



Stage ESIAL 2^{ème} année Rapport

CHRISTOPHE BENOIT
ESIAL 2^{ème} ANNEE
8, rue Joseph Kessel
54420 Saulxures-lès-Nancy
Tel : 06.09.83.96.94

BSI SA
Parc de l'Ohmbach
20, rue des Anémones
68280 Westhalten
Tel : 03.89.78.09.50



Stage ESIAL 2^{ème} année **Rapport**

« Déploiement d'une application client/serveur en
une application trois tiers, sur Internet. »

Responsable de stage : Mr Girod Pascal

Année Universitaire :
2001-2002

ESIAL
Domaine Scientifique Victor Grignard
B.P. 239
54506 VANDOEUVRE-LES-NANCY CEDEX
Tel : 03 83 91 25 94

Avant-Propos

Mon stage s'est déroulé du 1^{er} juillet au 31 août 2002, au sein de l'entreprise BSI, une société de services informatiques spécialisée sur le marché des PMI/PME.

Je tiens tout d'abord à remercier particulièrement Mr Benoît Sturny, président directeur général de BSI SA, pour m'avoir accueilli dans son entreprise. Je remercie également Mr Hédi Dardour, directeur général de BSI SA. Je tiens particulièrement à remercier Mr Pascal Girod, mon maître de stage, pour toute l'aide qu'il aura pu m'apporter. Je remercie également le pôle Oracle, de m'avoir bien accueilli pendant toute la durée de mon stage. Enfin, je tiens à remercier toute l'équipe de BSI SA, qui là encore a été très accueillante avec moi.

L'objectif du stage de deuxième année consiste d'une part, à se familiariser avec le travail en entreprise et à découvrir les processus de gestion du travail des ingénieurs en informatique. D'autre part, il s'agit de mettre en pratique les compétences techniques acquises plus particulièrement lors de notre deuxième année d'enseignement, de les confronter et de les adapter aux impératifs de temps et de moyens de l'entreprise d'accueil.

De début Juillet à fin Août 2002, j'ai travaillé au déploiement sur Internet d'une application développée avec les logiciels Oracle. Après avoir étudié les techniques permettant ce déploiement, j'ai analysé les logiciels qui pouvaient le réaliser. J'ai ensuite mis en place tous ces logiciels, avant de passer à la phase de déploiement proprement dite.

Table des Matières

AVANT-PROPOS	3
TABLE DES MATIERES	4
TABLE DES ILLUSTRATIONS.....	5
INTRODUCTION.....	6
1 L'ENTREPRISE BSI.....	7
1.1 LA SOCIETE BSI ET SA FILIALE MIRE.....	7
1.1.1 Le pôle Alsacien : pôle de développement	7
1.1.1.1 Intégration de solutions informatiques.....	7
1.1.1.2 Formation des clients	7
1.1.1.3 Mise à jour et assistance produits	8
1.1.2 Le pôle Nancéien : pôle commercial	8
1.1.3 La filiale MIRE.....	8
1.2 ACTIONNARIAT DU GROUPE.....	9
1.2.1 Actionnariat de BSI	9
1.2.2 Actionnariat de MIRE.....	9
1.3 L'EVOLUTION EN CHIFFRES DE BSI ET DE MIRE	10
1.3.1 Evolution du Chiffre d' Affaires de BSI.....	10
1.3.2 Evolution du Chiffre d' Affaires de MIRE	11
1.3.3 Evolution du personnel dans le groupe.....	11
1.4 ORGANIGRAMME DU POLE DE DEVELOPPEMENT	12
2 ANALYSE DES BESOINS DE L'ENTREPRISE.....	14
2.1 EXPOSE DU PROBLEME.....	14
2.2 LOGICIELS INSTALLES.....	16
2.2.1 Installation d'Oracle8i Database Release 3	17
2.2.2 Mise à niveau d'Oracle8i Database Rel 3.....	17
2.2.3 Installation d'Oracle9i Application Server	18
2.2.4 Installation de Forms & Report Builder 6i	18
2.3 LE DEPLOIEMENT DE L'APPLICATION.....	19
2.3.1 Rapatriement des données	19
2.3.2 Migration des programmes récupérés.....	19
2.3.3 Paramétrage du serveur.....	20
2.3.4 Paramétrage des icônes.....	21
2.3.5 Paramétrage des éditions	21
2.3.6 Paramétrage des touches de fonctions	22
2.3.7 Définition d'un profil d'utilisateur Internet	23
2.3.8 Résolution des bugs	24
2.3.8.1 Bugs liés aux versions des produits	25
2.3.8.1.1 Problème lié aux « Combo Box ».....	25
2.3.8.1.2 Problème lié aux nombre d'utilisateurs	25
2.3.8.2 Bugs propre à la version Internet	26
2.3.8.2.1 Problème lié aux « Polices ».....	26
2.3.8.2.2 Problème de fonctions non supportées	26
2.3.8.2.3 Problème de fonctions non admises	26
2.3.8.2.4 Problème de langues.....	27
2.3.9 Résultats du déploiement et remarques.....	27
CONCLUSION.....	28

ANNEXES.....	29
I. PROCEDURE LIANT OUTILS ET BASE DE DONNEES.....	30
II. PROCEDURE DE MISE A JOUR D'ORACLE8I DATABASE.....	31
i. Installation du patch.....	31
ii. Lancement des scripts.....	31
III. MIGRATION DES PROGRAMMES.....	34
i. Scripts de recompilation.....	34
ii. Paramétrage des icônes.....	35
IV. PARAMETRAGE SPECIFIQUE DU SERVEUR.....	35
i. Paramétrage de la méthode « servlet ».....	35
ii. Paramétrage spécifique à l'application.....	37
iii. Paramétrage des éditions.....	40
iv. Ajout de valeurs dans la base de registres de Windows.....	41
v. Paramétrage des touches de fonctions.....	42
V. CAPTURES D'ECRAN DE L'APPLICATION CERES SUR INTERNET.....	44
VI. CONFIGURATION MATERIELLE ET PRIX D'UNE MACHINE SUN.....	49

Table des Illustrations

FIGURE 1 : ACTIONNARIAT DU GROUPE BSI.....	9
FIGURE 2 : ACTIONNARIAT DE LA SOCIETE MIRE.....	10
FIGURE 3 : CHIFFRE D'AFFAIRES DE LA SOCIETE BSI.....	10
FIGURE 4 : CHIFFRE D'AFFAIRES DE LA SOCIETE MIRE.....	11
FIGURE 5 : ARCHITECTURE CLIENT/SERVEUR.....	14
FIGURE 6 : ARCHITECTURE TROIS TIERS.....	15
FIGURE 7 : ECRAN D'ACCUEIL DU NAVIGATEUR AU LANCEMENT DE L'APPLICATION.....	44
FIGURE 8 : FENETRE COMMANDANT L'OUVERTURE DE L'APPLET.....	44
FIGURE 9 : ECRAN D'ACCUEIL DE L'APPLICATION ET DE CONNEXION DE L'UTILISATEUR INTERNET.....	45
FIGURE 10 : ECRAN DE CONNEXION RENSEIGNANT SUR LA SOCIETE DE L'UTILISATEUR.....	45
FIGURE 11 : ECRAN D'ACCUEIL DE L'APPLICATION CERES.....	46
FIGURE 12 : EXEMPLE DE GRILLE AVEC VUE RESTREINTE POUR L'INTERNAUTE.....	46
FIGURE 13 : EXEMPLE DE GRILLE QUE L'UTILISATEUR INTERNET VERRA VIA L'APPLET.....	47
FIGURE 14 : EXEMPLE DE DOCUMENT PDF ENVOYE AU CLIENT.....	48

Introduction

Le but de mon stage dans l'entreprise BSI était de mettre en œuvre la méthode pour qu'une application de type client/serveur, puisse être mise sur Internet.

Ainsi, plusieurs étapes se sont dessinées, lors de mon stage. Il a tout d'abord fallu cerner le sujet, afin de bien comprendre les attentes de l'entreprise. Cela a principalement consisté à comprendre l'architecture trois tiers.

Ensuite, il a fallu se plonger dans la documentation des produits Oracle, afin de savoir quels produits pouvaient répondre aux besoins de l'entreprise. De plus, il me fallait savoir quelles étaient les configurations requises tant au niveau software¹ qu'au niveau hardware² pour installer ces logiciels.

Puis, il a fallu s'intéresser au développement qui avait déjà été entrepris. En effet, il m'a fallu savoir comment les « forms »³ avaient été générées, afin de comprendre la méthode de travail qui avait été mise en œuvre. Je me suis ensuite intéressé au fonctionnement du logiciel, permettant de déployer les « forms » pour les rendre opérationnelles sur Internet.

Enfin, j'ai recompilé ces « forms » en mode Internet et je les ai testées. En effet, une vaste pléiade de tests, fut nécessaire au bon fonctionnement de l'application. Parmi les principaux problèmes à résoudre, on peut citer les problèmes de montée en charge⁴ ou les problèmes liés au proxy⁵ notamment les problèmes de firewall⁶.

¹ Niveau Software : niveau logiciel, ce que l'ordinateur doit posséder comme logiciel(s).

² Niveau Hardware : matériel physique, ce que l'ordinateur doit posséder comme configuration minimum.

³ Forms : Les « forms » sont générées par programmation ; c'est par exemple l'interface que voit l'utilisateur devant son terminal.

⁴ Montée de charge : Terme signifiant une augmentation des connexions au site

⁵ Proxy : Un serveur proxy est une machine intermédiaire entre les ordinateurs d'un réseau local et le web. Le proxy peut servir de firewall

⁶ Firewall : littéralement : Mur de feu, barrière protégeant les données. Système filtrant les informations en ne laissant passer que les ports choisis pour des raisons de sécurité.

1 L'entreprise BSI

Présente en Alsace Lorraine depuis Mars 1980, l'entreprise BSI compte aujourd'hui une vingtaine de collaborateurs. Le siège social de l'entreprise BSI est établi à Westhalten, dans le vignoble Alsacien. Son agence commerciale est établie à Nancy. Enfin, une troisième société est reliée au groupe. Il s'agit de la société MIRE, spécialiste dans les installations réseaux.

1.1 La société BSI et sa filiale MIRE

Comme on l'a dit précédemment, l'entreprise BSI est divisée en trois pôles distincts, notamment au niveau de l'activité pratiquée. Ainsi, concernant la gestion du personnel, l'activité de chaque équipe est coordonnée par un chef de projet; c'est avec lui, et en fonction des besoins des clients et des priorités qu'est déterminé le planning des différents membres de l'équipe. Ensuite il y a un suivi hebdomadaire de l'état d'avancement. L'activité est suivie avec un logiciel "maison" de suivi de projets.

1.1.1 Le pôle Alsacien : pôle de développement

Il s'agit du pôle de développement et de maintenance informatique. Le site Alsacien travaille sur le développement de nouveaux logiciels ainsi que sur la mise à jour des logiciels utilisés par les entreprises clientes. On trouve dans ce pôle le fondateur de la société BSI, le président directeur général M. Benoît Sturny. Une de ses fonctions consiste à trouver de nouveaux clients au groupe.

1.1.1.1 Intégration de solutions informatiques

L'entreprise BSI permet d'intégrer des solutions informatiques dans une entreprise. Elle doit assurer la mise en place de ces solutions. Elle définit donc un planning avec ses consultants, pour optimiser l'organisation, le paramétrage et la mise en œuvre des logiciels BSI. L'entreprise BSI assure également l'équipement en machines informatiques, des sociétés clientes.

1.1.1.2 Formation des clients

L'entreprise BSI est une société de services distribuant des logiciels informatiques. C'est donc

également une entreprise assurant la formation des personnes qui utiliseront les logiciels. Cette formation est là encore assurée par des développeurs du pôle Alsacien. Elle a lieu, soit dans les locaux de BSI sur des postes dédiés à la formation des clients, soit dans les locaux du client sur des postes destinés à la production.

1.1.1.3 Mise à jour et assistance produits

Le produit phare de BSI est l'application CERES, sur laquelle j'ai pu travailler. Cette application est un ERP complet prenant en charge toute l'activité des coopératives et négociants céréaliers. Cela impose donc d'avoir une application qui soit mise à jour continuellement. En effet, lors de l'apparition de nouvelles normes ou lors du passage à l'euro... ; des modifications dans les programmes s'imposent. L'entreprise BSI a donc développé des contrats permettant des prestations personnalisées. En effet, tous les types d'agriculteurs peuvent utiliser CERES. Donc, selon un secteur ou un autre, les modifications ne seront pas les mêmes.

L'entreprise BSI travaille essentiellement avec la télémaintenance. En effet, elle permet de se connecter directement chez le client pour effectuer la mise à jour des programmes ou pour corriger d'éventuelles saisies erronées faites par les entreprises clientes.

1.1.2 Le pôle Nancéien : pôle commercial

L'entreprise BSI possède un pôle chargé uniquement de l'activité commerciale du groupe. Ce pôle se situe à Villers-lès-Nancy. Dans ce pôle, se trouve le directeur général M. Hédi Dardour.

1.1.3 La filiale MIRE

Enfin, l'entreprise BSI travaille avec un troisième pôle, il s'agit de la société MIRE. MIRE est une société spécialisée dans les études et la réalisation de câblages d'immeubles (TCP/IP, Wireless⁷...). MIRE et BSI sont deux sociétés indépendantes.

⁷ Wireless : Il s'agit d'un réseau sans fil. De plus en plus d'ordinateurs sont équipés d'une carte intégrée, permettant d'avoir un accès réseau sans fil.

Néanmoins, elles cohabitent dans les mêmes locaux à Villers-lès-Nancy et l'actionnariat de MIRE (voir page 10) explique la grande synergie existant entre les deux sociétés. En effet, BSI est la centrale d'achat de MIRE ainsi que son centre de gestion. L'activité de MIRE est totalement complémentaire à celle de BSI, notamment lors de l'intégration de solutions informatiques dans une entreprise.

1.2 Actionnariat du groupe

Le groupe BSI + MIRE est partagé entre plusieurs actionnaires.

1.2.1 Actionnariat de BSI

BSI est une SA au capital social de 300000 €. Il y a 7 actionnaires dans le capital de BSI, répartis comme suit :

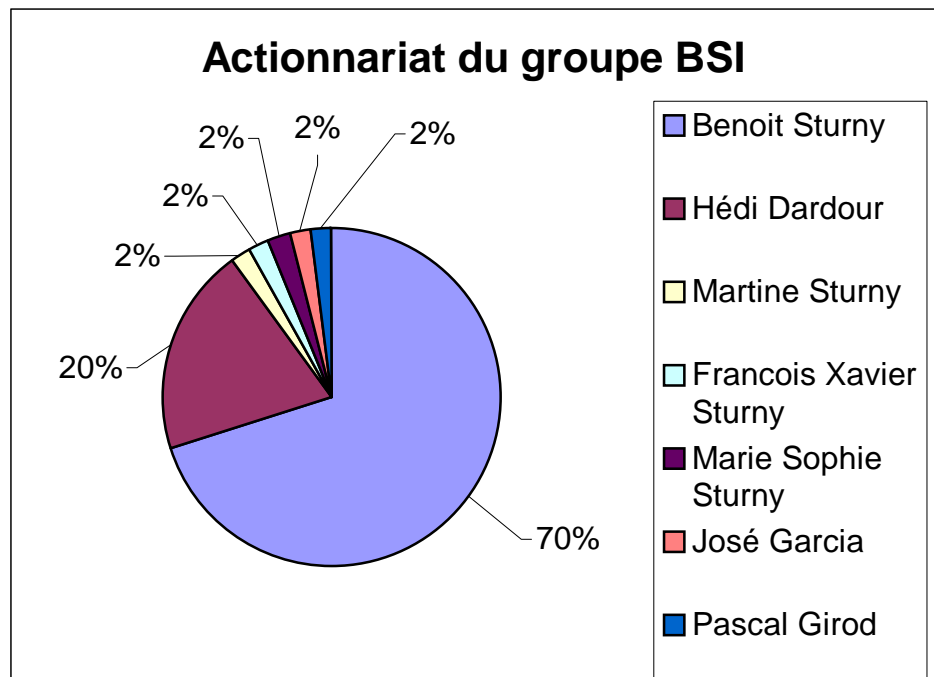


Figure 1 : Actionnariat du groupe BSI

1.2.2 Actionnariat de MIRE

MIRE et BSI travaillent en complémentarité. Comme on va le voir, l'actionnariat de MIRE permet cette entente entre les deux entreprises.

Par ailleurs, MIRE est une société au capital social de 90000 €. Ses actionnaires sont au nombre de trois, comme suit :

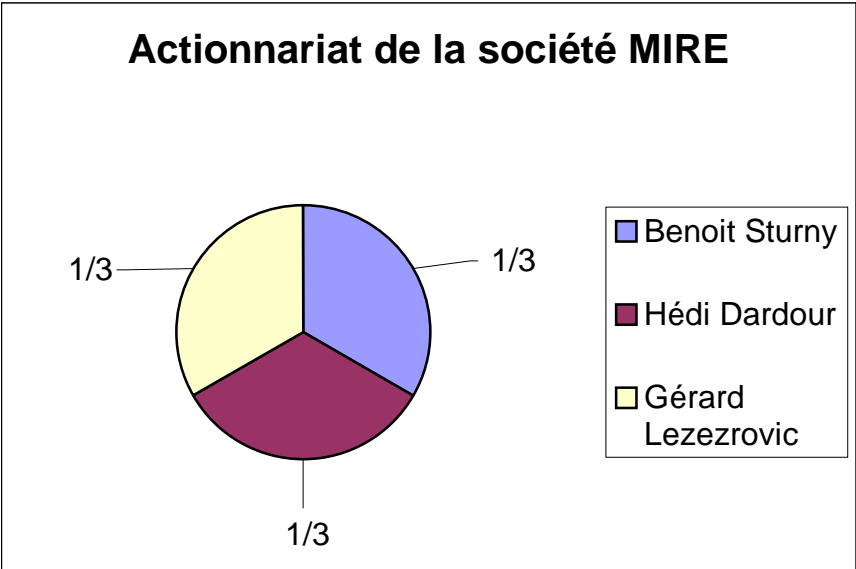


Figure 2 : Actionnariat de la société MIRE

1.3 L'évolution en chiffres de BSI et de MIRE

On ne présente ici que l'évolution de l'entreprise sur les quatre dernières années, tant au niveau du chiffre d'affaires qu'au niveau de l'évolution de l'effectif.

1.3.1 Evolution du Chiffre d'Affaires de BSI

L'exercice débute au 1^{er} octobre et s'arrête le 30 septembre.

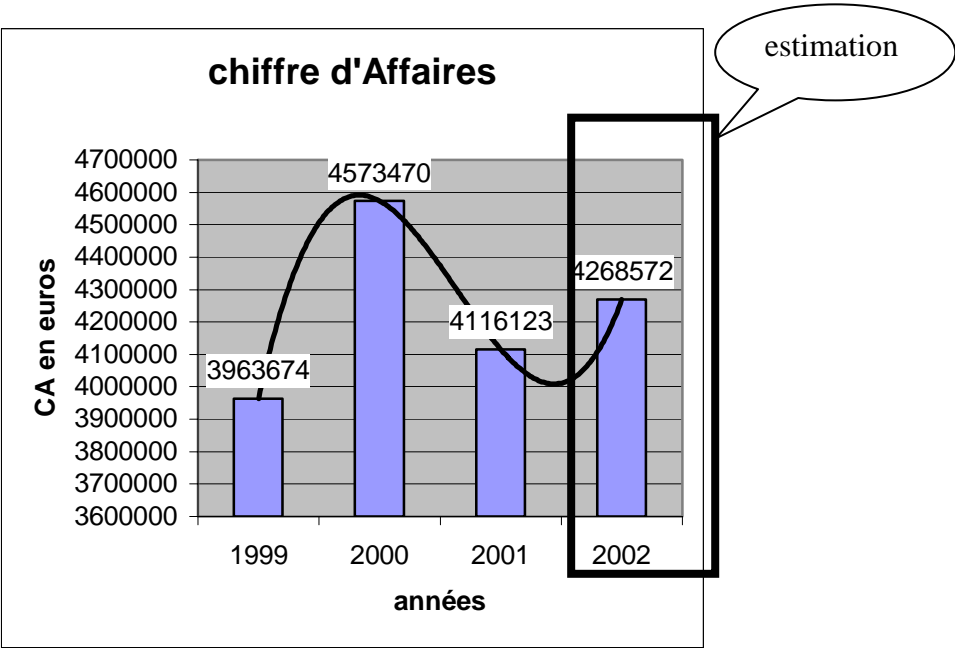


Figure 3 : Chiffre d'Affaires de la société BSI

La forte hausse du chiffre d'affaires de l'année 2000 est en partie due au remplacement prématuré du matériel, par crainte du bug de l'an 2000.

Concernant la place sur le marché de l'application CERES, aucune étude n'a à ce jour été menée. En effet, la connaissance de la part de marché de CERES ne présente pas d'intérêt dans la stratégie de l'entreprise BSI. Le potentiel de mise en place d'une nouvelle version par an constitue toujours des dossiers très lourds à mettre en oeuvre.

1.3.2 Evolution du Chiffre d'Affaires de MIRE

L'exercice débute au 1^{er} juillet et s'arrête au 30 juin.

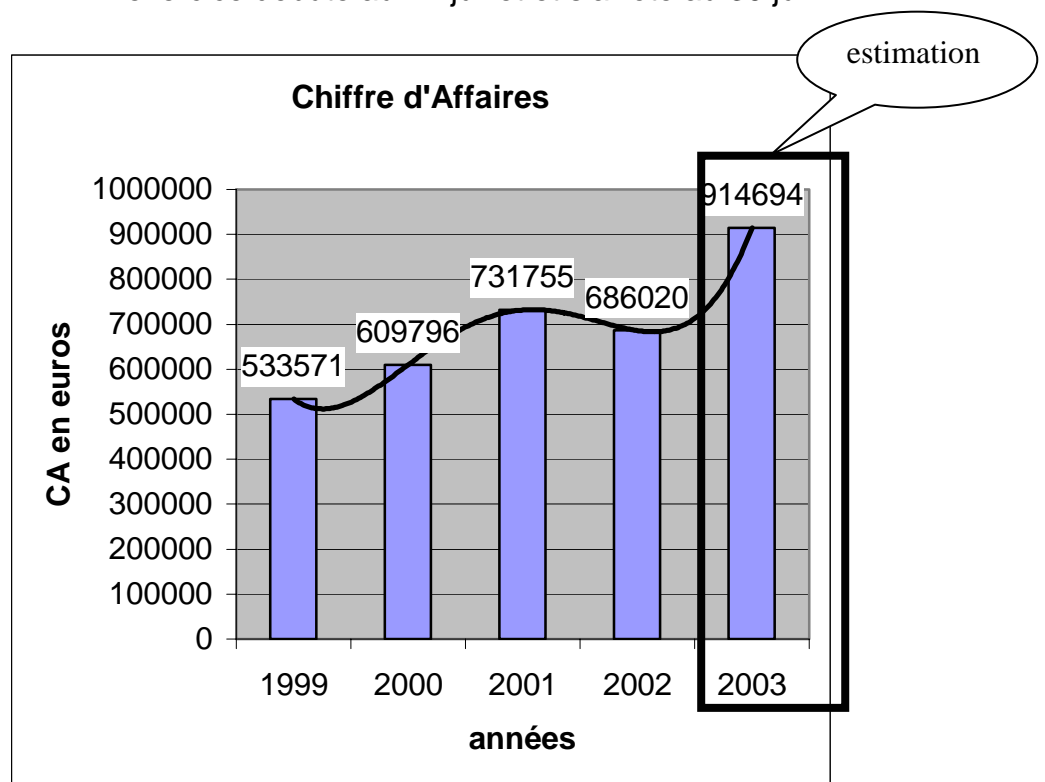


Figure 4 : Chiffre d'Affaires de la société MIRE

La baisse du chiffre d'affaires en 2002, s'explique par un gros investissement en formation sur les techniques du wireless.

1.3.3 Evolution du personnel dans le groupe

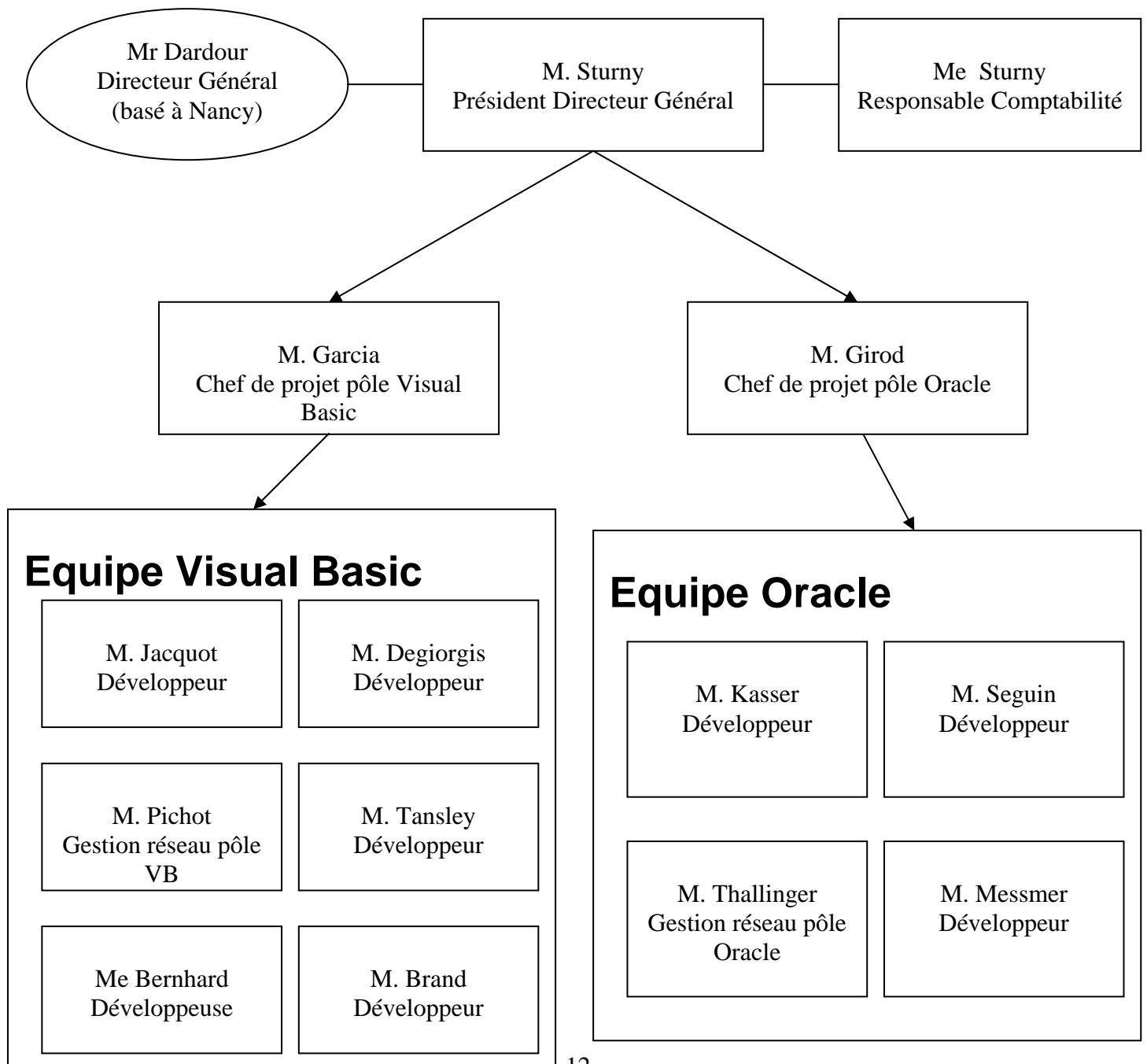
Depuis quatre ans, le nombre de collaborateurs dans la société BSI est resté constant (une vingtaine). Dans la société MIRE, le nombre de salariés a doublé.

1.4 Organigramme du pôle de développement

L'entreprise BSI compte une vingtaine de collaborateurs répartis comme suit :

- ⇒ Développement progiciel : 12,
- ⇒ Gestion et formation : 5,
- ⇒ Commercial : 4,
- ⇒ Réseau : 4.

Ayant été intégré dans le pôle Alsacien, je ne présente que l'organigramme de ce pôle. En effet, c'est le plus important des organigrammes car il montre autour de quels axes l'entreprise BSI s'organise.



L'équipe que j'ai rejoint pour effectuer mon stage, fut l'équipe Oracle. Chacun des deux pôles de développement se différencie par les outils qu'il utilise pour créer des logiciels. L'équipe Oracle développe des progiciels (tel CERES) reposant sur les produits Oracle.

Tous les développeurs de l'équipe sont également des formateurs. Leur semaine se divise en deux parties :

- ⇒ une partie de la semaine est consacrée au développement et à la maintenance (ou mise à jour).
- ⇒ L'autre partie de la semaine, les développeurs sont en clientèle, pour assurer la formation ou l'information sur le produit.

2 Analyse des besoins de l'entreprise

Ayant développé une application en mode client/serveur, l'entreprise BSI souhaitait la déployer sur Internet afin de la mettre à disposition d'un plus grand nombre de personnes. Le but était que les personnes aient juste besoin d'un navigateur avec une connexion à Internet, sans aucun autre produit Oracle.

Cette application ayant été développée avec « Developer », pack de logiciels de la gamme Oracle, utilisait une base de données Oracle. Il fallut donc me plonger dans la documentation des produits Oracle afin de savoir quels outils pouvaient répondre à la demande de l'entreprise. En effet, il me fallait savoir quels étaient les outils me permettant de passer d'une architecture client/serveur, à une architecture trois tiers.

2.1 Exposé du problème

En mode client/serveur, comme le montre la figure 1, les « forms »⁸ sont installées sur le poste du client. Elles sont donc chargées d'interroger la base de données, de construire des grilles avec les données renvoyées (partie logique de l'application) et de les présenter au client. Une telle architecture sur Internet était beaucoup trop lourde. En effet, il fallait absolument, pour rendre l'application fonctionnelle, que la partie logique des « forms » générées ne s'exécute pas sur le « browser »⁹ du client.

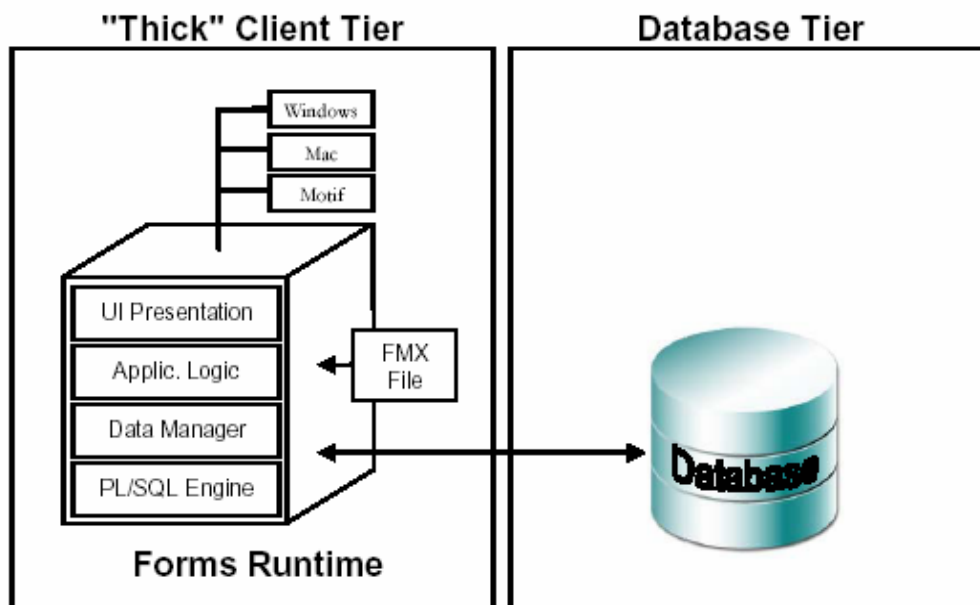


Figure 5 : Architecture Client/Serveur

⁸ Forms : voir page introduction.

⁹ Browser : outil permettant de visualiser des pages Web on peut citer par exemple Internet Explorer ou Mozilla

L'architecture trois tiers comprend un nouvel élément par rapport à la méthode client/serveur. Cet intermédiaire est un « serveur d'application » sur lequel seront stockées toutes les « forms ». Ainsi, c'est lui qui sera chargé de toute la partie logique de l'application. Il pourra discuter avec le client, via de simples requêtes (socket¹⁰ ou http). Il exécutera des procédures demandées par le client, interrogera la base de données, créera des pages Web dynamiquement en fonction des résultats obtenus et les renverra sur le browser du client. Ainsi, comme le montre cette image :

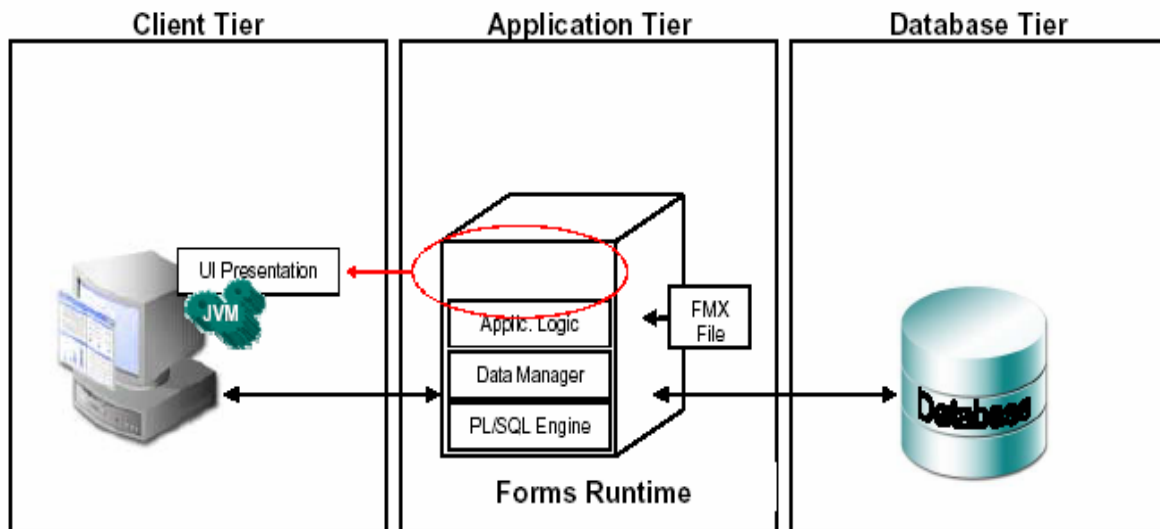


Figure 6 : Architecture trois tiers

On voit bien trois parties distinctes dans l'architecture trois tiers. Les procédures appelées par le client seront exécutées par la 2^{ème} tiers, ou « serveur d'application ». Ainsi, l'utilisateur fera du RPC¹¹ et son poste sera déchargé de toute la partie logique de l'application. On pourra donc tourner cette application vers Internet. Evidemment, le poste client n'aura pas beaucoup de tâches à exécuter. Néanmoins, le serveur d'application lui, va devoir gérer seul, les demandes de plusieurs utilisateurs. On utilise dans ce cas, des machines spéciales, dont le travail est uniquement destiné à cette tâche. La configuration matérielle peut donc être énorme sur un gros serveur d'application, par exemple, la configuration matérielle du serveur SUN FIRE[tm] V880 pourra être consultée en annexe, page 54.

¹⁰ Socket : point de connexion pour une communication. Il faut donc deux points de connexions pour les deux entités qui communiquent.

¹¹ RPC : abréviation de « Remote Procedure Call » : les procédures sont appelées par le client et exécutées à distance sur le serveur d'application.

2.2 Logiciels installés

Du fait que CERES était une application Client/Serveur, les logiciels Oracle utilisés n'étaient pas viables pour un passage de l'application sur Internet.

J'ai donc, en premier lieu, installé la base de données Oracle 8i Release 3, prenant en charge les technologies liées à Internet.

Puis, j'ai installé le serveur d'application Oracle 9i Application Server, représentant le deuxième tiers dans la technologie trois tiers.

Enfin, j'ai installé Oracle Developer Suite. Ces outils permettent de construire et de compiler des « forms ». En effet, les « forms » existantes ne géraient pas du tout le passage sur Internet. J'ai donc dû utiliser ces outils pour recompiler les « forms » existantes afin de permettre le déploiement sur Internet.

Il faut noter que ces logiciels ne sont vraiment pas simples à installer proprement. En effet, j'ai été vraiment ralenti dans cette phase de mon stage car il y avait quelques problèmes d'installation pour le serveur d'application. De plus, il y avait des problèmes de paramétrage pour pouvoir réaliser la discussion entre le serveur d'application et la base de données Oracle.

Par exemple, j'ai commencé par installer Oracle9i Database Release 2, mais la version était plus récente que celle du serveur d'application.

Puis, après avoir tout désinstaller, j'ai installé Oracle8i Database Release 3. Mais là encore, il m'a fallu renommer des bibliothèques ne fonctionnant pas avec mon processeur. J'ai donc dû télécharger un « patch »¹² d'Internet, afin de pouvoir mettre à jour le noyau de la version 8i pour que celle-ci supporte au mieux le serveur d'application.

Il faut également noter que les services Oracle requièrent une quantité très importante de mémoire, ce qui rend ces logiciels moins maniables. Enfin, il faut noter la quantité très importante en nombre d'heures qu'il faut passer pour certaines installations comme le serveur d'application : environ une journée, une journée et demi.

¹² « patch », « patcher » : Un patch comprend un ensemble de bibliothèques développées après la sortie du logiciel, afin de corriger des « bugs » relevés par les utilisateurs. Patcher est donc faire une mise à jour.

2.2.1 Installation d'Oracle8i Database Release 3

Pour pouvoir installer le serveur d'application et afin de pouvoir tester les « forms » axées Internet, j'ai installé une base de données Oracle orientée Internet.

L'installation de la base de données était requise afin que le serveur d'application sache avec quelle base de données il discuterait.

Cette installation engendre un nombre important de configurations à faire et de produits à installer. Elle crée tout d'abord un « HOME »¹³ du produit Oracle à installer.

Puis, pour créer véritablement la base de données locale, il faut lui donner un nom global, un SID¹⁴ et enfin, il faut créer un utilisateur avec un mot de passe.

Par exemple lors de mon stage, j'ai créé une base de données, dont le nom complet se composait du SID : « dbOracle » que j'allais donner à ma base de données, suivi du nom de domaine pour le réseau BSI : « BSIWEST ».

Une autre configuration très importante lors de la création de la base de données, est la mémoire que l'on va allouer à la base de données créée. Ici, la configuration standard a été respectée, à savoir, 81 Mo alloués en mémoire centrale, pour la base de données.

2.2.2 Mise à niveau d'Oracle8i Database Rel 3

Pour que la base de données Oracle 8i puisse supporter au mieux le serveur d'application, il a fallu « patcher » le noyau d'Oracle8i Database Release 3. Il m'a donc fallu télécharger un « patch » de 101 Mo, afin d'avoir les mises à jour des « bugs » rencontrés par d'autres utilisateurs.

Par ailleurs, la procédure à suivre pour mettre à jour la version 8i avec le « patch » était assez complexe. Par exemple, il fallait charger le « patch » mais ensuite, il y avait de nombreux scripts SQL à exécuter.

¹³ HOME : endroit où vont être stockés tous les composants installés. Deux Homes différents ne peuvent interagir entre eux que s'ils sont configurés pour. Par défaut, cela n'est pas possible.

¹⁴ SID : identificateur unique de la base de données. Le SID permet la connexion à la base de données.

Comme on pourra le voir, la procédure permettant de mettre Oracle 8i Database Release 3 à jour, est donnée en annexe, à titre indicatif

2.2.3 Installation d'Oracle9i Application Server

Il reste donc à installer Oracle9i Application Server, afin de pouvoir générer des « forms » axées Internet, dans le but de les placer sur le serveur d'application, pour pouvoir tester le tout.

L'installation est donc analogue à celle précédente. Néanmoins, il y a beaucoup plus de configuration manuelle à faire, du fait que lors de l'installation du serveur d'application, deux nouveaux « HOME » sont créés. En effet, on ne peut pas installer le serveur d'application dans le même « HOME » que le produit « Database » sous peine d'enlever tous les outils déjà installés.

Il m'a fallu faire manuellement des configurations¹⁵ permettant au serveur situé dans un certain « HOME », de communiquer avec la base de données créée précédemment dans un « HOME » différent.

Cette installation a été beaucoup plus longue que la précédente. En effet, Oracle9i AS est un « pack » comprenant des outils annexes au serveur d'application, tel « Oracle Portal 3.0 », destiné à concevoir des portails Internet ou encore des outils comme « Oracle Forms & Reports 6i » afin de pouvoir visualiser des « forms » axées Internet.

La place disque est également beaucoup plus importante. En effet, l'installation « Enterprise » requiert environ 2.67 GB.

2.2.4 Installation de Forms & Report Builder 6i

L'installation de « Developer Forms 6i » et de « Report Builder », était relativement plus courte que les autres. En effet, les seuls outils dont j'avais besoin étaient des outils de compilation afin de pouvoir mettre à jour des « forms » existantes.

Il m'a fallu installer ces produits Oracle dans un HOME différent des autres.

¹⁵ Ces configurations sont données en annexe, rubrique « Procédures liant outils et base de données »

2.3 Le déploiement de l'application

Lors de mon stage, le plan de travail suivi pour réaliser le projet, s'est décomposé en différentes parties assez distinctes.

2.3.1 Rapatriement des données

J'ai commencé par rapatrier toutes les données contenues dans la base de données de l'entreprise, vers ma base de données.

En effet, la base de données existante n'était pas du tout adaptée à la technologie Internet. Les structures entre la base de données existante et la base de données 8i installée sur mon poste étaient complètement différentes. Il était donc incontournable de rapatrier les données dans ma base, afin que les structures employées soient axées vers le monde Internet.

Nous avons donc utilisé le procédé d'export/import des tables contenant les données. Ainsi, nous avons créé un fichier représentant les données contenues dans les tables de la base de données de l'entreprise. Ensuite, nous avons téléchargé ce fichier sur mon poste. Enfin nous avons exploité ce fichier afin de recréer des tables sur ma base de données, ayant une structure orientée Internet.

Ensuite, nous avons téléchargé, toujours sur mon poste, toutes les « forms » qui avaient été développées en Client/Serveur, afin de les migrer en mode Internet.

2.3.2 Migration des programmes récupérés

Les programmes existants avaient été compilés et générés avec une version de Forms (6.0.5). Or, cette version ne permet pas le déploiement de l'application sur Internet ; par exemple, les « combo box » (objets contenant du texte) ne s'affichent pas. Il m'a donc fallu recompiler¹⁶ tous les programmes, toutes les bibliothèques et tous les menus, en version 6i, afin que ceux ci supportent le passage sur Internet.

Evidemment, vu la taille de l'application CERES, il aurait été trop long d'ouvrir tous les programmes par le logiciel Forms 6i et de les recompiler tous.

¹⁶ Le script de recompilation automatique est donné en annexe, rubrique « Migration des programmes » « script de recompilation »

J'ai donc utilisé un script (fournit en annexe) lançant en boucle une commande de compilation sur tous les fichiers possédant telle extension.

2.3.3 Paramétrage du serveur

Il me fallait paramétrer mon serveur d'application, afin que celui-ci puisse disposer des programmes

De plus, il me fallait paramétrer la méthode avec laquelle le serveur d'application allait lancer ses requêtes sur la base de données. En effet, la méthode proposée par défaut était une méthode utilisant les CGI. Or, il faut savoir que les CGI, dès la moindre requête, occupent toute la bande passante disponible. Il n'était donc pas imaginable de passer par cette méthode. J'ai donc dû paramétrer entièrement la méthode¹⁷ servlet, afin que ce soit via des servlets que le serveur d'application communique.

Pour que le serveur d'application sache où se trouve la base de données, afin qu'ils communiquent, j'ai dû paramétrer un fichier de configuration du serveur.

Puis, j'ai mis à jour (ou ajouté) certaines variables de la base de registres de Windows. En effet, il fallait positionner correctement le chemin disant où se trouvaient les programmes.

Un autre paramétrage était encore nécessaire. En effet, l'URL à rentrer pour pouvoir accéder à l'application CERES, était celle-ci :

<http://<poste>:<port>/servlets/f60servlet?form=<chemin de la 1ere forme a lancer>&userid=<name>/<pass>@<base>>

C'était donc une adresse très peu maniable qu'il fallait taper. De plus, le client ne devait pas à priori rentrer un mot de passe de connexion à la base de données.

J'ai donc créé un fichier « formsweb.cfg¹⁸ » avec en paramètres, l'endroit où se trouvait le premier programme à lancer et les paramètres de connexion à la base de données. Ainsi, l'URL de l'application est devenue :

<http://<poste>:<port>/servlets/f60servlet>

¹⁷ La procédure permettant de paramétrer la méthode servlet, est donnée en annexe.

¹⁸ Le contenu du fichier formsweb.cfg est disponible en annexe, rubrique « Paramètres Serveur » « Lancement de l'application ».

Ce qui raccourci considérablement l'URL et qui ne nécessite aucun mot de passe.

Dans la suite du déroulement du projet, j'ai été amené à résoudre un problème au niveau de la définition d'un profil d'utilisateur Internet. Or, en résolvant ce problème, j'ai également modifié l'adresse à rentrer et je l'ai encore réduite. Cette méthode et cette adresse pourront être trouvées en fin de section 2.3.7 (page 24).

2.3.4 Paramétrage des icônes

En mode Internet, les icônes que l'on peut habituellement voir dans les applications (fichier au format « .ico ») ne sont pas reconnues. J'avais donc, une fois l'application CERES lancée via un browser, des boutons sans images dessus donc inutilisables. Il fallait donc remédier à ce problème car l'application n'aurait pas été très fonctionnelle.

J'ai donc changer (et rajouter) des chemins sur le serveur, afin que celui ci prenne comme format d'icône, des images ayant l'extension « .gif ». De plus, il me fallait également reconstruire les icônes existantes en images gif. J'ai donc télécharger un logiciel de conversion de format d'image, afin de faire ce changement.

2.3.5 Paramétrage des éditions

J'ai paramétré les éditions afin de pouvoir, une fois que l'on aurait consulté des informations, les éditer.

Bien sûr, ce paramétrage ne devait pas modifier les sources. Ainsi, le seul paramétrage que j'ai pu faire pour que l'application trouve une imprimante, a été d'utiliser l'imprimante reliée au serveur.

En effet, cela n'est pas très fonctionnel dans le sens où un client utilisant l'application va vouloir imprimer une facture ou un bon... chez lui. Mais, cela reviendrait à dire au serveur de contrôler une ressource du client. Cela était donc impossible. Evidemment, il existe des moyens pour palier à cette limite, néanmoins cela impose de retoucher aux sources des programmes développés, ce qui dans un premier temps n'était pas le but.

Dans la suite du déroulement de mon stage, j'ai essayé sur quelques programmes de modifier les sources afin qu'un client puisse imprimer localement. En fait, cela m'a imposé de définir de nouvelles variables dans la base de registres de Windows et il m'a fallu changer quelques lignes dans les sources. Néanmoins, le résultat fut que l'édition était envoyée (sous forme de page web ou de document pdf) sur le poste client. Donc, ce dernier pouvait imprimer ce qu'il voulait en cliquant sur le bouton « imprimer » de son browser (un exemplaire de fichier pdf est donné en annexe, rubrique captures d'écran page 48).

2.3.6 Paramétrage des touches de fonctions

L'application existante utilisait des touches de raccourcis et notamment des touches de fonctions (F1 à F12). Or, après avoir migré l'application pour la rendre visualisable sur Internet, ces fonctions n'avaient pas gardé leurs propriétés.

J'ai pensé, au début de mon stage, qu'il était normal que ces fonctions soient comme désactivées en mode Internet, car il me paraissait probable que Windows se garde le droit de définir les touches de fonctions (par exemple, F5 indique au navigateur Web de rafraîchir la page). Cependant, des raccourcis comme CTRL+Q ne fonctionnait plus.

Je me suis donc demandé si il n'y avait pas un moyen de définir ces touches sur le serveur. En effet, du fait que l'application était visible via une applet JAVA, j'ai pensé que ce n'était pas Windows qui définissait les touches « *Fi* » à l'intérieur de l'application. Ainsi, j'ai trouvé un fichier « fmrweb.res » sur le serveur, définissant les touches de raccourcis d'une application sur Internet, à l'aide de code de fonctions.

J'ai donc recréé ce fichier (disponible en annexe) en mettant les bonnes définitions des touches, afin de rétablir les raccourcis.

Pour cela, j'ai dû respecter une certaine syntaxe comme suit :

JFN : JMN : URKS : FFN : URFD (espaces ignorés) JFN = Java function number JMN = Java modifiers number URKS = User-readable key sequence (double-quote) FFN = Forms function number URFD = User-readable function description (double-quote)

2.3.7 Définition d'un profil d'utilisateur Internet

En mode Client/Serveur, l'application CERES utilise un fichier de configuration (cereale.ini), pour savoir quel utilisateur se connecte à la base de données. Or, en passant par Internet, les utilisateurs n'ont plus à avoir ce fichier « cereale.ini ».

En effet, les internautes ne recevant que des grilles générées par le serveur d'application, ils n'interagissent jamais dans le fonctionnement de l'application Internet. Le serveur ne va donc pas aller chercher un fichier chez l'utilisateur (De plus, il n'en aurait pas le droit).

La méthode par défaut fut de prendre un fichier générique « céréale.ini » et de le mettre sur le serveur afin que ce dernier aille chercher les paramètres dans ce fichier. Mais, le problème en résultant était que tous les utilisateurs, Internet et Client/Serveur, allaient avoir les mêmes droits (le même profil).

Nous avons donc créé un fichier spécifique nommé « Internet.ini » et nous avons modifié les sources de l'application, afin de créer un profil d'utilisateur Internet.

A présent, on a déjà un premier niveau de restriction des droits, dans lequel un utilisateur Internet n'a pas les mêmes droits qu'un autre client qui utiliserait l'application en mode Client/Serveur. En effet, le projet de déploiement de l'application sur Internet était de rendre l'application consultable et il ne fallait pas que les utilisateurs Internet aient des droits de modifications.

Enfin, il restait un deuxième niveau de restriction à mettre en place, à savoir : en fonction de l'utilisateur Internet qui se connecte, allouer plus ou moins de droits à celui-ci. Il faut donc, à ce moment, arriver à récupérer son code client et son mot de passe, afin de savoir qui est cet utilisateur. De cette façon, on pourra paramétrer ses droits.

Ce deuxième niveau de restriction des droits est indispensable pour le bon fonctionnement de l'application CERES. En effet, la base de données liée à l'application, contient par exemple des renseignements sur les stocks de différents dépôts. Or, si on crée juste un profil Internet, sans récupérer d'informations sur les internautes, cela veut dire que tous les utilisateurs Internet interrogeront un et un seul dépôt. Les utilisateurs n'interrogeraient alors pas forcément leur dépôt !!

J'ai donc tenté de résoudre ce problème. Pour cela, je me suis demandé comment passer des paramètres à l'application. Je me suis alors rendu compte qu'il faudrait passer ces paramètres dans l'URL. Or, cela allait à l'encontre du travail

effectué auparavant, concernant le raccourcissement de l'URL. J'ai donc essayé, tout d'abord, de voir comment passer des paramètres dans l'URL. J'ai testé pour savoir si les paramètres que je mettais dans l'URL étaient bien pris en compte dans la « form ».

Ensuite, j'ai abordé le problème de la longueur de l'URL. En effet, l'URL à rentrer était de nouveau complexe :

```
http://<nom de_la_machine> :<port>/servlets/f60servlet
?otherparams=<param1>=<valeur1>%20<param2>
=<valeur2>...
```

Il fallait donc impérativement créer un script qui facilite le travail de l'utilisateur. J'ai donc créé une page HTML avec un script demandant au client des paramètres propres à lui même.

Puis j'ai lancé l'ouverture d'une seconde page HTML à laquelle je passais tous les arguments nécessaires à la cohérence de l'application CERES. Il ne restait plus qu'à mettre la page sur le serveur, afin qu'elle soit le point de départ de cette application.

J'ai donc déplacé cette page sur mon serveur et j'ai rajouté un alias afin qu'un chemin du serveur pointe sur la page de départ. J'ai également changé l'attribut « DocumentRoot » du serveur, afin de ne pas laisser trop de libertés au client sur le serveur.

Ainsi, j'ai résolu le deuxième niveau de restriction des droits mais j'ai également raccourci l'adresse Internet de l'application au maximum. En effet, étant donné que la page était devenue la racine du serveur, les clients pouvaient y accéder juste en tapant l'adresse :

http://nom_de_la_machine:<port>/

2.3.8 Résolution des bugs

Plusieurs bugs sont apparus lors du déploiement de l'application sur Internet. Il m'a donc fallu trouver ces bugs et dans la mesure du possible, les résoudre.

Il faut noter qu'il était assez difficile de savoir s'il fallait corriger ou non tel ou tel bug. En effet, si la fonctionnalité de l'application s'en trouvait améliorée, le travail effectué n'était pas d'une grande utilité. Du fait que des corrections étaient apportées assez fréquemment par les développeurs du logiciel CERES,

mes programmes devaient être souvent remis à jour, tout comme ma base de données. Ainsi, à chaque fois qu'une mise à jour était effectuée, je devais modifier à nouveau les bugs spécifiques au passage des programmes sur Internet. Ceci était dû au fait que les développeurs et moi travaillions avec des versions différentes.

Les différents bugs que j'ai corrigé, sont dus d'une part au passage d'une version (6.0.5) à une autre (6.0.8) mais aussi du passage des programmes sur Internet, ce qui engendre des fonctions n'étant pas prises en compte ou gelant l'application ...

2.3.8.1 Bugs liés aux versions des produits

Sans entrer dans le détail, je présente une liste non exhaustive des problèmes que j'ai rencontré lors de la migration des programmes développés avec l'outil Forms version 6.0.5 , vers la version Forms 6i (version 6.0.8).

2.3.8.1.1 Problème lié aux « Combo Box »

Les « combo box » sont des objets représentant un cadre que l'utilisateur doit remplir et proposant une liste déroulante de choix.

Or, si dans la version 6.0.5, la longueur maximale des combos pouvait être donnée comme étant la longueur d'un index, ce n'était pas le cas en version 6i. Il m'a donc fallu, à défaut de trouver un script permettant une modification globale, changer manuellement la longueur de tous les combos. Cette modification est un véritable problème car à chaque mise à jour de programmes, il faudra refaire ces changements.

2.3.8.1.2 Problème lié aux nombre d'utilisateurs

La version actuelle du serveur d'application (i.e : Oracle9i Application Server), est configurée pour supporter au plus 128 utilisateurs. Ce paramètre n'est pas modifiable par l'administrateur du réseau. Pour avoir plus de connexions, il faut attendre une version d'Oracle proposant un nombre maximal plus élevé et réinstaller un serveur d'application.

2.3.8.2 Bugs propre à la version Internet

2.3.8.2.1 Problème lié aux « Polices »

Si le texte apparaît correctement sur des boutons, des zones de texte... en mode Client/Serveur, cela n'est pas forcément le cas en mode Web. Pour ce faire, il faut juste agrandir les boutons ou les zones de texte, pour apercevoir le texte. De même, certains objets comme les « scrollbar » peuvent apparaître beaucoup plus fins en mode Web par rapport à la version Client/Serveur. Il faut là encore penser à les grossir.

2.3.8.2.2 Problème de fonctions non supportées

Lors des tests de l'application, certaines grilles que j'appelais, refusaient de s'ouvrir. Il m'a fallu déboguer et comprendre pourquoi j'avais ce problème. Ainsi j'ai trouvé qu'une fonction permettant d'écrire du texte sur deux lignes (équivalente au « \n » en langage C) générait ce problème.

2.3.8.2.3 Problème de fonctions non admises

Certaines fonctions, utilisées en mode Client/Serveur, ne sont pas admises en mode Web, notamment à cause de la quantité importante de bande passante qu'elles demanderaient. En effet, des événements de souris, renseignant sur sa position, n'apporte rien à l'utilisateur et surchargent très fortement le réseau. Il m'a donc fallu retirer ces événements, pour éviter, là encore, que l'application se gèle.

2.3.8.2.4 Problème de langues

J'ai également dû résoudre le problème du symbole « € » qui en version Internet, n'était pas compris. En fait, il me fallait juste changer dans la base de registres, la norme utilisée pour le codage alphabétique.

2.3.9 Résultats du déploiement et remarques

Lorsqu'un utilisateur rentre l'URL de connexion à l'application via un browser, une fenêtre Internet s'ouvre et lui demande de saisir des paramètres propres à lui seul¹⁹.

Puis, un script ouvre une fenêtre Internet qui passe à l'application les paramètres rentrés. Il faut préciser que l'utilisateur n'a besoin de rien d'autre qu'un browser et d'une connexion Internet. En effet, si l'utilisateur ne possède pas le logiciel permettant de visualiser l'applet, le serveur lui propose de le télécharger et le logiciel s'installe automatiquement. Ce logiciel est « Jinitiator » version 1.1.8.19, un produit possédant une signature d'authentification, donc sans danger pour l'utilisateur.

Attention : Il reste encore un problème lié au proxy. En effet, le réseau de l'entreprise BSI est sécurisé par un proxy. Un utilisateur venant de l'extérieur du réseau ne peut pas se connecter à l'application CERES.

Il faut, s'il veut utiliser l'application CERES sur Internet, qu'il passe par un chemin le conduisant sur un routeur avec lequel l'entreprise BSI a établi une passerelle.

Ainsi, pour se connecter à ce routeur, l'utilisateur pourra se servir du logiciel PC-Anywhere. Puis étant connecté sur le routeur, il pourra accéder à mon poste en tapant l'URL :

```
http://<adresse IP de ma machine> :<port>/
```

En effet, le nom BSI40 est reconnu à l'intérieur du réseau BSI. De l'extérieur, il n'y a pas de résolution de nom. Il faut donc entrer l'adresse IP de la machine.

¹⁹ Des captures d'écran mises en annexes, montrent les fenêtres qui s'ouvrent lorsqu'un utilisateur a rentré l'URL.

Conclusion

Ce stage aura été pour moi, l'occasion de découvrir le fonctionnement d'une entreprise de services informatiques. De plus, ce stage m'aura permis d'acquérir la démarche à suivre pour mener à bien un projet. Notamment de voir que l'étape la plus importante réside dans le choix des outils à installer et à utiliser. En effet, pendant notre année scolaire, nous n'avons pas à nous poser les questions de savoir quels logiciels utilisés.

Les deux premières semaines de mon stage m'auront permis de mesurer l'impact qu'aurait pu avoir un mauvais choix dans les produits à installer. J'ai pu ainsi comprendre l'importance de l'investissement dans un projet en entreprise.

Ce projet m'a également fait comprendre comment fonctionnait une application en mode trois tiers. En effet ce procédé m'était, au début de mon stage, complètement étranger.

Enfin sur le plan personnel, j'ai là aussi pu apprécier le travail en équipe dans une entreprise informatique.

ANNEXES

I. Procédure liant Outils et Base de données

Comme on l'a vu dans le rapport, les produits Oracle doivent s'installer dans des « HOME » différents. Ainsi, le fait qu'un produit soit dans un HOME différent de celui de la base de données, entraîne le rajout de certaines lignes dans certains fichiers pour permettre à un quelconque produit Oracle de savoir où se trouve la base de données avec laquelle il doit discuter.

Il m'a donc fallu mettre à jour les fichiers nommés « TNSNAMES.ORA » dans tous les « HOMES » différents de celui de la base de données.

Ainsi, il m'a également fallu tenir compte du fichier « SQLNET.ORA » afin de ne pas ajouter n'importe quoi comme ligne dans le premier fichier cité.

En fait, le fichier « SQLNET.ORA » peut contenir une ligne donnant une extension à toutes les instances de bases de données se trouvant dans le fichier TNSNAMES.ORA. Par exemple, j'ai rajouté les lignes suivantes :

```
DBORACLE.WORLD =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP)(HOST = BSI40)(PORT = 1521))
    )
    (CONNECT_DATA = (SERVICE_NAME = dbOracle.BSIWEST))
  )
```

Si le fichier « SQLNET.ORA » possède une ligne avec l'extension « world » :

```
names.default_domain = world
name.default_zone = world
```

Sinon, je devais juste ajouter ces lignes :

```
DBORACLE =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP)(HOST = BSI40)(PORT = 1521))
    )
    (CONNECT_DATA = (SERVICE_NAME = dbOracle.BSIWEST))
  )
```

Avec un fichier « SQLNET.ORA » sans extension.

II. Procédure de mise à jour d'Oracle8i Database

Après avoir téléchargé le patch permettant la mise à jour du noyau d'Oracle 8i Database Release 3, l'opération s'est déroulée en 2 phases.

i. Installation du patch

Le patch téléchargé était un « installer » Oracle chargé d'installer, en fait de remplacer, des outils dont se servait le noyau de la base de données Oracle 8i. Il faut noter que cette étape a due être modifiée. En effet, ayant une machine équipée d'un Pentium 4, je dus renommer des « dll » afin de pouvoir lancer l'installation.

ii. Lancement des scripts

Il me fallut ensuite exécuter des scripts sur la base de données, afin de mettre à jour certaines tables systèmes de la base de données. Cet extrait répertorie les corrections apportées par la mise à jour ainsi que les scripts à exécuter.

1. When applying this patchset on Windows NT and Windows 2000, you must log onto the system as a user with Administrative privileges (e.g., as a user which is a member of the local Administrators group).
2. Unzip the downloaded patchset file into a new directory.
3. In the Oracle Home in which you wish to install the patchset, stop any existing Oracle Server instances with normal or immediate priority. Stop any other Oracle-specific services or applications which are running out of the chosen Oracle Home.
4. You may install the Patch Set through either an interactive or a silent installation.
 - o To perform an interactive installation using the Oracle Universal Installer graphical interface, start the installer by running the `setup` executable located at the root of the unzipped patchset installation area. Then follow the steps given below within the installer:
 1. On the **Welcome** screen, click the [Next] button. This will display the **File Locations** screen.tory where you unpacked the Patch Set tar file.
 2. The default location for the **Source...** entry field should already be pointing to the correct `products.jar` file needed for the patchset installation. Leave this field unchanged.
 3. Select the Oracle Home in which you wish to install the patchset from the drop down list of Oracle Homes or Oracle Home Names.

4. Click the [Next] button.

The products file will be read and the installer will automatically load the product definitions for those components whose base release (8.1.7.0) version exists in the currently selected Oracle Home.

Once the products have been loaded from the patchset installation area, the installer will display those products which will be installed in the current session.

5. Verify the products listed and then click on the [Install] button.
6. Once the installation has completed successfully, it will display **End of Installation**. Click on [Exit] and confirm to exit the installer.
 - o To perform a silent installation requiring no user intervention:
 1. Copy the response file template provided in the \response directory of the unzipped patchset installation area.
 2. Edit the values for all fields labeled as "<Value Required>" according to the comments and examples in the template.
 3. From the unzipped patchset installation area, start the installer by running the setup executable with the following parameters:

```
setup.exe -silent -responseFile  
<full_path_to_your_response_file>
```

5. Disabling system triggers.

Before performing the next step, you must first set `_SYSTEM_TRIG_ENABLED = FALSE` in the initialization parameter file. To do this add the following line to your `init<sid>.ora` file and restart the database (see Note 149948.1 in Metalink for more information on this issue).

```
_system_trig_enabled=false
```

6. Invoke SQL*Plus (sqlplus), connect as internal and run the following:

```
select * from duc$ where PACK='JIS$INTERCEPTOR$' ;
```

If no rows were returned go to the next step. If a row was returned delete it using the following:

```
delete from duc$ where PACK='JIS$INTERCEPTOR$' ;  
commit;
```

7. Invoke SQL*Plus (sqlplus), connect as internal and run the following SQL scripts with [event 10520](#) set. **NOTE: This event is not intended for regular database operation and MUST be turned off as indicated below:**

```
ALTER SESSION SET EVENTS '10520 TRACE NAME CONTEXT  
FOREVER, LEVEL 10';  
?/rdbms/admin/catalog.sql  
?/rdbms/admin/catproc.sql  
?/rdbms/admin/catrep.sql (This only needs to be run if you are using  
symmetric/advanced replication. This is not necessary for sites using dblinks and  
read-only snapshots if symmetric/advanced replication is not installed)  
CONNECT / AS SYSDBA;  
update obj$ set status=5 where type#=29 and owner#!=0;  
commit;
```

8. You can ignore this step if you have already executed it as part of a previous 8.1.7

patch set install.

If Java has previously been loaded into the database, invoke SQL*Plus (sqlplus), connect as internal and run the following SQL DDL command(* **Note:If Java has not previously been loaded into the database, skip this step**):.

```
SQL> create or replace java system
2 /
```

(Note that the / (slash) is important to ensure the execution of the command).

To ensure the correct installation of the XDK in the database run the following 2 scripts (Bug 2115227):

```
$ORACLE_HOME/oracore/admin/initxml.sql
$ORACLE_HOME/rdbms/admin/catxsu.sql
```

*** To determine whether Java has previously been loaded into the database:**

invoke SQL*Plus (sqlplus), connect as internal and run the following:

```
SQL> select count(*) from all_objects where object_type like
'JAVA%';
```

Java is installed if you get a non zero result returned from the query.

9.

10. **You can ignore this step if you have already executed it as part of a previous 8.1.7 patch set install:**

If Java has previously been loaded into the database, invoke SQL*Plus (sqlplus), connect as internal and run the following scripts which are located in \$ORACLE_HOME/javavm/install (**Note: If Java has not previously been loaded into the database, you can skip this step**):

```
load_jis.sql
jisja.sql
initjsp.sql
jspja.sql
```

Notes:

When running the \$ORACLE_HOME/javavm/install/jisja.sql script the following error may be reported and can be ignored:

```
call jis_exit_java_session(0)
```

```
ERROR at line 1:
```

```
ORA-29515: exit called from Java code with status 0
```

Bug 1459233 was filed against this problem which is fixed in 9i.

11. If you deleted a row from duc\$ in step 8 then reinsert it again. Invoke SQL*Plus (sqlplus), connect as internal and run the following SQL:

```
insert into duc$ (OWNER, PACK, PROC, FIELD1, OPERATION#,
SEQ) values ('SYS', 'JIS$INTERCEPTOR$', 'USER_DROPPED',
0, 1, 1);
commit;
```

12. Enabling system triggers.

You must either remove _SYSTEM_TRIG_ENABLED from the initialization parameter file or explicitly set it to TRUE before attempting any other patch set post-install steps. Restart the database.

13. *This step is optional - it will recompile all invalid PL/SQL packages now rather than when accessed for the first time - you can also use [utlrcmp.sql](#) to parallelize this.*

Invoke SQL*Plus (sqlplus), connect as internal and run the following SQL script:

```
~/rdbms/admin/utl1rp.sql
```

14. Users of interMedia Text must run each of the following scripts in SQL*Plus, as CTXSYS, in the following order.

```
~/ctx/admin/upgrade/u0801071.sql  
~/ctx/admin/upgrade/u0801072.sql  
~/ctx/admin/dr0pkh.sql  
~/ctx/admin/dr0plb.sql  
~/ctx/admin/dr0type.plb  
~/ctx/admin/dr0typec.plb
```

15. **You can ignore this step if you have already executed it as part of a previous 8.1.7 patch set install:**

The installation of this patch set fixes a potential security hole in the XSQL Servlet - see [here](#) for more detail. The patchset installation does not, however, modify any of the Oracle HTTP Server configuration files to reflect changes to the XSQL Servlet installation. If you have previously installed the Oracle HTTP Server, you will need to update one of the configuration files manually after the patchset installation is complete. If you do not do so, the XSQL Servlet may not work in the context of the Oracle HTTP Server after the patchset is installed. To update your Oracle HTTP Server configuration, perform the following steps:

- 1) Edit the file: \$ORACLE_HOME/Apache/Jserv/conf/jserv.properties
- 2) Search for the string: "XSQLConfig.xml File location"
- 3) Modify the parameter wrapper.classpath on the line immediately following the above string to read: \$ORACLE_HOME/oracore/admin
- 4) Save the file.

You can ignore this step if you have already executed it as part of a previous 8.1.7 patch set install:

The installation of this patch set fixes a potential security hole in the PL/SQL XML parser - see [here](#) for more detail. To load the new XML parser in to the database, run the following script as user SYS:

```
? /xdk/plsql/parser/bin/load.sql
```

III. Migration des programmes

i. Scripts de recompilation

Comme je l'ai mentionné dans mon rapport, les programmes avaient été compilés avec une version qui ne gérait pas le passage de l'application sur Internet. Ces programmes peuvent être considérés comme appartenant à quatre types différents :

- ⇒ Les Forms (corps de l'application)
- ⇒ Les Menus (permettant la navigation dans l'application)
- ⇒ Les Bibliothèques (permettant de définir des fonctions génériques)
- ⇒ Les Aides (apportant une aide à l'utilisateur)
- ⇒ Les Rapports (entêtes des éditions)

Il m'a donc fallu tout recompiler et tout générer.

Pour ce faire, j'ai utilisé un script permettant de faire cette opération en boucle (on a juste à taper ces lignes dans une fenêtre DOS) :

```
-- Pour les forms :
for %x in (*.fmb) do start /W ifcmp60 module=%x
userid=user/psswd@base
batch=yes module_type=form compile_all=yes
window_state=minimize upgrade=yes

-- Pour les librairies :
for %x in (*.pll) do start /W ifcmp60 %x userid=user/psswd@base
batch=yes
module_type=LIBRARY compile_all=yes window_state=minimize
upgrade=yes

-- Pour les menus :
for %x in (*.mmb) do ifcmp60 %x user/pass@base compile_all=yes
batch=yes
upgrade=yes
```

ii. Paramétrage des icônes

Afin de pouvoir afficher des icônes en mode Internet, j'ai dû rajouter des lignes dans certains fichiers afin de dire au serveur où se trouvaient les icônes. De plus, j'ai dû télécharger un logiciel me permettant de générer des icônes au format « gif » à partir d'icônes au format « ico ».

⇒ Fichier 6iserver.conf :
ligne rajoutée : Alias /web_icons/ "e:\projet\icone/"

⇒ Fichier Registry.dat :
Lignes rajoutées : default.icons.iconpath=/web_icons/
default.icons.iconextension=gif

IV. Paramétrage spécifique du serveur

Les textes suivants sont des procédures à suivre, afin de pouvoir augmenter la fonctionnalité de l'application CERES sur Internet.

i. Paramétrage de la méthode « servlet »

Les servlets étant plus rapide en temps d'exécution que les cgi-bin, il m'a fallu paramétrer cette méthode. Voici donc la procédure qu'il fallait suivre :

SETTING UP JSERV FOR THE LISTENERSERVLET

To add the ListenerServlet's Java classes to the JServ configuration first load the file %IAS_HOME%\Apache\Jserv\servlets\zone.properties into an editor and add the line:

```
repositories=DEVELOPER_HOME\forms60\java\f60srv.jar
```

Make sure that you replace DEVELOPER_HOME with the correct path; placeholders are not allowed in .properties files.

For convenience also add the following line to zone.properties. This will enable you to abbreviate the servlet name as f60listener.

```
servlet.f60listener.code=oracle.forms.servlet.ListenerServlet
```

Now edit %ORACLE_HOME%\Apache\Jserv\conf\jserv.properties and add these lines:

```
wrapper.path=IAS_HOME\bin;DEVELOPER_HOME\bin;C:\winnt;C:\winnt\system32  
and  
wrapper.env=PATH=IAS_HOME\bin;DEVELOPER_HOME\bin;C:\winnt;C:\winnt\system32
```

If wrapper.path or wrapper.env=PATH lines already exist don't add new ones but make sure that the ORACLE_HOME\bin directories are contained in this order. Always make sure that wrapper.path and wrapper.env=PATH are set to identical values, because JServ does not document which of these two settings is preferred.

This configuration is necessary for the ListenerServlet to find a DLL with native C functions.

Finally change the line

```
wrapper.env=ORACLE_HOME=IASHOME
```

to

```
wrapper.env=ORACLE_HOME=DEVELOPERHOME
```

in jserv.properties. This is necessary for the ListenerServlet to find another very specific DLL, but...

... WARNING... changing this may cause malfunction of other Servlets that need ORACLE_HOME set to the IASHOME. A clean solution to solve this with Patch 5 features is described below.

Now test the Servlet by accessing the following URL:
<http://yourhost:port/servlets/f60listener>

As you can see you are using the abbreviation as declared the zone.properties file.

You should receive a page with the heading "Oracle Forms 6i Listener Servlet" and some options to test/troubleshoot the ListenerServlet.

SETTING UP THE FORMS HTML PAGE

In your existing HTML page in which your Webforms applet is embedded remove all connectMode, serverName and serverPort parameters and add these instead:

```
<param name="connectMode" value="http">
<param name="serverURL" value="http://yourhost:port/servlets/f60listener">
(for JInitiator in IE)
```

```
<embed ...
  connectMode="http"
  serverURL="http://yourhost:port/servlets/f60listener"
  ...>
```

(for JInitiator in Netscape)

Finally call your updated HTML file through the web server. As You can see from JInitiator's Java Console output (if enabled) several HTTP requests to the ListenerServlet are made while running the Forms module.

AVOIDING TO CHANGE JSERV'S ORACLE_HOME

Because you have been warned above, you will most likely not want to change JServ's ORACLE_HOME setting. Here are the steps to avoid this if you are using Patch 5 or higher:

Create a file called f60envfile with this contents:

```
ORACLE_HOME=DEVELOPERHOME
PATH=IAS_HOME\bin;DEVELOPER_HOME\bin;C:\winnt;C:\winnt\system32
FORMS60_PATH=...
NLS_LANG=AMERICAN_AMERICA.WE8ISO8859P1
and extend the zone.properties to this configuration:
```

```
servlet.f60listener.code=oracle.forms.servlet.ListenerServlet
servlet.f60listener.initArgs=envfile=X:\full\path\to\f60envfile
```

This will make the ListenerServlet use the environment variables from f60envfile. You can even declare several different Servlet aliases for the ListenerServlet where each alias uses a different envfile, so that You can run Webforms with different environment settings depending on which alias You are using in the HTML file.

ii. Paramétrage spécifique à l'application

Dans le but de faciliter la tâche de l'utilisateur, notamment pour taper l'URL de connexion à l'application, j'ai créé un fichier : « formsweb.cfg », afin qu'il contienne certains paramètres que l'utilisateur n'ait pas à rentrer, ni même à connaître. On va d'ailleurs retrouver dans ce fichier, des commandes montrant que l'on utilise bien la méthode des « servlets ».

```

; Forms Web Servlet Configuration File
;-----

; SYSTEM PARAMETERS
;-----

baseHTML=%FORMS60%\server\base.htm
baseHTMLJInitiator=%FORMS60%\server\basejini.htm
HTMLdelimiter=%
MetricsServerPort=9020
MetricsServerErrorURL=
; The next parameter specifies how to execute the Forms applet under
; Microsoft Internet Explorer 5.0. Put IE50=native if you want the
; Forms applet to run in the browser's native JVM.
IE50=JInitiator

; USER PARAMETERS
;-----

; 1) Runform arguments:

; endroit où se trouve la forme à lancer par défaut
form=e:/projet/ceres60/form/cereale.fmx

; paramètre de connexion à la base
userid=cereale/bsi@dbOracle

; autre paramètres (à passer à l'application par exemple)
otherparams=

; 2) page HTML , attributs permettant de mettre du code avant et après la forme.

pageTitle=Ceres - BSI SA &copy
HTMLbodyAttrs=bgcolor="#ffff0"

; added specific text for the future client
; by ben - 25/07/2002
HTMLbeforeForm=<center><font color="blue" face="Verdana"><h1>Bienvenue
&nbsp;sur&nbsp;la&nbsp;version<br>Internet&nbsp;du&nbsp;logiciel&nbsp;
CERES</h1></font><br><font color="red" face="verdana"><b>Attention&nbsp;
:&nbsp;Ne&nbsp;pas&nbsp;fermer&nbsp;la&nbsp;fenêtre&nbsp;internet.<br>
Cela&nbsp;fermerait&nbsp;aussi&nbsp;l'application&nbsp;CERES&nbsp;!</b>
</font><p style="border: 1px solid #000000;padding: 10px">Si&nbsp;en&nbsp;
revanche&nbsp;vous&nbsp;fermez&nbsp;l'application,<br>vous&nbsp;pouvez
&nbsp;la&nbsp;réouvrir&nbsp;en&nbsp;appuyant<br>sur&nbsp;le&nbsp;
bouton&nbsp;rafraîchir&nbsp;de&nbsp;la&nbsp;fenêtre&nbsp;de&nbsp;
votre&nbsp;navigateur.</p><br><p style="border: 1px solid #000000;
padding: 10px;width: 30%"><u><b>paramètres&nbsp;de&nbsp;
connexion&nbsp;:</b><br><i>login&nbsp;:&nbsp;INTERNET<br>mot&nbsp;de&nbsp;
passe&nbsp;:&nbsp;INTERNET</i></u></p></center>
HTMLafterForm=<h5><br><p align="right">fonctionnalité ajoutée par Benoît
CHRISTOPHE - BSI SA &copy</p></h5>

; 3) paramètres de l'applet:

; astuce : on met à zero sa taille dans la fenêtre du browser et on met a true separateFrame

```

```

; afin que l'applet s'affiche dans une fenetre detachee du browser.
width=0
height=0
separateFrame=true
splashScreen=true

; select default background by not specifying a value
;background=e:\projet\icone\bascule.gif
;lookAndFeel=Generic
;colorScheme=teal
serverApp=default
;serverPort=9001
;serverHost=BSI40
;serverURL=
archive_jini=f60all_jinit.jar
archive_ie=f60all.cab
;connectMode=Socket
archive=f60web.jar
archive_jini=f60all_jinit.jar
archive_ie=f60all.cab

; tests du 09/08/2002
connectMode=HTTP
serverURL=http://BSI40:7778/servlets/f60listener

; 4) Parametre de JInitiator

; Page displayed to Netscape users to allow them to download JInitiator.
; If you create your own version, set this parameter to point to it.
jinit_download_page=/jinitiator/us/jinit_download.htm

; Parameters related to the version of JInitiator.
;jinit_classid=clsid:ff348b6e-fd21-11d4-a3f0-00c04fa32518
;jinit_exename=jinit.exe#Version=1,1,8,7
;jinit_mimetype=application/x-jinit-applet;version=1.1.8.7

; *****
; changement de version de JInitiator : sur mon poste les changements fonctionnent
jinit_classid=clsid:5e2a3510-4371-11d6-b64c-00c04faedb18
jinit_exename=jinit.exe#Version=1,1,8,19
jinit_mimetype=application/x-jinit-applet;version=1.1.8.19
; fin changement

; *****
; CONFIGURATIONS SPECIFIQUES
; *****
; You may define your own specific, named configurations (sets of parameters)
; by adding special sections as illustrated in the following examples.
; Note that you need only specify the parameters you want to change. The
; default values (defined above) will be used for all other parameters.
; Use of a specific configuration can be requested by including the text
; "config=<your_config_name>" in the query string of the URL used to run
; a form. For example, to use the sepwin configuration, your could issue
; a URL like "http://myhost.mydomain.com/ifcgi60.exe?config=sepwin".

; Example 1: configuration to run forms in a separate browser window with
; "generic" look and feel (include "config=sepwin" in the URL)

```

```

[sepwin]
separateFrame=True
lookandfeel=Generic

; Example 2: configuration affecting users of MicroSoft Internet Explorer 5.0.
;       Forms applet will run under the browser's native JVM rather than
;       using Oracle JInitiator.
[ie50native]
IE50=native

; Example 3: configuration forcing use of the base.htm base HTML file in all
;       cases (means applet-style tags will always be generated and
;       JInitiator will never be used).
[applet]
baseHTMLJInitiator=

[trace]
otherparams=record=collect log=c:\forms.log

```

iii. Paramétrage des éditions

Pour pouvoir lancer une impression, il faut démarrer le service « forms server », en tant que l'utilisateur connecté.

Ici, on ouvre donc une session du service form server en tant que BSIWEST\BC. (BSIWEST était le nom du réseau de l'entreprise et BC mon login de connexion)

De plus, il faut configurer l'imprimante reliée au serveur (donc ici à ma machine) comme étant l'imprimante par défaut.

Il faut également que l'utilisateur qui a démarré le serveur d'application ait tous les droits sur cette imprimante.

Une fonctionnalité a cependant été testée. En effet, en changeant un peu les sources des programmes, on peut envoyer un « report » dans une fenêtre web, sur le poste client. Ainsi, le client peut, en utilisant la fonction « imprimer » de son navigateur, imprimer chez lui ce qu'il veut.

Modifications apportées au code du programme « edmvstk.fmb » :

```

/* changements : anciens parametres
  Add_Parameter(id_liste,'DESTYPE',TEXT_PARAMETER,'PRINTER');
  Add_Parameter(id_liste,'DESNAME',TEXT_PARAMETER,'');
*/
  Add_Parameter(id_liste, 'PARAMFORM', TEXT_PARAMETER, 'YES');
  Add_Parameter(id_liste,'DESTYPE',TEXT_PARAMETER,'FILE');
  Add_Parameter(id_liste,'DESNAME',TEXT_PARAMETER,'E:\ORACLEiAS\
      806\tools\web60\temp\edstk.html');
  Add_Parameter(id_liste,'DESFORMAT',TEXT_PARAMETER,'HTML');
...

```



```
...  
Run_Product(REPORTS,nom_edition,SYNCHRONOUS,RUNTIME,  
            FILESYSTEM,id_liste,'');  
Web.Show_Document('http://BSI40:7778/dev60temp/edstk.html','_BLANK');
```

Attention : Afin de pouvoir utiliser correctement la fonction WEB.SHOW_DOCUMENT, il faut s'assurer que certaines valeurs, dans la base de registres, sont présentes. De même, il faut s'assurer que l'alias (ici : dev60temp) est bien défini dans les paramètres du serveur.

Clés à ajouter sous le chemin :

Poste de travail\HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE

⇒ **Nom** : FORMS60_MAPPING

○ **Valeur** : /dev60temp/

⇒ **Nom** : FOMRS60_OUTPUT

○ **Valeur** : E:\ORACLEiAS\806\tools\web60\temp

⇒ **Nom** : FORMS60_REPFORMAT

○ **Valeur** : HTML (ou PDF)

Alias à paramétrer dans le fichier : e:\oracleiAS\806\conf\6iserver.conf

⇒ **Valeur** : Alias /dev60temp/ "e:\oracleiAS\806\tools\web60\temp/"

iv. Ajout de valeurs dans la base de registres de Windows

Certaines valeurs sont à rajouter dans la base de registres de Windows. Voici une liste de ces valeurs :

⇒ **Chemin** :

« Poste de travail »\HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE :

○ **Clé** :

FORMS60_PATH

- Valeur :
E:\projet\ceres60\form ;e:\projet\ceres60\lib ;e:\projet\ceres60\menu ;e:\projet\ceres60\aide
- Clé :
REPORTS60_PATH
- Valeur :
E:\projet\ceres60\form ;e:\projet\ceres60\lib ;e:\projet\ceres60\menu ;e:\projet\ceres60\aide

v. Paramétrage des touches de fonctions

Le fichier concernant le paramétrage des touches de fonctions, n'est pas donné en entier. En effet, seule la syntaxe à respecter montre la méthode qu'il m'a fallu suivre pour paramétrer les touches.

```
# fichier de configuration des touches de raccourci
# pour des forms en mode web
#
# syntaxe :
# JFN : JMN : URKS : FFN : URFD (espaces ignorés)
#
# JFN = Java function number
# JMN = Java modifiers number
# URKS = User-readable key sequence (double-quote)
# FFN = Forms function number
# URFD = User-readable function description (double-quote)
#
# JAVA FUNCTION NUMBER
# 33 = PageUp
# 34 = PageDown
# 35 = End
# 36 = Home
# 37 = LeftArrow
# 38 = UpArrow
# 39 = RightArrow
# 40 = DownArrow
# 65 - 90 = Ctrl+A thru Ctrl+Z (These will always have the control
# modifier explicitly included, as well as any other
# modifiers that might be used.)
# 112 - 123 = F1 thru F12
# 9 = Tab (Ctrl+I, without the control modifier)
# 10 = Return (Ctrl+J, without the control modifier)
#
```

```

# JAVA MODIFIERS NUMBER
# Equal to the sum of the values for the modifier keys:
# 0 = None
# 1 = Shift
# 2 = Control
# 4 = Meta
# 8 = Alt
#
# FORMS FUNCTION NUMBER
# The Forms function numbers match the function numbers found in a
#
# typical Forms key binding file.
#
# USER-READABLE STRINGS
# The double-quoted strings appear when users click [Show Keys], and
# are used for this purpose only. These strings can be translated as
# needed. Note that the strings do not affect what actually happens
# when end users press a particular key sequence.
#
#
# définitions des touches spécifiques à l'application CERES
# added by ben : 14/08/2002
#
# exemple de touche de raccourcis :
9 : 0 : "Tab" : 1 : "Next Field"

```

On a donc dans ce fichier, la syntaxe à respecter ainsi que les codes des principales fonctions.

V. Captures d'écran de l'application CERES sur Internet

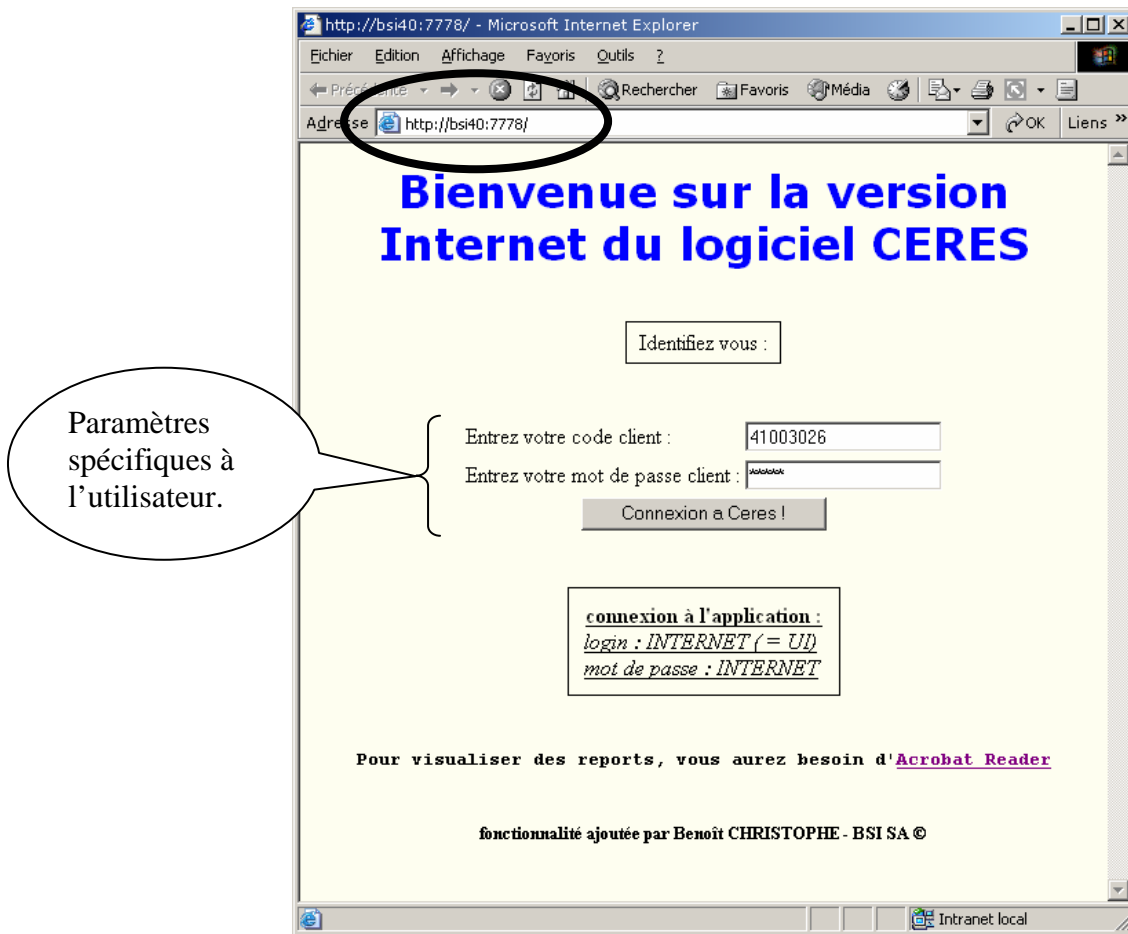


Figure 7 : Ecran d'accueil du navigateur au lancement de l'application

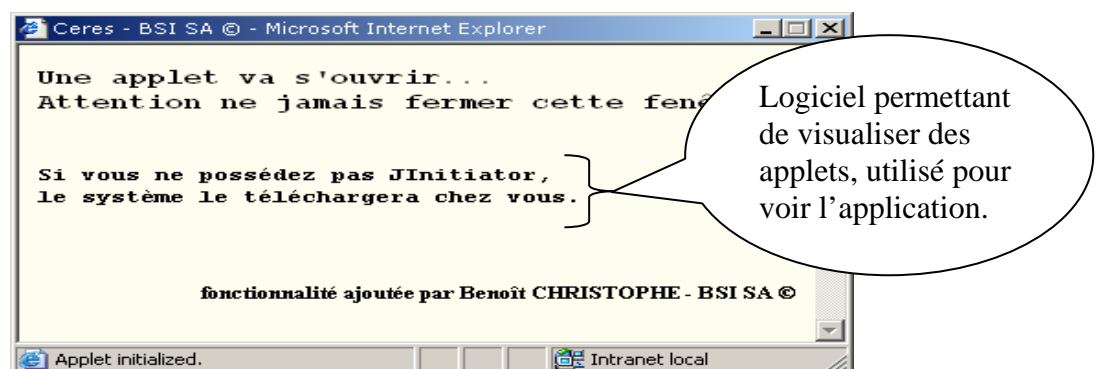


Figure 8 : Fenêtre commandant l'ouverture de l'applet



Figure 9 : Ecran d'accueil de l'application et de connexion de l'utilisateur Internet



Figure 10 : Ecran de connexion renseignant sur la société de l'utilisateur

Droits du profil Internet



Figure 11 : Ecran d'accueil de l'application CERES

Le code client est grisé : l'internaute ne pourra consulter que les données le concernant.

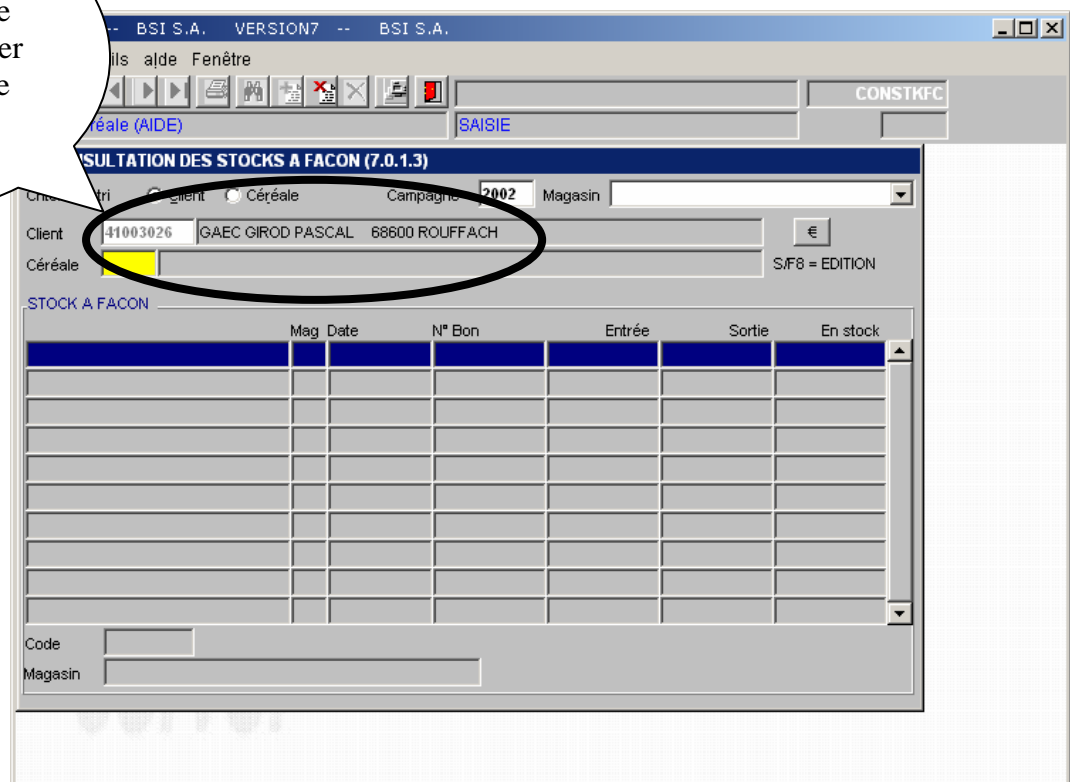


Figure 12 : Exemple de grille avec vue restreinte pour l'internaute

CERES -- BSI S.A. VERSION7 -- BSI S.A.

Action Outils aide Fenêtre

Code produit (AIDE) CONSULTATION 1

CONSULTATION DES STOCKS (7.0.2.7)

PRODUIT

Produit **268** TERN. 17/10/16+S VRAC Produit sans stock RECH. MATIERE

Unité TONNE Qté unitaire 0.001 Forme VRAC

Réapprovisionnement Minimum 0.000 Alerte 0.000 Quantité min. 0.000 Délai Jours

Maximum 9999.000

Matières actives

STOCK

Magasin	Stock Réel	A livrer	Disponible	A recevoir	Disponible	Livrés A	Livrés A
MAGASIN WESTHAL	0.000	0.000	0.000	0.000	0.000	0.000	0.000
SUNDHOFFEN	90.000	17.500	72.500	0.000	72.500	0.000	0.000
RANTZWILLER	0.000	0.000	0.000	0.000	0.000	0.000	0.000
DEPART RENO	0.000	0.000	0.000	0.000	0.000	0.000	0.000
LESSARD EN BRESSI	0.000	0.000	0.000	0.000	0.000	0.000	0.000
MOULIN DE LA ZORN	0.000	0.000	0.000	0.000	0.000	0.000	0.000
TOTAL	90.000	17.500	72.500	0.000	72.500	0.000	0.000

Prochain Arrivage

RAL RAR MVT

Figure 13 : Exemple de grille que l'utilisateur Internet verra via l'applet

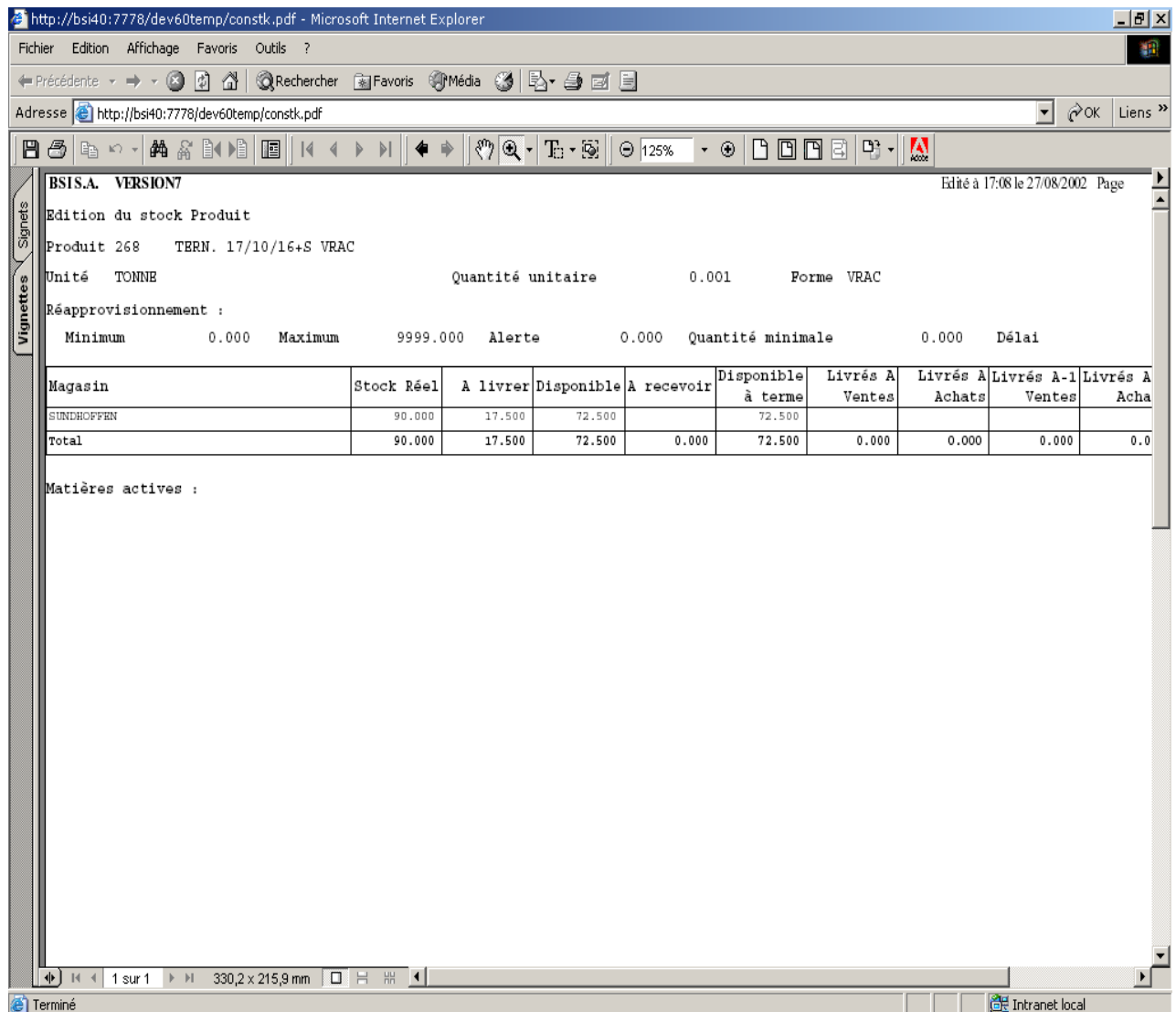


Figure 14: Exemple de document pdf envoyé au client

VI. Configuration matérielle et prix d'une machine SUN

Voici la configuration matérielle et le prix d'une machine SUN, qui pourrait servir de serveur d'application.

Bien entendu, cette configuration est énorme. Il faut donc également savoir si le serveur sera adapté à la taille de l'entreprise et s'il est nécessaire d'acheter une telle machine.

Select a Base Configuration				
	SMALL	MEDIUM	LARGE	EXTRA LARGE
UltraSPARC III Cu Processor	2 @ 900 MHz	4 @ 900 MHz	8 @ 900 MHz	8 @ 900 MHz
External Cache per Processor	8 MB	8 MB	8 MB	8 MB
Memory	4 GB (16 @ 256-MB DIMMS)	8 GB (32 @ 256-MB DIMMS)	16 GB (64 @ 256-MB DIMMS)	64 GB (64 @ 1-GB DIMMS)
10000 RPM FC-AL Disk Drive	6 @ 73 GB	6 @ 73 GB	6 @ 73 GB	12 @ 73 GB
DVD-ROM 10 Drive	1	1	1	1
Gigabit Ethernet Port	1	1	1	1
Ethernet Port	1 @ 10/100BASE-T	1 @ 10/100BASE-T	1 @ 10/100BASE-T	1 @ 10/100BASE-T
Serial Port	2 RS-232C/RS-423	2 RS-232C/RS-423	2 RS-232C/RS-423	2 RS-232C/RS-423
Power Supply	3 @ 1500 Watt	3 @ 1500 Watt	3 @ 1500 Watt	3 @ 1500 Watt
Solaris 8 Server License	Yes	Yes	Yes	Yes
Ships Within	8 business days	8 business days	8 business days	8 business days
List Price	\$34,995.00	\$59,995.00	\$99,995.00	\$249,995.00

Etant donné que ce projet n'en est qu'au stade initial, le serveur d'application (niveau logiciel) a été installé sur la machine sur laquelle j'ai travaillé, c'est à dire un Pentium 4 avec 384 Mo de mémoire vive.