

Projet de fin d'études  
POUR L'OBTENTION DU DIPLÔME  
D'INGÉNIEUR D'ÉTAT EN TÉLÉCOMMUNICATIONS

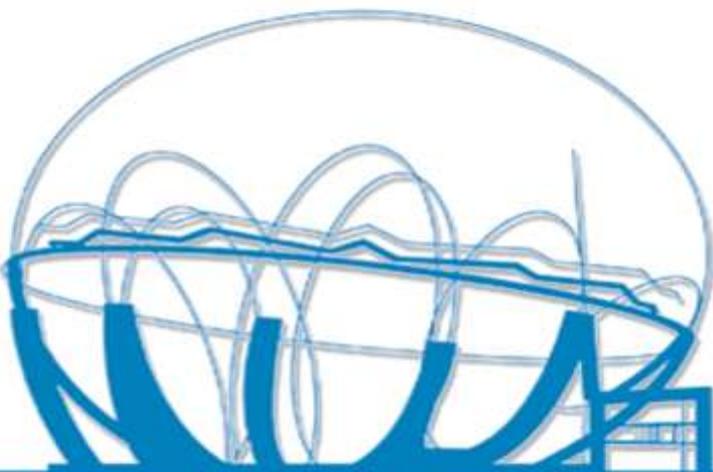
**Migration de Sage Ligne 100 vers  
OpenERP et la réalisation d'une  
solution BI**

Réalisé par :

**M. Tizki Riyad**

Membres de Jury :

**M. Bellafkih Moustafa (Président Jury)**  
**M. Zaouia Abdelilah (INPT)**  
**M. Oubrich Mourad (INPT)**  
**M. Sarhani Saad (RIBATIS)**



AGENCE NATIONALE DE RÉGLEMENTATION DES TÉLÉCOMMUNICATIONS  
INSTITUT NATIONAL DES POSTES ET TÉLÉCOMMUNICATIONS  
PROJET DE FIN D'ÉTUDES EN CYCLE INE

Jun 2011

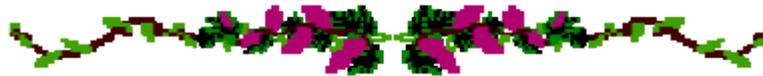
## Dédicace

### A mes très chers parents



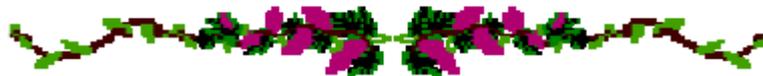
Dont leurs mérites, leurs sacrifices, leurs qualités humaines m'ont permis de vivre ce jour :  
Les mots me manquent pour exprimer toute la reconnaissance, la fierté et le profond amour que je vous porte pour les sacrifices qu'ils ont consenti pour ma réussite, qu'ils trouvent ici le témoignage de mon attachement ma reconnaissance, gratitude et respect, que dieu leur préserve bonne santé et longue vie. Tous mes sentiments de reconnaissance pour vous.

### A ma sœur



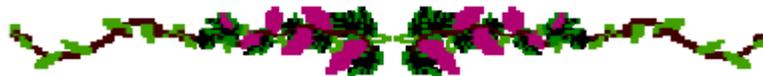
J'espère atteint le seuil de tes espérances. Que ce travail soit l'expression de ma profonde affection Je te remercie pour le soutien moral et l'encouragement que tu m'as accordés .Je te souhaite tout le bonheur que tu mérites.

### A ma famille



Que je ne pourrais nommer de peur d'en oublier mon attachement et mes affections les plus Sincères.

### A mes ami(e) s



A tout ceux qui ont su m'apporter aide et soutien aux moments propices, Je dédie ce travail, reconnaissant et remerciant chaleureusement.

*TIZKI Riyad*

## Remerciement

*J'adresse tous mes remerciements les plus vifs à M. **BENACHOU Faical** de m'avoir donné la chance de poursuivre mon projet de fin d'étude au sein de RIBATIS.*

*J'exprime ma très grande gratitude auprès de mon encadrant au sein de RIBATIS M. **SARHANI Saad** pour son aide gracieux, pour sa disponibilité et sa bienveillance.*

*Je remercie mes professeurs et encadrants internes Mr **ZAOUIA Abdelilah**, Mr **OUBRICH Mourad** et Mr **BELLAFKIH Moustafa** qui n'ont épargné aucun effort pour me soutenir et m'orienter tout au long de la période du PFE, Je leur remercie pour leurs conseils dans l'élaboration du présent rapport. Mes remerciements vont à l'ensemble des enseignants de l'INPT pour leurs contributions à ma formation.*

*Un grand merci pour ma très chères familles et aussi mes très chers amis au sein de l'INPT et à toute personne ayant contribué, du pré ou de loin, au bon déroulement de ce stage de fin d'études.*

## Résumé

Le présent rapport constitue le résultat d'un travail réalisé dans le cadre du projet de fin d'études, au sein de cabinet de conseil RIBATIS.

Notre projet de fin d'étude s'inscrit dans le cadre d'un projet réel (**e-dmaj BI**) réalisé en environnement pilote pour le compte d'un Client de RIABTIS. En effet, il s'agit d'assurer la « Migration du système d'information existant, propre au client et qui est intégré sous l'ERP SAGE, vers un futur système : le progiciel de gestion intégré, OpenERP » et cela pour une meilleure maîtrise/ gestion des processus support et métier de l'entreprise cliente ; ainsi qu'une exploitation des données migrées, en partie, pour instaurer un système décisionnel de Reporting / Tableau de bord.

## Abstract

The following report is the outcome of the work done for the "End of studies" project within the company RIBATIS.

Our final project study is part of a real project (E-Dmaj BI) made pilot environment on behalf of a client RIABTIS, Indeed, this is to ensure the "Migration of existing information system, client-specific and is integrated in the ERP SAGE towards a future system: ERP, OpenERP" and for a better control / process management and business support of the client company, and exploitative of migrated data, in part, to establish a decision-Reporting / Dashboard.

## ملخص

هذا التقرير هو تتويج لعمل في إطار مشروع السنة الختامية من سلك مهندسي دولة التي تم إنجازها في شركة " ريباتيس " .

يدخل هذا المشروع في إطار مشروع حقيقي (إدماج) لصالح أحد زبناء شركة " ريباتيس " ويهدف هذا المشروع إلى نقل المعلومات الموجودة في برنامج المحاسبة سايج خط 100 إلى برنامج تخطيط الموارد المنشأة "أوبن أورب " وذلك لتمكين الزبون من تطوير و هيكله عملياته. وكذلك من إنجاز نظام دعم لاتخاذ القرار عن طريق لوحة التحكم.

## Table des matières

Introduction Général.....	8
Chapitre I : Présentation générale du projet.....	11
<b>1. Présentation générale du projet</b> .....	12
1.1.1 Domaines d'activités :.....	12
1.1.2 Organisation :.....	14
Chapitre II :Analyse et Spécification des besoins.....	20
<b>2. Analyse et Spécification des besoins</b> .....	21
2.1 Etude de l'existant :.....	21
2.1.1 Volet fonctionnel :.....	21
2.1.2 Volet technique :.....	22
2.1.3 Volet technico fonctionnel :.....	23
2.1.3.1 SAGE ligne 100 :.....	23
2.1.3.2 OpenERP OLAP :.....	24
2.2 Spécification des besoins métiers :.....	25
Chapitre III : Etudes comparatives des outils décisionnels.....	27
<b>3. Etudes comparatives des outils décisionnels</b> .....	28
3.1 Choix de l'ETL :.....	28
3.2 Choix de l'outil de restitution de données :.....	37
3.2.1 Contraintes de l'étude comparative :.....	37
3.2.2 Présentation du « short list » préétabli :.....	38
3.2.3 Conformité des alternatives par rapport aux contraintes :.....	41
3.2.4 Bilan du comparatif des outils de restitution de données :.....	41
Conclusion :.....	42
Chapitre VI : Conception de la Solution.....	43
<b>4. Conception de la Solution</b> :.....	44
4.1 Conception de la montée en version (OpenERP v5 vers OpenERP v6) :.....	44
4.1.1 Reverse engineering de l'OpenERP 5 :.....	44
4.1.2 Reverse engineering de l'OpenERP 6 :.....	51

4.1.3	Mapping entre OpenERP v5 et OpenERP v6 : .....	53
4.2.1	Cartographie des tables Sage Ligne 100 (Module Comptable).....	54
4.2.2	Reverse engineering OpenERP 6 : .....	63
4.2.3	Mapping entre Sage Ligne 100 et OpenERP v6 : .....	63
4.3.1	Conception du modèle multidimensionnel : .....	63
4.3.1.1	Axes d'analyse et indicateurs : .....	63
4.3.1.2	Modèle de conception : .....	67
4.3.1.3	Structure du Datawarehouse : .....	68
4.3.1.4	Tables de dimensions : .....	70
4.3.1.5	Table de faits : .....	70
4.3.2	Conception de l'ETL : .....	70
Chapitre V : Réalisation de la solution .....		72
5.	Réalisation de la solution .....	73
5.1	Environnement de développement : .....	73
5.2	Solution finale : .....	73
5.2.1	Mise en place de la montée en version OpenERP (v5 → v6) : .....	73
5.2.2	Mise en place de la migration Sage Ligne 100 vers OpenERP 6.....	76
5.2.3	Mise en place du Datawarehouse : « Gestion_Activité_RIBATIS ».....	81
5.2.3.1	Réalisation de la phase ETL : .....	81
5.2.3.2	Restitution du cube OLAP : .....	82
5.2.3.3	Restitution des tableaux d'analyses croisées .....	83
5.2.3.4	Restitution des tableaux de bord : .....	84
Glossaire : .....		87

## Table des Figures

Figure 1 Organisation de RIBATIS.....	14
Figure 2 : Diagramme GANTT du projet .....	18
Figure 3 :Exemple d'enregistrement d'un compte sous l'ERP SAGE ligne 100 .....	23
Figure 4: Exemple d'un tableau de bord sous l'interface Web du module Open ERP OLAP .....	25
Figure 5 : Diagramme représentant les coûts des solutions en fonction du temps passé .....	28
Figure 6 : Tendence d'intérêt sur le Web vis-à-vis des 5 ETLs (long list).....	29
Figure 7 : Schéma récapitulatif de la suite décisionnelle du projet .....	38
Figure 8 : Connexion entre PostgreSQL et PowerAMC .....	44
Figure 9 : Sélection des tables concernées par le Reverse Engineering.....	45
Figure 10 : MCD du module comptabilité et finance (résultat du Reverse Engineering) .....	46
Figure 11 : MCD du module ressources humaines (résultat du Reverse Engineering).....	47
Figure 12 : Localisation des tables de la paie des employées dans le MCD du module RH .....	48
Figure 13 : Regroupement des tables de la paie (indépendantes des employées) .....	48
Figure 14 : MCD du module produit (résultat du Reverse Engineering).....	49
Figure 15 : Liaison des tables du module production_Ribatis avec le module Partenaires.....	50
Figure 16 : MCD du module partenaires (résultat du Reverse Engineering) .....	51
Figure 17 : Le champ Company_id ajouté aux tables de l'Open ERP .....	52
Figure 18 : Le concept du level d'un compte implémenté sous l'OpenERP 6.....	53
Figure 19 : Environnement de l'ERP Sage ligne 100 Comptabilité .....	55
Figure 20 : Conception multidimensionnelle du modèle .....	69
Figure 21 : Administration des modules sous l'OpenERP V6 .....	74
Figure 22 : Exemple d'un Job de migration sous l'ETL Talend (OpenERP V5 vers V6).....	75
Figure 23: Exemple du contenu du module comptabilité visualisé sous l'environnement cible de la migration (OpenERP V6), sous la machine serveur .....	76
Figure 24 : Le module Comptabilité (schéma cible) installé sous l'OpenERP V6 .....	76
Figure 25 : Assistant d'installation du PiloteODBC_SAGE Ligne100_Version14.04 .....	77
Figure 26: Configuration de la source de données « SAGE ligne 100 comptabilité » .....	77
Figure 27 : Etablissement de la connexion entre Talend et la source de données « SAGE ligne 100 » .....	78
Figure 28:Exécution du script PL-pgSQL pour la résolution du problème parent right / parent left....	79
Figure 29:Exemple d'un Job de migration sous l'ETL Talend (SAGE ligne 100 vers OpenERP6) .....	79
Figure 30 : Automatisation des Jobs de la migration SAGE ligne 100 vers OpenERP V6 sous Talend.....	80
Figure 31:Planification automatique de l'exécution des Jobs de la migration sous Talend .....	80
Figure 32 : Datawarehouse relationnel alimenté depuis les sources de données croisées.....	81
Figure 33:Flux de chargement de la dimension collaborateur sur l'outil Palo.....	81
Figure 34:Flux de chargement de la table de fait (dimensions+indicateurs calculés) sur l'outil Palo ..	82
Figure 35:Schématisation du cube OLAP sous JPalo Client.....	83
Figure 36:Exemple de tableau d'analyses croisées sous JPalo Client .....	84
Figure 37 : Tableau de bord circulaire illustrant la répartition des projets selon leur typologie .....	85
Figure 38 : Tableau de bord illustrant la l'implication des collaborateurs dans les projet .....	85
Figure 39 : Tableau de bord en histogramme illustrant la répartition des Clients par famille de projets lors du quatrième trimestre de l'année 2010 .....	86

## Table des Tableaux

Tableau 1: Eléments comptables à récupérer depuis SAGE ligne 100.....	24
Tableau 2: Tables du module comptabilité et finance de l'OpenERP 5 .....	45
Tableau 3: Tables du module ressources humaines de l'OpenERP 5.....	46
Tableau 4: Tables du module de paie de l'OpenERP 5.....	48
Tableau 5: Tables du module produit de l'OpenERP 5.....	49
Tableau 6: Tables du module de production_RIBATIS sous l'OpenERP 5 .....	50
Tableau 7: Tables du module partenaires de l'OpenERP 5 .....	51
Tableau 8: Cartographie de la table P_Devises sous Sage ligne 100 comptabilité.....	56
Tableau 9: Cartographie de la table F_Contactt sous Sage ligne 100 comptabilité.....	57
Tableau 10: Cartographie de la table F_Compteg sous Sage ligne 100 comptabilité .....	58
Tableau 11: Cartographie de la table F_Taxe sous Sage ligne 100 comptabilité .....	59
Tableau 12: Cartographie de la table F_Banque sous Sage ligne 100 comptabilité .....	60
Tableau 13: Cartographie de la table F_Journaux sous Sage ligne 100 comptabilité .....	61
Tableau 14: Cartographie de la table F_Ecriturec sous Sage ligne 100 comptabilité .....	62
Tableau 15: Correspondence du mapping Sage ligne 100 ☐ OpenERP6 (modle comptable) .....	63
Tableau 16: Définition des dimensions .....	65
Tableau 17: Récapitulatif des indicateurs .....	67
Tableau 18: Comparaison entre les deux modèles de conception .....	68

## Introduction Général

Notre projet de fin d'étude s'inscrit dans le cadre d'un projet réel (**BI e-dmaj**) réalisé en environnement pilote pour le compte d'un Client de RIABTIS. En effet, il s'agit d'assurer la « Migration du système d'information existant, propre au client et qui est intégré sous l'ERP SAGE, vers un futur système : le progiciel de gestion intégré, OpenERP » et cela pour une meilleure maîtrise/ gestion des processus support et métier de l'entreprise cliente ; ainsi qu'une exploitation des données migrées, en partie, pour instaurer un système décisionnel de Reporting / Tableau de bord.

Ainsi, notre mission consistait, entre autres, à concevoir en premier lieu un module de reprise de données vers l'architecture cible : OpenERP en assurant la compatibilité avec l'ancien système .Et en second lieu il s'agissait de mettre en place une solution d'analyse de données financières / et de production pour une génération de rapports dynamiques et de tableau de bord dédiés à la mesure de performance de l'activité interne de la société.

Notre contribution dans ce grand projet touchait essentiellement au métier : comptabilité/immobilisations. A cet effet, l'objectif majeur de notre tâche était de veiller à la récupération de plusieurs éléments fonctionnels concernant ce double volet, depuis l'ERP source : Ecritures comptables, partenaires, référentiels articles... puis de les adapter aux contraintes techniques et fonctionnelles du nouvel environnement à savoir la Bdd liée au progiciel Open ERP. Ces données seront manipulées par la suite par l'équipe de projet « Conception et développement » dans le cadre d'un paramétrage et/ou développement spécifique au sens d'intégration du nouveau SI Client.

Par ailleurs, leur croisement avec des données de production propres à la société alimentera un entrepôt de données décisionnel donnant naissance par la suite à des tableaux croisés avec des axes d'analyse variés qui répondent aux exigences du manager.

## Chapitre I : Présentation générale du projet

Dans la première section de ce chapitre, nous présenterons l'organisme d'accueil, à savoir le cabinet de conseil opérationnel RIBATIS, son organisation, son domaine d'activité et les différents projets qu'il réalise. La deuxième section donnera dans un premier temps une description générale du projet ainsi que les problématiques qui lui sont liées et terminera par préciser ses objectifs. La dernière section étoffera la démarche et le planning adoptés pour mener à bien ce travail.

# 1. Présentation générale du projet



## 1.1 Présentation de l'organisme d'accueil :



RIBATIS est un cabinet de consulting opérationnel fondé en Janvier 2007 agissant dans le domaine de l'organisation, du management et de la mise en œuvre des systèmes d'information et de communication.

Avec un positionnement, alliant le recul des cabinets de conseil et le pragmatisme des sociétés d'ingénierie et de services informatiques, RIBATIS représente un allié, à la fois des Directions Systèmes d'Information et des Directions Métier pour assurer un alignement optimal entre la stratégie métier et l'avantage indéniable que représente les technologies de l'information et de la communication.

### 1.1.1 Domaines d'activités :

RIBATIS offre divers pratiques du secteur, propose des démarches pratiques, pragmatiques et personnalisées pour atteindre les objectifs fixés conjointement avec ses clients.



#### **Consulting :**

Un ensemble de services élaborés et optimisés sur la base de l'expertise de ses consultants, les feedback de ses clients et les activités de veille anticipative qu'ils exercent de façon continue.

- Assistance à maîtrise d'ouvrage/œuvre (AMOA) et (AMOE).
- Audit projets systèmes d'information et réseaux.
- Business process engineering et reengineering (BPE/BPR).
- Organisation & conduite de changement.
- Plan de transformation SI et Télécom.
- PMO : Pilotage de projets et programmes.
- Systems selection.



#### **Technology :**

Ils sont convaincus au sein de RIBATIS que l'utilisation des technologies informatiques et télécom n'est pas une fin en soi. Ces dernières ne sont là que pour supporter le développement business de leurs clients.

La maîtrise technologique des consultants techniques couplée avec le sens métier des consultants fonctionnels, positionne le cabinet comme allié stratégique pour l'alignement de l'infrastructure SI avec les enjeux métier des ses clients.

- CRM : gestion relation client.
- Développement spécifique.
- E-dmaj ®: SI intégré pour PME.
- ERP : progiciel de gestion intégrée.
- Portails, intranets et e-business.
- Travail collaboratif & gestion documentaire.



### **Training :**

RIBATIS propose un ensemble de séminaires et de formations couvrant une cible très variée, allant du simple utilisateur jusqu'à l'expert, en passant par les professionnels des directions SI et des directions métier.

Ses consultants formateurs proposent 4 types de formations permettant d'adresser chaque besoin de façon simple, pratique et efficiente :

- **Quick Start kits :**

Des séminaires de 2 à 3 jours dont l'objectif est de former les participants sur un des métiers relatifs aux systèmes d'information (DSI, Chef de Projets, Responsable Exploitation, MOE, MOA,...). Ces séminaires se distinguent par leur côté pratique et pragmatique permettant aux participants d'être opérationnels très rapidement en les munissant de pratiques et d'outils ayant fait leurs preuves.

- **Tours d'horizon :**

Une collection de séminaires de 3 à 4 jours ayant pour objectif de passer en revue avec les participants l'ensemble des facettes d'un concept lié aux technologies de l'information (ERP, CRM, BPR, ...)

- **Etats de l'Art :**

Des formations approfondies de 4 à 5 jours mettant le focus sur un concept, une technologie, une approche, ... Ces formations sont destinées aux initiés désirant approfondir leurs connaissances et les confronter à l'état de l'art en la matière.

- **Référentiels qualité :**

Ces séminaires de 2 jours se proposent d'explorer avec les participants quelques uns des référentiels qualité les plus réponsus dans le domaine du management des systèmes d'informations : CMM-I, ITIL, ISO, Six Sigma.

### 1.1.2 Organisation :

RIBATIS s'appuie sur son équipe interne de consultants qualifiés ainsi qu'un réseau de partenaires-expert nationaux et internationaux.

L'équipe projet :

- Directeur de projet SI (1)
- Assistante de projet (1)
- Senior consultant B.I (1)
- Chef de projet technique (1)
- Consultant fonctionnel senior (1)
- Ingénieur conception et développement senior (1)
- Ingénieur conception et développement (2)
- Consultant technico-commercial (1)

L'organigramme ci-dessous (cf. Figure 1) présente les différentes directions de RIBATIS :

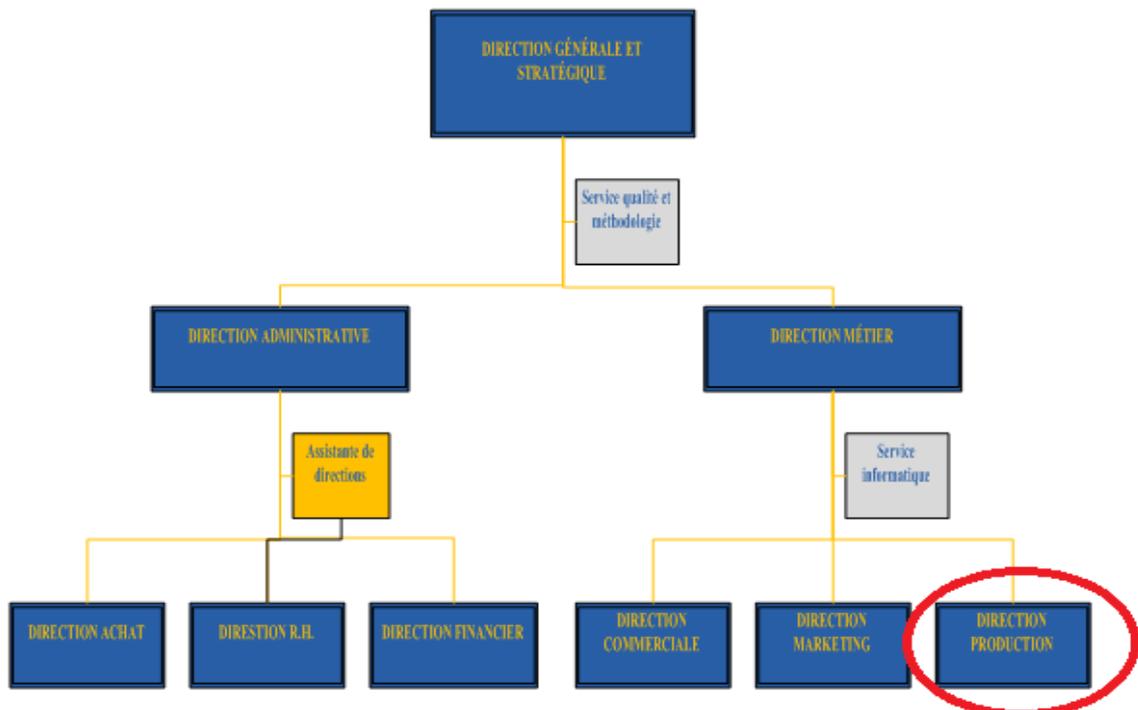


Figure 1 Organisation de RIBATIS

Notre stage de fin d'étude s'est déroulé dans l'entité production relative à la direction métier.

## 1.2 Contexte et objectifs du projet :

Notre projet de fin d'étude s'inscrit dans le cadre d'un projet réel (**e-dmaj**®: une solution informatique complète pensée et mise en place à partir de plusieurs dizaines de diagnostics SI réalisés auprès de PME marocaines pour la refonte de leurs systèmes d'information ), réalisé en **environnement pilote** pour le compte de RIBATIS.

Pour les clients de RIBATIS il s'agira d'assurer la « Migration du système d'information existant et qui est intégré soit sous l'ERP **OpenERP V5** soit sous la plateforme **SAGE ligne 100**, vers un futur système : le progiciel de gestion intégré, **OpenERP V6** » et cela pour une meilleure maîtrise/ gestion des processus support et métier de l'entreprise cliente ; ainsi qu'une exploitation des données migrées, en partie, pour instaurer un système décisionnel de Reporting / Tableaux de bord.

En effet, notre contribution dans ce grand projet touche essentiellement au métier : comptabilité/immobilisations. A cet effet, l'objectif majeur de notre tâche est de veiller à la récupération de plusieurs éléments fonctionnels concernant ce double volet, depuis l'ERP source SAGE ligne 100 (Ecritures comptables, partenaires, référentiels articles...), puis de les adapter aux contraintes techniques et fonctionnelles du nouvel environnement à savoir la BD liée au progiciel Open ERP V6. Ces données seront manipulées par la suite par l'équipe de projet « Conception et développement » dans le cadre d'un paramétrage et/ou développement spécifique au sens d'intégration du nouveau SI Client.

Par ailleurs, leur croisement avec des données de production propres à la société alimentera un entrepôt de données décisionnel donnant naissance par la suite à des tableaux croisés avec des axes d'analyse variés qui répondent aux exigences du manager. A cet effet, signalons que les décideurs disposent déjà d'une solution décisionnelle intégrée à l'OpenERP mais qui présente beaucoup de limitations.

Ainsi, notre mission consiste, entre autres, à concevoir en premier lieu un module de reprise de données vers l'architecture cible : OpenERP V6 en assurant la compatibilité avec l'ancien système ( OpenERP V5 ou SAGE ligne 100). Et en second lieu il s'agit de mettre en place une solution d'analyse de données financières / et de production pour une génération de tableaux d'analyse croisées dynamiques et de tableaux de bord dédiés à

la mesure de performance de l'activité interne de RIBATIS. Cette solution pourrait ainsi être adaptée à l'environnement d'un Client RIBATIS.

*En résumé, on pourrait décliner la finalité du travail qui nous a été assigné en les objectifs suivants :*

- Réaliser la montée en version (open ERP 5 → open ERP 6) couvrant ainsi tous les modules : Comptabilité , Ressources Humaines, Paie ...
- Concevoir un mapping entre SAGE ligne 100 et Open ERP6 au niveau du module comptabilité et procéder à la migration des données entre les deux plateformes ( depuis SAGE vers OpenERP ).
- Mettre en place un datawarehouse qui sera alimenté par le croisement des données de production issues de PostgreSQL (données journalières) avec les données financières émanant du module comptable de l'open ERP 6 (résultats de la migration depuis SAGE ligne 100).
- Elaboration de tableaux d'analyses croisées et de tableaux de bord adaptés à l'ergonomie exigée par le manager (filtres, forme des jauges, typologie des graphes ...). Ces tableaux ayant pour objectif l'analyse de l'écart entre le prévu et le réalisé en terme de l'activité interne du cabinet de conseil opérationnel RIBATIS .Via une démarche similaire et moyennant les données réelles ces même tableaux de bords pourront répondre aux besoins du Client.

## 1.3 Conduite du projet :

### 1.3.1 Démarche :

Pour bien mener notre mission, il est judicieux de suivre une démarche empreinte à la fois par l'aspect **migration de données** et le cycle classique d'un **projet décisionnel**, tout en optant pour une **conduite itérative** concernant les livrables ; et cela pour une meilleure agilité en terme de gestion de projet.

Pour ce qui est de la **migration**, la démarche sera décortiquée en deux principales phases :

- ✓ Extraction : l'objectif de cette phase est de concevoir une méthode d'extraction de données depuis l'ERP source : SAGE (SI existant du client).
- ✓ Alimentation : cette phase du projet s'articule autour des deux points suivants :
  - ✓ Elaboration d'une cartographie du schéma cible (base de données à alimenter, liée à Open ERP).
  - ✓ Mise en place d'un pack de transformations effectuées par un outil ETL (voir ci dessous) dans une optique d'adapter les deux schémas (source : SAGE, et cible : OpenERP).

Pour ce qui de la **partie décisionnelle**, la démarche sera conforme au cycle décisionnel classique :

- ETL (Extraction / Transformation / Chargement) :
  - ✓ Extraction des données sources : Résolution des problèmes d'intégration et de qualité des données depuis la source vers la cible.
  - ✓ Transformation : Application de filtres, agrégations, traitement des données manquantes ou aberrantes et contrôle des rejets, intégrité et cohérence.
  - ✓ Chargement des données : Synchronisation des chargements, transfert de fichiers ou transfert de base à base.
- Mise en place du Datamart « Gestion d'activités » (datawarehouse orienté métier) : Mesure de performance interne d'un Client RIBATIS.
- Alimentation des cubes OLAP.
- Elaboration des tableaux de bord.

Dans le cadre de notre projet, il sera aussi question de procéder à des **études comparatives** pour le choix des outils décisionnels open source (ETL et Outil de reporting ) les mieux adaptés. Le Chapitre III se chargera de décrire la logique et les résultats de ces études.

### 1.3.2 Planning :

La planification du projet est une phase importante d'avant-projet. Elle consiste à prévoir le déroulement du projet tout au long des phases constituant le cycle de développement (démarche adoptée). Grâce aux réunions tenues avec les intervenants au sein de RIBATIS, nous avons pu planifier dans le temps les taches soulevées précédemment (cf. section 3.1) :

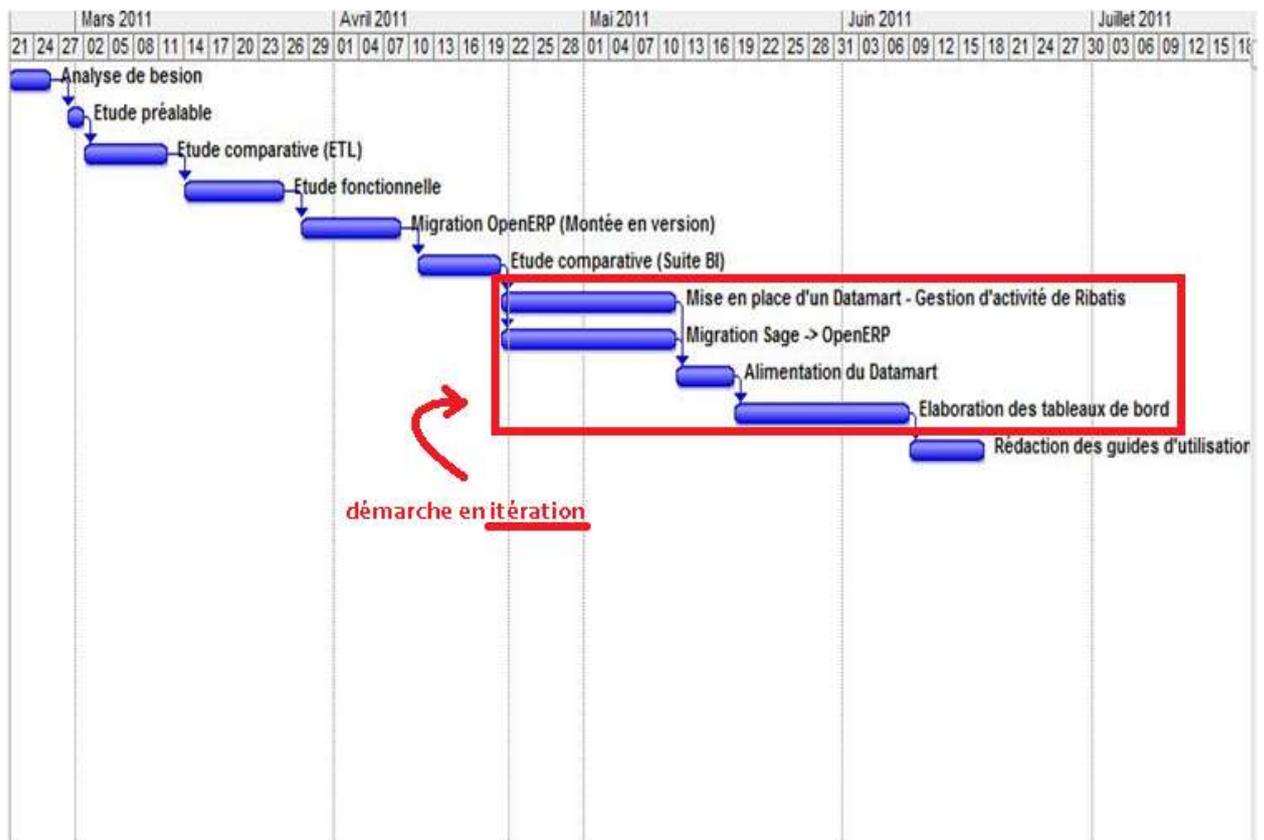


Figure 2 : Diagramme GANTT du projet

### 1.3.3 Livrables :

*En terme de livrables, les résultats attendus peuvent se résumer en :*

- Cartographie du schéma source du module comptabilité de l'OpenERP.
- Templates d'alimentation. (Mapping schéma SAGE → schéma Open ERP).
- Pack des transformations de l'ETL choisi (Jobs automatisés).
- Guides d'utilisation et de paramétrage pour l'alimentation de l'OpenERP 6 (depuis version 5/depuis SAGE).
- Guide d'utilisation des outils décisionnels open sources choisis (ETL+ Outil de restitution).

### Conclusion :

Dans ce premier chapitre, on a présenté le contexte général du projet et identifié ses objectifs majeurs. En l'occurrence la migration de données (montée en version de l'OpenERP V5 → V6 et mapping entre deux plateformes différentes SAGE → OpenERP ), ainsi que la mise en œuvre d'un système décisionnel exigé par le manager : « Gestion\_Activité » dédié à la gestion de performance de l'activité interne d'un Client RIBATIS. Ce chapitre a aussi justifié la démarche adoptée dans ce projet empreint par l'aspect migration de données, le cycle décisionnel et l'aspect itératif quant à la remise des livrables. En conclusion le chapitre a précisé le planning et les livrables visés dans le cadre de notre travail.

Le chapitre suivant sera dédié à l'analyse technico fonctionnelle du projet ainsi que la spécification des besoins métiers.

## Chapitre II :Analyse et Spécification des besoins

Dans la première section de ce chapitre, nous procéderons à l'analyse technico-fonctionnelle de l'existant de notre projet en terme de migration de données et relativement à la solution décisionnelle actuelle et des limitations qu'elle présente. Ce chapitre terminera en spécifiant les besoins métier du projet.

## 2. Analyse et Spécification des besoins

### 2.1 Etude de l'existant :

#### 2.1.1 Volet fonctionnel :

La version actuelle de l'OpenERP (**version 5**) avec laquelle travaillent les collaborateurs de RIBATIS a fait preuve d'un manque de flexibilité (dépendance forte entre les modules support), ainsi que des défauts de navigation au niveau de la version Web. Ces problèmes entravent le bon déroulement des projets d'intégration SI (paramétrage et/ou développement spécifique).

A cet effet, et afin de suivre l'évolutivité de la communauté de l'OpenERP. La direction a décidé de procéder à une montée en version de son **système de production interne** vers une intégration sous le progiciel **OpenERP 6** (ultime version). Ainsi la version 6.0 s'avère comme une vraie 'suite d'application métier' et non un ERP classique ; avec moins d'interdépendance entre les modules support et une seule interface centralisée pour toutes les applications.

Cette migration devrait inclure toutes les données liées aux modules support (comptabilité, RH, achats...) du système de production en question. Le système migré constituera le nouveau framework de travail.

D'autre part, Open ERP étant le *framework paramétrable* sur lequel se base le cabinet pour déployer des solutions SI à ses clients, alors le besoin d'utiliser les modules supports s'avère récurrent. En l'occurrence, le module comptabilité, pour la plupart des entreprises clientes, est intégré sous un autre progiciel à savoir **SAGE ligne 100**. D'où le besoin d'instaurer une moulinette de migration de données vers le nouvel environnement : OpenERP 6.

Dans notre cas, il s'agit de concevoir un mapping fonctionnel entre le schéma source : les **données comptables** telles qu'elles sont enregistrées sous l'ERP SAGE ligne 100, et le schéma cible (données de comptabilité sous l'OpenERP). Ces éléments comptables à migrer étant les suivants :

- Comptes comptables
- Ecritures comptables
- Journaux
- Devises
- Taxes
- Partenaires / adresse partenaires

Le croisement des données de production et des données comptables peut être exploité au niveau d'un module BI intégré sous l'OpenERP, pour la mesure de la performance interne de RIBATIS ou de son Client. Cependant, ce module est limité en termes de dynamique des axes d'analyse, et présente également un manque au niveau de l'ergonomie personnalisée des tableaux de bord. La solution cible (**système décisionnel exigé**) devrait combler ces lacunes.

### 2.1.2 Volet technique :

La cartographie du schéma source du module comptable de l'ERP SAGE ligne 100 s'avère foncièrement hétérogène par rapport à celle de l'OpenERP. En effet, et à titre d'exemple, les 'comptes' de la comptabilité générale sont décrits par les tables:

- Account\_account : sous OpenERP v6 (SGBD : PostgreSQL)
- F\_COMPTEG : sous SAGE ligne 100

La moulinette de migration à réaliser doit mapper, en particulier, les structures des deux tables d'une façon à ce qu'un compte enregistré initialement sous le système SAGE soit visible et modifiable sous l'interface de l'OpenERP.

En second lieu, les dirigeants de RIBATIS recevaient jusqu'à lors des tableaux de bord, intégrés à Open ERP, présentant de grosses limites en termes d'interactivité ce qui réduit les champs d'action.

Un des objectifs de la solution cible est donc de pouvoir instaurer un système de reporting / tableaux de bord **interactif** et **autonome**, de manière à donner aux dirigeants la possibilité d'assurer une navigation avec une grande flexibilité au niveau des axes d'analyse.

## 2.1.3 Volet technico fonctionnel :

### 2.1.3.1 SAGE ligne 100 :

S'étendant de la gestion comptable et financière à la gestion du recouvrement, le progiciel de gestion intégré Sage est destiné aux PME pour une amélioration de leur productivité mais également pour faire face à des problématiques telles que le risque client ou encore l'évaluation de la performance commerciale et financière de l'entreprise.

Pour les moyennes entreprises, Sage ligne 100 propose l'ensemble des éléments constitutifs d'une gestion comptable et financière. Gestion de la comptabilité, du recouvrement, de la trésorerie, des immobilisations et des moyens de paiement, des outils d'aide à la décision et la toute dernière solution de dématérialisation de factures, font de la gestion comptable et financière de Sage 100 un véritable centre de pilotage des données de l'entreprise axé sur le contrôle et l'analyse d'indicateurs clés dans une optique d'amélioration de la productivité. [ .... ]

Dans le cas du cabinet de conseil opérationnel RIBATIS, tout comme la majorité de ses entreprises clientes, la revue fiduciaire (informations fiscales, comptables..) est intégrée sous l'ERP **SAGE ligne 100 v14**. Ce logiciel assure la comptabilité générale, auxiliaire, analytique et budgétaire.

La figure ci-dessous, illustre l'enregistrement d'un compte sous l'ERP SAGE :

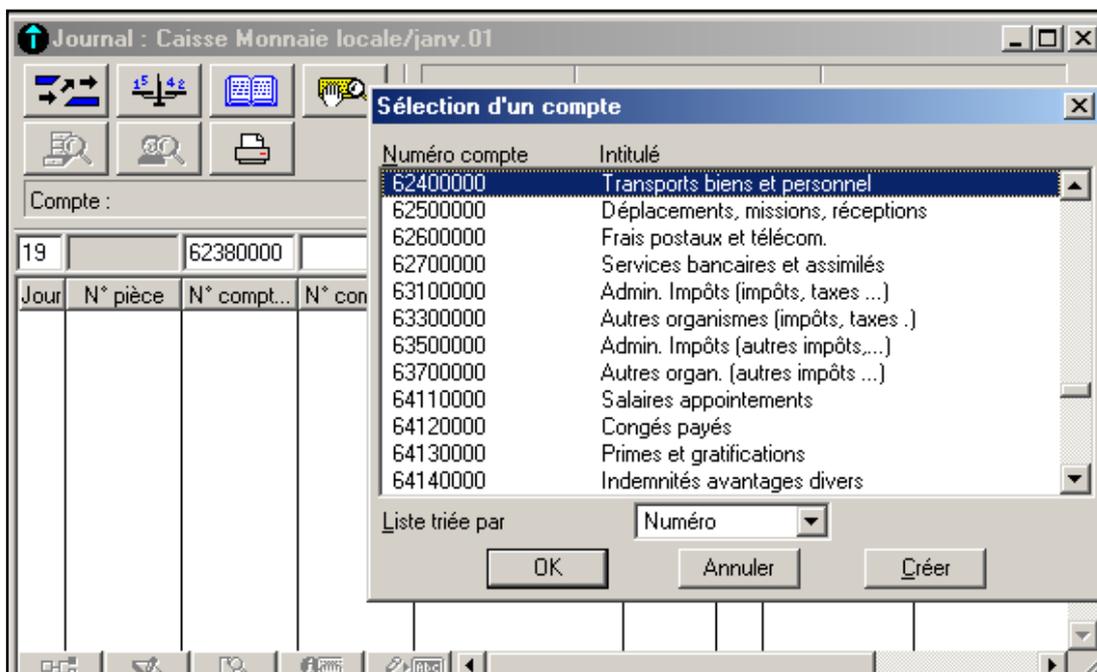


Figure 3 :Exemple d'enregistrement d'un compte sous l'ERP SAGE ligne 100

Ci dessous nous donnons un extrait de la cartographie du schéma source de la comptabilité générale sous SAGE ligne 100 :

Nom de la table	Descriptif
F_COMPTEG	Les comptes de la comptabilité générale
F_COMPTEA	Les comptes de la comptabilité analytique
F_JOURNAUX	Les journaux comptables
F_ECRITUREC	Les écritures comptables
F_TAXE	Les taxes
F_COMPTET	Les partenaires
F_CONTACTT	Les contacts des partenaires

**Tableau 1: Eléments comptables à récupérer depuis SAGE ligne 100**

Ainsi, le tableau ci-dessus décrit les éléments comptables à migrer depuis SAGE vers l'OpenERP pour le cas des données fiduciaires de RIBATIS. Ces données seront manipulées, avec plus de flexibilité, par la suite dans le cadre d'un paramétrage / développement spécifique au niveau de l'environnement cible (OpenERP).

#### 2.1.3.2 OpenERP OLAP :

Ce module fait partie, entre autres, des modules propres au progiciel OpenERP. Il a été conçu pour apporter des solutions BI intégrées au système d'information en question. Avec sa simplicité d'utilisation sous une interface Web et son architecture de stockage multidimensionnelle/relationnelle variée : HOLAP (Hybride OLAP = ROLAP + MOLAP), cette solution décisionnelle open source s'avère adéquate aux besoins des managers en terme de tableaux de bord.

Néanmoins, ce module présente des inconvénients coté pertinence d'analyse (niveaux d'agrégation). En effet, les dirigeants se limitent aux vues construites au préalable sans pouvoir aller à un niveau de détails plus fin sur des axes d'analyse avancés ni de délimiter certains périmètres pour mieux identifier les sources d'éventuelles problématiques.

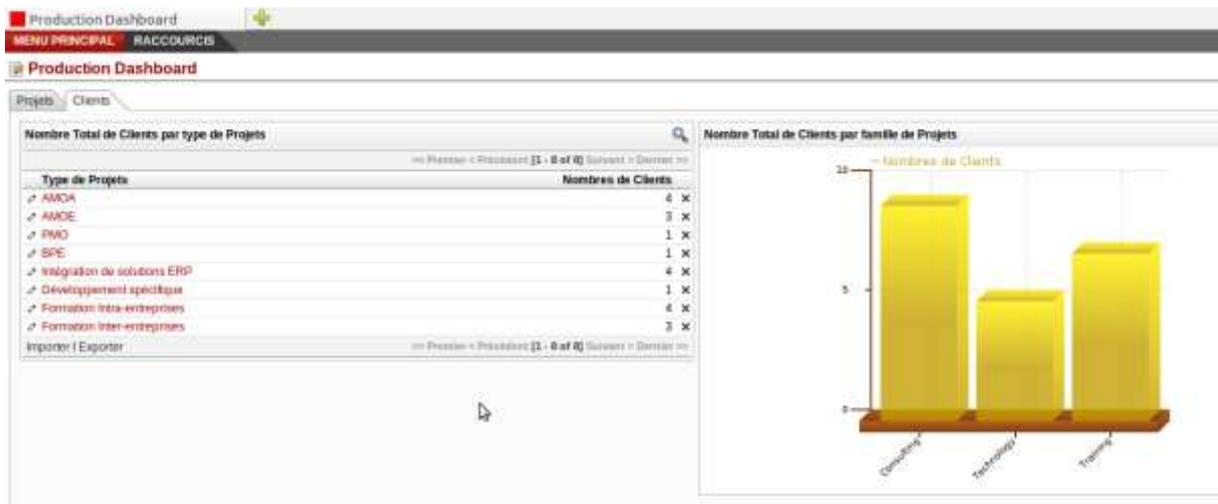


Figure 4: Exemple d'un tableau de bord sous l'interface Web du module Open ERP OLAP

## 2.2 Spécification des besoins métiers :

Le système à réaliser devrait permettre en premier abord, **une meilleure gestion des processus** métier et support via la migration d'un module incontournable dans n'importe quel système d'information intégré ; à savoir le module comptable. L'amélioration de la productivité d'une entreprise donnée passe forcément via le paramétrage de ce module suivant les contraintes métiers de celle-ci. Dans cette perspective, le système de reprise de données, mis en place, doit répondre au besoin récurrent des clients de RIBATIS pour une manipulation plus flexible des données comptables, sous l'environnement de travail cible qui n'est autre que le progiciel : OpenERP.

En second lieu, la solution BI à instaurer devrait **améliorer la prise de décision** pour un Client RIBATIS. Il s'agit donc d'exploiter les mesures principales (KPIs : indicateurs de performance) pour mieux gérer l'activité d'une entreprise donnée. Ainsi, notre travail consiste aussi à créer un Datawarehouse dédié à la mesure de performance interne d'un Client RIBATIS pour pouvoir répondre aux questions complexes (projets en dérapage ? degré d'implication des collaborateurs ? état d'avancement des projets menés ? mesure de l'écart entre le prévu et le réalisé...) en utilisant des tableaux de bord plus dynamiques qui génèrent des connaissances à jour et pertinentes concernant la « Gestion de l'activité » d'une entreprise donnée.

**Conclusion :**

Nous avons vu lors du premier chapitre que notre projet consiste en deux principales briques : d'une part une migration de données vers OpenERP V6 à partir de l'OpenERP V5 ou SAGE ligne 100 , et d'autre part la mise en place d'une solution décisionnelle pour la mesure de l'activité interne d'un client RIBATIS. Ce chapitre a concerné l'étude de l'existant : OpenERP 5 et SAGE ligne 100 en terme de migration et OpenERP OLAP en terme de solution décisionnelle. Ainsi nous avons illustré la faiblesse de l'architecture d'OpenERP V5 en terme d'interdépendance des modules et celle de l'« OpenERP BI » concernant la dynamique des axes d'analyse et justifié les besoins métiers principaux de la solution future : une meilleure gestion des processus métiers moyennant l'OpenERP 6 et une amélioration de la prise de décision via des tableaux de bord dynamiques générés par une solution décisionnelle bien adaptée. Le choix des composants open source de cette solution fera l'objet du chapitre suivant.

## Chapitre III : Etudes comparatives des outils décisionnels

Ce chapitre étoffera la démarche, le modèle ainsi que les résultats des études comparatives menées pour le choix des outils décisionnels à utiliser dans notre projet. Le premier comparatif concernera le choix de l'ETL qui va mettre en place le système de migration, tandis que le deuxième tranchera sur l'outil de restitution de données destiné à l'implémentation de la solution BI.

### 3. Etudes comparatives des outils décisionnels

Nous avons signalé lors du premier chapitre (cf. section 3.1) la nécessité de procéder à des études comparatives pour choisir les outils les plus appropriés aux deux volets de notre projet. Ainsi la problématique de choix concernera : les outils ETL pour la migration de données et les outils de restitution de données pour la partie reporting.

#### 3.1 Choix de l'ETL :

Pour l'intégralité de ses services de Technology / Consulting, RIBATIS se base sur des solutions **Open Source**. Ainsi toutes les briques logicielles utilisées : ERP, GED, CRM etc. sont librement redistribuées aux clients suite à des travaux dérivés (paramétrage et /ou développement spécifique). C'est alors dans ce sens que s'inscrit notre premier benchmark : choisir l'ETL le plus adapté à notre problématique de migration puisant ainsi du marché 'libre licence'.

En effet, un ETL Open Source représente une offre complète et peu chère relativement aux solutions classiques : ETL propriétaire ou en code spécifique. L'Open Source donne accès au code source, le développeur peut donc le modifier pour ajouter des fonctionnalités, le consulter pour regarder le fonctionnement du programme de plus près.. De plus, la communauté Open Source est souvent très active : de nombreux utilisateurs sont disponibles sur les forums pour toute aide.

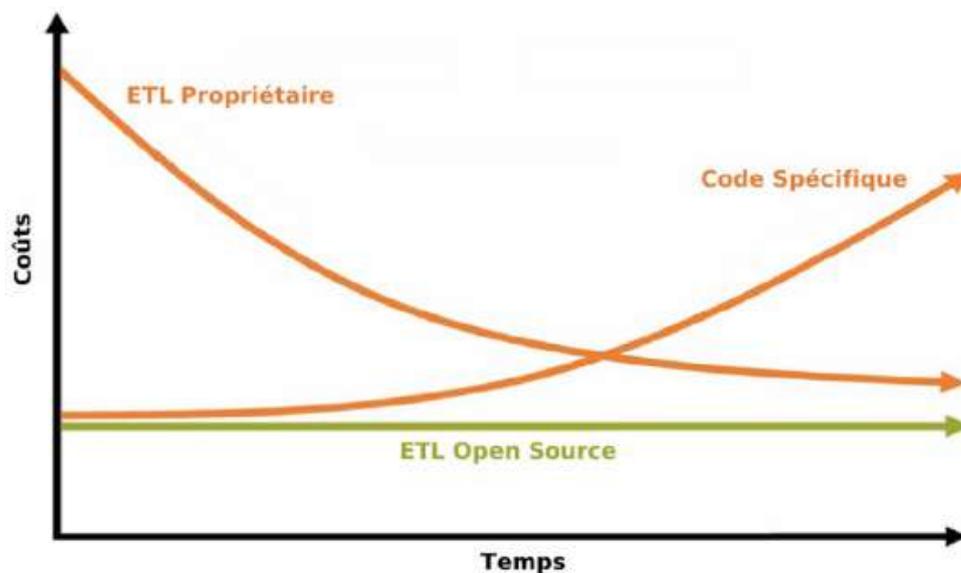


Figure 5 : Diagramme représentant les coûts des solutions en fonction du temps passé

### 3.1.1 Long list ( ETLs Open Source considérés ) :

A cet effet, il a fallu rechercher sur Internet les différents outils ETL Open Source disponibles.. Cette phase de recherche sur de nombreux sites internet (forums, sites officiels, blogs, etc.) a permis de répertorier les **5 ETLs** suivant :

- Pentaho Data Integration (PDI)
- Talend Open Studio (TOS)
- KETL
- CloverETL
- Palo ETL

Cette liste recense les outils ETL les plus utilisés et les plus reconnus dans le monde de l'Open Source. En effet, La figure ci-dessous illustre la tendance d'intérêt que portent, à priori, les internautes vis-à-vis des ETLs freeware en question, cela pendant les 5 dernières années. A noter, que d'autres ETL Open Source existent dans le marché avec un taux d'intérêt moins de 2 %, ce qui explique la raison pour laquelle ils ne font pas partie de cette **présélection**.

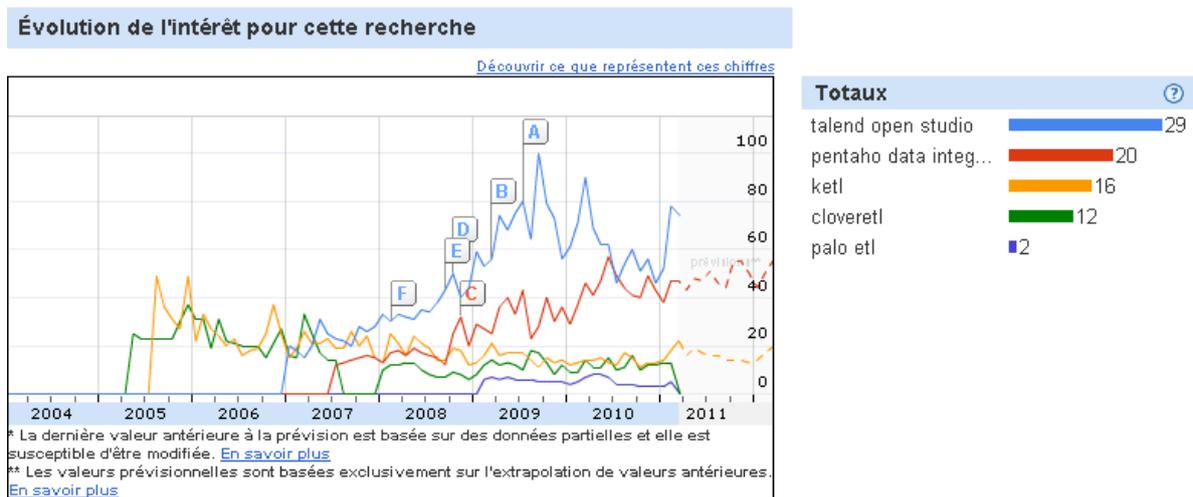


Figure 6 : Tendance d'intérêt sur le Web vis-à-vis des 5 ETLs (long list)

### 3.1.2 Critères de comparaison :

Après avoir procédé à une présélection des alternatives de notre étude comparative, nous avons pu établir les divers critères de choix via lesquels on pourrait réaliser le benchmark. Ces critères sont de deux types :

- *Critères éliminatoires :*

Un outil ETL ne peut s'avérer strictement performant, dans le sens basique du terme que, s'il s'approprie un minimum de caractéristiques technico fonctionnels qu'on requiert, les experts de la BI, pour ce genre d'outils. Ainsi, et dans le cadre de notre étude comparative, un ETL est éliminé d'emblée si et seulement s'il subit les trois points suivants (un ou plusieurs) :

- **Absence de communauté** : le cas échéant, toute aide via un support technique en ligne serait impossible. Ce qui entrave toute utilisation de l'outil en cas de blocage.
- **Absence de connecteurs applicatifs** : dans ce cas, l'ETL ne peut être interfacé avec des briques logiciels genres : CRM, GED, ERP etc. Ce qui restreint le champ d'utilisation de l'outil
- **Non compatibilité avec un nombre minimal de SGBD** : ni en connexion native ni via un pilote ODBC/JDBC.

- *Critères d'évaluation :*

Voici les deux familles de critères qu'on a retenues pour l'évaluation des ETLs candidats. La liste étant non exhaustive, puisque les critères composants choisis de chaque famille relèvent des exigences technico fonctionnelles de RIBATIS (préférences selon les perspectives de l'utilisation, la compatibilité avec les framework existants etc.).

**- Critères fonctionnelles :** *(à coefficients de pondération égaux)*

- ✓ Fonctionnalités : couverture fonctionnelle du logiciel (métadonnées, transformations etc.).
- ✓ Performance : consommation mémoire et du temps d'exécution des tâches.
- ✓ Professionalisme : méthodes employées dans le processus de développement et de l'organisation du projet.
- ✓ Convivialité : qualité de l'interface utilisateur et de l'accessibilité du logiciel.
- ✓ Architecture : modularité, portabilité, flexibilité, extensibilité, ouverture et facilité d'intégration.
- ✓ Qualité : qualité de la conception, du code et des tests.

**- Critères techniques :** *(à coefficients de pondération égaux)*

- ✓ Référentiel vers métadonnées : un historique des Jobs / Fichiers manipulés, et une description détaillée de ceux-ci (structure / format).
- ✓ Outil de création de requêtes : possibilité de pré visualiser le contenu d'une BdD via des requêtes : SELECT, UPDATE etc sous l'ETL.
- ✓ Validité des fichiers plats : possibilité de valider la structure d'un fichier CSV/XML/Excel etc. au préalable, conformément à un schéma prédéfini par l'outil de l'ETL.
- ✓ XML / RPC : possibilité d'envoyer une requête à un serveur XmlRPC et lire les résultats renvoyés en output.

### 3.1.3 Short list (ETLs Open Source retenus) :

On a constaté que deux parmi les cinq ETLs appartenant à la « long list » présentent des manques en termes des caractéristiques basiques citées dans la section précédente (cf. section 1.2) . En l'occurrence, **KETL** et **Palo ETL** sont à éliminer.

En effet, et suite à une exploration de ces deux outils, on a pu découvrir les inconvénients suivant :

Pour « Palo ETL » :

- C'est un outil récent qui n'a pas de communauté.
- Son temps de traitement est très élevé.
- Absence de connecteurs pour les applications d'entreprises.

Pour « KETL » :

- Il n'y a pas d'interface graphique, ce qui rend l'utilisation de l'outil difficile surtout quand il s'agit d'opérations complexes.
- Absence d'un support technique, parce qu'il n'existe pas une communauté qui supporte KETL.
- Il n'est pas compatible avec tous les SGBD.

Ainsi, conformément aux critères éliminatoires établis auparavant, on a décidé d'exclure ces deux solutions ETL de notre benchmark. La liste des alternatives se restreint alors aux trois composants suivant :

- CloverETL
- Talend Open Studio (TOS)
- Pentaho Data Integration (PDI)

### **3.1.4 Comparatif des composants du « short list » suivant les critères d'évaluation :**

Maintenant qu'on a retenu les 3 ETLs candidats du « short list », alors on va les évaluer par rapport à des critères bien précis.

L'étude des 3 ETLs a été réalisée sur un même ordinateur portable, dont la configuration est la suivante :

- Intel Pentium Dual-Core Processor T2310, speedStep Technology, 1.46GHz.
- 2 Go de RAM.
- Windows 7.

Avec:

- TOS version 4.2.0 (dernière version : community).
- Clover.ETL 3.0.1 (dernière version : community).
- PDI version 4.0.1 (dernière version : community).

Codification de l'évaluation (échelle):

++ : l'emporte largement, + : l'emporte, -- : largement dominé, - : dominé

- Selon les critères fonctionnelles : (à coefficients de pondération égaux)

- Fonctionnalités :

TOS a une gamme plus grande de composants (246 composants contre 57 pour CloverETL et 160 pour PDI) et admet donc plus de fonctionnalités. Ceci n'empêche pas PDI de remplir parfaitement son rôle d'ETL et de posséder des composants absents dans la palette de TOS.

	TOS	Clover ETL	PDI
Fonctionnalités	++	-	+
Explication	Plus grande gamme de composants pour TOS		

- Performance :

Clover.ETL et Pentaho Data Integration prennent une longueur d'avance, car TOS est un **grand consommateur de mémoire**, malgré la qualité de ses temps d'exécution sur un nombre de lignes manipulées modéré (jusqu'à 2 millions). Ce handicap lui vaut de ne pas pouvoir lire plus de 3 millions de lignes.

	TOS	Clover ETL	PDI
Consommation de mémoire	--	++	++
Explication	TOS est un grand consommateur de mémoire		

	TOS	Clover ETL	PDI
Temps d'exécution	+	-	-
Explication	TOS est plus rapide que CloverETL et PDI		

- Professionalisme :

TOS et PDI restent **plus organisés** que Clover.ETL dans la modification et l'extension de code. TOS utilise des feuilles de route par exemple, et gère donc mieux le développement de nouvelles fonctionnalités ou versions.

	TOS	Clover ETL	PDI
Professionalisme	++	--	+
Explication	TOS et PDI sont plus organisés que Clover.ETL		

- Convivialité :

TOS a une interface **très agréable** et **très maniable** (sous Eclipse) car beaucoup d'actions se font en Glisser-déposer. Clover.ETL et PDI restent plus faciles à prendre en main car leur interface est moins chargée (moins de composants) et donc plus claire.

	TOS	Clover ETL	PDI
Convivialité	++	+	+
Explication	TOS a une interface plus pratique et plus maniable		

- Architecture :

Le code de Clover.ETL est beaucoup **plus simple** que celui de TOS. Il est donc plus facile à modifier. PDI n'en dispose pas.

	TOS	Clover ETL	PDI
Architecture	+	++	--
Explication	Le code de CloverETL est moins complexe.		

- Qualité :

Les 3 ETLs possèdent des **bugtrackers** (Système de suivie des problèmes), mais il n'y a que TOS qui l'utilise.

	TOS	Clover ETL	PDI
Qualité	++	--	--
Explication	TOS utilise le BugTrackers		

En effet, on a essayé de créer un problème dans un job, en voilà le résultat lors de la compilation au niveau des 3 ETLs :

- **TOS** indique qu'il y a un problème « Impossible de convertir de String en int », en spécifiant la typologie de l'erreur.
- **CloverETL** ne réagit pas même. Le designer doit chercher la source du problème lui-même.
- Idem pour **PDI**, il ne réagit pas en cas d'erreur, donc on devrait chercher la source du problème nous même.

- **Selon les critères techniques** : (à coefficients de pondération égaux)

- Référentiel de métadonnées :

	TOS	Clover ETL	PDI
Référentiel de métadonnées	++	+	-

- En effet, dans l'ETL **TOS**, il existe un référentiel qui permet de stocker les métadonnées (type et format des données d'entrées) afin de pouvoir les exploiter dans différents jobs (il suffit de les mettre à jour). Cela permet d'éviter à chaque fois le processus d'intégration et de dépôt des composants et connecteurs sur le Job en question, en dessinant leurs connexions et relations, et en modifiant leurs propriétés.
- Dans **CloverETL**, il n'y a pas de méthode pour faire une mise à jour des métadonnées, donc s'il y a un changement au niveau de la metadata en doit refaire tout le travail.
- Tandis que dans l'ETL **PDI** cette fonctionnalité (référentiel de metadata ) est restreinte à un simple historique des transformations récemment utilisées.

- Outils de création de requête :

<i>Accès aux données relationnelles</i>	<b>TOS</b>	<b>Clover ETL</b>	<b>PDI</b>
Outil de création de requête	++	+	+

- En effet, l'outil de création de requêtes sur **TOS** facilite la construction des requêtes dans la base de données cible ou source, cela via une
  - ✓ Détection du schéma (éditeur SQL texte + designer graphique).
  - ✓ Détection des jointures entre les tables (relations entre tables).
  - ✓ Pré visualisation des résultats de requêtes SQL exécutées.
- De son coté, **CloverETL** dispose d'un outil de requêtes intégré, similaire à celui de TOS abstraction faite de l'option de détection des jointures entre les tables, ainsi que le designer graphique des requetes SQL.
- Idem pour **PDI**.

- Validité des fichiers plats :

<i>Fichier plats</i>	<b>TOS</b>	<b>Clover ETL</b>	<b>PDI</b>
Validité des fichiers plats	++	--	--

- En effet, **TOS** dispose d'une option qui permet de vérifier la validité des fichiers plats (.csv / .xls / .xml etc) relativement à leur structure de base (schéma prédéfini).
- Alors que les 2 autres ETLs n'en disposent pas.

- Déclenchement par message :

<i>Déclenchement par message</i>	<b>TOS</b>	<b>Clover ETL</b>	<b>PDI</b>
XML RPC	+	-	--

- En effet, **TOS** peut envoyer une requête à un serveur XmlRPC et lire ensuite les résultats renvoyés en output.
- Sous CloverETL le protocole XML-RPC est absent. Mais on note l'existence du protocole http (déclenchement par message également).

- Tandis que PDI n'en dispose même pas.

### 3.1.5 Bilan de l'étude comparative des ETLs :

Compte tenu des résultats de comparaison des trois outils ETLs Open Source : Talend Open Studio, Pentaho Data Integration et CloverETL (composants du short-list) ; et en prenant en considération l'échelle d'appréciation fixée au préalable et appliquée sur la totalité des critères d'évaluation, alors on conclut que : **TOS est l'ETL le plus puissant** parmi les 3 ETLs étudiés pertinemment et le plus adapté également au contexte de notre projet.

Dans cette perspective, et suite aux réunions tenues avec la direction de RIBATIS ; alors la mise en oeuvre de notre projet (migration et chaîne BI) aura lieu moyennant cet outil.

## 3.2 Choix de l'outil de restitution de données :

### 3.2.1 Contraintes de l'étude comparative :

Lors du benchmark technico fonctionnel précédent, l'objectif était d'opter pour une solution ETL open source adaptée à notre problématique. En effet, L'ETL choisi, à savoir **Talend Open Studio**, sera utilisé aussi bien dans la phase de la double migration :

- montée en version : Open ERP 5 → Open ERP 6
- mapping: SAGE ligne 100 → Open ERP 6 (module comptabilité);

que dans la seconde partie du projet : élaboration des tableaux de bord dédiés à la gestion de performance interne d'un client RIBATIS.

Pour cette deuxième perspective, une suite décisionnelle (client/serveur OLAP, report designer, concepteur de tableau de bord etc.) devrait être choisie sous les **contraintes techniques et métiers** suivantes :

- Le futur outil devrait combler les lacunes, étudiées auparavant, de l'outil existant (Open ERP OLAP : module décisionnel).
- Il s'agirait forcément d'un outil Open Source, pour rester fidèle à la stratégie de RIBATIS (investissement à base d'outils free).

- Compatibilité (en terme de connectivité) de la suite BI de restitution de données en question, avec l'ETL choisi auparavant à savoir Talend Open Studio.
- Simplicité de la conception des tableaux de bord pour les utilisateurs finaux, qui ne seront forcément pas des informaticiens spécialisés en la matière.

En dernier lieu, il faut noter que la base de cette étude est strictement une liste composée de deux alternatives, initialement imposées par la direction de RIBATIS, répondant bien évidemment aux contraintes précédentes. En effet, il s'agit bien d'une exigence du Client pour lequel le cabinet effectue le test en environnement pilote. Ainsi, on a commencé notre comparatif avec un « **short list** » **préétabli**.

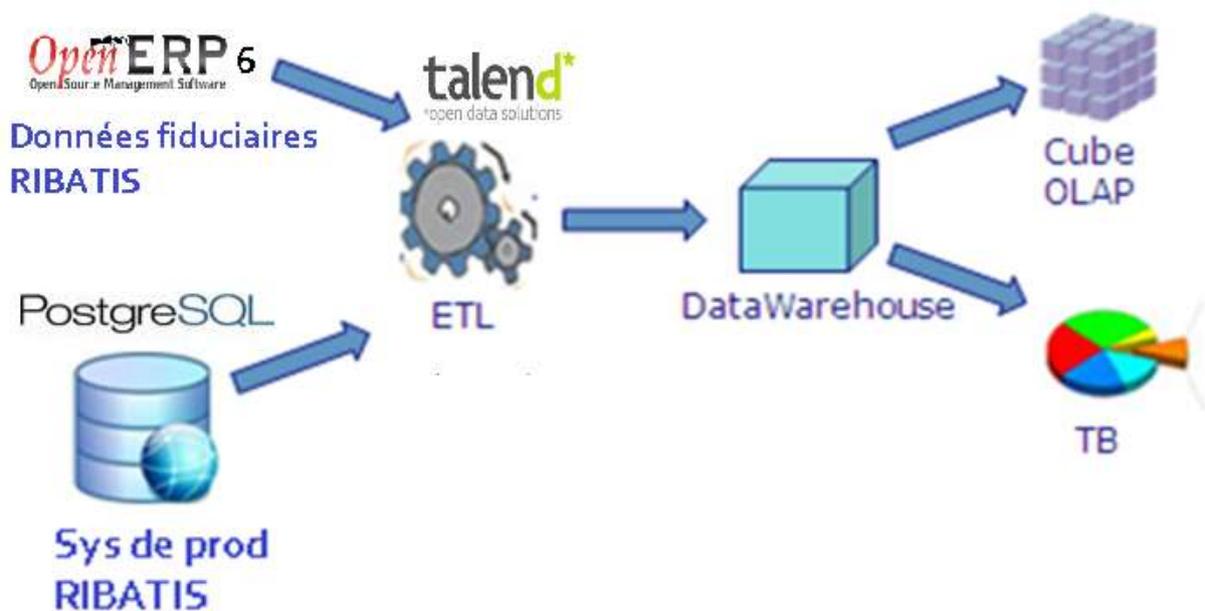


Figure 7 : Schéma récapitulatif de la suite décisionnelle du projet

### 3.2.2 Présentation du « short list » préétabli :

Les deux outils Open source destinés à la restitution de données et qui ont été imposés par la direction sont : **Palo BI suite** et **JasperSoft BI suite**.

➤ **Palo BI suite :**



Palo Suite combine les quatre **applications principales** de la société allemande Jedox – Serveur OLAP, Palo Web, Serveur ETL et Palo pour Excel – en une seule et personnalisable plateforme de Business Intelligence. La plateforme est complètement basée sur des produits Open Source représentant une solution de Business Intelligence t entièrement libre de tout coût de licences.

- Le **serveur OLAP** de Palo, module clés de Palo Suite, offre une stabilité et des performances accrues, ainsi que de nouveaux algorithmes logiques. C'est une application de base de données multiutilisateurs qui permet à tout collaborateur à travers une entreprise d'accéder, de modifier et de collaborer sur des données de BI et ceci de manière instantanée. De plus, il offre une agrégation en temps-réel à travers un modèle de données multidimensionnel.
- **Palo Web** combine tous les composants Palo dans une interface Web. Ici toutes les actions peuvent être effectuées, suivant les droits utilisateurs qui ont été déclarés et permettant l'accès à Palo Web. Les Designer vont pouvoir administrer et créer des rapports Web, modéliser la base de données OLAP et suivre les processus ETL. Les utilisateurs métiers vont pouvoir accéder aux applications de planification et aux rapports d'analyse dans Palo Web.
- Avec **Palo pour Excel**, Palo Suite permet aux utilisateurs un accès direct à travers Microsoft Excel et Open Office Calc. Avec cela, les applications existantes (principalement basés sur Excel) peuvent être facilement migrées et trouver de nouvelles utilisations. Découvrir Palo Suite à travers Palo pour Excel est simple, et se fait sans aucunes expériences Palo préalable. Palo for excel est le designer même des rapports et des tableaux de bord.

JasperSoft BI suite :  JasperServer

La plateforme décisionnelle Jaspersoft offre les services centralisés requis pour gérer l'ensemble de la suite décisionnelle, et ainsi traiter, organiser, sécuriser et accéder aux rapports, tableaux de bord et analyses.

Jaspersoft est une plateforme décisionnelle extrêmement perfectionnée, conçue pour évoluer et pour répondre aux exigences les plus pointues d'une grande variété d'organisations et de modèles de déploiement.

Une plateforme décisionnelle moderne doit exécuter des fonctions diverses capables de répondre aux besoins variés d'une vaste communauté d'utilisateurs. Elle doit sécuriser et fournir les données depuis la source jusqu'à l'entrepôt, puis jusqu'à l'utilisateur sous la forme d'un rapport ou d'une analyse. Des métadonnées sont souvent utilisées pour simplifier la complexité des données auprès des concepteurs et utilisateurs de solutions décisionnelles en libre-service. Enfin, elle doit être capable de gérer le déploiement et l'intégration dans différents environnements et modèles de production. Voici les **trois principales applications** de JasperSoft :

- **JasperReports Server** : L'analyse en mémoire permet une exploration simple et rapide des données
- **Jaspersoft OLAP** : Exécution de requêtes analytiques avancées dans de vastes ensembles de données avec une interface utilisateur Web interactive
- **Jaspersoft iReport Designer** : Puissant environnement de bureau destiné à la conception de rapports complexes ou non pixellisés avec JasperReports et JasperReports Server.

### 3.2.3 Conformité des alternatives par rapport aux contraintes :

Les deux outils décisionnels présentés ci-dessus, s'avèrent tous les deux conformes aux contraintes métiers et techniques du benchmark. En effet :

- Les deux solutions BI sont entièrement libres de tout cout de licence.
- Les deux suites BI peuvent être connectées à l'ETL Talend Open Studio :
  - ✓ **Palo** : via le composant « tPaloConnection » de TOS.
  - ✓ **Jasper** : via le composant « tJasperOutput » de TOS.
- Les deux solutions présentent un « plus » relativement à Open ERP OLAP :
  - ✓ **Palo** : offre une grande **flexibilité** concernant les axes d'analyse. Ainsi les tableaux croisés sont manipulables via la version Web et sous Palo For Excel. Cela, outre la puissance de cet outil coté diversification de **l'ergonomie** des tableaux de bord en fonction des problématiques d'analyse.
  - ✓ **Jasper** : offre un report designer **autonome** de la suite BI. S'ajoute à cela la grande **interactivité** des graphiques de mesure de performance qu'il déploie.

En particulier, **Palo BI suite se distingue** par la simplicité d'utilisation de son Report designer à savoir Palo For Excel. Cet outil étant facile à découvrir sans la moindre connaissance préalable de Palo BI, il constitue ainsi une solution de proximité vis-à-vis des utilisateurs finaux déjà familiers avec le fameux outil Excel. Il leur offre par ailleurs, des applications avancées pour le design de leur états de sortie (tableaux de bord).

### 3.2.4 Bilan du comparatif des outils de restitution de données :

Suite à la brève étude établie ci-dessus, et conformément aux décisions de l'équipe de projet de RIBATIS, dépendants ainsi des préférences du Client ; on a décidé de travailler avec **Palo BI suite** dans la suite du projet.

### **Conclusion :**

Ce chapitre a présenté en détails le modèle et les résultats des benchmark menés pour le choix des outils décisionnels open sources les plus adaptés à notre projet. Ainsi pour ce qui est du choix de l'ETL on est parti d'un « long list » composé de cinq ETLs candidats avant d'en retenir que trois (short list) conformément à des critères éliminatoires. Ensuite on a comparé les ETLs retenus suivant des critères d'évaluation (technico fonctionnels) pour trancher finalement sur le plus puissant et le plus approprié parmi eux (**Talend Open Studio**). Et pour ce qui est du choix des outils de restitution de données, on a commencé avec un « short list » préétabli (deux outils candidats) et on a par la suite évalué ses composants relativement à des contraintes techniques et métiers. Ce comparatif a mené au choix de **Palo BI Suite** comme outil de reporting.

Le chapitre suivant fera l'objet de l'étude conceptuelle détaillée concernant les deux volets de notre projet à savoir la migration de données et la solution décisionnelle.

## Chapitre VI : Conception de la Solution

Dans ce chapitre on va entamer la partie la plus critique de notre projet, c'est la conception.

Dans cette partie on va utiliser le concept du reverse-Engineering afin de pouvoir connaître la structure des différentes architectures soit Sage Ligne 100 et OpenERP.

## 4. Conception de la Solution :

### 4.1 Conception de la montée en version (OpenERP v5 vers OpenERP v6) :

Cette rubrique décrit la conception de la moulinette de migration qui réalise la montée en version de l'OpenERP de la version 5 vers la nouvelle version v6, à l'aide du concept du « Reverse Engineering ». Cette procédure concerne le système de production interne de RIBATIS intégré sous le SGBD PostgreSQL.

A cet effet, on a utilisé l'outil du Reverse Engineering à savoir : « PowerAMC v15.01 » qui nous a généré le model conceptuel de données (MCD) à partir de la BD en question, liée à l'OpenERP 5 (environnement source de la migration).

#### 4.1.1 Reverse engineering de l'OpenERP 5 :

Il s'agit d'une rétro-conception qui consiste à étudier les tables de l'OpenERP v5 afin de déterminer le fonctionnement interne de celui-ci, et de savoir les liaisons entre les tables, la signification des champs ainsi que les champs obligatoires.

La procédure commence tout d'abord par l'établissement d'une connexion entre le SGBD PostgreSQL et PowerAMC via un pilote ODBC de PostgreSQL.

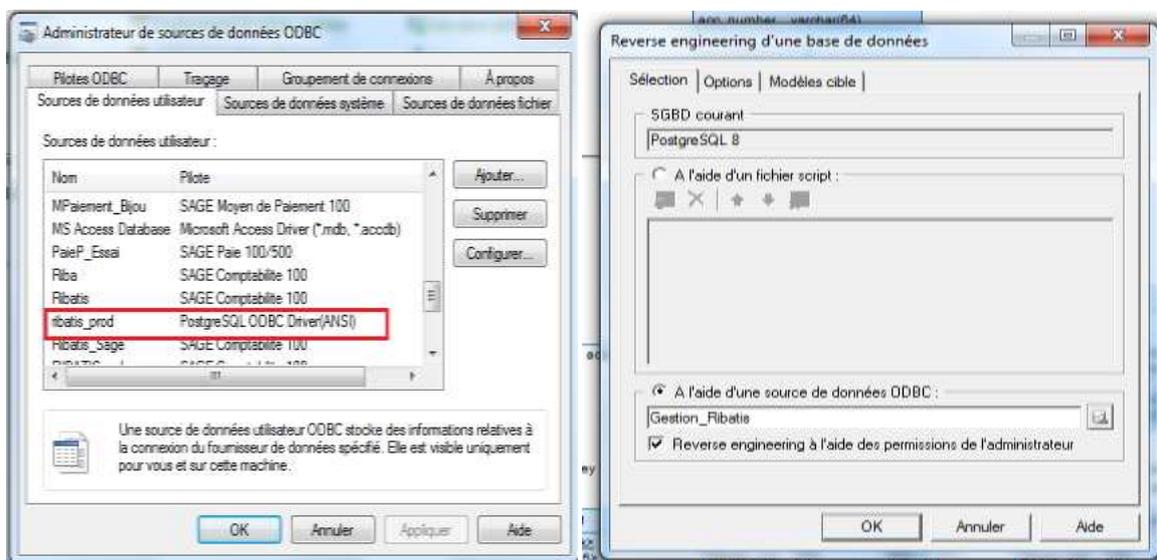


Figure 8 : Connexion entre PostgreSQL et PowerAMC

Ensuite on génère le Reverse Engineering de la base de données de l'OpenERP v5 en choisissant les tables qu'on cherche à étudier.

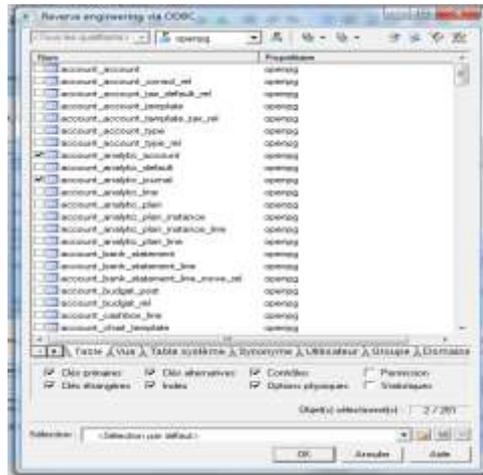


Figure 9 : Sélection des tables concernées par le Reverse Engineering

Suite à cette opération, on obtient le MCD de chaque module structuré suivant son schéma de base.

**Module : « Comptabilité et Finance » :**

Le module Comptabilité et Finance contient les tables suivantes :

Nom de la Table	Description
Account_account	Les comptes de la comptabilité générale
Account_account_type	Types des comptes
Account_analytic_account	Les comptes de la comptabilité analytique
Account_analytic_journal	Journal analytique
Account_analytic_line	Les lignes de la comptabilité analytique
Account_fiscalyear	Année fiscale
Account_invoice	Factures
Account_invoice_line	Lignes de facture
Account_invoice_line_tax	Les taxes des lignes de facture
Account_invoice_tax	Les taxes facturées
Account_budget_poste	Budget
Account_journal	Les journaux comptables
Account_journal_period	Périodes des journaux
Account_journal_view	La vue des journaux
Account_move	Les écritures comptables
Account_move_line	Les lignes d'écritures
account_payment_term	Types de paiement
Account_period	Périodes
Account_tax	Les taxes
Account_tax_code	Codes de taxes
Res_partner	Les partenaires
Res_partner_address	Les contacts des partenaires

Tableau 2: Tables du module comptabilité et finance de l'OpenERP 5



Dans le module ressources humaines, la table **Hr\_employee** (contenant les informations sur les employés) est la table centrale du module, donc elle est liée aux autres tables par des clés étrangères.

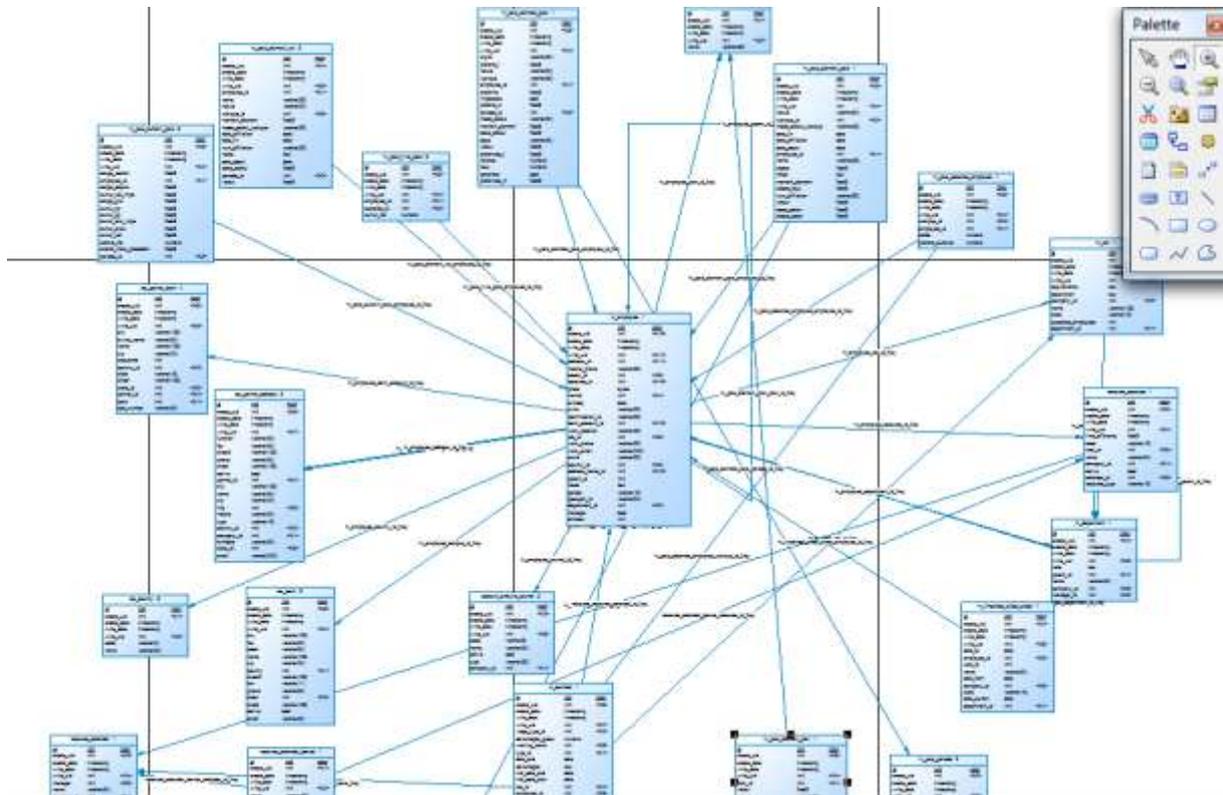


Figure 11 : MCD du module ressources humaines (résultat du Reverse Engineering)

➤ **Module Paie : (Développé par RIBATIS)**

Le module Paie contient les tables suivantes :

Nom de la Table	Description
Hr_paie_absence_employee	Absences des employés
Hr_paie_bareme_anciennete	Barèmes d'ancienneté
Hr_paie_bareme_ir	Barèmes de l'IR (Impôt sur revenu)
Hr_paie_bareme_ir_annuel	Barèmes de l'IR annuel
Hr_paie_bulletin_paie	Bulletin de paie
Hr_paie_contrat_type	Types de contrat
Hr_paie_controle_paie	Contrôle de paie
Hr_paie_element_paie	Éléments de paie
Hr_paie_exercice	Exercices
Hr_paie_livre_paie	Livres de paie
Hr_paie_organisme	Paiement des organismes (CNSS, CIMR,...)
Hr_paie_param_societe	Paramètre de la société

Hr_paie_periode	Période de paie
-----------------	-----------------

Tableau 4: Tables du module de paie de l'OpenERP 5

Dans le module Paie, on trouve que des tables comme : la gestion d'absence des employés, le barème d'ancienneté, le barème de l'IR etc. sont rattachées au module ressource humaines par la table hr\_employee.

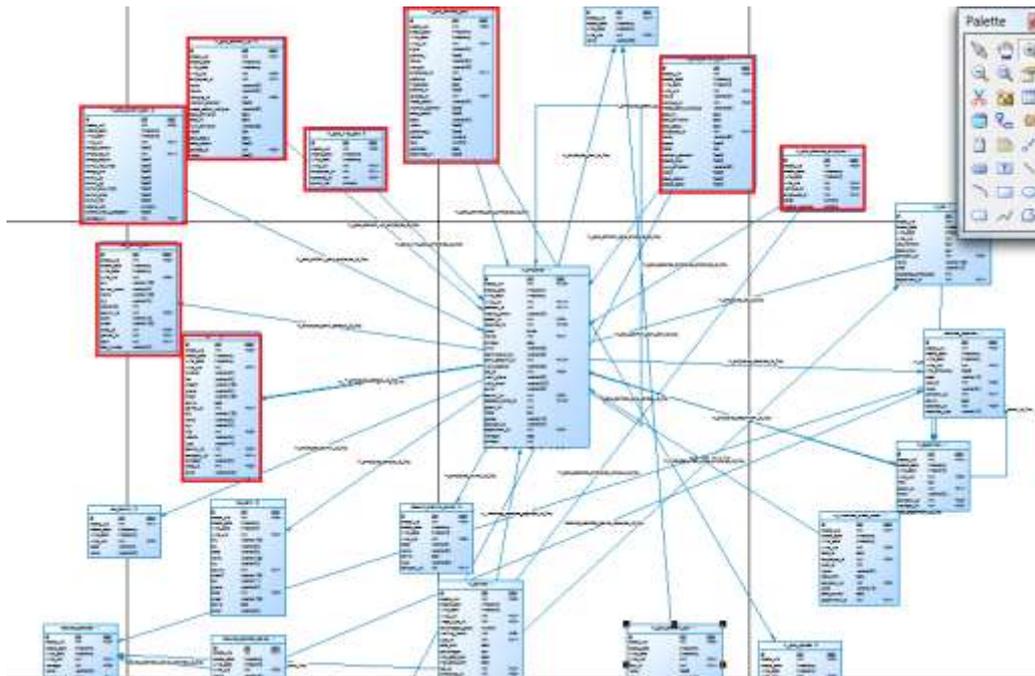


Figure 12 : Localisation des tables de la paie des employés dans le MCD du module RH

Tandis que les tables gérant la paie indépendamment des employés sont regroupées à part (elles ne dépendent pas du MCD du module RH) : Contrôle de paie, livres de paie, paiement des organismes (CNSS, CIMR,..) etc.

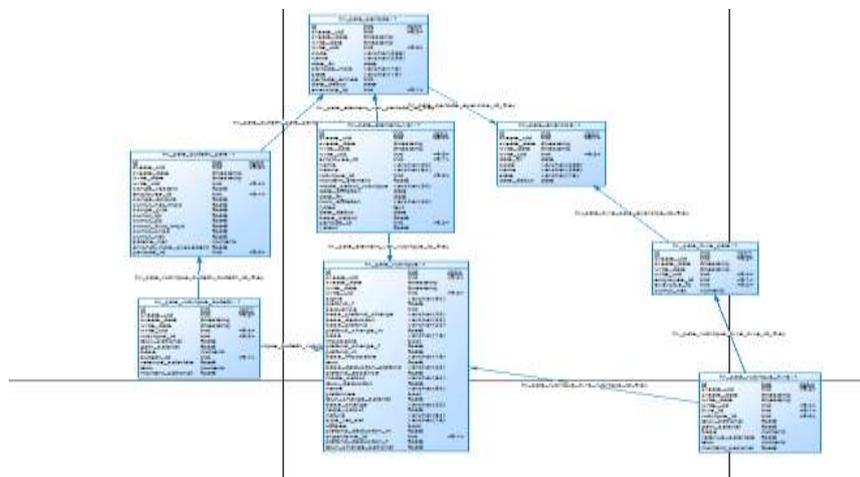


Figure 13 : Regroupement des tables de la paie (indépendantes des employés)

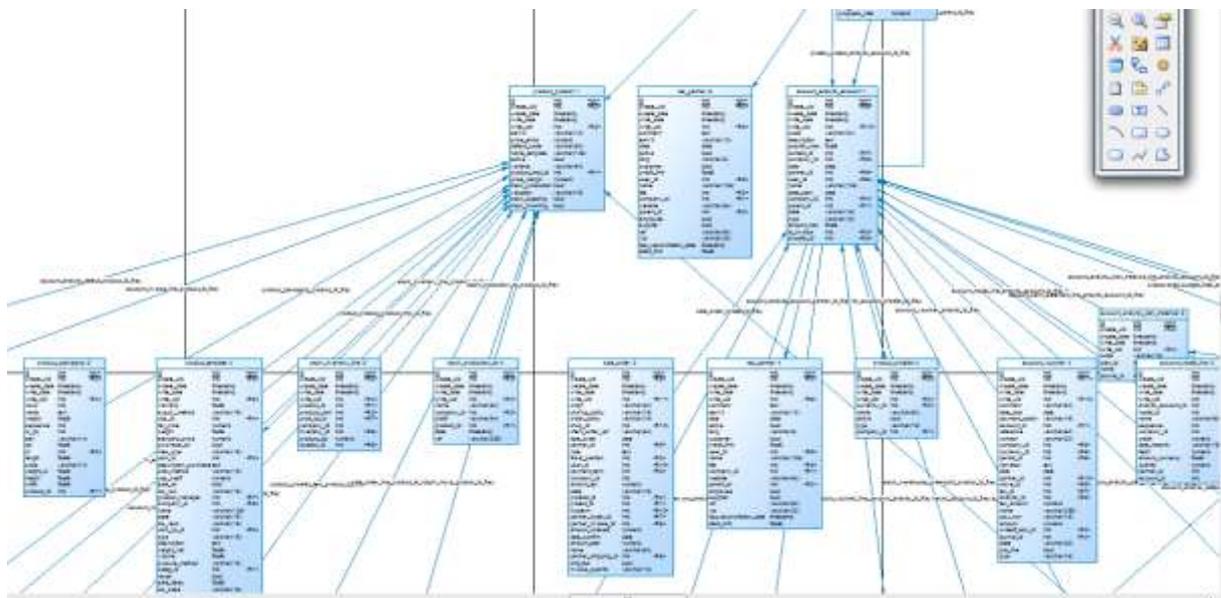
➤ **Module Produit:**

Le module Produit contient les tables suivantes :

Nom de la Table	Description
Product_category	Catégories des produits
Product_price_type	Type de prix (prix du public,..)
Product_pricelist	Listes des prix
Product_pricelist_items	Elément des listes des prix
Product_pricelist_type	Types des listes des prix
Product_product	Les produits
Product_suppliertaxe_rel	Les taxes des fournisseurs
Product_supplierinfo	Informations sur les fournisseurs
Product_taxes_rel	Les taxes sur les produits
Product_template	Modèles des produits

**Tableau 5: Tables du module produit de l'OpenERP 5**

Dans le module Produit, la table **Product\_product** (contenant les informations sur les produits) est la table centrale du module, donc elle est liée aux autres tables par des clés étrangères et elle est aussi attachée au module Partenaire par une clé étrangère.



**Figure 14 : MCD du module produit (résultat du Reverse Engineering)**

➤ **Module Production de RIBATIS : (Développé par RIBATIS)**

Le module Production de Ribatis contient les tables suivantes :



Res_partner_address	Adresse des partenaires
Res_partner_bank	Les banques
Res_partner_bank_type	Types des banques
Res_partner_canal	Moyen de communication avec le partenaire
Res_partner_category	Catégorie des partenaires (Fédération,SII,..)
Res_partner_event	Evenements
Res_partner_title	Titre du partenaire (Sarl,Mr,Administration,..)

Tableau 7:Tables du module partenaires de l'OpenERP 5

Dans le module Partenaires, la table **Res\_partner** (contenant les informations sur les partenaires) est la table centrale du module, donc elle est liée aux autres tables par des clés étrangères.

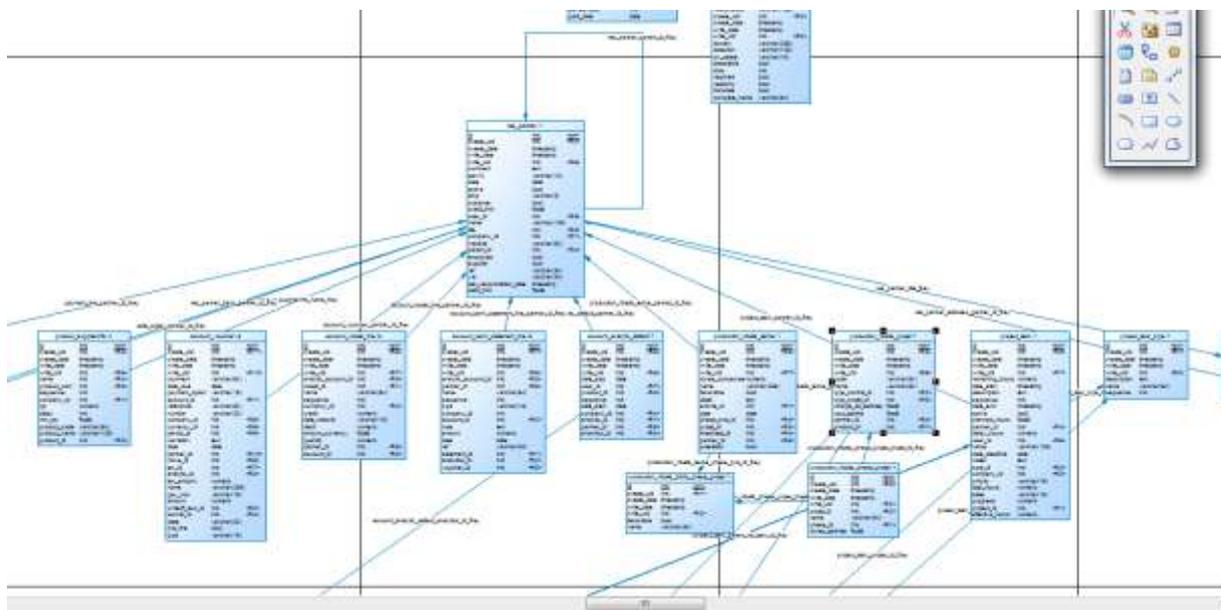


Figure 16 : MCD du module partenaires (résultat du Reverse Engineering)

#### 4.1.2 Reverse engineering de l'OpenERP 6 :

Après avoir appliqué le Reverse Engineering aux tables de la base de données liée à l'OpenERP v6, on a constaté que le schéma MCD de l'OpenERP v6 ressemble à celui de l'OpenERP v5, hormis quelques changements au niveau de la structure des tables d'OpenERP sous sa nouvelle version.

On peut citer quelques exemples de ces changements :

➤ **Multi-sociétés :**

La gestion des multi-sociétés est maintenant complètement intégrée dans la version 6 de l'OpenERP permettant ainsi de cloisonner entièrement ou partiellement les sociétés tout en ayant une vue consolidée immédiate. Un utilisateur appartenant à plusieurs sociétés pourra même choisir sur quelle société il agit sans changer d'identification.

Donc pour introduire le concept de multi-sociétés, la communauté de l'OpenERP a ajouté dans la majorité des tables de la base de données un nouveau champ : **Company\_id**, qui indique l'entreprise concernée par l'enregistrement. Ainsi on peut trouver par exemple les écritures comptables de plusieurs entreprises dans la même table.

The screenshot shows a PostgreSQL query result window titled 'Édition des données - Testfédats (localhost:5432) - restructuration - account\_move'. The table has the following columns: id [PK] serial, create\_uid integer, create\_date timestamp, write\_date timestamp, write\_uid integer, ref character var, name character var, journal\_id integer, company\_id integer (highlighted in red), state character var, period\_id integer, narration text, date date, partner\_id integer, and to\_check boolean. The table contains 33 rows of data, with the 'company\_id' column consistently containing the value '1'.

Figure 17 : Le champ Company\_id ajouté aux tables de l'Open ERP

➤ **Fonction du partenaire :**

En examinant de plus près la BD PostgreSQL liée à l'OpenERP v6, on peut constater que la table qui décrit la fonction du partenaire : « res\_partner\_function » et celle qui décrit le niveau de satisfaction du partenaire : « res\_partner\_som » sont supprimées dans la nouvelle version. En alternative, la communauté de l'OpenERP a intégré ces deux informations dans la table qui décrit l'adresse du partenaire à savoir : « res\_partner\_address ».

➤ **Concept du « level » :**

La nouvelle version de l'OpenERP a introduit un nouveau concept celui du « Level », ce concept est ajouté concrètement comme un nouveau champ dans la table

des comptes généraux account\_account. Ainsi, il s'agit de compter le nombre hiérarchique du parent, si par exemple un compte n'a pas de parents, alors son level est égal à 0 et s'il a un parent et il n'a pas de grand-père donc son level est égal à 1. En général le level d'un compte est le level du père de ce compte incrémenté par 1.

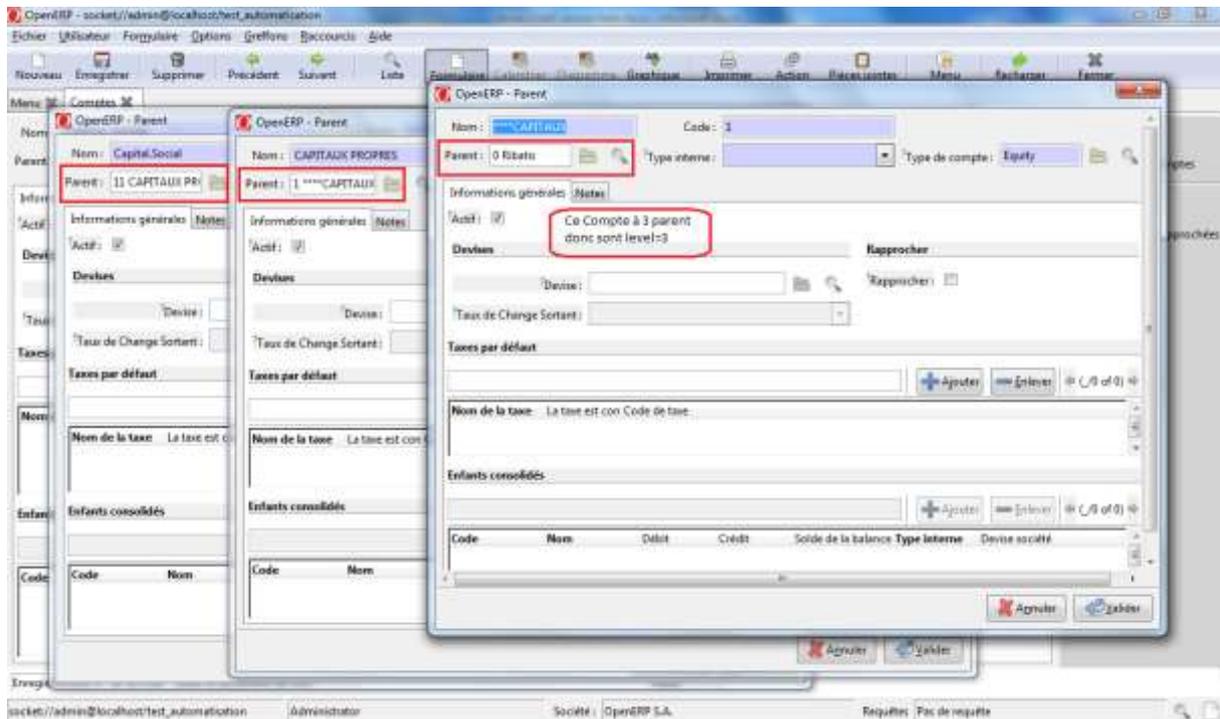


Figure 18 : Le concept du level d'un compte implémenté sous l'OpenERP 6

➤ **Nouvelle table Ressource ressource :**

Il s'agit d'une nouvelle table ajoutée à la nouvelle version d'OpenERP, cette table décrit les ressources existantes dans toutes les entreprises qui sont gérées par OpenERP. Elle a une relation avec le module des Ressources humaines.

**4.1.3 Mapping entre OpenERP v5 et OpenERP v6 :**

Cette phase nous a permis de faire un « Mapping direct » entre les tables de l'OpenERP v5 et celles de l'OpenERP v6, à l'aide des MCD générés grâce au Reverse Engineering déjà effectué.

Ce mapping va subir un enchaînement strict puisque les tables de correspondance sont reliées sémantiquement entre elles.

L'enchaînement de la migration de l'OpenERP v5 vers l'OpenERP v6 suit une correspondance bien précis puisque les tables sont lié entre eux.

## 4.2 Conception de la migration Sage ligne 100 vers OpenERP6 :

### 4.2.1 Cartographie des tables Sage Ligne 100 (Module Comptable)

Après avoir établi la montée en version du système de production interne de RIBATIS depuis l'ancienne version de l'OpenERP (v5) vers la nouvelle version (v6). Il a été question d'instaurer la migration du module comptabilité de l'ERP Sage ligne 100 vers le nouvel environnement cible, à savoir l'OpenERP6.

A cet effet, signalons que « Sage ligne 100 Comptabilité » est un progiciel destiné à la réalisation de toutes les opérations comptables :

- Création d'un plan comptable.
- Gestion de la comptabilité générale.
- Gestion de la comptabilité tierce.
- Gestion de la comptabilité analytique.
- Gestion du reporting.
- Gestion budgétaire.
- Gestion des devises.
- Règlements des tiers, relances clients et relevés tiers.
- Rapprochement bancaire.
- Transfert de données vers d'autres logiciels et communication client/expert comptable.
- Recherche d'écritures.
- Gestion d'écritures d'abonnement.
- Edition des états comptables.
- Déclaration de TVA.

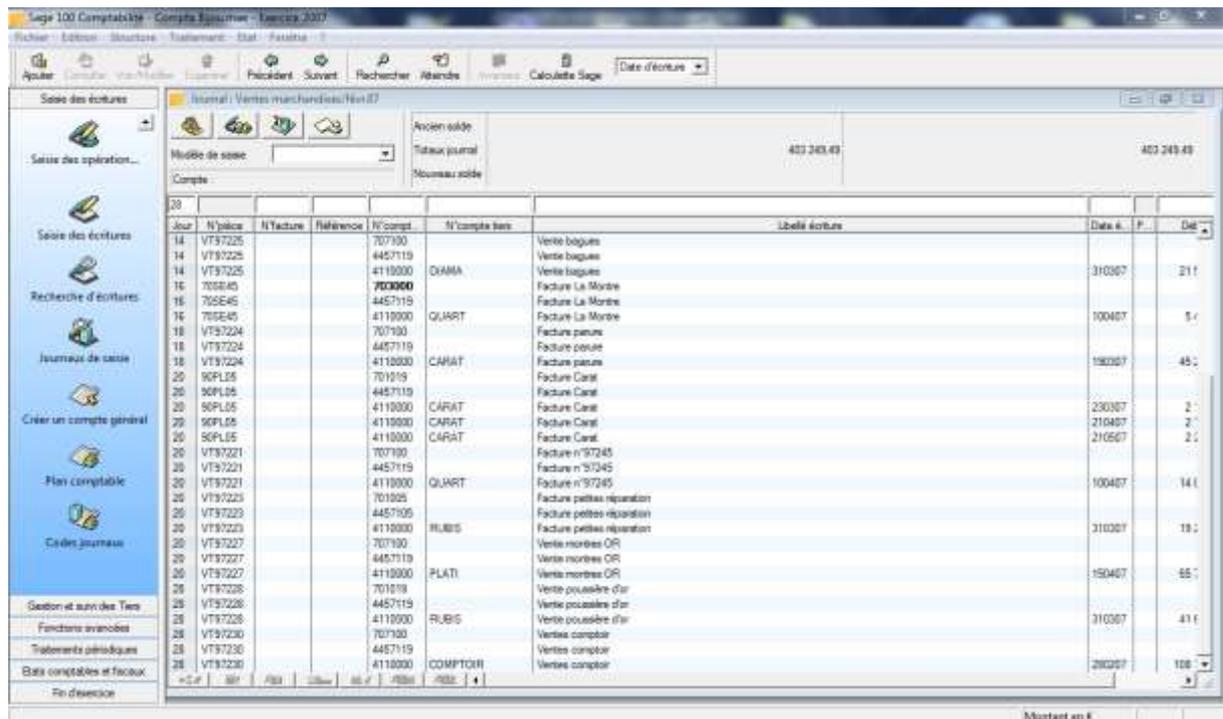


Figure 19 : Environnement de l'ERP Sage ligne 100 Comptabilité

Cette édition appartient à la famille des programmes de l'ERP Sage tout comme celles de : la **Gestion commerciale**, la **Paie**, les **Immobilisations**, les **Moyens de paiements** et le logiciel de **communication bancaire Telbac**.

Sage Ligne 100 est un progiciel propriétaire où le code source et la base de données interne constituent une boîte noire ; c'est-à-dire on ne peut pas accéder ni au code source ni à la base de Sage ligne 100. Alors il n'est pas possible d'explorer le schéma source des tables ni pouvoir détecter les liens entre celles-ci via le concept du Reverse Engineering.

Suite à ce problème on a cherché la documentation officielle de Sage Ligne 100 sur le Web. En effet, on a trouvé un document PDF qui s'intitule « Cartographie Sage Ligne 100 v14 » qui contient une description détaillée des tables de Sage et la signification des champs de ces tables.

Ainsi, la migration des données de comptabilité générale repose sur la reprise des tables sources suivantes sur l'environnement cible de l'OpenERP6 :

- P\_Devis.
- F\_Comptet.
- F\_Contactt.
- F\_Compteg.

- F\_Taxe.
- F\_Banque.
- F\_Journaux.
- F\_Ecriturec.

Ci-dessous la description détaillée de chaque table :

✓ **P Devise** :

Cette table contient des informations sur les devis enregistrés sous Sage.

Nom	Signification	Type de données	Domaine de validité
D_CodeDEI	Code DEI	Chaîne de caractères	3 caractères max.
D_Intitule	Intitulé	Chaîne de caractères	35 caractères max.
D_Format	Format	Chaîne de caractères	31 caractères max.
D_Cours	Cours	Numérique : Réel double	
D_CoursP	Cours période	Numérique : Réel double	
D_Monnaie	Unité monétaire	Chaîne de caractères	21 caractères max.
D_SousMonnaie	Sous-unité monétaire	Chaîne de caractères	21 caractères max.
D_CodeISO	Code ISO	Chaîne de caractères	3 caractères max.
D_Sigle	Sigle	Chaîne de caractères	5 caractères max.
D_Mode	Mode de cotation	Numérique : Entier	Cotation Ancienne cotation
N_DeviseCot	Devise de cotation	Numérique : Entier	1 à 32 (nbre de devises)
D_CoursClot	Cours de clôture	Numérique : Réel double	
D_AncDate	Date limite	Date	
D_AncCours	Cours ancienne cotation	Numérique : Réel double	
D_AncMode	Mode ancienne cotation	Numérique : Entier	
N_DeviseAncCot	Ancienne devise de cotation	Numérique : Entier	1 à 32 (nbre de devises)
D_CodeRemise	Code de remise	Numérique : Entier	
D_Euro	Monnaie zone Euro	Numérique : Entier	0 ou 1
CbIndice	Indice	Numérique : Entier	1 à 32
CbMarq	Marqueur	Numérique : Entier	Uniquement version SQL Champ identité Début compteur : 1

Tableau 8: Cartographie de la table P\_Devise sous Sage ligne 100 comptabilité

✓ **F Contactt** :

Cette table contient des informations sur les contacts des comptes tiers.

Nom	Signification	Type de données	Domaine de validité
CT_Num	Numéro Compte tiers	Chaîne Alphanumérique Maj.	17 caractères max.
CT_Nom	Nom	Chaîne de caractères	35 caractères max.
CT_Prenom	Prénom	Chaîne de caractères	35 caractères max.
N_Service	Service	Numérique : Entier	1 à 30
CT_Fonction	Fonction	Chaîne de caractères	35 caractères max.
CT_Telephone	Téléphone	Chaîne de caractères	21 caractères max.
CT_TelPortable	Téléphone Portable	Chaîne de caractères	21 caractères max.
CT_Telecopie	Télécopie	Chaîne de caractères	21 caractères max.
CT_Email	Adresse EMail	Chaîne de caractères	69 caractères max.
CbMarq	Marqueur de Table	Numérique : Entier	Uniquement version SQL Champ identité
			Début compteur : 1 Incrément : 1
CbProt	Champ de protection	Numérique : Entier	Uniquement version SQL Valeur par défaut : 0
CBCreateur	Créateur de l'application	Chaîne de caractères	4 caractères max.
CBModification	Date de modification de l'enregistrement	Date	
CBReplication	Etat de la réplication (uniquement utilisé par E_Commerce)	Numérique : Entier long	

**Tableau 9: Cartographie de la table F\_Contactt sous Sage ligne 100 comptabilité**

Champs à renseigner obligatoirement lors de l'ajout :

- CT\_Num
- CT\_Nom
- N\_Service

✓ **F Compteg :**

Cette table contient des informations sur les comptes généraux et le plan comptable.

Nom	Signification	Type de données	Domaine de validité
CG_Num	Numéro compte général	Chaîne Alphanumérique Maj.	13 caractères max.
CG_Type	Type de compte	Numérique : Entier	0 = Type Détail 1 = Type Total
CG_Intitule	Intitulé du compte	Chaîne de caractères	35 caractères max.
CG_Classement	Classement	Chaîne de caractères Par défaut : les 17 premiers caractères de l'intitulé, mais peut être modifié.	17 caractères max.
N_Nature	Numéro nature	Numérique : Entier	de 0 à 10 0 = Aucune 1 = Client 2 = Fournisseur 3 = Salarié 4 = Banque 5 = Caisse 6 = Amortissement/Provision 7 = Résultat bilan 8 = charge 9 = Produit 10 = Résultat Gestion
CG_Report	Type report	Numérique : Entier	de 0 à 2 0 = Aucun 1 = Solde 2 = Détail
CR_Num	Numéro compte reporting	Chaîne Alphanumérique Maj.	13 caractères max.
CG_Raccourci	Raccourci	Chaîne Alphanumérique Maj.	7 caractères max.
CG_Saut	Saut de lignes	Numérique : Entier	de 0 à 99 0 = Saut de page > = 1 : nbre Saut de lignes

Tableau 10: Cartographie de la table F\_Compteg sous Sage ligne 100 comptabilité

✓ **F Taxe :**

Cette table contient des informations sur les Taxes et le taux de taxe.

Nom	Signification	Type de données	Domaine de validité
TA_Intitule	Intitulé	Chaîne de caractères	35 caractères max.
TA_TTaux	Type taux	Numérique : Entier	0 à 2 0 = Taux % 1 = Montant (francs) 2 = Quantité
TA_Taux	Taux	Numérique : Réel double	
TA_Type	Type	Numérique : Entier	0 à 9 0 = TVA/Débit 1 = TVA/Encaissement 2 = TP/HT 3 = TP/TTC 4 = TP/Poids 5 = TVA/CEE 6 = Surtaxe

Tableau 11:Cartographie de la table F\_Taxe sous Sage ligne 100 comptabilité

✓ **F Banque :**

Cette table contient des informations sur les Banques de l'entreprise.

Nom	Signification	Type de données	Domaine de validité
BQ_No	Numéro interne	Numérique : Entier long	Compteur
BQ_Intitule	Intitulé	Chaîne de caractères	35 caractères max.

BQ_Adresse	Adresse	Chaîne de caractères	35 caractères max.
BQ_Complement	Complément	Chaîne de caractères	35 caractères max.
BQ_CodePostal	Code postal	Chaîne de caractères	9 caractères max.
BQ_Ville	Ville	Chaîne de caractères	35 caractères max.
BQ_CodeRegion	Code Région	Chaîne de caractères	25 caractères max.
BQ_Pays	Pays	Chaîne de caractères	35 caractères max.
BQ_Contact	Contact	Chaîne de caractères	35 caractères max.
BQ_Abrege	Abrégé	Chaîne de caractères	17 caractères max.
BQ_ModeRemise	Mode remise	Numérique : Entier	0 à 2 0 = Télétransmission 1 = Papier 2 = Disquette
BQ_BordDevis	Bordereau en devise	Numérique : Entier	0 ou 1 0 = Non 1 = Oui
BQ_DailyDateConv	Date convention Dailly	Date	
BQ_DailyNatJur	Nature Juridique Dailly	Chaîne de caractères	35 caractères max.
BQ_DailyAdresse	Adresse Dailly	Chaîne de caractères	35 caractères max.
BQ_DailyComplement	Complément Dailly	Chaîne de caractères	35 caractères max.
BQ_DailyCodePostal	Code postal Dailly	Chaîne de caractères	9 caractères max.
BQ_DailyVille	Ville Dailly	Chaîne de caractères	35 caractères max.
BQ_DailyRCS	Registre Commerce Société Dailly	Chaîne de caractères	35 caractères max.
BQ_BIC	Code BIC	Chaîne de caractères	11 caractères max.
BQ_CodIdent	Code Identification	Chaîne Alphanumérique Maj	16 caractères max.

Tableau 12: Cartographie de la table F\_Banque sous Sage ligne 100 comptabilité

### ✓ F Journaux :

Cette table contient des informations sur les Journaux Comptables.

Nom	Signification	Type de données	Domaine de validité
JO_Num	Code journal	Alphanumérique, Maj.	6 caractères max.

JO_Intitule	Intitulé	Chaîne de caractères	35 caractères max.
CG_Num	N° compte général de trésorerie	Chaîne Alphanumérique Maj.	13 caractères max.
JO_Type	Type	Numérique : Entier	0 à 4 0 = Achats 1 = Ventes 2 = Trésorerie 3 = Général 4 = Situation
JO_NumPiece	Type numérotation pièce	Numérique : Entier	0 à 3 0 = Manuelle 1 = Continue pour le journal 2 = Continue pour la base 3 = Mensuelle
JO_Contrepartie	Option contrepartie/ligne	Numérique : Entier	0 ou 1 0 = Non 1 = oui
JO_SaisAnal	Option saisie analytique	Numérique : Entier	0 ou 1 0 = Non 1 = Oui
JO_NotCalcTot	Masquer les totaux	Numérique : Entier	0 ou 1 0 = Non 1 = Oui
JO_Rappro	Type rapprochement	Numérique : Entier	0 à 2 0 = Aucun 1 = Contrepartie 2 = Trésorerie
JO_Sommeil	Mise en sommeil	Numérique : Entier	0 ou 1 0 = Non 1 = Oui

**Tableau 13:Cartographie de la table F\_Journaux sous Sage ligne 100 comptabilité**

Champs à renseigner obligatoirement lors de l'ajout :

- JO\_Num.
- JO\_Intitule.
- CG\_Num Si et seulement Si JO\_Type = 2 (Trésorerie).
- JO\_Type.

✓ **F Ecriturec :**

Cette table contient des informations sur les Ecritures comptables.

Nom	Signification	Type de données	Domaine de validité
JO_Num	Code journal	Chaîne Alphanumérique Maj	6 caractères max.
EC_No	Numéro interne	Numérique : Entier long	Compteur
EC_NoLink	Numéro interne de lien	Numérique : Entier long	
JM_Date	Période	Date	
EC_Jour	Jour	Numérique : Entier	1 à 31
EC_Date	Date de saisie	Date	
EC_Piece	Pièce	Chaîne Alphanumérique Maj.	13 caractères max.
EC_RefPiece	Numéro de facture	Chaîne de caractères	17 caractères max.
EC_TresoPiece	Pièce de trésorerie	Chaîne de caractères	17 caractères max.
CG_Num	Numéro compte général	Chaîne Alphanumérique Maj	13 caractères max.
CG_NumCont	N° compte général contrepartie	Chaîne Alphanumérique Maj	13 caractères max.
CT_Num	Numéro compte tiers	Chaîne Alphanumérique Maj.	17 caractères max.
EC_Intitule	Intitulé	Chaîne de caractères	35 caractères max.
N_Reglement	Numéro de règlement	Numérique : Entier	0 à 30
EC_Echeance	Echéance	Date	
EC_Parite	Parité	Numérique : Réel double	
EC_Quantite	Quantité	Numérique : Réel double	
N_Devis	Numéro devise	Numérique : Entier	0 à 32
EC_Sens	Sens	Numérique : Entier	0 ou 1 0 = Débit 1 = Crédit
EC_Montant	Montant	Numérique : Réel double	
EC_Lettre	Type Lettrage	Numérique : Entier	0 ou 1 0 = non lettrée 1 = lettrée
EC_Lettre	Numéro lettre	Chaîne Alphanumérique Maj.	3 caractères max.

**Tableau 14:Cartographie de la table F\_Ecriturec sous Sage ligne 100 comptabilité**

Champs à renseigner obligatoirement lors de l'ajout :

- JO\_Num
- EC\_No
- JM\_Date

## 4.2.2 Reverse engineering OpenERP 6 :

A cet effet, on a exploité les résultats du Reverse Engineering décrit en détails dans le sous-chapitre 3.1.2 .Ces résultats englobent la Comptabilité aussi.

## 4.2.3 Mapping entre Sage Ligne 100 et OpenERP v6 :

Moyennant la cartographie de Sage ligne 100 (module comptabilité) étudiée auparavant, et le MCD de la BD liée à l'OpenERP 6 (tables de la comptabilité), ce mapping entre les deux ERP en question est devenu réalisable.

Ainsi, la chronologie de la migration depuis Sage Ligne 100 vers OpenERP v6 passe via la correspondance entre les tables suivantes :

N°	OpenERP v6	Sage Ligne 100
1	Res_currency	P_Devises
2	Res_partner	F_comptet
3	Res_partner_address	F_contactt
4	Account_account	F_compteg/F_piece
5	Account_taxe	F_taxe
6	Res_banque	F_banque
7	Account_journal	F_journaux
8	Account_move	F_ecriturec
9	Account_move_line	F_ecriturec

Tableau 15:Correspondence du mapping Sage ligne 100 □ OpenERP6 (module comptable)

Lors de cette première phase de l'étude conceptuelle de la solution, il s'agissait de mettre au point le socle de la double moulinette de migration en termes de modélisation.

Pour la suite du projet, le résultat de la reprise du module comptable depuis Sage ligne 100 vers OpenERP6 (BD PostgreSQL), servira en partie comme une source pour notre entrepôt de données décisionnel (les données fiduciaires de RIBATIS alimenteront le datawarehouse de gestion de performance interne du cabinet). Tandis que le même DWH puisera les données journalières depuis les tables de PostgreSQL liées à l'OpenERP 6 (système de production de RIBATIS).

## 4.3 Conception du Datawarehouse « Gestion\_Activité\_RIBATIS » :

### 4.3.1 Conception du modèle multidimensionnel :

#### 4.3.1.1 Axes d'analyse et indicateurs :

En fonction des besoins au vue d'une meilleure gestion de performance interne du cabinet de conseil RIBATIS, et après une étude des sources de données :

- OpenERP6 : données fiduciaires migrées depuis Sage ligne 100  
(Comptabilité, finance ...)
- PostgreSQL : données journalières (système de production interne de RIBATIS) ;

Alors on a conclu que les **données pertinentes requises** pour les analyses cibles de la gestion d'activité du cabinet vont être extraites pour constituer les dimensions suivantes :

Nom de la dimension	Définition	Hiérarchie
Période	Détermine la période de calcul des indicateurs.	Année → Mois → Semaine
Projet	Elle contient des informations sur les projets menés par RIBATIS.	Type projet → Projet

Type projet	Elle spécifie la typologie des projets.	Famille projet → Type projet
Famille projet	Elle détermine les 3 grandes catégories suivant lesquelles les projets sont regroupés.	Famille projet
Collaborateur	Elle contient des informations sur les salariés de RIBATIS.	Collaborateur
Tache	Elle énumère les taches effectuées par un collaborateur conformément à un timesheet donné.	Collaborateur → Timesheet → Tache
Type contrat	Elle précise les 3 types de contrat suivant lesquels les projets sont réglementés.	Type contrat
Client	Elle contient des informations sur les clients de RIBATIS.	Partner → Client

**Tableau 16: Définition des dimensions**

Les mesures ou les indicateurs recensés dans l'ensemble des analyses à considérer pour la gestion d'activité de RIBATIS, se présentent comme suit :

Indicateurs	Description	Formule de calcul	Périodicité
<b>Budget_initial ( en Dh )</b>	Le cout total estimé d'un projet donné.	Budget initial en Dh	Durant tout le projet

<b>Charge_initiale</b> ( en J*H)	La charge totale estimée d'un projet donné.	Charge initiale en J*H	Durant tout le projet
<b>Budget_consomme</b> ( en Dh )	Le cout réellement consommé lors d'un projet donné.	SOMME $\sum_i ( NJC_i * TJM_i )$  Où : - <b>NJC</b> : Nbr de jours œuvrés par un collaborateur dans un projet donné.  - <b>TJM</b> : Taux journalier moyen par type de collaborateur.  <i>(Somme <math>\sum_i</math> : collaborateur)</i>	Mensuel / Trimestriel / Annuel
<b>Charge_consommee</b> ( en J*H)	La charge réellement consommée lors d'un projet donné.	SOMME $\sum_i ( Duree\_consacree_i )$  Où : <b>Duree_consacree</b> : durée de réalisation d'une tâche relative à un projet donné ( par un collaborateur donné)  <i>(Somme <math>\sum_i</math> : tâche )</i>	Mensuel / Trimestriel / Annuel
<b>Taux_consomm_budg et</b> ( en %)	Le pourcentage du budget consommé relativement à celui estimé	Budget_consomme / Budget_initial	Mensuel / Trimestriel / Annuel
<b>Taux_consomm_charg e</b> ( en %)	Le pourcentage de la charge consommée relativement à	Charge_consommee / Charge_initiale	Mensuel / Trimestriel / Annuel

	celle estimée.		
<b>Marge1</b> <b>( en valeur )</b>	L'écart (en valeur )entre le budget estimé initialement et celui consommé réellement.	$Budget\_initial - Budget\_consomme$	Mensuel / Trimestriel / Annuel
<b>Marge2</b> <b>( en % )</b>	L'écart (en %) entre le budget estimé initialement et celui consommé réellement	$(1 - (Budget\_consomme / Budget\_initial)) * 100$	Mensuel / Trimestriel / Annuel

**Tableau 17:Récapitulatif des indicateurs**

Ainsi, l'objectif de ce Datawarehouse est de donner aux décideurs la possibilité de **suivre les indicateurs de performance relatifs aux projets** menés par le cabinet : par famille / type de projet, par collaborateur, par client, par type de contrat et par intervalle temporel.

#### 4.3.1.2 Modèle de conception :

La phase de conception a pour objectif la détermination de la finalité du Datawarehouse. De ce fait, la deuxième étape de cette phase consiste à élaborer un modèle de données qui satisfait les besoins de l'analyse.

Conceptuellement, cette modélisation a donné naissance aux notions de fait et de dimension. Le **fait** modélise le sujet de l'analyse. Il s'agit de la mesure relative aux informations de l'activité analysée. La **dimension**, quant à elle, modélise une perspective de l'analyse. Elle se compose de paramètres correspondants aux informations faisant varier les mesures de l'activité. Ces informations sont généralement des niveaux ou des propriétés de la dimension. L'élaboration d'un modèle conceptuel décisionnel de données

peut être faite en utilisant un modèle en **étoile** ou en **flocon**. Dans les deux cas, le modèle est formé d'une table de fait (ou plusieurs) regroupant les indicateurs de dimensions sur les axes d'analyse. Le tableau suivant dresse une comparaison entre ces deux modes de conception :

	<b>Modèle en étoile</b>	<b>Modèle en flocon</b>
<b>Table de fait</b>	Table (ou plusieurs) central regroupant les mesures	Table (ou plusieurs) central regroupant les mesures
<b>Table de dimension</b>	Dénormalisation des dimensions (une table par dimension)	Normalisation des dimensions (possibilité de regrouper plusieurs tables par dimension)
<b>Avantages</b>	<ul style="list-style-type: none"> <li>- Facilité de navigation : Nombre de jointures limité</li> <li>- Fiabilité des résultats</li> </ul>	Réduction de volumes si les tables et les dimensions sont volumineuses
<b>Inconvénients</b>	<ul style="list-style-type: none"> <li>- Redondance dans les dimensions</li> <li>- Alimentation complexe</li> </ul>	<ul style="list-style-type: none"> <li>- Navigation difficile</li> <li>- Nombreuses jointures.</li> </ul>

**Tableau 18: Comparaison entre les deux modèles de conception**

Le modèle en étoile s'avère le plus adéquat dans notre cas. En effet, le but principal d'un système BI est de faciliter la navigation dans les données, et répondre rapidement aux requêtes des utilisateurs sans se soucier de la phase d'alimentation.

#### **4.3.1.3 Structure du Datawarehouse :**

Nous avons choisi pour la réalisation du DataWarehouse le modèle en étoile, constitué d'une **table de fait** et **8 tables de dimensions**. La table de faits est constituée des clés primaires de chaque table de dimension , plus les entités mesurables qui nous servirons après lors de la génération du cube OLAP. En effet, Le principe d'optimisation de ce modèle en étoile est le suivant : une clé calculée "technique" (clé générique) sert de

jointure relationnelle entre les tables de dimensions et la table de fait. La requête SQL réalise d'abord sa sélection sur les tables de dimensions (peu volumineuses) et ensuite seulement, via les clés ainsi sélectionnées, la jointure avec la volumineuse table de fait.

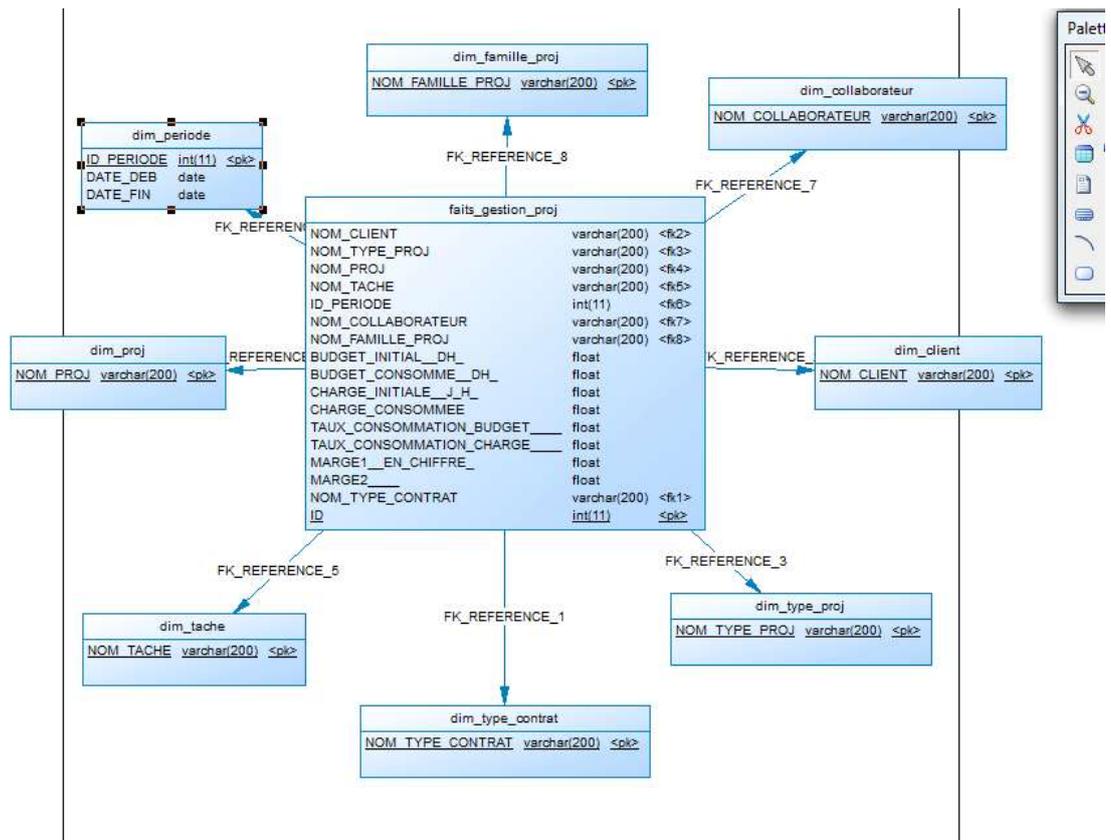


Figure 20 : Conception multidimensionnelle du modèle

#### 4.3.1.4 Tables de dimensions :

Comme précisé, le modèle contient 8 tables de dimensions dont le contenu et la signification ont été défini auparavant ,dans le *tableau 17: "Définition des dimensions "*

#### 4.3.1.5 Table de faits :

La table de faits de notre modèle comporte outre les clés primaires des tables de dimensions précitées, les mesures qui nous seront utiles pour notre analyse.

Ces mesures utiles pour l'analyse (KPI) sont énumérables en **8 indicateurs** . Ils ont été détaillées dans le *tableau 18* : « *Récapitulatif des indicateurs* ».

### 4.3.2 Conception de l'ETL :

Comme expliqué précédemment , les outils ETL gèrent toutes les étapes de la collecte de données au sein des systèmes d'information hétérogènes : SGBD, applications spécifiques, fichiers plats, bases hiérarchiques et base des ERP, depuis le nettoyage de données collectées, la consolidation et la mise en concordance des données éparses jusqu'à leur distribution auprès des applications cibles ou des systèmes décisionnels (analyse, tableaux de bord, etc.).

L'ETL se déroule en trois étapes majeures : l'extraction, la transformation et le chargement des données.

- Extraction des données sources : C'est un processus important car c'est lors de ce processus que nous devons résoudre les problèmes d'intégration et de qualité des données. Deux processus d'extraction sont à prévoir : le premier pour le peuplement initial du Datawarehouse et le second pour les chargements réguliers et incrémentiels.
- Transformation : Application de filtre, agrégation, traitement des données manquantes ou aberrantes et contrôle des rejets, intégrité et cohérence.
- Chargement des données : synchronisation des chargements, transfert de fichiers ou transfert de base à base.
- **Données de stockage** : cette première étape porte sur la consolidation des données source qui serviront à l'alimentation du Datawarehouse. En effet, avant d'effectuer l'extraction des données, il est important de bien connaître leurs contenus, leurs emplacements dans le Datawarehouse. Pour notre cas, les sources de données sont :
  - Système de production (PostgreSQL).
  - OpenERP 6.

- **Préparation de la zone de stockage de données** : La préparation de la zone de stockage permet le nettoyage et la mise en forme des données. Il ne suffit pas de savoir quelles données on souhaite extraire, il faut les mettre en relation les unes avec les autres, et les valider avant leur chargement dans le système décisionnel. C'est à son entrée qu'il faut la valider, éventuellement la rejeter, et la traiter pour l'harmoniser avec les autres informations auxquelles elle va être comparée. Dans cette première tâche nous avons travaillé sur la base de données de l'ERP OpenERP.
- **Extraction des données** : L'extraction des données est effectuée vers les tables de dimensions du Datawarehouse à partir de la source de données, il s'agit d'extraire les différents champs selon les besoins prédéfinis sous forme de colonnes pour pouvoir les insérer dans le Datawarehouse.
- **Contrôle et transformation des données** : Cette étape est la plus importante du processus d'acquisition des données, elle permet de contrôler les données entrantes, de lancer le processus de traitement des rejets, puis consolider et de transformer les données validées en vue de leur intégration dans l'entrepôt de données.

## Chapitre V : Réalisation de la solution

Ce chapitre commencera par étoffer les différentes étapes de réalisation de la montée en version OpenERP V5 → V6 ainsi que les phases de la migration de données (module comptabilité) depuis SAGE ligne 100 vers l'OpenERP V6. Il terminera par expliquer la démarche menée pour l'implémentation de la solution décisionnelle en termes d'étapes de mise en œuvre et de génération d'états de sortie .

## 5. Réalisation de la solution

### 5.1 Environnement de développement :

On avait traité lors du Chapitre III le choix des outils décisionnels Open source moyennant lesquels on va réaliser notre projet. Ainsi , on avait décidé au terme de nos études comparatives, d'utiliser les plateformes suivantes :

- **Talend Open Studio** ( voir *Annexe A* ) : l'ETL qui sera utilisé lors de la migration (SAGE ligne 100 / OpenERP V5 → OpenERP V6 ) ainsi que dans le déploiement de la solution décisionnelle.
- **Palo BI Suite** (cf. Chapitre III/section 2.2 ) : Suite BI destinée à la mise en place de la solution reporting.

### 5.2 Solution finale :

#### 5.2.1 Mise en place de la montée en version OpenERP (v5 → v6) :

La mise en place de la montée en version d'OpenERP de la version 5 vers la nouvelle version 6 , lancée par l'éditeur en janvier dernier, est répartie en plusieurs étapes :

- **Installation des Modules sous l'OpenERP V6 :**

Cette étape consiste à l'installation des modules nécessaires sous l'OpenERP V6 pour pouvoir effectuer la montée en version. C'est l'étape de la préparation du milieu de migration (environnement cible).

Dans cette perspective, on a installé les modules concernés par la montée en version, et qui sont les suivants :

- Module Comptabilité et Finance.
- Module Ressources Humaines.
- Module Production de RIBATIS.
- Module Paie.

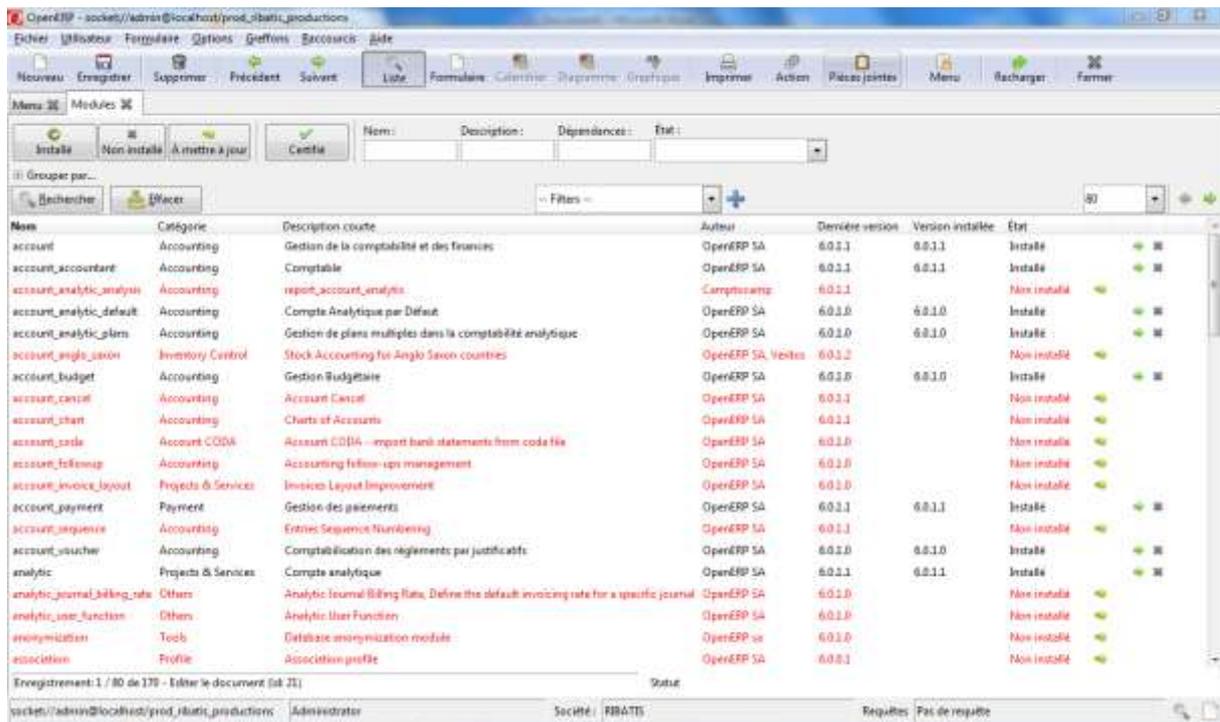


Figure 21 : Administration des modules sous l'OpenERP V6

- **Alimentation des bases de données cibles de l'OpenERP V6:**

Il s'agit de créer des jobs ,sous l'ETL Talend, contenant des composants d'extraction de transformation et de chargement . Moyennant ces derniers, on va extraire les données de leurs sources (base de données PostgreSQL liée à l'OpenERP v5) , faire les conversions de données nécessaires, alimenter les tables qui se correspondent (selon le mapping conceptuel déjà détaillé dans le Chapitre IV cf. section 1.3) et faire des jointures entre les tables lorsque cela est nécessaire. Cette phase est la plus critique de l'ETL puisqu' une simple erreur peut provoquer un bug sur l'OpenERP V6 (environnement cible de la migration).

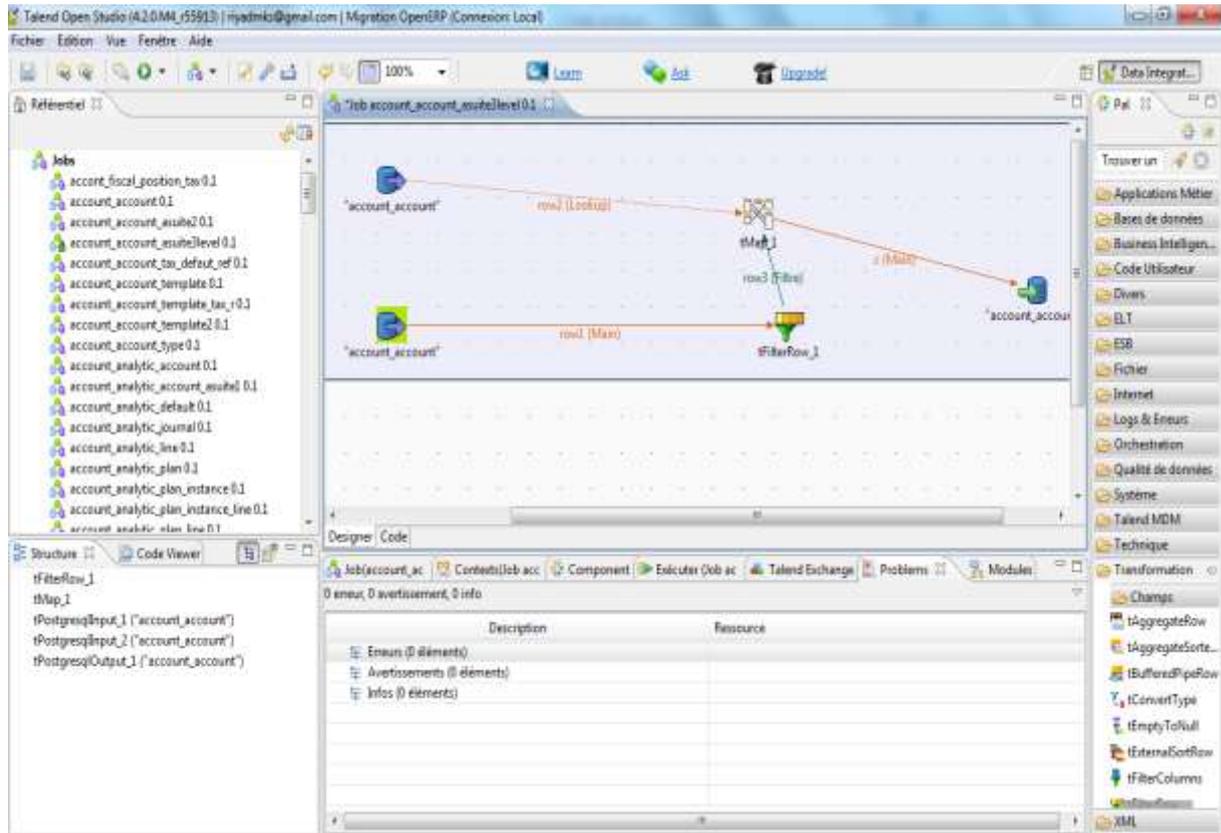


Figure 22 : Exemple d'un Job de migration sous l'ETL Talend (montée en version du module comptabilité : OpenERP V5  $\square$  V6)

- **Déploiement sur le serveur :**

Dans cette étape on va utiliser un serveur distant contenant la base de données propre à l'OpenERP V6, pour l'alimenter via les données de l'OpenERP V5 propres aux quatre modules cités précédemment (cf. section 2.1 installation des modules) . A cet effet, il suffit de reconfigurer les connexions entre l'ETL Talend et la base de données PostgreSQL liée à l'environnement cible de la migration (OpenERP V6 sous la machine serveur ), et exécuter ensuite les jobs déjà construits vers cette cible ( dont l'adresse IP : 192.168.2.81 : 8080)

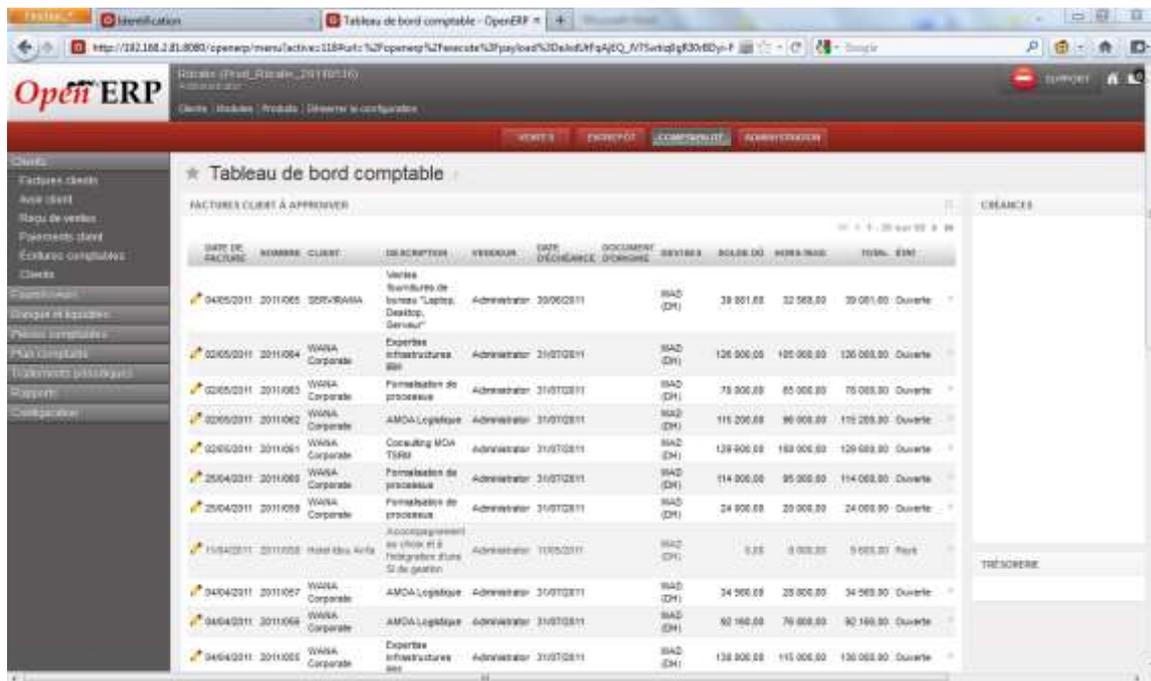


Figure 23: Exemple du contenu du module comptabilité visualisé sous l'environnement cible de la migration (OpenERP V6), sous la machine serveur

### 5.2.2 Mise en place de la migration Sage Ligne 100 vers OpenERP 6

La mise en place de la migration depuis la plateforme Sage Ligne 100 vers OpenERP V6 est structurée comme suit :

- **Installation du Module « Comptabilité » de l'OpenERP V6 :**

Cette étape consiste en l'installation du module Comptabilité et Finance sous l'OpenERP V6 (schéma cible de la mapping) , c'est l'étape de la préparation du milieu de migration.

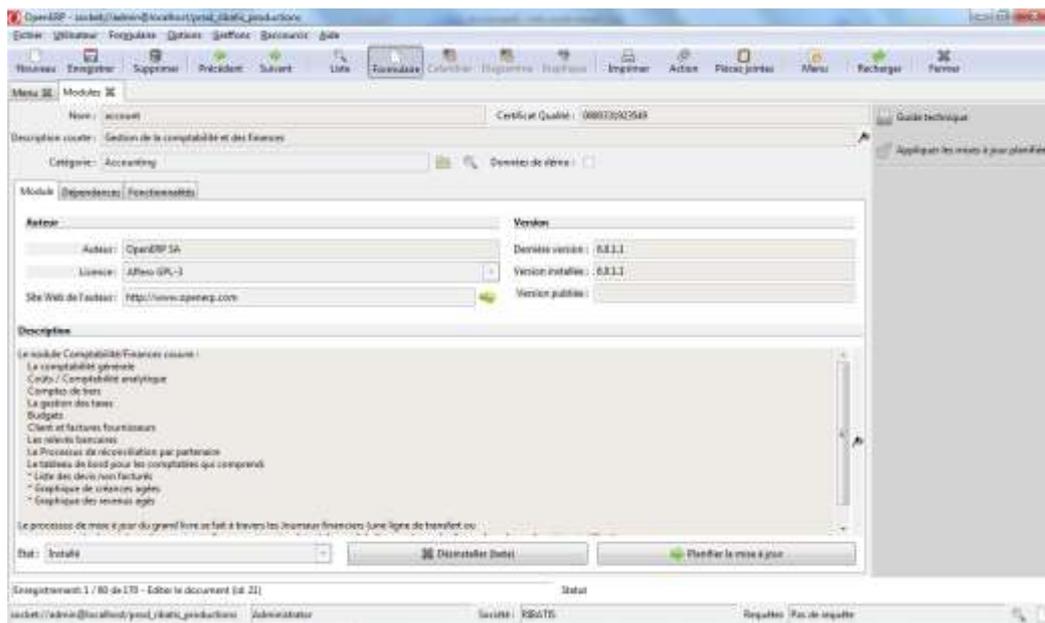


Figure 24 : Le module Comptabilité (schéma cible) installé sous l'OpenERP V6

- **Connectivité Sage Ligne 100 – Talend :**

Il s'agit d'établir la connexion entre Sage Ligne 100 et l'outil de la migration : l'ETL Talend Open Studio via un driver ODBC propriétaire : « PiloteODBC\_SAGE Ligne100\_Version14.04 » qui permet l'accès aux tables de « Sage ligne 100 comptabilité »

On commence par l'installation du driver ODBC sous Windows :

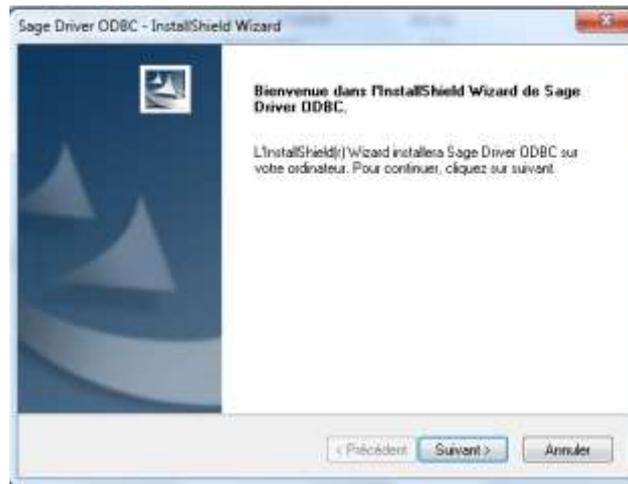


Figure 25 : Assistant d'installation du PiloteODBC\_SAGE Ligne100\_Version14.04

Ensuite on doit configurer une source de données sous Windows par le biais du pilote « Sage Comptabilité 100 » désormais utilisable via une connexion de type C\_Base (à configurer en précisant l'emplacement du fichier « .mae » : schéma source SAGE comptabilité)

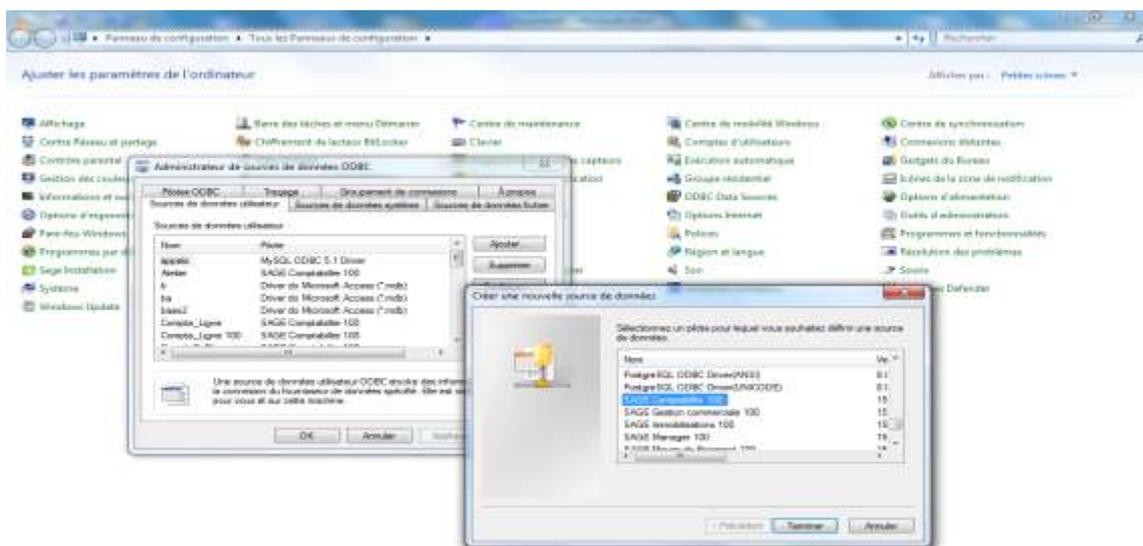
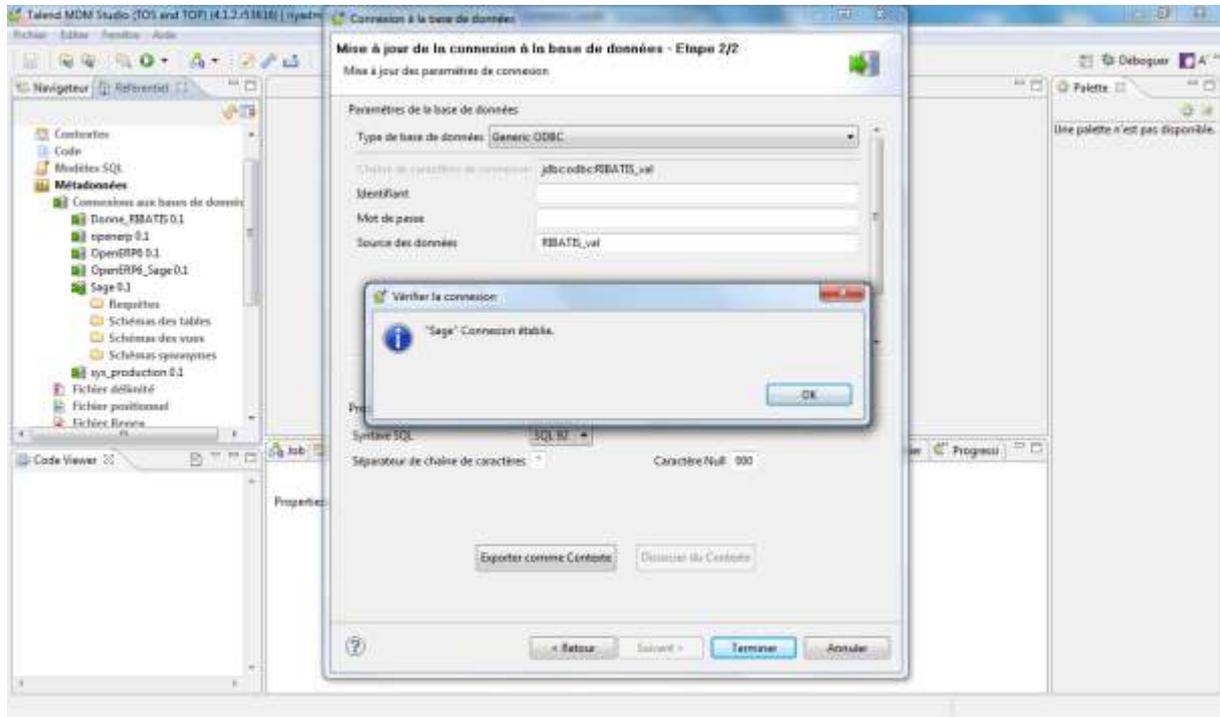


Figure 26: Configuration de la source de données « SAGE ligne 100 comptabilité »

L'étape qui suit consiste à paramétrer une connexion de type « Generic\_ODBC » sous l'ETL Talend Open Studio afin de se connecter directement à la source de données « Sage Ligne 100 » configurée précédemment :



**Figure 27 : Etablissement de la connexion entre Talend et la source de données « SAGE ligne 100 comptabilité »**

- **Exécution du script Parent Left, Parent right :**

Afin de résoudre le problème d'absence des champs *Parent\_Left*, *Parent\_Right* (Bug de l'affichage des comptes générales) dans le schéma source de Sage Ligne 100, et qui sont primordiales pour la table *Account\_Account* du module comptabilité de l'OpenERP V6, alors on doit exécuter un script PL-pgSQL qui va calculer ces 2 champs lors de l'alimentation de la table *Account-Account*.

```

1  --create language plpgsql;
2  create or replace function fix_parent(tabla text, campo_padre text, campo_hijo text) RETURNS varchar as $$
3  DECLARE
4  abuelo record;
5  padre record;
6  hijo record;
7  c integer; -- contador secuencia
8  mx integer; -- numero de registros a actualizar
9  p integer; -- profundidad del arbol de hijos.
10 r varchar; -- resultado
11 nivel_a integer; -- nivel anterior
12 padre_a integer; -- padre anterior
13
14 BEGIN
15     c := 0;
16     for abuelo in execute 'select id from ' || tabla || ' where ' || campo_padre || ' is null order by id' loop
17         nivel_a := 0;
18         for hijo in execute 'with recursive r(id,path,id,parent) as (select 1, ''',id,'' || campo_padre || ' from ' || tabla || ' where id=' || abuelo.id || ') union
19             while nivel_a > hijo.d loop -- de arriba
20                 execute 'update ' || tabla || ' set parent_right=' || c || ' where id=' || padre_a;
21                 c := c + 1;
22
23                 execute 'select ' || campo_padre || ' from ' || tabla || ' where id=' || padre_a into padre_a;
24                 nivel_a := nivel_a - 1;
25             end loop;
26
27             execute 'select count(*) from ' || tabla || ' where ' || campo_padre || '=' || hijo.id into p;
28             if p > 0 then
29                 execute 'update ' || tabla || ' set parent_left=' || c || ' where id=' || hijo.id;
30             else --ultimo nivel
31                 execute 'update ' || tabla || ' set parent_left=' || c || ' where id=' || hijo.id;
32                 c := c + 1;
33             end if;
34             execute 'update ' || tabla || ' set parent_right=' || c || ' where id=' || hijo.id;
35             c := c + 1;
36         end if;
37     end loop;
38
39 END;

```

Figure 28: Exécution du script PL-pgSQL pour la résolution du problème parent right / parent left

- Alimentation des bases de données cible de l'OpenERP V6 :

On applique la même démarche de l'ETL adoptée lors de la montée en version OpenERP V5→V6 sauf que cette fois ci on puise nos données de la « source de données » : SAGE ligne 100 comptabilité.

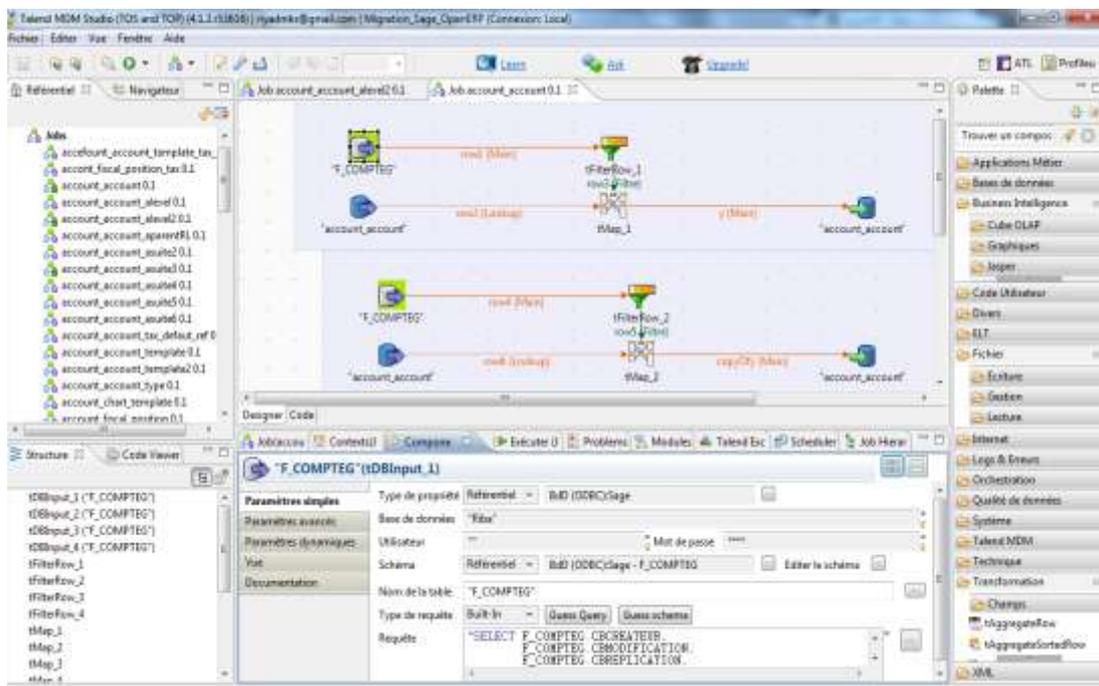


Figure 29: Exemple d'un Job de migration sous l'ETL Talend (migration des comptes de la comptabilité générale depuis SAGE ligne 100 vers OpenERP6)

- **Automatisation des Jobs :**

Il s'agit d'automatiser les jobs afin de déclencher la migration automatiquement, donc il suffit d'exécuter le job Root et attendre environ 30 min pour que la migration prenne fin.

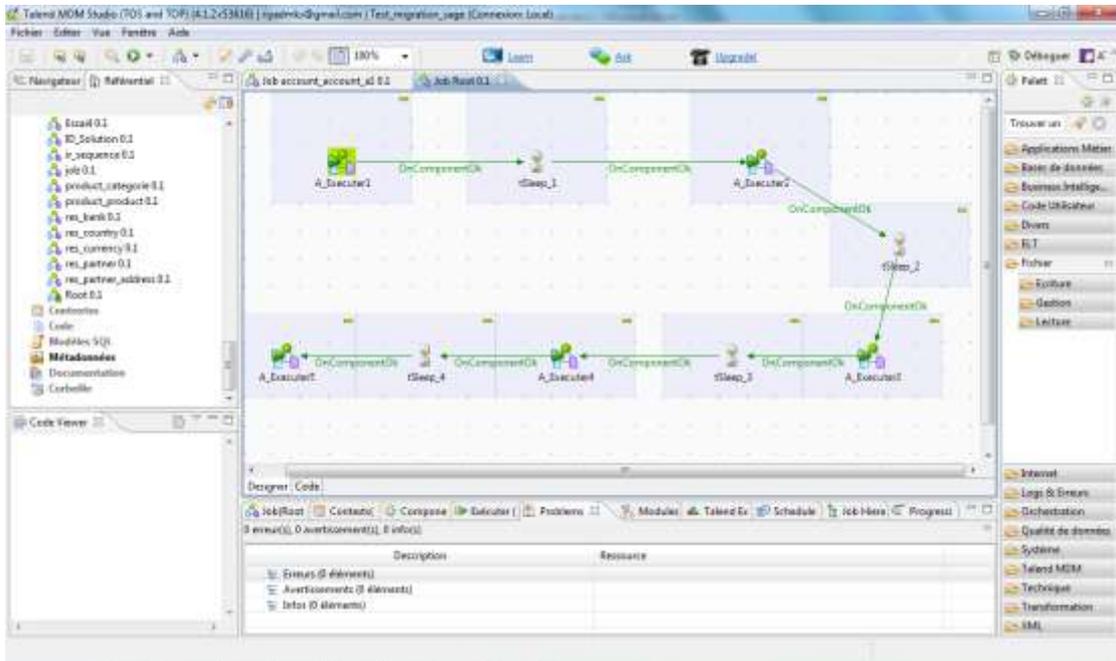


Figure 30 : Automatisation des Jobs de la migration SAGE ligne 100 □ OpenERP V6 sous Talend

- **Planification automatique de l'exécution :**

Après avoir construit les jobs de la migration de Sage Ligne 100 vers OpenERP V6 sous Talend, et l'automatisation de cette migration, on a intérêt à réaliser la planification automatique de l'exécution de cette moulinette de reprise de données. Ainsi on peut programmer le traitement de la migration à une date précise.

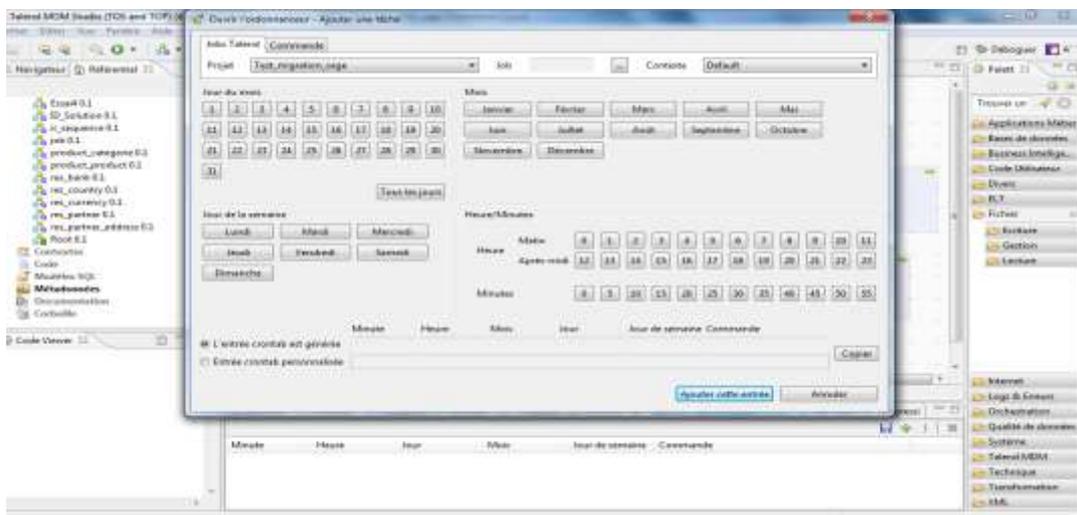


Figure 31: Planification automatique de l'exécution des Jobs de la migration sous Talend

### 5.2.3 Mise en place du Datawarehouse :

#### « Gestion\_Activité\_RIBATIS »

#### 5.2.3.1 Réalisation de la phase ETL :

On a traité lors du chapitre précédent (cf. section 3.2 ) les étapes de création et d'alimentation de notre Datawarehouse depuis les sources de données croisées (extraction depuis le système de production sous PostgreSQL ou via les données financières migrées depuis SAGE vers OpenERP6).

Table	Action	Lignes	Type	Interclassement	Taille	Perte
dim_client		82	InnoDB	latin1_swedish_ci	16,0 Kio	-
dim_collaborateur		12	InnoDB	latin1_swedish_ci	16,0 Kio	-
dim_famille_proj		3	InnoDB	latin1_swedish_ci	16,0 Kio	-
dim_periode		0	InnoDB	latin1_swedish_ci	16,0 Kio	-
dim_proj		21	InnoDB	latin1_swedish_ci	16,0 Kio	-
dim_tache		0	InnoDB	latin1_swedish_ci	16,0 Kio	-
dim_type_contrat		3	InnoDB	latin1_swedish_ci	16,0 Kio	-
dim_type_proj		30	InnoDB	latin1_swedish_ci	16,0 Kio	-
faits_gestion_proj		35	InnoDB	latin1_swedish_ci	144,0 Kio	-
<b>9 table(s)</b>	<b>Somme</b>	<b>198</b>	<b>MyISAM</b>	<b>latin1_swedish_ci</b>	<b>272,0 Kio</b>	<b>0</b>

Figure 32 : Datawarehouse relationnel alimenté depuis les sources de données croisées

La réalisation de l'étape de chargement est schématisée comme suit :

- **Charger les dimensions** : chaque table de dimension du datawarehouse ci dessus doit alimenter sa dimension correspondante au niveau de l'outil de restitution Palo BI Suite.

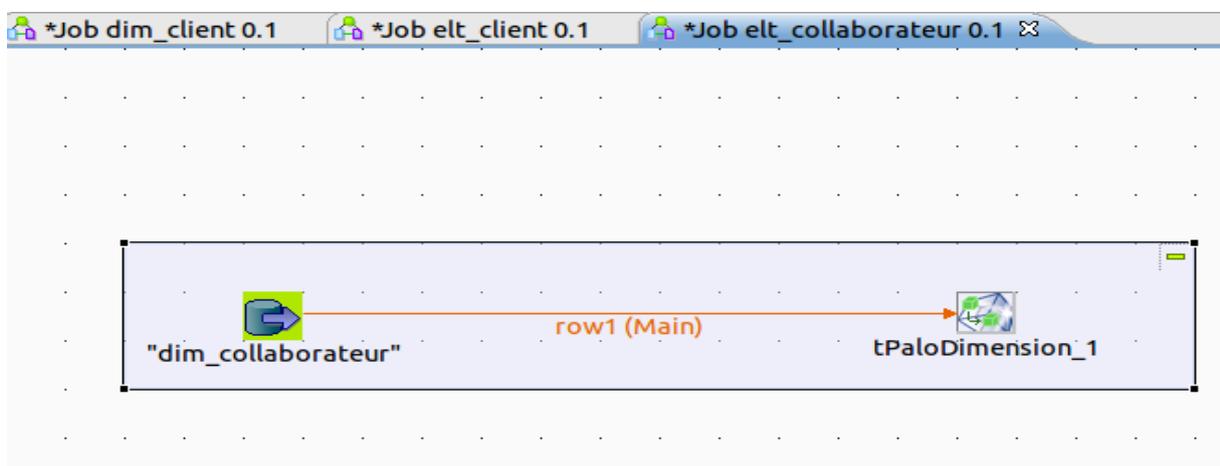


Figure 33:Flux de chargement de la dimension collaborateur sur l'outil Palo

- Charger la table de faits** : la table de faits étant déjà alimentée par l'extraction des données sources (Chapitre IV , cf. section 3.2), alors il reste de calculer les mesures prédéfinis qu'elle contient et de charger les résultats sur l'outil de restitution Palo BI Suite. Un fichier délimité (csv) contenant les règles et les résultats de calcul de ces indicateurs est utilisé à cet effet.

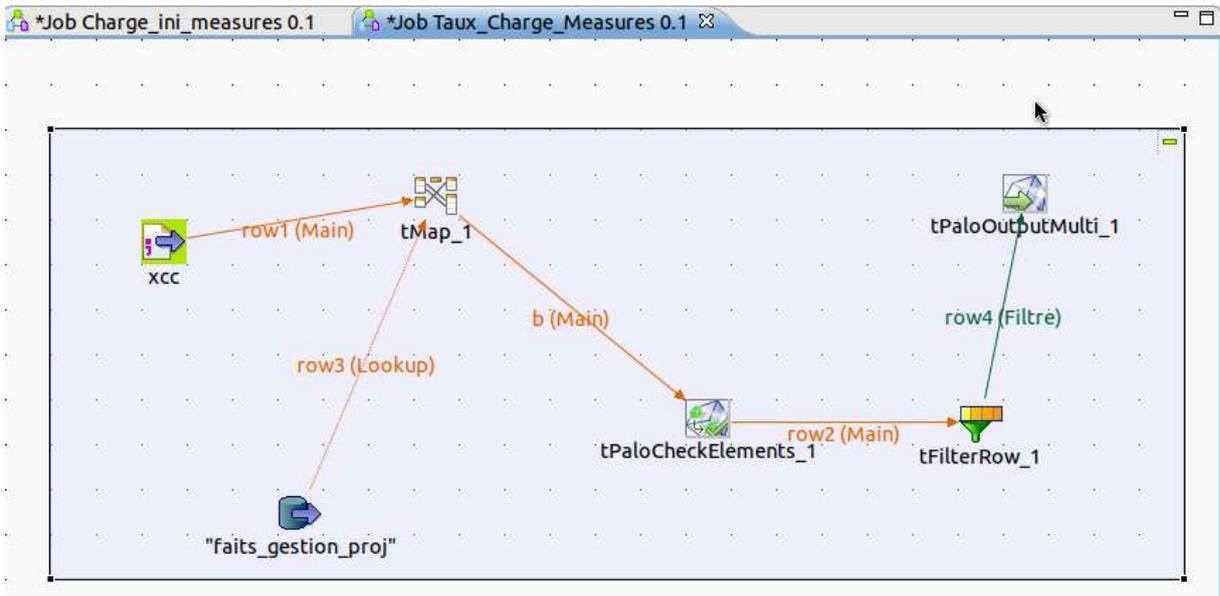


Figure 34: Flux de chargement de la table de fait (dimensions+indicateurs calculés) sur l'outil Palo

### 5.2.3.2 Restitution du cube OLAP :

Le chargement des dimensions et du contenu de la table de fait vers l'outil de restitution nécessite au préalable la construction du schéma du cube multidimensionnelle OLAP au niveau de l'outil «JPalo Client» qui va recevoir ces entités. En effet, l'alimentation de ces données transformées suivant le schéma OLAP permet de convertir les données sources croisées (issues des bases de données relationnelles) en information pertinentes et faciles à exploiter, grâce à la création d'un cube de données. La création d'un cube sous «JPalo Client» va permettre d'améliorer les performances d'analyse en mettant en place une base de données multidimensionnelle à partir de la base de données issus de PostgreSQL.

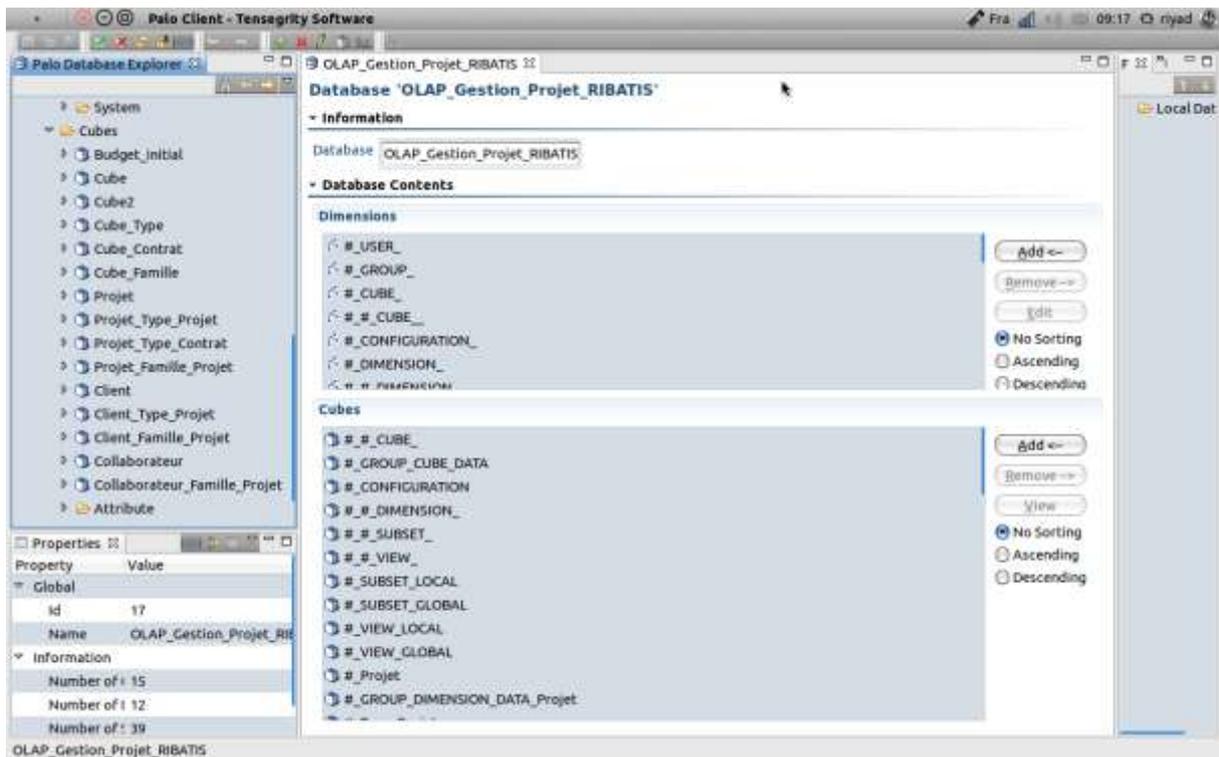


Figure 35: Schématisation du cube OLAP sous JPalo Client

### 5.2.3.3 Restitution des tableaux d'analyses croisées

JPalo Client permet facilement de passer en revue tous les indicateurs de performance clés déjà préétablis. Ces derniers constituent les mesures utilisées pour évaluer l'entreprise. L'interaction des facteurs régissant les indicateurs de performance permet de bénéficier d'informations pertinentes pour faciliter la décision.

Ainsi, il donne la possibilité d'avoir des réponses à des questions précises concernant la mesure souhaitée, à une date donnée, en interaction avec d'autres dimensions (agrégations suivant les axes d'analyse).

Dans ce qui suit nous allons présenter la restitution des différents tableaux d'analyses croisées moyennant l'outil JPalo Client.

En l'occurrence, la figure ci dessous illustre un exemple de restitution des mesures comme : budget initial / consommé , charge initiale / consommée, taux consommation budget ; relativement aux différents projets menés lors de l'exercice 2011 (jusqu'au mois5).

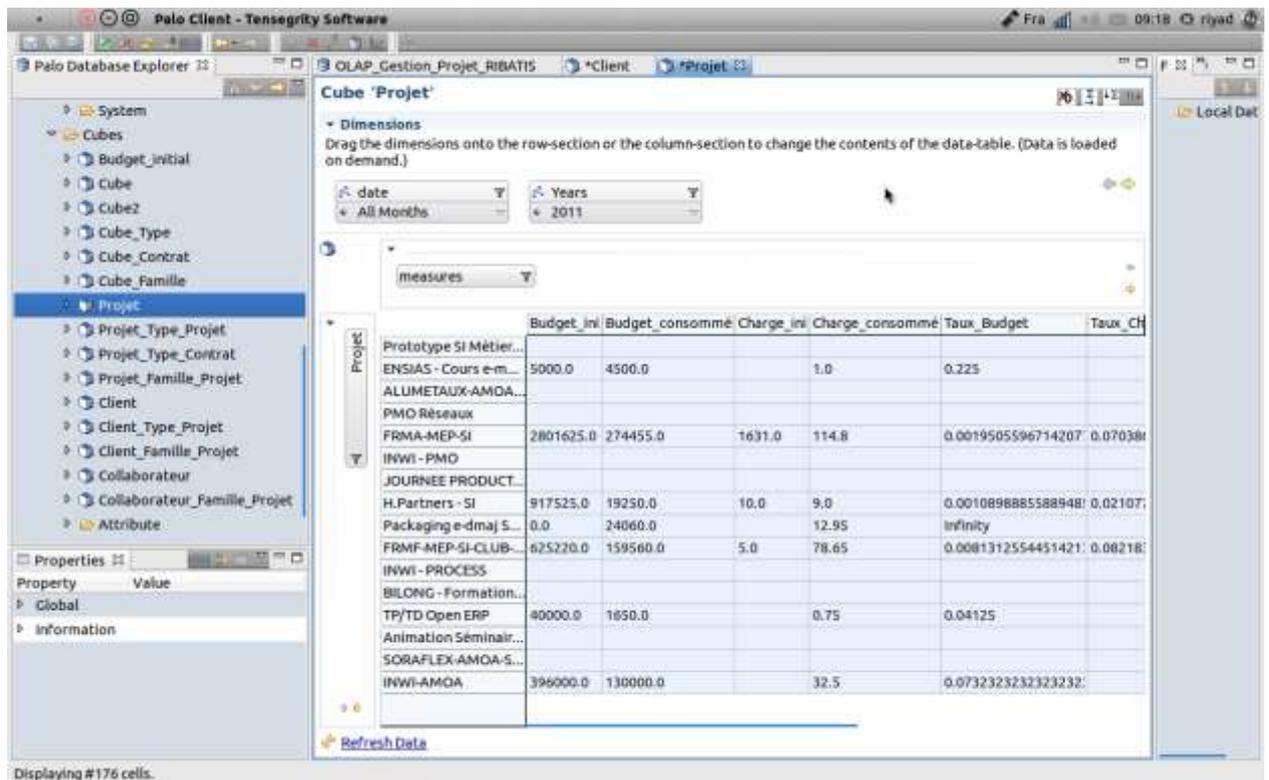


Figure 36:Exemple de tableau d'analyses croisées sous JPalo Client

### 5.2.3.4 Restitution des tableaux de bord :

Dans cette partie on va générer des tableaux de bord adaptés aux différentes problématiques d'analyses exigées par le Client RIBATIS en terme de gestion de son activité interne. Ainsi on va exploiter les tableaux d'analyses croisées créés précédemment (cf. section 2.3.3) touchant aux métiers projets, collaborateurs et finalement clients.

- **Exemple 1 :**

Dans cet exemple, on a généré un tableau de bord en graphique circulaire (camembert) afin de savoir la répartition des types de projets en pourcentage au cours du troisième trimestre de l'année 2010. On constate en effet que la majorité des projets menés lors de cette période bien précise étaient de type AMOA (moitié des projets) et en second lieu de type intégration des solutions SI (33,33%).

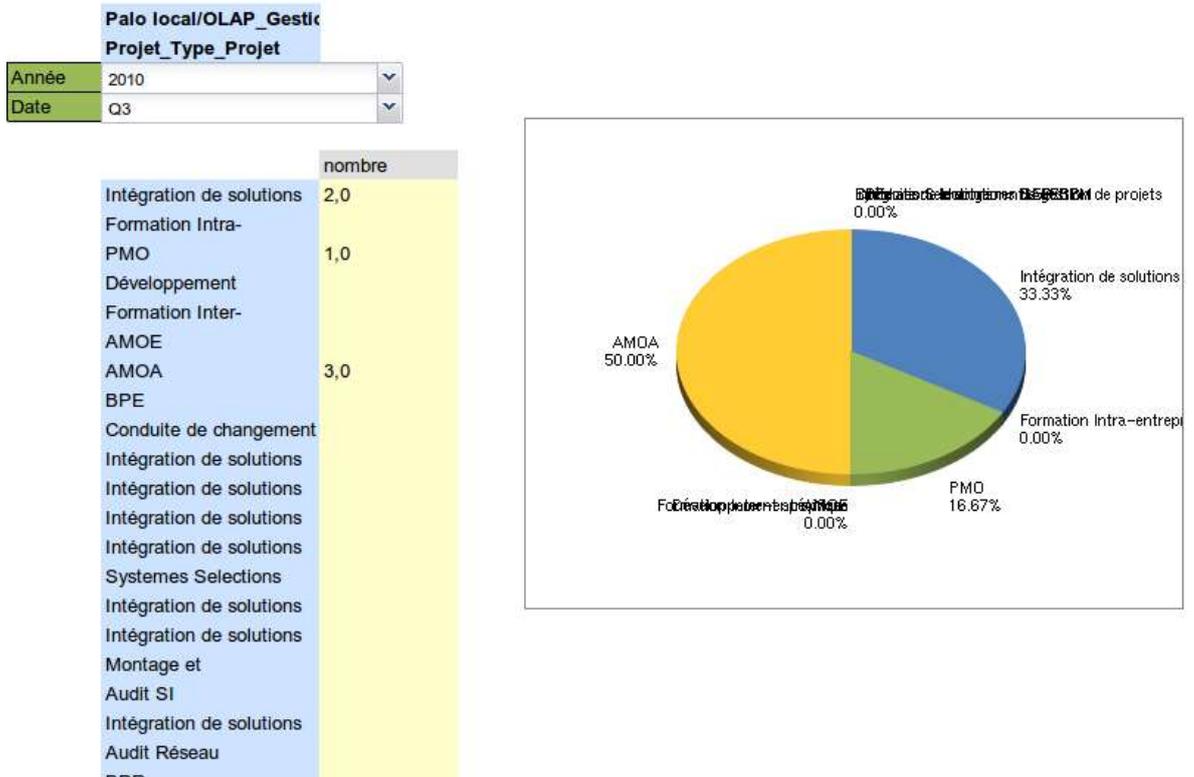


Figure 37 : Tableau de bord circulaire illustrant la répartition des projets selon leur typologie (Troisième trimestre de l'année 2010)

**Exemple 2 :**

Le but de ce second exemple est de donner une vision sur la charge consommée en J\*H et le budget consommé en Dh par chaque collaborateur durant l'exercice de 2010.

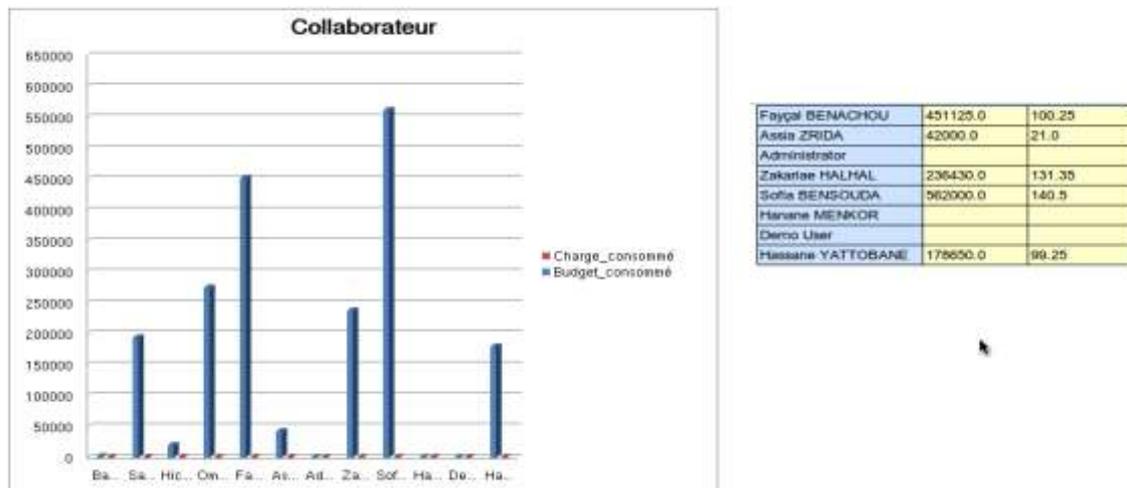
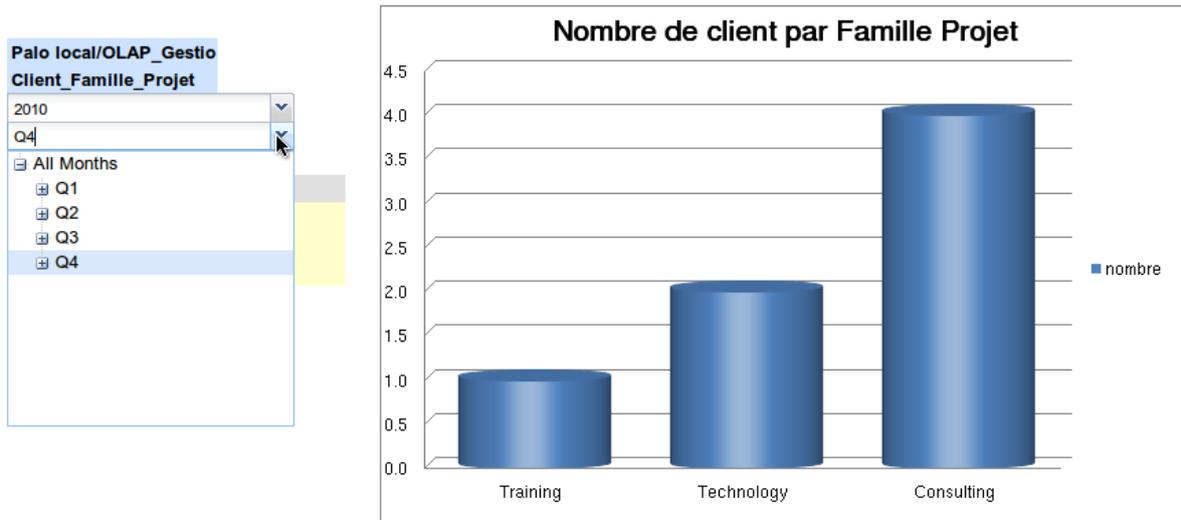


Figure 38 : Tableau de bord en histogramme illustrant la l'implication des collaborateurs dans les projet en terme de charge / budget

- **Exemple 3 :**

Ce troisième exemple de tableau de bord généré donne une idée claire sur le nombre de clients par famille de projets menées. Ainsi on remarque que la famille des projets « Consulting » s'acquiert le plus grand nombre de clients lors du quatrième trimestre de l'année 2010.



**Figure 39 : Tableau de bord en histogramme illustrant la répartition des Clients par famille de projets lors du quatrième trimestre de l'année 2010**

## Glossaire :

### A

**Agrégat :** Un agrégat est une consolidation d'indicateurs précalculée dans le but d'offrir de meilleures performances aux requêtes des utilisateurs. Le data warehouse stocke ainsi plusieurs agrégats, définis d'après les demandes récurrentes des utilisateurs.

### B

**Business Intelligence:** L'informatique décisionnelle ,désigne les moyens, les outils et les méthodes qui permettent de collecter, consolider, modéliser et restituer les données, matérielles ou immatérielles, d'une entreprise en vue d'offrir une aide à la décision et de permettre aux responsables de la stratégie d'entreprise d'avoir une vue d'ensemble de l'activité traitée.

**Benchmarking:** un processus continu de recherche, d'analyse comparative, d'adaptation et d'implantation des meilleures pratiques pour améliorer la performance des processus dans une organisation.

**Bugtrackers :** Un logiciel de suivi de problèmes ou système de suivi de problèmes est un logiciel qui permet d'aider les utilisateurs et les développeurs à connaître le problème (Bug) afin d'améliorer la qualité d'un logiciel.

### C

**Choix Multi- Critère:** Il s'agit de méthodes et de calculs permettant de choisir la meilleure solution ou la solution optimale parmi tout un ensemble de solutions, l'alternative de type OUI-NON n'étant qu'un cas particulier du cas général.

### D

**DataWarehouse:** Pôle de données organisées spécifiquement pour répondre à des besoins décisionnels. Les données issues des sites de production sont extraites, transformées et enregistrées dans le data warehouse afin de permettre leurs analyses. Les données du data warehouse sont orientées sujet, intégrées, non volatiles, agrégées dans le temps et documentées.

**DataMart:** Le data mart est un pôle de données répondant à toutes les caractéristiques du data warehouse, mais d'une volumétrie restreinte. Un data mart peut ainsi être construit autour d'une fonction particulière (par exemple contrôle de gestion), d'un sujet précis (par exemple la promotion) ou d'une granularité de données réduite

**Dimension :** Ensemble logique d'informations constituant un axe d'analyse des indicateurs. Exemple : le temps, la structure de gamme, le client, la promotion, les établissements ou agences...

E

**ERP:** Appelé également PGI (Progiciel de Gestion Intégré), il propose une bibliothèque de processus paramétrables à l'échelle de l'entreprise pour automatiser la gestion de ses activités.

**ETL :** Terme générique pour les logiciels d'alimentation du data warehouse, chargés de collecter les données sources, de les transformer et de les intégrer en informations, et de les charger dans la base de données d'aide à la décision.

F

**Fait :** Ensemble logique d'informations décrivant un événement de gestion. Exemple : ligne de vente. Le fait est porteur de l'indicateur qu'il mesure. Exemple : montant TTC de la ligne de vente.

G

**Gestion de performance :** La gestion de la performance est un moyen d'évaluer la rentabilité et les exploits de tous les employés et employées qui sont présents au sein d'une entreprise.

H

**Hierarchie :** On définit une hiérarchie dans une dimension lorsque plusieurs membres (éléments ou occurrences) de cette dimension permettent de réaliser une analyse de type drill-down. Exemple : dans la dimension temps, les informations « année », « mois » et « jour » définissent une hiérarchie.

I

**Indicateur:** Information généralement numérique destinée à mesurer un événement représentatif du métier de l'entreprise. Les indicateurs peuvent être additifs (ils se somment selon toutes les dimensions définies), semi-additifs (ils ne se somment que selon certaines dimensions) ou non-additifs (ils ne se somment pas).

J

**Jointure:** un produit cartésien de deux tables. On appelle équijointure une  $\theta$ -jointure dont la qualification est une égalité entre deux colonnes.

K

**KPI :** (Voir Indicateur).

M

**Méta-données :** Donnée décrivant une donnée. De manière générale, on définit comme métadonnées l'ensemble de la documentation technique et fonctionnelle du data warehouse. Les métadonnées constituent ainsi le dictionnaire ou référentiel du data warehouse.

**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.

**Modèle en étoile** : Type de modèle consistant à regrouper les indicateurs dans une table centrale (la table des faits) et les dimensions dans des tables satellites à cette table centrale. Certains outils permettent d'optimiser les performances des requêtes interrogeant un modèle en étoile. En anglais : star schema.

## O

**ODBC**: Interface définie par Microsoft, basée sur CLI, pour accéder aux bases de données.

**OLAP** : Littéralement processus d'analyse en temps réel. Par extension, organisation des données décisionnelles selon une approche multidimensionnelle.

## P

**PLSQL**: Un langage procédural propriétaire créé par Oracle et utilisé dans le cadre de bases de données relationnelles. Il a été influencé par le langage Ada. Il permet de combiner des requêtes SQL et des instructions procédurales (boucles, conditions...), dans le but de créer des traitements complexes destinés à être stockés sur le serveur de base de données (objets serveur), comme par exemple des procédures stockées ou des déclencheurs.

## R

**Reverse engineering**: La rétro-ingénierie (traduction littérale de l'anglais reverse engineering), également appelée rétro-conception, ingénierie inversée ou ingénierie inverse, est l'activité qui consiste à étudier un objet pour en déterminer le fonctionnement interne ou la

## S

**SGBD**: Ensemble de logiciels qui fait vivre une base relationnelle ou multidimensionnelle. On parle de SGBDR lorsqu'il s'agit d'un système de gestion de base de données relationnelle.

**SQL** : Langage de requête structuré permettant d'interroger et de mettre à jour les informations des bases de données relationnelles. Le langage SQL est un standard défini par les organismes de normalisation ANSI et ISO.

## T

**Transformation de données** : Processus automatisé qui utilise les données brutes extraites pour les transformer selon les règles de gestion de l'entreprise et les préparer selon les besoins des utilisateurs.