

République Algérienne Démocratique et Populaire  
Université Abou Bakr Belkaid– Tlemcen  
Faculté des Sciences  
Département d'Informatique

Mémoire de fin d'études  
pour l'obtention du diplôme de Master en Informatique

*Option: Système d'Information et de Connaissances (S.I.C)*

## Thème

# Suivi et gestion répartie des clients d'une banque Cas de la Banque BADR

**Réalisé par :**

- Bouanani Hanane

*Présenté le 19 septembre 2013 devant le jury composé de MM.*

- **Mr BOUDEFLA Amine** (Président)
- **Mr MATALLAH Hocine** (Encadreur)
- **Mr SMAHI Mohammed Ismail** (Examineur)
- **Mme ELYEBDRI Zeyneb** (Examinatrice)

# *Remerciements*

Louange à Dieu le clément, le miséricordieux, qui m'a donné le courage et la patience de mener à bien ce travail.

J'ai le plaisir d'adresser mes vifs remerciements à tous mes enseignants pour la qualité de formation qu'ils m'ont dispensé et particulièrement mon encadreur Monsieur Matallah Houcine qui m'a été d'un véritable soutien pour la réalisation de ce travail.

Je tiens également à remercier toute la direction et le personnel de la banque de l'agriculture et du développement rural BADR de m'avoir réservé un accueil chaleureux pendant mon stage notamment mon maître de stage Melle Sirate Nouria qui m'a formé et accompagné tout au long de cette expérience professionnelle avec beaucoup de patience et de pédagogie.

Un grand merci aux membres du jury qui me font l'honneur d'examiner mon travail.

Je n'oublie pas de remercier mes parents, pour leur contribution, leur soutien et leur patience.

Ainsi qu'à toutes les personnes qui ont contribué de près ou de loin à l'élaboration de ce présent mémoire.

*Un immense merci à vous tous*

# *Dédicaces*

Louange à Allah le Miséricorde

Je dédie ce mémoire :

A mon père pour sa présence et sa générosité

A ma mère qui a éclairé mon chemin et qui m'a encouragé et soutenu tout au long de  
mes études

A mes très chers frères Hichem et Zakaria

A mes grands parents pour leur bonté et leur amour

A mes tantes et mes oncles

A mes cousins et cousines

A mes professeurs

A mes camarades de la promotion Master2 SIC 2012/2013

A tous mes ami(e)s.

Et à tous ceux qui m'ont encouragé de près ou de loin pour l'achèvement de ce  
Mémoire.

## *Résumé*

Le but de ce projet de fin d'étude est la réalisation d'un système informatique de gestion bancaire couvrant les agences de la wilaya de Tlemcen.

Notre contribution consiste à proposer une solution informatique permettant de palier aux problèmes de gestion des clients rencontrés par la banque BADR, à savoir la redondance des informations dans les documents, la perte considérable de temps et particulièrement le problème majeur dû au fait qu'un client peut avoir plusieurs numéros de comptes dans plusieurs agences différentes.

L'élaboration de ce travail a été faite avec le SGBD Oracle 10g et implémentée en langage java.

**Mot clés :** SGBD, Oracle, BADR

## *Abstract*

The goal of this final year project is the design of a management information system to be used by the banking agencies of the Tlemcen province.

Our contribution is to provide a software solution to overcome the management issues faced by the BADR bank, namely, the duplication of information and a considerable waste of time and particularly the major problem due to the fact that a client may have multiple account numbers in several different agencies.

The development of this software has been done using Oracle 10g DBMS and it has been written in Java language.

**Keyword:** DBMS, Oracle, BADR.

## *ملخص*

الغرض من هذه الدراسة هو تنفيذ نظام معلومات لإدارة بنك و التي تغطي الوكالات المصرفية في تلمسان. مساهمتنا هي توفير الحلول البرمجية للتغلب على المشاكل التي تواجهها إدارة البنك بدر، و هي تكرار المعلومات في الوثائق، هدر الوقت و خاصة المشكلة الرئيسية التي تتمثل في حقيقة أن الزبون يمكن أن يكون له حسابات متعددة في عدة وكالات مختلفة.

و قد تم تطوير هذا العمل بواسطة نظام إدارة قاعدة بيانات أوراكل 10g و يتم تنفيذ ذلك مع لغة جافا

## Table des matières

Liste des figures .....	4
Liste des tableaux.....	5
Introduction générale .....	7
I. Chapitre 1 : Bases de données réparties .....	10
I.1 Introduction .....	10
I.2 Principe des bases de données réparties.....	10
I.2.1 Définition.....	10
I.3 Avantages .....	11
I.4 Inconvénients de la répartition des données : .....	12
I.5 Les objectifs des bases de données réparties.....	12
I.6 SGBD Réparti .....	13
I.6.1 Définition d'un Système de gestion de bases de données ( DataBase Management System) .....	13
I.7 Rôle d'un SGBD .....	14
I.8 La conception des bases de données réparties .....	14
I.8.1 Méthode de conception.....	15
I.8.2 Architecture d'une base de données répartie.....	16
I.8.3 La fragmentation .....	17
I.8.4 Allocation .....	20
I.8.5 Réplication.....	20
I.9 Gestion des BBD Réparties.....	22
I.9.1 Mise à jours des bddr.....	22
I.10 Conclusion.....	23
II. Chapitre 2 : Les bases de données réparties sous Oracle .....	25
II.1 Introduction .....	25
II.2 Présentation de Oracle.....	25

II.2.1	Les outils d'administration .....	25
II.3	Création d'une base de données .....	32
II.3.1	L'assistant DBCA .....	32
II.3.2	Les utilisateurs .....	32
1-	Création d'un utilisateur.....	33
a.	Les privilèges utilisateurs .....	33
II.4	Conclusion.....	35
III.	Chapitre 3 : Présentation de la banque BADR.....	37
III.1	Introduction .....	37
III.2	Présentation de la société BADR : .....	37
III.2.1	Description de la BADR Banque .....	37
III.2.2	Structure de la BADR .....	38
III.3	Conclusion.....	43
IV.	Chapitre 4 : Analyse et Conception .....	45
IV.1	Introduction .....	45
IV.2	Analyse du besoin .....	45
IV.2.1	Étude des postes de travail.....	45
IV.2.2	Étude des documents .....	47
IV.3	Spécification du besoin .....	47
IV.4	Solution proposée.....	48
IV.5	Conception de la solution.....	48
IV.5.1	Modélisation du système .....	48
i)	Règles de passage du MCD vers le MLD .....	52
IV.5.2	Répartition de la base de données.....	53
IV.6	Conclusion.....	54
V.	Chapitre 5 : Réalisation .....	56
V.1	Introduction .....	56

V.2	Outils de développement.....	56
V.2.1	NetBeans .....	56
V.2.2	Java .....	57
V.2.3	Oracle, JDBC et NetBeans.....	57
V.3	Création des bases de données .....	57
V.4	Configuration d'Oracle .....	57
V.5	Implémentation de la base de données répartie.....	58
V.5.1	Création des Utilisateurs et des tables .....	58
V.5.2	Création des liens entre les bases.....	59
V.6	Développement de l'application .....	60
V.6.1	Les méthodes de connexion de Netbeans vers Oracle 10g.....	60
V.6.2	Fermeture d'une connexion .....	60
V.7	Présentation de l'application .....	60
V.7.1	Structure générale .....	60
V.7.2	Les interfaces de l'application .....	61
V.8	Conclusion.....	69
	Conclusion générale.....	71
	Bibliographie .....	72
	Annexe .....	75

## Liste des figures

Figure I-1: Exemple de BD Réparti .....	11
Figure I-2: Schéma d'un SGBD .....	14
Figure I-3 Approche descendante .....	15
Figure I-4 Approche ascendante .....	16
Figure I-5 Architecture d'une BDDR .....	16
Figure I-6: Fragmentation horizontale .....	18
Figure I-7: Fragmentation verticale .....	19
Figure I-8: Fragmentation mixte .....	19
Figure II-1: Assistant de configuration Oracle Net .....	29
Figure II-2 Interface en Mode ligne de commande .....	30
Figure II-3 Interface SQL*Plus Graphique .....	31
Figure II-4 Connexion via iSQL*Plus .....	31
Figure II-5: Interface iSQL*Plus .....	31
Figure III-1: Organisation générale de la BADR .....	40
Figure III-2: Organisation de la GRE .....	41
Figure III-3: Organigramme d'une Agence BADR .....	42
Figure IV-1: Organigramme du service de caisse .....	46
Figure IV-2: Le modèle conceptuel de données .....	52
Figure V-1: NetBeans IDE .....	56
Figure V-2: Oracle Net Manager .....	58
Figure V-3: Fenêtre Accueil .....	62
Figure V-4: Authentification Client .....	62
Figure V-5: Confirmer la connexion de l'utilisateur à la base .....	63
Figure V-6: Espace Agence .....	63
Figure V-7: Rechercher et gérer les clients .....	64
Figure V-8: Ajouter un client .....	65
Figure V-9: Ajouter un compte .....	65
Figure V-10: Authentification Administration .....	66
Figure V-11: Confirmer l'authentification de l'administrateur .....	66
Figure V-12: Espace administration .....	67
Figure V-13: Administration / Consultation des opérations .....	67
Figure V-14: Administration / Consultation des Clients .....	68

Figure V-15: Administration /Gestion des employés .....	69
---	----

## **Liste des tableaux**

Tableau II-1: les privilèges d'accès à une base .....	34
Tableau IV-1 : Propriétés du dictionnaire de données.....	49
Tableau IV-2: Dictionnaire de données .....	50
Tableau IV-3: Le dictionnaire de données épuré.....	51

# Introduction générale

### **Introduction générale**

La discipline « Bases de Données » (BD) a pris aujourd'hui une place essentielle dans l'informatique et plus particulièrement en gestion. Afin de coordonner son activité et d'atteindre les objectifs fixés par sa stratégie, gérer l'information est devenue fondamentale pour l'entreprise. Pour cela, il existe un grand nombre d'outils logiciels notamment les Systèmes de Gestion de Bases de Données (SGBD) qui permettent de gérer efficacement de grandes bases de données.

### **Problématique**

A présent, de nouveaux besoins sont apparus, toute organisation automatisée souhaite stocker et échanger ses informations qui sont géographiquement éloignées, ce qui rend la tâche de la collecte et du traitement d'une grande quantité d'informations dispersées très délicate, de ce fait, l'amélioration des systèmes d'informations est devenu une priorité pour les gérants des entreprises. Notre problématique est liée à l'absence de communication et coordination automatique en temps réel entre les différentes agences d'une banque parmi les banques nationales les plus importantes à savoir la BADR. Plusieurs difficultés sont générées de cette rupture entre les agences qui s'opposent au bon fonctionnement de la banque. Pour notre PFE nous nous focalisons sur la gestion et suivi des clients dû au fait qu'un client peut avoir plusieurs numéros de comptes dans plusieurs agences différentes.

### **Solution**

La solution qui s'impose est de distribuer les données et les organiser dans des bases de données sur différents sites de stockage. L'ensemble de ces sites constitue un système de bases de données réparties offrant la possibilité aux utilisateurs de manipuler les différentes bases via un réseau de manière transparente, comme dans une base de données globale.

### **Cadre d'étude**

Du 26 mai 2013 au 06 Juin 2013, nous avons effectué un stage au sein de l'entreprise BADR située à Remchi. Au cours de ce stage au service caisse, nous avons pu nous intéresser au fonctionnement d'une agence bancaire.

### Objectif de ce travail

Notre projet a pour objectif le développement d'un système d'information de gestion bancaire dont les données sont intégrées dans un environnement réparti. Le but de ce travail est d'essayer de solutionner les problèmes de localisation des données et d'exécuter la distribution des requêtes posées par la répartition d'une base de données à travers un réseau d'ordinateurs. Pour cela, Nous avons conçu et mis en œuvre une base de données répartie sous *Oracle* pour la gestion des clients d'une banque.

### Organisation du mémoire

Notre mémoire est structuré en une introduction générale, une conclusion générale et 05 chapitres comme suit :

- Dans le premier chapitre, on introduit avec un état de l'art sur les bases de données réparties ainsi que leurs avantages, leurs inconvénients, et les principes de leurs mises en œuvre. Aussi on aborde les différentes techniques de conception et de gestion des bases de données réparties et les principes de la réplication.
- Le deuxième chapitre présente les bases de données réparties sous Oracle.
- Le troisième chapitre est consacré à la présentation de l'entreprise BADR.
- Le quatrième sera consacré à l'étude de l'existant, à l'étude conceptuelle.
- Le cinquième et dernier chapitre présente la mise en place du nouveau système à l'aide de différents outils tels que *ORACLE* et Netbeans.
- On terminera par une conclusion et quelques pistes de recherches à développer dans le futur.

# Chapitre I

*Bases de données*

*réparties*

**I. Chapitre 1 : Bases de données réparties****I.1 Introduction**

Le développement des techniques informatiques depuis ces dernières années a permis d'appliquer les outils informatiques dans l'organisation des entreprises. Vu, l'immense volume de données maniées par ces dernières, la puissance des micro-ordinateurs, les performances des réseaux et la baisse considérable des coûts du matériel informatique ont permis l'apparition d'une nouvelle approche afin de remédier aux difficultés causées par la centralisation des données, et ce en répartissant les ressources informatiques tout en préservant leur cohérence.

Les bases de données réparties sont un moyen performant pour diminuer les problèmes provoqués par l'approche centralisée, mais ne restent pas sans failles.

**I.2 Principe des bases de données réparties****I.2.1 Définition**

Une base de données répartie (BDR) est une base de données dont différentes parties sont stockées sur des sites, généralement géographiquement distants, reliés par un réseau. La réunion de ces parties forme la base de données répartie.

Un système de bases de données réparti ne doit donc en aucun cas être confondu avec un système dans lequel les bases de données sont accessibles à distance (selon le principe client serveur). Il ne doit pas non plus être confondu avec un système multibase. Dans ce dernier cas, chaque utilisateur accède à différentes bases de données en spécifiant leur nom et adresse, et le système se comporte alors simplement comme un serveur de BD et n'apporte aucune fonctionnalité particulière à la répartition.

Au contraire, un système de bases de données réparti est suffisamment complet pour décharger les utilisateurs de tous les problèmes de concurrence, de fiabilité, d'optimisation de requêtes ou de transaction sur des données gérées par différents SGBD sur plusieurs sites. (Spaccapietra, Stefano, 2002)

A Titre d'exemple, une banque peut posséder des agences à Remchi et à Maghnia. Dans une BD centralisée, le siège social de la banque va gérer tous les comptes des clients et les agences devraient communiquer avec le siège social pour avoir accès aux données.

Dans une BD répartie, les informations sur les comptes sont distribuées dans les agences et celles-ci sont interconnectées (entièrement ou partiellement) afin qu'elles puissent avoir accès aux données externes (Figure 1). Cependant, la répartition de la base de données bancaire est invisible aux agences en tant qu'utilisateurs, et la seule conséquence directe pour elles est que l'accès à certaines données est beaucoup plus rapide et très fiable.

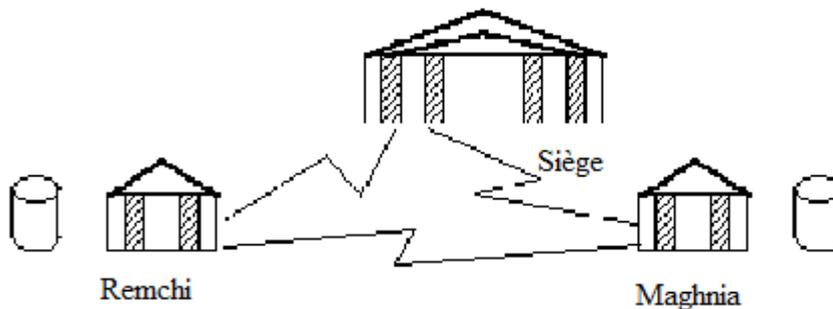


Figure I-1: Exemple de BD Réparti

### I.3 Avantages

Les bases de données réparties ont une architecture plus adaptée à l'organisation des entreprises décentralisées. (Rim Moussa, 2006)

- **Plus de fiabilité** : les bases de données réparties ont souvent des données répliquées. La panne d'un site n'est pas très importante pour l'utilisateur, qui s'adressera à un autre site.
- **Meilleures performances** : réduire le trafic sur le réseau est une possibilité d'accroître les performances. Le but de la répartition des données est de les rapprocher de l'endroit où elles sont accédées. Répartir une base de données sur plusieurs sites permet de répartir la charge sur les processeurs et sur les entrées/sorties.
- **Faciliter l'accroissement**: l'accroissement se fait par l'ajout de machines sur le réseau.

**I.4 Inconvénients de la répartition des données :**

L'inconvénient majeur de la répartition des données d'une BD entre plusieurs sites est la complexité résultant de leur coordination.

Cette complexité se répartit de la façon suivante : (Ecole de technologie supérieure)

- Le coût de mise au point du logiciel;
- Le nombre d'erreurs logicielles plus important;
- Les servitudes du système accrues pour la coordination.
  - o Echange de messages.
  - o Calcul supplémentaire.
  - o Récupération de système plus complexe après panne (Réintégration des sites ou liaison en pannes).

**I.5 Les objectifs des bases de données réparties**

Les principaux objectifs sont : (Rim Moussa, 2006)

- Transparence pour l'utilisateur.
- Autonomie de chaque site
- Absence de site privilégié
- Continuité de service
- Transparence vis à vis de la localisation des données
- Transparence vis à vis de la fragmentation
- Transparence vis à vis de la réplication
- Traitement des requêtes distribuées
- Indépendance vis à vis du matériel
- Indépendance vis à vis du système d'exploitation
- Indépendance vis à vis du réseau
- Indépendance vis à vis du SGBD

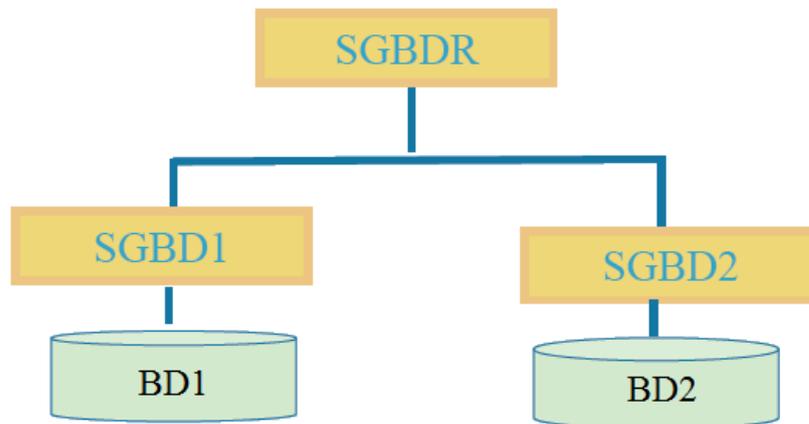
## I.6 SGBD Réparti

### I.6.1 Définition d'un Système de gestion de bases de données ( DataBase Management System)

Le SGBD (*Système de Gestion des Bases de Données*) est l'outil principal de gestion d'une base de données. Il permet d'insérer, de modifier et de rechercher efficacement des données spécifiques dans une grande masse d'informations. C'est une interface entre les utilisateurs et la mémoire de masse. Il facilite ainsi le travail des utilisateurs en leur donnant l'impression que l'information est organisée comme ils le souhaitent. Le SGBD est composé de plusieurs couches :

- Le SGBD externe (*user interface handler*). Sa tâche est d'interpréter les commandes utilisateurs.
- Le contrôleur sémantique des données (*sémantic data controler*). Il utilise les différentes contraintes définies sur la base de données afin de vérifier qu'une requête d'un utilisateur peut être effectuée.
- Le processeur de requêtes (*query processor*). Il détermine une stratégie afin de minimiser le temps d'exécution d'une requête.
- Le gestionnaire de transactions (*transaction manager*). Il assure la coordination des différentes demandes des utilisateurs.
- Le gestionnaire de reprise (*recovery manager*). Il s'occupe d'assurer la cohérence des données lorsque des pannes surviennent.
- Le système de gestion des fichiers (*run-time support processor*). Il gère le stockage physique de l'information. Il est dépendant du matériel utilisé.

Un SGBD réparti doit rendre la répartition des bases de données transparentes aux utilisateurs. La base de données étant répartie, il faut également répartir certaines fonctionnalités du SGBD. Le schéma d'un SGBD réparti est résumé dans la figure suivante. (Desfontaines, 2000)



**Figure I-2: Schéma d'un SGBD**

### **I.7 Rôle d'un SGBD**

Le logiciel de gestion d'un système de base de données (SGBD) a pour rôle :

- d'assurer la confidentialité des données : implémentation d'un mécanisme d'authentification par compte avec un mot de passe, attribution de rôles aux utilisateurs permettant d'ouvrir ou de réduire la surface d'exposition des données ;

- d'assurer la cohérence des données : vérifier les connaissances d'unicité (clés primaires) et les contraintes d'intégrité fonctionnelles ( s'assurer qu'une clé étrangère référence bien un clé primaire et que la suppression d'une clé primaire ne crée pas d'enregistrement orphelins) ;

- d'assurer la gestion des incidents : le système doit s'assurer que l'échec d'une requête ne remet pas en cause l'intégrité des données. Il doit également pouvoir procéder aux reprises sur incident suite à une panne du serveur hébergeant la base de données, par exemple.

(Michelle, et al., 2011)

### **I.8 La conception des bases de données réparties**

Une BDR reprend les mêmes principes que ceux d'une BD centralisée mais en étendant les techniques existantes ou en proposant certains concepts nouveaux qui sont particuliers à la répartition des données. (Spaccapietra, Stefano, 2002)

## I.8.1 Méthode de conception

### I.8.1.1 Conception descendante (top down design)

On commence par définir un schéma conceptuel global de la base de données répartie, puis on distribue sur les différents sites en des schémas conceptuels locaux.

La répartition se fait donc en deux étapes, en première étape la fragmentation, et en deuxième étape l'allocation de ces fragments aux sites.

L'approche *top down* est intéressante quand on part du néant.

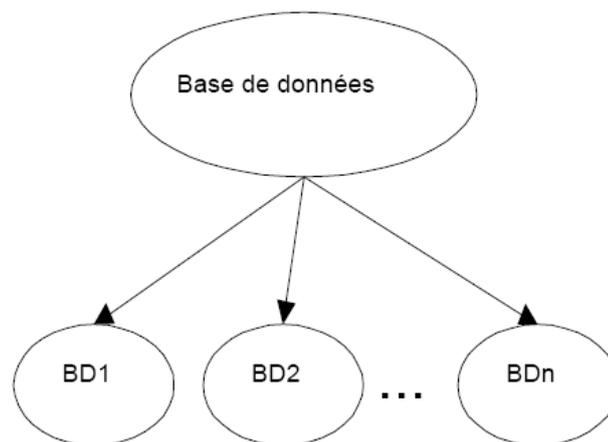


Figure I-3 Approche descendante

### I.8.1.2 Conception ascendante (bottom up design)

L'approche se base sur le fait que la répartition est déjà faite, mais il faut réussir à intégrer les différentes BDs existantes en une seule BD globale. En d'autres termes, les schémas conceptuels locaux existent et il faut réussir à les unifier dans un schéma conceptuel global. Si les BDs existent déjà la méthode *bottom up* est utilisée.

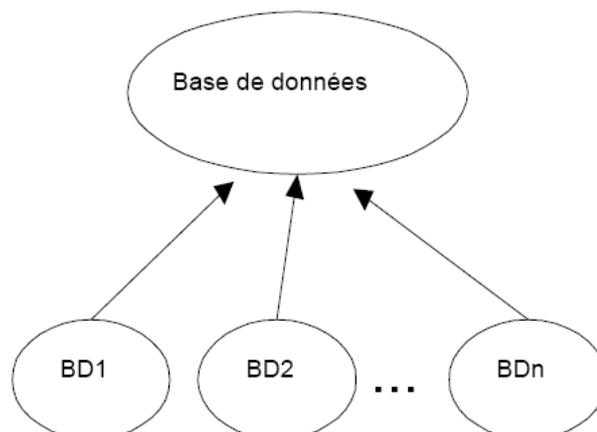


Figure I-4 Approche ascendante

## I.8.2 Architecture d'une base de données répartie

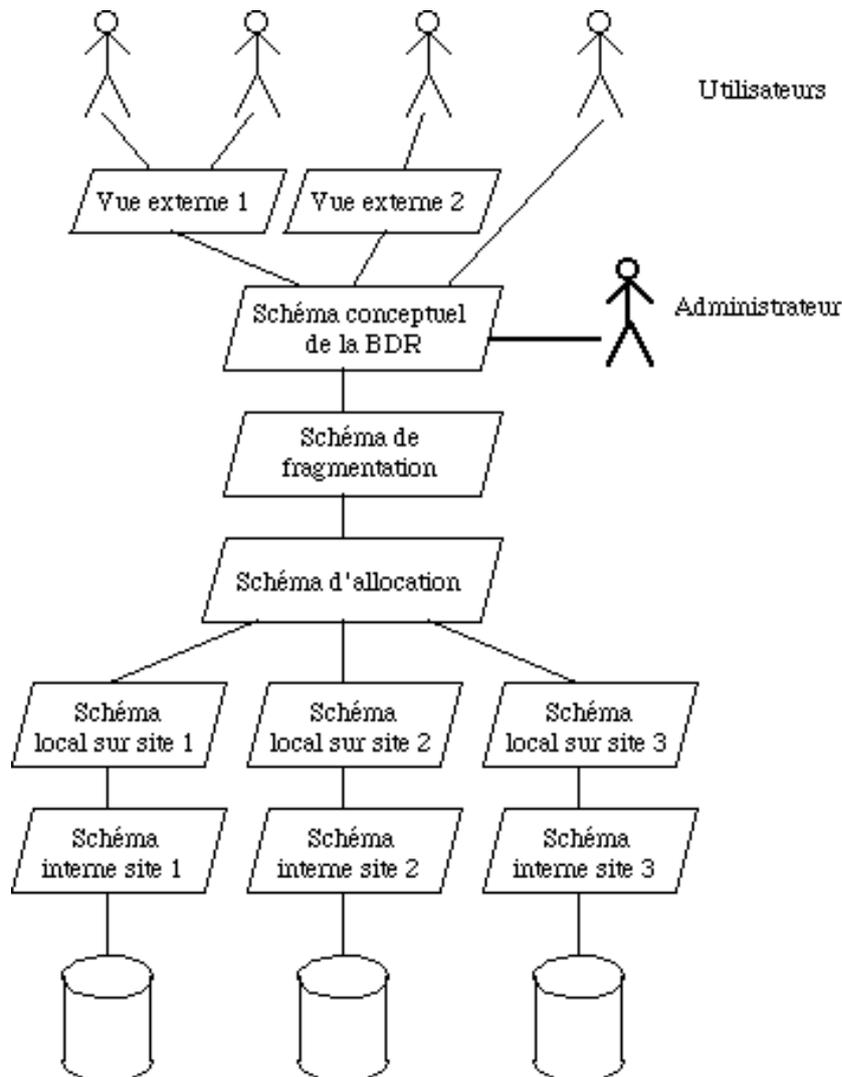


Figure I-5 Architecture d'une BDDR

- **Niveau de répartition** : La répartition d'une base de données intervient dans les trois niveaux de son architecture en plus de la répartition physique des données :
  - **Niveau externe** : les vues (schémas externes) sont distribuées aux utilisateurs sur leurs sites, les *sites utilisateurs*.
  - **Niveau conceptuel** : le schéma conceptuel des données (schéma logique) est associé, par l'intermédiaire du schéma de répartition (lui-même décomposé en un schéma de fragmentation et un schéma d'allocation), aux schémas locaux qui sont répartis sur plusieurs sites, les *sites physiques*.

- **Niveau interne** : le schéma interne global n'a pas d'existence réelle mais fait place à des schémas internes locaux répartis sur différents sites (en principe les sites d'accueil des schémas logiques répartis).
- Tous les niveaux doivent être concernés par la répartition pour que l'on puisse parler d'une vrai BD répartie

(A. ABDALLAH, 2012)

### **I.8.3 La fragmentation**

La fragmentation est le processus de décomposition d'une base de données logique (telle qu'elle est vue par les utilisateurs) en un ensemble de "sous" bases de données. Cette décomposition doit évidemment être sans perte d'information pour être acceptable. (Spaccapietra, Stefano, 2002)

#### ***I.8.3.1 Objectif de la fragmentation***

Les applications ne travaillent que sur des sous-ensembles des relations. Une distribution complète des relations générerait soit beaucoup de trafic, soit une répllication des données avec tous les problèmes que cela occasionne : problèmes de mises à jour, problèmes de stockage. Il est donc préférable de mieux distribuer ces sous-ensembles.

L'utilisation de petits fragments permet de faire tourner plus de processus simultanément, ce qui entraîne une meilleure utilisation des capacités du réseau d'ordinateurs. (Hakim MADI , 2009)

#### ***I.8.3.2 Les problèmes de la fragmentation***

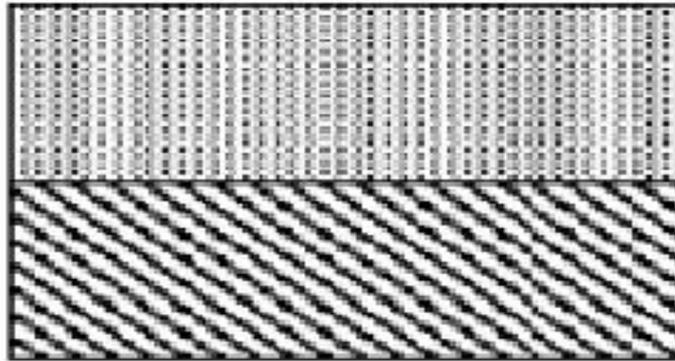
La fragmentation peut être coûteuse s'il existe des applications qui possèdent des besoins opposés. On est en quelque sorte dans le cas d'une exclusion mutuelle qui empêche une fragmentation correcte.

Par ailleurs, la vérification des dépendances sur différents sites peut être une opération très longue. (Hakim MADI , 2009)

### *1.8.3.3 Types de fragmentation*

Il existe 2 types de fragmentations :

- a. la fragmentation horizontale** : la relation est divisée en plusieurs sous-relations contenant chacune un sous-ensemble des tuples (lignes) de la relation.



**Figure I-6: Fragmentation horizontale**

- b. la fragmentation verticale** : la relation est divisée en plusieurs sous-relations contenant chacune un sous-ensemble des attributs (colonnes) de la relation.

La fragmentation horizontale a tout son intérêt pour une société dispersée aux quatre coins du globe et qui maintient une relation contenant ses employées. Cette relation logique, peut-être fragmentée horizontalement en plusieurs groupes contenant chaque fois les employés selon leur localisation.

La fragmentation verticale est plus complexe et moins intuitive. Les sous-relations ne contiennent pas tous les attributs mais tous les tuples. Un peu comme une vue permet de cacher les attributs inutiles selon le contexte, la fragmentation verticale peut-être utile d'un point de vue hiérarchique.

(Aurélien, Gillet; Vladimir, Svoboda, 2010)

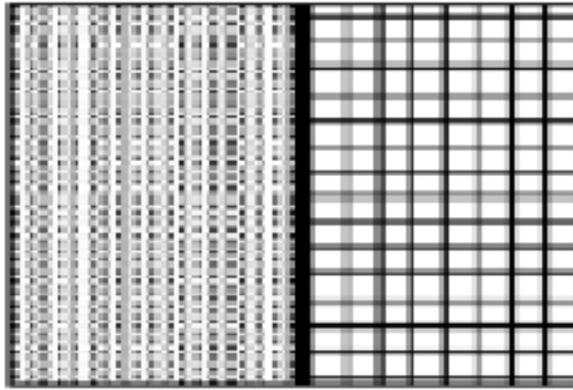


Figure I-7: Fragmentation verticale

**C. La fragmentation mixte :**

Elle résulte de l'application successive d'opérations de fragmentation horizontale et verticale sur une relation globale.

(Hakim MADI , 2009)

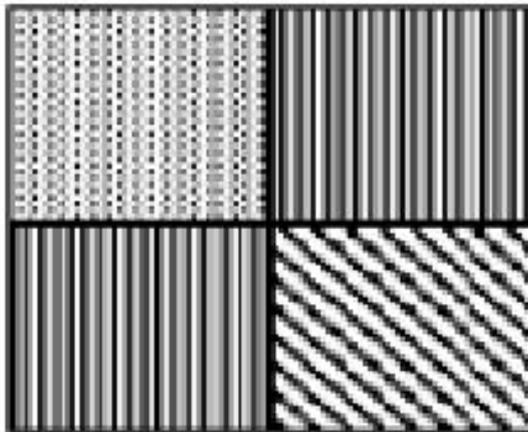


Figure I-8: Fragmentation mixte

**1.8.3.4 Les règles de la fragmentation**

Le problème qui se pose pour la fragmentation est comment définir un bon degré de fragmentation. Il existe trois règles pour la fragmentation : (Hakim MADI , 2009)

- **Complétude** : pour toute donnée d'une relation globale R, il existe au moins un fragment  $R_i$  de la relation R qui possède cette donnée.

- **Reconstruction** : pour toute relation R décomposée en un ensemble de fragments  $R_i$ , il existe une opération de reconstruction à définir en fonction de la fragmentation. Pour les fragmentations horizontales, l'opération de reconstruction est une union. Pour les fragmentations verticales c'est la jointure.
- **Disjonction** : une donnée n'est présente que dans un seul fragment, sauf dans le cas de la fragmentation verticale pour la clé primaire qui doit être présente dans l'ensemble des fragments issus d'une relation.

#### I.8.4 Allocation

L'affectation des fragments sur les sites est décidée en fonction de l'origine prévue des requêtes qui ont servi à la fragmentation. Le but est de placer les fragments sur les sites où ils sont les plus utilisés, et ce pour minimiser les transferts de données entre les sites.

L'allocation peut se faire avec réplication ou sans réplication. Sachant que la réplication favorise les performances des requêtes et la disponibilité des données, mais est coûteuse en considérant les mises à jour des fragments répliqués. (Rim Moussa, 2006)

#### I.8.5 Réplication

Certaines informations ne subissent pas souvent de modification (comme le nom de famille, l'adresse ou le nombre d'enfants des employés) et par conséquent une copie même ancienne de ces informations est, dans sa grande majorité, exacte.

La réplication consiste en l'utilisation de clichés (ang. *snapshot*). Un cliché représente un état de la base de données à un instant donné.

La pertinence d'un cliché diminue donc au fur et à mesure que le temps passe.

Les deux critères qui sont à prendre en compte pour définir l'intérêt d'un cliché sont d'une part l'ancienneté du cliché, et d'autre part le temps d'attente qui serait nécessaire avant d'obtenir l'information originale (à jour). Ces deux informations, l'ancienneté et le temps d'attente, peuvent être pondérées par un taux de satisfaction pour le système d'information. (A. ABDALLAH, 2012) (Rim Moussa, 2006)

##### I.8.5.1 Les avantages de la réplication

Les avantages de la réplication sont assez nombreux, selon le type on trouve : (Hakim MADI, 2009)

- Allègement du trafic réseau en répartissant la charge sur divers sites. Par conséquent, rapidité des accès aux données.
- Amélioration des performances des requêtes.
- Résistance aux pannes par l'augmentation de la disponibilité des données.

### ***1.8.5.2 Type de réplication***

- **Réplication asymétrique**

La réplication asymétrique distingue un site maître appelé site primaire, chargé de centraliser les mises à jour. Il est le seul autorisé à mettre à jour les données, et chargé de diffuser les mises à jour aux copies dites secondaires.

Le plus gros problème de la gestion asymétrique est la panne du site primaire. Dans ce cas, il faut choisir un remplaçant si l'on veut continuer les mises à jour. On aboutit alors à une technique asymétrique mobile dans laquelle le site primaire change dynamiquement. On distingue l'asymétrie synchrone et l'asymétrie asynchrone : (Hakim MADI , 2009)

- **Réplication asymétrique synchrone** : elle utilise un site primaire qui pousse les mises à jour en temps réel vers un ou plusieurs sites secondaires. La table répliquée est immédiatement mise à jour pour chaque modification par utilisation de trigger sur la table maître.
- **Réplication asymétrique asynchrone** : elle pousse les mises à jour en temps différé via une file persistante. Les mises à jour seront exécutées ultérieurement, à partir d'un déclencheur externe.

- **Réplication symétrique**

A l'opposé de la réplication précédente, la réplication symétrique ne privilégie aucune copie c'est-à-dire chaque copie peut être mise à jour à tout instant et assure la diffusion des mises à jour aux autres copies.

Cette technique pose problème de la concurrence d'accès risquant de faire diverger les copies. Une technique globale de résolution de conflits doit être mise en œuvre. On distingue la symétrie synchrone et la symétrie asynchrone : (Hakim MADI , 2009)

- **Réplication symétrique synchrone** : Lors de la réplication symétrique synchrone, il n'y a pas de table maître. L'utilisation de trigger sur chaque table doit différencier une mise à jour client à répercuter d'une mise à jour par réplication.
- **Réplication symétrique asynchrone** : Dans ce cas, la mise à jour des tables répliquées est différée. Cette technique risque de provoquer des incohérences de données.

## I.9 Gestion des BBD Réparties

### I.9.1 Mise à jours des bddr

La principale difficulté réside dans le fait qu'une mise à jour dans une relation du schéma global se traduit par plusieurs mises à jour dans différents fragments.

Il faut donc identifier les fragments concernés par l'opération de mise à jour, puis décomposer en conséquence l'opération en un ensemble d'opération de mise à jour sur ces fragments.

#### - **Insertion**

Retrouver le fragment horizontal concerné en utilisant les conditions qui définissent les fragments horizontaux, puis insertion du tuple dans tous les fragments verticaux correspondants.

#### - **Suppression**

Rechercher le tuple dans les fragments qui sont susceptibles de contenir le tuple concerné, et supprimer les valeurs d'attribut du tuple dans tous les fragments verticaux.

#### - **Modification**

Rechercher les tuples, les modifier et les déplacer vers les bons fragments si nécessaire.

(A. ABDALLAH, 2012)

**I.10 Conclusion**

Ainsi, se termine cette première partie qui été une présentation générale des notions de base de données répartie et ses principales caractéristiques. Dans la suite, nous allons exposer le cas particulier d'Oracle 10g qui est le SGBD le plus utilisé dans la répartition et de loin le plus efficace.

# Chapitre II

*Les bases de données  
réparties sous Oracle*

## II. Chapitre 2 : Les bases de données réparties sous Oracle

### II.1 Introduction

Dans le cadre de notre travail, on a utilisé le SGBD répartie Oracle 10g. Oracle est le système de gestion de base de données le plus utilisé dans le monde informatique et initialement basé sur le modèle relationnel. Il fonctionne de façon relativement identique sur tout type d'ordinateur.

### II.2 Présentation de Oracle

#### II.2.1 Les outils d'administration

Oracle fournit un certain nombre d'outils standards. Puissants et performants, ces outils se retrouvent sur toutes les plateformes quelque soit la version.

##### **1- Oracle net :**

Initialement appelé SQL\*Net, renommé Net8, et maintenant connu sous le nom d'Oracle Net, est l'ensemble d'outils de réseau d'Oracle qui peut être utilisé pour se connecter à des bases de données distribuées.

Oracle Net facilite le partage de données entre plusieurs bases, même si ces dernières sont hébergées sur des serveurs différents qui exécutent des systèmes d'exploitation et des protocoles de communication différents. Il permet aussi la mise en œuvre d'application trois tiers ; le serveur gère principalement les E/S de la base de données tandis que l'application est hébergée sur un serveur d'application intermédiaire et que les exigences de présentation des données de l'application sont supportées par les clients. (Razvan Bizoï, 2005)

#### **a. L'architecture d'Oracle Net**

##### **i) Architecture Client / Serveur**

Il y a plusieurs types de connexion entre le client et le serveur :

- Application basée Serveur.
- Application Client/Serveur deux niveaux.
- Application Client/Serveur trois niveaux.

- Application Serveur/Serveur.
  - ii) **Modèle OSI** : Les communications réseau dépendent de plusieurs couches de services ; un standard connu sous le nom de modèle OSI (Open System Interconnecting) met en œuvre un modèle à sept couches. Voyons les différentes couches réseau :
    - ✓ Application.
    - ✓ Présentation.
    - ✓ Session.
    - ✓ Transport.
    - ✓ Réseau.
    - ✓ Physique.

Le client et le serveur contiennent les mêmes composants. Lorsqu'une requête est emballée et envoyée d'un côté, l'autre côté doit la lire et la déchiffrer.

Si les développeurs devaient se préoccuper de chaque exécution ou variation possible de ce modèle, il serait impossible d'écrire des applications client-serveur. Ainsi Oracle a développé des composants réseau pour isoler le développeur (et l'utilisateur de l'application) des complexités de communications réseau. En employant ces composants, Oracle Net cache l'exécution et permet aux développeurs d'établir une application simple qui peut s'exécuter à travers tous les réseaux.

En employant les interfaces standards fournies par ces bibliothèques, un développeur peut être exempt de réalisations physiques. (Razvan Bizoï, 2005)

### iii) Architecture JDBC

JDBC (Java DataBase Connectivity) est une interface de programmation ou API (Application Programming Interface), qui permet d'accéder à la base et à SQL depuis Java. Il existe une API standard, mais chaque éditeur de base de données fournit aussi des pilotes additionnels offrant des fonctionnalités spécifiques à son produit. Oracle supporte trois types de pilotes JDBC. (Razvan Bizoï, 2005)

- Pilote JDBC thick (OCI Oracle Call Interface, driver lourd) requiert un code Oracle spécifique côté client.
- Pilote JDBC thin (driver léger): il est écrit entièrement en Java. Pour cette raison, il est indépendant de la plate-forme et ne requiert donc aucun code

Oracle additionnel côté client. Il est chargé dynamiquement lors de l'exécution, emploie TCP/IP, et fonctionne bien avec des applications autonomes et des pare-feu sur un intranet.

- Pilote JDBC côté serveur : Les programmes java qui s'exécutent dans le serveur de base de données utilisent un pilote JDBC côté serveur. Ils peuvent ainsi accéder directement aux données Oracle sans avoir à passer par le réseau.

Lorsque des programmes Java sont utilisés, un pilote JDBC thick ou thin communique avec Oracle Net pour traiter les requêtes. Dans le cas d'un pilote JDBC thick, ou OCI (Oracle Call Interface), Oracle Net doit être présent sur le système distant et sur le serveur de base de données.

Le type de pilote JDBC doit être défini lors de la connexion à la base de données. Les applications Java qui s'exécutent côté client peuvent employer un pilote thin ou thick. Les pilotes thick sont placés sur le niveau intermédiaire lorsqu'il existe des serveurs d'applications. Un pilote côté serveur peut uniquement s'exécuter dans la base Oracle.

### **b. Le processus de connexion**

Un utilisateur qui veut se connecter à une base Oracle à travers le réseau doit utiliser un identifiant, un mot de passe et une chaîne de connexion à la base de données.

La connexion est établie en cinq étapes : (Razvan Bizoï, 2005)

- 1) Les sessions réseau sont établies via un processus d'écoutes LISTENER, qui est un processus séparé résidant sur le serveur de base de données. Le module d'écoute reçoit les demandes de connexion client entrantes et gère leur acheminement vers le serveur.

L'étape 1 consiste en l'envoi d'une chaîne de connexion à travers le réseau au processus LISTENER.

La chaîne de connexion comporte :

- Le nom d'utilisateur.
- Le mot de passe.
- Le descripteur de connexion ( connect descriptor).

- 2) Le module d'écoute (LISTENER) prend en charge la demande du client en la transmettant au serveur. Chaque fois qu'un client demande une session réseau au serveur, un module d'écoute reçoit la demande. Si les informations du client correspondent à celles du module d'écoute, ce dernier autorise une connexion au serveur.
- 3) Le processus serveur dédié redemande au client le renvoi de la chaîne de connexion.
- 4) Le processus serveur valide ou rejette la demande de connexion.
- 5) Le processus serveur notifie soit la connexion soit l'abandon de la session.

### **c. Configuration du client**

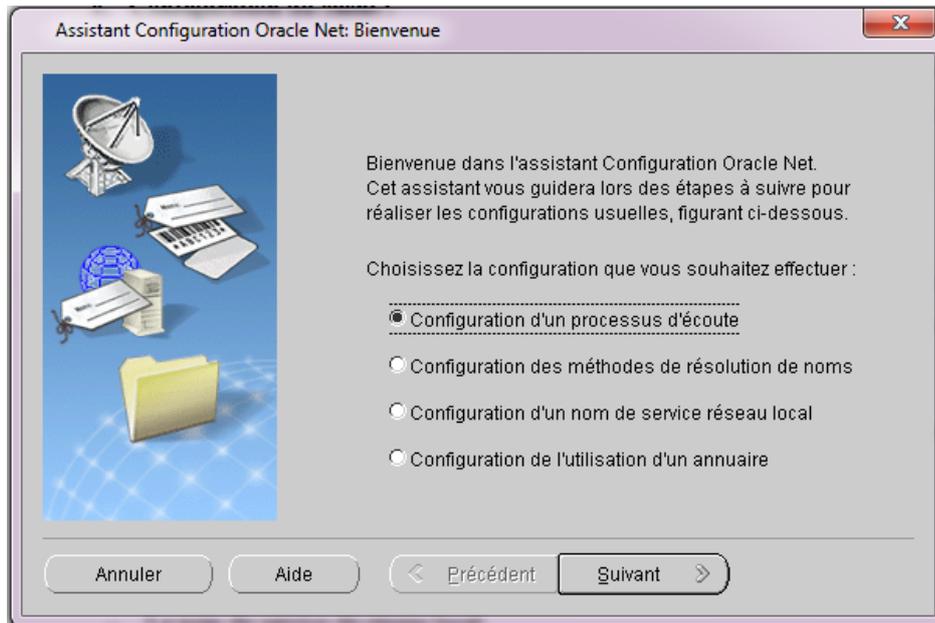
Oracle Net se fonde sur la technologie TNS (Transparent Network Substrate) pour assurer la connectivité de niveau serveur, et s'appuie sur les fichiers de configuration du client et du serveur pour gérer la connectivité de la base.

### **d. Assistant de configuration Oracle Net**

L'assistant de configuration Oracle Net exécute les étapes de configuration initiales du réseau après l'installation d'oracle et crée automatiquement les fichiers de configuration de base, par défaut. (Razvan Bizoï, 2005)

Il dispose d'une interface utilisateur graphique pour la configuration des éléments suivant :

- Le processus d'écoute LISTENER.
- Les méthodes de résolution de noms.
- Le nom de service de réseau local.
- L'utilisation d'un annuaire.



**Figure II-1: Assistant de configuration Oracle Net**

## **2- SQL\*Plus**

C'est l'outil de prédilection de l'administrateur ; il permet de manipuler les données de la base de données et aussi d'effectuer les tâches d'administration de la base. C'est un outil en ligne de commande très utile pour les tâches répétitives d'exécution des scripts et batch. (Razvan Bizoï, 2005)

Les interfaces SQL\*Plus permettent de dialoguer avec la base de différentes manières : (Christian, août 2011)

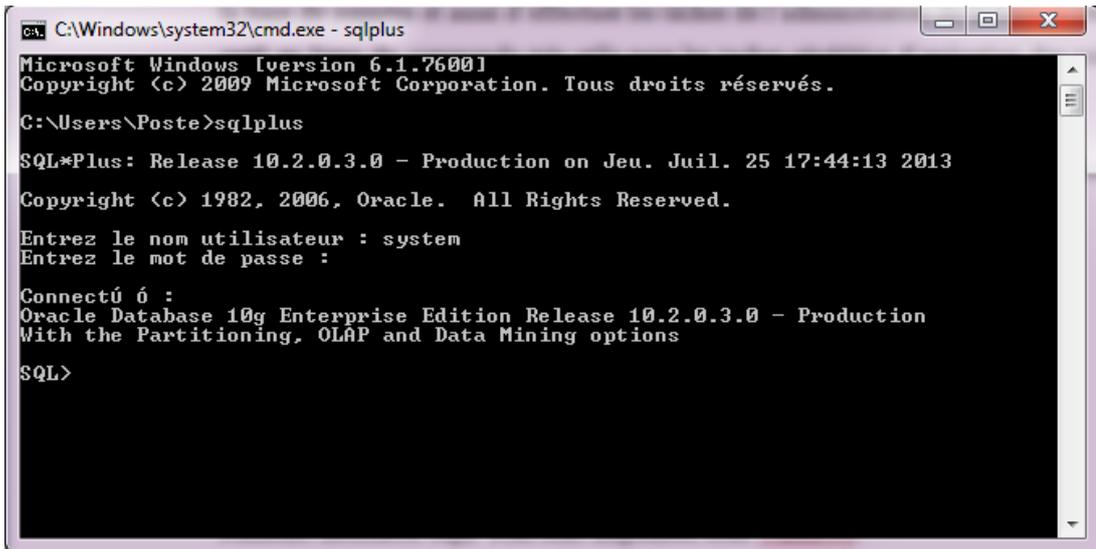
- Exécution de commande SQL, SQL\*Plus et de blocs PL/SQL ;
- Echange de messages avec d'autres utilisateurs ;
- Création de rapports d'impression en incluant des calculs ;
- Réalisation des tâches d'administration en ligne.

### **a- Les interfaces SQL\*Plus**

Plusieurs interfaces SQL\*Plus sont disponibles sous windows : (Christian, août 2011)

- En mode ligne de commande (qui ressemble à une fenêtre DOS ou telnet) :

Dans la fenêtre de commande, on lance sqlplus. Un nom d'utilisateur et un mot de passe sont demandés.



**Figure II-2 Interface en Mode ligne de commande**

- Avec l'interface graphique : SQL\*Plus graphique :

Cette interface est la plus pratique, car elle permet facilement de Copier/Coller des blocs instruction SQL et PL/SQL.

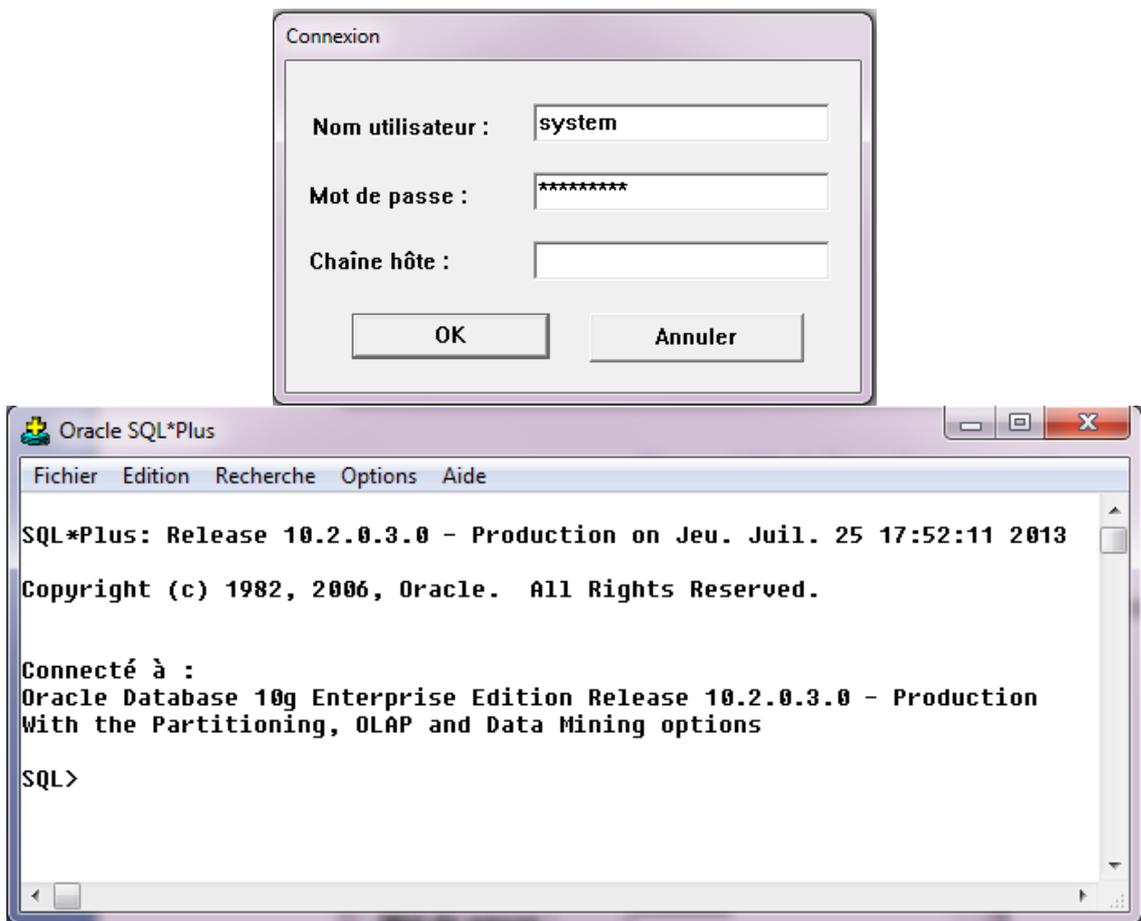


Figure II-3 Interface SQL\*Plus Graphique

- Avec le navigateur via l'interface web iSQL\*Plus (i comme « Internet ») :

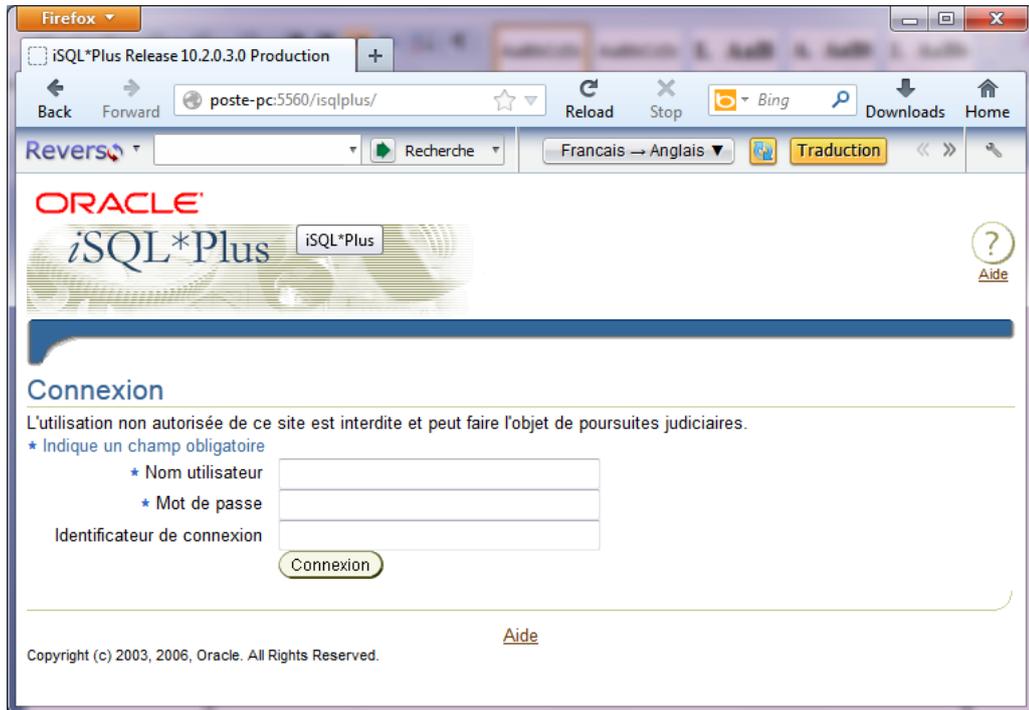


Figure II-4 Connexion via iSQL\*Plus

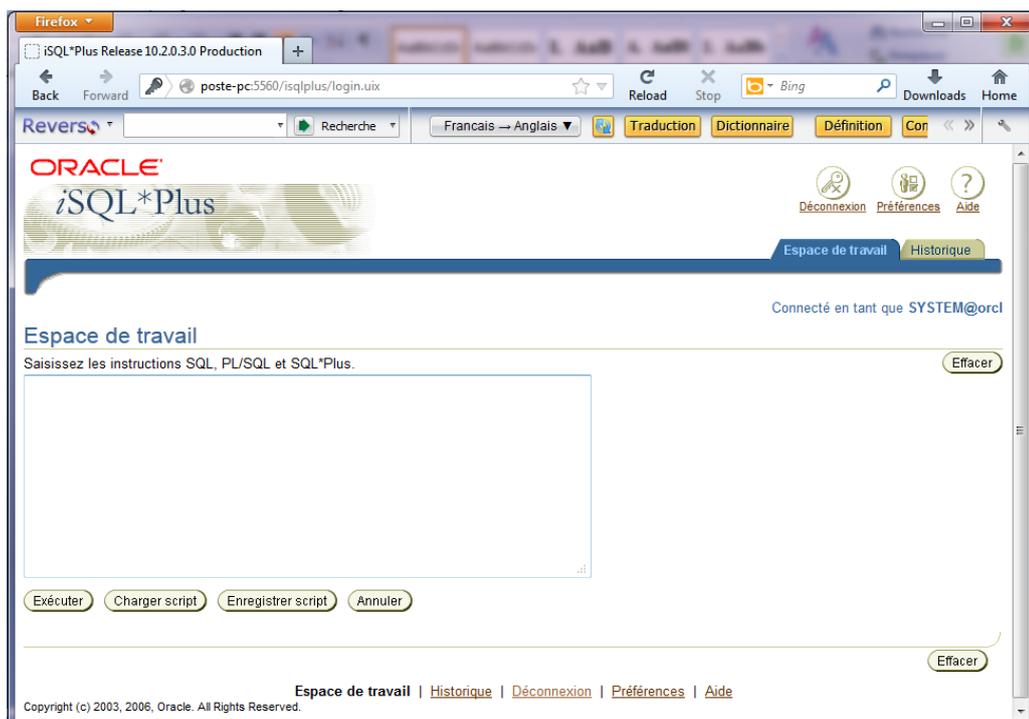


Figure II-5: Interface iSQL\*Plus

### II.3 Création d'une base de données

Le nom de la base de données est stocké dans le paramètre « DB\_NAME » ; il est également utilisé dans la commande SQL « CREATE DATABASE ». Il n'est plus possible de modifier le nom de la base de données après sa création.

#### II.3.1 L'assistant DBCA

L'assistant DBCA (DataBase Configuration Assistant) est un assistant de configuration de bases de données. Il guide dans le processus de création d'une nouvelle base de données, de modification de la configuration d'une base de données existante ou de la suppression d'une base de données. (Razvan Bizoï, 2005)

L'assistant de configuration de base de données permet de :

- Créer une base de donnée ;
- Configurer les options d'une base de données.
- Supprimer une base de données.
- Gérer les modèles d'une base de données.

#### II.3.2 Les utilisateurs

Lorsqu'on parle d'utilisateurs de base de données, il est généralement question de : (Razvan Bizoï, 2005)

- Utilisateurs finaux :

Les utilisateurs finaux sont des utilisateurs qui se connectent à la base Oracle pour interagir avec les données qui y sont stockées et les maintenir. Ils doivent être autorisés à créer des sessions pour pouvoir se connecter à la base et aussi disposer de privilèges de niveau objet sur les données qu'ils ont besoin de visualiser et de modifier. Ils ne reçoivent habituellement aucun privilège système général, de sorte qu'ils ne peuvent accéder à l'ensemble de la base. Leur accès est donc strictement limité aux zones qui leur sont nécessaires pour pouvoir accomplir leur travail.

- Administrateurs :

Les administrateurs de bases de données surveillent et maintiennent la base elle-même ; ils ont donc besoin du plus haut niveau de privilèges. Ils sont notamment chargés de créer des bases de données et des rôles, et d'octroyer des privilèges d'accès aux utilisateurs finaux.

### 1- Création d'un utilisateur

Lors de la création d'un utilisateur de base de données, l'objectif est de définir un compte utile et sécurisé qui détient les privilèges adéquats et les paramètres par défaut appropriés. (Razvan Bizoï, 2005)

La syntaxe de création d'un utilisateur est :

```
CREATE USER nom_utilisateur  
IDENTIFIED BY password ;
```

#### a. Les privilèges utilisateurs

Oracle permet à plusieurs utilisateurs de travailler en toute sécurité sur la même base.

Chaque donnée peut être confidentielle et accessible à un seul utilisateur, ou partageable entre plusieurs utilisateurs.

Les ordres **GRANT** et **REVOKE** permettent de définir les droits de chaque utilisateur sur les objets de la base.

Tout utilisateur doit communiquer son nom d'utilisateur et son mot de passe pour pouvoir accéder à la base. C'est ce nom d'utilisateur qui déterminera les droits d'accès aux objets de la base.

L'utilisateur qui crée une table est considéré comme le propriétaire de cette table. Il a tous les droits sur cette table et son contenu. En revanche, les autres utilisateurs n'ont aucun droit sur cette table (ni lecture ni modification), à moins que le propriétaire ne leur donne explicitement ces droits avec un ordre **GRANT**. (Richard Grin, 2008)

##### i) **GRANT**

L'ordre **GRANT** du langage SQL permet au propriétaire d'une table ou d'une vue de donner à d'autres utilisateurs des droits d'accès à celles-ci : (Richard Grin, 2008)

**GRANT** privilège **ON** table /vue **TO** utilisateur [**WITH GRANT OPTION**]

Les privilèges qui peuvent être donnés sont les suivants :

Privilèges d'accès à la base	Description
<b>SELECT</b>	droit de lecture
<b>INSERT</b>	droit d'insertion de lignes
<b>UPDATE</b>	droit de modification de lignes
<b>UPDATE (col1, col2, ...)</b>	droit de modification de lignes limité à certaines colonnes
<b>DELETE</b>	droit de suppression de lignes
<b>ALTER</b>	droit de modification de la définition de la table
<b>INDEX</b>	droit de création d'index
<b>ALL</b>	tous les droits ci-dessus

**Tableau II-1: les privilèges d'accès à une base**

Les privilèges **SELECT**, **INSERT** et **UPDATE** s'appliquent aux tables et aux vues. Les autres s'appliquent uniquement aux tables.

Un utilisateur ayant reçu un privilège avec l'option facultative **\_WITH GRANT OPTION\_** peut le transmettre à son tour.

Les droits peuvent être accordés à tous les utilisateurs en utilisant le mot réservé **PUBLIC** à la place d'un nom d'utilisateur : (Richard Grin, 2008)

**GRANT** privilège **ON** table /vue **TO PUBLIC**

*ii) REVOKE*

Un utilisateur ayant accordé un privilège peut le reprendre à l'aide de l'ordre **REVOKE** : (Richard Grin, 2008)

**REVOKE** privilège **ON** table /vue **FROM** utilisateur

**II.4 Conclusion**

Dans ce chapitre on a pu présenter un aperçu sur le SGBD Oracle 10 g qui, remarquons-le, se démarque des autres gestionnaires de bases de données par son côté administration très développé.

# Chapitre III

*Présentation de la  
banque BADR*

**III. Chapitre 3 : Présentation de la banque BADR****III.1 Introduction**

Une **banque** est une entreprise qui fait le commerce de l'argent. Cette activité peut être exercée pour le compte de clients de différentes manières : recevoir et garder des fonds, proposer divers placements (épargne), fournir des moyens de paiement (chèques, carte bancaires, etc.) et de change, prêter de l'argent, et plus généralement se charger de tous services financiers.

Une Banque peut également intervenir pour réaliser des opérations et des interventions sur les marchés financiers pour son compte ou celui de sa clientèle.

Dans le cadre de notre étude ; on a choisis la BADR Banque étant la première banque en Algérie.

**III.2 Présentation de la société BADR :****III.2.1 Description de la BADR Banque**

La Banque de l'Agriculture et du Développement Rural est une institution financière nationale issue du démembrement de la BNA (Banque Nationale d'Algérie). Elle a été créée le 13 mars 1982.

A sa création, la BADR avait pour principale mission le financement du secteur de l'agriculture et tout ce qui en dépend, notamment au plan des industries et des activités artisanales.

Dix ans après, la BADR consciente de son rôle de banquier, se devait d'élargir son champ d'intervention à tous les autres secteurs de l'économie nationale à tel point qu'aujourd'hui, le secteur agricole ne représente plus que 25% de son chiffre d'affaires.

Constituée initialement de 140 agences cédées par la Banque Nationale d'Algérie (BNA), son réseau compte aujourd'hui 310 agences et 42 GRE (ex-succursales) (Direction régionale). Prés de 7000 cadres et employés activent au niveau des structures centrales, régionales et locales. Son chiffre d'affaires s'élève à plus de 26.800.000.000 DA.

De part de la densité de son réseau et l'importance de ces effectifs, la BADR est considérée par le « BANKERS ALMANACH » (édition 2001) comme la première banque en Algérie. Elle occupe le premier rang au plan national et le 668<sup>ème</sup> au niveau mondial, sur environ 4100 banques classées.

### **III.2.2 Structure de la BADR**

La BADR est structurée en 21 directions centrales, le réseau qui se compose de 42 GRE (ex-succursale), 310 agences réparties sur l'ensemble du territoire national.

#### ***a- La direction générale***

Sous l'autorité du PDG elle est chargée du suivi de toutes les activités de la BADR, elle est assistée de six départements généraux :

- Inspection générale ;
- Direction de l'audit interne « DAI » ;
- Département de la communication « DC » ;
- Secrétariat de la direction ;
- Conseil commissions et comités ;
- Conseillers.

#### ***b- Les directions centrales***

Elles sont au total 21 directions à savoir :

- Direction générale adjointe « ressources et emplois » ;
- Direction générale adjointe « informatique, comptabilité et trésorerie » ;
- Direction générale adjointe « administration et moyens » ;
- Division internationale ;
- Direction du financement des grandes entreprises (D.F.G.E) ;
- Direction « Informatique central » (D.I.C) ;
- Direction du personnel (D.P) ;
- Direction des opérations techniques avec l'étranger (D.O.T.E) ;
- Direction du financement des petites et moyennes entreprises (D.F.P.M.E) ;
- Direction « informatique réseau d'exploitation » (D.I.R.E) ;

- Direction de revalorisation des ressources humaines (D.R.R.H) ;
- Direction des relations internationales (D.R.I) ;
- Direction du financement des activités agricoles (D.F.A.A) ;
- Direction de la télétransmission et de la maintenance informatique (D.T.M.I) ;
- Direction des moyens généraux (D.M.G) ;
- Direction du contrôle et des statistiques (D.C.S) ;
- Direction des études de marches et produits (D.E.M.P) ;
- Direction de la comptabilité générale (D.C.G) ;
- Direction de la réglementation, études juridiques et du contentieux (D.R.E.J.C) ;
- Direction du suivi et du recouvrement (D.S.R) ;
- Direction de la trésorerie (D.T) ;
- Direction de la prévision et du contrôle de gestion (D.P.C.G).

Nous allons présenter dans ce qui suit l'Organigramme général de la BADR :

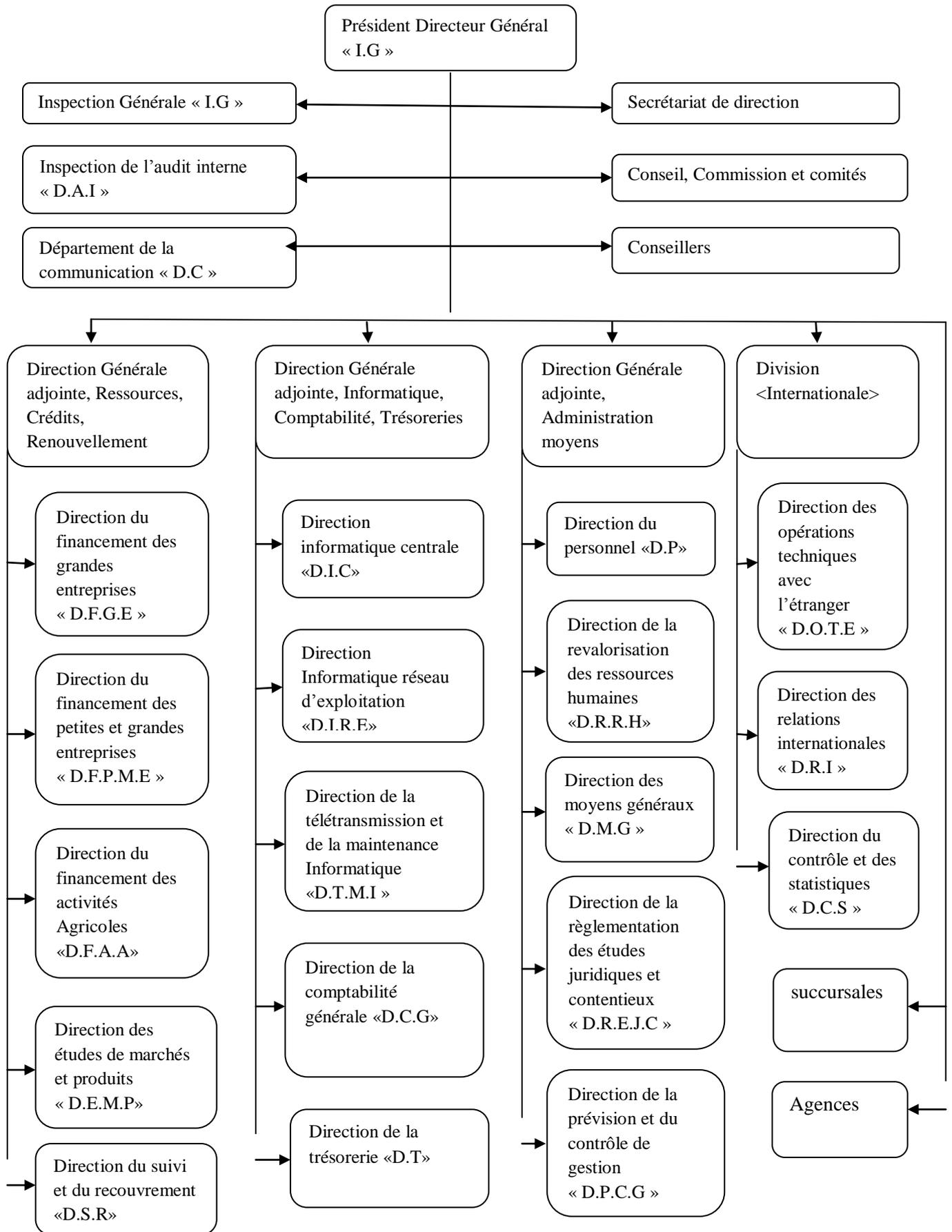
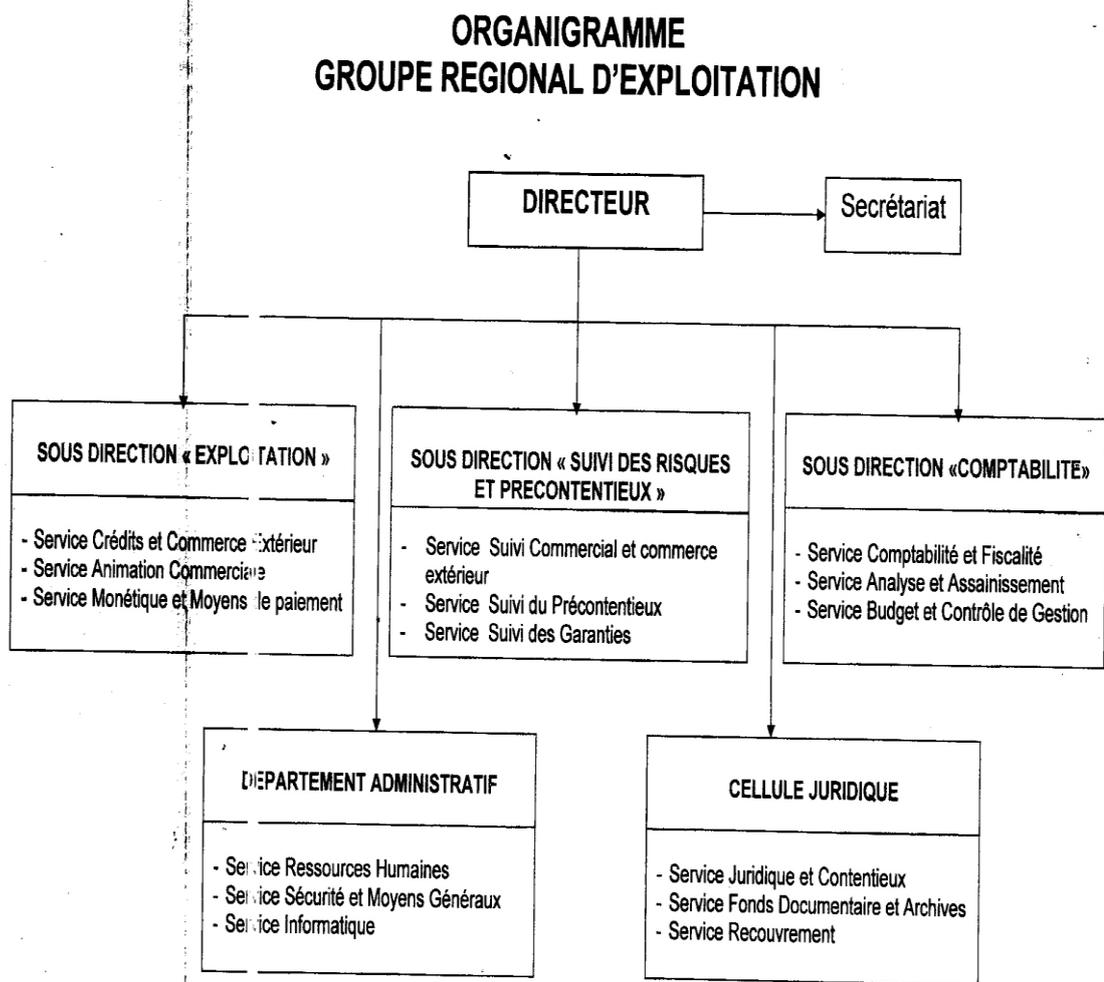


Figure III-1: Organisation générale de la BADR

De plus on trouve des GRE et des ALE réparties sur l'ensemble du territoire algérien.

La GRE est structurée comme suite :

- Sous direction d'Exploitation.
- Sous direction Suivi des risques et précontentieux.
- Sous direction Comptabilité.
- Cellule juridique.
- Le département Administratif.



I.G/2009

Figure III-2: Organisation de la GRE

Et chacune des ces sous directions comporte des sous services.

Afin de présenter une organisation de la BADR BANQUE encore plus spécifique que la précédente, on présente dans ce qui suit l’organigramme de l’agence qui englobe différents services.

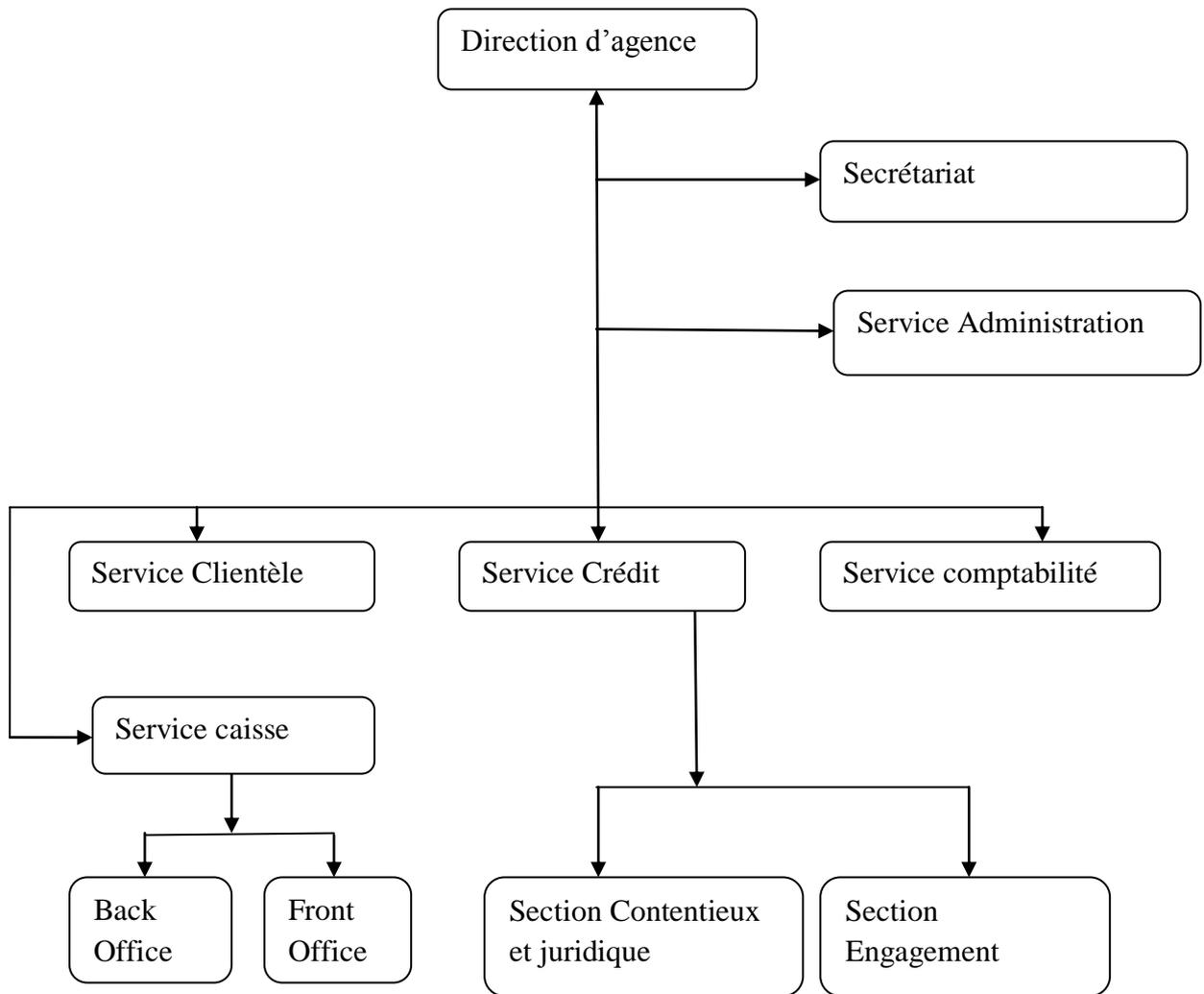


Figure III-3: Organigramme d'une Agence BADR

Au sein de la BADR, il est aisé de percevoir l’interaction constante entre les différents services décrits plus haut.

**III.3 Conclusion**

A titre de conclusion, il semble intéressant de mettre en évidence les questions actuelles qui se posent sur l'avenir de la banque BADR.

Un service informatique est créé, et la banque développe sa propre gestion assistée par ordinateur. Cependant, le système bancaire se doit d'incorporer l'informatique d'une manière complète dans tous ses traitements et ce afin d'avoir une gestion et un contrôle efficace et fiable.

# Chapitre 4

## *Analyse et Conception*

**IV. Chapitre 4 : Analyse et Conception****IV.1 Introduction**

Avant de concevoir un système informatique, il est essentiel de faire une analyse du domaine afin d'observer les différentes lacunes et de proposer une solution aux problèmes posés.

L'analyse de l'existant constitue l'étape fondamentale de l'étude préalable de la méthode Merise. Elle consiste à étudier toutes les procédures existantes au niveau du service caisse afin d'examiner la situation de gestion actuelle en vue de l'améliorer par des procédures et des méthodes bien adaptées. Pour cela, nous nous sommes intéressés :

- Aux postes de travail.
- Aux documents et fichiers existants.
- Aux moyens de traitement et de circulation de l'information.

**IV.2 Analyse du besoin**

Le but de notre projet est de développer une application simulant le fonctionnement d'un système de gestion bancaire. Il met en œuvre un système de gestion de base de données répartie (SGBDR). Notre étude s'est basée sur la description d'une partie du domaine d'activité concerné (Le service caisse).

Le service caisse est un service clé et stratégique de l'agence, il comporte des postes de travail qui ont des rôles fixés. Il a pour fonction essentielle de recevoir les dépôts d'espèces, d'exécuter les virements de toute origine effectués par ou pour le compte de la clientèle et d'assurer les règlements ou paiements sur instruction des clients.

Comme il est chargé de l'ouverture et de la tenue des comptes en traitant les opérations de versement et de retrait.

**IV.2.1 Étude des postes de travail**

L'étude des postes de travail consiste à les présenter d'une manière détaillée, ceci en dégagant les différentes tâches qui constituent ces derniers, ainsi que l'ensemble des documents circulant et les fiches manipulées. En effet, le service caisse tient à sa

disposition deux paramètres spécifiques : le front-office et le back-office. Le premier contient la section caisse chargée de retrait et du versement que ce soit en dinars ou en devise. Le second regroupe d'autre section : Portefeuille, virement et exploitation.

Voici l'organigramme qui englobe les différentes parties du commentaire précité.

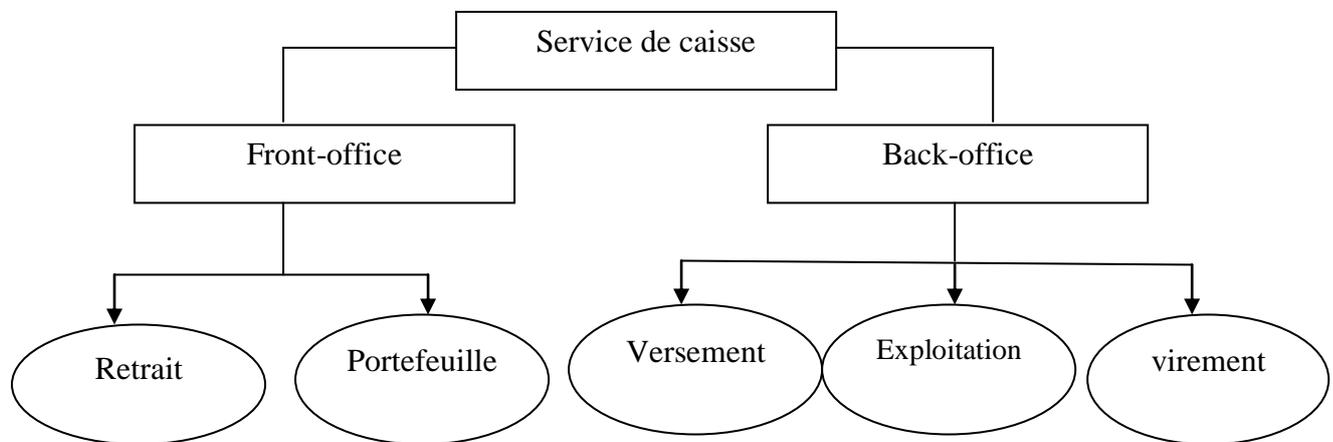


Figure IV-1: Organigramme du service de caisse

**a- Le service caisse en question**

En accordant ses premières priorités à l'informatisation de ses systèmes d'information, le service caisse s'est adjugé deux atouts qui lui permettent non seulement d'être le noyau de la banque, mais aussi un service qui offre les services les plus attractifs à la clientèle.

Il y a plusieurs postes de travail au niveau du service caisse :

- **Le chef de service caisse** : superviser toutes les tâches en relation avec se service.
- **L'Agent responsable des comptes** : charger de l'ouverture des comptes, tenu des comptes et de clôturer les comptes.
- **Le caissier / Guichetier** : exécuter des retraits (paiement) et versement.
- **Agent de virement** : transférer une somme d'argent au compte d'un bénéficiaire sur ordre d'un client.

**IV.2.2 Étude des documents**

Le système d'information du service caisse est principalement constitué de documents à établir manuellement.

La liste de ces documents est comme suite :

- Demande d'ouverture de compte : c'est un document par lequel le client demande à la banque son consentement pour l'ouverture d'un compte.
- Fiche de signature : document qui prouve l'authenticité et la régularité de la signature du chèque et des autres imprimés à remplir.
- Détail de monnaie : document qui permet la reconnaissance des espèces par le caissier.
- Ordre de virement : c'est un moyen d'exprimer clairement la volonté d'un client.

**IV.3 Spécification du besoin**

Venons-en à la problématique du sujet. Le service caisse comporte plusieurs postes de travail, et l'agent responsable des comptes est chargé de l'ouverture et des tenues des comptes ; et puisque les agences ayant chacune sa base de données locale, aucun échange ou communication automatique n'est permis entre deux agences, alors le problème majeur qui s'oppose au service est le fait qu'un client peut avoir plusieurs numéros de comptes dans plusieurs agences différentes.

Ainsi les problèmes identifiés, à résoudre principalement, sont les suivants :

- Redondance d'informations dans les documents de travail.
- Répétitivité des mêmes tâches de contrôle et vérification des documents qui ralentit le processus.
- Lenteur dans la circulation des documents entre les différents postes de travail à tous les niveaux hiérarchiques.
- La sous-exploitation des infrastructures informatiques matérielles et réseautiques acquises et mises à la disposition des agences.

#### IV.4 Solution proposée

La solution adéquate pour palier à ces problèmes est la mise en place d'une base de données répartie sur les différentes agences ainsi que la direction de la banque, afin de les faire interagir automatiquement sans problèmes. Ainsi chaque agence aura sa propre base de données indépendante et toute donnée enregistrée au niveau d'une agence pourra être consultée par les autres agences sans aucun déplacement ou manipulation humaine, et en transparence totale.

#### IV.5 Conception de la solution

A l'issue de ce qui précède, on suggère une solution informatique qui répondra au mieux aux problèmes de gestion du service caisse.

Reste à définir maintenant la conception de cette nouvelle solution. Pour cela, nous avons opté pour la méthode Merise<sup>1</sup> pour la modélisation de notre système.

##### IV.5.1 Modélisation du système

###### *a- Règle de gestion*

- Un client possède un compte dans une agence.
- Un employé travaille dans une agence.
- Une agence possède plusieurs employés.
- L'employé effectue différentes opérations sur des clients.
- Un client peut avoir plusieurs comptes.

###### *b- Construction du dictionnaire de données*

###### **1- Dictionnaire de données :**

---

<sup>1</sup> 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 (commentcamarche, 2013).

La phase d'analyse consiste à extraire des documents les informations Primaires qui vont Construire la future base de données.

La collection de l'ensemble des données primaire, appeler **attributs** ou **champs**, constitue le dictionnaire des données. Chaque attribut (champ) du dictionnaire de données peut être caractérisé par les propriétés suivantes :

<b>Propriété</b>	<b>Signification</b>
<b>Mnémonique</b>	Abréviation du nom de l'attribut.
<b>Libellé</b>	Libellé contenant la signification précise de l'attribut.
<b>Type de donnée</b>	Type de l'attribut : Alphabétique, Numérique, Date, ....
<b>Taille</b>	La taille de l'attribut

**Tableau IV-1 : Propriétés du dictionnaire de données**

Le dictionnaire de données Correspondant aux documents précédents est le suivant :

<b>Libellé</b>	<b>Mnémonique</b>	<b>Type de donnée</b>	<b>Taille</b>
Code du client	Code_Cli	AN	15
Nom du client	Nom_Cli	A	20
Prenom du Client	Prenom_Cli	A	20
Adresse du client	Adrs_Cli	AN	30
Rue du client	Rue_Cli	AN	15
Ville du client	Ville_Cli	A	15
Commune du Client	Comm	A	15
Willaya du client	Will	A	15
Pays du client	Pay_Cli	A	15
Code postale	Code_post	N	15
Date de naissance du client	DatNais_Cli	D	10
Ville de naissance du client	VillNais_Cli	A	15
Pays de naissance du Client	PayNais_Cli	A	15
Numéro de téléphone du Client	NumTel_Cli	N	15
Employeur	Employeur	A	20
Code de l'employé de la banque	Code_Emp	AN	15

Nom de l'employé	Nom_Emp	A	20
Prenom de l'employé	Prenom_Emp	A	20
Adresse de l'employé	Adrs_Emp	AN	15
Ville de l'employé	Ville_Emp	A	15
Willaya de l'employé	Will_Emp	A	15
Pays de l'employé	Pay_Emp	A	15
Code postale	Code_post	N	15
Numéro de téléphone	Num_tel	N	15
Date de naissance de l'employé	DatNais_Emp	D	10
Ville de naissance de l'employé	Vill_Nais	A	20
Pays de naissance de l'employé	Pay_Nais	A	15
Code du compte	Code_Compte	AN	15
Type du compte	Type_Compte	A	20
Date de l'ouverture du compte	Date_ouv_Compte	D	10
Somme contenue dans le compte	Somme	N	30
Code de l'agence	Code_Agence	AN	15
Nom de l'agence	Nom_Agence	A	30
Adresse de l'agence	Adrs_Agence	AN	15
Rue de l'agence	Rue_Agence	A	15
Ville de l'agence	Vill_Agence	A	15
Commune de l'agence	Comm_Agence	A	15
Willaya de l'agence	Will_Agence	A	15
Code de l'opération	Code_Op	AN	15
Type de l'opération	Type_Op	A	20
Date de l'opération	Date_Op	D	10

**Tableau IV-2: Dictionnaire de données**

**2- Dictionnaire de données épuré :**

Le dictionnaire de données épuré se présente comme suite :

<b>Libellé</b>	<b>Mnémonique</b>	<b>Type de données</b>	<b>Taille</b>
Le code du client	Code_Client	AN	15
Le nom du client	Nom_Client	A	20
Le prénom du client	Prenom_Client	A	20
L'adresse du client	Addr	AN	30
Code postale	Cod_post	N	15
La commune du client	Comm	A	15
La wilaya du client	Wil	A	15
Date de naissance du client	Dat_naiss	D	10
L'employeur du client	Employeur	A	20
Le code le l'employé	Code_Employe	AN	15
Le nom de l'employé	Nom_Emp	A	20
Le prénom de l'employé	Prenom_Emp	A	20
Date de naissance de l'employé	DatNais_Emp	D	10
Le code du compte	Code_Compte	AN	15
Le type du compte	Type_Compte	A	20
La date d'ouverture du compte	Date_ouv_Compt	D	10
La somme contenue dans un compte	Somme	N	30
Le code le l'agence	Code_Agence	AN	15
Le nom de l'agence	Nom_Agence	A	30
L'adresse où se situe l'agence	Adrss_Agence	AN	20
Code opération	Code_Op	AN	15
Type de l'opération	Type_Op	A	20
Date de l'opération	Date_Op	D	10

**Tableau IV-3: Le dictionnaire de données épuré**

**A** : Alphabétique ;

**AN** : Alphanumérique ;

D : Date ;

*c- (rien)Le 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. (commentcamarche, 2013)

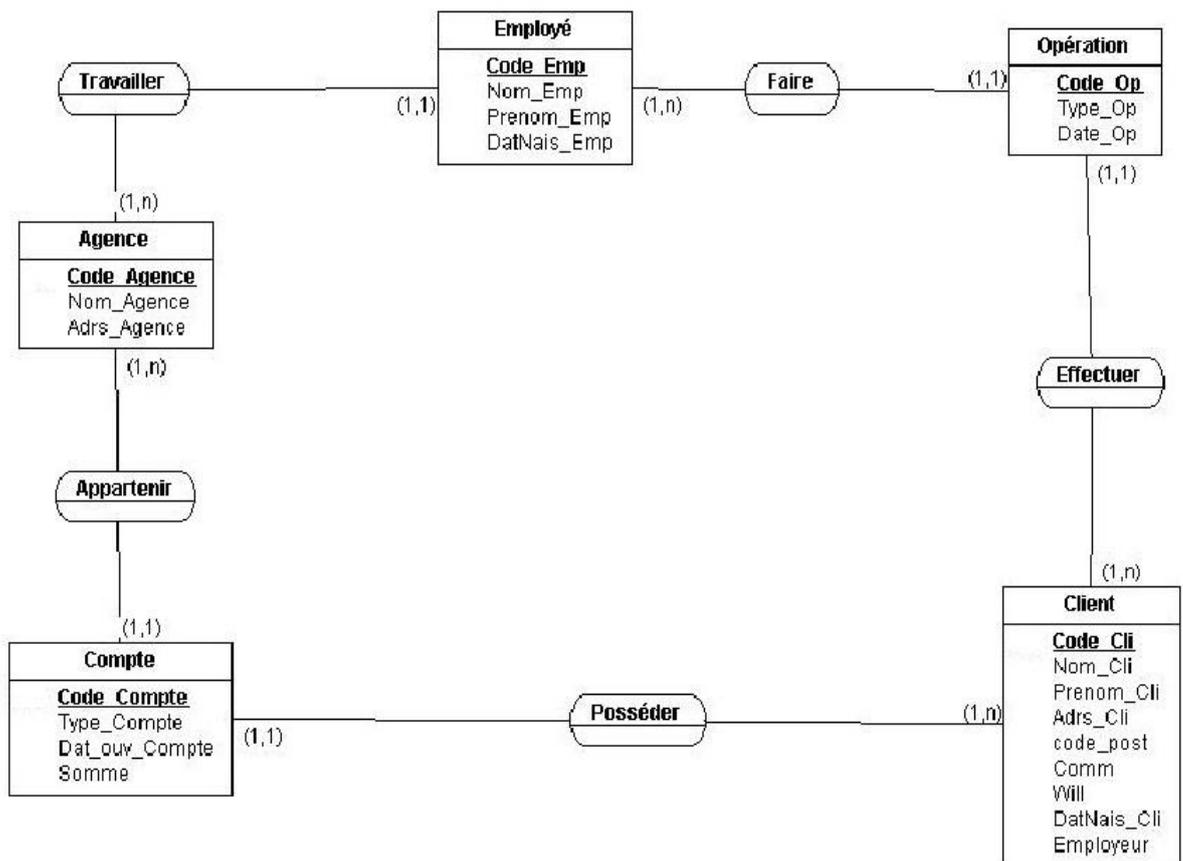


Figure IV-2: Le modèle conceptuel de données

*d- Le modèle logique de données*

**i) Règles de passage du MCD vers le MLD**

Nous allons définir les règles de passage du MCD vers le MLD : (Bagourram, 2013)

- Chaque entité devient une relation.
  - o L'identifiant devient clé primaire.

- Les propriétés deviennent attributs.
- Lorsqu'une entité est source de cardinalités comme 0,1 ou 1,1 alors la règle précédente s'applique. De plus, l'identifiant de l'entité père dans l'association devient clé étrangère dans l'entité fils.
- Une association de type 0,n - 1,n (ou 0,n - 0,n ou 1,n - 1,n) devient une relation dont :
  - La clé primaire se compose des identifiants des entités engagés dans la relation.
  - Les attributs sont les propriétés portées par l'association.

Après avoir appliqué ses règles à notre MCD ; on a obtenu le MLD suivant :

**Employe** ( Code-Emp, Nom\_Emp, Prenom\_Emp, DatNais\_Emp, **Code\_Agence\*** ) ;

**Operation** ( Code\_Op, Typr\_Op, Date\_Op, **Cod\_Cli\***, **Code\_Emp\*** ) ;

**Client** ( Code\_Cli, Nom\_Cli, Prenom\_Cli, Adrs\_Cli, Code\_post, Comm, Will, DaNais\_Cli, Employeur ) ;

**Compte** ( Code\_Compte, Type\_Compte, Date\_ouv\_Compte, Somme, **Code\_Agence\***, **Code-Cli\*** ) ;

**Agence** ( Code\_Agence, Nom\_Agence, Adrs\_Agence ) ;

#### IV.5.2 Répartition de la base de données

##### a- Fragmentation

Dans notre étude on va se limiter dans la répartition de la base de données à 3 sites, qui sont la Centrale et deux agences A et B. Pour nous la centrale c'est la succursale se trouvant à Tlemcen et les agences A et B sont à Remchi et à Maghnia.

La centrale est un site, qui n'est accessible que par quelques utilisateurs privilégiés. On peut interroger un certain type de compte (courant, étudiant...), consulter un certain type d'opération (Ajout, modification...) ainsi que la gestion des employés, ajouter, modifier et supprimer un employé de la base de données.

Dans les agences, les employés peuvent ajouter des clients, modifier ou supprimer des clients, créer des comptes, et rechercher les clients sur toutes les bases.

Nous avons mené une étude statistiques sur les différentes requêtes utilisées dans le cadre du fonctionnement d'une banque qui a amené au fait que les requêtes les plus fréquemment utilisées sont des requêtes à base de restrictions de type :

**Select \* from ..... Where Agence =.....**

Alors puisque les plus fréquentes sont les plus critiques pour la base de données, on a procédé à une fragmentation horizontale qui va affecter à une base de données les lignes des tables des comptes, des opérations, et des clients correspondants, en distribuant la base globale sur les agences selon une approche de conception descendante.

#### **IV.6 Conclusion**

Dans ce chapitre et lors de notre conception, nous avons construit un modèle conceptuel de données valide décrivant les données qui seront utilisées par notre système, on l'a convertit en modèle logique de données selon le modèle relationnel. On a fragmenté le MLD de la base de données globale en différents schémas locaux ainsi qu'allouer les schémas locaux des bases de données à leurs agences correspondantes.

Dans la suite du document, nous allons procéder au développement de notre application qui va concrétiser et mettre en œuvre notre solution.

# Chapitre 5

## *Réalisation*

## V. Chapitre 5 : Réalisation

### V.1 Introduction

Dans ce chapitre, nous allons aborder le côté pratique de notre projet. Nous définirons les étapes de réalisation, et la mise en œuvre de notre système de gestion bancaire. Nous commencerons par une brève présentation des outils de développement que nous avons utilisé ; et nous terminerons par l'implémentation de notre système.

### V.2 Outils de développement

Pour réaliser notre système réparti, nous avons utilisé différents outils. Comme outils de développement, on a choisi **NetBeans** avec comme langage de programmation **Java** ; et le système de gestion de base de données **Oracle 10g** dont on a fait une brève présentation précédemment.

#### V.2.1 NetBeans

NetBeans est un projet open source fondé par Sun Microsystems. L'IDE NetBeans est un environnement de développement permettant d'écrire, compiler, déboguer et déployer des programmes. Il est écrit en Java, mais peut supporter n'importe quel langage de programmation. Il y a également un grand nombre de modules pour étendre l'IDE NetBeans. L'IDE NetBeans est un produit gratuit, sans aucune restriction quant à son usage. ( Hyacinthe MENIET, 2011)

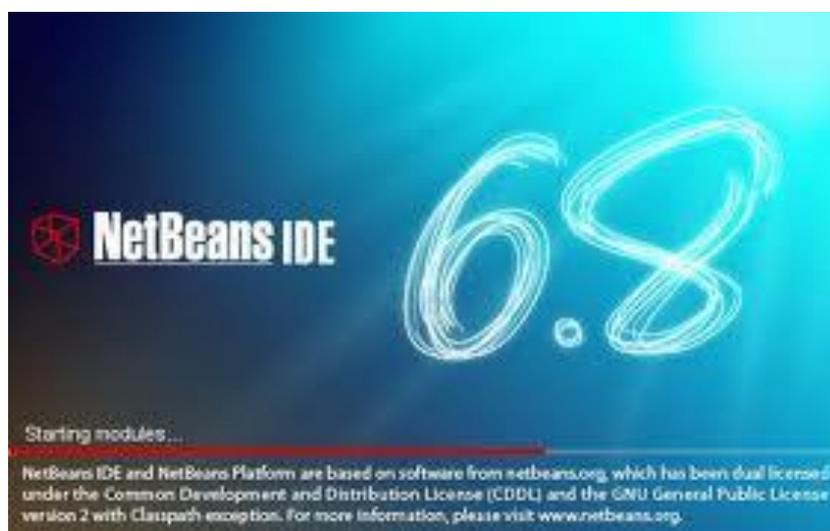


Figure V-1: NetBeans IDE

**V.2.2 Java**

Java est un langage Objet permettant le développement d'applications complètes s'appuyant sur les structures de données classiques (tableaux, fichier) et utilisant abondamment l'allocation dynamique de mémoire pour créer des objets en mémoire. La notion de structure, ensemble de données décrivant une entité (un objet en Java), est remplacée par la notion de classe au sens de la programmation Objet. Le langage Java permet également la définition d'interfaces graphiques (GUI : Graphical User Interface) facilitant le développement d'application interactives et permettant à l'utilisateur de piloter son programme dans un ordre non imposé par le logiciel. (Michel Divay, 2006)

**V.2.3 Oracle, JDBC et NetBeans**

Nous avons présenté dans le chapitre II les notions d'Oracle et de JDBC ; pour que nous puissions utiliser NetBeans avec ces deux outils, nous devons intégrer une librairie JDBC (.jar) à notre projet. On peut trouver des drivers appropriés sur le site officiel d'Oracle.

**V.3 Création des bases de données**

En premier lieu, il faut installer Oracle 10g serveur sur les serveurs de la société.

Création des trois bases : à l'aide de l'assistant DBCA déjà défini dans le chapitre II :

- Création de la base Tlemcen située à la direction.
- Création de la base Remchi située à l'agence n°1.
- Création de la base Maghnia située à l'agence n°2.

**V.4 Configuration d'Oracle**

A l'aide d'Oracle Net Manager :

- Configurer un processus d'écoute LISTENER
- Configurer un nom de service réseau.

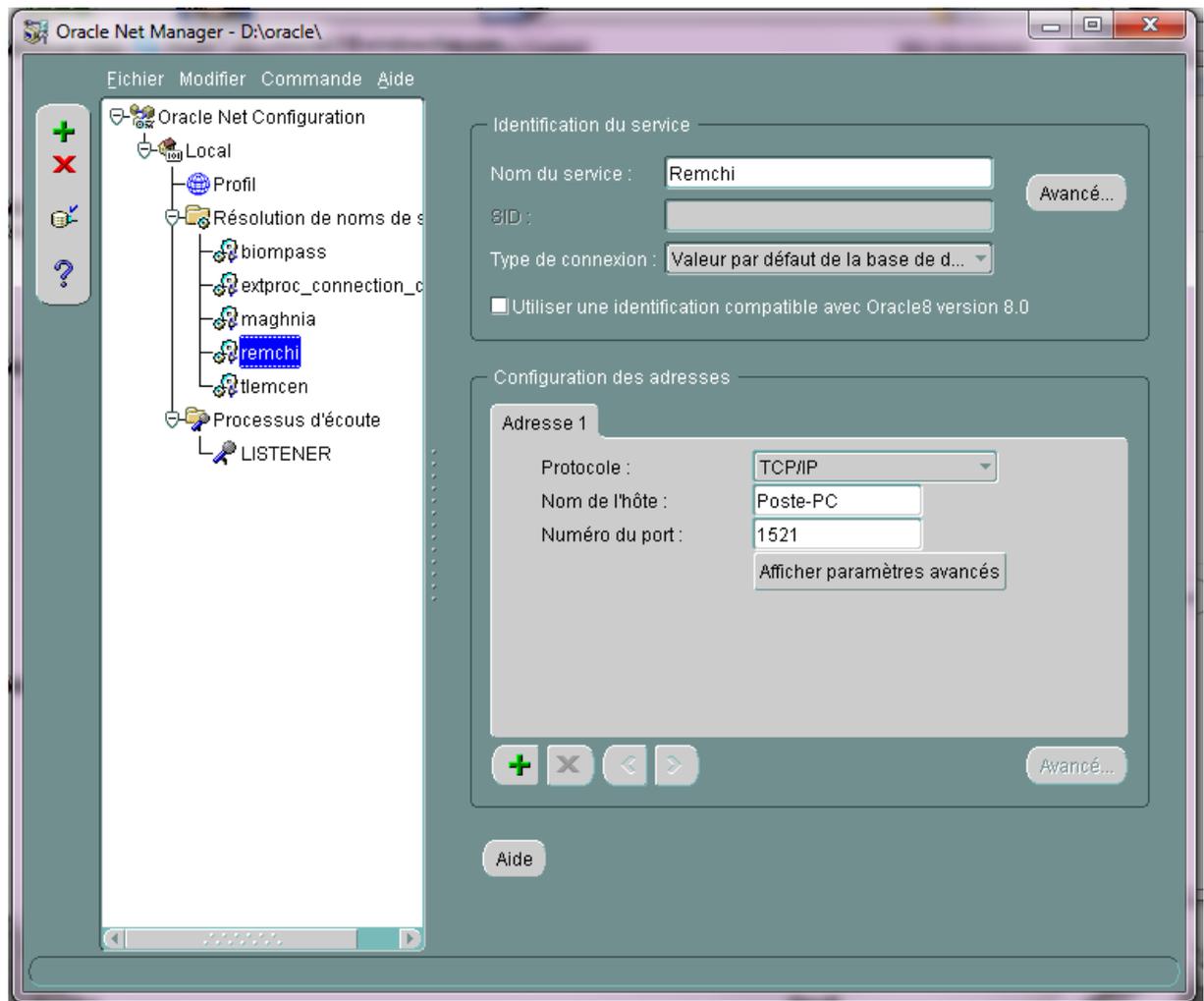


Figure V-2: Oracle Net Manager

## V.5 Implémentation de la base de données répartie

### V.5.1 Création des Utilisateurs et des tables

- Au niveau de la centrale Tlemcen :
  - Créer un utilisateur :

```

CREATE USER agentlemcen
IDENTIFIED BY agentlemcen;

GRANT all privileges TO agentlemcen;

```

- Créer toutes les tables au niveau de la base Tlemcen.
- Au niveau de l'agence 1 Remchi:

- Créer un utilisateur:

```
CREATE USER agenremchi  
IDENTIFIED BY agenremchi;  
GRANT all privileges TO agenremchi;
```

- Créer les tables au niveau de la base Remchi.
- Au niveau de l'agence 2 Maghnia :

- Créer un utilisateur:

```
CREATE USER agenmaghnia  
IDENTIFIED BY agenmaghnia;  
GRANT all privileges TO agenmaghnia;
```

- Créer les tables au niveau de la base Maghnia.

### V.5.2 Création des liens entre les bases

- Entre Centrale tlemcen et Agence Remchi :

```
CREATE DATABASE LINK lien_tlem_rem CONNECT TO agenremchi  
IDENTIFIED BY agenremchi USING 'Remchi' ;
```

- Entre Centrale tlemcen et Agence Maghnia:

```
CREATE DATABASE LINK lien_tlem_magh CONNECT TO agenmaghnia  
IDENTIFIED BY agenmaghnia USING 'Maghnia' ;
```

- Entre les deux Agences:

```
CREATE DATABASE LINK lien_rem_magh CONNECT TO agenmaghnia  
IDENTIFIED BY agenmaghnia USING 'Maghnia' ;
```

## V.6 Développement de l'application

### V.6.1 Les méthodes de connexion de Netbeans vers Oracle 10g

La connexion à la base de données Oracle est rendue possible par l'utilisation de la classe *DriverManager* et de l'interface *Connection*. La connexion s'effectue par la méthode *getConnection* de l'interface *DriverManager*.

Donc deux étapes sont alors nécessaires pour qu'un programme se connecte à une base :

- Le chargement du pilote par l'appel de la méthode `java.lang.Class.forName` .
- l'établissement de la connexion en appelant un objet de l'interface *Connection* par l'instruction : `objet= DriverManager.getConnection (chaîneConnexion, logine, password) ;`

Illustration dans le code suivant :

```
Class.forName ("oracle.jdbc.driver.OracleDriver");  
String URL = "jdbc:oracle:thin:@localhost:1521:Tlemcen";  
String USER = "agentlemcen";  
String PASSWD = "agentlemcen";  
cnx =DriverManager.getConnection(URL,USER,PASSWD);
```

### V.6.2 Fermeture d'une connexion

La connexion est fermée en appliquant la méthode *close()* à tous les objets *Connection* ouvert.

## V.7 Présentation de l'application

### V.7.1 Structure générale

Nous présentons là-dessus la structure générale du fonctionnement de notre application :

- Accueil :
  - Agence
    - Interface d'authentification Agence
      - ✓ Agence
        - Ajouter des clients
          - ❖ Ajouter un compte
        - Rechercher et gérer les clients
          - ❖ Afficher tous les clients.
          - ❖ Rechercher.
          - ❖ Ajouter Compte
          - ❖ Modifier des clients
          - ❖ Supprimer des clients
  - Administrateur
    - Interface d'authentification Administration
      - ✓ Administration
        - Gestion des employés
          - ❖ Afficher les employés
          - ❖ Ajouter des employés
          - ❖ Modifier des employés
          - ❖ Supprimer des employés
        - Consulter les opérations
        - Consulter les clients

### V.7.2 Les interfaces de l'application

Nous allons présenter dans ce qui suit les différentes interfaces de notre application, commençant par la fenêtre *Accueil*. C'est la première interface de l'application apparaissant lors de l'exécution ; elle permet d'accéder via deux boutons, à deux interfaces d'authentification, un pour l'agence et un pour les administrateurs.



Figure V-3: Fenêtre Accueil

- Agence :

Nous commençons par présenter les différentes interfaces du côté de l'agence. En premier lieu, nous aurons une première fenêtre *Authentification Client* ; après la saisie du nom d'utilisateur, du mot de passe et du nom de la base, l'utilisateur pourra accéder à l'espace Agence.



Figure V-4: Authentification Client

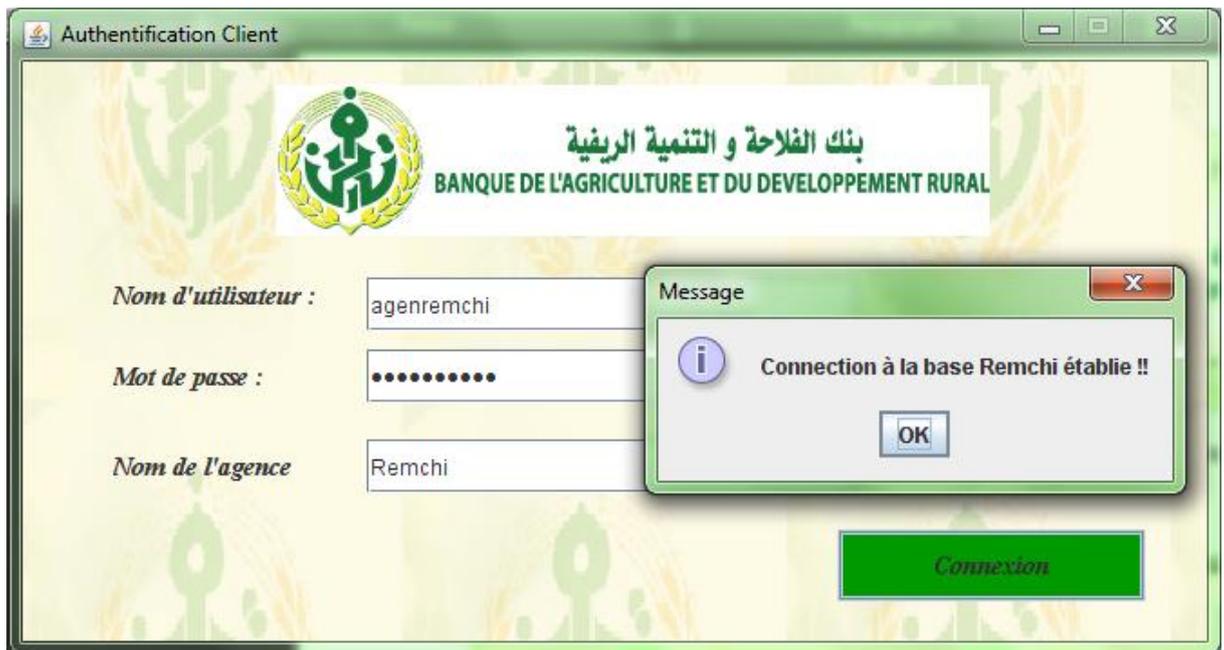


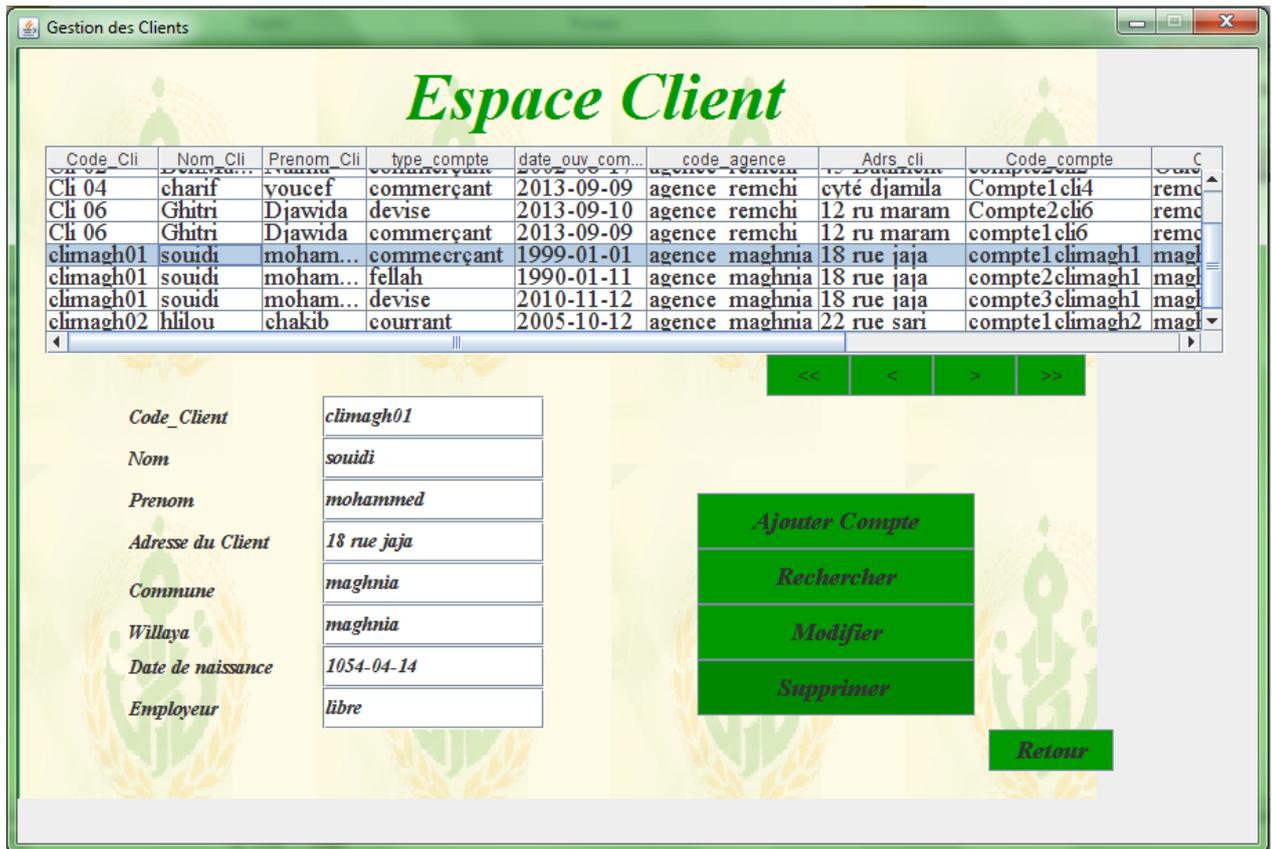
Figure V-5: Confirmer la connexion de l'utilisateur à la base

Lorsque l'utilisateur s'est connecté, il sera dirigé vers l'interface **Agence**



Figure V-6: Espace Agence

Pour rechercher ou gérer un client, l'utilisateur doit cliquer sur le bouton « **Rechercher et gérer les clients** » ; il sera rediriger vers la fenêtre suivante :

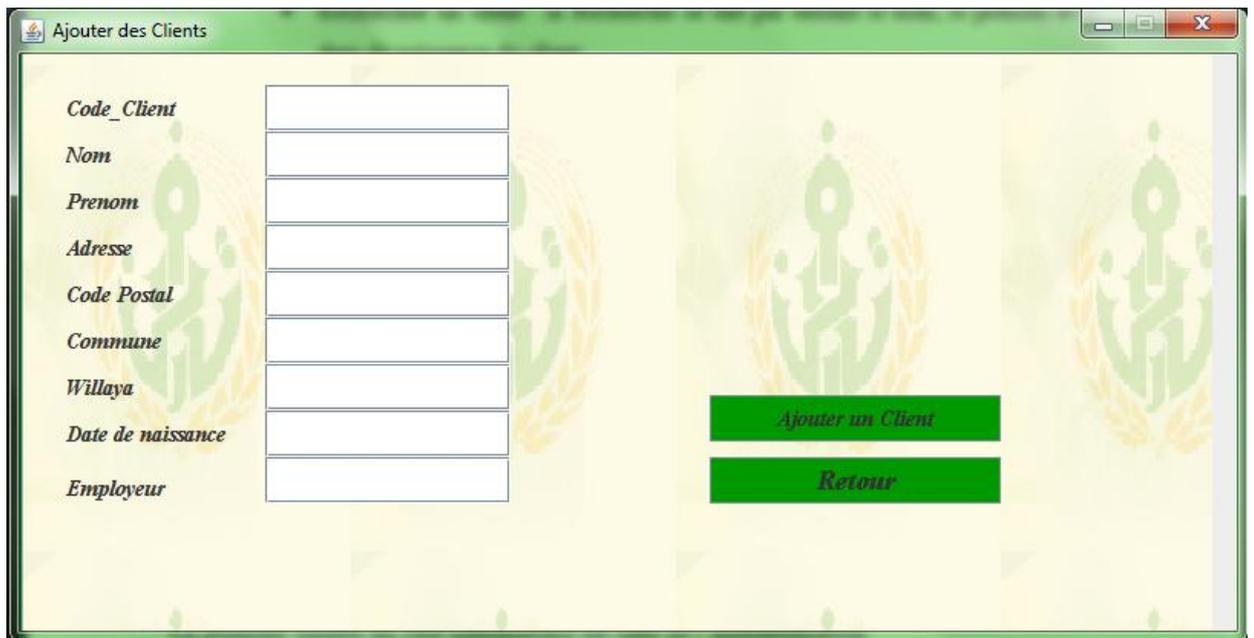


**Figure V-7: Rechercher et gérer les clients**

Cette interface permettra à l'utilisateur de :

- Rechercher un client : la recherche se fait en entrant le nom, le prénom et la date de naissance du client.
- Modifier un client.
- Supprimer un client.
- Et ajouter un compte à un client.

Le deuxième bouton de l'interface agence, permet de créer un nouveau client à la base. Lorsqu'un nouveau demandeur de compte se présente à l'agence, l'utilisateur doit vérifier s'il a un numéro dans la base (dans toutes les agences de la BADR). Sinon il va être créé sur l'interface suivante :



Ajouter des Clients

*Code\_Client*

*Nom*

*Prenom*

*Adresse*

*Code Postal*

*Commune*

*Willaya*

*Date de naissance*

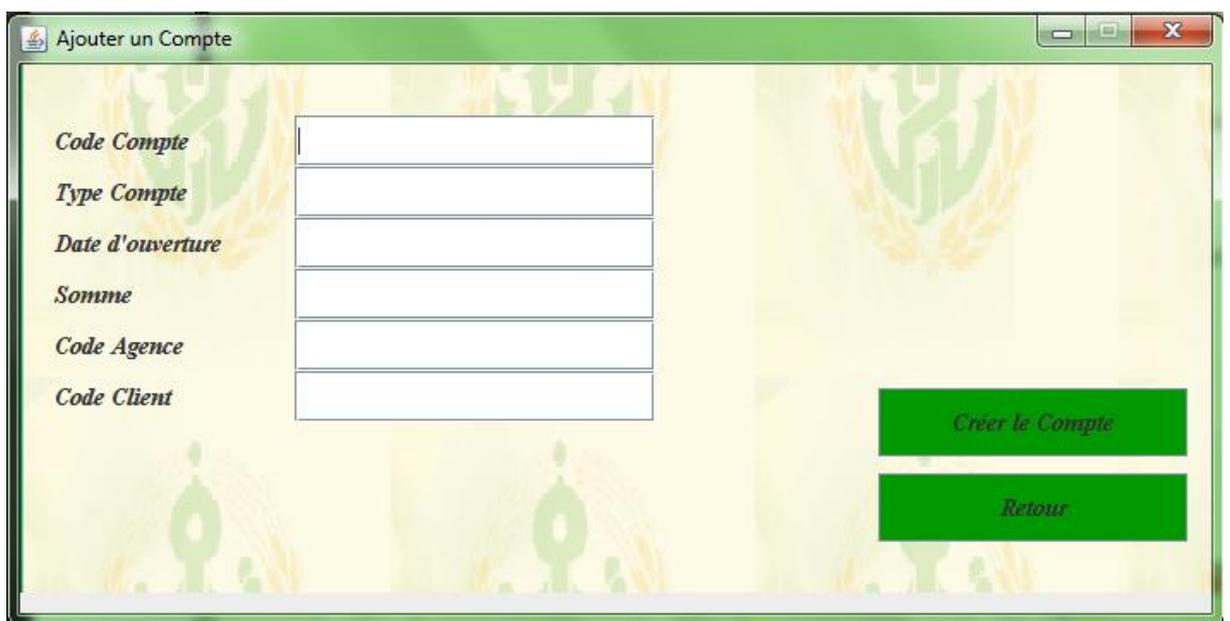
*Employeur*

*Ajouter un Client*

*Retour*

**Figure V-8: Ajouter un client**

Une fois le client ajouté à la base, l'utilisateur sera automatiquement dirigé vers l'interface suivante qui lui permettra de lui affecter un compte :



Ajouter un Compte

*Code Compte*

*Type Compte*

*Date d'ouverture*

*Somme*

*Code Agence*

*Code Client*

*Créer le Compte*

*Retour*

**Figure V-9: Ajouter un compte**

- Administration :

La première fenêtre du côté administratif est celle de l'authentification.



Figure V-10: Authentification Administration



Figure V-11: Confirmer l'authentification de l'administrateur

Après que l'administrateur s'est connecté, il sera orienté vers l'espace administratif suivant :



Figure V-12: Espace administration

Dans cet espace l'administrateur pourra consulter tous les clients de toutes les agences, toutes les opérations effectuées sur les clients et gérer les employés.

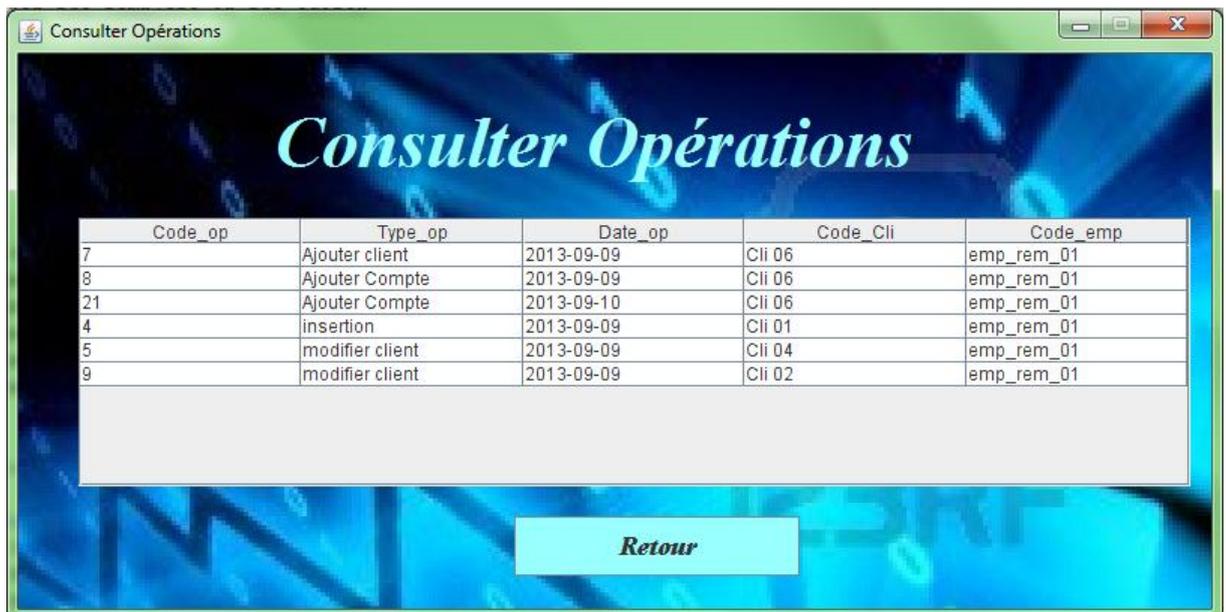


Figure V-13: Administration / Consultation des opérations

Tous les Clients

*Clients de toutes les agences de Tlemcen*

Code_Cli	Nom_Cli	Prenom...	Adrs_cli	Comm	Will	Danaiss_cli	Employ...	code_compte	type_compte	date_ouv_com...	somme	nom_agence
Cli 01	Boua...	Zak...	12 Ru...	Remchi	T...	1985-05...	seciété	compte1Cli1	courrant	2013-09-07	5000	agence rem...
Cli 02	Ben...	Naïma	45 Bat...	Oule...	T...	1953-09...	CE...	compte1cli2	fellah	2013-09-05	80000	agence rem...
Cli 02	Ben...	Naïma	45 Bat...	Oule...	T...	1953-09...	CE...	compte2cli2	commer...	2002-06-17	100...	agence rem...
Cli 04	charif	youcef	cyté dj...	remchi	tl...	1988-09...	cavir	Compte1 cli4	commer...	2013-09-09	50000	agence rem...
Cli 06	Ghitri	Dja...	12 ru ...	remchi	tl...	1930-09...	libre	Compte2cli6	devise	2013-09-10	5000	agence rem...
Cli 06	Ghitri	Dja...	12 ru ...	remchi	tl...	1930-09...	libre	compte1cli6	commer...	2013-09-09	690...	agence rem...
climagh01	souidi	moh...	18 rue...	magh...	...	1054-04...	libre	compte1 cli...	commec...	1999-01-01	12000	agence mag...
climagh01	souidi	moh...	18 rue...	magh...	...	1054-04...	libre	compte2cli...	fellah	1990-01-11	140...	agence mag...
climagh01	souidi	moh...	18 rue...	magh...	...	1054-04...	libre	compte3cli...	devise	2010-11-12	9000	agence mag...
climagh02	hlilou	chakib	22 rue...	magh...	...	1981-12...	badr	compte1 cli...	courrant	2005-10-12	50000	agence mag...

Retour

**Figure V-14: Administration /Consultation des Clients**

La fenêtre de gestion des employés permet à l'administrateur d'ajouter un nouvel employé dans une des agences, de modifier et/ou de supprimer les employés.



Figure V-15: Administration /Gestion des employés

## V.8 Conclusion

Dans ce dernier chapitre, nous avons présenté les différentes parties nécessaires à la réalisation de notre système de gestion bancaire réparti, à savoir la création de la base de données, présentation des différents outils utilisés pour réaliser notre solution, illustration des différentes interfaces composant notre application.

Conclusion

Générale

### Conclusion générale

Ainsi s'achève notre projet de fin d'étude. Notre objectif était la réalisation d'un système de gestion bancaire réparti couvrant toutes les agences de la banque BADR avec la succursale (Direction Régionale), pour résoudre des problèmes qui ont un impact direct sur la gestion de la banque, un système qui sera très bénéfique aussi bien pour les responsables que pour les employés de la BADR.

En ce qui nous concerne, ce travail a été pour nous à la fois, un sujet de recherche dans le domaine universitaire, et d'affirmation dans le monde professionnel. En effet cette expérience nous a permis de joindre l'utile à l'agréable en évaluant aussi bien les profondeurs théoriques que pratiques de ce vaste et passionnant domaine qu'est celui des bases de données réparties. D'un côté, ce projet nous a permis de nous familiariser avec le SGBD Oracle 10g, et d'un autre côté d'introduire des requêtes SQL dans un langage comme le Java.

Comme perspectives, nous envisageons d'une part, d'élargir le système pour tous les autres services de la banque, et d'autre part de généraliser l'expérience en 1<sup>er</sup> lieu au niveau de toutes les agences de la direction et en 2<sup>ème</sup> lieu à toutes les agences au niveau national.

## Bibliographie

**Dave Odilon DJAMOU YIKAM . 2008.** Bases de données réparties sous Oracle. *memoireonline.com*. [En ligne] Ecole supérieur de management commerce et informatique, 2008. [http://www.memoireonline.com/05/10/3459/m\\_Bases-de-donnees-reparties-sous-Oracle.html](http://www.memoireonline.com/05/10/3459/m_Bases-de-donnees-reparties-sous-Oracle.html).

**Hyacinthe MENIET. 2011.** NetBeans. *trustonme*. [En ligne] 2011. [http://www.trustonme.net/didactels/82.html#java\\_netbeans\\_intro](http://www.trustonme.net/didactels/82.html#java_netbeans_intro).

**A. ABDALLAH. 2012.** Chapitre 1 : Introduction aux bases de données réparties. 2012. Vol. V1.0.

**Anne Tasso. 2011.** *Le livre de Java premier langage*. 7e édition. s.l. : Eyrolles, 2011. p. 515 Pages.

**Aurélien, Gillet; Vladimir, Svoboda. 2010.** Les bases de données réparties. 2010.

**Bagourram. 2013.** Introduction a la methode Merise et a ses extensions. *scribd.com*. [En ligne] 2013. <http://www.scribd.com/doc/27979634/Introduction-a-la-methode-Merise-et-a-ses-extensions>.

**Christian, Soutou. août 2011.** *SQL pour Oracle Applications avec Java, PHP, et XML*. 5ème édition. mayrre : eyrolles, août 2011. p. 697.

**Claude Delannoy. 2011.** *Programmer en Java*. 6ème édition. s.l. : EYROLLES, 2011. p. 830 pages.

**commentcamarche. 2013.** MERISE - Initiation a la conception de systemes d'information. *commentcamarche*. [En ligne] Juin 2013. <http://www.commentcamarche.net/contents/655-merise-initiation-a-la-conception-de-systemes-d-information>.

**Desfontaines, Vincent. 2000.** Introduction aux bases de données réparties. *hds.utc.fr*. [En ligne] Septembre 2000. <https://www.hds.utc.fr/~ducourth/TX/BDD/BDD-intro.html>.

**Ecole de technologie supérieure.** ETS: cours par sigle; chapitre 10- Architecture des systèmes de base de données. [www.etsmtl.ca](http://www.etsmtl.ca). [En ligne] <https://cours.etsmtl.ca/gpa775/Cours/>.

**Gardarin, Georges. 2003.** *Bases de données*. 5ème tirage. s.l. : Eyrolles, 2003. p. 788.

**Gilles, Briard. 2006.** *Oracle 10g sous Windows*. s.l. : eyrolles, 2006. p. 846 pages .

**Hakim MADI . 2009.** Conception et réalisation d'une base de données répartie sous oracle : cas de l'hébergement des résidences universitaires. [memoireonline.com](http://memoireonline.com). [En ligne] Université A/Mira de Bejaia , 2009. [http://www.memoireonline.com/02/11/4278/m\\_Conception-et-realisation-dune-base-de-donnees-repartie-sous-oracle--cas-de-lhebergement-d0.html](http://www.memoireonline.com/02/11/4278/m_Conception-et-realisation-dune-base-de-donnees-repartie-sous-oracle--cas-de-lhebergement-d0.html).

**Michel Divay. 2006.** *La programmation Objet en Java*. s.l. : DUNOD, 2006. p. 443 Pages .

**Michelle, Gillet et Patrick, Gillet. 2011.** *Les systèmes d'information de A à Z*. [éd.] Savoirs. s.l. : DUNOD, 2011. p. 217.

**Razvan Bizoï. 2005.** *Oracle 10g - Administration*. [éd.] Tsoft Eyrolles. 1ère édition. s.l. : Eyrolles, 2005. p. 744.

**Richard Grin. 2008.** *Langage SQL*. s.l. : Université de Nice Sophia-Antipolis, 2008. p. 108.

**Rim Moussa. 2006.** Teaching\_fichiers/BDR. [lamsade.dauphine](http://lamsade.dauphine.fr). [En ligne] 6 juin 2006. <http://www.lamsade.dauphine.fr/~litwin/Rim/>.

**Spaccapietra, Stefano. 2002.** RessourceBD. [tice.univ-nc.nc](http://tice.univ-nc.nc). [En ligne] 06 Nov 2002. [http://tice.univ-nc.nc/~taladoire/Pedagogie/RessourcesBD/EPFL/poly3\\_fichiers/15/15.html](http://tice.univ-nc.nc/~taladoire/Pedagogie/RessourcesBD/EPFL/poly3_fichiers/15/15.html).

**Thierry, Coq. 2012.** *Méthode et informatique*. s.l. : Lavoisier, 2012. p. 295.

# *Annexe*

## Annexe 1 : Fiche de signature

## بنك الأقاليم والتنمية الريفية

BANQUE DE L'AGRICULTURE ET DU DEVELOPPEMENT RURAL

بطاقة امضاء  
FICHE DE SIGNATURE

SIEGE : ..... : مركز

Date d'ouverture	تاريخ التسجيل	N° de compte	رقم الحساب
TITULAIRE	المستفيد	Signature	الامضاء
NOM : .....	اللقب : .....		
PRENOMS : .....	الاسم : .....		
PROFESSION : .....	المهنة : .....		
ADRESSE FISCALE : .....	عنوان الجبائي : .....		
ADRESSE COURRIER : .....	عنوان البريد : .....		
N° PIECE D'IDENTITE : .....	رقم بطاقة التعريف : .....		
DATE ET LIEU DE NAISSANCE : .....	تاريخ ومكان الولادة : .....		
NATIONALITE : .....	الهاتف : .....	TEL. : .....	الجنسية : .....
MANDATAIRE	المفوض	Signature	الامضاء
M : .....	السيد : .....		
DATE : .....	تاريخ : .....		
M : .....	السيد : .....		
DATE : .....	تاريخ : .....		
M : .....	السيد : .....		
DATE : .....	تاريخ : .....		
M : .....	السيد : .....		
DATE : .....	تاريخ : .....		

Imp. B.A.D.R. - Alger

Réf. : CA 10 / CA 10 Bis / ARCH. 15 Ans

(Noter Instructions et Contentieux au Verso)

## Annexe 2 : Documents pour l'ouverture d'un compte

**OUVERTURE D'UN COMPTE ▼****POUR LES PARTICULIERS**

- Présentation d'une pièce d'identité en cours de validité
- Document officiel établissant la preuve de l'adresse

**POUR LES PROFESSIONNELS**

- Présentation d'une pièce d'identité en cours de validité du gérant et cogérant
- Document officiel établissant la preuve de l'adresse
- Copie des statuts de l'entreprise, du registre de commerce et/ou agrément
- Justificatif des numéros d'identification fiscale et statistique

# Venez

**ouvrir un compte à la BADR  
& bénéficier de nos produits  
et services**

*Pour plus d'informations,  
rapprochez-vous de l'agence BADR la plus proche.*

## Annexe 3 : Fiche d'ouverture Compte chèque

Dossier .....

ملف .....

Compte n°

--	--	--	--

حساب رقم

--	--	--	--

Je soussigné (nom et prénom - la femme mariée indique le nom et le prénom du mari suivis de ses nom et prénoms de jeune fille)

انى الموقع اسفله ( اللقب والاسم، المرأة المتزوجة تذكر لقب واسم زوجها متبعين بلقبها واسمها قبل الزواج )

agissant (en mon nom personnel) ou (en qualité de président, trésorier, etc... de l'Association... ou autre personne morale désirant ouvrir le compte)

المتصرف ( باسمي الشخصي ) أو ( بصفتي رئيسا أو أمين مال ،، الخ . للجمعية ،، أو أي شخص معنوي آخر يرغب في فتح حساب )

Date et lieu de naissance (1)

تاريخ ومكان الولادة (1)

Nationalité (1)

الجنسية (1)

Profession (1)

المهنة (1)

Adresse fiscale

العنوان الجبائي

Autre adresse (courrier)

عناوين أخرى ( البريد )

N° de téléphone

رقم الهاتف

Pièce d'identité présentée (nature, numéro, date, lieu, autorité l'ayant délivrée)

بطاقة التعريف المقدمة ( نوعها، رقمها، تاريخها، مكان تسليمها، السلطة التي سلمتها )

Documents (2) remis, à remettre : statuts, journaux de publications, pouvoirs, etc...)

الوثائق (2) (التي سلمت والتي ستسلم : القانون الاساسي منشورات الاعلانات الشرعية، الفويضات ،، الخ . )



## Annexe 4 : Convention de compte courant bancaire



بنك الفلاحة و التنمية الريفية  
Banque de l'Agriculture et du Développement Rural

**BANQUE DE L'AGRICULTURE ET DU DEVELOPPEMENT RURAL**

**CONVENTION DE COMPTE COURANT BANCAIRE**

Les soussignés :

La Banque de l'Agriculture et du Développement Rural, société par actions au capital social de trente trois (33) milliards de dinars, immatriculée au Centre National du Registre de Commerce d'Alger sous le N° 00B 0011640, ayant son siège social au 17, Boulevard Colonel Amirouche- Alger.

Représentée par :

en sa qualité

de : .....

ci-après désignée la Banque d'une part

Et,

le client : .....

Représenté par (pour les personnes morales) :

.....  
.....

en sa qualité de :

.....

en vertu des Pouvoirs statutaires

du : .....

ci-après désigné le Client ou le Titulaire du compte d'autre part

ont convenu ce qui suit :

La Banque ouvrira un compte courant au nom du client qui sera régi par les conditions générales de la présente convention ainsi que par les conditions particulières stipulées sur le formulaire d'ouverture du compte de dépôt.



En cas de refus du client d'accepter les modifications, la Banque ou le Client peut résilier la convention.

#### **Article 45 — Renseignements - Réclamations**

##### **45.1 Renseignement**

Le Client peut obtenir tous renseignements sur le fonctionnement de son compte auprès de son agence domiciliataire.

##### **45.2 Réclamations**

Pour toutes ses réclamations, le Client peut s'adresser au service qui lui sera indiqué à cet effet par son agence.

#### **IX- / ELECTION DE DOMICILE – LOI APPLICABLE – LITIGES**

##### **Article 46 –Election de domicile**

Pour l'exécution de la présente convention, les parties font élection de domicile à leurs adresses respectives indiqués ci-dessus.

##### **Article 47 – Loi applicable**

La loi applicable à la présente convention est la loi algérienne.

##### **Article 48 – Règlement des litiges – Juridiction compétente**

En cas de litiges, la banque et le client titulaire du compte rechercheront une solution amiable. A défaut, le litige sera porté devant la juridiction dans le ressort de laquelle se trouve le siège de l'agence domiciliataire du compte

Fait à .....le.....

Signature du client  
Avec apposition de la mention  
«lu et approuvé»

Signature pour la banque