

Proposition de PFE : Module de conciliation bancaire OFX

Résumé :

Ce projet consiste à écrire et intégrer un module de conciliation bancaire pour le logiciel financier GnuCash. Le logiciel GnuCash est un logiciel de finances personnelles similaire à Quicken, mais imposant les pratiques comptables de la double entrée. Il est distribué sous licence GPL¹.

Le projet :

L'écriture d'un module de communication OFX² général avait été tentée par les développeurs de GnuCash. Celui-ci était prévu pour supporter la norme OFX au complet (conciliation bancaire, paiement de factures, transferts bancaires, transactions boursières, etc.). Ce projet a échoué. Les développeurs estimaient à 12 mois-personnes le travail nécessaire. Le projet ne comptant que sur des bénévoles, il n'avancait plus.

Je propose donc un projet moins ambitieux : n'implanter qu'une partie de la norme OFX : L'importation de ce qui est communément appelé un « fichier OFX » soit un fichier de conciliation bancaire suivant la norme OFX. Cela permet d'implanter la fonction la plus désirée de la norme OFX, soit de concilier un compte GnuCash avec des transactions téléchargées à partir d'un compte bancaire ou d'une carte de crédit. Je crois ce projet réalisable comme un PFE de 6 crédits.

Considérations techniques :

Mon module sera basé sur la norme OFX 1.6. C'est la version la plus récente des « anciennes » normes OFX, basées sur SGML³, et elle est compatible avec les versions précédentes. Il existe une nouvelle norme, OFX2.01, mais celle-ci est basée sur XML⁴, et est encore très peu déployée par les institutions financières. Utiliser OFX2.01 poserait probablement des problèmes de compatibilité arrière. Par contre, les deux normes sont des cousines rapprochées⁵ et il est possible de passer de l'un à l'autre. Comme il existe une multitude d'outils travailler avec XML, et moins pour SGML, il est envisageable qu'il soit plus simple de convertir le DTD au format XML, et qu'il soit ensuite simple de supporter les deux normes. Mais à cette étape préliminaire, je n'ai aucun moyen de le savoir sans une étude plus approfondie des deux normes et des différents outils. De plus, mon institution financière (le service Accès-D de Desjardins) exporte ses fichiers dans le format OFX 1.02, et il me faut impérativement un moyen de me générer des fichiers tests. Donc, pour que mon module soit utile à la majorité des utilisateurs et que je puisse le tester, je m'en tiendrai à priori à OFX1.6 dans le cadre de ce PFE.

Mon projet devra être intégré à un logiciel existant. Il faudra donc que je travaille avec les personnes présentement en charge, et que je me conforme à certaines de leurs normes et procédés. De plus le modèle de développement ouvert utilisé par les logiciels libres implique que je devrai demander des modifications ou éliminer des erreurs dans des modules maintenus par d'autres développeurs, et que d'autres feront sans doute de même pour les miens. J'ai déjà contacté la communauté de développeurs de GnuCash, et j'ai été bien reçu.

¹ GNU General Public License: <http://www.gnu.org/licenses/licenses.html#TOCGPL>

² Open Financial Exchange: <http://www.ofx.net/ofx/default.asp>

³ Standard Generalized Markup Language

⁴ Extensible Markup Language

⁵ Comparison of SGML and XML, World Wide Web Consortium Note 15/12/1997, <http://www.w3.org/TR/NOTE-sgml-xml>

Implantation et temps prévu :

Familiarisation avec le code source du logiciel et la norme OFX. À cette étape, je prévois également discuter des modifications au logiciel actuel afin de faciliter l'intégration aux dernières étapes. Je prévois 20 h pour cette étape préliminaire.

Ajout du numéro de transit aux structures de données des comptes de GnuCash. L'importation de fichiers OFX dans le populaire logiciel Quicken a des faiblesses qui laissent une ambiguïté quant au compte que l'on désire concilier si l'on met à jour le logiciel ou si notre service bancaire en ligne fait certaines modifications. Je ne veux pas répéter ce problème. Le numéro de transit étant unique, il n'y aura plus d'ambiguïté quant au compte à concilier. Cependant, les comptes de cartes de crédit n'ont pas de numéro de transit, un autre identificateur unique devra être utilisé. Je devrai aussi ajouter l'interface usager et l'interface programmeur pour accéder et modifier ces données. Le changement devrait être relativement trivial. Temps prévu : 7h.

Parseur du fichier OFX. C'est bien sûr le travail le plus complexe et le plus important de ce projet. Plusieurs approches sont possibles. Il serait possible de simplement parcourir linéairement le fichier à la recherche des jetons définissant un compte bancaire, puis d'interpréter directement les jetons qui suivent. Ce serait probablement une approche assez rapide, mais extrêmement risquée, car elle rendrait quasi-impossible la détection d'erreur. Le plus le nombre d'exceptions à gérer serait très élevé, et la maintenance très difficile. Il vaudrait probablement mieux utiliser une librairie SGML ou XML existante, l'utiliser pour parser le DTD et accéder ensuite aux éléments du fichier en parcourant l'arbre généré. Cette approche offre la validation, et donnera également une excellente base pour que d'autres développeurs puissent éventuellement étendre qui voudraient étendre mon module pour ajouter d'autres fonctions que la conciliation bancaire (paiement en ligne, etc.). Un bon nombre d'heures d'analyse restent à passer pour choisir la meilleure implantation de cette fonction. Ce module transformera les données contenues dans le fichier OFX en un compte temporaire de GnuCash, et affichera les autres données d'intérêt (Version, date, banque, transit, etc.). 125h prévues.

Fonction de fusion. La fusion des données importées devrait être séparée de l'importation elle-même pour des raisons de robustesse et de lisibilité du code. Ainsi on ne fera pas des opérations de copie et de comparaison sur des structures de données différentes, et on pourra utiliser les fonctions de copie et de recherche qui font déjà partie du logiciel. De plus, ce module a d'autres utilisations potentielles, et pourra être réutilisée dans d'autres parties du logiciel.

La fusion consiste à prendre le compte temporaire et le fusionner avec le compte correspondant (identifié par son no. de transit) de GnuCash. Si le no. de transit du compte en question n'existe pas dans les autres comptes, le logiciel offre à l'utilisateur de choisir le compte à utiliser, ou en créer un nouveau. Il ajoute le numéro de transit au compte choisi. Si deux comptes ont le même numéro de transit, un message d'erreur est affiché.

Un fois le compte identifié, pour chaque transaction, le module :

- cherche une transaction ayant le même identificateur, la même date et le même montant. Si il en trouve une et une seule, il considère l'opération comme conciliée.
- si il n'en a pas trouvé, il cherche une transaction ayant le même identificateur et le même montant dans les 4 jours précédent l'opération conciliée, ou une opération ayant le même montant et la même date. Si il en trouve une, il offre de faire correspondre.
- Si il n'en a pas trouvé ou si l'utilisateur a refusé la correspondance, il offre d'ajouter l'opération au compte.

La gestion des duplicata est importante. C'est là une autre faiblesse du logiciel Quicken que je ne voudrais pas reproduire. Si il trouve plus d'une opération correspondante à une opération téléchargée, le module informera l'utilisateur d'une erreur potentielle. Par défaut Quicken fait correspondre toute opération téléchargée avec la récente opération ayant le même identificateur et le même montant, même si elle est plus vieille de plusieurs mois! Limiter à un certain nombre de jours comme je l'ai fait plus haut règle une partie du problème, mais j'aimerais aussi régler le problème posé par deux retraits d'un même montant dans un guichet automatique faits la même journée. Mais je ne suis pas sûr qu'il y ait une solution plus élégante dans ce cas que d'offrir à l'utilisateur d'ajouter les deux opérations. De plus, il ne faut pas rejeter complètement l'approche de Quicken, car pour les cartes de crédit, il peut arriver de recevoir la transaction plusieurs semaines après un achat. (Récemment j'ai payé un repas dans un aéroport d'Europe et j'ai reçu la transaction plus d'un mois après)

Une partie des fonctions nécessaires à la fusion étant déjà offerte par le GnuCash, cette partie sera probablement intégrée à un module existant. C'est à ce niveau que se situe la majorité de l'intégration au reste du logiciel, c'est l'une des principales difficultés pressenties. 75h prévues.

Documentation : Une partie de tout développement structuré est d'écrire de la documentation à l'intention des autres développeurs ainsi que des utilisateurs. 10h prévues.

Tests et déverminage : L'échelle et le temps alloué à ce projet ne me permet pas de prévoir autre chose que des tests unitaires et quelques tests fonctionnels. Je vais devoir écrire des tests pour exercer les fonctions du module de fusion (particulièrement pour la vérification des duplicata). Les tests fonctionnels seront très limités, puisque je n'ai qu'une source de fichiers OFX pour l'instant (il est cependant possible que d'autres développeurs acceptent de me fournir des fichiers OFX). Je ne prévois pas générer une campagne de tests automatisés. 25h prévues

Rapport : Une partie de la documentation produite et de la correspondance échangée pourra être récupérée pour rédiger le rapport, j'y alloue donc peu de temps : 20h prévues.

Total : 282 heures

Liens et références :

GnuCash

Le logiciel GnuCash <http://www.gnucash.org/>

OFX

Le consortium OFX, <http://www.ofx.net/ofx/default.asp>

XML et SGML

-Comparison of SGML and XML, World Wide Web Consortium Note, 15/12/1997, <http://www.w3.org/TR/NOTE-sgml-xml>

Extensible Markup Language (XML), <http://www.w3.org/XML/> TRÈS bien....

-XML from the inside Out, <http://www.xml.com/>

-XML White Papers http://www.xml.org/xml/resources_whitepapers.shtml

-Converting an SGML DTD to XML, Norman Walsh, July 08, 1998, <http://www.xml.com/pub/a/98/07/dtd/>

A Gentle Introduction to SGML, C. M. Sperberg-McQueen and Lou Burnard, <http://www-tei.uic.edu/orgs/tei/sgml/teip3sg/index.html>

-XML - Questions & Answers, Jon Bosak, Sun Microsystems, <http://www.isgmlug.org/n3-1/n3-1-18.htm>

-Differences between SGML, XML and HTML (avec graphiques) <http://matwww.ee.tut.fi/~onykane/courses/prj/xmlsem2000/ht/leppanen.htm>

Outils

Free XML tools and software, <http://www.garshol.priv.no/download/xmltools/>

Free SGML Tools, Christopher Browne, <http://www.ntlug.org/~cbbrowne/sgmlfreestuff.html>

-Nsgmls : outil de validation SGML

-perlSGML : Outil de parsing et manipulation des fichiers SGML et de leurs fichiers de définition (DTD : Document Type Definition)