

Stagiaire :
Bastien JORGE

Responsable pédagogique UTT :
Pierre BEAUSEROY

Branche : ISI
Année : 2014
Semestre : P14

Université
de technologie
de Troyes

Intégration de Business Objects dans les solutions logicielles Millésime CCAS version Web

Ce stage s'est déroulé dans l'entreprise *Implicit*, au sein du service développement.

Il a consisté à intégrer de manière la plus naturelle possible pour l'utilisateur final le progiciel d'informatique décisionnelle de l'entreprise SAP : *Business Objects* (BO). Le but est de fournir aux agents des CCAS (centre communal d'action sociale) une solution d'édition de statistiques.

Les différentes phases de travail successivement réalisées sont :

- Le remplacement de l'ancien outil de requêtage dans la solution web existante par une intégration de BO via la sollicitation de webservices ; le développement a été effectué en WebDev et javascript,
- L'ajout d'une surcouche cartographique à BO afin d'offrir aux travailleurs sociaux une interface aboutie d'aide à la décision.

L'enjeu de cette intégration est de participer à l'amélioration continue du progiciel de gestion vendu par *Implicit* aux CCAS.

Mots clés

Entreprise : **Implicit**
Lieu : **Saint-Martin sur le pré**
Responsable : **Pascal Baligout**

1. Conception
2. Services Marchands
3. Informatique
4. Gestion - Logiciels

Remerciements

Je tiens tout d'abord à remercier Richard THOUVENOT pour m'avoir offert l'opportunité unique de piloter un projet en relative autonomie durant mon stage, ainsi que celle d'avoir pu intégrer une entreprise au cadre aussi agréable que celui d'Implicit.

Je remercie tout particulièrement Pascal BALIGOUT, mon responsable, qui a effectué un suivi régulier de mes progressions. En me confiant ces missions, il m'a fait confiance sur tous les points et m'a donné carte blanche la plupart du temps.

Je tiens également à remercier Emmanuel VATTAT pour le suivi enrichissant qu'il a pu avoir sur mon stage, et pour avoir été aussi attentif à ce que mes différentes activités au sein de l'entreprise correspondent au mieux à mon projet professionnel.

Je veux remercier Pierre BEAUSEROY pour avoir pris sur son temps et être venu me visiter à Saint-Martin sur le Pré, et s'être assuré du bon déroulement de mon stage.

Je remercie également Amélie, Fabien, Steeve, Mickael, Grégory, Jérémy et Christophe, l'équipe d'analystes-développeurs, pour l'aide apportée au quotidien et l'ambiance très agréable tout au long de ces six mois.

Je veux aussi remercier Isabelle MAITRE pour son soutien régulier depuis mon entrée à l'Université de Technologie de Troyes et tout particulièrement durant ce stage.

Je remercie enfin Margaux et mes parents pour leur relecture attentive et l'aide à la rédaction de ce rapport.

Sommaire

I. Rapport

Introduction	4
1. Présentation de l'entreprise	6
1.1. Historique	6
1.2. Domaine d'activité	7
1.3. Structure interne	7
1.4. Modèle économique	9
1.5. Situation dans le groupe	11
2. Présentation du stage	13
2.1. Rappel du sujet	13
2.2. Présentation du logiciel CCAS Web	13
2.3. Présentation du projet	15
2.4. Planification de la mission et méthodologie de travail	19
3. L'intégration Business Objects	20
3.1. Organisation temporelle	20
3.2. L'intégration : choix de programmation et difficultés	21
3.3. Bilan de l'intégration et évolutions envisagées	35
4. L'implémentation de GeoBI	37
4.1. Organisation temporelle	37
4.2. Choix d'implémentation et difficultés rencontrées	37
4.3. Bilan de l'intégration et évolutions envisagées	45
Bilan	47
Enrichissement personnel	47
Difficultés rencontrées	47
Réponse à la problématique	48

II. Annexes

Première partie

Rapport

Introduction

Ce stage de 26 semaines s'est déroulé au sein de l'entreprise Implicit, une société de mise en place de solutions informatiques à destination des Centres Communaux d'Action Sociale (*voir 1.2 Domaine d'activité*), et filiale du groupe Chèque Déjeuner. Il a été effectué au siège social, à Saint-Martin sur le Pré (Marne), dans le cadre de mon cursus universitaire en tant qu'élève ingénieur de l'Université de Technologie de Troyes en Informatique et Systèmes d'Information.

J'ai été intégré dans l'équipe de développement de l'entreprise au poste d'analyste-développeur, sur **un projet d'amélioration continue de leur progiciel de gestion** — appelé *CCAS Web*. J'ai été affecté à l'intégration de *Business Objects* (solution SAP) dans leur logiciel, ainsi qu'à l'ajout d'une surcouche cartographique dans la solution de Business Intelligence.

J'ai ainsi pu m'investir dans l'ajout de fonctionnalités de leur programme de gestion en implémentant une nouvelle interface ; de ce fait, j'ai pu travailler sur chacune des phases du développement logiciel : de l'analyse et la recherche du réalisable, au débug après phase de tests, en passant par la rédaction d'un cahier des charges ou la production de code.

J'ai donc pu m'impliquer dans les phases de Recherche et Développement de la solution, ainsi que dans les phases de conception technique et de réalisation fonctionnelle.

À l'occasion de cette mission, j'ai ainsi utilisé et découvert divers outils sur des domaines très variés, tels que *Business Objects XI 4.1* (solution de Business Intelligence), *PCSoft Webdev 15* et *JQuery 1.11.0* (langages de programmation), *Git* (éditeur de texte avancé), *Balsamiq Mockups* (outil de maquettage), ou encore *JOSM* (logiciel d'édition cartographique).

Le choix de ce stage s'inscrit dans un projet professionnel basé sur mes expériences passées et un désir de découvertes.

En effet, la seule vision du monde professionnel dont j'avais pu bénéficier auparavant concerne un stage de 10 semaines effectué en fin de cursus de DUT informatique dans une association qui ne disposait pas de SI ; bien que très intéressant, il m'a fallu m'auto-former et m'adapter aux différentes problématiques rencontrées par un développeur. L'entreprise Implicit étant éditrice de logiciels, je savais que je serai ici supervisé par des responsables directement concernés par ma formation.

Par ailleurs, la mission proposée regroupait la quasi-totalité du programme étudié à l'Université de Technologie de Troyes durant le tronc commun de branche, et me permettait d'avoir un aperçu de ce que proposaient les deux filières qui m'intéressaient pour la suite de mes études — à savoir Management de Projet Logiciel et Management

du Système d'Information. Elle avait aussi l'avantage de me faire découvrir la pratique — qui ne peut être étudiée à l'UTT — du développement logiciel au sein d'une équipe, d'en comprendre la dynamique et de me faire progresser au niveau technique.

L'objectif de ce rapport est de présenter le déroulement de mon stage effectué au sein d'Implicit.

Dans un premier temps, nous allons nous intéresser à l'entreprise en elle-même : son positionnement sur le marché ainsi que son fonctionnement interne, atypiques mais qui présentent des particularités intéressantes.

Après une brève description du stage, nous verrons plus en détail son contexte, ainsi que les différents enjeux de ces six mois.

Enfin, nous nous intéresserons successivement aux deux missions sur lesquelles j'ai été positionné. Nous aborderons ainsi des éléments techniques et surtout des choix de conception.

À l'issue de ce document, nous effectuerons un bilan du stage : mon vécu personnel et la mise en rapport avec mon projet professionnel, ainsi que les difficultés auxquelles j'ai pu être confronté dans la globalité du stage.

1. Présentation de l'entreprise

Implicit, ainsi que le groupe Chèque Déjeuner dont elle fait partie, sont deux structures atypiques. Leur particularité — tant structurelle qu'à travers les valeurs qu'elles véhiculent — ont permis de rendre mon stage d'autant plus intéressant.

1.1. De la branche d'une start-up à la filiale d'un groupe multinational, un parcours atypique

L'histoire de la société *Implicit* commence en 1983 : c'est cette année que Jacky VAUTHIER fonde la SARL *JVS*, une entreprise de création de logiciels de gestion territoriale, et embauche très vite Richard THOUVENOT. Elle devient rapidement une référence dans son domaine, le marché des collectivités territoriales.

En 1988, *JVS* lance sa branche de développement spécialisée dans les logiciels de gestion à destination des Centres Communaux d'Action Sociale¹ (CCAS). Cette branche deviendra indépendante en 1998, une jeune entreprise du nom de *JVS CCAS*. A cette époque, la petite start-up *JVS* est devenue un groupe, propriétaire de cette nouvelle entité.

JVS CCAS rachète en 2004 l'un de ses concurrents d'alors, *IMPLICIT Solutions*, afin d'étendre son marché. Ils forment ensemble *JVS-IMPLICIT*.



FIGURE 1. – Logo de la société JVS-IMPLICIT, et logo actuel d'Implicit

Enfin, suite à un désir d'unifier ses solutions sociales (*voir 1.4 Modèle économique*), le Groupe Chèque Déjeuner souhaite acquérir cette filiale en 2010 ; elle reprend alors le nom d'*Implicit* et Richard THOUVENOT reste à sa tête².

Aujourd'hui, *Implicit* est leader dans l'informatisation des CCAS et accompagne plus de 1 000 clients (55% de parts de marché sur les collectivités de 10 000 à 50 000 habitants).

1. Un Centre Communal d'Action Sociale est un établissement public communal intervenant dans les domaines d'aides sociale (*voir 1.2*).

2. Richard THOUVENOT est devenu directeur général à l'époque de *JVS CCAS*.

1.2. Le domaine d'activité : qu'est-ce que le CCAS

« Un Centre Communal d'Action Sociale (CCAS) est, en France, un établissement public communal intervenant principalement dans trois domaines :

- L'aide sociale légale qui, de par la loi, est sa seule attribution obligatoire
- L'aide sociale facultative et l'action sociale, matières pour lesquelles il dispose d'une grande liberté d'intervention
- L'animation des activités sociales »

(Source : Wikipedia)

Plus simplement, un CCAS est l'organisme responsable de redistribuer des aides aux administrés en difficulté, qu'il s'agisse d'un besoin financier ou matériel, ponctuel ou régulier (pour trouver un logement, aider exceptionnellement à payer une facture d'électricité...). Et comme la majorité des structures, sa gestion passe par des outils informatiques.

La société Implicit est spécialisée dans l'édition de solutions logicielles à destination de ces organismes.

1.3. Mon poste dans la société

A l'heure actuelle, l'entreprise Implicit est une PME qui compte 58 salariés et 9 agences en France (voir Figure 2).

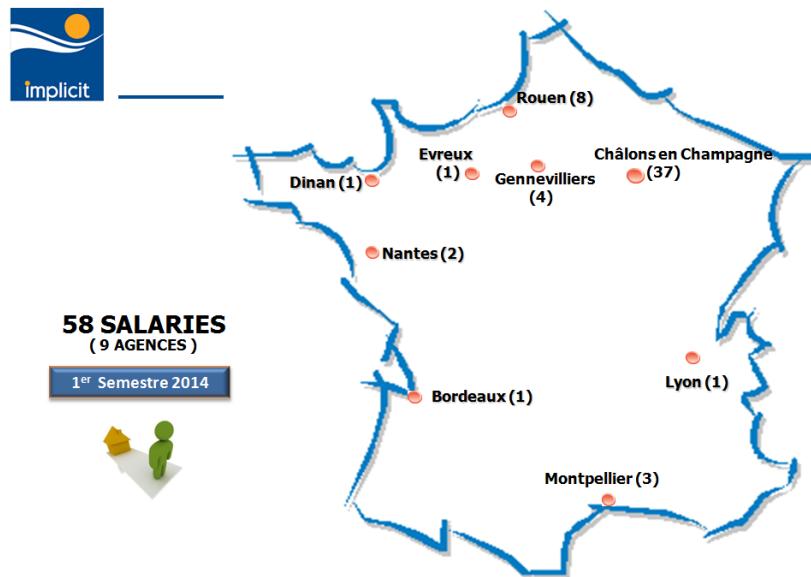


FIGURE 2. – Répartition géographique des salariés

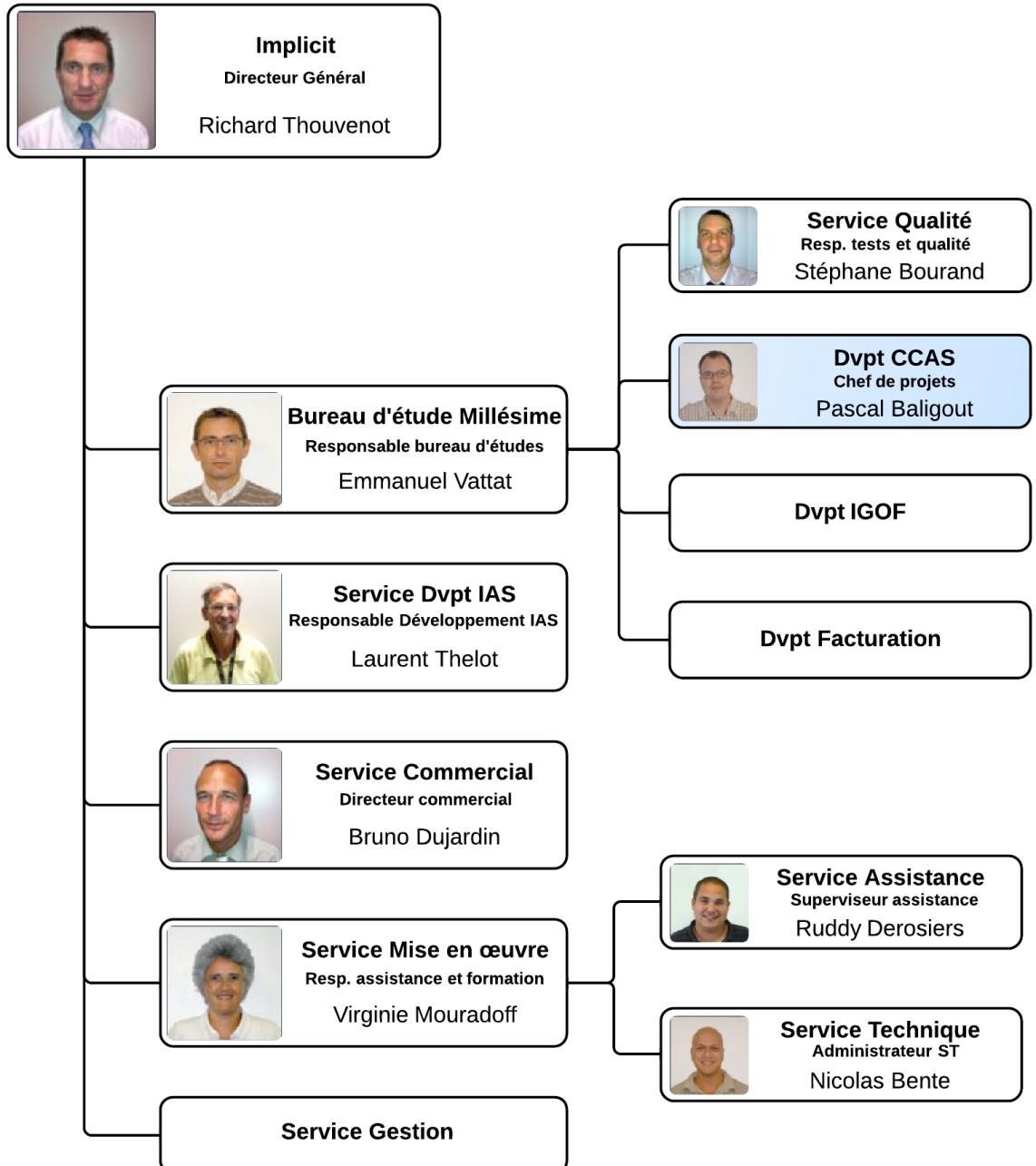


FIGURE 3. – Organigramme de la société Implicit (58 collaborateurs)

Mon poste a été rattaché au Bureau d'étude Millésime, et plus précisément au service de développement CCAS (*voir l'organigramme, Figure 3*). Le stage a donc eu lieu au siège social d'Implicit, à Châlons-en-Champagne (Marne).

J'ai travaillé sous la responsabilité directe du chef de projets, Pascal BALIGOUT. Et bien que j'aie été pleinement intégré dans l'équipe de développement CCAS/Facturation, j'ai eu la chance d'être très autonome.

1.4. Le modèle économique : une philosophie sociale et solidaire

La particularité d'Implicit réside dans la structure dont elle fait partie. Le Groupe Chèque Déjeuner est rattaché au secteur de l'Economie Sociale et Solidaire (ESS), qui cherche à produire, consommer et décider autrement. Cet « autrement », le site web jeun-ess.fr [2] a essayé de lui donner une définition :

L'économie sociale et solidaire (ESS) représente des milliers d'entreprises très diverses dans leurs réalités, qui partagent des caractéristiques essentielles : un projet économique au service de l'utilité sociale, une mise en œuvre éthique, une gouvernance démocratique et une dynamique de développement fondée sur un ancrage territorial et une mobilisation citoyenne.

Cette philosophie permet au Groupe Chèque Déjeuner de conserver un statut de multinationale leader sur plusieurs de ses domaines (*voir Figure 5*) tout en conciliant son activité économique avec une vision très sociale, tant de son utilité que de son fondement.



FIGURE 4. – Logo du Groupe Chèque Déjeuner

Par ailleurs, le groupe a été créé par *l'entreprise* Chèque Déjeuner, qui a pour forme juridique le statut de Société COopérative et Participative³ (SCOP). Il bénéficie donc d'une gouvernance démocratique (un salarié = une voix) et d'une répartition des résultats prioritairement affectée à la pérennité des emplois et du projet d'entreprise. Ce choix est représentatif des valeurs sociales et humaines que souhaitait véhiculer son fondateur.

3. Une SCOP « est un groupement de personnes poursuivant des buts économiques, sociaux et éducatifs communs, par le moyen d'une entreprise dont le fonctionnement est démocratique et la propriété collective »(Source : cheque-dejeuner.com [6])

N°3 MONDIAL

SUR LE MARCHÉ DES TITRES DE SERVICES PRÉPAYÉS

1^{RE} ENTREPRISE
 DE SON SECTEUR À ÊTRE ÉVALUÉE SUR SA
RESPONSABILITÉ SOCIÉTALE

4,1 MILLIARDS
 D'EUROS
 DE VOLUME
 D'AFFAIRES

1,1 MILLION
 DE COMMERÇANTS AFFILIÉS AU GROUPE

21 MILLIONS
 D'UTILISATEURS DE TITRES

13 PAYS

168 000

CLIENTS ENTREPRISES ET COLLECTIVITÉS


2000
 COLLABORATEURS

50 ANS

N°1 EN FRANCE

SUR LE MARCHÉ DE L'INFORMATISATION
DE L'ACTION SOCIALE

24,4 MILLIONS DE TITRES
 ÉMIS AU SERVICE DES POLITIQUES
 PUBLIQUES ET SOCIALES

1230 
 COLLECTIVITÉS
 ET INSTITUTIONS
 UTILISATRICES DE NOS TITRES DE PAIEMENT

4815
 CLIENTS
UTILISATEURS DE NOS SOLUTIONS
 DE GESTION INFORMATIQUE DÉDIÉES
 À L'ACTION SOCIALE
 ET MÉDICO-SOCIALE

LEADER
 SUR LE MARCHÉ
 DES COLLECTIVITÉS
TERRITORIALES

FIGURE 5. – Extrait de la brochure *Les solutions action sociale et médico-sociale du groupe Chèque Déjeuner, ça fait du bien au quotidien*

Cette vision du groupe, basée sur l'ESS, irradie sur chacune des entreprises qui en font partie, même si elles ne sont pas elles-même des SCOP⁴. De fait, on retrouve partout les valeurs qui positionnent l'homme au centre des débats et des organisations :

- Solidarité,
- Équité,
- Valeurs sociales,
- Développement durable,
- Humanisme.

Ainsi, Emmanuel VATTAT, directeur du bureau d'études au sein d'Implicit, comme Pascal BALIGOUT mon responsable, m'ont souvent sollicité afin de s'assurer que la mission était intéressante pour moi, et pour qu'elle soit au maximum en adéquation avec mon projet professionnel et ma formation actuelle. Par ailleurs, mon intégration a été pleine et entière, ainsi que la confiance qui m'a été donnée.

1.5. La position d'Implicit au sein du groupe Chèque Déjeuner

Suite au rachat d'Implicit par le Groupe Chèque Déjeuner, il est légitime de se demander comment un groupe spécialisé sur le marché des titres de services prépayés va intégrer une entreprise d'informatique à ses solutions commerciales.

Il s'agit en fait pour le Groupe d'unifier informatique sociale et vente de titres. En effet, il faut rapprocher Implicit de cinq autres sociétés, regroupées sous le nom de *Cohésia* (*voir Figure 6*).



FIGURE 6. – Cohésia, la solution d'« informatique sociale » du groupe

La solution Cohésia est composée de six entreprises distinctes ; leur rôle est différent, mais elles sont toutes complémentaires. On trouve aujourd'hui en son sein :

4. En tous cas pour le moment : l'Assemblée Générale des salariés de juin 2014 a entériné l'ouverture d'une réflexion sur l'élargissement du statut de SCOP aux différentes filiales du Groupe.

- Quatre sociétés informatique :
 - *Info.DB*, qui édite des logiciels de gestion de l'action sociale côté financeurs, destinés aux Conseils Généraux.
 - *Implicit*, qui produit des solutions à l'intention de l'instructeur communal.
 - *Apologic*, qui réalise les prestations financées avec notamment l'informatisation des structures d'aide et de soin à domicile.
 - *Arcan*, qui fournit les structures médico-sociales (HAD, EHPAD...).
- Deux sociétés émettrices de dispositifs prépayés :
 - *Chèque Domicile*, 1^{er} émetteur de CESU⁵, qui permet l'accès aux services à la personne.
 - *Chèques de Services*, émetteur de titres permettant d'acheter des biens et services essentiels à la vie quotidienne (alimentation, habillement, transport, hébergement...), souvent utilisés dans la délivrance d'aide sociale.

Dès lors, on constate l'importance d'une telle synergie : les entreprises de solutions informatiques ne se contentent pas de fournir aux organismes des logiciels de gestion, mais elles proposent également un lien naturel avec deux sociétés réalisant des prestations nécessaires à l'apport d'aides sociales.

5. Chèque Emploi-Service Universel, moyen de paiement utilisé dans les emplois de service à la personne.

2. Présentation du stage

La mission de ce stage repose sur l'intégration de *Business Objects* (BO). C'est pour cette raison qu'il est intéressant de s'attarder sur le choix de ce sujet ainsi que sur les deux outils sur lesquels a porté mon travail.

2.1. Ma mission au sein d'Implicit

Le sujet du stage qui m'a été confié est le suivant : « *Intégration de Business Objects dans nos solutions logicielles* ».

Il s'agit d'un sujet de Recherche et Développement, dont la durée peut difficilement être estimée. Il a ainsi été restreint à trois activités, et dans l'ordre de priorité suivant :

- Exploitation des états BO (Webi XI 4) directement dans la solution logicielle.
- Intégration des solutions de cartographie de *Business Geografic* (qui s'appuient sur BO)
- Implémentation d'un connecteur pour BO XI R3

Face à l'ampleur du projet, il a été rapidement convenu que la dernière activité serait écartée du sujet du stage. D'autant plus que, s'agissant toujours de R&D, nous n'étions pas assurés de pouvoir parvenir à réaliser le premier item. Sa concrétisation a relégué à l'arrière plan l'implémentation d'un connecteur pour une version plus ancienne de BO.

2.2. Présentation du logiciel CCAS Web

L'outil *CCAS Web* est la solution online de l'ancienne application client-serveur *Millésime*. En effet, une première version en client lourd a été déployée pendant des années, jusqu'à ce que les usagers lui préfèrent un outil en client léger. Elle permet de se passer de la configuration sur chaque poste, nécessaire lors d'une mise à jour ou d'une installation : la seconde solution ne nécessite qu'un explorateur web.

CCAS Web est un progiciel de gestion des aides sociales (*voir Annexe A page III*). Il doit permettre d'éditer des formulaires, d'attribuer des aides, d'effectuer un suivi...

Il s'agit d'un portail pointant vers chacun des modules⁶ utilisés par les usagers d'une commune (*voir Figure 7*), et dans lequel les bénéficiaires d'aides sociales s'articulent autour d'un *foyer* — notion représentant une famille au sens large, souvent identique au foyer fiscal — et défini par son responsable.

6. Chaque module représente un dispositif d'aide, parmi : Aides légales, aides facultatives, aides extérieures, RSA, demande de logement social, SDS (Sans Domicile Stable), suivi social, élection de domicile.



FIGURE 7. – Page d'accueil de l'application CCAS Web

Une fois les bénéficiaires enregistrés en base, CCAS Web permet d'effectuer principalement deux types d'actions :

- L'instruction simple d'un dossier. Dans le cas de l'aide légale⁷ par exemple, ce n'est pas la commune qui est décisionnaire, mais généralement le Conseil Général. Le logiciel permettra alors d'enregistrer la demande et de la formater pour être transmise à l'autorité décisionnelle compétente.
- La gestion complète du dossier. Dans le cadre d'une aide facultative⁸ par exemple, la demande sera enregistrée, étudiée en commission et mise en place *via* l'interface CCAS Web.

Ce type de logiciel étant indispensable aux employés des CCAS, il est vital de le maintenir à jour et d'être à l'écoute des évolutions légales, mais aussi des besoins des utilisateurs. À cet égard, leur revendication principale était d'avoir accès à un outil de statistiques avancé et simple d'utilisation pour pouvoir générer différentes éditions, apporter une aide décisionnelle et surtout se plier à l'obligation de fournir l'*ABS*⁹ (*voir 2.3.1 Un cadre légal*).

C'est à ce niveau que je suis intervenu.

7. Aide dont le montant, les conditions d'attribution, le nombre d'affectations annuelles autorisées... sont cadrés par la loi. [4]

8. Aide définie par la politique sociale de la commune, comme un prêt sans intérêt, l'attribution de CAP (Chèque d'Accompagnement Personnalisé)... [4]

9. Analyse des Besoins Sociaux.

2.3. Présentation du projet

Le sujet du stage portant sur l'intégration de Business Objects, nous allons voir comment le bureau d'études a raisonné pour sélectionner cette solution.

2.3.1. La nécessité d'intégrer une solution statistique

Un cadre légal

Pour pouvoir effectuer un suivi de l'activité de ses services, un CCAS est légalement tenu de fournir un certain nombre de statistiques au conseil d'administration¹⁰ : l'ABS. Ces dernières permettent également à la structure d'éditer des rapports d'activité divers, permettant par exemple de visualiser à tout moment le montant total des aides attribuées par rapport à son budget annuel.

Sur la base de cette analyse, les CCAS doivent également mettre en œuvre une action sociale générale de prévention et de développement social dans la commune.

Pour ces raisons, Implicit a inclus depuis plusieurs années dans son logiciel un outil de statistiques.

La nécessité d'un nouvel outil

L'outil actuel est un simple requêteur intégré au site : il a été développé afin de permettre aux travailleurs sociaux n'ayant aucune connaissance en base de données d'effectuer des requêtes vers Oracle DB, via une interface graphique utilisateur (*voir Figures 8 et 9*). Le résultat se présente sous forme d'un tableau dont les en-têtes ont été sélectionnés par le gestionnaire du CCAS, à la manière d'un retour de `SELECT` de base de données.

Cependant, ce requêteur souffre d'un certain nombre de problèmes : bien qu'intuitif pour l'utilisateur, il présente de nombreux bugs. De plus, les délais de réponse sont inadaptés et ses fonctionnalités sont assez limitées par rapport aux besoins.

Forts de ces éléments, la décision a finalement été prise d'abandonner le requêteur et d'utiliser une solution de *Business Intelligence*¹¹ éprouvée, à intégrer directement dans CCAS Web. Parmi différents éditeurs — Oracle (OBI), Microsoft (SQL Server Analysis Services), etc. — Implicit a choisi d'implémenter *Business Objects*, une solution développée par SAP.

2.3.2. Présentation de l'outil *Business Objects* de SAP

Business Objects est une solution de Business Intelligence. À l'origine française, elle a été rachetée en 2008 par l'Allemand SAP — il s'agit du plus gros concepteur de logiciels en Europe, et du quatrième mondial [5].

10. Articles 1^{er} et 2^e du *Décret n° 95-562 du 6 mai 1995 relatif aux centres communaux et intercommunaux d'action sociale [...] (voir Annexe D page XV)*

11. L'informatique décisionnelle, ou Business Intelligence (BI), est un ensemble d'outils destinés à offrir une vue d'ensemble de l'activité traitée à un dirigeant d'entreprise, et à donner une aide à la décision.

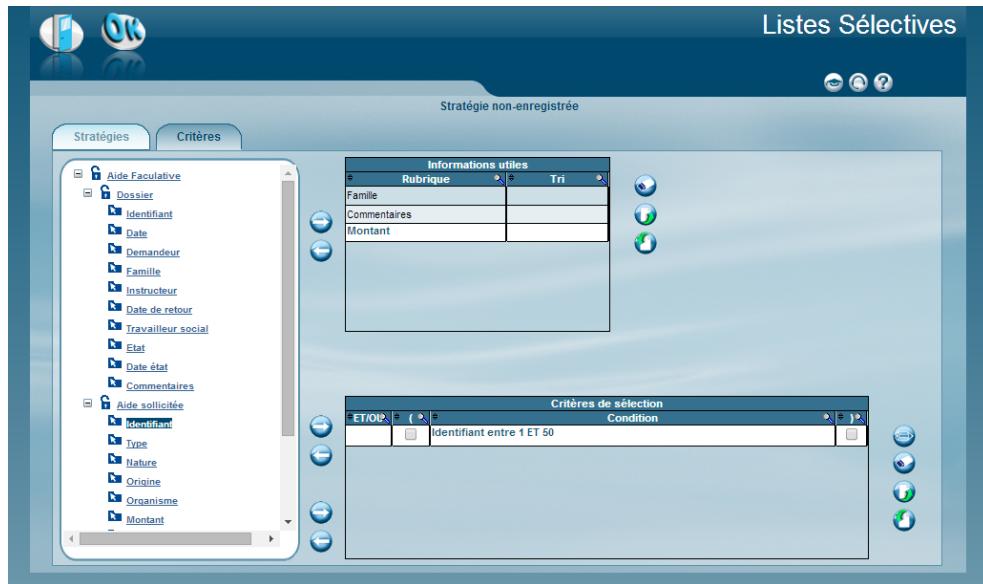


FIGURE 8. – Requêteur – Choix des filtres

Famille	Commentaires	Montant
TROUSIO LUCIA		120.25
ROLDAN LOIC		100
DUJARDIN BRUNO		150
DUJARDIN BRUNO		400
GURAUD SYLVAIN		500
GURAUD SYLVAIN		200
GURAUD SYLVAIN		0
GURAUD SYLVAIN		50
RAJABUL ARTHUR		50
RAJABUL ARTHUR		200
RAJABUL ARTHUR	Second essai de création de demande d'AF	500
RAJABUL ARTHUR		500
RAJABUL ARTHUR		120
PAREL FABIEN		230
Suite à la perte d'emploi de		

FIGURE 9. – Requêteur – Résultat de l'exécution

Il existe en version client lourd, mais dans le cas de CCAS Web, c'est l'interface navigateur qui va être utilisée. Il sera ainsi plus aisé de l'intégrer directement dans l'application d'Implicit.

Il propose, principalement, un outil de construction de requêtes et de mise en page des résultats, sous forme de rapports (*voir* Figure 10) ou de tableaux de bord. Les requêtes s'appuient sur une base de données externe, couplée à BO via un *Univers* : il s'agit du liant entre les données brutes de la base et l'applicatif métier, plus parlant aux décisionnaires.

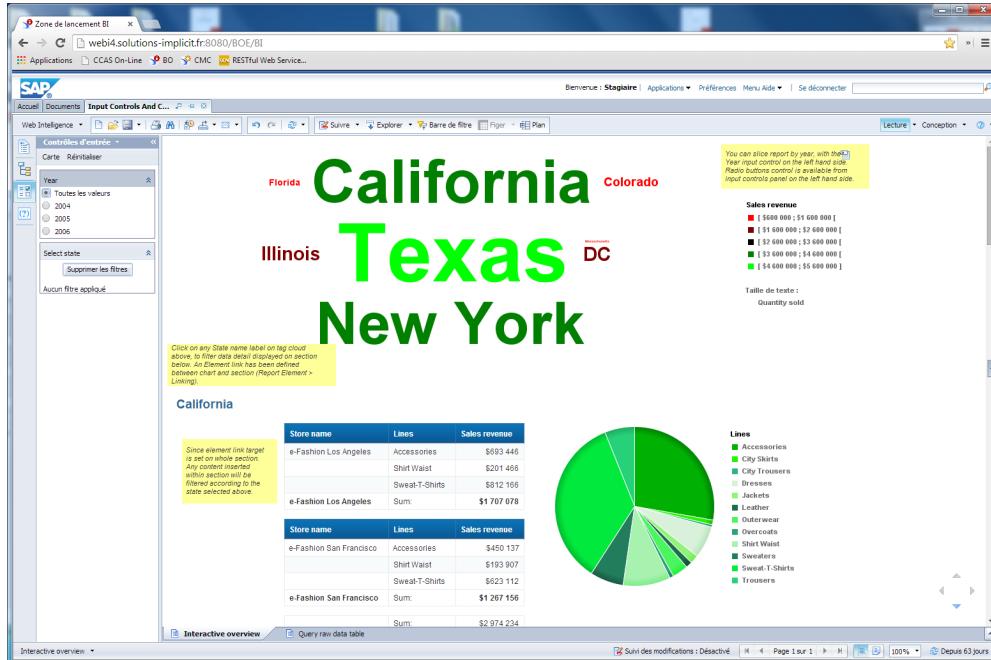


FIGURE 10. – Exemple d'un report BO

Business Objects est un produit très abouti, relativement complexe — c'est d'ailleurs pour simplifier son utilisation qu'il m'a été demandé de l'intégrer à CCAS Web — mais la manière de construire les requêtes est assez intuitive pour l'utilisateur final (*voir* Figure 11) : c'est là que réside sa force.

Si le choix de l'équipe dirigeante d'Implicit s'est porté sur Business Objects plutôt qu'un autre outil d'informatique décisionnelle, au delà des conditions des contrats qui ont pu être négociées et des accords financiers, c'est parce que certains CCAS sont déjà équipés de cet outil, et que la transition entre l'ancien requêteur et Business Objects pourrait se faire plus en douceur. En effet, Implicit utilisait déjà BO depuis plusieurs années au vu de sa position de leader, et le proposait à ses clients, mais sans intégration

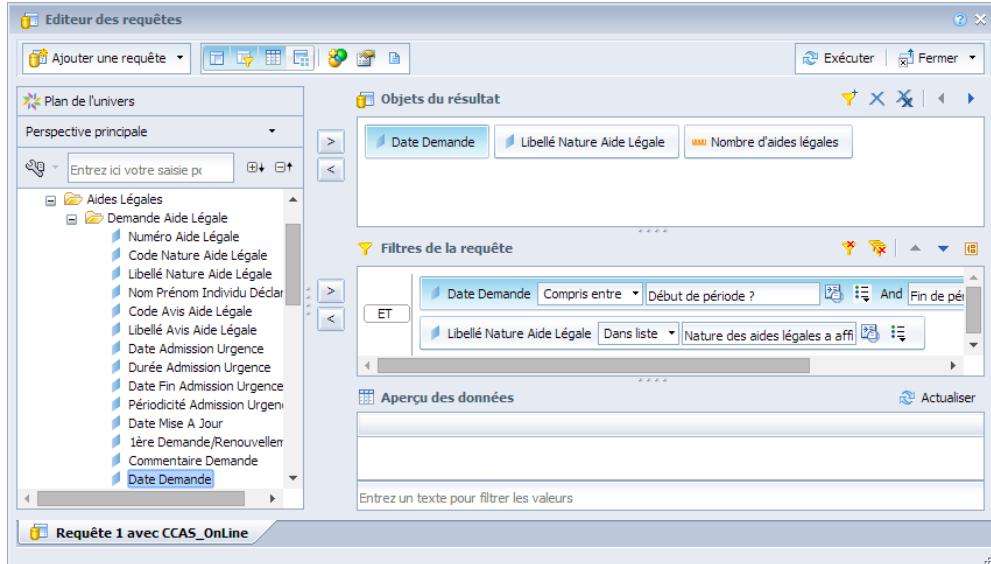


FIGURE 11. – L’interface de création de requête BO : la sélection de filtres et d’intitulés se fait par cliqué-glisssé

poussée à CCAS Web.

De plus, Business Objects supporte de très nombreux outils externes, tel que le connecteur *GeoBI*¹², qui constitue l’étape suivante de l’intégration BO que j’ai eu à effectuer dans CCAS Web.

Business Objects gère aussi une notion de droits et de groupes : basiquement, il y a trois niveaux d’utilisateurs (en plus de l’administrateur) :

- Le créateur de documents : cet utilisateur dispose des droits de création/édition d’un document. Il peut notamment accéder à l’interface de création de requête (*voir* Figure 11).
- Le lecteur simple : cet utilisateur ne peut que consulter un document déjà généré.
- Un lecteur avancé : il ne peut pas modifier le document, mais peut affecter des valeurs aux différents filtres pré-sélectionnés.

L’intégration de Business Objects dans CCAS Web s’adresse particulièrement aux deux derniers types d’utilisateurs : le créateur de documents aura suivi une formation afin de créer des états¹³ et sera donc à l’aise avec toutes les subtilités de BO en natif, mais les autres seront beaucoup plus novices. Il sera donc nécessaire que les différents accès via CCAS Web soient le plus intuitif possible pour l’utilisateur final.

12. GeoBI, proposé par Business Geografic, est une extension cartographique pour Business Objects de SAP.

13. Un état est une requête BO sauvegardée. Il est possible d’y appliquer à postériori des filtres sans altérer le document de manière définitive.

2.4. Planification de la mission et méthodologie de travail

Lors de la définition du sujet du stage, mon responsable Pascal BALIGOUT n'avait effectué que des recherches superficielles quant aux possibilités d'intégration. Ainsi, il ne disposait que d'une vision de surface de l'état du réalisable lorsqu'il m'a confié le projet.

S'agissant de recherche appliquée, un projet de Recherche & Développement comme celui-ci peut difficilement être planifié avec précision, et établir un rétro-planning est encore plus complexe. Néanmoins, différentes étapes ont pu être identifiées, pour chacun des objectifs du stage (*voir* 3.2 et 4.2) :

- L'exploitation des états BO directement dans la solution logicielle est l'objectif qui devra impérativement être atteint avant l'échéance du stage, au 31 juillet.
- L'intégration des solutions de cartographie de l'entreprise *Business Geografic* est considérée comme une activité annexe. Implicit a prévu de mettre en place cet outil à terme, mais cet objectif n'étant pas prioritaire sur l'implémentation de BO dans CCAS Web, il pourra être différé sans problème.

Par ailleurs, il faut noter que j'ai travaillé de façon indépendante sur ce projet : en effet, bien qu'intégré au sein de l'équipe technique, j'ai été relativement autonome sur ces missions. Pascal BALIGOUT (chef de projets) et Emmanuel VATTAT (responsable du bureau d'études) m'ont donc fait confiance sur de nombreux éléments du développement.

Nous avons néanmoins eu des réunions et des comités réguliers (un à deux par mois) pour faire le point, définir les besoins, valider le cahier des charges (*voir* Annexe C page IX), etc. J'ai également profité de suivis hebdomadaires sur l'état d'avancement.

3. L'intégration Business Objects

La première étape de mon stage, et la plus longue, a été l'intégration de Business Objects dans CCAS Web.

3.1. Organisation temporelle

Initialement, la durée d'intégration prévue était celle de l'intégralité de la période couverte par ma présence en entreprise, à savoir 6 mois. Néanmoins, j'ai eu besoin de moins de temps pour remplir cette mission.

La phase d'intégration de BO n'a finalement duré que quatre mois et demi : de début février à mi-juin, j'ai pu travailler en autonomie sur ce projet. De la recherche des fonctionnalités mises à disposition par l'API au développement en lui-même au sein de l'application, j'ai été amené à effectuer différentes tâches (*voir* Figure 12).

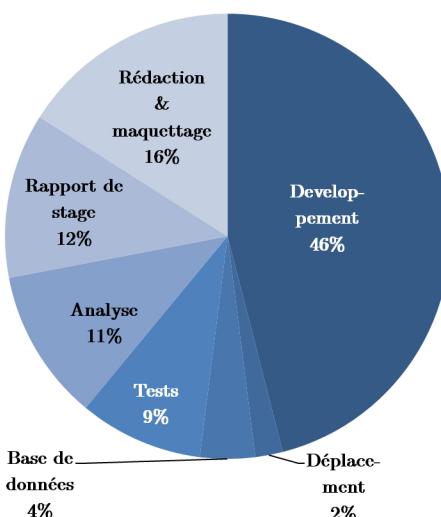


FIGURE 12. – Charge de travail pour l'intégration de BO

Parmi mes tâches principales, on peut voir que l'analyse, la rédaction de documents et le développement représentent à eux seuls près de 75% de la charge de travail. Cette dernière a été principalement axée sur le développement de l'application demandée, mais on voit qu'une partie non négligeable concerne les tâches annexes, à savoir

la recherche d'information (analyse) et la rédaction de documents divers (cahier des charges, manuel utilisateur...).

On peut constater que ma mission au sein d'Implicit a été très complète et les tâches diversifiées.

3.2. Étapes de l'intégration, choix de programmation et difficultés rencontrées

Un des intérêts du stage que j'ai effectué chez Implicit est que j'ai pu étudier et proposer moi-même une partie des technologies utilisées. Elles ont fait l'objet d'une démarche réfléchie.

3.2.1. Etude du réalisable

Avant toute chose, il m'a fallu opérer une étude du réalisable. En effet, Pascal BALIGOUT s'était renseigné sur l'intégration de BO via la sollicitation de web-services¹⁴, afin de s'assurer que le projet était à ma portée. Mais il n'était pas allé plus loin et il m'appartenait donc d'étudier ce que cette technologie mettait à disposition et ce qui restait hors d'atteinte.

Dans un premier temps, je ne me suis pas du tout attaché à l'applicatif métier qu'est CCAS Web et me suis concentré sur les possibilités offertes par l'API BO.

Cette API repose sur l'architecture REST.

L'architecture REST

Elle est très bien adaptée au web dans la mesure où ses contraintes d'exécution reposent, entre autres, sur l'environnement client-serveur et une mise en cache.

Il s'agit simplement d'envoyer une requête formatée au serveur, et d'attendre un éventuel retour. L'avantage de cette architecture est qu'elle ne nécessite pas de connaissance approfondie du fonctionnement interne du serveur, ni d'authentification préalable (si nécessaire, elle peut s'effectuer également par webservice). Ces requêtes, en HTTP, s'effectuent de la manière suivante : il faut spécifier l'adresse à solliciter et le type de requête (`GET`, `POST`, `PUT`, ou encore `DELETE`).

On peut y adjoindre un ou des paramètres d'en-tête sous la forme de couple clé-valeur, ainsi qu'un corps de texte en XML ou JSON. L'en-tête comme le corps de texte sont facultatifs.

Premiers appels au serveur

Afin d'effectuer mes tests sans avoir à développer un mini-projet, j'ai utilisé une application de Google Chrome : *Advanced REST Client*. Cet outil à intégrer au navi-

¹⁴. Un *webservice* (ou service web en français) est une méthode de communication entre deux appareils électroniques via un réseau. Cette méthode est normalisée via le protocole SOAP, qui décrit un mode de communication vers des objets distants.

gateur (*voir Figures 13 et 14*) permet de solliciter de façon très simple le serveur BO et de visualiser les retours.

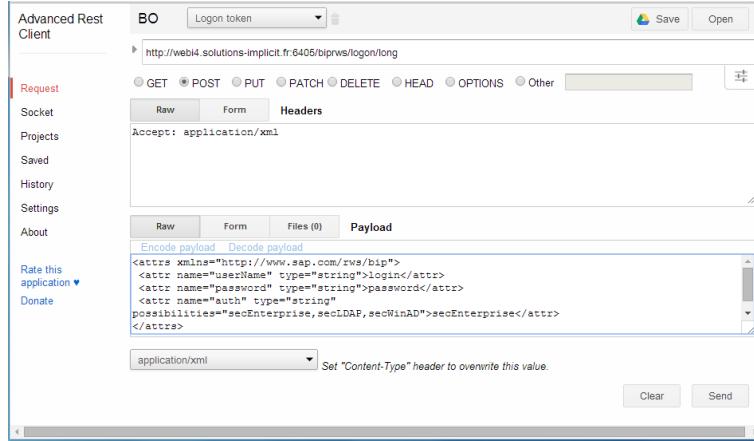


FIGURE 13. – Envoi d'une requête avec *Advanced REST Client*

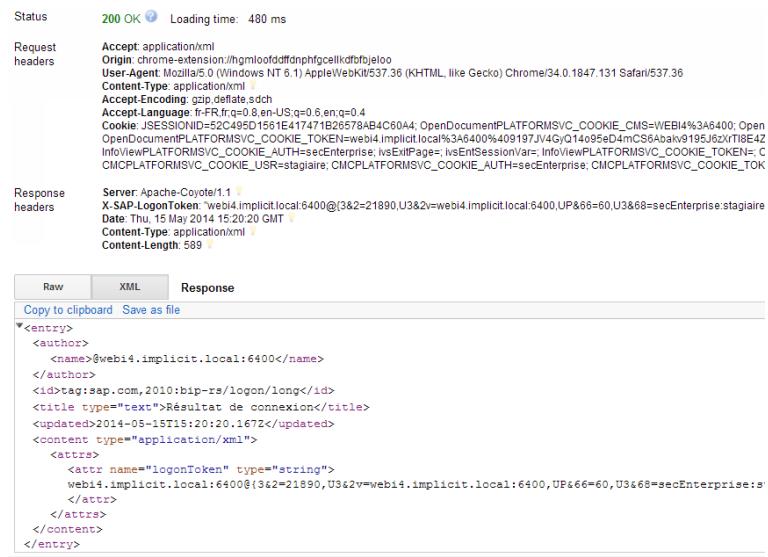


FIGURE 14. – Retour d'une requête sous *Advanced REST Client*

Ainsi, j'ai pu tester l'API BO vers le serveur REST. En effet, les ressources SAP sont très prolixes, et principalement en anglais, j'ai donc passé ma première semaine

au sein de l'entreprise à collecter les fichiers mis à disposition et à les étudier.



Advanced REST client

FIGURE 15. – *Advanced Rest Client* dans le lanceur d'applications Chrome

Ces tests m'ont conduit à rédiger un document condensant toutes les possibilités offertes par BO ainsi que leurs moyens de réalisation.

Rédaction de l'étude

Cette étude (*voir Annexe B page III*) a été la base de travail qui a permis de voir l'étendue des possibilités offertes par l'API REST de BO.

Elle a été le point de départ du projet : sans elle, le comité de pilotage n'aurait pas pu savoir quelle serait la limite dans l'intégration de BO à CCAS Web. J'y ai donc recensé ce que permettait l'API, mais également ce qu'il était impossible de réaliser via requêtes REST. Par exemple, nous nous sommes résignés à ne pas gérer la création d'utilisateur BO et l'attribution des droits dans l'application CCAS Web car il aurait fallu utiliser impérativement l'API java, trop lourde à gérer, en particulier pour ce seul besoin.

3.2.2. Prise en main de WebDev

Les logiciels d'Implicit ne reposent pas sur du PHP, mais sur l'AGL¹⁵ du français PCSOft : *WebDev* (*voir Figures 16 et 17*). Il présente l'avantage d'utiliser le WLangage — langage commun aux environnements web, client/serveur et mobile de cet éditeur — ce qui permet de porter très facilement un site web vers un logiciel Windows.

```

1 PROCEDURE vGénèreFils(oBo est un objet dynamique gBO, ...
2     ARB_arbre = Null, bGénèrePetitFils est un booléen = Faux, ...
3     cFilsAignorer est une chaîne = "")
```

FIGURE 16. – Exemple de définition d'une méthode en WLangage

15. Le terme AGL, ou Atelier de Génie Logiciel, désigne « un ensemble de programmes informatiques permettant eux-mêmes de produire des programmes de manière industrielle » (Source : Wikipedia)

La prise en main du WLangage en lui-même est assez aisée : proche du langage naturel, le développement se fait en français. Cependant, de nombreux éléments évidents à programmer avec d'autres langages ne le sont pas du tout sous WebDev ; c'est le cas pour une requête REST native par exemple (*voir 3.2.5 Le choix de la technologie du requêtage REST*).

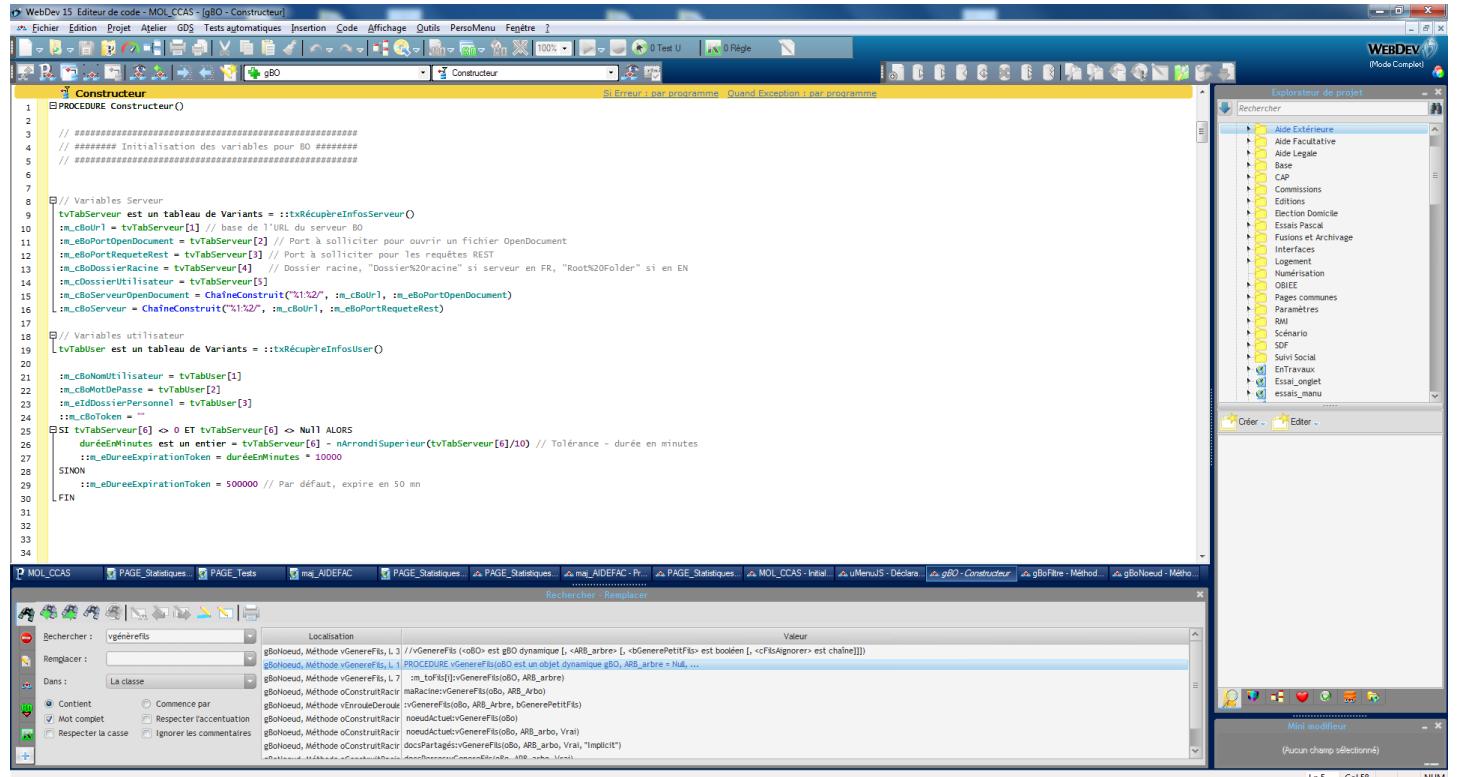


FIGURE 17. – Environnement de travail WebDev

J'ai donc lu le livret d'auto-formation à WebDev 15 [3] afin de me familiariser avec le WLangage et ses subtilités, avant d'installer une version du projet CCAS Web sur ma machine.

Dans le même temps, il a fallu s'intéresser à ce qu'il faudrait intégrer de façon concrète. Cela passe par une analyse des besoins des utilisateurs.

3.2.3. Analyse des besoins utilisateurs

Étant dans l'impossibilité de monter un comité utilisateur ou même de rencontrer les usagers d'un CCAS, j'ai pu participer à deux réunions afin de déterminer l'orientation

du projet :

La première le 11 mars 2014 avec Stéphane BOURAND (responsable du service qualité) et Pascal BALIGOUT (chef de projets). Le but était avant tout de m'exposer la vision qu'ils avaient de l'implémentation de BO.

La seconde a eu lieu le 14 mars 2014 en présence de Pascal BALIGOUT, Emmanuel VATTAT (responsable du bureau d'études), Farid EL KADIRI (commercial sur la gamme CCAS Web), et Ruddy DEROISIERS (responsable du service assistance). La rencontre de ces acteurs proches des utilisateurs a permis de comprendre leurs envies et attentes.

A l'issue de ces deux réunions, il est ressorti plusieurs désirs et remarques. Elles serviront de base à mon développement.

- Au lancement du logiciel, il n'y a aucun point de menu "Statistiques" dans l'application CCAS Web. Il faut commencer par en ajouter un puisqu'il s'agit de la principale lacune de l'application.
- Il pourrait être intéressant d'accéder directement aux documents stockés dans BO puisqu'il peut être utilisé comme une GED¹⁶.
- Il faut pouvoir visualiser directement un report BO, soit via l'interface web native, soit — et cette solution est à privilégier si possible — en le téléchargeant. Cela signifie qu'il sera également nécessaire de saisir les valeurs des filtres d'un report directement dans CCAS Web (ces valeurs sont assimilables au `_x_` dans le `WHERE filter = _x_` d'une requête SQL).
- Si possible, ces invites seront pré-remplies en fonction du module dans lequel se trouve l'utilisateur (par exemple saisir automatiquement le numéro d'un dossier lorsqu'on le consulte).

Suite à ces observations, je les ai formalisées dans un cahier des charges afin de partir sur des bases de programmation saines.

3.2.4. Rédaction du cahier des charges et maquettage

Document interne destiné à poser les règles de construction depuis les fondations jusqu'aux détails les plus fins, le cahier des charges doit faire preuve d'une précision absolue.

Au sein d'Implicit, il est composé aussi bien des informations techniques et fonctionnelles que graphiques. Ainsi, de nombreuses maquettes ont pu y être insérées (*voir* Figure 18), et il m'a fallu me projeter dans la position de l'utilisateur final afin de rendre chacune des nouvelles fonctionnalités à implémenter la plus intuitive possible. Cet exercice est d'autant plus difficile qu'à ce stade aucun développement n'a encore été commencé.

J'ai donc rédigé ce cahier des charges qui a bénéficié de plusieurs mises à jour : il a fait la navette entre le chef de projets, le responsable du bureau d'études et moi. Cela m'a permis d'obtenir une version finale validée (*voir* Annexe C page IX).

16. « La Gestion électronique des documents désigne un procédé informatisé visant à organiser et gérer des informations et des documents électroniques. » (Source : Wikipedia)

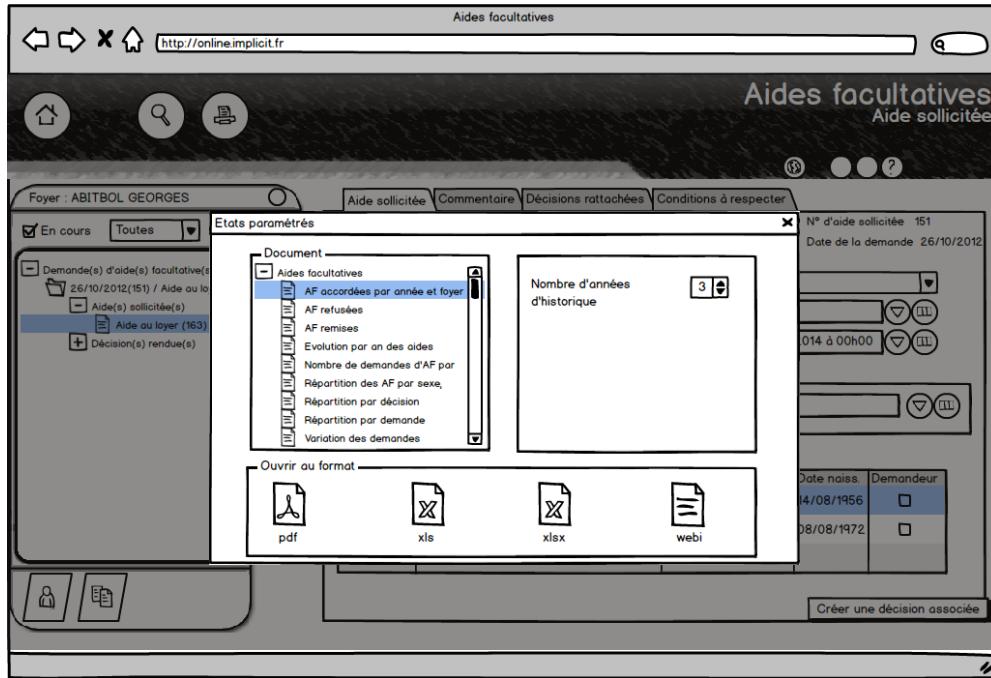


FIGURE 18. – Maquette d'une nouvelle fenêtre dans l'application

3.2.5. Phase d'intégration

L'intégration de BO à CCAS Web a été la partie principale de mon stage, et la plus longue. C'est lors de celle-ci que j'ai été confronté à la majorité des difficultés que j'ai pu avoir lors de ces six mois.

Cette phase peut être divisée en plusieurs étapes principales.

Le choix de la technologie du requêtage REST

La décision d'utiliser l'API REST de BO n'a pas été de mon ressort : elle a été motivée par le fait que les échanges sont rapides, que l'installation sur le serveur n'est pas particulièrement lourde et qu'elle permet une grande diversité de fonctionnalités. Je ne la détaillerai donc pas plus ici.

En revanche, on m'a laissé la liberté du choix de la technologie à utiliser pour effectuer ces requêtes, ainsi, de nombreuses possibilités m'étaient offertes. Ayant rencontré des problèmes sur la plupart d'entre elles, cela m'a conduit à en éliminer un grand nombre.

• **Utiliser les fonctionnalités natives WebDev** a été ma première initiative. En effet, l'API de PCSoft propose deux fonctions : `HTTPRequête()` et `HTTPEnvieFormulaire()`, mais qui malheureusement ne sont pas assez complètes.

La première fonction interdit l'utilisation de requêtes de type `UPDATE` ou `DELETE` ainsi que des en-têtes multiples, et ne me permet donc pas de solliciter tous les types de webservices. Et si la seconde autorise les en-têtes multiples et les quatre types existants, il est en revanche impossible d'y adjoindre un corps de requête, ce qui est bien plus bloquant.

En conséquence, j'ai tenté d'envoyer un e-mail à l'assistance de la société éditrice, PCSoft, mais la seule réponse que nous ayons reçue était bancale, ce qui m'a encouragé à passer à une autre technologie.

• **Développer une appli externe en PHP** : il s'agit de la solution la plus souvent proposée sur les forums d'entraide à WebDev, officiel ou communautaires. Envoyer une requête HTTP se fait alors très simplement, mais mon responsable a rejeté cette possibilité. En effet, il faudrait installer un serveur PHP à côté de celui exécutant WebDev, ce qui serait excessif pour un seul besoin.

Je me suis donc tourné vers un langage nativement accepté par WebDev.

• **JavaScript, jQuery et Ajax** En effet, l'Ajax est une architecture qui permet d'effectuer une requête en sollicitant une autre page via le `XMLHttpRequest`¹⁷. L'utilisation de cet objet est grandement facilitée par la bibliothèque libre jQuery, qui est déjà utilisée par CCAS Web.

J'ai ainsi réalisé une classe JavaScript permettant d'effectuer des requêtes spécifiques à BO (voir Figure 19).

```

32 // ##### Métodes #####
33 // ##### Méthode générique de requête RESTful
34 // Paramètres :
35 // * reqUrl (string) : fin de l'url, qui va influencer l'action à effectuer
36 // * reqType (string) : prend en valeur "POST", "GET", "PUT" ou "DELETE" en fonction de la requête à effectuer
37 // * reqToken (string) : prend le token de la session à exploiter. Peut être égal à chaîne vide ""
38 // * payload (string) : corps de la requête si nécessaire. En xml.
39 // * extension (string) : type du retour à exploiter. Généralement "xml", mais peut prendre d'autres valeurs ("pdf", "vnd.ms-excel"...)
40 // * callback (function) : fonction de rappel pour effectuer la suite de la requête AJAX
41 MyBO.prototype.restRequest = function(reqUrl, reqType, reqToken, payload, extension, callback) {
42   $('html').css('cursor', 'progress');
43   $.ajax({
44     url: this.server + "/bipws/" + reqUrl,
45     type: reqType,
46     contentType: "application/xml",
47     dataType: "text",
48     headers: {"X-SAP-LogonToken": reqToken, "Accept": "application/" + extension },
49     data: payload,
50     // ce déroule
51   });
52 }
53
54

```

FIGURE 19. – Extrait de la classe JavaScript de gestion de BO

Mais j'ai à nouveau été confronté à deux problème : l'un majeur, et l'autre, moins important, soulevé par mon responsable de stage :

17. « XMLHttpRequest est un objet JavaScript qui a été créé par Microsoft [...] pour récupérer facilement des données via HTTP. » (Source : developer.mozilla.org)

- Il est impossible en JavaScript de générer à la volée un fichier. En effet, avec la librairie de fonctions BO, je peux récupérer le contenu *binnaire* d'un pdf et rediriger la sortie standard vers un nouveau fichier. Malheureusement, sans pouvoir créer ce fichier dynamiquement, impossible de proposer à l'utilisateur de le télécharger.
- Si les requêtes se font en AJAX, elle seront effectuées côté client et non côté serveur. Cela impliquerait d'autoriser auprès du serveur BO les requêtes provenant de nombreux postes au lieu d'un seul, ce qui risque de poser problème aux plus grosses structures parmi les CCAS.

Pour ces raisons, l'utilisation du JavaScript a été abandonnée, en tous cas pour la partie requête HTTP (*voir 3.2.5 La gestion des filtres*).

• **Un exécutable de type batch** est de loin ce qui m'a fourni les résultats les plus probants. WebDev étant hébergé sur un système Windows Server, il est tout à fait possible de lancer l'exécution d'un `.bat`.

J'ai donc choisi l'utilitaire *cURL*¹⁸, une interface s'appuyant sur la bibliothèque `libcurl` permettant d'effectuer une requête HTTP et d'en récupérer son retour. Il peut donc être utilisé en tant que client REST (*voir Figure 20*).

```

1  @echo off  rem désactive l'affichage des commandes
2  cls          rem efface les lignes précédentes
3
4  rem on envoie les paramètres dans des variables
5  rem l'ajout du tilde supprime les double quote entourantes ("")
6  set requestType=%1
7  set params=%~2
8  set url=%3
9  set redirect=%4
10
11 rem remplacement de "" par " et de #egal# par =
12 set params= %params:""="#"
13 set url= %url:#egal##%
14
15 curl -X "%requestType%" "%params% %url% > %redirect%

```

FIGURE 20. – Code batch utilisé dans le projet

Grâce à ces quelques lignes, j'étais alors en mesure de solliciter le serveur de webservices pour effectuer toutes les requêtes REST nécessaires. En effet, je peux ici choisir le type de requête, les différents paramètres — et donc les différents en-têtes —, l'adresse à attaquer, et enfin, une redirection du contenu retour dans un fichier, si besoin.

18. « Abréviation de Client URL Request Library : "Bibliothèque de requêtes aux URL des clients" ou See URL : "Voir URL" »(Source : Wikipedia)

L'avantage de cURL est qu'il s'agit d'une bibliothèque open-source et gratuite, y compris dans le cadre d'un usage commercial, son utilisation ne posait donc aucun problème pour l'intégration.

Cette solution m'a néanmoins posé quelques soucis dans la mesure où je tombais parfois sur l'erreur cURL suivante : `curl: (27) Out of memory`. Voici ce qu'il ressort de la documentation officielle :

`CURLE_OUT_OF_MEMORY (27)`

A memory allocation request failed. This is serious badness and things are severely screwed up if this ever occurs.

Je n'ai pu comprendre d'où venait cette erreur qu'après quelques temps : l'ordinateur sur lequel je travaillais ne disposait que de 2 Go de RAM et cette dernière était presque saturée lors du lancement du serveur local WebDev. C'est pourquoi l'erreur semblait survenir de manière aléatoire.

J'ai solutionné plus tard cette difficulté en libérant des ressources. Ce problème ne se posera plus une fois l'implémentation déployée, puisque le serveur sera autrement plus puissant que ma machine.

- **Importer la librairie `libcurl.dll`** Il s'agit de la librairie sur laquelle cURL s'appuie pour pouvoir effectuer ses requêtes, c'est pourquoi j'ai décidé de descendre encore d'un niveau. Cette bibliothèque dispose d'une API très complète, mais là encore j'ai fait face à un blocage qui m'a empêché de l'utiliser.

En effet, pour pouvoir effectuer une requête avec `libcurl.dll`, il faut entre autres passer un pointeur vers une fonction callback¹⁹ en paramètre. Malheureusement, à la compilation, le pointeur n'est pas reconnu et WebDev tombe.

Ainsi, je suis revenu à la solution précédente (le batch) en essayant de trouver d'où venait le problème. Une fois résolu, j'ai pu avancer dans l'intégration de BO.

La gestion des filtres

Lors de l'ouverture d'un report, il peut être nécessaire de restreindre sa recherche (sur un bénéficiaire particulier, une plage de dates...) plutôt que d'avoir l'intégralité des résultats : c'est ici qu'interviennent les filtres.

À la construction de la requête dans BO (*voir* Figure 11), il est possible d'appliquer des filtres : un équivalent d'une « clause `WHERE` ». Ainsi, lors d'une consultation ultérieure dans l'interface native, il sera alors nécessaire d'entrer dans des champs HTML les valeurs de restriction souhaitées. Cela permettra de générer un report adapté aux besoins.

Avec les webservices REST, il m'est possible de solliciter les serveurs BO et de récupérer les différentes limitations appliqués, ainsi que leurs valeurs par défaut. Cependant, j'ai été confronté à une difficulté lors du développement de l'interface pour

19. Un *callback*, ou fonction de rappel, est une fonction passée en paramètre à une deuxième. La seconde ne connaît pas par avance la première, mais peut y faire appel directement.

afficher ces filtres. En effet, pour pouvoir les présenter au travailleur social, il me faudrait générer dynamiquement des champs HTML — texte, checkboxes, liste déroulante, etc — en fonction du type et du nombre de filtres (*voir* Figure 21).

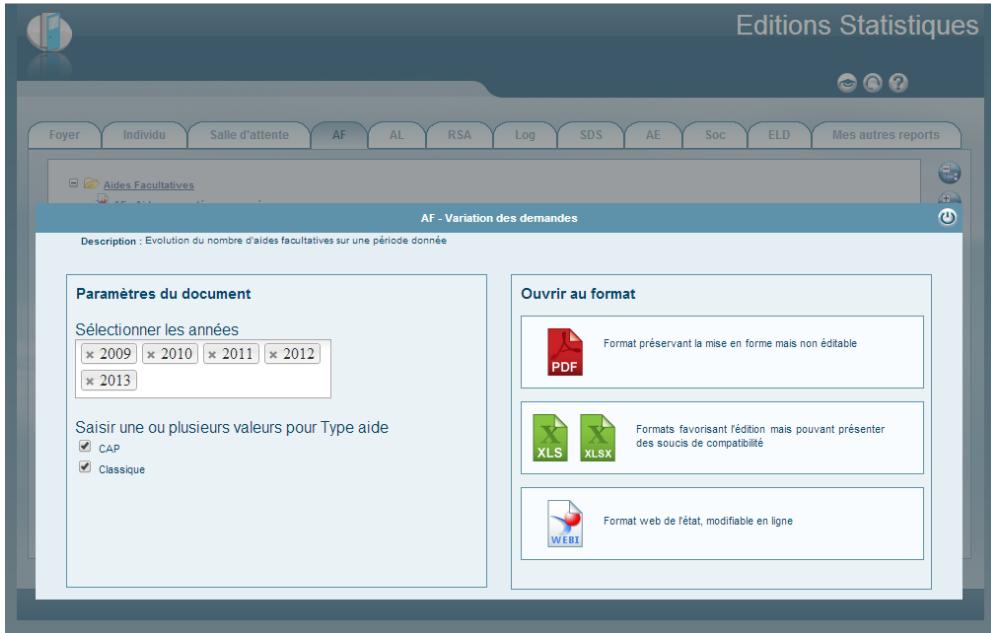


FIGURE 21. – Demande d'accès à un report avec filtres sur l'année et le type d'aide

On touche ici à nouveau à une limite de WebDev : dans sa version 15, il ne permet pas de générer dynamiquement des champs de saisie avec une finesse suffisante. C'est pourquoi je me suis tourné vers le JavaScript (via sa bibliothèque jQuery) : intervenant directement sur le poste utilisateur, il peut générer dynamiquement du HTML très facilement.

Il m'a donc fallu gérer deux connecteurs. On l'a vu (cf. *supra*, **3.2.5 JavaScript, jQuery et Ajax**), il était préférable de ne pas utiliser Ajax pour solliciter le serveur BO, mais plutôt l'application CCAS Web. Ainsi, je ne pouvais récupérer les informations des filtres en JavaScript, ni faire la demande de mise à jour avec ce même langage.

Pour cette raison, il m'a fallu établir un lien entre le serveur WebDev qui va interroger BO et le navigateur client, dans lequel s'exécute le code JavaScript. Ainsi, la fenêtre présentant les filtres dispose de deux champs de saisie cachés. Lors d'une requête en vue de récupérer les filtres, WebDev écrit dans le premier champ le contenu XML renvoyé par BO, permettant ainsi au JavaScript d'accéder aux informations du document.

Interventions dans le Manager

En plus de l'intégration de BO au sens propre, j'ai eu besoin d'intervenir dans le *Manager*. Il s'agit d'un projet annexe, mais néanmoins lié, qui sert notamment de lanceur à l'application, de point d'entrée vers CCAS Web. C'est également via le Manager que l'administrateur dispose d'une interface pour paramétriser le serveur ou créer un utilisateur (*voir* Figure 22).



FIGURE 22. – Le Manager : accès aux paramétrages

Ainsi, c'est via cette application que seront enregistrées les informations du serveur BO — telles que l'url d'accès au serveur, les ports spécifiques pour les envois de requêtes, etc. — ou les informations de connexion d'un utilisateur BO. En effet, pour pouvoir effectuer des requêtes REST, il faut disposer d'un jeton de connexion généré à partir du login BO et du mot de passe associé, qui sont différents de ceux de CCAS Web ; c'est pour cela qu'il faut que l'administrateur lie les comptes dans le Manager.

Base de données

En plus des échanges avec le serveur BO, l'application CCAS Web communique avec une base de données. C'est cette dernière qui va stocker les informations de connexion au serveur BO et les couples identifiant/login de chaque utilisateur Business Objects.

Il m'a donc fallu m'adapter au MCD existant (*voir* Annexe E page XVII) en restant dans une même logique. Dans un système aussi complexe, il n'est pas aussi évident qu'il y parait d'ajouter deux simples colonnes ; en effet, de nombreux éléments dans chacune des pages de l'application sont générés en fonction du contenu des tables. Ainsi, j'ai dû m'inspirer des appels effectués vers la base depuis le *framework*²⁰ et les

20. Un framework est un ensemble cohérent de composants logiciels structurels, qui sert à créer les fondations ainsi que les grandes lignes de tout ou d'une partie d'un logiciel — son architecture.

reproduire.

En l'espèce, plutôt que d'ajouter deux colonnes `LOGIN_BO` et `PWD_BO` dans la table `OL_LOGIN` — celle qui gère les informations d'identification de l'utilisateur — j'ai pu m'appuyer sur trois tables différentes, mais en utilisant des colonnes déjà existantes. Ce choix a eu pour conséquence de simplifier mon développement dans le manager — puisque la zone concernée est générée dynamiquement d'après la base de données (*voir Figure 23*) —, mais également de minimiser les interventions dans la structure de la base de données.

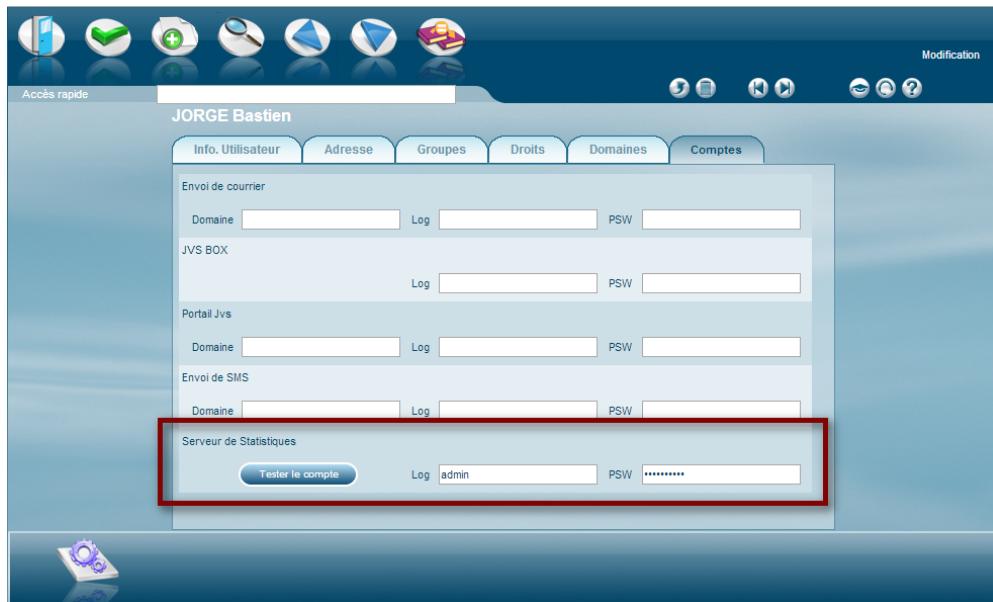


FIGURE 23. – Zone générée dynamiquement par l'ajout d'une occurrence en base de données

Améliorations

J'ai pu proposer une idée d'amélioration qui a été acceptée : établir un token persistant. Pour mieux comprendre cette modification, quelques explications sont nécessaires.

Pour la quasi-totalité des requêtes REST, un paramètre d'en-tête contenant un token²¹ valide est nécessaire. Un token est une chaîne de caractères retournée par le serveur BO suite à une requête lui envoyant un login et un mot de passe. Cette chaîne

²¹. "Token" peut être traduit littéralement par "jeton", qui a ici le sens de jeton de connexion.

sera par la suite reconnue en tant que connexion valide et le serveur pourra donc répondre favorablement aux requêtes qui suivront.

Il faut noter qu'un token a une durée d'expiration : pour ne pas saturer sa mémoire, le serveur purge régulièrement les jetons non utilisés. C'est pour cette raison que nous avons décidé de générer un unique jeton par page qui sera utilisé. On peut trouver trois inconvénients à cela :

- Si l'utilisateur reste trop longtemps sur une même page et réutilise un jeton périmé, la requête n'aboutira pas.
- À chaque chargement d'une page amenée à effectuer une requête, ce jeton est demandé, même s'il n'est pas utilisé.
- Il en découle une perte de rapidité. En lançant un test de performance (*voir Figures 24 et 25*) on s'aperçoit que de manière générale, la requête de récupération d'un token est extrêmement rapide (≈ 20 centièmes de seconde) mais peut parfois dépasser une seconde (cf. *infra* tests 08 et 10). Ce délai est bien trop long si répété régulièrement.

C'est pourquoi j'ai commencé à réfléchir à une alternative. La solution qui semblait la plus évidente était de ne demander qu'un seul token pour tout le projet et le réutiliser quelle que soit la page. Cependant, dans la documentation SAP, il n'existe aucun moyen de récupérer auprès du serveur BO ce délai d'expiration — qui peut être modifié par l'administrateur. Malgré tout, j'ai pu remarquer que, dans la forme du token lui-même, on trouve une valeur égale au nombre de minutes avant destruction : `v=url,UP&66=60` (extrait). Ici, le token expire après 60 minutes.

J'ai donc décidé d'utiliser ce paramètre `66` pour générer automatiquement un nouveau token si besoin. Mais n'étant pas documenté, j'ai sécurisé ce délai en hiérarchisant les priorités :

- Utilisation de la valeur du token donnée par le paramètre `66`.
- Si impossible, on utilise une valeur en base de données correspondant au délai d'expiration modifié par l'administrateur BO.
- S'il n'existe aucune valeur en base de données, alors on utilisera la valeur par défaut, à savoir 60 minutes.

Une fois cette solution trouvée, la phase de développement était terminée : l'intégration était maintenant en phase alpha. En attendant que le service qualité puisse mettre à l'épreuve mon programme avec des tests, j'ai rédigé le manuel utilisateur.

3.2.6. Rédaction du livret utilisateur

C'est en attendant le rendez-vous avec la société *Business Geografic* (*voir 4.1 Organisation temporelle*) que je me suis attelé à la rédaction du manuel utilisateur (*voir Annexe F page XIX*).

L'objectif était de produire une première version du document qui sera fourni aux utilisateurs en même temps que la mise à jour incluant l'intégration de BO. La consigne qui m'a été donnée est la suivante : rédiger un document d'explication pas à pas des trois profils existants — administrateur, créateur de document, lecteur — qui soit le plus compréhensible possible pour un novice. Il ne s'agira en revanche pas de la version

```

C:\Windows\system32\cmd.exe
#####
##### bench.bat #####
#####

Starting test 01 at: 16:05:46,20
Test 01 terminated at: 16:05:47,07
Execution time: .87"
-----
Starting test 02 at: 16:05:47,07
Test 02 terminated at: 16:05:47,27
Execution time: .20"
-----
Starting test 03 at: 16:05:47,27
Test 03 terminated at: 16:05:47,47
Execution time: .20"
-----
Starting test 04 at: 16:05:47,47
Test 04 terminated at: 16:05:47,67
Execution time: .20"
-----
Starting test 05 at: 16:05:47,67
Test 05 terminated at: 16:05:47,88
Execution time: .21"
-----
Starting test 06 at: 16:05:47,88
Test 06 terminated at: 16:05:48,09
Execution time: .21"
-----
Starting test 07 at: 16:05:48,09
Test 07 terminated at: 16:05:48,29
Execution time: .20"
-----
Starting test 08 at: 16:05:48,29
Test 08 terminated at: 16:05:48,79
Execution time: 1.50"
-----
Starting test 09 at: 16:05:48,80
Test 09 terminated at: 16:05:49,09
Execution time: .29"
-----
Starting test 10 at: 16:05:49,09
Test 10 terminated at: 16:05:52,96
Execution time: 3.87"
Appuyez sur une touche pour continuer...
  
```

FIGURE 24. – Benchmark d'une requête de récupération du token

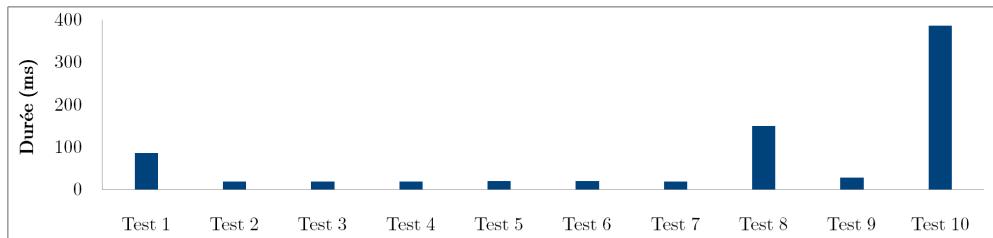


FIGURE 25. – Durée par exécution

définitive puisque les formateurs se le réapproprieront pour le remanier et l'adapteront au métier, mais il servira de base au document final.

Dans le même temps, j'ai assisté à une réunion de présentation des nouveautés aux commerciaux de l'entreprise. Pascal BALIGOUT a repris une par une les fonctionnalités apportées suite à l'intégration BO, lesquelles ont reçu un accueil très favorable de leur part.

Enfin, j'ai pu passer à la dernière étape de l'intégration.

3.2.7. Phase de recettage et corrections

Le recettage est la phase de tests que le Service Qualité a fait subir au code que j'ai écrit. Ainsi, François MAROTTE a eu une semaine pour mettre à l'épreuve les pages que j'ai développées et générer un maximum d'états, y compris avec des paramètres aberrants, pour analyser les erreurs de code et les corrections à apporter.

Au final, j'ai eu assez peu d'améliorations à apporter (une dizaine de points). Cela peut se comprendre par le fait que j'ai eu beaucoup plus de temps pour travailler sur ce projet que ce dont dispose habituellement l'équipe de développeurs ; j'ai ainsi pu débugger naturellement mon code au fur et à mesure de l'intégration.

3.3. Bilan de l'intégration et évolutions envisagées

Lors de ma dernière semaine de stage, un transfert de connaissance avec Steeve GAUMONT, développeur sur le projet CCAS Web, a été programmé. En effet, il m'a fallu partager la technique acquise pour pouvoir la reporter dans le produit final, ou pour être à même de la maintenir.

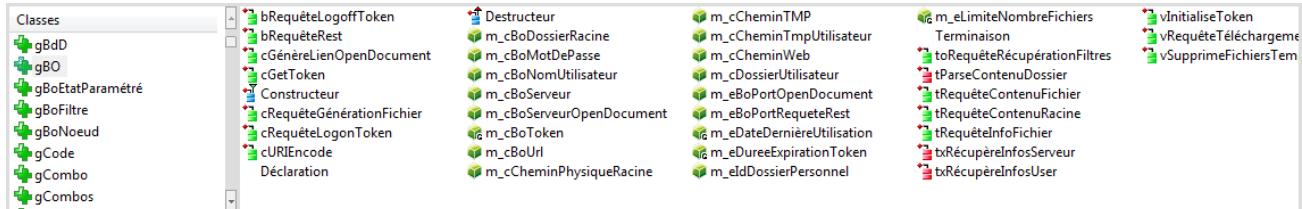


FIGURE 26. – Classe gBo dans WebDev, un des éléments transmis à Steeve

Par ailleurs, il m'a été demandé de rédiger un document destiné aux développeurs pour y consigner tous les apports que j'ai intégré afin d'assurer un suivi dans l'application.

En outre, le but de l'intégration de BO était d'apporter une fonctionnalité demandée depuis longtemps par les utilisateurs, même si elle ne concernera pas les plus petits d'entre eux. Ainsi, il est probable qu'elle évolue encore suite aux retours des différentes mairies, une fois la solution mise en production.

Enfin, j'ai pu aussi proposer des pistes d'amélioration, notamment concernant une notification sur les rapports planifiés (*voir Annexe C, page IX*). Elles seront étudiées plus tard en fonction de la disponibilité du service développement et de l'estimation du niveau de besoin de cette nouveauté.

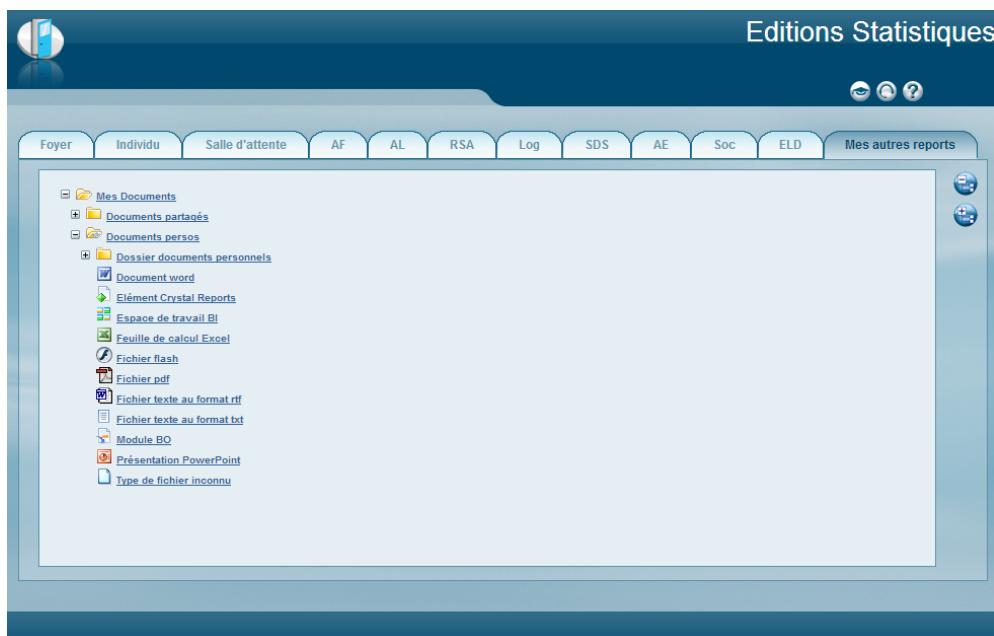


FIGURE 27. – Aperçu : documents stockés dans BO visibles sous CCAS Web

4. L'implémentation de GeoBI

Ma mission principale ayant été terminée plus tôt que prévue, j'ai pu m'atteler à la seconde tâche de mon stage : implémenter le connecteur GeoBI (cf. *infra*, **4.2.1 GeoBI, surcouche de BO** pour comprendre ce connecteur).

4.1. Organisation temporelle

L'organisation de cette deuxième mission a été moins évidente du fait de la nécessité de rencontrer la société *Ciril*²² avant de commencer l'intégration. En effet, étant localisée à Lyon, il a fallu convenir d'une rencontre à une date durant laquelle leur équipe de développeurs, mon responsable et moi-même étions disponibles.

Ce rendez-vous pris, nous avons pu récupérer les informations et les ressources fournies par Business Geografic. Les tâches ont été réparties ainsi :

- Lecture des différents manuels utilisateurs fournis par Business Geografic.
- Installation de GeoBI sur le serveur Tomcat de BO et paramétrage de l'application.
- Téléchargement puis importation des couches cartographiques.
- Création de nos propres fichiers de cartographie.
- Présentation du réalisable à Richard THOUVENOT, DG d'Implicit.

4.2. Choix d'implémentation et difficultés rencontrées

Ici, peu de développement aura été nécessaire. S'agissant d'une surcouche à BO, il sera principalement question de paramétrages.

4.2.1. Pourquoi GeoBI ?

Dans un premier temps, nous allons voir ce qu'est GeoBI réellement, pour ensuite en comprendre l'utilité.

Découverte de GeoBI

Le 11 février 2014, soit une semaine après mon arrivée dans l'entreprise, avait été programmé un webinar²³ animé par *Décivision*, une SSII spécialisée en informatique décisionnelle et partenaire distributeur de SAP Business Objects.

22. Ciril est le groupe propriétaire de l'éditeur Business Geografic, qui propose notamment GeoBI parmi sa gamme progicielle.

23. Un webinar est une conférence qui se tient à travers le réseau Internet.



FIGURE 28. – Logo de l'entreprise Décivision

Cette webconférence avait pour principal objet de présenter les nouveautés apportées par la version 4.1 de BO (alors sortie depuis peu) ; mon objectif personnel était différent puisque je m'étais fixé comme but de mieux appréhender cet outil, découvert seulement quelques jours auparavant.

Néanmoins, lors de cette formation à distance, les intervenants ont présenté une fonctionnalité qui a rapidement attiré l'attention de Pascal BALIGOUT et d'Emmanuel VATTAT : une surcouche à BO permettant de produire de la statistique géographique. Cette extension ne leur était alors déjà pas inconnue, mais c'était la première fois qu'elle s'intégrait avec la version web de Business Objects.

Il s'agissait de GeoBI.

GeoBI, surcouche de BO

GeoBI est un plugin cartographique de BO développé par Business Geographic. Il s'agit d'une extension de Business Intelligence facilitant encore plus les prises de décision en proposant à l'utilisateur un support très visuel, reposant sur une carte géographique.



FIGURE 29. – Logos de la société Business Geografic, et de leur produit GeoBI

Le site web de GeoBI décrit son logiciel comme « un outil d'analyse, d'aide à la décision et de simulation cartographique, exploitant directement les données contenues dans [les] rapports d'Entreprise BusinessObjects »[7].

Son principal attrait est donc d'offrir une solution pleinement intégrée aux données présentes dans BO.

L'importance d'un connecteur cartographique

Toujours dans un souci de proposer aux utilisateurs une application la plus complète possible, il apparaît intéressant d'intégrer cette solution au sein de CCAS Web. Dans la même optique d'obligation légale des CCAS de fournir une Analyse des Besoins Sociaux

(ABS) annuellement, l'extension à BO est très intéressante et permet de nombreuses possibilités. L'intérêt là encore est de ne pas réinventer la roue et de s'appuyer sur des technologies éprouvées.

On constate par exemple très rapidement l'importance d'une telle extension pour des localités qui travaillent beaucoup sur de l'intelligence géographique : il suffit de comparer les données statistiques brutes (*voir* Figure 30 page suivante) avec les mêmes reportées sur une carte (*voir* Figure 31 page suivante) pour se rendre compte du côté indispensable de GeoBI.

Forts de ces observations, nous avons décidé de rencontrer l'éditeur de GeoBI afin de s'assurer de la compatibilité de leur logiciel avec notre solution et des conditions d'implémentation.

4.2.2. Rencontre de l'entreprise ***Business Geografic***

Début juin, mon responsable et moi-même avons rencontré sur Lyon le directeur du bureau d'études et le chef de projets de Business Geografic.



FIGURE 32. – Page d'accueil du Designer GeoBI

Ils nous ont présenté l'outil durant la matinée et ont déroulé devant nous tout un panel de solutions qui allaient bien au delà de ce à quoi nous nous attendions. Le résultat a été assez probant et nous avons décidé de tester leur application dans notre environnement.

4.2.3. Premier essai

Une fois installée sur le serveur Tomcat (*voir* Figure 33) et paramétrée — installation extrêmement aisée et rapide en mode graphique — nous nous sommes rapidement lancés dans un premier essai d'utilisation de GeoBI. La première étape a été de me rendre dans le *Designer GeoBI*, l'interface de paramétrage des données. En effet, nativement, le logiciel ne dispose d'aucun contenu de découpage géographique : il m'a donc fallu en importer.

Analyse des logements et des flux régionaux

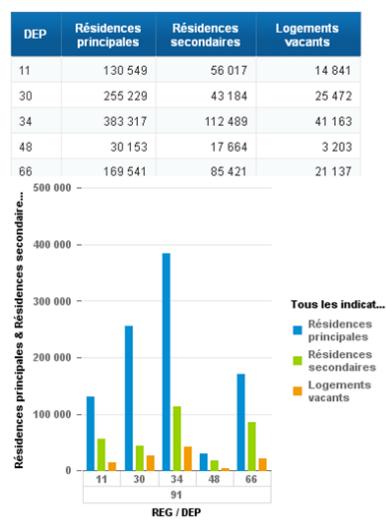


FIGURE 30. – Exemple de statistique fournie par Business Geografic

Analyse des logements et des flux régionaux

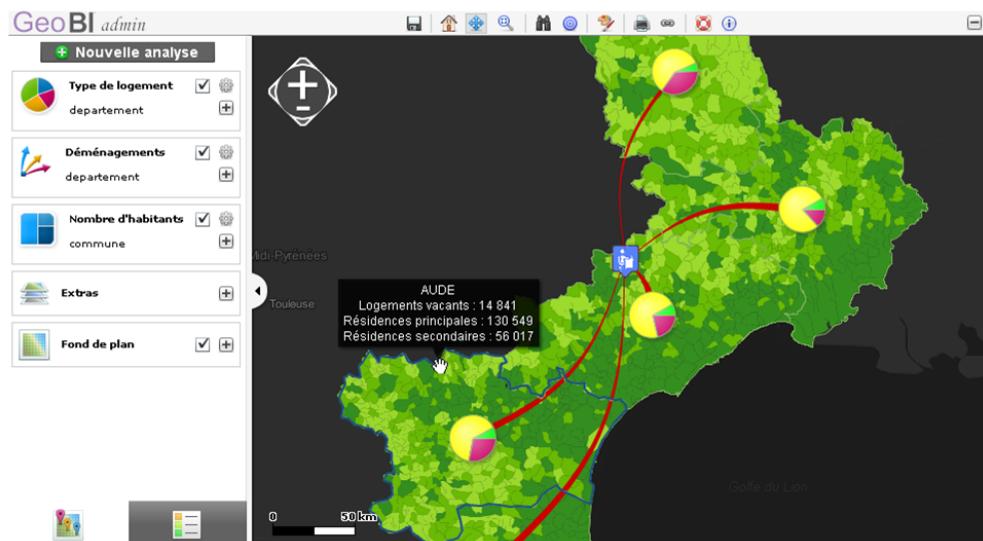


FIGURE 31. – Cartographie issue des statistiques de la *Figure 30*

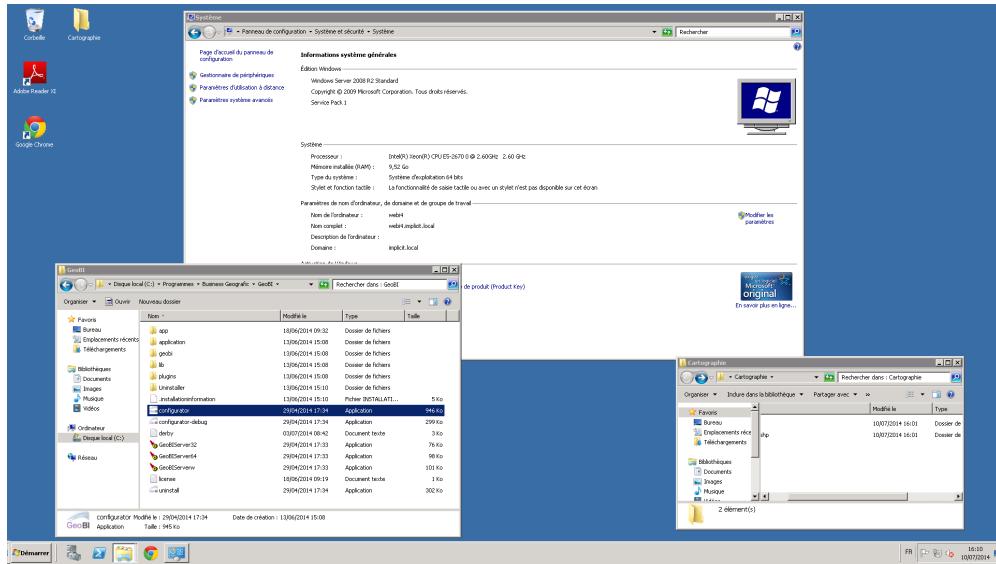


FIGURE 33. – Aperçu du Windows Server 2008 sur lequel est installé GeoBI

J'ai néanmoins réussi à parvenir à un résultat exploitable en à peine une heure, l'outil étant vraiment intuitif.

Import de fichiers de découpage géographique

Pour cette première étape, j'ai d'abord dû effectuer quelques recherches sur le format de ces fichiers avant de les importer de manière effective. Ce format spécifique est appelé *Shapefile*.

• **Le format Shapefile** est un peu particulier. Pour bien le comprendre, quelques notions de cartographie informatique sont nécessaire — attention néanmoins : ces notions sont volontairement schématisées afin d'en saisir l'essentiel, je n'ai nullement la prétention d'effectuer ici un exposé sur la cartographie.

Le format Shapefile, initialement développé par ESRI²⁴ pour ses logiciels commerciaux, est rapidement devenu un standard *de facto*. Il est composé de plusieurs fichiers — dans notre cas, au moins 4 — : .shp (métadonnées), .shx (index de la géométrie), .dbf (données relatives aux objets du shapefile) et .prj (informations sur le système de coordonnées). Ces fichiers comportent donc divers contours vectoriels (d'où *shapes*)

24. Environmental Systems Research Institute est la société qui a inventé le concept logiciel des SIG (Système d'Information Géographique) et donc le Shapefile.

ainsi que des informations associées à chaque zone — point, ligne, polygone, etc. — sous forme de couple clé–valeur.

Pour notre projet, nous allons différencier deux notions :

- Le fond de carte (*voir* Figure 34-1) : il peut être statique ou dynamique (chargement par "tuiles", différents niveaux de précision en fonction du zoom...) et comportera généralement les villes, les rues, leurs noms, etc. Il ne s'agit donc pas de Shapefile.
- Les couches cartographiques (*voir* Figure 34-2) : elles représentent des découpages de zones, ce sont elles qui comportent les données clé–valeur et qui sont importées dans GeoBI.



FIGURE 34. – À gauche (1) : fond de carte du centre de Troyes - À droite (2) : découpage arbitraire du centre de Troyes

Une fois ces informations assimilées, il m'a fallu trouver des couches Shapefile utilisables.

• **GéoPortail, le site de données cartographiques** est la source française de référence en matière de géographie territoriale. Il est l'œuvre de deux établissements publics, l'IGN²⁵ et le BRGM²⁶.



FIGURE 35. – Logo du site GéoPortail

25. Institut national de l'information géographique et forestière, qui a pour mission d'assurer la production, l'entretien et la diffusion de l'information géographique de référence en France.

26. Bureau de recherches géologiques et minières, l'organisme public français de référence dans le domaine des sciences de la Terre pour la gestion des ressources et des risques du sol et du sous-sol.

Ce site met à disposition différents types de découpages, notamment certains gratuits : régions, départements, communes. Je me suis donc appuyé sur ces shapefiles pour effectuer mon premier test.

Premiers résultats

Le paramétrage en ligne se fait très rapidement : il m'a suffi d'uploader les fichiers Shapefile dans le Designer (l'interface de GeoBI) pour les superposer à un fond de carte *OpenStreetMap*²⁷ (OSM) et voir se dessiner les découpages des communes.

La deuxième étape a été de lier une clé de la couche cartographique avec un élément issu de l'Univers BO, pour faire une jonction entre la zone physique sur la carte et la zone définie par une donnée particulière. Ici, nous avons choisi le code INSEE de la commune, puisqu'il est unique, et qu'à la différence des noms de villes il ne peut pas y avoir d'ambiguïté (par exemple, comment savoir si la casse dans le Shapefile est « Ste Savine », « SAINTE-SAVINE » ou « Sainte Savine » ?). Cependant, bien qu'existant dans la base de données, cette information n'était pas présente dans l'Univers BO. Il a donc fallu procéder à une mise à jour pour intégrer cet objet.

J'ai ensuite effectué une requête très simple : ressortir le nombre de bénéficiaires du CCAS d'une grande ville du Sud de la France — nous disposons déjà de la DB associée — en fonction de la commune de leur foyer. Si le résultat dans un tableau n'est pas parlant, une carte de chaleur est parfaitement adaptée (*voir* Figure 36).

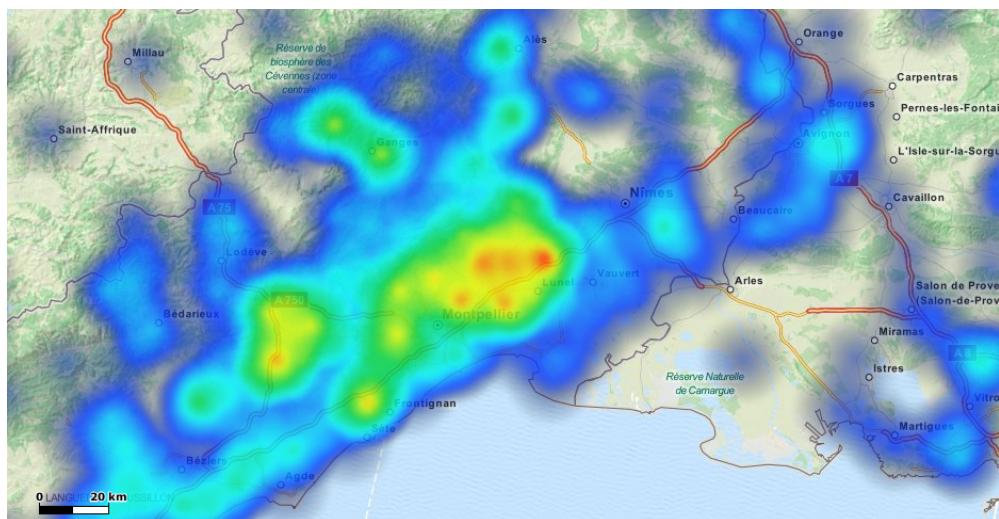


FIGURE 36. – Répartition des bénéficiaires d'un CCAS sous forme de zone de chaleur

27. Projet open-source qui a pour but de constituer une base de données géographique mondiale libre de droits. Plus d'informations sur le wiki dédié : <http://wiki.openstreetmap.org/wiki/> [en].

À la vue de ces résultats plus que satisfaisants, je me suis attelé à la génération de nos propres fichiers Shapefile, la finalité étant de proposer aux utilisateurs, par exemple, un découpage par quartier.

4.2.4. Utilisation de Shapefiles personnalisés

Cette phase est probablement celle qui m'a posé le plus de problèmes au cours de cette seconde mission. En effet, la communauté des utilisateurs de SIG ne jure que par un seul logiciel pour générer des Shapefiles : *ArcGIS*. Édité par ESRI — l'inventeur du format —, ce programme est très puissant (beaucoup trop pour nos besoins), mais aussi très cher. Il m'a donc fallu trouver une solution équivalente, et de préférence dans le domaine open-source, tout du moins durant cette phase de tests.



FIGURE 37. – Logo du logiciel ArcGIS

Malheureusement, il n'existe que très peu de ressources sur la conception de Shapefiles, et encore moins de logiciels spécialisés sur cette activité. Certains permettent de créer des contours, mais pas de les poser sur un fond de carte OSM (donc impossible de dessiner correctement les frontières des zones), d'autres ne donnaient pas la possibilité d'intégrer un attribut clé-valeur (aucun moyen de l'identifier avec un objet BO), d'autres enfin étaient livrés sans documentation et je n'ai jamais réussi à les utiliser correctement.

J'ai donc fait de nombreuses recherches, et subséquemment, me suis rabattu sur une solution alternative. Il existe un logiciel, JOSM (pour Java OpenStreetMap) qui permet de créer ces polygones sur une carte OSM et de leur associer des identifiants.



FIGURE 38. – Logo du logiciel JOSM

Malgré cela, le logiciel n'est pas capable d'enregistrer le résultat au format Shapefile : il ne peut le faire qu'en .osm — dont le contenu n'est autre que du XML. Il m'a fallu passer une fois de plus par des moyens détournés.

J'ai découvert sur GitHub un script python²⁸ que j'ai adapté pour qu'il ne soit plus spécifique au fichier de démonstration, et qui permet de transformer un document XML en Shapefile fonctionnel.

Grâce à ce moyen détourné, j'ai pu fournir à Implicit un outil leur permettant de générer leurs propres découpages de communes.

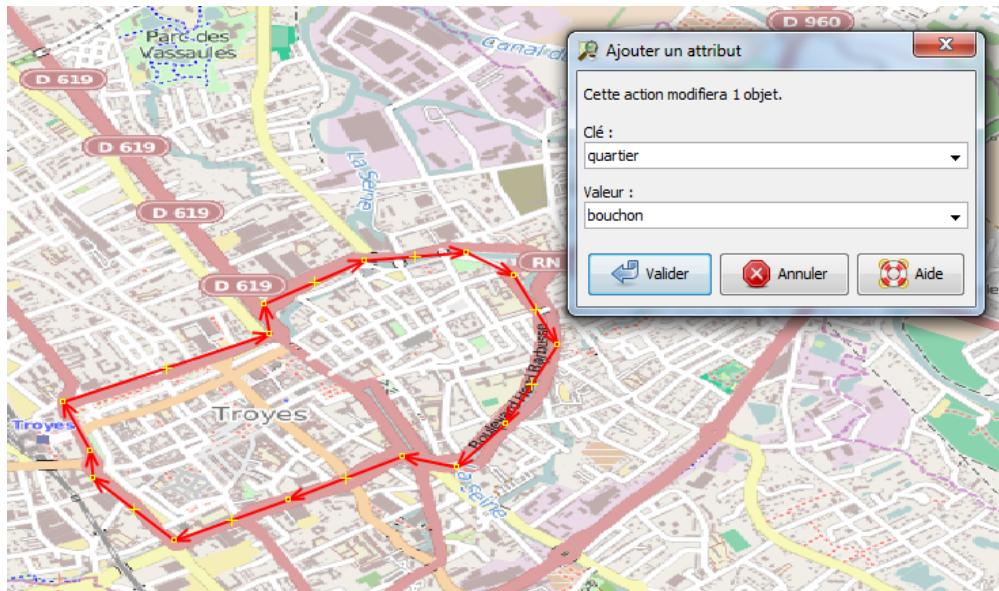


FIGURE 39. – Affectation d'un attribut d'une couche dans JOSM

4.3. Bilan de l'intégration et évolutions envisagées

Le 8 juillet 2014, mon travail sur la BI géographique a été produit devant le comité « synergie PPS²⁹ » avec la présence de Jacques-Yves HARSCOUËT, Directeur Général du groupe sur le marché Action Sociales.

Lors de cette réunion a été abordée la monétisation de l'implémentation. En effet, cela faisait plusieurs années qu'Implicit — et à fortiori les sociétés éditrices de solutions informatiques du groupe — cherchait à développer un système similaire sans succès, et

28. Code disponible sur gist.github.com/robintw/9366322

29. Programmes Publics et Sociaux

les différents tests que j'ai pu effectuer ont fait la preuve qu'au niveau technique, de nombreuses possibilités étaient offertes.

Il a été décidé de l'utiliser dans un premier temps pour créer via BO des ABS génériques. Pour rappel, l'Analyse des Besoins Sociaux est le dossier qu'un CCAS est tenu de présenter tous les ans sur les différentes évolutions démographico-sociales de la commune afin d'envisager le plan d'action pour les années à venir. Cependant, de nombreux CCAS y consacrent des ressources non négligeables qui pourrait être diminuées, d'autres au contraire passent trop peu de temps dessus pour fournir un document de qualité.

Ainsi, Implicit va leur proposer une étude pré-formatée et générée via un état BO, qui exploitera les données propres à chaque CCAS. Pour créer ces états, il a été décidé de prendre contact durant l'été avec les clients privilégiés de la société pour décider avec eux de ce que devra contenir cet ABS généré.

Enfin, il est à noter que certaines évolutions vont également voir le jour, basées sur l'outil GeoBI. Ce dernier il dispose d'une fonction très utile appelée isochrone, laquelle permet de dessiner une zone autour d'un point donné, comportant tous les endroits accessibles en fonction de la durée nécessaire pour s'y rendre (*voir Figure 40*). Son utilisation finale reste néanmoins à définir.

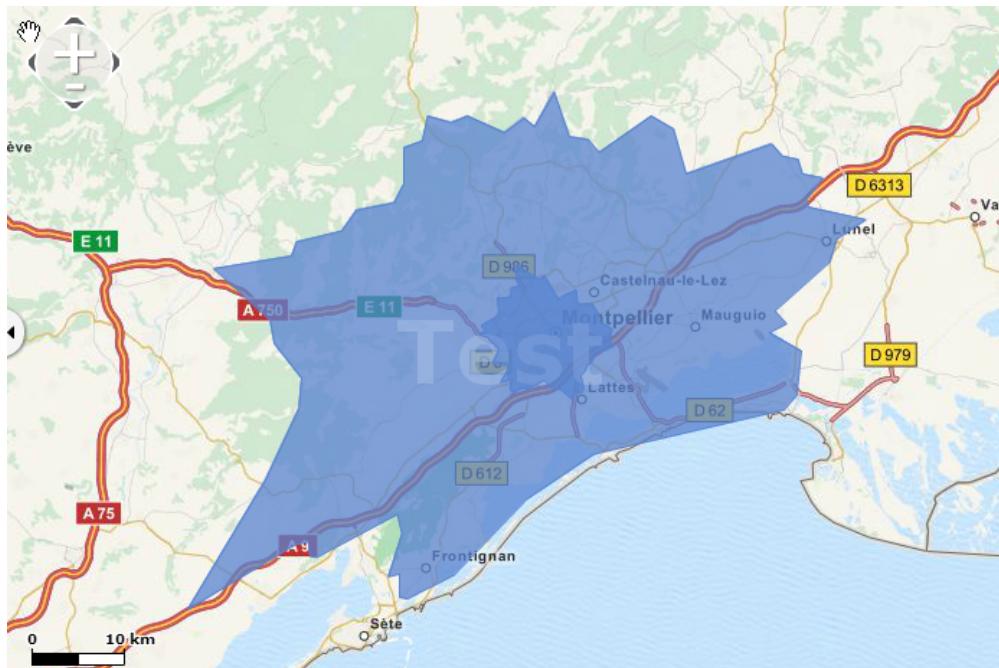


FIGURE 40. – Isochrone des zones accessibles en 15 et 30 minutes depuis le centre-ville de Montpellier

Bilan du stage

Ce stage au sein de l'entreprise Implicit a été très agréable pour moi. J'ai pu en tirer de nombreuses leçons malgré les difficultés auxquelles j'ai pu être confronté, mais aussi grâce à elles.

Enrichissement personnel

Avoir développé deux solutions qui seront intégrées à la gamme d'Implicit, savoir que mon travail est reconnu et utile — les deux nouveaux modules seront présentées lors de webconférences en fin d'année, et lors de rencontres ciblées par les chargés de clientèle — resteront des expériences professionnelles très valorisantes pour moi.

J'ai en effet beaucoup appris et me suis personnellement enrichi : méthodes de travail (projet en équipe, dates butoir à respecter, méthodes agiles...), apprentissage de nouveaux langages, perfectionnement dans d'autres, application de cours étudiés au long de mon année d'études à l'Université de Technologie de Troyes (analyse et conception) ou découverte de nouveaux domaines (Business Intelligence, Système d'Informatisation Géographique...). Tout cela m'a conforté dans mon projet professionnel.

L'ambiance de travail m'a aidé à m'épanouir dans cet univers de l'entreprise. Je tire également une certaine fierté d'avoir pu réaliser la deuxième mission de mon stage, en dépit du caractère facultatif qu'elle revêtait à l'origine. J'ai pu saisir l'opportunité d'expérimenter de nouvelles technologies très intéressantes.

J'ai également pu tirer d'autres compétences personnelles grâce à diverses discussions avec mes collègues, dans des domaines qui n'ont pas attrait aux activités de l'entreprise, comme l'amélioration de mes connaissances en php ou la découverte de l'administration linux.

Difficultés rencontrées

À mon arrivée, j'appréhendais un peu la prise en main d'un nouveau langage et la phase de recherche concernant BO, de crainte de ne pas y arriver. Mais la confiance qu'ont pu m'accorder Emmanuel VATTAT et Pascal BALIGOUT m'a permis de m'intégrer très rapidement au projet, et j'ai ainsi pu être opérationnel très peu de temps après le début de mon stage.

J'ai aussi pu être confronté à des difficultés du quotidien de l'entreprise, comme lorsque le serveur Oracle est tombé (deux fois durant mon stage), me forçant alors l'espace d'une journée au chômage technique.

Les autres difficultés — éliminations successives de technologies inadéquates (*voir 3.2.5 Le choix de la technologie du requêtage REST*), peu d'informations sur la

génération de fichiers personnels (*voir 4.2.4 Utilisation de Shapefiles personnalisés*) — sont, à mon sens, à considérer plutôt comme un défi que comme des problèmes à proprement parler, et les surmonter a été d'autant plus enrichissant.

Réponse à la problématique

L'enjeu du stage était de participer à l'amélioration continue de l'outil CCAS Web, pour répondre aux demandes des utilisateurs.

Je pense que l'objectif a été non seulement rempli, mais dépassé, dans la mesure où l'implémentation de la BI géographique n'émerge pas d'une demande des travailleurs sociaux ou des équipes dirigeantes des CCAS, mais d'une volonté de l'entreprise d'aller toujours plus loin dans les services qu'elle propose. Cela permettra peut-être de toucher de nouveaux clients potentiels.

Deuxième partie

Annexes

A. Fiche commerciale de l'application Millésime Action Sociale

Cette annexe est le document qui sert de support commercial pour présenter l'outil CCAS Web aux potentiels clients (*voir* page III).



AIDE SOCIALE ET ACCOMPAGNEMENT



Les Plus

- **Dossier unique**
- Personnalisation de l'espace de travail
- **Navigation simple, efficace et intuitive**
- **Traitements en temps réel** des commissions
- Dispatching logement
- Alertes paramétrables
- **Saisie mode formulaire**
- Téléprocédures sécurisées

Vous souhaitez en savoir plus sur notre solution

Action sociale

Notre équipe se tient à votre disposition

03 26 26 67 30

contact@implicit.fr

8 rue Raymond Aron
Saint-Martin sur le Pré - CS 40222
51010 CHALONS-EN-CHAMPAGNE Cedex



www.implicit.fr

OBJET DE L'APPLICATION

- Suivi de l'usager
- Coordination des dispositifs sociaux
- Extraction et exploitation des données



Déploiement

- Environnement technique standard (compatible avec les versions Windows XP à Windows 8)
- Installation monoposte ou sur le réseau de la collectivité
- Possibilité d'hébergement



Fonctionnalités

- Accueil et orientation
- Agenda collaboratif
- Aide légale et facultative
- Préparation et suivi des commissions
- Gestion des CAP
- RSA (insertion et suivi)
- Logement Social interfacé avec le portail national
- Suivi social et accompagnement
- Domiciliation
- Gestion des droits d'accès



Traitements

- Numérisation des pièces justificatives
- Edition des imprimés légaux
- Tableaux de bord et statistiques
- Fusions bureautiques (compatible Microsoft Office et Open Office)
- Archivage pour mise en conformité CNIL
- Dématérialisation (interface Département, comptabilité...)

B. Étude du réalisable

L'Annexe B est un extrait de l'étude du réalisable effectuée en début de stage, document sur lequel je me suis appuyé pour définir le cahier des charges.

Sont inclus :

- Le sommaire, aperçu rapide du contenu
- Un exemple des paramètres à passer pour une requête
- Les explications de téléchargement du document en ligne, avec redirection de la sortie standard qui m'a posé problème.
- Les informations sur la création d'utilisateur en REST, malheureusement impossible.
- Les consignes pour ouvrir un document sans connexion préalable, via une URL (pas de sollicitation REST, mais vitale pour le projet final)



Etude du réalisable

Intégration de SAP Bo
dans Millésime CCAS version web

Millésime CCAS version web



Sommaire

1	Versions	3
2	Introduction.....	3
a.	Produit concerné	3
b.	Objet	3
c.	Notations	3
3	Paramétrages préalables.....	3
4	Connexion / Déconnexion : utilisation de <i>tokens</i>	4
a.	Connexion.....	4
b.	Déconnexion.....	5
5	Naviguer dans l'arborescence BO.....	6
6	Modifier les filtres d'un webi.....	9
7	Télécharger un webi.....	11
a.	pdf	11
b.	xls	11
c.	xlsx	11
8	Créer un nouveau dossier / webi	12
a.	Nouveau dossier	12
b.	Nouveau webi.....	12
9	Supprimer un webi	13
10	Duplicer un webi	14
11	Création d'un utilisateur / gestion des droits.....	15
12	Personnalisation de l'interface.....	16
13	Accès direct aux éléments	17
a.	Ouvrir un document sans se connecter.....	17
b.	Accéder au Launchpad (ancien InfoView) sans se connecter.....	17
c.	Accéder au CMC sans se connecter	17
14	Technologies envisagées	18
a.	Requêtes HTTP en WLangage.....	18
b.	Utilisation d'AJAX avec JQuery (JavaScript).....	18
c.	Appel à une dll externe avec la fonction API()	18
d.	Utilisation de CURL côté serveur via un fichier batch.....	18
15	Intégrations envisagées.....	19
a.	Parcours de l'arborescence des fichiers disponibles pour l'utilisateur	19
b.	Ouverture directe en édition d'un webi.....	19
c.	Téléchargement du webi.....	19
d.	Onglet "Infos" : derniers documents mis à jour	19

1 Versions

Date	Version	Evolutions	Auteur
05/02/2014	0.1	Création	Bastien Jorge
20/03/2014	0.2	Retour E. Vattat	Bastien Jorge

2 Introduction

a. Produit concerné

Cette étude concerne le produit Business Object XI 4.1 de la société SAP.

b. Objet

Ce document décrit les possibilités d'accès, d'édition, de création et de suppression offertes par l'API de BO en accès RESTful, sans connexion préalable via l'interface de l'outil.

c. Notations

Lors des requêtes ci-dessous, des variables sont utilisées : %var% dans le code, <nom> dans l'url. Il faudra les remplacer par les valeurs correspondantes (6405 pour <port> lors des requêtes REST).

3 Paramétrages préalables

Pour pouvoir fonctionner correctement, les requêtes REST nécessitent des interventions :

- Ouvrir les ports 6400 – 6405 – 6410 (serveur et firewall) en entrant et sortant
- Autoriser REST (dans \Business Objects\Tomcat55\webapps\dswsbobje\WEB-INF\conf, éditer axis2.xml, remplacer


```
<parameter name="disableREST" locked="true">true</parameter> par
<parameter name="disableREST" locked="true">false</parameter>, sauver et redémarrer le serveur.)
```
- Vérifier que les fichiers *jquery.min.js* et *bo_filtres.js* sont bien présents dans *MOL_CCAS_WEB*, ainsi que le plugin *select2*
- Installer cURL sur la machine dans le dossier *Exe*
- Déposer le batch *Exe\curl\curl_exec.bat*

4 Connexion / Déconnexion : utilisation de tokens

Avant toute utilisation de BO, il faut pouvoir s'identifier pour accéder aux diverses fonctionnalités. C'est là l'utilité du jeton de connexion.

a. Connexion

Pour générer un jeton de connexion, il faut effectuer la requête REST suivante :

Adresse : <http://<domaine>:<port>/biprws/logon/long>

Requête POST	
Header	
Key	Value
Accept	application/xml
Content-Type	application/xml

Accept contient le format de la réponse retournée par le serveur BO

Content-Type contient le format des données envoyées au serveur BO

Body

```
<atrs xmlns="http://www.sap.com/rws/bip">
  <attr name="userName" type="string">
    %id%
  </attr>
  <attr name="password" type="string">
    %password%
  </attr>
  <attr name="auth" type="string">
    possibilities="secEnterprise,secLDAP,secWinAD">
      secEnterprise
    </attr>
  </atrs>
```

Le premier attribut (*userName*) contient le nom de l'utilisateur (sensible à la casse), le deuxième (*password*) contient le mot de passe en clair, et le troisième (*auth*) est le type d'authentification (laisser *secEnterprise* normalement)



Retour du serveur de la forme de :

Header	
Key	Value
Server	Apache-Coyote/1.1
X-SAP-LogonToken	"webi4.implicit.local:6400@{3&2=6337,U3&2v=webi4.implicit.local:6400,UP&66=60,U3&68=secEnterprise:stagiaire,UP&S9=5820,U3&qe=100,U3&vz=Y61x6aZTvepnBjlcMLu_.sli <div_aeyf7uapo4icw2e,up}"< td=""></div_aeyf7uapo4icw2e,up}"<>
Date	Wed, 05 Feb 2014 07:41:24 GMT
Content-Type	application/xml
Content-Length	588

Le seul élément important dans le header du retour est le contenu de X-SAP-LogonToken qui contient le jeton de connexion.

Body	
<pre><entry xmlns="http://www.w3.org/2005/Atom"> <author> <name>@webi4.implicit.local:6400</name> </author> <id>tag:sap.com,2010:bip-rs/logon/long</id> <title type="text">Résultat de connexion</title> <updated>2014-02-05T07:41:24.733Z</updated> <content type="application/xml"> <attr xmlns="http://www.sap.com/rws/bip"> <attr name="logonToken" type="string"> webi4.implicit.local:6400@{3&2=6337,U3&2v=webi4.implicit.local:6400,UP&66=60,U3&68=secEnterprise:stagiaire,UP&S9=5820,U3&qe=100,U3&vz=Y61x6aZTvepnBjlcMLu_.sli<div_aeyf7uapo4icw2e,up} <="" attr>="" content>="" entry><="" pre=""> </div_aeyf7uapo4icw2e,up}></pre>	

On vient de récupérer le token (on n'a pas besoin du body pour l'exploiter).

b. Déconnexion

Une fois le jeton utilisé, il peut être utile de le détruire. Il faudra alors utiliser la requête REST suivante :

Adresse : <http://<domaine>:<port>/biprws/logoff>

Requête POST	
Header	
Key	Value
Accept	application/xml
Content-Type	application/xml
X-SAP-LogonToken	"webi4.implicit.local:6400@{3&2=6337,U3&2v=webi4.implicit.local:6400,UP&66=60,U3&68=secEnterprise:stagiaire,UP&S9=5820,U3&qe=100,U3&vz=Y61x6aZTvepnBjlcMLu_.sli <div_aeyf7uapo4icw2e,up}"< td=""></div_aeyf7uapo4icw2e,up}"<>

X-SAP-LogonToken contient le jeton à détruire.

7 Télécharger un webi

Il est possible de télécharger un webi, sous plusieurs formats finaux, notamment .pdf, .xls et .xlsx

Adresse :

[http://<domaine>:<port>/biprws/raylight/v1/documents/<id>\[parameters\]](http://<domaine>:<port>/biprws/raylight/v1/documents/<id>[parameters])

Requête GET

Header	
Key	Value
Accept	application/%format%
Content-Type	application/xml
X-SAP-LogonToken	%token%

a. pdf

- Pas de paramètres
- %format% = "pdf"

b. xls

- Paramètres : "optimized=true&dpi=150"
- %format% = "vnd.ms-excel"

c. xlsx

- Paramètres : "optimized=true"
- %format% = "vnd.openxmlformats-officedocument.spreadsheetml.sheet"

Dans tous les cas, il faudra rediriger la sortie standard vers un fichier pdf, xls ou xlsx.



11 Cr ation d'un utilisateur / gestion des droits

Via l'API RESTful, il sera impossible de g erer les utilisateurs. Une tentative de solution a cependant  t e imagin e apr s navigation dans le dossier utilisateurs : en effet, les utilisateurs eux-m mes sont consid r s comme des fichiers. Ainsi, la possibilit  de cr er manuellement ce type de fichier (avec les dossiers favorites_folder, inbox, etc., associ s, les lier   des groupes...) a  t e envisag e , mais deux probl mes se sont pos s s s :

- Il n'est,   notre connaissance, pas possible de cr er des dossiers via l'API RESTful
- Cr er des utilisateurs manuellement peut s'av rer dangereux en cas de changement de syst me de fonctionnement par SAP.

Mais il serait cependant   priori possible d'effectuer une gestion des utilisateurs via l'API Java (<http://scn.sap.com/community/bi-platform/java-sdk/blog/2013/05/14/scripts-for-user-management>). Il faudra cependant se demander s'il est r ellement utile de proposer cette gestion directement sur l'outil Millesime ou s'il vaut mieux g rir les autorisations via le CMC.

De plus, la prise en main de l'API Java est relativement complexe pour l'usage de cette seule fonctionnalit .

13 Acc s direct aux  l ments

a. Ouvrir un document sans se connecter

Il est possible d'ouvrir un document sans avoir   se connecter via l'interface BO, gr ce   un token. Cela permet notamment d'entrer en fen tre  dition du document (cas d'un webi) ou de visualiser le document via l'interface BO (cas d'un pdf par exemple).

Il suffit d'avoir un token disposant des droits n cessaires pour l'ouverture du webi.

L'url appel e sera la suivante :

[http://\(1\):\(2\)/BOE/OpenDocument/opendoc/openDocument.jsp?iDocID=\(3\)&sidType=CUID&X-SAP-LogonToken=\(4\)](http://(1):(2)/BOE/OpenDocument/opendoc/openDocument.jsp?iDocID=(3)&sidType=CUID&X-SAP-LogonToken=(4))

(1) : Nom ou adresse IP du serveur BO (ex : localhost)

(2) : Port du serveur de jsp utilis  par BO (ex : 8080)

(3) : CUID du document   ouvrir (ex : AVOEBpPXOvpGhczzE_RAeY)

(4) : Token (sans les doubles quotes) URLEncod  (en utf8)

b. Acc der au Lauchpad (ancien InfoView) sans se connecter

De la m me mani re, il faut appeler l'url suivante :

[http://\(1\):\(2\)/BOE/BI/logon/start.do?ivsLogonToken=\(3\)](http://(1):(2)/BOE/BI/logon/start.do?ivsLogonToken=(3))

(1) : Nom ou adresse IP du serveur BO (ex : localhost)

(2) : Port du serveur de jsp utilis  par BO (ex : 8080)

(3) : Token (sans les doubles quotes) URLEncod  (en utf8)

c. Acc der au CMC sans se connecter

Il est impossible d'acc der directement au CMC via un token ou de bypasser le formulaire de login de quelque fa on que ce soit sans intervenir lourdement sur le contenu des fichiers de configuration sur le syst me.

C. Cahier des charges

L'Annexe C est un extrait du cahier des charges. Ce document a été validé par Pascal BALIGOUT, mon responsable.

Sont inclus :

- Le sommaire, aperçu rapide du contenu
- Les exigences fonctionnelles et un extrait des modifications apportées au MCD, le point de vue technique du Cahier des Charges
- Un chapitre sur une modélisation particulière : les éditions statistiques
- Une annexe concernant des potentielles évolutions futures



Cahier des charges

Intégration de SAP Bo
dans Millésime CCAS version web

Millésime CCAS version web



Sommaire

Sommaire.....	2
1 Versions	3
2 Introduction.....	3
a. Produit concerné	3
b. Objet.....	3
c. Cadre.....	3
3 Exigences fonctionnelles et techniques	3
4 Modèles de données	4
a. MCD	4
b. MLD.....	4
5 Paramétriser le serveur BO dans le Manager	6
a. Contraintes	6
b. Interface gestion des domaines	6
6 Lier un compte utilisateur du Manager avec un compte BO	7
a. Contraintes	7
b. Interface modification utilisateur.....	7
7 Accueil : accès aux statistiques.....	9
a. Contraintes	9
b. Interface Accueil.....	11
c. Interface d'ouverture d'un rapport d'activité	12
8 Accueil : Accès au Launchpad (anciennement portail Infoview).....	13
a. Contraintes	13
b. Mise en œuvre.....	13
9 Editions statistiques.....	14
a. Contraintes	14
b. Interface page statistiques, onglet d'un module.....	15
c. Interface page statistique, onglet documents perso	17
10 Ouverture d'un état paramétré.....	18
a. Contraintes	18
b. Interface aide facultative sollicitée	18
c. Interface aide facultative sollicitée – GFI	19
11 Invites obligatoires	20
Annexes	21
Annexe I : Evolutions futures	21
Annexe II : Arborescence des documents Business Object à respecter	23

1 Versions

Date	Version	Evolutions	Auteur
17/03/2014	0.1	Création	Bastien Jorge
20/03/2014	0.2	Maquettage	Bastien Jorge
02/04/2014	0.3	Retour P. Baligout	Bastien Jorge
12/05/2014	0.4	Retour E. Vattat	Bastien Jorge

2 Introduction

a. Produit concerné

Ce cahier des charges concerne le produit CCAS version web de la gamme Millésime.

b. Objet

Ce document décrit les évolutions à apporter afin de proposer une intégration progressive de l'outil décisionnel Business Object de la société SAP dans l'applicatif métier Millésime CCAS en version web.

c. Cadre

Il est nécessaire de pouvoir proposer un accès aux fonctionnalités décisionnelles et statistiques de l'outil BO afin de permettre aux agents des CCAS de produire l'ensemble des documents statistiques nécessaires à leur travail.

3 Exigences fonctionnelles et techniques

Le but final de l'intégration est de proposer les fonctionnalités de l'outil BO aux utilisateurs CCAS en leur facilitant les accès et l'utilisation à l'outil. Dans le meilleur des mondes, l'expérience utilisateur devrait laisser penser que l'on reste dans un seul et unique produit. Dans les faits, nous ne pourrons pas aller aussi loin (notamment à cause du pilotage des droits et groupes utilisateurs dans le CMC de BO). Malgré tout, une utilisation quotidienne après paramétrage sera quasi-transparente pour l'utilisateur final.

L'intégration sera réalisée à partir de la dernière version de Business Object (BO XI 4.1) car elle apporte la nouveauté de pouvoir solliciter le serveur avec des requêtes à des webservices de type REST.

Il faudra veiller à ce que cette intégration soit fonctionnelle dans les deux cas de déploiement suivants :

- Installation des produits chez le client (éventuellement sur des serveurs séparés)
- Installation hébergée des 2 produits (CCAS et BO)

L'étude va donc proposer une mise en place progressive de l'intégration de l'outil selon les axes suivants :

- Accéder directement aux éditions statistiques en fonction du contexte depuis CCAS.
- Permettre l'accès et la connexion automatique de l'utilisateur CCAS à son compte BO Launchpad.
- Customiser l'environnement BO pour donner une unité au couple CCAS - BO.
- Visualiser depuis CCAS les éditions planifiées nouvellement générées.

4 Modèles de données

a. MCD

OL_DOM_PARAM

ID_DOM_PARAM

...

DP_BO_DOMAINE

DP_BO_PORT_LAUNCHPAD

DP_BO_PORT_SERVEUR_REST

DP_BO_DOSSIER_RACINE

DP_BO_DOSSIER_UTILISATEUR

DP_BO_VALIDATE_TOKEN

OL_TYPE_COMPTE

ID_TYPE_COMPTE

...

TC_BTN_TEST

b. MLD

On ajoute dans la table OL_DOM_PARAM (qui contient les paramètres de connexion au domaine) les colonnes suivantes :

Colonne	Description	Type	Obligatoire
DP_BO_DOMAINE	URL du serveur BO	chaine	X
DP_BO_PORT_LAUNCHPAD	Numéro du port pour l'accès à BO	numérique	X
DP_BO_PORT_SERVEUR_REST	Numéro du port à solliciter pour les requêtes REST	numérique	X
DP_BO_DOSSIER_RACINE	Nom du dossier racine	chaine	X
DP_BO_DOSSIER_UTILISATEUR	Nom du dossier utilisateur	chaine	X
DP_BO_VALIDATE_TOKEN	Durée de validité du token en minutes	entier	

Par défaut :

Colonne	Valeur
DP_BO_DOMAINE	http://webi4.solutions-implicit.fr/
DP_BO_PORT_LAUNCHPAD	8080
DP_BO_PORT_SERVEUR_REST	6405
DP_BO_DOSSIER_RACINE	Dossier racine
DP_BO_DOSSIER_UTILISATEUR	Dossiers utilisateur
DP_BO_VALIDATE_TOKEN	

Le délai de validité du token est récupérable dans la forme du token lui-même, mais n'étant pas documentée il est possible que dans une version ultérieure, SAP décide de changer ce mode de communication. Ainsi, il est nécessaire de stocker en DB cette durée (non accessible à l'utilisateur final, il faudra la modifier directement dans la base si besoin). Enfin, si rien n'est spécifié, la durée utilisée sera celle par défaut dans BO : 60mn.

9 Editions statistiques

a. Contraintes

L'accès aux éditions statistiques est l'évolution la plus importante de ce cahier des charges. L'utilisateur devra accéder à une page centralisant tous les reports.

Il a été demandé s'il était possible d'enregistrer un report dans les raccourcis de la fenêtre d'accueil, mais cela nécessiterait que l'édition soit directement accessible depuis un point de menu, et il a été clairement spécifié que ce ne soit pas le cas.

La page centralisant toutes les éditions disposera d'un onglet pour chaque module (Foyer, AF, AL...) qui ne sera affiché que si l'utilisateur dispose des droits sur l'unité fonctionnelle correspondante, plus un onglet final affichant l'arborescence complète de tous les reports personnels ainsi que ceux partagés n'entrant pas dans les modules.

Pour l'affichage dans chaque module, il a été demandé de privilégier un affichage sous forme de liste plutôt que sous forme d'arborescence : les utilisateurs ayant déjà régulièrement fait remonter des réticences quant à leur utilisation. Mais cela pose le problème de la gestion des sous-dossiers : en effet, nous ne maîtrisons pas l'utilisation que feront les utilisateurs finaux, et s'ils décident de stocker plus d'un niveau de fichiers, l'affichage par liste est rendu impossible.

La solution pourrait être de proposer un arbre déjà déroulé recouvrant la totalité de l'espace disponible : en effet, les reproches qu'ont pu faire les utilisateurs concernent plus la taille disponible pour visualiser l'arbre que l'arbre lui-même.

Il faudra par contre imposer une certaine rigueur⁹ de nommage aux administrateurs pour les dossiers BO de chaque module afin de gérer l'affichage ou non des onglets dans la page.

La page des reports personnels (contenant à la fois les documents non partagés et les documents ne se situant pas dans le dossier racine *Implicit*¹⁰) quant à elle ne pourra se présenter que sous la forme d'un arbre. Par contre, la problématique de proposer à l'utilisateur de dérouler intégralement l'arbre se pose : en effet, dans un souci de réactivité, le contenu d'un sous-dossier n'est récupéré sur le serveur BO que lorsque l'utilisateur souhaite y accéder : si tout était calculé, un délai de l'ordre de 5 à 10 secondes serait nécessaire avant de déplier l'arbre.

Cette problématique ne se pose pas pour les modules car ils auront suffisamment peu de sous-dossiers (voire pas du tout) que ce délai de réponse serait rendu négligeable.

Néanmoins, il faut noter que BO propose de stocker dans sa GEIDE des liens hypertextes. Mais à la différence d'un autre type de fichier, il n'est pas possible de l'exploiter côté CCAS Web. En effet, il n'existe aucun moyen de récupérer cet URL ou d'ouvrir une page quelconque redirigeant directement vers la fenêtre concernée. Ainsi, ils seront tout simplement ignorés.

b. Interface page statistiques, onglet d'un module

- 1 Chacun de ces onglets ne s'affiche que si l'utilisateur dispose des habilitations nécessaires
- 2 Un clic sur l'un des documents ouvre une fenêtre GFI demandant de choisir des paramètres de filtres le cas échéant ainsi que le format du téléchargement
- 3 Icônes pour enrayer/dérouler l'intégralité de l'arborescence

Dans chacun des onglets, l'arborescence peut déjà être déroulée.

⁹ Rigueur néanmoins relative, qui reste souple sur l'égalité des noms (cf. Annexe II)

¹⁰ cf. Annexe II



Aperçu de la fenêtre GFI s'ouvrant au clic sur un document.

Si aucun filtre n'est présent sur l'état, la case paramètres ne s'affiche pas.

c. Interface page statistique, onglet documents perso

Ici, il serait trop long de dérouler l'arbre complètement (comme vu dans les contraintes), seul le premier niveau sera être ouvert par défaut.

Cependant, on laisse quand même la possibilité à l'utilisateur de dérouler intégralement l'arbre s'il le souhaite, nonobstant un délai de réponse qui peut être de l'ordre de quelques dizaines de secondes en cas de dossier personnel lourdement rempli.

Cet onglet comporte les documents présents dans son dossier personnel.

Attention, dans le cas des dossiers personnels, sont exclus les dossiers déjà présents dans le cadre des autres onglets et menus, à la fois pour des raisons de performance et pour le côté pratique des habilitations.

De plus, le nombre maximal de fichiers qui peuvent être lus par l'application est enregistré en dur, car nativement limités à 50 par BO. A l'heure actuelle, il est égal à 5000 : l'arbre WebDev sera saturé bien avant, mais cette limite agit également sur le nombre d'utilisateurs visibles par CCAS Web. Il faudra relever cette limite si le nombre d'utilisateur de la collectivité devient croissant et qu'ils ne sont pas régulièrement nettoyés.



Annexes

Annexe I : Evolutions futures

Il est envisagé de permettre une visualisation directe des instances de documents générées depuis la dernière consultation par l'utilisateur. Cette fonctionnalité apporte l'intérêt de pouvoir planifier un document long à générer pendant la nuit et de notifier l'utilisateur des nouveaux documents disponibles via une consultation directe depuis l'application.

Cette évolution est réalisable pour peu que l'administrateur sélectionne, lors de l'enregistrement de la planification, les utilisateurs à notifier via la messagerie interne BO lors de la mise à jour.

The screenshot shows the SAP BusinessObjects Planning interface. The main title is "Planifier - AF - Aides accordées par année". On the left, there's a sidebar with options like "Planifier", "Titre de l'instance", "Périodicité", "Formats", "Mise en cache", "Événements", and "Groupe de serveurs de planification". The main area has two tables: "Destinataires disponibles" and "Destinaires sélectionnés". Under "Destinataires disponibles", there are lists for "Liste des utilisateurs", "Liste des groupes", and "Héritage de groupe". Under "Destinaires sélectionnés", there are entries for "Administrator", "Guest", "pascal_balgout", "QaaVSServlePrincipal", "SMAdmin", and "stephane.bourard". Below these tables, there are sections for "Nom cible" (with radio buttons for "Utiliser le nom généré automatiquement" and "Utiliser le nom spécifique") and "Envoyer comme" (with radio buttons for "Raccourci" and "Email"). At the bottom right are "Planifier" and "Annuler" buttons.

Exemple d'ajout de destinataires BO pour la planification

Une fois l'état généré, un message est automatiquement envoyé dans la boîte de réception BO des utilisateurs (ou groupes) sélectionnés. On sait qu'il est possible d'accéder aux documents contenus dans cette boîte de réception par requêtes REST, ainsi que la date à laquelle ils ont été générés. Il suffit de stocker la date de dernière visualisation de ces documents pour pouvoir insérer en page d'accueil une pastille de notification avec le nombre de mises à jour, qui renverrait sur une nouvelle page contenant l'historique des documents planifiés.

Il est également possible de planifier un document envoyé en alerte lorsqu'un indicateur est atteint, qui sera affiché de la même manière.



The screenshot shows a web browser window with the URL "http://online.implicit.fr". The page is titled "Historique". It features a header with a home icon, a search bar, and a navigation menu. The main content area has two sections: "Historique des documents générés" and "Anciens documents générés". Under "Historique des documents générés", it says "2 nouveaux documents" and lists "14/03/2014 - Aides distribuées au deuxième semestre 2013 (corrige)" and "13/03/2014 - Aides distribuées au deuxième semestre 2013". Under "Anciens documents générés", it lists several documents with their dates: "13/09/2013 - Aides distribuées au premier semestre 2013", "12/03/2013 - Aides distribuées au deuxième semestre 2012", "13/09/2012 - Aides distribuées au premier semestre 2012", "13/03/2012 - Aides distribuées au deuxième semestre 2011", "14/09/2011 - Aides distribuées au premier semestre 2011", and "12/03/2011 - Aides distribuées au deuxième semestre 2010".

Maquette de page d'historique des documents planifiés

Lors d'un clic sur l'un des documents, une fenêtre GFI s'affiche demandant le format d'ouverture : pdf, xls, xlsx ou webi.

D. Décret no 95-562 du 6 mai 1995 (articles 1^{er} et 2^e)

Décret n° 95-562 du 6 mai 1995 relatif aux centres communaux et intercommunaux d'action sociale ainsi qu'aux sections de centre communal d'action sociale des communes associées et portant dispositions particulières applicables aux centres communaux d'action sociale de Marseille et de Lyon :

CHAPITRE I^{er} – Dispositions générales

Art. 1^{er}. – Les centres communaux et intercommunaux d'action sociale mentionnés au chapitre II du titre III du code de la famille et de l'aide sociale procèdent annuellement à une analyse des besoins sociaux de l'ensemble de la population qui relève d'eux, et notamment de ceux des familles, des jeunes, des personnes âgées, des personnes handicapées et des personnes en difficulté. Cette analyse fait l'objet d'un rapport présenté au conseil d'administration.

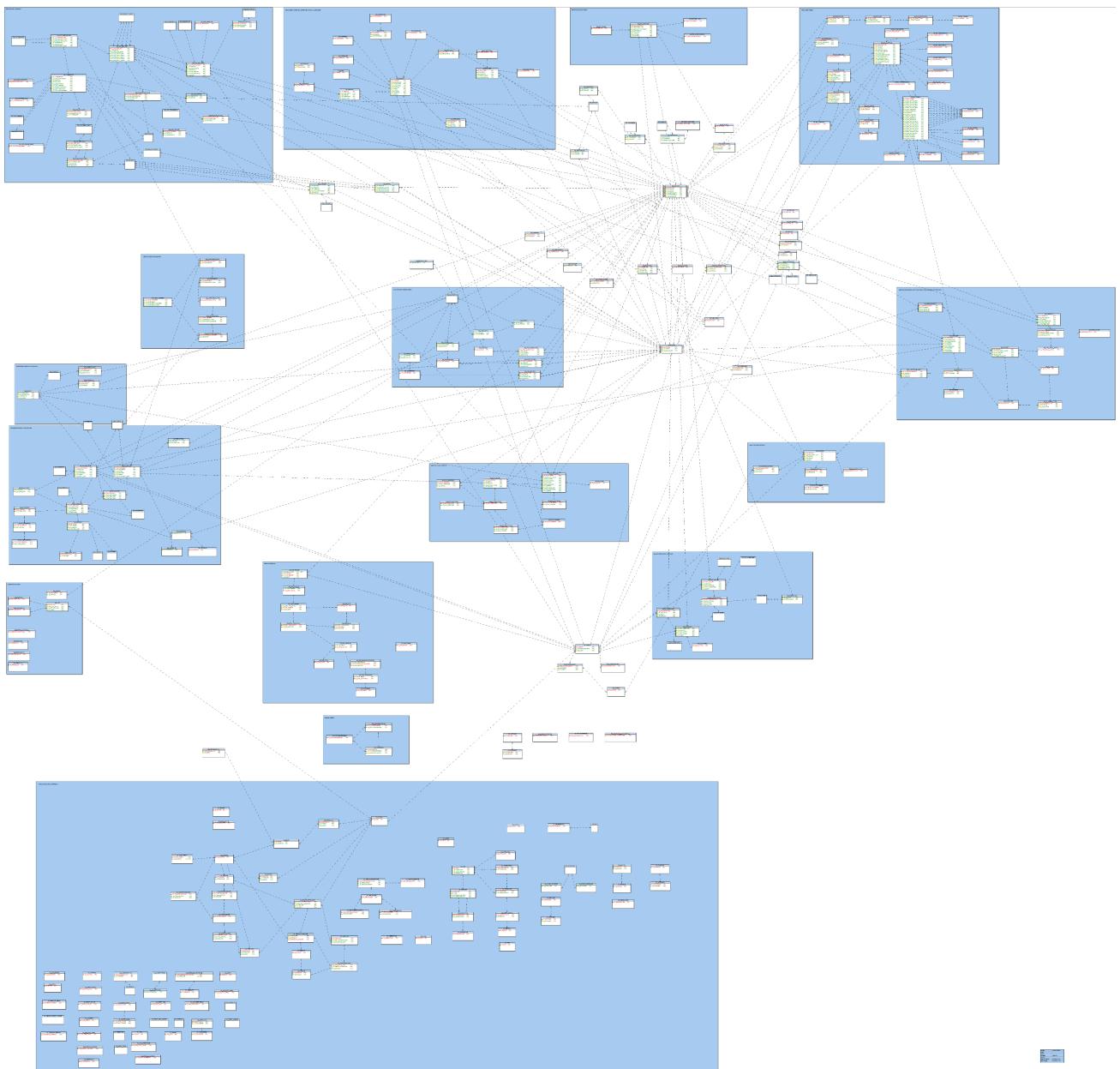
Art. 2. – Les centres d'action sociale mettent en oeuvre, sur la base du rapport mentionné à l'article 1^{er}, une action sociale générale, telle qu'elle est définie par l'article 137 du code de la famille et de l'aide sociale et des actions spécifiques. Ils peuvent intervenir au moyen de prestations en espèces, remboursables ou non, et de prestations en nature.

Texte complet disponible sur Legifrance :
<http://www.legifrance.gouv.fr/affichTexte.do?cidTexte=JORFTEXT000000187191>

E. Vision globale du MCD

L'Annexe E est une représentation graphique du Modèle Conceptuel de Données.
Sa présence sert ici à illustrer la complexité de la base de données utilisée.

On décompte 415 tables différentes.



F. Manuel utilisateur

L'Annexe F est un extrait du manuel utilisateur de la version intégrée à CCAS Web de Business Objects. Ce livret sera retravaillé par les formateurs pour mieux correspondre aux standards attendus par les utilisateurs.

Cet extrait comporte le chapitre traitant du lecteur de documents, l'utilisateur qui ne dispose que des droits de consultation.

Dans sa version complète, on y trouve deux autres profils : celui de l'administrateur CCAS Web et celui du créateur de documents.

4 Profil : lecteur de documents webi

4.1 Accès aux éditions statistiques

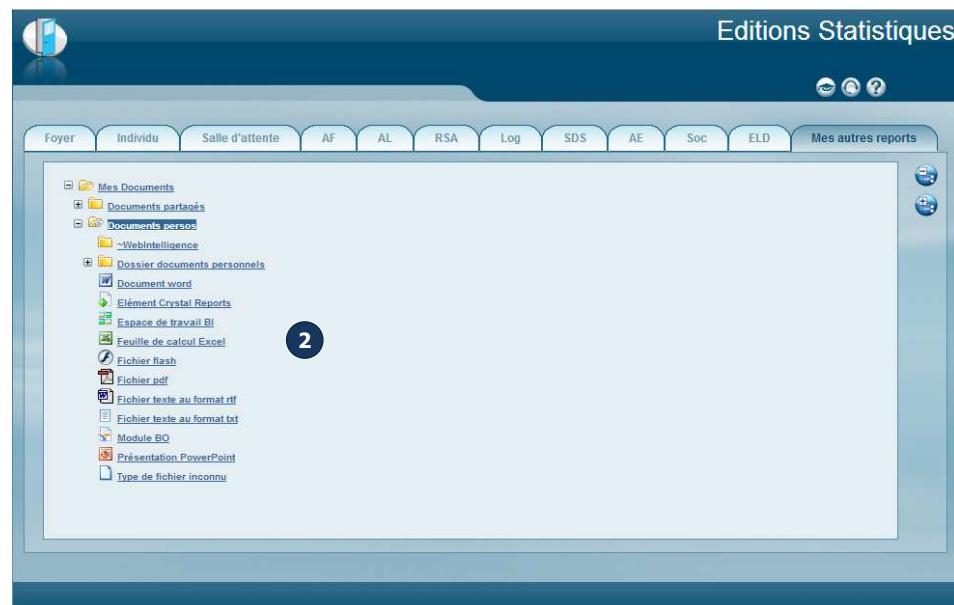
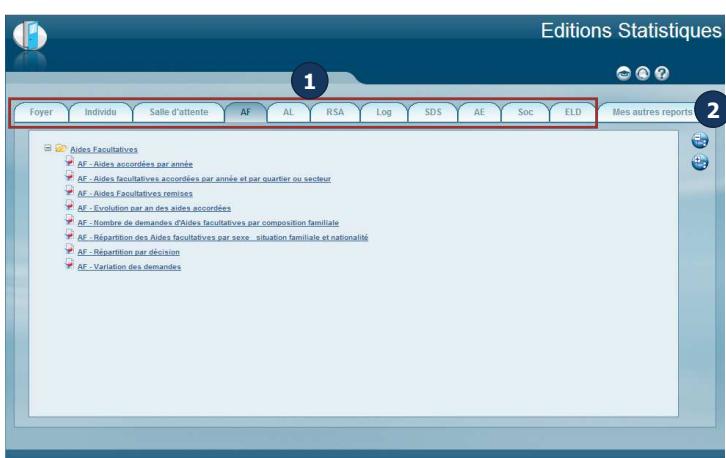
Il est possible d'accéder à toutes les éditions statistiques mises à disposition, ainsi qu'à ses documents personnels.

L'accès se fait à partir du point de menu [Statistiques], puis [Editions].



La fenêtre d'éditions statistique dispose de deux types d'onglets :

- Les onglets propres à chaque module, présents ou non selon les habilitations correspondant aux unités fonctionnelles associées **1**
- Un onglet « Mes autres reports » qui regroupe tous les documents partagés qui ne sont pas dans le dossier « Implicit » (voir Annexe 1) et tous vos documents personnels **2**



Zoom

Au clic sur un fichier, plusieurs types d'action peuvent être effectués en fonction de son type.

- S'il s'agit d'un **document pdf**, il sera ouvert en lecture directement via BO.
- S'il s'agit d'un **autre type de document non webi**, il sera proposé au téléchargement.
- S'il s'agit d'un **webi**, deux cas :
 - Aucun filtre n'est présent et on peut l'ouvrir directement **1**
 - Il présente des filtres, et il faut les remplir avant d'ouvrir le webi **2**





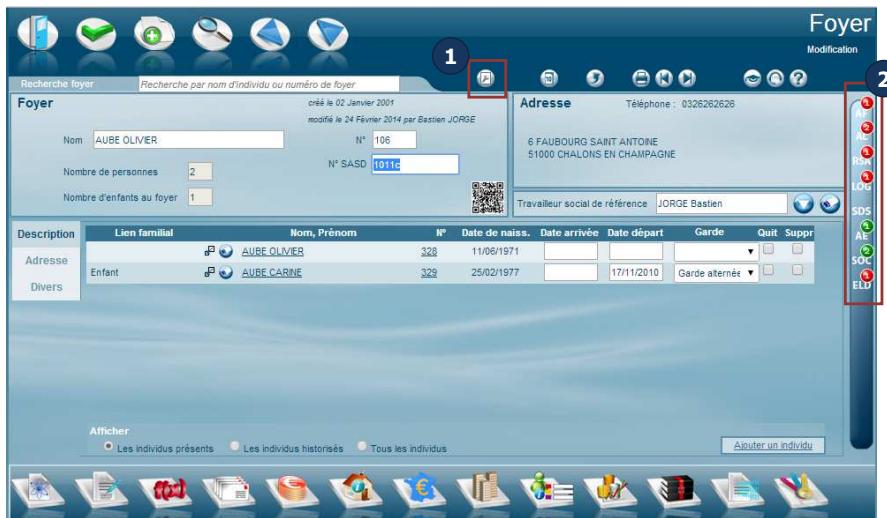
4.2 Accès aux états paramétrés

Les états paramétrés – au sens de l’application Millésime qui se basait sur Crystal Report – sont des reports qui s’ouvrent en fonction du contexte.

Par exemple

Je me trouve sur le dossier du foyer Dupont, en aide facultative, sur une demande précise. Au clic sur le bouton je vais accéder à une fenêtre me permettant d’ouvrir un webi pré-filtré sur le foyer, et le numéro de demande.

Les états paramétrés sont disponibles sur les fenêtres de foyer **1**, d’individu, de salle d’attente, de commission, et sur chacune des fenêtres de modules **2**



4.3 Les rapports d’activité

L’accès aux rapports d’activité se fait depuis le point de menu [Rapports d’activité].



Information

Les rapports d’activité fonctionnent de la même façon que les états paramétrés.

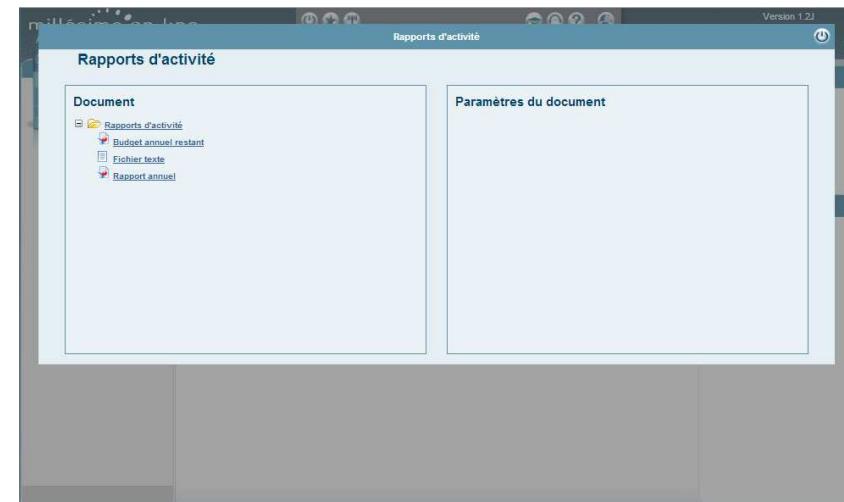


Table des figures

1.	Logo de la société JVS-IMPLICIT, et logo actuel d'Implicit	6
2.	Répartition géographique des salariés	7
3.	Organigramme de la société Implicit (58 collaborateurs)	8
4.	Logo du Groupe Chèque Déjeuner	9
5.	Extrait de la brochure <i>Les solutions action sociale et médico-sociale du groupe Chèque Déjeuner, ça fait du bien au quotidien</i>	10
6.	Cohésia, la solution d'« informatique sociale » du groupe	11
7.	Page d'accueil de l'application CCAS Web	14
8.	Requêteur – Choix des filtres	16
9.	Requêteur – Résultat de l'exécution	16
10.	Exemple d'un report BO	17
11.	L'interface de création de requête BO : la sélection de filtres et d'intitulés se fait par cliqué-glissé	18
12.	Charge de travail pour l'intégration de BO	20
13.	Envoi d'une requête avec <i>Advanced REST Client</i>	22
14.	Retour d'une requête sous <i>Advanced REST Client</i>	22
15.	<i>Advanced Rest Client</i> dans le lanceur d'applications Chrome	23
16.	Exemple de définition d'une méthode en WLangage	23
17.	Environnement de travail WebDev	24
18.	Maquette d'une nouvelle fenêtre dans l'application	26
19.	Extrait de la classe JavaScript de gestion de BO	27
20.	Code batch utilisé dans le projet	28
21.	Demande d'accès à un report avec filtres sur l'année et le type d'aide	30
22.	Le Manager : accès aux paramétrages	31
23.	Zone générée dynamiquement par l'ajout d'une occurrence en base de données	32
24.	Benchmark d'une requête de récupération du token	34
25.	Durée par exécution	34
26.	Classe gBo dans WebDev, un des éléments transmis à Steeve	35
27.	Aperçu : documents stockés dans BO visibles sous CCAS Web	36
28.	Logo de l'entreprise Décivision	38
29.	Logos de la société Business Geografic, et de leur produit GeoBI	38
32.	Page d'accueil du Designer GeoBI	39
30.	Exemple de statistique fournie par Business Geografic	40
31.	Cartographie issue des statistiques de la <i>Figure 30</i>	40

33.	Aperçu du Windows Server 2008 sur lequel est installé GeoBI	41
34.	À gauche (1) : fond de carte du centre de Troyes - À droite (2) : découpe arbitraire du centre de Troyes	42
35.	Logo du site GéoPortail	42
36.	Répartition des bénéficiaires d'un CCAS sous forme de zone de chaleur .	43
37.	Logo du logiciel ArcGIS	44
38.	Logo du logiciel JOSM	44
39.	Affectation d'un attribut d'une couche dans JOSM	45
40.	Isochrone des zones accessibles en 15 et 30 minutes depuis le centre-ville de Montpellier	46

Bibliographie

- [1] Sébastien DE LA MARCK et Johann PARDANAUD. *Dynamisez vos sites web avec Javascript*. Livre du Zéro, Aurillac : Albedia, 2012, 512p. ISBN 979-10-90085-04-6.
- [2] JEUN'ESS. C'est quoi l'ess ?
<http://www.jeun-ess.fr/cest-quoi-l-ess/>, Mis à jour le 27 mai 2013 [Consulté le 6 mai 2014]. [En ligne].
- [3] PCSOFT. *WebDev 15.0 – Auto-Formation*. Courtry : PCSoft, 2009, 458p.
- [4] UNION NATIONALE DES CENTRES COMMUNAUX D'ACTION SOCIALE. Présentation des centres communaux et intercommunaux d'action sociale (ccas/cias).
<http://www.unccas.org/unccas/ccas-cias.asp>, [Consulté le 29 avril 2014]. [En ligne].
- [5] Michel VAN KOTEN. Global software top 100 - edition 2011.
<http://www.softwaretop100.org/global-software-top-100-edition-2011>, Mis à jour le 23 août 2011 [Consulté le 2 mai 2014]. [En ligne].
- [6] Chèque Déjeuner. Des salariés qui vont bien, c'est une entreprise qui va bien !
<http://www.cheque-dejeuner.com/une-entreprise-pas-comme-les-autres/valeurs-sociales-humaines-SCOP-chequedejeuner.jsp>, [Consulté le 10 juillet 2014]. [En ligne].
- [7] Business Geografic. Qu'est-ce que geobi ?
<http://www.geo-bi.com/fr/geobi-pour-businessobjects/quest-ce-que-geobi.html>, Mis à jour en 2013 [Consulté le 17 juin 2014]. [En ligne].

À Propos

Ce document a été réalisé en L^AT_EX sous Windows avec TeXStudio par Bastien JORGE et a été compilé grâce à MiK^TE_X.
Le style s'appuie sur le package open-source **koma-script**

