



Interview de Frank Wille - Juin 2006

Par Philippe "Elwood" Ferrucci,
Relu par Mathias "Corto" Parnaudeau.

Te rappelles-tu ce qui t'a décidé à acheter un Amiga ? Comment as-tu commencé à programmer ?

J'ai vu un Amiga pour la première fois en 1986, après avoir lu des infos sur ce nouvel ordinateur de rêve dans les magazines. Des amis en ont acheté un et je leur ai souvent rendu visite pour jouer et essayer leur A1000.

C'était le temps des Defender of the Crown, Winter Games, Faery Tale et Test Drive. J'étais complètement épaté et fasciné par cette machine. Je savais qu'il m'en fallait une absolument. Et au début de 1987, avec mon frère, j'ai acheté un Amiga 1000.

Vous ne serez pas surpris de savoir que mon frère ne pouvait pas beaucoup utiliser l'Amiga, vu que j'étais dessus toute la journée. ;) Du coup un an après il a eu son propre A500, alors que j'étendais l'Amiga 1000 avec 2 Mo de Fast RAM, un disque dur de 30 Mo, un processeur 68010, un flicker fixer, etc..

Cette fascination pour l'Amiga n'a jamais disparue, elle a même augmentée au fur et à mesure que j'en apprenais plus sur le matériel et son super système d'exploitation. Je n'ai jamais eu autant de fun en informatique que pendant ces 7 années jusqu'à la faillite de Commodore, et je n'en aurai probablement plus.

J'ai programmé dessus depuis le premier jour parce que c'est ce que j'avais toujours fait. J'écrivais des programmes en assembleur depuis 1983 sur les VIC-20 et C-64. Donc j'ai tout de suite commencé à apprendre le 68000 et j'ai acheté le compilateur Aztec-C. Mais pendant plusieurs années je restais toujours intéressé par la programmation en assembleur.

Mon premier programme sur l'Amiga a dû être un truc du genre test d'entrée/sortie dans une console ou juste un test pour le bouton gauche de la souris en lisant le registre hardware \$bfe001. Je ne me rappelle pas vraiment. :)

Qu'est-ce que VBCC ? D'où vient-il ? Comment l'as-tu rencontré ?

VBCC, qui s'écrit souvent en minuscule, vbcc, est le compilateur C de Volker Barthelmann. Un compilateur ISO-C portable et pouvant avoir plusieurs cibles. Il supporte ISO/IEC 9899:1989 et une partie du nouveau standard ISO/IEC 9899:1999 (C99).

Son gros avantage est qu'il est très compact et facile à porter sur toute plate-forme qui possède un compilateur ANSI-C en quelques minutes. Il supporte aussi très facilement la cross-compilation pour les architectures 8, 16, 32 ou 64 bits.

Mais vbcc est plus qu'un simple compilateur. Il a évolué vers un environnement de développement complet, incluant un assembleur portable pour 6 processeurs différents (vasm), un linker portable pour 8 formats de fichiers différents (vlink), un profileur, un planificateur d'instruction, une bibliothèque d'outils, un afficheur de messages, etc..

C'était il y a à peu près 10 ans (1995 ou 1996) que j'ai reçu un email d'un inconnu appelé Volker Barthelmann. Il était intéressé par PhxAss, mon assembleur M68k. Quelques mails plus tard il m'a demandé si je voulais tester son compilateur C, qu'il venait d'écrire. Je ne pensais pas qu'une seule personne pouvait écrire un compilateur utilisable pendant son temps libre mais j'ai été curieux et j'y ai jeté un oeil.

Le compilateur n'était pas parfait car encore en développement mais il était au moins au même niveau que DICE-C par exemple. Les mois suivants j'ai commencé à utiliser vbcc pour mes projets car je n'avais pas SAS/C, mon Aztec-C était trop vieux et plus supporté et je n'ai jamais aimé GNU-C. Celui-ci était trop Unixien à mon goût et il se comportait comme un extra-terrestre sur mon AmigaOS.

Les années qui suivirent je me suis impliqué de plus en plus dans le projet vbcc avec mon assembleur 68k PhxAss et mon linker PhxLnk qui furent intégrés au projet. En 1997, j'ai commencé à écrire l'assembleur PowerPC portable "pasm" et le linker multi-cible "vlink" pour vbcc, qui était une base pour ajouter ensuite le support de PowerUp et WarpOS à vbcc.

À partir de ce moment-là nous formions une équipe de deux. Volker se concentrait sur le cœur du compilateur vbcc, alors que je travaillais sur les assembleurs, linkers, startup-codes et bibliothèques.

Volker a quitté l'Amiga il y a longtemps et il travaille sur Linux. Je n'ai jamais quitté mon « vieil amour » et je développe toujours sur Amiga. Nous avons un accord qui m'autorise à distribuer des versions de vbcc pour tous les systèmes d'exploitations Amiga-like autant que je veux et Volker a le droit de publier et vendre le compilateur avec mes outils dans les marchés non-Amiga (par exemple sur le marché des composants embarqués).

GCC est un compilateur reconnu mondialement et très populaire. Qu'en est-il de vbcc ? As-tu une idée de sa popularité ?

Je n'ai pas de chiffres précis. Il arrive souvent pour un logiciel freeware que les retours qu'on peut avoir viennent uniquement des personnes qui l'utilisent de façon intense. Ceux-ci trouvent un bug gênant et voudraient le voir corrigé. Nous avons un retour de ce genre et nous ne pouvons qu'estimer le nombre d'utilisateurs en appliquant un facteur au nombre de retours reçus.

Vbcc est utilisé mondialement dans des environnements très différents. Je suppose que la majeure partie est constituée d'utilisateurs Amiga mais nous avons aussi été contactés par des gens qui utilisent vbcc sur des environnements de composants embarqués (la plupart fonctionnent avec des générateurs de code pour C16x, HC12, PPC ou Coldfire). Certains aussi utilisent vbcc en tant que cross-compileur sous Windows. D'autres encore, écrivent leur propre générateur de code pour supporter leur propre processeur, ou encore utilisent vbcc comme moyen d'éducation. Il y a quelques temps, nous avons été contactés par un professeur au Vietnam qui utilise vbcc pour ses cours avec des processeurs C16x.

Comparez-vous vbcc avec GCC (ou d'autres compilateurs) notamment pour comparer le code généré ou la vitesse de compilation ? Lequel est le meilleur ? :-)

Bien sûr que nous faisons des comparaisons, de temps en temps. Et gcc est bien un compilateur avec lequel on peut comparer vbcc car il y a beaucoup de ressemblances entre eux. Mais nous avons la chance de faire mieux dans vbcc ce que nous n'aimons pas dans gcc.

Souvent une comparaison entre gcc et vbcc n'est pas juste. Il y a des hordes de programmeurs, souvent payés, qui travaillent sur gcc alors que nous ne sommes que deux. Donc dans de nombreux cas, gcc génère un code un peu meilleur. Ce serait une honte pour eux si ce n'était pas le cas.

Mais il y a toujours des cas où vbcc est meilleur. Surtout quand on compare avec le vieux gcc 2, le générateur de code PPC de vbcc produit souvent un code plus optimisé. Vous pouvez consulter des tests comparatifs à l'adresse <http://sun.hasenbraten.de/vbcc/index.php?view=benchmarks>

Le principal avantage de vbcc est la simplicité. Il est très facile de le porter sur une plate-forme et l'utiliser comme cross-compileur où que vous vouliez. L'interface est bien documentée et il est possible d'ajouter un nouveau générateur de code en un rien de temps. C'est un compilateur poids plume qui ne remplit pas votre disque dur de douzaines de méga-octets, répartis sur plusieurs répertoires. À titre d'exemple, la version complète pour OS4, contenant les exécutables, les bibliothèques et les fichiers entêtes .h fait environ 1,8 Mo seulement.

Tu crées encore des exécutables pour PowerUP/WarpOS. Prévois-tu de supporter ces architectures pendant longtemps ? Avec MOS et un jour OS4 sur Classic, on est amené à se demander pourquoi tu fais du travail en plus.

C'est vrai. Je ne les supporterai pas indéfiniment. Mais tant qu'il y a plus d'utilisateurs de BPPC et de CSPPC que de Pegasos et AOne, cela a un sens.

J'aimerais bien abandonner rapidement le support pour les cartes PowerUp, car il faut beaucoup de travail supplémentaire pour créer ces versions. Peut-être que je supporterai AROS à la place. Le premier compilateur vbcc expérimental pour AROS/x86 est déjà en test.

Travailles-tu sur d'autres projets que vbcc ? Dis-nous en plus.

Pas de projets importants mais de temps en temps, j'aime porter des logiciels sur Amiga. Surtout des jeux mais aussi des émulateurs (STonAmiga) ou des outils de développement (cvs). La plupart des Amigaïstes doivent connaître mes portages de Quake et QuakeWorld, qui atteignent une vitesse raisonnable grâce à de grosses optimisations en assembleur. C'est sans aucun doute le portage sur lequel j'ai passé le plus de temps.

Je m'intéresse aussi à NetBSD. Toutes mes machines qui ne tournent pas sous AmigaOS ou MorphOS tournent sous NetBSD. Mais à part aider au développement d'une nouvelle interface réseau, je n'ai pas bien eu le temps de faire du hacking de kernel.

J'ai commencé le portage de NetBSD sur Pegasos avec des programmeurs d'Espagne, USA et Danemark, mais malheureusement nous n'avons pas eu assez de temps et/ou d'expérience pour le finir.

Comme vbcc est utilisé pour certains composants d'OS4, cela nous amène une à la question habituelle : que penses-tu d'OS4 et qu'est-ce qui te manque le plus dedans ?

C'est une question dangereuse. Quelque soit ce que je répons, je vais avoir des problèmes avec le clan des rouges ou des bleus. :)

En conséquence, il faut que je précise d'abord que je ne fais aucune différence entre MorphOS et OS4 et que je continuerai à porter tous mes logiciels sur les deux systèmes d'exploitation, si possible.

Comme je connais plus ou moins l'état d'avancement d'OS4, je dirai que l'avance qu'avait MorphOS a disparu au fil des années. Je ne suis pas au courant des problèmes restants pour les pilotes de périphériques pour A1 donc je ne ferai pas de commentaires, mais sur ma carte CSPPC le système me paraît très stable et il n'y a pas grand chose qui me manque.

J'aime RoadShow, la pile TCP/IP d'OS4. Ce qui est un point faible de MOS. J'utilise MOSNet mais ce n'est pas vraiment comparable.

Ensuite il y a le triste problème de manque de matériel. Même le meilleur OS n'est rien sans hardware pour le faire fonctionner. J'ai un Pegasos2 depuis 3 ans et il pourrait aussi faire un très bon hardware OS4. Mais nous connaissons tous les difficultés.

Du côté logiciel, il manque à OS4 la même chose que pour tous les systèmes Amiga-like : un navigateur internet au goût du jour, GIMP, Java et peut-être une suite office. Je pourrai bien vivre sans les deux derniers mais d'autres ne peuvent pas.

En tant que développeur, qu'est-ce que tu aimes ou n'aimes pas dans OS4/MOS ?

La nouvelle interface d'OS4 est une bonne idée, mais l'inconvénient est l'augmentation de l'incompatibilité avec les APIs de OS3.x et MOS. De manière générale, il est plus long de porter quelque chose sur OS4, mais c'est aussi à cause de plusieurs modifications et corrections nécessaires dans les fichiers header. Et c'est aussi plus difficile de garder un logiciel portable sur toutes les plate-formes.

Le SDK et les outils de débbugging sont un gros avantage d'OS4. Le SDK est constamment mis à jour et c'est très important d'avoir des outils comme le debugger gdb (bien que les dernières versions de gdb que j'ai pu tester n'acceptaient pas les sorties DWARF2 de vbcc).

La documentation développeur d'OS4 est aussi ok. C'est quelque chose que je déteste vraiment sur MorphOS ! Il n'y a pas de nouveau SDK depuis des années et les autodocs que j'ai ne décrivent que MUI, CyberGfx, quelques extensions de Exec et Intuition. Beaucoup de fonctions et paramètres ne sont même pas indiquées, et si vous êtes chanceux, vous trouverez des définitions dans les fichiers header avec lesquels vous pouvez faire des essais.

J'aime bien aussi que OS4 utilise de vrais exécutables ELF. Cela me semble un peu plus « clean » que de traiter des fichiers objets repositionnables comme des exécutables.

Volker Barthelmann :

Peux-tu nous dire quelques mots sur vbcc ? As-tu prévu des mises à jour de vbcc ? Support de C++ ou d'autres fonctionnalités ?

De manière générale il y aura, à l'avenir, un travail continu sur vbcc comme cela a été le cas par le passé. Frank est le décideur pour les fonctionnalités liées à l'Amiga. À part les modifications sur le noyau du compilateur, je fais des améliorations relatives aux systèmes embarqués et des recherches avec vbcc. À titre d'exemple, je donne actuellement des conseils à un étudiant d'université qui écrit une thèse qui est liée à vbcc.

Il est prévu (et du travail a déjà été réalisé) pour ajouter à vbcc EC++ (Embedded C++, un sous-ensemble de C++ pour les systèmes embarqués). Mais il n'y a rien de prévu pour implémenter un C++ complet. Étant donné le travail nécessaire, il y a peu de chance que cela se produise à moins que quelqu'un ne paye pour cela.

À tous ceux qui utilisent vbcc: Have fun !

◆ 2004-2011 IntuitionBase