

## **EPIGRAPHE**

« N'oubliez pas les merveilles que l'Éternel a faites. »

Psaume 105 : 5

## DEDICACES

Je dédie ce travail

A ma famille, elle qui m'a doté d'une éducation digne, son amour a fait de moi ce que je suis aujourd'hui :

Particulièrement à ma mère Pétronille MUKUNA, pour le goût à l'effort qu'il a suscité en moi, de par sa rigueur.

A vous maman Mimie, ceci est ma profonde gratitude pour ton éternel amour, que ce rapport soit le meilleur cadeau que je puisse t'offrir.

A vous mes frères Fortina MBALA, Chris KANANGILA et sœurs Sephorah NGALULA, Marthe NGALULA, Goretty WEBE qui m'avez toujours soutenu et encouragé durant ces années d'études.

## REMERCIEMENTS

Nous remercions l'Eternel Dieu le Tout Puissant pour sa grâce qu'il nous a accordé depuis le début de notre parcours universitaire en Sciences Informatiques et Intelligence Artificielle au sein de l'Université William Booth à

Kinshasa/R D Congo « UWB » en sigle.

Nous remercions aussi les autorités académiques et les membres du corps professoral pour leur encadrement et formation de qualité dont nous sommes bénéficiaires.

De façon particulière, notre gratitude va tout droit à la personne du

Professeur KUTANGILA MAYOYA David qui, en dépit de son horaire académique surchargé, a accepté de diriger ce travail. Sans les judicieuses remarques et nobles conseils de sa part, ce travail ne serait-ce qu'il est. Ces mêmes sentiments de gratitude s'adressent à la personne de MALANDA Anderson, l'encadrement de ce travail, pour les judicieuses remarques et nobles conseils.

Nous remercions tous les membres de ma famille et très particulièrement à ma mère MUKUNA Pétronille à cause de son amour qu'elle a partagé avec moi durant tout mon parcours académique en mettant en œuvre son propre moyen (financier, matériel et moral) pour l'épanouissement de ma connaissance en me conduisant au sein de l'Université William Booth pour me rendre utile dans la société.

A mes collègues et camarades qui ont apporté un plus à la réalisation de ce dernier : Sephorah NGALULA, Jemima BOTULI et Tatiana NTEMO.

Que tous ceux qui ont contribués de loin ou de près à notre épanouissement scientifique et progrès de ce travail dont leurs noms ne sont pas cités ci-haut trouvent ici l'expression de nos sincères grâtes.

Clarisse KALENGA ILUNGA

## 0. INTRODUCTION GENERALE

### 0.1. Généralité

Depuis des décennies environs, l'évolution du monde est le fait conjugué de certains événements dont le plus important qui est l'informatique représente un atout majeur pour les rénovations de la science. En particulier, il a permis de perfectionner les méthodes utilisées en automatisant autant de système de correspondance, sur laquelle l'homme à toujours rêvé d'améliorer son état du travail.

La nouvelle technologie de l'information et de la communication (NTIC) a une immense importance sur presque tous les aspects de notre vie. L'évolution rapide de cette technologie crée des occasions complètement nouvelles de parvenir à des niveaux de développement plus élevés. Leurs capacités à réduire bon nombre d'obstacles classiques, notamment ceux que constituent le temps et la distance permet pour la première fois dans l'histoire de faire bénéficier de leur potentiel.

Grace à l'automatisation et à l'agencement accompagnée par l'ordinateur. L'ordinateur est un outil important dans tous les domaines professionnels de la vie nécessitant une modélisation préalable (artisanats, architecte, industrie, etc....).

Pour ce faire toutes entreprises ont donc pour exhortation principale de gérer le système d'information de suivi en fin de récolte de données économiques et financières internes et externes, ...

L'homme est parvenu à mettre au centre une science compétitive que nous appelons « Informatique » science du traitement rationnel notamment par machines automatique, de l'information considérée comme les supports des connaissances humaines des communications dans les domaines techniques, économiques et sociaux.

L'importance de l'informatique pour ce qui concerne notre sujet est d'améliorer la qualité de services rendues en mettant en place un **système automatisé pour la gestion d'enregistrement des orphelins** à l'aide d'un outil que l'homme a mis au centre que nous appelons « Ordinateur » qui est une machine du traitement de l'information.

C'est dans ce cadre que les réalités dans notre domaine concernent l'automatisation des données recueillies par un outil informatique plus performant. Ce néologisme est composé de mots « informatique » et « automatique », nous désignons l'automatisation du traitement de l'information par un système concret (machine) ou abstrait.

### 0.2. Problématique

Nous devons parfois exposer de manière organisée nos idées, d'où la problématique est l'ensemble des questions anticipatives à une question du départ, en vue d'apporter les solutions éventuelles. Etant donné que notre investigation soit basée au sein d'un orphelinat sur le sujet dénommé « **MISE EN PLACE D'UNE APPLICATION INFORMATIQUE DE GESTION D'ENREGISTREMENT DES ORPHELINS** », et dans le but

de résoudre les différents problèmes auxquels font face les gestionnaires de l'orphelinat, nous nous sommes posés certaines questions comme suivent :

- ❖ Comment pourrait-on avoir un bon moyen de faire le suivi dans le cadre de la réinsertion familiale des enfants ?
- ❖ Est-il nécessaire de mettre en place un système pour enregistrer les orphelins ?
- ❖ Ce système servira-t-il à bien gérer les données et permettre la traçabilité des différentes origines des enfants ?

### 0.3. Hypothèses et intérêt du sujet

L'hypothèse est une proposition à partir de laquelle on réfléchit pour résoudre un problème, pour démontrer un théorème, au sens spécifique, le terme hypothèse est une proposition résultant d'une observation et que l'on soumet au contrôle de l'expérience ou que l'on vérifie par raisonnement, en effet tout chercheur doit en effet présupposer au départ un point de vue lequel constitue ce qu'on appelle le concept opérationnel ou hypothèse de travail. Dans ce même ordre d'idée, nous partirons de l'information et de la communication qui sont actuellement considérées comme le socle de la recherche.

Relatif à l'achèvement de notre sujet de travail de fin d'étude en sciences informatiques, la **gestion** a conservé notre concentration, ainsi le sujet de notre rayonnement scientifique se présente comme suit : « **Mise en place d'une application informatique de gestion d'enregistrement des orphelins** » cas de l'**Orphelinat notre dame de LIESSE**.

Pour mener à bon port nos recherches, nous nous sommes préconisés provisoirement une série des réponses envisageables à traiter tout au long de notre travail :

- ❖ La crédibilité d'accès à la l'information et cela à un temps record ;
- ❖ La rapidité dans le traitement et la bonne conservation de données ;

### 0.4. Délimitation du sujet

Pour être efficace, vu la complexité du travail qui se prescrit scientifique, il est avéré nécessaire de se fixer les idées en limitant dans le temps et dans l'espace pour une meilleur mise en œuvre.

La délimitation temporelle de notre travail se situe dans l'intervalle de l'année 2023 à nos jours.

Du point de vue spatial, notre sujet et porte principalement sur l'**orphelinat notre dame de Liesse** sise sur l'avenue Essence N°21 au Quartier KIMPE dans la Commune de Ngaliema dans la ville-province de Kinshasa.

### 0.5. Méthode et techniques utilisées

#### 0.5.1. Méthodes

Tout travail scientifique nécessite l'utilisation des procédés méthodologiques pour atteindre l'objectif visé.

Les méthodes choisies sont utilisées du temps et de moyens dont dispose le chercheur, pour ce fait nous avons opté pour les méthodes suivantes :

#### **0.5.1.1. Méthode historique**

La méthode historique désigne l'ensemble des réflexions qui portent sur les procédés, les moyens, les règles et les contextes des travaux des historiens, elle nous a aidés à la découverte des origines qui sont proche et lointaines pour la société l'**orphelinat notre dame de liesse**.

#### **0.5.1.2. Méthode descriptive**

La description consiste à déterminer la nature et les caractéristiques des phénomènes et parfois à établir des associations entre eux. Elle permet aussi de décrire l'organisation de l'**orphelinat notre dame de liesse**.

#### **0.5.1.3. 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 à effectuer en plusieurs modèles conceptuels et physiques. La séparation des données et des traitements assure une longévité au modèle. En effet, l'agencement des données n'a pas à être souvent remanié, tandis que les traitements le sont plus fréquemment.

La méthode MERISE date de 1978-1979, et fait suite à une consultation nationale lancée en 1977 par le ministère de l'Industrie dans le but de choisir des sociétés de conseil en informatique afin de définir une méthode de conception de systèmes d'information. Les deux principales sociétés ayant mis au point cette méthode sont le CTI (Centre Technique d'Informatique) chargé de gérer le projet, et le CETE (Centre d'Etudes Techniques de l'Equipement) implanté à Aix-en-Provence.

### **0.5.2. Techniques**

Les techniques sont de supports d'un travail scientifique qui viennent appuyer les méthodes et constituent de ce fait des outils efficaces dans la récolte des données. Ces sont des instruments, des procédés opératoires pour récolter les données

#### **0.5.2.1. Technique documentaire**

Cette technique nous permettra de consulter et d'explorer différentes ouvrages et supports tels que : les dictionnaires, les livres, les revues, les sites internet afin de s'imprégner des différentes informations en rapport avec notre travail.

### **0.5.2.1. Technique d'entretien**

Celle-ci consistera à avoir des entretiens en tête à tête avec les agents de l'orphelinat en vue de pouvoir conquérir d'avantage les informations sur l'apport des Technologies de l'information et de la communication dans la recherche documentaire dans une bibliothèque ou centre de recherche documentaire.

La méthode comme étant l'ensemble des opérations intellectuelles par lesquelles une discipline cherche à atteindre les vérités qu'elle poursuit, le démontre et les vérifie.

Ainsi, tout au long de notre travail, nous ferons recours à la méthode merise qui est une méthode est de conception et de réalisation de système d'information basée, elle possède un certain nombre de modèle ou (schéma).

### **0.6. Difficultés rencontrées**

Au cours de l'élaboration de ce travail, nous avons fait face à certaines difficultés, pendant lesquelles nous n'avons pas baissé les bras, mais avons trouvé une source de motivation pour bien nous préparer à ce qui nous attend dans le monde professionnel.

Parmi les difficultés rencontrés, l'accès aux ressources et données de l'orphelinat n'a pas été aussi facile à avoir, néanmoins avec insistance et explication claire, les agents ont été coopératifs.

Nous allons finir par une note optimiste, en essayent de me convaincre que pour tout problème il y a une solution. Petit à petit, à force de travailler et de persévérer, on peut obtenir des bons résultats

### **0.7. Subdivision du travail**

Outre l'introduction et la conclusion, notre travail comprend trois (3) parties dont chacune d'elles comporte des chapitres à savoir :

**Chapitre I : CONSIDERATION THEORIQUES SUR LES CONCEPTS DE BASE**

**Chapitre II : EVALUATION DU PROJET**

**Chapitre III : CADRAGE DU PROJET**

**Chapitre IV : CONCEPTION DU NOUVEAU SYSTEME**

**Section I : CONCEPTION DU SYSTEME D'INFORMATION ORGANISATIONNEL**

**Section II : CONCEPTION DU SYSTEME D'INFORMATION INFORMATISE**

**Chapitre V : MISE EN PLACE ET FONCTIONNEMENT DU SYSTEME**

# CHAPITRE I : CONSIDERATIONS THEORIQUES SUR LES CONCEPTS DE BASE

## I.1. Introduction

Ce présent chapitre montre les différents théories et concepts utilisés dans notre travail, les notions sur Réseau Informatique et Merise, d'éclaircir les mots et siglés abrégés, démontrer notre main-forte par rapport à ce qui a été fait par des prédécesseurs chercheurs en ce qui ayant trait à l'analyse de notre sujet.

### I.1.1. Définition opérationnelle des concepts clés utilisés

- ❖ Implémentation : Action d'implémenter<sup>1</sup>.
- ❖ Système d'information : C'est l'ensemble des moyens, humains et matériels et des méthodes se rapportant au traitement des différentes formes d'information rencontrées dans des organisations<sup>2</sup>.
- ❖ Gestion : Action ou manière de gérer, d'administrer, de diriger, d'organiser quelque chose<sup>3</sup>.
- ❖ Procédure : Manière de procéder, série de formalités qui doivent être remplies<sup>4</sup>.
- ❖ Octroi : Action d'octroyer.
- ❖ Document administratif : C'est toute information, sous quelque forme que ce soit, dont une administration dispose.
- ❖ Informatisation : mise en place d'un système de traitement automatique de l'information dans un service n'utilisant pas l'Informatique au préalable.
- ❖ Entité administrative : C'est espace dont les limites ont été imaginées ou choisies à la suite d'une décision humaine ;
- ❖ Conception : Idée, création de l'esprit ;
- ❖ Réalisation : Action de réaliser ;
- ❖ Système : Est un ensemble avec des sous-ensembles dont le but est commun afin d'atteindre des objectifs qu'elle s'est assignés<sup>5</sup>. Ces sous-ensembles constituent eux aussi des systèmes dans un système. Il s'agit de :
- ❖ Système de Pilotage (S.P ou décisionnel) : a pour objet d'arrêter les stratégies pour le bon fonctionnement de l'entreprise. Il est appelé autrement système décisionnel, car il décide du sort de l'entreprise à court, moyen et long terme ;
- ❖ Système Opérant (S.O ou d'exécution) : a pour objet d'exécuter les ordres provenant du système décisionnel via le système d'information et d'en faire un rapport après exécution ;
- ❖ Système d'information (S.I) : joue le rôle de la courroie de transmission entre le système de pilotage et le système opérant. Il est un ensemble d'informations et de moyens utilisés pour exploiter ses informations. Il s'agit des moyens : humains, matériels, logiciels, financiers, ...

<sup>1</sup> 36 Dictionnaires et Recueils

<sup>2</sup> Prof. Dr. Ir. Jp. MBIKAYI MPANYA, Cours de Méthode d'Analyse en Informatique, URKIM/G3 Informatique de gestion, 2012-2013, inédit

<sup>3</sup> Dictionnaire Encyclopédique Universel, éd. 1993, Québec, 2002, p. 692

<sup>4</sup> Dictionnaire Le Robert, éd. 2006, Paris, p. 1017

<sup>5</sup> J.A. MVIBUDULU, L.D. KONKFIE, Technique des bases de données Etude et Cas : 1<sup>ère</sup> Edition, Kinshasa, CRIGED, janvier 2010, p.4

## I.1.2. Etat de la question

Produit documentaire établissant le bilan critique des travaux effectués sur un sujet donné pendant une période déterminée et pouvant se présenter sous forme écrite ou orale<sup>6</sup>. Il définit le sujet dans le temps et dans l'espace, en précisant les acteurs et les différents aspects (politiques, économiques, juridiques, etc.), les sources et ressources d'information.

Le sujet traité dans ce travail est l'implémentation d'un système d'information pour la gestion des enregistrements des enfants dans un orphelinat. Nous n'avons pas pu retrouver un sujet similaire au notre parmi tant de travaux que nous avons lu. Mais néanmoins, nous ne disons pas ou croyons que nous sommes les premiers à présenter un tel sujet. Il se peut que ce présent sujet ait déjà fait l'objet de plusieurs recherches sous d'autres cieux.

Qu'en cela ne tienne, nous ne pouvons présenter que notre problématique et hypothèse.

Notre problématique est constituée des questions telles que :

- ✓ Comment gérer toutes les demandes d'enregistrement des enfants et la gestion d'intégration dans des familles d'accueil ?
- ✓ Quel mécanisme mettre en place pouvant faciliter le suivi des documents délivrés ainsi que ceux en cours de délivrance ?
- ✓ Comment produire les rapports des arrivées et des départs des enfants ?
- ✓ Quelle architecture de gestion mettre en place pour permettre l'accès à une catégorie des utilisateurs choisis dans le cadre de la gestion dudit système ?

Quant à l'hypothèse, nous avons soutenu ce qui suit :

- ✓ L'informatisation de procédure d'enregistrement des enfants dans cette structure serait un moyen idéal pour assurer une gestion fiable et rapide ;
- ✓ Pour une bonne gestion, développer et déployer une application sur mesure dans le système ;
- ✓ De mettre en place un système d'authentification pour les utilisateurs de l'application.

## I.2. Notions sur la méthode MERISE

### I.2.1. PRINCIPES GENERAUX ET FONDEMENTS DE LA METHODE MERISE

MERISE est une méthode d'analyse destinée à concevoir et à développer les systèmes d'information. Elle conduit à recenser et à décrire toutes les informations nécessaires au bon fonctionnement de l'entreprise ; que ces informations soient utilisées manuellement ou par des machines<sup>7</sup>.

Cette description de l'information est instantanée (elle concerne les informations actuellement traitées), mais elle se veut également prospective dans la mesure où les informations qui ne deviendront pertinentes que quelques années plus tard doivent être recherchées et recensées. Elle est aussi objective en permettant la confrontation des points de vue de différents acteurs.

Que signifie bien MERISE ? Rien d'officiel. Nous donnons ici quelques définitions de cet acronyme :

- ❖ Méthode pour Rassembler les Idées Sans effort (Hubert Tardieu).
- ❖ Méthode d'Etude et de Réalisation Informatique pour les Systèmes d'Entreprise.

La méthode MERISE a puisé ses fondements théoriques dans plusieurs domaines :

<sup>6</sup> Dictionnaire.sensagent.com/état%20de%20la%20question/fr-fr consulté le 16/04/2015 à 19h30'

<sup>7</sup> Prof. Dr. Ir. J. MBIKAYI, Modéliser un système d'information avec l'approche systémique Tome 1 : MERISE, 2<sup>ème</sup> génération, p.21-32

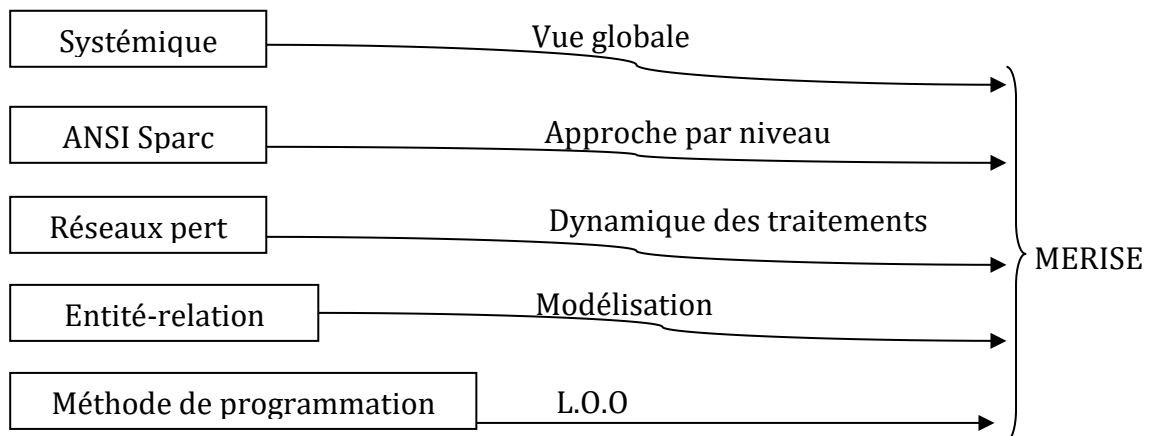


Figure 1 : Schéma des principes généraux et fondements de la méthode MERISE

Les principes fondamentaux de la méthode MERISE reposent sur :

- ✓ Apport de la systémique (organisation en tant que système) ;
- ✓ Découpage de l'organisation en domaine ;
- ✓ Analyse indépendante donnée-traitement ;

Une démarche à trois dimensions :

- ✓ La démarche : cycle de vie ;
- ✓ Le raisonnement : niveaux d'abstraction ;
- ✓ La maîtrise : niveau de décision.

## 1.2.2. LES TROIS COMPOSANTES DE LA METHODE MERISE

### 1.2.2.1. La démarche ou cycle de vie

- ✓ **Etude préalable** où l'on définit les propositions et l'évaluation des solutions d'organisation et technique pour le système d'information d'un domaine.
- ✓ **Etude détaillée** pour la spécification complète du futur (SIO) en tenant compte des points de vue de l'utilisateur.
- ✓ **Etude technique** pour la spécification complète du futur (SII) du point de vue réalisation.
- ✓ **Production logicielle** où l'on doit écrire le programme, la génération des fichiers de la base de données et les tests.
- ✓ **Mise en service (en œuvre)** qui correspond à l'installation de l'application informatique produite et la mise en place de la nouvelle organisation du travail.
- ✓ **Maintenance** pour rectifier les éventuelles anomalies, améliorer le système mis en place, participer à son évolution.

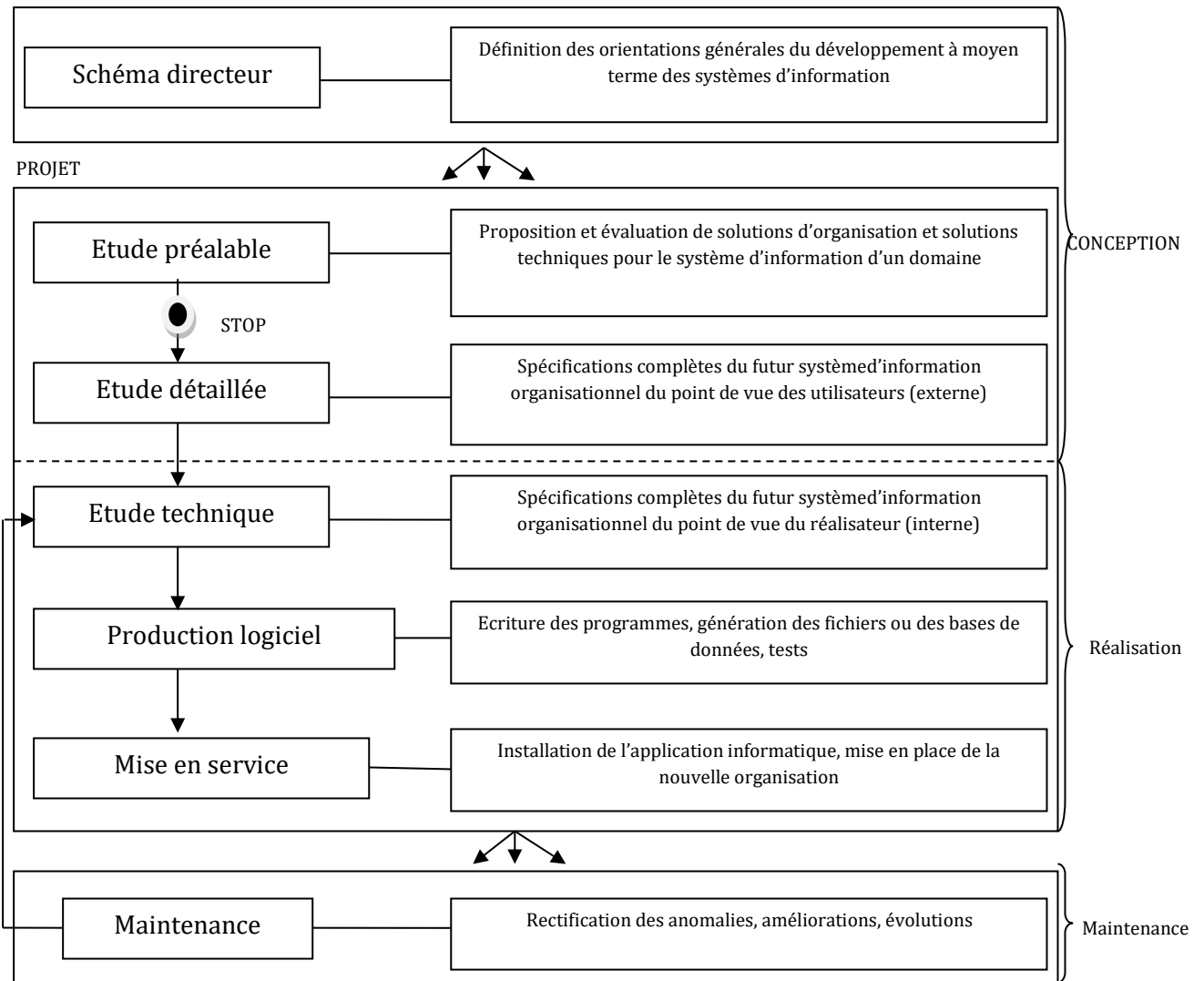


Figure 2 : Schéma de la démarche ou cycle de vie de la méthode MERISE.

#### I.2.2.1.1. Le schéma directeur

Première étape de la conception, le schéma directeur définit le cadre général du développement des systèmes d'information, principalement en termes d'objectifs et de contraintes. Il détermine, pour les systèmes d'information :

- ✓ Le découpage en domaine ;
- ✓ Les orientations de l'informatisation ;
- ✓ Les axes organisationnels ;
- ✓ Les options socioprofessionnelles ;
- ✓ La planification globale du développement
- ✓ Les cadres budgétaires.

#### I.2.2.1.2. L'étude préalable

Dans la ligne du schéma directeur, l'étude préalable constitue une étape fondamentale de MERISE. Elle permet, avant de se lancer à fond dans un projet, d'élaborer globalement différentes solutions et d'en évaluer les diverses conséquences. Cette étape est confrontée à deux exigences contradictoires :

- ✓ Une durée relativement courte (quelques mois ou plus) ;
- ✓ Une analyse suffisamment complète pour permettre une évaluation raisonnable.

Elle portera en conséquence sur un sous-ensemble représentatif du domaine étudié. L'étude préalable permet de proposer des solutions en précisant pour chacune :

- ✓ Le processus de fonctionnement du domaine ;
- ✓ Le degré et le type d'automatisation ;
- ✓ La perception des informations ;
- ✓ Le coût des moyens à mettre en œuvre (informatique en particulier) ;
- ✓ Les détails et étapes transitoires ;
- ✓ Les avantages et contraintes de la solution ;
- ✓ La situation par rapport au schéma directeur.

#### **1.2.2.1.3. L'étude détaillée**

Cette étude permet, à partir des choix issus de l'étude préalable, de spécifier complètement le futur système d'information. Cette conception comporte deux phases :

- ✓ La conception générale, dont l'objet est d'étendre à l'ensemble du domaine les principes du fonctionnement retenus pour le sous-ensemble représentatif. Les différentes spécifications sont complétées et validées ;
- ✓ La conception détaillée, qui produit, au niveau de chacune des tâches à automatiser. Une description complète en termes de support (dessin, écran, imprimé), d'algorithme (règle de calcul, de contrôle, etc.), d'action sur les données (mis à jour, consultation).

Elle permet d'obtenir pour l'utilisateur, une description complète et contractuelle du futur système d'information organisationnel. Elle permet également de réajuster les évaluations de moyens, coûts et délais estimés lors de l'étude préalable.

#### **1.2.2.1.4. L'étude technique**

Elle est la traduction informatique de spécifications issues de l'étude détaillée permet de déterminer :

- ✓ La structure informatique de la base de données ;
- ✓ L'architecture des programmes (transactionnels et batch) ;
- ✓ La structure de chaque programme et des accès aux données.

La position de cette étape est souvent ambiguë. Au demeurant cette étape peut être considérée comme la partie informatique de l'étude détaillée. Toutefois, son aspect fortement technique la rend très proche de la réalisation et l'assimile à la spécification de celle-ci.

#### **1.2.2.1.5. La production du logiciel**

Elle consiste à traduire, dans des langages appropriés, les spécifications exprimées dans les étapes précédentes. Cette production comprendra, entre autres :

- ✓ L'écriture des programmes dans un langage de programmation ;
- ✓ La génération des fichiers ou des bases de données ;
- ✓ Les tests de mise au point.

A l'issue de cette étape, une recette du logiciel est effectuée, prononçant la conformité aux spécifications.

#### **I.2.2.1.6. La mise en service**

Elle consiste à installer les logiciels réalisés et à mettre progressivement l'ensemble du système d'information au service des utilisateurs. Au cours de cette étape, on procède à :

- ✓ La mise au point d'un planning d'installation tenant compte des phases transitoires ;
- ✓ La création et le chargement des informations de base ;
- ✓ La formation des utilisateurs ; la vérification du bon fonctionnement du logiciel ;
- ✓ La mise en place progressive de la nouvelle organisation.

A l'issue de cette période de lancement, on pourra procéder, en fonction des événements, à la recette provisoire d'abord, définitive du système d'information ensuite.

#### **I.2.2.1.7. La maintenance**

Elle consiste à prendre en compte les évolutions apparaissant après le lancement opérationnel. Il faut, en fait, distinguer une étape supplémentaire, antérieure à la maintenance : le fonctionnement opérationnel.

La maintenance qui demeure la plus importante de la vie d'un projet ne devrait pas se manifester autrement que par des tâches d'exploitation. Les évolutions conduisant à une modification de l'application initiale proviennent des progrès technologiques, de la modification de l'environnement et des utilisateurs.

Cette maintenance se traduit par un « débouclage » du cycle de vie :

- ✓ L'étude de l'impact de la modification ;
- ✓ La spécification des modifications à effectuer ;
- ✓ La réalisation ;
- ✓ La mise en service.

### **I.2.2.2. Les raisonnements ou cycle d'abstraction**

Il existe quatre niveaux d'abstraction en MERISE deuxième génération. Le découpage en niveaux a été confirmé par la communauté internationale (ANSI-X3-SPARC 75) :

Le SIO renferme les deux premiers niveaux :

- ✓ Niveau conceptuel (exprimant les choix fondamentaux de gestion): on doit rechercher des éléments stables indépendamment des moyens à mettre en œuvre, de leurs contraintes et de leur organisation ;
- ✓ Niveau organisationnel exprimant les choix d'organisation des ressources humaines et matérielles au travers de la définition des sites, des postes de travail.

Le SII renferme, quant à lui, les deux derniers niveaux :

- ✓ Niveau logique exprimant le choix des moyens et des ressources informatiques en faisant abstraction de leurs caractéristiques techniques précises ;

✓ Niveau physique : traduisant les choix techniques et la prise en compte de leurs spécificités.

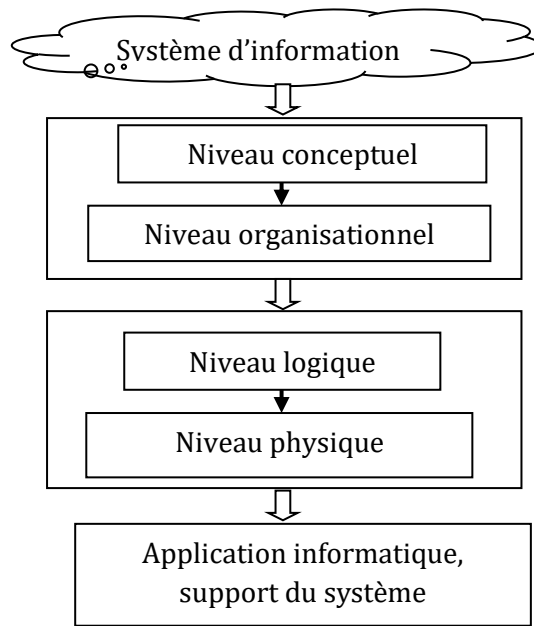


Figure 3 : Schéma de cycle d'abstraction

Ce cycle d'abstraction avec quatre niveaux conduit à l'obtention de huit modèles dont chacune concerne les données et les quatre autres les traitements.

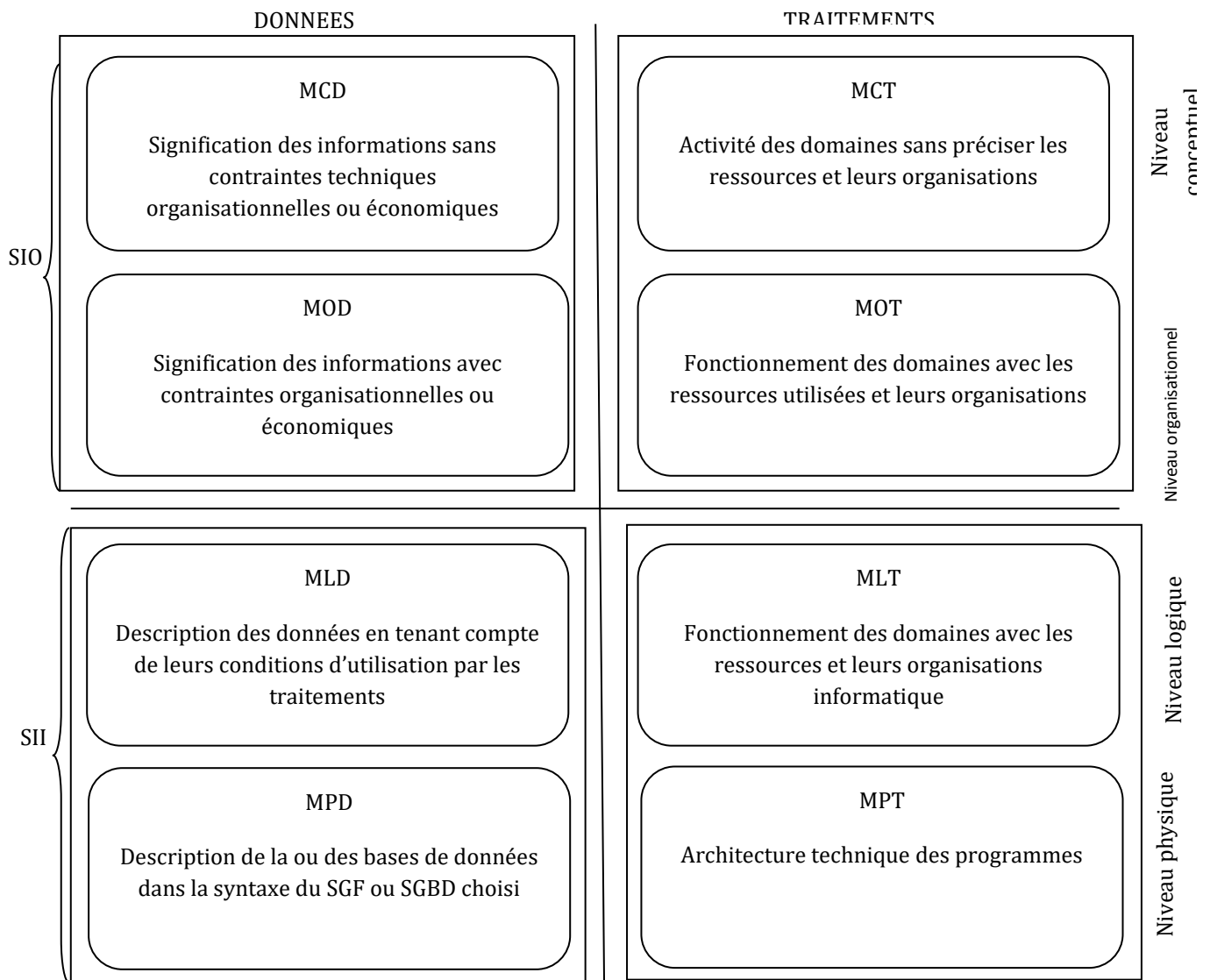


Figure 4 : Schéma de cycle d'abstraction avec quatre niveaux

N.B : le SIO prend en compte la préoccupation des gestionnaires et des utilisateurs tandis que le SII celles des informaticiens.

### I.2.2.3. La maîtrise du projet ou cycle de décision

Le déroulement simultané de la démarche et des raisonnements doit être maîtrisé. Dans chaque modèle, à chaque étape, des choix doivent être effectués. Vers quel projet veut-on aller ? Quels moyens veut-on lui affecter ?

La mise en œuvre de la méthode MERISE se traduit en outre, par une succession de choix permettant, d'une part, de contrôler la durée globale de la conception/réalisation du projet, d'autre part, de définir un système en harmonie avec les objectifs généraux de l'entreprise.

Ce cycle concerne donc les différents choix qui sont effectués tout au long du cycle de vie de la méthode. La plupart de ces décisions marquent la fin d'une étape et le début d'une autre. Pour chaque étape, MERISE prévoit une action :

- ✓ Préparer ;
- ✓ Exécuter ;
- ✓ Contrôler.

Dans la pratique, le cycle de décision est intégré dans le cycle de vie. Cela se traduit par des résultats types à l'issue de chaque étape et par des décisions attendues, comme le montre la figure suivante<sup>8</sup> :

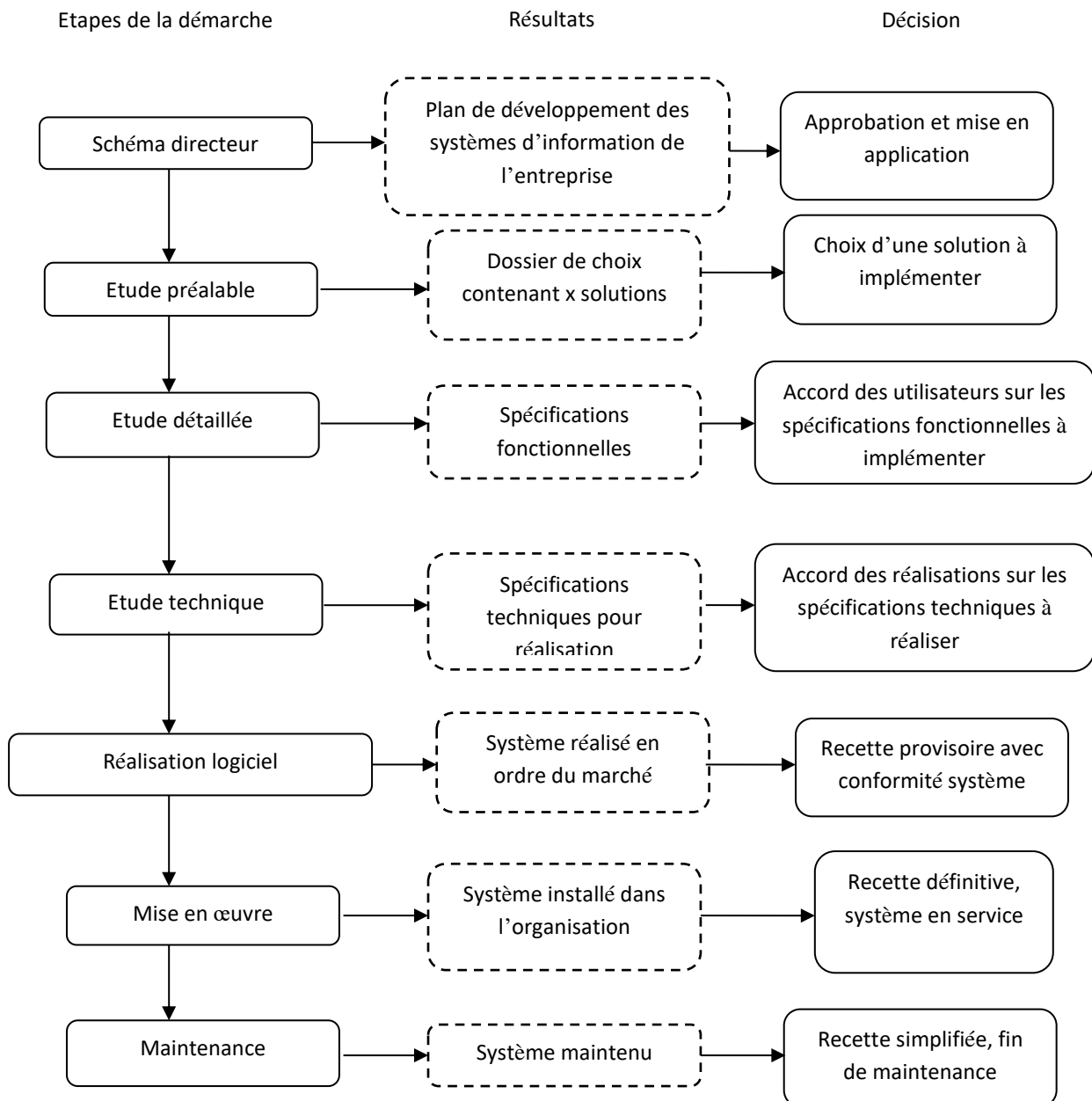


Figure 5 : Schéma de cycle d'abstraction avec quatre niveaux

<sup>8</sup> J.A. MVIBUDULU, L.D. KONKIE, *Technique des bases de données Etude et Cas* : 1<sup>ère</sup> Edition, Kinshasa, CRIGED, janvier 2010, p. 21-31

## I.4. Notions sur les bases des données

### I.4.1. Définition

#### a. Une base de données

Une base de données est un ensemble d'informations de même nature, organisées dans un ou plusieurs endroits selon certains critères en vue de permettre leur exploitation.

- On les appelle "BDD" par commodité : les bases de données ont pour but de stocker, organiser et analyser les données.
- Elles désignent une collection d'informations organisées pour être facilement consultables, gérables et mises à jour.
- Ainsi, au sein d'une base des données, les données sont soumises à une organisation rigoureuse : en ligne, colonnes et tableaux.
- Elles sont indexées pour être trouvées rapidement via un logiciel informatique, et mises à jour ou supprimées à chaque fois que de nouvelles informations sont ajoutées.

#### b. Une base de données informatisée

Une base de données informatisée est un ensemble structuré de données enregistrées sur des supports accessibles par l'ordinateur, représentant des informations du monde réel et pouvant être interrogées et mises à jour par une communauté d'utilisateurs.

### I.4.2. Caractéristiques d'une base de données

Une base de données est caractérisée par ce qui suit :

- **L'intégrité** : C'est La cohérence, la fiabilité, et la pertinence des données qu'elle contient.
- **La sécurité** : La base de données doit être restaurée après une panne qui soit d'origine logicielle ou matérielle par le Système de Gestion de Base de Données (SGBD).
- **La concurrence** : La base de données doit être cohérente lorsque plusieurs transactions se font et lorsque plusieurs utilisateurs se partagent la même base de données.
- **La confidentialité** : Le Système Gestion Base de Données doit permettre d'interdire à certaines personnes de réaliser certaines opérations sur une partie ou toute la base de données.
- **Le non redondance** : Les incohérences provoquées par la redondance d'information représentent le principal souci du concepteur d'une base de données. En effet, lorsque les données sont dupliquées, aucun mécanisme ne peut garantir que le changement de la valeur d'une donnée est répercuté correctement sur les autres données.
- **Le non exhaustif** : La base de données doit contenir des informations qui ne se répètent pas c.à.d. chaque donnée doit être présente qu'une seule fois dans la base

### I.4.3. Utilité d'une base de données

Une base de données permet de mettre des informations à disposition d'utilisateurs pour qu'ils puissent les consulter, les modifier, les mettre à jour etc...

Elles sont largement utilisées en entreprises, dans de nombreuses entreprises et à de nombreuses fins : gestion des réservations pour les compagnies aériennes, enregistrements médicaux dans les hôpitaux, enregistrements légaux dans les compagnies d'assurance, services publics, ...

### I.4.4. Avantages d'une base de données

La base de données présente des avantages ci - après :

- ❖ L'indépendance entre données et traitements ;
- ❖ La duplication des données est réduite ;
- ❖ La base de données dote l'entreprise d'un contrôle centralisé de données opérationnelles qui représentent d'après H.S. MELZER le capital important de l'entreprise ;
- ❖ L'ordre dans le stockage de données ;
- ❖ L'utilisation simultanée des données par différents utilisateurs.

### I.4.5. Types de base de données<sup>9</sup>

On peut classer les bases de données selon différentes approches : en fonction de leur contenu (bibliographique, texte, chiffres, images...) ou selon leur approche organisationnelle, qui est souvent celle retenue en informatique.

- ❖ **La BDD hiérarchique** : Souvent présentée sous forme d'arbre avec ses ramifications, il s'agit du tout premier programme qui a permis de structurer l'information de façon hiérarchique. Ici, chaque enregistrement dépend d'un seul enregistrement, et chaque niveau d'enregistrement découle sur un ensemble de catégories plus petites.
- ❖ **La BDD réseau** : Dans ce cas, contrairement à la BDD hiérarchique, un objet peut avoir plusieurs objets parents et plusieurs objets enfants, ce qui permet de s'approcher du monde réel plus fidèlement. Des liens multiples sont ainsi créés entre les ensembles, permettant une vitesse et une polyvalence qui ont permis leur adoption massive.
- ❖ **La BDD SQL ou relationnelle** : Les bases de données relationnelles sont un type de base de données qui stockent et organisent des points de données avec des relations définies pour un accès rapide. Avec une base de données relationnelle, les données sont organisées en tables qui contiennent des informations sur chaque entité et représentent des catégories prédéfinies par le biais de lignes et de colonnes. La structure des données de cette façon rend l'accès efficace et flexible, c'est

---

<sup>9</sup> <https://www.free-work.com/fr/tech-it/blog/actualites-informatiques/bases-de-donnees-definition-fonctionnement-et-typologie>

pourquoi les bases de données relationnelles sont les plus courantes. Les bases de données relationnelles sont également conçues pour comprendre langage SQL (SQL), un langage de programmation standardisé qui est utilisé pour stocker, manipuler et récupérer des données. Dans SQL, il existe un langage intégré pour la création de tables appelées DDL (Data Definition Language) et un langage de manipulation de données appelé DML (Data Manipulation Language).

Relationnel désigne l'indication ou la création d'une relation. Dans le contexte des bases de données, la façon dont nous définissons les données relationnelles s'applique principalement aux données. Les jeux de données relationnels ont des relations prédéfinies entre eux. Par exemple, une base de données qui inclut des informations client pour une entreprise peut également inclure des données de transaction individuelles jointes à chaque compte. Les bases de données relationnelles concentrent l'attention sur la relation entre les éléments de données stockés.

- ❖ **La BDD orientée objet** : Cette typologie de base de données est encore en cours d'élaboration. Elle est focalisée sur la base de données des objets en tant que concept de programmation qui va permettre de simplifier la création de logiciels.
- ❖ **La BDD orientée texte** : La "flat file database" se présente sous la forme d'un fichier .txt ou .ini, qui est soit un fichier texte, soit un fichier combinant du texte avec un fichier binaire. Chaque ligne ne comporte généralement qu'un seul enregistrement.
- ❖ **La BDD distribuée** : Ce type de base de données présente des portions stockées au sein de différents endroits physiques, avec un traitement réparti ou répliqué entre différents points d'un réseau. Elle peut être homogène ou hétérogène : soit tous les emplacements physiques fonctionnent avec le même hardware et tournent sous le même système d'exploitation et les mêmes applications de bases de données, soit ils varient entre différents endroits physiques.
- ❖ **La BDD cloud** : Optimisée ou directement créée pour les environnements virtualisés, elle peut être relative à un cloud privé, un cloud public ou un cloud hybride. Ses avantages sont multiples : paiement pour la capacité de stockage et la bande passante en fonction de l'usage, changement d'échelle sur demande, disponibilité plus élevée...
- ❖ **La BDD NoSQL** : Poussées par l'essor du Big Data, elles sont utiles pour les larges ensembles de données distribuées, et parfaites pour analyser des quantités importantes de données non structurées, ou stockées sur plusieurs serveurs cloud virtuels.
- ❖ **La BDD orientée graph** : C'est un type de database NoSQL qui utilise la théorie des graphes pour stocker, cartographier et effectuer des requêtes sur les relations entre les données.

## I.4.6. Conception d'une base de données

### A. Modèle conceptuel de données (MCD)

Le modèle conceptuel des données (MCD) a pour but d'écrire de façon formelle les données qui seront utilisées par le système d'information. Il s'agit donc d'une représentation des données, facilement compréhensible, permettant de décrire le système d'information à l'aide d'entités<sup>10</sup>.

### B. Modèle logique de données (MLD)

Le modèle logique des données consiste à décrire la structure de données utilisée sans faire référence à un langage de programmation. Il s'agit donc de préciser le type de données utilisées lors des traitements<sup>11</sup>.

Ainsi, le modèle logique est dépendant du type de base de données utilisé.

### C. Modèle Physique de données (MPD)

Cette étape consiste à implémenter le modèle dans le SGBD, c'est-à-dire le traduire dans un langage de définition de données<sup>12</sup>.

Le langage généralement utilisé pour ce type d'opération est le SQL, et plus spécialement le langage de définition de données du SQL.

## I.2. SYSTÈME DE GESTION DE BASE DE DONNÉES

### I.2.1. Définition

Un système de gestion de base de données (SGBD) est un logiciel conçu pour stocker, récupérer et gérer des données. Le SGBD le plus répandu dans un système de base de données d'entreprise est le RDBMS. La forme complète de RDBMS est le système de gestion de base de données relationnelle. Bases de données c'est-à-dire :

- ❖ Permettre l'accès aux données de façon simple ;
- ❖ Autoriser un accès aux informations à de multiples utilisateurs ;
- ❖ Manipuler les données présentes dans la base de données (insertion, suppression, modification, ...).

### I.2.2. Objectif

Des objectifs principaux ont été fixés aux SGBD dès l'origine de ceux-ci et ce, afin de résoudre les problèmes causés par la démarche classique.

Ces objectifs sont ci-après :

---

<sup>10</sup> <http://www.commentcamarche.net/contents/659-merise-modele-conceptuel-des-donnees>

<sup>11</sup> <http://www.commentcamarche.net/contents/661-merise-modele-logique-des-donnees>

<sup>12</sup> <http://www.commentcamarche.net/contents/663-merise-modele-physique-des-donnees>

- ❖ Indépendance physique : La façon dont les données sont définies doit être indépendante des structures de stockage utilisées.
- ❖ Indépendance logique : Un même ensemble de données peut être vu différemment par des utilisateurs différents.
- ❖ Accès aux données : Se fait par intermédiaire d'un langage de manipulation de données (LMD).
- ❖ Administration centralisée de données (intégration) : toutes les données doivent être centralisées dans un réservoir unique commun à toutes les applications.
- ❖ Non redondance : Chaque donnée doit être présente qu'une seule fois dans la base.
- ❖ Cohérence de données : Les données sont soumises à un certain nombre de contraintes d'intégrité qui définissent un état cohérent de la base.
- ❖ Partage de données : Il s'agit de permettre à plusieurs utilisateurs d'accéder aux données au même moment de manière transparente.
- ❖ Sécurité des données : Les données doivent pouvoir être protégées contre les accès non autorisés.
- ❖ Résistance aux pannes.

### I.2.3. Principe de fonctionnement

#### Administrateur de bases de données

- ❖ Analyse fonctionnelle ;
- ❖ Définition du schéma conceptuel ;
- ❖ Choix des méthodes de stockage et d'accès ;
- ❖ Modification de l'organisation
- ❖ Conceptuelle / physique
- ❖ Gestion des droits d'accès ;
- ❖ Des droits d'accès ;
- ❖ Spécification des contraintes d'intégrité.

#### Classes d'utilisateurs

- ❖ Utilisateurs occasionnels (programmes d'application) ;
- ❖ Utilisateurs experts (SQL) ;
- ❖ Programmeurs d'applications (SQL + langage de programmation "hôte").

### I.2.4. TYPE DE SYSTEME DE GESTION DE BASE DE DONNEES

Tout SGBD est basé sur un **modèle de données**, constitué de :

- Une façon de **structurer** les données
- Des **opérations** pour agir sur les données

Au fil de l'évolution de la technologie et des besoins pour des bases de données, différents modèles de données ont vu le jour. Les modèles présentés dans le cadre du cours sont quelques-uns des principaux modèles que l'on retrouve en contexte documentaire, mais d'autres existent. Nous pouvons citer plus spécifiquement :

- ❖ Le *modèle textuel* (fichier plat)
- ❖ Le *modèle relationnel*
- ❖ Les *familles NoSQL* (pour information)
- ❖ Le SGBD hiérarchique ;

- ❖ Le SGBD réseau ;
- ❖ Le SGBD objet ;
- ❖ Etc.

## 1. Système de Gestion de Base de Données hiérarchique

Les modèles de bases de données hiérarchiques et en réseau constituent les deux premières approches pour organiser et données structurantes, mais ils diffèrent considérablement par leurs architectures et leurs capacités.

Dans un modèle de base de données hiérarchique, les données sont organisées dans une structure arborescente, dans laquelle chaque enregistrement est lié à un enregistrement parent, qui peut avoir plusieurs enregistrements enfants. Cette relation rigide parent-enfant restreint la flexibilité de la représentation des données, car chaque enregistrement enfant ne peut avoir qu'un seul parent.

À l'inverse, dans un modèle de base de données en réseau, les données sont organisées de manière plus flex structure de type graphique, dans laquelle les enregistrements peuvent avoir plusieurs connexions avec d'autres enregistrements, formant des relations complexes. Cette approche permet de multiples relations entre les enregistrements, permettant une représentation plus riche et plus polyvalente des données. (Voir figure suivante)

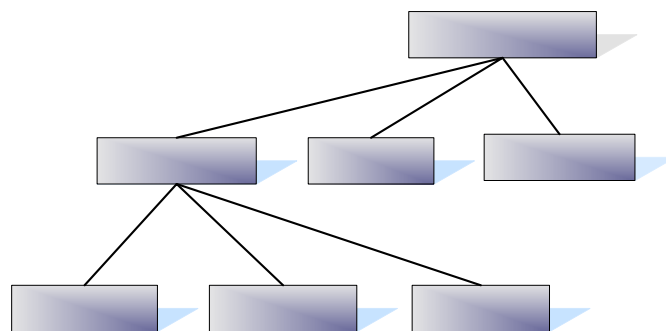


Figure 6 : Exemple de SGBD hiérarchique

## 2. Système de Gestion de Base de Données Réseau<sup>13</sup>

Le modèle réseau est une manière de représenter les données dans le cadre d'une base de données. Ce modèle est en mesure de lever de nombreuses difficultés du modèle hiérarchique grâce à la possibilité d'établir des liaisons de type n-m en définissant des associations entre tous les types d'enregistrements.

Ce modèle est une extension du modèle précédent (hiérarchique), les liens entre objets peuvent exister sans restriction. Pour retrouver une donnée dans une telle modélisation,

<sup>13</sup> [https://fr.wikipedia.org/wiki/Base\\_de\\_donn%C3%A9es\\_r%C3%A9seau](https://fr.wikipedia.org/wiki/Base_de_donn%C3%A9es_r%C3%A9seau)

il faut connaître le chemin d'accès (les liens), ceci rend encore les programmes dépendants de la structure de données.

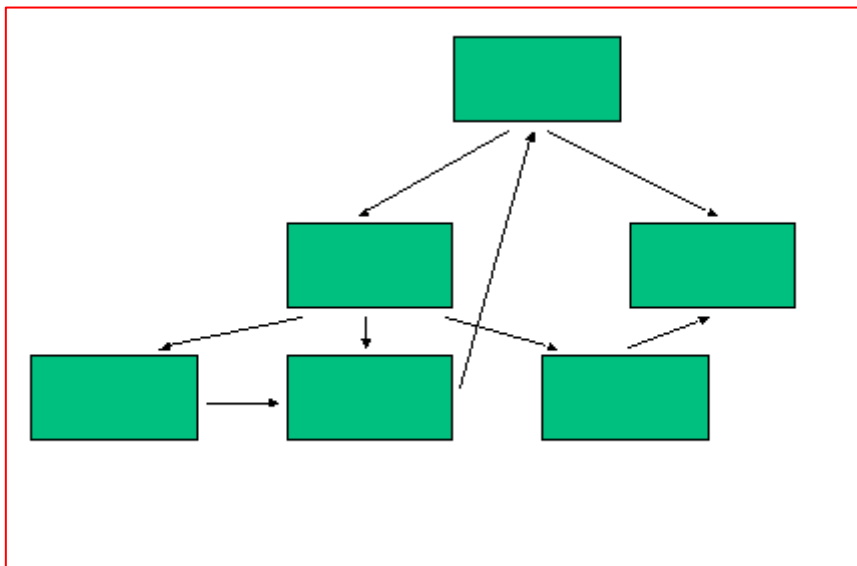


Figure 7 : Base de données réseau

### 3. Système de Gestion de Base de Données Relationnel

Une base de données relationnelle est un ensemble d'informations qui organise les données dans des relations prédéfinies et les stockent dans une ou plusieurs tables (ou "relations") de colonnes et de lignes, ce qui permet de les consulter et de comprendre les relations entre différentes structures de données. Les relations sont des connexions logiques entre différentes tables qui sont établies en fonction de l'interaction entre ces tables.

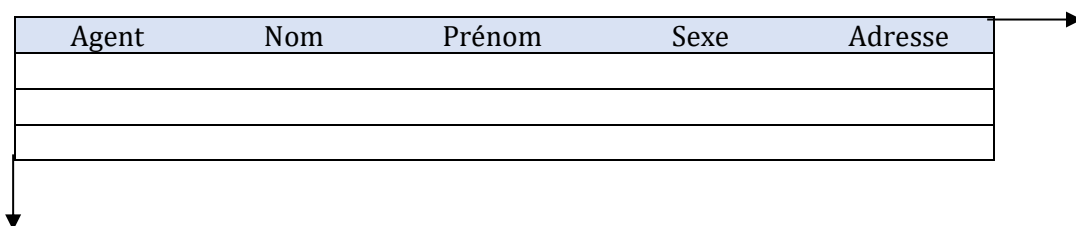


Figure 8 : Exemple de SGBD Relationnel

### 1.3. SYSTEME DE GESTION DE BASE DE DONNEES REPARTIS (SGBDR)

#### 1.3.1. DEFINITION

Un Système Gestion de base de données Répartis (SGBDR) est un logiciel qui se charge de la création et de la maintenance des bases de données réparties. Le Un Système Gestion de base de données Répartis (SGBDR) repose sur un système réparti qui est constitué d'un ensemble de processeurs autonomes appelés sites (micro-ordinateurs, stations de travail, ... etc.) reliés par un réseau de communication qui leur permet d'échanger des données.

Un Système Gestion de base de données Repartis (SGBDR) est constitué de composantes suivantes :

- ❖ La composante de gestion de base de données
- ❖ La composante de communication de données
- ❖ Le dictionnaire de données qui peut représenter l'information à propos de la répartition des données dans le réseau informatique.
- ❖ La composante de base de données répartie qui contient le logiciel qui s'occupe du parallélisme, de l'exécution mutuelle et de la synchronisation.

### **1.3.2. FONCTIONNEMENT**

L'accès lointain par un programme d'application rendu possible grâce à la composante de base de données répartie.

- ❖ Garantir certains degrés de transparences réparties.
- ❖ Le support d'administration et de contrôle de base de données. Le contrôle, l'utilisation des bases de données et la vue globale des fichiers existants dans les divers sites.
- ❖ Le contrôle de concurrence et la rentabilité des transactions réparties.

## Conclusion partielle

En somme, la méthode Merise constitue un cadre méthodologique puissant pour la conception et la modélisation des systèmes d'information, en particulier les bases de données. À travers ses différentes phases, de l'étude conceptuelle à la mise en œuvre, Merise permet d'aborder avec rigueur la structuration des données et des processus.

Les notions de bases de données, quant à elles, forment le socle technique indispensable pour assurer la gestion efficace et sécurisée de l'information au sein d'une organisation. Des concepts clés comme les modèles entité-association (MCD), les modèles relationnels et la normalisation nous offrent une vision claire de la manière dont les données doivent être structurées pour éviter la redondance et garantir leur intégrité.

Maîtriser ces concepts fondamentaux ouvre la voie à une meilleure compréhension des systèmes d'information modernes, et à une gestion optimisée des données, qui sont aujourd'hui au cœur de toute stratégie d'entreprise. Dans les chapitres suivants, nous approfondirons ces bases en explorant les aspects plus avancés de la conception et de l'optimisation des bases de données, ainsi que leur mise en œuvre pratique.

## CHAPITRE II : EVALUATION DU PROJET

### II.1. Introduction

L'évaluation est un processus permet de déterminer si les résultats obtenus au moment X concurrent aux objectifs pour suivies de façon efficient et efficace. La programmation et l'évaluation sont complémentaires dans la mesure où l'on établit ces paramètres guidés, et l'autre s'en sent pour mesure l'espace parcourue et l'impact des actes posés vers les résultats des objectifs.

L'évaluation peut porter sur :

- ✓ La structure ;
- ✓ La technique, les hommes ;
- ✓ Les procédures, stratégies ;
- ✓ Les résultats et l'impact réel du projet.

Un projet est un ensemble d'opération qui doivent permettre d'atteindre un objectif clairement exprimable et présentant un certain caractère d'unicité<sup>14</sup>.

### II.1. Technique d'ordonnancement des tâches

Les problèmes d'ordonnancement sont apparus au départ dans la planification de grands projets. Le but était de gagner du temps sur leur réalisation.

L'ordonnancement est effectué par le chef de projet et lui permet d'identifier rapidement les activités critiques, d'élaborer ce budget et de choisir à réaliser par le personnel propre à l'organisation.

La connaissance des différentes tâches accomplir ne suffit pas pour réaliser un projet. Il faut encore une parfaite connaissance de l'articulation de ces tâches, établir un plan complet d'action permettant de la réaliser dans les conditions de coûts et du temps imposés, et pendant son déroulement, vérifier constamment si le plan établi est respecté, en cas d'écart observé, recherche à le réduire.

La réalisation d'un projet nécessite souvent une succession des tâches auxquelles s'attachent certaines contraintes :

- De temps délai à respecter pour l'exécution des tâches ;
- Antériorité : certaines tâches doivent être exécutées avant d'autres ;
- De simultanéité : certaines tâches peuvent – être réalisées en même temps.

### II.2. Méthode d'ordonnancement des tâches

Il existe plusieurs méthodes permettant de présenter un projet. Il s'agit de :

- ✓ Diagramme gant ;
- ✓ La méthode CPM ;
- ✓ La méthode MPM ;

---

<sup>14</sup> KUTUC, notes du cours d'Evaluation des projets, L2 info, 2012-2013, inédites

- ✓ La méthode PERT.

## 1. Le diagramme de gannt

Dans ce type de diagramme, on représente au sein d'un tableau de signes des différentes tâches en colonnes les unités en données les unités de temps (exprimées en mois, semaines, jour, heures, ...) la durée d'exécution d'une tâche est matérialisée par un trait au sein d'un diagramme.

Il représente 4 étapes différentes pour sa réalisation qui sont les suivantes :

- ✓ 1<sup>ère</sup> étape : déterminer ces différentes tâches ou opération à réaliser et leur durée ;
- ✓ 2<sup>ème</sup> étape : définit les relations d'antériorité entre tâches ;
- ✓ 3<sup>ème</sup> étape : représente d'abord les tâches n'ayant aucune antériorité, puis les tâches dont les tâches antérieures ont déjà été représenté et ainsi de suite ;
- ✓ 4<sup>ème</sup> étape : représente par un trait parallèle en pointillé, la tâche planifie la progression réelle du travail.

## 2. Méthode CPM

La méthode du chemin critique se penche sur les options les plus critiques (importantes et difficiles à réaliser) pour évaluer le temps requis. Elle répond à la question : quel est le temps le plus critiqué pour que toutes les activités du projet soient accomplies ? La réponse à cette question permet d'évaluer la durée du projet. La méthode débute en pratique par l'élaboration d'un réseau PERT pour aboutir à l'établissement, l'identification du chemin critique.

Elle comprend ainsi les ci – après :

- ✓ Indentification et établissement de toutes les activités nécessaires (numération des activités) ;
- ✓ Estimation de la durée de chaque activité en tenant compte de ressources disponibles et des paramètres spécifiques de performance ;
- ✓ Construire le réseau séquentiel PERT ;
- ✓ Identification du début et de la fin du projet ;
- ✓ Détermination du début le plus tôt et de la fin la plus rapprochée de chaque activité ;
- ✓ Identification du chemin critique pour le projet ;
- ✓ Détermination de la durée du projet en estimant le temps le plus court pour la réalisation des activités ;
- ✓ Utilisation du diagramme critique pour gérer le projet.

## 3. Méthode des potentiels métra (MPM)

La solution aux P.C.O est donnée par le graphe potentiel (MPM) pondéré est aussi un réseau de transport quasi- fortement connexe, un cas particulier du réseau de transport<sup>15</sup>.

### a) Eléments du graphe MPM

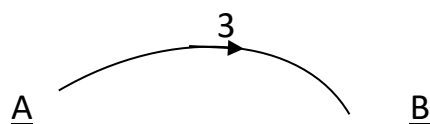
---

<sup>15</sup> MANYAN, notes du cours de Recherche Opérationnelle, UNIKIN G3 Math info, 2009-2010, inédites

Le graphe MPM est constituée des éléments suivants, à s'avoir :

- ✓ Chaque tâche représente un sommet (ou nœud) ;
- ✓ Chaque arc représente un sommet de succession (ou d'antériorité) ;
- ✓ La pondération (ou potentiel) d'un arc donne le temps qui doit s'écouler au minimum entre le début de la tâche origine de l'arc et le début de la tâche extrémité de l'arc.

Exemple :



Ce schéma ci- haut signifie que la tâche A doit précéder la tâche B et il s'écouler au minimum trois unités de temps entre le début de la tâche A et le début de la tâche.

#### 4. La méthode PERT

La méthode PERT est « une méthode consistant à mettre en ordre sous forme de réseau plusieurs tâches qui grâce à leur dépendance et à leur chronologie concurrente toutes à l'obtention d'un projet fini. Pour l'élaboration du planning de notre projet, nous allons utiliser la méthode PERT. Le signe PERT veut dire : Program of Evaluation an Review Technique (technique d'ordonnancement et de contrôle des programme).

Il s'agit d'une technique couramment utilisée en gestion de projet et en management. Elle a été mise au point en 1957 pour le compte de la marine américaine (USNAVY) afin d'établir et de mettre à jour des programmes de gestion.

En réalité, il s'agit d'un réseau (communément appelé diagramme) qui représente les tâches par des cercles, reliés entre eux, par des flèches ayant une direction. Ces flèches symbolisent les étapes du programme, tandis que leur sens indique l'antécédent entre les tâches<sup>16</sup>.

#### II.3. Principes de représentation en P.E.R.T

Ici, il faut noter que la méthode PERT permet de résoudre le problème central d'ordonnancement par le graphe PERT. Dans ce travail, nous ne considérons que les graphes PERT temps c'est-à-dire graphe temps déterministe.

Dans un graphe PERT :

Chaque tâche est représentée par un arc, auquel on associe un chiffre entre paramètre qui représente la durée de la tâche ;

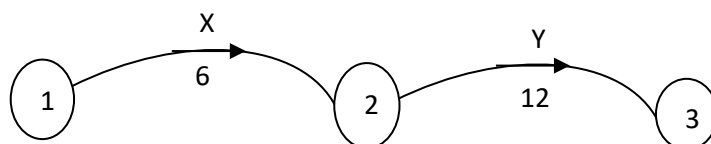
Entre les arcs figurent des cercles appelés « Sommet » au « Evénement » numéro afin de suivre l'ordre de succession des divers événements.

<sup>16</sup> MANYAN, op.cit.

Les sommets du graphe PERT représentent les éléments déterministes chaque opération représente la fin des opérations.

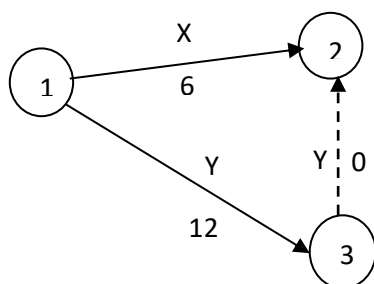
La disposition des arcs doit traduire la succession ou au contraire la simultanéité des opérations si extrémités finales d'un arc coïncide avec l'extrémité initiale d'un autre, cela signifie que l'opération associée au premier arc doit être achevée pour que celle associée au 2<sup>ème</sup> arc puisse débiter<sup>17</sup>.

Exemple 1 : Succession des opérations (tâches)



L'opération Y de durée 12 unités de temps ne peut débiter qu'avant que l'opération X de durée 6 unités de temps puisse être achevée. Cela veut dire qu'y débiter nécessairement à la fin de X si non il n'y aura pas de souplesse.

Exemple 2 : Simultanéité des opérations (tâche)



Ceci signifie que l'opération X peut être exécutée.

Remarque : il est nécessaire d'introduire une tâche fictive de durée égale à 0, pour représenter la relation d'autorité entre deux nœuds permet de déterminer la date au plus tôt de la réalisation du projet. Cette succession de tâches constitue le chemin critique.

Construction du graphe PERT.

Chaque sommet se représente comme suit :

DTO	DTA
X	

Où : X : désigne le numéro du sommet (événement) ;

✓ DTO : désigne la date attendue E de l'événement X ou la date au plus tôt de l'événement.

<sup>17</sup> Jean F., cours de Graphe, Paris 16 février 2007, OS, inédit

✓ DTA : désigne la date limite ou la date au plus tard de l'événement X<sup>18</sup>.

## II.4. Contraintes d'un graphe P.E.R.T

Dans ce paragraphe, nous avons des tâches pour des contraintes de postériorité qui signifient simplement qu'on ne peut démarrer une tâche avant que celle qui la précède ne soit terminée ou arrivée à sa fin<sup>19</sup>.

### II.4.1. Identification des tâches

Ces tâches suivantes sont identifiées pour notre projet :

1. Recueil des besoins
2. Analyse des besoins
3. Validation des besoins
4. Etudes de faisabilité
5. Elaboration du cahier de charges
6. Validation du cahier de charges
7. Conception du système
8. Implémentation et tests unitaires
9. Déploiement et validation logiciel
10. Documentation et Formation utilisateurs
11. Maintenance

### II.4.2. Tableau d'ordonnement des tâches du projet

Tableau 1: Tableau d'ordonnement des tâches du projet

Code Tache	Activités	Durée (Jours)	Tache antérieure
A	Recueil des besoins	4	-
B	Analyse des besoins	7	A
C	Validation des besoins	7	B
D	Etudes de faisabilité	10	C
E	Elaboration du cahier de charges	5	D
F	Validation du cahier de charges	3	E
G	Conception du système	10	F
H	Implémentation et tests unitaires	20	G
I	Déploiement et validation logiciel	7	H
J	Documentation et Formation utilisateurs	7	I et H

### II.4.3. Contraintes d'élaboration du projet

✓ En répondant aux questions suivantes :

<sup>18</sup> Jean, op.cit.

<sup>19</sup> SULA, op.cit.

- ✓ Quelle (S) tâche (S) doit être terminée immédiatement
- ✓ Avant qu'une autre recommence ?
- ✓ Quelle tâche doit suivre une tâche déterminée ?

#### II.4.3.1. Construction du graphe en P.E.R.T

Le P.E.R.T (Programme of Evaluation and Technic) est une méthode qui consiste à mettre en ordre sous forme de réseau plusieurs tâches qui, grâce à leur chronologie, concurrent toutes à l'obtention d'un produit fini<sup>20</sup>.

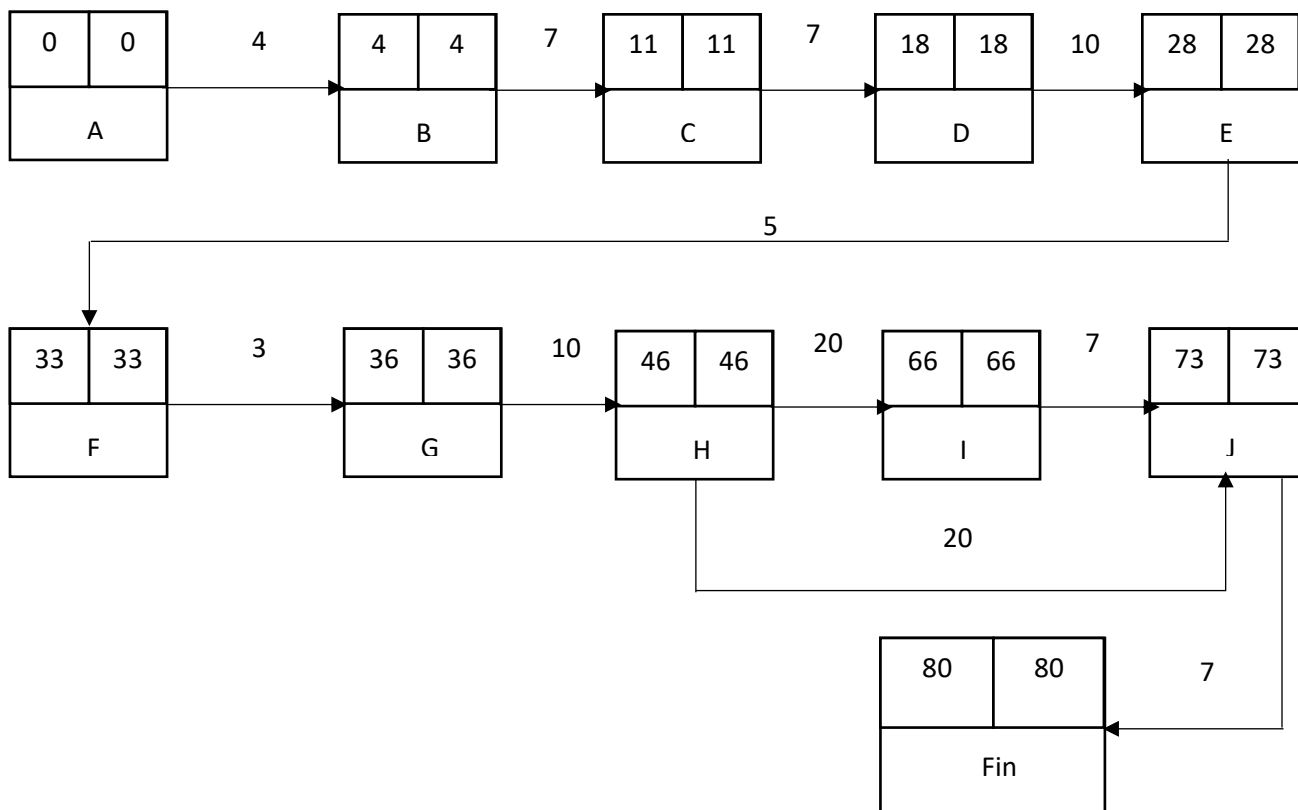


Figure 9 : Graphe Pert

#### II.4.3.2. Détermination du chemin critique

Le chemin critique est l'ensemble des sommets du graphe par lesquels, si on dépasse la date au plutôt pour le démarrage d'une tâche, ce dernier affectera l'ensemble du projet.

<sup>20</sup> SULA, op.cit.

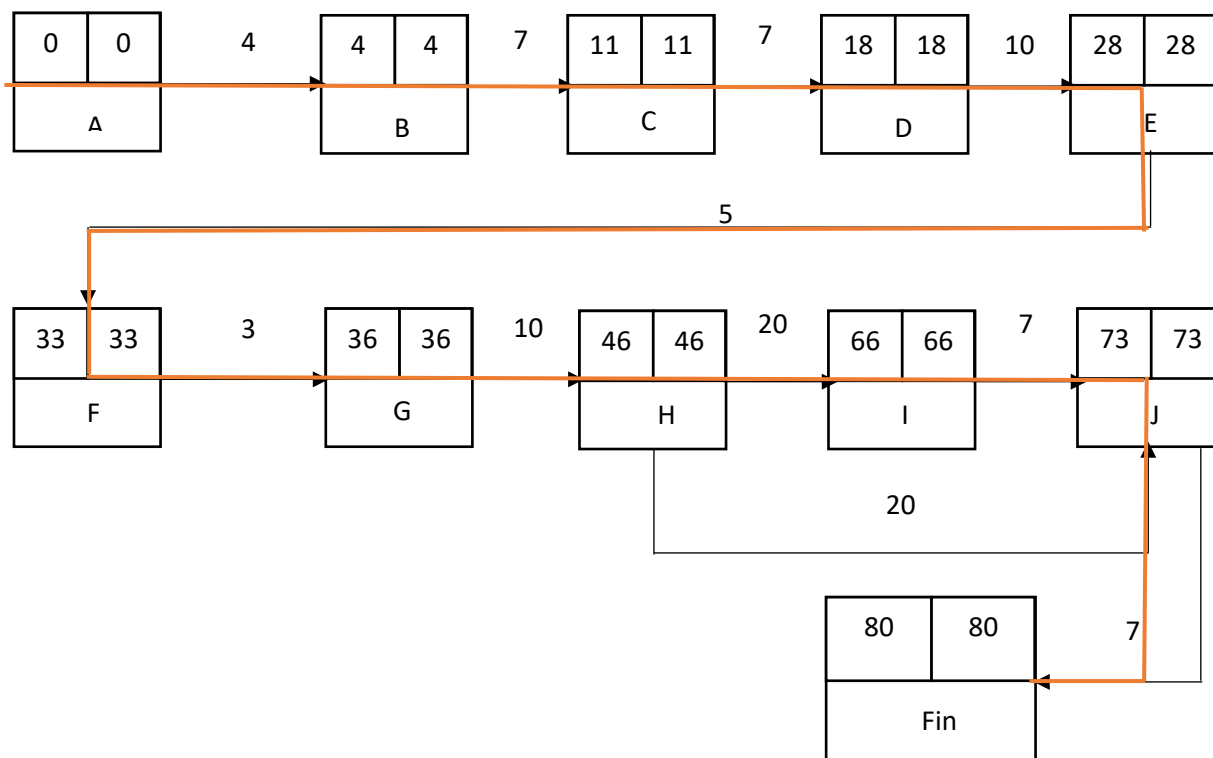


Figure 10 : chemin critique

### II.5. COUT DU PROJET

Ici nous allons essayer d'évaluer le coût des équipement nécessaires pour la réalisation et le fonctionnement de notre système. De ce fait, nous allons donner les différents prix de

Tableau 2: Estimation du coup

Matériel	Prix Unitaire en USD	Quantité	Prix Total en USD
Ordinateur Portable	500	2	1000
Imprimantes	65	2	130
Onduleur 1500 Watts	250	2	500
Main d'œuvre			500
<b>TOTAL</b>			<b>2130</b>

## CHAPITRE III : CADRAGE DU PROJET

### III.1. Présentation de l'environnement du système

#### III.1.1. Historique

L'orphelinat a été créé en 2002 par la sœur Pétronelle Mukuna.

#### III.1.2. Objectifs sociaux et activités principales

##### a. Objectifs sociaux

Son objectif social consiste en l'accompagnement de ...

- Soutien émotionnel et psychologique : Offrir un encadrement affectif pour aider les enfants à surmonter des traumatismes.
- Éducation : Assurer l'accès à une éducation de qualité pour favoriser leur développement personnel et professionnel.
- Intégration sociale : Aider les enfants à s'intégrer dans la société, en leur enseignant des compétences sociales et en les impliquant dans des activités communautaires.
- Santé et bien-être : Fournir des soins médicaux et une alimentation adéquate pour garantir leur santé physique.
- Préparation à l'autonomie : Former les jeunes à devenir autonomes lorsqu'ils quitteront l'orphelinat, en leur enseignant des compétences pratiques et professionnelles.

##### b. Activité principale

- Accueil et hébergement : Offrir un lieu sûr où les enfants peuvent vivre, dormir et se sentir en sécurité.
- Soutien éducatif : Organiser des activités scolaires et aider les enfants à poursuivre leur éducation, que ce soit en les inscrivant à l'école ou en proposant des cours de soutien.
- Activités récréatives : Proposer des loisirs, des jeux et des activités culturelles pour favoriser le développement social et émotionnel des enfants.
- Accompagnement psychologique : Mettre en place un soutien psychologique pour aider les enfants à gérer leurs émotions et leur passé.
- Formation à la vie quotidiennes : Enseigner aux enfants des compétences pratiques comme la cuisine, le ménage ou la gestion de l'argent, afin qu'ils soient préparés à la vie adulte

#### III.1.3. Situation géographique

L'orphelinat **notre dame de liesse est** situé sur l'avenue Essence N°21 au Quartier KIMPE dans la Commune de Ngaliema dans la ville-province de Kinshasa.

### III.1.4. Organigramme général

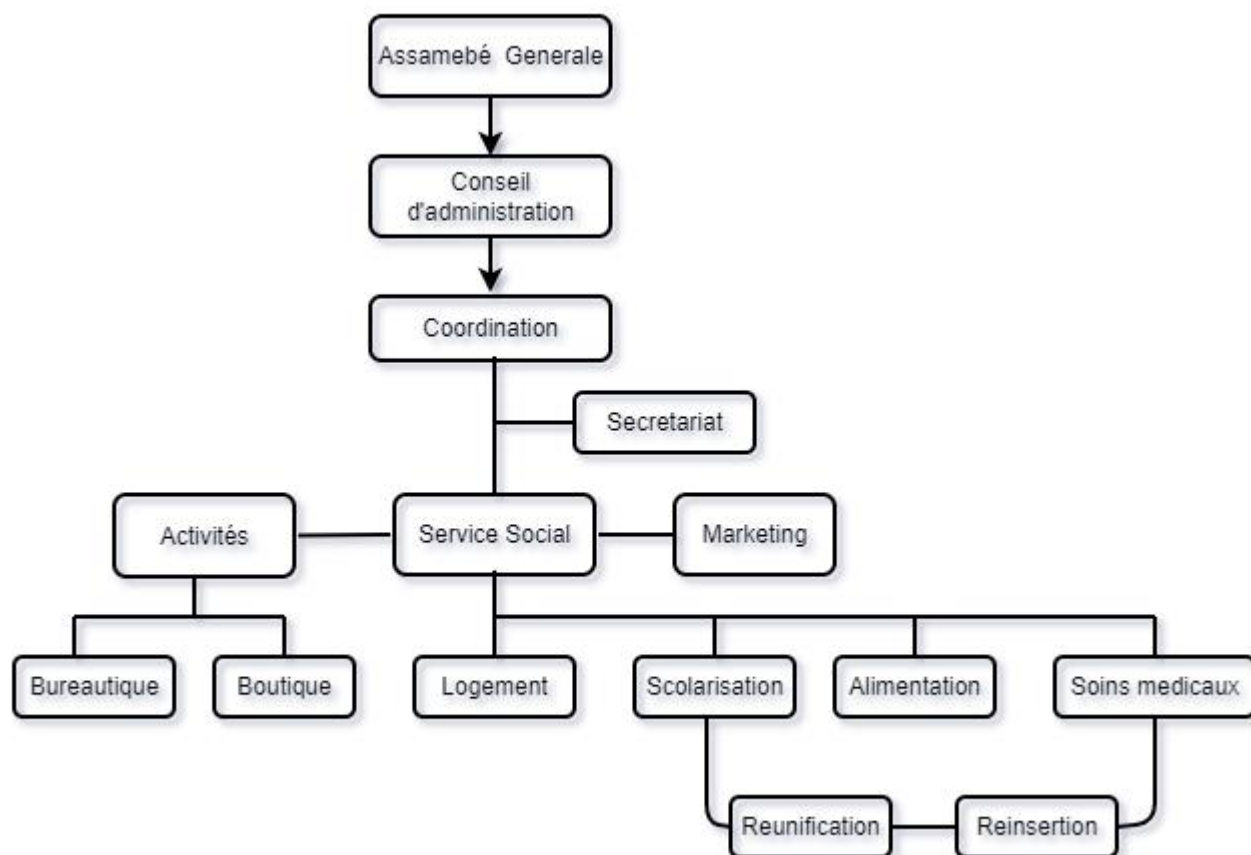


Figure 11 : Organigramme de l'orphelinat

### III.1.6. Organigramme de service concerné

#### III.1.6.1 Structure fonctionnelle

Ici, il sera question de savoir : comment est organisée l'orphelinat « Notre dame de LIESSE » ? C'est la question principale. Une autre qui lui est subséquente, c'est celle de savoir comment cet orphelinat fonctionne-t-elle service par service ?

La subdivision administrative de l'orphelinat « Notre dame de LIESSE » nous met en face de 3 principaux niveaux d'organisation :

- ✓ La secrétaire (Réceptionniste) ;
- ✓ Le coordonnateur ;
- ✓ Autres services (administratifs ou techniques).

#### 1. La Secrétaire

Il joue aussi le rôle de réceptionniste et est chargé d'accueillir toute personne au sein de l'orphelinat en lui donnant des indications et toute information nécessaire selon les demandes. Il est également chargé d'effectuer les enregistrements des enfants, la rédaction des rapports, des lettres et bien d'autres tâches administratives.

#### 2. Le coordonnateur

Comme son poste l'indique, il est chargé de la gestion de l'ensemble de toutes les activités, et la prise de décision sur l'acceptation d'une demande au sein de l'orphelinat.

#### 3. D'autres services

Ces autres services ne faisant pas partis de notre travail, nous avons choisi de ne pas intégrer dans le développement de notre travail.

### III.1.7. Connaissance du plan directeur

Les entreprises de grande taille qui, en général se servent de l'information depuis longtemps que périodiquement, elles doivent évaluer les systèmes d'information en place et définir plan directeur d'utilisation de ces systèmes pour les années à venir, dès lors que toutes ces entreprises (petites et grandes) doivent se servir à ce même exercice.

### III.1.5. Présentation et fonctionnement

Du point de vue fonctionnement, le secrétaire est chargé d'enregistrer les enfants, ainsi que les personnes qui les accompagne dans leurs nouveaux cadres, ainsi que les personnes qui adoptent ces enfants après leur passage au sein de la structure.

### III.1.6. Diagramme du contexte

C'est un diagramme nous permettant de contextualiser l'environnement de travail c'est-à-dire de représenter les flux d'informations entre l'organisation et les acteurs externes selon une représentation standard dans laquelle chaque objet porte un nom.

- ✓ L'organisation est représentée par un rectangle ;

- ✓ Les acteurs externes sont représentés par des ellipses en pointillés ;
- ✓ Les flux d'information sont représentés par des flèches dont l'orientation désigne le sens du flux d'information.

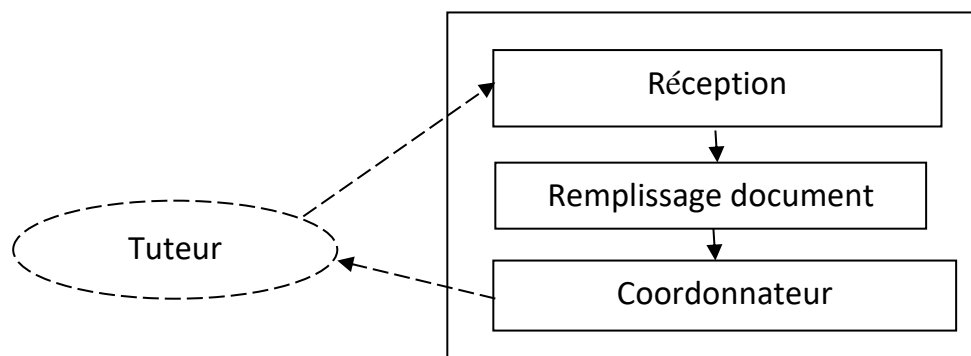
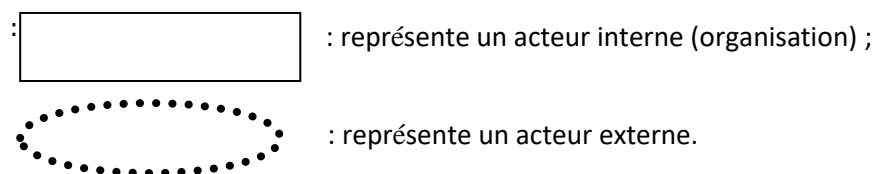


Figure 12 : Présentation du diagramme de contexte

### III.1.7. Etude du réseau d'information

#### III.1.7.1. Narration

Le processus d'enregistrement des données sur les enfants se déroule comme suit :  
 A l'arrivée, le tuteur se présente à la réception ou il exprime ses besoins et présente sa carte. Après échange avec le réceptionniste, ce dernier lui remet un document à remplir renseignant sur son identité et celui de l'enfant qu'il accompagne. Le document étant rempli, ce dernier est dirigé vers le Coordonnateur pour un examen du dossier. Après cet analyse, si la demande est jugée recevable, le document est ramené à la réception (secrétariat) pour une insertion dans le registre des enfants ; dans le cas contraire, le document est classé et la demande du tuteur est jugée irrecevable.

#### III.1.7.2. Diagramme et matrice de flux

##### III.1.7.2.1. Présentation du diagramme de flux

Ce diagramme (appelé aussi Modèle Conceptuel de la Communication) permet de compléter le diagramme de contexte en décomposant l'organisation en une série d'acteurs internes. Dans ce diagramme la représentation standard est la suivante :

- ✓ Les acteurs internes sont représentés par des ellipses ;
- ✓ Les messages internes sont représentés par des flèches ;
- ✓ Un flux c'est une information circulante entre un ou plusieurs acteurs.

Il existe le flux logique (ne s'accompagne pas d'un document physique) et le flux physique (s'accompagne avec un document physique).

Le formalisme se fait d'une manière directe ou indirecte c'est-à-dire en prévoyant une légende à la fin de la présentation.

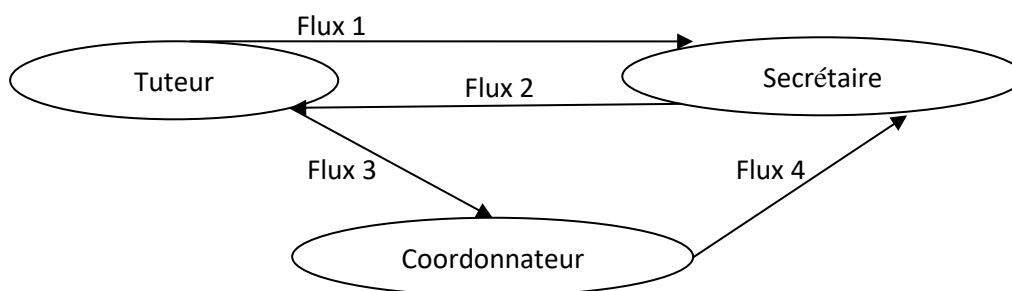


Figure 13 : Présentation du diagramme de flux

### III.1.7.2.2. Description du diagramme de flux

Tableau 3: Description du diagramme de flux

Flux	Acteur émetteur	Acteur récepteur	Document
Flux 1	Tuteur	Secrétaire	Identité
Flux 2	Secrétaire	Tuteur	Fiche d'identification
Flux 3	Tuteur	Coordonnateur	Fiche d'identification
Flux 4	Coordonnateur	<b>Secrétaire</b>	Fiche d'identification recevable

### III.1.7.2.3. Présentation de matrice de flux

La matrice des flux est une représentation matricielle des acteurs et des flux échangés. Les acteurs forment les lignes et les colonnes du tableau. Situé en ligne, l'acteur a un rôle d'émetteur de flux ; situé en colonne, il a un rôle de destinataire de flux. Les flux sont indiqués dans les « cases » du tableau, à l'intersection de la ligne « acteur-émetteur » et de la colonne « acteur-destinataire ».

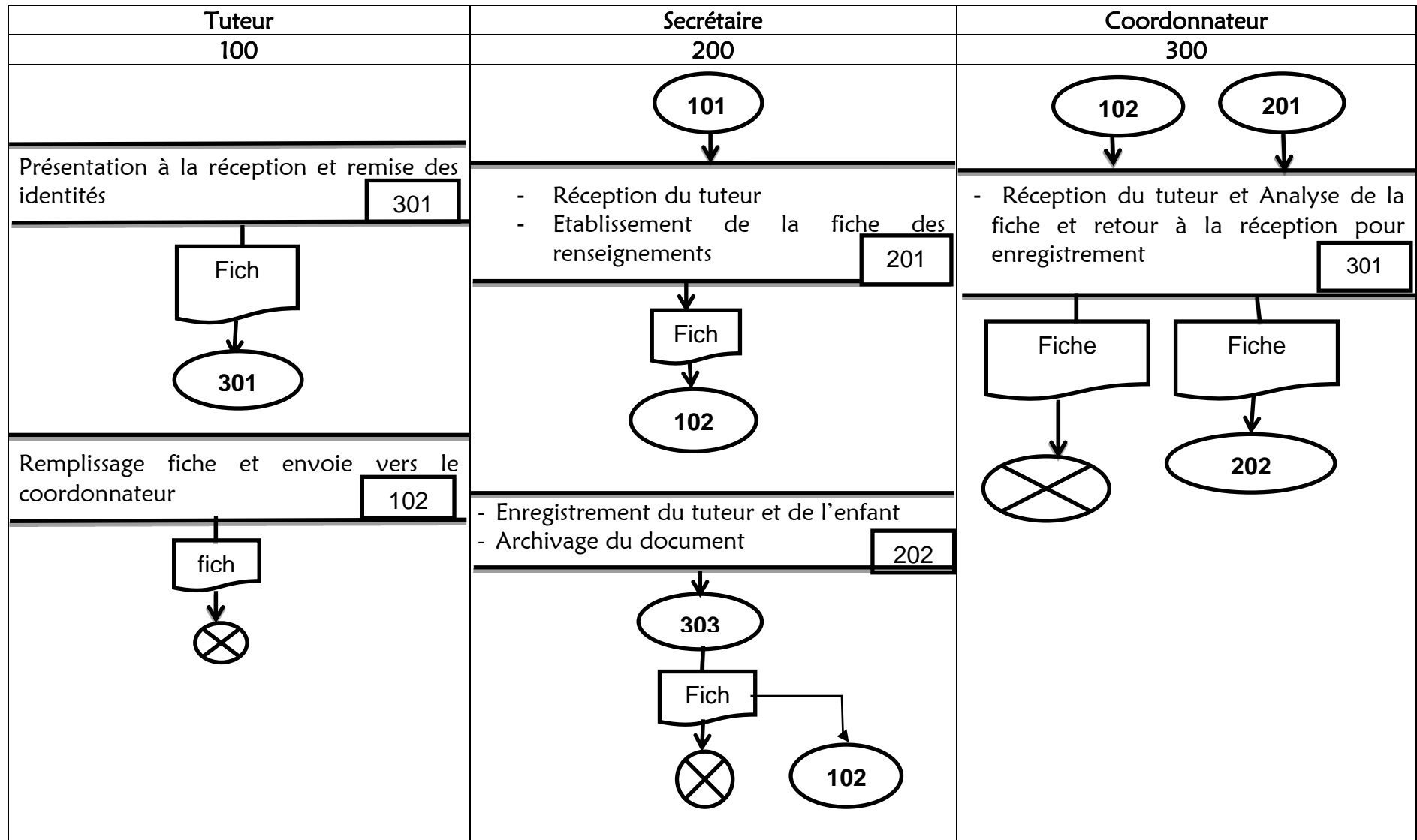
La matrice des flux est un outil de synthèse permettant au concepteur d'avoir une vision synthétique sur l'ensemble des flux du domaine considéré.

### III.1.7.2.4. Description de matrice de flux

Tableau 4 : Matrice des flux

Emetteur Récepteur	Poste externe	Poste interne	
	Tuteur	Secrétaire	Gérant
Tuteur	-----	Identité	-----
Secrétaire	Fiche d'identification	-----	Fiche d'identification
Gérant	-----	Fiche d'identification recevable	-----

### III.1.7.3. ETUDE DU SCHEMA DE CIRCULATION DES INFORMATIONS



### III.1.8. Etude des documents utilisés

#### III.1.8.1. Recensements

En ce qui nous concerne, nous avons recensé les documents tels que le (l) :

- ✓ Carte d'identité
- ✓ Fiche d'enregistrement

#### III.1.8.2. Description

##### 1. Carte d'identité

**a) Rôle :** Permet à une personne d'être identifiée.

**b) Description**

Tableau 5 : Description de la carte d'identité

Code rubrique	Rubrique	Caractères	Taille
Num	Numéro	AN	10
Nom	Nom	AN	15
Postnom	Postnom	AN	15
Prenom	Prénom	AN	15
Adres	Adresse	AN	50
sexreq	Sexe	AN	1
Datenais	Date naissance	DATE	<b>45</b>
Provre	Province	AN	20
Sectre	Secteur	AN	20
Territre	Territoire	AN	20
District	District	AN	20

##### 2. Fiche d'enregistrement

**a) Rôle :** C'est pour l'enregistrement du requérant pour un quelconque document sollicité.

**b) Modèle :**


Fiche adaptée pour le travail social des ENFANTS DE LA RUE

C  
REPUBLIQUE DEMOCRATIQUE DU CONGO

Fiche d'identification individuelle des enfants en rupture familiale

(Confidentiel)

Nom de la structure :



1. Identité de l'Enfant

Nom TUBANE Postnom MINANI  
 Prénom PISCASSE Sobriquet \_\_\_\_\_

Sexe M  F

Age donné par l'enfant 19/04/2009 Age estimé par l'enquêteur 19/04/2009 Mans

Nationalité CONGOLAISE

LIEU DE NAISSANCE: INDUÉ PROVINCE D'ORIGINE SUD KIVU  
 DISTRICT D'ORIGINE INDWI SUD TERRITOIRE/COMMUNE D'ORIGINE KINTAMA  
 SECTEUR/QUARTIER D'ORIGINE \_\_\_\_\_ COLLECTIVITE D'ORIGINE NTALANGWA  
 LOCALITE D'ORIGINE \_\_\_\_\_ TRIBU D'ORIGINE BAHAYU  
 Pays de Résidence RDC Religion ACK Eglise de Reveil  
 Langue(s) parlées SWAHILI LINDALA  
 Niveau scolaire ou profession exercée avant la séparation 2<sup>e</sup> primaire  
 Signes physiques distinctifs \_\_\_\_\_

Nom du père TUBANE L Postnom/Prénom/Sobriquet LEONARD

Ne sait pas  En vie  Décédé

Etat civil ; Marie  Célibataire  Divorce  Veuf  Divorce / remarie

Locataire  Propriétaire  Sous logé  Nombre de personne à charge

---

Nom de la mère MINANI Postnom/Prénom/Sobriquet RACHEL

Ne sait pas  En vie  Décédée

Etat civil ; Marie  Célibataire  Divorce  Veuf  Divorce / remarie

## 3. Différentes adresses connues par l'enfant

Adresses	Points de repère	Nom de la personne	Lien de parenté	Observation

## 4. Frères et/ou sœurs avec l'enfant, (le cas échéant)

A. Nom/Postnom/Prenom/Sobriquet \_\_\_\_\_ Age \_\_\_\_\_ Sexe M  F B. Nom/Postnom/Prenom/Sobriquet \_\_\_\_\_ Age \_\_\_\_\_ Sexe M  F 

## 5. Prise en charge actuelle de l'enfant

a. Centre  b. Famille d'accueil "intérimaire"  c. Famille d'accueil "spontanée"   
d. Famille apparentée  f. Autre  (préciser) \_\_\_\_\_Nom du centre ou de la personne qui prend en charge l'enfant \_\_\_\_\_  
Adresse \_\_\_\_\_

Tel. \_\_\_\_\_

Date de la prise en charge \_\_\_\_\_

Adresse de la famille d'accueil où l'enfant aimerait retourner ou se rendre (le cas échéant) \_\_\_\_\_  
Tel. \_\_\_\_\_

## 6. Vœux de l'enfant pour la réunification

Personne que l'enfant voudrait retrouver

Père  Mère  Frère  Sœur  Autre  (Veuillez spécifier) \_\_\_\_\_

A. Nom complet de la personne et lien avec l'enfant \_\_\_\_\_

Fiche adaptée pour le travail social des ENFANTS DE LA RUE

Dernière adresse connue \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_ Tel \_\_\_\_\_

B. Nom complet de la personne et lien avec l'enfant \_\_\_\_\_  
 Dernière adresse connue \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_ Tel \_\_\_\_\_

L'enfant est-il en contact / a t'il eu des nouvelles de quelque parenté que ce soit? (Veuillez spécifier):  
 \_\_\_\_\_  
 \_\_\_\_\_

**7. Interview par d'autres organisations impliquées dans la recherche**

L'enfant a t'il été interviewé par d'autres organisations?    •    Oui     Non

Nom de l'organisation \_\_\_\_\_

Lieu de l'interview \_\_\_\_\_

Pays \_\_\_\_\_    Date \_\_\_\_\_

Numéro de référence donné à l'enfant par une/d'autre(s) organisation(s) : \_\_\_\_\_

**8. Liste des documents et des biens importants que l'enfant possède**

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

**9. Permission d'utiliser l'information**

Est-ce que l'enfant est d'accord pour que les informations de base soient utilisées dans le but de rechercher sa famille?    Oui     Non

\_\_\_\_\_

**10. Informations sur les modifications éventuelles des données de l'enfant** \_\_\_\_\_

\_\_\_\_\_

**11. Organisation responsable de la recherche familiale** \_\_\_\_\_

**12. Lieu et date de l'interview** \_\_\_\_\_

**13. Nom de l'interviewer** \_\_\_\_\_

**14. Signature de l'interviewer** \_\_\_\_\_

\_\_\_\_\_

\* Il y a parfois un écart important entre ce que l'enfant dit et son âge véritable. (parfois 4 ans d'écart).  
 \* Préciser l'adresse le plus possible (Province/ District/ Territoire ou Commune/ Secteur ou Quartier/ Collectivité/ Localité).

Fiche adaptée pour le travail social des ENFANTS DE LA RUE

Comment l'enfant sait-il que le/les parents sont décédés?  
 sa maman biologique

Etat civil : Marié  Célibataire  Divorce  Veuf  Divorce / remarié

Locataire  Propriétaire  Sous logé  Nombre de personne à charge

Dernière adresse connue des parents Hotel OKAPI case

Nom du tuteur \_\_\_\_\_ Postnom/Prénom/Sobriquet \_\_\_\_\_

Locataire  Propriétaire  Sous logé  Nombre de personne à charge

Nom du tuteur \_\_\_\_\_ Postnom/Prénom/Sobriquet \_\_\_\_\_

Date d'arrivée au centre : 21/05/2020

Durée passée dans la rue : /

**2. Histoire de la séparation**

Date de la séparation 21/05/20 Lieu de séparation\* Cave / OKAPI

Circonstances de la séparation Après la mort de leur papa biologique la maman Rachel s'est remarié (Yaka Towafda, concubi nage) avec M. Ilunga de FARDC avec qui elle a eu 3 enfants. Ils ont quitté Bukavu pour Kinshasa Cave Okapi. Le papa Ilunga ne veut plus de 3 enfants, il les persécute frappe avec les sabots et foyets, blessure. c'est ainsi que Maman Rebecca a demandé pas la commune l'hebergement.

## c) Description

Code rubrique	Rubrique	Caractères	Taille
Enfant	Identifiant Enfant	Idenf	5
	prenom Enfant	Prenom	15
	Nom Enfant	Prenom	15
	Post-nom Enfant	Postnom	15
	Sexe	sexe	1
	date de naissance	datenaiss	15
	lieu de naissance	lieunaiss	20
	Adresse	Adress	50
	Noms du père	NomPer	30
	Noms de la mère	NomMer	30
	Village	Village	30
	Secteur	Secteur	30
	Chefferie	Chefferie	30
	Territoire	Territ	30
	Province	Prov	30
Nombre des freres	Nbfr	5	
Nombre des soeurs	Nbsr	5	

### III.1.9. Etude des moyens matériels et humains

#### III.1.9.1. Etude des moyens matériels

Tableau 6: Présentation des moyens matériels

Nom outils	Nombre	Nombre personnes	Année acquisition	Marque	Etat
Table	2	1	-	-	Bon
Chaise	12	6	-	-	Bon
Stylo	6	6	-	-	Bon
Agrafeuse	2	6	-	-	Bon
Papier duplicateur	Plusieurs	6	-	-	Bon

#### III.1.9.2. Etude des moyens humains

Tableau 7: Présentation des moyens humains

Nom poste	Nbre personnes/poste	Spécialité	Fonction	Ancienneté	Niveau d'étude
Secrétaire	1		220	+10	-
Gerant	1	-	232	+5	-
Secrétaire	1	-	243	+5	-
Agent	3	-	223	-	-

### III.1.10.1. Critique du système d'information existant

#### III.1.10.1.1. Définition :

C'est une étude minutieuse (rigoureuse) conduisant à détecter les causes de la non évolution dudit système.

#### III.1.10.1.2. Critique proprement dit

##### III.1.10.1.2.1. Les atouts

- ✓ Il y a une bonne collaboration entre les agents internes au sein dudit orphelinant et un bon climat de travail y règne ;
- ✓ L'orphelinant est doté d'ordinateurs.

##### III.1.10.1.2.2. Les faiblesses

- ✓ Bureau de petite dimension ;
- ✓ Insuffisance d'armoires pour la conservation des documents ;
- ✓ Le traitement des informations se fait manuellement ;
- ✓ Difficultés sur la recherche des données surtout celle de longue date ;

- ✓ Insécurité et perte facile des documents.

### **III.1.11. Proposition des solutions**

#### **III.1.11.1. Solution non Informatique (Manuelle)**

Par rapport à la solution non informatique, nous préconisons ce qui suit :

- ✓ La construction de nouveaux bureaux spacieux ;
- ✓ Achat de nouvelles armoires à tiroirs afin d'éviter les pertes d'informations et d'assurer l'efficacité de la conservation.

##### **III.1.11.1.1. Avantages**

- ✓ La non dépendance à l'énergie électrique (courant) ;
- ✓ Solution moins coûteuse.

##### **III.1.11.1.2. Inconvénients**

- ✓ Le problème de l'insécurité ainsi que de la perte des documents n'est pas sensiblement résolu ;
- ✓ L'inconfort pendant le travail ;
- ✓ La solution reste attachée aux faiblesses de l'être humain, car celui-ci est au centre du traitement.

#### **III.1.11.2. Solution Informatique**

Puisque l'émergence de l'Informatique en matière de gestion n'est plus à douter, l'ordinateur demeure l'outil privilégié d'aide à la gestion d'information. Ainsi, nous préconisons :

- ✓ La mise en place d'une base de données pour le système d'information concerné ;
- ✓ La mise en place d'un réseau informatique permettant aux acteurs différents de pouvoir manipuler les données. Ceci nécessitera donc l'achat des matériels adéquats pour le déploiement et l'exploitation d'un système informatisé ainsi que le financement nécessaire pour l'initiation du personnel à l'emploi du nouveau système, voire le recrutement d'un agent spécialiste (Informaticien).

##### **III.1.11.2.1. Avantages**

- ✓ Rapidité dans le traitement ;
- ✓ Sécurité des données ;
- ✓ Confort pendant le travail.

##### **III.1.11.2.2. Inconvénients**

- ✓ Prix élevé de mis en place ;
- ✓ La dépendance à l'énergie électrique ;
- ✓ La nécessité d'une main d'œuvre spécialisée.

## **Conclusion partielle**

Après analyse des hypothèses par rapport à la problématique, après étude du service concerné de notre travail, objet de cette section et après étude des différentes solutions préconisées, nous estimons qu'au regard des avantages et inconvénients présentés pour les deux solutions, l'orphelinat Notre Dame de Liesse ferait mieux d'adopter la Solution Informatique pour sa bonne marche et pour une meilleure gestion des informations (données) accompagnées d'un rendement meilleur.

## CHAPITRE IV : CONCEPTION DU NOUVEAU SYSTÈME

### Section I : CONCEPTION DU SYSTEME D'INFORMATION ORGANISATIONNEL (CSIO)

#### I.1. Etape conceptuelle

L'objectif est de représenter l'activité de l'entreprise et de formaliser son "système d'information" indépendamment de son organisation. Le compte rendu de cette étude est matérialisé sous la forme de dessins normalisés, de modèles complétés par un dossier explicatif. Le but de ce chapitre est d'expliquer comment décrire l'entreprise concernée en respectant les normes de chaque modèle.

##### 1. Présentation des règles de gestion

Après analyse fonctionnelle et analyse de flux d'informations dans le système dont il est question d'informatiser, les règles de gestion se présentent comme suit :

R1 : Un enfant est enregistré par un et un seul secrétaire

R2 : un Secrétaire enregistre un ou plusieurs enfants

R3 : Un tuteur fait enregistrer un ou plusieurs enfants auprès d'un et un seul secrétaire

R4 : Un enfant est autorisé par un et un seul coordonnateur

R5 : un coordonnateur autorise un ou plusieurs enfants pour une ou plusieurs familles d'accueil

R6 : Dans une famille d'accueil est autorisé un ou plusieurs enfants

#### I.1.1. Modélisation Conceptuelle de Données (MCD)

##### 1.1.1. Définition et But de la modélisation conceptuelle de données

Le modèle conceptuel des données est une représentation statique du système d'information de l'entreprise qui met en évidence sa sémantique. Il a pour but d'écrire de façon formelle les données qui seront utilisées par le système d'information. Il s'agit donc d'une représentation des données, facilement compréhensible. Cet aspect recouvre les mots qui décrivent le système ainsi que les liens existants entre ces mots. Le formalisme adopté par la méthode Merise pour réaliser cette description est basé sur les concepts « entité-association ».

#### A. Définition des concepts

##### a. Entité

Une entité (ou objet) est une représentation d'un élément matériel ou immatériel ayant une existence propre et jouant un rôle dans le système que l'on désire décrire. La figure ci-dessous nous montre la représentation d'un objet.



Figure 14 : représentation d'un objet

## b. Occurrence

Une **occurrence**, c'est tout simplement une « **ligne** » de valeurs. Dans une entité, une occurrence correspond à l'ensemble des valeurs des propriétés rattachées à un seul identifiant. Dans une relation, une occurrence correspond à l'ensemble des valeurs des propriétés de la relation (représenté par les clés de chaque entité liée) : on l'appelle alors une **occurrence de relation**.<sup>22</sup>

## c. Identifiant

L'identifiant est une propriété particulière d'un objet telle qu'il n'existe pas deux occurrences de cet objet pour lesquelles cette propriété pourrait prendre une même valeur. Le modèle conceptuel des données propose de souligner les identifiants (parfois de les faire précéder d'un #).

## d. Relation

Une association (appelée aussi parfois relation) est un lien sémantique entre plusieurs entités.

- **La dimension d'une association**

La dimension d'une relation indique le nombre d'entités participant à l'association. Les dimensions les plus courantes sont 2 (association binaire) et 3 (association ternaire) :

- ✓ Une relation **récursive** (ou *réflexive*) relie la même classe d'entité ;
- ✓ Une relation **binaire** relie deux classes d'entité ;
- ✓ Une relation **ternaire** relie trois classes d'entité ;
- ✓ Une relation **n'aire** relie n entité.

Les relations sont représentées par des hexagones (parfois des ellipses) dont l'intitulé décrit le type de relation qui relie les entités (généralement un verbe à l'infinitif). On peut éventuellement ajouter des propriétés aux relations.

Une relation n'a pas d'existence propre et doit être définie par les individus auxquels elle est rattachée.

- **Une relation peut être porteuse des propriétés**

<sup>22</sup> <http://www.base-de-donnees.com/occurrence/>

Les propriétés qui dépendent fonctionnellement de plusieurs identifiants d'entités sont portées par les associations entre ces entités. C'est une dépendance fonctionnelle multi propriété au niveau de la source.

**Retenons** : Les relations de type père-fils ne sont plus porteuses des propriétés

## B. Recensement et description des entités

### a. Recensement des entités

L'analyse de règles de gestion nous a conduit à ressortir les objets suivants :

- ✓ Coordonnatrice
- ✓ Enfant
- ✓ Tuteur
- ✓ Secrétaire
- ✓ Famille d'accueil

### b. Description des entités

Tableau 8: Description des entités

Objets	Propriétés	Code propriété	Taille	Type	Identifiant
Enfant	Identifiant Enfant	Idenf	5	AN	#
	prenom Enfant	Prenom	15	AN	
	Nom Enfant	Prenom	15	AN	
	Post-nom Enfant	Postnom	15	AN	
	Sexe	sexe	1	AN	
	date de naissance	datenaiss	15	AN	
	lieu de naissance	lieunaiss	20	AN	
	Adresse	Adress	50	AN	
	Noms du père	NomPer	30	AN	
	Noms de la mère	NomMer	30	AN	
	Village	Village	30	AN	
	Secteur	Secteur	30	AN	
	Chefferie	Chefferie	30	AN	
	Territoire	Territ	30	AN	
	Province	Prov	30	AN	
	Nombre des frères	Nbfr	5	N	
	Nombre des sœurs	Nbsr	5	N	

Tuteur	Identifiant Tuteur	Identut	5	AN	#
	prenom Tuteur	Prenom	15	AN	
	Nom Tuteur	Nom	15	AN	
	Post-nom Tuteur	Postnom	15	AN	
	Sexe	sexe	1	AN	
	date de naissance	datenaiss	15	AN	
	lieu de naissance	lieunaiss	20	AN	
	Adresse	Adress	50	AN	
	Telephone	Tel	30	AN	
	Adresse Mail	Email	50	AN	
	Village	Village	30	AN	
	Secteur	Secteur	30	AN	
	Chefferie	Chefferie	30	AN	
	Territoire	Territ	30	AN	
Province	Prov	30	AN		
Secrétaire	Identifiant Secrétaire	IdSec	5	AN	#
	Nom	nom	15	AN	
	Postnom	postnom	15	AN	
	Prenom	Prenom	15	AN	
	Sexe	Sexe	1	AN	
	Lieu de Naissance	Lieunais	30	AN	
	Lieu de Naissance	DateNais	30	AN	
	Adresse	Adress	50	AN	
	Telephone	Tel	30	AN	
	Adresse Mail	Email	50	AN	
Coordonnatrice	Identifiant Coordonnatrice	IdSec	5	AN	#
	Nom	nom	15	AN	
	Postnom	postnom	15	AN	
	Prenom	Prenom	15	AN	
	Sexe	Sexe	1	AN	
	Lieu de Naissance	Lieunais	30	AN	
	Lieu de Naissance	DateNais	30	AN	
	Adresse	Adress	50	AN	
	Telephone	Tel	30	AN	
	Adresse Mail	Email	50	AN	
Fam Accueil	Identifiant Famille	Idfa	5	AN	#
	Prenom Represent Famille	Prenom	15	AN	
	Nom Represent Famille	Nom	15	AN	
	Post-nom Represent Famille	Postnom	15	AN	
	Sexe	sexe	1	AN	
	date de naissance	datenaiss	15	AN	
	lieu de naissance	lieunaiss	20	AN	
	Adresse	Adress	50	AN	
	Telephone	Tel	30	AN	
	Adresse Mail	Email	50	AN	
	Village	Village	30	AN	
	Secteur	Secteur	30	AN	
	Chefferie	Chefferie	30	AN	
Territoire	Territ	30	AN		

	Province	Prov	30	AN	
--	----------	------	----	----	--

### 1.1.2. Recensement et description des relations

#### a. Recensement des relations

L'analyse de règles de gestion nous a conduits à ressortir les relations suivantes :

- Enregistrer
- Autoriser

#### b. Description des relations

Tableau 9: Description des relations

Relation	Propriété	Dimension	Object associé
Enregistrer	Date	Ternaire	Enfant Secrétaire Tuteur
Autoriser	Date Motif Detedepart Date retour Confirm	Ternaire	Enfant Secrétaire Famille d'accueil

### 1.1.2. Présentation des contraintes de cardinalités

La Contrainte de Cardinalité caractérise le lien qui existe entre une entité et la relation sur laquelle elle est reliée.

- La borne minimale (généralement 0 ou 1) décrit le nombre minimal des occurrences d'un objet participant dans une relation ;
- La borne maximale (généralement 0 ou 1) décrit le nombre maximal des occurrences d'un objet participant dans une relation ;
- La cardinalité (1, n) signifie un ou plusieurs ;
- La cardinalité (1,1) signifie un et un seul ;
- La cardinalité (0,1) signifie zéro ou un ;
- La cardinalité (0, n) signifie zéro ou plusieurs.

### 1.1.3. Présentation des contraintes d'intégrité fonctionnelles (CIF)

D'une manière générale les contraintes d'intégrités fonctionnelles sont, en dehors d'une spécification par les cardinalités sur les pattes de la relation concernée, représentées, selon que la relation soit binaire ou n-aire, par les graphismes suivants :

Figure 15 : présentation des contraintes d'intégrité fonctionnelle

### 1.1.4. Présentation du modèle conceptuel des données

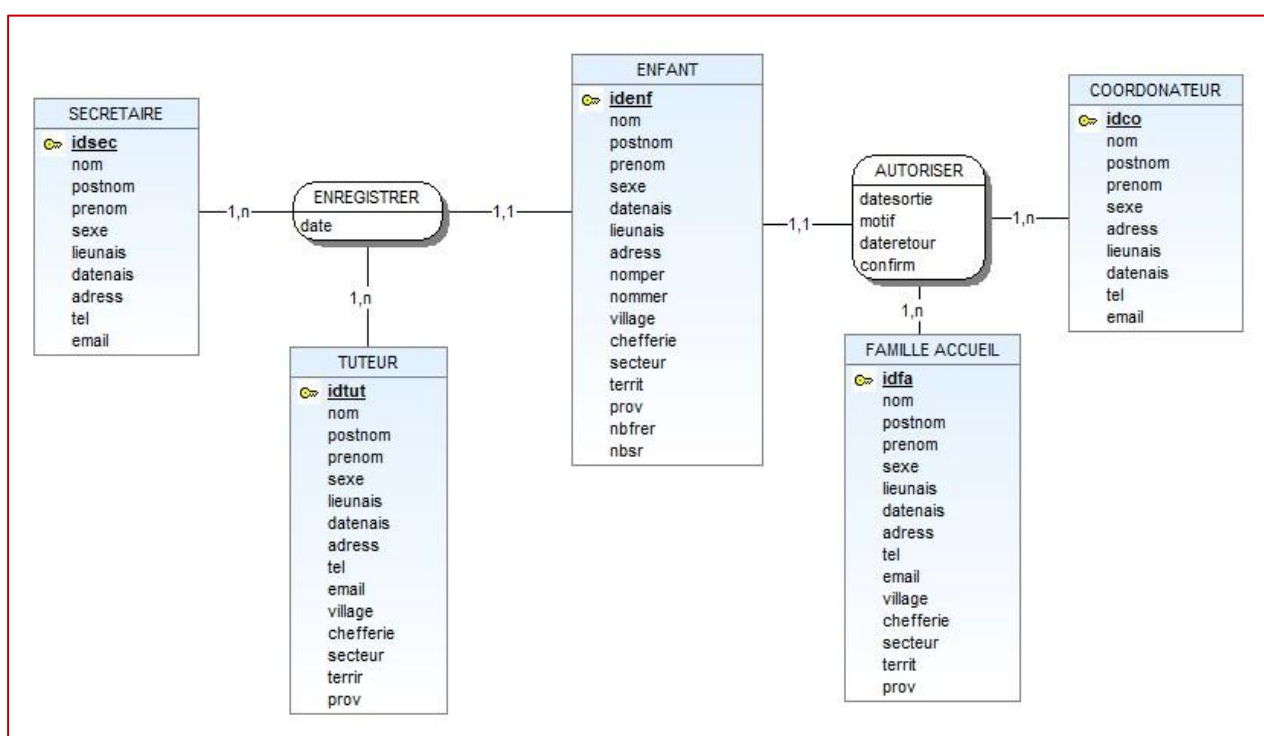


Figure 16 : présentation du MCD

### I.1.2. Modélisation Conceptuelle de Traitement (MCT)

La modélisation conceptuelle de traitement représente la dynamique du système d'information c'est-à-dire les opérations sont réalisées en fonction d'évènements sans faire référence aux choix organisationnels ou des moyens d'exécution. Il a pour but de décrire le processus de traitement sans tenir compte des moyens utilisés et détermine le processus en tenant compte de l'organisation de l'entreprise.

#### A. Définitions des concepts

❖ **Evénement** : c'est un déclencheur ou stimulus d'opération provoquant un effet

- ❖ **Opération** : c'est l'ensemble d'action déclencheur de résultats.
- ❖ **Synchronisation** : c'est une condition d'exécution d'une opération et provoqué par l'événement ;
- ❖ **Processus** : constitue un sous ensemble d'activités de la société (domaine) dont les points d'entrée et de sortie stables et indépendant du choix d'organisation.

## B. Formalisme

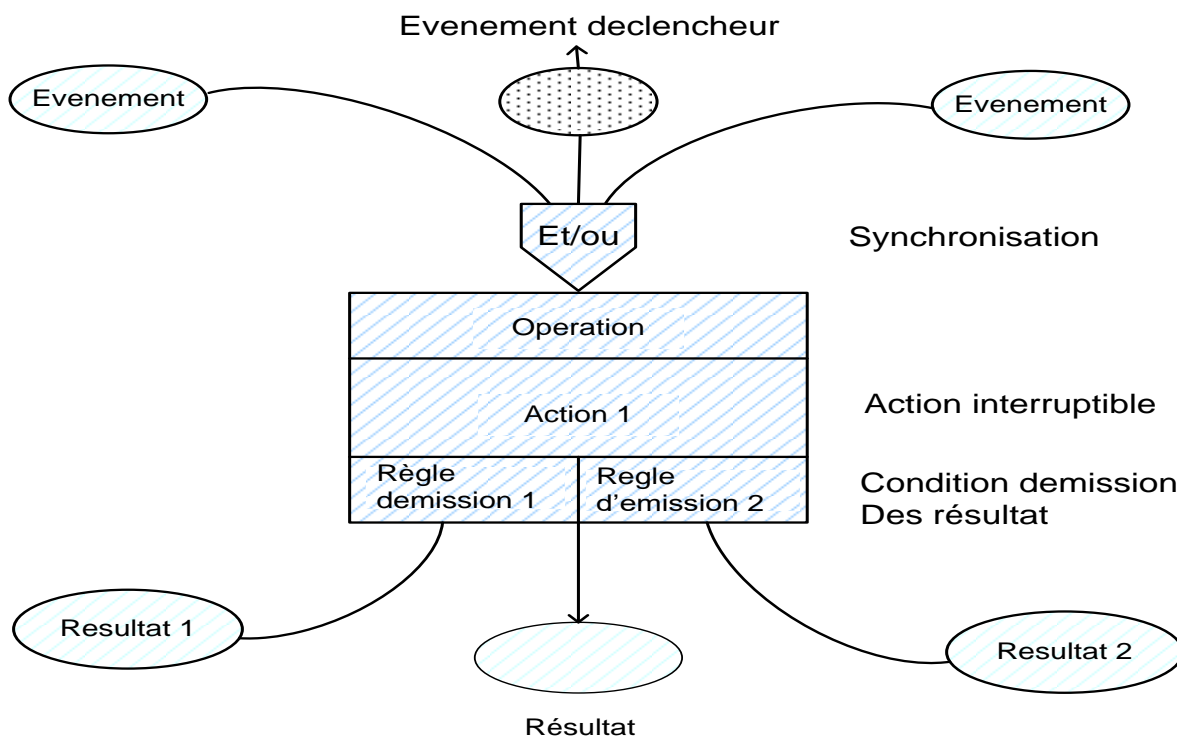
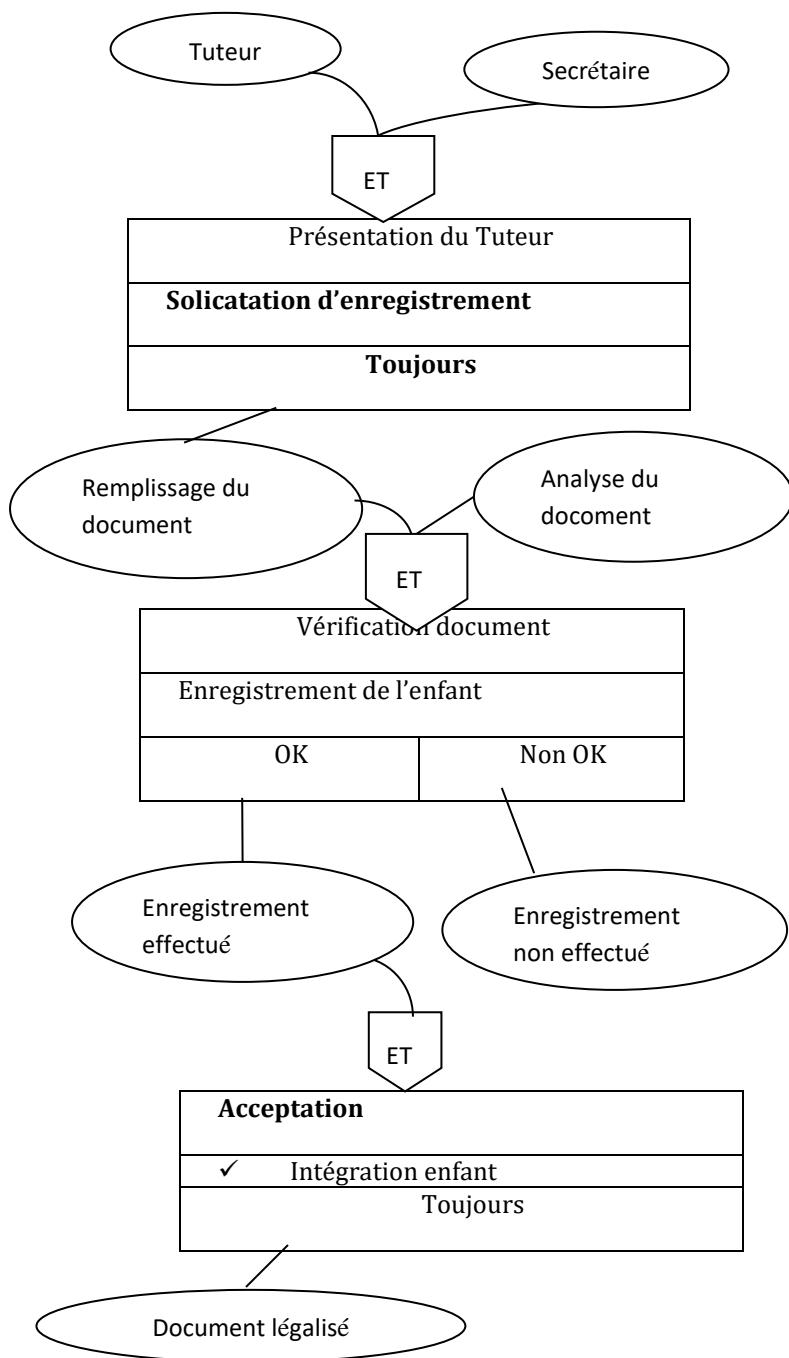


Figure 17 : formalisme du MCT

### I.1.2.1. Présentation du Modèle conceptuel de traitement



## I.2. Etape organisationnelle

### I.2.1. Modélisation organisationnelle des données

Ce modèle d'organisation de données est élaboré à partir du modèle conceptuel de données, dont on ne conserve que les objets appelés à être mémorisés informatiquement.

### 2.2.1. Règles de passage du MCD au MOD

Ce passage se fait de la manière suivante :

- ❖ La suppression des objets ou des propriétés qui ne seront pas à modéliser informatiquement ;
- ❖ La modification au besoin de certains éléments, compte tenu du choix de mémorisation informatique ;
- ❖ La création évidente des nouveaux éléments de substitution pour garder un lien avec les supprimés.

### 2.2.2. Présentation du Modèle Organisationnel de Données (MOD)

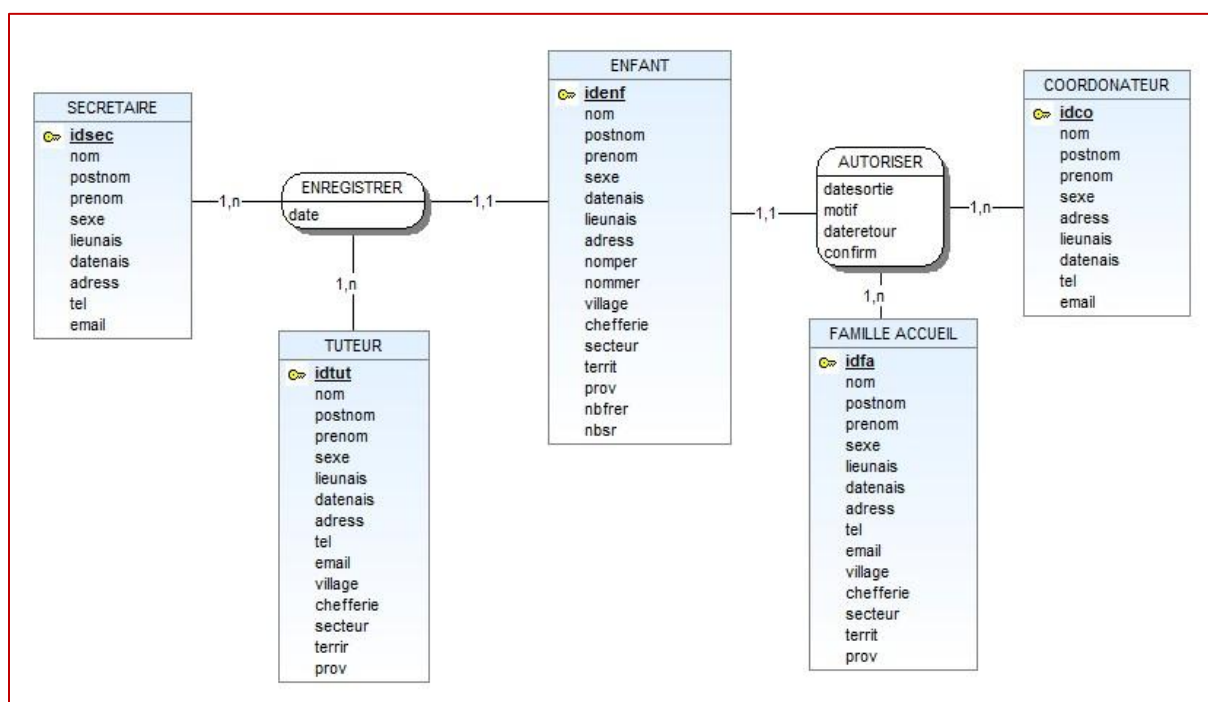


Figure 18 : Présentation du MOD

### 2.2.3. Quantification de la multiplicité des cardinalités

Par multiplicité de cardinalité, on sous-entend un couple dont la cardinalité maximale est égale à N c'est-à-dire le nombre d'occurrences maximum d'un objet participant aux occurrences d'une relation donnée. Pour ce faire, on calcule la cardinalité moyenne qui se base sur certaines variables statiques ainsi que les occurrences des objets et relations.

La cardinalité moyenne est obtenue à partir de la formule suivante :

$$Cm = [(Min + 2Mo + Max) / 4] * p$$

$$P = Mo / Max$$

- **Cm** : la cardinalité minimale
- **Max** : la cardinalité maximale
- **Mo** : la valeur modale
- **Cm** : la cardinalité moyenne
- **P** : Taux de participation.

Pour quantifier l'effectif des relations du type autre que père fils on utilise la formule :  $N3=N1*cm2$

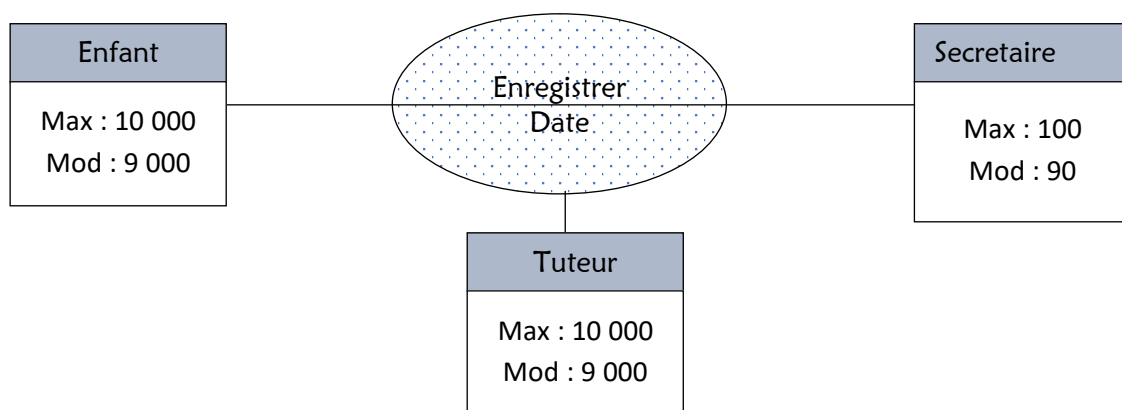
Où **N1** : Nombre d'occurrence de l'objet source

**N3** : Nombre d'occurrence de la relation

**Cm2** : Cardinalité moyenne de l'objet cible.

La connaissance des effectifs des objets et des relations nous permettra de calculer le volume approximatif du MOD global.

Le calcul de la multiplicité des cardinalités de notre MOD global se fait de la manière suivante :



$$P=Mo/Max= 9000/10000 = 0,9$$

$$Cm1= [(Min+2*Mo+Max)/4]*p$$

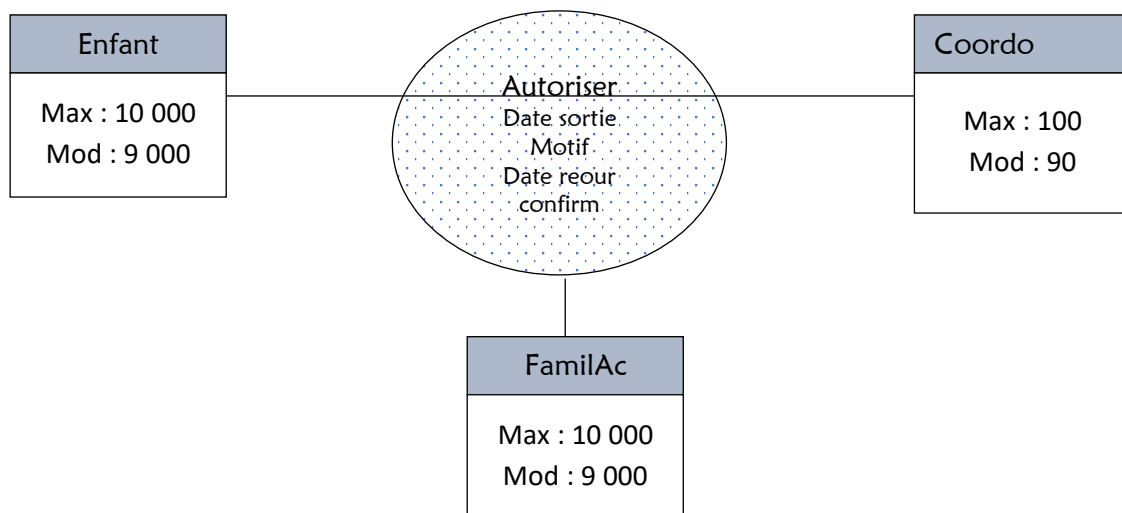
$$Cm1= [(1 + 2 * 9000 + 10 000)/4] * 0,9 \\ = 100 808$$

$$P=Mo/Max= 90/100 = 0,9$$

$$Cm2= [(Min+2*Mo+Max)/4]*p$$

$$Cm2= [(1 + 2 * 90 + 100)/4] * 0,9 = 64$$

$$N3=N1* Cm2= 10 000 * 64= 640 000$$



$$N3 = N1* Cm2 = 10 000 * 3151 = 31 510 000$$

$$P=Mo/Max= 9000/10000 = 0,9$$

$$Cm1= [(Min+2*Mo+Max)/4]*p$$

$$Cm1= [(1 + 2 * 9000 + 10 000)/4] * \\ 0,9 = 100 808$$

$$P=Mo/Max= 4 500/5 000 = 0,9$$

$$Cm2= [(Min+2*Mo+Max)/4]*p$$

$$Cm2= [(1 + 2 * 4 500 + 5 000)/4] * \\ 0,9 = 3 151$$

$$P = Mo/Max = 9000/10000 = 0,9$$

$$Cm1 = Cm2 = [(Min + 2 * Mo + Max) / 4] * p$$

$$= [(1 + 2 * 9000 + 10\ 000) / 4] * 0,9 = 100\ 808$$

$$N3 = N1 * Cm2 = 10\ 000 * 100\ 808 = 1\ 008\ 000\ 000$$

#### 2.2.4. Quantification des objets.

Tableau 10: Quantification des objets

N°	Entité	Nombre d'occurrences
1	Enfant	100808
2	Secrétaire	3151
3	Tuteur	100808
4	Coordonnateur	351
5	Famille Accueil	100808
Total		

#### 2.2.5. Quantification des propriétés

Tableau 11: Quantification des propriétés

Objets	Propriétés	Code propriété	Taille	Type	Tailles
Enfant	Identifiant Enfant	Idenf	5	AN	356
	prenom Enfant	Prenom	15	AN	
	Nom Enfant	Prenom	15	AN	
	Post-nom Enfant	Postnom	15	AN	
	Sexe	sexe	1	AN	
	date de naissance	datenaiss	15	AN	
	lieu de naissance	lieunaiss	20	AN	
	Adresse	Adress	50	AN	
	Noms du père	NomPer	30	AN	
	Noms de la mère	NomMer	30	AN	
	Village	Village	30	AN	
	Secteur	Secteur	30	AN	
	Chefferie	Chefferie	30	AN	
	Territoire	Territ	30	AN	
	Province	Prov	30	AN	



	Adresse Mail	Email	50	AN	
	Village	Village	30	AN	
	Secteur	Secteur	30	AN	
	Chefferie	Chefferie	30	AN	
	Territoire	Territ	30	AN	
	Province	Prov	30	AN	

### 2.2.6. Calcul du volume théorique du MOD

Le volume du MOD modèle organisationnel de données est l'espace qu'occupera la base de données sur un support quelconque. Ce calcul nous permet d'envisager le choix sur les types de supports que contiendront les données que nous allons enregistrer (stocker) dans notre base de données. Il est exprimé en nombre de caractères.

#### ❖ Calcul du volume des objets : volume = effectif \* taille

Tableau 12: calcul du volume des objets

Objet	Taille	Effectif	Volume
Enfant	356	10 000	1 660 000
Tuteur	366	10 000	300 000
Secrétaire	241	5 000	1 400 000
Coordonnateur	241	700	42 000
Famille Accueil	366	100	2 000
<b>Volume total objet</b>		<b>=</b>	<b>3 404 000</b>

#### ❖ Calcul du volume des relations : volume = effectif \* taille

Tableau 13: Calcul du volume des relations

Objet	Taille	Effectif	Volume
Enregistrer			
Autoriser			
<b>Volume total relation =</b>			<b>270 439 000 000</b>

### 2.2.7. Calcul du volume de la base de données

#### ❖ Calcul du volume des tables

On calcule le volume d'une table en multipliant l'effectif de cette table par sa taille. D'où :  $N \times \text{Taille}$

Tableau 14: Calcul du volume des tables

Objet	Taille (T)	Effectif (N)	Volume
Enfant	356	10 000	1 660 000
Tuteur	366	10 000	300 000
Secrétaire	241	5 000	1 400 000
Coordonnateur	241	700	42 000

Famille Accueil	366	100	2 000
Enregistrer	260	1 008 000 000	262 080 000 000
Autoriser	260	31 510 000	8 192 600 000
<b>Total volume tables =</b>			<b>270 442 404 000</b>

### ❖ Calcul du volume des index

Tableau 15: Calcul du volume des index

Table	Index	Taille	Effectif	Volume
Enfant	#idenf	5	10 000	100 000
Secrétaire	#idsec	5	100	1 000
Tuteur	#idtut	5	10 000	100 000
Famille Accueil	#idfa	5	10 000	100 000
Secrétaire	#idsec	5	100	1 000
Enregistrer	#idenf	5	31 510 000	315 100 000
	#idsec	5		
	#idtut	5		
Autoriser	#idenf	5	640 000	6 400 000
	#idco	5		
	#idfa			
<b>Total volume des index =</b>				<b>5 361 754 500</b>

### ❖ Calcul du volume de la base de données

Le calcul du volume du MOD global nous a permis de calculer l'espace approximative qu'occuperait notre base de données. Mais ici on calcul l'espace réel ou exacte qu'occupera la base de données, car à ce stade on connaît déjà la liste exhaustive des tables qui seront créer base de données relationnelle. Pour ce faire, on utilise la formule suivante :

Volume de la base de données = ( $\Sigma$ volume tables +  $\Sigma$ volume index) \* coefficient de multiplication. Le coefficient varie entre 2,5 à 3.

D'où, le volume de la base de données :  $270\,442\,404\,000 + 5\,361\,754\,500 = 2\,775\,804\,158\,500$  octets =  $2\,775\,804\,158\,500 / 1\,048\,576 = 2,64$  Mo

## I.2.2. Modélisation Organisationnel de Traitement (MOT)

Le modèle organisationnel de traitement (MOT) intégré les notions de temps et durées (déroulement) de ressources, de lieu et de responsabilité (poste de travail) et de nature de traitements (manuel ou automatique)

### I.2.2.1. Formalisme

Le MOT dérive du MCT ajoutant les différentes contraintes liées à l'organisation dont nous avons :

- ❖ Le déroulement de la tâche répondant à la question « quand ? » ;
- ❖ La nature de la tâche, répondant à la question « qui ? » ;
- ❖ Le poste de travail, répondant à la question « où ? ».

### I.2.2.2. Règles de passage du MCT au MOT

Les opérations définies dans les différents processus du MCT découpées en procédures fonctionnelles (PF) pour lesquelles on précise le déroulement de la tâche, le poste de travail et la nature de la tâche (manuel ou automatisée) pour passage au MOT, il faudrait qu'on ajoute les réponses aux questions « quand ? qui ? Et où ? »

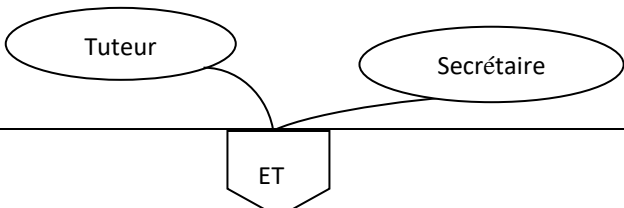
- ❖ Le déroulement de la tâche est la réponse à la question quand ? on détermine le moment ou la fréquence d'exécution de la tâche ;
- ❖ Le poste de travail est la réponse à la question où c'est l'endroit où sera exécutée la tâche ;
- ❖ La nature de tâche est la réponse relative à la question qui ? Elle implique aussi des réponses :
  - a) Si c'est l'homme, la tâche est manuelle (TM) ;
  - b) Si c'est la machine, la tâche est informatisée temps réel (TR) il faut ajouter aussi deux paramètres :

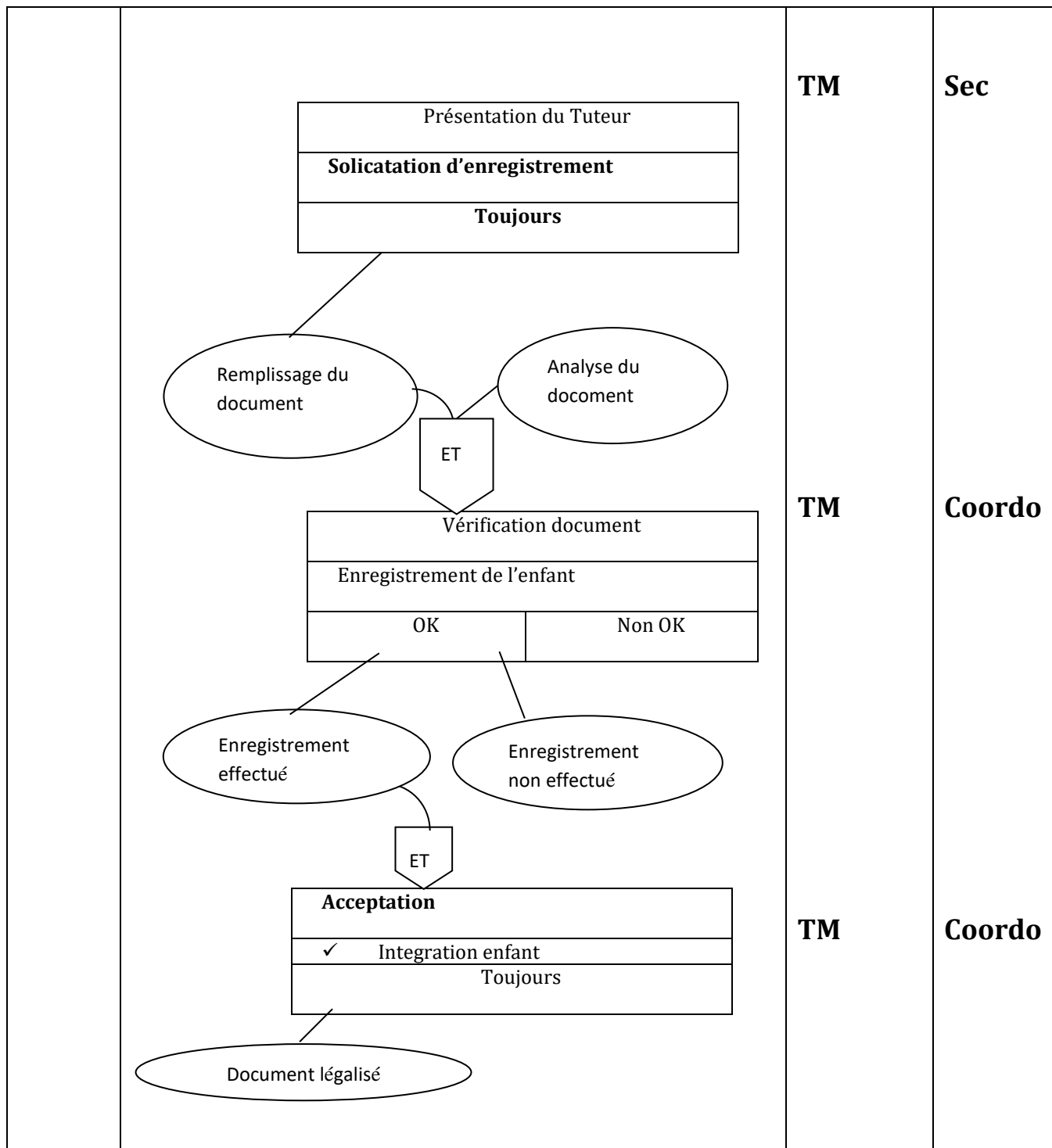
#### ➤ Le modèle fonctionnement de la tâche :

- La tâche peut être en mode unitaire (u). C'est-à-dire exécutée une à une ;
- Elle peut être en mode de traitement par lot (1) c'est-à-dire exécutée en bloc.

#### ➤ Le délai de réponse : la réponse peut être immédiate (i) ou en différé (d).

## 2.1.1. Présentation du Modèle Organisationnel de Traitement

Période	Procédure fonctionnelle	Nature	QUI ?
			



**TM**

**Sec**

**TM**

**Coordo**

**TM**

**Coordo**

## Section II : CONCEPTION DU SYSTEME D'INFORMATION INFORMATISE (CSII)

### II.1. Etape logique

#### II.1.1. Modélisation Logique de Données (MLD)

Comment? Est la question posée à cette étape pour déterminer les moyens et les ressources informatiques en faisant à l'abstraction de leurs caractéristiques techniques précises. Elle exprime la forme que doit prendre l'outil informatique pour être adapté à l'utilisation, a son poste de travail et celles-ci se fait indépendamment du langage de programmation et de système de gestion de base de données.

##### A. Vocabulaire spécifique utilisé.

- ❖ Les objets deviennent des tables
- ❖ Les propriétés deviennent des attributs
- ❖ Les identifiants deviennent des clés primaires

#### 1.1.2 Règle de passage du MOD au MLD.

Le passage du MOD global au MLD brut doit respecter les règles suivantes :

- Les objets deviennent des entités dans le sens mathématique du terme ; donc les lignes aux colonnes sur forme des tables ;
- Les propriétés des objets deviennent les attributs des tables ;
- Les identifiants des entités deviennent des clés primaires ;
- Les relations dans le sens conceptuel ou sémantique subissent plusieurs traitements selon le cas notamment :
  - La relation du type Père et Fils disparaît mais la sémantique sera maintenue. Comme la table fils dépend de la table Père, elle va recevoir la clé de son Père et cette dernière sera migrée dans la table Fils comme clé étrangère. Le fils va pointer le père ;
  - Pour des relations du type autre que Père et fils, cette relation devient une table et ses attributs seront la concaténation de deux autres tables. Si la relation portait une propriété, celle-ci demeurera dans la table comme attribut.

#### 1.1.4 Vérification et Normalisation.

Etant donné que le passage du MOD au MLD brut a permis de créer des tables qui peuvent contenir de redondances, l'opération de normalisation nous permet de pouvoir valider le MLD brut. C'est ainsi que nous allons vérifier ces objets de manière que ceux-ci aient des propriétés non répétitives. Cette opération se fait en trois formes normales à respecter afin de pouvoir valider notre MLD brut.

- ❖ **1<sup>ère</sup> forme normale :**  
Une table est à la 1<sup>ère</sup> forme normale, si elle possède une clé primaire et que ses attributs dépendent directement de la clé primaire ;
- ❖ **2<sup>ème</sup> forme normale :**

Une table est à la 2ème forme normale, étant déjà à la 1ère forme normale, ses attributs sont élémentaires c'est-à-dire non décomposables ;

❖ **3ème forme normale :**

Une table est à la 3ème forme normale, si étant déjà à la 2ème forme normale, les attributs qu'elle porte ont une dépendance fonctionnelle directe avec la clé primaire, sans passer transitivement à un autre attribut.

Dès lors que les trois formes sont respectées, les tables peuvent être déclarées normalisées et le MLD sera validé.

### 1.1.5. Présentation du modèle logique des données

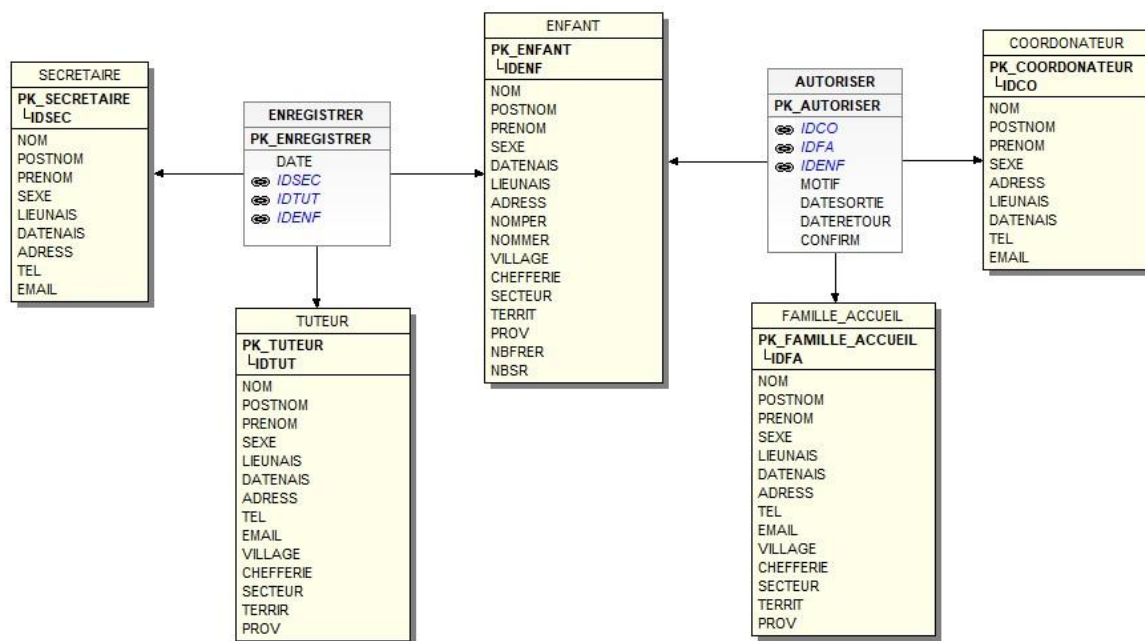


Figure 19 : Présentation du MLD

## II.1.2 Modélisation Logique de Traitement (MLT)

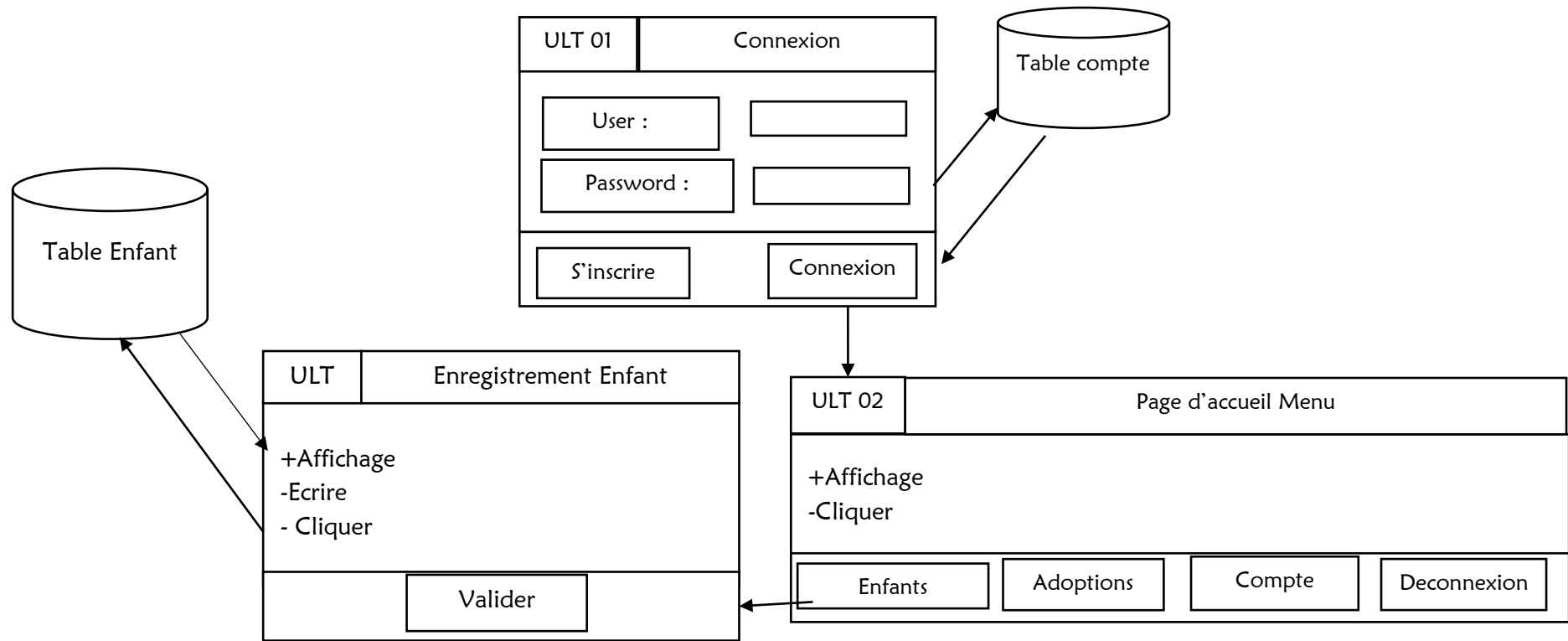
Dans le modèle logique des traitements, nous présentons les moyens que le concepteur va utiliser pour construire les programmes informatiques traduisant les solutions adoptées afin de répondre aux exigences définies au niveau organisationnel. En d'autres termes il s'agit de décrire le modèle précédent suivant un formalisme compréhensible.

### 1.2.1 Règle de passage du MOT au MLT

Ce passage concerne la réflexion et/ou l'imagination du développeur de l'application, selon sa maîtrise et/ou sa pensée dans laquelle il appliquera sur la conception de ses interfaces graphiques. Cependant, nous allons éliminer à partir du MOT les tâches qui ne seront pas informatisées et les tâches restantes. Celles-ci détermineront l'unité logique de traitement ou les événements fonctionnels disparaissent et cèdent la place aux actions des utilisateurs, notamment : clic ou clique, saisir, etc. Les tâches deviennent des unités logiques de traitement ; les procédures fonctionnelles deviennent des procédures

logiques ; l'ensemble des procédures logiques constituent le MLT et les postes de travail deviennent des sites.

### **1.2.2 Présentation du Modelé Logique de Traitement**



**Figure 20 : Présentation du MLT**

## **II.2 Etape physique**

### **2.2.1 Modélisation physique de données**

Le modèle physique de données consiste à la prise en compte des contraintes techniques liées aux matériels et aux logiciels de traitements choisis pour élaborer la solution informatique. Ainsi le modèle physique de données apparaît finalement étant celle qui prend en compte les préoccupations et choix techniques pour fournir les éléments nécessaires à l'implantation de données et la mise en place des traitements.

### **2.2.2. Règle de passage du MLD validé au MPD**

Le passage du modèle logique de données au modèle physique de données respecte le formalisme d'accès au système de gestion de base de données choisi pour la réalisation de la base de données. Les tables du modèle logique de la base de données deviennent des fichiers, les propriétés deviennent les champs, les identifiants deviennent de clés primaire (champ indexé sous doublons), les clés indexes deviennent de clés étrangères.

### **2.2.3 Présentation du MPD**

Il n'existe pratiquement pas aujourd'hui une approche normalisée pour la description et la présentation du niveau physique de données. Ce niveau est étroitement lié au choix technique informatique en rapport avec le système de gestion de base de données. Cependant, quelques règles sont à émettre selon l'orientation de choix technique effectué.

Il permet aussi de faire une description des solutions techniques retenues pour prendre en compte les aspects de performance, les conditions d'accès aux données, le mode de traitement et d'enregistrement, le matériel et logiciel choisis.

Il répond à la question « comment ? » et a pour objectif l'intégration des préoccupations et des choix techniques déterminant l'implémentation des données et la mise en place des traitements.

Ici nous allons présenter les structures des différentes tables :

#### **2. Table Enfant**

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI
idEnf	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
nom	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
postnom	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
prenom	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
sexe	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
lieunais	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
datenais	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
nomper	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
nommer	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
village	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
chefferie	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
secteur	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
territ	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
prov	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
nbfrer	INT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
nbsr	INT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

### 3. Table Tuteur

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI
idTut	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
nomt	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
postnom	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
prenom	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
sexe	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
lieunais	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
datenais	VARCHAR(25)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
adress	VARCHAR(255)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
tel	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
email	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
village	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
chefferie	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
secteur	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
territ	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
prov	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

### 4. Table Autoriser

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI
idAut	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
motif	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
datesortie	DATE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
dateretour	DATE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
confirm	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
idEnf	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
idCo	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
idFa	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

### 5. Table Famille d'accueil

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI
⚡ idFa	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
◇ nom	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
◇ postnom	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
◇ prenom	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
◇ sexe	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
◇ adress	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
◇ lieunais	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
◇ datenais	DATE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
◇ tel	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
◇ email	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
◇ village	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
◇ chefferie	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
◇ secteur	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
◇ territ	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

## 6. Table Secrétaire

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G
⚡ idSec	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
◇ nom	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
◇ postnom	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
◇ prenom	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
◇ sexe	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
◇ adress	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
◇ lieunais	VARCHAR(25)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
◇ datnais	VARCHAR(25)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
◇ tel	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
◇ email	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

## 7. Table Enregistrer

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI
⚡ idEn	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
◇ datee	VARCHAR(25)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
◇ idSec	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
◇ idTut	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
◇ idEnf	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

### 2.2.3. Modélisation Physique des Traitements

Le modèle physique de traitement (MPT) constitue l'ensemble de tous les programmes informatiques que nous allons exécuter dans l'application. Cet ensemble des programmes devra être structurés et organisés en une architecture technique de programme selon un langage de programmation spécifique. Cette technique matérialise la logique de traitement en fonction des possibilités techniques et de moyens de programmation.

Le modèle organisationnel de traitement spécifie l'enchaînement de l'ensemble des tâches à réaliser du point de vue de l'utilisateur ; le modèle logique de traitement traduit à son tour la logique informatique correspondant à l'informatisation de toutes les tâches automatiques et réelles décrites dans le modèle organisationnel de traitement. Enfin, le modèle physique de traitement nous permettra de décrire la solution technique de construction du logiciel.

Il consiste en l'écriture du programme. Celui-ci peut être généré dans le cadre d'un "atelier de génie logiciel". La finalité de la méthode MERISE est la production de "code" automatique à partir de la conception.

### 2.3.1. Passage du MLT au MPT

Le MPT s'élabore à partir du modèle logique de traitement en faisant un regroupement de toutes unités logiques en programmes. Il se présente ainsi sous forme d'une structure arborescente des programmes à réaliser

### 2.3.2. Présentation du Modèle Physique de Traitement (MPT)

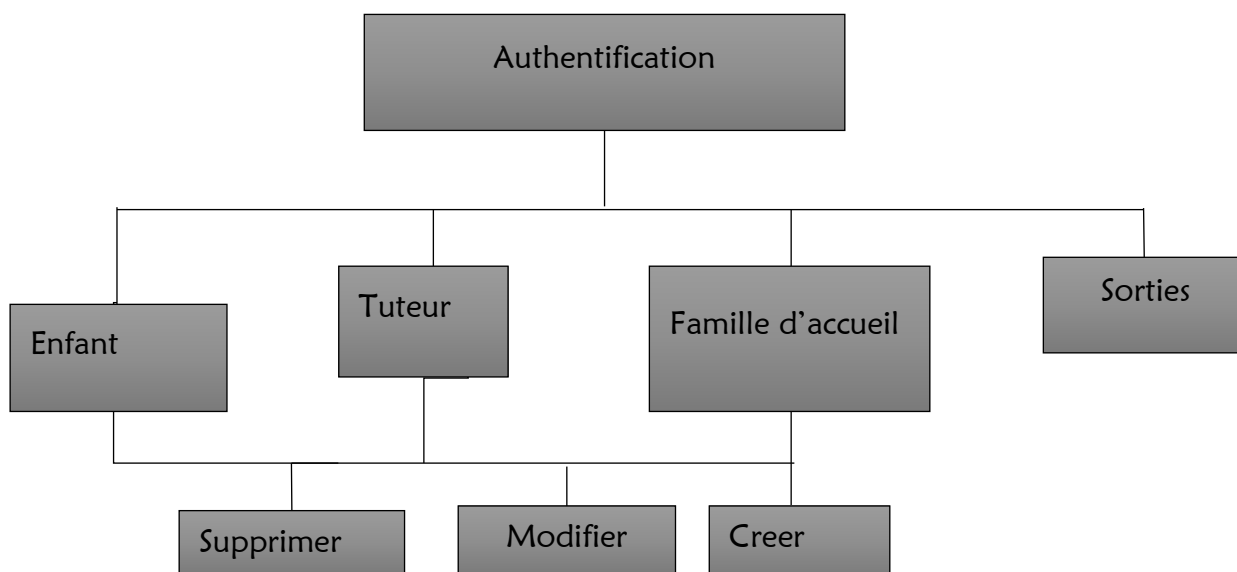


Figure 21 : Présentation du MPT

## **Conclusion Partielle**

Dans ce chapitre subdivisé en deux sections, il a été question de faire une analyse détaillée du système en utilisant la méthode MERISE et les différents modèles qu'elle nous propose.

Dans la première section intitulée Conception du Système d'Information Organisationnel, nous avons analysé le système au niveau organisationnel en passant en relevant le Modèle conceptuel, ainsi que le modèle organisationnel.

Dans la seconde section, nous avons parlé du système d'information informatisé. Cette section se focalise beaucoup plus sur les aspects logiques du système en présentant les différents modèles normalisés conduisant à la réalisation de la base de données.

## Chapitre V : MISE EN PLACE ET FONCTIONNEMENT DU LOGICIEL

### V.1. Introduction

Dans cette partie nous allons essayer de parler de tout ce qui est du point de vue de la mise en œuvre proprement dite et du fonctionnement du logiciel. Nous passerons en revue les différents outils utilisés dans la réalisation de notre application, le type de système de gestion des bases de données utilisé, les designs patterns (Patron de conception), le langage de programmation utilisé, pour enfin finir par la présentation des quelques interfaces homme-machine et quelques codes source de l'application.

### V.2. Présentation des outils et environnements de développement

#### IV.2.1. Le système de gestion de base de données (SGBD) utilisé

Nous avons jeté notre dévolu sur MySQL qui est un système de gestion de base de données relationnelles (SGBDR). Il s'agit d'un SGBDR open-source développé et supporté par Oracle, le leader mondial de la base de données. MySQL est aujourd'hui un des SGBDR les plus utilisés dans le monde. Il est donc indispensable pour un futur Data Analyst de bien comprendre son fonctionnement.

MySQL est né le 23 mai 1995. Il est développé par MySQL AB, une société suédoise fondée par David Axmark, Allan Larsson et Michael Widenius.

Son nom vient de l'association de « My » (le nom de la fille d'un des co-fondateurs) avec SQL.

Pendant les années 90, mSQL est le SGBDR le plus populaire. Il s'agit d'un système open-source qui simplifie l'utilisation des requêtes SQL.

La création de MySQL vient de l'idée de développer une version de mSQL plus rapide et plus flexible.

Ce pari est une réussite car MySQL prend rapidement la place de mSQL sur le marché des SGBDR.

MySQL connaît ensuite une série d'améliorations jusqu'au rachat de MySQL AB par Sun Microsystems en 2008. Le succès de MySQL suscite alors l'intérêt du géant de l'informatique Oracle, qui en fait l'acquisition en 2010, par le biais du rachat de Sun Microsystems.

**MySQL** est un SGBDR qui se base sur un modèle **Client-Serveur**. Ce modèle décrit la communication entre le stockage de la BDD et l'utilisateur :

- **Le serveur** correspond à l'endroit où sont stockées les **données**. Par exemple, un serveur chez un fournisseur Cloud.
- **Le Client** correspond à l'**utilisateur** qui va interroger la **base de données**. Le client va utiliser une **requête SQL** qui va créer une demande sur le **serveur**.

Nous allons implémenter cette base de données en utilisant un outil de conception graphique MySQL Workbench 8.

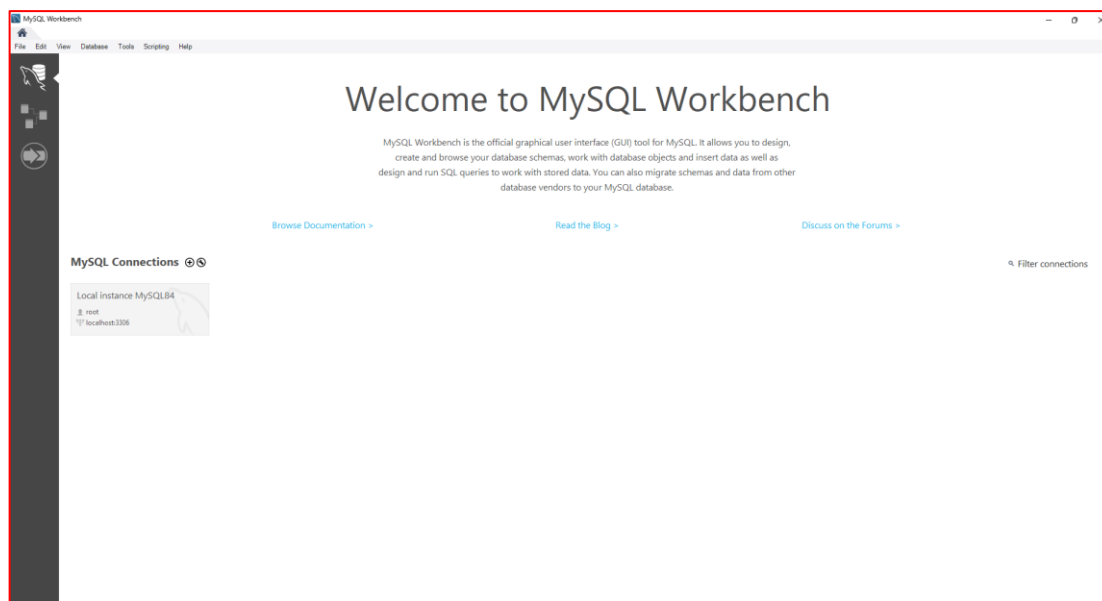


Figure 22 : Interface d'accueil MySQL Workbench 6.3

### IV.3.2. Le langage de programmation et environnement utilisé

Pour la réalisation de notre application, nous avons opté pour le langage Java et l'environnement de développement intégré (IDE) Apache NetBeans.

Le Java est un langage de programmation orienté objet, créé par Gosling et introduit par Sun Microsystems en 1995. Il est populaire pour sa portabilité, grâce au slogan « Write Once, Run Anywhere » (écrire une fois, exécuter partout). Java est utilisé pour développer des applications de bureau, mobiles, web et des systèmes embarqués. Ses principales caractéristiques incluent la gestion automatique de la mémoire, la sécurité, et un vaste écosystème de bibliothèques et frameworks.

Les avantages de Java sont nombreux. D'abord, sa portabilité : grâce à la machine virtuelle Java (JVM), vous pouvez exécuter votre code sur n'importe quelle plateforme qui a une JVM. Ensuite, sa robustesse : Java a une gestion automatique de la mémoire et un système de gestion des exceptions très développé, ce qui réduit les bugs.

De plus, sa sécurité est excellente : les applets Java ont des restrictions pour protéger l'utilisateur. Enfin, Java a une communauté énorme et des milliers de bibliothèques et frameworks qui facilitent le développement.

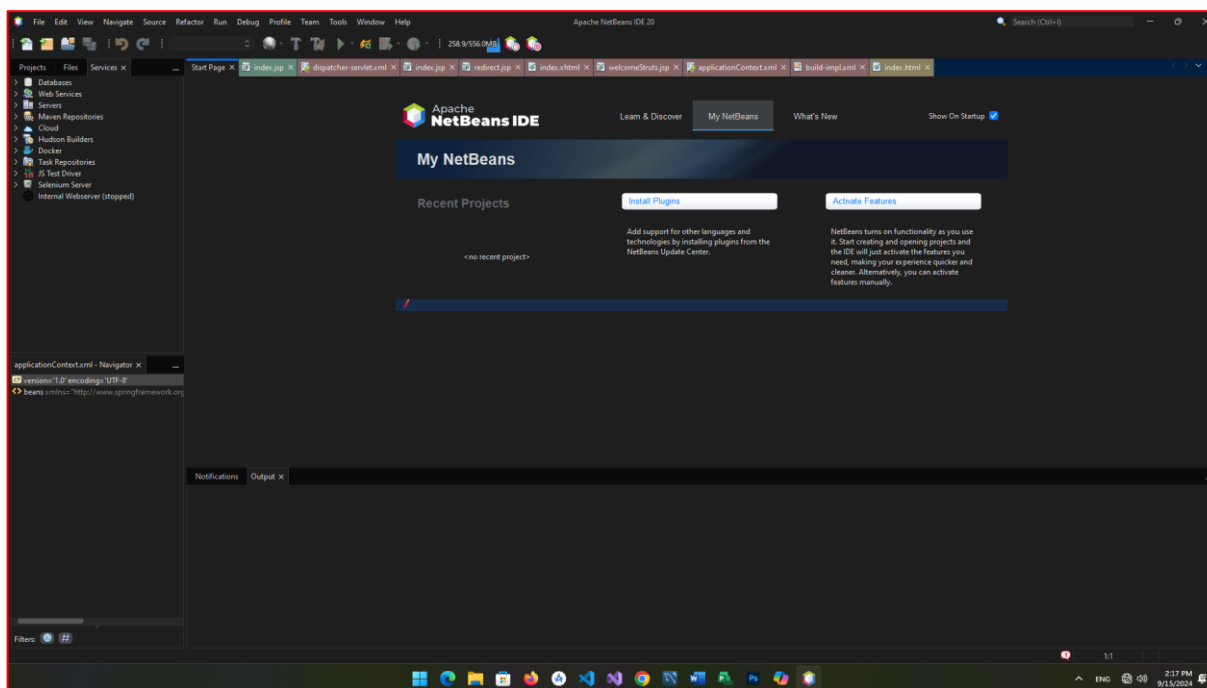


Figure 23 : Accueil NetBeans 20

### V.3.3. Les Méthodes de programmations et Modelés de conception utilisés

#### 1. La Méthode objet

Cette méthode qui consiste à créer une représentation informatique des éléments du monde réel du monde auxquels on s'intéresse, sans forcément se préoccuper de l'implémentation, ce qui signifie indépendamment d'un langage de programmation.

Il s'agit donc d'identifier les objets présents dans le système et d'isoler les données et les fonctions qu'ils utilisent. En bref, c'est la description de ce qui est représenté dans un diagramme de classe.

#### ❖ Avantages de la POO

Choisir la méthode objet comme sa méthode de programmation, présente un bon nombre d'avantage dont :

- ✓ La POO permet l'encapsulation : Le code constituant l'objet est caché de l'utilisateur (l'utilisateur dont il est question ici c'est celui qui se servira de vos objets afin d'animer le site ou l'application, et non le visiteur qui demande la page depuis son navigateur, ou simplement l'utilisateur de l'application). Cela diminue le risque d'erreurs puisque l'utilisateur ne pourra pas modifier le cœur même de votre code et évitera ainsi beaucoup d'erreurs.
- ✓ Maintenance et évolutivité : C'est-à-dire que la POO permet de concevoir une application dont les différents constituants sont indépendants les uns des autres. De ce fait, modifier un de ces constituants n'affectera pas les autres et n'entraînera donc pas d'erreurs ou autres comportements erratiques, du fait de la clarté du code vous retrouverez facilement les éléments que vous souhaitez modifier.
- ✓ Possibilité de réutilisation : L'indépendance de vos modules vous permet de les réutiliser dans d'autres applications : module de news, galerie photo : ces éléments sont présents sur une grande majorité de sites, il est donc très intéressant de pouvoir les réutiliser à chaque fois que ce sera nécessaire, sans avoir à les réinventer !

### ❖ Inconvénients de la POO

- ✓ La performance souffre parfois en utilisation intensive du polymorphisme en temps d'exécution, patrons de conception imbriqués et autres artifices POO.
- ✓ La redondance de code parfois imposée par certaines contraintes et recommandations de l'OOP. Taille de la base de code conséquente, résultat de l'accumulation des niveaux d'abstraction.
- ✓ La sur-ingénierie ou Overengineering est un piège qui guette les concepteurs (parfois expérimentés) et qui peut causer des dégâts allant du retard jusqu'à l'échec du projet.
- ✓ Exigence en savoir-faire en génie logiciel pour les projets importants (architectes, ingénieurs, et développeurs qualifiés en POO).
- ✓ Pas très pratique pour des petits projets ou test de fonctionnalités non PPO.

En somme, La réutilisation, l'encapsulation, l'extensibilité et la maintenabilité ont parfois des prix à payer : le temps et l'espace entre autres.

Il est important de notifier, qu'il existe des méthodologies pouvant être combinées à la POO afin d'accroître ces avantages en proposant une structuration bien particulière et un niveau d'abstraction et d'organisation bien plus élevé. Ces méthodologies sont connues sous le nom de « design patterns ».

## 2. Les Design patterns

Les design patterns (en français patrons de conception, Modèle de conception ou encore Motifs de conception) représentent chacun un ensemble de bonnes pratiques de conception pour un certain nombre de problème récurrents en programmation orienté objet.

Ce concept de Design patterns est issu des travaux de 4 chercheurs (Erich Gamma, Richard Helm, Ralph Johnson, et John Vlissides, groupe de quatre chercheurs connus sous le nom de « Gang of four » ou « La bande des quatre ») dans leur ouvrage intitulé « Design patterns : Elements of Reusable Object-Oriented Software » édité en 1995 et proposant 23 motifs de conception.

Il existe plusieurs types de patterns, mais dans le cadre de notre implémentation, nous n'en avons utilisés que deux, à savoir : le pattern singleton et le pattern DAO.

### ❖ Le Pattern Singleton

Le « singleton » [Gamma 95] est l'une des techniques les plus utilisées en conception orientée objet. Il permet de référencer l'instance d'une classe devant être unique par construction. Certains objets techniques prennent en effet une responsabilité particulière dans la gestion logique d'une application.

C'est par exemple le cas d'objets comme le « contrôleur des objets chargés en mémoire » ou le « superviseur des vues », qui sont les seuls et uniques représentants de leur classe. Ces objets sont le plus souvent publiquement accessibles.

De tels cas de figure sont extrêmement fréquents en conception objet, et le singleton est requis pour les concevoir.

Le singleton repose sur l'utilisation d'une opération de classe, `getInstance()` : Instance, chargée de rapporter à l'appelant la référence de l'objet unique. De plus, le singleton se charge automatiquement de construire l'objet lors du premier appel.

## ❖ Le Pattern DAO

Un objet d'accès aux données (en anglais Data Access Object) est un patron de conception (en la langue de voltaire, un modèle pour concevoir une solution) utilisé dans les architectures logicielles objet. Ce modèle permet d'encapsuler la façon dont un programme récupère les données d'un système de stockage (fichier, BDD...).

Il fait donc abstraction de la façon dont les données sont stockées au niveau des objets métier, Ainsi, le changement du mode de stockage ne remet pas en cause le reste de l'application. En effet, seules ces classes dites "techniques" seront à modifier (et donc à ré-tester). Cette souplesse implique cependant un coût additionnel, dû à une plus grande complexité de mise en œuvre.

## ❖ Le Pattern MVC

Le pattern MVC permet de bien organiser son code source. Il va vous aider à savoir quels fichiers créer, mais surtout à définir leur rôle. Le but de MVC est justement de séparer la logique du code en trois parties que l'on retrouve dans des fichiers distincts.

- **Modèle** : cette partie gère ce qu'on appelle la **logique métier** de votre site. Elle comprend notamment la gestion des données qui sont stockées, mais aussi tout le code qui prend des décisions autour de ces données. Son objectif est de fournir une interface d'action le plus simple possible au contrôleur. On y trouve donc entre autres des algorithmes complexes et des requêtes SQL.
- **Vue** : cette partie se concentre sur l'**affichage**. Elle ne fait presque aucun calcul et se contente de récupérer des variables pour savoir ce qu'elle doit afficher. On y trouve essentiellement du code HTML mais aussi quelques boucles et conditions PHP très simples, pour afficher par exemple une liste de messages.
- **Contrôleur** : cette partie gère les **échanges** avec l'utilisateur. C'est en quelque sorte l'intermédiaire entre l'utilisateur, le modèle et la vue. Le contrôleur va recevoir des requêtes de l'utilisateur. Pour chacune, il va demander au modèle d'effectuer certaines actions (lire des articles de blog depuis une base de données, supprimer un commentaire) et de lui renvoyer les résultats (la liste des articles, si la suppression est réussie). Puis il va *adapter* ce résultat et le donner à la vue. Enfin, il va renvoyer la nouvelle page HTML, générée par la vue, à l'utilisateur.

## V.4. Implémentation du système

### V.4.1. Implémentation de la base des données

#### V.4.1.1. Script de création de la base des données et ses tables

```
-- MySQL Workbench Forward Engineering
SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_
FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION';
```

```
-----
-- Schema GestEnfant
-----
```

```
CREATE SCHEMA IF NOT EXISTS `GestEnfant` DEFAULT CHARACTER SET utf8 ;
USE `GestEnfant` ;
```

```
-----
-- Table `GestEnfant`.`Secretaire`
-----
```

```
DROP TABLE IF EXISTS `GestEnfant`.`Secretaire` ;
```

```
CREATE TABLE IF NOT EXISTS `GestEnfant`.`Secretaire` (
  `idSec` INT NOT NULL AUTO_INCREMENT,
  `nom` VARCHAR(45) NULL,
  `postnom` VARCHAR(45) NULL,
  `prenom` VARCHAR(45) NULL,
  `sexe` VARCHAR(45) NULL,
  `adress` VARCHAR(45) NULL,
  `lieunais` DATE NULL,
  `datnais` DATE NULL,
  `tel` VARCHAR(45) NULL,
  `email` VARCHAR(45) NULL,
  PRIMARY KEY (`idSec`))
ENGINE = InnoDB;
```

```
-----
-- Table `GestEnfant`.`Tuteur`
-----
```

```
DROP TABLE IF EXISTS `GestEnfant`.`Tuteur` ;
```

```
CREATE TABLE IF NOT EXISTS `GestEnfant`.`Tuteur` (
  `idTut` INT NOT NULL AUTO_INCREMENT,
  `nom` VARCHAR(45) NULL,
  `postnom` VARCHAR(45) NULL,
  `prenom` VARCHAR(45) NULL,
  `sexe` VARCHAR(45) NULL,
  `lieunais` VARCHAR(45) NULL,
  `datenais` DATE NULL,
  `adress` VARCHAR(45) NULL,
  `tel` VARCHAR(45) NULL,
  `email` VARCHAR(45) NULL,
  `village` VARCHAR(45) NULL,
  `chefferie` VARCHAR(45) NULL,
  `secteur` VARCHAR(45) NULL,
  `territ` VARCHAR(45) NULL,
  `prov` VARCHAR(45) NULL,
  PRIMARY KEY (`idTut`))
ENGINE = InnoDB;
```

```
-----
-- Table `GestEnfant`.`Enfant`
-----
```

```
DROP TABLE IF EXISTS `GestEnfant`.`Enfant` ;
```

```
CREATE TABLE IF NOT EXISTS `GestEnfant`.`Enfant` (
```

```

`idEnf` INT NOT NULL AUTO_INCREMENT,
`nom` VARCHAR(45) NULL,
`postnom` VARCHAR(45) NULL,
`prenom` VARCHAR(45) NULL,
`sexe` VARCHAR(45) NULL,
`lieunais` VARCHAR(45) NULL,
`datenais` DATE NULL,
`nomper` VARCHAR(45) NULL,
`nommer` VARCHAR(45) NULL,
`village` VARCHAR(45) NULL,
`chefferie` VARCHAR(45) NULL,
`secteur` VARCHAR(45) NULL,
`territ` VARCHAR(45) NULL,
`prov` VARCHAR(45) NULL,
`nbfre` INT NULL,
`nbsr` INT NULL,
PRIMARY KEY (`idEnf`))
ENGINE = InnoDB;

-----
-- Table `GestEnfant`.`Enregistrer`
-----
DROP TABLE IF EXISTS `GestEnfant`.`Enregistrer` ;

CREATE TABLE IF NOT EXISTS `GestEnfant`.`Enregistrer` (
  `idEn` INT NOT NULL AUTO_INCREMENT,
  `date` DATE NULL,
  `idSec` INT NOT NULL,
  `idTut` INT NOT NULL,
  `idEnf` INT NOT NULL,
  PRIMARY KEY (`idEn`),
  CONSTRAINT `fk_Enregistrer_Secretaire`
    FOREIGN KEY (`idSec`)
    REFERENCES `GestEnfant`.`Secretaire` (`idSec`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_Enregistrer_Tuteur1`
    FOREIGN KEY (`idTut`)
    REFERENCES `GestEnfant`.`Tuteur` (`idTut`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_Enregistrer_Enfant1`
    FOREIGN KEY (`idEnf`)
    REFERENCES `GestEnfant`.`Enfant` (`idEnf`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

-----
-- Table `GestEnfant`.`Coordonateur`
-----
DROP TABLE IF EXISTS `GestEnfant`.`Coordonateur` ;

```

```

CREATE TABLE IF NOT EXISTS `GestEnfant`.`Coordonateur` (
  `idCo` INT NOT NULL AUTO_INCREMENT,
  `nom` VARCHAR(45) NULL,
  `postnom` VARCHAR(45) NULL,
  `prenom` VARCHAR(45) NULL,
  `sexe` VARCHAR(45) NULL,
  `adress` VARCHAR(45) NULL,
  `lieunais` VARCHAR(45) NULL,
  `datenais` DATE NULL,
  `tel` VARCHAR(45) NULL,
  `email` VARCHAR(45) NULL,
  PRIMARY KEY (`idCo`))
ENGINE = InnoDB;

```

```

-----
-- Table `GestEnfant`.`Famaccueil`
-----

```

```

DROP TABLE IF EXISTS `GestEnfant`.`Famaccueil` ;

```

```

CREATE TABLE IF NOT EXISTS `GestEnfant`.`Famaccueil` (
  `idFa` INT NOT NULL AUTO_INCREMENT,
  `nom` VARCHAR(45) NULL,
  `postnom` VARCHAR(45) NULL,
  `prenom` VARCHAR(45) NULL,
  `sexe` VARCHAR(45) NULL,
  `adress` VARCHAR(45) NULL,
  `lieunais` VARCHAR(45) NULL,
  `datenais` DATE NULL,
  `tel` VARCHAR(45) NULL,
  `email` VARCHAR(45) NULL,
  `village` VARCHAR(45) NULL,
  `chefferie` VARCHAR(45) NULL,
  `secteur` VARCHAR(45) NULL,
  `territ` VARCHAR(45) NULL,
  PRIMARY KEY (`idFa`))
ENGINE = InnoDB;

```

```

-----
-- Table `GestEnfant`.`Autoriser`
-----

```

```

DROP TABLE IF EXISTS `GestEnfant`.`Autoriser`

```

```

CREATE TABLE IF NOT EXISTS `GestEnfant`.`Autoriser` (
  `idAut` INT NOT NULL AUTO_INCREMENT,
  `motif` VARCHAR(45) NULL,
  `datesortie` DATE NULL,
  `dateretour` DATE NULL,
  `confirm` VARCHAR(45) NULL,
  `idEnf` INT NOT NULL,
  `Coordonateur_idCo` INT NOT NULL,
  `idFa` INT NOT NULL,

```

```

PRIMARY KEY (`idAut`),
CONSTRAINT `fk_Autoriser_Enfant1`
FOREIGN KEY (`idEnf`)
REFERENCES `GestEnfant`.`Enfant` (`idEnf`)
ON DELETE NO ACTION
ON UPDATE NO ACTION,
CONSTRAINT `fk_Autoriser_Coordonateur1`
FOREIGN KEY (`Coordonateur_idCo`)
REFERENCES `GestEnfant`.`Coordonateur` (`idCo`)
ON DELETE NO ACTION
ON UPDATE NO ACTION,
CONSTRAINT `fk_Autoriser_Famaccueil1`
FOREIGN KEY (`idFa`)
REFERENCES `GestEnfant`.`Famaccueil` (`idFa`)
ON DELETE NO ACTION
ON UPDATE NO ACTION)
ENGINE = InnoDB;

SET SQL_MODE=@OLD_SQL_MODE;
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;

```

#### V.4.1.2. Présentation du Schéma global de la base des données

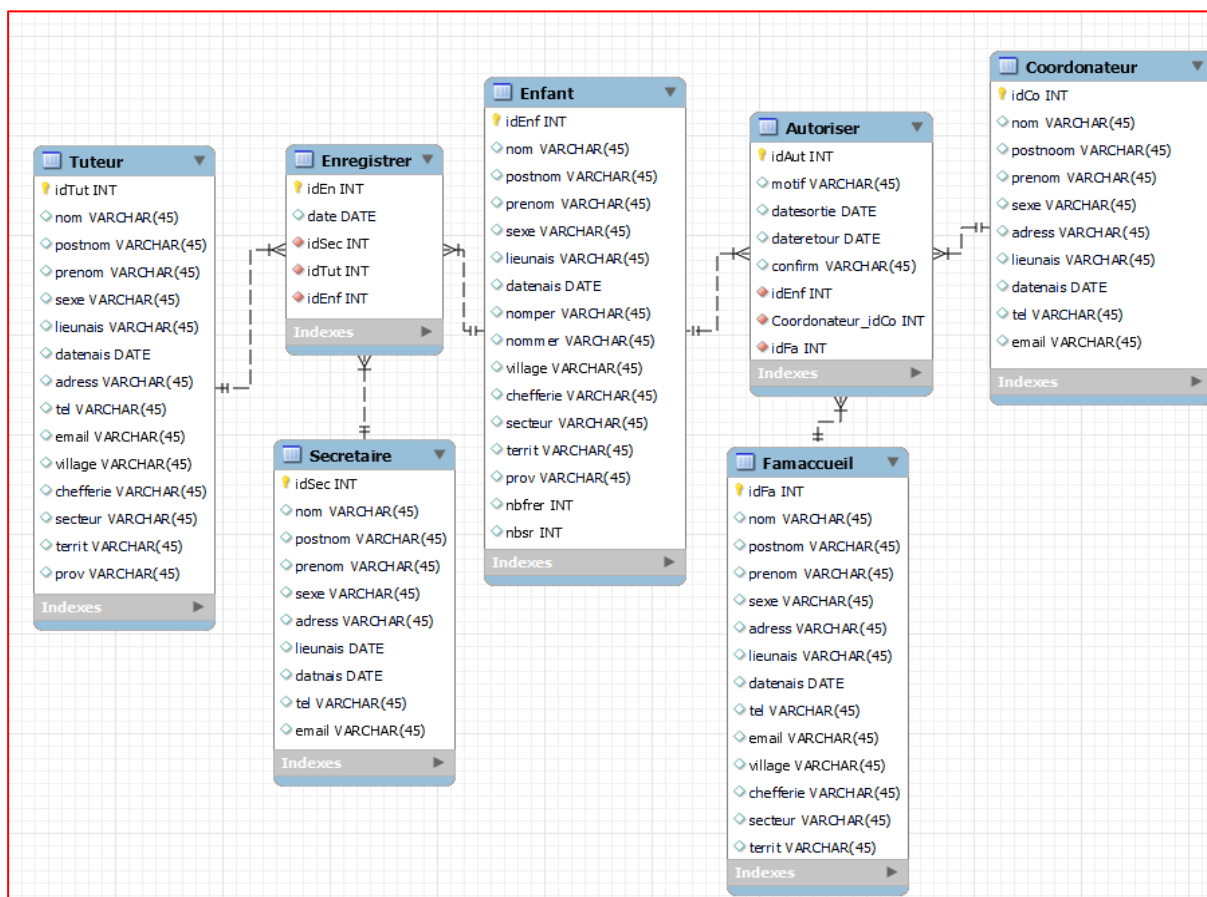


Figure 24 : Schéma global de la base des données

## V.4.2. Codage et présentation de quelques interfaces homme machine

### V.4.2.1. Présentation de quelques interfaces homme - machine



Figure 25 : Interface d'authentification

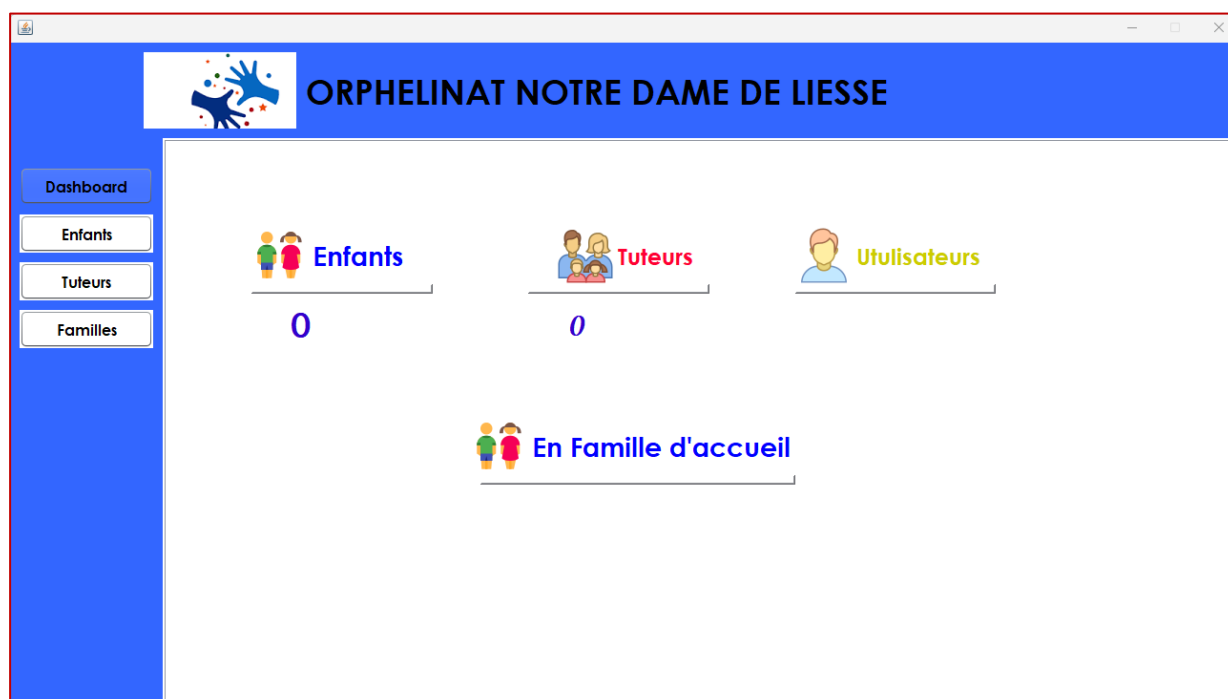


Figure 26 : Interface du tableau de bord

Figure 27 : Interface d'enregistrement d'enfants

Figure 28 : Interface d'enregistrement de tuteurs

#### V.4.2.2. Présentation de quelques codes sources

##### 1. Classe ConnectionDB

```
package data;
```

```
import java.sql.Connection;
```

```

import java.sql.DriverManager;
import java.sql.SQLException;

/**
 *
 * @author Lenovo
 */
public class ConnectionDB {

    private static Connection cnx;

    private ConnectionDB(){
        String url ="jdbc:mysql://127.0.0.1:3306/gestenfant";
        String usr = "root";
        String pswd = "felicity";

        try {
            cnx = DriverManager.getConnection(url,usr,pswd);
        }
        catch (SQLException e) {
            System.out.println(e);
        }
    }
    public static Connection getInstance(){
        if(cnx == null){
            new ConnectionDB();
        }
        return cnx;
    }
}

```

## 2. Classe EnfantController

```

package data;
import java.awt.HeadlessException;
import java.sql.SQLException;
import java.sql.Statement;
import javax.swing.JOptionPane;

/**
 *
 * @author Lenovo
 */
public class EnfantController {
    private String nom, postnom, prenom, Sex, lieu, datnais,
        nomper, nommer, village, cheff, sect, territ, prov;
    private int nbfrer, nbsr, iden;

    public EnfantController(String nom, String postnom, String prenom, String Sex, String lieu,
        String datnais, String nomper, String nommer, String village, String cheff, String sect,
        String territ, String prov, int nbfrer, int nbsr, int iden) {
        this.nom = nom;
        this.postnom = postnom;
        this.prenom = prenom;
        this.Sex = Sex;
        this.lieu = lieu;
        this.datnais = datnais;
        this.nomper = nomper;
        this.nommer = nommer;
        this.village = village;
    }
}

```

```

    this.cheff = cheff;
    this.sect = sect;
    this.territ = territ;
    this.prov = prov;
    this.nbfrer = nbfrer;
    this.nbsr = nbsr;
    this.iden = iden;
}

public EnfantController(){
}
public void setIden(int iden) {
    this.iden = iden;
}
public int getIden() {
    return iden;
}
public void setNom(String nom) {
    this.nom = nom;
}
public void setPostnom(String postnom) {
    this.postnom = postnom;
}
public void setPrenom(String prenom) {
    this.prenom = prenom;
}
public void setSex(String Sex) {
    this.Sex = Sex;
}
public void setLieun(String lieun) {
    this.lieun = lieun;
}
public void setDatnais(String datnais) {
    this.datnais = datnais;
}
public void setNomper(String nomper) {
    this.nomper = nomper;
}
public void setNommer(String nommer) {
    this.nommer = nommer;
}
public void setVillage(String village) {
    this.village = village;
}
public void setCheff(String cheff) {
    this.cheff = cheff;
}
public void setSect(String sect) {
    this.sect = sect;
}
public void setTerrit(String territ) {
    this.territ = territ;
}
public void setProv(String prov) {
    this.prov = prov;
}
public void setNbfrer(int nbfrer) {
    this.nbfrer = nbfrer;
}
public void setNbsr(int nbsr) {
    this.nbsr = nbsr;
}
public String getNom() {
    return nom;
}
public String getPostnom() {

```

```

    return postnom;
}
public String getPrenom() {
    return prenom;
}
public String getSex() {
    return Sex;
}
public String getLieun() {
    return lieu;
}
public String getDatnais() {
    return datnais;
}
public String getNomper() {
    return nomper;
}
public String getNommer() {
    return nommer;
}
public String getVillage() {
    return village;
}
public String getCheff() {
    return cheff;
}
public String getSect() {
    return sect;
}
public String getTerrit() {
    return territ;
}
public String getProv() {
    return prov;
}
public int getNbfrrer() {
    return nbfrrer;
}
public int getNbsr() {
    return nbsr;
}
}

public void creerEnf(){
    try {
        String sql = "INSERT INTO `gestenfant`.`enfant` (`nom`,`postnom`,`prenom`,`sexe`,`lieunais`,`datenais`,`nomper`,`nommer`,`village`,`chefferie`,`secteur`,`territ`,`prov`,`nbfrrer`,`nbsr`) "
            + "VALUES (" + getNom() + ", " + getPostnom() + ", " + getPrenom() + ", " + getSex() + ", " + getLieun() + ", "
            + getDatnais() + ", "
            + "" + getNomper() + ", " + getNommer() + ", " + getVillage() + ", " + getCheff() + ", " + getSect() + ", " + getTerrit() + ", "
            + "" + getProv() + ", " + getNbfrrer() + ", " + getNbsr() + ")";
        Statement st = ConnectionDB.getInstance().createStatement();
        st.executeUpdate(sql);
        JOptionPane.showMessageDialog(null, "Effectué");
    } catch (HeadlessException | SQLException e) {
    }
}
public void supprimerEnf(int id){
    try {
        Statement st = ConnectionDB.getInstance().createStatement();
        st.executeUpdate("delete from enfant where idEnf ='" + id + "'");
        //JOptionPane.showMessageDialog(null, "Suppression reussie");
    } catch (SQLException e) {
    }
}

```

```

}

public void majEnf(){

    try {
        String request = "UPDATE `gestenfant`.`enfant` "
            + "SET `nom` = '"+getNom()+"', "
            + "`postnom` = '"+getPostnom()+"', "
            + "`prenom` = '"+getPrenom()+"', "
            + "`sexe` = '"+getSex()+"', "
            + "`lieunais` = '"+getLieun()+"', "
            + "`datenais` = '"+getDatnais()+"', "
            + "`nomper` = '"+getNommer()+"', "
            + "`nommer` = '"+getNommer()+"', "
            + "`village` = '"+getVillage()+"', "
            + "`chefferie` = '"+getCheff()+"', "
            + "`secteur` = '"+getSect()+"', "
            + "`territ` = '"+getTerrit()+"', "
            + "`prov` = '"+getProv()+"', "
            + "`nbfre` = '"+getNbfre()+"', "
            + "`nbsr` = '"+getNbsr()+" "
            + "WHERE (`idEnf` = '"+getIden()+"");
        Statement st = ConnectionDB.getInstance().createStatement();
        st.executeUpdate(request);
        JOptionPane.showMessageDialog(null, "Modification effectuee");
    } catch (SQLException e) {
    }
}
}
}

```

## CONCLUSION GENERALE

Dans le cadre de ce mémoire intitulé "Mise en œuvre d'une application informatique pour la gestion des enregistrements des orphelins", nous avons exploré les différentes étapes et méthodologies nécessaires pour mener à bien un projet informatique. Nous avons notamment abordé la méthode Merise, qui s'est avérée être un outil essentiel pour la modélisation des données et des traitements, en garantissant une structure rigoureuse et cohérente du système d'information.

Le cadrage du projet et l'utilisation du diagramme de Gantt nous ont permis de bien organiser et planifier les différentes phases du développement, tout en respectant les contraintes de temps et de ressources. Cela a été crucial pour assurer la réussite du projet, en permettant une gestion efficace des tâches et des intervenants.

Enfin, le développement de l'application avec le langage Java a concrétisé ces fondements théoriques et méthodologiques. Grâce à la flexibilité et à la puissance de Java, nous avons pu mettre en place une solution informatique robuste, sécurisée et évolutive, répondant aux besoins spécifiques de la gestion des enregistrements des orphelins. Ce projet démontre non seulement la capacité à intégrer les concepts théoriques de gestion de projet avec les compétences techniques de développement, mais aussi l'importance de disposer d'un système informatisé fiable pour faciliter la gestion d'un domaine aussi sensible que celui des orphelinats.

En conclusion, cette réalisation montre la pertinence de l'approche méthodologique adoptée et la valeur ajoutée qu'une solution informatique bien conçue peut apporter à des organisations, en optimisant leur gestion et en facilitant leur quotidien.

## **BIBLIOGRAPHIE et WEBOGRAPHIE**

### **A. Ouvrages**

1. MBIKAYI MPANYA Jean Marcel, note de cours de MAI 2, G3 info, URKIM, 2017
2. BUENO, Base de données, conception Méthodique des bases de Données, éd. Ellipse, Paris 2008
3. J.A. MVIBUDULUK, LD KONKFIE IPEPE, Technique des bases de données (Décembre 2012)
4. MBIKAYI MPANYA Jean Marcel, Modéliser un système d'information avec l'approche systématique : MERISE
5. Michel GRENIE, Dictionnaire la Micro-informatique, Paris.
6. M. BOUZEGHOUB, G. GARDARIN, P. VALDURIEZ, merise éd Eyrolles, tirage, paris 1995
7. Jean F., cours de Graphe, Paris 16 février 2007, OS, inédit

### **B. Sites Web**

1. <http://commentcamarche.fr> (Septembre 2012)
2. [http://fr.wikipedia.org/wiki/merise-\(informatique\)#](http://fr.wikipedia.org/wiki/merise-(informatique)#)
3. <http://laurent-audibert.developpez.com/Cours-BD/>
4. <https://www.free-work.com/fr/tech-it/blog/actualites-informatiques/bases-de-donnees-definition-fonctionnement-et-typologie>

## LISTE DES TABLEAUX

Tableau 1: Tableau d'ordonancement des taches du projet .....	23
Tableau 2: Estimation du coup .....	25
Tableau 3: Description du diagramme de flux .....	30
Tableau 4 : Matrice des flux .....	31
Tableau 5 : Description de la carte d'identité .....	33
Tableau 6: Présentation des moyens matériels .....	38
Tableau 7: Présentation des moyens humains.....	38
Tableau 8: Description des entités .....	43
Tableau 9: Description des relations.....	45
Tableau 10: Quantification des objets.....	51
Tableau 11: Quantification des propriétés .....	51
Tableau 12: calcul du volume des objets .....	53
Tableau 13: Calcul du volume des relations .....	53
Tableau 14: Calcul du volume des tables.....	53
Tableau 15: Calcul du volume des index.....	54

## LISTE DES FIGURES ET ILLUSTRATIONS

Figure 1 : Schéma des principes généraux et fondements de la méthode MERISE.....	3
Figure 2 : Schéma de la démarche ou cycle de vie de la méthode MERISE.....	4
Figure 3 : Schéma de cycle d'abstraction.....	7
Figure 4 : Schéma de cycle d'abstraction avec quatre niveaux.....	8
Figure 5 : Schéma de cycle d'abstraction avec quatre niveaux.....	9
Figure 6 : Exemple de SGBD hiérarchique.....	15
Figure 7 : Base de données réseau.....	16
Figure 8 : <b>Exemple de SGBD Relationnel</b> .....	16
Figure 9 : Graphe Pert.....	24
Figure 10 : chemin critique.....	25
Figure 11 : Organigramme de l'orphelinat.....	27
Figure 12 : Présentation du diagramme de contexte.....	29
Figure 13 : Présentation du diagramme de flux.....	30
Figure 14 : représentation d'un objet.....	42
Figure 15 : présentation des contraintes d'intégrité fonctionnelle.....	46
Figure 16 : présentation du MCD.....	46
Figure 17 : formalisme du MCT.....	47
Figure 8 : Présentation du MOD.....	49
Figure 19 : Présentation du MLD.....	58
<b>Figure 20 : Présentation du MLT</b> .....	60
Figure 21 : Présentation du MPT.....	64
Figure 22 : Interface d'accueil MySQL Workbench 6.3.....	67
Figure 23 : Accueil NetBeans 20.....	68
Figure 24 : Schéma global de la base des données.....	74
Figure 25 : Interface d'authentification.....	75
Figure 26 : Interface du tableau de bord.....	75
Figure 8 : Interface d'enregistrement d'enfants.....	76
<b>Figure 28 : Interface d'enregistrement de tuteurs</b> .....	76

## TABLE DES MATIÈRES

EPIGRAPHE .....	i
DEDICACES .....	ii
REMERCIEMENTS.....	iii
0. INTRODUCTION GENERALE .....	1
0.1. Généralité .....	1
0.2. Problématique .....	1
0.3. Hypothèses et intérêt du sujet .....	2
0.4. Délimitation du sujet .....	2
0.5. Méthode et techniques utilisées.....	2
0.6. Difficultés rencontrées .....	4
0.7. Subdivision du travail .....	4
CHAPITRE I : CONSIDERATIONS THEORIQUES SUR LE CONCEPT DE BASE .....	1
I.1. Introduction .....	1
I.1.1. Définition opérationnelle des concepts clés utilisés.....	1
I.1.2. Etat de la question.....	2
I.2. Notions sur la méthode MERISE.....	2
I.2.1. PRINCIPES GENERAUX ET FONDEMENTS DE LA METHODE MERISE.....	2
I.2.2. LES TROIS COMPOSANTES DE LA METHODE MERISE .....	3
I.4. Notions sur les bases des données .....	10
I.4.1. Définition .....	10
I.4.2. Caractéristiques d'une base de données .....	10
I.4.3. Utilité d'une base de données .....	11
I.4.4. Avantages d'une base de données.....	11
I.4.5. Types de base de données.....	11
I.4.6. Conception d'une base de données.....	13
I.2. SYSTÈME DE GESTION DE BASE DE DONNÉES .....	13
I.2.1. Définition .....	13
I.2.2. Objectif.....	13
I.2.3. Principe de fonctionnement.....	14
I.2.4. TYPE DE SYSTEME DE GESTION DE BASE DE DONNEES.....	14
1.3. SYSTEME DE GESTION DE BASE DE DONNEES REPARTIS (SGBDR).....	16
1.3.1. DEFINITION .....	16
1.3.2. FONCTIONNEMENT.....	17
CHAPITRE II : EVALUATION DU PROJET .....	19
II.1. Introduction.....	19
II.1. Technique d'ordonnancement des tâches .....	19
II.2. Méthode d'ordonnancement des tâches.....	19
1. Le diagramme de gannt .....	20
2. Méthode CPM.....	20
3. Méthode des potentiels métra (MPM) .....	20
4. La méthode PERT .....	21
II.3. Principes de représentation en P.E.R.T .....	21
II.4. Contraintes d'un graphe P.E.R.T .....	23

II.4.1. Identification des tâches .....	23
II.4.2. Tableau d'ordonnement des tâches du projet .....	23
II.4.3. Contraintes d'élaboration du projet .....	23
II.4.3.1. Construction du graphe en P.E.R.T .....	24
II.4.3.2. Détermination du chemin critique .....	24
II.5. COUT DU PROJET .....	25
CHAPITRE III : CADRAGE DU PROJET .....	26
III.1. Présentation de l'environnement du système .....	26
III.1.1. Historique.....	26
III.1.2. Objectifs sociaux et activités principales.....	26
III.1.3. Situation géographique.....	26
III.1.4. Organigramme général.....	27
III.1.6. Organigramme de service concerne.....	28
III.1.7. Connaissance du plan directeur.....	28
III.1.5. Présentation et fonctionnement.....	28
III.1.6. Diagramme du contexte.....	28
III.1.7. Etude du réseau d'information.....	29
III.1.8. Etude des documents utilisés .....	31
III.1.9. Etude des moyens matériels et humains.....	38
III.1.10.1. Critique du système d'information existant .....	38
III.1.11. Proposition des solutions .....	39
Conclusion partielle.....	40
CHAPITRE IV : CONCEPTION DU NOUVEAU SYSTÈME.....	41
Section I : CONCEPTION DU SYSTEME D'INFORMATION ORGANISATIONNEL (CSIO)....	41
I.1. Etape conceptuelle.....	41
I.1.1. Modélisation Conceptuelle de Données (MCD) .....	41
1.1.1. Définition et But de la modélisation conceptuelle de données.....	41
A. Définition des concepts .....	41
B. Recensement et description des entités .....	43
1.1.2. Recensement et description des relations.....	45
1.1.2. Présentation des contraintes de cardinalités .....	45
1.1.3. Présentation des contraintes d'intégrité fonctionnelles (CIF).....	45
1.1.4. Présentation du modèle conceptuel des données .....	46
I.1.2. Modélisation Conceptuelle de Traitement (MCT).....	46
A. Définitions des concepts .....	46
B. Formalisme.....	47
I.2. Etape organisationnelle .....	48
I.2.1. Modélisation organisationnelle des données .....	48
2.2.1. Règles de passage du MCD au MOD .....	49
2.2.2. Présentation du Modèle Organisationnel de Données (MOD) .....	49
2.2.3. Quantification de la multiplicité des cardinalités.....	49
2.2.4. Quantification des objets. ....	51
2.2.5. Quantification des propriétés .....	51
2.2.6. Calcul du volume théorique du MOD.....	53
2.2.7. Calcul du volume de la base de données.....	53
I.2.2. Modélisation Organisationnel de Traitement (MOT) .....	54

I.2.2.1. Formalisme.....	54
I.2.2.2. Règles de passage du MCT au MOT.....	55
Section II : CONCEPTION DU SYSTEME D'INFORMATION INFORMATISE (CSII) .....	57
II.1. Etape logique.....	57
II.1.1. Modélisation Logique de Données (MLD) .....	57
A. Vocabulaire spécifique utilisé.....	57
1.1.2 Règle de passage du MOD au MLD.....	57
1.1.4 Vérification et Normalisation.....	57
1.1.5. Présentation du modèle logique des données.....	58
II.1.2 Modélisation Logique de Traitement (MLT) .....	58
1.2.1 Règle de passage du MOT au MLT.....	58
1.2.2 Présentation du Modelé Logique de Traitement.....	59
II.2 Etape physique .....	61
2.2.1 Modélisation physique de données.....	61
2.2.2. Règle de passage du MLD validé au MPD .....	61
2.2.3. Modélisation Physique des Traitements.....	63
2.3.1. Passage du MLT au MPT.....	64
2.3.2. Présentation du Modèle Physique de Traitement (MPT).....	64
Conclusion Partielle.....	65
Chapitre V : MISE EN PLACE ET FONCTIONNEMENT DU LOGICIEL.....	66
V.1. Introduction .....	66
V.2. Présentation des outils et environnements de développement.....	66
IV.2.1. Le système de gestion de base de données (SGBD) utilisé.....	66
IV.3.2. Le langage de programmation et environnement utilisé.....	67
V.4. Implémentation du système .....	70
V.4.1. Implémentation de la base des données .....	70
V.4.1.1. Script de création de la base des données et ses tables.....	70
V.4.1.2. Présentation du Schéma global de la base des données.....	74
Figure 24 : Schéma global de la base des données .....	74
V.4.2. Codage et présentation de quelques interfaces homme machine.....	75
CONCLUSION GENERALE.....	81
BIBLIOGRAPHIE et WEBOGRAPHIE .....	82
LISTE DES TABLEAUX .....	83
LISTE DES FIGURES ET ILLUSTRATIONS .....	84
TABLE DES MATIÈRES.....	85