

IUT Nancy Charlemagne

Université de Lorraine

2 ter boulevard Charlemagne

BP 55227

54052 Nancy Cedex

Département informatique

**Réalisation de reports,**

**Automatisation d’impression de factures en masse,**

**Création d'une interface d'ajout de personnel,**

**Gestion de points logistiques dans la cartographie,**

**Création d’un serveur de supervision**

**Rapport de stage de DUT Informatique**

**Entreprise : Transports VIGNERON**

**Alexis CESARO**

**Tuteur : El Haj LAAMRI**

**Année universitaire 2017-2018**

# Remerciements

Je tiens à remercier tout d’abord Mr Jean-François BONNE pour avoir retenu ma candidature de stage, puis pour toute l’aide, les conseils et la confiance qu’il m’a accordés tout au long du stage.

Je tiens également à remercier l’ensemble des employés du groupe VIGNERON avec qui j’ai été amené à travailler.

Je remercie également Mr El Haj LAAMRI pour le suivi qu’il a effectué pendant mon stage.

# Sommaire

[Introduction 1](#_Toc516846865)

[1) Présentation 2](#_Toc516846866)

[1.1) Présentation du groupe VIGNERON 2](#_Toc516846867)

[1.2) Présentation du service informatique 2](#_Toc516846868)

[1.3) Présentation rapide du mode de fonctionnement et de la structure de la base de données 3](#_Toc516846869)

[1.4) Sujets du stage 4](#_Toc516846870)

[1.5) Logiciels utilisés 5](#_Toc516846871)

[1.5.1) OMP R2000X 5](#_Toc516846872)

[1.5.2) Oracle SQL Developer 6](#_Toc516846873)

[1.5.3) Crystal Reports 7](#_Toc516846874)

[1.5.4) Microsoft Office Excel 2010 9](#_Toc516846875)

[1.6) Remarques sur la base de données 10](#_Toc516846876)

[2) Ajout de points logistiques pertinents (Les points Tred Union TU) 12](#_Toc516846877)

[2.1) Besoins du sujet 12](#_Toc516846878)

[2.2) Réalisation de ce sujet 12](#_Toc516846879)

[2.3) Difficultés rencontrées 13](#_Toc516846880)

[2.4) Améliorations possibles 14](#_Toc516846881)

[3) Interface d’ajout de personnel dans la base de données 14](#_Toc516846882)

[3.1) Besoin et sujet 14](#_Toc516846883)

[3.2) Réalisation du formulaire 15](#_Toc516846884)

[3.3) Réalisation du code VBA 17](#_Toc516846885)

[3.3.1) Contrôles de cohérence 17](#_Toc516846886)

[3.3.2) Soumettre la demande de création de personnel 17](#_Toc516846887)

[4) Convertisseur d’export OMP vers logiciel de paie (SAGE) pour les congés 17](#_Toc516846888)

[4.1) Besoin et sujet 17](#_Toc516846889)

[4.2) Réalisation du sujet 18](#_Toc516846890)

[4.2.1) Analyse préliminaire 18](#_Toc516846891)

[4.2.2) Mise en page lors de l’exécution 19](#_Toc516846892)

[4.2.3) Interprétation des périodes de congés 19](#_Toc516846893)

[4.2.4) Traduction des codes R2000X vers les codes SAGE Paie 20](#_Toc516846894)

[4.3) Description de l’exécution du programme 20](#_Toc516846895)

[4.4) Amélioration du programme 21](#_Toc516846896)

[5) Impression de factures en masse 22](#_Toc516846897)

[5.1) Besoin du sujet 22](#_Toc516846898)

[5.2) Analyse préliminaire 22](#_Toc516846899)

[5.3) Réalisation 23](#_Toc516846900)

[5.3.1) Première étape : Procédure VT\_ECRIRE\_PDF\_Facture 23](#_Toc516846901)

[5.3.2) Deuxième étape : Script Bash d’impression 23](#_Toc516846902)

[5.3.3) Troisième étape : Procédure VT\_EDITER\_PDF\_FACTURE 24](#_Toc516846903)

[5.3.4) Quatrième étape : VT\_EDITER\_PDF\_FACTURES et Liaison avec OMP 25](#_Toc516846904)

[5.4) Résolution du soucis d’intégration dans R2000X 26](#_Toc516846905)

[6) Maintenance des serveurs 26](#_Toc516846906)

[7) Mise en place d’un serveur de supervision avec Centreon 27](#_Toc516846907)

[7.1) Présentation de Centreon 27](#_Toc516846908)

[7.2) Besoins 27](#_Toc516846909)

[7.3) Réalisation 28](#_Toc516846910)

[7.3.1) Documentation et recherches 28](#_Toc516846911)

[7.3.2) Installation et premières difficultés. 28](#_Toc516846912)

[7.3.2) Configuration des premiers appareils 29](#_Toc516846913)

[7.3.3) Avantage et configuration futur 30](#_Toc516846914)

[8) Script de contrôle de sauvegarde 30](#_Toc516846915)

[8.1) Besoins 30](#_Toc516846916)

[8.2) Réalisation et fonctionnement 31](#_Toc516846917)

[8.3) Les différentes versions 31](#_Toc516846918)

[9) Création ou modification de reports 32](#_Toc516846919)

[9.1) Récépissé depuis OT et Récépissé par voyage 32](#_Toc516846920)

[9.1.1) Première modification : informations cachées 32](#_Toc516846921)

[9.1.2) Seconde modification : dates erronées et ajouts d’informations 33](#_Toc516846922)

[9.2) Bordereau de groupage 34](#_Toc516846923)

[9.2.1) Première modification : montant du transport 34](#_Toc516846924)

[9.2.1) Deuxième modification : ajout d’informations 34](#_Toc516846925)

[9.2.1) Troisième modification : suppression du point logistique Garonor pour Claude 35](#_Toc516846926)

[9.3) Fourniture Date 35](#_Toc516846927)

[9.4) Majoration Gazole 36](#_Toc516846928)

[9.5) Relevé kilométrique 37](#_Toc516846929)

[Conclusion 39](#_Toc516846930)

# Introduction

Etant étudiant en deuxième année en DUT Informatique à l’IUT Charlemagne à Nancy, j’ai dû effectuer, pour pouvoir valider mon diplôme, un stage en entreprise de dix semaines. J’ai réalisé le mien au service informatique des Transports VIGNERON. Il a duré du 09 avril au 15 juin 2018. Mon maître de stage était Mr Jean-François BONNE, qui est Directeur des Services d’Information du groupe Vigneron.

Les sujets de mon stage étaient très diversifiés et répondaient à un très grand nombre de besoins de l’entreprise. Du développement d’une interface en VBA permettant aux utilisateurs d’ajouter un nouvel employé dans la base de données, à la mise en place d’un serveur de supervision, en passant par la réalisation de différents « reports » à l’aide du logiciel Crystal Reports pour répondre à divers besoins et demandes des utilisateurs.

# 1) Présentation

## 1.1) Présentation du groupe VIGNERON

Le **groupe VIGNERON** est un groupe de sociétés dirigées par Mr Joël VIGNERON. L’activité du groupe est le **transport** (principalement le transport routier de marchandises) et la **logistique**. Le groupe dispose d’une flotte de plus de 250 camions et emploie plus de 300 salariés. La société-mère, basée à Ludres, s’est agrandie au cours de son existence en rachetant d’autres sociétés de la région pour agrandir et diversifier son activité.

Le périmètre de mon stage comprenait douze sociétés : VIGNERON Transports, VIGNERON Logistique, VIGNERON Distribution, THIEBAUT Transports, CLAUDE Transports, SOLUTRANS Logistique, Transports MOINE, FRA Transport Spécial Rapide, FRA Presse, FRA Diffusion, Trans Euro Presse et Sud Distribution Logistique. Chacune de ces sociétés a un domaine d’activité spécialisé dans le secteur du transport et de la logistique. Les sociétés sont basées sur plusieurs sites, situés principalement aux alentours de Nancy (Ludres, Champigneulles, Richardménil) mais également Claye-Souilly (77 Seine et Marne).

Le groupe est client depuis 2013 de la société OMP Informatique Transport, qui lui fournit son environnement de gestion appelé R2000X, accompagné d’une base de données Oracle. Les douze sociétés travaillent avec cette base de données unique, composée de plus de 1000 tables et de plus de 10 giga-octets.

## 1.2) Présentation du service informatique

Le Service Informatique du groupe VIGNERON est constitué d’un seul employé, Mr Jean-François BONNE. Il doit gérer et maintenir le parc informatique et téléphonique, gérer les seize serveurs de la société et assurer la hotline de dépannage. Il est responsable également du bon fonctionnement de l’ERP (Enterprise Ressource Planning) R2000X dans l'entreprise. Mr Bonne dispose de deux bureaux : un à Richardménil dans les locaux de la société FRA Transport Spécial Rapide, et un à Ludres dans les locaux administratifs de la société Transports VIGNERON, dans lequel j’ai effectué mon stage.

## 1.3) Présentation rapide du mode de fonctionnement et de la structure de la base de données

Tous les sujets sur lesquels j’ai été amené à travailler ont eu un rapport avec un ou plusieurs aspects de la base de données. J’ai dû me familiariser au mode de fonctionnement de ces sociétés pour bien comprendre la structure des données et des tables que j’avais à traiter.

Avant qu’une commande soit acceptée, des étapes de négociation de gré à gré et de devis se produisent avec le client. Lorsque la commande est passée, un **ordre de transport** est inséré dans la base de données dans la table ORTR, contenant les informations de la commande du point de vue externe (adresse de chargement, adresse de livraison, dates, contraintes…).

Cet ordre de transport sera désigné à tout moment par deux identifiants principaux : OT et ORTR, dont les méthodes de comptage diffèrent et qui sont chacun utilisés par différents services.

Le service d’exploitation va affecter un ou plusieurs camions, chauffeurs, etc., sur cet ordre de transport. On appelle **phase** une étape de ce transport au cours de laquelle la marchandise est chargée, transportée et déchargée. Les données des phases sont stockées dans la base dans la table OTTP.

Certaines des sociétés du groupe transportent généralement des chargements entiers (lot complet : société VIGNERON Transport) et d’autres ne transportent que quelques palettes par commande (lot partiel : société CLAUDE) ou des marchandises en vrac (société MOINE). Un camion réalise un **voyage** lorsqu’on lui assigne un certain nombre de phases à accomplir (notion importante en cas de lot partiel puisqu’à tout moment la cargaison appartiendra à plusieurs clients et aura plusieurs destinations, on parle alors de groupage). Les informations du voyage sont dans la table VOYA. Les informations d’une **séquence de chargement et de livraison** (notamment les quantités chargées) vont être stockées dans diverses tables ayant pour préfixe SEQC.

Parmi les autres tables à citer, différentes tables contiennent respectivement les informations sur : les **camions** ; les **chauffeurs** ; les **clients** ; les adresses de chargement et déchargement fréquemment utilisées, appelés **points logistiques**; les différentes **tarifications** de ces transports ; les **coûts** de ces différentes étapes ; etc.

Les **factures** destinées au client regroupent un certain nombre de commandes. Lors de chaque livraison, le chauffeur va faire signer (émarger) un **bordereau de livraison** pour prouver que cette livraison a bien été effectuée et tous ces bordereaux doivent légalement être fournis avec les factures. Pour simplifier la gestion de ces bordereaux qui peuvent se retrouver partout en France, ils sont numérisés puis stockés dans la base de données dans un champ au format LONG RAW. Ils sont un des types de documents que contient la table DATABANK.

Le logiciel fournit également un certain nombre de vues effectuant certaines jointures utilisées fréquemment par les utilisateurs et le logiciel.

## 1.4) Sujets du stage

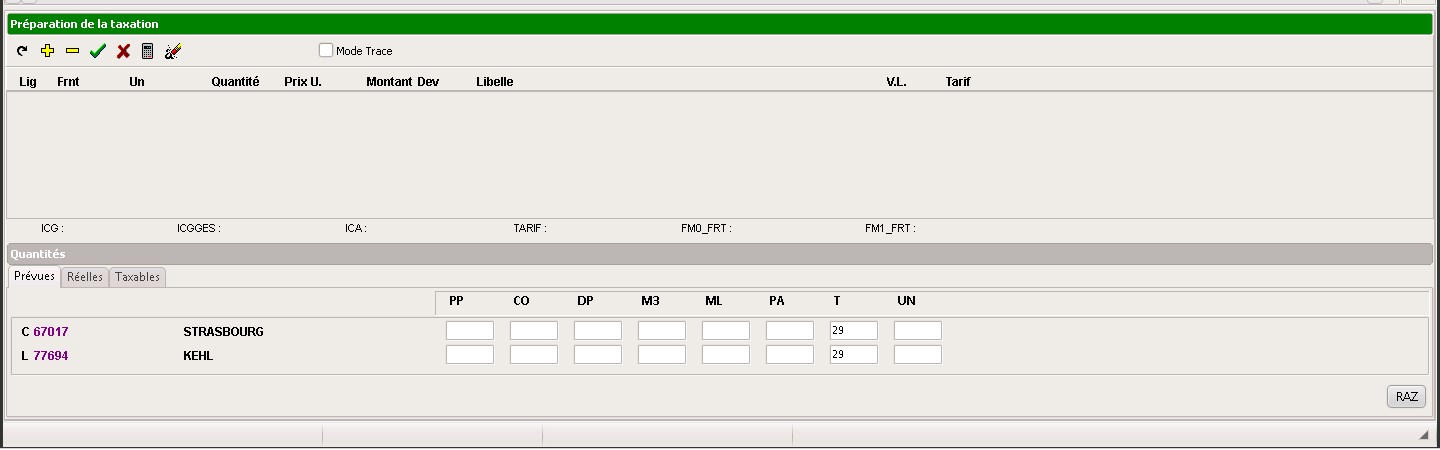
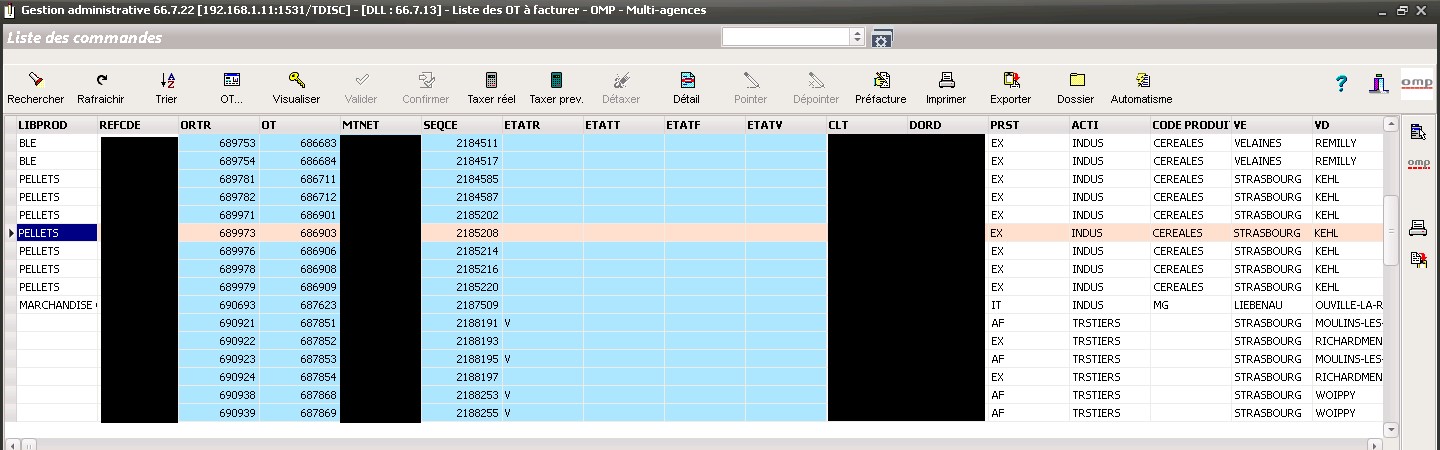
Les sujets qui ont constitué l’offre de stage sont les suivants :

* Réaliser l’automatisation d’impression des factures sur une période donnée.
* Réaliser une interface de création automatique de création de nouveau personnel dans la base de données.
* Réaliser divers reports demandés par différents services et sociétés.
* Mettre en place un serveur de supervision avec Centreon.
* Ajout de points logistiques spécifiques dans la base de données et dans le logiciel R2000X

Ces cinq grands axes constitueront les cinq grandes parties de ce rapport de stage.

## 1.5) Logiciels utilisés

### 1.5.1) OMP R2000X



*Capture d’écran d’une des facettes du progiciel R2000X. Cet écran liste et permet d’interagir et de compléter les valeurs des ordres de transports qui n’ont pas encore été facturés, donc qui sont encore ‘en cours’.*

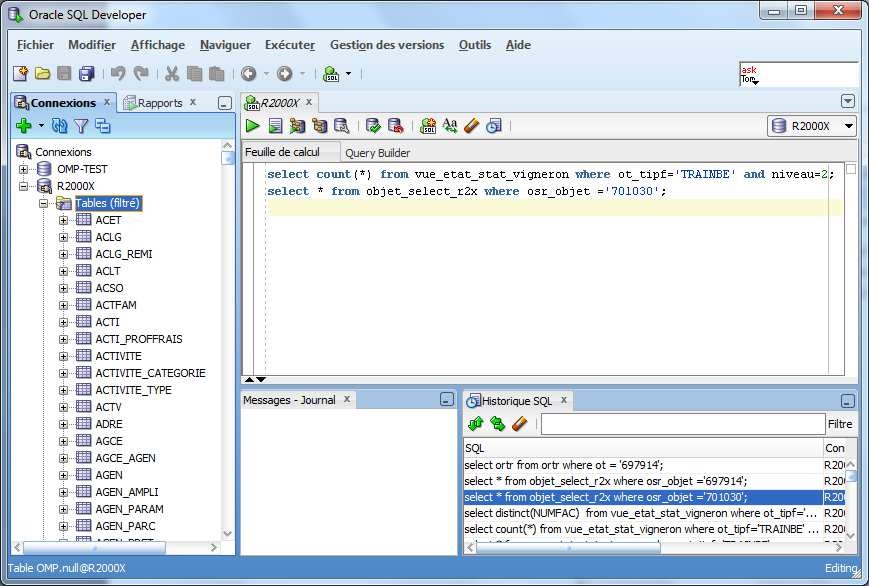
Le progiciel R2000X est un ERP destiné au secteur du transport développé par la société française OMP Informatique Transport. Un ERP est un progiciel qui permet de gérer l’ensemble des processus opérationnels d’une entreprise en intégrant plusieurs fonctions de gestion. Ce progiciel vaste est présent dans tous les aspects de l’entreprise, permettant de gérer le parc véhicule, les commandes, les transports, les tarifications, les ressources humaines, les clients, les factures, etc. Il s’appuie sur la base de données Oracle qui contient pratiquement toutes les données de fonctionnement du groupe VIGNERON. Il fonctionne également avec d’autres logiciels, appelant notamment Crystal Reports lorsqu’un report doit être imprimé pour une facture, etc.

Le progiciel étant particulièrement lourd et consommant beaucoup de bande passante, il est lancé sur deux serveurs TSE situés à Ludres. Tous les utilisateurs qui ne se situent pas sur le réseau local de Ludres se connectent par une session TSE en bureau à distance.

Le groupe VIGNERON est client de la société OMP pour ce progiciel. Un chef de projet de la société OMP a été assigné au groupe et se tient disponible pour les demandes de nouvelles fonctionnalités ou résolution de problèmes. Malheureusement, de nombreuses demandes reçoivent des réponses négatives ou des devis trop importants, ce qui force Mr Bonne à chercher à répondre à certains des besoins par ses propres moyens. Il a par ailleurs des contacts dans d’autres sociétés clientes d’OMP dans la même situation.

J’ai été amené à travailler avec les facettes du logiciel qui concernent l’impression des factures en masse, la gestion des congés, l’ajout de nouveaux points logistiques, l’ajout de personnel mais également lors de la réalisation de report (sujet exposé plus tard dans le dossier).

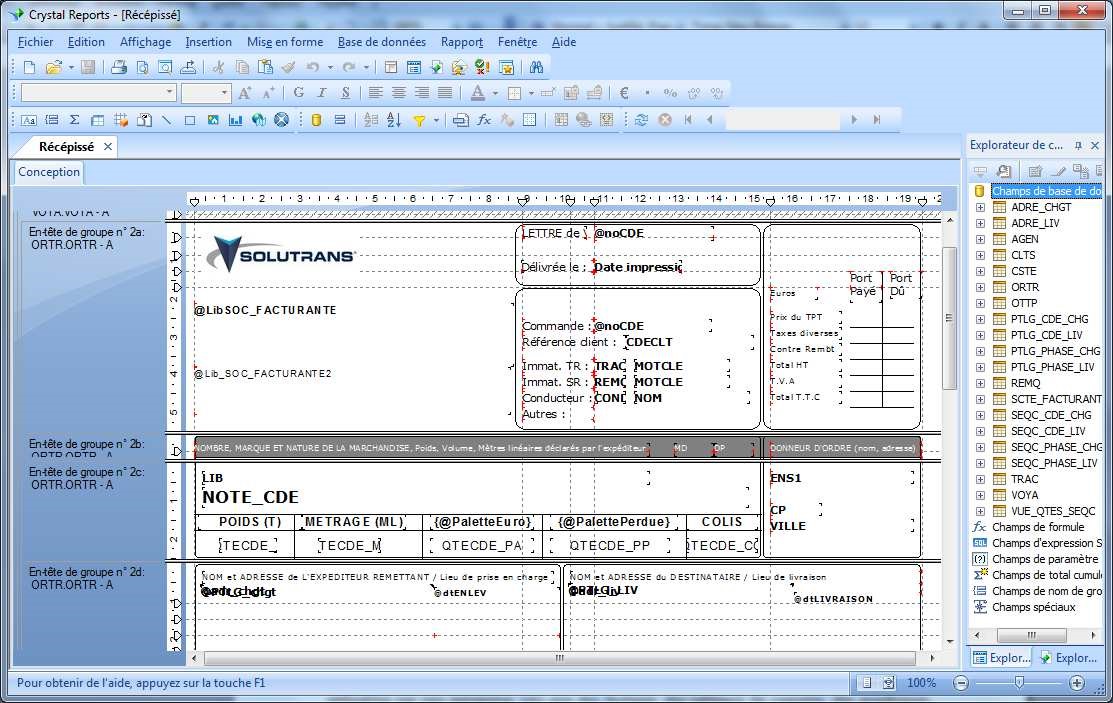
### 1.5.2) Oracle SQL Developer



*Capture d’écran du logiciel Oracle SQL Developer*

Oracle SQL Developer est un environnement de développement intégré (IDE) fourni par Oracle. Il permet d’interagir avec la base de données à l’aide de requêtes SQL. Ce logiciel a été un outil indispensable pour parcourir la base, tester les requêtes, vérifier les données contenues dans la base de données et développer et installer les différentes procédures stockées PL/SQL.

### 1.5.3) Crystal Reports



*Capture d’écran du logiciel SAP Crystal Reports.*

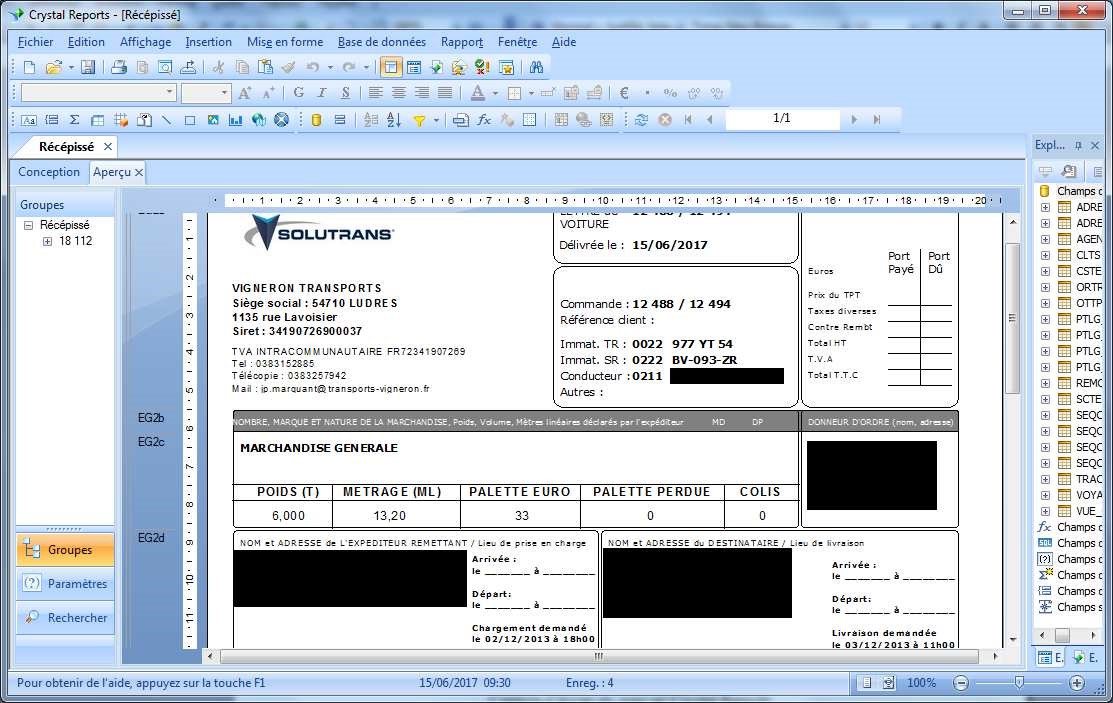
*On y voit un report dans l’écran de conception, avec les différents champs.*

Crystal Reports est un logiciel d’édition de rapports, qu’on appellera ‘reports’ (à prononcer à l’anglaise). Ce report permet de mettre en page des données, qu’on peut notamment extraire d’une base de données à l’aide d’une ou plusieurs requêtes SQL. Il permet donc, une fois le report conçu, de présenter ces données sous forme de documents finis utilisables par une entreprise, tels que des factures, des tableaux de comptes, des graphiques, etc.

Tous les bons de confirmation, les factures, les récépissés, ainsi qu’un certain nombre de tableaux statistiques ou comptables des sociétés du groupe VIGNERON sont actuellement des reports générés par Crystal Reports. Ils sont liés et appelés par le logiciel R2000X qui fournit à Crystal Reports les différents paramètres. Les clients peuvent réclamer leur facture à tel ou tel format, avec les données triées selon tel ou tel critère, fournissant ou masquant certaines informations, etc. Différents reports doivent donc être créés selon les types de documents et selon les besoins particuliers.

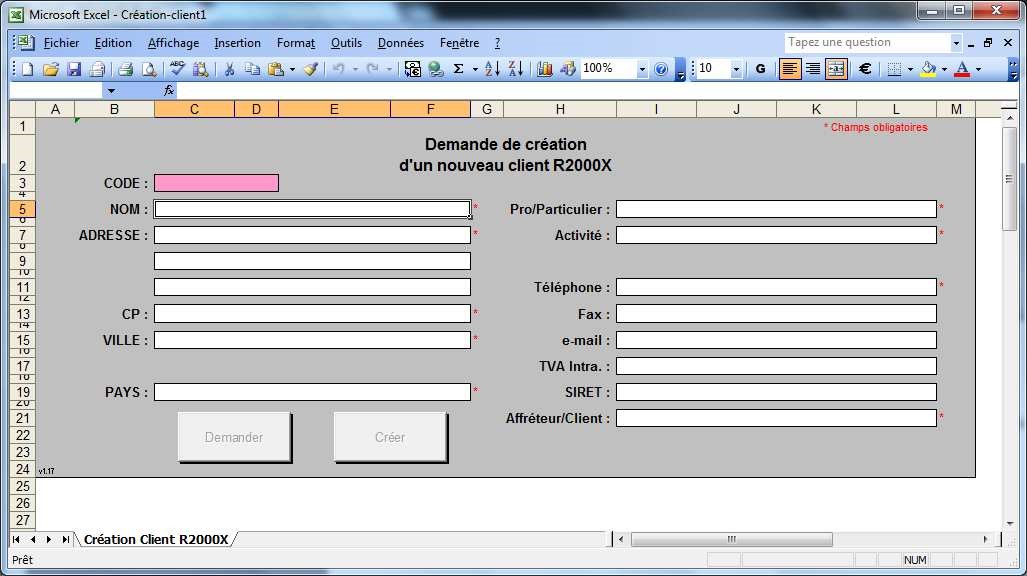
Après avoir défini la ou les sources des données du report, on peut disposer sur la page des zones de textes, les champs de données qui seront fournis par le logiciel, des zones de totaux, des images, des graphiques, etc., d’une façon comparable à Microsoft Publisher. On peut ensuite définir des formules qui sont exécutées selon les besoins avant, pendant ou après la lecture des données et qui permettent de calculer des valeurs, modifier la mise en page, masquer ou afficher certains champs, etc. Pour des reports complexes, on peut être également obligé d’inclure des sous-rapports.

La plupart des reports utilisés dans le groupe VIGNERON ont été réalisés par Mr Bonne. Etant très occupé et son travail fait qu’il est interrompu en permanence. Il n’a donc plus le temps requis pour créer les reports complexes demandés par certains utilisateurs.



*Le même report, dans l’écran d’aperçu, à qui on a fourni un paramètre et qui donne un récépissé prêt à imprimer.*

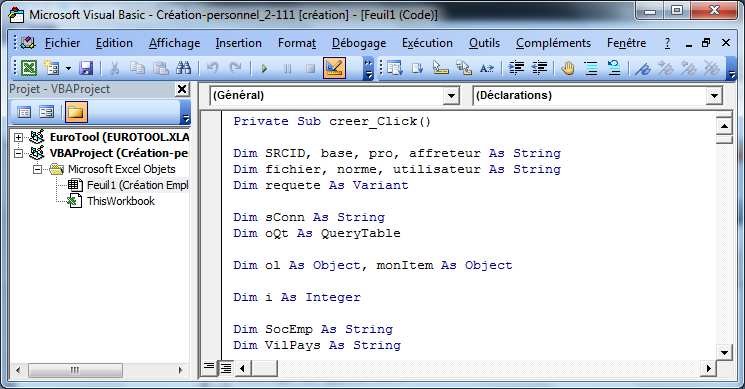
### 1.5.4) Microsoft Office Excel 2010



*Capture d’écran du formulaire de création d’un nouveau client réalisé par Mr Bonne dans Microsoft Office Excel 2010.*

Microsoft Office est la suite bureautique la plus utilisée à travers le monde. Elle propose des logiciels très utilisés en entreprise, comme Word, Excel, PowerPoint, Access, Outlook et Publisher. La suite est utilisée sur tous les ordinateurs du parc informatique du groupe VIGNERON car tous les utilisateurs se servent d’Outlook (logiciel de messagerie) et certains ont besoin de Word (traitement de texte) ou d’Excel.

Microsoft Excel est un tableur puissant et très polyvalent. Il est utilisé dans le groupe VIGNERON pour traiter et exploiter des données depuis la base de données Oracle, effectuer des calculs pour les services comptables, produire des graphiques et des statistiques.



*Capture d’écran de Visual Basic Editor, inclus dans Excel.*

Excel dispose également d’un langage de programmation, le VBA (Visual Basic for Applications), pouvant être utilisé pour faire des macros simples ou très complexes. On modifie les macros complexes dans Visual Basic Editor. Puisque Excel est forcément installé sur tous les ordinateurs du groupe, que le système de cellules est adapté pour créer des cases pour un formulaire, le langage permet d’envoyer des mails et d’émettre des requêtes SQL, Excel a été retenu pour la création du formulaire de création de nouveaux clients dans la base de données ainsi que pour la gestion des congés.

## 1.6) Remarques sur la base de données

La base de données étant utilisée par plusieurs sociétés ayant des activités, des contraintes et des façons de travailler différentes, il faut prendre en compte un nombre impressionnant de cas particuliers pour exploiter correctement les données. Voici quelques exemples de cas particuliers :

* Un nombre de palettes transporté n’est pas nécessairement un nombre entier.
* La valeur chargée peut ne pas être égale à la valeur déchargée.
* On ne peut pas faire totalement confiance à un prix ou un coût valant zéro, car l’utilisateur n’a peut-être pas eu connaissance de la valeur lorsqu’il l’a renseigné. Inversement, on ne peut pas ignorer les zéros car certains transports ont réellement un coût ou un prix de zéro euro.
* Il n’existe pas de valeur ‘coût total d’un ordre de transport’ dans la base de données, car le coût vaut la somme des coûts des phases impliquées. Du fait de la mutualisation des moyens (groupage), différentes interprétations et donc clés de répartitions des coûts (qui doivent être choisies par les responsables) peuvent s’appliquer. Ceci est de plus renforcé par la disparité et mixité des unités de mesure (tonnes, mètres, palettes…), des moyens (véhicule, personnel), des géographies et des contextes clients qui peuvent être panachés. De plus, lorsqu’une phase a un coût renseigné de zéro (comme vu à l’item précédent), la valeur totale est également remise en question.
* Certaines sociétés ont des coûts fixes, et veulent dans leurs statistiques ignorer tel ou tel coût car déjà comptabilisé ou considéré comme négligeable. Cela implique qu’un report de statistiques destiné à une société n’est pas applicable à une autre et doit être adapté.
* Les données sont éparpillées dans les tables et dépendent des jointures. Quand une valeur importante manque, des jointures peuvent ne pas se faire, et des pans entiers d’informations sont inaccessibles.
* Inversement, sans entrer dans le mode de fonctionnement de comptage des ordres de transports, on a parfois pour un seul numéro OT plusieurs records avec des numéros ORTR distincts, avec d’autres différences de données. Dans cette situation, un seul de ces records est valable. Lorsque les jointures sont effectuées sur le numéro OT, il faut toujours s’assurer de trouver et ne prendre en compte que le bon record.
* L’important volume de données implique un certain nombre d’erreurs de saisie.
* Certains utilisateurs ont fourni leurs bordereaux émargés en format JPG au lieu de PDF. D’autres ont joint dans le document PDF contenant leurs bordereaux de livraison émargés des photos en haute qualité d’une palette endommagée, malgré le fait que ces photos auraient dû être à part.

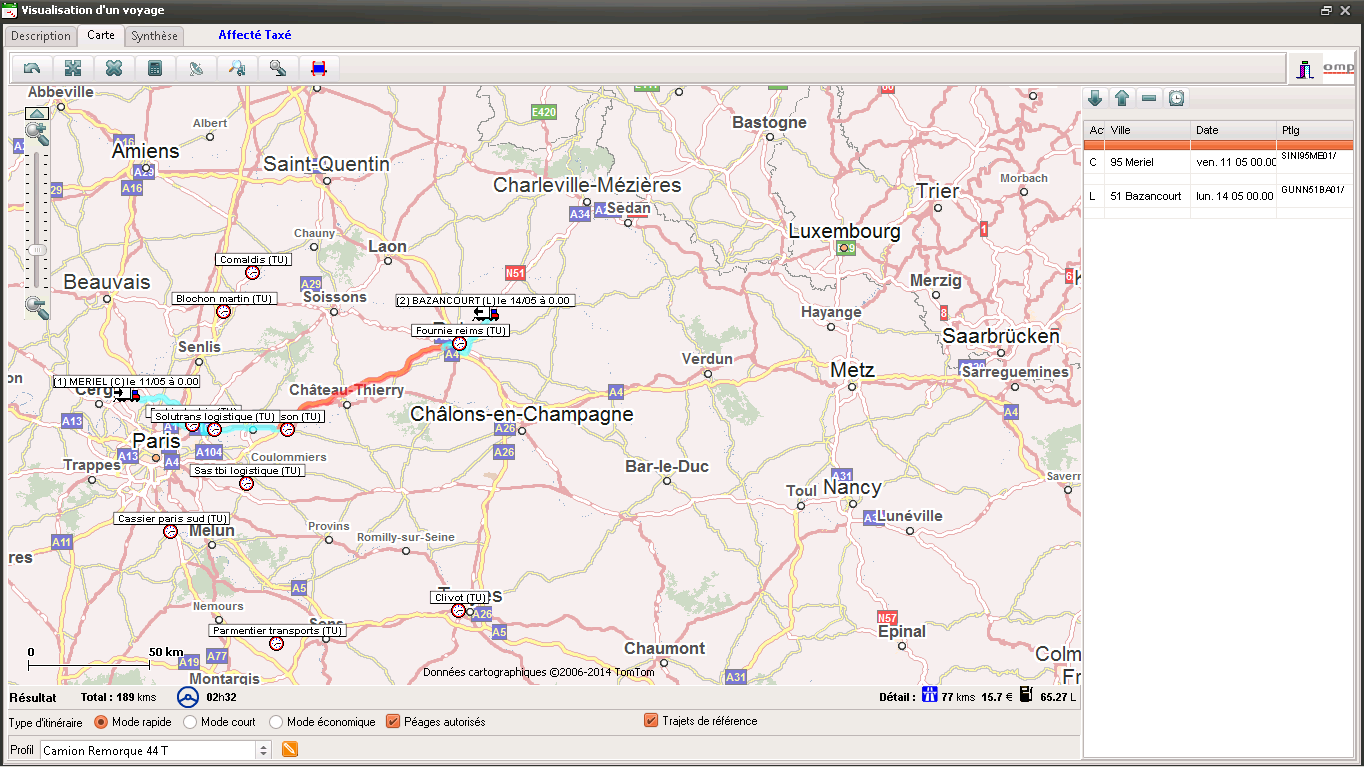
# 2) Ajout de points logistiques pertinents (Les points Tred Union TU)

## 2.1) Besoins du sujet

Dans l’ancien logiciel de transport utilisé par l’entreprise une fonctionnalité plus que pratique avait été mise en place par Mr Bonne. La visualisation de points appelés points Tred Union qui recensaient toutes les entreprises ou lieux appartenant au groupe de transporteur Tred Union. Lors du changement de logiciel pour OMP (R2000X) la fonctionnalité n’était pas implémentée par défaut. Cependant, le logiciel permettait l’affichage de points logistiques (des points d’intérêts) mais sous certaines conditions. Cette fonctionnalité était très utile pour les exploitants lors d’un souci de livraison pour déposer la remorque ou simplement chercher le partenaire Tred Union le plus proche d’un camion (géo localisable sur la carte).

## 2.2) Réalisation de ce sujet

Pour avoir l’affichage de ces points j’ai dans un premier temps renseigné les différents lieux et contacts associés dans la base de données par l’intermédiaire de l’interface du logiciel OMP à partir d’un listing fourni par Tred Union qui contenait près de 200 contacts à renseigner un à un dans la base de données. Une fois toutes les valeurs renseignées la seconde étape consistaient en l’installation du bouton pour exécuter la fonctionnalité. Je suis passé sur la vingtaines de PC présent au service des exploitants pour leur installer. Une fois installée il m’a fallu former les exploitants à l’utilisation cette nouvelle fonctionnalité.



*Aperçu des points Tred Union (« TU ») sur la carte. Ils sont représentés par des horloges. Les icônes de camions sont les positons des camions en temps réelle.*

## 2.3) Difficultés rencontrées

Plusieurs contraintes ont été rencontrées lors de la réalisation de cette tâche notamment lors du renseignement des données. Lors de la première itération, l’annuaire des contacts comportait près de 200 contacts. Chaque contact possédait une localisation, un nom, un prénom, et un voire deux numéro de téléphone. Il a fallu donc ajouter, si le point logistique existait, les coordonnées du contact et modifier son type (ici ‘TU’) en veillant à ne rentrer aucune valeur erronée et sans en oublier. Dans le cas où le lieu n’existait pas il fallait créer un nouveau point logistique (en respectant la nomenclature des identifiants dans la base de données), le localiser et renseigner les informations de l’annuaire. Une fois cette première itération réalisée, je l’ai présenté à un utilisateur potentiel et lors de la démonstration, ce dernier nous a signalé que certains points renseignés ne faisaient plus partie du groupe Tred Union. L’annuaire utilisé n’était pas à jour et Tred Union n’avait pas fait parvenir la dernière version au service informatique. Il a fallu donc revérifier tous les points renseignés et les modifier en fonction de ce nouvel annuaire. Soit environ 170 points.

Une troisième itération a été réalisée lorsque que je me suis rendu compte que les contacts indiqués dans l’annuaire n’étaient pas les exploitants de l’entreprise concernée mais leurs dirigeants. Mr Bonne a donc contacté Tred Union pour obtenir la bonne version à jour de l’annuaire. Cette fois environ 130 valeurs étaient à vérifier et à modifier.

## 2.4) Améliorations possibles

Le point relevé le plus par les utilisateurs est la représentation graphique du point logistique dans la carte. Sur l’ancien logiciel Mr Bonne avait modifié l’icône du point de sorte que ce soit le logo du groupe Tred Union. Cependant dans OMP nous n’avons pas la possibilité de changer l’apparence de l’icône. A l’heure actuelle, ce dernier ressemble à une horloge assez petite. Pas très pratique quand la carte se charge un peu.

Le second point est que la carte où l’on peut visualiser ces points n’est disponible uniquement lorsque l’on a sélectionné un transport planifié. Mr Bonne a donc contacté OMP pour avoir des renseignements vis-à-vis de ces améliorations et s’il est possible de les intégrer dans une future mise à jour.

# 3) Interface d’ajout de personnel dans la base de données

## 3.1) Besoin et sujet

Les sociétés du groupe VIGNERON sont souvent amenées, à cause de leur secteur d’activité, à engager du personnel. Lorsqu’une personne vient travailler ainsi, la procédure standard consiste à remplir un dossier écrit contenant un **formulaire de demande d’embauche** accompagné de toutes les pièces requises (photocopies de la carte d’identité, du permis de conduire, RIB, etc.). Ce dossier va être remis au Directeur des Ressources Humaines, prendra quelques jours pour être traité, et sera finalement remis à Mr Bonne au service informatique pour un ajout manuel dans la base de données. La personne n’apparaitra pas dans la base avant plusieurs jours. Or, à cause des contraintes du secteur du transport, il arrive par exemple que suite à un besoin immédiat un chauffeur doive être opérationnel immédiatement. À partir de ce moment-là, le fait qu’il ne soit pas dans la base de données est pénalisant car le chauffeur n’est pas géolocalisable par exemple. Dans ce genre de cas, les directeurs du service d’exploitation appelleront Mr Bonne par téléphone pour qu’il ajoute le chauffeur en urgence dans la base de données, court-circuitant le dossier papier.

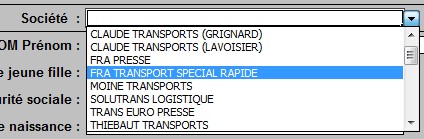
Les employés du service d’exploitation ont déjà à leur disposition un formulaire de demande de création d’un client. C’est Mr Bonne qui a réalisé ce formulaire dans un classeur Excel. Lorsqu’un employé est confronté à une société client qui semble ne pas être dans la liste, il doit remplir ce formulaire. Les macros, codées en VBA, contrôlent la cohérence des informations et si le formulaire est complet, les utilisateurs peuvent cliquer sur le bouton ‘Envoyer la demande’. Le classeur Excel va alors s’envoyer par mail aux trois responsables, ainsi que l’utilisateur à l’initiative de cette demande. Les responsables vérifient les informations, et vérifient que la société n’existe pas dans la base sous un nom différent (par exemple éviter d’ajouter une société SARL CORA si la base contient la société CORA SARL). Ces responsables ont accès à un bouton supplémentaire, qui ajoute de façon définitive le client dans la base de données et prévient par mail les autres responsables que le client a été ajouté.

Un des objectifs de mon stage était de reprendre ce formulaire de création client et de l’adapter pour la création d’un employé dans la base de données. Les contraintes sont en parties différentes car le but de ce formulaire est de créer directement l’employé et non pas de compter sur une vérification par un utilisateur intermédiaire. Egalement, il est difficile de supplanter le dossier papier qui est un document officiel signé et accompagné de toutes les pièces jointes requises.

## 3.2) Réalisation du formulaire

J’ai réutilisé une ancienne version du formulaire qui suivait le modèle du formulaire de création client déjà existant. Toutes les cases dans lesquelles l’utilisateur ne peut pas entrer de données sont protégées et ne peuvent pas être sélectionnées.

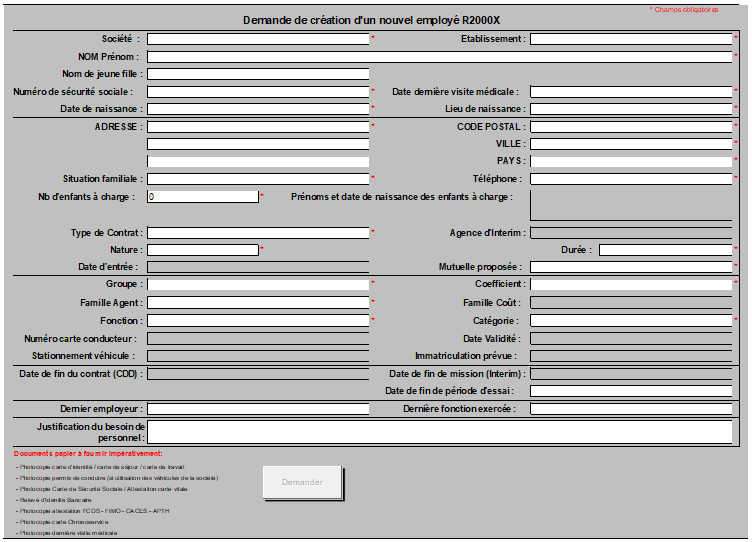
Certaines cases ne peuvent prendre que certaines valeurs (par exemple Société et Pays qui ont des listes déroulantes énumérant les options possibles, et il n’est pas possible d’apporter une valeur n’étant pas dans la liste).



*Capture d’écran de la liste déroulante de la case Société*

De même, certaines cases devant contenir des dates ou des nombres n’acceptent que des données du type correct. Entrer de mauvais caractères ou des dates impossibles renvoie directement un message d’erreur.

Si une case obligatoire n’est pas remplie, le formulaire prévient l’utilisateur que cette case est nécessaire pour faire valider le formulaire.

**

*Capture d’écran du formulaire de création du personnel dans Microsoft Excel*

## 3.3) Réalisation du code VBA

### 3.3.1) Contrôles de cohérence

La macro vérifie dès que l’utilisateur modifie une case que la valeur est conforme aux règles pour les cases liées à des champs de la base de données. Par exemple, un message d’erreur s’affichera si la valeur de la case Nom Prénom dépasse les 32 caractères, et la valeur s’affichera en rouge tant que le problème n’est pas corrigé. Le bouton Créer est désactivé tant que les données obligatoires ne sont pas toutes renseignées et correctes.

Certaines cases seront grisées selon les valeurs d’autres cases. Par exemple, tant que la case Nature ne vaut pas ‘CDD’, la case Date de fin du contrat (CDD) sera grisée et ne sera pas sélectionnable.

Lorsque l’utilisateur modifie une case, la macro va ensuite sélectionner la prochaine case qui n’est pas grisée dans l’ordre logique du formulaire.

### 3.3.2) Soumettre la demande de création de personnel

Lorsque le bouton « Demander » est activé et que l’utilisateur clique dessus, la macro va envoyer une requête SQL à la base de données pour intégrer les données dans la table de personnel ITFRH. Il enregistre également le formulaire complété comme un fichier Excel d’extension xls, puis il va envoyer un mail aux personnes concernées (à commencer par Mr Bonne) indiquant la demande de création d’un client, et fournira en fichier joint du mail le fichier xls.

# 4) Convertisseur d’export OMP vers logiciel de paie (SAGE) pour les congés

## 4.1) Besoin et sujet

Le logiciel R2000X de OMP permet de recenser les congés, les RTT et autres périodes de pauses des employés. Cependant le logiciel de paie n’est pas intégré à R2000X mais est un logiciel à part nommé SAGE Paie. Les codes des congés sont donc différents d’un logiciel à l’autre. Même si OMP permet une exportation des données dans des fichiers au format « .csv » et que SAGE permet l’importation de fichier Excel, la contrainte citée précédemment empêche de les effectuer.

Cette tâche était donc faite manuellement et cela faisait perdre énormément de temps avec des risques d’erreurs importants.

Le but de ce projet est donc d’interfacer l’export du logiciel OMP avec le logiciel SAGE pour ainsi automatiser la comptabilisation des congés dans la paie.

## 4.2) Réalisation du sujet

### 4.2.1) Analyse préliminaire

La première étape a consisté en l’analyse du besoin. Pour cela plusieurs ressources étaient à ma disposition. Un fichier étalon a été généré et fourni par Mr Bonne. Celui-ci contenait les différentes périodes de pause d’un chauffeur sur le mois de janvier. Un second fichier a été fourni. Ce dernier donnait la mise en page finale du fichier accepté par SAGE Paie (modèle). En somme après la transformation du fichier d’exemple par le programme, celui-ci devrait ressembler au modèle pour SAGE.

|  |  |
| --- | --- |
| C:\Users\Alexis\AppData\Local\Microsoft\Windows\INetCache\Content.Word\2.png | C:\Users\Alexis\AppData\Local\Microsoft\Windows\INetCache\Content.Word\1.png |

*À gauche le fichier initial (export de R2000X) à droite le fichier attendu (import SAGE Paie)*

*Note : le fichier d’import a sa première ligne vide. Ici des intitulés ont été rajouté manuellement pour avoir la correspondance des colonnes.*

Le langage de ce programme a été obligatoirement sélectionné car les fichiers traités sont tous supportés par Excel. Ce langage était donc le VBA.

### 4.2.2) Mise en page lors de l’exécution

L’objectif est donc d’obtenir par l’intermédiaire du code la mise en page attendue dans le modèle.

Lors de l’exportation des données par le logiciel OMP, celui-ci génère un fichier Excel comprenant 11 colonnes en tout.

Or le logiciel SAGE importe un fichier Excel qui ne doit comporter que sept colonnes. Cela signifie que des colonnes sont inutiles et que peut être d’autres colonnes non générées par R2000X sont nécessaires et dans quels cas elles doivent être générées par le programme.

Lors de l’analyse les colonnes en supplément et manquantes ont été déterminées.

Une autre particularité a été remarquée. Le fichier Excel dont SAGE a besoin doit avoir sa première ligne vide.

### 4.2.3) Interprétation des périodes de congés

L’interprétation des périodes de congés est le cœur du programme. Celui-ci doit analyser la date et le libellé de congé et déterminer si cette date fait partie d’une période de congé en cours ou si c’est tout simplement une nouvelle période.

La feuille à traiter étant remplie dans l’ordre chronologique (du début à la fin du mois) cela facilite l’étude des dates et périodes. Ainsi plus on avance dans la feuille plus on se rapproche de la fin du mois.

Même si cela facilite l’étude des périodes, elle n’en reste pas moins complexe à mettre en place. En effet les contraintes pour déterminer si une date fait partie d’une période ou non sont nombreuses.

Les Week-End (Samedi et dimanche) ne sont pas concernés. Il faut par conséquent ne pas en tenir compte. Si, par exemple, l’employé a une période de congé entrecoupé d’un Week-End (et qu’il soit toujours sous le même type de congé c’est-à-dire congé payé ou RTT) alors le programme doit l’interpréter comme une unique période et non deux périodes.

Le programme doit également être capable de détecter si la période de pause commence le matin ou l’après-midi et si elle finit le matin ou l’après-midi et cocher des cases en fonction des résultats.

Le libellé de congé généré par R2000X est présenté sous la forme : « soir » / « matin » / « après-midi ». Lors de la construction du code j’ai noté que lorsqu’une période de congé n’était effective qu’une seule matinée, le progiciel R2000X le génère sous la forme « soir » / « matin ». Par conséquent il manquait un « / » nécessaire lors de l’exécution du programme et le faisait planter.

Le programme doit également faire attention au changement de libellé en plein milieu d’une période. Par exemple un employé est en congé payé tout une semaine sauf le mercredi matin, le programme doit donc détecter le changement de libeller mettre un terme à la période en cours et en commencer une nouvelle.

Toutes ces contraintes ont nécessité de longues analyses et une très longue phase de test.

### 4.2.4) Traduction des codes R2000X vers les codes SAGE Paie

Le programme doit être capable de faire correspondre les codes congés de OMP avec les codes congés de SAGE Paie.

Le programme va donc détecter le code exporté par R2000X et le traduire en un code compréhensible par le logiciel SAGE.

## 4.3) Description de l’exécution du programme

Une fois l’exportation de R2000X effectué et le fichier à traiter ouvert dans Excel, l’utilisateur va exécuter le programme.

Il va, dans un premier temps, mettre en forme le document en supprimant les colonnes inutiles et en ajoutant les colonnes nécessaires pour SAGE.

Il va ensuite effectuer un premier passage dans les données pour rajouter les séparateurs « / » manquants afin d’éviter des erreurs et un plantage.

Le programme va ensuite effectuer un second passage et supprimer les Week-End, période non concernée par les congés.

C’est au troisième passage que le programme va étudier les codes congé de R2000X ainsi que les périodes.

Une fois qu’une période est commencée, le programme va vérifier quand cette période démarre dans la première journée (matin ou après-midi) et coche la case correspondante ou non. Il étudie ensuite la ligne suivante. Si elle appartient à la période en cours alors le programme la supprime. Une fois qu’il détecte que la ligne suivante ne fait plus partie de la période en cours, il va analyser quand, dans la dernière journée de la période, les congés se terminent et cocher si elles se finissent le matin ou l’après-midi. Une fois fait le programme passe à la ligne suivante et démarre une nouvelle période.

Une fois toutes les périodes détectées, le programme va effectuer un dernier passage pour traduire les codes congé de R2000X vers les codes congé de SAGE.

Une fois fait le programme enregistre le fichier sur le bureau de l’utilisateur dans un fichier nommé comme le fichier initial mais au format « xls » (importable dans SAGE)

## 4.4) Amélioration du programme

R2000X permet dans son exportation de mettre plus de 1 seul employé à la fois dans le fichier. Cependant le programme ne fonctionnait que pour 1 seul employé, c’est-à-dire qu’il ne distinguait pas plusieurs employés et donc leurs périodes respectives. Les périodes étaient donc calculées comme si seul employé était concerné.

Pour effectuer cette amélioration, un nouveau programme a été développé. Ce dernier va détecter les périodes par employé et pour chacun d’eux appeler le programme de base.

# 5) Impression de factures en masse

## 5.1) Besoin du sujet

Les entreprises du groupe Vigneron peuvent avoir besoin, pour des clients ou lors d’un contrôle fiscal, de fournir les factures associées à une entreprise sur des périodes plus ou moins longues. Le logiciel R2000X permet d’accéder à ces factures.

Lors d’une telle demande, la procédure consistait à sélectionner via R2000X la société concernée et à effectuer une requête sur la période souhaitée. Une fois la requête exécutée, les factures s’affichent sous forme de liste. Il fallait donc sélectionner la facture, la pré-visualiser pour ensuite lancer l’impression. Cela s’avère être un travail long et fastidieux surtout quand les périodes demandées peuvent aller jusqu’à plusieurs mois.

L’objectif est donc de réduire significativement le temps de traitement d’une demande similaire et que la procédure soit donc automatisée.

## 5.2) Analyse préliminaire

Le choix du langage pour programmer cette nouvelle fonctionnalité a été imposé par la structure du logiciel R2000X. En effet R2000X se repose sur une base de données qui contient toutes les informations sur la société y compris les factures. La fonctionnalité devait donc être codée en PL/SQL.

Cette fonctionnalité s’inspirant d’une autre existante, qui consistait dans l’impression des bordereaux de livraison, a déterminé les outils et utilitaires à utiliser dont la plupart étaient déjà en place.

Je devais donc, dans un premier temps, me concentrer uniquement sur le côté réception, traitement et transfert des données.

## 5.3) Réalisation

### 5.3.1) Première étape : Procédure VT\_ECRIRE\_PDF\_Facture

La première étape a été de coder la procédure VT\_ECRIRE\_PDF\_FACTURE qui extrait les fichiers de la base. J’ai créé une table de transition, et la procédure convertit le fichier LONG RAW en BLOB dans cette table. Le code va lire le contenu du BLOB dans un buffer, puis l’écrire dans un fichier sur le disque du serveur, à un emplacement défini.

Pour que ce code puisse fonctionner, il faut créer un répertoire SQL, c'est-à-dire définir dans la base de données Oracle un emplacement système. Ensuite, il faut s’assurer que l’utilisateur qui est destiné à utiliser la procédure dispose des autorisations en lecture et en écriture sur le répertoire. L’algorithme est inspiré de modèles sur Internet, mais appliqués dans notre cas les PDF étaient corrompus (il n’est pas impossible que le logiciel R2000X n’importe pas les données telles qu’elles devraient être selon le type LONG RAW) et il a fallu adapter jusqu’à réussir à obtenir les données correctement.

### 5.3.2) Deuxième étape : Script Bash d’impression

La deuxième étape était de pouvoir exécuter un script Bash depuis une procédure stockée. Pour cela, il faut créer un travail SQL (SQL Job.) Un travail est une tâche, qui peut être une procédure stockée ou un exécutable du système, et qui est lancée par la base de données régulièrement et ponctuellement sur demande. Dans notre cas, il est appelé par la procédure stockée.

J’ai récupéré ensuite le script Bash qui permet l’impression. Ce script lançait l’impression du fichier, avec SumatraPDF pour les PDF et Paint pour les JPG, sur l’imprimante fournie en paramètre. Une fois l’impression terminée, le script supprime le fichier.

Il est à noter qu’on m’a fixé comme contrainte d’utiliser SumatraPDF. Le logiciel a été choisi parce qu’il est adapté à l’utilisation en ligne de commande, supportant plusieurs paramètres d’impression nécessaires selon les utilisateurs (impression mise à l’échelle du papier, impression simple ou recto verso, noir et blanc ou couleur), et dont le processus se ferme automatiquement lorsque l’impression est terminée.

### 5.3.3) Troisième étape : Procédure VT\_EDITER\_PDF\_FACTURE

La troisième étape était de créer une procédure, que j’ai nommée VT\_EDITER\_PDF\_FACTURE, qui permet de lister les fichiers qu’il faut imprimer pour un numéro de facture. On passe en paramètre à la procédure le numéro de facture, l’adresse IP de l’utilisateur qui lance l’opération, le nom de l’imprimante sur le réseau et les paramètres d’impression désirés. Pour chaque facture, la procédure appelle VT\_ECRIRE\_PDF\_FACTURE pour récupérer, dans le dossier, le fichier à imprimer, et appelle un travail qui exécute le script Bash. Le souci d’utilisation d’un « travail » est que cela peut amener beaucoup de problèmes lors de l’exécution de la procédure. Par exemple, si l’on n’attribue pas au travail une durée de répétition automatique la plus grande possible (99 ans), ce dernier peut se désactiver ou générer des erreurs lors de son appel.

Pour que le travail puisse être exécuté, un service spécial (OracleJobScheduler) doit être lancé dans le système Windows du serveur par un compte disposant de suffisamment d’autorisations pour effectuer des opérations de lecture et d’écriture sur le disque, sinon le fichier ne sera jamais écrit sur le disque.

On a donc lancé OracleJobScheduler avec le compte administrateur. Mais c’est ensuite du compte administrateur que partait l’ordre d’impression et non pas l’utilisateur connecté. Les imprimantes, qui ont un système d’authentification, n’étaient pas configurées pour accepter l’administrateur. L’impression semblait se dérouler correctement du point de vue du système mais l’imprimante refusait l’impression à son niveau sans laisser de message d’erreur.

Si l’utilitaire d’impression SumatraPDF plantait et restait bloqué, les essais suivants de la procédure échouaient en renvoyant cette fois-ci un message d’erreur, mais qui indiquait que l’utilisateur ne disposait pas des autorisations requises pour effectuer cette tâche.

Lorsque ces problèmes ont été résolus, la procédure fonctionnait complètement lorsqu’elle était appelée dans SQL Developer par un administrateur de la base, mais pas avec un compte utilisateur. Il a fallu se renseigner, découvrir l’existence du droit CREATE

EXTERNAL JOB et l’attribuer aux utilisateurs.

### 5.3.4) Quatrième étape : VT\_EDITER\_PDF\_FACTURES et Liaison avec OMP

La quatrième étape consistait en la liaison de cette procédure à un bouton personnalisé dans l’interface R2000X. Il faut manuellement ajouter un nouveau bouton, choisir d’associer le bouton avec une procédure de la base de données, sélectionner la procédure VT\_EDITER\_PDF\_FACTURE, et lier les valeurs avec les différents paramètres.

Cette étape s’est révélée plus complexe que cela car la procédure apparaissait correctement dans la liste lorsqu’on paramétrait le bouton, mais lorsqu’on voulait utiliser la procédure, après la saisie des paramètres, le logiciel d’OMP n’exécutait la procédure que pour une seule facture.

Nous avons donc remarqué que peu importe les paramètres d’appel de la procédure, R2000X n’exécutait qu’un seul appel. Et n’imprimait donc qu’une seule facture.

J’ai donc remodifié les procédures, les types d’entrées mais rien ne changeait et une seule facture était imprimée.

J’ai donc écrit une nouvelle procédure nommée VT\_EDITER\_PDF\_FACTURES pour tenter de parer le souci du nombre d’appels des procédures par le logiciel R2000X. Cette nouvelle méthode récupérait les paramètres de VT\_EDITER\_PDF\_FACTURE mais prenait en plus les date de la période voulue. VT\_EDITER\_PDF\_FACTURES va, par l’intermédiaire d’un curseur, récupérer les factures et appeler VT\_EDITER\_PDF\_FACTURE.

Suite à des tests nous avons remarqué que l’impression des factures s’effectuait sans aucun souci lorsque la procédure était exécutée directement dans SQL Developer.

En attendant une solution de OMP, il avait été décidé que la procédure resterait une procédure administrateur. C’est-à-dire que lors d’une demande d’impression de facture en masse, Mr Bonne l’exécutera directement dans SQL Developer.

Pour aider Mr Bonne dans une éventuelle intégration de cette procédure dans une autre base de données ou tout simplement installer les utilitaires nécessaires à son fonctionnement sur l’ordinateur d’un utilisateur, j’ai été amené à écrire une petite documentation d’installation.

## 5.4) Résolution du soucis d’intégration dans R2000X

Lors de mon dernier jour de stage, Mr Bonn m’a demandé de vérifier et tenter de trouver une solution sur l’impossibilité d’exécution de la procédure VT\_EDITER\_PDF\_FACTURES dans R2000X. Grâce à des tables de débuggages et de nombreux tests, je suis parvenue a détecter l’anomalie.

En effet le format des dates pour la sélection de la période dans R2000X n’était pas bon. SQL Developer devait, lors de l’exécution dans son interface, interpréter la date ce que R2000X ne faisait pas. Le format utilisé à la base était « jj-mm-aaaa » alors que R2000X attendait « aaaa-jj-mm ». Une fois cette modification effectuée, la procédure était opérationnelle sur R2000X

# 6) Maintenance des serveurs

Chaque jour, Mr Bonne doit vérifier le bon fonctionnement du parc informatique du groupe Vigneron. S’il est inutile de vérifier chaque ordinateur un à un, il est plus intéressant de vérifier les serveurs, pilier des logiciels utilisés par le groupe.

Un problème se pose donc si Mr Bonne est absent et ne peut donc pas effectuer les vérifications et en cas de défaillance, résoudre les problèmes. Un membre du personnel s’occupant de l’informatique embarquée, Mr Hacquard, devra donc effectuer ces vérifications à sa place.

Mon travail a donc été de construire un manuel comportant les différentes tâches à effectuer sur la vingtaine de serveurs du groupe (serveurs de secondaires comme serveurs principaux) de manière à construire un manuel donnant les instructions et les tests à effectuer pour vérifier le bon fonctionnement de ces derniers.

Bien que le sujet soit simple et ne touche pas la programmation, cela n’empêche pas que c’est une tâche longue à réaliser et que la maintenance du parc informatique fait partie du travail du service informatique.

# 7) Mise en place d’un serveur de supervision avec Centreon

Ce sujet a été réalisé tout au long du stage. Aucune restriction particulière sauf vis-à-vis du logiciel à utiliser. Grâce à une étude préalable, Mr Bonne a sélectionné ce logiciel pour répondre à ce sujet. Ce sujet a pour but de faciliter la vérification du parc informatique (vu au paragraphe précédent).

## 7.1) Présentation de Centreon

Centreon est un logiciel de supervision informatique édité par la société du même nom. Il est basé sur le Centreon Engine qui est open Source. Il est accompagné de plugins lui permettant de communiquer avec différents types d’appareils sur le réseau (imprimante, machine Windows, machine linux, …). Certains plugins sont gratuits et open source tandis que d’autres sont payants.

Au travers du réseau et différents protocoles de communication (notamment SNMP), Centreon interroge la machine ciblée avec son IP afin de recueillir les informations qui ont été demandées lors de la configuration du périphérique à surveiller.

## 7.2) Besoins

Comme vu au chapitre précédent, Mr Bonne doit vérifier le bon fonctionnement de la vingtaine de serveurs du parc informatique du groupe Vigneron. Cependant cette tâche, même si elle n’est pas complexe, est longue et fastidieuse. Centreon a pour but d’automatiser cette vérification. En effet Centreon peut interroger les périphériques de telle sorte à effectuer les vérifications de base. Que ce soit l’espace disque, la quantité de mémoire vive, la quantité d’encre d’une imprimante ou la bonne exécution d’un service, Centreon permet une vérification en temps réel (actualisation des données toutes les 5 minutes mais paramétrable selon le besoin de l’utilisateur). Grâce à son système d’alerte, Centreon peut notifier (par mail, alarme, etc…) l’administrateur du système d’une défaillance. Il permettra donc à Mr Bonne de gagner beaucoup de temps lors de la vérification quotidienne.

## 7.3) Réalisation

### 7.3.1) Documentation et recherches

La première étape de ce sujet est de se documenter. En effet Centreon est un logiciel que je ne connaissais pas. Il a fallu différencier les versions et sélectionner le bon type (Il existait 2 types de Centreon : Centreon Central et Centreon Poller). Pour démarrer il fallait donc utiliser Centreon Central. Il fallait ensuite sélectionner sur quel support l’installer. Centreon n’est pas un simple logiciel mais un système d’exploitation basé sur Unix Red Hat. Par conséquent les méthodes d’installation passaient par des images disque « bootable » pour l’installer comme un système d’exploitation à part entière. Mais une version va retenir mon attention et Mr Bonne confirmera mon choix. J’ai donc choisis la version « machine virtuelle » de Centreon. La documentation de Centreon n’étant pas toujours à jour avec l’interface actuelle, j’ai favorisé la version anglaise de cette documentation qui est plus à jour que la version française.

### 7.3.2) Installation et premières difficultés.

La machine que j’utilisais pour mon stage était un PC portable sous Windows 7 32 bits. La particularité pour faire fonctionner une machine virtuelle sur son ordinateur et qu’il faut que son processeur supporte la Virtualisation de matériel. Ce qui n’était pas le cas de ma machine. J’ai donc utilisé une tour que Mr Bonne m’a fournie, qui était une ancienne machine d’un employé de Transports Vigneron.

Après le téléchargement des fichiers et utilitaires nécessaires pour la machine virtuelle, j’ai donc lancé l’importation de cette dernière dans le logiciel VirtualBox. Une fois l’importation terminée la machine Centreon se lance et survient le premier souci ; la machine virtuelle était protégée par un identifiant et un mot de passe non précisés dans la documentation. C’est après plusieurs heures de recherches sur les forums de Centreon que j’ai réussi à les trouver et donc à me connecter à la machine Centreon. L’interface de ce système d’exploitation est uniquement en ligne de commande.

Une fois démarré je décide de lancer les mises à jour des composantes de Centreon. Cependant aucune connexion internet n’était disponible. Le souci pouvait provenir d’une mauvaise configuration de l’interface internet de Centreon mais également à des plugins manquants de VirtualBox ou une mauvaise connexion entre la carte réseau virtuelle et la carte réseau physique de la machine. Après plusieurs jours de recherches et tentatives pour réparer cette défaillance je parviens à détecter le souci. L’interface réseaux de Centreon n’était pas définie. Par conséquent aucune connexion internet ne pouvait être trouvée. J’ai donc suivi un tutoriel en anglais permettant de résoudre ce souci et une fois ce tutoriel finalisé, la connexion internet était rétablie.

### 7.3.2) Configuration des premiers appareils

Après avoir lu attentivement la documentation et plusieurs essais non fructueux, je suis parvenu à mettre en place la surveillance de Centreon lui-même. Le principe est donc le même que pour tout autre dispositif que Centreon surveillera.

On définit dans un premier temps l’agent qui va surveiller le périphérique choisi. Soit une « Central » soit un « Poller » (les 2 versions de Centreon disponible). On doit donc le configurer et l’exporter de sorte à ce que le cœur de Centreon le reconnaisse.  
Une fois fait il faut déclarer le périphérique à surveiller. Pour cela Centreon fonctionne avec un système de configuration de base (template). Pour chaque type de périphérique (Imprimante, Linux, Windows, …) correspond un template qui prédéfinit ses paramètres. Il suffit donc de nommer le périphérique, fournir son adresse IP à Centreon et appliquer le bon template.

Une fois le périphérique choisi configuré correctement, Centreon le reconnaît mais aucune surveillance n’est encore appliquée. Pour cela il faut créer des « Services » (Surveillance de processeur, de processus, de la mémoire vive, …) et appliquer ce « Service » au périphérique. Les types de services à créer dépendent du template utilisé pour créer le périphérique à surveiller.

Par exemple un périphérique utilisant un « template » pour un ordinateur sous Linux pourra utiliser des services qui ne seront pas disponibles pour un périphérique utilisant un « template » pour surveiller une imprimante.

Les premiers appareils pour qui j’ai mis en place cette surveillance sont mon ordinateur de travail de stage (Windows), le Centreon lui-même (Linux Red Hat) et l’imprimante de mon étage. Ce dernier a remonté des alertes concernant l’encre et la quantité de papier avec le tiroir concerné.

### 7.3.3) Avantage et configuration futur

Au vu de mon choix pour la plateforme de Centreon (machine virtuelle), un avantage non négligeable se présente. Toutes les configurations que j’ai effectuées ne seront pas perdues et ne seront pas à refaire. En effet la machine virtuelle est contenue dans un fichier avec une extension de type « ova ». Etant donné que la machine sur laquelle j’ai configuré Centreon n’est pas la machine finale qui doit l’accueillir, il suffira de déplacer le fichier de la machine virtuelle vers une autre machine physique et de l’importer dans un logiciel permettant de lancer des machines virtuelles (comme VitualBox). Lors de la mise en place de la surveillance d’autres dispositifs du parc informatique, il ne suffira alors qu’à les déclarer dans Centreon.

# 8) Script de contrôle de sauvegarde

## 8.1) Besoins

Mr Bonne a mis en place un programme qui sert d’indicateur. Lorsqu’une sauvegarde a été effectuée, le programme crée un dossier SAVEJFB horodaté à la date du jour De cette manière on peut contrôler quand la dernière sauvegarde a été effectuée et dans le cas où elle échoue, enquêter sur les raisons de cet échec (employé en congé, ordinateur non démarré, …)

Pour effectuer cette vérification Mr Bonne détenait une liste des utilisateurs et, grâce à la fonction « rechercher », Windows listait tous les dossiers « SAVEJFB\_2018 » et permettait ainsi à Mr Bonne de déterminer depuis quand les sauvegardes d’un utilisateur étaient en échec.

Un travail assez long et une perte de temps non négligeable.

Mon objectif a été de créer un script en Batch (langage de commande pour Windows) qui permet d’effectuer cette vérification et de sortir un fichier de log recensant les utilisateurs qui n’ont pas de sauvegarde du tout ou qui ont une sauvegarde ancienne (par rapport à la date du jour actuel).

## 8.2) Réalisation et fonctionnement

La réalisation a été assez complexe car le langage Batch m’était inconnu (utilisant essentiellement le Bash Linux). C’est donc grâce à des forums et des adaptations de codes que le code a été construit.

Le script va donc parcourir les différents dossiers principaux des utilisateurs (nommé comme ceci : prénom.NOM).

Pour chaque utilisateur, le script va vérifier si le dossier « SAVEJFB\_<date actuel> existe ». Dans le cas où il existe (et qu’il est bien daté) rien ne se passe, le script va étudier l’arborescence de l’utilisateur suivant. Mais dans le cas où il n’existe pas ou que la date de sauvegarde ne correspond pas à la date actuelle, le script crée un fichier texte sur le bureau de l’ordinateur où le script est exécuté et inscrit le nom du dossier où il manque le dossier de sauvegarde.

Le script devait suivre une certaine arborescence précise des dossiers (prénom.Nom > Mes Documents > SAVEJFB > SAVEJFB\_<date de sauvegarde>) cependant pour éviter tout risque de plantage, à cause du script (boucle infinie, …), sur le server j’ai reproduit une arborescence des dossiers similaire sur mon pc de stage pour faire le développement et les tests nécessaires pour le script.

## 8.3) Les différentes versions

Plusieurs versions de ce script ont été faites. Une version n’écrivait pas dans un fichier texte mais affichait directement dans la console avec une petite mise en page pour faciliter la lecture. Mais Mr Bonne préférait obtenir un fichier texte.

La version retenue par Mr Bonne a également subi une nouvelle version car nous nous sommes rendu compte que lors de l’exécution du script, si l’on ne supprimait pas le fichier de log existant, le script écrivait à la suite des résultats de l’ancienne exécution. Cela donnait donc une fausse liste de dossier manquant. J’ai donc corrigé l’erreur en purgeant le fichier texte à chaque exécution du script.

# 9) Création ou modification de reports

Tout au long de mon stage, ponctuellement, j’ai été amené à réaliser ou modifier des reports avec Crystal Reports

## 9.1) Récépissé depuis OT et Récépissé par voyage

Pour me familiariser avec Crystal Reports, Mr Bonne m’a chargé de petites modifications mineures. Pour cela il a demandé aux employés, par mail, s’ils avaient détecté des bugs sur les différents reports qu’ils utilisent au quotidien. Mon premier report à modifier était donc le report générant le récépissé d’un voyage.

Les récépissés peuvent être générés de deux façons différentes par le logiciel

R2000X : par le numéro d’ordre de transport ou par le numéro de voyage. Il y a donc deux reports distincts pour générer ce document.

### 9.1.1) Première modification : informations cachées

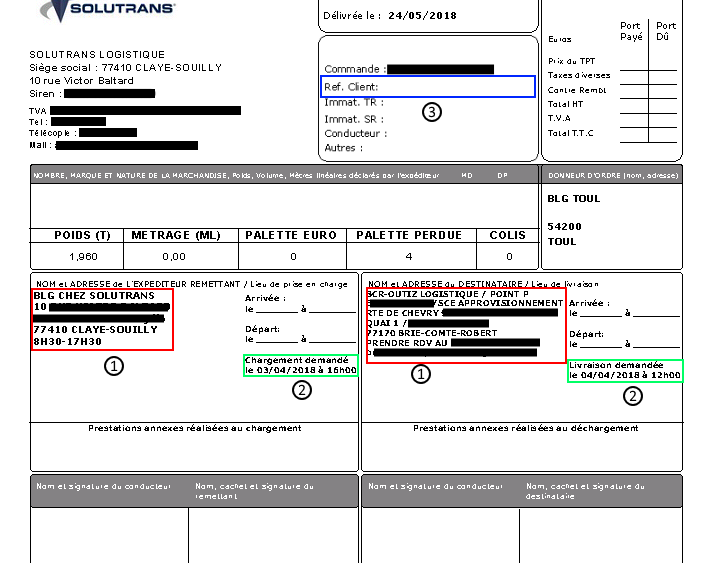
Lors de la génération d’un récépissé, ce dernier n’affichait pas entièrement les adresses lorsque ces dernières étaient trop grandes. Certaines informations étaient par conséquent cachées (comme le numéro de téléphone par exemple). J’ai donc programmé la mise en page dynamique et contextuelle des deux emplacements où les adresses se situent. Ainsi si les différentes informations (adresses, numéro, …) dépassent les 25 caractères, la taille de la police est réduite. Ce qui résout le problème d’information cachées. (Voir 1 sur l’image ci-dessous)

### 9.1.2) Seconde modification : dates erronées et ajouts d’informations

Plus tard dans le stage, les utilisateurs de ce report ont fait remonter des incohérences avec les dates de chargement et de livraison présentes dans les encadrés des adresses.

Pour résoudre ce souci j’ai modifié les données que Crystal Reports récupère lors de la génération. Crystal Reports récupérait les données de la mauvaise table. J’ai donc rectifié cette erreur et les dates de chargements et de livraisons sont redevenues cohérentes. (Voir 2 sur l’image ci-dessous)

Les utilisateurs en ont également profité pour demander l’ajout d’une information qui leur est utile. J’ai donc ajouté une rubrique « Référence Client » sur le report. (Voir 3 sur l’image ci-dessous)



## 9.2) Bordereau de groupage

Ce report regroupe tous les transports d’un même affrété sur une journée. Il lui permet de connaitre ce qu’il possède dans sa remorque et permet au centre de réception de l’affrété de s’organiser pour récupérer et livrer au plus vite.

### 9.2.1) Première modification : montant du transport

La première modification que j’ai effectuée sur ce report concernait le montant du transport de l’affrété du voyage. En effet des incohérences et des erreurs sur ces champs avaient été signalées à Mr Bonne. La valeur récupérée par le report n’était la bonne. Le champ à récupérer faisait partie d’une autre table mais avait le même nom de code (MTTRAF).

Une fois cette erreur rectifiée, les valeurs du report étaient redevenues cohérentes.

### 9.2.1) Deuxième modification : ajout d’informations

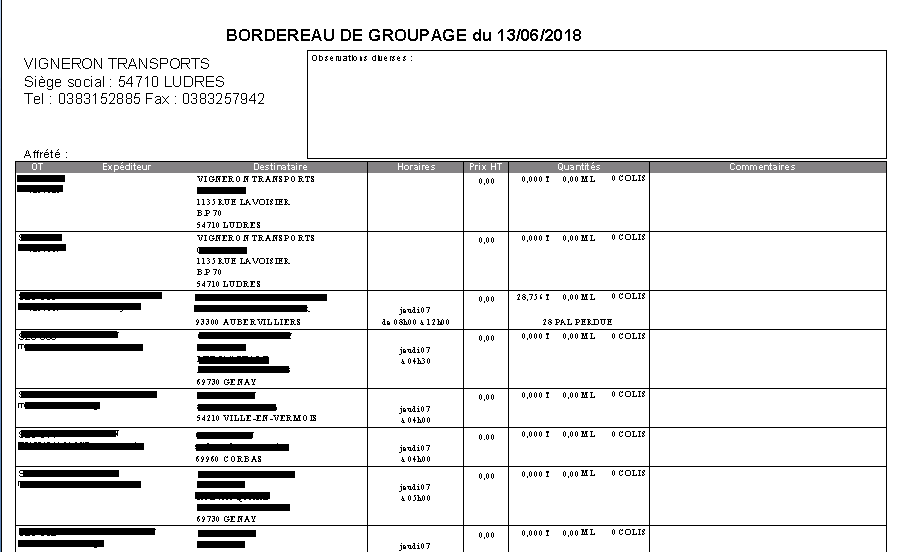
Le report récupérait des informations de contacts (adresse, numéro fixe, …) via les champs associés dans la base de données. Cependant les utilisateurs lors d’ajout de nouveau client dans la base de données, lorsque plusieurs numéros étaient à renseigner, ils choisissaient d’ajouter le numéro de mobile dans le champ du numéro de fax. OR ce champ n’apparaissait pas dans le report car il n’était pas récupéré.

J’ai donc ajouté l’affichage du numéro de fax pour que les utilisateurs puissent avoir pleinement accès à toutes les coordonnées des clients.

### 9.2.1) Troisième modification : suppression du point logistique Garonor pour Claude

Le but de cette modification était de supprimer l’affichage des différentes informations lorsque l’ordre de transport venait de Claude et qu’il avait pour destination Garonor car il s’agissait enfaite d’un trafic de retour.

Pour cela j’ai ajouté des conditions de suppression, comme le permet Crystal Reports, lors de sa génération pour atteindre l’objectif de cette modification.

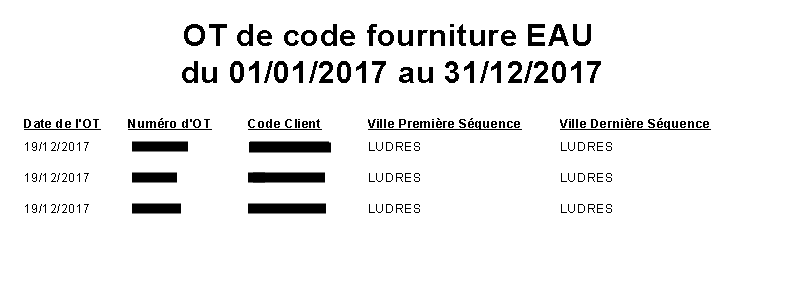


## 9.3) Fourniture Date

Le service facturation de Transports Vigneron a demandé à Mr Bonne de réaliser un report. Ce report a pour but d’afficher les différents ordres de transports facturés en fonction d’un code fourniture et d’une période saisie par l’utilisateur.

J’ai donc réalisé ce report entièrement. Les éléments à afficher ainsi que d’autres paramètres (ordre de tri, …) ont été choisie par le service facturation.

Ce report permet de gagner du temps aux employés du services facturation lorsqu’ils doivent rechercher les ordres de transport associés à une fourniture spécifique.



## 9.4) Majoration Gazole

Il arrive que Thiebaut Transport facture à Transports Vigneron un ordre de transport car, Transports Vigneron a, par exemple, affrété le transport à Thiebaut Transport.

Thiebaut Transport facture donc à Vigneron la prestation. Cependant en fonction des clients de cet ordre de transport, la majoration gazole ajoutée n’est pas forcément la même (la majoration gazole pour un transport de matériels n’est pas la même que pour un transport de lait).

Mr Bonne avait donc mis en place un report permettant de récupérer et calculer le coût moyen de la majoration gazole d’un ordre de transport (OT) de Thiebaut Transport. Cependant des erreurs avaient été repéré car le report ne tenait pas compte des dates et donc de la mise à jour des valeurs de la majoration gazole des différents donneurs d’ordres.

Mon travail a donc été de modifier ce report de sorte à ce que les valeurs soient cohérentes, de rectifier également la valeur moyenne de la majoration qui était du coup faussée lors de son calcul. Je devais également ajouter une condition périodique, c’est-à-dire que le report, après que l’utilisateur ait sélectionné une période, doit récupérer les ordres de transport que Transports Vigneron lui a affrété et ceux-ci dans la période voulue.

La difficulté réside dans le fait que les valeurs de ces majorations changent régulièrement. D’un point de vue base de données, ces valeurs (quand elles n’étaient plus à jour) se situaient dans une table d’historisation (TRHS), différente de la table principale qui, elle, garde les valeurs à jour (TRPX). Il fallait donc pour chaque donneur d’ordre de l’ordre de transport vérifier que sa valeur de majoration gazole était toujours d’actualité par rapport à la période demandée. C’est-à-dire qu’il ne suffisait pas de récupérer la dernière valeur de la majoration gazole mais de récupérer la valeur valide pendant la période entrée par l’utilisateur.

Ce report s’est avéré être très complexe car il fallait saisir parfaitement les conditions de récupération des valeurs pour obtenir les bons résultats.

Une fois les valeurs générées dans le report, il m’était possible de les vérifier via R2000X pour contrôler leurs cohérences.



## 9.5) Relevé kilométrique

L’objectif de ce report est d’obtenir un suivi kilométrique (relevés par l’informatique embarquée) de tous les tracteurs de la société Vigneron. Chaque mois le responsable du parc des tracteurs doit dresser un suivi kilométrique des tracteurs, pour y parvenir il utilise une base de données gérée par le logiciel Access de Microsoft. Seulement des anomalies ont été détectées. Pour faciliter le travail du responsable du parc, Mr Bonne m’a demandé de réaliser un report listant tous les camions du parc et y associer le dernier kilométrage avec la dernière date de mise à jour.

Dans sa première version le report ne devait récupérer que le kilométrage des camions qui avaient roulé pendant le mois demandé (donc passé en paramètre). Mais au vu du nombre de données présentes dans les tables (plus de dix millions), Mr Bonne m’a demandé de rajouter une restriction périodique. Les données traitées n’étaient donc que des données qui dataient du 1er janvier 2018 ou plus.

Dans une seconde version, le report devait sélectionner les camions qui étaient encore actifs et donc non sortis du parc.

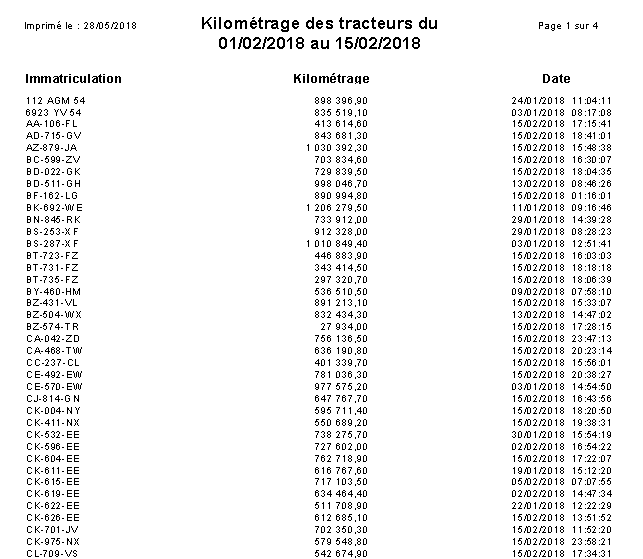
Le souci de cette version est que lors de l’essai sur l’ordinateur du responsable du parc, même après avoir attendu une vingtaine de minutes, le report ne s’était toujours pas généré. Nous avons donc décidé de stopper l’exécution et mon travail a été d’améliorer ce temps d’exécution.

Pendant l’attente le responsable du parc nous a exposé ses idées pour améliorer le report.

La troisième version devait donc être plus rapide pour son exécution et devait charger le dernier kilométrage des camions et ceux avec une antériorité de 4 mois. C’est-à-dire que si un camion qui est encore actif dans le parc n’a pas roulé depuis moins de 4 mois, le report doit récupérer ses informations. Au-delà de cette période il doit être ignoré. Dans ce cas le camion est affiché mais aucune information kilométrique n’y est liée.

Après beaucoup de temps et d’essais (les temps d’exécution pouvaient aller jusqu’à une trentaine de minutes) le report ne mettait plus qu’une dizaine de minutes à se générer.

Une fois générée tout une procédure de vérification des données faites par moi-même était effectuée via SQL Developer et des requête SQL basique pour vérifier que ses données étaient correctes (une fois encore les temps d’exécution étaient très longs).

**

# Conclusion

Le stage s’est révélé être très instructif et enrichissant. Il m’a permis de découvrir le monde de l’entreprise et le secteur du transport et de la logistique, que je ne connaissais pas particulièrement. Il m’a permis d’être confronté aux contraintes du développement pour répondre à des besoins concrets (modifications du cahier des charges, etc…), faire fonctionner des projets dans des cas où on ne maîtrise pas forcément chaque engrenage. Il m’a également permis de voir le métier de technicien de maintenance et de chef de service informatique qu’exerce Mr Bonne. C’est un métier comportant beaucoup de responsabilités et qui occupe beaucoup de temps.

La recherche, la documentation et le débogage ont constitué la plus grande partie du stage. L’utilisation du VBA et du logiciel Crystal Reports n’ont pas été vu lors de mes cours et je les ai apprises pendant le développement des projets concernés. De nombreuses difficultés se sont manifestées mais des aides étaient présentes sur Internet. Dans les cas où des aides étaient introuvable, il a fallu improviser et réussir à trouver et adapter des solutions pour répondre aux besoins.

J’ai apprécié avoir utilisé de nouveau logiciel tel que Centreon ou Crystal Reports. L’expérience acquise sur ces logiciels me sera très utile vu que Crystal Reports est un logiciel très utilisé en entreprise et que Centreon est un logiciel très utile pour surveiller un très grand nombre de machines simplement.

J’ai également découvert toute une partie du logiciel Excel (dont j’ignorais qu’une telle programmation était possible avec ce dernier). C’est donc encore une fois une nouvelle expérience qui me sera sans doute très utile.

Il n’y a pas eu de problèmes au niveau du déroulement du stage. Je me suis très bien entendu avec Mr Bonne. Les autres employés ont été très gentils et ce fut très agréable de travailler avec eux pendant ces 10 semaines.

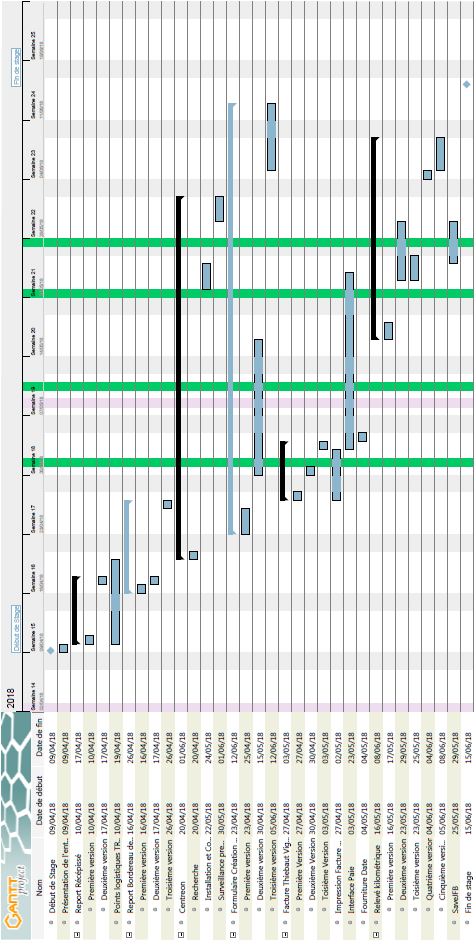
Tous les sujets de base ont été réalisés. Les reports ont été modifiés avec succès. L’impression des factures en masse, mis à part le souci d’intégration dans R2000X, est pleinement fonctionnelle et utilisable sur la base de données de production.

Les points logistiques sont installés et fonctionnels. Et la plupart des employés ont été formés à son utilisation.

Le serveur de supervision est fonctionnel et prêt à l’emploi même s’il n’est pas installé sur la machine définitive.

Quant à l’interface de création de personnel, plusieurs versions ont été proposées aux employés concernés. Des ajouts et suppression de champs ont été effectués. Le formulaire est fonctionnel mais doit encore attendre une validation.

J’ai également réalisé des sujets qui n’étaient pas prévus à la base comme l’interface entre R2000X et SAGE Paie pour les congés ou encore la notice pour la maintenance des serveurs ainsi que le script pour vérifier que les sauvegardes ont bien été effectuées.



**Bibliographie**

**Documentation**

SQL.SH (2017), Cours et tutoriels sur le langage SQL.

http://sql.sh

ORACLE HELP CENTER (2017), Database Documentation.

https://docs.oracle.com/en/database/

TECHONTHENET.COM (2017), Oracle / PLSQL Functions.

https://www.techonthenet.com/oracle/functions/index.php

Fleur-Anne BLAIN (2007), Initiation à Crystal Reports XI.

http://fablain.developpez.com/tutoriel/crystal-report/

SAP.DEVELOPPEZ.COM (2014), FAQ Crystal Reports.

https://sap.developpez.com/faq/crystal-reports

MICROSOFT (2005), Crystal Syntax.

https://msdn.microsoft.com/en-us/library/ms225310(v=vs.80).aspx

OMP Informatique Transport (2015), Rapport Modèle Physique de Données R2000X.

Documentation fournie avec la base de données, 1 016 pages.

Site Web Centreon

https://www.centreon.com/

**Forums**

StackOverflow.com .

http://stackoverflow.com

SAP, Forum officiel de Crystal Reports.

https://archive.sap.com/discussions/space/crystal-reports

Forum Developpez.net .

https://www.developpez.net/forums/

Forum Experts Exchange.

https://www.experts-exchange.com/topics/crystal-reports/