Développement de Wims

Soumis par Bernadette Perrin-Riou
24-12-2007
Dernière mise à jour : 11-01-2008

Pendant le colloque wims à Nice en mai dernier, Gang Xiao nous a dit qu'il ne désirait pas continuer à s'occuper de la maintenance et des patchs des versions intermédiaires de WIMS. Nous avons donc avec son accord décidé de prendre le relais.

Pour cela, nous avons déposé les sources du noyau de WIMS (noyau en C et modules administratifs) au CRU (<http://sourcesup.cru.fr/> et plus précisément <http://sourcesup.cru.fr/projects/wimsdev>) qui propose en particulier un hébergement et une gestion collaborative de sources par Subversion et un environnement avec GForge. Après une période d'apprentissage et d'adaptation, le projet est devenu public récemment.

Nous mettons régulièrement les patchs de bogues décelés dans la version 3.63a et proposons quelques améliorations. Il s'agit pour l'instant de maintenance plutôt que d'innovations.

Actuellement, les modifications faites sont visibles sur le site de sourcesup et les sources sous svn peuvent être récupérées par svn par des développeurs. Il est donc possible de proposer des patchs par rapport à cette version. Notre but pour l'instant est de conduire tranquillement wims vers une version 3.64 qui soit acceptée par Gang Xiao.

Un package 3.63a.tgz sera bientôt mis à la disposition et une version de test sera mis à disposition. C'est d'ailleurs aussi le cas du serveur wims.auto.u-psud.fr (tant que cela ne pose pas de problème).

Les modules développés à l'aide de fonctionnalités introduites après la version 3.62 ne doivent pas être publiés dans le système de publications de Nice. Pour l'instant, aucune vérification n'est faite.

Afin de pouvoir aussi partager des modules en développement qui ne se trouvent pas dans la distribution (administratifs ou pédagogiques), nous avons aussi créé un projet MODULESWIMS. Les modules qui sont à l'intérieur ont vocation s'ils tournent bien à entrer ensuite dans la distribution "officielle".

Il nous semble aussi indispensable qu'une concertation soit menée lors d'ajouts d'interfaces à d'autres logiciels afin de mesurer les différents aspects négatifs ou positifs "plus du logiciel par rapport à ceux déjà interfacés/interdépendance supplémentaire, coût du point de vue de la sécurité et de la complexité d'utilisation". Bien sûr, les développeurs font ce qu'ils veulent sur leur version locale, à condition de vérifier que les modules publiés ensuite sont compatibles.

Cordialement

Fabrice Guérimand, Bernadette Perrin-Riou, Eric Reyssat