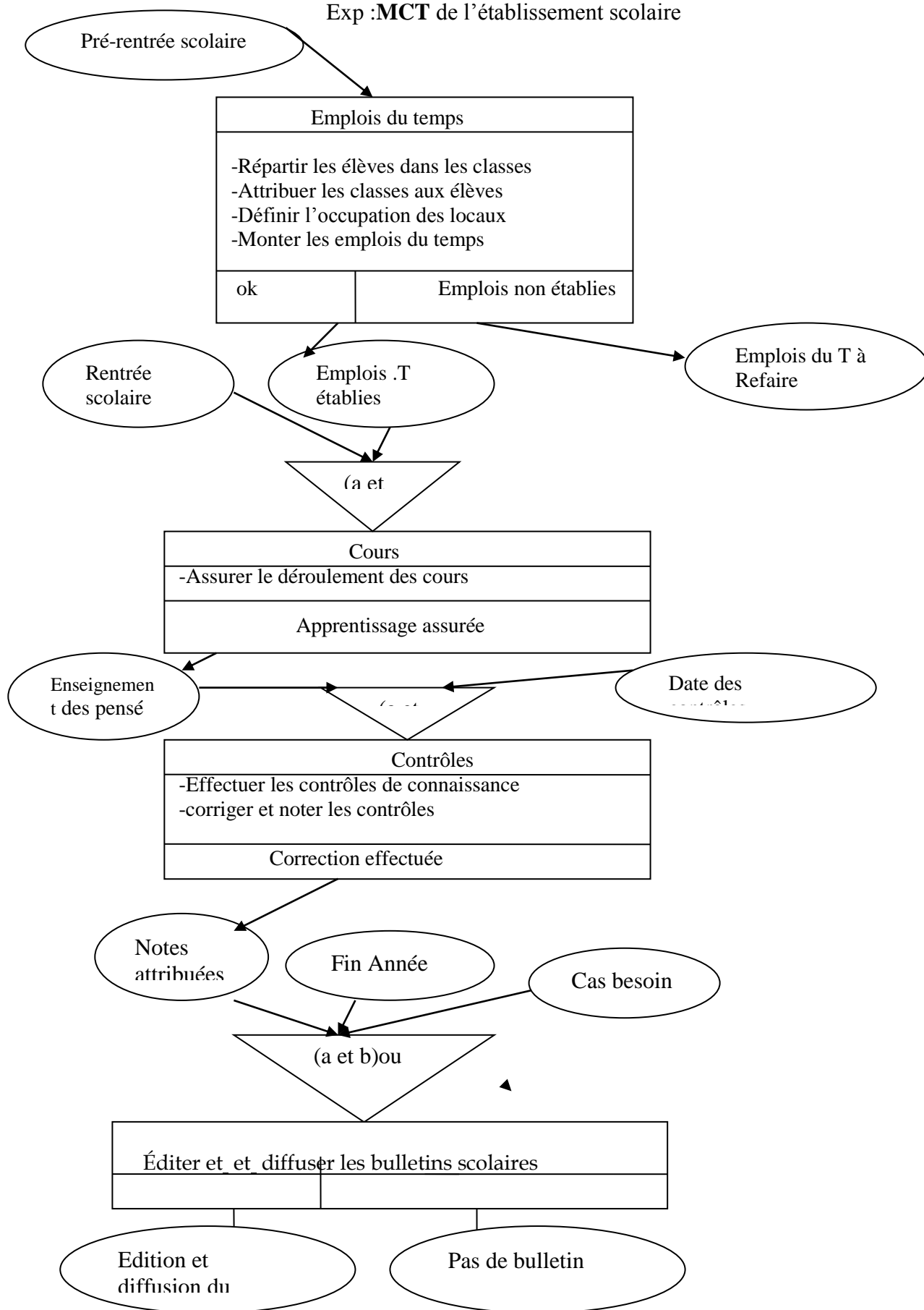
Le niveau organisationnel s'accroche à décrire le **S.I** en répondant aux questions **Où ? qui ? Quand ?**

A la première interrogation répondra le **poste de travail** concerné à la seconde le choix entre un traitement **automatique ou manuel** et la dernière question précise le **déroulement dans le temps** des différentes actions

L'objectif est de fournir une représentation schématique de l'organisation de l'entreprise .pour cela le formalisme utilisé au niveau conceptuel sera repris et On Organise les opérations conceptuelles

Pour chaque **action** du niveau conceptuel, on intègre les notions de lieu (**poste de travail**), de nature des traitements (**manuels ou automatiques**), de temps (**temps de déroulement**) de l'action,

Exp :MCT de l'établissement scolaire



Les actions du niveau conceptuel qui sont définies par le poste de travail, la nature du traitement et sa durée, Seront affirmées des **règles d'organisation**, concernent alors le niveau organisationnel de traitement.

Les objectifs et contraintes dégagés lors des interviews qui sont affirmées comme règles d'organisation

Les règles de gestion appliquées dans cet établissement

- A chaque classe est attribuée une et une seule salle de cours
- chaque matière n'est enseignée que par un seul professeur
- pour chaque classe et chaque matière est défini un nombre fixe d'heures de cours
- l'établissement gère les emplois du temps des professeurs et des élèves ainsi que le contrôle des connaissances

Avant la rentrée scolaire l'établissement établit les emplois du temps des classes et des professeurs alors pour réaliser ceux-ci :

- le bureau de gestion répartir les élèves par classe par un traitement conversationnel
- le bureau de gestion attribuer les classes aux professeurs par un traitement conversationnel
- le bureau de gestion définir l'occupation des locaux par un traitement conversationnel
- le bureau de gestion monter les emplois du temps par un traitement conversationnel

Lorsque les élèves ont rejoint leurs classes l'établissement éveille sur leur formation

- à chaque vacation des professeurs font des cours avec des classes suivant leurs emplois du temps, choisissant des méthodes pédagogiques appropriés

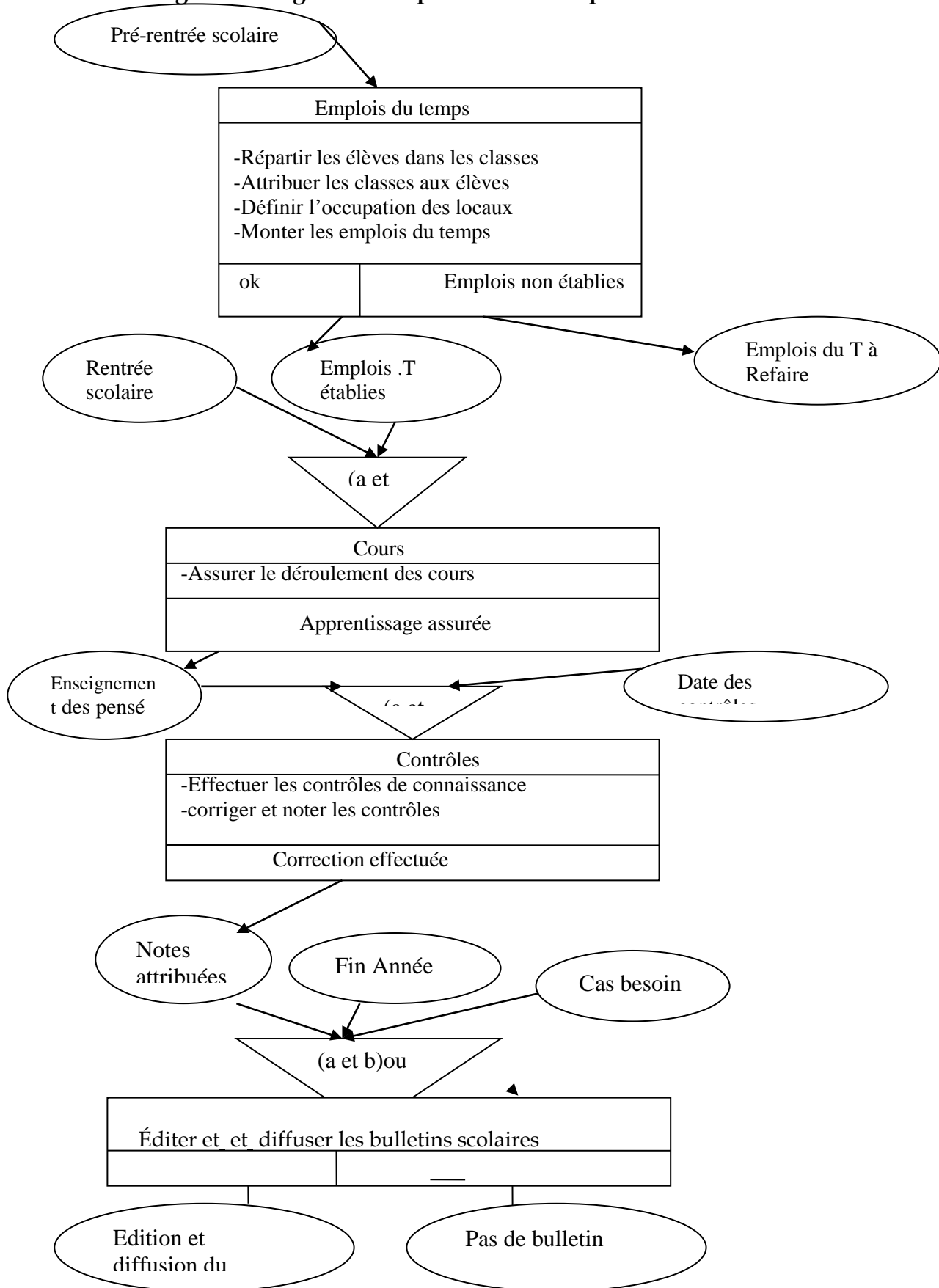
Après une période d'apprentissage des élèves, l'établissement fixe des dates des contrôles

- chaque professeur effectue les contrôles de connaissances avec sa classe écrite ou orale
- chaque professeur corrige les contrôles, et attribue des notes aux élèves suivant le jugement sur les réponses
- etc.

L'établissement diffuse les résultats à la fin de chaque période:

- l'atelier de saisie édite les notes des élèves par un traitement conversationnel
- le centre de calcul détermine la moyenne dans chaque discipline de chaque élève et sa moyenne générale par un traitement par lot
- le bureau de gestion diffuse les bulletins scolaires aux parents par voie postale

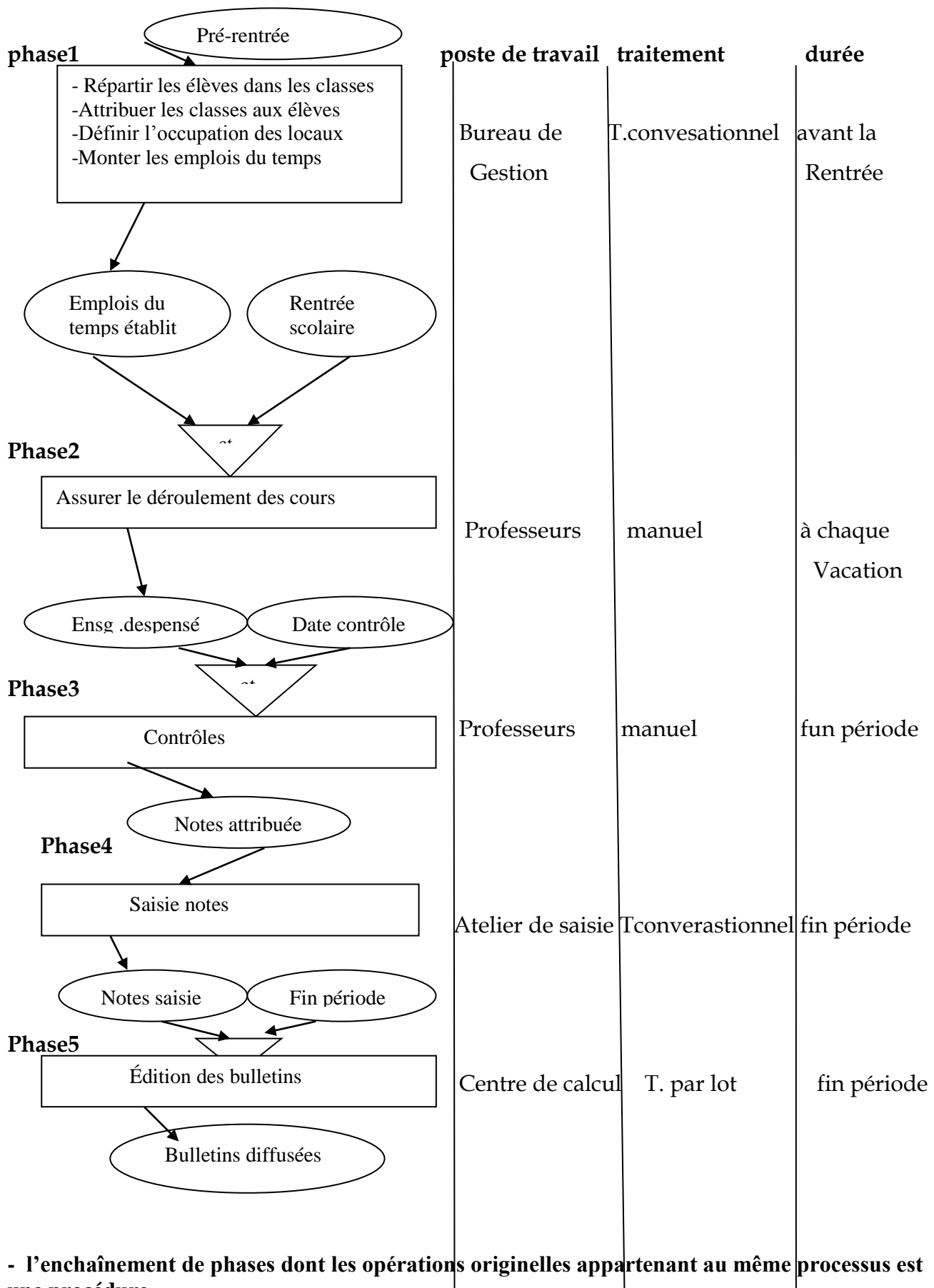
Suivant ces règles On Organise les opérations conceptuelles :



Pour décrire le MCT on réalise les schémas suivants

| | Poste de travail | traitement | durée |
|---|-------------------|-------------------------------|----------------------|
| <pre> graph TD A[Répartir les élèves aux classes] --> B((Classes formées)) B --> C[Attribution des classes aux professeurs] C --> D((Matières-classe)) D --> E[Occupation des salles] E --> F((Salles.determiné)) F --> E F --> G[Montage des emplois du temps] G --> H(()) H --> I((Emplois du temps établi)) H --> J((Rentrée scolaire)) I --> K[Assurer le déroulement des cours] J --> K K --> L(()) L --> M((Ensg .dispensé)) L --> N((Date contrôle)) M --> O[Contrôles] N --> O O --> P((Notes attribuée)) P --> Q[Saisie notes] Q --> R(()) R --> S((Notes saisie)) R --> T((Fin période)) S --> U[édition des bulletins] T --> U U --> V((Bulletin diffusée)) </pre> | Bureau gestion | traitement Conversationnel | AV.rentree |
| | Bureau gestion | traitement Conversationnel | AV. rentrée |
| | Bureau gestion | traitement Conversationnel | AV. rentrée |
| | Bureau gestion | traitement Conversationne | AV. rentrée |
| | Professeur | manuel | à chaque Vacation |
| | Professeur | manuel | |
| | Atelier de saisie | conversationnel | après correction |
| | Centre calcul | T.par lot | fin periode |
| - Chaque action organisée devient une tâche. | | | |
| Préparé par : Abdellah ELAouni Malika Sami | | | |

- Les tâches pour les quelles ces trois points sont identiques seront regroupées, un tel ensemble cohérent est appelé phase



Remarque le formalisme, très proche de celui du niveau MCT. Les concepts de **tâche** et de **phase** remplacent ceux d'**action** et d'**opération**: L'assemblage des phases ainsi présenté constituera le modèle organisationnel des traitements.

6-1- définitions et formalisme

Les notions qui figurent sur le **MOT** sont définies comme suit :

Règles d'organisation : c'est l'expression de l'organisation mise en place en termes de postes de travail, de nature de traitement et de chronologie.

Exp : - le bureau de gestion monte les emplois du temps par un traitement conversationnel

tâche : action pourvue d'une organisation définie par les règles d'organisation.

Epx : - le centre de calcul édite les notes des élèves par un traitement automatique

Événement : fait réel dont la venue a pour effet de déclencher l'exécution d'une ou plusieurs tâches.

Exp : fin de période

Synchronisation : condition booléenne, traduisant les règles de gestion et d'organisation, que doivent vérifier les événements pour déclencher les tâches.

Exp : notes attribuée **et** fin de période

phase : ensemble de tâches dont l'enchaînement ininterrompible compte tenu de l'organisation mise en place n'est conditionné par l'attente d'aucun événement autre que le déclencheur initial

Exp : Avant la rentrée scolaire l'établissement établit les emplois du temps des classes et des professeurs alors pour réaliser ceux-ci :

- répartir les élèves par classe
- attribuer les classes aux professeurs
- définir l'occupation des locaux
- monter les emplois du temps

Règle d'émission : condition traduisant les règles de gestion et d'organisation auxquelles est soumise l'émission des résultats d'une phase.

Exp : Si les classes sont formés et attribuées aux professeurs alors emplois du temps établit

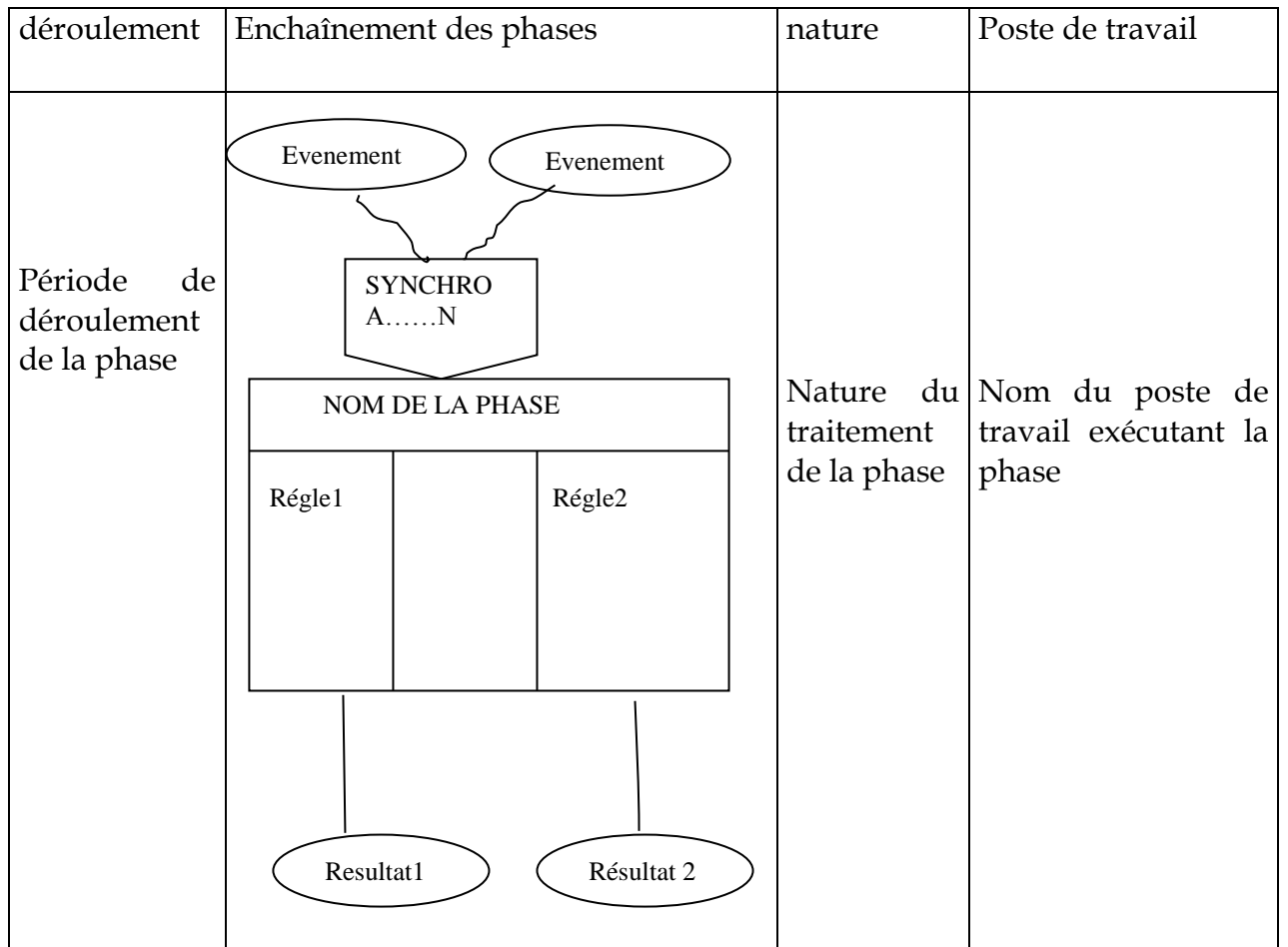
Résultat : produit de l'exécution d'une phase. Le résultat est un fait réel de même nature que l'événement, il pourra être déclencheur d'une autre phase.

Exp : bulletin diffusé

6-3 FORMALISME :

Le formalisme est le même que celui du modèle conceptuel des traitement à condition de remplacer les opérations par les phases. Trois colonnes précisent pour chaque phase :

- le déroulement chronologique
- la nature du traitement
- le poste de travail concerné



6-4- constitution du modèle

- Création des tâches
- Création des phases (passage des tâches aux phases, découpage en procédure, enchaînement des phases)
- Événements, Règles de synchronisation et d'émission

6-5 Modèles externes

Objectif : de faire le rapprochement entre le MCD et les traitements définies pour les phases automatisées

Pour les phases manuelles le travail s'est achevé avec des tâches qu'elles englobent, des règles d'organisation qui les gouvernent, du mode opératoire décrivant leur mise en œuvre, alors ne sont concernés que les phases automatisées,

Pour chaque phase automatisée on définit des **traitements et leurs fonctions** et un MCD pour chaque traitement qu'on appelle modèle externe **ME**

Remarque :

- le **MCD** appelé **ME (ou la vue externe)** car est un modèle qui représente la vue des données à travers les traitements automatisés

Règle de construction ME

Règle1 : *On construit un ME pour une fonction particulière de traitement*

Il existe deux types de fonction de traitement soit :

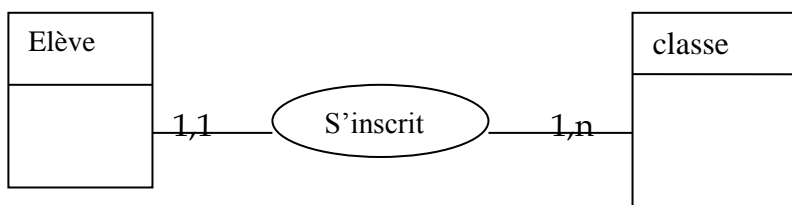
-la fonction de MAJ

-la fonction de consultation

Exp : les fonctions de traitement pour la phase1

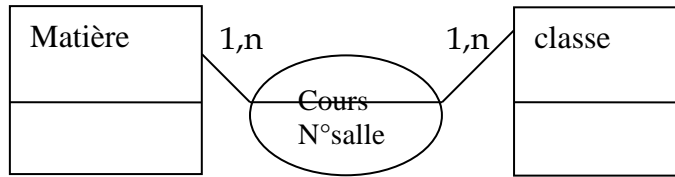
- réparation des élèves dans les classes : Fonction de MAJ

ME pour cette fonction de traitement



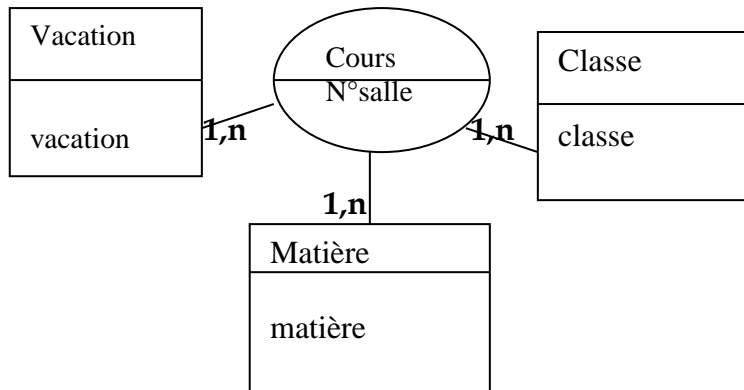
- fixer les salles pour les classes suivant les matières : fonction de MAJ

ME pour cette fonction de traitement



- établissement des emplois du temps : fonction **de MAJ**

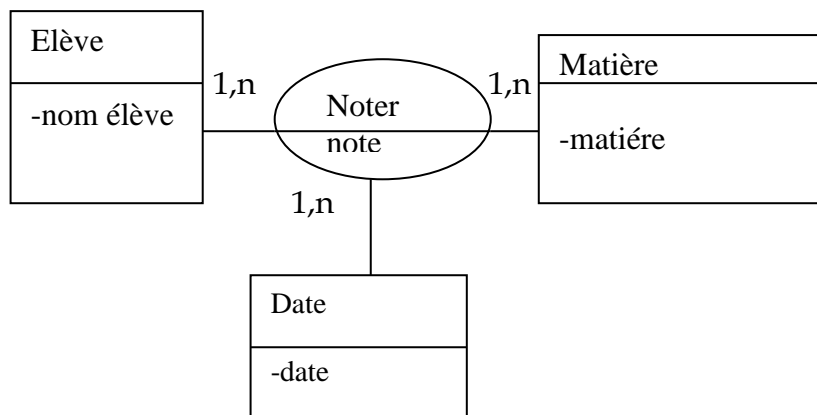
ME pour cette fonction



Exp : les fonctions de traitement pour la phase 4

- saisie des notes des élèves : fonction de MAJ

ME pour cette fonction de traitement



Alors pour chaque phase automatisée on définit les fonctions de traitements nécessaires à l'objectif du système étudié. Et pour chaque fonction on établit **ME**

Règle2 : on listera pour chaque modèle externe les données manipulées en se référant au dictionnaire de données

Le recensement des données dans le MOT sera donc confronté au dictionnaire de données de façon à éviter de créer des synonymes ou des polysèmes

Règle 3 : *on exprime les modèles externes dans le formalisme du modèle conceptuel des données*

Le ME doit être vérifié (pas de propriété répétitive ou sans signification) on s'efforcera de respecter au moins les deux premières règles normales

Règle 4 : *on utilisera pour construire les ME , les blocs logiques d'entrée-sortie*

Exp : comment construire un ME

Soit la fonction consistant à éditer, l'emploi du temps d'une classe.

le bloc logique de sortie est alors

| classe de 6°As3 | | | |
|-----------------|-----------------------|---------------------------|-------|
| | lundi | mardi | |
| 9h à 10h | Français Salle n°2 | Mathématique Salle n°6 | |
| 10h à 11h | Arabe Salle n°2 | | |
| | | | |

Enfin, supposons que le dictionnaire nous fournisse les données suivantes liées à ce domaine

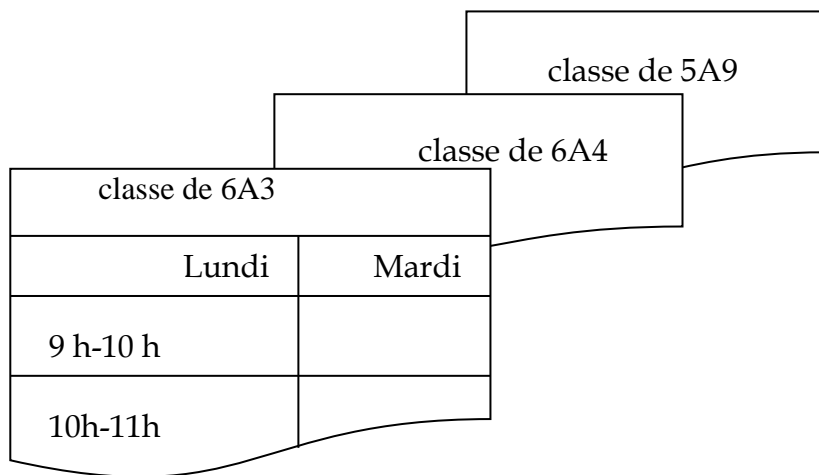
- classe,
- numéro de salle,
- nom de l'élève,
- vacation,
- matière

Question :

1- Etablir le modèle externe pour cette fonction

2- si la fonction avait été l'édition de l'emploi du temps de chaque classe.

Etablir le modèle Externe correspondant ,sachant que Le bloc logique de sortie aurait alors une dimension supplémentaire, due aux diverses classes, selon le dessin suivant



Modèle Externe

1- Règle N° 1 : Le traitement d'édition, correspond à une fonction de consultation. Le ME

Règle N° 2 : Référence au dictionnaire

- Les données « Classe » et « Matière » sont conformes.

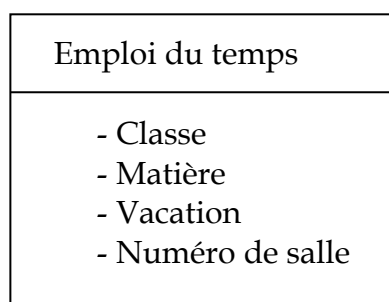
- le « Jour » et la « Tranche horaire » sont absents du dictionnaire où figure la donnée « Vacation ». Le traitement ne manipulant pas ces deux données séparément, on peut ne conserver que la « Vacation » qui est, en fait, le couple (Jour, Tranche horaire).

Les données que l'on va formaliser sont donc

- classe,
- matière,
- vacation,
- numéro de salle,
- nombre d'élèves.

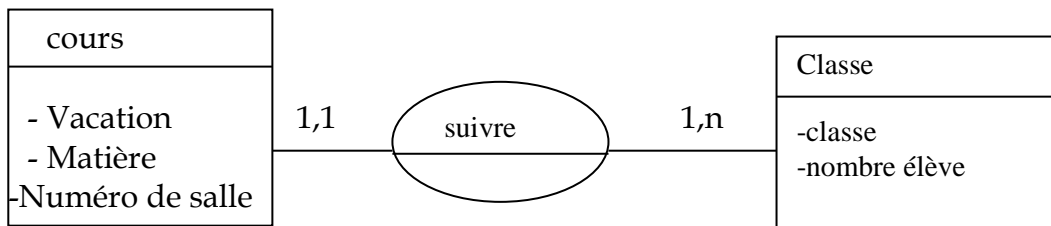
Règles N° 3 et N° 4 : Formalisation

On créera d'abord un objet «Emploi du temps», identifié par la «Classe»



La non répétitivité des propriétés impose de sortir de cet objet « Vacation », « Matière » et « Numéro de salle ». Seuls subsistent « Classe », dès lors, il est plus judicieux d'appeler «Classe» cet objet. Une « Matière » et un « Numéro de salle » se trouvent dans plusieurs vacations. Par conséquent, aucune de ces deux données ne pourra identifier un objet portant la «Vacation».

La solution est alors de créer un objet que l'on pourrait appeler « Cours », identifié par la «Vacation », portant la « Matière » et le «Numéro de salle », et rattaché à la « Classe ».

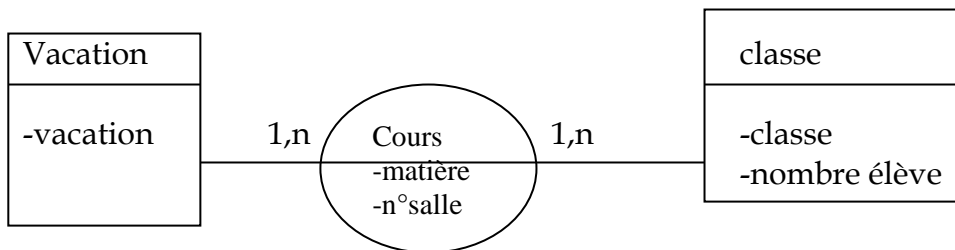


Du point de vue de ce traitement : édition de l'emploi du temps d'une seule classe, ce modèle externe est satisfaisant.

Il est formellement correct les objets dans le système de référence de ce traitement ont une existence propre. Notre modèle externe est donc bien de la nature d'un modèle conceptuel des donné".

2- Alors, l'objet « Cours » n'aurait plus été conforme au formalisme, puisqu'à une « Vacation » pouvait correspondre plusieurs « Matière » et « Numéro de salle ». La solution la plus « économique » aurait été la création d'une relation « Cours », porteuse des propriétés « Matière » et « N° de salle », si, pour une vacation et une classe, existait une seule matière enseignée dans une seule salle (pas de fractionnement des classes).

Dans ce cas le modèle externe aurait été



6-6 -Validation du MOT et du MCD

Le système d'information intégrant à la fois la vue statique et la vue dynamique.
La vue statique est schématisée par un **MCD**, La vue dynamique est schématisée par Les **ME** .Pour ne pas avoir des incohérence entre ces deux représentations alors, il faut rendre l'aspect statique et l'aspect dynamique compatible

1-Validation de chaque ME par rapport au MCD brut :

- Valider un ME par rapport au MCD s'assurer qu'il est déductible du MCD

* corrections éventuelles des ME

Règles de validation

Il faut vérifié que toutes les données conceptuelles et les objets manipules dans chaque opération sont présent dans le dictionnaire de données .la validation se poursuit par la mise à jour du dictionnaire de données. La validation procédera par

- la validation des propriétés externes
- la validation des objets externes
- la validation des relations externes
- la validation des cardinalités externes

2- Validation du MCD par rapport à l'ensemble des ME

- valider un MCD : ne garder de celui-ci que ce qui est strictement nécessaire aux ME validés

* correction éventuelle du MCD brut

La validation aura donc fournie un MCD validé et des sous modèles validés qui garanties la faisabilité d'un ensemble de traitements

6-7-production des sous -modèles conceptuels

Définir au sein du MCD, les sous modèles permettant de déduire chacune des vues externes. On est alors assuré, que la vue globale des données permet d'effectuer l'ensemble des traitements souhaités

7- LE MODELE LOGIQUE DE DONNÉES (MLD)

la validation a produit un MCD validé et des ME validés pour les traitements souhaités

Alors les objectifs pour ce niveau

- définir l'organisation logique des données traduisant le MCD validé
- optimiser cette organisation par rapport aux besoins des traitements

7-1. La représentation logique des données

Cette représentation constitue le passage du MCD à l'implémentation physique de la base de données.

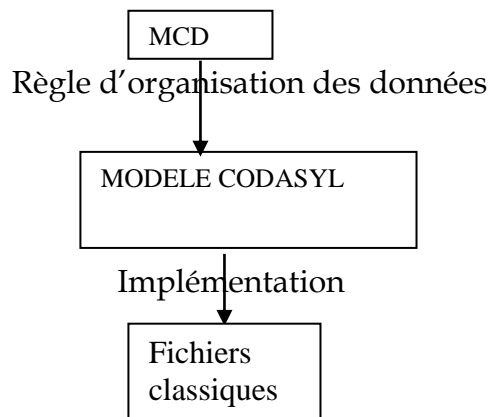
On exprime alors le lien logique entre les données dans le formalisme approprié au système étudié. Merise propose au niveau logique des données deux formalismes associées chacune aux solutions techniques de type SGBD :

- Formalisme **CODASYL**
- formalisme **RELATIONNEL**

7-1-1-le modèle Codasyl.

Merise a adopté pour la description logique des données le modèle Codasyl. (Conférence On Data Système Langage) pour accéder à l'implémentation des bases des données fichiers classiques

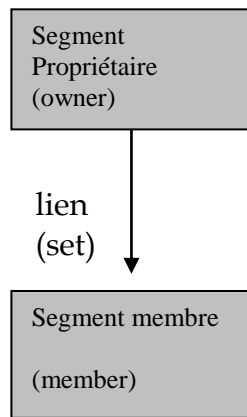
le modèle Codasyl.



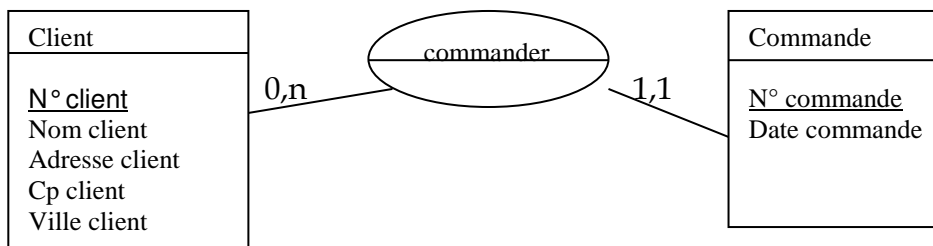
Modèle codasyl

7-1--2- La norme Codasyl

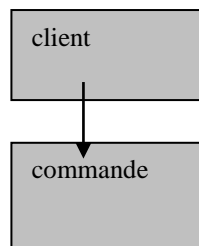
Le module de base de ce schéma est composé de deux segments de données (RECORD), dont l'un est appelé **propriétaire** (OWNER) et l'autre **membre** (MEMBER). Entre ces deux segments est établi un chemin nommé **lien** (SET)



Reprenons l'exemple, mettant en scène les objets « CLIENT » et « COMMANDE »



La cardinalité (1,1) détermine le record membre, et la cardinalité (0,n) le record propriétaire.



7-2-Règles de passage

Le passage du modèle conceptuel des données au modèle Codasyl se fait en appliquant des règles d'algorithme très simples.

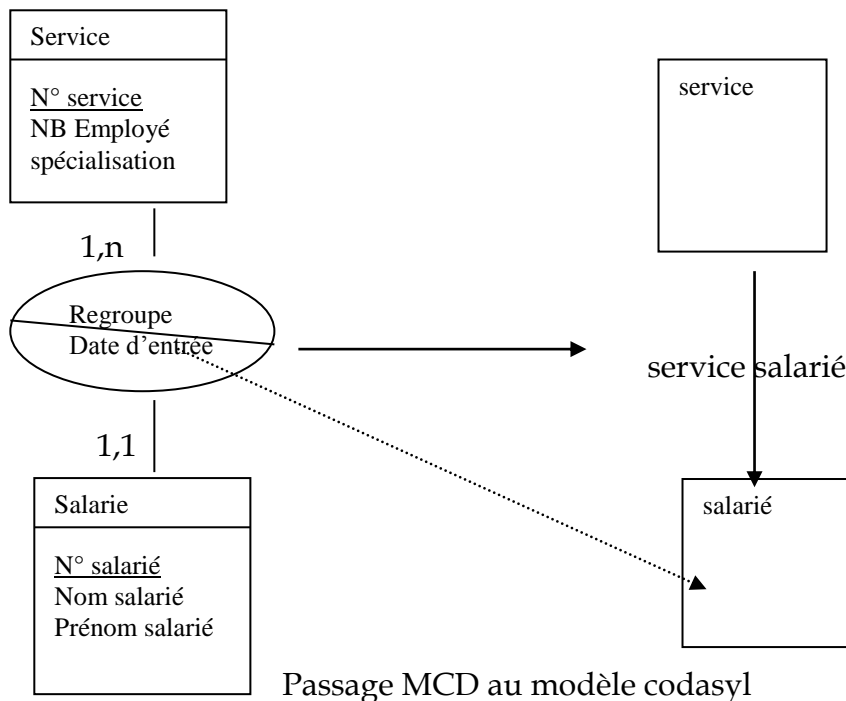
1^{re} règle

La propriété se transforme en **champ** (item), et l'objet se transforme en **Record**.

2^{re} règle

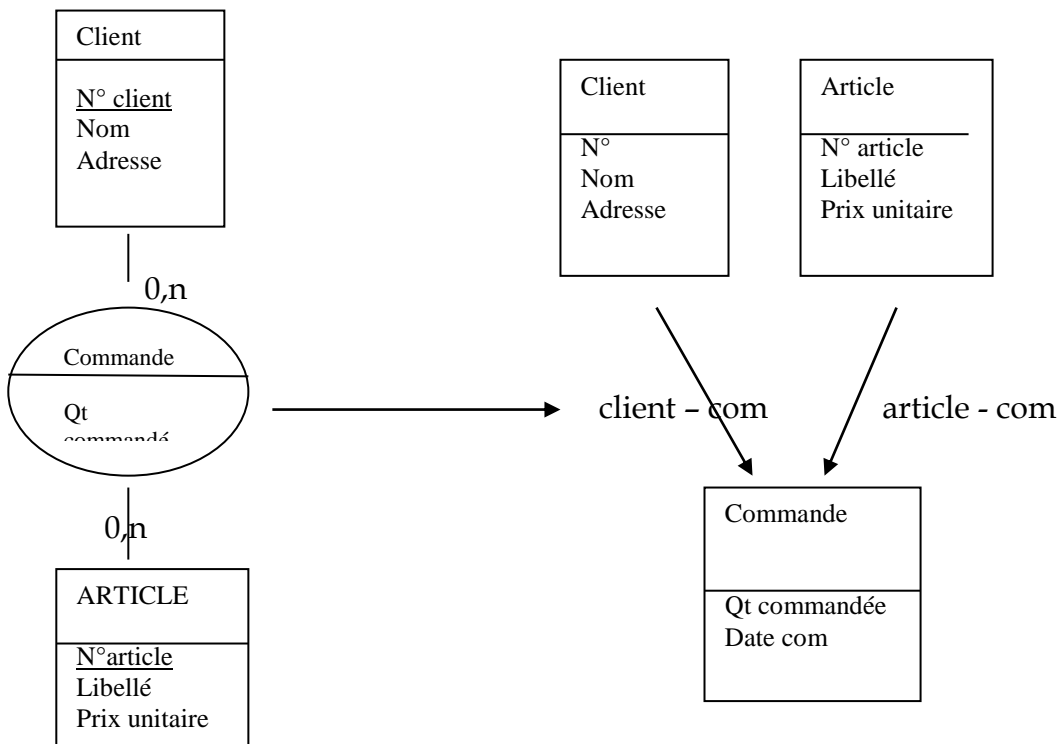
Une relation binaire de **type père -fils** ayant des cardinalités (0,n - 0, 1), (1,n - 0,1), (0 ,n - 1,1) ou (1,n , 1,1) se transforme en un **set** orienté vers le record issu de l'objet ayant les cardinalités 0,1 ou 1,1.

Si cette relation est porteuse de propriétés, ces propriétés migrent vers le record issu de l'objet ayant les cardinalités 0, 1 ou 1,1



3^{ème} règle

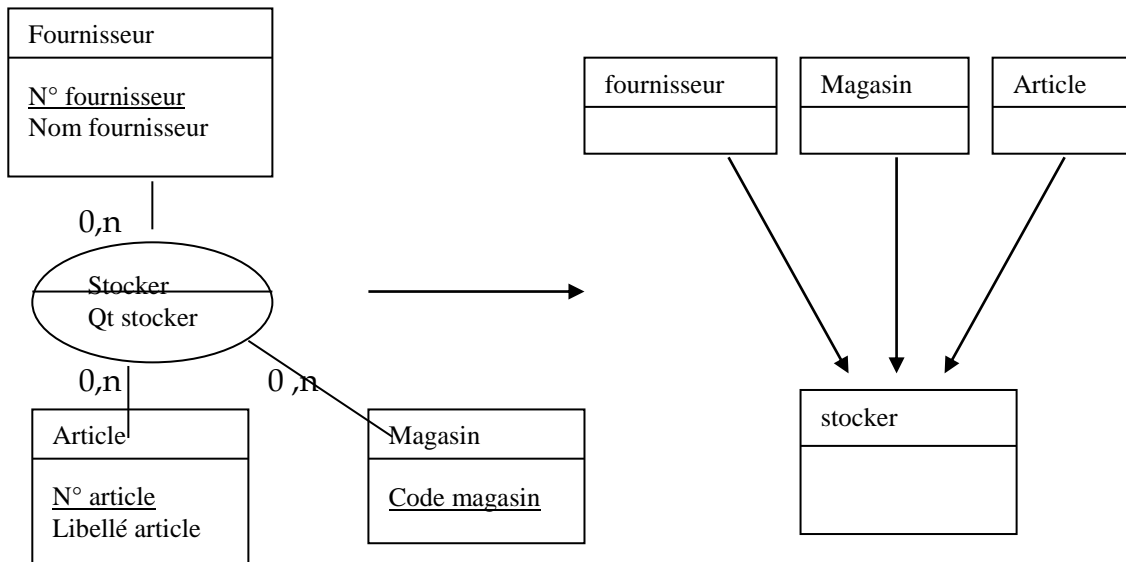
Les autres relations binaires de type **autre que père - fils** se transforment en un record et deux sets orientés vers ce record



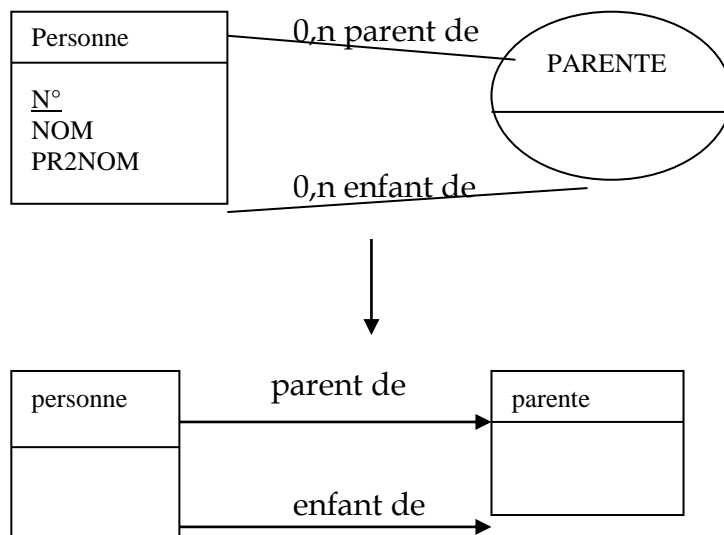
Si la relation est une relation vide, le record issu de cette relation ne contient que des pointeurs. C'est un pseudo-record.

4eme règle

Une relation **n-aire** (**de dimension supérieur à 2**)se transforme en un record et autant de sets que d'objets présents dans sa collection. Tous ces sets sont orientés vers le record issu de la relation



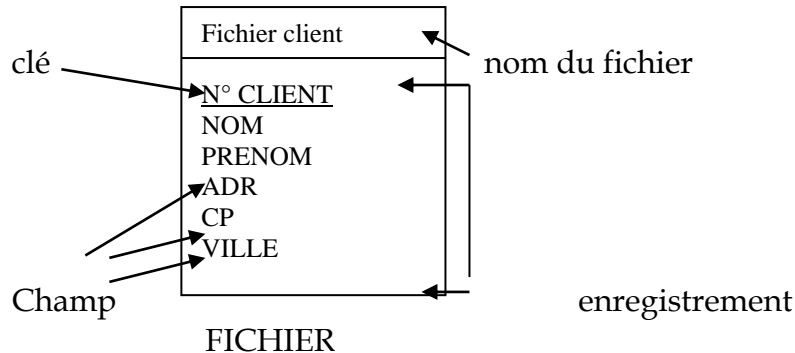
Pour La relation réflexive se transforme en record ou pseudo-record et deux sets pointés vers celui ci



cas particulier de la relation réflexive

7-2- Les fichiers

Un fichier regroupe des informations de même nature (par exemple le fichier fournisseurs, le fichier clients, le fichier article ...), sous forme de champs (code fournisseur, nom fournisseur, code article ...), regroupés en enregistrements" (client, fournisseur, article ...). La clé, ou identifiant est un champ qui permet de connaître chaque occurrence d'enregistrement



7-2-1- Règles de transformation du modèle Codasyl en fichiers classiques

- Un record identifié est un record issu d'un objet conceptuel. Un record non identifié est issu d'une relation conceptuelle.
- Un fichier principal est un fichier issu d'un record identifié.
- Un fichier d'association est-un fichier issu d'un record non identifié.
- Un fichier de correspondance est un fichier issu d'un set, et permettant de mettre en lien deux fichiers principaux ou un fichier principal et un fichier d'association.

1re - règle

Tout record non membre (qui n'a pas de record propriétaire) se transforme en fichier principal L'identifiant du record devient clé du fichier, et les data-items en deviennent les champs.

2ème règle

Tout record membre possédant un identifiant génère un fichier principal et éventuellement un fichier de correspondance. L'identifiant du record devient clé du

fichier principal et l'(les) identifiant (s) du (des) record (s) propriétaire(s) migre(nt) en champ (s) non-clé dans ce fichier. Des fichiers de correspondance sont créés entre le fichier

issu du record membre et le fichier issu d'un record propriétaire chaque fois que le set les unissant est exploiter dans le sens propriétaire membre. La clé d'un tel fichier est composée des clés des fichiers qu'il met en lien.

3eme règle

Tout record membre sans identifiant devient fichier d'association et éventuellement fichier de correspondance. La clé de ce fichier est composée des clés de ses fichiers propriétaires.

8 - MODELE OPERATIONNEL DES TRAITEMENTS (MOPT) :

MOPT exprime le **comment**.

On s'interroger sur la façon dont la machine accomplira ce que l'on attend d'elle

8-1- objectif :

Le niveau opérationnel du traitement a pour but de concevoir et visualiser clairement l'architecture de l'application informatique, en définissant les modules de traitement et leur diagramme d'enchaînement.

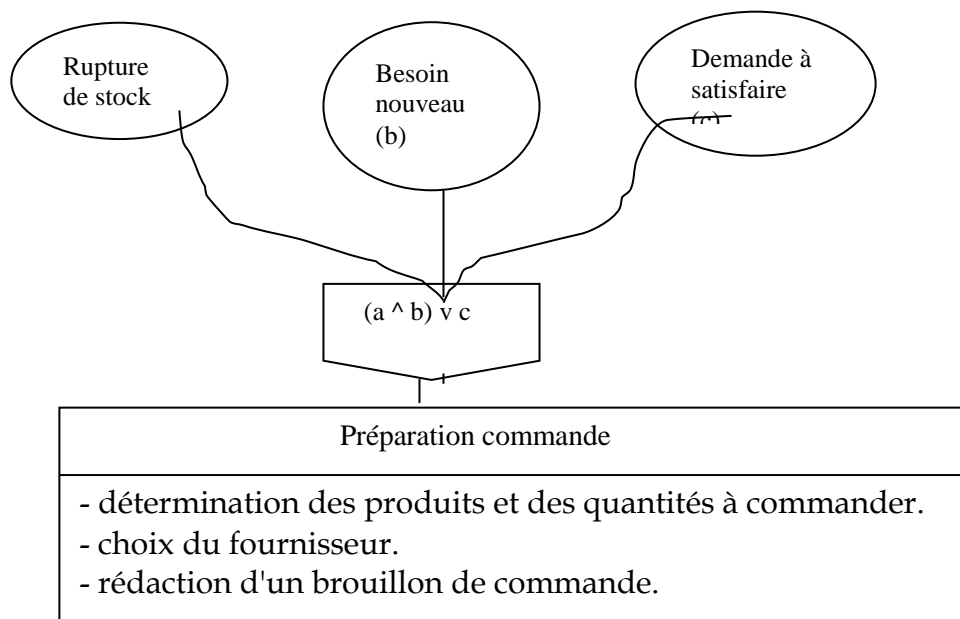
8-2- définition des modules de traitement :

Au sein de chaque phase, seront établis des modules de traitement. Pour définir les modules de traitement à l'intérieur de chaque phase on définit des fonctions, soit de **mise à jour**, soit de **consultation** relative à chaque tâche

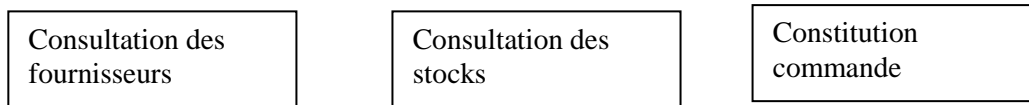
Exemple : L'opération « Préparation d'une commande » regroupe les actions ininterruptibles suivantes:

- détermination des produits et des quantités à commander.
- choix du fournisseur.
- rédaction d'un brouillon de commande.

On suppose L'opération « Préparation d'une commande » correspond à une seule phase « Préparation d'une commande » réaliser par un magasinier par un traitement conversationnelle à une période donnée



Exemple : dans une phase de Préparation commande il y a des modules tel que

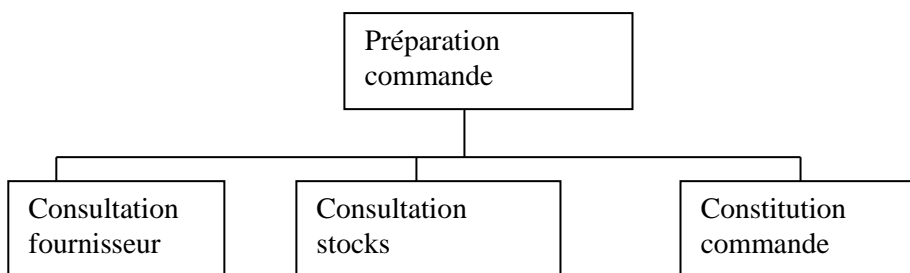


8-3- Enchaînement des modules :

Pour chaque phase l'ensemble des traitements est ainsi structuré en **modules fonctionnels**

Au descriptif de la phase, On adapte une représentation arborescence des modules fonctionnels. Un module de tête portant le nom de la phase sera ajouté.

Chaque phase est représentée par un diagramme hiérarchique composé des modules de type PERE-FILS.

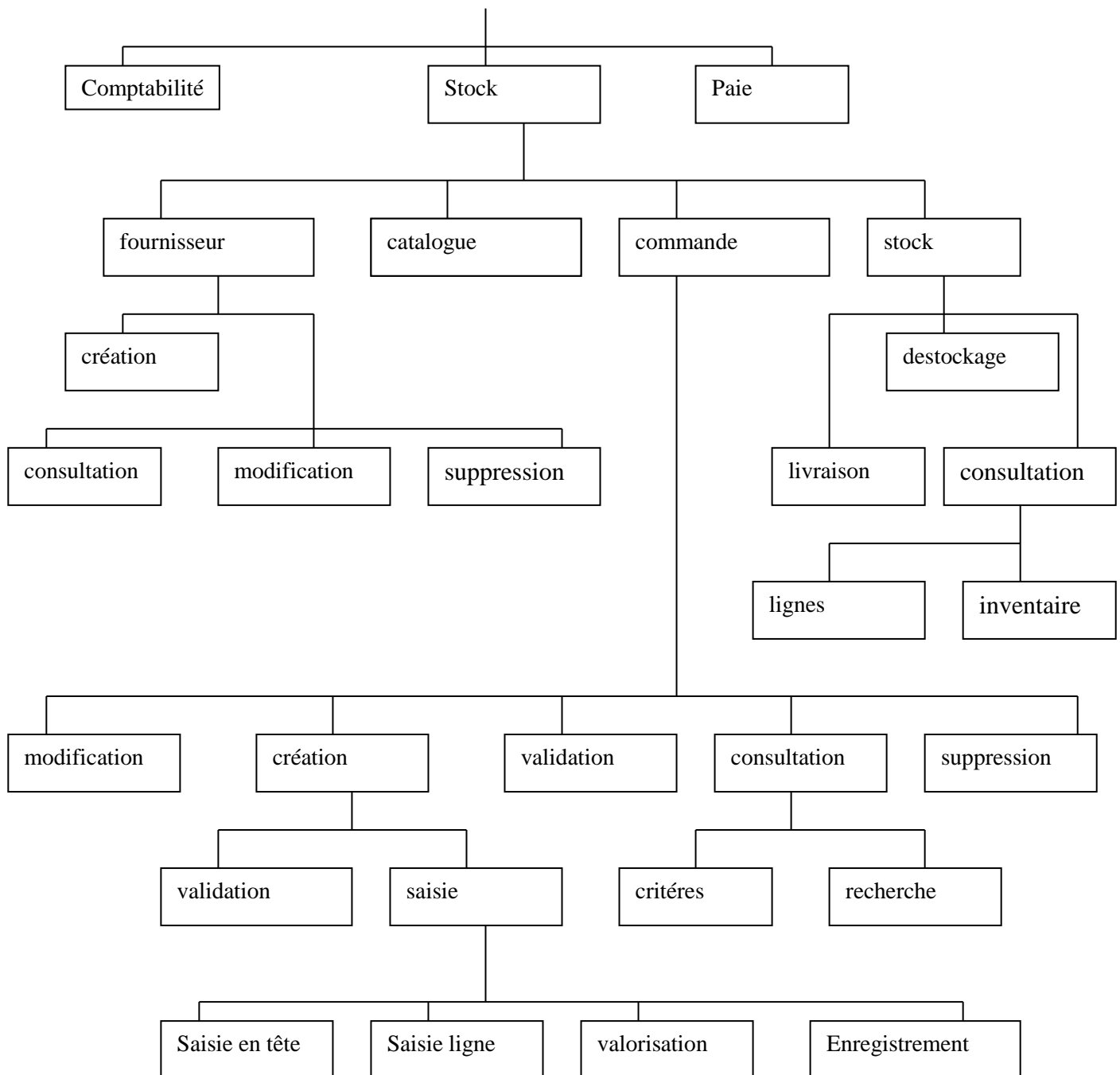


8-4- Constitution du diagramme global

Les diagrammes hiérarchiques de chaque phase doivent être intégrés les uns aux autres pour constituer le diagramme unique d'enchaînement de l'application.

le diagramme d'enchaînement de l'application constitué permet :

- de visualiser facilement l'ensemble de l'application,
- de développer l'application du haut vers le bas,
- en coupant le diagramme au diagramme d'étude, de ne présenter que la partie de l'application suffisante à l'objectif poursuivi.



L'ensemble des diagrammes hiérarchiques est nommé un schéma logique du traitement

Le diagramme d'enchaînement de l'application permet :

- de visualiser l'ensemble de l'application

- de développer l'application du haut vers le bas en intégrant progressivement les modules de plus bas niveau
- En coupant le diagramme en programme d'étude

8-5- Outils de description

Chacun des modèles du diagramme hiérarchique constituant une unité de programmes autonomes, on précisera :

a- les entrées/sorties

- Les entrées ;
- Les traitements ;
- Les sorties.

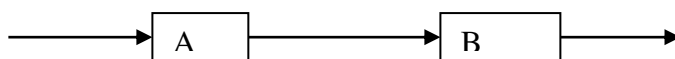
Exp : pour un module de consultation du stock on indique

- Une saisie sur un clavier des critères de sélection
- une recherche des produits correspondants
- un affichage sur écran des résultats

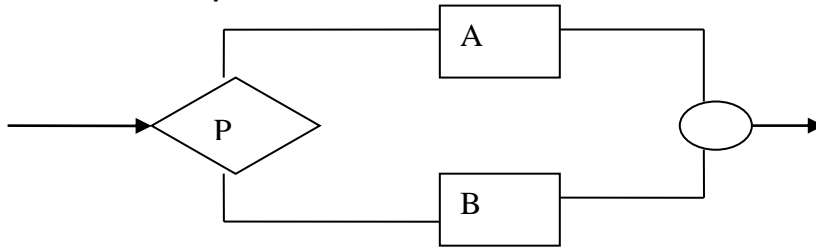
b- le langage de description

Pour chaque module sera donc établie une description des traitements, afin de préparer le mieux possible une programmation structurée, on utilise un pseudo code

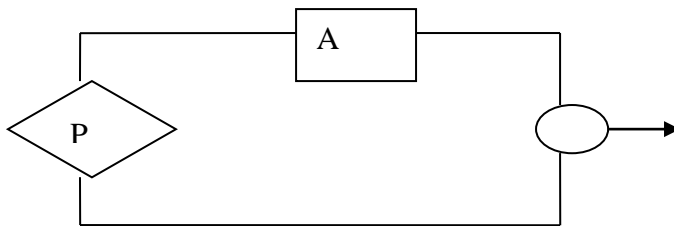
- la séquence



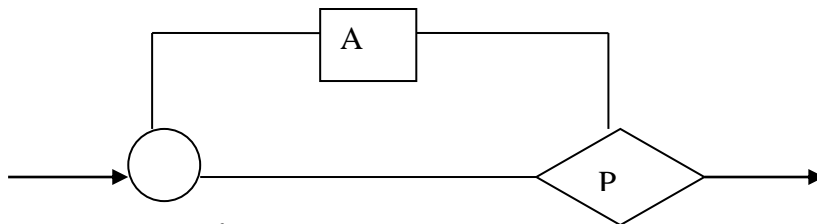
Traduit l'enchaînement de deux traitements

- la condition si/ sinon

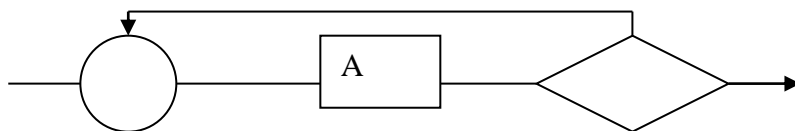
Si La proposition P est vrai, exécuter A, sinon B

- la condition si

si la proposition P est vrai ,exécuter A

- la condition Tant que

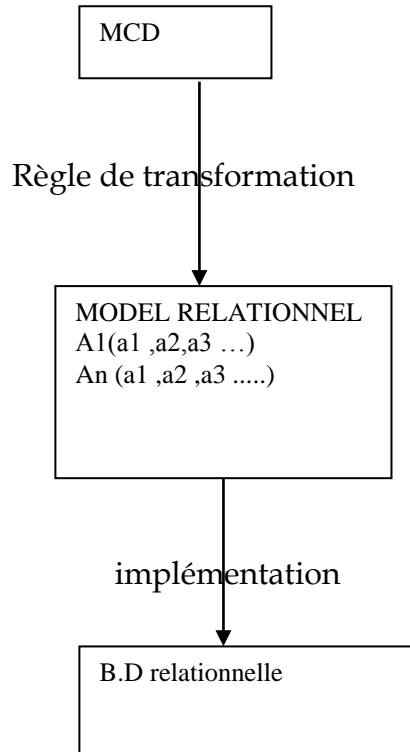
Tant que P est fausse, ne pas exécuter A

- la condition Jusqu'à ce que

Exécuter A jusqu'à ce que P soit vraie

9- formalisme RELATIONNEL

La traduction du **MCD** en modèle logique relationnel s'effectue directement par la transformation des Entités (objets) conceptuelles en relations, en fonction des règles de passage précises



Passage du MCD à la base de données relationnelle

Le passage du modèle conceptuel des données au modèle logique des données s'effectue en appliquant des règles s'appuyant sur les cardinalités des couples objet - relation

9-1- Le modèle relationnel

Concepts

L'élément de base est la relation au sens La relation relationnelle se distingue de la relation conceptuelle.

Dans le modèle conceptuel, la relation représente une association de plusieurs objets. Dans le modèle relationnel, la relation est une association d'attributs (données).

Nom de la relation (Attribut 1, attribut 2..... attribut n).

L'ensemble des occurrences de la relation est représenté par une table, dont les colonnes contiennent les valeurs prises par les attributs de cette relation. Les lignes de la table

représentent les occurrences de la relation, ou tuples. Chacune de ces lignes est identifiée par un attribut ou un ensemble d'attribut appelé clé primaire.

Les attributs clés sont placés en tête de la relation et soulignés.

Exemple:

La relation « CLIENT » entre les attributs N° client », « Nom client », « Adresse client », « Cp client » et « Ville client » se présente de la manière suivante:

CLIENT (N°client , Nom client, Adresse client, Cp client, Ville client)

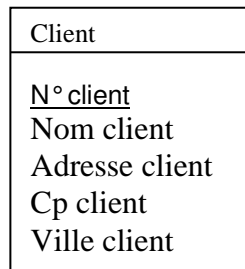
La relation «COMMANDER » entre les attributs N° commande », « N° article », « N° client » et « Quantité commandée » prend la forme suivante

COMMANDER (N° commande, N° client , N°Article, Quantité commandée)

9-2- Règles de passage

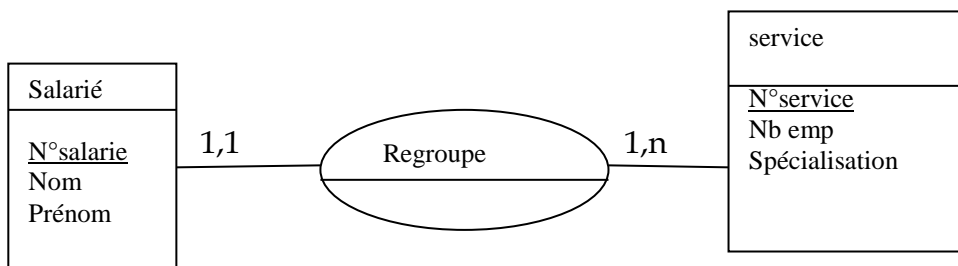
Règle 1

Un objet conceptuel se transforme en relation Chaque propriété se transforme en attribut. L'identifiant de l'objet devient la clé primaire de la relation.



Règle 2

Une relation binaire (ou réflexive) ayant des cardinalités (1,1) -(1,n) ou (1,1)-(0,n) se traduit par une redondance de l'identifiant de l'objet à cardinalité (1,n) ou (0,n) dans la relation issue de l'objet à cardinalité (1,1)



Salarié (N°salarie , Nom ,prénom N°service)

Service (N°service , Nb emp , spécialisation)

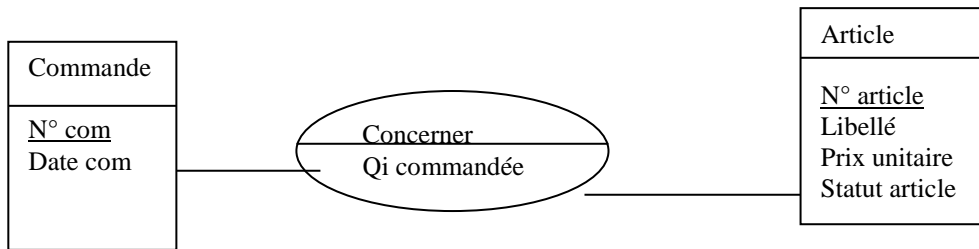
L'identifiant de l'objet à cardinalité (1,1) devient la clé primaire de la relation.

La propriété dupliquée devient clé étrangère dans la relation. Si la relation est réflexive, c'est l'identifiant de l'objet qui est dupliqué dans la relation issue de ce même objet après avoir été renommé.

Si la relation (conceptuelle) est porteuse de propriétés, celles-ci se retrouvent comme attributs dans la relation relationnelle issue de l'objet à cardinalités (1,I).

Règle 3

Une relation n-aire du modèle conceptuel, porteuse ou non de données, se transforme en une relation du schéma relationnel ayant comme clé primaire composite les attributs issus des identifiants des objets participant, à cette relation conceptuelle

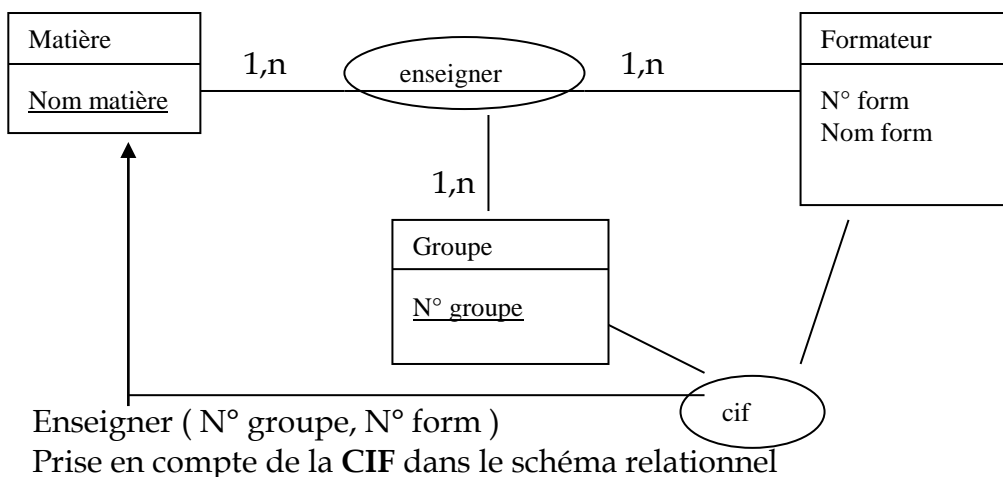


Commande (N°com , date commande)
Article (N°article , libellé , prix unitaire , statut article)
Concerner (N°com , N°artice , quantité commandée)

Cas particulier :

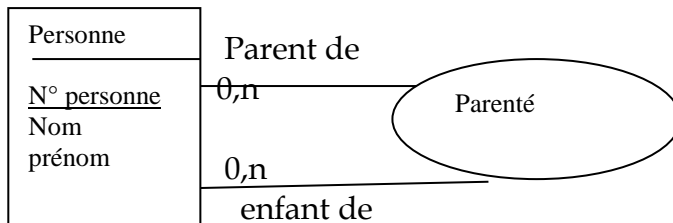
Une relation binaire à cardinalités (0, I)-(0,n) ou (0, I).(I,n) se traduit soit suivant la règle 2, soit selon la règle 3.

S'il existe une CIF sur la relation du modèle conceptuel, l'identifiant de l'objet concerné est sorti de la clé de la relation relationnelle générée



Règle 4

La relation réflexive, si elle ne répond pas à la définition de la règle 2 se traduit par une relation porteuse de deux attribues. Duplications de l'identifiant de l'objet, et toutes deux renommées.



parenté (N°parent , N° enfant)

personne (N°personne , nom , prénom)

Cet ensemble d'attributs constitue la clé primaire composite de cette relation.

9-3. Les clés

Dans une table relationnelle, il ne peut y avoir deux lignes identiques, d'où la notion de clé. La valeur prise par la clé doit nous permettre d'identifier la ligne.

a) Clé primaire (primary key)

C'est cette clé qui assure l'unicité de la ligne. Elle est dite « clé primaire simple » si elle ne comporte qu'un attribut, ou « clé primaire composite » si elle en comporte plusieurs.

b) Clé étrangère (foreign key)

Un attribut est dit clé étrangère dans une relation lorsqu'il est également clé primaire dans une autre.

c) Clé candidate

Cet attribut n'est pas clé primaire mais est également apte assurer l'unicité d'une ligne.

9-4- Les formes normales

Nous ne donnerons ici que les trois premières formes normales, la normalisation étant traitée en détail dans les ouvrages spécialisés en bases de données. Nous retrouvons la notion de dépendance fonctionnelle qui a guidé la construction du modèle conceptuel.

Les trois premières formes normales ont pour objectif de permettre la décomposition de relation sans perte d'information, à partir de **la notion de DF**. L'objectif de cette décomposition est d'aboutir à un schéma conceptuel représentant les entités et les relations canoniques du système étudié

a) - Première forme normale (1FN)

1FN permet simplement d'obtenir des tables rectangulaires

Une relation est en première forme normale (1FN) si :

- la relation possède une clé
- tout attribut est atomique (élémentaire)

Exemple1 : Clients

| nom | Adresse |
|--------|----------------------------|
| Alaoui | 21 rue de résidence Tanger |
| Chakir | 45 avenue pasteur Rabat |
| Bakali | 123 rue Lavoisier Casa |

Si on considère l'Adresse est composée de 3 parties (Rue , N°magasin , ville) . la relation Clients N'est pas conforme à la 1FN alors la restructurer (décomposition des champs non élémentaires)

| Nom | N°magasin | Rue | ville |
|--------|-----------|-----------|--------|
| Alaoui | 21 | résidence | Tanger |
| Chakir | 45 | Pasteur | Rabat |
| Bakali | 123 | Lavoisier | Casa |

b) Deuxième forme normale (2FN)

2FN permet d'assurer l'élimination de certaines redondances en garantissant qu'un attribut n'est déterminé seulement par une partie de la clé

Une relation est en deuxième forme normale (2FN) si :

- elle est en première forme normale (1FN)
- et que tout attribut non clé ne dépend pas d'une partie de cette clé.

Exemple :

| <u>Num salarié</u> | Nom | Numprojet | Heure |
|--------------------|--------|-----------|-------|
| 236 | Said | 1 | 18,5 |
| 236 | Said | 2 | 6,7 |
| 369 | Ikram | 2 | 8,5 |
| 450 | Hassan | 3 | 23,5 |
| 450 | Hassan | 1 | 4,8 |

Soit (Num_salarié , Numprojet) clé de la relation

Num_salarié + Numprojet \longrightarrow Heure

Num_salarié + Numprojet $\not\longrightarrow$ Nom

Num_salarié \longrightarrow Nom (seul)

Alors il faut scinder la table en 2

| <u>Num salarié</u> | Nom |
|--------------------|--------|
| 236 | Said |
| 369 | ikram |
| 450 | Hassan |

| <u>Num salarié</u> | Numprojet | Heure |
|--------------------|-----------|-------|
| 236 | 1 | 18,5 |
| 236 | 2 | 6,7 |
| 369 | 2 | 8,5 |
| 450 | 3 | 23,5 |
| 450 | 1 | 4,8 |

C) Troisième forme normale (3FN)

3FN permet d'assurer l'élimination de redondances dues aux dépendances transitives

Une relation est en troisième forme normale (3FN) si :

- elle est en 2FN
- et que tout attribut non clé n'est en dépendance fonctionnelle que de la clé (pas de transitivité)

Exemple :

Soit 3 attributs (A ,B ,C) ,A la clé primaire

Si A \longrightarrow B et B \longrightarrow C
 A $\not\longrightarrow$ C

Exemple

| Nom | Num_salarie | Date naissance | service | Nomservice | Numchef |
|---------|-------------|----------------|---------|--------------|---------|
| chaouki | 51 | 15/10/19888 | 5 | informatique | 46 |
| daoudi | 52 | 12/04/1989 | 6 | vente | 41 |

La table n'est pas en 3FN

Num_salarie \longrightarrow service \longrightarrow Nomservice
 \longrightarrow

Alors on scinde la table en 2

| Nom | Num_salarie | Date naissance | service |
|---------|-------------|----------------|---------|
| chaouki | 51 | 15/10/19888 | 5 |
| daoudi | 52 | 12/04/1989 | 6 |

| service | Nomservice | Numchef |
|---------|--------------|---------|
| 5 | informatique | 46 |
| 6 | vente | 41 |

d- forme normale de BOYCE-CODD (FNBC)

FNBC est une extension de la **3FN** (FNBC est une forme spécifique de la 3FN)

Elle seule supprime toute dépendance transitive

Une relation est en FNBC si :

- Seulement les seuls DF élémentaires sont celles dans les quelles une clé détermine un attribut

FNBC si pour tout $X \rightarrow Y$, X est une clé .c'est tout

- Si la table possède plus d'un candidat pour la clé primaire, elle doit être examinée selon le point de vue de chacune de ces clés choisie (choix suivant l'utilisation de la table)

Après le choix de la clé, si après un tel examen de la relation, elle se trouve toujours dans la (3FN) Alors elle est dans (FNBC)

Exemple :

| <u>Num salarié</u> | Nom | Numprojet | Heure |
|--------------------|--------|-----------|-------|
| 236 | Said | 1 | 18,5 |
| 236 | Said | 2 | 6,7 |
| 369 | Ikram | 2 | 8,5 |
| 450 | Hassan | 3 | 23,5 |
| 450 | Hassan | 1 | 4,8 |

La clé peut être (Numsalarié +Numprojet) **Ou** (Nom+ Numprojet)

Quelque soit la clé choisie la relation n'est pas en **3FN** alors on doit la rendre dans la 2FN

Alors il faut scinder la table en 2

R1

| <u>Num salarié</u> | Nom |
|--------------------|--------|
| 236 | Said |
| 369 | Ikram |
| 450 | Hassan |

R2

| <u>Num salarié</u> | Numprojet | Heure |
|--------------------|-----------|-------|
| 236 | 1 | 18,5 |
| 236 | 2 | 6,7 |
| 369 | 2 | 8,5 |
| 450 | 3 | 23,5 |
| 450 | 1 | 4,8 |

Ces 2relations sont toujours être dans la troisième forme normale (quel que soit la clé primaire choisie) alors elle est dans la forme normale de boyce -codd

Alors les 2 relations sont en FNB

9-5 Les opérations sur les relations

Pour manipuler les données, le modèle rationnel dispose d'un ensemble d'opérateurs

Les opérateurs

a)-La Restriction (sélection)

La sélection permet le choix d'un certain de lignes en fonction d'un critère de sélection.

| tbl R1 | | | |
|---------|--------|-------------------|--------------------|
| Nom | Prénom | date de naissance | ville de naissance |
| DAOUDI | SAID | 31/07/77 | Rabat |
| DAHMANI | KHALID | 02/05/79 | Tétouan |
| BERADA | KHALID | 27/02/83 | Tanger |

| Restriction (Sélection) | | | |
|-------------------------|--------|-------------------|--------------------|
| Nom | Prénom | date de naissance | ville de naissance |
| DAHMANI | KHALID | 02/05/79 | Tétouan |
| BERADA | KHALID | 27/02/83 | Tanger |

b)-La projection

Elle permet le choix d'un sous-ensemble de colonnes

| tbl R1 | | | |
|---------|--------|-------------------|--------------------|
| Nom | Prénom | date de naissance | ville de naissance |
| DAOUDI | SAID | 31/07/77 | Rabat |
| DAHMANI | KHALID | 02/05/79 | Tétouan |
| BERADA | KHALID | 27/02/83 | Tanger |

| Projection | |
|------------|--------------------|
| nom | ville de naissance |
| DAOUDI | Rabat |
| DAHMANI | Tétouan |
| BERADA | Tanger |

c)-L'union

L'union de deux tables (obligatoirement de même structure) donne une table contenant les lignes de la première et de la deuxième.

| tbl R1 | | | |
|---------|--------|-------------------|--------------------|
| Nom | Prénom | date de naissance | ville de naissance |
| DAOUDI | SAID | 31/07/77 | Rabat |
| DAHMANI | KHALID | 02/05/79 | Tétouan |
| BERADA | KHALID | 27/02/83 | Tanger |

| tbl R2 | | | |
|---------|---------|-------------------|--------------------|
| Nom | Prénom | date de naissance | ville de naissance |
| DAHMANI | Khalid | 02/05/79 | Tétouan |
| DAHMANI | Najat | 01/01/82 | Tétouan |
| TOUSANI | youssef | 27/02/83 | Tanger |

L'union

| Nom | Prénom | date de naissance | ville de naissance |
|---------|---------|-------------------|--------------------|
| BERADA | KHALID | 27/02/83 | Tanger |
| DAHMANI | KHALID | 02/05/79 | Tétouan |
| DAHMANI | Najat | 01/01/82 | Tétouan |
| DAOUDI | SAID | 31/07/77 | Rabat |
| TOUSANI | youssef | 27/02/83 | Tanger |

d) -L'intersection

L'intersection de deux tables (obligatoirement de même structure) donne une table contenant les lignes qui sont à la fois dans la première et dans la deuxième.

| tbl R1 | | | |
|---------|--------|-------------------|--------------------|
| Nom | Prénom | date de naissance | ville de naissance |
| DAOUDI | SAID | 31/07/77 | Rabat |
| DAHMANI | KHALID | 02/05/79 | Tétouan |
| BERADA | KHALID | 27/02/83 | Tanger |

| tbl R2 | | | |
|---------|---------|-------------------|--------------------|
| Nom | Prénom | date de naissance | ville de naissance |
| DAHMANI | Khalid | 02/05/79 | Tétouan |
| DAHMANI | Najat | 01/01/82 | Tétouan |
| TOUSANI | youssef | 27/02/83 | Tanger |

L'intersection

| Nom | Prénom | date de naissance | ville de naissance |
|---------|--------|-------------------|--------------------|
| DAHMANI | KHALID | 02/05/79 | Tétouan |

e)- La différence

La différence entre deux tables (obligatoirement de même structure) donne une table contenant toutes les lignes de la première à l'exception de celles qui sont également dans la deuxième.

| tbl R1 | | | |
|---------|--------|-------------------|--------------------|
| Nom | Prénom | date de naissance | ville de naissance |
| DAOUDI | SAID | 31/07/77 | Rabat |
| DAHMANI | KHALID | 02/05/79 | Tétouan |
| BERADA | KHALID | 27/02/83 | Tanger |

| tbl R2 | | | |
|---------|---------|-------------------|--------------------|
| Nom | Prénom | date de naissance | ville de naissance |
| DAHMANI | Khalid | 02/05/79 | Tétouan |
| DAHMANI | Najat | 01/01/82 | Tétouan |
| TOUSANI | youssef | 27/02/83 | Tanger |

La différence

| Nom | Prénom | date de naissance | ville de naissance |
|--------|--------|-------------------|--------------------|
| DAOUDI | SAID | 31/07/77 | Rabat |
| BERADA | KHALID | 27/02/83 | Tanger |

g)- Produit cartésien

le produit cartésien entre deux tables permet de créer une troisième table à partir d'une concaténation des attributs qui composent les 2 relations .

| tbl R1 | | | |
|---------|--------|-------------------|--------------------|
| Nom | Prénom | date de naissance | ville de naissance |
| DAOUDI | SAID | 31/07/77 | Rabat |
| DAHMANI | KHALID | 02/05/79 | Tétouan |
| BERADA | KHALID | 27/02/83 | Tanger |

| tbl R5 | |
|------------|----------|
| professeur | Matière |
| Ahmed | français |
| Hassan | physique |

Produit cartésien :R1*R5

| Nom | Prénom | date de naissance | ville de naissance | professeur | Matière |
|---------|--------|-------------------|--------------------|------------|----------|
| DAOUDI | SAID | 31/07/77 | Rabat | Ahmed | français |
| DAOUDI | SAID | 31/07/77 | Rabat | Hassan | physique |
| DAHMANI | KHALID | 02/05/79 | Tétouan | Ahmed | français |
| DAHMANI | KHALID | 02/05/79 | Tétouan | Hassan | physique |
| BERADA | KHALID | 27/02/83 | Tanger | Ahmed | français |
| BERADA | KHALID | 27/02/83 | Tanger | Hassan | physique |

F) -La jointure

La jointure entre deux tables permet de créer une troisième table à partir du produit cartésien entre les 2 tables et d'une sélection sur un attribut commun.

| tbl R1 | | | |
|---------|--------|-------------------|--------------------|
| Nom | Prénom | date de naissance | ville de naissance |
| DAOUDI | SAID | 31/07/77 | Rabat |
| DAHMANI | KHALID | 02/05/79 | Tétouan |
| BERADA | KHALID | 27/02/83 | Tanger |

| tbl R9 | | |
|---------|----------|------|
| Nom | Matière | Note |
| DAOUDI | français | 10 |
| DAHMANI | français | 12 |
| DAHMANI | math | 10 |
| DAHMANI | math | 15 |
| TOUSANI | histoire | 11 |

La jointure

| tbl r1.Nom | Prénom | date de naissance | ville de naissance | tbl r9.Nom | Matière | Note |
|------------|--------|-------------------|--------------------|------------|----------|------|
| DAOUDI | SAID | 31/07/77 | Rabat | DAOUDI | français | 10 |
| DAHMANI | KHALID | 02/05/79 | Tétouan | DAHMANI | français | 12 |