



Université du Havre



CAF de Rouen

Rapport de Stage à la CAF de Rouen

Gestion des Habilitations

Étudiant : B. Julien

Formation : Informatique 2^{ème} année

Tuteur de stage : M Dabancourt
Tuteur en entreprise : Mme Dupond

29 mars 2004 / 4 juin 2004

Remerciements :

Je tiens avant tout à remercier Madame Dupond pour l'aide qu'elle m'a apportée en Visual Basic ainsi que dans l'élaboration du programme. Je tiens aussi à remercier Monsieur Durand Responsable Informatique pour m'avoir accueilli au sein du service informatique, ainsi que l'ensemble du personnel du service informatique pour l'accueil qu'il m'a réservé.

Je tiens également à remercier Monsieur Dupont Directeur de la CAF de Rouen et Madame Dupuis Directeur des Ressources Humaines et de la Logistique pour avoir approuvé mon entrée au sein de la Caisse d'Allocations Familiales pour la durée de mon stage.

Je tiens enfin à remercier M Dabancourt pour avoir encadré le stage que j'ai effectué à la CAF de Rouen.

Sommaire

1. Présentation de l'entreprise.	2
1.1. Les grandes dates.	2
1.2. Présentation de la CAF de Rouen.	2
1.3. La place du service informatique au sein de la CAF	2
1.3.1. Le parc informatique	2
2. Sujet du Stage	3
3. Le Déroulement du stage	5
3.1. Le Fonctionnement du programme	5
3.1.1. Mise à jour de la base	5
3.1.2. Le lancement du programme	5
3.1.3. Modifier une habilitation	5
3.1.4. Les options de la GRH	7
3.2. Programmation.	10
3.2.1. L'accès à la base de données en Visual Basic	10
3.2.2. La connexion des utilisateurs	11
3.2.3. Crystal Report	12
3.3. Application supplémentaire	16
3.4. Les demandes d'interventions	16
4. Conclusion	17
4.1. Améliorations futures	17
4.2. L'apport du stage	18
5. Index et Bibliographie	19
5.1. Index	19
5.2. Bibliographie.	20
5.3. Liste des Figures.	20

1. Présentation de l'entreprise.

Cette partie a pour but de présenter succinctement la Caisse d'Allocations Familiales (CAF) au travers des grandes dates de cette institution et aussi de présenter rapidement la CAF de Rouen

1.1. Les grandes dates.

- 1932 : Création de la première prestation d'allocation familiale.
- 1945 : La sécurité sociale est créée.
- 1967 : C'est la Caisse National des Allocations Familiales (CNAF) qui est créée par les ordonnances de 1967.

1.2. Présentation de la CAF de Rouen.

La Caisse d'Allocations Familiales (CAF) est un organisme privé qui gère des fonds publics. La CAF de Rouen compte plus de 300 employés, répartis en 39 services. En France, 123 caisses sont chargées de gérer les prestations familiales. La CAF de Rouen est une des quatre CAF de la Seine Maritime.

La CAF de Rouen gère 107 000 allocataires. Elle verse des prestations pour 113 884 enfants. En 2003, les prestations familiales versées ont représenté un montant de 448 670 605 €

Par jour et en chiffre, la CAF c'est :

- En réception
 - 525 allocataires
 - 1520 communications téléphoniques
 - 2220 courriers
- En traitement
 - 2540 pièces
- En expédition
 - 3940 courriers

1.3. La place du service informatique au sein de la CAF

Le service informatique est le cœur de la CAF de Rouen, en effet tous les documents reçus par courrier sont traités informatiquement. Ils sont scannés et envoyés sur des serveurs qui stockent ces documents au format image. Ensuite lorsqu'un allocataire se présente à la CAF, le technicien chargé de l'accueillir à juste à ouvrir le dossier sur son ordinateur. Il est évident que si un problème survient au service informatique la totalité de la CAF le ressent.

1.3.1. Le parc informatique

Le parc informatique de la CAF de Rouen se compose de plus de 300 stations et un peu plus de 20 serveurs. La configuration des machines commence au 633 MHz pour aller jusqu'au 2.6 GHz, j'étais d'ailleurs équipé d'une de ces machines. Le système d'exploitation Windows NT 4.0 équipe les stations de travail et les serveurs. On peut noter la présence d'un serveur Unix qui gère la connexion avec le centre régional de traitement informatique situé à Rennes. Tous les ordinateurs sont reliés au réseau local.

2. Sujet du Stage

Le programme créé est un programme de gestion des demandes d'habilitations. Celui-ci doit permettre aux chefs de service de sélectionner rapidement les applications que les agents sont autorisés à utiliser, il doit également mettre en page rapidement la demande sans que le chef de service n'est à intervenir.

Un programme déjà existant, appelé Odyssee, gérait déjà les habilitations. Néanmoins celui-ci était mal sécurisé et complexe d'utilisation. Dans ce but, je me suis vu confier la tâche de développer un nouveau programme plus simple et plus sécurisé qu'Odyssee.

Les habilitations des agents sont en fait les programmes qu'ils ont le droit d'utiliser. Ils apparaissent dans une barre de tâches. Cette barre de tâches est gérée par le PTMF (Poste de Travail MultiFonction). Celui-ci gère toutes les habilitations des utilisateurs. Le programme développé n'intervient que dans la partie administrative de la gestion des habilitations, c'est à dire, l'étape ou celles-ci sont approuvées par la Direction.

Le service GRH (Gestion des Ressources Humaines) doit contrôler régulièrement les demandes de modification faites par les chefs de service. Dans le cas où une demande a été formulée, il leur suffit d'imprimer le document contenant la liste des habilitations, puis de le soumettre à la direction qui se charge de vérifier la demande et de la faire signer par l'Agent et le Directeur.

Il est à noter que différentes autorisations sont données aux utilisateurs pour utiliser un programme. Par exemple, il y a pour les applications développées par la CNAF des autorisations utilisateur, vérificateur et administrateur. Un agent ne peut avoir qu'une seule autorisation par application. Un agent ayant une autorisation administrateur peut utiliser le programme comme un utilisateur, tout en ayant des fonctionnalités supplémentaires.

Pour revenir sur le PTMF, il n'existe aucune liaison entre le PTMF et le programme créé dans le cadre de mon stage. Il n'en existait d'ailleurs pas entre Odyssee et le PTMF. N'ayant pas eu accès au code source du PTMF et n'étant pas autorisé à mettre en place cette solution je ne me suis donc que très peu intéressé au problème. En fait la difficulté réside dans la compatibilité de Visual Basic et du langage utilisé par le PTMF qui est "Progress".

Il a été décidé lors de la conception du programme, d'utiliser Visual Basic 6.0. Visual Basic est un langage objet. La mise en place de l'interface est très intuitive. Il est fourni avec celui-ci un programme qui met en place de façon très rapide une interface graphique simple et efficace. Il a en mémoire la plupart des actions que l'on va générer sur un bouton ou n'importe quel autre objet. Ce qui permet de coder directement à l'intérieur de cette méthode les actions que l'on veut voir effectuer à la suite d'une action sur un objet. Pour faire une comparaison avec JAVA, il n'est pas utile de se demander quelle méthode on devra utiliser pour vérifier un clic sur un bouton. En effet, Visual Basic s'en charge. Ceci permet de se concentrer plus sur la partie conception que sur la partie interface. D'ailleurs, cette partie se met très rapidement en place avec un simple clic de souris.

Suite à l'autorisation de la CAF, j'ai pu joindre le code au dossier. Vous pourrez trouver celui-ci dans les annexes.

Il est à noter que le programme a été conçu pour être le plus portable possible. En effet, ce programme a pour vocation de remplacer Odyssee qui est trop complexe d'utilisation et qui est utilisé dans quelques CAF.

3. Le Déroulement du stage

3.1. Le Fonctionnement du programme

Ce programme se compose d'une base de données et d'un exécutable. La base de données se trouve sur un serveur indépendant et le programme va chercher les informations directement dans la base de données. Le programme est constitué de plusieurs fenêtres. Dans cette partie, nous expliquerons succinctement quelles modifications ont été apportées à la base de données existante. Nous nous attarderons sur le lancement du programme. Puis nous terminerons en parlant de la procédure pour modifier une habilitation ainsi que des options supplémentaires accessibles par la GRH.

3.1.1. Mise à jour de la base

La base de données était déjà existante. Il a donc fallu s'adapter à celle-ci en ajoutant des tables et des objets dans des tables sans supprimer d'éléments. En effet, cette base est utilisée par d'autres applications. La base de données existante doit pouvoir être mise à jour le plus facilement possible. Quelques tables se sont vu ajouter de nouveaux objets. La plupart des modifications a consisté en l'ajout de nouvelles tables qu'il est facile de relier entre elles sous Access.

3.1.2. Le lancement du programme

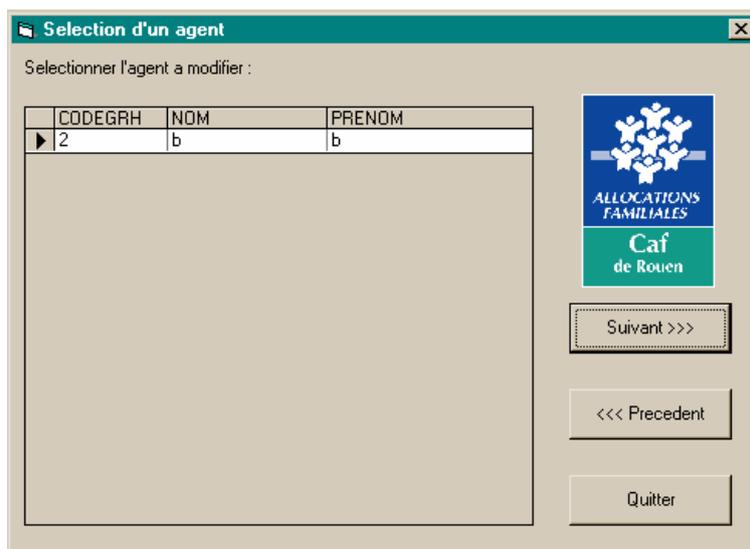
Lors du lancement du programme une reconnaissance de l'utilisateur Windows NT (tous les utilisateurs ont un identifiant Windows NT) est lancée par le biais d'une commande système se trouvant dans une fenêtre non affichée. L'utilisateur doit avoir été préalablement entré dans la base de données avec son numéro de branche, son code de service et son droit d'accès à l'ajout d'agent. Cet ajout se fait manuellement dans la base de données.

Le numéro de branche est en fait le code indiquant à quelle branche le service appartient, il faut savoir que la CAF de Rouen compte 5 branches. Ce numéro n'est pas primordial dans l'application développée, néanmoins il sera utile dans une application qui sera développée ultérieurement à partir de la base de données par le service de développement informatique. On m'a donc dans ce but, demandé de mettre en place ces données dans la base.

Le droit à l'accès «ajout_agent » est en fait un booléen placé dans la base de données. Celui-ci définit en fait si la personne a le droit de modifier ou d'ajouter un agent. De plus, il permet aussi d'avoir accès à l'affichage des habilitations de tous les agents. Seul le service GRH a besoin de cette option, c'est pour cela que seul le service GRH a accès à une fenêtre d'option au lancement. Les autres utilisateurs sont directement envoyés à la page des employés appartenant à leur service. Le code service est un code référençant chaque service dans la base de données.

3.1.3. Modifier une habilitation

L'utilisateur choisit l'agent qu'il souhaite modifier ou visualiser par le biais d'une liste que vous pouvez voir ci-dessous (pour des raisons de confidentialité la totalité des agents de la CAF a été retiré des captures d'écrans).

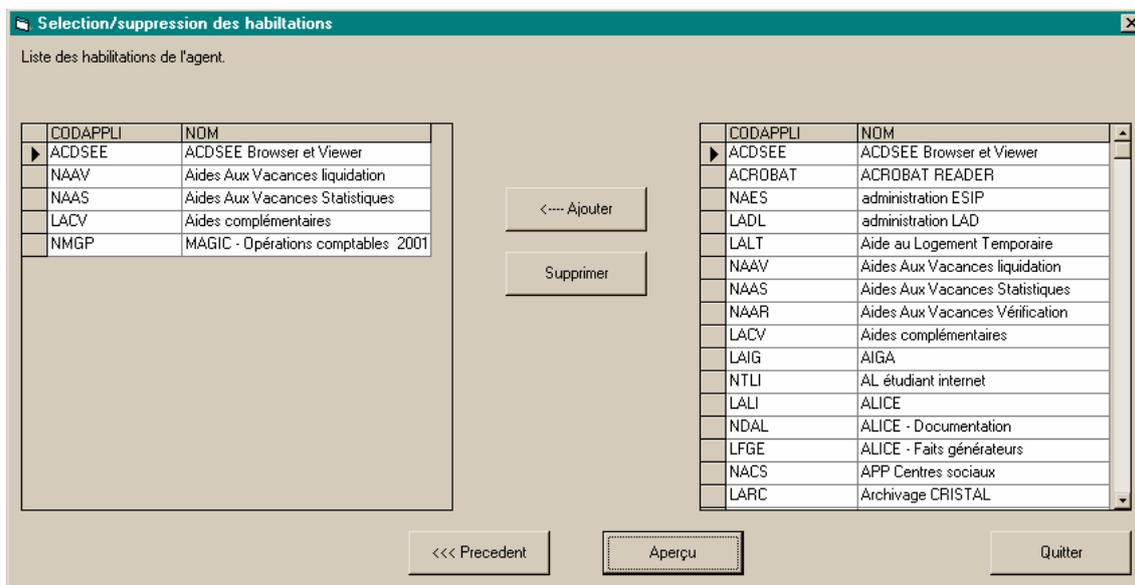


1 : captures d'écrans

Le programme met à jour en direct les données ce qui permet de visualiser immédiatement quelles applications sont accessibles par l'agent.

Une fois les applications ajoutées ou supprimées, le chef de service a la possibilité de visualiser les applications qu'il a sélectionné grâce à une fenêtre Crystal Report. Crystal Report est un utilitaire qui permet de mettre en page facilement les données extraites d'une base et de les afficher à l'écran. Il regroupe à peu près toutes les fonctionnalités d'un traitement de texte. De plus, une fois compilée la feuille Crystal Report est totalement indépendante, il n'est donc pas nécessaire d'installer celui ci sur la machine ou le programme sera exécuté.

Pour visualiser la feuille Crystal Report il suffit au chef de service de cliquer sur "Aperçu".



2 : la feuille Crystal Report

Afin de bien différencier les deux types de document, l'imprimé édité par un chef de service est très sommaire. En effet, celui ci n'est pas la demande définitive d'habilitation. Cette dernière ne peut être imprimée que par la GRH. Il s'agit en fait d'un brouillon ou le chef de service pourra vérifier que ces demandes de modifications d'habilitations sont correctes. Il est

à noter qu'à la suite de plusieurs entrées manuelles dans la base de données par le biais d'Odyssée (application qui permettait cette fonctionnalité), certaines applications sont présentes dans la liste des habilitations mais ne le sont pas sur la page Crystal Report. En effet, certaines autorisations d'applications entrées dans les profils ne correspondent pas aux autorisations possibles des applications. Nous reviendrons plus en détail sur ce point dans la partie consacrée à Crystal Report.

Une gestion des profils a également été développée. Il sera possible de gérer un ou plusieurs profils type par service. Lorsqu'un profil type existe pour un service, si l'agent sélectionné ne dispose d'aucune application, le chef de service verra s'afficher une fenêtre avec la possibilité de sélectionner un profil type. Dans un service les agents ont sensiblement les mêmes autorisations. Cette fonctionnalité permettra au chef de service de gagner du temps dans sa demande d'habilitation.

3.1.4. Les options de la GRH

La GRH a accès à une rubrique spéciale qui lui permet d'ajouter des agents à la base de données. Un agent est constitué d'un code GRH, d'un nom, d'un prénom, d'un code service, d'un code utilisateur, d'un code fonction, d'une date d'entrée à la CAF, d'une date de fin de contrat pour les CDD, d'une date de modification d'habilitation ainsi que de trois booléens définissant si ses habilitations sont en attente de validation, validées et modifiées.



3 : page d'accueil de la GRH

Ces quatre derniers objets ont été ajoutés pour faciliter le travail de recherche de la GRH. Pour l'impression des habilitations en attente de validation, une rubrique a été créée pour afficher uniquement ce type d'habilitation. Au moment de quitter la section, l'utilisateur se voit demander s'il a réussi à imprimer. Si l'agent de la GRH répond positivement, alors l'agent n'apparaîtra plus dans la section des agents en attente de validation. Il n'est pas possible de s'assurer que l'impression a été correctement réalisée dans le programme pour la simple raison qu'un problème technique pourrait survenir au moment de l'impression sans

qu'un message d'erreur n'arrive sur l'ordinateur. Pour éviter ce désagrément c'est la personne qui imprime les documents qui s'assure du bon déroulement de cette étape.

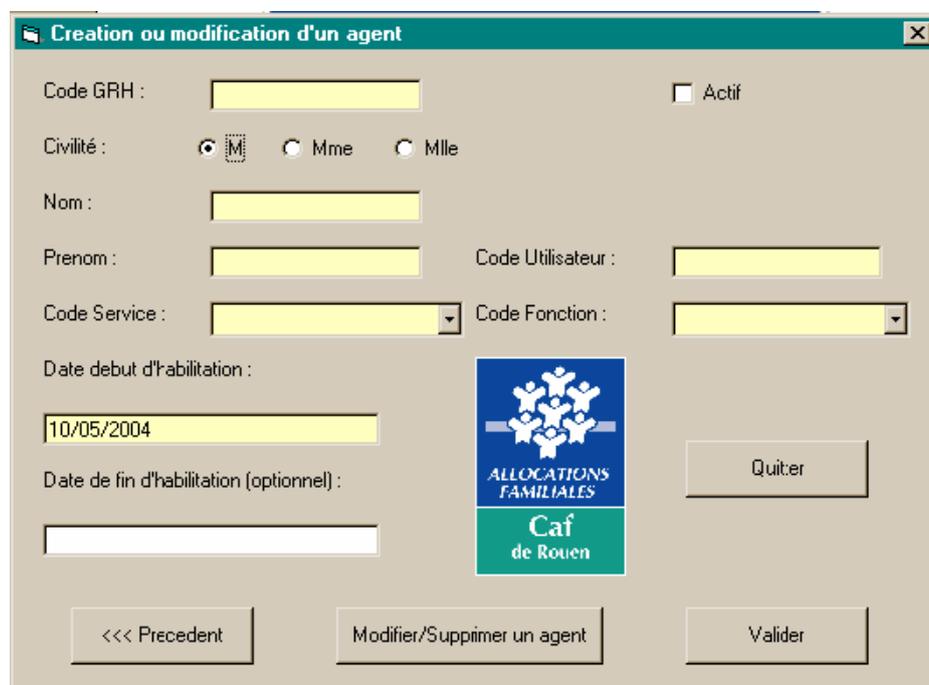
Nous pouvons aussi voir la présence d'un bouton nommé « Valider une demande ». Dans la pratique, une fois la demande imprimée un booléen est changé dans l'enregistrement de l'agent modifié, ce qui permet de l'afficher dans les demandes à valider. Une fois la demande validée un autre booléen sera modifié ce qui permettra d'afficher la demande dans une fenêtre Crystal Report qui sera utilisée dans l'aide à l'installation pour le service informatique (Cf.3.3).

Il faut préciser que si la demande est refusée, le chef de service est chargé de modifier lui-même les habilitations de son agent et de renvoyer sa demande à la GRH. Il faut préciser que le refus d'une autorisation est extrêmement rare.

Enfin, pour conclure sur ce problème de conception on peut noter qu'Odyssee fonctionnait de la même façon au niveau des autorisations. Toutefois, le nouveau programme impliquera plus les chefs de service. En effet, auparavant, les chefs de service appelaient la GRH pour demander l'ajout d'une application pour un agent et, c'était la GRH qui s'occupait de toutes les demandes.

La GRH a la possibilité de visualiser toutes les habilitations. Elle peut également visualiser les habilitations d'un agent en le sélectionnant dans une liste. Cette liste comporte la dernière date de modification de l'agent. Il est à noter que cette date ne figure pas sur les documents que la GRH imprime, car c'est la date de l'impression qui a été intégré au document pour des raisons administratives.

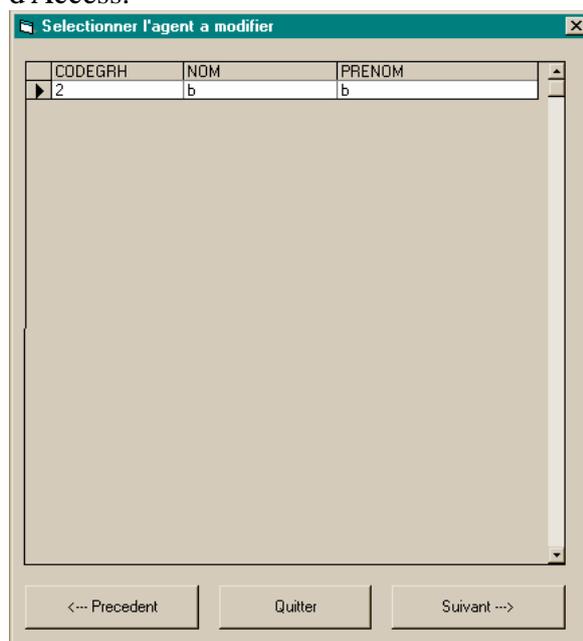
L'ajout d'un agent est uniquement accessible par la GRH. En effet, seule la GRH est habilitée à gérer les agents de la CAF.



4 : Ajout d'un agent

La section modification ressemble beaucoup à la section ajout. Toutefois, le code GRH ne peut pas être modifié. En effet, celui ci est attribué à un agent et à un seul, pour toute la durée ou il sera à la CAF. De plus, le code GRH est une clé primaire utilisée dans la table profil2 (table regroupant l'ensemble des autorisations des agents de la CAF avec le code d'application et le code d'autorisation sur ces mêmes applications, la table profil1 quant à elle

regroupe les autorisations comptables le programme ne devait se charger que de la gestion des habilitations informatiques). La modification du code GRH est bloquée par une sécurité d'Access.



5 : Sélection d'un agent dans la liste

Code GRH : 2 Actif
Civilité : M Mme Mlle
Nom : b
Prenom : b Code Utilisateur : d
Code Service : INFORMATIQUE Code Fonction : Documentaliste
Date debut d'habilitation : 7/05/2004
Date de fin d'habilitation (optionnel) :
ALLOCATIONS FAMILIALES Caf de Rouen
Valider Annuler Supprimer l'agent Quitter

6 : Modification ou suppression d'un agent

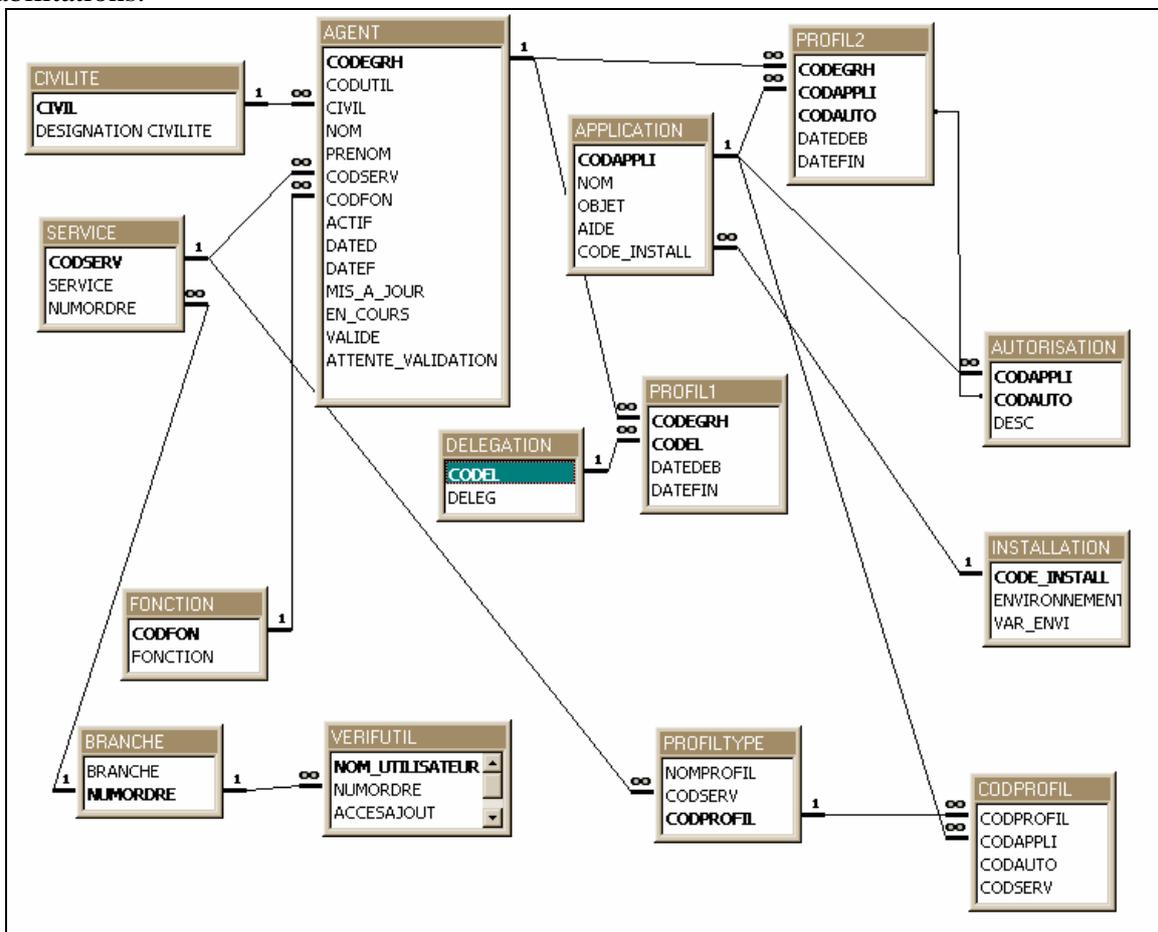
La section modification d'un agent permet aussi de supprimer un agent après sélection de celui ci dans une liste d'agent. Une confirmation de suppression est demandée. En cas de suppression, toutes les lignes ou le code GRH de l'agent apparaît sont supprimées en cascade, de la base de données.. Cette option est activée dans la base de donnée Access.

3.2. Programmation.

Dans cette partie nous nous attarderons plus en détail sur l'aspect programmation et développement du programme. Dans une première partie nous traiterons de l'accès à la base de données que Visual Basic propose. Puis nous parlerons un peu plus en détail de la connexion des utilisateurs et le fonctionnement du programme. Enfin nous concluons cette partie en parlant de Crystal Report.

3.2.1. L'accès à la base de données en Visual Basic

Comme indiqué dans l'introduction, Visual Basic permet de se concentrer sur la conception et sur la programmation plutôt que sur le codage de l'interface qui peut se révéler fastidieux avec certains langages de programmation. L'essentiel du programme repose sur l'accès à la base de données qui regroupe tous les renseignements nécessaires à la gestion des habilitations.



7 : Schéma de relation sous accès

Visual Basic utilise des objets qui stocke en mémoire les tables de la base de données. Ces objets sont les recordset. Ils permettent d'agir sur la base de données. Pour sélectionner une ligne particulière ou un ensemble de ligne il suffit d'utiliser une méthode qui prend en paramètre une requête SQL et qui sélectionne uniquement les lignes répondant aux critères de la requête SQL. Pour y accéder, il est nécessaire d'avoir un objet de type database qui donne l'adresse de la base de données et ensuite l'objet recordset va chercher la table passée en paramètre dans la base définie par l'objet database.

Il existe d'autres accès à la base de données que les recordset et les database. L'autre objet souvent utilisé dans le programme est en fait un objet de type data. Il permet de faire le lien entre des grilles, des listes et des menus déroulant accédant directement à la base de données. Il est d'ailleurs intéressant de préciser qu'il est nécessaire de verrouiller les accès data n'ayant pas à modifier la base de données en écriture car la mise à jour est automatique si un changement survient dans un objet étant lié à la base de données. Par contre, lorsque la base de données doit être modifiée, il est préférable d'utiliser dans ce cas un objet recordset faisant appel à un contrôle database qui ne s'exécute que lors d'une validation volontaire de l'utilisateur, ces deux éléments étant codés en dur dans le programme.

Le contrôle data permet malgré tout de simplifier grandement la programmation puisque presque tous les renseignements nécessaires sont entrés directement dans le tableau de propriété du contrôle data, ce qui permet de lier très rapidement un objet à la base de données. L'adresse de la base, la requête SQL de sélection des données et d'autres informations sont dans cette fenêtre de propriété. Néanmoins le tableau de propriété ne permet pas d'effectuer des sélections en fonctions de variables initialisées dans le programme. Pour pouvoir sélectionner en fonction de ces variables, il est nécessaire d'utiliser une méthode qui remplacera la requête de sélection initiale par une nouvelle requête. Cette nouvelle requête, dans le cas de la sélection des habilitations d'un agent par exemple devra se faire au moment du chargement de la fenêtre.

Il est à noter que malgré tous les avantages apportés par Visual Basic pour l'accès à la base de données, il existe un problème sur la syntaxe des données à rentrer. En effet, si dans un nom ou tout autre élément issu de la base de données il y a le caractère ' la requête SQL ne fonctionnera pas. En fait Visual Basic utilise le caractère ' pour définir s'il s'agit ou non d'une chaîne de caractère. Si ce caractère est présent dans une variable, Visual Basic fermera la chaîne de caractères et ceci provoquera une erreur. Pour éviter ce problème, il a été nécessaire de verrouiller cette touche par l'intermédiaire de Visual Basic en indiquant que dans le cas où le code ASCII de cette touche était envoyé celui-ci ne doit pas être pris en compte. Cette méthode a été placée dans tous les endroits où l'utilisateur peut entrer une information manuellement.

3.2.2. La connexion des utilisateurs

Revenons maintenant aux demandes du programme. Lors du lancement de celui-ci, un objet recordset analyse le nom de l'utilisateur NT et détermine s'il est ou non autorisé à lancer le programme. Si celui-ci n'est pas dans la base (ce qui est normalement impossible puisque seuls les personnes habilitées à utiliser le programme le voit dans leur barre d'habilitation), un message d'erreur apparaît précisant que cet utilisateur n'est pas autorisé à utiliser l'application. Pour récupérer le nom NT il a fallu accéder à une DLL de Windows. Il est utile de décrire pourquoi le choix d'utiliser le login NT a été fait, puisque c'est sur ce nom que repose l'accès au programme.

D'un point de vue pratique, il est impossible d'avoir deux noms utilisateurs identiques sur le réseau. C'est cet identifiant qui permet à un utilisateur d'accéder au réseau de l'entreprise. Seul les administrateurs peuvent créer un compte utilisateur. Pour se connecter un utilisateur doit donner son nom et son mot de passe. Il n'était donc pas nécessaire de développer un système plus important du point de vue sécurité. La première version du programme reposait elle sur un système mot de passe et login inclus dans la base de données. Le mot de passe était caché grâce à une option d'Access. Ceci permettait d'empêcher un utilisateur malveillant de trouver les mots de passe en ouvrant simplement la base de données qui se trouve sur le serveur.

Néanmoins la demande de rendre le plus automatique possible le programme et de limiter au maximum les interventions de l'utilisateur dans celui-ci, on est obligé d'utiliser la solution du code utilisateur NT. Une fois l'utilisateur identifié, il accède à la liste des membres de son

service ou s'il est de la GRH il est orienté vers la fenêtre décrite dans la partie précédente. L'affichage des agents s'effectue en fonction du code de service de l'utilisateur connecté, le code service étant une des composantes constituant la table « verifutil » dans la base de données. Une fois la liste des agents affichée l'utilisateur sélectionne l'agent, ce qui provoque le stockage de cette valeur dans une variable qui va ensuite afficher la totalité des habilitations de l'agent en fonction du code GRH. Pour effectuer la sélection de la colonne où se trouve le renseignement, Visual Basic dispose de méthodes permettant la sélection de l'élément d'une des colonnes de la ligne sélectionnée. Compte tenu de l'encombrement de la page de sélection des habilitations, il a été décidé de réduire au maximum les renseignements disponibles sur cette page. Pour visualiser la totalité des renseignements il suffit simplement de cliquer sur la commande visualisation. Ceci provoque l'affichage d'une fenêtre crystal report et tous les éléments utiles à la mise en place d'une autorisation informatique sont affichés.

3.2.3. Crystal Report

3.2.3.1. Le fonctionnement

Il paraît important maintenant de parler du fonctionnement de Crystal Report un peu plus en détail. Crystal Report comme indiqué précédemment permet de créer des pages de texte utilisant des données directement extraites d'une base de données. Avant tout il faut définir l'emplacement de la base de données puis ensuite indiquer quels champs de la base de données seront utilisés. Une fois cette étape effectuée, on doit définir des groupes qui sont en fait des parties de la page qui regrouperont les données extraites. Par exemple dans le cas du programme il a fallu établir deux groupes principaux. Un groupe pour le "Nom" et un groupe pour le "Code" GRH. Il faut faire très attention à la position d'un groupe par rapport à l'autre, par exemple si c'est le groupe "Nom" qui est à l'intérieur du groupe "Code" GRH et que la page change une fois que l'on passe à un autre nom on aura un problème. En effet, pour les personnes qui portent le même nom, nous n'aurons en fait qu'une seule page qui apparaîtra, avec les renseignements de la première personne et dans la partie affichant les habilitations nous aurons l'ensemble des applications de toutes les personnes portant le même nom. C'est pour cela qu'il faut changer de page une fois que le code GRH change puisque celui-ci est propre à chaque personne. De plus, ici le fait que le nom soit en haut permet d'effectuer un tri par ordre alphabétique.

N 1 : NOM - C					
N 2 : CODEGRH - C	AUTORISATIONS INFORMATIQUES				
	PROFIL UTILISATEUR DE L'AGENT				
	NOM	XXXXXXXXXXXXXXXXXXXXXXXXXXXX			
	PRENOM	XXXXXXXXXXXXXXXXXXXXXXXXXXXX			
	CODEGRH	-555 555 555			
	FONCTION	XXXXXXXXXXXXXXXXXXXXXXXXXXXX			
	SERVICE	XXXXXXXXXXXXXXXXXXXXXXXXXXXX			
	CODE UTILISATEUR	XXXXXXXXXXXX			
	AUTORISATIONS ACCORDEES PAR APPLICATION				
	NOM APPLICATION	CODE	DESCRIPTION	A partir du	Jusqu'au
Détails	XXXXXXXXXXXXXXXXXXXXXXXXXXXX	XXXXXX	XXXXXXXXXXXXXXXXXXXX	31/12/1999	31/12/1999
N 2 : CODEGRH - C	Le 31 Décembre 1999				
	L'agent :		Le Directeur :		
	<div style="border: 1px solid black; height: 40px;"></div>		<div style="border: 1px solid black; height: 40px;"></div>		

8 : fenêtre de configuration de crystal report

La partie détail est la partie où les habilitations vont être affichées à chaque occurrence du code GRH dans la base de données. Ceci permet d'avoir toutes les habilitations d'une seule personne à la suite.

L'avantage de Crystal Report réside dans le fait qu'il est très rapide de mettre en page des données issues d'une base.

AUTORISATIONS INFORMATIQUES

PROFIL UTILISATEUR DE L'AGENT

NOM titi

PRENOM tata

CODEGRH 2

FONCTION Technicien

SERVICE INFORMATIQUE

CODE UTILISATEUR tatit

AUTORISATIONS ACCORDEES PAR APPLICATION

NOM APPLICATION	CODE	DESCRIPTION	A partir du : Jusqu'au :
Aides Aux Vacances liquidation	ADM000	Administrateur	7/05/2004
ACROBAT READER	DEF000	Utilisateur	7/05/2004
ACDSEE Browser et Viewer	USR000	Utilisateur	7/05/2004

Le 7 Mai 2004

L'agent :

Le Directeur :

9 : Exemple de page imprimée par la GRH

Néanmoins malgré tous ces avantages Crystal Report connaît quelques petits défauts. En effet, lors de l'élaboration d'une page il me suffisait d'agrandir très légèrement une case pour voir le nombre de calcul sur la base de données décupler. Le seul moyen de revenir à une configuration correcte en temps de calcul était de charger une sauvegarde du fichier Crystal Report quand il marchait correctement. Je n'ai malheureusement pas trouvé de solution à ce problème et je ne sais toujours pas à quoi est dû ce bug (le nombre de combinaisons à vérifier était normalement de 14 000, avec le bug il passait à plus de 370 000 calculs ce qui prenait près de 40s au lieu de 3s).

3.2.3.2. Problème rencontré pour l’affichage des habilitations

Le problème cité précédemment concernant le problème d’affichage de certaines applications est dû au fait que la requête SQL dans Crystal Report est plus sélective que la requête utilisée pour la fenêtre d’ajout des habilitations. La requête de la grille d’affichage des habilitations ne sélectionne que le nom des habilitations en fonction du code GRH et du code d’application dans la table profil alors que la fenêtre Crystal Report sélectionne en plus la totalité des renseignements concernant les habilitations de l’agent sur cette application, pour que le service informatique puisse ajouter les applications au PTMF. Le problème vient du fait que lorsque l’on sélectionne les applications en fonction de leur code et que l’on souhaite afficher les autorisations, toutes les autorisations possibles sur le programme apparaissent ce qui est normal puisque la sélection est faite à partir du code d’application. Pour palier à ce petit désagrément, il est nécessaire de vérifier que les autorisations de la table profil2 correspondent avec celles de la table autorisation. Suite à des entrées manuelles, certains profils se retrouvent avec des autorisations qui n’existent pas dans la table autorisation. C’est pour cette raison que certaines applications entrées manuellement dans la table profil2 ne sont pas affichées lors de l’ouverture de Crystal Report puisque celles ci sont exclues par la requête SQL. Il faut donc revenir en arrière et supprimer l’application de la liste des applications de l’agent puis de la rajouter. L’ajout d’une autorisation va chercher directement dans la table autorisation celles qui sont affiliées au programme et ainsi corrige dans la table profil2 le problème.

Il aurait été possible de le faire en automatique mais cela aurait été une perte de temps sur le projet pour seulement une poignée d’applications qui ont été mal gérées. Il est aussi à préciser que dans le programme, la base de données n’est plus accessible en édition pour éviter de nouveau ce problème. Il est à noter qu’un agent ne peut avoir qu’une seule autorisation par d’application.

Odyssée était une application Access et lors de la programmation les développeurs avait décidé de laisser la possibilité aux agents d’accéder à la base de données. Ce choix a provoqué des problèmes au niveau de la correspondance entre les tables citées précédemment.

3.3. Application supplémentaire

Durant le stage il m'a été demandé de mettre au point un autre programme qui pourrait servir d'aide lors de la mise en place des programmes par le service informatique.

Pour les besoins de ce programme j'ai dû ajouter une table appelée "INSTALLATION" qui permet de regrouper les renseignements sur les « Environnements » nécessaires au fonctionnement des applications. Ici le terme environnement désigne simplement si l'application a besoin de Excel, de Word ou d'une autre application particulière pour fonctionner. Il a aussi été nécessaire d'ajouter divers renseignements supplémentaires dans les tables "APPLICATION" et "AGENT". Dans ce petit programme les agents du service informatique ont la possibilité de :

regarder comment on installe une application

ajouter une procédure d'installation

consulter la liste des agents ayant eu leur nouvelle demande d'habilitation validée

(il est à noter que la feuille d'habilitation arrive directement au service informatique après la validation).

La visualisation des applications ayant une procédure particulière d'installation s'effectue par l'intermédiaire d'une feuille Crystal Report qui affiche les procédures en fonction de chaque agent. Je ne reviendrais pas sur la programmation de cette application puisque celle ci utilise les mêmes fonctions que le programme principal et il n'est donc pas nécessaire d'en expliquer le fonctionnement en détail. Vous pourrez trouver le code source de ce programme ainsi que celui du programme principal en annexe.

3.4. Les demandes d'interventions

Au cours du stage j'ai effectué plusieurs demandes d'interventions allant du remplacement de souris à la recherche de problèmes matériels en passant par l'installation d'écrans et de postes. Au cours de ces interventions, j'ai constaté que certains problèmes qui nous paraissent évidents ne le sont pas pour les utilisateurs.

4. Conclusion

4.1. Améliorations futures

Une des améliorations qui pourrait être apportée au programme serait de le relier directement à la base de données du PTMF. Cette option permettrait de retrouver pour chaque agent des autorisations identiques entre le PTMF et le programme développé. En effet, les demandes d'habilitations transmises par la GRH doivent être ressaisies manuellement, d'où des risques de divergences entre les deux programmes. Cette amélioration apporterait un gain de temps énorme pour la gestion des habilitations. Néanmoins n'ayant pas eu accès au code source du PTMF je n'ai pas pu essayer de mettre en place cette solution.

Une autre amélioration qui elle pourrait être intéressante est la gestion des habilitations comptables. Ceci ne rentrait pas en compte dans le programme puisqu'on m'a demandé de ne gérer que les applications informatiques. Les autorisations comptables sont en fait la liste des autorisations comme l'ajout d'argent sur un compte, l'émission de chèques et autres prérogatives propres au service comptabilité. Cela ne demanderait pas beaucoup de temps puisqu'il suffirait de récupérer les autorisations répertoriées dans la table profil1 qui regroupe les autorisations comptables.

La principale amélioration qui sera apportée sera la mise en place de l'application au niveau national si elle a lieu. Le programme a été conçu dans cette optique afin d'être le plus portatif possible. Tous les renseignements sont directement issus de la base de données et toutes les variables accédant à la base de données sont communes à toutes les CAF. Si une CAF souhaitait utiliser le programme, il lui suffirait de remplir la base de données avec les mêmes tables mais avec des renseignements différents. Néanmoins il est évident qu'il est nécessaire de garder le même schéma de table puisque le programme repose sur des requêtes utilisant des noms précis de variables de la table. Il est à noter qu'il serait aussi nécessaire de modifier le chemin d'accès à la base de données dans le programme, ceci ne prendrait que quelques minutes. L'application n'ayant pas encore été validée au moment de la fin de mon stage par le Comité de Direction des instructions pour l'installation ont été laissées pour la mise en place de l'application sur la base de données du réseau. Ces modifications consistent simplement à ajouter les tables de la base de données créées pour les besoins du programme à la base de données du réseau, d'ajouter des liaisons entre les tables et quelques variables à certaines tables.

4.2. L'apport du stage

Le stage m'a permis de connaître un peu mieux le monde de l'entreprise et de mieux comprendre le rôle de l'informaticien dans celle-ci. J'ai pu constater que le rôle de l'informaticien est un peu différent de la formation que l'on reçoit puisque à plusieurs reprises je suis parti en dépannage informatique pour des problèmes bénins. J'ai néanmoins passé près des trois quarts de mon temps à programmer et à vérifier les bugs possibles du programme. L'élaboration du programme m'a aussi permis de prendre conscience des difficultés de certains utilisateurs à utiliser des applications et cela m'a donc poussé à le modifier à de multiples reprises pour le rendre le plus simple et le plus efficace possible. La plupart de mon travail a consisté à travailler sur une base de données et les cours dispensés en première année m'ont été très utiles. Les cours de programmation par objet m'ont permis de m'adapter plus vite à un langage que je découvrais et ainsi à ne passer qu'une semaine à me former au langage. Ce stage a donc constitué une bonne expérience puisque j'ai pu appliquer les notions théoriques enseignées à la pratique sans trop de difficultés.

Index et Bibliographie

4.3. Index

Access : Système de base de données développé par Microsoft.

CAF : Caisse d'Allocations Familiales.

CNAF : Caisse Nationale d'Allocations Familiales. Elle assure le financement de l'ensemble des régimes de prestations familiales. Elle gère également un fonds d'action sanitaire et sociale.

Crystal Report : Application permettant de mettre sous forme de document des éléments issus d'une base de données.

JAVA : Langage de programmation objet.

PTMF : Poste de Travail MultiFonction. Il s'agit en fait d'une barre des tâches regroupant toutes les applications que l'agent peut exécuter.

Visual Basic : C'est un langage de programmation objet développé par Microsoft.

4.4. Bibliographie.

Visual Basic 5.0, Benoît BLIER, Institut Media Plus
MSDN Library : Visual studio 6.0, Microsoft
Le Programmeur : Visual Basic 6, Greg Perry, CampusPress

4.5. Liste des Figures.

1 : captures d'écrans	6
2 : la feuille Crystal Report	6
3 : page d'accueil de la GRH	7
4 : Ajout d'un agent.....	8
5 : Sélection d'un agent dans la liste	9
6 : Modification ou suppression d'un agent	9
7 : Schéma de relation sous accès	10
8 : fenêtre de configuration de crystal report	13
9 : Exemple de page imprimée par la GRH	14