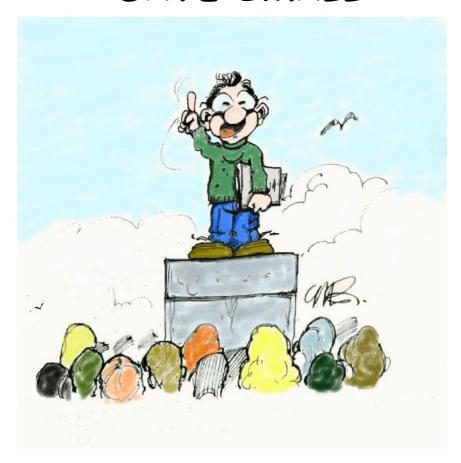
# LE PETIT MONDE DE DAVE SMALL



Recueil des articles d'un bidouilleur de génie parus dans



Entre aout 1991 & mars 1994

# Le petit monde de Dave Small



Ces articles sont parus dans ST Mag.
Tous les articles ont été écrit par David Small et traduit par Password90 (Frédéric Mora)
Recueil par Franck Meric http://www.labibleatari.fr

# Le petit monde de Dave Small

Le petit monde de Dave Small (ST magazine n°53).	4
Souvenir de hacker (ST magazine n°54).	9
Philosophie et Conception du SST 68030 (ST magazine n°55).	16
Je me souviens (ST magazine n°56).	23
Aidez quelqu'un d'autre (ST magazine n°57).	29
Le salon WAACE Atarifest' 91 (ST magazine n°58).	37
Les affres de la création (première partie) (ST magazine n°59).	42
Les affres de la création (deuxième partie) (ST magazine n°60).	48
Guerre dans l'ordinateur (ST magazine n°61).	53
<u>Le feu sacré (ST magazine n°62)</u>	58
Le test de Myers-Briggs (ST magazine n°64)	67
Hérésie (première partie) (ST magazine n°65)	73
Hérésie (seconde partie) (ST magazine n°66).	79
Rationalisme (première partie) (ST magazine n°67)	86
Rationalisme (deuxième partie) (ST magazine n°68).	91
MMU et magie (ST magazine n°70)	97
La face cachée de la télématique (ST magazine n°71).	103
Le cauchemar des études (ST magazine n°73).	110
Qui est vraiment Jerry Pournelle? (ST magazine n°74).	117
Tesla, le plus grand des bidouilleurs (ST magazine n°75).	123
Des ordinateurs qui doutent (ST magazine n°76).	129
Le sorcier blanc et le spectre couleur (ST magazine n°77).	137
Le principe d'incertitude d'Heisenberg (ST magazine n°78).	143
CYBER par Dave Small (ST magazine n°79).	149
Les quatre lois de Small 1- La loi de l'effort maximum (ST magazine n°80)	156
Les quatre lois de Small 2- La loi du chaos grandissant (ST magazine n°81).	162

# Le petit monde de Dave Small

Dave Small est le développeur bien connu de Spectre, l'émulateur Mac pour ST, à ce titre, il mérite bien sa gloire de Grand Bidouilleur parmi les Bidouilleurs (en anglais, "Hacker"). Mais être un hacker, qu'est-ce vraiment? Dave a écrit une série d'articles où il nous fournit justement un aperçu de ce que fut sa jeunesse tumultueuse. Ce mois-ci Dave nous plonge dans Star Trek et l'assembleur.

Prenez-en de la graine...

#### Motivation

Pourquoi ai-je appris l'assembleur?

Pour gagner de l'argent ? Non. Pour améliorer ma carrière ? Non. Pour sortir de la fac avec un diplôme ? Non.

J'ai appris l'assembleur pour pouvoir mieux jouer à Empire. Attention, nous ne parlons pas ici du jeu "Empire" sur ST. Nous parlons d'un fantastique jeu multi-joueur basé sur la série Star Trek et joué sur le réseau PLATO s'étendant à travers les États-Unis. Il permettait de faire s'affronter jusqu'à 32 joueurs, répartis en 4 équipes, qui chacune devait "conquérir la galaxie".

Le système PLATO [NdT ce système était normalement destiné à l'enseignement assisté par ordinateur, et se composait de terminaux graphiques reliés par modem à un ordinateur central] tournait sur un gros ordinateur central Cyber de Control Data Corporation (CDC), situé à Minneapolis, dans le Minnesota.

L'affichage était constitué d'écrans graphiques de 512 x 512, ressemblant beaucoup aux écrans monochromes de 640 x 400 du ST. On y voyait des petits vaisseaux spatiaux, des torpilles à photons, des faisceaux de phasers, des planètes, des soleils, etc. Il y avait environ 50 commandes, comme tourner, régler la vitesse, brancher les écrans de camouflage, les rayons tracteurs et autres gadgets. Tout ça se déroulait en 1978, alors que j'étais étudiant à la Colorado State University (CSU). Durant mon séjour à la CSU, j'ai bien dû passer deux mille heures sur le réseau PLATO, dont pas mal sur Empire.

Empire était apparu depuis un certain temps sur PLATO, et était un jeu très populaire. C'était vraiment le jeu pour les machos du clavier : le secret de la victoire résidait en une frappe très rapide et très précise. De ce fait, les dactylographes au toucher, qui frappaient à toute allure avec tous les doigts sans jamais regarder le clavier, gagnaient en général les batailles.

Par exemple, supposez que vous soyez proche d'un ennemi. Une séquence de commande typique était : k (flèche) 9 : vitesse 9 dans la direction de la flèche ; P (flèche) RETURN : tire 3 torpilles à photons dans la direction de la flèche;

P (flèche> RETURN : idem; P (flèche> RETURN : idem;

F tire aux phasers dans la dernière direction entrée ; BACK D BACK D : force trois mises à jour du système pour faire exploser les torpilles en bout de course, de préférence sur l'ennemi.

L'idée générale était de prérégler un saut hyperspatial pour vous éloigner de l'ennemi, lui balancer neuf torpilles dans la figure, puis de sauter au loin immédiatement avant de faire exploser les torpilles. L'hypersaut était nécessaire parce que l'ennemi était en train de vous faire subir exactement le même traitement, et il fallait tâcher d'être loin quand ses torpilles détonnaient.

La séquence de touches ci-dessus devait être entrée aussi rapidement que possible. L'effectuer en une à deux secondes faisait de vous un adversaire médiocre face à un joueur déterminé.

Et certains joueurs avaient pris des cours de dactylographie [NdT c'est courant, cela rapporte des

unités de valeurs dans les universités américaines, et cela rend service en cette ère de claviers omniprésents], et avaient donc une frappe extrêmement rapide et précise. Maintenant, imaginez une salle sombre, éclairée par les écrans de 15 utilisateurs, tous jouant à Empire. C'était hallucinant.

#### Le secret de Dave

Maintenant, voici un petit secret que je n'ai jamais encore révélé à quiconque : je ne frappe pas au toucher. Je n'ai jamais su le faire.

Quoi ? dites-vous. Dave Small, frapper avec deux doigts ? Eh oui, c'est vrai. J'admets que je suis rapide avec mes deux doigts, mais je fais beaucoup d'erreurs, la touche Backspace de mon système s'use rapidement (et imaginez le code source de Spectre CCR, 1500 000 touches frappées, tout ça avec deux doigts...). Du coup, je me faisais régulièrement massacrer à Empire par les super dactylographes. En général, je m'approchais d'un vaisseau Klingon, frappais des commandes, manquais une touche, regardais l'écran, voyais que toutes les commandes suivant la faute de frappe avaient été rejetées. Puis je voyais le message "Vous avez été tué par..."

Je faisais partie de l'équipe de la Fédération, les "Feddy Bears", qui était au dernier rang du classement d'Empire.

La plupart des employés de Control Data, ainsi que les utilisateurs de la grande rivale détestée, la Colorado University (ne pas confondre avec la Colorado State University), jouaient dans l'équipe des Klingons.

#### Le terminal IST-1

Mon terminal PLATO était un IST-1 de Control Data. Il contenait un microprocesseur Intel 8080, avec environ 4 ko de RAM disponible (et attention, on pouvait commander une extension de 16 ko, ce qui donnait une incroyable quantité de mémoire!

Rappelez-vous que c'était en 1978. Et de plus, il y avait sur le central un assembleur pour ce microprocesseur, appelé "pptasm". On pouvait télécharger un programme depuis le central vers l'IST-1 puis l'exécuter.

Une nuit, une pensée malveillante se glissa dans mon esprit : et si je pouvais programmer l'ordinateur pour frapper des touches à ma place ? Tout ce que faisait le 8080 de l'IST-1, c'était surveiller le clavier, et s'il détectait une touche enfoncée, il l'envoyait au central. Si des octets arrivaient du central, il réalisait les affichages correspondants. En fait, le 8080 exécutait un programme de gestion de terminal. Ce qui me tentait, c'était un programme de macrocommandes comme beaucoup de gens en utilisent de nos jours sur leurs émulateurs de terminaux. Je commençai donc à apprendre le 8080. Heureusement, j'avais un exemple de programme, par Mark Sandmann, qui me facilita bien les choses. le plongeai dans les manuels. j'appris tout sur l'assembleur 8080 : accumulateurs, registres, JMP et CALL... Et chaque nuit, alors que je continuais à me faire massacrer à Empire, ma détermination croissait.

Je décidai de connaître le 8080 ou de mourir à la tâche. Je séchais les cours, sautais des repas, dormais dans la salle PLATO, entre autres excès... Finalement, je fis tourner la première version de mon programme. Si on pressait juste une touche de flèche, le terminal envoyait un P, puis la flèche, puis RETURN - trois touches d'un coup i Cela améliora considérablement les choses pour moi. Je pou vais simplement presser trois touches de flèches pour lancer des torpilles. Mon score commença à progresser.

Je me mis alors à rajouter des options. Je créai la redoutable combinaison plus flèche, qui mettait le cap à l'opposé de l'ennemi, réglait la vitesse 9, envoyait à l'ennemi neuf torpilles et trois coups de phasers, puis forçait le central à faire détonner les torpilles. Tout ce qu'il y avait à faire pour déclencher

cette longue séquence était de presser la touche "+" puis une touche de flèche. Puis je me demandais pourquoi il fallait presser une flèche. Le système savait où était l'ennemi, puisqu'il venait de le dessiner à l'écran. Je modifiai le programme pour calculer la direction où était l'ennemi et l'envoyer automatiquement. J'ajoutais aussi une option pour voir où vous expédiait exactement un hypersaut, sous la forme d'une série de cercles concentriques centrés sur votre vaisseau, comme un sonar.

C'était vraiment drôle de presser seulement une touche et de voir le témoin de transmission s'éclairer sur le modem du terminal, tandis que mon programme expédiait les séquences de touches à la vitesse maximale acceptée par le système. J'étais devenu, si vous voulez, le dactylographe virtuel absolu.

J'en informais l'équipe de la Fédération, et bientôt nous jouions tous en utilisant mon programme. Et nous gagnâmes une partie contre les Klingons tant détestés, puis une autre, puis d'autres encore. Les Klingons étaient vraiment abasourdis ils tapaient à la vitesse maximale, et nous les surpassions.

Le moment que je me rappelle le mieux se trouvait lorsque j'étais l'un des deux défenseurs en orbite autour de la Terre, dernière planète restante de la galaxie. Quatre Klingons apparurent. Je fis un hypersaut au milieu d'eux, et les détruisis dans un tourbillon d'action.

Cette année-là, l'équipe de la Fédération dépassa les Klingons et devint championne, ce qui donna lieu à moult réjouissances. En fait, lorsque je me mariai à Sandy en 1981, je conduisis une voiture de location depuis Austin (Texas) jusqu'à Minneapolis (Minnesota) pour fêter l'enterrement de ma vie de garçon, car la plupart des membres de l'équipe de la Fédération étaient au siège de Control Data de Minneapolis. J'avais loué la voiture sous contrat de kilométrage illimité, et je me rappelle toujours le visage horrifié des employés du bureau de location lorsqu'ils calculèrent la distance que j'avais ajouté au compteur en un seul week-end.

Bien entendu, l'équipe des Klingons (parmi laquelle figuraient de nombreux programmeurs système de CDC) découvrit vite mon astuce, et écrivit ses propres programmes "presse-touches". Bientôt, Empire devint une guerre de terminaux automatiques.

Il y avait même des programmes capables de réaliser une mission entière (par exemple, mettre le cap vers une planète ennemie éloignée, bombarder toutes les armées s'y trouvant, et lancer un message aux forces amies annonçant que la planète était prête à être capturée). Si vous lanciez ce programme sur dix terminaux à la fois, vous pouviez vraiment voir l'ennemi s'étrangler de rage.

Je trouvais tout cela très distrayant. J'aimais opposer mes capacités de programmeur à celles des programmeurs système, et nous nous amusions beaucoup.

#### Politique d'entreprise

En même temps, je tentais de corriger un problème vraiment irritant sur l'IST, tout en me faisant des ennemis à CDC. Chaque fois qu'il y avait du bruit sur la ligne connectant l'IST-1 au central, l'IST-1 plantait, le voyant ERREUR s'allumait, et le terminal était bloqué. Il fallait faire un RESET du terminal, ce qui avait pour effet de vous déconnecter. Bien sûr, on évitait à l'IST-1 de recevoir des données endommagées, mais il aurait fallu qu'il récupère après erreur... C'était un problème logiciel que CDC ne voulait pas corriger. Ça me rendait dingue. Souvent, je jouais à Empire, je massacrais l'ennemi, et brusquement, la ligne prenait un parasite, le terminal était mort, et j'étais déconnecté.

Je fouinais alors dans la mémoire de l'IST-1 jusqu'à ce que je trouve l'indicateur d'erreur, puis je fis un programme simple toutes les demi-secondes, il testait cet indicateur, et, s'il était à 1, le remettait à 0. C'était gagné, le terminal ressuscitait.

Désormais, l'IST-1 encaissait des parasites sur la lignes et les ignorait. Bon, d'accord, de temps à autre, des affichages aléatoires apparaissaient, mais peu importait. Cela valait bien mieux que d'avoir à se reconnecter au système.

Ce petit bidouillage devint un programme extrêmement populaire sur PLATO. De plus en plus de gens se mirent à l'utiliser. Du coup, ils cessèrent de signaler des erreurs de ligne au service central des télécommunications de PLATO, puisque les erreurs étaient ignorées. Et les membres de ce service des

télécoms se trouvaient être dans les Klingons d'Empire. Vous voyez le coup venir, non?

Les gens des télécoms de CDC se mirent à m'en vouloir beaucoup. Même énormément. Imaginez le bel exemple de politique d'entreprise que cela donnait «Les clients ne nous signalent plus d'erreurs à cause du programme de ce Dave Small.» «Oui, l'assembleur, ce n'est vraiment pas à laisser aux clients. Retirons-le.» Ce fut mon introduction à la politique d'entreprise. CDC ne voulait pas corriger ce problème idiot dans le programme du 8080 de l'IST-1 ; par contre, il retirèrent aux clients l'outil grâce auquel il pouvait résoudre le problème lui-même. Cela revenait à retirer une parcelle de responsabilité au client pour la rendre à CDC. Une bataille de plates-bandes, en d'autres termes.

Une annonce très controversée fut donc faite, selon laquelle tout accès au langage assembleur de l'IST-1 serait supprimé après une certaine date. Cela outragea vraiment les clients de CDC, parmi lesquels certains utilisaient l'assembleur à des fins légitimes - l'entraînement au vol à American Airlines, par exemple. Mais CDC fut têtu comme une mule, et supprima l'accès à l'assembleur malgré tout (des exceptions furent faites pour certains clients qui payèrent cher ce privilège, mais pas pour le commun des mortels).

Naturellement, les programmeurs système gardèrent accès aux outils de téléchargement et à l'assembleur, ainsi qu'aux presse-touches d'Empire. Quelle coïncidence ! Les Klingons recommencèrent à gagner à Empire, et je sais parfaitement comment ils faisaient.

Je ne me le tenais pas pour dit. Je déteste voir l'accès à quelque chose d'utile être refusé, surtout pour de stupides raisons politiques. C'est là une base de la philosophie des hackers : liberté d'échange des informations. Mon premier geste fut d'obtenir un compte sur le CERL [NdT : un autre réseau national américain], qui était un autre système PLATO où l'accès à l'assembleur était toujours permis.

J'appelais donc la machine CERL dans l'Illinois, me connectais sur la machine, téléchargeais mon presse-touches pour Empire qui était stocké là-bas, et me connectais ensuite sur le système CDC avec mon terminal convenablement configuré. Le problème était que cela représentait un appel longue distance et que je ne pouvais pas donner ce programme à quiconque ne disposait pas d'un compte CERL. (CERL a toujours maintenu un accès bien plus libre et ouvert à son système que CDC. Aujourd'hui, on peut toujours leur demander un compte PLATO peu coûteux et utiliser un émulateur de terminal Mac ou IBM. Oui, le programme Mac tourne sous Spectre Leur version d'Empire ou de Moria est excellente.)

J'ai donc ouvert l'IST-1, trouvé l'EPROM de démarrage, et examiné ce qui se passait durant un RESET. J'en appris assez sur les EPROM pour en faire une nouvelle moi-même.

Cette nouvelle EPROM contenait un programme qui, durant le RESET, regardait si une certaine donnée se trouvait sur l'écran ; si oui, le RESET était interrompu, et un téléchargement pouvait avoir lieu. J'écrivis alors un petit programme tournant sur le Cyber central qui envoyait à l'écran du terminal un programme en assembleur 8080. A ce moment, je pressais RESET, ce qui chargeait ce programme assembleur dans la mémoire de l'IST-1, et j'étais à nouveau dans la course.

En d'autres termes, j'avais bidouillé le terminal pour me permettre de télécharger des programmes en assembleur, en dépit de CDC. Et ce fut là que j'appris à connaître les ROM, EPROM, brochages... ce qui me fut par la suite bien utile lorsque je créais le Magic Sac, puis le Spectre.

Le bruit se répandit parmi les hackers que mon petit bidouillage marchait, et bientôt je fus submergé par des demandes d'EPROM. J'en envoyais aux quatre coins du monde - certaines en Islande, d'autres en Norvège, et partout aux USA.

Le programme de téléchargement du central se mit à être utilisé intensivement, car les gens téléchargeaient le programme antiparasite et le presse-touches d'Empire. Les programmeurs système étaient furieux, et ne pouvaient absolument rien y faire. C'était une situation politique idiote, j'avais récupéré les plates-bandes qu'ils avaient accaparées.

Cependant, je me fis des ennemis parmi les grosses légumes de PLATO, et cela me joua plus tard des tours, lorsque les hackers quittèrent PLATO et que la faction politique fut aux commandes sans opposition. La dernière fois que je me suis connecté à PLATO, c'était il y a quelques années, alors que je testais un émulateur de terminal pour Mac sous Magic Sac (ça marchait). PLATO était alors devenu un panier de crabes où diverses factions s'arrachaient des territoires. Des barrières étaient érigées de toutes parts. C'était devenu un état policier sur ordinateur. Je m'en allais en décidant de ne m'en

#### Conclusion

Je n'ai pas appris l'assembleur en classe. Je n'ai pas écrit les programmes classiques avec lesquels on apprend l'assembleur. je ne l'ai pas appris pour passer un examen. Je l'ai appris parce que brûlait en moi un profond désir de massacrer des Klingons outrecuidants.

Cela m'a suffisamment motivé pour tout apprendre des compléments à deux, registres, affichages, etc. C'était une croisade, qui s'est terminée par une victoire pour moi.

Si vous voulez apprendre un langage, que ce soit l'assembleur ou le C, je vous recommande fortement de vous assigner un but, une raison pour laquelle vous voulez l'apprendre, qui vous motivera. Cela fera de l'apprentissage un jeu au lieu d'une corvée. J'ai appris de nombreux langages depuis le 8080, et j'ai appliqué ce principe à chaque fois. Les premières fois, c'était inconscient, par la suite, ce fut délibéré. Cela m'a permis de continuer à m'amuser en programmant.

Le Magic Sac et le Spectre 128 étaient plus ou moins la même chose. Je finissais par être las d'entendre les gens dire qu'il était impossible pour un ST d'exécuter les programmes d'un Mac. J'ai donc prouvé que cela pouvait être fait. Pour cela, il m'a fallu apprendre le 68000.

Ces temps-ci, il me faut retourner au C, puisque ce langage devient l'Espéranto de l'informatique ; c'est pourquoi je suis à la recherche d'un but qui me motiverait pour l'apprendre. Je n'en ai pas encore trouvé, mais je continue à chercher.

Une autre chose que j'ai apprise dans cette affaire est l'importance de la politique d'entreprise. La défense du territoire est bien plus importante que le service rendu au client, et PLATO a payé le prix de cette conception : aujourd'hui, le système est bien plus petit que ce qu'il était il y a 10 ans. J'ai découvert que c'était le cas dans un très grand nombre de corporations, ce qui est la raison pour laquelle de petites boîtes compétitives peuvent vaincre de grosses corporations à chaque fois ou presque. Ce



n'est que lorsque les gens ont suffisamment de temps libre pour jouer au politicard qu'une compagnie commence à avoir des problèmes.

Ma solution à ce problème a été de ne plus jamais travailler pour une corporation.

En conclusion de cette histoire, sachez que lors du Comdex 1990, j'ai rencontré un vieil ennemi Klingon qui m'a rendu visite au stand Spectre 128. Il était employé chez CDC depuis longtemps lorsque nous nous affrontions à Empire. Il avait toujours eu assez de classe pour que nous restions amis, et lorsque nous nous étions parlé la dernière fois, huit ans auparavant, nous étions amis. Il m'a donné sa carte commerciale, il travaille à présent chez Apple.

Traduction: Password

# Souvenir de hacker

Ce mois-ci, Dave Small nous fait un exposé d'archéo-informatique, science des ordinateurs des temps reculés où ces machines étaient énormes, ruineuses, et assurées contre les incendies et les raids des Vikings. En même temps, il nous prouve que sa vocation de bidouilleur (en anglais, hacker) s'affirmait déjà avant même qu'il ne passe son bac.

### Souvenir de hacker

Beaucoup de gens affirment que le bon vieux temps des hackers est révolu. Le livre "Hackers" de Steve Levy (chaudement recommandé, il contient même une partie sur l'Atari 8 bits et sur John Harris, auteur des jeux Jawbreakers et Frogger), a un épilogue intitulé "le dernier hacker" sur Richard Stallman, de la Free Software Foundation, père du projet GNU (NdT voir ST-MAG numéro 47). Et Rick Cook, auteur de SF et d'Heroic Fantasy, a écrit de moi sur BIX (NdT : le serveur du magazine américain Byte) "Dave est l'un des derniers bidouilleurs fous restant." J'ai tout simplement envie de pousser un soupir désespéré. C'est de l'égotisme aigu! Voici des gens prétendant que, sous prétexte qu'ils ont vécu durant le Bon Vieux Temps, ce qui se fait actuellement n'est pas de la Bidouille. Vous savez, la vraie bidouille, avec le menton hautain et tout. C'est l'équivalent du grand-père disant "Moi, quand j'étais enfant et que j'allais à l'école, il fallait que je marche dix kilomètres pieds nus, dans les tempêtes de neige, en montant la pente à l'aller et au retour." Peuh ! Petite pause le temps d'une définition le bidouillage, ou hacking, ne consiste pas à faire intrusion dans des ordinateurs bancaires, entre autres méfaits vulgarisés par les mass-médias.

Le bidouillage consiste à repousser les limites du possible, à "pousser l'enveloppe" (NdT : expression d'ingénieur aéronautique, l'enveloppe étant celles des courbes paramétriques définissant le domaine de vol d'un avion. Pousser l'enveloppe, c'est élargir les conditions limites entre lesquelles l'engin tient toujours en l'air). Cela ne se limite pas à l'informatique. Il y a des bidouilleurs dans toutes les occupations et tous les loisirs.

Goddard était un bidouilleur de fusées, Scholz un bidouilleur de guitares et d'électronique analogique, etc. Il faut que j'y aille de mon couplet chaque fois que j'aborde le sujet, de façon à ce que les gens sachent que je ne parle pas du film WarGames ni du "ver" du réseau Internet.

De plus, je commence à être fatigué d'entendre des gens se vanter de ce qu'ils arrivaient à faire avec huit malheureux kilooctets sur un PDP-8 (ou autre bécane antédiluvienne), tout en ricanant à propos des mégaoctets de mémoire des machines d'aujourd'hui, comme si les défis n'étaient pas toujours aussi grands. J'ai une nouvelle à annoncer à ces gens-là : les bidouilleurs ne sont pas (mais alors, pas du tout!) dans une maison de retraite à bavasser sur le labo d'intelligence artificielle du MIT dans les années 60. Le bidouillage est vivant, bien vivant.

#### **Quelques exemples**

Dez (Jeremy) San est un très bon programmeur. Son jeu StarGlider II relève du génie pur. Il a créé une disquette qui boote et fonctionne à la fois sur Amiga et sur ST. Un seul programme pour deux machines, et qui plus est, un programme écrit en assembleur 68000 ultra-optimisé.

C'est incroyable. Des machines complètement différentes, des architectures, Principes d'affichages, sons, complètement différents, et ce gars fait un disque fonctionnant sur les deux machines

J'aimerais beaucoup travailler avec Jez San. Je l'ai rencontré lorsque je suis allé en Angleterre avec ma femme Sandy, il y a quelque temps, et nous sommes restés en contact. Il dépense ses royalties en appels transatlantiques pour se connecter à BIX (!).

- De viens d'avoir une copie de la Union Demo (NdT : c'était en 1989, on a fait mieux depuis !). C'est gratuit, téléchargez-la, elle vaut largement le temps de transfert. Grâce à des temporisations bien calculées, le logiciel parvient à supprimer les marges imposées autour de l'écran par les circuits du ST. Cela leur permet de faire des graphismes plein écran. Et bien sur, cela a été fait en langage assembleur.
- Charles Johnson et John Eidsvoog de Codehead continuent de bidouiller le TOS du ST audelà de toute vraisemblance (et comme le dit leur T-shirt "Ici on parte assembleur." Je crois que je suis un peu partial envers l'assembleur, hein?).

L'air de ne pas y toucher, ils ont écrit un remplacement pour le GDOS (nommé G+ PLUS), un extenseur d'accessoires de bureau (MultiDesk), et autres choses réputées "impossibles". Ils m'ébahissent en permanence. J'aimerais beaucoup travailler avec eux aussi.

Mon petit bidouillage personnel, Magic Sac, puis Spectre 128, à présent le Spectre GCR, est plutôt bien accepté à présent. Même les bigots du Mac en ont entendu parler, et admettent à contrecœur qu'il marche. De plus, j'ai honnêtement le sentiment que je ferais avec mon projet Hyperweb la même chose que ce que SideKick a fait sur PC complètement changer la façon dont nous utilisons nos machines. Hyperweb m'accapare actuellement tant l'esprit qu'il interfère avec les touches finales que j'apporte à GCR.

Je m'efforce de ne pas écrire au sujet d'Hyperweb tant qu'il n'est pas publié et offert à l'admiration générale. Notez que Hyperweb est une marque déposée de Gadgets By Small... Donc, ceci n'est pas un article sur le bon vieux temps à jamais révolu du vrai bidouillage. C'est juste une poignée de souvenirs amusants que je vous livre.

#### Viens-en au fait Dave!

Ça va, d'accord, d'accord

Je suis allé au collège de XXXXX (Note: le nom de l'école a été effacé pour protéger les coupables). Notre système informatique consistait en un télétype (NdT : *ou téléscripteur*) ASR-33~ Nous nous connections par modem et téléphone à un système en temps partagé appelé "Hewey", un ordinateur Hewlett-Packard 2000C.

Le système HP 2000C était en lui-même un joli bidouillage. Il était constitué de deux mini-ordinateurs, un HP 2114 et un HP 2116, et se débrouillait pour supporter 32 utilisateurs simultanément. Ce qui n'était absolument pas trivial. Pour fournir le contexte historique (je suis né en 1958), disons qu'il utilisait une mémoire à tambour pour son espace de pagination et disposait de 64 ko de mémoire centrale. Le tout nouveau système 2 000 F utilisait un disque amovible de cinq mégaoctets - vous savez, ceux qui faisaient 60 cm de diamètre (trop jeune pour vous en rappeler, hmmm ?). Naturellement, vous appreniez d'abord à faire tout ce que l'ordinateur vous laisse faire avant de vous mettre à bidouiller. Je plongeai donc dans les manuels HP, qui étaient aussi mauvais que le sont les manuels du ST ou d'Unix aujourd'hui, et j'appris le basic HP. Il n'y avait pas d'autre langage, ni d'éditeur. Ensuite, j'ai commandé à Cupertino (NdT : siège de Hewlett-Packard) les manuels de l'assembleur HP 2100. Je connaissais un peu le basic, mais je devins expert en basic HP. J'appris les finesses de l'instruction ASSIGN, et me tins au courant des nouveaux appels systèmes introduits par HP. Par exemple, la fonction PRO(), qui interdisait à l'utilisateur d'interrompre le programme. Je fis tout cela au détriment du travail scolaire, que je trouvais ennuyeux à en crever. D'où mon modeste 13/20 de moyenne générale. Pas terrible, direz-vous. Mais supposons que VOUS ayez eu un cours d'histoire

américaine enseigné par un prof atteint d'un complexe de culpabilité aigu, qui aurait passé presque tout son temps sur les Bons Sauvages Indiens et comment les vilains blancs les ont oppressés. Une journée sur les trois milliards d'années de la Terre jusqu'à Christophe Colomb, puis la révolution américaine, puis un millénaire sur les Bons Sauvages, puis une journée, à la fin, sur les Indiens à travers la guerre de Sécession. Encore aujourd'hui, je ne peux m'empêcher de frémir en regardant un Bon Sauvage en peinture. Eh bien! je suppose que VOUS aussi préféreriez vous asseoir à ce télétype ASR33 et réfléchir à des bidouilles Ce terminal était rapide : dix caractères par seconde. Je l'avais monté à onze, ce que l'ASR-33 supportait, même s'il faisait un drôle de bruit (mais impossible d'aller iusqu'à douze). Cette vitesse correspondait à une transmission de 110 bauds, ce qui est la raison pour laquelle tous les programmes de télécommunication du monde doivent encore supporter cette vitesse lamentablement lente : quelque pauvre hère pourrait encore avoir, quelque part, un télétype. L'ordinateur disposait de disques durs d'environ 20 mégaoctets. Les trois guarts de ces disques étaient interdits pour une raison purement politique l'administrateur essayait d'obtenir un disque plus gros, et tentait d'appuyer sa demande par des plaintes d'utilisateurs mécontents, plaintes qui, il l'espérait, se multiplieraient à cause de l'espace disque insuffisant et l'aideraient à faire aboutir sa demande. Ma mémoire de masse personnelle consistait en bandes de papier perforé. Elles offraient une inépuisable source d'amusement : à la fin de la journée, on ramassait les minuscules confettis dans le perforateur et on les jetait dans les cheveux de quelqu'un. Quoi qu'on fit, l'électricité statique les maintenait dans la chevelure jusqu'à la fin du semestre. Même aujourd'hui, j'ai de ces bandes de papier dans ma boîte à souvenirs, et les regarder me fait chaud au cœur. Ma femme Sandy veut que je les jette. Argh j'avais aussi acheté une bande magnétique (!) et avais demandé aux opérateurs système d'enregistrer mes fichiers dessus. Ce qui avait probablement occupé deux mètres sur les 800 de la bande, mais quelle sensation!

#### Joyeux noël à tous!

Un jour, je remarquai que l'ASR-33 était "à réponse automatique", comme un modem du même métal. En fait, quand on appelait le télétype depuis un autre terminal, il se mettait en marche automatiquement comme par magie. Cela était nécessaire pour que les télétypes de la Western Union remplissent leur fonction originale, la transmission des télex. Un caractère de fin de transmission, Control-D, remettait le terminal à l'arrêt (les parasites sur la ligne créaient beaucoup de faux Control-D, qui arrêtaient le terminal, jusqu'à ce que je bloque la tige de détection de ce caractère à l'aide d'un élastique).

Hélas!, la salle où se trouvait le télétype était celle où enseignait un prof de math, un certain jack Myslik, dit Jack le Pantin. Nous, les bidouilleurs, étions régulièrement expulsés de la salle, à notre grande irritation. J'allais donc au second terminal, dans un autre bâtiment, et la veille des vacances de Noël, je mis en place un complot. Jack faisait son cours. Dans un coin se trouvait le télétype. Brutalement, sans que quiconque y touche, l'ASR-33 se mit en marche. Et faisant joyeusement sonner sa clochette, il cracha du papier, imprima "Joyeux Noël à tous", signé "Hewey l'ordinateur", puis se déconnecta. Dès cette époque, j'avais appris à me couvrir. J'étais accompagné d'un autre professeur, et nous étions tous deux morts de rire tandis que j'introduisais une bande de papier dans le second terminal qui avait appelé celui de la salle de maths. Je pensais que la double caution d'un professeur et de la saison des voeux-aux-hommes-de-bonne-volonté me mettraient à l'abri des représailles. Curieusement, cela s'avéra exact (depuis, je n'ai plus guère eu de chance en pariant sur la bonne volonté des gens). Jack le prit remarquablement bien. La classe en resta, me dit-on, stupéfaite.

#### Les experts es HP

Bientôt, je commençai à explorer le monde qui gravitait autour du HP. Il y avait une bibliothèque de programmes systèmes très intéressante. C'était les mêmes noms qui apparaissaient sans arrêt dans le code source, et c'est de ces gens que j'ai appris le plus. Les opérateurs du HP s'appelaient Phil Tubb et John Ridges, et ils étaient vraiment brillants. Plus tard, ils formèrent ALF, une boîte qui faisait des cartes musicales et à coprocesseur 68000 pour Apple II, puis une affaire de duplication de disques très profitable. John Ridges est sans conteste le meilleur programmeur que j'aie jamais rencontré, HackerCon ou pas, et un concepteur de matériel hors pair. Même Dan Moore, qui n'est pas manchot, le dit sans hésiter. Et John a également une personnalité attachante, ce qui manque cruellement à bien des informaticiens géniaux.

Je découvris que ces gens avaient formé un club d'utilisateurs, le Jefferson County Computer Club, ou J3C, qui tenait des réunions mensuelles. Je commençai à y assister, et y découvris encore davantage. Ils avaient créé sur le HP un serveur RTC qui était excellent. C'est là que je participai à mes premières discussions télématiques. De plus, ils enseignaient l'assembleur, et avaient même réussi à persuader le gestionnaire du système de les laisser arrêter le HP le samedi matin, pour que les élèves de ce cours puissent essayer leurs programmes en assembleur (ces programmes ne pouvaient pas s'exécuter sous le basic en temps partagé, qui tenait lieu de système d'exploitation).

Ainsi, ils avaient écrit un assembleur en basic HP, qui transformait un code source assembleur en binaire craché sur une bande de papier. Petit problème : les ASR-33 utilisaient des mots de sept bits en

parité paire, ce qui interdisait des codes assembleurs supérieurs à 127, puisque le 8 bit de chaque octet était pris par le bit de parité.

"Pas de problème", disait John Ridges. Son assembleur produisait aussi une seconde bande de papier qui masquait le bit de parité correctement. Vous aligniez et superposiez les deux bandes, l'une masquant l'autre, vous glissiez le tout dans un lecteur de bandes non connecté à la machine, en mode recopie, et le perforateur recopiait le tout sur une bande qui contenait le code assembleur correct. Une bidouille typique de John Ridges.

John écrivit aussi MultiComm, le premier programme que je n'ai jamais compris, quoi que j'y fasse. MultiComm permettait à seize utilisateurs de discuter simultanément via leurs terminaux. Il était incroyable qu'il marche, vu que le HP n'était pas du tout prévu pour permettre la communication interterminaux.

#### Mon premier programme en assembleur HP

C'est ainsi que j'écrivis mon premier programme en assembleur HP. Il faisait défiler les LED du panneau de commande de gauche à droite et vice versa, comme une balle de ping-pong. (NdT: le HP était doté de rangées de LED visualisant les registres de son unité centrale, une LED par bit. Les données pouvaient être entrées à l'aide d'inverseurs.) Cela donnait :

```
move.w #1,d0
encore: move.w #15,d1
;
gauche: move.w d0,lumieres
rol.w #1,d0
dbf d1,gauche;
move.w #15,d1
droite: move.w d0,lumieres
ror.w #1,d0
dbf d1,droite;
bra encore
```

Si vous ne comprenez pas ce programme, apprenez donc l'assembleur 68000. Si je vous en convaincs, je vous aurai rendu un fier service. Vous allez adorer (même si l'assembleur 8088 des PC vous a traumatisé, vous allez apprécier le 68000). Donc, en tremblant, j'entrai mon programme dans le HP ce

samedi-là, et, ô surprise!, il ne marchait pas. Tim Gill, un autre membre fondateur du groupe, vint m'aider à essayer de le déboguer. Il fit s'afficher le programme, un mot à la fois, sur les 16 LED du panneau de commande. Il n'avait jamais vu le source auparavant : il visualisait le programme uniquement à partir des 16 LED. "Ha ! ha", dit-il, "erreur fréquente. Voyons ça." Il manipula des inverseurs, pressa quelques fois le bouton "LOAD", modifia le programme légèrement, et celui-ci s'exécuta, faisant défiler les LED. Voyez-vous, Tim connaissait le binaire de l'assembleur HP par cœur, un passe-temps, quoi Tim est très fort. Il fonda par la suite la firme Quark, qui faisait des disques durs pour Apple Il et pour Mac, et qui à présent édite Quark XPress, un excellent programme de PAO causant beaucoup de soucis à Aldus Corp., leader du marché (NDLR : et servant à la mise en page de ST Magazine...). Je crois me souvenir que d'après le magazine InfoWorld, Quark avait cédé à Claris, ex-branche logicielle d'Apple, un vieux code source de traitement de textes pour un rondelet million de dollars... Et je le répète, le J3C n'était que le club informatique de Denver, pas celui de la baie de San Francisco qui est mentionné chaque fois que quelqu'un se lamente en vantant le bon vieux temps. Quand je me souviens de mon adolescence, le J3C m'a vraiment donné le meilleur de ces années. Mon adolescence a été, à part ça, absolument épouvantable, et je ne voudrais à aucun prix la revivre.

#### Bidouille et mauvais coups

Je crois que compte tenu de la mentalité de ces gens qui cherchaient sans cesse à "pousser l'enveloppe", il était inévitable que je cherche à les imiter et que je commence à bidouiller le HP. Ces gens étaient mes héros, ils faisaient des programmes que je n'arrivais même pas à comprendre! Entre autres, nous faisions planter le système. Mais entendons-nous bien, planter le système était, en soi, totalement sans intérêt. Le véritable attrait de la chose, c'était de parvenir à contourner les mesures de sécurité conçues par les génies de HP, et vous prouver que vous pouviez les surpasser. je l'admets (et vous pouvez froncer les sourcils), une fois, j'ai planté le HP... et j'ai compris la leçon (soupir). Notez bien que j'étais ultra-prudent et que je veillais à ne jamais corrompre un fichier. J'ai juste arrêté le système. Ouais !, super, j'étais génial et tout. En attendant, je ne pouvais plus me connecter à Hewey. Plus de serveur RTC, plus de fichiers, plus rien. La leçon fut apprise définitivement durant la demiheure que mit John à relancer le système à partir de bandes de papier "Planter le système, c'est rasoir." Quand je leur révélai l'origine du plantage, john et l'équipe ne s'intéressèrent qu'à la bidouille qui l'avait causé. Ils en informèrent HP, et HP l'ignora autant que je sache. Mais mon prestige augmenta légèrement aux yeux de mes héros : je devenais un hacker, et j'échangeais des informations librement.

Nous découvrîmes le programme Ralentisseur. Sur un système partagé, l'unité centrale accorde à chaque utilisateur à tour de rôle une tranche de temps, et chacun a ainsi l'illusion de disposer de la

machine pour lui seul. Notre HP accordait ainsi 1/32 de son temps à chacun des trente-deux utilisateurs. Nous avions un programme qui utilisait 98% de sa tranche de temps avec deux instructions ASSIGN, et ces instructions étaient suivies d'une instruction ENTER qui forçait l'unité centrale à vous accorder plus de temps qu'il ne l'aurait fallu. On faisait tourner cela sur trois ou quatre terminaux, et le système s'arrêtait pour tous les autres utilisateurs, restant focalisé sur ces quelques terminaux. Nous fîmes l'expérience une fois, et en informâmes John. "Pas mal", dit-il. Il n'eut pas à nous dire : "Ne recommencez pas." C'était implicite. Après avoir découvert cette bidouille, nous nous attelâmes à des choses plus intéressantes.

#### Bidouille créative

A la suite d'un défi lancé par john, j'écrivis en basic un interpréteur basic. Ce programme, bien que très lent, exécutait correctement des sources écrits en basic HP. Il disposait en outre de possibilités de trace et de débogage. Nous avions un jeu Star Trek sur le HP. Je le bidouillais pour qu'il permette de sauvegarder la partie en cours et de la restaurer, de façon à ne pas devoir recommencer la partie

chaque fois que Jack Myslik nous expulsait de la salle. J'en vins à détester la programmation non structurée. (Une telle phrase venant de Dave Small ? C'est vrai, c'est vrai, mais j'ai vraiment dû dépatouiller d'infâmes plats de spaghettis en basic HP, et j'en sais maintenant assez pour écrire du basic lisible à présent). Et puis, pour la première fois de ma vie, je ressentais le besoin de créer quelque chose. Nombreuses furent les nuits où je m'assis au terminal, armé de mes manuels de basic et d'assembleur, et créai. Il m'est difficile d'expliquer l'importance que cela pouvait avoir pour un élève de collège (NdT : à peu près notre classe de troisième). C'était très important pour moi à cette époque, où j'avais besoin de faire quelque chose de neuf et de le faire bien. Parfois, c'était nécessaire. Ainsi, nous avions un problème : un crétin s'amusait à effacer nos fichiers sur le HP, où il n'y avait pas de sécurité. Je créai donc un fichier nommé "000000", une succession de lettres O et de chiffres zéro, que l'ASR-33 imprime quasiment de manière identique. Et ce gars qui se demandait pourquoi il ne pouvait pas effacer un fichier nommé 6-fois-zéro. Hé hé

D'autres fois, c'était amusant. Mon frère Jim avait fait un "jeu du canonnier" (Vous vous rappelez? On entrait la hausse en degrés et la quantité de poudre, et le programme imprimait à quelle distance on avait tiré et de combien on avait manqué la cible). Je le saisis en basic HP, et, à ma grande fierté, les responsables système le placèrent dans la bibliothèque de programmes du HP. Je mis la main sur un superbe nouveau gadget qu'on me prêta: un Hazeltine 2000.

C'était un terminal vidéo d'une vitesse incroyable de 300 bauds, ce qui était alors fabuleux pour moi. Il était doté de l'adressage du curseur : on pouvait déplacer le curseur n'importe où sur l'écran! Bientôt, j'avais écrit un programme d'alunissage avec graphiques pour le Hazeltine. Petite parenthèse en passant.

Lorsque j'entrai plus tard au lycée, mes connaissances en basic et en graphiques me tirèrent d'embarras. Je manquai tout juste de redoubler la première année, mais j'appris le secret des études indépendantes du lycée.

Trouvez un prof acceptant de vous soutenir, étudiez quelque chose que vous voulez absolument apprendre (par exemple, l'assembleur CDC 6600), faites- en un projet scolaire, et vous obtenez un 'A' tout en vous amusant. Ou bien étudiez le synthétiseur ARP 2600, créez une bande de deux minutes complètes, avec un effet sonore à la fin, dont le duplicata se trouve à présent dans la page secrète de dédicace de chaque disquette Spectre, et vous obtenez un autre 'A' en classe de musique. J'ai survécu au lycée, et ai obtenu une moyenne générale d'environ 10,001 sur 20 (NdT : on redouble avec 10 ou moins) grâce à cela.

Fin de la parenthèse.

#### Trivia

Le plus grand événement pour moi fut le défi que nous lançâmes aux responsables système, un défi sous la forme d'un jeu de Trivia (NdT : questions-réponses) concernant le HP. Ils acceptèrent et furent très bon joueurs. Ils bricolèrent même un système électronique de sonnettes et de boutons-poussoirs, à base de circuits intégrés, qui permettait au premier qui avait la réponse à une question de presser un bouton, ce qui activait une sonnette et verrouillait le circuit, empêchant les autres joueurs de l'activer. Phil Tubb accepta d'arbitrer. Je sentis des frissons glacés en m'asseyant et en observant, de l'autre côté de la pièce, John Ridges et Tim Gill. Derrière chaque équipe se trouvait un ASR-33, connecté à Hewey, et nous croulions sous les manuels et les listings dont nous aurions pu avoir besoin. Chaque équipe posait à tour de rôle des questions à l'autre.

Ma question fut la première : "Quels sont les prénoms de Hewlett et Packard?" Sourires dans la salle, ce qui établit l'ambiance pour la soirée. Dring. Tim Gill: "Bill Hewlett, Dave Packard." (Tim travailla plus tard pour HP avant de fonder Quark). A leur tour : "Combien de trous y a-t-il dans 12,9 mètres de bande perforée ?" On sortit les calculatrices, tandis que le chrono avançait. Évidemment, nous manquâmes la réponse à une erreur d'arrondi près.

Nous répliquâmes : "Combien de terminaux peuvent exécuter une boucle faite de deux ASSIGN suivis d'un ENTER avant de bloquer le système?", ce qui rétablit l'égalité des scores. Et ainsi de suite.

Nous en arrivâmes à la dernière question alors que nous étions à égalité. Nous décidâmes d'un commun accord que la dernière question vaudrait suffisamment de points pour que la première équipe y répondant remporte l'épreuve.

Phil demanda aux deux équipes de calculer, avec une précision de deux cents chiffres significatifs, la valeur de 2/41457 (ou quelque chose d'approchant). Je songeai avec désespoir qu'il fallait écrire un programme de division à précision infinie. Vous savez, comme pour la division manuelle : reporter le chiffre, soustraire, multiplier... Tim Gill était déjà sur leur terminal, saisissant le programme auquel il songeait, ligne par ligne. Pris de sueurs froides, je commençais à écrire sur papier l'algorithme que Tim avait écrit dans sa tête. Leur terminal commença à cracher des chiffres : blip (pause), blip (pause), très lentement. Et soudain, je me souvins du programme de calculs en précision infinie de Tim. Il faisait des calculs avec le nombre de chiffres significatifs choisis par l'utilisateur. Et il était court. J'en sortis le listing de mon sac, et demandais à mon équipe de me le lire tandis que je le saisissais. Déjà, en ce temps, je pouvais taper trop vite pour les dix caractères par seconde de l'ASR-33 (et même pour les onze de ceux-là, vous pensez bien que nous les avions bidouillés). En quelques minutes, le programme était saisi, et nous le lançâmes. Notre terminal commença à éditer la réponse, sans aucune ème

hésitation entre deux chiffres. Le programme des responsables système avait dépassé le 150

chiffre lorsque le programme de Tim, sur notre terminal, imprima finalement le 200 . Nous avions gagné.

Tim, hochant la tête, admit qu'il avait tout simplement oublié son programme calculateur en précision infinie. Pour lui, ce n'était qu'une petite bidouille rapide.

Rien, absolument rien de ce que j'ai fait au collège ne peut rivaliser avec ces moments où mes camarades et moi surpassions les meilleurs des bidouilleurs à l'aide de leur propre programme. Ce fut l'un des grands moments de ma vie. Et quelque part à la cave, il y a une boite marquée "Gare à vous si vous jetez cela", contenant les vieux listings et bandes perforées du HP 2000C, jaunissant doucement, les premières choses que j'aie jamais créées sur un ordinateur.

Vive la bidouille		

**Traduction Password** 

# Philosophie et Conception du SST 68030

Ce mois-ci, Dave Small nous montre un aperçu bien tentant de la carte accélératrice SST 68030 pour ST, qui sera peut-être déjà commercialisée par Gadgets By Small lorsque paraîtront ces lignes. La carte s'annonce rapide, bien conçue, et... mais suivons plutôt Dave dans sa conception de cette carte, qui devrait mettre les ST au niveau des TT, avec même des répercussions pour les heureux possesseurs d'un TT.

#### Les accélérateurs

Un beau jour, on voulut accélérer le ST avec une idée brillante on mit un 68000 cadencé à 16 MHz dans un ST. L'horloge du ST fut modifiée pour battre à cette fréquence, et l'on espéra obtenir une augmentation de vitesse d'exécution des programmes. Après tout, le temps d'accès des RAM du ST est assez bas pour supporter cette fréquence d'accès.

L'expérience échoua. Cette méthode ne donna qu'une accélération d'environ 10%, au lieu du doublement de vitesse que l'on aurait pu naïvement espérer. Que s'était-il passé ? Simple : conflits avec la vidéo.

Voyez-vous, un 68000 lit chacune de ses instructions en mémoire, puis les exécute. A 8 MHz, le 68000 et la vidéo interfèrent rarement, et font leurs opérations avec la mémoire de façon bien séparée. Mais ce n'est plus le cas à 16 MHz A cette fréquence, le CPU requiert plus souvent l'accès à la mémoire. Face à cette Situation, les circuits vidéo du ST forcent tranquillement le 68000 à accomplir un cycle d'attente (les fameux wait states) pour pouvoir accéder sans encombre à la mémoire. C'est toujours la vidéo qui est prioritaire. Car on ne peut pas ralentir le balayage du canon à électrons dans le tube TV. Pendant le bref instant où le canon balaie un pixel, on veut pouvoir l'alimenter en signal vidéo, car l'on ne veut pas voir apparaître de points noirs sur l'écran. Donc on fait attendre le 68000, qui le supporte très bien. Du coup, l'avantage d'une fréquence plus élevée est largement entamé, puisque l'on ne peut fournir au 68000 ses instructions assez vite. Il est impossible de couper la vidéo ou de rendre sa charge plus tolérable visa-vis de l'accès mémoire. Les circuits d'Atari ("custom chips") ne peuvent être modifiés, et il est extrêmement délicat d'intercepter leurs interconnexions pour altérer leur fonctionnement. Les circuits GLUE et MMU fournissent sans arrêt des octets vidéo au Shifter, et ils se moquent de la gêne qu'ils nous apportent.

C'est ce qu'on appelle le goulot d'étranglement de la mémoire vidéo. Pas le choix Si vous voulez voir quelque chose sur votre écran, vous devez abandonner 8 MHz de bande passante de mémoire à la vidéo sur les 16 qu'elle peut fournir. Restent 8 MHz pour le 68000. Vous vous rappelez certains ordinateurs 8 bits où il fallait couper l'affichage graphique pour pouvoir aller à la vitesse maximum ? C'était déjà pour le même motif.

#### Accélérateur avec antémémoire

Le stade suivant en matière d'accélérateurs arriva avec les antémémoires, également appelées caches. Une antémémoire repose sur un principe simple. Mais d'abord une convention : appelons

"RAM ST" la mémoire normale du ST, celle où voisinent la mémoire vidéo et les programmes d'un ST non trafiqué. Cela nous aidera à nous y retrouver lorsque nous introduirons une nouvelle sorte de mémoire. Une antémémoire fonctionne en réduisant le nombre d'accès à la RAM ST que doit faire le 68000 (accès au cours des quels il entre en conflit avec la vidéo). Elle est constituée d'une petite RAM statique (en général 16 ko) rapide. Ce qui signifie qu'elle est chère. Cela marche grâce au fait que la plupart des programmes tendent à accéder de manière répétée aux mêmes instructions.

Par exemple, supposons que nous programmions un effacement d'écran du ST. Cela consiste à écrire dans les 32000 octets de la RAM vidéo (elle-même n'étant qu'une simple zone de la RAM ST) l'octet représentant des pixels blancs. Cela donne à peu près

Compteur = 0

Adresse = Début de la RAM vidéo

**BOUCLE:** 

Mettre un blanc dans Adresse Incrémenter Compteur Incrémenter Adresse Si Compteur < 32000.

alors aller à BOUCLE Si ce programme est en RAM ST et que nous essayons de le faire tourner à 16 MHz, les conflits avec la vidéo vont nous engendrer quantité de cycles d'attente, bien que les instructions exécutées soient 32000 fois les mêmes. Mais une antémémoire permet de garder dans une RAM spéciale les valeurs que vous lisez ou écrivez dans la RAM ST, ainsi que leurs adresses. Bien sûr, Si vous avez 4 Mo de RAM ST et 16 ko d'antémémoire, vous ne pouvez pas espérer tout y caser. Il a donc fallu adopter une règle d'occupation des précieux octets de l'antémémoire.

On utilise habituellement la règle de l'accès le plus récent chaque fois que vous accédez à une nouvelle adresse, celle ayant été accédée le plus longtemps auparavant est éliminée et recouverte par la nouvelle valeur.

Reprenons donc le programme ci-dessus sur une machine 16 MHz dotée de RAM ST à 8 MHz et d'une antémémoire. A présent, lorsque nous lisons les instructions de la boucle la première fois, elles sont lues dans la RAM ST avec la lenteur imposée par celle-ci, et sont à la fois exécutées par le 68000 et stockées dans l'antémémoire, accompagnées d'une étiquette indiquant leur adresse. Lorsque nous exécutons la dernière instruction ("aller à BOUCLE"), le 68000 engendre une adresse que 'antémémoire détecte c'est une vieille connaissance, elle correspond à des octets qui sont dans sa RAM Du coup, c'est le contrôleur d'antémémoire qui répond et non la RAM ST. Ainsi, le 68000 n'a plus à attendre les octets de la RAM ST.

Les antémémoires peuvent être une bonne ou une mauvaise solution, cela dépend du programme que vous exécutez. Si celui-ci contient une boucle tenant intégralement dans sa RAM, celle-ci peut être remplie durant le premier passage dans la boucle, et les itérations suivantes en bénéficient. Si le programme folâtre de saut en branchement, l'antémémoire n'a jamais réellement l'occasion de remplir son rôle, car 6 ko sont vite remplis, et il faut faire de la place pour les adresses suivantes.

Mon expérience personnelle confirme ces explications. J'ai un clone de PC tournant à 8 MHz comme le ST. J'y ai monté un accélérateur doté d'un 80386 à 16 MHz et de 16 ko d'antémémoire. J'ai découvert en l'utilisant que certaines opérations s'en trouvent accélérées tandis que d'autres y sont insensibles. Les petites boucles sont propulsées à toute allure, tandis que les gros programmes n'en retirent rien. Les benchmarks disent que la machine est 17 fois plus rapide qu'avant. C'est ridicule, ce n'est pas l'impression en ressortant à l'usage. Toutefois, les benchmarks se trouvent être précisément de petites boucles rapides, sur lesquelles l'antémémoire marche bien. D'où le chiffre mirifique. J'ai utilisé également des clones à base de 386 à 16 MHz dotés d'une RAM rapide, sans cycle d'attente (pas besoin d'antémémoire, par conséquent). Eh bien, la machine fonçait dans tout ce qu'elle faisait!

Impossible de lire le répertoire craché par un DIR tant cela va vite rien à voir avec mon clone modifié.

#### Conception, pas sabotage

Une méthode, pour faire une carte 68030, est de mettre un 68030 et une petite antémémoire (16 ko) sur un circuit imprimé. Ainsi, vous aurez de bons résultats aux benchmarks (très important, en particulier pour la pub : même si, dans des applications réelles, vous n'allez pas huit fois plus vite que le vieux ST, un benchmark claironnant un gain de 800% est bon pour la pub). Mais cela ne me tentait guère : c'est que j'avais jadis acheté des accélérateurs à antémémoire pour diverses machines. J'ai même acheté un accélérateur pour ST, doté d'un 68000 à 16 MHz et de 16 ko d'antémémoire. Des benchmarks sans signification les créditaient d'une vitesse double de celle du ST. Mais pour ce que j'en faisais (développement de logiciels), l'accélération obtenue était vraiment minime. C'était un accélérateur fantôme tantôt il répondait, tantôt il disparaissait. En moyenne, les programmes que je tenais pour significatifs obtenaient 50% d'accélération, comme pour une horloge à 12 MHz. Or, des mégahertz, j'en avais acheté 16, moi! Refaire ce bricolage ne m'intéressait donc guère.

Du coup, je commençais à réfléchir aux moyens de bien faire les choses. Je passais beaucoup de temps à me demander comment éviter que les circuits vidéo ne drainent la bande passante de la RAM ST. Je n'en trouvais aucun. Pendant ce temps, je voyais l'industrie évoluer autour de moi tandis qu'apparaissaient des ordinateur dotés de processeurs plus rapides et de capacités mémoires plus grandes.

#### Notre propre RAM

A ce moment, une pensée me vint pourquoi ne pas tout simplement ajouter de la mémoire au ST ? Ajoutons-y une mémoire interdite d'accès à ces satanés circuits vidéo, de façon à ce qu'elle ne soit pas réduite à 8 MHz de bande passante et qu'elle puisse tourner aussi vite que possible. Tant qu'on y est, organisons cette mémoire sur une "largeur" supérieure à celle du ST. Sur le ST, quand vous accédez à une adresse mémoire, vous lisez un mot à la fois, soit 16 bits ou deux octets. Moi, je voulais que ma mémoire m'envoie des longs mots de 32 bits (4 octets). Accéder à 4 octets consécutifs ne prend alors plus qu'un seul cycle d'accès mémoire au lieu de deux, ce qui économise des cycles d'horloge et se répercute aussi sur les cycles de précharge et de rafraîchissement de la RAM dynamique [NdT : voir à ce propos l'article "DRAMS" paru dans ST Mag 54, page 76]. Quant au CPU, il fallait que ce soit un 68030, pour des raisons de vitesse à tout le moins. Le 68000 n'est pas disponible en version plus rapide que 16 MHz, et d'ailleurs, 16 MHz ne constituent plus la pointe de la technologie aujourd'hui (bien qu'une véritable architecture à 16 MHz soit très rapide). Le 68030 était disponible en versions 16,25,33, 40 et 50 MHz, ce qui me plaisait bien je voulais une accélération suffisante pour m'arracher les oreilles.

Après avoir pris la décision d'ajouter de la mémoire, il fallait en fixer le type. Nous sommes entrés dans une nouvelle ère, même par rapport à seulement deux ans auparavant : la RAM dynamique est bon marché, elle est disponible en paquet familial. La RAM statique est ruineuse et ne se vend que par petit pot. Nous choisîmes donc la RAM dynamique, malgré les problèmes qu'elle ajoutait pour la conception, car elle permettait de réduire le prix de vente de la carte.

Cette RAM rapide nous donnait plusieurs avantages. D'abord, un programme chargé dans cette RAM tournerait très vite. Les conflits mémoires seraient réduits à zéro, ce qui est, nous l'avons vu, très important. Et il travaillerait sur 4 octets à la fois. Il pourrait tirer parti du mode "turbo" du 68030, appelé mode "burst" [NDLR : ou "en rafales" en français], dans lequel le 68030 lit des séries de mots consécutifs bien plus vite qu'en mode normal. Utiliser ce mode impose une conception spéciale de la carte, mais c'était faisable. Philosophiquement enfin, les utilisateurs de ST ayant besoin de davantage de RAM me décidèrent! Nous avons déjà dépassé le stade où 4 Mo de RAM étaient un espace si vaste que l'on ne pouvait rêver de le remplir. Certains utilitaires, comme celui de Codehead, permettent d'installer en mémoire autant d'accessoires que vous voulez. Des "switchers" vous offrent de passer instantanément d'un programme déjà chargé en mémoire à un autre. Les sons numérisés sont vraiment à l'étroit (ce sont de vrais goinfres à RAM). Les programmes de PAO trouvent aisément l'usage de RAM supplémentaire, surtout ceux utilisant des images numérisées... Et en dressant ainsi la liste de ce dont les gens avaient besoin, je concluais que je devais fournir davantage de vitesse et de RAM.

Nous fixâmes donc la RAM à 8 Mo (eh oui !, deux fois plus que n'en offre le haut de la gamme ST !), sous forme de connecteurs standard SIMM. Comme beaucoup de gens peuvent trouver facilement des barrettes SIMM chez leurs revendeurs, nous n'avons pas voulu imposer les nôtres et ajouter notre marge. Nous pouvons toujours vous en fournir si vous le voulez, mais tous les magazines informatiques regorgent d'annonces pour des barrettes SIMM. Mais attention, ces 8 Mo s'ajoutent aux 4 Mo déjà installables dans le ST. Ils n'en prennent pas la place. Vous aboutissez donc à une machine à 68030 et à 12 Mo de RAM. Ce qui est une puissance très honorable. Aller au-delà de 8 Mo ne nous a pas semblé prudent. Tout d'abord, il y a des problèmes d'adressage mémoire. L'agencement des zones mémoires du ST devient délicat vers la barrière des 14 Mo. Ensuite, nous ne voulions pas solliciter l'alimentation du ST trop fortement. Enfin, 12 Mo semblent sacrément confortables pour la quasi-totalité des gens! Mon Mac II en a 8 et je suis rarement à court de mémoire. Si nous tombons sur des forcenés de la mémoire voulant à tout prix utiliser des SIMM de 4 Mo, une modification mineure de la carte devrait leur donner satisfaction, à condition qu'ils écrivent des programmes vraiment intéressants! Cela leur donnerait 32 + 4 Mo, ce qui devrait suffire à tout le monde. Sinon, qu'ils aillent acheter un Cray comme Apple.

Cette RAM rapide, nous l'avons vu, est spéciale. Les circuits vidéo n'y ont pas accès. Rien ne vient interférer avec le 68030. D'où le nom de fastRAM que nous lui avons donné. Un programme chargé en fastRAM résoudra vos problèmes avant que vous ne les ayez exposés, tant il s'exécutera vite! Les 4 Mo de la RAM ST demeureront parfaitement utilisables. Mais bien sûr, ils sont lents, la vidéo les limitant à 8 MHz. Vous ne devriez pas les utiliser à moins d'en avoir explicitement besoin - par exemple, si un programme a été écrit par quelqu'un supposant que la RAM du ST ne pourrait jamais dépasser 4 Mo, chose rare. Tous les accès disque et vidéo doivent se faire au travers de la RAM ST. Mais cela ne présente guère de problème, même pour moi qui doit écrire le logiciel faisant marcher tout ceci! En fait, la vidéo est toujours attachée directement à la RAM ST, et si vous essayez d'écrire un bloc de données du disque vers la fastRAM, le bloc sera d'abord lu du disque et écrit en RAM ST, puis transféré (et sacrément vite!) en fastRAM. L'écriture sur disque utilise le mécanisme inverse le bloc est transféré de la fastRAM vers la RAM ST puis écrit sur disque. La vitesse du 68030 est telle qu'en dépit de ce transfert, les performances du disque n'en souffrent pas. Je ne pourrais le supporter, d'ailleurs. Sachez que je suis un fanatique de la rapidité des accès disques depuis que j'ai écrit le formateur Twister.

#### Connecteur d'extension

La carte SST est dotée d'un connecteur d'extension de bonne qualité pour permettre d'y brancher une carte supplémentaire. Car je ne suis pas satisfait d'avoir a me contenter de la vidéo du ST.

J'aimerais (j'aimerais même beaucoup) avoir une meilleure résolution, du genre Super-VGA. Avec tous les efforts de R&D, dans le monde IBM PC, contribuant à réduire les coûts, il semble scandaleux de ne pas tirer profit de ces circuits vidéo bon marché et d'excellente qualité...

Je suis par ailleurs convaincu de l'utilité des connecteurs d'extension, et nous avons l'intention de publier les spécifications de notre connecteur pour les gens voulant l'utiliser. Il ne s'agit pas d'un connecteur "réduit" ou "basse vitesse", mais d'un accès direct aux bus de données, d'adresse et de contrôle du 68030, plus quelques autres signaux dont vous pourriez avoir besoin. Si vous développez, et Si vous voulez utiliser des processeurs puissants, vous allez pouvoir vous amuser avec la SST.

#### Compatibilité avec le ST

La SST originale fut bien sûr conçue pour le Méga ST. Nous devions bien commencer quelque part, et tous les membres de l'équipe de développement avaient un Méga I Toutefois, aux USA, seul un faible pourcentage d'utilisateurs de ST possède un Méga, et nous n'avons pas l'intention de renoncer à un marché aux USA ou en Europe. Nous pouvons donc vous assurer que des versions pour 520 et

1040 ST sont en chantier. Mais je dois préciser qu'elles seront délicates, car franchement, 8 Mo de SIMM prennent de la place et dégagent de la chaleur (ni le 520, ni le 1040 n'ont de ventilateur, et les pannes d'origine thermique sont légion en informatique, tous les vieux routards le savent) (NdT :exact, défaillance ou absence de ventilateur m'ont déjà mis en panne un disque dur de 1040 et une station Unix).

Une technique, dont on m'a parlé plusieurs fois, consiste à placer le 520 dans le boîtier d'un clone PC, en laissant le dessus du ST ouvert, et à introduire ensuite la SST avec des supports 64 broches en guise d'écarteurs. Cela marche et laisse la chaleur se dissiper. Digression pour bidouilleurs vous pouvez, si vous le souhaitez, utiliser un clavier externe de PC en le connectant à l'un des divers adaptateurs du marché. Les boîtiers de clones permettent de plus un montage aisé des disques durs et d'alimentations très répandues. Une "alim" de 65 watts pour clone suffit amplement. N'exagérez pas sa puissance, sinon votre "alim" ne sera pas assez chargée pour que sa régulation à découpage fonctionne correctement! Ma collection de disques durs tient actuellement dans deux boîtiers de clones. Fin de la digression.

Le 1040 est plus délicat encore. Il y a un certain nombre d'agencements de cartes mères en circulation pour les 1040, dont une avec le 68000 directement sous le clavier! Impossible pour celle-là d'y loger la SST. Sous réserve, nous devrions logiquement proposer un petit câble allant de la carte SST au socle du 68000, de façon à pouvoir installer la carte même dans ces cas.

#### Compatibilité logicielle

Pour être franc, nous espérons atteindre le même niveau de compatibilité que le TT, qui est d'environ 80%. Mais je dois avouer que je garde une carte dans la manche. Durant le développement du Spectre, j'ai appris quelques trucs particulièrement vicieux, pour corriger des logiciels Mac qui plantaient de façon vraiment bizarre, et il se trouve que la cause principale de plantage sur le TT se rapproche beaucoup d'une des bogues du Spectre que j'ai corrigée. je travaille actuellement sur un programme réparant les erreurs causées par certains dysfonctionnements logiciels, et qui, je le souhaite, permettra à beaucoup de programmes qui ne tournaient pas sur le SST de s'exécuter proprement. Si cela marche aussi bien que je l'espère, je pourrais le livrer avec le SST et en sortir une version pour le TT. Ce programme antiplantage est issu d'une de ces idées semblant parfaitement évidente lorsqu'on y songe et où il est clair que la solution sera efficace. Je me réfère à mes cinq ans d'expérience à faire fonctionner des programmes Mac sur du matériel non Apple.

J'espère bien que l'antiplantage marchera également sur le TT. Il y a déjà beaucoup trop de programmes pour ST seulement, ne fonctionnant pas sur le TT, et qui ne seront jamais mis à jour, et si je peux les corriger par un programme a mettre dans le dossier AUTO, je le ferai. L'expérience acquise sur le Mac, après l'effort ardu de correction de certains logiciels, fait que le concept est parfaitement clair dans ma tête et je n'entrevois aucun problème. Le 68000 et le 68030 sont en effet très compatibles, et la plupart des programmes ne plantent sur TT que pour des raisons ridiculement minimes. Il ne restera plus alors que les programmes, qui suivent Si peu les règles élémentaires qu'ils plantent pour des raisons plus profondes, et continueront, hélas!, à planter.

Une petite recommandation en passant, qui s'applique à l'achat d'un accélérateur quel qu'il soit. Beaucoup de logiciels sont protégés contre la copie par des techniques basées sur une cadence fixe et connue de l'horloge du microprocesseur, et défaillants si le processeur devient plus rapide. De même, il est difficile de charger un quelconque utilitaire antiplantage depuis une disquette autoboot (Hmmm... Peut-être devrais-je mettre ce programme dans les ROMs de la SST ?). Donc, prudence, monter un accélérateur non débrayant peut signifier devoir renoncer à vos jeux et logiciels protégés.

#### Installation

Installer la SST est simple. Vous devez prendre les précautions antistatiques habituelles (pas de chaussures, pas de vêtements synthétiques, pas de tapis en laine, pas de copine aux longs cheveux longuement brossés...). Enlevez votre 68000 de son socle. Pour cela, coupez les broches à l'aide d'une pince pointue spéciale (celle de Tandy est parfaite, en limant un peu ses bords extérieurs, on arrive à la glisser partout!), puis dessoudez les morceaux de broches à l'aide d'une pompe à dessouder et d'un fer à souder.

Ensuite, inspectez la carte du ST à l'endroit du 68000 et vérifiez l'absence de bavures de soudure faisant court-circuit et de pistes endommagées. Employez une loupe et un bon éclairage. Cette phase est cruciale.

Puis insérez dans la carte du ST le support à 64 broches fourni. Nous vous donnons même un 68000 de rechange pour le cas où vous voudriez revenir à un ST normal (ou le revendre sans la SST). Soudez le support. Vérifiez comme auparavant.

Mettez le nombre de barrettes SIMM que vous désirez dans les emplacements fastRAM de la SST. La largeur du bus du SST est de 32 bits, vous devez donc insérer 4 barrettes à la fois (0, 4 ou 8). Je vous recommande au moins 4 barrettes, toute la carte est conçue pour tirer parti de cette fastRAM.

Bien que l'on puisse constater un gain de performance dû au seul 68030 sans fastRAM, la carte n'atteint ses performances réelles qu'avec ses barrettes.

Enfin, mettez la carte dans le connecteur 64 broches et branchez le câble d'alimentation. Vérifiez que vous n'avez pas inséré en sautant une broche, et mettez sous tension.

#### Et l'antémémoire?

Non, pas d'antémémoire spécifique dans la SST. Nous avons longuement réfléchi. Nous avons conclu que disposer de 4 ou 8 Mo de fastRAM était mieux que d'avoir une petite antémémoire de vitesse équivalente. J'avoue ne pas tout comprendre quant aux chronogrammes des DRAM, mais George m'assure qu'en mode "burst", notre montage surclasse même une antémémoire haute vitesse. Mais rassurez-vous, il y a quand même une antémémoire dans la SST à l'intérieur du 68030 se trouvent 256 octets dédiés à cet usage pour chacun des bus de données et d'instructions. Donc, Si vous exécutez une boucle serrée tenant dans ces 256 octets, votre 68030 aura la vitesse foudroyante typique des antémémoire. Et en dehors de ces boucles, le 68030 ira chercher ses instructions en mode "burst", assurant la vitesse de débit maximale entre le processeur et la RAM.

#### Compatibilité?

A notre grande surprise, bien après que les "specs" de la SST aient été fixées et que George ait commencé sa conception matérielle, noue entendîmes parler du TT. Le TT et la SST ont en commun quelques caractéristiques très intéressantes I D'abord, tous deux emploient un 68030. Le TT est doté d'un 68030 à 32 MHz, la SST utilise un processeur à 33 MHz (si vous le commandez, nous offrons aussi la version 16 MHz et envisageons les versions 25 et 40 MHz). Un malheureux mégahertz de plus ou de moins n'importera guère. Mais le TT utilise aussi de la fastRAM! Voire de la fastRAM à l'intérieur du TT, quand il apparut enfin, me rassura. Et - c'était quasiment inévitable - il y avait un cavalier pour sélectionner le mode "burst" I Ce qui m'avait rassuré et conforté, c'était de voir que les ingénieurs d'Atari s'étaient penchés sur les mêmes problèmes et avaient apporté des solutions identiques.

Le TT n'utilise d'ailleurs pas d'antémémoire autre que celle intégrée au 68030, laquelle, à en juger par ses performances, est largement suffisante. Mais après nous être penchés sur le TT en détail, nous avons découvert des points communs plus étonnants encore. Je le répète, la SST était en chantier quand le TT sortit. Or, les adresses de la fastRAM étaient les mêmes! Nous avions choisi notre adresse de début de la fastRAM plus ou moins au hasard, cela nous apparaissait comme un bon endroit pour

mettre un décodage d'adresse. Et là, hasard irréel, c'était la même que celle du TT! Cela signifie que les logiciels développés spécifiquement pour le TT marcheront sur la SST, et la réciproque est très probablement vraie. Nos tests préliminaires nous donnent bon espoir. Beaucoup de choses fonctionnent déjà, et nous pouvons facilement changer les quelques détails qui manquent encore pour rendre la compatibilité totale.

#### Le test

Enfin, le Grand Jour arriva.

Il y a un programme très populaire pour le Mac appelé Speedometer, mesurant les performances de la machine. C'est encore un de ces benchmarks stupides, me direz-vous, mais celui-là est très connu. Bon. Nous avons donc chargé Spectre 3.1 (pas 3.0,3.11) dans la SST. La version 3.1 contient des optimisations spécifiques au TT et à la SST. Par exemple, il s'arrange pour travailler en fastRAM, pas en RAM ST. C'est assez délicat et exige de remapper la mémoire à l'aide de la MMU du 68030. Nous fîmes ensuite tourner Speedometer sous l'émulateur Spectre. Lequel nous annonça que nous étions plus rapides que le Mac Ici. Or le Mac LCII est le second Mac le plus rapide de chez Apple, c'est une machine à 25 MHz avec toutes sortes de gadgets que seul surpasse le très coûteux Mac Ilfx à 40 MHz. A ce stade, j'ai bien envie de déballer mon sac à clichés les fenêtres s'ouvrant et se fermant en un éclair, les menus apparaissant instantanément, les documents reformâtes en un clin d'œil, etc. Bref, ça fonçait.

Certes, le mode émulation Mac nous donne un avantage, car les Mac à base de 68020 sont sortis depuis suffisamment de temps, pour que tous les logiciels Mac tournent en utilisant le jeu d'instructions et les ressources du 68030, compatibilité ascendante oblige. Mais même en mode ST (pourtant limité; sauf exception, au jeu d'instructions du 68000), ce que j'ai vu m'a impressionné. En particulier, le changement de taille des fenêtres se fait instantanément. On ne voit plus l'écran se redessiner, ce qui montre que cette opération est faite le temps d'un balayage vertical, soit bien plus vite qu'avant.

Encore quelques petits réglages à faire - que je ne révélerai pas ici, car mes concurrents lisent mes articles - pour faire encore gagner un peu de vitesse à la SST, et je la lance dans la course.

Traduction et adaptation: Password

### Je me souviens...

Qui a dit que les bidouilleurs étaient insensibles à tout ce qui ne touchait pas leur domaine technique? Dave Small nous prouve ce mois-ci le contraire, en nous parlant d'un sujet extrêmement délicat. Les grandes précautions verbales dont Dave s'entoure sont indispensables aux USA, où le problème abordé est politique. D'autant plus qu'il touche aux aspects sociaux des entreprises informatiques américaines, dont Dave nous dévoile des facettes méconnues. Le sujet est, hélas!, d'actualité en France...

#### Pas de technique aujourd'hui

Il y a toutes sortes de sujets techniques dont j'aurais pu vous entretenir ce mois-ci. Nous avons la carte accélératrice SST, les réseaux, etc. Je pourrais jeter dans la conversation des wagons de notions techniques passionnantes, comme vitesse d'horloge, temps de montée des signaux, comptage d'événement déclenché par fronts montants, unité de gestion de mémoire par pagination, et mémoire virtuelle. Youpi!

L'ennui, c'est que je ne me sens vraiment pas d'attaque pour aborder ces sujets pour l'instant. Je crois que c'est dû au fait qu'après avoir vu le jeu d'arcade Battlezone, j'ai parlé à un ami de Cupertino qui en est l'auteur, ce qui m'a amené à écrire ceci (vous allez comprendre pourquoi). Cela fait longtemps que je pense à l'article que vous lisez, mais je n'avais jamais encore eu le courage de l'écrire. D'une part, il est dur d'écrire sur une question vous affectant personnellement. Et d'autre part, le sujet que je compte aborder a le don de déranger les gens, et je n'ai nulle envie de recevoir des lettres désagréables. Beaucoup de gens ont des sentiments très marqués à ce propos.

Bon, je me jette à l'eau. Il s'agit de choses très personnelles pour moi. Je me souviens de Nick. Je l'avais rencontré sur un serveur télématique. Nick m'avait dégoté un boulot chez XXX (un grand constructeur informatique - nom de la firme non dévoilé pour raisons juridiques. Certaines personnes se sont confiées à moi sous la condition expresse que leur nom ou celui de leur compagnie ne serait jamais dévoilé). Nick était gay et n'en faisait pas une histoire. Il vivait avec mon chef de cette époque. Eh oui, ils étaient gays tous deux ! Aucun ne prenait la peine de débattre du sujet, et cela ne les dérangeait aucunement. Ils vivaient à Saratoga dans la Silicon Valley, dans un beau pavillon, et travaillaient sur un projet fascinant. Dans la Silicon Valley, la plupart des informaticiens se fichent de ce que vous soyez gay ou non. A l'occasion, vous rencontrez un fanatique ou un gars du genre prêcheur, mais la plupart des gens s'en moquent et s'occupent de leurs propres affaires. Nick devint un expert autodidacte en optimisation des gros ordinateurs XXX. Cela consistait à allouer un disgue plus rapide pour les opérations accédant au disque de manière intensive, ou encore à défragmenter un fichier très utilisé, etc. Il accrût ainsi les performances d'une machine de 200 % en un mois de travail. Il était sacrément bon dans sa partie. J'avais avec Nick le même type de relations gu'avec mes autres collègues à ce centre. Il avait ses espoirs et ses rêves, j'avais les miens, et nous en discutions parfois au cours du déjeuner. Nous étions tous deux fanas du jeu d'arcade "Battlezone" qui venait juste de sortir, et lorsque nous découvrîmes une façon d'échapper aux smart bombs à tête chercheuse, cela nous réjouit pour la semaine entière.

En 1980, j'étais très occupé à essayer de convaincre Sandy de m'épouser, et n'étais donc pas intéressé par d'autres relations. Nick était un ami, quoi. Vous comprendrez donc qu'il m'est dur de vous dire que Nick est récemment décédé du sida. Je ne sais même pas si mon ancien chef est vivant ou mort, et je ne puis trouver le courage de prendre le téléphone et d'appeler pour tenter de le savoir. Le simple fait d'écrire ceci me noue la gorge, car Nick était un ami, et tout ce qu'il a jamais été a disparu. je crois que je ne jouerai plus à Battlezone pour un bout de temps, trop de souvenirs s'y associent.

#### L'année 1980

En 1980, la communauté médicale commençait à peine à se demander ce qui pouvait bien donner tous ces sarcomes (cancers de la peau) à des homosexuels... Eh bien, nous savons à présent! Nous ne savons même que trop bien. Quand l'on y songe, nous en sommes au stade où beaucoup de gens connaissent quelqu'un que les tests ont révélé être séropositif. Mon frère, qui est médecin, avait un ami proche depuis le collège, il était élève officier [NdT le corps des élèves officiers de réserve (ROTC) permet à de jeunes Américains de faire leurs études tout en étant pris en charge par l'armée, d'où le spectacle surprenant de ces étudiants en rangers et treillis, en côtoyant d'autres en baskets et jeans]. Il avait des 'A' partout et faisait partie des tireurs d'élite. Il est mort récemment. Mon frère voit tant de patients atteints du sida qu'il commence à perdre son détachement impersonnel et médical à ce sujet, et déteste littéralement cette maladie. Autre exemple, mon père a perdu il y a peu un collègue.

En 1980, j'ai discuté avec un gars du service du personnel à la compagnie XXX. Je quittais la compagnie pour épouser Sandy, ce qui me conduisait à changer de ville. Comme ce gars et moi nous connaissions depuis quelque temps, nous nous offrîmes un pot d'adieu autour de guelques bières.

Durant la discussion, le sujet de nos collègues gays fut abordé. Il fut très franc : c'était plus ou moins un standard de fait dans la compagnie de préférer embaucher des gays si possible. Oh pas pour remplir les quotas de discrimination positive. [NdT loi américaine très controversée obligeant les employeurs à embaucher des quotas de membres de diverses minorités officiellement reconnues, à la grande fureur de ceux s'estimant membres de minorités injustement méconnues, et au grand dam des employeurs parfois obligés de remplir leurs quotas en catastrophe avec le tout-venant.] Non, simplement parce que la compagnie estimait que les gays fournissaient plus de travail à moindre coût. Écoutez, me dit-il, vous (il pointa son doigt vers moi), vous allez vous marier. Vous allez avoir une épouse, une famille, toute une série de dépenses d'assurances maladies, surtout avec des gosses. [NdT: aux USA, les employeurs ne paient pas à la sécurité sociale une part fixe dépendant du salaire comme en France, mais des assurances maladies pour chaque membre de la famille. Les employeurs paient aussi une part des dépenses médicales de leurs ex-employés retraités. Coût pour IBM, par exemple, au premier trimestre 1991: 2,3 milliards de dollars!] Vous allez aussi avoir des horaires. Il vous faudra être à la maison à cinq heures et demie le soir, et votre femme voudra vous voir pendant que vous serez à la maison. Impossible de faire des nuits blanches. Au contraire, si vous êtes célibataire et sans attaches, vous pouvez faire des heures supplémentaires. Et comme vous êtes "au forfait", vous n'êtes pas pavé pour ces heures vous êtes rémunéré sur des objectifs, pas à l'heure. Et surtout, vous n'avez pas de famille, pas d'assurance maladie à payer, sauf pour vous-même, et que diable, vous êtes jeune et sain (Les discussions actuelles de couverture médicale du sida pour les proches n'existaient pas en 1980.) C'est excellent pour la compagnie, dit-il. C'est pour ça que nous les embauchons. La compagnie XXX, en fait, avait un programme de recrutement de gays si actif, que 7 des 9 personnes de mon groupe de travail l'étaient. La discrétion qui régnait était telle, que je l'ai appris seulement le jour où i'ai guitté la boîte. J'en entends qui ricanent. Soyons adultes: c'était un environnement de travail tout ce qu'il y a de plus normal, avec des gens ordinaires. Je m'entendais avec certains, pas avec d'autres ; il y avait les politiques de la compagnie, les mémos internes, les bilans... Bref, un travail parmi tous ceux que j'ai eu depuis. Je n'étais ni harassé ni mis à l'écart d'une clique quelconque. J'ai également travaillé dans d'autres compagnies depuis lors, où des collègues étaient gays, et cela n'importait aucunement. Dans mon optique, soit une personne sait programmer et faire le boulot, soit elle ne sait pas. Rien d'autre n'importe réellement, vu qu'il y a si peu de programmeurs vraiment compétents que lorsque vous en dégotez un, vous vous moquez du reste.

#### Épitaphes magnétiques

Mais, comme vous le savez, les temps ont changé depuis 1980. La compagnie XXX est à présent submergée par les coûts vertigineux des hospitalisations d'employés atteints du sida [NdT: dans le cas des grosses compagnies, ce sont les mutuelles de la société qui règlent ces coûts d'assurance maladie]. J'en ai parlé à un manager de haut niveau de la compagnie XXX que je connais depuis des années. Il n'a pas voulu mentionner de chiffres, à ceci près qu'il a un peu pâli, mais selon lui, la compagnie avoue que sa politique de recrutement des années 70 et 80 engendre à présent des coûts très lourds en assurance maladie, et que l'embauche d'une personne atteinte du sida est ces temps-ci formellement proscrite. Il m'a aussi dit que le projet sur lequel Nick et mon ancien chef travaillaient était annulé du fait du décès de Nick.

Je ne citerai pas de nom, mais si vous avez possédé un Atari 8 bits, et utilisé quelques-uns de ses logiciels les plus populaires, vous avez sans doute fait tourner du code auquel mon groupe de travail a participé. Et si vous avez utilisé certains des utilitaires les plus puissants pour ces machines, vous avez sans aucun doute fait tourner des programmes écrits par des gens à présent morts. Des gens que je connaissais déjà en 1982, lorsque j'écrivais pour le magazine Creative Computing (tenez, je me rappelle même le temps où l'étage supérieur entier du 1196 Borregas Avenue constituait le bureau de Chris Crawford. Maintenant, il constitue la moitié de tout l'immeuble d'Atari). Je m'arrête un instant pour vous dire que cela me fait une impression très étrange, et même cauchemardesque, que la seule épitaphe de beaucoup de ces gens soit les quelques impulsions magnétiques d'une disquette composant leur programme. Tout le reste a disparu.

#### **Discrétion**

Dans l'informatique, beaucoup de gens ont un comportement, heu !..., assez peu sociable, et ne s'intéressent guère aux relations humaines pour commencer. A leur contact, vous ne percevez pas vraiment leur orientation sexuelle parce qu'ils présentent tant d'autres particularités.

Soyons concrets : quand vous parlez a un programmeur d'un top niveau mondial, ayant son ordinateur, son lit et son vélo d'appartement dans la même pièce, et qui n'a pas pris de douche depuis cinq jours, vous êtes déjà assez occupé à rester dos au vent (ça m'est arrivé plus d'une fois). Beaucoup de ces gens sont gays ; et parce que l'opinion dominante est "cela ne me regarde pas", on n'en parle jamais (je ne dis d'ailleurs pas qu'il faille en parler).

Incidemment, beaucoup de journalistes de la presse informatique [NdT : aux USA, et cela n'implique nullement Pressimage] partagent cette préférence sexuelle, et cela influence leur façon d'écrire. Quand je lis un reportage sur une personne que je sais être gay, et que certaines choses sont mentionnées, cela change de signification... Il faut alors lire entre les lignes, car dans ce milieu, les gays ont appris à rester réservés. Par exemple, supposons que dans une interview imaginaire, quelqu'un dise que "ce nouvel ordinateur a été conçu par un groupe qui était d'une grande cohésion de style, car il n y avait pas de place pour les conflits personnels". Vous pouvez traduire cela de vous-même. Vous ne me croiriez pas si je vous disais que c'est un cas réel, n'est-ce pas ? Eh bien, je crains de vous décevoir, car c'est vrai, j'en ai la certitude absolue. Notez que j'ai été extrêmement vague car je n'aime guère les visites d'huissiers. Et, de plus, cela constituerait une intrusion dans la vie privée que de nommer la compagnie, les gens, l'époque, le code et le processeur, mais je le sais.

Ou, tenez, à propos d'Apple. J'ai discuté ces dernières années avec de nombreuses personnes de cette firme. Certains d'entre eux proviennent même d'Atari (durant l'Époque Noire de 1984, Apple avait gelé ses embauches, mais avait levé ce gel le temps de récupérer des gens d'Atari qui étaient licenciés en masse). D'autres au contraire se sont convertis au ST. Avec tous les gens d'Apple, j'ai toujours entendu la même chose sur ce sujet. C'est une compagnie très fermée, soit vous êtes dans le clan, soit vous restez en dehors, point final. D'abord, ils commencent par discuter avec vous, par essayer de vous connaître, et s'assurer que vous n'êtes pas du genre à les sermonner ou à être choqué par la vie privée d'autrui. En général, ils glissent dans la conversation qu'untel est gay, et regardent si cela vous choque. Si vous ne sourcillez pas, il n'y a plus de problème. Après quoi, ils vous confient qu'Apple supporte des coûts de soins médicaux très, très élevés pour certains de ses employés, et ils vous ouvrent peu à peu les yeux vis-à-vis de ceux-ci.

Un ami, en lequel j'ai confiance, m'a dit qu'Apple était même l'une des rares compagnies qui pouvait envisager de vous embaucher si vous étiez séropositif, ou participerait à vos frais médicaux si vous viviez avec quelqu'un qui l'était. Cet ami trouvait stupéfiant qu'Apple puisse toujours se le permettre en ces temps de crise quant à moi, je sais où est passé un peu du prix des deux Mac IIfx de Gadgets By Small (et je trouve que c'est tant mieux). Mais le débat est toujours de personne à personne. Il ne fait pas les gros titres, parce que la première chose qui se passe, lorsque ce Sujet est débattu en public, est que les gens commencent à s'écrier : "Bien fait, c'est périr par où l'on pèche", ou bien "On ne s'en occupe pas assez". Le débat fait rage ailleurs et je ne compte pas y entrer.

Franchement, je me moque des hauts débats philosophiques et des grands discours, et je ne pense pas que mes sentiments affecteront les recherches en cours. Des amis à moi sont en train de mourir, c'est tout.

#### Douche froide, déprime et drogues douteuses

A l'expo MacWorld, il y a quelque temps, j ai assiste à la démo d'Hypercard donnée par Bill Atkinson, son auteur. Après avoir montré les fantastiques possibilités de ce logiciel et son orientation objet, la dernière chose qu'il montra fut la pile "Sida" [NdT : les applications écrites en Hypercard sont représentées sous forme de piles de cartes], affichant le nombre de gens morts du sida, avec mises à jour toutes les minutes. Ce fut une douche froide, qui mit fin abruptement à ce qui fut par ailleurs une présentation enthousiaste, devant une audience en délire qui admirait un authentique hérosprogrammeur. Mais comme le dit un de mes amis de chez Apple : "Bill connaît des gens qui sont mourants. Ce sont ses amis depuis des années. Quelle démo serait plus appropriée à ses yeux ?" Et je suis bien obligé d'être d'accord. Simplement, je crois que je n'aurais pas eu le cran de faire ce qu'a fait Bill.

Je ne voudrais pas vous déprimer, mais Si vous avez utilisé un Mac ou un Spectre GCR sur votre ST, vous avez exécuté des programmes écrits par des gens qui sont morts du sida.

En Californie, on en parle même dans les journaux. Ailleurs, le débat est bien moins apparent, sauf dans les quelques communautés ayant un fort pourcentage de personnes gays. Sur les serveurs télématiques, il y a souvent des conférences fermées secrètes (pour éviter les perturbateurs), car parfois les gens se sentent seuls, désespérés, et ont besoin de parler. Si vous vous liez d'amitié avec ces personnes, et qu'elles vous font réellement confiance, vous pouvez être admis dans ces conférences télématiques. Ce fut mon cas. La conversation peut devenir effroyablement déprimante. Que pouvez-vous dire à quelqu'un dont le meilleur ami vient de mourir ? Que pouvez-vous dire à quelqu'un ayant découvert qu'il est séropositif, donc bientôt malade, donc bientôt mort ? Je n'ai jamais été capable de répondre, je ne saurais que dire. Par exemple, je suis incapable d'imaginer de perdre ma femme Sandy, lentement, jour après jour, douloureusement. Et que dire de l'énorme ressentiment vis-à-vis des lenteurs de la FDA ? [NdT : la Food and Drug Administration délivre les équivalents américains de nos Autorisations de Mise sur le Marché aux nouveaux médicaments]. Je comprends un peu mieux ce problème grâce à mon frère médecin la FDA doit s'entourer de précautions draconiennes. Mais le nombre de gens en train de mourir du sida rend morale l'expérimentation humaine des médicaments, et l'augmentation du taux de mortalité indique que l'on ferait mieux de se dépêcher. Les jusqu'au-boutistes parmi les participants à ces conférences télématiques discutent de la façon d'obtenir de nouvelles drogues expérimentales venant de Chine (« Composé Q «) ou de Tijuana, au Mexique, ou encore vous mettent en rapport, si vous le souhaitez, avec des gens qui fabriquent euxmêmes d'autres composés (certains chimistes marrons sont excellents), et vous révèlent tout sur l'AZT (qui est pour l'instant ]e seul remède agréé, avec ses avantages et ses inconvénients). S'il est un sujet de discussion bien adapté aux conférences informatisées, c'est bien celui-là. Les gens devant leurs consoles ont besoin de parler. Je souhaiterais voir ce débat abordé dans des conférences ouvertes. mais les préjugés de notre société les en empêchent. Parfois, quelqu'un se connecte sur une conférence, avoue qu'il est gay, et récolte des avanies pour sa peine. Alors qu'une conférence fermée est idéale pour permettre à chaque interlocuteur isolé de déshumaniser l'autre.

#### La vraie raison des retards?

Toujours est-il que vous devriez essayer de deviner le nombre de projets, matériels ou logiciels, étant annulés ou retardés parce que les gens ne sont pas là pour y travailler, surtout dans la Silicon Valley. Mon ami qui est manager à XXX affirme tout bonnement que plusieurs gros projets Ont dû être abandonnés, réduits ou étalés sur plusieurs années - et retarder un projet dans cette industrie allant si vite est souvent signer l'arrêt de mort de celui-ci. De plus, les coûts d'assurance maladie accaparent de grosses sommes. On aboutit donc à manquer à la fois de gens qualifiés et de fonds.

Maintenant, repensez à tout ce que vous avez lu dans la presse informatique, concernant des projets retardés ou annulés, particulièrement ceux mobilisant beaucoup de gens et d'argent. Lisez entre les lignes! Les compagnies sont impassibles, mais je me demande dans quelle mesure le sida a contribué à ces retards? Je ne pense pas qu'il faille se demander "si", mais bien "dans quelle mesure". Et en dépit des précautions des gestionnaires, une compagnie ne vit en général que grâce à quelques personnes clés. Quand ils s'en vont, la compagnie dépérit. Idem bien sûr s'ils meurent.

Si vous étiez déjà un passionné d'informatique en 1980, repensez aux noms des développeurs de jeux qui marchaient bien à l'époque. Et demandez-vous où sont ces gens à présent. Attention, la réponse peut vous hanter (elle me hante bien, moi). Certains noms se rencontrent toujours (je suis toujours là). D'autres ont changé de secteur : Russ Wetmore est passé chez Apple après avoir écrit "Preppies!" où il a développé le cdev [NdT : programme résidant du Mac] Kolor, plus tout un tas de choses dont il ne peut pas parler... John Harris, connu pour son JawBreakers, s'est mis au vert et élève des chevaux, mais il apprécie toujours les Atari 8 bits... D'autres personnes, elles, ont disparu à jamais. Des personnes que j'avais rencontrées à des salons informatiques, avec lesquelles j'avais dîné, des gens avec lesquels j'avais débattu des mérites comparés de Mac/65 et de ASMIEDIT. Des gens qui m'avaient offert leurs planchers pour y dormir durant les salons informatiques. Je ne citerais aucun nom, mais cela me semble toujours irréel de voir une publicité avec une liste de logiciels 8 bits, et d'y voir un programme écrit par quelqu'un qui est mort.

#### Conclusion

Je voudrais bien avoir une conclusion à vous livrer. Mais je n'en ai aucune. Je constate que l'industrie informatique américaine compte une forte proportion de gens gays, et que cette maladie les tue. Je laisse à d'autres le soin de critiquer les efforts de recherche anti-sida découvrir quelque chose qui pourrait éradiquer ce virus serait écrire une nouvelle page de la médecine, pas une découverte mineure, et les médecins sont aussi déprimés que vous ou moi lorsqu'ils ont un patient mourant qu'ils ne peuvent aider. (Ils le sont même davantage. Songez que le progrès a accoutumé les médecins à l'idée de pouvoir guérir leurs patients.)

En définitive, peu importe l'opinion de chacun sur les homosexuels ou sur le sida. Le fait est que notre univers informatique s'en trouve affecté. Les choses ne vont pas aussi vite qu'elles le pourraient, et certains de nos meilleurs talents ont déjà disparu. D'autres vont y passer. Certains de ces gens apportent une contribution inestimable à notre industrie, et ils nous manqueront cruellement. De plus, nos participations aux dépenses de santé s'alourdissent. Et cela empirera bien avant qu'il y ait seulement des espoirs d'amélioration. Ne croyez pas lire ici quelque blâme à l'encontre des gens atteints du sida.

Je déteste cette maladie, pas ses victimes. De même, je ne blâme aucunement les gays.

Voilà ce que j'avais à dire. Cet article est fini, vous pouvez tourner la page. D'autres ne le peuvent pas. En comparaison, cela rend enviable même des situations tragiques comme revendeur d'Atari aux États-Unis. Mais quand je repense à Nick, à Levon, à Gary, à Jill, je ne peux me contraindre à tirer un trait. Je me rappellerai d'eux, et des autres qui les suivront sans nul doute dans l'avenir. Je me

souviendrai, c'est tout ce que je peux faire.

Je ne suis pas satisfait de cet article. Comme les autres, je l'ai relu et repris plusieurs fois avant de le livrer, et je n'en suis toujours pas satisfait. Je crois que je sais pourquoi quand je conçois un article, il y a un début, un chemine-ment et une fin. Or, cet article-ci n'a pas de fin. Toutefois, j'aimerais préciser que c'était un des articles les plus durs à écrire que j'ai connu, car les souvenirs sont pénibles. Voyezvous, quand mon copain de Cupertino m'a appelé il y a quelques jours, j'ai appris la disparition d'un autre ami, que j'avais revu en octobre 90.

Traduction: Password (joignable sur 3615 ST Mag en BAL PASSWORD90)

# Aidez quelqu'un d'autre

Dave Small nous parle ce mois-ci de quelle manière il a bénéficié de l'entraide, ce qu'elle signifie pour lui et comment il la pratique. En ces jours où les appels à la charité publique se substituent aux politiques nationales, beaucoup estiment s'être acquittés de tout devoir envers leur prochain, avec un chèque au profit de l'un de ces organismes dépensant cent millions en affiches pour en envoyer dix dans le tiers-monde. Pour Dave, l'entraide, ce n'est pas cela, surtout en tant que jeune entrepreneur. Et le tiers-monde est à l'est autant qu'au sud. Au fait, la Russie vient de rétablir les cartes de rationnement...

#### Lakewood

Vous ne le saviez sûrement pas, mais les acheteurs du Spectre GCR ont longtemps aidé des handicapés de Lakewood (banlieue ouest de Denver, dans le Colorado). Ma firme, Gadgets By Small, a fait effectuer l'assemblage de nos circuits imprimés par un atelier employant des personnes handicapées. Ne pas confondre un atelier de fabrication de circuits imprimés produit nos cartes imprimées (faites en un matériau qui, je crois, s'apparente à la fibre de verre), avec des pistes en cuivre et des contacts dorés. C'est seulement ensuite que commence la rigolade. Il faut que quelqu'un mette les bons composants dans les bons trous et les y soude. C'est ce que l'on appelle l'assemblage des cartes. Il y a des myriades de problèmes possibles dans un atelier d'assemblage.

Incidemment, recourir à un atelier d'assemblage aux USA, y compris à celui de Lakewood, coûte beaucoup plus cher que de faire souder les cartes à la vague à Taïwan, et de les faire expédier ici. Faire assembler des cartes au Mexique, juste derrière la frontière, dans des zones franches conçues pour attirer les investissements américains, coûte encore moins cher. J'éviterai de vous déprimer, en vous révélant combien de gens m'ont conseillé de laisser tomber Lakewood, et de faire effectuer leur travail à Ciudad Jaurez. Néanmoins, Denver a besoin d'emplois, et en particulier les gens de Lakewood. Ils voulaient travailler mais n'avaient aucune commande. En toute bonne foi, je dois dire que je ne m'attendais qu'à un travail moyen de la part de cet atelier. J'ai vu des ateliers d'assemblage de tous calibres en dix ans, d'excellents à ignobles, et je me suis résigné à l'habituelle loi de l'emm... maximum, problèmes d'approvisionnement en composants, mauvaises communications, etc. Mais, ô surprise!, les gens de Lakewood m'ont fait le plus beau boulot d'assemblage et de soudage que j'aie jamais vu. Le meilleur, point, à la ligne. Les cartes sont soudées à la vague dans ce procédé, tous les composants sont insérés dans leurs trous, puis la carte est placée juste au-dessus d'une cuve de soudure en fusion. Un vibreur engendre une «vague», comme un caillou tombant dans l'eau, qui balaie les points à souder. Après soudage, nos cartes étaient inspectées à la loupe, et toute soudure douteuse était refaite à la main. Nous n'avons jamais vu un composant «en l'air» (sorti de ses trous avant soudage). Tout circuit imprimé douteux, dont les pistes en cuivre n'étaient pas parfaites, était étiqueté pour nous. Les cartes étaient apparemment très minutieusement inspectées. J'ai parfois cherché plusieurs minutes un défaut dans une carte étiquetée, sans trouver la mauvaise piste. Le pourcentage de pannes de ces cartes chez les clients était d'un pour cent, ce qui est excellent, vu la complexité des cartes et la fiabilité des composants. Car quelques-uns des circuits intégrés (CI) d'un lot de mille vont toujours se révéler défectueux. Multipliez cela par le nombre de CI de la carte, et vous aurez un taux de panne minimum. S'y ajoute également ce que nous appelons la «faute de pilotage» beaucoup de cartes renvoyées et re-testées s'avèrent bonnes.

Nous avons fait travailler Lakewod, aussi longtemps qu'ils ont eu du temps à nous consacrer. En fait, nous avons été leur seul client pendant longtemps. L'économie de Denver est encore très atteinte, après s'être désagrégée à la suite de la baisse des coûts du pétrole. Mais finalement, nous avons dû aller trouver un autre atelier d'assemblage, car celui de Lakewood a décroché un gros contrat, pour

produire des chargeurs à grande capacité pour fusils d'assaut genre AR-15/M-16 et pour Uzi, et n'avait plus de personnel pour nos cartes. Il y a une forte demande pour ces chargeurs ces temps-ci, à cause d'une loi en cours d'examen au Congrès, qui pourrait les interdire. Mais je suis heureux d'avoir pu aider Lakewood au temps des vaches maigres.

#### Jamaïque

En ce moment, les acheteurs du Spectre aident l'économie de la Jamaïque, qui en a bien besoin. Un de nos ex-employés lance d'ailleurs une affaire là-bas, et elle est très bien partie. Les employés sont jamaïcains, ils ont des salaires décents, et la boîte vit du tourisme américain, elle n'a rien à voir avec l'informatique.

Nous lui fournissons du capital pour acquérir de l'équipement, et nous lui avons donné un Mac II et une Laserwriter, devenus un peu trop lents pour traiter notre base de données. Pourquoi les aidonsnous ? Parce que cette personne nous a énormément soutenu durant les jours critiques des débuts du Spectre, et que c'est une façon de lui dire merci de tout cœur.

#### Pourquoi?

Pourquoi faisons-nous ces choses ? C'est vrai qu'elles nous coûtent cher. Bon sang, me dis-je, si j'avais pris un atelier bon marché, utilisé des contacts étamés au lieu de dorés, refusé d'aider un employé à réaliser son rêve, et n'avais pas envoyé ces mises à jour et ces lettres d'information gratuites, crénom, je pourrais avoir placé de l'argent et en percevoir les intérêts (humpf!, et en laisser une bonne part aux impôts!). En fait, beaucoup de cartouches pour le ST sont construites de la façon décrite ci-dessus, aussi peu cher que possible. Et le ST lui-même, sous bien des aspects, est construit et architecturé de cette façon.

Oui, je pourrais avoir plus d'argent devant moi, si j'avais fait les économies décrites ci-dessus. Mais pour moi, l'argent ne signifie rien, comparé au bien qui peut être fait en l'utilisant. Spectre (ainsi que certains autres de mes travaux, dans une bien moindre mesure) a touché de nombreuses personnes. Quand je reçois par exemple une lettre de quelqu'un, ayant pu lancer un magazine grâce à un ST, un Spectre, et un programme de PAO pour le Mac dont il avait besoin, le tout accompagné d'un exemplaire du magazine, cela me remplit d'une joie inexprimable... et je ne pourrais vous citer en cent pages toutes les choses que font les gens avec Spectre. Quand je déprime, je relis les lettres de gens ayant pris le temps de m'écrire, pour me remercier d'avoir créé Spectre.

De plus, quoi que fasse Atari Corp., le ST peut facilement devenir un Mac disposant de certains des meilleurs logiciels. C'est peut-être la meilleure chose que j'aie jamais faite pour le ST, lui fournir une assurance. En vérité, je ne suis ici que grâce à des gens qui m'ont fourni une aide que je ne peux leur rendre. Ainsi, le Dr. Kuder et Lee Maxwell de la Colorado State University (CSU) m'ont évité d'être éjecté du lycée. Or, Si mon diplôme d'informatique n'a pas été déterminant, il m'a bien aidé pendant un moment. Sherwin Gooch m'a enseigné l'Éthique du Bidouilleur sur PLATO [NdT : voir «Le Petit Monde de Dave Small» dans ST Mag n° 53]. John Ridges et Phil Tuhb, de ALF Products, m'ont longtemps soutenu, et grâce à eux j'en sais long sur les disquettes. Jerry Pournelle (NdT : auteur de best-sellers de science fiction aux USA [un seul livre, hélas!, traduit en français] et journaliste très influent dans le magazine américain Byte) m'a également beaucoup aidé. Le moins que l'on puisse dire étant qu'il n'y était pas forcé, il y a dix mille autres firmes rivalisant pour accaparer son attention et faire parler d'elles dans ses articles. Mais il a toujours pris le temps d'annoncer les petits bidouilleurs ayant de bonnes idées, leur consacrant autant d'espace qu'aux grosses boîtes (que je déteste d'ailleurs).

Et pourtant, d'après mon expérience journalistique, il est bien plus facile d'écrire sur les grosses boîtes, elles peuvent vous fournir des pages entières de documentation. Or, quand je demande à tous ces gens, mes sauveurs, ce que je peux faire pour leur rendre leur aide, la réponse est toujours la

même : «Aidez quelqu'un d'autre.» Bien sûr, j'ai un peu aidé Roberta Pournelle (NdT : femme de Jerry, institutrice pour enfants difficiles, et néanmoins programmeuse et rousse flamboyante) à longuement tester son programme d'apprentissage de la lecture, pour rendre un peu de ce qu'ils ont fait pour moi. Mais j'ai entendu cette réponse : « Aidez quelqu'un d'autre », tant de fois que j'y crois fermement. Certes, cela flatte l'ego d'aider quelqu'un à réaliser son rêve, mais l'important est d'appuyer un débutant inconnu pour le lancer. Et croyez-moi, je suis heureux de l'avoir fait.

#### Un regard sur l'URSS

Ceci étant dit, il est beaucoup question ces temps-ci des besoins de l'URSS (NdT *et encore, cet article a été écrit avant le « putsch »*). Je sais, je sais, il est étrange de penser à aider l'URSS. Même pour moi qui n'ai jamais eu des exercices d'alerte atomique en classe, ni vécu de crises comme les missiles de Cuba. Beaucoup de gens pensent que nous devrions laisser ce pays s'écrouler et retourner à l'âge de pierre. D'autres croient que nous devrions commencer immédiatement à leur expédier des vivres. Et moi ? je ne sais pas, je le dis franchement. Lisez, vous allez comprendre. En fait, je n'ai aucune acrimonie envers les gens d'URSS. C'est envers leur gouvernement que j'en aie. Les Soviétiques, ainsi que nous, avons énormément souffert à cause de ce gouvernement. Attention, ne m'étiquetez pas hâtivement comme un forcené de droite. (NdT: les étiquettes gauche droite des États-Unis n'ont rien à voir avec les nôtres. Aux USA, les libéraux sont à gauche par exemple, et « libéral » signifie plus ou moins « gauchiste » Et Bush est centriste.) Voyez ce que je pense sur d'autres sujets. Mais je vous signale néanmoins, que le gouvernement soviétique détient le record absolu du plus grand massacreur que l'humanité ait jamais produit. Il est même difficile de savoir à cinq millions près, combien de personnes sont mortes depuis la «révolution» de 1917. Et chacune de ses personnes aurait pu être vous ou moi.

Pour cette raison, je ne veux rien faire qui puisse aider ce gouvernement, car je ne pourrais ensuite me regarder dans un miroir. (J'ai même rejeté une offre pour des Spectre CCR pour cette raison, bien que l'on m'ait offert bien plus que le prix au détail, en liquide sous la table ! Pas question.) Mais les informations me provenant d'URSS, depuis la levée de certaines restrictions, m'ont conduit à un terrible dilemme. Dois-je aider l'URSS ? Si quelqu'un me demande de lui vendre des GCR pour les exporter vers l'URSS, dois-je les lui vendre ? Le Département d'État autorise l'exportation de la technologie 68000 (mais pas des 68020, 68030 ou 68040 [NDLR : *le 68030 est maintenant autorisé à l'export vers l'URSS*]), donc légalement je le pourrais, mais...

Bien sûr, je souhaiterais aider les gens là-bas, dans la mesure où le Spectre leur est utile (après tout, ce n'est gu'un modeste outil, et l'on ne peut pas manger une cartouche!), je n'ai rien contre ces gens, encore une fois. j'ai été très touché par une carte postale, que le programmeur de Tetris a pris la peine de m'envoyer par un intermédiaire, pour me saluer et me dire qu'il avait entendu parler de Spectre jusqu'en URSS. Et le témoignage de quelqu'un revenant d'URSS me confirme que les Soviétiques sont des gens au grand cœur. j'ai reçu d'URSS une lettre, accompagnée d'un CV, d'un homme marié, père d'un enfant, désirant se faire embaucher, et très déterminé d'après sa lettre. Je ne sais comment vous dire combien j'aurais souhaité pouvoir l'embaucher, rien que pour pouvoir le tirer hors d'URSS et l'amener ici. Mais à présent que la censure du courrier a été rétablie par les Soviétiques, j'ai peur de lui attirer des ennuis rien qu'en lui répondant. L'histoire démontre que les gens, ayant essayé de quitter le paradis des prolétaires, ont eu des ennuis allant des persécutions minimes aux traitements psychiatriques intensifs (drogues, lobotomie) en passant par la prison. Cela me fait mal. J'ai donc fait un effort pour faire parvenir des Spectre 128 et des GCR en Europe de l'Est. C'était même l'un des principaux buts de mon voyage en Allemagne, lors du CeBIT de Hanovre en mars dernier, et j'ai consacré beaucoup de temps à des rendez-vous dans ce but. Il y a peu de profits à la clé, vu les problèmes de devises là-bas (rappelez-vous que le rouble n'est pas convertible !). Mais cela n'entre pas en ligne de compte. Vous seriez surpris par l'importance de la présence du ST en Europe de l'Est. Certaines revues américaines consacrées au ST y sont très lues, et leur nombre de lecteurs est bien supérieur au nombre d'abonnements. L'on y trouve aussi un nombre très élevé de programmeurs sur ST, et une quantité vraiment surprenante de bidouilleurs extrêmement compétents. En jetant un œil sur leur code en assembleur 68000, j'ai cherché un bon moment à comprendre, et quand finalement j'ai

compilé, je suis resté admiratif. j'ai vu une astuce géniale, utilisant MOVEP, pour jouer des sons numérisés en immobilisant le processeur le moins possible. Personne, ou presque, n'utilise cette instruction pour quoi que ce soit, moi inclus, ou du moins jusqu'à ce que ces gens m'ouvrent les yeux.

Je sais qu'une compagnie américaine produisait que des programmes en Hongrie, et leur travail était considéré comme du haut de gamme ici ; je ne crois pas que je sois de leur niveau. J'espère pouvoir un jour visiter l'ex-Allemagne de l'Est pour voir quelles sortes de programmeurs vivaient là-bas.

L'Europe de l'Est n'est plus dominée par le Pacte de Varsovie, et certains pays se sont assouplis quelque peu, mais pas l'URSS... Je crains que si des GCR y étaient envoyés, ils seraient, comme les autres technologies occidentales, utilisés par la Nomenklatura, pour rendre la vie des gens un peu plus difficile. J'ai longtemps songé à faire un geste, comme par exemple vendre des Spectre 128 à très bas prix pour l'URSS seulement, et je ne suis toujours pas sûr. (Si vous voulez me donner votre avis et m'écrire, vous êtes les bienvenus, voir à la fin de l'article.

Je suis suffisamment ouvert pour avoir déjà adopté certaines des idées de mes correspondants...) Par ailleurs, John Sculley d'Apple a fait don d'un lot de Mac II à Gorbatchev. Alors que pour d'autres, seul compte le profit. Prenons Gulf Oil, par exemple ils n'ont aucun complexe à financer l'Angola et son régime, grâce à un grand programme de forage offshore. Or l'Angola n'est vraiment pas réputé pour respecter et prendre soin de ses citoyens. Cuba y envoie, en tant que mercenaires, des hommes très jeunes combattent ceux essayant de lutter pour la liberté, comme Jonas Savimbi (Cuba a grand besoin des devises que rapporte ce trafic). Et, bien que Gulf Oil considère ces pratiques comme de bonnes affaires, elles ont leur revers. Tenez, zut !, flûte suis-je maladroit !, ma carte de crédit Gulf vient juste de se faire déchiqueter et leurs lettres de relance sont parties au panier! Curieusement, c'est arrivé également à bien d'autres personnes. Ouaip !, de bonnes affaires! Toute ironie mise à part, j'ai le sentiment profond que si je cessais de respecter les idéaux et les gens, mes réalisations finiraient rapidement d'exister. Cela n'est pas exprimable en dollars, mais pourtant beaucoup de ce qui constitue le Spectre est en rapport avec les gens, lesquels ne peuvent figurer sur une feuille de bilan financier.

#### Voyage en URSS

Ainsi, lorsque nous sûmes que Kristie, la fille d'amis de Sandy, ne pouvait réunir les fonds nécessaires à un voyage en Union Soviétique, dans le cadre d'un échange d'étudiants, le «bon cœur» de Gadgets by Small permit de compléter la somme (en fait, elle va nous servir de baby-sitter en échange, chose que ceux d'entre vous qui sont parents, savent être plus précieuse que l'or). Le collège de Kristie échange durant quatre semaines des élèves avec une école soviétique, et une étudiante soviétique était auparavant venue aux USA. L'échange avait commencé durant l'administration Reagan, grâce à un accord entre lui et Gorbatchev concernant plus de 50 écoles.

Au fait, ne confondez plus la Russie et l'URSS. La Russie n'est que l'une des nombreuses républiques de 'URSS, et comme la susceptibilité nationaliste croissante, et le désir d'indépendance de ces républiques, s'opposent au déclin du pouvoir central, il n'y a plus d'entité monolithique dans ce pays. A ce sujet, je me rappelle, lorsque je vivais à Austin, au Texas, un livre à succès traitait de la sécession du Texas hors des USA... Il m'a été donné de bien connaître Kristie, après qu'elle ait travaillé ici pour ma firme l'été dernier. Elle a fait des boulots ingrats, tels que le nettoyage de mon bureau l'occupant une bonne semaine (je ne rigole pas, il n'avait pas été nettoyé depuis avant le GCR), ou la saisie de mon carnet d'adresses sous Hypercard. S'y sont ajoutés des travaux informatiques plus sophistiqués pour Sandy. J'ai constaté qu'elle parlait parfaitement le russe, et elle apprend l'allemand par ailleurs (NdT : très utile en russe, langue où la plupart des mots techniques ou modernes viennent de l'allemand, et se prononcent de la même façon, par exemple, lager, camp ; informatsion, information pounkt, point spion, espion). Son don pour les langues est peu commun, et il est parfaitement déprimant pour moi, infoutu de demander où sont les toilettes au Mexique après cinq ans d'espagnol. Si vous avez besoin d'un traducteur anglo-russe, faites-moi signe, je transmettrai... Ayant l'assurance qu'elle pourrait discuter avec les Soviétiques durant son voyage, je lui ai demandé à son retour un entretien, que je retranscris ici pour vous fournir un point de vue original dans la cacophonie médiatique entourant l'URSS, abondant en « reportages en profondeur de trois minutes ». Ce n'est pas un

entretien professionnel bien léché, mais spontané. J'y ai ajouté quelques notes. Je me suis contenté d'ôter certains noms. L'URSS peut sombrer demain dans le chaos, et qui sait ce que les interlocuteurs de Kristie peuvent subir...

#### **Entretien: Voyage en URSS**

Question : Dans quel cadre s'est déroulé ton séjour?

Réponse : Notre école échange des étudiants avec une école soviétique. Un groupe soviétique est venu aux USA passer quatre semaines, et à notre tour nous sommes allés séjourner un mois en URSS. Ma correspondante soviétique, une lycéenne, a été particulièrement éberluée par les supermarchés. Il y a tout ce dont l'on a besoin dans un seul magasin, et de nombreuses marques concurrentes de produits semblables. En URSS, les magasins sont spécialisés, et il n'y a pas de marques concurrentes. Il y a, de plus, de fréquentes pénuries pour tous les produits. En allant faire les courses en URSS, j'ai compris pourquoi elle était si ahurie. Durant leur séjour ici, les Soviétiques n'en ont guère parlé, mais ouvraient de grands yeux. Cela se comprend si, en visitant un pays étranger, je découvrais des choses bien mieux qu'aux USA, je n'en parlerais pas non plus. Certains Russes pensaient même que nous ne travaillions guère et que tout nous était donné. C'est sans doute parce qu'ils sont venus nous visiter en période de vacances, et que beaucoup de parents étaient aussi en congé, de telle sorte que les jeunes Russes ne voyaient personne travailler.

Q : As-tu eu des problèmes en URSS parce que tu étais américaine ? (Note : mes parents ont fait un voyage en URSS dans les années 70. Tous les clichés étaient au rendez-vous ils ont été suivis, il semblait y avoir des micros dans la chambre d'hôtel, le personnel soviétique souffrait d'espionnite aiguë, ils ont eu droit à la visite obligatoire de Lénine embaumé dans son cercueil de verre comme Blanche-Neige, et les douaniers, particulièrement hostiles, les ont retenus durant des heures.)

R : Absolument aucun problème. Les douanes nous ont regardés passer sans rien ouvrir, ce qui valait mieux pour nous. Nous n'avons jamais eu l'impression d'être suivis. Et nous avons partout été particulièrement bien traités, parce que justement nous étions américains, ce que les Américains ont du mal à croire quand je le leur raconte. Tout le monde savait que nous étions américains, et nous nous faisions constamment aborder, principalement pour du marché noir. En ces occasions, nos interlocuteurs avaient très peur de se faire prendre. Mais tous étaient très chaleureux, très sympathiques.

#### Marché noir et militaire

Q : Que voulaient-ils acheter ou vendre?

A: Ils voulaient vendre des vêtements militaires, parce qu'ils nous en savent friands. Ils vendaient aussi beaucoup de montres. Ils voulaient tout particulièrement nous acheter du chewing-gum, ainsi que tous nos vêtements et nos chaussures, que ce soit leur taille ou non. Les vêtements soviétiques, que j'ai vus, étaient taillés à la mode des années 70, pantalons en pattes d'éléphants et habits disco. Tous parlaient très bien l'anglais. Beaucoup étaient adolescents, ne les empêchant d'ailleurs pas d'être redoutables en affaires et de bien marchander. Je n'ai vu aucune femme faire du marché noir. Le féminisme n'a pas encore atteint ce secteur d'activité! J'en ai profité pour acheter des vêtements, qui auraient fait sourciller les douaniers s'ils avaient ouvert mon sac au retour. Le marché noir est omniprésent. Un Américain de mon groupe a vu deux vendeurs au noir se battre férocement dans un magasin de disques, parce que l'un opérait sur le «territoire» de l'autre. J'ai été quotidiennement, à de très nombreuses reprises, abordée par des marchands au noir, et le marché noir est une partie essentielle de la vie soviétique, car il y a tant de choses impossibles à trouver par les canaux « officiels »

Q : Il y a eu des échos de soldats soviétiques vendant leurs armes et leurs vêtements avant de quitter certains pays de l'Est (NdT : on mesure l'ampleur du problème en rappelant que 300 000 soldats de

l'Armée Rouge stationnent toujours sur le territoire allemand). Penses-tu que les vêtements proviennent de là? (Note d'après les médias occidentaux, ces soldats vendent avant de partir non seulement leurs uniformes, mais aussi leurs fusils d'assaut AK47, AK74 et Dragounov, des munitions, etc.)

R : Oui. Tous les garçons doivent là-bas faire leur service après leur bac. Je le sais bien, car j'habitais juste en face d'un baraquement militaire. Nous l'appelions «la baraque aux bébés», parce que les garçons avaient tous 17 ans, l'air très jeune, et portaient une lettre V cyrillique sur leurs épaulettes, s'écrivant comme notre B romain, comme «Bébé»! Avant d'aller en classe le matin, les garçons doivent aller à l'entraînement militaire.

#### Les femmes

Q: Tu as parlé de féminisme. Du point de vue féminin, décris-nous ce que tu as vu en URSS.

A : La vie des femmes est infernale là-bas. Elles travaillent toutes, quel que soit leur âge. Elles sont au travail environ 9 heures par jour, parfois plus, après quoi elles doivent chaque soir faire la queue pendant environ deux heures, pour essayer de trouver de quoi dîner. Nos références n'ont plus cours là-bas. Une des Soviétiques qui participaient à notre échange était copropriétaire de la première manufacture privée de XX (nom d'une grande ville, tenu secret sur demande) fabriquant des articles ménagers. Cette usine appartient à ses employés, qui en partagent les bénéfices. C'est une grande nouveauté chez eux, alors que c'est acquis chez nous. Les femmes n'ont guère d'espoir quant à l'avenir.

Ce sont elles qui subissent le plus de contraintes. Une enseignante soviétique, étant revenue des USA après son mois de séjour, a été complimentée sur sa bonne mine de vacancière. C'est que les femmes commencent très tôt là-bas à avoir l'air hâve. Et depuis mon séjour, la censure a été rétablie, les dernières lettres que j'ai reçues ayant des découpes ou des zones noircies à l'encre. C'est un très mauvais signe, et je m'inquiète pour les Soviétiques participant à l'échange.

#### L'alimentation

Q : Mais pourquoi les femmes font-elles la queue tous les soirs, au lieu de faire des courses pour une semaine ?

A : D'une part, les ménages n'ont guère de bons réfrigérateurs. Nous avions acheté de la crème glacée, mise au congélateur, où elle a fondu. D'autre part, le plus souvent, il n'y a pas assez de nourriture à vendre pour commencer. Depuis mon voyage, j'ai appris que les cartes de rationnement pour la nourriture avaient été instaurées. Et à cause de la censure, je ne sais pas vraiment ce qui se passe d'autre. Nous étions hébergés chez une Soviétique ayant un chat.

Un jour, l'écuelle de ce chat était vide, et elle est allé lui acheter un poisson. Dans le bus, les gens avaient l'air si tristes et si faméliques. Elle pensait qu'elle ne pouvait donner un poisson à son chat, alors que les gens crevaient de faim, si bien qu'elle y a renoncé. Mais ce qui m'a frappé, c'est ce qui lui est arrivé avec les médecins.

#### La médecine

Cet épisode est significatif de la médecine soviétique en général. Notre hôtesse souffrait de terribles douleurs stomacales. Elle est allée voir un docteur et a subi une ablation de l'appendice, pour laquelle

elle a eu une anesthésie, ce qui est inhabituel. Mais les douleurs ont continué. Elle s'est fait réopérer, et on lui a trouvé une tumeur bénigne, ce n'était pas une appendicite finalement. Et cela n'a rien d'extraordinaire là-bas.

Q : Une anesthésie inhabituelle, as-tu dit?

A : Oh, oui! Ils ne font pas d'anesthésie pour la « petite » chirurgie ou les soins dentaires, là où nous la tenons pour allant de soi! C'est pour ça que les Soviétiques n'aiment vraiment pas aller chez le dentiste.

On m'a aussi dit là-bas que les plombages sont d'une très mauvaise qualité, ils sont en alliages ferreux s'usant très vite, comparés aux amalgames (NdT: *plomb-mercure*) que nous utilisons. Pour seulement pouvoir consulter un docteur, il faut en avoir un parmi ses amis. Impossible de débarquer dans un cabinet. Théoriquement, la médecine est gratuite. Vous pouvez aller dans un dispensaire coopératif étatique, mais ils sont très mauvais (comme toutes les coopératives là-bas, d'ailleurs). Alors, en allant voir un docteur, vous êtes censé lui apporter un cadeau, « par pure bonté d'âme », peut-être sera-t-il plus attentionné. Ce qui peut signifier la vie ou la mort pour vous.

Les docteurs sont hideux. Ils portent des sortes de toques de boulanger et des tabliers de boucher, avec même des taches de sang! C'est à la fois drôle et macabre. Nous avons pu les voir de près, quand un garçon de notre groupe s'est blessé. Pourtant, les Soviétiques ont inventé une méthode de correction de la vision par des incisions précises du cristallin, donc certains de leurs médecins sont excellents. Peut-être n'ai-je pas vu de vrais pros ou de docteurs en ayant l'air. A noter qu'il y a plus de femmes que d'hommes chez les médecins soviétiques.

Q : Qu'est-il arrivé à cette personne de votre groupe?

R : Ce garçon s'est blessé au genou et souffrait beaucoup. Il avait emporté de bons médicaments, antidouleurs et décontractant, des USA, car il s'était déjà blessé au genou. Il est allé à l'infirmerie de l'école. Mais l'infirmière ne voulait pas le laisser prendre ses médicaments, elle disait que cela affecterait ses radiographies Elle voulait lui faire une piqûre. Le directeur du groupe n'a pas laissé faire, parce qu'ils emploient des seringues sales, et que leurs médicaments sont d'une qualité douteuse.

Il a été transporté à l'hôpital. Il a eu une chambre pour lui seul, parce qu'il était américain. D'autres chambres comptaient jusqu'à six personnes (NdT ben quoi, en France aussi !) Il y avait des flopées de cafards, nous devions les tuer jusque dans son lit. Il y avait des patients jusque dans les couloirs. Ils lui ont plâtré la jambe sans l'envelopper dans une chaussette, ni même la lui raser d'abord, Si bien que ses poils ont été pris dans le plâtre et qu'il a eu des ampoules. De retour chez lui, il s'est fait enlever le plâtre, et les médecins ont été effarés par ce travail. Ils en ont longuement parlé.

#### Les gens

Q : Les gens t'ont-ils paru amers et agressifs envers toi ?

R : Oh non! Les gens sont formidables. Ils font un effort pour que vous soyez content. Ils sont très, très amicaux envers les Américains. Et ce n'était pas de la mise en scène, je l'ai constaté partout. Je me souviens d'un magasin bondé, dans lequel je voulais regarder des bagues. Le père de mon amie russe a dit: «C'est une Américaine.» Aussitôt tout le monde m'a cédé le passage. Je me sentais un peu gênée. Il est impossible que le magasin, pris au hasard, ait été rempli de membres de la police secrète (NdT : dans les années 60, le KGB organisait souvent des mises en scène édifiantes à l'intention de touristes occidentaux étroitement encadrés. Le NKVD faisait déjà de même avant-guerre. C'est devenu un cliché.)

Q : Pourquoi cet empressement après la guerre froide et des années de propagande?

R : Aucune idée! J'aimerais bien que les Américains agissent de même. Ils étaient très gentils, c'est ce qui m'a marqué, et c'est de cela que je me souviendrai. J'aimerais y retourner un jour.

#### L'école

Q : As-tu visité des écoles?

R : Les classes s'échelonnent de la Première à la Onzième (NdT : équivalent de notre Cours Élémentaire à notre Terminale). Jusqu'en Huitième (NdT notre Troisième), les élèves sont très polis. Pour poser une question, ils lèvent la main de la manière prescrite, le bras gauche horizontal, le bras droit vertical. Ils ont, à la limite, le droit de se lever et d'agiter le bras droit s'ils sont impatients, mais en silence. A partir de la Neuvième, les élèves parlent en classe ouvertement et n'écoutent pas, comme ici.

Q: As-tu vu des ordinateurs?

R : Je n'en ai vu aucun Les écoles n'en ont pas. Ici, nous sommes habitués à l'informatisation partout, comme dans les supermarchés. Mais en URSS, les caissières utilisent des bouliers et des caisses enregistreuses manuelles. Le paiement de vos achats semble durer une éternité, à supposer que le magasin soit approvisionné. Par contre, les écoles utilisent des rétroprojecteurs. La situation est comparable à celle des USA dans les années 50, avant que l'informatique ne se répande. (Note : le magazine Byte parle de l'informatique soviétique dans son numéro d'avril 1991. Les observations de Kristie recoupent celles que l'on y trouve. Par exemple, les microprocesseurs soviétiques sont livrés avec une liste des instructions ne marchant pas sur cette puce, un peu comme la liste des mauvais secteurs sur un disque dur. Imaginez la programmation pour des processeurs, dont certaines instructions peuvent ne pas marcher... Et Atari a parlé, il y a quelque temps, d'envisager d'acheter des RAM dynamiques soviétiques. Vu leur qualité, je m'étonne qu'ils y aient seulement pensé, quel qu'en soit le prix.)

Q : Ton entourage soviétique parlait-il d'informatique ?

R : Oui, ma correspondante voulait être informaticienne, et allait à « l'école technique », ce qui est une mauvaise traduction du russe, car cette école est une maison dont les pièces sont transformées en classes. Quand elle est venue aux USA, elle a été enthousiasmée par mon Apple II GS (NDLR : dérivé de l'Apple II original, n'ayant rien à voir avec), et elle a passé des heures à jouer. Elle a particulièrement aimé les jeux.

#### Les transports

Q: Comment te déplaçais-tu?

R : Surtout par les transports en commun. Les bus et les métros sont très propres. Les seuls graffitis que j'ai vus étaient en anglais (!) ou des croix gammées. On m'a dit qu'il y avait un groupuscule jouant aux nazis et imitant nos skinheads, mais qu'il comptait suffisamment peu de monde pour tenir dans une cabine téléphonique. Les transports en commun sont très réguliers et ponctuels, ce qui est une bonne chose, car les gens en dépendent complètement. Je ne voudrais d'ailleurs pas posséder une voiture soviétique.

A ce sujet, une anecdote : à un arrêt de tram, une femme nous a demandé si nous pouvions l'emmener aux États-Unis, comme ça. Nous avons répondu que nous ne pouvions pas. Elle a dit : «Saluez l'Amérique de la part de la malheureuse Russie. » Elle était très sérieuse. Nous étions profondément attristés.

#### Conclusion

J'espère que cet article vous a apporté quelques informations. Si vous voulez faire des commentaires, vous êtes les bienvenus. Je n'ai pas le temps de répondre au courrier « papier », et je préfère de loin le courrier électronique. Soyez patients si je mets parfois longtemps à répondre.

Traduction et adaptation: Password 90

## Le salon WAACE Atarifest' 91

WAACE n'est pas une onomatopée, mais l'abréviation d'«enthousiastes pour les ordinateurs Atari de la région de Washington». Ce salon a eu lieu les samedi 12 et dimanche 13 octobre 1991 à l'hôtel Sheraton de Reston, dans le nord de la Virginie, à quelques kilomètres de Washington que l'on appelle là-bas «Washington, D.C.» (District de Colombus), pour différencier de l'État de Washington, qui est, lui, sur la côte ouest. Les salons dédiés à Atari attirent moins l'attention aux États- Unis qu'un Comdex, mais n'en constituent pas moins des événements majeurs. Plus de 2000 visiteurs y ont afflué, et de nombreux vendeurs y ont écoulé la totalité de leurs stocks. Dave Small était lui-même présent, et a donné une des conférences du salon. Son stock à lui, Dave l'a écoulé en cette occasion d'une manière peu ordinaire, bien dans son style...

#### L'automne au Colorado

Le temps des feuilles mortes est revenu. C'est sans aucun doute la plus belle saison au Colorado. C'est aussi l'époque du salon WAACE. Mais pour que vous compreniez l'importance du WAACE à mes yeux, permettez-moi de m'attarder sur le Colorado en cette saison, et de vous expliquer pourquoi il est dur de le quitter.

Le Colorado est tout en contrastes, c'est quatre états en un. A l'est, vers le Kansas, le paysage s'arrondit en douces collines parsemées de fermes, en faisant un des greniers à blé de l'Amérique. Les labours y préparent les sols pour l'hiver.

juste à l'ouest de Denver (NdT : proche de la limite nord de l'État), les plaines sont bordées par les Montagnes Rocheuses, comptant plusieurs sommets de plus de 4000 mètres éparpillés dans le Colorado. Ils forment un paysage d'une splendeur terrible ; ils grimpent si haut que les arbres ne peuvent y pousser, et que les hommes ont peine à y respirer. Au sommet de ces pics, vous atteignez un endroit particulier l'eau qui y ruisselle d'un côté s'écoule dans la rivière Platte, puis dans le Mississippi, et finalement dans le golfe du Mexique, alors que de l'autre côté, cette eau atteint le bassin du fleuve Colorado qui se dirige vers le sud-ouest et se jette dans le golfe de Californie. C'est pourquoi cette ligne de pics est appelée le Partage Continental.

Ces montagnes sont un endroit magique, comme peut vous le dire quiconque y a été. Il suffit de s'asseoir au sommet et on le ressent... En cette saison dans les montagnes, les arbres prennent des couleurs spectaculaires, et les peupliers déploient une fantastique palette passant du vert à l'orange, puis au rouge. Les nouvelles télévisées parlent même du «changement des peupliers».

L'ouest du Colorado est un désert chaotique, une région aride et sauvage, semblable en bien des points à un Grand Canyon inachevé. Mais il recèle des trésors. C'est dans ce désert que se ruèrent jadis les prospecteurs d'uranium, compteurs Geiger en bandoulière, pour chercher le précieux minerai nécessaire aux centrales. Là, les gens rassemblent du bois pour l'hiver, et les tas de bois près des maisons de Kremmling arrivent souvent à la hauteur du premier étage.

Enfin, le sud du Colorado est très semblable au Nouveau-Mexique qui le borde, une grande région de paysages trompeurs, cédant abruptement le terrain à des formations comme la Gorge Royale ou le Grand Canyon. Je me souviens y avoir traversé en voiture une petite ville, où les écoliers avaient allumé des luminaires faits d'une chandelle dans un sac en papier, dans toute la ville et jusqu'à un kilomètre à l'extérieur sur la grande route.

C'est pour cela que quitter le Colorado à cette époque est presque un crime. L'air y est extrêmement pur, les arbres déclinent des couleurs qu'on a décrit comme «le feu d'artifice de la nature», la

température convient aux manches de chemises, l'humidité est très faible, les gosses jouent au foot, tout se ligue pour me demander de rester... Et pourtant chaque année je pars pour assister au WAACE, le salon dédié à Atari, organisé par le club d'utilisateurs d'Atari de la région de la capitale, ainsi qu'à la Hackers Convention ensuite. J'ai assisté à ces deux manifestations chaque année depuis 1986, sans exception. J'écris ceci le jeudi qui suit le WAACE, et par ma fenêtre j'aperçois les pics du Partage Continental. C'est dur de partir. Heureusement, Dieu merci, j'ai préféré ne pas aller au Comdex cette année, pour m'éviter les cinq jours de folie non-stop auxquels j'ai eu droit les années précédentes. Nous allons donc parler du WAACE de cette année. Ce n'est que ma vision des choses, et sans doute en aurez-vous d'autres échos. Mais cette année, au WAACE, nous avons fait des choses, heu!... inhabituelles

#### **WAACE 91**

Cette année fut différente pour moi, car je n'ai pas eu à transporter des machines jusqu'au WAACE, ce qui est vraiment pénible. George Richardson et Jeff Creenblatt avaient amené trois ST munis de cartes accélératrices SST, et je n'avais donc à trimbaler que quelques habits, ma brosse à dents et des logiciels.

George est l'expert en électronique qui nous a rejoint lors de la conception de la MegaTalk et de la carte SST. Jeff Greenblatt s'occupe des bêta-tests de mes logiciels depuis cinq ou six ans. Ensemble, aidés par la femme et le fils de George, nous avons tenu le stand de Gadgets By Small.A notre stand, nous avons exhibé les ST les plus rapides du monde. Le titre était disputé par les ST de Jeff et de George.

J'avais promis à Jeff un microprocesseur 68030 à 50 MHz, ainsi que le quartz associé, s'il m'aidait à tenir le stand. Autour de nous se trouvaient d'autres vendeurs ayant bénéficié du flot de gens assiégeant notre stand. L'affluence était telle que j'étais vraiment content qu'il y ait une table entre la foule et nous!

Nous avons distribué des boîtes entières de prospectus, avons pris des commandes, mais n'avions aucun SST à vendre. En fait, le premier lot fabriqué de SST est déjà vendu, et donc retenu, et je me vois mal vendre les mêmes cartes deux fois! Cette situation nous est familière, car en dépit de tous nos efforts, le Spectre GCR n'est jamais parvenu à sortir de son état de rupture de stock en deux ans. Nous les fabriquons aussi vite que nous pouvons, en quantité qu'il nous est possible de produire (acheter ces composants est ruineux!), mais nous ne pouvons jamais en faire assez, particulièrement lorsqu'on nous en commande 500 à la fois.

Nous avons sans arrêt fait tourner sur nos SST des tests de vitesse (les nôtres et les «autres»), ainsi que divers programmes de test. Nous avons montré Chromax, la nouvelle carte vidéo pour SST, mise au point par George. Le samedi, après la fermeture du salon, je me suis même mis en 20 minutes à programmer en assembleur sur Chromax, histoire de dessiner à l'écran des rectangles en résolution de 640 x 480 et 256 couleurs... qui n'est que l'un des modes de la Chromax. Un test plus spectaculaire pour moi, mais sans doute moins parlant pour la foule mon test-du-mode-rafale-en-adressage-plat. Cela consiste à charger, dans les 4 mégaoctets de la SST, un programme simple occupant chacun des 4 millions et quelques d'octets, et à l'exécuter. Ce qui prend environ un quart de seconde, et nous donne à peu près 7,2 MIPS en traitement continu. Un record. Autrement, nous lancions régulièrement Q-Index qui indiquait une vitesse de plus de 1200%, soit 12 fois plus vite qu'un ST normal.

#### Le discours

J'avais une conférence à faire à treize heures le samedi. L'heure fatidique arrivait. Certes, j'en ai l'habitude. Cela fait des années que j'en donne. Les spectateurs étaient debout (pas de chaises) dans la salle des séminaires, et il y avait de la place pour beaucoup de monde...

A ma grande surprise, je découvris que j'étais nerveux! J'avais l'un des symptômes usuels du trac, à savoir que mon estomac était plein de papillons qui voletaient. Dans les cours de communication que j'ai aidé à enseigner, l'on donne une astuce tenter de faire voler ces papillons en formation. C'est donc ce que je fis quelques minutes avant le discours. Voyez-vous, la plupart du temps, je tente d'être détaché durant mes discours publics, et je ne parle jamais de choses ayant un impact émotionnel (donnant sa force à un discours). Cette fois, j'avais décidé de faire l'inverse. Mais d'abord, j'avais prévu un peu de détente.

Nous avions installé un magnétoscope et un écran de télé pour que les spectateurs le voient. Et soudain, ce fut à moi. Il y avait même dans l'audience des gens qui me filmaient en vidéo!

La pression était telle qu'il m'était difficile de contrôler ma voix... Je commençai à parler exprès dans ce que j'appelle le Mode Monotone et Ennuyeux. C'est la façon de s'exprimer de la plupart des gens n'ayant pas l'habitude de parler en public, et c'est radical pour endormir les auditeurs. Je saluai, et expliquai doctement que durant ce séminaire, j'allais expliciter beaucoup de termes techniques. Car, dis-je, depuis l'avènement des cartes accélératrices à base de 68030, tant de nouveaux termes de jargon informatique étaient apparus, que personne n'y comprenait rien. Les auditeurs furent suffisamment polis pour ne pas bâiller, mais leurs mines indiquaient assez que cette heure allait être une des plus longues de leur vie.

J'entamai avec le mode rafale. «Le mode rafale, expliquai-je, permet de lire 16 octets d'une fastRAM organisée en 32 bits, et de les stocker dans l'antémémoire d'instructions en 5 cycles, avec post-incrémentation exécutée par la mémoire externe.» Les gens ouvraient de grands yeux, les sourires s'étaient figés, et chez beaucoup, ils avaient totalement disparus.

J'ajoutai alors que Gadgets By Small préparait une vidéo pour expliquer ce concept, vidéo que j'allai leur montrer. J'enfonçai la touche de lecture du magnétoscope et montai le son et la luminosité. J'avais positionné la bande avec beaucoup de soin avant le discours, et je priai que tout marche bien. Bizarrement, la vidéo montrait quelqu'un dans une forêt, et ce quelqu'un, c'était moi. J'expliquai que le cameraman était un certain Tom Hudson, auteur de Degas et de CAD-3D (ce qui était la stricte réalité).

Brusquement, la personne sur l'écran leva un M-16 et tira une longue rafale continue de 30 cartouches sur une cible! Les douilles éjectées décrivaient un arc gracieux et scintillant dans l'air. Le tir s'arrêta, tandis que roulait l'écho interminable des détonations dans les montagnes du Colorado... Il était temps d'arrêter la vidéo, il y avait un écho dans la salle aussi. «C'est ça, le mode rafale», dis-je.

Dans l'assistance, plus personne ne s'ennuyait ou ne s'assoupissait. Je vis quelques mâchoires inférieures pendre jusqu'à ce que leur propriétaire se rappelle qu'il fallait la refermer. J'avais assurément retenu leur attention. Par ailleurs, cette scène est réellement tirée de la vidéo de GBS, sur laquelle nous travaillions vraiment. Mais elle est encore en projet.

A partir de là, la conférence devint plus sérieuse. Je suis connu pour faire à ce salon des choses spéciales. Une fois, par exemple, j'ai amené une bobine de Tesla (NdT: *générateur à induction de haute tension à haute fréquence*), j'ai donné aux spectateurs un tube fluorescent, et je leur ai montré qu'il pouvait s'allumer entre leurs mains par induction, sans aucun fil. Alors, vous pensez bien qu'un simple fusil d'assaut n'allait pas arrêter les spectateurs du WAACE. Mais c'est la partie d'après qui fut pour moi la plus dure.

J'ai alors parlé aux gens en toute sincérité. je leur parlai de Dave et Sandy Small, tels qu'ils étaient en 1988, fauchés et découragés. je leur parlai de Sandy qui dessina le circuit imprimé de la cartouche Spectre 128 avec MacDraw, n'étant guère conçu pour cet usage, imprima le dessin sur imprimante laser, et l'utilisa pour les films d'insolation des cartes. je leur parlai des trois mois d'efforts intenses passés à programmer Spectre, tandis que notre compte bancaire oscillait autour de zéro. Enfin, je leur parlai de la fabrication des deux cents premières cartouches financées en tirant sur les cartes de crédit VisaMastercard. Comme je le dis : «Si nous ne les avions pas vendues, nous aurions été dans une sale situation.»

Mais nous les avons vendues! Notamment au WAACE de cette année. Ce qui nous a permis de rembourser les bonnes gens des organismes de cartes de crédit avant qu'ils ne nous lynchent... Et le Spectre s'est bien porté depuis lors. je vous l'ai dit, en deux ans, nous n'avons jamais pu en avoir en stock, non encore vendus.

A ce moment, j'en ai bien peur, une certaine émotion transparût dans ma voix (alors que j'apprends aux gens à parler en public !), tandis que je me remémorais trop précisément cette époque difficile. Je me tournai et attrapai derrière moi une grande boîte. je dis au public qu'à l'intérieur se trouvait de nombreuses cartouches Spectre 128. Tandis que je marchais dans la foule, micro en main, j'expliquai que j'avais discuté avec Sandy du soutien que le WAACE et son public m'avaient toujours apporté, et que nous avions décidé de distribuer gratuitement ces Spectre 128 pour les remercier. «Immédiatement», dis-je. je tenais à finir cette phrase rapidement, cela devenait très dur de parler, car ma gorge se serrait.

Pendant une seconde, l'on aurait pu entendre tomber une épingle. Puis il y eut un son curieux, celui de gens inspirant enfin après avoir retenu leur souffle. Tim Powers, un auteur que j'admire beaucoup, a fort justement décrit ce bruit : «On aurait dit un parapluie que l'on enfonçait dans une cheminée.» Ce bruit, je m'en souviendrais longtemps.

J'en distribuais une dizaine et trouvais deux volontaires pour m'aider. L'ambiance s'anima très vite, menaçant de tourner à l'émeute. Plus tard, on m'a raconté que des gens avaient couru dans le hall principal du salon en criant «Dave Small donne des Spectre!». George Richardson m'a dit que la foule qui entourait les stands disparût, pour se rematérialiser dans la salle des séminaires quelques millisecondes plus tard.

Lorsque nous tombâmes à court de cartouches (nous en avions quelques centaines), ma voix était redevenue normale. J'annonçai à présent: «nous abordons la seconde partie de la conférence.» Mais je ne me rappelle pas un traître mot de ce que j'ai bien pu dire ensuite. Cette première partie, avec la mise en boite, la vidéo, puis la distribution, m'avait vidé. je ne crois pas pouvoir faire mieux dans un prochain salon!

C'est sans aucun doute la conférence la plus surprenante que j'ai jamais tenue, celle où j'ai le plus «poussé l'enveloppe» (notre expression favorite à GBS). Mais j'avais voulu dire merci. Et bien sûr, personne d'autre que Dave et GBS n'est assez fondu, pour donner gratuitement quelques centaines d'émulateurs Mac en parfait état de marche (certes, ils ne contenaient pas les ROM d'Apple, mais j'ai autorisé les heureux propriétaires à copier le logiciel auprès de quelqu'un l'ayant déjà>. Mais c'était la chose à faire. Le WAACE a contribué beaucoup pour l'existence même de Gadgets By Small. Ce sont des organisations comme ce salon, permettant à de petites compagnies, sans grand capital de départ, de s'établir, comme ce fut notre cas.

Ce n'est qu'une heure après la conférence que mes mains ont cessé de trembler. C'est qu'elle avait vraiment donné lieu à une grande décharge émotionnelle.

#### Le banquet

Chaque année, le WAACE comporte un banquet avec un discours. Cette année, les gens assis à notre table s'avérèrent être un peu plus chahuteurs que d'ordinaire... mais cela pouvait être attribué au bar et à la réception précédant le banquet. On eut à déplorer quelques bagarres à coups de nourriture (moi aussi ? Vous êtes sûr ? Grands dieux !), une interprétation improvisée d'une chanson de Neil Young par quelques développeurs qui étaient aussi musiciens, et enfin, nous applaudîmes des gens qui le méritaient, comme des champions sur un stade. Je voyais des gens aux autres tables nous montrer à leurs voisins et dire : «je ne sais pas ce qu'on leur a servi à dîner, mais je veux la même chose !»

Nathan Potechin d'ISD fit un beau discours, décrivant des parallèles entre Atari et «Le Seigneur des anneaux» de Tolkien. (NdT : j'ai appris par ailleurs que son discours comparait Frodon à Sam Tramiel, Gollum à Leonard Tramiel, et l'Anneau unique au marché). Le prix du meilleur auteur de l'année du magazine Current Notes fut décerné à David Troy (NdT : distributeur Atari, bidouilleur de disques durs et auteur d'articles), qui le méritait bien à mon avis. Je suis persuadé que dans dix ans, je serais content de pouvoir dire que j'ai bien connu David Troy, alors qu'il n'était encore que responsable de son magasin, car il ira loin.

La compagnie Codehead a affirmé que ce fût son meilleur salon, même en incluant «le bon vieux temps d'Atari». Tous les développeurs que je vis étaient souriants. L'endroit était bondé samedi, un peu moins dimanche. Je préfère les affluences raisonnables, car l'on peut y parler avec des clients potentiels sérieux, sans être interrompu 50 fois par des gens ne pouvant attendre pour poser une question. Les gens étaient vraiment très gentils. Ainsi, j'ai dit à quelqu'un que je n'avais pas trouvé de Pepsi avant le début du salon, et il est revenu avec six boites. Il s'est avéré qu'il avait dû marcher plus d'un kilomètre pour me les trouver (Conseil j'ai découvert que ma voix tient mieux, malgré les discussions incessantes, si je sirote une boisson pendant le salon).

J'ai tenu une discussion publique avec Jim Allen, qui a sorti lui aussi une carte accélératrice 68030, durant laquelle nous avons fait taire les rumeurs de disputes personnelles entre nous. j'ai souligné que j'avais une Camarro que je chérissais, et qu'outre moi-même, seules deux personnes pouvaient la conduire, ma femme Sandy et jim quand il passait nous voir.

J'ai également expliqué que si une personne voulait une carte accélératrice, mais que la mienne ne lui convenait pas, il fallait qu'elle considère l'achat de celle de Jim. J'ai noté que quelques spectateurs se tortillaient, mal à l'aise en cette occasion. Peut-être voulaient-ils voir du sang ? En tout cas, ils sont restés sur leur faim. Et finalement, tard le dimanche, j'ai pu prendre le temps de visiter le salon. En marchant rapidement, j'ai pu éviter d'être intercepté pour répondre à «juste une question». Et j'ai trouvé des extracteurs de circuits intégrés PLCC! (ce sont les circuits carrés qui sont sur la carte mère de votre ST).

Ces outils sont très durs à trouver. J'ai aussi acheté, entre autres, un correcteur grammatical qui m'évitera des erreurs embarrassantes dans mes prochains articles (NdT : pas trop tôt !). Bref, un salon formidable, et si cela ne dépend que de moi, on m'y reverra l'an prochain.

Merci à tous les volontaires bénévoles, souvent pas même remerciés ayant contribué à faire de ce salon une réussite. Et merci à Charles Smeton (NdT : *président du groupe WAACE*) de m'avoir prêté une télé et un magnétoscope, pour un usage auquel il n'aurait jamais songé...

Traduction et adaptation Password

# Les affres de la création (première partie)

Le processus de la création intellectuelle est une chose fascinante. Les grands créateurs de ce monde - qu'ils produisent des œuvres de papier ou des inventions géniales - ont souvent une aura de mystère ou de légende. Et Dave Small ? En tant que bidouilleur de génie, Dave a souvent connu les angoisses de l'idée géniale qui prend tant de temps et de peine à se concrétiser. A cela se sont ajoutés les soucis de l'entrepreneur. Cette expérience vaut la peine d'être partagée. Ce premier article d'une série de deux nous décrit les phases durant lesquelles germe l'Idée...

#### Perles de sagesse

D'après le courrier que je reçois et les discussions que j'ai lors des salons informatiques, bon nombre de personnes voudraient savoir quel effet cela fait d'écrire un programme comme Spectre ou d'autres (le formateur Twister, l'utilitaire de sauvegarde Meg-a-Minute, etc.). Il semble que certains de ces gens recherchent des perles de sagesse dont ils pourraient bénéficier pour entreprendre quelque chose de similaire.

Eh bien, je n'ai malheureusement pas grand chose à leur fournir en ce domaine. Les idées arrivent, sans que je sache trop pourquoi, et je n'en sais guère plus sur la façon dont elles arrivent aux autres. Mais je suis passé à travers la moulinette d'un projet de programmation suffisamment souvent pour savoir reconnaître les symptômes de tout le processus. Passons donc en revue les différentes phases de la création...

#### Le stade où l'on se cogne aux portes

Durant ce premier stade, vous pensez à quelque chose que vous aimeriez créer et vous commencez à vous concentrer dessus. Progressivement, l'idée occupe une part croissante de votre esprit vous commencez à y penser pendant que vous attendez à un stop, pendant que vous faites la queue, pendant que vous parlez à votre femme, bref chaque fois que votre pleine attention n'est pas requise par le monde extérieur. (Mais oui, je plaisantais, Sandy, lâche ce fusil!)

Votre esprit se focalise tant sur le problème que vous commencez à vous heurter aux obstacles, d'où le nom de ce stade. Beaucoup de gens appellent cela de la distraction. Ce n'est pas que l'on soit distrait, c'est que l'esprit est très occupé à travailler sur quelque chose d'autre (et tout chercheur en Intelligence Artificielle vous dira qu'un cerveau humain qui se focalise sur un problème est infiniment plus créatif et plus puissant que n'importe quel ordinateur). Cette situation est similaire à ce qui se passe quand on travaille sur un ordinateur multitâche lorsque quelque chose consomme la majeure partie du temps de calcul. Pour voir ce que je veux dire, il suffit de prendre une machine Unix quelconque et de lancer en tâche à fond dix programmes gourmands en calcul. Vous obtenez alors l'équivalent informatique de ce premier stade. Si vous demandez alors quoi que ce soit à votre machine, vous constatez que vous commencez à pianoter d'impatience avant qu'elle ne réponde. Si votre ordinateur pouvait marcher, il se cognerait aux murs: parce qu'il ne les remarguerait pas.

De plus, ce que vous voulez faire est parfois vraiment impossible dans l'état actuel du matériel et du logiciel. Ces problèmes-là sont à devenir fou vous avez beau repasser la question en revue dans votre tête et accumuler les pansements sur votre front, rien n'y fait, vous ne trouvez pas de solution... Cela

vous démange, impossible de l'oublier. Mais là où cela devient vraiment dur, c'est quand plus tard un mois ou un an après, le même problème resurgit dans votre esprit, vous narguant calmement, se faisant insistant... Se pourrait-il qu'au fond, ce que l'on nomme "sénilité" ne soit que l'accumulation d'un trop grand nombre de ces "tâches de fond" insolubles, sur lequel l'esprit perd à travailler?

J'ai jadis écrit un article [NdT : à paraître dans un prochain ST Mag] où je parlais des deux grands types psychologiques "N" et "S". Cette classification répartit la personnalité des individus en ces deux types selon qu'ils sont introvertis, intuitifs et vivent dans leur propre monde, ou qu'ils sont extravertis, ancrés dans leur cinq sens et tournés vers les sensations extérieures. je n'ai absolument jamais rencontré de programmeur qui ne soit pas du type intuitif "N", bien que la moitié de la population soit du type "S". Les non-programmeurs de ma connaissance qui ont des personnalités de type "S" ont en général des activités sportives genre baseball où ils peuvent mettre à profit les avantages que leur procurent leur système sensoriel développé. A l'opposé, les programmeurs travaillent le plus souvent en bâtissant des structures ou des images mentales et n'ont guère besoin de développer leurs cinq sens. Se cogner distraitement dans les portes convient assez à ce type de personnalité.

Donc, votre problème vous tourmente. Peut-être est-il insoluble. Cependant, il convient de ne jamais faire de conclusion hâtive et supposer que quelque chose ne peut pas se faire. Laissez donc au problème le temps de vous obséder, de saturer votre esprit, car Si vous vous accrochez, vous arriverez au stade suivant.

#### Le stade de l'inspiration fulgurante

Ce stade commence lorsque vous prenez votre bain tranquillement, réfléchissant au problème, lorsque soudain votre crâne se met à bouillir et la solution apparaît! Vous vous mettez alors à courir dans la maison, complètement à poil et dégoulinant, en hurlant "Eurêka".

(Ne croyez pas que ce soit une plaisanterie, la légende veut que ce soit réellement ce qu'a fait Archimède lorsqu'il a trouvé la façon d'obtenir le volume exact d'une couronne d'or dont on lui avait demandé de s'assurer qu'elle ne contenait pas de plomb. Archimède s'est aperçu dans sa baignoire trop pleine que le volume d'eau qui débordait lors de l'immersion d'un objet était précisément le volume de l'objet. Simple, mais génial.)

Ce n'est sûrement pas drôle d'être l'épouse de quelqu'un qui traverse cette phase, comme ma femme Sandy vous le dirait. D'abord, on se prend les pieds dans la serviette humide qui traîne par terre, ensuite il y a des flaques dans toute la maison, et enfin la Police Municipale de Denver qui ramène encore l'inventeur nu à la maison. (Cependant, c'est peut-être bien un progrès par rapport à la phase précédente où le conjoint endommage les papiers peints avec son front, marmonne "musique orientée objet" sans arrêt, et gribouille des schémas sur les serviettes et nappes en papiers des restaurants. Mais je n'ose poser la question.)

Prenons un exemple. Supposons que vous vouliez doubler la vitesse d'entrée/sortie des lecteurs de disquettes du ST. Le problème est la vitesse de rotation de la disquette: pendant que vous déplacez la tête de lecture de piste en piste, la disquette tourne. Et quand, à la fin de la piste courante, vous passez à la piste suivante, le temps de déplacement de la tête est suffisant pour que vous ratiez le début de cette piste, qui est le secteur 1. Cela vous oblige à attendre un tour complet pour revenir au début. C'est la phase d'identification du problème. La phase Eurêka survient lorsque vous réalisez qu'on peut résoudre le problème en décalant la numérotation des secteurs de trois unités. Ainsi, durant le déplacement de la tête, les trois derniers secteurs de la piste suivante défilent et quand la tête est positionnée, elle trouve immédiatement le secteur 1. Et le lecteur de disquette peut à présent travailler à pleine vitesse. C'est la base du fonctionnement de Twister.

Ou bien supposons que vous réfléchissiez au moyen de faire tourner des logiciels Mac sur l'architecture matérielle du ST. La plupart de gens ne dépassent jamais le stade où ils lancent un "impossible" dédaigneux. Mais ceux qui le dépassent commencent à se cogner aux portes tandis qu'ils s'attaquent aux problèmes au fur et à mesure qu'ils se présentent. Puis Jaillissent toutes sortes de petits Eurêkas quand les problèmes sont résolus.

Ainsi, le Mac réserve les adresses mémoires de \$100 à \$13F pour ses variables globales, des données essentielles à tout le système. Le ST, lui, a besoin de cet espace pour la table des vecteurs d'interruptions du MFP (en clair, cet espace lui sert à savoir où aller lorsque vous bougez la souris, frappez une touche ou, d'une manière générale, quand vous engendrez une interruption). La solution ? Une commande peu connue qui déplace la table à l'adresse \$CO, hors du chemin du Mac, vu que celui-ci se contrefiche de ce qu'il y a entre \$C0 et \$FF.

Ou encore, les disquettes du Mac sont enregistrées au format GCR (Croup Coded Record, enregistrement codé par groupes) et sont écrites avec une vitesse de rotation variable. Les disquettes du ST sont au format MFM (Modified Frequency Modulation), appelé aussi "FM double densité" et tournent à une vitesse constante. La solution ? Pour pouvoir lire des disquettes de Mac sur le ST, le circuit de lecture/écriture doit pouvoir faire varier sa cadence de fonctionnement pour émuler une variation de la vitesse de rotation.

Souvent, les logiciels du Mac écrivent accidentellement à l'adresse mémoire zéro. Ce qui est sans conséquence sur un Mac. Mais sur le ST quelque peu différent (l'adresse zéro est à lecture seule), cela cause une erreur de bus (deux bombes) et plante l'ordinateur. La solution ? (Le manuel Motorola du 68000 dit qu'il n'y en a pas.) Écrire un gestionnaire d'erreur de bus! Plutôt que d'afficher deux bombes à l'écran et de tout bloquer, le gestionnaire tente de déterminer quelle instruction a causé ce cafouillis, et essaie de la sauter intelligemment ou de la ré-exécuter différemment de façon à ne pas causer le problème, puis poursuit l'exécution. A peu près la moitié des logiciels Mac utilisés sur un ST causent des erreurs de bus régulièrement, et les utilisateurs ne s'en aperçoivent pas. Seules de nouvelles manières, particulièrement irritantes, d'écrire à l'adresse zéro parviennent à présent à planter le Spectre.

Et ainsi de suite. Une foule de "petits problèmes", autant d'"Eurêkas", et énormément de bosses au front ! Après tant d'années d'émulation Mac (rappelez-vous, mes premières cartes datent de novembre 1985), ma femme a bien des raisons de me surnommer "tête de linotte".

Mais on ne peut jamais prédire quand un Eurêka va se produire et que votre subconscient va résoudre un problème et le livrer tout mâché à votre esprit conscient. Je sais par contre que l'on peut bloquer complètement ce processus en essayant de le forcer. La plupart des Eurêkas se produisent lorsque votre esprit conscient est à la dérive, comme durant un bain ou pendant que vous tondez la pelouse. Et lorsque cela arrive, vous avez l'Exaltation.

#### Le stade de l'Exaltation

Après la phase Eurêka, une indicible exaltation envahit tout votre corps. Plutôt que de m'évertuer à la décrire, je vais tenter d'établir quelques analogies, après quoi vous verrez ce que je veux dire, ou vous mettrez en doute ma santé mentale... Mais je vous assure que je n'exagère pas.

Écoutez donc l'Ouverture de 1812 [NdT: de Tchaikovski, se caractérise par un tir de canons à la 6ème minute]. Elle cherche son rythme, semble heurter un moment les limites du raisonnable, puis s'échappe dans un fracas de canons.

Le meilleur représentant de la musique rock est le premier album de Boston. Le morceau "Long Time" contient ce stupéfiant son de guitare qui a rendu le groupe célèbre : la "distorsion propre". Ordinairement lorsque vous distordez un son de guitare, vous ne pouvez faire vibrer qu'une seule des six cordes à la fois, car sinon le bruit devient horrible. Mais il est plutôt difficile de jouer un air avec une seule corde. Bien des gens y sont obligés pour pouvoir utiliser la distorsion (voir comme exemples typiques, Neil Young, "Like a hurricane", "Out of the blue", "loto the black"). La distorsion de Boston se débrouille pour donner un son magnifique en utilisant les six cordes. En fait, le leader du groupe, Tom Scholz, commercialise à présent une boite qui réalise cet exploit, le Rockman. Je le sais, j'en ai une. Et comme beaucoup de guitaristes aiment ce son, le Rockman se vend bien.

Si vous permettez une digression... Un peu avant Noël, Sandy m'a emmené à un magasin de musique; elle a branché une guitare, un Rockman sur la guitare, et m'a tendu le tout. J'ai gratté un mi, et voici que ce son merveilleux m'a envahi et m'a emporté. Une bonne minute plus tard, je suis revenu sur

terre, remarquant distraitement que mes poils étaient toujours dressés sur mes bras et qu'un frisson me parcourait l'échine, tandis que Sandy remplissait déjà un chèque pour le Rockman.

L'exaltation vaut la peine d'être vécue. Elle justifie les bosses au front et toutes les difficultés traversées. Plus il vous a fallu lutter pour parvenir à résoudre un problème, plus sa résolution s'avère gratifiante. Je pense en être arrivé à aimer me fixer un problème apparemment insoluble, puis le résoudre, en repoussant les limites connues un peu plus loin. Je n'ai aucune idée de la raison pour laquelle les solutions me tombent du Ciel. Peut-être Dieu me prend-il en pitié et dit: "Donnons-lui une centaine d'Eurêkas dans sa vie, il en a bien besoin!"

Ou bien est-ce en fait une malédiction ? Vous aimeriez, vous, vous trouver avec des tas de problèmes qui vous démangent l'esprit ? J'en traîne toujours, des non résolus qui me titillent. Par exemple, la trisection de l'angle (partager un angle en trois angles égaux avec une règle et un compas), un vieux classique des problèmes insolubles, qui m'a turlupiné depuis que j'en ai entendu parler pour la première fois, ce qui remonte, oh, au Cours Moyen. j'ai fini par admettre que quel que soit le nombre de fois où l'on divise un angle en deux, ce qu'on sait faire avec une règle et un compas, on n'obtient jamais un dénominateur commun avec une trisection.

Sans compter les autres problèmes informatiques, comme faire utiliser par un PC plus de 640 K de mémoire... J'ai l'intuition qu'il existe un moyen auquel on n'a pas encore pensé.

Qui que vous soyez, l'Exaltation peut soudain vous frapper, vous envahir, et les gens constatent alors que vous avez cessé de parler et qu'une très étrange lueur s'est mise à briller dans vos yeux... Et c'est alors que commence...

#### Le stade du bafouillage

Ce stade commence juste après celui de l'Exaltation. Il se manifeste par un désir d'expliquer à un sceptique votre solution à un problème. Et comme votre esprit est encore sous l'influence de l'Exaltation, vos phrases ne sont guère cohérentes. C'est que vous n'avez plus assez de temps de calcul à consacrer à votre interface verbale. Souvent, vous croyez devoir éclaircir vos bouts de phrases incohérents en dessinant de petits diagrammes sinueux. Ce qui n'aide pas les sceptiques à changer d'avis. En général, à ce point les gens prennent un air nonchalant, et s'assurent discrètement qu'il y a une sortie à proximité et que vous n'avez aucun objet tranchant à portée de main.

Inévitablement, vous avez droit à un certain nombre de haussement de sourcil et d'expressions amusées qui iraient très bien à Monsieur Spock ("Vraiment, Docteur, vous devriez apprendre à maîtriser vos émotions")

Certaines personnes ne parviennent pas à saisir le message. D'autres ont une réaction embarrassée quand elles sont à leur tour atteintes par l'exaltation et qu'elles réalisent ce que vous avez essayé de formuler. (Naturellement, elles se mettent ensuite à feindre de ne pas être impressionnées... Mais parfois, elles trouvent votre idée si fantastique qu'elles ne peuvent pas décemment rester calmes - car sinon, elles pourraient laisser croire qu'elles n'ont rien compris!)

Que le Ciel vous vienne en aide si vous vous trouvez en pleine phase de bafouillage avec quelqu'un qui ne partage pas vos connaissances en programmation. "Mais, Monsieur l'agent, je viens juste de résoudre le problème du conflit de table d'interruption du MFP! Qu'est-ce que cela peut bien faire que je sois nu?"

Bien, la partie de plaisir est finie. Vous vous êtes cognés dans tous les murs, avez fini par trouver votre Eurêka, êtes passés par les stades d'exaltation et de bafouillage. Maintenant, vous devez implémenter votre idée.

#### Le stade où, aie, c'est parti pour être un sacre boulot

Après avoir réalisé que c'est faisable, le moment est venu de foncer. Le monde est rempli de gens qui ne peuvent pas mener à bien ce stade ils ont de bonne idées, essaient parfois d'en parler autour d'eux, mais ne les mettent pas en pratique. La programmation est particulière en ce sens où vous devez implémenter votre idée pour arriver à montrer où vous vouliez en venir, et c'est un processus long et acharné. (Vous ne me croyez pas ? Comparez le manuel d'utilisation d'un Mac à sa pratique - en regardant les fenêtres, les icônes, en déplaçant la souris. Le plus gros problème de marketing d'Apple c'est de faire en sorte que les acheteurs de Mac puissent le prendre en main et l'essayer, ce que ne remplace aucune description écrite.)

(Atari prenez note de cette dernière phrase. Vous aussi. Cela explique pourquoi les groupes d'utilisateurs, les démonstrations et les vendeurs sont si importants. Jack Tramiel passe pour avoir dit quelque chose comme "Il est important de laisser au moins un ordinateur à un vendeur, même à perte le vendeur a besoin de le toucher, de le comprendre, de faire corps avec lui, avant qu'on ne lui en achète. " Qu'il ait dit cela ou non, c'est tout à fait vrai.) Pendant que vous transpirez sur l'implémentation, seule votre volonté vous permet de mener à bien ce stade. Il n'y a rien d'aussi exténuant que de contrer un ordinateur avec son seul esprit. Absolument rien. (Obliger votre femme à faire ce que vous voulez ? Non, ça, c'est complètement impossible!)

Quand je lis des histoires fantastiques de sorciers invoquant des démons pour leur faire accomplir certaines tâches, je peux réellement m'identifier au personnage. Les ordinateurs sont des êtres diaboliques. Ils ne se sentent pas le moins du monde concernés par ce que vous voulez leur faire exécuter, et n'ont pas la moindre intention de le faire.

Beaucoup de gens, particulièrement les jeunes diplômés tout juste sortis de leur école, partagent l'idée risible que les ordinateurs constituent un monde précis, exact, encadré par des limites nettes, sans nuances, à la grande différence du monde réel. C'est totalement faux, et il est presque douloureux de voir ces gens se heurter violemment aux réalités. La vérité, c'est qu'il y a tant d'inconnues, y compris dans un ordinateur que vous connaissez parfaitement, que vous pouvez toujours faire de fausses manipulations. Et bien des fois, vous planterez tout sans comprendre pourquoi.

Croyez-moi, l'air de chagrin incrédule d'un jeune programmeur face à un ordinateur irrécupérablement planté est quelque chose de triste. C'est le même air que quand ils apprennent que la Petite Souris n'était en fait que Papa et Maman, que le Père Noël n'existe pas et que les œufs apportés par la Cloche de Pâques viennent du supermarché. Car jusqu'à ce moment-là, ils s'étaient enfoncés dans le crâne que si un ordinateur se conduisait mal, c'était entièrement leur faute et celle de leur mauvais programmes. C'était si souvent vrai qu'ils avaient fini par se persuader que ça l'était toujours. Raté

Dans un ordinateur se passent toutes sortes de bizarreries. Il peut y avoir des bugs dans l'unité centrale (exemple du microprocesseur 68000 du ST: l'instruction MOVE SR, D0 qui accède à la pile sans vous le dire). Il peut y avoir des parasites électriques, ou bien des résonances dans un câble de disque dur. Les rayons cosmiques peuvent littéralement (je ne plaisante pas) inverser des bits au hasard à l'intérieur des puces mémoires, ce qui est la raison principale de la présence d'un neuvième bit de parité dans les compatibles PC (qui permet au moins de tout arrêter quand la mémoire a été corrompue). Et ainsi de suite.

Tous les vieux programmeurs que je connais, qui ne se soient pas lassés ou évadés vers une profession plus facile (terrassier. par exemple), traitent les ordinateurs comme s'ils étaient des démons. Ils se méfient de leurs machines, gardent un nombre ahurissant de sauvegardes.. pratiquent la "programmation préventive", et tentent de se préserver de la loi de l'emm... maximum. Bien sûr, il leur arrive aussi de se faire avoir par des bugs, mais la différence c'est qu'eux n'ont pas abandonné l'informatique.

Les vieux programmeurs ont tendance à collectionner les outils, les utilitaires, et surtout les debuggers. Ils ont appris à leurs dépens que souvent, il n'y a qu'un seul outil qui puisse leur montrer ce qui ne va pas - particulièrement lorsqu'il s'agit d'une erreur due à un parasite. Beaucoup de gens se demandent pourquoi je ne suis pas très chaud pour les "langages de haut niveau". C'est que très souvent j'ai dû chercher des bugs sans pouvoir les voir parce que le langage me les cachait; c'était l'équivalent de débugger en aveugle. A ce moment, il vous faut abandonner toute idée préconçue de ce que la machine est censée faire, et essayer d'imaginer ce qu'elle fait vraiment.

Bien sûr, cela nous ramène aux autres problèmes, et des bugs peuvent être trouvés de la même façon.

surtout quand ils sont particulièrement vicieux. Vous commencez à penser à ce bug, il commence à imprégner votre esprit...

(Vous vous cognez aux portes. Eurêka. Exaltation. Bafouillage. Boulot. Vous voyez, nous n'avons pas encore résolu le problème principal et nous appliquons à ce sous-problème les mêmes étapes. C'est ce qu'on appelle la récursivité.)

D'ailleurs, Si vous croyez que les gens qui écrivent les compilateurs sont parfaits, vous croyez sans doute également au Père Noël. Si vous utilisez un langage compilé de haut niveau et que l'auteur du compilateur ne connaissait pas le bug du MOVE SR,D0 (et qui le connaît donc ?), attendez-vous à des problèmes. Votre programme va se planter à tout bout de champ sans raison discernable. Seules peuvent vous renseigner une session prolongée d'exécution pas à pas des instructions en langage machine, aidée d'une trace de ce qui s'est passé juste avant le plantage.

Et ça, ça signifie l'assembleur. C'est pour cela que je travaille ainsi. Écoutez, la moitié du temps d'un programmeur se passe à débugger. C'est pourquoi les bons programmeurs sont toujours en train d'écouter les récits des bugs découverts par les autres programmeurs, tout en prenant mentalement des notes. Et c'est aussi pourquoi aussi les bons programmeurs hantent les messagerie informatiques. cherchant des solutions et posant des guestions lorsqu'ils sont coincés.

Enfin, si vous tenez toujours à contraindre votre machine à vous obéir, vous finissez par découvrir qu'il vous faut écrire une routine de 500 lignes pour résoudre votre problème. A ce moment, il faut serrer les dents et l'écrire Résistez à la tentation de vous trouver quelque corvée domestique urgente et faîtes-le. Rien ne remplacera ce travail. C'est long et pénible, mais c'est nécessaire, c'est le jeu. Ce n'est guère différent de se trouver confronté à une toile vierge à peindre ou une page blanche à remplir. Soit vous vous montrez créatif, et menez votre programme à bien pour qu'il tourne et que d'autres gens puissent le voir, soit vous le garderez pour toujours dans un coin de votre esprit. rien que le vôtre. et à quoi cela servirait-il ?

Et si vous vous accrochez et vous obstinez, votre programme finira un jour par tourner rond si vous le traitez gentiment. Et après mise au point il résistera même aux abus. Il vérifiera les données en entrée, par exemple.

Il sera alors temps de montrer aux autres votre création.

(Suite et fin le <u>mois prochain</u>.) Traduction et adaptation Password 90

# Les affres de la création (deuxième partie)

Le mois dernier Dave a commencé à nous exposer les différentes phases de la création, plus particulièrement en matière de programmation. Nous avons ainsi pu découvrir "le stade où l'on se cogne aux portes", "le stade de l'inspiration fulgurante", "le stade de l'exaltation", "le stade du bafouillage", "le stade où, aïe, c'est parti pour être un sacré boulot". Une fois cette phase surmontée, les idées du départ sont concrétisées en un produit. Mais les difficultés ne sont pas finies pour autant. Il faut à présent faire savoir au monde ce que vous avez créé! L'aventure continue...

#### La démo

Cela peut arriver n'importe où. Chez un ami, lors d'une réunion d'un club d'utilisateurs, ou encore à un salon. Vous mettez la disquette dans le lecteur, vous double-cliquez sur l'icône, et votre bébé est lancé. C'est un moment chargé d'émotions pour les créateurs, et cela me fait toujours sourire d'en voir qui font semblant de ne pas faire attention à ce que l'on dit de leur cher enfant.

Soyons clairs, Si les commentaires laissent vraiment un développeur indifférent, c'est que le programme ne vaut probablement pas la peine d'être acheté. Un programmeur est censé être l'interface entre les êtres humains et les ordinateurs, non ? Et Si quelqu'un fait un programme tout en ignorant totalement les humains, il n'est pas programmeur. En fait, il n'y a pas encore de mot dans le langage pour désigner une telle sorte d'individu.

Lors d'une démo de votre dernière création, soyez paré à ce que beaucoup de gens n'en voient pas l'utilité ("Un émulateur Mac ? Mais d'abord, qui donc pourrait bien vouloir faire tourner des applications Mac ?"). Préparez-vous aussi à ce que d'autres n'aient qu'une compréhension partielle du concept ("Accélérer les accès disque, hein ? C'est quoi, un disque ?"). Enfin, soyez prêt à tenir tête à ceux qui comprennent le sujet, mais qui ne sont pas décidés à reconnaître vos mérites pour avoir résolu le problème et leur avoir apporté votre bidouille ("Oh, vous n'avez fait qu'écrire quelques programmes pilotes pour le Mac").

Il y a en informatique énormément de gens blasés qui croient que toute innovation géniale est sans valeur. La plupart des gens qui écrivent des articles dans des revues informatiques finissent ainsi [NdT: Dieu m'en préserve], le plus souvent parce qu'ils ont été trop fréquemment déçus. Préparezvous à subir quelques bancs d'essais disant que votre idée est nulle. Mais gardez à l'esprit la Loi de Small pour la Survie après les Bancs d'Essais tout testeur qui refuse de reconnaître qu'il y a un tant soit peu de bon et de mauvais dans tout produit n'est pas digne d'être entendu. Par conséquent, quiconque jugerait que la totalité d'une création n'est qu' "un tas de (bip)" ne doit pas être écouté, car rien n'est jamais entièrement mauvais. De même, l'avis de quelqu'un qui s'extasie sur un programme "complètement dément" ne vaut pas le papier sur lequel il est imprimé. Souvenez-vous que beaucoup de critiques musicaux sont incapables de jouer d'un instrument. Si un testeur est si brillant, pourquoi n'écrit-il donc pas des programmes ?

En dernier ressort, c'est le public qui doit comprendre ce que vous avez fait Si vous voulez en obtenir quelque chose. "Si vous ne pouvez, à la longue, faire savoir à tout le monde ce que vous avez fait, c'est que ça ne vaut rien". C'est d'Erwin Schrödinger (1887-1961), un des plus grands physiciens qui ait jamais vécu. Et c'est l'une des raisons pour lesquelles j'écris des articles. A quoi bon faire quelque chose Si personne ne s'en aperçoit jamais ?

#### La vente

Je ne prétends même plus écrire ces articles pour favoriser mes ventes. A la vérité, j'écris et je crée articles et programmes dans un élan similaire de créativité. C'est presque entièrement égoïste... A ceci près qu'il est difficile de traiter d'égoïste quelqu'un qui survit à tous ces stades ("masochiste" conviendrait mieux).

Il se trouve que je suis marié à quelqu'un qui a le sens du commerce (Sandy) et que j'ai eu de la chance. Nous connaissions quelqu'un qui maîtrise à fond l'art du service clientèle (Barb), ainsi qu'un programmeur qui a encore autant d'énergie à consacrer aux nuits blanches que j'en avais à 22 ans (Doug). Ajoutez-y quelques personnes qui travaillent en indépendants pour nous, et vous avez Gadgets By Small.

Ainsi, toutes les petites idées que j'ai eu (ainsi que celles que d'autres me proposent à présent) s'additionnent. Certaines déboucheront sur des articles pour des magazines allez savoir pourquoi, je semble avoir des facilités pour expliquer de façon concise l'idée générale d'un programme dans l'espace restreint d'un article, ce que recherchent les rédacteurs-en-chef.

Je n'ignore pas que les autres programmeurs ne sont pas aussi chanceux que moi; ils vendent leurs créations à des boites de programmation en échange d'un (maigre) pourcentage, ou encore ils les distribuent en tant que sharewares, pour de tout aussi maigres bénéfices. Beaucoup d'entre eux se retrouvent dans la catégorie des artistes sans le sou. Mais j'en connais beaucoup d'autres qui programment par plaisir, comme j'aime moi-même à le faire. Je dois vous dire que cela ne m'a pas pris très longtemps de faire apparaître dans mon premier émulateur Mac l'icône du " Mac malade ", un écran qui apparaît quand les routines de diagnostic du Mac découvrent à l'allumage que la machine fonctionne mal. Un vrai supplice de Tantale. je voulais en obtenir plus. Et j'aurais sans doute fait l'émulateur Mac de toute façon.

L'une des raisons qui font que je garde espoir pour le ST est qu'il est agréable de créer quelque chose sur cette machine. Son apprentissage n'est pas trop abrupt, comparé à d'autres, et il peut être d'une simplicité enfantine à programmer. Il y aura toujours des gens qui écriront des programmes pour le seul plaisir de repousser les limites, de "pousser l'enveloppe".

#### Caser la marchandise

Je n'emploie pas cette expression avec le dédain qu'elle sous-entend pour la vente. Il y a des développeurs qui ne peuvent gagner assez pour survivre en vendant leurs créations propres, et qui n'ont hélas aucune chance d'obtenir des subventions. Il leur faut donc rengainer leur créativité, faire du porte-à-porte et chercher à gagner leur vie en programmant. Ils font alors des choses passionnantes, comme des vérificateurs orthographiques pour un produit dont le développement sera plus tard arrêté par la direction, lors d'une dispute portant sur la politique de la compagnie. Ou bien ils trouvent des idées de nouveaux programmes qui sont dénaturées par le marketing, l'ennemi suprême de tous les programmeurs (" Vous ne pouvez pas le faire ressembler, disons, à CECI?" Le marketing, c'est le domaine dans lequel bossent les gens qui ne savent pas programmer).

Tout cela, je l'ai fait, c'est pourquoi je mesure la chance que j'ai en ce moment d'avoir ma firme GBS (et ça me donne une forte motivation pour continuer à la faire marcher!). Je me rappelle l'horrible frustration qu'engendrent de bonnes idées que l'on n'a ni le temps, ni l'énergie de concrétiser, à la fin d'une journée au bureau. Il est très dur de retrouver un peu de tonus après une journée entière passée à se défendre contre des couillons du marketing qui ont "quelques soucis" quant à l'état d'avancement de votre programme et veulent "une réunion hebdomadaire pour être tenus au courant". Et pourtant, on ne peut pas dire que travailler en indépendant ou avoir sa propre boîte soit bien vu. Les facs ont

tendance à vous jeter dans les bras de grosses compagnies, avec la bonne conscience que donne la sensation d'accomplir une noble tâche. C'est ce que je croyais : je suis entré chez Control Data Corporation après la fac. Mieux encore, j'ai été assez stupide pour m'installer dans la Silicon Valley. Eh oui, j'ai vécu à San José, en Californie, alors que Sandy était à Austin, au Texas, après ses études. J'avais là un cas de conscience auquel j'ai mis bon ordre quatre mois plus tard en déménageant. Certaines personnes démarrent ainsi dans la vie, fatiguent de s'entendre dire que "c'est comme ça qu'on fait quand on est un pro", et commencent à créer en à-côtés. C'est ainsi que j'ai débuté. Mon premier article pour le magazine Creative Computing fut écrit sur un mini-ordinateur Texas Instruments, et non sur une machine à écrire, vu l'usage fréquent que je faisais de la touche "Correction" (je tape affreusement mal). Je l'ai imprimé pendant un week-end sur une des imprimantes du mini, et ce fut mon grand départ. Encore une fois, j'ai eu de la chance. Mais il faut donner un coup de pouce à la chance le hasard doit bien trouver une occasion de vous aider, et la plupart des gens ne la lui fournissent pas.

En attendant, je vois souvent de grandes innovations dans le domaine public ou parmi les sharewares. La seule différence entre moi et certains de ces auteurs est que j'écris...

#### L'épuisement

Il y a un danger bien plus insidieux, qui est de se consumer et de tomber en panne sèche. Je l'ai vu arriver bien souvent. Cela m'est même arrivé plusieurs fois, et la rémission fut à chaque fois éprouvante. Ne l'ébruitez pas, mais je sors à peine en ce moment d'une de ces périodes.

Imaginez donc de vous trouver assis dans un SR-71 Blackbird, réacteurs au ralenti. Cette énorme puissance, totalement sous votre contrôle, vous enveloppe. Deux turboréacteurs attendent votre bon vouloir... Il n'y a rien de meilleur que d'enfoncer les deux manettes des gaz à fond, de sentir la détente de la post-combustion qu'on enclenche. Dans les jets de gaz brûlants se créent des ondes stationnaires en forme de losanges. Vous vous retrouvez littéralement écrasé dans votre siège par l'accélération, vous tirez un peu sur le manche, atteignez les cinq mille mètres d'altitude, dix mille, quinze mille... Exalté, vous stabilisez à vingt-cinq mille, Mach 3... [NdT : les détails sont authentiques, le père de Dave a été pilote de SR-71]

Mais on ne peut indéfiniment encaisser autant d'adrénaline, et il y a une limite au nombre de fois où l'on peut ainsi se dépasser. Les gens jeunes y arrivent mieux que leurs aînés, c'est pourquoi les jeunes pilotes poussent leurs zincs à leurs limites, et conduisent de la même manière.

Durant ma tumultueuse jeunesse, j'avais installé un moteur de Corvette (5,4 litres de cylindrée, 365 CV, version à carburateur du moteur à injection) sur le châssis léger d'une Chevy. Et j'ai accéléré très fort. Par le plus pur des hasards, il y avait parfois à côté de moi une autre voiture qui accélérait autant, mais non, monsieur l'agent, on ne faisait pas la course! Ayant eu ma part de conduite brutale, je suis maintenant un conducteur plutôt calme. Il est même rare à présent que je double, rétrograde en troisième, pied au plancher, me délectant du feulement du moteur de 7 litres à haute compression (11:1) et de l'accélération de ma Camarro. Je l'ai fait suffisamment de fois. En un sens, c'est devenir adulte. Mais comme l'a dit Robert Heinlein, "Il est stupéfiant de constater combien devenir adulte ressemble à devenir trop fatiqué". Les ordinateurs peuvent vous épuiser, brûler toutes vos ressources. Comment? Par dessus tout, je pense que le grand coupable est le débugage, qui est très négatif. En débuguant, vous cherchez vos propres erreurs. Si vous voulez être un "bon programmeur", vous devez vous concentrer sur vos propres gaffes. Pas sur votre code correct. Souvenez-vous que les ordinateurs ne disent absolument rien quand votre programme marche. Un compilateur C, par exemple, se contente de vous rendre la main si vous n'avez pas cafouillé. Pas de bravo, rien! Et n'importe quel psychologue vous dira les malheurs qui arrivent à ceux qui ne regardent que les côtés négatifs de la vie. En termes humains, supposez que vous travailliez pour un patron qui n'ouvre la bouche que pour vous faire remarquer vos erreurs. C'est fondamentalement mauvais pour les gens qui subissent cela, et je crois que c'est ce qui épuise bon nombre de programmeurs. Ils découvrent soudain qu'ils ne sont plus heureux de rester assis devant leur écran, en se grattant la tête et en se demandant ce qu'il ont fait de mal cette fois-ci.

Il est très pénible de vouloir créer et d'avoir une dépression qui vous en empêche. On en arrive à en vouloir à la machine, ce qui bloque toute nouvelle idée. La solution est de s'arrêter, de se reposer, d'espérer, et de prier pour que le vieux démon de la créativité revienne vous chatouiller. Le temps guérit tout. Inutile de forcer l'inspiration, il vous faut attendre qu'elle daigne revenir. J'ai quant à moi vécu de vraies périodes de folie comme, disons, Spectre, qui sont parfois totalement épuisantes. Tant de choses à faire, le téléphone qui sonne et vous interrompt, l'entourage dont il faut s'occuper... Le tout peut vous vider complètement. La seule solution à ce point est un trou dans le sol doté d'un couvercle que vous pouvez verrouiller. (Je me contente de la porte de la salle de bain). Le lecteur de disque du Spectre GCR, par exemple, m'a presque achevé. Je peux vous le dire, les gars, ce fut vraiment affreux. Ah, oui, c'est très simple d'emploi - vous introduisez la disquette au format Mac et ça marche, facile, hein ? - mais l'implémentation m'a donné moult occasions de serrer les dents à les rompre. Je suis tombé sur tous les bugs que j'aie jamais imaginés, et sur bien d'autres encore. De nombreuses choses "impossibles" sont arrivées. Vous n'avez pas vraiment vécu avant d'avoir découvert qu'un lecteur de disquette peut aller à la piste -1 (oui, "moins un" !) et se plaindre de n'y trouver rien à lire, et c'est impossible parce que "tout le monde sait" qu'il y a un arrêt pour empêcher la tête d'aller en-deçà de la piste zéro. Ben voyons! C'est faux, et j'ai mis une semaine à le trouver! Quant au formateur de disquette, ce fut l'enfer. Je parie que vous ne saviez pas que la vitesse de rotation des disquettes varie énormément. Allez donc essayer d'écrire un formateur qui compense cela. Une piste doit être très précisément calibrée à 32 microsecondes près; trop long, et vous écrasez les données au début de la piste; trop court, et vous laissez à la fin de la piste des données du précédent formatage, qui reviendront vous hanter plus tard. ("Quoi ? J'ai DEUX secteurs 10 ? C'est impossible!")

Heureusement, Doug est arrivé à GBS à peu près à ce moment-là. Après s'être penché sur ce problème pendant bon nombre d'heures, il a trouvé quelques erreurs très subtiles dans le code et dans le formateur, et il les a corrigées bien après que j'y aie moi-même renoncé. J'en étais arrivé au stade où je ne pouvais même plus supporter d'y jeter un œil. Et il a fallu beaucoup de temps pour que l'envie de programmer me revienne après avoir poussé les choses aussi loin. J'ai essayé autant que faire se peut, de mettre en valeur la contribution de Doug aux dernières versions de Spectre. Ce n'est assurément pas moi qui ai tout fait; Doug y est pour beaucoup. Je ne voudrais plus jamais refaire ce que j'ai fait pour ce projet. Le printemps et l'été 1990 entiers passés à travailler comme un forcené sont pour moi de mauvais souvenirs. J'ai découvert que détourner votre créativité vers n'importe quoi d'autre vous aide à revenir à l'informatique. Par exemple, j'écris parfois sur des sujets qui laissent les lecteurs perplexes. je pousse la part de créativité assez loin. Jouer de la guitare, ou sur un clavier, ou encore sur une Hotz Box, fait également l'affaire pour moi. (Il faudra que je vous parle de la Hotz Box ! Enfin, un génie a inventé la musique orientée objet Hypercard appliqué à la musique est loin d'être une mauvaise analogie...)

C'est comme s'il y avait un instinct créatif profondément ancré en vous qui cherchait à remonter à la surface, un peu comme la lave d'un volcan. Si la sortie "ordinateur" est bloquée, il cherche une autre voie. Par exemple, je peux me mettre à jouer un petit contre-point à une mélodie à la radio, chose que je n'étais jamais assez rapide pour faire avant d'avoir une Hotz Box, sauf en sifflotant. Et finalement, quand tout le reste échoue, vous devez prendre du recul et vous détendre. "Laisser refroidir les moteurs", comme le dit une chanson du groupe Boston. Et vous ne devez pas laisser les gens vous forcer à continuer tant que vous n'êtes pas prêt. A moins que vous ne consentiez à bâcler quelques mauvais boulots, sans y avoir consacré cet indispensable temps de "précharge" durant lequel vous vous cognez aux murs. Ce qui est nécessaire pour en retirer quelque chose de vraiment bon.

Point n'est besoin de le dire, cela peut être très dur pour les gens avec qui vous travaillez. Ils voudraient vous voir au turbin de 9h à 5h chaque jour, "normalement". Mais cela ne marche pas du tout comme ça (NDLR : c'est ben vrai, ça !).

Pendant que j'écris cet article de grandes choses sont en train de se passer. L'épuisement consécutif au Spectre se dissipe, la créativité revient. Quelque chose de nouveau, que je désire depuis au moins dix ans, commence à prendre forme. J'ai des visions de la façon dont un ordinateur devrait travailler - d'une nouvelle manière que je n'ai encore jamais vue. j'y ai déjà fait allusion dans de précédents articles [NdT : Dave veut parler du projet Hyperweb, qu'il a commencé avant de se consacrer ensuite à sa carte SST], mais je ne peux encore le décrire. Mon entourage insiste pour que je le brevète, et je ne pourrais le faire si je bavarde. Mais pour moi, en comparaison, Spectre prend l'air d'un joujou en termes d'impact sur le monde informatique. Je pense sincèrement qu'il pourrait changer celui-ci, et devenir un

élément de base que tout ordinateur devrait avoir très rapidement (je sais, ce n'est quère modeste).

Les idées se bousculent. J'ai subi le stade de l'exaltation à force d'examiner à fond l'aspect que je veux lui donner. A présent, il est temps de serrer les dents. En définitive, ce sera une lutte entre l'ordinateur et moi, à qui sera le plus obstiné, votre serviteur tentant de créer quelque chose suffisamment utile pour faire bouger l'informatique partout.

Bon d'accord, je vise plutôt haut. Mais j'affirme que si vous ne ratez pas un but trop haut placé de temps en temps, c'est que vous ne placez pas la barre assez haut. Ce qui revient à ne pas être assez ambitieux.

#### Conclusion

Je voudrais remercier les nombreuses personnes qui ont pris le temps (qui est la ressource la plus rare) de m'envoyer des messages, de m'écrire des lettres et des petits mots, comparant les performances d'Atari à celles d'autres marques. Je n'ai jamais assez de temps pour répondre au courrier que je reçois; je crois qu'il est temps d'embaucher quelqu'un pour m'assister dans cette tâche (sans compter bien d'autres pour lesquels je manque également de temps).

Il y a beaucoup de choses dont je n'ai pas parlé, mais cet article est déjà trop long. La carte 68030. MegaTalk. La page de dédicace cachée de Spectre 3.0 - oui, la 3.0 marche. L'Atari TT...

Je traîne toutes ces idées d'articles et/ou de programmes qui cherchent à requérir mon attention. Je pense qu'il est temps d'aller se cogner contre les murs, qu'en dites-vous ?

A la prochaine fois!

Traduction : Password (Titre original: "Idling, part II")

### **Guerre dans l'ordinateur**

Pour paraphraser le Victor Hugo qu'on apprenait jadis a l'école, l'article de ce mois-ci aurait pu s'intituler "Tempête sous un capot de bécane". En fait, tous les programmeurs ont un jour connu des frustrations à cause d'un système ou d'un langage informatique. Dave, grand bidouilleur devant l'Éternel, ne fait pas exception, et a d'autant plus souffert qu'il a écrit beaucoup de code. Or, on peut profiter des frustrations d'autrui de deux manières en évitant de refaire les mêmes erreurs, et en se divertissant du spectacle. C'est à joindre ainsi l'utile et l'agréable que nous convie Dave. En démolissant au passage certaines idées reçues...

#### **Avertissement**

Dans certains de mes articles précédents, j'ai fait preuve d'une certaine modération, d'une retenue un peu guindée. Si, si. Des histoires gentillettes, des questions métaphysiques. Je m'amollissais.

Mais cette douceur joviale, quoique fort respectable, peut se révéler lassante. C'est pourquoi ce moisci, j'engage un chargeur, je mets une bastos dans le canon, j'ôte le cran de sécurité, je mets le sélecteur de tir sur "rafale", et j'ouvre le feu! Accrochez-vous, c'est parti!

#### Qu'est-ce que la programmation?

Certaines personnes croient encore que la programmation est un ensemble de principes théoriques rigoureux, décrits en algorithmes approuvés, délicatement écrits en "Pascal structuré", et compilés par quelque pauvre microprocesseur soigneusement tenu en laisse. En tout cas, c'est comme ça qu'on enseigne la programmation. J'en ai été victime lorsque j'ai passé mon Bachelor of Science [NdT: équivalent d'un bac ou d'une première année de licence] dans la section d'informatique d'une grande université. Ce n'est pas du tout ça. Jack Tramiel a dit que « les affaires, c'est la guerre ». Peuh! Les affaires, c'est en donner aux gens pour leur argent. La véritable guerre, c'est la programmation!

Prenons par exemple ma bidouille de l'année 1988, le Spectre 128. Un cliché usé voudrait que j'aie arpenté le gazon d'un campus par une journée ensoleillée, créant les algorithmes voulus dans ma tête, puis que je me sois assis devant un écran dans une salle emplie de musique New Age, et que j'aie frappé le programme en Pascal avec les dix doigts. Eh bien, croyez-moi, cela ne s'est pas passé ainsi. Oh. non.

Le développement du Spectre ressembla plutôt à un combat à la baïonnette. Je me battais contre des circuits intégrés diaboliques, des monstres froids, sans aucun doute communistes, qui faisaient tout ce qui était en leur pouvoir pour ne pas fonctionner.

#### Dave incendie le Pascal

Pour comprendre combien les ROMs du Macintosh sont maléfiques, vous devez d'abord apprendre à connaître le Pascal, car les ROMs ont été en majeure partie écrite en ce langage.

Le Pascal a rampé hors de sa crevasse au début des années 70, et s'est répandu comme une épidémie dans les universités, corrompant l'esprit de milliers d'innocents. Les créateurs du Pascal

étaient épouvantés par les « langages non structurés ». Pour eux, le monde de la programmation était une jungle. Personne n'y suivait aucune règle. Personne ne s'y pliait à aucun standard. Rien n'y était normalisé.

Or, les langages non structurés, comme le Basic, permettaient au programmeur de faire absolument ce qui lui chantait. Faire un saut en tout point du programme, n'importe quand. Passer d'un endroit à l'autre avec moult ruses. S'arranger pour que ça tourne, quoi. « Horreur ! », s'exclamèrent les Pascaldroïdes. En parfaits bureaucrates, ils avaient pour le contrôle absolu un amour à faire blêmir Big Brother. Ils entamèrent donc une campagne pour ôter au programmeur sa liberté de mouvement.

D'abord virent les slogans. « Codage en spaghetti ! » ; « Mauvaises habitudes de programmation ! » ; « L'important, ce n'est pas de faire un code qui marche, c'est d'en faire un qui soit beau ! »... Puis, dans un grand brouillard de confusion mentale, les Pascaldroïdes lâchèrent le Grand Mensonge : vous devez décomposer tout programme en quelques structures de contrôle. Si-Alors-Sinon. Faire-Tant que. Et le terrifiant GOTO fut banni.

Arrivèrent d'autres slogans. « Structures de contrôle » ; « code élégant ». Et le pire de tous : « Code auto-documenté ». Celui-ci est bien le pire. Il suppose que la prochaine personne qui lira votre code est un mutant télépathe, capable de deviner l'usage de votre programme en en lisant le code, pas les commentaires.

#### Première loi de Small:

Il n'y a pas de code auto-documenté. Si vous n'avez pas davantage de commentaires que de code, c'est du boulot de cochon.

Ce que les zélateurs du Pascal n'ont jamais, au grand jamais, pris en compte, c'est la couche de séparation que le Pascal interpose entre l'ordinateur et vous. La vitesse d'exécution des programmes se mit à baisser, parce que les compilateurs Pascal engendraient un code abominable. Vous ne pouviez plus comprendre pourquoi la machine se plantait, sauf si par miracle vous trouviez un bug dans votre programme en Pascal. Vous ne pouviez plus simplement taper RUN pour savoir si votre programme tournait. Non, vous deviez d'abord apprendre à utiliser un éditeur stupide, frapper votre code, puis le faire passer dans un compilateur, et enfin attendre qu'un « interpréteur de P-Code » vif comme une limace daigne exécuter vos idées.

Peu de temps après, le Pascal devint une religion. Les programmeurs furent privé de tout contrôle direct sur leurs processeurs, leurs programmes durent ralentir, et rares étaient les programmeurs qui savaient ce qui se passait au niveau des bits élémentaires. Le Pascal est en cela similaire à la politique stupide de la limitation de vitesse sur autoroute. « C'est pour votre bien » ; « c'est pour votre propre sécurité » !

J'ai entendu ce mensonge bien souvent au collège, je m'en suis échappé juste avant que le Pascal ne devienne obligatoire. A ce moment-là, la section informatique avait sa clique d'activistes du Pascal. Ce langage était imposé aux programmeurs avec des méthodes à faire pâlir un gourou de secte. Si vous ne codiez pas en Pascal, vous n'étiez qu'un pauvre macho, vous n'apparteniez pas à la Nouvelle Vague, vous en étiez resté à la préhistoire. Tenez, j'ai même entendu un apprenti-programmeur me dire que « programmer en Pascal, c'est comme s'imposer de porter une camisole de force ».

Pour bien vous rendre compte de la valeur de cet argument, imaginez-vous en père de famille. Votre fille attend un jeune homme avec lequel elle passera la soirée. Il frappe à la porte. Vous l'accueillez. Vous lui demandez ce qu'il fait dans la vie. Il répond: « je m'impose de porter une camisole de force ». Vous le chassez en pensant « quel abruti ! », pas vrai ?

#### Deuxième loi de Small:

En vérité, on peut écrire du bon ou du mauvais code dans n'importe quel langage. On peut appliquer des règles de structure dans n'importe quel langage.

#### Troisième loi de Small:

Tout programmeur assez stupide pour s'imposer une camisole de force la mérite

amplement.

Le Pascal ne réussit à s'imposer que grâce au conformisme le plus servile. Mais peu à peu, la vérité quant à ce langage commença à filtrer, en de discrets courants souterrains. Attention, si vous leviez la tête en disant « Eh bien moi, j'aime le Basic », les molosses du Pascal vous submergeaient en hurlant. Mais une saine réaction alimentait des murmures : « Pascal est si inefficace que mes programmes mettent un siècle à s'exécuter » ; «Il y a des choses qu'on ne peut même pas imaginer en Pascal, comme mélanger les types, alors que j'en aurait sacrément besoin! » ; « Eh, vous avez vu ? Mon petit programme en Pascal donne après compilation un exécutable de 100 Ko de long! ».

La triste vérité, c'était que le Pascal était un langage mal conçu. Toutes les professions de foi, tous les bêlements affirmant que le Pascal est bon pour vous, butaient contre le mur des réalités : un programmeur codant en assembleur pouvait écrire un logiciel qui ridiculisait en vitesse le même logiciel écrit en Pascal, tout en étant beaucoup plus court. Pire : un programmeur pouvait se permettre en assembleur des choses à jamais impossibles en Pascal.

C'est pourquoi les courants souterrains démarrèrent chez les gens assez malins pour voir que le Pascal, c'était du bluff. Pour les gens qui n'avaient pas peur de faire valser leur machine à pleine vitesse, l'assembleur s'imposa. Regardez plutôt :

- ① Lotus 1 -2-3 est écrit en assembleur [NdT : du moins les premières versions] ;
- Wordstar est écrit en assembleur ;
- ① dBase est écrit en assembleur :
- ① Pratiquement tous les jeux d'arcade sont écrits en assembleur.

Me fais-je bien comprendre? Les programmes qui ont fait la fortune de leurs auteurs sont écrits en assembleur. Ce sont des programmes remarquables par leur vitesse, qui faisaient de votre ordinateur une voiture de compétition, qui lui faisaient chausser des bottes de sept lieues. Ces programmes se sont bien vendus parce que leurs utilisateurs aiment leur apparence, leur « look-and-feel ».

Les posters de Samantha Fox ne se vendent pas parce qu'elle est « structurée ». Ou « élégante ». Encore moins parce qu'elle « s'impose une camisole ». Mais parce qu'elle est sexy!

#### Alors vinrent les années 80

Venons-en aux années 80. Les gens commencèrent à abandonner le Pascal en masse : ses limitations étaient vraiment insupportables. Émergea alors le C, un Pascal qui était en fait efficace et rapide et vous laissait faire ce que vous vouliez. C'est pourquoi sa popularité croît sans cesse actuellement.

Sur ce, Apple débarque. Apple aime le Pascal. Et même, Apple adore le Pascal. Et crée un ordinateur nommé Lisa.

Quand une personne allumait un Lisa, dont le système d'exploitation était écrit en Pascal, elle avait toujours la même réaction : pourquoi est-ce si long ? Pourquoi cette machine est-elle si lente ? Et surtout, comment ont-ils fait pour ralentir autant un 68000 ?

L'échec du Lisa fut le signe qu'il était temps de faire le Macintosh. Les routines d'affichage graphique étaient écrites en Pascal, puis optimisées en assembleur là où il y avait des goulots d'étranglement. Elles occupaient finalement 60% de la ROMs du Mac. Quant au système d'exploitation, il fut en grande partie réécrit en assembleur, pour l'accélérer.

Mais hélas ! la machine gardait un affreux parfum de Pascal. Il fallait la supplier d'attribuer de la mémoire pour y charger un programme. Il fallait lui plaider votre cause pour qu'elle ne sorte pas le programme de la mémoire une fois que vous l'aviez chargé. Votre programme était déplacé dans la mémoire sous vos pieds à intervalles plus ou moins aléatoires, et le tout était justifié dans la

documentation du Mac, sur un ton vertueux et compassé, comme étant « bon pour vous ».

Vous ne pouviez pas faire les choses simplement, comme PRINT "BONJOUR". Non, il vous fallait d'abord initialiser le gestionnaire de mémoire, le tas, et QuickDraw, puis ouvrir un GrafPort, choisir une police, choisir sa taille, ses caractéristiques (gras, italique, etc.), lancer une boucle de traitement d'événements, lire un fichier ressource contenant le texte « BONJOUR », et peut-être alors pouviez-vous espérer afficher « BONJOUR ». Il fallait neuf pages de Pascal pour faire un affichage simple.

Peut-être avez-vous entendu dire que le Mac est la machine la plus difficile à programmer. A mon humble avis, c'est à cause de son héritage pascalien. Le Pascal a obligé les programmeurs à faire les choses d'une seule manière, selon le vieux rêve des gardes-chiourme qui veulent à tout prix standardiser les programmes. Rappelez-vous Nikita Khrouchtchev martelant la tribune de l'ONU de sa chaussure, du genre « moi veux tout le monde il faire comme moi décide ! », et vous aurez un bon aperçu de la façon dont les comités de normalisation aiment à contrôler les programmeurs.

La plupart du temps, cette contrôlite aiguë finit par rendre les programmes lourds et lents. Et c'est ce que le Mac force les programmeurs à faire. C'est pourquoi on lit si souvent dans la presse que tel logiciel pour le Mac est « retardé de 3 mois ». Ou « encore instable ». Ou que « ce produit a été abandonné ».

#### Dave descend en flammes les ROM du Mac

Prenez donc les ROM, ces monstres froids, avec cet héritage dictatorial, et insérez-les dans un ST, comme je l'ai fait en 1985 (c'était facile à cette époque, il suffisait de les enfoncer dans les supports prévus pour les ROMs du TOS, vu que le TOS était encore sur disquette). Penseriez-vous que les ROMs aiment cela ? Certainement pas. Elles se mettent à agir exactement comme un agent du KGB qui a été kidnappé et forcé à vivre aux États-Unis. Les ROMs jettent un œil sur le bus du ST. Il y a là un bon disque dur bien rapide. Il y a aussi un bon affichage vidéo monochrome, conçu proprement, pas un de ces bricolages qui vous ralentissent un processeur. Et même un lecteur de disquette tout à fait correct. Croyez-vous que les ROMs en soient satisfaites ? Pas du tout ! imaginez un agent du KGB contemplant un supermarché d'un air dégoûté, et vous aurez une approximation. L'agent est horrifié par le manque de normes, l'absence de contrôle... Tenez, le port de l'uniforme n'est même pas obligatoire ici !

Du coup, les ROMs déclarent la djihad. Leur but, dans la vie, devient le Plantage. En bref,elle se montrent si inaptes à s'adapter à la vie dans le ST qu'elles passent un pacte de suicide mutuel. Elles veillent tard dans la nuit, complotant dans l'ordinateur des machinations ignobles pour mieux planter. Glissons une oreille :

- Camarade Lobachevsky! Au rapport!
- Da, camarade officier !... Les lecteurs de disquettes semblent inattaquables. Par contre, la routine de formatage des disquettes n'est pas très robuste, si bien que nous pourrions planter là. Il y a aussi de bonnes possibilités du côté des circuits de l'interface série.

Le lendemain, quelqu'un essaya de formater une disquette en mode Mac sur ce ST. Vlan! L'écran du « Mac planté » apparut. Les ROMs poussèrent un soupir de soulagement béat. Plus tard, quelqu'un tenta de faire tourner un programme de communication qui s'adressait directement aux circuits de l'interface série. Plantage.

J'ai bien essayé de menacer ces ROMs. Je les ai reliées à une prise électrique commandée par un interrupteur, et, le doigt sur le bouton, j'ai dit : « Obéissez, ou sinon... ». Dans un ricanement méprisant, elles m'ont répondu : « Vas-y, appuie, sale impérialiste ! On va te montrer comment savent mourir des ROMs écrites en Pascal !... ».

Comme ça ne marchait visiblement pas, j'ai commencé à dresser des barrières autour des ROMs via le logiciel, en limitant les dégâts qu'elles pouvaient causer par leurs sabotages.

- Camarade Lobachevsky ! Au rapport ! Vous nous aviez assuré que vous alliez planter cette machine aujourd'hui !
- Je... je ne comprend pas, camarades. Aujourd'hui, le port série nous a été retiré. J'ai essayé toutes les

fonctions, et elles marchent toutes !...

Le camarade Borodine, un homme à l'air mauvais, prend la parole : « J'ai une solution. Beaucoup de programmes Mac écrivent accidentellement à l'adresse zéro. Sur cette machine, cela provoque un plantage. Ainsi, le système ne sera jamais stable».

La réunion se termine alors dans de grands éclats de rires. Le lendemain, je fis tourner Excel de Microsoft. Plantage. A cause d'une écriture à l'adresse zéro.

Là, j'eus bien peur d'être vraiment coincé. Le manuel du 68000 de Motorola est clair : une écriture à l'adresse zéro provoque une erreur de bus, et on ne peut pas rattraper une erreur de bus. En fait, pour pouvoir le faire, il faut un 68010 ou mieux. (Ecoutez le tuyau, les étudiants en informatique ! Le rattrapage des erreurs de bus, c'est la manière d'implémenter la mémoire virtuelle et la pagination sur disque).

Je refermai soigneusement le manuel Motorola, et j'essayai quand même une bidouille pour tenter de rattraper les erreurs de bus. Et vous savez quoi ? Le manuel avait tort! Je continuai et écrivis du code pour que la machine ignore les erreurs de bus. Ce ne fut pas chose aisée, mais ça marchait.

A présent, les ROMs se réunissent, pétrifiées d'horreur :

- Camarade Borodine! Nous avons fait aujourd'hui cinquante écritures à l'adresse zéro, et il n'y a eu aucun plantage! Gardes!... Emmenez Borodine au peloton d'exécution!... (*Bruit de rafales d'AK-47*.)

Cela a duré quelque temps. Au fur et à mesure que les ROMs découvraient de nouvelles façons de planter, j'installais de nouvelles barrières, de manière à l'interdire. Et tandis que les programmes Mac essayaient un tour de cochon après l'autre, créant des problèmes qui réussirent à planter même de vrais Mac, je trottinais derrière avec mon ramasse-crottes pour nettoyer les débris.

Le Spectre 1 28 contient ainsi des zones démilitarisées, des barbelés, des champs de mines, des pièges, des alarmes... entre autres. Les ROMs sont quasiment enfermées dans un goulag. Bien sûr, elles élaborent toujours des plans démoniaques. Elles ont trouvé un moyen de saboter l'option « prévisualisation de page » dans Microsoft Word3.02. Mais tôt ou tard, je découvrirai ce moyen, et je lui enfoncerai un pieu de bois dans le cœur.

Cela a été une lutte continuelle, une guerre intense. J'ai perdu bon nombre de batailles, et il y a toujours des choses qui refusent de marcher, mais finalement, j'ai gagné la guerre.

#### Conclusion

Voilà, c'est ce dont je voulais vous parler. Une critique du Pascal à faire faire des apoplexies aux amoureux de ce langage. La vérité sur ce qui se passe la nuit dans les réunions secrètes des ROMs du Mac. Et un aperçu sur l'étrange psychologie tordue d'un programmeur en assembleur.

J'apprécie toujours les remarques et commentaires. Si vous voulez bien me faire part des vôtres, exécutez le programme suivant pour obtenir mes adresses télématiques :

IF {LANGAGE(Favori) ==PASCAL} &AND& {EMOTION ==INDIGNATION} DO Adresse CompuServe = 00000,000; Adresse GEnie = Nil; Adresse Internet = ping.pong
ELSE
 AdresseCompuServe =76606,666; Adresse GEnie = DAVESMALL; Adresse Internet = dsmall@well.sf.ca.us
ENDIF

Traduction et adaptation: Password Titre original : "War inside the computeur"

### Le feu sacré

Dave nous entretient ce mois-ci d'un sujet qui touche à la haute technologie mais est très controversé la recherche spatiale. Tout le monde reconnaît que l'informatique est un outil indispensable. Mais l'unanimité est loin de régner en ce qui concerne l'espace. Et pourtant, la recherche spatiale engendre sans doute autant de retombées intéressantes que l'informatique. De plus, Dave connaît des gens qui s'ingénient à prouver que non seulement on peut privatiser une partie des activités spatiales, mais qu'en plus on n'a pas besoin de moyens colossaux pour atteindre l'orbite terrestre. Ces bidouilleurs de l'espace, Dave les a naturellement rencontrés dans un salon de la bidouille, loin des administrations et de leurs bévues. Car les autorités, comme la NASA et le FBI, semblent faire mauvais ménage avec les "hackers"...

#### Vous vous rappelez?

Chacun d'entre nous a eu des rêves, étant enfant. Des désirs, des buts insensés qui donnent forme à une existence entière, bien que le plus souvent, on ne puisse hélas les réaliser. Des rêves catalysés par une rencontre, par un événement. Vous vous rappelez les vôtres ? Moi, oui. Je ne pourrai jamais les oublier. Ils forment un jardin secret profondément enfoui en chacun, et comme tel, il est très difficile d'en parler. Une grande part de ce qui constitue le Dave Small actuel est construit autour d'eux. Mais un événement récent me pousse à vous en parler aujourd'hui.

Mais d'abord, retour en 1972. Je devais être en quatrième. J'étais allé voir un film de science-fiction nommé "La tache venue d'Andromède". Le film commença. Il montrait, à ma grande surprise, un personnage en robe cueillant des légumes dans un potager. Hein ? Et mes lasers, mes soucoupes volantes ? Puis le personnage entra dans une cuisine, commença à laver ses légumes, et ouvrit une fenêtre pour regarder au dehors. Le volet de métal qui occultait la fenêtre (hein ?...) s'abaissa lentement, découvrant le vide spatial d'un noir intense constellé d'étoiles brillantes. La caméra bascula en vue de l'extérieur de ce hublot, s'éloignant en travelling arrière, en même temps que commença la musique... Et bon sang, mais il était dans un vaisseau spatial!

Vaisseau parsemé de nombreux hublots, dont celui de cette cuisine, et le potager était sous un des énormes dômes géodésiques, et le vaisseau était en orbite autour de Saturne d'après les anneaux, et...

C'était si inattendu que j'en fus époustouflé. Un enthousiasme, une excitation que je n'avais jamais connus m'envahirent jusqu'au tréfonds de mon âme. J'en fus changé à jamais. J'avais découvert mon rêve secret: l'espace.

Durant quelques minutes, la caméra détailla le vaisseau géant, en comparaison duquel celui de "2001, l'odyssée de l'espace" ressemblait à une miniature, puis retourna à la cuisine et à l'intrigue. Je découvris que j'avais oublié de respirer pendant tout ce temps et que les mains me faisaient mal tant j'avais serré mes accoudoirs. Il y avait de quoi : voilà ce que je voulais, voilà ce pour quoi j'aurais tout donné! Aller là-haut, contempler la terre, voir le soleil se réfléchir sur les océans, chercher les USA dans l'hémisphère nord... Pour la première et unique fois, je savais quel sens avait ma vie.

Vous rappelez-vous pareille émotion, ami lecteur ? Vous souvenez-vous d'avoir un jour été empoigné par une force qui vous a montré votre plus cher désir, votre but ultime ? Avez-vous la chance de travailler dans ce but, à alimenter votre feu sacré ?

Ou peut-être avez-vous abandonné ce "rêve de gosse" et passez-vous votre vie à faire quelque chose qui ne vous motive pas, qui vous ennuie et vous désespère à la seule idée de devoir vous lever le matin pour cela...

Quant à moi, j'ai découvert l'automne dernier que j'avais oublié mon rêve, ce qui m'a presque déchiré.

Je suis content d'écrire cela au lieu d'en parler, car cette découverte me pèse et m'étouffe. En fait, j'ai dû attendre quelques mois et prendre du recul avant de pouvoir l'écrire.

"Que s'est-il passé ?", demandez-vous. Le choc qui m'a bouleversé est parti de retrouvailles à une convention annuelle.

#### HackerCon 7.0

Chaque automne depuis six ans, j'ai le privilège de participer à une convention nommée Hackers Conférence (en abrégé HackerCon), la conférence du bidouilleur. C'est une réunion, exclusivement sur invitation, où environ deux cents personnes, considérées comme les leaders de l'industrie informatique, se rencontrent durant deux jour dans un lieu éloigné de tout. Et, en fait de rencontre, certains résultats sont semblables au mélange de matière et d'anti-matière, d'une puissance qui dépasse toute attente.

Oh, certes, il y a des groupes de discussions techniques. Mais c'est principalement une occasion de rencontrer des gens dont on parle dans la presse. J'ai ainsi rencontré des gens comme John Draper (génie de la téléphonie), Todd Runtgren (musicien), Ed Rotberg (auteur de Battlezone), Nolan Bushnell (fondateur d'Atari), Jay Miner (concepteur des circuits des Atari 8 bits et de l'Amiga) et Jimmy Hotz (musique orientée objet). Toutes sortes de cartes de visites s'échangent là, calmement, et bien des choses s'y déroulent, à l'abri de la presse je suis très discret au sujet de mes articles sur HackerCon!).

Tom Hudson, auteur des splendides CAD-3D et Degas sur nos machines, reçut des applaudissement debout pour l'animation qu'il a montré. Il a écrit le magnifique logiciel Studio-3D pour AutoCAD sur PC, avec Gary Yost et Jack Powell, anciens d'ANTIC Publishing [NdT ancien éditeur du magazine ANTIC et de logiciels pour Atari, qui tire son nom d'un circuit intégré des Atari 8 bits].

#### Dave et le F.B.I.

Et attention, la Conférence est une réunion sérieuse, pas un week-end passé à bronzer. Si sérieuse, même, qu'il m'a fallu aller m'expliquer avec le FBI au sujet de HackerCon 5.0 ! En effet, quelqu'un, qui avait pour pseudo "NuPrometheus League", avait à cette époque, il y a deux ans, volé le code source des ROM d'Apple, et un couillon avait dit au FBI que ce quelqu'un avait certainement participé à la HackerCon

[NdT : le nom NuPrometheus - le nouveau Prométhée - est calqué sur le nom NuBus, le bus d'extension utilisé par Apple.]

En fait, tous les participants à HackerCon 5.0 hors de la région de San Francisco furent contactés par le FBI. Et croyez-moi, il n'est pas très rassurant de trouver en rentrant chez vous un message sur votre répondeur téléphonique vous disant de rappeler le FBI pour prendre rendez-vous Je suis néanmoins allé les voir.

Je crois que c'est moi qui ait été une surprise pour le FBI. Bon, bien sûr, ils m'avaient fait peur. Mais je fus coopératif, et j'expliquai précisément ce que sont les bidouilleurs, à savoir des gens qui cherchent à "pousser l'enveloppe", comme les pilotes d'essais, les grands programmeurs, ou les leaders de l'industrie. Je mentionnai des gens ayant participé à la Conférence qui sont des plus respectables je ne cafardais pas, ils avaient la liste des invités). Puis je montrai au FBI ce que contenaient de vrais serveurs télématiques anarchistes réellement dangereux ("Improvisez du plastic dans votre cuisine"), et je leur ai laissé un listing de serveurs pirates et de leur contenu, en les priant de les faire fermer. A leur tour, ils parlèrent et m'expliquèrent qu'ils manquaient de personnel pour courir après les pirates, et des sommes qu'il fallait mettre en jeu avant qu'ils ne puissent intervenir. Nous discutâmes des heures.

A ce que je sais, après avoir rassemblé des rapports, le FBI conclut que s'en prendre aux invités de la HackerCon était comme harceler le moteur qui anime l'industrie informatique. Et ils abandonnèrent ce

tuyau percé. Cependant, une furieuse activité naquit chez les bidouilleurs, en raison de ces interrogatoires, certes, mais également à cause de l'opération Soleil du Diable, une bévue ahurissante que la presse décrivit comme une rafle contre les pirates informatiques, mais qui fut en fait réalisée de façon aberrante, sans discernement, et parfois illégalement (bien sûr, les médias ne daignèrent pas rapporter les faits exacts).

Durant cette opération, les services secrets prirent d'assaut les bureaux de la firme de jeux Steve Jackson Games, qui préparait un jeu de rôles se déroulant dans un monde futuriste de "cyberpunks" vers l'an 2100. La firme faillit bien en être anéantie: les services secrets confisquèrent leurs équipements informatiques (traitement de texte, etc.) et les gardèrent très longtemps. La raison ? Le manuel du jeu expliquait que les personnages joueurs devaient s'introduire dans des ordinateurs. On jetait un dé, un 1 indiquait une réussite, sinon c'était un échec. Si Si, je passe mon temps à pénétrer par effraction dans des ordinateurs en jetant des dés (Ben voyons! Des façons de m'introduire dans des systèmes, j'en connais qui glaceraient les sangs de certains constructeurs informatiques, mais aucune ne fait appel à un dé - ni à moi, d'ailleurs.)

Le plus effrayant pour nous tous, c'est que les agents prirent d'assaut la firme de jeux de bon matin, fusils à pompe en main, avec, tenez-vous bien, des mandats de perquisitions non signés, et alignèrent face au mur les employés tandis qu'ils confisquaient tout ce qu'ils trouvaient.

Je me demande s'ils ont découvert le dé. En tout cas, c'est une violation caractérisée des droits civiques.

Quand le groupe de la Conférence apprit cela, ses membres explosèrent.

Les messages fusèrent sur les réseaux électroniques. Un certain David Small se connecta sur The Well [NdT : *réseau télématique*] et appela à l'aide : "Au secours Le FBI veut me voir au sujet de HackerCon 5.0 Que diable se passe-t-il ?" J'appris ce qui s'était passé, non sans soulagement. Et à présent, il y a la Electronic Frontier Foundation (EFF) pour défendre les droits des gens en matière télématique, en partie à cause de ces entretiens avec le FBI. L'EFF, dont l'un des co-fondateurs est Mitch Kapor (auteur de Lotus 1-2-3), aida entre autres Steve Jackson Games. Faute de cette aide, les seuls frais de justice auraient coulé cette firme. J'ai rencontré Steve Jackson à HackerCon 6.0, il n'a rien d'un pirate (Electronic Frontier Foundation - 155 Second Street - Cambridge, MA 02141 - USA - Télécopie : (1) 617 864-0886 internet : eff.eff.org. C'est là une bonne cause s'il en fut jamais.)

Vous auriez dû entendre les participants parler de leurs entretiens avec le FBI. A peu prés la moitié du groupe y avait eu droit. Jerry Pournelle [NdT : journaliste du magazine américain Byte et auteur de SF], qui est politiquement aussi à fond à droite qu'on peut l'être, était furieux de ces violations des droits civiques, et a décerné dans Byte son "prix oignon de l'année" [NdT : la Gamelle d'Or, en quelque sorte] à l'agent des Services Secrets qui avait concocté cette gaffe.

Mais je suis au regret de vous dire que certains invités ont eu assez peur pour ne plus jamais vouloir revenir à HackerCon.

#### Retour aux bidouilleurs

L'un des attraits de la Hackers Conference est le buffet aux friandises, où sont servis tous les amusegueule qu'on peut manger pendant un débogage nocturne, depuis le soda jusqu'aux barres de confiserie.

J'étais donc à côté du saladier de chips, prenant un peu de sauce à l'avocat [NdT: beuh NDLR: Ho lui !] - c'est vous dire comme tout cela est encore gravé dans ma mémoire lorsque mon regard tomba sur le badge portant le nom de la personne se trouvant de l'autre côté de la table. C'était un homme avec une légère barbe qui avait l'air un peu perdu des nouveaux venus à la Conférence, qui cherchent à rompre la glace (les nouveaux venus ont toujours un choc, mais après ils plongent). Ce badge disait : "Gary C. Hudson, Pacific Amencan Launch Systems". Oh mon Dieu, me dis-je, mais je connais ce nom!

Et tout ce que j'avais perdu il y a si longtemps, depuis mes treize ans en classe de Quatrième, ce que j'avais oublié, tout cela me revint d'un coup, car j'avais en face de moi l'Homme aux Fusées. Je n'osais

tout d'abord pas lui adresser la parole, car non seulement j'avais rencontré l'un des plus grands héros qui soient, mais de plus mes rêves perdus me revenaient et me faisaient apprécier la vie avec une force que j'avais cru oubliée... Je ferais peut-être mieux de m'expliquer, hein ?

#### Un petit historique de la NASA

Au début des années 60, une organisation nommée NASA fut formée pour regrouper les différents efforts d'exploration spatiale entrepris par divers organismes, dont principalement l'Air Force (le X-15, vous vous rappelez ?). John Kennedy confia à la NASA la tâche d'envoyer un homme sur la Lune avant la fin de la décennie.

La NASA était une agence gouvernementale et n'apprécia vraiment pas cela. A nouveau, laissez-moi vous expliquer. Ayant travaillé pour différents contrats gouvernementaux et avec des amis employés au gouvernement, je peux vous assurer que le but de la NASA n'était pas d'envoyer un homme sur la Lune. Le but, c'était que la NASA ait encore du travail l'année suivante. Ce qui signifiait qu'il fallait faire traîner les choses.

Notez qu'ils sont corrects en terme de contrats gouvernementaux. Après juillet 1969, quand l'excitation fut dissipée, la NASA perdit beaucoup d'emplois en raison de coupes budgétaires décidées par des gens qui n'y voyaient pas plus loin que le bout de leur nez, comme cet idiot de sénateur William Proxmire [NdT: sénateur célèbre pour ses critiques acerbes contre les dépenses gouvernementales, critiques souvent justifiées, il faut bien le dire].

Je suis navré de le dire ainsi, car l'ai vécu et j'ai rêvé une grande aventure grâce à la NASA, moi aussi, dans les années 60, me levant à 4 heures du matin pour regarder les lancements des Gemini et Apollo, la montée de la fusée dont les lettres peintes sur le fuselage épellent "UNITED STATES", pour voir Neil Armstrong sur la Lune, Jack Swigert dans Apollo 13 luttant pour survivre après l'explosion du réservoir d'oxygène, pour voir Deke Slayton finalement revenir intact...

...pour voir ensuite une fusée Saturn-V, prête à s'envoler avec un équipage, le plus puissant engin jamais bâti par l'homme, que la NASA a exposé couchée sur le côté à Houston et jeté en pâture aux touristes voir les interminables années 70 et 80, durant lesquelles Skylab cafouilla puis tomba et brûla parce que la NASA ne pouvait pas le maintenir en orbite, durant lesquelles nous n'avions plus de missions lunaires, nous lançâmes sans pré-vérification le télescope spatial Hubble (allons, secouez-vous!), qui échoua naturellement...

...jusqu'au jour présent, où le programme de la navette spatiale, d'une technologie datant des années 60-70, a connu ses fameux déboires, et où des gens sont morts. L'an dernier, avant le salon WAACE, nous avons visité le cimetière d'Arlington où reposent certains astronautes de Challenger et avons versé des larmes avec beaucoup d'autres gens.

Je n'ai plus confiance en la NASA. Jamais je ne pourrais un jour aller en orbite et atteindre mon but Si je m'en remet à la NASA. Voyez leurs exploits ! (Si vous croyez qu'Atari a des problèmes, regardez les leurs.)

La NASA et la firme Morton Thiokol [NdT constructeur du booster à poudre dont un joint déformé par le gel a causé l'explosion de la navette Challenger] ont passé outre les avertissements des ingénieurs, c'est-à-dire ceux qui savaient, quant à la température minimale de sécurité pour les joints toriques, afin de précipiter le lancement de Challenger pour des raisons essentiellement politiques. Voyez-vous, il y avait à bord une institutrice qui devait le lendemain donner à la télé un cours retransmis en direct dans toutes les écoles de la nation, et il ne fallait pas qu'elle ait un jour de retard. Alors ils ont quand même lancé Challenger.

Quand on pense que quelques lancements auparavant, un booster faillit se consumer en vol, tandis que le moteur principal de la navette était tombé en panne durant la montée... Et la NASA avait ignoré tout cela!

Durant les essais au sol d'Apollo I,ils avaient enfermé Gus Grissom et deux autres astronautes dans la capsule remplie d'oxygène pur (ce qui est inexcusable) [NdT : mais justifié par le gain de poids par

rapport à l'air], et un court-circuit a fait brûler vifs trois hommes valeureux qu'on ne put faire sortir parce qu'il fallait trente minutes pour démonter l'écoutille. La NASA a commis bien d'autres bévues à faire pâlir d'horreur. Certaines n'ont été rattrapées que grâce aux réactions rapides de pilotes d'essais et d'équipes au sol héroïques. On se rappelle Apollo 13, dont l'équipage survécut miraculeusement à l'explosion de son réservoir principal d'oxygène à ml-course vers la Lune, à cause d'une erreur d'un sous-traitant que la NASA avait laissé passer... Quant à la mission Apollo il, première à atteindre la Lune, elle n'a pu alunir que parce que Neil Armstrong, pilote d'essais depuis des années, regarda vers le bas au dernier moment, constata que le site vers lequel l'ordinateur de bord menait le LEM était semé d'énormes rochers, et passa~en commandes manuelles.

La commission d'enquête présidentielle sur Challenger a découvert que la conception de la navette était contestable sur de nombreux points, là où la politique ou le budget l'avaient emporté sur les lois de la physique ou le simple bon sens. Ainsi, les pneus éclataient régulièrement à l'atterrissage. Les freins étaient douteux. Les turbo-pompes qui alimentaient les moteurs principaux fonctionnaient sans marge de sécurité. Et l'équipage n'avait aucun moyen de s'éjecter!

Et je dois ajouter un détail très pénible qui a été soigneusement étouffé : l'équipage de Challenger a survécu quelque temps après l'explosion. Au moins un membre de l'équipage passa son masque à oxygène de secours, mais la cabine tomba dans l'océan et il y périt noyé.

Vous vous souvenez de Saturn V, ce puissant engin qui expédia une lourde charge jusqu'à la Lune ? Eh bien, la NASA a égaré - je ne plaisante pas - la documentation absolument indispensable sur la séquence de mise à feu des moteurs multiples, séquence dont la parfaite synchronisation fait la différence entre un beau lancement et une boule de feu. La nation la plus puissante du monde ne sait plus comment on allume une Saturn ! Parmi mes amis, de vieux employés cyniques de la NASA disent que la documentation a été volontairement jetée pour préparer la voie à la navette, ce qui revient à brûler les ponts derrière soi.

Quant à l'informatique de pointe de la NASA... La navette utilise des mémoires à tores dans ses trois ordinateurs principaux. Vous vous souvenez de cette cochonnerie ? Vous pouvez essayer d'en trouver dans des musées, ou dans des films des années 50 sur les ordinateurs et les robots. Un tore magnétique stocke un bit dans un petit anneau de ferrite et servait de mémoire principale avant que la RAM ne soit disponible en grandes quantités. De plus, la navette utilise trois ordinateurs pour prévenir une panne de l'un d'eux. Un tout petit calculateur sélectionne la décision majoritaire à deux contre un Si les trois machines sont en désaccord. Je suis stupéfait qu'IBM continue à maintenir un ordinateur Si vieux.

Et la station spatiale Freedom ? Mon meilleur ami s'y intéresse et observe que le projet est lentement annulé, morceau par morceau. La mission vers Mars est en train de couler avec. Honnêtement, Si nous avions un brin de cervelle, nous achèterions la station spatiale russe Mir, qui est toujours en orbite et toujours munie d'un équipage à ce jour. Au moins, elle marche [NdT: justement, la Russie cherche à vendre certaines technologies spatiales].

Au cas où vous n'auriez pas compris, j'ai des griefs contre la direction de la NASA, pas contre les excellents ingénieurs qui accumulent des réussites spectaculaires en dépit de leurs chefs. En fait, des cosmonautes russes en visite à la NASA ont examiné des plans de Freedom et ont signalé des erreurs qu'ils avaient déjà eu à subir. Comme par exemple les panneaux coulissants. Qui dans l'espace ne coulissent plus, mais adhèrent. Sans doute comprenez-vous à présent pourquoi j'ai perdu confiance en la NASA, et pourquoi je retiens mon souffle chaque fois qu'ils lancent une navette.

#### Les efforts de Gary

Et là, de l'autre côté de la table, se trouvait un homme qui avait de toutes ses forces tenté d'atteindre l'espace depuis 20 ans, un constructeur indépendant de fusées. C'est un ingénieur très qualifié, consultant pour de nombreuses organisations aérospatiales. Et il a fait ce qui pour moi est la caractéristique du bidouilleur : il a quand même essayé! Crénom, vous n'êtes pas un bidouilleur tant que vous n'avez pas tapé "GO" pour voir ce que ça fait, que ça marche ou non. Et comme le savent les

gens qui ont visité Gadgets By Small, c'est sans importance si ça rate, et ça rate souvent. Vous réessayez. Ce n'est qu'ainsi qu'on arrive à surmonter l'impossible. (Nous avons une scie ici : "Et alors ? Qu'est-ce que ça va faire ? Rater? La belle affaire !") Et le but est à présent Si terriblement proche...

Gary Hudson a perdu une fusée dans laquelle lui et 25 de ses collaborateurs avaient mis tout leur cœur et leurs tripes. Durant les tests de moteur, elle a fait un "mauvais départ", c'est-à-dire que le combustible, au lieu de s'enflammer instantanément dans la tuyère, s'est accumulé et a explosé. Les morceaux de ferraille résiduels ne faisaient guère plus de 50 cm, raconte Gary. (Interrogez à la NASA sur les mauvais départs, sur le nombre de fusées qu'ils ont perdu avant d'oser y mettre quoi que ce soit de vivant. Les gens d'ici qui se rappellent Spoutnik et les années 50 se souviennent bien de la phrase "Les nôtres explosent toujours!") Je n'ai jamais quant à moi subi quelque chose d'aussi affreux que cette explosion. Ce serait comme planter un programme durant les tests et perdre tout le code source en même temps.

Sans compter qu'après, Gary a dû affronter l'émission de TV "Nightline" où le journaliste Ted Koppel l'a interrogé sur l'état des firmes privées de lancements spatiaux. Et Ted Koppel fait ressembler l'interrogatoire du FBI que j'ai subi à un thé mondain. Un vrai baptême du feu!

Mais Gary a essayé, s'est obstiné. Il a conçu les engins Phoenix, qui sont des lanceurs réutilisables employant une technologie saine des années 80-90, qui peut monter en orbite et redescendre sur Terre aussi souvent que vous le souhaitez... Et il a découvert ce que je vous ai dit au sujet de la NASA lorsqu'il leur en a parlé. Intérêt zéro.

Phoenix utilise un concept dit SSTO (Single Stage To Orbit, mise en orbite avec un seul étage), parce qu'il ne gaspille pas plusieurs étages à chaque lancement. Après l'atterrissage, on fait le plein et on repart (bon, c'est un peu plus complexe, mais vous saisissez l'idée).

Mais voilà, Phoenix ne coûte pas les sommes énormes que coûte la navette, et Gary ne craint pas de perdre son boulot après avoir atteint ses objectifs. Ce n'est donc décidément pas un projet pour la NASA. Et vous pouvez être certain qu'il n'y a pas de mémoires à tores échappées des années 50 dans Phoenix.

Gary et quelques autres, Phil Bono par exemple, ont fait vivre un rêve durant bien des années de solitude, les années 70 et 80, où presque personne ne les écoutait. Oh, certes, les écrivains compétents connaissaient Gary Hudson. Jerry Pournelle, qui a travaillé à la NASA pendant des années, lui a dédié un livre, et en a même fait un personnage jouant un rôle important dans son dernier livre, "Fallen Angels", qui comprend bien sûr une fusée Phoenix. [NdT toujours aucune traduction des romans de SF de Pournelle n'est prévue en France, à cause, m'a t-on dit, de son agent littéraire.]

Le chiffre 7 est supposé porter chance. En tout cas, pour moi, il l'a fait, puisque c'est à HackerCon 7.0 que j'ai rencontré Gary. Je me suis enfin rappelé le sens de ma vie,après vingt ans d'oubli. J'ai fini par rassembler mon courage et suis entré en conversation avec Gary, qui est très sympathique. Il a un regard particulier que je n'ai vu auparavant que dans le regard du père de ma femme Sandy, un regard de pilote d'essais. (Le père de Sandy a piloté le prototype YF-12 et les SR-71 Blackbird. Sandy a grandi à la légendaire base Edwards de l'Air Force. Son père a participé aux tests du SR-71, ce qui signifie qu'il était de ceux qui poussèrent l'appareil à ses limites pour voir jusqu'où il pouvait aller. Avec l'avion le plus rapide du monde, et celui qui monte également le plus haut, ce n'est pas chose aisée. Nous lui avons dédié la carte 68030 SST.) Et dans ce regard, j'ai vu qu'on ne pouvait pas arrêter aisément Gary. C'est une force de la nature.

Nous avons parlé longtemps, et durant ce temps, la vieille flamme d'enthousiasme que je venais de ranimer n'a fait que grandir. D'autres personnes nous ont rejoint, principalement des fanas de l'espace, qui ont offert leurs idées sur l'aide qu'ils pourraient apporter. J'ai décidé d'y contribuer, afin que l'on sache qu'il y a d'autres voies d'accès à l'espace. La conversation a continué pendant le dîner et une bonne partie de la soirée, discutant de ce la façon dont certains bidouilleurs pourraient apporter leur aide. (Nous nous sommes cependant interrompus pour voir un simulateur de vol sur une station Indigo de Silicon Graphics... Il faut le voir pour le croire!)

Oui, je sais, il semble présomptueux de croire que quelques petits bidouilleurs pourraient aider la construction d'une fusée. Mais certaines de ces personnes sont celles qui ont créé l'industrie de la micro-informatique. Alors, Si quelqu'un peut le faire, c'est bien eux. Mais au fond, l'espace, pourquoi faire ?

#### L'espace, pourquoi faire?

Beaucoup de gens croient encore que le programme spatial est une perte de temps et d'argent, et que tout ce que nous avons fait, c'est ramener quelques cailloux de la Lune. C'est parce que personne ne leur a jamais expliqué ce que le programme spatial a fait pour nous. Le service des relations publiques de la NASA fait encore moins de vagues que celui d'Atari, c'est dire... Et il y a bien des gens qui voudraient s'emparer du ridicule budget de la NASA et l'ajouter à leur empire personnel (cf. Proxmire).

Si je vous dis "Nylon" ? Oui, les bas et les fils de pêche. Ou bien "Velcro" ? Vous en avez l'usage, non ? Eh bien, ce sont des dérivés directs de la R&D du programme spatial. Les satellites aussi, bien sûr, qui nous permettent de suivre la formation des cyclones et de sauver des centaines de milliers de vies humaines. Et saviez-vous que vous pouvez acheter aujourd'hui une balise de secours que des satellites peuvent repérer partout sur Terre? Ou bien un système de positionnement par satellite qui vous indiquera votre position à quelques mètres près partout sur Terre ? Vous pouvez oublier la crainte de vous perdre (ces systèmes ont été abondamment utilisés durant la Guerre du Golfe). Des photos prises par satellite montrent des ressources que nous pouvons utiliser, et aidaient à maintenir la paix du temps de l'URSS.

Les satellites Keyhole KH-1 1 étaient paraît-il Si performants qu'ils pouvaient lire une plaque d'immatriculation sur une base soviétique, que ce soit les essais de lasers anti-missiles à Dou shanbe, les silos de missiles soviétiques ou les énormes postes de commandement souterrains construits pour l'état-major en cas de conflit nucléaire. Durant le conflit israélo-égyptien de 1970, un passage de satellite fut absolument crucial pour maintenir la trêve, il indiquait les positions des troupes et aida les USA à persuader l'un des belligérants de ne pas violer les limites des traités.

Je peux sans problème prouver que les observations par satellite sont une des raisons pour lesquelles nous avons tous survécu à la course aux armements. C'est ce que j'appelle être utile.

#### L'espace vert

Je pourrais bavasser des heures sur les avantages de l'exploration spatiale, mais son plus grand mérite est le suivant : si vous vous souciez d'écologie et de préservation de l'environnement, vous devez soutenir le programme spatial. C'est notre seul espoir de nettoyer la planète et de la maintenir propre (mais la NASA n'est pas notre seule voie vers l'espace).

Prenons par exemple l'énergie, que nous devons absolument avoir en abondance pour notre économie. Avec la technologie des années 70, que nous pourrions mettre en œuvre aujourd'hui Si nous le voulions, nous pourrions placer en orbite des satellites de collecte d'énergie solaire, qui pourraient récolter l'énergie solaire brute, sans interférences atmosphériques, et l'envoyer sur Terre sous forme de rayonnement. Cela nous permettrait de nous passer de sources d'énergie controversées, comme les centrales thermiques au charbon génératrices de pluies acides, ou les centrales nucléaires engendrant des déchets qui resteront dangereux pendant 250 000 ans et que nous nous demandons où enterrer. C'est une idée si ancienne, qui a été calculée et vérifiée dans les moindres détails tant de fois qu'il est stupéfiant que rien n'ait été fait en ce domaine.

Avec de l'énergie bon marché (qui est l'ingrédient essentiel), nous pouvons nettoyer des zones polluées, et ne pas créer davantage de pollution. Généralement, une pollution résulte d'un processus industriel choisi pour sa moindre consommation en énergie et donc son moindre coût. Eh bien, la lumière du soleil est gratuite, combien vous en faut-il ? [NdT: en France, l'industrie aérospatiale a mis au point des lances à plasma, capables de "casser" à très haute température les molécules des polluants chimiques les plus stables, idéales pour changer en matière inerte de grandes quantités de déchets. Seul obstacle, leur énorme consommation en énergie...]

D'après le magazine Time, la couche d'ozone se détruit si vite que cela va poser problème très vite aux Etats-Unis, nous ne pouvons donc plus nous contenter de compatir aux problèmes des Australiens. Et pourtant on pourrait tenter d'y remédier, avec des lanceurs orbitaux.

Et de grandes choses peuvent être faites en apesanteur, depuis des fils incroyablement résistants appelés "mono filaments" jusqu'à de nouveaux médicaments, en passant par des billes de roulement parfaites (qui sont déjà utilisées !), mais toutes ces applications ont été terriblement retardées par les problèmes accumulés par la navette.

#### IDS et aile delta

Il y a quelque chose que vous trouverez sans doute ironique. Vous rappelez-vous tout le raffut autour de l'IDS, l'initiative de Défense Stratégique, surnommée "Guerre des Etoiles" ? A présent, les temps ont changé, et l'Office de l'IDS (SDIO) a plus de soucis à se faire pour un seul missile détenu par un fou que pour les milliers de l'ex-URSS. Et cette défense est un but réaliste, atteignable, et très moral. Au cas où vous ne sauriez pas, on ne peut dans l'immédiat rien faire d'autre pour éviter un missile nucléaire que s'abriter. Se donner un moyen d'action, c'est l'objet de l'IDS.

Un aparté intéressant au sujet de tout ce que les médias nous cachent.

Le missile Patriot, comme vous le savez, a sauvé de nombreuses vies en détruisant des missiles SCUD durant la guerre du Golfe. Mais savez-vous qui fut le fer de lance de la transformation du projet Patriot, à l'origine un missile anti-avions, en missile anti-missiles ? Ce fut Dan Quayle, alors Sénateur. Ce qui démontre que ne pas savoir parler en public n'implique pas qu'on soit idiot, chose que nous avons oubliée [NdT Dan Quayle, le vice-président américain, est réputé pour ses maladresses et est la tête de Turc de la presse].

Et savez-vous qui veut à présent un véhicule orbital réutilisable, qui pourrait nous ramener à l'espace et nous donner accès à des technologies salvatrices pour la Terre ? La NASA ? Vous n'y êtes pas.

C'est l'Office de l'IDS, le 5Db. Oui, les gens de la "Guerre des Etoiles". En fait, le SDIO a investi 58 millions de dollars dans un concept, le "DC-X Delta Clipper" de McDonnell-Douglas, qui ressemble étonnamment au Phoenix. Des gens comme Gary Hudson ont mis en équations ce concept depuis des années. Le nombre de moteurs qu'il faut par sécurité, en supposant que certains tombent en panne, le centre de gravité, la taille du nez, etc. Le concept est basé sur le moteur-fusée RL-10. Le RL-10 est aussi sûr que peut l'être un moteur, et son taux de réussite est excellent. Une revue très exacte et très fiable, "Aviation Week and Space Technology", en a fait l'éloge.

Nous parlons de choses concrètes. Premiers vols d'essais en avril 1993. Dans moins d'un an ! Vol orbitaux en été 1996. Voyez, ce n'est pas seulement un bel espoir, c'est du réel. Le DC-X utilise des technologies existantes, ce que tout programmeur appréciera (déboguer les nouveaux systèmes, c'est un enfer), et, comme mon rêve, enfin, se réalise.

Et Gary Hudson ? Ici, cet article a subi un coup de ciseaux massif, hélas. Gary a été assez aimable pour me parler de son projet et me fournir de la documentation dessus, mais à la condition que je n'en parle pas avant qu'il le dévoile. Je dirais que des gens comme Gary Hudson, comme tout vrai bidouilleur, ne doivent jamais être sous-estimés. Les grosses compagnies sortent des choses comme OS/2, les petites des choses comme Spectre GCR. Dites-moi laquelle marche le mieux.

Je vous garantis que vous entendrez parler des projets de Gary, qu'ils réussissent ou qu'ils échouent. En attendant, il a déjà gagné sur un point, puisque le SDIO investit dans un projet qui semble incroyablement familier à quiconque a déjà vu les plans de Phoenix. J'aurais souhaité les voir faire appel à Gary plutôt qu'à McDonneil-Douglas pour le construire, mais je soupçonne le gouvernement d'être partial envers les "fournisseurs connus". Et dès que je le pourrai, j'en écrirai davantage.

C'est bien la chose la plus formidable que j'aie connue de toute ma vie, en tout cas plus que je puis en communiquer par l'intermédiaire d'un traitement de texte.

#### Conclusion

Je vois les USA se replier sur eux-mêmes, perdre leur supériorité, et se dissoudre dans une multitude de groupes ne défendant que leur propres intérêts, au lieu d'une éthique constructive commune. Le Japon est l'exemple classique de nos concurrents. Et pourtant, il y a un domaine où nous excellons, à savoir la technologie spatiale et envoyer des gens là-haut. Dans moins d'un an, nous serons à nouveau sur la route qui mène à la prochaine frontière!

L'espace est l'endroit où nous pouvons changer notre devenir. Il peut nous fournir les ressources dont nous avons tant besoin (tôt ou tard, nous serons à court de pétrole, et plutôt tôt que tard), et l'espace est si innovateur que les nombreuses retombées déjà obtenues ne sont qu'un mince avant-goût des résultats futurs.

Gary s'est battu pendant 20 ans pour nous ramener dans l'espace, et on semble y venir. Si quelqu'un a jamais mérité d'être traité en héros, c'est bien lui, car s'il n'avait pas promu son idée de SSTO réutilisable, on ne construirait probablement pas de Delta Clipper aujourd'hui. Vols d'essais dans un an...

Je me rends bien compte que cet article n'est pas centré sur le monde Atari, mais il est toutefois centré sur l'univers de la bidouille, et sur l'espace qui nous affecte tous. J'ai pensé que vous aimeriez cet aperçu du futur, et de ce qui fait courir un bidouilleur sur Atari.

Je suis volontaire pour aider Gary au pilotage, aux tests, à n'importe quoi, n'importe quand ! J'ai redécouvert ce que je désire le plus au monde. Et béni soit Gary pour me l'avoir rappelé. Je vous laisse sur une pensée qui m'enthousiasme : Gary possède un exemplaire du moteur RL-10.

Traduction et adaptation: Password Titre original : The Fire Within.

## Le test de Myers-Briggs

Les grands événements ont toujours été le fait d'une poignée d'individus. La bataille de Midway, par exemple, qui changea le cours de la guerre du Pacifique, vit s'affronter quelques dizaines d'aviateurs et une poignée de commandants d'escadre de part et d'autre. De nos jours se déroulent de féroces batailles pour le contrôle des marchés de haute technologie dans le monde. Multinationales et financiers s'y affrontent sans merci. l'enjeu : des millions d'emplois, des milliards de francs, un impact énorme sur notre vie quotidienne. A la base, on trouve des technologies inventées par quelques bidouilleurs de génie. Mais comment pensent, comment raisonnent ces gens dont les créations ont tant d'influence ? Quels sont leurs rapports avec le reste de l'humanité ? Et pourquoi l'informatique est elle incompréhensible par le commun des mortels ?

#### Note de Dave Small aux lecteurs de ST Mag

J'ai écrit cet article en novembre 1987. Il expose certains travaux de psychologues peu connus en Europe, mais qui mériteraient attention car ces travaux expliquent bien des comportements dans l'univers informatique. Et il faut les connaître pour comprendre mes articles qui paraîtront dans les deux prochains ST Mag. c'est pourquoi j'ai souhaité voir celui-ci paraître d'abord.

#### Le retour de Jung

Si vous voulez bien m'accompagner dans une petite discussion psychologique, je vais vous parler de la chose la plus effarante que je connaisse sur les programmeurs - et qui présage mal de l'avenir de l'informatique. Et je répondrai enfin à la question que nous nous posons tous, "pourquoi les femmes n'aiment- elles pas l'informatique ?"

Dans les années 20, Carl Jung publia une théorie intéressante, selon laquelle il y avait des "types" de personnalité innés, que les gens ne pouvaient pas davantage changer que la couleur de leurs yeux. Jung remarqua également que les sages de la Grèce antique avaient "comme par hasard" donné à certains de leurs dieux le profil exact d'un type de personnalité donné. Les Grecs de l'antiquité connaissaient donc les différences entre les gens et les symbolisaient dans leurs dieux [NdT: Cette théorie est exposée dans "Types psychologiques" de Jung, ed. Librairie de l'Université Georg & Cie, Genève]. Les théories de Jung subirent une éclipse durant les années 30 et 40, ce qui n'est guère surprenant au vu des turbulences alors traversées par le monde. Mais dans les années 50, les travaux de Jung réapparurent. Deux femmes brillantes, Isabel Myers et Katherin Briggs, en firent la base d'une classification des types de personnalités. Elles créèrent l'Indicateur de Personnalité de Myers-Briggs(IPMB), qui depuis lors est devenu l'un des princlpaux indicateur de caractérologie, et est utilisé par de nombreux psychologues [NdT: aux USA du moins].

Les personnalités sont réparties en quatre types symbolisés par des lettres, composés de deux types de base, chacun se décomposant en deux sous- types. Le tableau suivant décrit les différents types, ainsi que leurs archétypes, leurs caractéristiques dominantes et les pourcentages qu'ils représentent dans la population occidentale.

Туре	Archétype	Caractéristiques dominantes	Pourcentage
NF	Artiste - Écrivain - Acteur	Intuition - Sentiments	10%
NT	Architecte - Programmeur	Intuition - Réflexion	10%
SP	Exécutant - Recherche de la performance	Sensations - Perception	40%
SJ	Manager - Membre d'une structure	Sensations - Jugement	40%

Par exemple, les Grecs avaient doté Prométhée d'une personnalité de type NT.

Mais attention, l'IPMB n'a jamais prétendu décrire exactement une personne. C'est impossible. On peut cependant esquisser une dominante globale, qui peut aider à comprendre certains individus. Voyons les différents types de plus près.

#### Les types de Myers-Briggs

Le "N" s'applique à vous si vous avez coutume de vous abandonner à vos intuitions, à vos intuitions, à vos réflexions, votre vie intérieure, tandis que le "S" décrit une personne qui recherche les sensations et tout ce qui vient de l'extérieur. Un poète, par exemple, est extrêmement intuitif, tandis qu'une vedette de rock est très tournée vers les sensations.

Les "F" et "T" décrivent la façon dont vous prenez vos décisions. Un "F" indique des décisions basées sur les sentiments, les émotions, les "tripes". Un "T" dénote une réflexion logique, des décisions qui procèdent du raisonnement. Je n'ai cité le "P" et le "J" que pour mémoire, car ces types sortent du cadre de cet article.

Les quatre types sont discutés et analysés en détail dans un livre intitulé "Please Understand Me" (Prometheus Press, 1984) de David Keirsey et Marylin Bates. C'est un ouvrage fascinant, probablement le plus utile que j'aie lu ces dix dernières années. Bien que j'aie toujours su que les gens étaient différents, je n'avais jamais rien lu qui explique en quoi ils l'étaient [NdT: ce livre est hélas inédit en Français].

Ce livre permet de définir son type de personnalité selon l'IPMB, ce qui m'a été d'une aide précieuse. Par exemple, ma propre personnalité s'est avérée être du type NF (écrivain). Or, 90% des écrivains sont des NF. Rien d'étonnant jusque là, hein ? Il se trouve que les NF ont des objectifs dans la vie qui sont incompréhensibles pour les autres types. Un bon exemple est "la recherche de soi-même". Si vous êtes en ce moment "à la recherche de vous-même", vous êtes sans doute un NT. Si vous pensez que cette expression n'a aucun sens, vous n'êtes probablement pas de ce type. C'est le genre même de choses qui n'a de sens que pour un NF. Le réalisateur Bill Murray avait fait un film sur "un homme à la recherche de lui-même". Mais seul 10% des gens peuvent s'identifier à un tel personnage. Et naturellement, le film fut un échec.

Et jusqu'à ce que je découvre que 90% de la population ne pensait pas de la même façon que moi, je m'étais toujours demandé à qui il manquait une case, à moi ou aux autres. Maintenant, je comprends en quoi je diffère de la majorité des gens, et je l'assume.

Les deux types minoritaires, NT et NF, représentent 10% de la population chacun. Les NF sont plutôt des artistes, des acteurs, des écrivains, des gens très créatifs. Quant aux NT, ce sont des êtres de logique. Ils aiment les chiffres, les maths, et aiment se représenter le monde en noir et blanc. Les scientifiques sont les personnalités NT classiques. Ils ont souvent des opinions binaires et voient le monde comme étant totalement d'une façon ou totalement de l'autre, jamais entre les deux. Robert Heinlein est l'écrivain NT classique et est à ce titre révéré parmi les fanas de science-fiction NT.

Les NT ne connaissent par exemple que deux catégories de logiciels. Ou bien un programme est "lamentable", ou bien il est "complètement dément". Jamais de nuances, jamais de "pas trop mal".

Pensez à votre dernière réunion de club d'utilisateurs... Ces expressions vous disent-elles quelque chose ?

Les NT sont plutôt froids et distants (bien qu'ils se défendent certainement d'agir comme tels), ce qui conduit les autres à les trouver assez arrogants. Par ailleurs, les NT aiment la documentation technique. Ils adorent découvrir comment marchent les choses, et vous l'expliquer en détail. Enfin, les NT aiment les casse-tête logiques et les énigmes. Un roman policier est pour eux un régal.

Avant que l'on me fasse remarquer que je ne suis pas qualifié pour parler des NT, qu'il me soit permis de signaler que j'en ai épousé une, et que j'en côtoie deux quotidiennement dans mon travail.

#### Une statistique effarante

Ce livre, comme je l'ai dit, me fascinait, c'est pourquoi j'ai commencé à tarabuster mes amis pour les soumettre à l'IPMB afin de déterminer leur type de personnalité. Et comme j'évolue dans le milieu informatique, mes relations comptent un grand nombre de programmeurs.

Et tous, absolument tous ces programmeurs se sont révélés être des NT! C'est ahurissant, vu que les NT ne représentent qu'un dixième de la population. Cette statistique tient depuis deux ans. C'est ça, la chose effarante que je voulais vous apprendre concernant les programmeurs. Je me trouve être le seul programmeur que je connaisse qui soit un NF, et encore est-ce parce que je suis un programmeur qui écrit au sujet de l'informatique.

Et je n'ai jamais vu quelqu'un heureux d'utiliser un ordinateur qui soit un SJ ou un SP. Et ces types représentent 80% de la population.

Cela signifie que les gens qui écrivent les logiciels que nous utilisons tous sont des NT, avec tous les bons et les mauvais côtés de ce type de personnalité. Et ces auteurs de logiciels orienteront leurs programmes selon leur penchants, même s'ils n'en ont pas l'intention, et feront un programme qui plaira aux autres NT. Cela leur semblera raisonnable, après tout.

C'est pourquoi vous obtiendrez un logiciel qui ressemblera à un casse-tête logique (pensez aux jeux d'aventure comme Zork). Ces logiciels dépeindront le monde sous forme de chiffres (pensez aux jeux de rôles qui résument un personnage en quelques chiffres, comme sa "force" ou son "charisme"). Ou bien ce seront des jeux apocalyptiques, où l'on ne peut que gagner ou perdre à 100% (pensez aux jeux d'arcade). Ou encore des tableurs, bases de données ou traitements de textes, orientés vers les applications techniques qui peuvent être représentées sous forme de chiffres.

Vous verrez des abstractions pleines de cet infinité de détails qu'adorent les NT, comme le langage C, qui est d'une complexité aussi inutile que vaine, mais un vrai régal pour les amateurs de casse-tête logiques. Savez-vous même qu'il existe un livre intitulé "Puzzles en C" ? Si, si, ça existe ! Avez-vous jamais vu un programmeur en C demander à un autre "Qu'est-ce que fait ceci? " en exhibant un bout de code particulièrement tordu ? Moi, je l'ai vu. En fait, le C est presque inutilisable pour moi.

Tout ceci vous rappelle-t-il quelque chose ?

Plongeons maintenant dans notre documentation technique. Un NF écrit une doc ainsi :

- 1. Insérez la disquette dans le lecteur
- 2. Allumez la machine.

Et ainsi de suite. En général, ces docs vous expliquent ce que le logiciel fera pour vous.

Par contre, un NT écrit une doc en voulant tout expliquer:

1. Insérez la disquette dans le lecteur. La disquette est recouverte d'oxyde de fer, lequel peut être magnétisé par endroits. Elle tourne à 300 (trois cents) tours/minute, et les données sont cadencées à 1 MHz. Elles sont contenues dans des cellules cadencées à 250 (deux cent cinquante) kHz. Il y a 80 (quatre vingts) pistes sur la disquette, et 10 (dix) secteurs par piste... (puis, après cinq pages sur la théorie des lecteurs de disquettes...)

2. Allumez l'ordinateur. Cela s'effectue en abaissant l'interrupteur à bascule situé à 5 (cinq) cm depuis le haut du capot arrière, jusqu'à ce que vous entendiez un déclic franc. A ce stade, le ventilateur doit se mettre à tourner...

Cela vous rappelle-t-il quelque chose ? Si vous avez lu des docs de chez IBM, TI, CDC ou HP, ce langage vous est probablement familier. Car les gars qui écrivent ces docs sont des NT.

Et je suis persuadé que c'est là la raison pour laquelle la majeure partie de la population (les 90% de non-NT) trouvent que toute la documentation informatique est abominable. Les NT qui l'écrivent la conçoivent comme pour eux, et n'ont pas la moindre idée que cela ne convient pas aux autres. Pis, cette idée ne les a jamais effleurés. Allons donc, qui voudrait lire une documentation sans tout apprendre sur l'ordinateur ? Ce ne serait pas raisonnable, pensent-ils, d'en savoir juste assez pour faire tourner le système.

A GBS, les notions de NF et de NT ont intégré notre lexique privé. Et cela nous a aidé à mieux nous entendre. En tant que NF, je suis souvent en désaccord avec les NT qui m'entourent sur de nombreuses choses. Mais ces désagréments sont rendus tolérables lorsqu'on sait qu'ils proviennent d'une façon différente d'appréhender le monde, et non d'une inexplicable bizarrerie dans le comportement de l'autre. J'ai mentionné que j'ai épousé une NT. Nous ne nous entendions pas vraiment avant que je n'aie lu ce livre, où sa personnalité, qui m'irritait par bien des aspects, était parfaitement décrite (elle a un diplôme en informatique...). Pour la première fois, je compris que si elle agissait comme elle le faisait, c'est que c'était sa nature innée, elle était née comme ça. Elle ne faisait pas ça rien que pour m'énerver! Ce qui faisait une grosse différence.

Dans l'univers micro-informatique, nous avons traversé l'âge de la première génération d'ordinateurs, qui étaient des régals pour les NT. Le DOS des PC, par exemple, est un casse-tête logique désespérant. Or, on a assené à la majorité des gens que l'informatique était importante. Que s'est-il alors passé ? La crise de la micro-informatique de 1984, quand ces mêmes gens décidèrent que l'informatique ne les intéressait pas, après tout, en dépit du battage. Imaginez la frustration des concepteurs d'ordinateurs NT, qui ne pouvaient pas voir pourquoi les gens ne raffolaient pas de leurs machines.

Pensez donc... Tous ces ordinateurs faits pour 10% de la population.

Mon avis personnel est que si le Macintosh a eu du succès, c'est parce que c'était le premier ordinateur que les NF ont aimé utiliser. Voyez les gens que vous voyez généralement utiliser un Mac. Des artistes, des musiciens, des concepteurs graphiques. Dans leur quasi totalité, ces gens sont des NF. Ils n'aiment pas devoir se rappeler ces petites commandes obscures qui sont indispensables pour faire marcher un PC sous DOS. La plupart d'entre eux ignorent sans doute qu'un disque se compose d'oxyde de fer et a 80 pistes. Ce qui les intéresse, c'est l'icône "Disque" sur laquelle il n'ont qu'à cliquer. Le Mac les isole de tous ces détails, ces jeux de logiques que les NF détestent...

Je pense également que personne n'a encore inventé un ordinateur que le reste de la population (les 80% de SJ et SP) aimeraient utiliser. Oh, certes, s'il y est forcé, un exécutant utilisera un traitement de texte. Ou bien un manager, mis au pied du mur, emploiera un tableur. Mais cela reste rare. Et leur machine ne sera certainement pas utilisée pour leurs loisirs.

Nous avons donc jusqu'à présent touché 20% de la population et laissé pour compte le reste. Mes parents ont un ordinateur, bien à l'abri dans un placard (ce ne sont ni des NT ni des NF). Et je parie que vous connaissez des tas de gens qui pensent de même vis-à-vis de l'informatique.

Le profil de personnalité explique aussi dans une certaine mesure pourquoi les programmeurs et les artistes sont généralement difficiles à gérer pour un manager. Les NT et NF ont des points de vue différents des SJ, les managers classiques. Aucun des deux types n'a la moindre idée des buts personnels de l'autre. Par exemple, les SJ tendent à considérer les NF comme des "artistes volages" et les NT comme des "programmeurs arrogants". Les NT ne comprennent pas pourquoi le manager veut quelque chose dans les délais impartis, au lieu de quelque chose qui soit parfait dans les moindres détails. Les NF, eux, ne comprennent pas pourquoi le manager refuse de refaire ce coffret d'emballage qui est si laid, si peu esthétique.

Et j'ai vu ces confrontations survenir dans toutes les firmes de programmation. c'est une des principales raisons pour lesquelles les boîtes de programmation se plantent.

C'est triste à dire, mais les programmeurs eux-mêmes sont en général des gens assez malheureux. Ils ont grandi dans l'isolement, car seul une personne sur dix leur ressemble. Au collège, ils étaient plutôt solitaires, étant considéré au mieux comme des "grosses tronches". Ce n'est qu'au lycée et à la fac qu'ils ont commencé à trouver des gens comme eux. C'est pourquoi ils ont pris l'habitude d'être distants, renfermés, ou bien ils sont devenus glaciaux et arrogants vis-à-vis d'une population qu'ils trouvent irrationnelle. Les ordinateurs sont pour eux une parfaite coquille dans laquelle ils s'isolent avec joie : les ordinateurs, eux, sont toujours logiques, rationnels, et jouent cartes sur tables. Au contraire des gens.

En allant à la HackerCon 3.0 (la troisième Hackers Conference), j'ai vu beaucoup, beaucoup de programmeurs NT malheureux. Ils s'épanouissaient à la conférence au contact d'autres NT, vis-à-vis desquels ils pouvaient s'ouvrir un peu. Mais dès la fin de la conférence, on les voyait revenir à leur mode de vie isolé. Au fait, oui, les expressions "lamentable" et "complètement dément" étaient omniprésentes à la conférence. J'ai rencontré là l'ancienne rédactrice- en chef d'un journal de langages informatiques qui était démoralisée et se demandait pourquoi l'informatique et les invités de HackerCon s'étaient mis à tant l'ennuyer. Je lui ai parlé du livre, elle s'est révélée être une NF, pas du tout le type NT du rédacteur technique.

#### Comment écrire un scénario

Les scénaristes de séries TV ont découvert depuis longtemps que si vous vous attachez à dépeindre un personnage en le dotant des attributs classiques d'un type de personnalité donnée, vous aurez des spectateurs qui s'identifieront rapidement à ce personnage. Une chose peu connue au sujet de la série "Star Trek", une des raisons qui lui assure la loyauté indéfectible de fans enragés, est que les personnages de cette série collent de très près aux différents types, donnant ainsi à chacun la possibilité de s'identifier à l'un d'eux. ( C'est d'ailleurs le problème avec "Star Trek, la nouvelle génération", dont les scénaristes ne suivent plus cette règle. Il se peut que la série originale ait créé ses personnages-archétypes par hasard.) Dans "Star Trek", il y a le capitaine Kirk, le manager SJ, défenseur des institutions de la société. Il y a Spock, le NT classique. Et McCoy, le NF. Sans oublier Scotty et Sulu, les SP.

Voyons si ceci vous rappelle quelque chose :

"Spock, espèce de monstre froid, Vulcain inhumain, comment pouvez-vous montrer si peu de compassion?" Le NF McCoy.

"Vraiment, docteur. (Haussement d'un sourcil). Vous êtes si illogique." Le NT Spock.

"Messieurs, retournons au travail. Vitesse 9, Scotty." Le manager SJ Kirk.

"A vos ordres, mon capitaine!" l'exécutant SP Scotty.

Ça devrait évoquer des souvenirs : les scénaristes placent un dialogue de ce style au moins une fois par épisode. Et pour moi, ça évoque furieusement les types de personnalités classiques.

Les écrivains emploient également ce système. Dans mon quatrième livre, "Le sorcier blanc", qui n'était pas assez bon pour être publié (eh oui, c'est dur de se faire éditer), j'ai tenté de falre coller mes personnages à ces types de personnalité, afin de les rendre plus crédibles, parce que les gens agissent dans la réalité comme le décrit le livre. D'autres auteurs le font mieux que moi, certains instinctivement, d'autres plus pesamment. c'est pourquoi ils sont publiés.

#### Conclusion

J'espère que cette petite introduction à l'indicateur de Personnalité de Myers-Briggs vous aura été utile. D'ailleurs, si vous lisez ceci, dans ST Mag, vous êtes probablement un NT, car qui d'autre utilise un

ordinateur pour ses loisirs?

Mais peut-être trouvez-vous cet article "lamentable" ? Gagné !

A ma décharge, je dirais qu'il m'a été extrêmement difficile de condenser un excellent livre dans cet article. Mais si vous êtes un programmeur NT, peut-être ceci vous aiderait-il à comprendre ce qui ne va pas dans vos relations avec les autres, femme et patron compris.

Et ma promesse de répondre à la question de savoir pourquoi les femmes n'aiment pas l'informatique, me direz-vous ? J'y viens.

C'est une question-piège, parce que certaines femmes l'aiment. Les NT. Mais 80% des femmes ne l'aiment pas, tout comme 80% des hommes. Si vous êtes une femme NT, vous l'appréciez sans doute. Mais les femmes, NT comprises, tendent à préférer les activités théoriques aux bidouillages technologiques, ce qui les rend plus rares dans nos rangs. J'avais fait un article à ce sujet, et j'ai eu en retour mon comptant de lettres disant, dans les classiques termes très tranchés des NT : "votre article est totalement faux... Nous avons beaucoup de femmes dans nos clubs d'utilisateurs... Etc."

Mais pourquoi alors un NF comme moi est il devenu programmeur ? c'est une autre histoire. Que je raconterais peut-être quand je ne serais plus "à la recherche de moi même"...

Traduction et adaptation : Password Titre original : The NF,s, NT,s, SP,s and SJ,s

## Hérésie (part one)

L'archétype du programmeur est l'individu qui va tous las matins à son bureau dans une grosse compagnie et développe en langage de haut niveau des choses pas toujours drôles mais qui appartiennent au domaine du possible. Dave Small, lui, a créé sa boîte, vit de l'émulation Mac sur Atari pourtant jugée "impossible" par des spécialistes, et code en assembleur. C'est en ce sens un anti-conformiste. C'est pourquoi il est bien placé pour vous proposer ce mois-ci une petite étude caustique du milieu des développeurs qui se voulant tous exactement aussi anti-conformistes que les autres.

#### Non-conformistes?

De prime abord, beaucoup de programmeurs semblent s'afficher comme des non conformistes absolus. Le plus gros choc survient en général lors de la première rencontre. Imaginez un type portant une queue de cheval lui descendant jusqu'aux fesses, maintenue par un élastique et une généreuse couche de sébum, ainsi qu'une barbe à faire pâlir d'envie Fidel Castro. En outre, l'individu fait de l'exercice chaque jour, et prend une douche... euh... tous les mois? Ce qui fait que, tandis que vous lui serrez la main, vous vous placez habilement contre le vent et vous vous demandez où vous pourriez trouver un désinfectant au plus vite. Car cette poignée de main évoque les léchouilles gluantes d'un chien errant sortant de quelque poubelle.

J'ai rencontré tant de gens comme celui-là que je ne les trouve même plus exceptionnels. Si vous voulez vivre cette expérience exaltante, allez au centre informatique de l'université la plus proche, et cherchez des étudiants de troisième cycle. Mais cette impression de non-conformisme ferait place à la stupéfaction si vous voyiez une cinquantaine de ces bidouilleurs ensemble. Ils ont l'air de clones. Ils sont en fait tous absolument conformes à un certain code! Il est hilarant de voir qu'ils ressemblent en cela à ces adolescents "non-conformistes" qui portent les mêmes jeans, les mêmes coiffures punk, et conduisent des voitures de sport. [NdT: on trouve aux USA des voitures de sport d'occasion à moins de 2000 dollars, le permis est à 16 ans, il n,y a pas de vignette, et l'essence 4 est fois moins chère qu'en France. C'est pourquoi même un ado désargenté peut s'offrir une telle voiture au prix d'un boulot d'été.].

Par ailleurs, ces bidouilleurs ont en commun beaucoup d'opinions sur la programmation et ne sauraient mettre en doute ces croyances. Osez le faire, et, croyez-moi, vous vous ferez expulser de la clique avant de comprendre ce qui arrive. Je le sais, j'ai émis un doute. Je ne peux m'en empêcher, ça fait partie de mon type de personnalité. D'ailleurs, j'ai passé jadis un test mesurant le degré de conformisme, et mon score était presque nul. C'est pourquoi leur attitude me donne des boutons, en particulier à raison d'une douche par mois.

Pour votre édification, je me livre donc ici à quelques propos hérétiques, et m'attends d'ailleurs à recevoir des tomates la prochaine fois que je me présenterai à un salon informatique. Sans compter les tomates "virtuelles" lancées via les réseaux Internet et USENET (voir mes coordonnées à la fin de l'article). Faites attention, mon sélecteur de tir, qui est d'habitude en position "coup-par-coup de précision", est pour cet article placé sur " rafales continues". Comme vous le savez sans doute, l'arme se relève en tir automatique, et tout servant de mitrailleuse vous dira qu'il faut ajuster le tir, c'est-à-dire observer les impacts et corriger. C'est pourquoi, si certains de mes arguments trop généraux vous semblent à côté de la plaque, essayez de voir quel but je vise maladroitement, quels principes j'essaie d'exprimer, avant d'ouvrir le feu en représailles!

Le bon professeur Wirth est l'inventeur du Pascal et de Modula. C'est lui qui a mis le monde entier sur les rails de la programmation structurée. Le Pascal est uniquement un langage d'enseignement de la programmation, qui était censé montrer comment la programmation structurée marchait. Il n'a jamais été conçu pour un usage dans le monde réel, ce qui est par contre le but de Modula (c'est Wirth luimême qui le dit). Franchement, si j'étais lui, je ne serais pas sûr de vouloir passer à la postérité pour le Pascal. Mais malheureusement, tout ce que dit Wirth est noté aussi scrupuleusement que si c'était tiré des Tables de la Loi, tout comme le moindre mot de Brian Kernighan ou Dennis Ritchie est pris comme référence pour le langage C. Ce qui est aussi absurde que si on on suivait les préceptes d'Henry Ford pour construire une Ford modèle 1992 sous prétexte qu'il a inventé le Modèle T. Il y a des gens qui ne deviennent jamais adulte : quand ils étaient petits, leurs héros étaient Musclor et Superman; maintenant, c'est en Wirth, ou Kernighan et Ritchie, qu'ils ont la foi absolue. Cette vénération irréfléchie est toujours malsaine. Je ne sais plus qui a dit que le mot "croire" est le plus effrayant, car il implique que le cerveau est déconnecté.

Ainsi, bien que Wirth ait inventé le Pascal pour l'enseignement, ce langage a été implémenté sur de nombreux ordinateurs. Et dans le monde réel, des programmeurs se sont rapidement rendu compte de ses défauts. En particulier, ce langage est restrictif. Vous aviez beau savoir ce que vous vouliez faire, ce stupide langage ne vous le permettait pas. (Les puristes vous diront que ce que vous vouliez était donc mauvais. Billevesées! Vous avez payé votre machine, vous pouvez en exiger ce que vous voulez.) Le Pascal est presque dépourvu d'entrées\ sorties, et se limite à lire ou écrire un caractère ou une ligne à la fois dans un fichier. Très impressionnant, hein? Ce n'est pas moi qui irait écrire des programmes dans un langage qui vous lie une main dans le dos.

Je pense que j'ai une profonde haine du Pascal pour deux raisons. Primo, on a essayé de m'en gaver de force au lycée, d'où je me suis échappé juste avant qu'il ne devienne obligatoire (et rende à mon avis les ordinateurs incompréhensibles à la majorité des gens auxquels la notion de structure est étrangère). Et secondo les ROM du Macintosh, sur lesquelles j'ai passé tant de temps, sont principalement écrites en Pascal, ce qui m'a obligé à avaler les horreurs produites par le compilateur qui a engendré le code assembleur stocké dans ces ROM.

Il y a aussi dans ces ROM de l'assembleur fait-main par Andy Hertzfeld, un magicien, et il est très instructif de comparer ce code propre et rapide avec l'abominable saleté pondue par le Pascal (toutes ces instructions LINK inutiles, par exemple). Le code machine engendré par cet abruti de compilateur est horriblement inefficace. Moi, quand j'écris un programme, j'aime pouvoir être fier de sa rapidité et en tirer quelque gloire.

C'est peut-être pour cela que les programmeurs chez Apple ne mettent plus leurs nom dans leurs programmes. Peut-être est-ce aussi pour cela que beaucoup d'entre eux viennent juste d'être licenciés? Non, je suis sûr qu'ils écrivaient du beau code. C'est plutôt dans la haute hiérarchie que cela n'allait pas.

Avis aux puristes : je suis conscient des différences entre le code machine binaire et l'assembleur qui est sa forme symbolique. Pour cette article, j'ignore cette différence. Rasseyez- vous, ça continue.

#### La programmation structurée est une antiquité

Les bidouilleurs utilisent la programmation structurée, ça fait partie de leur conformisme. Eh bien, moi pas, et mes tentatives pour l'utiliser se sont soldées par des échecs cuisants. Il me semble qu'écrire un programme de cette manière revient à me faire violence. La programmation structurée reflète un certain type de personnalité, le type NT selon le profil de Myers-Briggs, que j'ai découvert être celui de pratiquement tous les programmeurs. j'ai écrit un article à ce sujet [NdT : voir ST-Mag n° 64]; brièvement, je rappelle que presque tous les bidouilleurs ont une personnalité d'un type rare, présent chez seulement une personne sur dix, et ont des raisonnements et des réactions très différentes de ceux de la majorité. Des psychologues l'ont également remarqué. Les NT sont fondamentalement des architectes, et tout dans un programme informatique reflète cette mentalité. Je ne suis pas un NT, je me situe même à l'opposé dans les tests. C'est pourquoi je comprend l'étonnement de la majorité des gens devant l'informatique. Les partisans de la programmation structurée croient que tous les problèmes de

programmation peuvent être scindés en quelques constructions de base : la boucle DO-WHILE, l'instruction CASE (qui équivaut à une série de IF-THEN-ELSE), et quelques autres. Et ils croient aussi que c'est ce qui doit être fait dans tous les cas. Pour un bidouilleur NT, c'est tout-à-fait évident et intuitif. Pour les 90% du reste de la population, c'est ahurissant. Imaginez un enfant avec une boîte Lego, séparant et ré-assemblant ses pièces, et vous avez un futur NT qui fera de la programmation structurée.

## Il faut utiliser des sous-programmes

Pour s'accommoder de la complexité de certains programmes, les adeptes de la programmation structurée recommandent de scinder un programme en 36 000 petits sous- programmes, dont chacun ne fait qu'une seule chose. Pour moi, cela rend le tout encore moins compréhensible qu'un seul gros programme. C'est comme écrire un roman dont chaque paragraphe est mis dans un fichier séparé! (Ou comme faire un magazine à partir d'articles contenus dans des fichiers différents... Pauvre rédacteur-en chef!).

Adoptez par exemple le style de programmation de chez AT&T, où chaque sous-programme doit se trouver dans un fichier séparé, et vous vous retrouvez aux prises avec un bourbier innommable. Vous avez alors besoin du programme MAKE auquel vous devez préciser, dans un fichier de commande, dans quel ordre s'emboîtent ces morceaux pour construire votre programme. Dans ce genre de situation, il faut tout arrêter et vérifier que la belle idée colle toujours à la réalité (ce que j'appelle une vérification de réalité). Donc, le programme a été scindé à grand-peine en 500 modules, avec force variables importées et exportées, et, toujours à grand-peine, un fichier de commande pour MAKE a été écrit pour ré-assembler tous ces morceaux. Question : qu'y a-t-il d'anormal dans cette situation? Formulez votre réponse en utilisant les mots "heures" et "jours".

Contre-exemple : comparez avec Dave le simplet. Moi, j'écris le programme en un seul morceau, je le compile, et c'est fini. Je m'épargne l'effort considérable de le scinder et le non moins considérable travail de ré-assemblage.

## Les ordinateur ne sont pas parfaits

ème

Ajoutez à cela la 17 Loi de l'Informatique de Small, "Les ordinateurs ne sont pas encore parfaits". Les outils de scindage et de ré-assemblage des morceaux ne le sont pas non plus.

Pendant que vous compilerez séparément vos différents fichiers avant de les passer tous ensemble à l'éditeur de liens, vous mettrez votre machine à rude épreuve. Et vous découvrirez, à la manière forte, chacun des bogues de votre système d'exploitation, de votre compilateur et de votre éditeur de liens. J'en ai vu tellement que je ne soupire même plus de dégoût. J'apprécie vraiment lorsque ALN (le nouvel éditeur de lien d'Atari) me donne un message d'erreur en traitant des fichiers qui passaient sans erreur sur le vieux LO68/RELMOD, par exemple (devinez un peu lequel j'utilise). Franchement, si quelque chose n'allait pas, je l'aurait découvert avant; j'ai recompilé ce code des milliards de fois. Et je pourrais citer bien d'autres exemples.

Ne faites pas l'erreur de croire que les ordinateurs sont parfaits. Un tantinet de saine paranoïa s'impose. Si vous contemplez l'historique de n'importe quel ordinateur vous verrez immanquablement le même scénario : l'ordinateur sort, et les corrections de bogues suivent. Prenons Atari. Citons en vrac le bogue des 40 dossiers, celui du caractère "souligné" dans le sélecteur d'objet, les 97 bogues d'AS68 dont j'ai la liste, le bogue exaspérant de RELMOD qui lui fait remplir tout le disque si le fichier en entrée est incorrect, les choix discutables de conception du matériel (amener le signal RESET jusqu'aux lecteurs de disquette sans amplification? Non, mais je rêve!), et j'en passe.

Alors, savez-vous ce que je fais? Je m'en tiens à un seul assembleur et éditeur de liens, et quand je leur trouve un bogue, je l'ajoute au dossier BOGUES de ma machine. Je ne change l'outil que dans les cas désespérés, parce que je sais pouvoir m'affranchir des bogues d'un outil donné. (Ne pas mettre

deux labels à la suite sans instruction dans AS68, par exemple. Ne pas utiliser la compilation conditionnelle avec AS68.) C'est en tremblant de frousse que je suis passé à l'assembleur HiSoft, je reconnais que, mis à part quelques anomalies (comment diable le fait-on produire du code relatif au PC?), il marche bien.

Encore une petite vérification de réalité. Figurez-vous que mon émulateur Mac, le Spectre GCR, est obligé de corriger des défaillances de programmes Mac lorsqu'ils essaient d'écrire à l'adresse zéro à cause d'erreurs de logique ou de programmation. Pour mémoire, je dirais seulement que la moitié des programmes Mac, y compris la plupart de ceux, très connus, qui ont fait le succès de la machine, sont truffés d'erreurs d'écriture à l'adresse zéro. Ce sont des programmes écrits principalement en C et en Pascal, tournant sous un système l'exploitation écrit en Pascal, et écrits par des gens qui sont considérés comme étant parmi les meilleurs de la profession. Eh bien, même eux finissent par ne plus pouvoir se souvenir de leurs pointeurs et de la validité de ceux-ci, tant sont subtiles certaines opérations de déplacement de mémoire.

Rien qu'à l'allumage du Mac, avant d'en arriver au bureau, je compte au moins huit erreurs d'écriture en zéro! Et qui a écrit ça?La seule raison pour laquelle le Mac s'en tire est qu'à l'adresse zéro se trouve, par le plus pur des hasards, de la RAM, ce qui évite donc une erreur de bus (sur le ST, au contraire, c'est de la ROM, c'est pourquoi il m'a fallu arranger cela!). Mais souvent, ces bogues reviennent hanter le programme; après tout, cette écriture en zéro était faite pour stocker une donnée, et quelque chose d'important n'a pas été stocké à sa place. Quand le programme le relit, les bonnes valeurs ne s'y trouve pas, et boum! Erreur d'adresse, le plus souvent (3 bombes sur le ST, ID=2 sur le Mac). Finalement, la méthode du ST est la meilleure et il serait bon sur le Mac de modifier l'adresse zéro pour que toute écriture y engendre une erreur, afin que les programmeurs corrigent leurs codes. Apple, rendons-lui cette justice, encourage cela.

Si vous acceptez de signer un accord de non-divulgation, je veux bien vous faire faire un petit tour à vous soulever le cœur à travers la galerie des programmes Mac qui ne tournent que par accident. Je ne veux pas les nommer ici parce que j'ai horreur des procès.

Loi de Small n° 15 : les vrais programmeurs s'abonnent à tous les magazines et se connectent à tous les réseaux qu'ils peuvent atteindre, et découpent ou sauvent dans des fichiers les mentions de bogues et les astuces de programmation. Vous seriez surpris de voir combien un seul petit article peut vous aider.

## En assembleur, on voit ce que fait la machine

J'aime tirer une leçon de mes erreurs. Je trouve que lorsque je me tiens en retrait de ce que fait mon code dans la machine, comme par l'intermédiaire du langage C, ce code devient beaucoup plus dur à déboguer. Or, le débogage est un gros pourcentage du temps du programmeur! En utilisant l'assembleur, je peux tracer, pas à pas, mes instructions, et trouver l'erreur.

En C? Laissez tomber. Il faut revenir au débogage de l'âge des cavernes en mettant dans le code des lignes affichant "J'ai exécuté la ligne 10!", "J'ai exécuté la ligne 20!" et autres atrocités. Je ne voudrais pas passer ma carrière à lire et relire mon code C à la recherche d'un bug que je ne peux pas tracer directement. Vous savez, ces fameux problèmes à se gratter la tête de perplexité. Et le pire, c'est qu'il s'agit parfois d'un problème dans le compilateur, ses bibliothèques ou le système d'exploitation, pas d'une erreur de votre part, et vous ne le trouverez jamais. Point final.

En assembleur, vous le trouverez en traçant votre code, parce que vous voyez l'ordinateur à son niveau réel, et non à travers quelque fiction de machine parfaite. Vous pouvez même bidouiller des choses... disons, du code relatif au PC... que vous ne pouvez pas écrire directement dans votre source, en entrant des octets manuellement.

Du coup, j'ai acquis une accoutumance à l'assembleur. Quand mon programme plante, je trotte jusqu'aux adresses \$380-\$3FF et je regarde à quelle adresse précisément je me suis planté. Ou bien je lance le débogueur en mode trace et je regarde le plantage se dérouler. Ou mieux, j'utilise l'émulateur en circuit Zax [NdT : les ICE (In-Circuit Emulators) sont des appareils qui reproduisent la circuiterie d'un microprocesseur comme le 68000, et donnent accès en temps réel aux états internes, aux registres et

au moindre bit du microprocesseur émulé.] Si votre programme plante, le Zax vous révélera les 4096 dernières instructions exécutées par votre machine. Une sacré loupe de Sherlock Holmes! Ce qui me permet de corriger le bogue immédiatement. Fini la perplexité. Le problème apparaît en général instantanément.

Si vous aimez le son du Spectre, sachez qu'il est là parce que le Zax m'a prouvé que la circuiterie de l'Atari accomplissait quelque chose réputé "impossible". Qui pourtant est là. Le Zax a beaucoup amélioré le Spectre à partir de la version 3.0. (pour être honnête, les éditeurs de compilateurs C sortent maintenant des débogueurs symboliques qui permettent de tracer ce que fait le code C, et ces outils ont plus d'une ressemblance avec les traces assembleurs. Disons que comparé au débogage en assembleur et au Zax, ces outils abordent l'aube de l'ère industrielle.)

## Plus facile à maintenir

Les zélateurs de la programmation structurée affirment que l'on doit diviser un programme en utilisant seulement quelques structures simples. En procédant ainsi, affirment les conformistes, vous le rendez plus simple à maintenir.

Foutaises! Les problèmes les plus difficiles à résoudre en programmation sont les plus simples conceptuellement. Ces problèmes ne sont pas simplifiés par un quelconque style de programmation. (Les autres problèmes sont sans doute des fautes de frappe ou des bugs dans ce crétin de compilateur C. Exemple : "Est-ce que le fabricant a sorti une mise à jour du compilateur qui ne force pas les définitions SHORT à être alignées aux mots ? Crénom, pas étonnant que ce code ait eu des problèmes!...")

Tenez, considérez un pré carré entouré d'une clôture, dont les poteaux sont espacés de 4 m. Le pré fait 40 x 40 m. Combien de poteau y a-t-il sur chaque côté? Et au total? Déjà, contrairement à ce qu'on pourrait penser hâtivement, il n'y a pas 10 poteaux par côté, mais 11. En effet, il faut un poteau au coin pour marquer le début d'un côté, et ensuite on en compte 10, un par mètre. En programmation, cela correspond à une boucle qui débute à 0. Pour aller de 0 à 10 inclus, il y a 11 itérations. Cette erreur classique de programmation est si connue qu'elle a justement été baptisée "erreur du poteau" . Pour moi, cela représente parfaitement le problème philosophique extrême de faire correspondre au monde réel ce que dit l'ordinateur. Il y a beaucoup d'erreurs classiques de ce genre. Et diviser le problèmes en structures ne peut vous aider à les résoudre, car ces problèmes ont leur origine dans votre tête, pas dans la machine!

Ah oui, et le nombre total de poteau? Méditez cet exemple parfait de bonne réponse bâtie sur un mauvais raisonnement, comme ces programmes qui marchent par hasard : "40 poteaux au total, bien sûr, puisqu'il y en a 10 par côté."

## L'erreur du roi Salomon

Je voudrais mentionner une erreur classique de méthode de programmation que j'appellerais "l'erreur du roi Salomon". Vous vous souvenez de l'histoire biblique du roi Salomon qui proposa de couper un bébé en deux pour résoudre une dispute entre deux femmes s'en prétendant mère? Eh bien, découpez un programmes en morceaux, et vous aussi découvrirez qu'il est pénible de chercher les interactions entre les différents modules, simplement parce que vous ne pouvez pas les voir à l'écran en même temps. Je suis d'autant plus sensibilisé à ce problème que j'écris. Quand je rédige un article, j'aime en voir le maximum à l'écran.

Quand je le peux, j'utilise un écran Moniterm, et j'offre une forte récompense à quiconque modifiera un traitement de texte pour qu'il puisse utiliser les deux "pages" d'un Moniterm [NdT: Dave Small ignorait alors que des traitements de texte européens ont cette possibilité]. En voir beaucoup aide à rendre l'article cohérent. Non? Alors allez écrire sur un écran de huit lignes et relisez la bouillie informe qui en

émergera. Quand je veux maintenir quelque chose, je charge le code source entier dans un éditeur. J'ai parsemé le code source de commentaires contenant des mots-clés qui me servent de repères. Si je veux, disons, modifier le code qui gère la souris dans Spectre, je fais une recherche du mot-clé "trouve-souris", lequel se trouve dans un commentaire en tête du segment de code en question. Je peux alors regarder ce bout de code et ses interactions avec le reste sans avoir à ouvrir des fenêtres, charger d'autres fichiers, etc. Je ne saurais vous dire combien d'erreurs l'interaction j'aurais manqué si tout n'avait pas été lisible en un seul endroit.

#### **A Suivre**

J'arrête mes hérésies pour ce mois-ci. Le <u>mois prochain</u>, je jetterai un œil à l'hérésie en C et en Basic. A la prochaine fois!

Traduction et adaptation : Password

# Hérésie (seconde partie)

et non "part two" comme cela a été écrit le mois dernier durant la maquette au plus grand désespoir de Password 90 qui fut accusé de ne pas faire de traduction complète... encore pardon

Dans la <u>première partie</u> de cet article, Dave s'en prenait à certaines idées reçues et dénonçait un insidieux terrorisme intellectuel. Ce mois-ci, Dave continue le massacre en s'attaquant entre autres au langage C. Inutile de dire qu'à la rédaction de ST-Mag, cela fait grincer pas mal de dents, car les grands programmeurs que nous côtoyons ont tendance à oublier les difficultés qu'ils éprouvèrent lorsqu'ils étaient débutants. Dave en profite aussi pour égratigner la mode du langage "politiquement correct" qui fait fureur outre-Atlantique et semble gagner nos rivages. Dave a spécialement écrit pour les lecteurs de ST-Mag une petite explication de ce phénomène...

## Tout le monde trouve le langage C facile, sauf les crétins

J'en ai plus qu'assez du langage C. J'ai codé en C et c'est pour moi aussi intuitif et agréable que de changer une couche sale. Et pourtant, les bidouilleurs considèrent que le C et son successeur orienté objet, le C++, sont les langages de base. AAARGH!

Mais désormais, d'autres que moi critiquent le C. J'ai eu récemment le plaisir de lire Jerry Pournelle [NdT: journaliste du mensuel américain Byte et auteur de SF le mettre à mal. Le problème de fond est que les programmeurs, presque tous du type de personnalité NT, adorent le C pour ses pointilleux casse-tête logiques, alors que c'est précisément ce que détestent les autres types de personnalité - dont les N, le type des artistes et écrivains, auquel j'appartiens. Les gens qui ne sont pas des programmeurs professionnels sont souvent d'un type autre que NT. Or les non-NT représentent 90% de la population.

Qu'est-ce que ça implique pour votre marché potentiel si vous décidez de faire des langages de programmation? Soyez réalistes! Voulez-vous viser 10% des gens ou 90% ?

Vous avez déjà vu des programmes en C ? Ils sont bourrés de casse-tête logiques qui, selon les études psychologiques, sont le régal des NT. Il est même fréquent de voir des sources en C presque totalement dénués de commentaires : allons donc, ce serait de la triche! Un petit exemple bien connu nous est donné par une routine en C tirée du noyau d'Unix, dont le commentaire dit : "Je ne m'attends pas à ce que vous compreniez ce que je fais, alors je ne vais pas essayer de l'expliquer." C'est pour moi l'exemple parfait d'un programme en C.

#### Un casse-tête en C

Tenez, faites-vous donc les dents sur cette instruction d'une ligne en C. Attention : ceci est considéré comme simple, élémentaire, facile et enfantin par les maîtres du C.

while ( \*ptr2++ = \*ptr1++ );

Question : que fait cette instruction ?

NB : j'espère que je n'ai pas fait d'erreur en écrivant cette ligne, parce que le C est si incroyablement pinailleur que je ne m'en souviens jamais assez pour écrire du code. [NdT cette ligne est parfaitement correcte.] Un langage ne devrait pas être si difficile à mémoriser. J'ai appris le Basic à 16 ans, et à 33, je m'en souviens encore.

Alors, que fait cette ligne? Eh bien, c'est un casse-tête logique! Et à moins de pouvoir le résoudre, vous n'êtes pas un Vrai Bidouilleur (ou Bidouilleuse)... La foule vous observe! Le suspense monte! J'attends! Alors?...

Vous avez trouvé ?... Non ?... Alors, coupez-moi cette queue de cheval, crétin ! Allez vous raser et prenez une douche! Et dégagez!

Permettez-moi de vous expliquer ce que fait ce code. Ptr2 et ptr1 sont des pointeurs, c'est-à-dire qu'ils désignent un emplacement mémoire, et contiennent l'adresse d'un octet. l'astérisque qui les précède veut dire que l'on travaille non pas avec les adresses elles-mêmes, mais avec le contenu des adresses. Le signe égal entre les deux (l'affectation) signifie qu'on copie le contenu de l'emplacement mémoire pointé par ptr1 dans celui pointé par ptr2. Le ++ qui suit les noms des pointeurs veut dire que nous ajoutons automatiquement une unité à chacun après usage, si bien qu'ils pointeront tous deux sur l'emplacement mémoire suivant. Si ptr1 valait 1, il vaudra 2 après exécution de ptr++. ce qui s'appelle l'auto-incrémentation.

Mais nous sommes dans une boucle "while", qui continue à s'exécuter tant que la condition à l'intérieur de la parenthèse est vraie. Ce n'est que lorsqu'elle devient fausse que l'on passe à l'instruction suivante. C'est en somme une boucle IF-THEN-GOTO (horreur! ). Donc, cette instruction copie vers l'adresse ptr2 ce qu'il y a à l'adresse ptr1, passe aux adresses suivantes, et continue jusqu'à ce que la condition soit fausse. En C, faux équivaut à 0. Or, il se trouve que les chaînes de caractères sont, par convention, délimitées en C par un zéro ! (Ce qui entre en conflit avec la convention du Pascal, pour lequel une chaîne commence par un entier donnant sa longueur. Vous ne pouvez imaginer combien on s'amuse quand on veut, dans un programme en C, appeler des routines en Pascal traitant des chaînes... Comme quand on programme sur Macintosh. C'est l'enfer. Beaucoup de programmes plantent en ne trouvant jamais la fin d'une chaîne dont il manque le zéro final.)

Donc la boucle "while" se termine lorsqu'elle rencontre le 0 de fin de chaîne, parce que 0 équivaut à faux ! Et... j'en tremble d'excitation... puisque ptr1 et ptr2 pointent sur des chaînes... Nous venons de copier une chaîne !...

Super! Génial! C'est-y pas futé comme ruse, hmmm? (Au fait, vos cheveux ne vous démangent pas quand vous ne les lavez qu'une fois par mois?)

Si vous pensez que cette instruction est simple et élégante, apprenez le C. Vous serez heureux de passer le reste de votre vie à déchiffrer du code qui a délibérément été écrit pour être difficile à comprendre. Vous adorerez l'absence de commentaires parce que "le code est auto-commenté", ce qui revient à dire que si vous êtes trop bête pour comprendre, par télépathie, ce que pensait l'auteur du code, c'est que vous n'êtes pas à votre place. Ça me rappelle ce que répondait le milliardaire J.P. Morgan, quand des invités montaient à bord de son yacht et lui demandaient combien avait coûté le navire : "S'il vous faut demander son prix, c'est que c'est au-dessus de vos moyens."

Pour moi, cette ligne se résume tout simplement à une instruction A\$ = B\$ en Basic, ce malheureux langage obsolète pour esprits primitifs (comme dirait Spock, l'archétype du NT). Comme moi.

En fait, à moins de comprendre que la boucle "while" prendra fin lorsqu'elle rencontrera un zéro de fin de chaîne, vous êtes perdu. Et croyez-moi, je vous assure qu'il y a de très nombreux cas où une chaîne peut perdre sa terminaison, causant un plantage. J'en ai tracé tant... Le C est bourré d'aberrations de ce genre. De plus, il est fait pour des gens qui ont peur de frapper un caractère en trop. La boucle ci-dessus pourrait être "expansée" pour être plus claire, comme par exemple :

```
/* Tant que ptr1 ne pointe pas
sur le caractère de code 0 */
while ( *ptr1 ! = ' \0 ' )
{
*ptr2 ++ = *ptr1 ++; /* Copie
avec auto-incrémentation */
```

}

Mais l'écrire ainsi serait trop long. De même, pour ajouter un nombre b à un nombre a, au lieu d'écrire

a = a + b;

le vrai codeur en C écrira

a += b;

Crétonnerre, mais c'est qu'on a économisé la frappe de deux caractères en procédant ainsi!

Pour comparer deux variables, comme dans l'instruction Basic | A = B, on doit en C utiliser une parenthèse et un double égal :

if (a == b) . Et ce, à cause d'un subtil distinguo d'informaticien théoriste qui affirme qu'il faut marquer une différence entre le signe égal qui sert à transférer une donnée (l'affectation, A = B) et la comparaison (le test, est-ce que A = B?). Néanmoins, écrire if (a = b) en C n'engendre pas d'erreur à la compilation, et se contente de semer la panique dans votre programme ! Tous les programmeurs en C sont tombés dans ce piège plus d'une fois.

Or, curieusement, la torture qu'ils s'infligent est considérée comme un signe honorifique et est qualifiée de "bonne". Ce qui me rappelle furieusement les religieux fanatiques se flagellant durant l'épidémie de peste noire du XIVè siècle, en espérant que s'ils s'infligeaient assez de tourments, Dieu leur épargnerait la peste parce qu'ils avaient déjà assez souffert.

Si vous n'êtes pas encore mort de rire, sachez qu'il y a des discussions ô combien sérieuses sur le temps que les programmeurs en C ont gagné en utilisant ces petits raccourcis, comme si les années-hommes de débogage ne comptaient pas. Et si toutes ces turpitudes du C vous emplissent l'un sain dégoût, un peu comme quand on observe le Parlement en action, alors, ami lecteur, apprenez et utilisez le Basic et l'assembleur. Vous avez la chance d'avoir à votre disposition sur le ST les Basic les plus puissants jamais écrits, ainsi que le langage assembleur 68000, qui est le plus propre, le mieux conçu, le plus facile à apprendre que l'on puisse imaginer. Si vous avez déjà souffert sur l'infernal langage-machine du 8088 des compatibles PC, qui est une abomination vomie par le démon, vous êtes probablement dégoûté de l'assembleur. Aussi, je vous en prie, essayez ce pur, ce merveilleux assembleur 68000 (GenSt et GenTT de HiSoft sont formidables). C'est un plaisir, et les gens qui l'utilisent en tombent amoureux (comme moi).

#### Le sondage

Récemment, un torch... heu... un magazine consacré au PC a sondé ses lecteurs pour savoir dans quel langage ils programmaient. Les rédacteurs ont dressé une liste comprenant les langages C, Pascal, Modula, Assembleur, Forth et autres... Quelqu'un fit alors timidement remarquer qu'un brave bouseux nommé Small habitant Denver, dans le Colorado, pourrait bien encore utiliser le Basic. Tous eurent un petit rire dédaigneux et ajoutèrent le Basic "pour compléter la liste".

Je dois préciser que les rédacteurs techniques sont presque exclusivement des NT. Les autres rédacteurs et écrivains sont des N.

Quand les réponses des lecteurs commencèrent à arriver, la rédaction était pâle et tremblante d'horreur. L'énorme majorité des lecteurs programmait en IBM Basic ou en GW-Basic! C'était à n'y rien comprendre. Comment, avec tous les avantages qu'offrait le C, il fallait être un troglodyte criminel pour utiliser le Basic! Mais le fait était que beaucoup de gens, y compris votre serviteur, emploient le Basic. Il faut être masochiste pour utiliser le C. Vous haïssez-vous vraiment?

## Médias hallucinés

Les médias arrivent ainsi parfois à s'autopersuader que le petit terrorisme intellectuel qu'ils imposent dans leur domaine est obéi par le public. Dans ce cas-là, le démenti fut cinglant. Mais sous d'autres formes, la remise des pendules à l'heure tarde à venir. Comme par exemple pour la vague du "politiquement correct" qui sévit aux USA. Une "Personne Correcte Politiquement, , (PCP) ne fera pas l'aumône à un aveugle mais "apportera une aide humanitaire à une personne visuellement handicapée", ne parlera pas d'un vieillard mais l'une "personne du troisième âge", et autres non-sens. Personnellement, je pense que cette "langue de coton" permet de répartir les gens en catégories pour les contrôler plus facilement, ce que je déteste. Il faut se rappeler que PCP désigne en chimie la phencyclidine, un puissant anesthésique pour animaux que fument les drogués, et qui provoque chez eux de redoutables hallucinations les coupant totalement de la réalité. Voilà une belle métaphore pour ce conformisme servile.

## Le basic n'est pas obsolète

Jadis, durant les années 1960, c'est-à-dire il y a tout de même trente ans, le Basic vous permettait d'abuser de certaines choses, comme l'instruction GOTO. Si vous étiez idiot, vous écriviez du "code spaghetti", où le flot du programme partait dans tous les sens. Mettre fin à ces abus devint apparemment le but dans l'existence du Pr. Wirth. Mais il jeta le bébé avec l'eau du bain : le Basic était extrêmement puissant comparé au Pascal.

Loi de Small n° 21 : on peut écrire du mauvais code dans n'importe quel langage, y compris le C et le Basic. Mais de plus, le C est orienté vers le "code à écriture seule", par analogie avec les ROM qui sont des mémoires à lecture seule.

Loi de Small n° 22 : du "code à écriture seule" est du code si illisible qu'il ne pourra jamais plus être déchiffré, même par son auteur trois mois après. Le langage APL est l'exemple classique en la matière, mais le C n'est pas mal non plus.

Loi de Small n° 23 : Écrire pour votre entreprise du "code à écriture seule" dont vous gardez en secret une version commentée s'appelle "prendre une assurance anti-licenciement".

Loi de Small n° 24 : Le logiciel de l'IRS (le service de l'impôt sur le revenu américain) a été écrit par des gens qui sont souvent morts aujourd'hui. ait authentique : parfois, les structures fiscales ne peuvent être changées parce que personne ne sait plus où faire les modifications dans le logiciel. l'IRS continue à embaucher des programmeurs.

Pour en revenir au Basic des années 60, on pouvait très bien abuser de l'instruction GOTO, tout comme l'auteur d'un article peut très bien abuser des digressions et parenthèses pour s'écarter du sujet... Hem... heu... Reprenons.

Des gens bien inspirés ajoutèrent au Basic ce qui était alors le nec plus ultra, entre autres les structures de programmation, de sorte qu'on pouvait les utiliser si on le désirait. J'estime quant à moi que se prosterner devant les diktats de la programmation structurée est idiot. Mais peut-être votre compagnie a-t-elle des règlements stricts, disant que vous devez être à votre bureau à 9 h, vous laver les mains après avoir été aux toilettes, et utiliser des boucles " while" au lieu de IF-THEN-ELSE. Dans ce cas, pauvre hère, toutes mes condoléances.

Et si jamais un comique vous dit que le Basic est obsolète, voici quelques questions à lui poser pour engager la conversation:

- ① Que pensez-vous de l'affaire Ben Barka?
- ① La France va-t-elle vraiment guitter l'OTAN ?
- ① Où est votre pendentif Amour et Paix?
- ② Avez-vous été au concert de Jim Morrison et Jimi Hendrix ?
- ② Est-ce vous vous habituez aux Nouveaux francs?

Parce que le malheureux en est irrémédiablement resté aux années 60.

Quand je veux écrire quelque chose très vite, je l'écris en Basic, qui est une sorte de super-assembleur et qui fait pour vous beaucoup de travaux fastidieux. Quand je veux faire un programme qui va le plus vite possible, je l'écris en assembleur, et il pulvérise tout ce qui peut être écrit en d'autres langages. Je n'écris pas des saletés qui se traînent à la même allure que ce que peut faire n'importe qui, allure qui

n'a guère changé depuis 1980. (Vous savez, la même vitesse de défilement affligeante, la même vitesse de recalcule gastéropode...)

Loi de Small n° 25 : Tandis que le matériel devient de plus en plus rapide, les programmeurs engendrent un code de plus en plus mauvais, de sorte que la vitesse d'exécution reste constante. Le code engendré par les compilateurs C++ modernes, par exemple, est incroyablement inefficace. Je vous défie de me contredire après avoir utilisé System 7.0 sur un Mac à 8 MHz.

Loi de Small n° 26 : La raison pour laquelle Windows 3.0 est considéré comme "utilisable" est qu'il tourne désormais en général sur de puissants 386 ou 486 à plus de 16 MHz. l'ancienne version de Windows, qui fut condamnée par la presse comme étant "trop lente", ne bénéficiait pas encore de ce matériel. Je parie que Microsoft s'est contenté d'attendre ! Windows n'est d'ailleurs pas codé en assembleur.

Loi de Small n° 27 : Un programmeur digne de ce nom peut créer des codes époustouflants en assembleur, car rien qu'en évitant les grossières inefficientes des générateurs de code du C ou du Pascal, le programme sera beaucoup plus rapide. C'est ainsi que l'on crée des légendes. Prenez l'éditeur de texte Tempus, vous avez vu sa vitesse de défilement ?!

Loi de Small n° 28 : Comment croyez-vous que j'ai fait un émulateur Mac ? A votre avis, pourquoi est-il si rapide ?

Loi de Small n° 29 : Pensez-vous que l'on puisse gagner sa vie en suivant le troupeau du conformisme ? Si un avis est donné gratuitement dans un magazine, il vaut probablement ce que vous l'avez payé.

## Mais, Dave, le C est portable

Ben voyons. Si un bidouilleur du C vous affirme cela, regardez-le droit dans les yeux et faites-le répéter Je sais ce ce qu'il en est, j'ai fait des portages de sources C. C'est presque aussi amusant que de désenliser sous la pluie une voiture qui a de la boue jusqu'aux portières.

En théorie, vous devriez pouvoir prendre un code source écrit en C sur une machine et le recompiler sans problème sur une autre. A vrai dire, un source en C est partiellement portable entre machines. Pour l'être totalement, le source devrait n'utiliser que le plus petit commun dénominateur entre les machines, ce qui veut dire qu'il ne pourrait utiliser par exemple les caractéristiques particulières du ST. Ou alors, il faut recourir à l'écriture de modules "pilotes" spéciaux, ce qui réfute l'idée même de portabilité.

Et si vous réussissez à faire votre portage entre deux machines ? En ce cas, bravo ! Vous avez maintenant deux machines se traînant bien en deçà de leur vitesse optimale, parce que les compilateurs C sont encore très peu malins quand ils engendrent le code machine. Rappelez-vous que les compilateurs ne sont que des programmes et n'ont en eux rien de l'intelligence humaine pour les aider à optimiser. [NdT: sur station de travail Unix, certains compilateurs disposent d'optimiseurs qui, eux, appliquent des astuces de programmeurs chevronnés pour améliorer leur code machine.] Je dirais que la science des compilateurs a encore besoin de 50 ans de progrès avant de concurrencer les codeurs humains. Pour l'instant, n'importe quel programmeur peut améliorer considérablement du code compilé.

Note : la "remarquable augmentation de vitesse" des ROM 128K du Mac par rapport aux 64K est due à un programmeur qui a réécrit en assembleur les routines du Quickdraw compilées en Pascal.

Bref, je suis persuadé que faire un programme qui se traîne sur deux machines n'est pas un exploit. Le faire foncer, voilà qui est bien !

Prenez le très portable X-Windows, l'environnement graphique pour stations de travail. Il est admis par presque tous que ce système est lent, lourd et très touffu. Il faut des processeurs graphiques spécialisés pour avoir des vitesses acceptables, ce qui fait le bonheur des fabricants de puces. [NdT: il est vrai qu'il faut des machines d'au moins 20 MIPS pour avoir sous X-Windows des vitesses de

réaction comparables à celles du GEM sur ST.]

Maintenant que j'y pense, je signale que le Basic de l'IBM PC est plutôt rapide, et plutôt portable... même sur le ST.

#### La vérité

La vérité est qu'il y a un hiatus entre un concept et son implémentation, à savoir la programmation structurée.

Structurer un programme, tout programmeur sain d'esprit le fait, mais pas forcément dans la forme imposée par le terrorisme intellectuel ambiant. Tenez, mon style particulier de structuration est que le début d'un programme est la première chose qu'il fait, la dernière ligne est la dernière chose. Si je débogue quelque chose qui est fait à mi-chemin dans le programme, je commence par regarder au milieu du listing. Oui, j'utilise des sous-programmes, je les mets d'habitude au bas du listing, mais aussi parfois près de la routine qui les utilise. Cela rend le débogage beaucoup plus facile, ce qui me permet d'achever le programme plus vite.

La structure des programmes Pascal me paraît insensée. Elle est sens dessus-des-sous. d'abord, vous devez écrire tous les sous-programmes. Ensuite, au fin fond du listing, vous écrivez le code qui les appelle. Et c'est censé être cela, la programmation "haut-en-bas" ? Calembredaines ! Oui, je sais que le Pascal est un compilateur mono passe, mais c'est une mauvaise excuse, mon gaillard.

Je veux utiliser un langage dans lequel je puisse donner des ordres à la machine, et non me livrer à des contorsions pour lui faciliter la tâche.

De même, structurer un article, tout auteur sain d'esprit le fait, mais pas forcément de manière conformiste. Le "flot de contrôle" de l'article ne doit pas être exagérément obscurci par des digressions, des écarts et des considérations hors-sujet. J'ai délibérément écrit cet article de manière non structurée, en y ajoutant force digressions, de manière à bien montrer l'analogie avec la programmation. Dans d'autres articles, je vais rigoureusement du début à la conclusion de manière linéaire, ce qui montre que je peux aussi écrire - ou coder - de manière structurée. Mais l'important, c'est le plaisir du lecteur. A votre avis, de l'agrément ou de la rigueur, qu'est-ce qui prime ?

Ensuite, n'oubliez pas que la structure, c'est votre affaire, pas celle de Wirth (et je suis certain qu'il m'approuverait). Vous pouvez écrire de l'assembleur structuré qui est facile à déboguer si vous le souhaitez. Ou bien vous pouvez pondre le plus abominable des plats de spaghettis en C si vous le voulez, et le soumettre à vos confrères dans le cadre des concours "Que fait ce code?" organisés chaque année, où des bouts de codes particulièrement obscurs sont soumis à la sagacité de programmeurs C.

Le langage dans lequel vous codez n'a rien à voir avec vos capacités de bidouilleurs. Un langage de programmation est censé être une façon conventionnelle de donner des instructions à un ordinateur, et doit pouvoir être écrit et relu par des personnes différentes. Il n'est pas censé être obscur, bizarre et plein de pièges subtils (ça, ce sont les attributs de la magie). Ce qui importe en matière de créativité, ce sont vos rêves et votre volonté de repousser les limites, de faire ce que personne ne croyait possible (un bon exemple est fourni par presque toutes les créations de Codehead Software). Le reste n'est que vaine discussion entre peintres pour savoir quel pinceau est le meilleur. Certes, il se trouve des gens pour apprécier ces discussions, mais elles ne sont guère au cœur du débat.

Et si vous tenez absolument à ce que vos créations portent le fardeau que ne manquera pas de leur imposer le C, c'est votre affaire, allez-y. Moi, je dis seulement que je préfère voir ma machine dépenser utilement chaque cycle d'unité centrale.

#### Rebellez vous

Alors, rebellez-vous. A la prochaine réunion de votre groupe d'utilisateurs, interrompez les monologues des gourous du C, et dites : "Moi, je pense que le C est inutilisable et qu'on s'en souviendra comme d'un cauchemar dans dix ans. Quant aux structures, c'est moi qui les utilise, et non l'inverse."

Voyez ce qui arrivera. Vous devrez peut être couper votre queue de cheval, vous doucher et vous raser. Mais vous serez alors un véritable bidouilleur.

## La cible de vos tomate

Si vous voulez me lancer des tomates, voici mes adresses télématiques :

GEnie: DAVESMALL

Compulsive: 76606,666 ou 76004,2136 Internet/USENET :dsmall@well.sf.ca.us

# Rationalisme (première partie)

Le Falcon débarque, certes, et l'image d'Atari renoue enfin avec celle de bon vieux temps: une firme à contre-courant, mais aux machines hors du commun. Mais, voici à peine quelques mois, les ataristes étaient plutôt moroses. Des ventes en stagnation, une machine vieillissante, ne pouvait guère engendrer l'euphorie. En fait, le monde entier sauf quelques semblait snober Atari, artistes bidouilleurs... parmi lesquels il y a Dave. Or, devant l'apparent immobilisme d'Atari, Dave lui-même a connu des moments de doute, au point de visiter (horresco referens) des magasins de PC! Oui, vous avez bien lu! De cette visite aux enfers. Dave sortira-t-il indemne? Les méchants



vendeurs réussiront-ils à lui laver le cerveau et à lui fourguer un compilateur Pascal? Insoutenable suspense...

#### **BIBERONS POUR ATARISTES**

En ce moment, mon fils Jamers fait de son mieux pour me déconcentrer en hurlant à ma femme Sandy "Veut bibon!... VEUT BIBON!..." Traduction: "très chère mère, il me serait fort agréable que vous me procuriez un biberon." (De plus, Jamers croit que répéter quelque chose avec force lui donne plus d'importance).

Mais voilà. Jamers (trois ans) a un rhume et se sent vaseux, et hélas, un "bibon" ne lui fera aucun bien. Mais il n'en a cure: depuis des temps immémoriaux pour lui (plusieurs mois), il a l'habitude d'aller au lit après un biberon réconfortant, et il veut ce réconfort.

Ce qui nous amène à comparer Jamers à l'utilisateur moyen d'ordinateur Atari (ou de tout autre marque d'ailleurs). Cet utilisateur a acheté une machine quelque peu marginale et veut être conforté dans l'idée que c'est un bon achat. Cela n'est nullement indispensable, et ne lui apportera aucune aide, tout comme le "bibon" pour soigner un rhume. Mais psychologiquement, l'enfant et l'utilisateur recherchent tous deux une sensation rassurante, chaude et confortable.

Car pendant des années, les utilisateurs ont été bombardés de propagande affirmant qu'il leur fallait le meilleur et le plus rapide des ordinateurs. Ce qui est un but à jamais inaccessible, d'ailleurs, puisque la situation change tous les six mois environ et qu'il y aura toujours plus récent. (Tenez, le portable sur lequel j'écris ceci était considéré comme excellent lorsque Sandy me l'a offert, mais à présent, il y a beaucoup mieux.) Et malheureusement, cette propagande est orientée vers les Mac et les PC, pas vers les Atari. Quand ces utilisateurs ouvrent un magazine "généraliste" d'informatique, ils sont inondés de pubs pour des 486 à 33 MHz, des 386 à 40 MHz, des pré-annonces du processeur P5 (le futur 586) et Dieu sait quoi encore. Il y a des wagons de magazines PC, offrant aussi bien des conseils aux débutants que des cours de programmation en assembleur et en C. (Ce qui est d'ailleurs préférable: vu la nature de l'assembleur du PC, le C est un meilleur langage. Et vous savez en quelle estime je tiens le C.) Les librairies techniques regorgent de livres expliquant le maniement des interfaces utilisateur du PC (il faut bien ça, elles sont remarquablement mauvaises) et celui de centaines de logiciels pour PC (Microsoft affirme réaliser le quart de ses ventes en livres!)

En comparaison, il n'y a guère de périodiques sur le ST, et peu de livres. Ce n'est certes pas le désert, mais on a la sensation d'être à l'écart du marché. Et cela provoque un profond malaise chez les

utilisateurs d'Atari, car on leur tend un biberon rempli d'un indigeste lait au PC (du lait bleu, pouah!). Or ils voudraient bien qu'on les réconforte et qu'on leur dise que leur machine est encore belle et bonne. Hélas, Atari est resté longtemps sans sortir de nouvelles machine, et les signes décourageants se sont multipliés. Computer Shopper, revue américaine pour acheteurs de micro-informatique exigeants, a par exemple abandonné sa section sur l'Atari ST, l'une des seules dans la presse généraliste. Je ne dis pas cela pour démoraliser les ataristes, je trouve au contraire naturel de rechercher l'approbation de ses semblables.

Et, n'obtenant pas ce qu'ils veulent, les utilisateurs d'Atari se sont mis à vociférer, à l'instar de Jamers. Je ne voudrais pas être à la place des gens chez Atari qui ouvrent le courrier, où l'on trouve dans le meilleur des cas des suggestions et des plaintes. (Le courrier de Gadgets By Small menace déjà de me noyer.) Sans compter que l'abondance de lait au PC a tendance à décourager les développeurs sur Atari. Lorsqu'on vous fait ingurgiter quotidiennement un régime de Windows-sur-486-c'est-génial, il devient dur de conserver un intérêt pour le ST. Et quand on apprend que Windows a vendu 3 millions d'exemplaires de Windows, il est dur de ne pas vouloir prendre le train en marche (de fait, beaucoup de développeurs ont déjà sauté dans un wagon).

Un chose que les utilisateurs d'Atari voudraient bien voir, par exemple, c'est de la publicité [NdT: aux USA, le coût de la pub et son peu de résultat face aux Américains saturés explique pourquoi Atari préfère ne pas en faire, sauf très occasionnellement]. Pourquoi? En partie pour se voir confirmer qu'Atari est toujours dans la course, mais aussi pour pouvoir affirmer aux copains qu'on a acheté une machine qui n'est ni un Mac, ni un PC, et qu'on en est ravi, et que d'abord, Atari est une grande compagnie, regardez leur pubs... Je soupçonne que la raison pour laquelle tant de possesseurs de ST, qui ne sont donc plus à convaincre, voudraient voir de la pub pour Atari à la télévision, et qu'ils veulent que leurs amis cessent de leur dire "ah oui, la machine de jeux" quand on leur parle d'Atari. Il n'y a là de la part d'Atari un manque de support psychologique, ainsi qu'un manque d'attrait face aux charmes insidieux déployés par les magazines américains pour Mac et pour PC. De même que Jamers tentera d'obtenir un biberon auprès de moi si sa mère ne lui en donne pas, les utilisateurs d'Atari, mûs par un même besoin désespérés d'être rassurés, essaieront d'obtenir quelques consolations auprès des Mac et des PC.

A tel point que j'en suis moi-même victime. La propagande en faveur des Mac et des PC, idéologie dominante de la micro, se fait si insidieuse (elle est probablement subliminale) que je suis parfois tenté, je l'avoue, d'aller voir ce qui se passe dans ce domaine. Et je fais alors une vérification de réalité.

Je sais que c'est idiot, mais périodiquement, je gaspille mon temps à lire des magazines pour Mac et PC. Attention, je sais parfaitement ce que valent certains: au moins deux importants magazines américains, par exemple, proposent aux éditeurs d'acheter un banc d'essai flatteur de leur dernier produit! Et un autre ne publie rien d'autre que les communiqués de presse des compagnies, façon Pravda (qui, je le rappelle, signifie "vérité"). N'empêche que mon jugement se trouve diminué par ce véritable lavage de cerveau

#### **LECHE-VITRINES**

C'est ainsi que tous les six mois à peu près, je l'avoue, je tombe dans le piège des sirènes qui chantent dans tous ces journaux, et je m'en vais "regarder" les PC et les Mac. Mais rassurez-vous: à la fin de la journée, je suis de nouveau convaincu que rien ne vaut le monde Atari. Car en un jour, je vois un incroyable amoncellement de matériels et de logiciels incompatibles, et nombre de vendeurs ineptes! (Le plus nul techniquement d'entre eux fût également le plus gentil, il me confia qu'il venait juste de quitter un emploi de vendeur de résidences de vacances.) Je vous recommande cette expérience de tournée des boutiques informatiques pour réaliser les avantages de nos machines Atari, et pour relativiser les problèmes de la marque.

La première chose que font les vendeurs dans ces boutiques est de jauger votre fortune d'après vos vêtements. Avez-vous l'allure d'un acheteur de système haut de gamme? Moi, je n'ai sans doute pas l'air d'un gros client avec mon jean et mon T-shirt de "Soldier of Fortune" [NdT: "Soldier of Fortune" (mercenaire), magazine pour baroudeurs et amateurs d'armes, très conservateur, vend des T- shirts

dont raffolent les anticonformistes car ils cultivent l'art de la provocation, du genre "Paix... grâce à une puissance de feu supérieure" ou "Quand tu veux, Khadafi!". S'adresser au journal pour les commandes.]

Il me faut donc commencer par secouer un peu les vendeurs en leur lançant des mots de jargon technique comme autant de missiles SCUD, et en m'extasiant sur le confort qu'on éprouve le samedi à sortir sans costard-cravate, loin du bureau. Mon boulot? "Nous faisons des cartes accélératrices pour PC" (presque exact). Mon ordinateur? "Oh, nous utilisons des Mac II FX en réseau avec une LaserWriter NTX." Ils pâlissent un peu et m'éloignent du rayon des clones 286 jetables pour me guider vers celui des 486 musclés.

"Et vous ai-je présenté ma femme Sandy? (qu'ils ont jusqu'alors soigneusement ignoré). C'est la présidente de notre compagnie, et elle gère tout le marketing et les achats de machines. Elle est diplômée en informatique." A ce stade, les vendeurs en sont presque à faire pipi dans leur chaussures. Inutile de dire qu'ils essaient de rattraper le temps perdu avec Sandy. Laquelle leur lance son fameux Regard Énigmatique.

## **IGOR, ABAISSE LE LEVIER!**

Ils m'amènent alors devant une boîte monstrueuse qui pourrait servir d'abri antiatomique à un 520 ST, et l'allument. Les lumières faiblissent, le ventilateur me décoiffe. Une bonne année plus tard, l'écran s'illumine enfin, et affiche un monceau d'incohérences sur des CONFIG.SYS et autres AUTOEXEC.BAT. Je me prépare au choc, ça y est, la voilà, "C:>", ô exaltante invite!

Eh oui, me voici revenu quinze ans en arrière, à l'époque où le CP/M me lançait son "A:". C'est beau, le progrès. Cela me fait penser à un certain H. G. Wells. La seule analogie qui me vienne à l'esprit pour qualifier ce gâchis, est un V-8 monté sur un patin à roulettes. Bien sûr, c'est très rapide, mais l'utilisateur subit toujours la même ligne de commande grossière et peu ragoûtante. Et qu'on me dise à quoi sert une commande DIR qui fait s'afficher une liste de fichiers défilant si vite à l'écran qu'on ne peut en voir que la fin.

Et peut-on imaginer qu'au lieu de cliquer sur un dossier avec une souris, je dois taper le nom complet de ce sous-répertoire? Je ne peux pas traîner un fichier avec la souris, il faut que je me préoccupe de "chemins" et autres fadaises! Et quand je fais une faute de frappe, ce qui m'arrive souvent, je dois retaper en entier cette stupide ligne! Mais nous sommes retournés à l'âge de pierre! Où sont les cartes perforées, les bandes magnétiques, les tableaux de lampes clignotantes? Où sont passés Robby le robot et les IBM 360? J'ai subi toutes ces nuisances au collège et je ne regrette vraiment pas leur disparition.

Quant à copier un groupe de dossiers, c'est un exercice de pur masochisme. Pour y parvenir, il me faut une demi-heure, un éditeur de texte sophistiqué avec macro-commandes, la redirection des entréessorties et quelques tours de passe- passe. Sur ST, bien sûr, il me suffit de sélecter les dossiers à copier avec shift + clic et de les traîner. Et ne me parlez pas de XCOPY: il ne vérifie pas la copie, même si on lui met l'option qui est censée le lui ordonner. J'en ai fait la triste expérience, et j'ai passé une journée à chercher quels fichiers XCOPY avait vérolés.

J'affirme que le roi est nu!

#### **WINDOWS**

A ce point, les vendeurs lancent fièrement Windows. Ils espèrent que je vais m'extasier, sans doute parce que je l'ai vu remplir l'écran avec une couleur de fond sans se planter. Hélas, j'utilise quelque chose de bien supérieur depuis 1985, et il m'en faut plus pour m'épater. Windows m'apparaît être une version revue et corrigée par un service de marketing, c'est à dire une mauvaise version.

Loi de Small n°8: aucun bon programme n'est jamais sorti d'une étude de marché, sinon par accident. Demandez donc à quelqu'un qui travaillait chez Atari avant l'arrivée de Tramiel. Et essayez le jeu Pacman sur VCS (!!!).

Loi de Small n°7: les meilleurs logiciels sont des œuvres d'art amoureusement fignolées, généralement par un à deux programmeurs au maximum.

Windows procure l'inimitable sensation de marcher dans de la boue, qui colle aux chaussures et qui leste chaque pied de 10 kilos. La moindre manip' dure une éternité. Il faut fouiller dans des menus rien que pour copier des fichiers. Le pire est que Windows a des réactions lentes. Or, un débutant en marketing informatique vous dirait qu'il est très important qu'un logiciel ait une grande vitesse apparente de réaction. S'il ne tenait qu'à moi, je m'attellerais à la réécriture en pur assembleur de certaines parties de Windows... Je connais bien le genre de code que fait Microsoft, je l'ai débogué bien des fois sur Mac.

(Message personnel pour Darek Michocka, développeur de GEMulator: tu n'es pas mauvais en assembleur, n'est-ce pas? Alors, je t'en prie, viens en aide aux 3 millions de possesseurs de Windows et optimise-le. GEMulator est une idée intéressante, mais ces malheureux ont besoin d'aide, ils vont prendre racine devant leurs PC.)

#### **DURS, LES DISQUES**

Je cuisine ensuite les vendeurs quant au disque dur de ce super-PC. Est-il à la norme SCSI, propre et net? Non, au format IDE ou ST-506. Ouais, génial (hummphf). Puis-je en mettre un plus grand? Puis-je mettre plus de deux disques durs dans le PC? Ils n'en savent rien, et je n'en sais toujours rien d'ailleurs. Lecteurs, si vous avez la réponse, contactez-moi! Il me semble qu'il faudrait une deuxième carte d'interface de disque, et les interruptions entreraient alors en conflit. Si vous savez, n'hésitez pas, mes adresses sont en fin d'article.

Comparez cela à votre ST, sur lequel seuls quelques disques particulièrement exotiques ne peuvent se brancher sur l'unité centrale moyennant un tournevis, quelques minutes et le logiciel de formatage d'ICD. En ce qui me concerne, j'ai... voyons... six disques pour le développement de la carte SST et de Spectre, partitionnés en aires de développement et de test. Leur sauvegarde est d'ailleurs cauchemardesque.

De plus, les disques durs du PC sont une source d'amusement et de profit, si si. Mon copain Alex Pournelle gagne sa vie à récupérer des données sur des disques durs endommagés. Il peut remplacer les yeux fermés le circuit intégré qui grille le plus souvent sur les disques Seagate (le circuit de commande du moteur pas-à- pas).

## **RECUPERATION DE DONNEES**

Il y a quelques mois, j'ai passé trois nuits, de 21h à 3h du matin, à récupérer des données médicales cruciales sur le disque dur du PC de mon frère, qui est médecin. Ce fût épouvantable. Apparemment, ce stupide contrôleur de disque dur stocke quelque part les caractéristiques, comme son nombre de pistes, et je n'ai pas réussi à trouver où! Et croyez-moi, j'ai essayé. Et le contrôleur m'interdisait d'accéder au-delà d'un certain point (les données à récupérer étant bien sûr dans la zone interdite). Oui, je connais les RAM statiques CMOS non volatiles des PC-AT ainsi que les logiciels de configuration, mais cela n'a pas marché: ce clone là était incompatible. D'ailleurs, c'était un clone de PC- XT.

Cela représente parfaitement à mes yeux la philosophie du PC: "nous savons mieux que vous ce qu'il vous faut". Le contrôleur refusait de me laisser accéder aux données parce qu'il savait, et impossible de l'en faire démordre. D'ordinaire, je flinque les disques qui sont aussi bornés (j'ai pour cet usage un

AK-47 et un M-16, je vous raconterai ça un de ces jours).

Vous savez ce que j'ai dû faire pour résoudre ce problème? J'ai dû lancer un formatage du disque, le laisser tourner sur les vingt premières pistes à peu près, et éteindre le PC. Ceci a eu pour effet de réécrire les tables de paramètres du disque et de me laisser enfin accéder aux données brutes. Il m'a ensuite fallu copier les données secteur par secteur (elle étaient Dieu merci, sur des secteurs contigus) sur un autre disque, puis sur disquette, en utilisant les Norton Utilities en mode "Niveau expert de Monsieur Spock - vous assumez toutes les responsabilités". Enfin, j'ai détruit les données inutiles sur les derniers secteurs. Il m'a fallu aller explorer les tables d'allocation de fichiers, ce qui n'est pas drôle, et c'est pourquoi cela m'a pris tant de temps. Mais j'ai tout récupéré.

Je sais parfaitement que couper le courant au beau milieu d'un formatage glace le sang à quiconque connaît les disques durs. Mais c'est vraiment la seule astuce qui me restait, j'avais littéralement vidé mon sac à malices. J'avais par exemple connecté un second disque au contrôleur, l'avais formaté, puis avait interverti les câbles reliant le contrôleur aux disques pour tenter de tromper ce circuit stupide. J'avais même écrit des programmes pour aller directement modifier les paramètres dans les registres du contrôleur. Tout cela en vain. Machine stupide et bornée!

Sur ST, l'utilitaire SUPEDIT de Supra m'aurait arrangé ce problème en un clin d'œil. Je crois d'ailleurs ce programme indispensable à tout possesseur de disque dur. Il figure en bonne place dans ma trousse de survie spéciale pour salons informatiques, et c'est l'utilitaire dont je me sers le plus. Très chaudement recommandé!

J'arrête pour ce mois-ci, mais j'accueille déjà volontiers vos réactions.

Traduction et adaptation : Password 90

Titre original: Rationality

# Rationalisme (deuxième partie)

La Saga continue avec ce mois-ci la seconde partie d'une passionnante pluie de considération sur le rationalisme sous toutes ses formes...

Dave poursuit son exploration du monde des PC et des Mac, machines dont il possède, entendonsnous bien, plusieurs exemplaires. Mais si ces machines ont certes des qualités par rapport au ST, Dave affirme que les défauts l'emportent, du moins pour le bidouilleur qu'il est. Dave a travaillé sur de gros systèmes, comme sur toutes sortes de micro-ordinateurs, et en a dépiauté suffisamment pour connaître les entrailles de la plupart d'entre eux. Et de ce point de vue, le Mac et le PC ne le satisfont pas. Le mois dernier, Dave nous a confié pourquoi il avait peu d'affection pour le matériel des PC, notamment leurs disques durs. Il y a d'autres raisons, logicielles quant à elles...

#### **GRAND ECRIVAIN**

Supposons que vous ayez un PC et que vous vouliez y connecter un moniteur à grand écran. Parfait, vous en achetez un. Ilvous faut à présent une carte vidéo qui puisse piloter ce moniteur. Mais voilà, les fréquences de balayages, les bandes passantes, les résolutions, c'est toute une science qui remplit des manuels entiers! Votre moniteur peut-il avaler ce qu'envoie la carte sans faire appel au mode entrelacé, dont le scintillement vous conduirait à la cécité? D'ordinaire, les vendeurs ont une liste des cartes qui fonctionnent avec un moniteur donné, écrite avec un vocabulaire réduit à l'intention des néophytes.

Eh bien, si votre carte n'est pas sur la liste, extorquez au vendeur une garantie écrite selon laquelle vous pouvez la lui rendre et vous faire rembourser, je vous en conjure!

Très bien. Maintenant, essayez de trouver pour votre application favorite un programme pilote qui vous permettra d'utiliser la carte! Quoi, dites-vous, les applications ne savent pas l'utiliser spontanément? Ouarf ouarf. Je ris. Le plus souvent, les fabricants de cartes vidéo écrivent des pilotes pour quelques applications répandues, comme Lotus et AutoCAD, et c'est tout. Point final. Pourquoi? Parce que les autres n'ont pas besoin de grand moniteur, répondent-ils. Le traitement de texte qu'il m'arrive d'utiliser sur PC n'était pas sur la liste des pilotes pour les cartes vidéo que j'ai vues, si bien que j'ai laissé tomber le PC. Voyez-vous, un ordinateur personnel est censé s'adapter à vous, et non l'inverse.

Quand j'ai expliqué mes besoins aux vendeurs de la boutique de PC où j'étais entré, j'ai l'impression qu'on m'a pris de haut et éconduit. C'est typique : les spécialistes des PC savent ce que je veux mieux que moi (mais mon argent est resté dans ma poche). Je vois le genre : "Eh, Willie, ce comique avec le jean et le T-Shirt voulait un traitement de texte sur grand moniteur!... Ha ha ha!... Allons boire une bière et reluquer les serveuses du bar en sortant du boulot!... Avec un peu de chance, il y aura une bagarre!... Passe-moi une clope, veux-tu?"

Sur le ST (ou le Mac), les programmes bien écrits marchent sans problème sur grand écran, ou sont le plus souvent mis à jour dans le cas contraire. Car sur les ST, la capacité à utiliser un grand moniteur est incluse dans les ROM, dans le GEM (les puristes me permettront cet amalgame entre le GEM et le VDI). Toute application peut donc y faire appel, si le programmeur n'est pas trop flemmard. Sur les PC, il n'y a presque rien dans les ROM, et chacun développe dans son coin, sans standard.

Je vous défie de trouver une imprimante pour PC qui fasse ce que vous voulez, particulièrement une laser. Quand vous voulez acheter une application, il vous faut vérifier au dos de la boîte qu'elle supporte votre marque d'imprimante, et spécifiquement votre type, votre modèle et votre numéro de version de ROM. Sans quoi, préparez-vous à une hilarante partie de "jeu de l'imprimante", un sujet auquel chaque magazine pour PC consacre une rubrique tous les mois. D'ailleurs, des applications utilisant cette imprimante, vous en trouverez péniblement une, certainement pas deux.

Imaginez-vous des êtres velus, vêtus de peaux de bêtes, serrés dans l'entrée l'une caverne près d'un

feu, autour du standard pour imprimantes Epson, rognant de la viande crue sur des os, en marmonnant "Gah ounk, MX-80". On en est resté là! Perdu dans une multitudes de standards d'imprimantes, les concepteurs l'applications se raccrochent à l'antique jeu de commandes Epson et vous forcent à utiliser des imprimantes ultramodernes dans ce mode s'ils n'ont pas explicitement prévu son support. Tenez, rien ne peut me terroriser davantage que la pensée d'utiliser une application engendrant du PostScript sur un clone PC avec une imprimante contenant un clone de PostScript, avec pour toute aide un

malheureux nigaud de vendeur en guise de support technique. Je préférerais encore me faire marquer au fer rouge. J'ai utilisé PostScript sur son terrain de prédilection, le Mac, et c'est déjà assez dur. (Sandy arrive d'habitude à faire marcher notre LaserWriter. Mais nous avons dû installer un disque dur dans l'imprimante! Vous vous rendez compte? Bientôt, je suppose que c'est le clavier qui aura besoin de son disque dur.)

Le manque de standards sur le marché du PC a conduit celui-ci à l'anarchie totale. A la rigueur, si vous vous restreignez à un certain nombre de programmes sans quitter leurs domaines de compatibilité, vous vous en sortez. C'est ce que font les grosses firmes, qui normalisent leurs PC et y mettent le traitement-de- texte-que-tout-le-monde-utilise (qui est en général celui que détestent les secrétaires, selon la loi de Murphy). Idem pour les réseaux, qui ne fonctionnent qu'avec certains logiciels. Cela ressemble beaucoup au bas moyen-âge, où l'on pouvait prêter serment d'allégeance à un seigneur et lui payer l'impôt, ou bien tenter de survivre seul, pour échouer le plus souvent. Le programmeur sur PC doit donc rester fidèle au royaume d'AutoCAD ou au duché de Lotus, et ne jamais errer hors de ces fiefs.

Il faut un sacré cran pour installer un nouveau TSR (équivalent d'un programme du dossier AUTO) sur un PC. Il y en a tant qui entrent en conflit mutuellement! Sidekick de Borland était le pire, bien qu'il y ait peutêtre plus dangereux encore à présent - je ne me tiens pas vraiment au courant de ce qui sort sur PC. En fait, la plupart des PC ne servent qu'à faire tourner deux ou trois programmes leur vie durant. C'est bien fait pour eux? Peut-être, mais c'est de l'esclavage.

Les ST échappent à ce sort. Dans le monde du ST, nous avons quelques standards. Pas assez dans certains domaines, et le GEM n'est ni complet ni totalement débogué. Mais on s'y fait. Et grâce au Ciel, il n'y a pas surabondance de standards, comme pour le Mac. De plus, les ST sont incroyablement bon marché, assez faciles à utiliser, et on parvient à leur

faire accomplir une tâche sans utiliser le dixième des jurons qu'on lâche sur un PC. Si vous voulez un échantillon de ce qu'est le travail sur PC, écrivez un fichier ASSIGN.SYS et faites-le marcher. Les possesseurs de PC y passent leurs journées.

#### **LE MAC**

Ce qui nous amène au marché du Mac. Celui-ci souffre d'un constructivisme outrancier. Un syndrome stalinien sévit dans la documentation de programmation du Macintosh, où "tout ce qui n'est pas obligatoire est interdit". Ses auteurs pensent sincérement qu'ils ont trouvé le Chemin de la Vérité, et qu'il est bon de forcer tout le monde à suivre ce chemin.

En réalité, la gestion de la mémoire est si confuse que les meilleurs programmeurs s'y perdent et se mélangent les pinceaux dans leurs pointeurs qui finissent régulièrement par accéder à l'adresse zéro. Je le sais, je dois réparer les plantages de leur code. La firme Apple, à mon avis, est allé beaucoup trop loin en spécifiant ses standards et s'est montrée totalement incohérente à certains endroits. Par exemple, leur documentation bavasse sur des pointeurs secrets en mémoire en vous interdisant strictement d'utiliser ces informations, et à présent, nous avons des tonnes de logiciels qui plantent

sous System 7 en 32 bits parce que leurs auteurs ont fait usage de ces informations... Mais écrire un simple pilote d'impression est un grand secret que seules quelques compagnies ont découvert! (Sans cela, il y a longtemps que le Spectre pourrait imprimer en 300 points par pouce sur la laser du ST.)

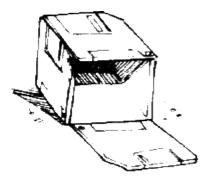
On avoue à présent qu'il faut une bonne année pour qu'un programmeur se forme au développement sur Mac. C'est effarant. C'est beaucoup trop long lorsqu'un programmeur coûte 50 000 dollars par an! [NdT: ce chiffre -300 000 F - correspond au salaire d'un bon programmeur aux USA. En France, le salaire est moins élevé, mais les charges sociales et les coûts indirects multiplient aisément par deux ce coût pour l'employeur.] Une raison de ce temps d'apprentissage élevé? Dans la documentation du Mac, il manque une chose vitale, un exemple de programme qui tourne. Savez-vous qu'ils ne se soucient même pas de fournir des exemples en assembleur de programmes simples? Mark Russel Benioff en avait écrit quelques-uns pour eux, mais le projet a été annulé.

C'est une conception macho de la programmation. Soit vous avez l'étoffe du Programmeur Mac, et vous apprenez en dépit de tout, soit vous ne l'avez pas. Je ne l'ai pas. Savez-vous que si vous commandez le kit Apple de développement SCSI, ils vous envoient un code source qui est, je cite, un "exemple non fonctionnel" de pilote de périphérique SCSI? Ou encore, sachez que dans le monde du Mac, on ne s'abaisse jamais à utiliser de vulgaires nombres. On utilise des "équivalents", des noms symboliques qui seront remplacés par les nombres correspondants lors de l'assemblage. On ne vous donne jamais les nombres en question. Ce qui rend quasiment impossible l'écriture de programmes d'après la doc, à moins d'acheter des livres qui listent ces données. Et qui rend infernale la lecture d'une image-mémoire [NdT : crash-dump, les données écrites sur le disque lorsqu'un programme plante pour permettre une étude des causes du plantage.] C'est pourquoi j'ai acheté tous les livres sur le Mac. Apple a également la très mauvaise habitude d'afficher des codes d'erreurs en nombres négatifs, comme "erreur -5561". Quoi? Devons-nous sortir nos calculatrices pour en faire le complément à deux, le convertir en entier positif puis en hexadécimal pour pouvoir retrouver cette valeur dans un listing? Je parie que dans les couloirs, chez Apple, se déroulent des concours pour savoir qui peux convertir le plus vite un nombre décimal négatif en nombre hexadécimal positif. Enfin, Messieurs, sovez adultes! C'est pour le plaisir de programmer macho? En tout cas, c'est stupide.

Enfin, je suis au regret de vous dire qu'à cause de cette prolifération de standards, dont certains se contredisent, il devient de plus en plus dur l'écrire des programmes Mac qui marchent sur toutes les machines. Autrefois, j'en maîtrisait la théorie. Plus maintenant, hélas. Avec les ajouts de MultiFinder, des couleurs en 32 bits, de Slot Manager, Time Manager, Power Manager, etc., la quantité de détails à mémoriser pour programmer sur Mac ferait frémir les concepteurs de la bombe atomique. Et tous les programmeurs sur ST qui tentent de passer sur Mac le savent.

Grâce aux standards, les programmes Mac sont mieux capables de communiquer entre eux que les programmes sur PC ou ST. Mais, parce que ces standards n'ont pas toujours été élaborés de manière saine (ils souffrent d'idéalisme aigu), ils ne donnent pas toujours un résultat convaincant, voire ne sont pas toujours utilisés. Les formats d'images, par exemple Greyview et PICT-2, sont très nombreux sur le Mac, ce qui ne serait pas le cas si les standards avaient été bien faits. Mais la culture d'entreprise d'Apple l'a interdit. Dans leur idéologie, l'idéalisme prime toujours sur le pratique.

Lorsqu'on examine l'histoire de cette firme, cette idéologie apparaît nettement. Vous souvenez-vous de l'Apple III? Du Lisa? Du Mac original sans SCSI? Hmmm... Ces gens sont des fanatiques sincères, et n'ont jamais compris pourquoi ces créatures stupides du monde extérieur ne se précipitaient pas pour



acheter leurs machines. Le succès de l'Apple II leur a permis de survivre malgré ces erreurs. Le Mac Plus a subi d'urgence une greffe de port SCSI et de 4 méga-octets de RAM, et quand la LaserWriter et PageMaker sont sortis, la firme a pu survivre malgré elle - et non à dessein. (Et déjà, l'Apple II n'avait dû son succès qu'au tableur VisiCalc.) Allez donc vous promener dans les allées du salon MacWorld (qui a été accaparé par les costards-cravates), et vous verrez des gens qui ont tous l'air hébété des membres de sectes. Des fanatiques. Effrayant.

Apple est une compagnie qui a réussi la prouesse de se mettre en péril en créant une machine qui a eu un grand succès! Comment? En sortant le Mac Classic à bas prix, qui s'est très bien vendu, et sur lequel la marge bénéficiaire d'Apple était insuffisante pour payer les frais généraux absolument délirants de la firme. Ce qui les a conduit à licencier beaucoup de gens. J'espère que tous appartenaient aux départements du Marketing et de l'Idéologie d'Entreprise. Quant aux revendeurs Mac, ils sont encore plus risibles et moins informés que leurs collègues du monde PC! Je sais, ça paraît impossible, mais c'est vrai. Tenez, avant la sortie du Mac portable, Sandy et moi avions réussi à intimider les employés d'une boutique en y apportant un Stacy et en le lançant en mode émulation Mac. Si jamais vous portez un beau costume, ils essaieront de vous vendre des machines trop chères et pas toujours trais fiables (rappelez-vous du fiasco des disques Quantum il y a quelques années), ainsi que des logiciels qui ne font pas ce qui est écrit sur la boîte (comme le scandale des "compatibles 32 bits", où Connectix leur a évité des procès : Apple faisait ce qui était interdit aux autres, à savoir écrire du code 24 bits qui plantait en mode 32 bits...)

Tuyau : si vous voulez un Mac vraiment puissant, achetez le Mac II d'occasion le moins cher que vous trouvez, et procédez à un échange de carte pour le changer en Mac II FX. Vous aurez une machine rapide et récente pour moins de 3000 dollars. Et n'achetez pas de Quadra : cette machine souffre de sérieux problèmes d'incompatibilité, et franchement, elle est très lente! Quand je lis dans un banc d'essai que le Quadra est à peine 30% plus rapide que le FX, cela me stupéfait. Les tests devraient montrer qu'il est beaucoup plus rapide, vu qu'il contient un 68040! Il y a là quelque chose de très anormal.

#### PROGRAMMATION ET UTILISATION

C'est pourquoi, lorsque j'entends des développeurs parler de quitter le ST pour aller vers le Mac ou le PC, je les invite à réfléchir.

Il faut un long apprentissage pour programmer sur PC, à cause de l'anarchie totale qui règne dans ce monde et vous oblige à rendre hommage au seigneur de chaque fief afin de rendre votre code compatible. Voyons, ce programme marche-t- il sur une carte vidéo Targa Super-VGA en mode non entrelacé avec un BIOS AMI 2. 1 sur un processeur 486 DX doté d'antémémoire? Il faudrait toute une vie pour tester toutes les combinaisons possibles. Je le sais, je déjeune régulièrement avec un développeur de jeux sur PC (il a écrit "Empire") qui endure cet enfer quotidiennement, et qui m'en parle.

Et sur Mac, il faut donc un an d'apprentissage. Encore faut-il serrer les dents et subir un endoctrinement politique ("Il est bon d'utiliser un relogeur de segment et de ne jamais savoir où votre code se situe, sachant qu'il peut être déplacé sous vos pieds"). Je n'ai jamais pu me faire violence à ce point, c'est pourquoi je reste à la périphérie de la programmation Mac, Dieu merci. En fait' je n'ai jamais écrit que de très courts programmes Mac, et seulement pour des tests : je ne veux pas être pourchassé de village en village par des gens qui me crient "impur! infidèle!".

Alors que sur ST, il ne faut qu'un mois pour découvrir l'environnement de programmation, surtout si vous avez de bons livres (je recommande ceux de Clay Walnum) et un bon assembleur ou compilateur. Vous pouvez facilement trouver des exemples de programmes qui tournent et les comprendre en quelques jours. Atari avait vraiment gaffé lors des débuts du ST, en ne documentant pas bien le GEM et en n'aidant pas les développeurs, mais des tiers ont subvenu à ce besoin, et désormais il est facile de se former au ST. Atari a en fait désespérément besoin d'un évangéliste l'entreprise qui porterait la bonne parole aux développeurs.

Et c'est pourquoi le monde du ST mérite qu'on y reste. On peut y programmer la machine en un temps raisonnable tout en s'amusant. De plus, Atari n'a pas commis les erreurs extrêmes de ses concurrents, à savoir l'anarchie totale ou la réglementation totale.

Si vous me permettez une analogie entre le marché de la micro et celui de l'automobile, voici les raisons pour lesquelles je ne veux pas développer sur PC ou Mac.

PC : Je ne veux pas aller chez un vendeur de voitures qui me propose des clones ayant tous les mêmes caractéristiques, mais utilisant chacun leur propre marque d'essence, à condition de trouver le bon vendeur. Si je change de marque l'essence, je dois changer le carburateur, la culasse et le pot

catalytique.

Mac : Je ne veux pas aller chez un vendeur de voitures qui me propose chaque roue à 500 dollars alors qu'elles ne coûtent que 75 dollars ailleurs. De plus, je devrais apprendre à conduire avec une trackball au lieu d'un volant, un embrayage au pied droit, un accélérateur au pied gauche, et un périscope en guise de pare-brise, sous prétexte que c'est la meilleure façon de conduire. Sans oublier de laisser Jamers en gage pour payer la voiture tant elle est chère. Enfin, pas question de servir de cobaye en attendant que le moteur soit débogué et de le changer trois fois avant la version définitive (ce qui s'est produit pour System 7).

Non, ce que je veux, c'est une voiture confortable, qui m'amène où je veux, et que je puisse facilement apprendre à bidouiller si je veux m'amuser. C'est le cas de mon ST. Il est doté de différents langages dans lesquels on peut se faire plaisir en programmant, ce qui est très important. J'apprécie le Basic et l'assembleur du ST, et une lointaine rumeur m'est parvenue, à la suite d'articles précédents, selon laquelle certains pourraient même s'amuser en C - un racontar si invraisemblable que je reste sceptique. Et tout compte fait, le ST est si supérieur au PC et au Mac de ce point de vue, même tant années après sa sortie, que j'en reste content. Si vous avez un ST, vous avez fait le bon choix.

## **BOULE DE CRISTAL**

Si je devais prédire l'évolution de la micro informatique, je dirais que le marché du PC va continuer à se fragmenter en centres d'intérêts spécifiques, jusqu'à ce que nous ayons des magazines "PC-AutoCAD", "PC-Lotus" et autres, dédiés à un petit nombre de logiciels (c'est ce qui est déjà en train de se produire). Les gens du Mac perdent déjà progressivement le contrôle des standards de leur machine : pour vérifier la configuration matérielle de la machine, par exemple, l'adresse \$B22 (où se trouvaient jadis les bits l'indication de configuration) a été remplacée par SysEnvirons, lui-même remplacé par Gestalt. Ce qui me rappelle les incohérences du gouvernement... On se souvient que l'Office de la santé et de la sécurité avait imposé des avertisseurs sonores de recul sur les engins de travaux public, pour se faire vertement tancer par l'Agence pour la protection de l'environnement qui leur reprochait d'être une source de pollution sonore! Apple a un urgent besoin de quelques responsables (attention, seulement quelques-uns) pour arrêter certains projets qui ne sont que travail inutile, ou qui n'aboutiront pas avant deux ou trois siècles.

Bien sûr, le ST pourrait être mieux. La plupart des reproches qu'on lui faisait ont d'ailleurs été pris en compte pour la conception du falcon. Mais surtout, le ST pourrait être bien pis! Faites le tour des boutiques d'informatique et voyez vous-mêmes. Les disputes qui portent sur les quelques défauts mineurs du ST sont si anodines, comparées aux hurlements provoqués par les vices majeurs des deux autres, que c'en est presque touchant. Comme si on rentrait à la maison alors que les enfants se disputent, après une grosse bagarre politique au travail.

En ce qui me concerne, j'ai cherché à améliorer le ST, par la carte accélératrice SST et par la compatibilité Mac, histoire de ne pas prendre ma retraite et rester à la maison à 30 ans. D'autres gens y pourvoient aussi tout en apprenant lentement comment survivre financièrement dans ce domaine. L'équipe de CodeHead, par exemple, fait du très bon travail. De plus, n'oubliez pas que si vous tenez absolument à éprouver les joies ineffables du PC et du Mac, vous pouvez le faire grâce aux émulateurs. Une raison de plus de garder votre Atari.

#### **CONCLUSION**

D'accord, il est difficile de garder la foi sans entendre régulièrement la bonne parole, surtout lorsqu'on vous injecte régulièrement une dose de désespoir via une certaine presse. Mais nous, possesseurs de ST, avons une bonne machine. Nous sommes une minorité relativement petite comparé aux marchés des Mac et des PC, mais il y a des aspects positifs à cela : si vous écrivez guelque chose de bon, vous

serez reconnu, au lieu d'être le 36ème inventeur de votre trouvaille! La communauté du ST a le confort d'une petite ville, et j'aime ça. Quand je vais au Comdex ou à MacWorld, je me perds dans un océan de costards-cravates qui tentent de m'extorquer 20% de marge de distribution pour très peu de travail.

La gamme Atari offre déjà, d'origine ou grâce à des cartes additionnelles, des machines extrêmement rapides, compétitives avec tout ce qu'offrent le Mac et le PC. Vous pouvez trouver des cartes vidéo qui ont une bonne chance de marcher avec vos applications sans requérir un pilote spécial. Il y a le Falcon, aux possibilités graphiques et sonores bien supérieures (j'en ai un) et dont le système d'exploitation supporte le multitâche. Et vous pouvez toujours y ajouter un émulateur Mac ou PC si vous le voulez.

Qu'est-ce que vous perdez en n'achetant pas un PC? La chance unique de devoir résoudre un conflit de numéros d'interruptions? La joie de découvrir qu'après avoir configuré ledit PC pour avoir presque la même souplesse qu'un ST, vous avez quasiment épuisé ses capacités l'extensions? La barrière des 640 Ko? La propagande? Franchement, je ne crois pas que cela puisse vous manquer, et à GBS, nous avons des ST, des PC, des Mac, des machines Unix... en autres. Je parle donc en connaissance de cause.

Enfin, en ces temps d'économie morose, le ST est particulièrement abordable. Mais il resterait le meilleur choix même si vous pouviez choisir n'importe quoi.

Traduction et adaptation : Password 90

Titre original: Rationality

## MMU et magie

Le Falcon débarque bientôt. Son processeur, le 68030, offre une liste de caractéristiques impressionnantes, parmi lesquelles on voit fréquemment apparaître un sigle énigmatique, "MMU". Qu'est-ce donc? Dave se fait ce mois-ci un plaisir de nous l'expliquer, et nous montre quels sont les avantages d'avoir dans une machine ces trois lettres magiques. Et en profite pour nous rappeler que l'informatique abonde en "tours de magie" a priori impossibles ou incompréhensibles. Car selon l'adage de l'ingénieur, une technologie suffisamment évoluée est indistinguable de la magie...

#### Introduction à la magie

J'ai toujours aimé la magie, et j'y ai toujours cru. Dans ce domaine, je ne suis jamais passé par ce curieux stade que l'on nomme "devenir adulte". Et cela m'a bien servi, comme à beaucoup de bidouilleurs. Il est parfois important de croire que quelque chose peut arriver, en dépit de ce que tout le monde dit. C'est ça, la magie : par exemple, vous savez pertinemment qu'il n'y a pas de lapin dans le chapeau, et pourtant, il en sort un.

Je me sens un peu dans la peau d'un prestidigitateur lorsque je fais une démonstration du Spectre GCR dans un salon, devant des gens qui ne l'ont jamais vu. Surtout lorsque l'assistance comporte des fanatiques du Mac. Je lance le programme, passe par les menus de configuration, règle les paramètres, et appuie sur Return pour tout lancer. Et pendant quelques secondes, rien ne se produit. Rien ne transparaît de l'activité furieuse de la machine. L'écran resté figé, et le programme semble s'être planté. Les secondes s'écoulent pesamment, et le suspense commence à devenir angoissant pour moi. S'est-il planté ou non? Après tout, c'est moi qui ai vu le Spectre se planter le plus souvent, et souvent à ce stade, parfois même durant une démo...

Et soudain, hop, le lapin sort du chapeau! Un petit Mac souriant apparaît sur l'écran, accompagné du message de bienvenue. Quelques secondes plus tard surgit le bureau du Mac. Je camoufle en sourire un soupir de soulagement et j'observe l'assistance. J'adore l'expression des visages à cet instant. C'est celle de gens qui ont vu un tour de magie. Ils savent que c'est impossible; pourtant ça a marché, et ils n'ont pas la moindre idée du trucage employé. Une bonne moitié de l'assistance cherche discrètement des câbles sous la table, qui relieraient le moniteur du ST à un Mac caché. C'est évidemment ce qu'a fait ce vieux dur à cuire de Jerry Pournelle la première fois que je lui ai montré le Spectre. (Il est journaliste dans Byte, écrivain de science-fiction, et ancien officier d'artillerie. On ne la lui fait pas.)

L'émulation Mac a fait ses débuts en 1986. Au fil des années, tandis que le programme évoluait, il est devenu de plus en plus difficile de le planter. Le point critique fut le gestionnaire d'accès à l'adresse zéro : beaucoup de programmes Mac écrivent par erreur à cette adresse, où se trouve, sur le ST, de la ROM, ce qui plante aussitôt le 68000. Les manuels de Motorola disent que le 68000 ne peut pas rattraper une erreur de ce type, et que c'est même pour ça qu'il y a le 68010. C'est là qu'intervient la plus puissante magie du Spectre après une telle erreur, mon gestionnaire d'accès à l'adresse zéro ramasse le 68000 qui s'était étalé par terre, l'époussette, remonte son ressort et le remet sur ses rails. Puisque les concepteurs du 68000 affirment que c'est impossible, il m'a fallu désapprendre ce que disait le manuel et essayer malgré tout. C'est ça, la magie : c'est faire ce que le manuel affirme être impossible.

C'est pourquoi nous avons surnommé Zeus notre gestionnaire, en l'honneur du plus puissant des dieux grecs. Zeus fait toute la différence entre un semi-émulateur et un produit stable. Comment marche-t-il, demandez-vous? Oh, allons, allons, rien n'est plus ennuyeux que voir un magicien vous montrer comment marche un tour... Mais Si vous êtes vraiment bidouilleur dans l'âme, vous noterez l'intéressante similitude entre le registre d'instruction IR, sauvé dans la pile lors du plantage, et l'instruction qui a planté le 68000. C'est le point de départ.

Je fais tellement confiance à mon petit tour de magie que je m'amuse même à demander aux utilisateurs de Mac de faire tourner sous Spectre des applications que je n'ai jamais vues. Certes, je suis toujours un peu anxieux, mais ces nouveaux programmes marchent dans 95% des cas, et le reste du temps, je peux mettre en évidence dans le code de l'application les problèmes qui ont provoqué le plantage : "Eh, ce programme accède directement au circuit SCSI! Il cherche les ennuis, ou quoi? Apple spécifie que c'est interdit, et ça plante sur un portable ou un Mac llfx."

## Magie peu discrète

J'adore faire à des ordinateurs des choses qui semblent être magiques. Tenez, au lycée, je n'ai survécu que grâce à des études indépendantes qui m'ont permis de rattraper les mauvaises notes de mes cours de statistiques (entre autres); ces études gravitaient autour de la Magie des Grands Systèmes. Nous avions un grand système Cyber de Control Data (CDC), qui m'a permis d'apprendre des choses très intéressantes comme le langage assembleur Cyber, les appels systèmes, comment interrompre une compilation Fortran afin de laisser dans un fichier une copie du système entier (tiens, tiens, il y en a, des choses intéressantes, là-dedans!), et même, summum de la bidouille, comment parler directement aux entrailles de la machine.

Je me rappelle avoir écrit un programme qui permettait à plusieurs utilisateurs jusqu'à 32) de discuter simultanément par terminaux interposés. Lorsque je voulus le tester, je le lançai sur quatre des terminaux de notre centre de calcul, où travaillaient beaucoup d'étudiants. Et je commençai à passer d'un terminal à l'autre, tapant des messages de test. Tout à la joie de ma bidouille, je mis un moment à remarquer le silence surnaturel qui s'était établi dans la salle. La bruyante imprimante à ruban de la salle voisine s'était tue, et les matricielles DecWriter avaient cessé leur crin-crin suraigu. Je levai le nez de mes terminaux et vis que tout le monde m'observait, car mes terminaux étaient les seuls encore actifs. Je parvins à garder l'air innocent suffisamment de temps pour arrêter mon programme. Aussitôt, la salle ressuscita, les imprimantes recommencèrent leurs bruits. Je me dépêchai de m'enfuir avant que ne surgisse quelque garde du Service de Sécurité Informatique.

Depuis, j'ai realisé que le répartiteur de travaux du Cyber "préférait" les petits programmes interactifs comme le mien plutôt que les gros travaux faisant de nombreux accès disque. Les manuels de CDC affirment que ce comportement est impossible sous TELEX, le système d'exploitation en temps partagé du Cyber. On me permettra d'en douter.

#### Magie intégrée

A présent, mes accessoires de magie sont les machines Atari, qui contiennent une magie potentielle encore inexploitée. Cette magie a pour nom MMU (Memory Management Unit, unité de gestion de mémoire). Je m'en suis servi pour accomplir des tours spectaculaires, dont chacun "sait" qu'ils sont impossibles.

Et qu'est-ce qu'une MMU, demandezvous? C'était jadis un circuit intégré à grande échelle séparé dans le catalogue Motorola, le 68851, qui fonctionnait avec le 68020. Dans le 68030, Motorola a intégré l'équivalent d'un 68851, si bien que le TT et le Falcon en sont dotés d'office. Une MMU est un circuit qui sépare les adresses mémoires en domaines "logiques" et "physiques", et vous permet de lier les uns aux autres. Une traduction, en quelque sorte.

A l'intérieur de votre ST, vous avez jusqu'à 4 mégaoctets de mémoire (un mégaoctet = 1 million de caractères, "Mo" en abrégé). Chaque octet est repéré par son adresse, de 1 à 4·000·000. Le numéro de chaque octet est nommé "adresse physique". Les programmes que vous exécutez sont chargés quelque part dans ces adresses, ainsi que les données qu'ils utilisent. Même ce que vous voyez apparaître sur votre écran vidéo est stocké en mémoire, sous forme d'une image occupant 32·000 octets.

Par contre, les programmes, eux, n'utilisent que des "adresses logiques". Il s'agit en fait de numéroter différemment les adresses. Mais sur le ST, les adresses physiques et logiques sont identiques. Si un programme accède à l'adresse logique n°36, il obtient l'adresse physique 36. C'est parce que le 68000 du ST n'a pas de MMU.

Maintenant, voyons ce que peut faire le 68030 d'un TT ou d'un Falcon. La MMU de celui-ci permet de complètement séparer les adressages physique et logique. Grâce à elle, le système d'exploitation (le TOS) peut faire croire aux programmes qu'ils sont seuls dans la machine et sont libres d'utiliser toute la mémoire, alors qu'en fait ce n'est pas le cas et que plusieurs programmes coexistent dans le système. L'astuce consiste à scinder la mémoire en "partitions" grâce à la MMU. Supposons que nous ayons 4 Mo de RAM et 4 programmes à utiliser simultanément. Nous attribuons la partition 1 (adresses physiques de 1 à 1·000·000) au programme 1, la partition 2 (adresses physiques de 1·000·001 à 2·000·000) au programme 2, et ainsi de suite pour les quatre partitions. Quand le programme i veut accéder une adresse (logique, car les programmes ne manipulent que des adresses logiques), mettons l'adresse 10, la MMU convertit cette adresse logique 10 en "dixième adresse de la partition 1". Et comme la partition 1 commence a l'adresse physique 1, il s'agit donc de l'adresse physique 10. Quand le programme 2 veut accéder à l'adresse logique 10, la MMU "sait", grâce à des tables de correspondance internes, que la partition 2 commence à l'adresse physique 1·000·001, et traduit donc cette adresse logique en l'adresse physique 1·000·010.

Le programme 2 accède donc à son octet, qui, pour lui, porte toujours l'adresse 10. Pour le programme, cette traduction au vol est totalement invisible. Et de plus, elle est sans importance en effet, le programme tourne normalement, car tous les emplacements mémoires manipulés par le programme ont été déplacés, ou "mappés" [NdT: du vieux français "mappe", ou carte, comme dans "mappemonde". C'est donc bien français, et en plus, c'est la traduction officielle.]

Supposons maintenant qu'à cause d'un bogue, le programme 1 tente d'accéder à une autre partition que la sienne, par exemple en accédant à l'adresse logique 1·500·000. La MMU détecte grâce à ses tables que l'adresse logique demandée appartient à la partition 2, à laquelle le programme 1 n'a pas le droit d'accéder. Elle signale donc une erreur de bus et arrête le programme.

A présent, soyons très malins. Installons dans notre système une horloge qui bat la seconde, et faisons en sorte qu'au premier battement, le 68030 fasse tourner le programme se trouvant dans la partition 1, puis celui de la partition 2 au second battement, et ainsi de suite tour à tour pour les quatre partitions, en recommençant avec la partition 1 au cinquième battement, en boucle. L'horloge peut interrompre et reprendre l'exécution d'un programme, de façon à pouvoir donner la main à tous les programmes tour à tour. Les programmes arrêtés reprennent bien sûr là où ils en étaient restés. Ceci s'appelle le multitâche préemptif. Les programmes ne peuvent en effet choisir les moments où le système d'exploitation les arrête et lès reprend. lis s'exécutent tranquillement, quand tout à coup, ils sont préemptés, et la main est donnée à un autre programme.

Dans le monde réel, cette horloge est celle du répartiteur de tâches, et elle est cadencée beaucoup plus vite que la seconde. Car s'il y a quatre tâche, il s'écoule pour une tâche donnée 3 secondes durant lesquelles le processeur est occupé à faire tourner les autres tâches, ce qui est beaucoup trop lent (même pour les humains, qui sont dotés d'un processeur plutôt lent). D'ordinaire, il y a environ 100 changements de tâche par seconde.

Avec une vitesse de changement de tâche suffisamment grande, tous les programmes semblent s'exécuter simultanément, bien que chacun n'ait en fait droit qu'à une tranche de temps de processeur un certain nombre de fois par seconde. Les changements de tâche doivent être assez rapides pour que le ou les utilisateurs ne s'aperçoivent de rien, sauf dans des cas particuliers. Les humains sont suffisamment lents pour qu'un ralentissement de l'ordinateur passe généralement inaperçu. De plus, les ordinateurs passent le plus clair de leur temps à attendre que l'utilisateur fasse quelque chose, comme frapper une touche du clavier ou bouger la souris. S'il n'y a rien à faire, comme souvent, la machine se tourne les pouces. Ce temps passé à attendre peut être utilisé pour donner la main à une autre tâche, s'il y en a une.

Il existe une autre forme de multitâche, dit coopératif, où les programmes signalent au système les endroits où ils peuvent être interrompus. Dans le ST, ces endroits sont ceux où le programme attend un événement, comme un clic de souris ou un appui sur une touche. Lorsque le programme se met en attente, le système donne la main à la partition suivante. Encore faut-il que le programmeur l'ait prévu.

Si un programme n'a pas été écrit dans l'esprit du multitâche, il peut très bien accaparer toute la machine en n'offrant jamais au système d'exploitation l'occasion de passer à une autre tâche. (C'est en fait ce qui s'était produit dans mon programme de communications entre terminaux sur le CDC au lycée).

Mais attention, une MMU est indispensable pour tout cela. C'est elle qui permet de faire croire aux programmes qu'ils sont seuls dans la machine et les empêche de marcher sur les plates-bandes de leurs collègues, par bogue ou par malice. Si vous faites du multitâche sans l'aide des circuits de la MMU, ces protections vous font cruellement défaut, et vous obtenez moult plantages. C'est d'ailleurs ce qui se produit sur l'Amiga, qui n'a qu'un 68000 en version de base et pas de MMU. Les utilisateurs de cette machine ont vite appris quels étaient les programmes qui ne suivaient pas les règles du multitâche coopératif, et qui donnaient l'équivalent des bombes sur le ST: les "Guru Meditation". (Quoi?... Non, juré, je ne travaillais pas chez Commodore à l'époque, et ce n'est pas moi qui suis le petit plaisantin auteur de cette trouvaille.) Seuls les programmes bien élevés peuvent tourner en multitâche sur l'Amiga. Les programmes mal écrits, dont les utilisateurs doivent tenir la liste, plantent le système.

## Magie virtuelle

Ça va, vous suivez? Accrochez-vous, ça va devenir subtil. Supposons que l'ordinateur comporte 4 Mo de RAM, et un disque dur. Supposons que nous exécutions un programme, et que lorsque celui-ci demande la taille mémoire disponible, nous lui répondions 12 Mo. De plus, nous ajoutons au système d'exploitation une petite routine spéciale pour l'aider. Enfin, nous créons sur notre disque dur un fichier spécial de 12 Mo de long, nommé image mémoire.

Tant que notre programme reste à l'intérieur des 4 mégaoctets réels, aucun problème. Mais supposons que le programme demande accès à l'adresse logique numéro 7·000·000? Aha! La MMU le détecte, invoque notre petite routine, et commence à pratiquer de la magie de haut vol. Elle cherche un endroit dans la RAM qui n'a pas été accédé depuis un bon moment, mettons l'adresse physique 2·000·000. Pour des raisons de sorcellerie appliquée trop ennuyeuses à exposer ici, la mémoire est divisée en morceaux appelés "pages", faisant entre 512 et 4.096 octets de long. Eh bien, le contenu de la page située à cette adresse physique 2·000·000 est sauvé sur disque. Ensuite, la MMU mappe l'adresse logique 7·000·000 à cette adresse physique. Désormais, tout accès à cette adresse logique se référera en fait à cette adresse physique. C'est ce qu'on appelle une "permutation de page". Et la MMU rend la main au programme, qui ne s'est aperçu de rien. Naturellement, Si, plus tard, le programme veut accéder à l'adresse logique 2·000·000, qui a été sauvée sur disque, la page sauvée est rechargée depuis le disque et est mise à un autre emplacement mémoire inusité. La MMU met à jour ses tables de correspondance, et l'exécution continue. Il n'y a aucune relation entre les adresses logiques et physiques à l'exception des tables de la MMU.

Naturellement, cela marche pour n'importe quelle page. Il suffit d'avoir suffisamment de place dans l'image mémoire sur le disque. Conclusion? Nous avons non plus 4, mais 12 Mo de mémoire. Certes, les accès au disque très lent comparé à la RAM - vont ralentir le système. Mais la plupart des programmes ont tendance à boucler dur des sections de code relativement courtes, et à ne pas demander accès à trop de pages à la fois. Ce mécanisme, dit de "mémoire virtuelle", marche donc assez bien. Il a été inventé par IBM dans les années 60, et a été popularisé par UNIX depuis plus de 20 ans!

Mettre la main sur un 68030, c'est facile : il suffit d'avoir un TT, un Falcon, ou d'en mettre un dans votre ST par une des cartes additionnelles qu'offrent différentes firmes (y compris la carte SST de votre serviteur). Cela permet de goûter aux joies de la mémoire virtuelle. Je l'ai vue tourner sous une forme spécialisée en Allemagne dès 1991 : un bidouilleur avait doté le programme de mise en page Calamus d'une mémoire virtuelle. Et Si vous voulez devenir un héros dans la communauté TT, munissez-vous d'un livre sur les systèmes d'exploitation et la mémoire virtuelle, et implémentez les algorithmes qui y sont expliqués. Motorola vous fournira même la plupart des routines nécessaires Si vous les leur demandez.

## Magie commutée

Continuons, nous commençons seulement à nous amuser! Supposons maintenant que vous mettiez en place trois partitions de mémoire. La première contient un programme ST. Dans la deuxième tourne un programme PC sous émulateur (matériel ou logiciel), et dans la troisième se trouve un programme Mac tournant sous Spectre. Nous avons donc un système comprenant trois tâches.

La MMU permet de passer instantanément de l'une à l'autre (ce qu'on nomme la commutation de tâches). Chacune de ces tâches "croit" qu'elle a la machine pour elle seule, et s'exécute normalement. C'est ainsi que grâce à un programme de commutation de tâches, vous pouvez passer en un éclair d'un programme à l'autre - et dans ce cas, d'un environnement à l'autre. La MMU s arrange pour mettre en mémoire basse la partition demandée, et le programme de commutation charge les registres avec les valeurs voulues, tout en gérant les aspects temps réel (comme le déplacement de la souris) durant la commutation. Ce n'est pas aussi dur que ça en a l'air. Le programme Revolver le fait déjà, mais sans l'aide d'une MMU. C'est pourquoi il est à la merci de tout programme mal écrit, qui plante le système entier. Ajoutez à cela une MMU, et le système devient très stable. Si c'est correctement fait, vous devez pouvoir planter un programme dans une partition tandis que les autres continuent à tourner.

Il n'y a aucune raison qui nous empêche de faire tourner plusieurs tâches à la fois dans autant de fenêtres. La commutation de tâche peut aisément se faire à une cadence assez rapide, par exemple 50 fois par seconde. L'idéal est alors de disposer d'un grand écran genre Moniterm, ou bien d'un grand écran virtuel (qui utilise un moniteur ordinaire qui est une fenêtre sur l'écran virtuel plus grand, et un programme grâce auquel on fait dérouler l'écran, par exemple vers la droite quand la souris atteint le bord droit du moniteur). Et, de fait, c'est exactement ce qui est possible grâce à MultiTOS, le TOS multitâche basé sur le système Mint d'Eric Smith. Et ce, grâce à la MMU du 68030.

## Magie et gadgets

La 68030 SST, la carte accélératrice pour ST que nous avons développée ici, à Gadgets By Small, nous a permis de faire d'intéressantes acrobaties à sa MMU. Par exemple, le TOS, le fidèle système d'exploitation du ST, est stocké dans deux EPROM. Or, ce type de mémoire est déjà plutôt lent. En plus, chaque lecture ne fournit au 68030 que 16 bits, au lieu des 32 qu'il réclame. La solution : copier les EPROM dans la RAM rapide de la carte, et utiliser la MMU pour remapper cette portion de mémoire à l'adresse physique qu'occupent normalement les EPROM. Cela ne prend après tout que 256 Ko sur les 8 Mo de RAM rapide dont dispose la SST. Après quoi, le TOS tourne en 32 bits, en mode rafale, soit le mode le plus rapide d'une mémoire qui ne lambine déjà pas. Et, comme par magie, toutes les détestables lenteurs du GEM sont oubliées.

Mieux encore, faisons une croix sur les 64 Ko les plus bas dans la RAM du ST (qui est de toute façon ralentie à 50% par les accès vidéo qui sont toujours prioritaires), et remappons-les en RAM rapide. La machine accède désormais sans ralentissement à cette portion de mémoire, qui contient les tables et variables du système, et le TOS bénéficie donc d'une nouvelle accélération.

Je vois d'ici certains bidouilleurs sourire malicieusement à la pensée que le système d'exploitation se trouve désormais en RAM, et non plus dans une ROM inaltérable. Si vous le désirez, vous pouvez changer tout ce que vous voulez en retouchant des octets dans la fastRAM. Il y a quelque chose qui vous a toujours manqué dans le TOS? Vous voudriez modifier l'aspect du bureau? (Ricanement démoniaque) Allez-y, vous avez le champ libre! Réécrivez par dessus le TOS, au lieu de devoir vous résoudre à des rafistolages douteux, comme l'interception des vecteurs du XBIOS. Vous en avez assez du bogue de la routine Malloc? Allez-y, corrigez-le.

#### Magie multiple

Si vous vous sentez d'humeur taquine, vous pouvez aussi mettre en place deux ST logiques côte à côte dans la mémoire du système, avec deux images différentes du TOS, deux écrans différents, etc. Et, bien sûr, un commutateur de tâches entre les deux. Vous pouvez avoir le deuxième ST que vous avez toujours voulu, par exemple pour héberger un serveur télématique sans interruption, tandis que le second ST logique est toujours disponible. Si l'un se plante, le second tourne toujours. (On pourrait même concevoir de faire tourner deux systèmes redondants pour mieux rattraper les erreurs. C'est ce que font les systèmes militaires et spatiaux à tolérance de panne.)

De même, il est possible de mettre dans ce ST deux émulateurs Mac logiques, ou deux PC, ou même, pourquoi pas, deux CP/M (d'ailleurs, j'ai conservé beaucoup de programmes CP/M, et avec les 33 MHz de la SST, l'émulateur est très rapide).

## Voulez-vous devenir magicien?

Si vous êtes en train de baver sur vos chaussures en lisant ceci, c'est qu'il est temps de vous procurer le manuel du 68030 de Motorola (pour sa partie spécifique sur la MMU de ce processeur), ainsi que celui de la PMMU 68851, qui vous dira tout ce que vous avez toujours voulu savoir sur les MMU. Et tenez-vous bien, Motorola emploie des gens dont le travail consiste à veiller à ce que ses manuels parviennent aux gens qu'ils intéressent. Appelez votre agence commerciale Motorola locale, et commandez ces manuels. [NdT: Motorola SemiConducteurs, 2 rue Auguste-Comte, 92173 Vanves, (1)40 95 59 00.]

Et Si un jour, dans un salon, quelqu'un m'appelle et me montre un beau tour de magie monté à l'aide d'une MMU, je saurai que cette article a été utile.

## Dave est préempté

Holà, il est temps de rendre la main aux autres tâches de ce magazine. Je dois libérer les autres pages. Je vous retrouve le mois prochain, lors de ma prochaine tranche de temps!

Traduction et adaptation Password 90 Titre original : MMU & Magic, Inc.

# La face cachée de la télématique

Aux Etats-Unis comme en France, la télématique a pris une extension fantastique. Après avoir traîné les pieds dans notre pays, le mouvement a pris de l'ampleur grâce au Minitel, dont le modem s'utilise facilement via un simple câble. Les multiples messageries et services télématiques sont à présent indispensables à bien des bidouilleurs. Mais hélas, tout n'est pas rose en ce domaine (sans même parler des messageries "pour adultes"). En marge des serveurs honnêtes, amoureusement bichonnés par des opérateurs scrupuleux, il existe une catégorie de serveurs nuisibles, voire dangereux. Suivons Dave dans sa visite de cet univers caché...

## Le problème

Qu'est-ce qu'un serveur télématique? C'est un ordinateur, en général doté d'un disque dur, relié au réseau téléphonique via un modem. On distingue les "grands" serveurs, dits multivoies, qui acceptent plusieurs appels à la fois, et desservent donc plusieurs utilisateurs. Aux USA, les plus grands, comme CompuServe ou GEnie, sont accessibles à travers tout le pays par des numéros de téléphone locaux répartis dans les plus grandes villes. Ceux-là tournent sur de gros systèmes. Mais il existe aussi de très nombreux serveurs, la plupart du temps mono-voie (un seul appel à la fois), qui tournent sur des machines domestiques, comme les ST. Contrairement aux grands serveurs, qui emploient des systèmes sophistiqués de concentrateurs d'appels pour être accessible dans tout le pays [NdT: ou bien le réseau Transpac et les numéros 3614 ou 3615 en France], ces petits serveurs utilisent le réseau téléphonique ordinaire, dit "Réseau Téléphonique Commuté", d'où le nom de "serveur RTC", ou RTC tout court.

Tout possesseur de modem, ou de Minitel en France, peut appeler un RTC. Le modem du RTC détecte l'appel, prend la ligne et répond à l'ordinateur appelant. Les deux principaux usages des RTC sont:

- ① les discussions. Il y a en général une rubrique Infos, une section "Rumeurs", ainsi que des rubriques concernant des sujets spécifiques, comme le C, le GEM ou l'assembleur.
- ① les transferts de fichiers. Le RTC a une bibliothèque de fichiers dans laquelle l'utilisateur peut choisir des fichiers à télécharger. Il peut aussi "remonter" des fichiers depuis sa machine vers le RTC. Ces fichiers sont censés appartenir au domaine public, c'est-à-dire que leurs auteurs ont expressément spécifié qu'ils pouvaient être copiés librement. Par exemple, des listes d'astuces et de solutions de jeux, ou bien des programmes, des accessoires, des formateurs de disquette, etc.

Si vous n'avez pas de modem ou de Minitel, vous perdez quelque chose. Il y a tout un univers à votre portée, pour le prix d'un appel. Les grands serveurs ont des opérateurs chargés de les maintenir, les "sysops" [NdT : *System Operator*], et ont des milliers de fichiers à télécharger, des rubriques nombreuses et très fréquentées, et des services attrayants. Les petits serveurs sont davantage tributaires du temps libre de leur propriétaire, qui n'en tire habituellement aucun profit.

La plupart des serveurs multivoies sont honnêtes, surtout les grands, très connus donc très observés. De même, la majorité des RTC sont parfaitement honnêtes, et offrent le meilleur de ce formidable outil qu'est la télématique. Mais comme tout outil, celui-ci peut être utilisé à de mauvaises fins. Comme la piraterie.

## Les RTC pirates

Il y a en fait plusieurs formes de piraterie. La première survient lorsque vous achetez un programme, mettons un jeu, et en donnez une copie à un ami. Ces cas de piraterie, il s'en produit sans arrêt. Il y a sur le ST des programmes qui facilitent la copie, et les gens qui vendent ces programmes ont gagné beaucoup d'argent. (Les gens dont les programmes sont copiés ont généralement fait faillite, mais ceci est une autre histoire.) Mais ce n'est pas la forme la plus grave. Certes, c'est illégal, mais ce n'est pas le pire.

La seconde forme de piraterie consiste à télécharger vers un RTC les fichiers de la disquette que vous venez d'acheter. Après quoi, tout le monde peut "descendre" les fichiers du RTC vers sa propre machine, en un seul appel. Soudainement, ce n'est plus une seule copie que vous faites en donnant une disquette à un ami, mais des centaines, que vous donnez à tous les habitués du serveur. Certaines de ces copies seront remontées sur d'autres RTC, d'où elles seront descendues par encore davantage d'utilisateurs. Le processus est similaire à une réaction en chaîne nucléaire. Et c'est pourquoi cette forme de piraterie est bien la pire, vu les dommages qu'elle cause aux éditeurs de logiciels. Il ne s'agit plus de quelques ventes perdues, mais de plusieurs milliers!

L'honnêteté ne se divise pas. Pour qu'un serveur RTC soit pirate, il suffit qu'il propose en téléchargement un programme hors du domaine public, soumis à droit d'auteur. Et il y a des milliers de RTC pirates aux Etats-Unis.

De plus, ces RTC sont extrêmement bien interconnectés. Leurs utilisateurs sont pour la plupart membres d'une même clique et fréquentent plusieurs serveurs. Quand ils découvrent un nouveau fichier sur un RTC, ils se dépêchent de le remonter à plusieurs autres. Ces pirates appellent souvent en interurbain [NdT: du moins aux USA, où les tarifs des communications interurbaines sont beaucoup plus raisonnables qu'en France.] Il suffit souvent d'à peine 24 heures pour qu'un logiciel soit disséminé d'un bout à l'autre du pays. En outre, il y a de plus en plus d'échanges avec l'Europe. Il est devenu fréquent de voir sur des RTC américains des logiciels venus du vieux continent.

#### Le déplombage

Les développeurs sont souvent conscients de ce problème. C'est pourquoi ils "plombent" leurs logiciels, ce qui consiste à y ajouter une protection contre les copies. En général, un programme protégé vérifie qu'il s'exécute depuis le disque original avant de s'exécuter. Le disque original est bien sur pourvu d'une marque impossible à dupliquer. Si quelqu'un remonte sur un RTC un programme ainsi protégé, rien ne sert de le descendre le disque sur lequel il aura été copié ne possède pas cette marque, et le programme refusera de s'exécuter.

Une autre méthode de protection consiste à changer le format de la disquette de sorte qu'elle ne contienne plus de fichier clairement identifiable et facile à télécharger. Le pirate examine alors la protection et l'inactive, ce qui s'appelle le déplombage. Le programme déplombé peut alors être remonté vers un RTC.

Le déplombeur est en général fier d'avoir pu déjouer la protection, et met son pseudonyme dans un écran du programme, de façon à ce que ses petits copains sachent qui a fait le coup. Ces "pseudos" sont similaires aux indicatifs des radio-amateurs CB, en plus fantaisistes, du genre "Captain Wizard" ou "Dr Typo". Bien entendu, aucun pirate n'est assez stupide pour signer ses déplombages de son véritable nom.

Autour des RTC pirates gravite toute une structure sociale de pirates se connaissant mutuellement par leur pseudo plutôt que par leur nom, et qui échangent des logiciels. Pour vous faire prendre conscience de l'ampleur les dégâts, permettez-moi de vous faire visiter brièvement un RTC pirates parmi des milliers d'autres. Il s'agit de "Coastline", un RTC en apparence tout à fait honnête, mais qui fonctionne à deux niveaux. Au premier niveau, il semble n'être qu'un RTC licite, offrant une messagerie et des fichiers du domaine public aux utilisateurs d'Atari ST. Mais lorsque vous êtes en bons termes avec le sysop, celui-ci vous laisse accéder à la section pirate, où l'on trouve une liste impressionnante de logiciels piratés, y compris des jeux et utilitaires récents. Ou plutôt, on y trouvait des logiciels récents : Coastline a été fermé, son équipement confisqué, et un procès est en route contre son sysop, un

adolescent qui doit à présent faire face à une accusation d'infraction à la législation des droits d'auteur, pour un montant de plusieurs milliers de dollars. Et comme Coastline était appelable depuis un autre état, cela devient un délit fédéral, dont s'occupe donc le FBI. Les parents de ce jeune homme affirment qu'ils ne savaient pas ce que leur gamin fabriquait des heures durant avec son modem, et sont probablement sincères. Ce qui n'empêche pas ce gars de se retrouver doté d'un casier judiciaire, et ses parents d'être pénalement responsable pour un montant énorme, tout ça pour avoir joué au pirate!

## Téléchargement

Les RTC pirates ont le plus souvent des règles en matière de téléchargement. Si vous y remontez un programme, vous pouvez en descendre trois. Cela oblige l'utilisateur à y remonter de nouveaux logiciels fréquemment pour continuer à descendre des programmes. Et de plus, cela encourage la piraterie, puisque les sysops rejettent les programmes qu'ils ont déjà en bibliothèque de téléchargement et n'acceptent que les nouveautés.

Les programmes obtenus de cette façon sont nommés "progz" [NdT : "warez" en anglais], la faute (z au lieu de s) étant intentionnelle. De même, les fichiers d'information permettant d'utiliser illicitement certains numéros de téléphone se nomment "phichier" [NdT : "philes" en anglais], avec un "ph" comme dans "téléphone". J'y reviendrai.

## Que fabrique Dave dans ce monde douteux?

Vous vous demandez sans doute ce que je fais sur ces serveurs douteux? Eh bien, l'ancêtre de Spectre, mon émulateur Mac, s'appelait Magic Sac, et apparaissait fréquemment sous forme déplombée sur des RTC pirates. J'en ai vu au moins dix versions différentes, dont trois d'Allemagne, deux d'Angleterre, et une de France (comprenant même les messages traduits!). J'essayais de savoir quand la version courante du Magic Sac avait été déplombée, et je savais alors qu'il était vital de sortir une mise à jour.

Notez au passage que je me consolais en me disant qu'au moins, j'avais écrit quelque chose digne d'être copié. Mon mentor, Wayne Smith, affirme que c'est le critère pour connaître la valeur de vos créations. Cela reste une piètre compensation quand on voit son programme piraté et qu'on se souvient de tout le travail qu'on y a investi.

J'ai infiltré un certain nombre de RTC, sous différents noms, juste pour les surveiller. Je ne me fais pas d'illusion, je n'ai vu que le sommet de l'iceberg. Il y a des milliers de RTC pirates dans le monde du ST. Mais si ça peut vous rassurer, le nombre de serveurs pirates pour PC est effarant, et j'en connais un qui a une liste de progz de 15 pages de long, à raison d'un par ligne.

Après avoir plongé dans ces eaux troubles, je me suis dit qu'il y avait là matière à un article. C'est pourquoi je vous convie à me suivre dans un voyage que j'ai entrepris parmi les RTC pirates, où j'ai écouté, observé, et appris. Il s'y passe des choses ahurissantes. La piraterie n'est qu'un début, vous allez voir. Tout d'abord, laissez-moi vous conter comment je me suis brutalement retrouvé confronté à ce genre de pirates.

#### Dave remonte la filière

Certains développeurs sont fatalistes quant aux RTC pirates. Ils pensent qu'ils ne peuvent rien y faire, donc ils ne s'en soucient pas "Quand tu ne peux rien faire dans une situation donnée, fais un somme",

fait dire Howard à son héros Conan. D'autres, par contre, tentent de faire fermer ces RTC. Ils contactent le FBI et engagent des poursuites. A ce sujet, signalons qu'il existent des sysops pirates qui se sentent en sécurité parce qu'on ne connaît d'eux que le numéro de téléphone de leur RTC. N'importe quel détective privé digne de ce nom, entre autres, peut facilement obtenir l'adresse correspondante.

Parmi les développeurs, certains sont eux-mêmes des pirates et appartiennent à la clique susmentionnée. Aussi triste à dire que cela soit, certaines des plus extraordinaires collections de progz que j'ai vues appartenaient à des développeurs.

D'autres développeurs, enfin, ont une rancune personnelle contre les RTC pirates qui les ont détroussés, sans avoir les moyens d'engager des poursuites. C'est pourquoi ils ont entamé une action clandestine, dont je vais vous parler, tout en admettant que cela va faire grincer pas mal de dents. Ils évitent d'en parler, voire le nient, pour des raisons juridiques. Mais ces agissements insidieux se poursuivent néanmoins. D'aucuns trouvent que c'est une bonne idée, d'autres pensent que c'est une abomination. La controverse bat son plein.

L'idée de la manip a peut-être germée suite aux activités d'un pirate qui utilise les pseudos de "Dr Typo" et "Captain Wizard". Disons tout de suite que ce n'est pas un mystérieux inconnu pour moi, j'ai même appris de sources différentes que c'est le gérant d'une boite de la côte est qui vendait des périphériques et des logiciels pour Atari 8 bits. Car contrairement au dicton, les loups se mangent entre eux : il m'a suffit d'offrir un Magic Sac pour obtenir le nom et le numéro de téléphone du Dr Typo. Bien sûr, je n'ai aucune preuve tangible, mais comme vous allez le voir, ce brave gars a déjà eu ce qu'il méritait.

## L'erreur qui tue

Ce cher Dr Typo avait donc déplombé la version 4.52 du Magic Sac, et avait même mis les ROM du Mac sur la disquette. Ainsi, inutile d'avoir la cartouche de l'émulateur pour faire tourner des logiciels Mac. Non seulement c'était illégal, mais en plus c'était incroyablement stupide : enfreindre les copyrights d'Apple est totalement suicidaire. Ils ont des armées d'avocats et des budgets colossaux pour anéantir toute velléité de ce genre. (Citons par exemple ce RTC de Cincinnati qui fut fermé, avec confiscation de l'équipement, et dont l'opérateur se vit réclamer des dizaines de milliers de dollars de dommages et intérêts pour avoir proposé en téléchargement une copie des ROM d'Apple. Paix à ses cendres...)

Dr Typo remonta ensuite ladite version 4.52 déplombée vers son RTC favori, et attendit les applaudissements de ses petits copains pirates. Pensez donc : la première version du Magic Sac à pouvoir utiliser le disque dur. Imaginez ma tête lorsque je découvris ce progz dans un des RTC que je surveillais. Mais le pirate avait commis une légère erreur, ses retouches au code étaient décalées d'un octet. Ce qui faisait que lors d'une interruption, si le registre DO du 68000 avait une valeur inférieure à 7, le secteur O du disque dur était réécrit, le plus souvent avec des zéros. Et la valeur de ce registre est parfaitement aléatoire lors d'une interruption, puisque l'on ne peut savoir à quel moment on interrompt le programme. Je suis sûr que cette erreur était accidentelle, car n'eut été ce décalage d'un octet, ses retouches étaient correctes.

Or le secteur 0 est absolument crucial pour le disque dur. Il contient la table des partitions, qui indique où se trouvent les différentes partitions du disque. Sans cette table, le disque est totalement inutilisable, et toutes vos données sur le disque sont perdues, à moins de trouver un sorcier qui puisse vous reconstruire ce secteur. Impossible sinon d'accéder au disque, sauf pour le reformater.

Ainsi, l'ingénieuse bidouille de Dr Typo avait pour effet de bousiller votre disque dur à intervalles aléatoires. C'est ce qui m'arriva au bout de cinq minutes de test de cette version déplombée (car naturellement, je l'avais descendue sur mon ST pour l'examiner). Diantre, mon disque dur était mort! Heureusement, cela survint tôt le matin, et j'avais sauvegardé le disque dur (en le copiant sur un autre) au début de la journée. De plus, j'avais les connaissances techniques pour découvrir ce qui s'était passé et pour reconstruire le secteur. Mais beaucoup de gens n'ont ni ces connaissances, ni les outils

adéquats.

Bientôt, la version 4.52 pirate du Dr Typo commença à bousiller les disques durs des RTC pirates qui l'essayaient. La plupart des opérateurs durent reformater leurs disques durs et recharger leurs sauvegardes, ce qui est en général l'occasion de s'apercevoir qu'elles sont ô combien obsolètes. Un bon nombre de RTC perdirent leur progz favoris.

Dan Moore et moi mîmes des messages dans CompuServe, GEnie et autres grands serveurs nationaux, mettant en garde contre le Magic Sac 4.52 déplombé. [NdT: Dan Moore est un programmeur sur Atari 8 bits, puis sur ST, redoutable tireur, barbe à la Castro, qui est suffisamment brillant pour avoir été embauché par AT&T bien qu'il n'ait aucun diplôme - une dérogation exceptionnelle dans cette firme.] Nous ne pleurions pas sur les malheurs de Dr Typo, mais désirions éviter à des utilisateurs de perdre leur disque dur juste parce qu'un copain leur avait amené une disquette "pour démonstration". Perdre 20 mégaoctets de données peut être tragique, surtout quand les sauvegardes sont incomplètes. (Plutôt qu'un lecteur de bande, achetez un disque dur d'occasion pour vos sauvegardes!)

Les pirates, qui fréquentent également ces grands serveurs, virent l'avertissement et l'affichèrent également dans leurs RTC - du moins dans ceux qui tournaient encore. Je vis ainsi apparaître ma mise en garde sur Coastline, qui avait la 4.52 pirate jusqu'à ce moment. Le résultat est que le seul nom de Dr Typo suscite l'animosité dans la communauté des pirates. Au lieu d'être applaudi, celui-ci commença à être victime des techniques de harcèlement que déploient les pirates les uns envers les autres. Programmer un modem pour faire sonner son téléphone toute la nuit, par exemple... Vu le temps que j'ai passé à développer le Magic Sac, je ne puis éprouver la moindre compassion pour cet aigrefin.

Le plus drôle est que c'est sans doute à cause d'une faute de frappe [NdT : "typo" en anglais] que cette retouche était défectueuse. Dans le cas évoqué plus haut, le programme sautait à la routine de gestion du disque, elle-même initialisée pour lire le secteur 0, etc... Un accident fortuit s'il en est. Mais cela n'améliora en rien sa réputation dans le milieu des RTC pirates. Le nom de Dr Typo reste gravé dans la mémoire de tous les opérateurs qui ont dû reconstituer leur disque dur après avoir essayé sa bidouille. Après un tel désastre, le Dr Typo dut se faire oublier, et le pirate opère désormais sous le pseudo de "Captain Wizard". Aux dernières nouvelles, il déplombait toujours mes émulateurs Mac, et niait farouchement être Dr Typo, de crainte d'être lynché. (J'espère que cet article tombera entre de bonnes mains. Si, Si, c'est lui, Captain Wizard est bien le nouveau pseudo de cet abruti.)

#### La vengeance des développeurs

Cette erreur subtile me donna matière à réflexion. Et si la méthode de Dr Typo était reprise par des développeurs? Si des développeurs mettaient délibérément en circulation des programmes annonçant "déplombé par Untel" qui endommagent les ST sur lesquels ils tournent? J'en discutai avec quelques personnes, et à ma grande surprise, il me dirent : "Tout à fait entre nous, Dave, ne l'ébruite pas, mais nous le faisons déjà."

Je ne vais pas divulguer les noms de ceux qui ont avoué se livrer à ces pratiques controversées. Mais croyez-moi, il y a déjà d'excellents développeurs à l'esprit militant qui agissent ainsi. La technique courante semble être celle de la corruption à retardement de la FAT [NdT: File Allocation Table, table d'allocation des fichiers, qui indique sur un disque quels secteurs sont occupés par quels fichiers]. Cela fonctionne de la manière suivante : un développeur réalise une version spéciale de son programme, qu'il désigne clairement comme étant une version pirate. Si quelqu'un exécute ce programme, il est impossible d'ignorer qu'il s'agit d'une copie illicite. Si l'utilisateur persiste, le programme commence son insidieux travail, tout en semblant fonctionner parfaitement.

Le logiciel vérifie d'abord la date. Le but est de laisser le temps aux pirates de diffuser le programme sur suffisamment de RTC pirates. Il ne faut donc pas les déglinguer trop vite. Au bout d'un délai qui peut varier entre quinze jours et six mois, le disque dur du RTC commence à souffrir de problèmes quasi imperceptibles au niveau de sa FAT. Le problème s'étend peu à peu, les pointeurs internes, qui

indiquent au système d'exploitation quels sont les secteurs qui composent un fichier, sont réaffectés à d'autres secteurs. Le résultat est que la bibliothèque du pirate est progressivement corrompue. Quelqu'un qui tentera de descendre un émulateur de terminal, par exemple, obtiendra un bout du programme d'émulation, un bout d'un autre fichier, un secteur vide... Tout ce qui est dicté par la FAT corrompue. Le programme voulu ne marchera certainement pas.

Bien entendu, les dommages surviennent tout doucement, si bien qu'il est dur de dire lequel des programmes (il y en a souvent des centaines) est le coupable. Les utilitaires de vérification du disque dur n'y voient que du feu, car les modifications de la FAT ont une apparence normale. Les utilisateurs du RTC pirate, eux, vont pousser les hauts cris lorsqu'ils verront que les progz téléchargés depuis le serveur ne marchent pas. C'est alors que l'opérateur du RTC découvrira que toute sa bibliothèque est esquintée. Avec un peu de chance, ses sauvegardes sont également endommagées, puisqu'elles ont probablement été faites lorsque le problème des FAT était déjà latent. Autant reformater le disque et recommencer. Notez que, contrairement à un virus, cette méthode n'inflige aucun dégât à un utilisateur innocent. Le problème ne survient que si le système exécute le programme piraté.

J'ai discuté de ceci, un soir, avec un pirate repenti, dans une conférence télématique. Il dit que les RTC pirates en sont réduits à utiliser des programmes de "protection en écriture" des disques durs, afin de tenter de protéger leurs disques durant les tests de programmes. Ce qu'il ignorait est que ce genre de programme (comme PROTECT.ACC que j'ai aidé à écrire) peut être court-circuité en accédant directement au matériel, sans passer par les routines du système d'exploitation. Face à un programmeur déterminé, il n y a aucun moyen d'interdire l'accès au disque dur.

Depuis l'avènement des virus sur ST, le problème a pris de grandes proportions. Je songe à modifier mon disque dur pour relier le signal d'écriture à un interrupteur physique actionnable de l'extérieur du ST pour créer une protection infranchissable par logiciel. Des virus délibérément nuisibles sont en effet à présent fréquents dans la communauté du ST.

#### Vos réactions?

Voyez-vous, les RTC pirates ont mangé leur pain blanc. Auparavant, on pouvait télécharger un programme et sa documentation en quelques minutes, et il marchait. A présent, les choses ont changé. Ce programme qui affiche fièrement "déplombé par SuperPirate" pourrait bien attendre sournoisement le jour fatidique où il effacera votre disque dur. Et ce, en dépit de vos protections. Il pourrait même être assez roué pour altérer vos fichiers Si graduellement que vos sauvegardes en seraient inutilisables.

(D'autres méthodes existent, comme par exemple brûler un moniteur couleur en forçant le ST en mode monochrome. Tous les développeurs contactés ont rejeté cette méthode comme étant trop radicale. Sauf un...)

Je veux souligner ici que je ne suis pas l'auteur de ces manips, et que même le Dr Typo n'en est pas responsable (ce qu'il faisait était accidentel). Mais la solution semble raisonnable. Si une personne exécute un programme clairement marqué comme étant piraté, elle vole la propriété intellectuelle de quelqu'un. De plus, je suis aux premières loges pour comprendre l'acrimonie de ces développeurs qui veulent se venger. J'ai vu des amis faire faillite, se retrouver ruinés, et quitter le monde du ST à cause de la piraterie. Eh oui, on en est là. Et les RTC pirates en sont le plus pernicieux vecteur. Leur malhonnêteté tue le marché du logiciel pour ST, donc le ST lui-même. C'est pourquoi je comprends qu'on puisse utiliser des méthodes quelque peu déplaisantes pour la combattre.

Vos réaction m'intéressent. Mes coordonnées figurent en fin d'article, n'hésitez pas à m'écrire. Moi, je ne prends pas position. Je lirai votre avis avec intérêt. Et je ne puis me résoudre à trahir la confiance des développeurs qui m'ont fait des aveux. Après tout, c'est la première fois à ma connaissance que des développeurs s'en prennent activement aux RTC pirates. Et contrairement à la protection des disquettes, qui empêche l'acheteur honnête de sauvegarder son logiciel, cette attaque ne nuit qu'aux pirates. Et si cette méthode se répandait? Si un programme piraté sur deux massacrait les disques durs qui le reçoivent? Les mots "logiciel piraté" prendraient la même connotation que "partenaire sexuel de rencontre" en ces temps de Sida...

## Le mois prochain...

Nous poursuivrons notre exploration des RTC clandestins, et verrons comment on passe du bricolage amusant au vol, à l'escroquerie, ou pis encore.

Traduction et adaptation : Password 90 - Les précisions concernant le Minitel sont des ajouts du traducteur.

Frédéric Mora

# Le cauchemar des études

Les études sont parfois un véritable cauchemar pour un élève un peu timide, comme le fut Dave Small. Le programme scolaire devient plus souple après le bac, car on peut choisir sa filière. Mais au lycée, en France comme aux USA, règnent des programmes scolaires très rigides et très lourds. Tous n'engendrent pas le même intérêt chez les élèves. Que faire pour survivre en dépit des professeurs, du matériel indigent et du stress? Dave nous présente ses règles de survie. Il les a appliquées alors qu'il avait entre 14 et 17 ans, car le système scolaire américain propose des études par modules "à la carte" similaire à nos unités de valeurs universitaires dés le lycée. Dave avait choisi de passer un bac scientifique avec l'informatique comme matière principale. L'étudiant français peut aisément mettre en pratique ces conseils dés qu'il aborde les études après son baccalauréat.

### Le survivant

J'ai survécu au lycée. Plus précisément, j'ai survécu à des études qui m'ont gratifié d'un diplôme de bachelier en science informatique. Ce fut dur. J'ai émergé de ce champ de ruines qu'est le système éducatif supérieur, escaladant des débris fumants, contournant des cratères, évoluant dans un paysage calciné à l'irakienne, tel un Mad Max s'échappant d'un camp de prisonniers. Douze années me séparent à présent de ce douloureux internement, et il est de mon devoir de transmettre quelques mots d'avertissement aux jeunes gens qui vont se risquer, peut-être pour la première fois, dans cet enfer. Ainsi mon expérience ne sera-t-elle pas perdue.

A quoi servent les études? De mon point de vue, il y a deux approches possibles. Première approche, vous pouvez sortir de vos études sans diplôme. [NdT en France, où la condition suffisante à l'obtention du bac est d'être en vie, "diplôme" signifie "post-baccalauréat". Le bac n'est plus rien en soi.] Impossible alors d'obtenir un travail bien payé, voir un travail tout court. Dans le meilleurs des cas, vous échouez dans des boulots qui ne sollicitent guère votre créativité, comme tondre des pelouses, cuire des hamburgers, ou tenir le guichet d'une station service. (J'allais dire "faire les pleins", ce que je suis assez vieux pour avoir fait. Mais il y a belle lurette que personne ne vous sert plus dans les stations.) Et encore devez-vous vous estimer heureux.

On peut faire mieux. C'est la seconde approche. Il s'agit de décrocher un diplôme. Hélas, personne ne vous explique jamais les détails importants de la vie estudiantine qui permettent d'y parvenir! C'est pourquoi je me dois de combler cette lacune. Oh, vous entendez bien parler de soirées mémorables où la bière coule à flots, de filles aguichantes, et de bals de fin d'année. Mais pas de la façon d'obtenir ce précieux bout de papier, grâce auquel des centaines de firmes vont rivaliser pour vous embaucher, et vont aller jusqu'à envoyer des représentants dans votre établissement pour discuter avec vous et vous prier de venir programmer chez elles, moyennant un gros salaire! [NdT c'était le cas en France jusqu'à il y a quelques années, mais depuis, la chasse aux jeunes recrues a quasiment disparu, hélas.]

Cela me semblait la voie royale. C'est celle que j'ai adoptée. Mais...

Règle 1 : Il y a des façons d'échapper au système.

### Les autres voies

Personne ne m'avait dit qu'il y avait d'autres options. Il y a des gens pour qui les études sont une catastrophe. Certaines personnes, parmi les plus heureuses que je connaisse, ont pris une ou deux

années après le lycée pour travailler, voyager, ou voir à quoi ressemblait le monde (le vrai, pas l'univers clos de l'éducation), avant de commencer leur troisième cycle. [NdT : renseignez-vous soigneusement avant de tenter cela en France, tous les établissement n'acceptent pas cette "fantaisie".]

Certains des meilleurs bidouilleurs que je connaisse considéraient les études comme une perte de temps. Ils étaient déjà en train de concevoir des synthétiseurs sonores pour Apple II ou des duplicateurs de disquette, et leurs préoccupations étaient de mesurer précisément les variations de flux magnétique sous une tête de lecteur de disquette toutes les 62,5 nanosecondes. Le meilleur bidouilleur, en matériel comme en logiciel, que j'aie jamais rencontré n'a pas été au-delà du lycée. J'ai eu la chance de l'avoir comme opérateur système sur l'ordinateur de notre lycée, et il m'a permis de découvrir que je pouvais faire quelque chose de créatif tout seul : programmer sur un Hewlett-Packard 2000C en temps partagé. Mais attention, ces gens ont pu se le permettre parce qu'ils étaient géniaux. A l'inverse, le fait de ne pas dépasser le bac ne vous rend pas nécessairement aussi doué.

Il vous faut savoir que la moyenne d'âge de la population ne cesse de s'élever (elle dépasse 33 ans aux USA), et que les employeurs [NdT ceux qui embauchent encore...] cherchent désespérément à recruter des gens jeunes et qualifiés pour apporter un sang neuf à leur entreprise. Ces jeunes recrues sont dures à trouver, et encore faut-il qu'elles soient utilisables directement. C'est pourquoi il n'est sans doute pas inutile de consacrer un ou deux ans à obtenir un diplôme supplémentaire très terre-à-terre, complété par un stage en entreprise, afin d'acquérir une expérience informatique dans le monde réel. Un employeur appréciera qu'un candidat ait une certaine expérience des langages de programmation ainsi que des grands classiques parmi les logiciels (Lotus 1-2-3, Word Perfect, dBase...), voire qu'il sache se dépatouiller avec un logiciel de PAO et un Mac.

Votre expérience de l'univers Atari ne constitue aucunement un handicap, au contraire. Vous serez passablement ennuyé par les autres ordinateurs et leur stupide façon d'opérer, comparé au confort de votre ST, et votre Atari servira souvent de modèle pédagogique. Voir des dossiers dans une fenêtre au lieu d'une liste de sous-répertoire, par exemple, a instantanément clarifié la notion de répertoires contenant des fichiers pour un de mes amis.

## La cruelle vérité

Durant votre première année de lycée, vous allez découvrir un nouvel univers où se passent des choses étranges. La première avec laquelle vous allez devoir vous familiariser est ce que l'on appelle les notes. Je vais vous expliquer en détail cette institution ubuesque. Vous vous voyez attribuer une note pour votre travail, pour vos tests, vos devoirs, selon le nombre de fois où le professeur vous voit demander son aide, selon que vous étiez assis ou non au premier rang de façon à ce qu'il se rappelle votre faciès, etc. Il vous faut savoir qu'une note de 10/20 est considérée comme moyenne, et donc, en théorie, vous place au milieu du troupeau : la moitié des gens sont devant vous, la moitié derrière.

Et pourtant, 10/20 est la barre limite en deçà de laquelle on vous éjecte de l'école. Je ne le sais que trop bien, puisque mon premier semestre se solda par un 9/20 je plaide le choc culturel). Je fus mis à l'épreuve durant le second semestre, et parvins à hisser ma moyenne générale bien au-delà des 10/20 fatidiques [NdT : *le cycle scolaire américain est divisé en semestres*]. Alors, cherchez l'erreur : Si vous êtes "moyen", vous êtes 50ème sur 100, et il y a 50 personnes derrière vous qui sont censées être chassés de l'école, chaque semestre!

Cela signifie que chaque semestre, la population scolaire devrait décliner de moitié. Ou encore, que sur 100 personnes, il en reste 0,39 (quatre dixièmes d'un étudiant?) au bout de quatre ans, ce qui est tout de même exagéré. [NdT : le lycée américain comporte quatre années avant l'équivalent du bac, qu'on obtient en général à 17 ans.]

Règle 2 : Les notes, comme la physique quantique, procèdent d'une autre réalité.

Et pourtant, les universités regorgent de bacheliers, ce qui prouve que la sélection n'est pas aussi sévère. Il y a donc un fossé ahurissant entre les notes et la réalité, et vous devez retenir que les notes n'ont aucune signification dans le monde réel. C'est une notion que vous pouvez utiliser à votre profit. Ou qui peut se retourner en votre défaveur.

Règle 3 : Les notes de lycée doivent être considérées comme une série de chiffres plus ou moins aléatoires, et ne sont pas représentatives de votre intelligence, de vos capacités ou de votre ardeur au travail

Exemples, dans les deux sens. Dans un cours d'informatique particulièrement nul, le professeur nous infligea un test d'une heure, où il s'agissait d'écrire sur papier un programme. J'écrivis le mien et le vérifiai du mieux que je pus. Durant la correction, le prof marqua "faux" un sous-programme parfaitement correct. J'allai le voir, et ligne par ligne, nous exécutâmes manuellement le programme entier. Naturellement, mon sous-programme était bon. Mais plus loin, vers la fin de mon code, j'avais fait une erreur banale, Si bien que le prof me déclara "Je savais bien qu'il y avait une erreur quelque part."

Règle 4 : Les professeurs sont prêts à toutes les bassesses pour se débarrasser de la corvée de la notation le plus vite possible.

Soyons sérieux. Combien de programmes tournent-ils parfaitement lors du premier essai? Moi, cela ne m'est arrivé qu'une fois. Peut-on attribuer une note sur un tel coup de chance?

C'est pourquoi j'eus une mauvaise moyenne à ce cours, en raison d'une attitude sceptique et irrespectueuse.

J'assistais aussi à un cours d'électronique, où le budget limité de l'école nous contraignait à n'utiliser que des portes NAND - des circuits intégrés logiques implémentant la fonction booléenne Non-Et, dont la sortie valait O si et seulement si les deux entrées valaient 1. Il nous fallait donc reconstituer tous les autres types de portes uniquement à partir de circuits NAND afin de faire des montages. Rien d'infaisable, notez bien : Seymour Cray a construit toute la logique du Cray 1 uniquement à partir de porte NAND en technologie ECL (circuits bipolaires, extrêmement rapides mais à forte consommation et difficiles à miniaturiser.) Bien sûr, nous passions 90% de notre temps à câbler et décâbler des portes logiques. Au moins, ça nous occupait, à défaut de nous plaire...

Dans un autre cours, le manuel utilisé était celui écrit par le professeur. Et il n'était pas terminé. Nous ne disposions que de photocopies du manuscrit, avec de petits croquis dessinés à la main. Nous devions trouver tous les "bogues" et les fautes du manuel (il y en avait des tonnes!), et gratuitement, bien sûr. Le manuel, lorsqu'il sortit, ne coûtait pourtant pas moins cher.

Règle 5 : Souvenez-vous que vous payez tout ça.

Vous le payez directement ou avec vos impôts. Alors autant vous débrouiller pour faire fonctionner ce système stupide à votre profit, grâce à une astuce, les "travaux pratiques libres". Les TP libres, cela consiste à avoir des rapports amicaux avec un prof et concocter avec lui un TP que vous ferez seul, à la place d'un cours ou d'un TP classique, et dans lequel vous excellerez, parce que vous serez vraiment motivé.

## Les tp libres

J'avais ainsi fait un TP libre sur un synthétiseur sonore ARP-2600 (oui, je sais, ça date...), et mon "compte-rendu" consistait en une bande de démonstration de deux minutes. Dans cette bande, je mis toute les technologies sonores possibles. Des simples, comme un sustain de guitare commandé par feedback. Et des moins simples, comme une pédale wah-wah commandée par une photorésistance placée devant une lampe mise à la sortie d'un ampli Sherwood j'envoyais au Sherwood un signal sinusoïdal, augmentais sa fréquence, et cela produisait un son que j'adorais (je l'ai même mis dans la page cachée de mon émulateur Mac). Je me suis régalé. Cela m'a permis d'étendre mes connaissances, et de pousser à leurs limites mes capacités (et celles des transistors de l'ARP!). J'ai obtenu un 20/20. Les très bonnes notes ont le même effet que les potions de soins d'un jeu de rôle, elles annulent les 5/20 récoltés dans des classes enseignées par des rigolos.

C'est là le plus important conseil que je puisse donner pour sortir à peu près intact de vos études: les TP libres.

Règle 6 : Si les cours sont médiocres, faites-vous vos propres TP. Au moins, vous apprendrez quelque chose!

C'est ce qui m'a permis d'obtenir mon diplôme en dépit de l'une des plus basses moyennes générales qu'on ait jamais couronnée d'un diplôme à cet établissement. De plus, j'ai appris beaucoup de choses qui me furent plus tard utiles pour ma vie active, beaucoup plus que si je m'en étais tenu au cursus classique.

Apprenez à connaître vos professeurs. Ce sont des êtres humains, même s'ils sont un peu bizarres. Ne faites rien pour énerver les cinglés, mais sachez plaire à ceux qui sont vraiment sympas. Ils y en a toujours quelques-uns qui acceptent volontiers d'aider un élève à sortir des sentiers battus. Beaucoup de profs font de la politique ou s'écoutent parler, mais certains essaient sincèrement d'étendre le champ des connaissances de leurs élèves, et les poussent à étudier par eux-mêmes.

C'est ce que j'ai fait j'ai étudié par moi-même le langage assembleur sur l'ordinateur central de l'école. J'ai beaucoup appris. Oui, et cela m'a bien servi! J'ai découvert des choses étonnantes au sujet des partitions mémoires, par exemple, ou sur le superviseur de travaux. J'ai aussi découvert ce à quoi les constructeurs informatiques devraient penser avant d'autoriser des travaux interactifs en temps partagé sur un système initialement conçu pour le traitement par lots, et les faiblesses qui en découlent.

Règle 7 : Parfois, vous pouvez apprendre quelque chose d'important.

En temps partagé, chaque terminal d'un gros système dispose d'une fraction du temps de calcul de la machine. Chaque programme est traité tour à tour. Alors qu'en traitement par lot, les programmes sont traités individuellement, un par un, comme un ordinateur monotâche.

Eh bien, à ma grande surprise, je découvris que sur notre machine, lorsqu'on interrompt une compilation FORTRAN depuis un terminal en temps partagé, l'image mémoire de tout le système d'exploitation de l'ordinateur est copiée sous forme d'un fichier local. En traitement par lots, une telle interruption est impossible, c'est pourquoi le constructeur n'avait pas couvert cette éventualité. Tout étudiant pouvait donc désassembler ce programme superviseur et l'étudier, nuit après nuit, dans le relatif confort de sa chambre.

Voyez-vous, c'est triste à dire, mais le système des notes implique que vous êtes en compétition avec vos camarades de classe. C'est bien dommage, mais le professeur doit attribuer des notes, des bonnes comme des mauvaises. Si besoin est, il inventera un moyen de distribuer ces notes (comme faire écrire un programme en classe).

Règle 8 : Les notes sont nécessaires, et même si tout le monde est excellent, il faut en trouver qui récolteront les mauvaises notes, et d'autres les bonnes.

J'ai subi cette compétition d'une manière très visible, en essayant de trouver assez de temps de calcul pour déboguer mon programme, en compétition avec ma classe. Quand les quelque 50 étudiants de ma classe s'abattaient sur le centre de calcul, chaque programme pouvait avoir à passer des heures dans la file d'attente avant d'avoir son tour et d'être exécuté une seule fois! Cela rendait le débogage impossible : nous écrivions des programmes complexes, comme des assembleurs et des compilateurs. Beaucoup de gens excellents se sont épuisés à passer des nuits blanches complètes devant leur terminal.

Règle 9 : Concernant la compétition, au moins, le lycée peut former pour affronter le monde réel.

Et tout le monde était logé à la même enseigne, gaspillant son temps à attendre une énième recompilation. A moins, bien sûr, d'avoir étudié le code assembleur du répartiteur de tâches de la machine, et d'avoir spécifié une très petite partition mémoire pour votre programme, au lieu de la taille par défaut, bien trop grande. Le répartiteur de tâche était conçu pour "remplir les trous" dans la mémoire de l'ordinateur, et pouvait y caser mes petites partitions, alors que les énormes programmes Pascal (55 Ko) de mes camarades n'y rentraient pas. Ce qui me permettait d'avoir autant de recompilations et d'exécutions que je pouvait en dèboguer, là où les autres en avaient une par heure et peinaient pour trouver leurs bogues en déployant des trésors d'ingéniosité.

Le répartiteur de tâches avait également une autre "caractéristique". Tel qu'il était écrit, il gardait les travaux de N utilisateurs en mémoire, tout en les exécutant. Quand votre programme avait passé 10 secondes en mémoire et obtenu un peu de temps de calcul, il était "paginé" sur disque et laissé là à

moisir un bon moment, se retrouvant en fait derrière tous les autres dans la file d'attente. Bon, maintenant, appuyez sur la touche Break de votre terminal. La machine doit obligatoirement traiter votre interruption. Elle vous demande alors si vous voulez continuer. Répondez O (oui). Du coup, puisque le systéme a dû traiter votre interruption, il a remis votre programme en mémoire, et vous voilà reparti pour 10 nouvelles secondes!

Et devinez ce que vous faites 10 secondes après? Et ainsi de suite jusqu'à ce que votre programme soit entièrement exécuté.

Grâce au ciel, les autres ne découvrirent pas le truc jusqu'à l'examen final. Il m'aurait été impossible de terminer mes programmes sans cette astuce j'étais à court de temps, j'assistais à beaucoup de cours, essayant de ramasser assez d'unités de valeurs pour m'échapper de cette prison.

### Le stress

L'absurdité de ces notations engendra un stress très éprouvant pour nombre de mes camarades. Toute leur vie, on leur avait répété que les notes étaient significatives et qu'une bonne note était importante. Ils se retrouvaient dans une situation où obtenir une bonne note devenait impossible, parce qu'un programme ne peut pas marcher correctement sans de nombreux essais et sans suffisamment de temps de calcul pour dèboguer à fond les algorithmes de base, comme nos tables de hachage. Il y a certes d'autres facteurs, mais en définitive, c'est le stress qui épuise les gens durant leurs études. (Certes, c'est une bonne préparation à la vie, où l'on doit subir un patron, des impôts et autres nuisances, mais je pense que c'est à éviter.) Vous devez absolument trouver un moyen de vous détacher, de ne pas être obsédé par les notes.

Règle 10 : Eviter le stress est sans doute la chose la plus importante que vous devez apprendre durant vos études.

Une fille que je connaissais a fait une surdose accidentelle. Elle travaillait trop dur, et elle avait commencé à se droguer pour essayer de se détendre et d'échapper au stress. (Elle s'en est finalement sortie mais sa vie en a été gâchée, et elle cela lui a coûté ses études.) Il est difficile pour moi de porter un jugement sur elle, vu que j'ai constaté que le stress vous détruit encore plus rapidement, et vous fait mourir encore plus jeune, que la drogue. J'ai vu beaucoup plus de gens se droguer dans l'industrie informatique que je n'en ai vu au lycée ou à l'université. Si, au lieu de prendre des calmants licites, quelqu'un choisit une... hem... façon différente de traiter les problèmes que lui donne son travail chez une firme informatique renommée, je ne vais pas lui faire la morale. Le stress tue.

Maintenant que je vous ai déprimé en vous montrant la futilité des études, laissez-moi vous parler de leur utilité réelle.

Les études vous placent dans des conditions qui se rapprochent du combat de la vie réelle. Elles exigent de la discipline pour en réchapper et vous préparent à ce que vous trouverez quand vous aurez votre diplôme. Et que signifie réellement un diplôme? Seulement qu'une firme peut vous embaucher en sachant qu'en plus de quelques connaissances de base, vous avez acquis une aptitude à vous accrocher, à vous battre, à survivre malgré la tambouille des restos universitaires et les conditions de travail inconfortables. Ainsi avez-vous une chance de survie dans cette firme durant les deux premières années où vous allez lui coûter cher, le temps que vous appreniez votre métier. Le diplôme signifie que vous êtes un investissement qui en vaut probablement la chandelle. Rien de plus, mais rien de moins.

Règle 11: Les meilleurs, les plus brillants, ne s'embarrassent pas de perdre 4 ans à subir un test d'endurance au stress, et ne vont jamais au lycée. Wozniak est allé au lycée après avoir fait d'Apple le succès que l'on sait.

### Les bons aspects

Les études ont de bons aspects, et Si vous savez en profiter, elles en deviennent presque supportables. Un défouloir est indispensable, ce qui explique le succès des équipes universitaires de rugby.

Mais les études fournissent aussi l'occasion de rencontrer de belles jeunes filles dont l'œil pétille de ce regard si particulier (je parle évidemment de mon point de vue), qui fréquentent de nombreuses soirées et sortent beaucoup. Je me souviens d'une charmante Mindy à mon école. Curieusement, au bout de deux ou trois ans, elles arborent une bague de fiançailles, puis quittent l'école et se marient. C'était si répandu que nous avions donné un nom poétique à cette activité, "préparer un diplôme de recherche de bon parti". Étant passé dans le collimateur, je sais à présent ce que ressent un cerf le jour de l'ouverture de la chasse.

Règle 12 : Il y a des choses qui peuvent coûter trop cher.

Règle 13 : Tout le monde regrette de n'être pas davantage sorti au lycée.

Règle 14 : Tout le monde oublie combien il peut être terrifiant de demander ou d'accepter de sortir avec quelqu'un.

Durant vos études, vous pouvez découvrir pour la première fois de votre vie qu'il y a beaucoup de gens avec qui vous vous entendez, surtout si vous avez une personnalité de type solitaire (NF ou NT) [NdT : voir la chronique de Dave Small dans ST-Mag n° 64]. Les second et troisième cycles concentrent tout particulièrement les NT dans les sciences, l'informatique, l'architecture. Bidouilleurs solitaires, prenez-en bonne note! Avec un peu de chance, vous vous rappellerez comme vous vous entendiez bien avec ces autres NT, comme leur compagnie rendait tolérables les longues nuits de travail. Et vous chercherez à travailler dans un endroit où ils se regroupent.

(Et quant à ceux qui sont dans des coins isolés, les réseaux télématiques sont ce qu'il y a de mieux pour communiquer avec des gens qui vous comprennent.)

# Ma façon de faire

Bon, et que m'est-il finalement arrivé? Eh bien, j'ai passé les deux premières années de lycée en tâtonnements maladroits, jusqu'à ce que je me fixe un objectif: TP libres et la note minimale requise pour les cours obligatoires, que je considérais comme inutiles de toute façon (ce qui s'est avéré vrai : la théorie des gros systèmes centraux est plutôt désuète de nos jours).

J'ai vécu en chambre de cité universitaire pendant un an, me suis vite lassé de la nourriture, et ai atterri dans un appartement occupé par deux filles. N'allez pas vous figurer des choses, elles cherchaient un troisième co-turne pour partager le loyer. Partager l'unique salle de bain fut une épreuve redoutable, mais elles réussirent plus ou moins à m'apprendre la propreté ("Si, on doit nettoyer la crasse dans la baignoire après son bain".) Chose rare, nous sommes toujours amis : l'une est programmeuse, l'autre est en Angleterre avec son officier de l'Air Force de mari.

Ceci dit, j'ai appris à apprécier la popote du réfectoire. Elle est fade, mais elle est abondante [NdT : *il faudra parler à Dave des restos U français où elle est fade et chichement servie*]. Une boîte de haricots en guise de dîner, ce n'est pas la joie.

En outre, grâce à un ami, j'ai rencontré dans mon dortoir une fascinante jeune femme nommée Sandy Heidlebaught, qui avait lu les mêmes livres que moi, dont les rêves n'avaient pas de limites, qui savait où se situait Mordor, et connaissait son nom en elfique. Celui-ci est d'ailleurs toujours gravé dans les rochers qui surplombent le barrage dans la montagne.

Ma seconde année de lycée gravita autour de Sandy. Rien n'était plus important. L'informatique? Soyons sérieux. J'ai beaucoup appris à son contact, traversant orages et disputes, et nous nous sommes mutuellement civilisés, abandonnant des particularités irritantes l'un au contact de l'autre. Ce fut, et de loin, la chose la plus importante que je fis au lycée.

Règle 15 : Le lycée est l'endroit rêvé pour rencontrer quelqu'un d'intéressant, car c'est là que se concentrent les gens que vous cherchez à rencontrer. Ça vous économise bien des sorties inutiles.

Finalement, nous avons passé notre bac, et j'ai tant pourchassé Sandy qu'elle a fini par m'attraper. C'est une façon de faire. Il y en a d'autres. Tâchez d'en trouver une qui vous convienne et de réussir.

Titre original : Goin'to School Traduction et adaptation: Password 90 [qui a réussi ses études grâce aux TP libres!]



# **Qui est vraiment Jerry Pournelle?**

PAR DAVID SMALL De nombreux lecteurs de ST-Mag lisent également le magazine américain Byte, réputé pour sa qualité. Et la chronique la plus lue de Byte est celle de Jerry Pournelle. Pournelle est un écrivain de science-fiction qui possède un solide bagage en informatique et en haute technologie. Il appartient à un comité qui conseille le gouvernement américain sur la politique spatiale. Il est également docteur en psychologie et ancien ingénieur. Pournelle a la chance de pouvoir essayer les tout derniers programmes et machines de l'industrie, que les firmes se battent pour lui faire tester. Encore que cette chance soit parfois une malédiction, car le succès de ses articles fait de Pournelle un personnage aussi harcelé par les fabricants de micro-informatique que le Prince Charles par les journalistes. Pournelle serait-il un de ces nantis conformistes ne jurant que par leur PC ? Pas du tout! Dave nous décrit ce personnage haut en couleur du monde de la micro-informatique.

### Un auteur à succès

Beaucoup de gens lisent les revues américaines d'informatique Byte et Infoworld. Dans ces deux revues paraissent des chroniques de Jerry Pournelle. Ces articles sont remarquables, parce qu'ils n'hésitent jamais à prendre des positions controversées, ce qui détonne dans le ronron ambiant.

Davantage de gens encore lisent des romans de science-fiction. Et parmi les amateurs de SF, le nom de Jerry Pournelle est bien connu. [NdT pour de sombres raisons de bisbille entre agents littéraires, équivalents des imprésarios pour les romanciers, les romans de Jerry Pournelle ne sont pas traduits en français, à l'exception de "La poussière dans l'œil de Dieu", ce qui est regrettable.] Il a écrit des romans qui ont renouvelé le genre, comme la série des "Janissaries" (4 tomes parus), et a également collaboré avec Larry Niven, coauteur de plusieurs autres titres.

Lorsque l'ai sorti mon premier émulateur Mac, Jerry fut l'un des premiers à en parler, et c'est en grande partie grâce à son soutien que l'ai pu le terminer. Nous n'avions pas d'argent pour en faire de la publicité, et personne ne se doutait qu'il était possible de faire tourner des programmes Mac sur un ST. Mais grâce aux articles de Jerry, des clients potentiels nous ont contacté, et nous avons su qu'il y avait un marché pour cet émulateur, ce qui justifiait nos efforts et nous a poussé à en finir la mise au point. J'ai régulièrement envoyé à Jerry les mises à jour du produit, et l'ai souvent rencontré lors de salons informatiques, ce qui m'a permis de garder le contact avec les Pournelle.

Aussi, quand on parle de Jerry (c'est un sujet qui revient souvent dans l'univers de la micro), on me demande pourtant qui il est vraiment. C'est pourquoi je souhaite vous présenter ce personnage exceptionnel.

### Un sain anticonformisme

Tout d'abord, dans une industrie où les personnalités marquées sont rares et où le conformisme est de rigueur, Jerry s'impose comme une tâche de couleur dans un monde de grisaille. Par exemple, dans un salon informatique de San Francisco bourré de costards trois pièces, il débarqua une fois en tenue camouflée de brousse, avec un sac publicitaire du magazine "Soldier of Fortune" [NdT *magazine pour baroudeurs et amateurs d'attirail militaire, très anticommuniste*].

Cela provoqua quelques haussements de sourcils (aimable litote). Je ne voudrais pas entrer ici dans un débat politique, mais il vous faut savoir que dans la région de San Francisco, la presse informatique

est mise en coupe réglée par un réseau d'activistes aux idées très marquées à gauche, et qui favorisent par cooptation uniquement des gens partageant leurs idées. Cela remonte au noyau universitaire des premiers bidouilleurs qui innovèrent dans la micro. Le magazine informatique "Dr. Dobbs" avait par exemple publié en 1987 un article où l'auteur doutait qu'il soit moral de programmer pour une firme en rapport avec l'industrie militaire (l'auteur affirmait ne vouloir travailler sur des projets liés à la défense nationale que si la moitié de son salaire était donné à des mouvements pacifistes.)

Or, déplaire à ces gens est généralement suicidaire pour un écrivain ou un journaliste. Ils vous blackboulent, puis donnent votre place à un gratte-papier plus docile d'opinion. Et la profession grouille de gens avides d'écrire des articles...

Par son anticonformisme, Jerry ne fait donc pas qu'apporter un peu de fantaisie, il prend aussi des risques.

### Une voix de stentor

Si vous rencontrez Pournelle pour la première fois, attendez-vous à un choc. Ses articles sont d'un style calme, aimable, posé, du genre "notre ami Pournelle nous parle d'informatique". Et là, soudain, vous vous retrouvez face à un gars arborant un badge du magazine Byte, en tenue camouflée, et qui hurle dans vos oreilles!

C'est que Jerry a fait la guerre de Corée dans une unité d'artillerie. Il s'est retrouvé au mauvais moment (le début) dans un sale coin, où les troupes US ont bien failli se faire rejeter à la mer. Son unité a fait feu jour et nuit, et son ouïe en a été atteinte. Depuis, comme tous les gens qui n'entendent pas bien, il parle très fort.

Je me souviens qu'une fois, lors d'un salon, peu de temps après avoir fait sa connaissance, j'avais timidement parlé àJerry de mon émulateur Mac, encore en cours de conception. Et, ô joie, Jerry avait accepté d'y jeter un œil une fois qu'il serait prêt, alors qu'il était très sollicité. Je lui avais demandé de me dédicacer un de ses livres, puis m'étais éclipsé pour aller au toilettes. Et tandis que j'étais face à un urinoir, vidant consciencieusement ma vessie, Jerry était entré sans bruit derrière moi, et, m'apercevant, me lança de sa voie retentissante: "VOTRE LIVRE EST PRET, MONSIEUR SMALL !!!" Je sursautai, et eu le réflexe stupide de me retourner. Je manquai de peu d'asperger les chaussures de Jerry, ce qui, je le crains, aurait marqué la fin de ma carrière en informatique...

Autre chose frappante chez Jerry son regard. J'ai essayé de trouver les mots exacts pour le décrire, mais je crains d'avoir échoué. C'est un regard perçant, insistant, précis, que je n'ai vu que chez de très rares personnes. L'une était un très bon chasseur en compagnie duquel j'ai fait quelques randonnées. Une autre est mon beau-père, pilote d'essai qui risqua sa peau sur les prototypes des SR71 Blackbird. Un tel regard ne se trouve apparemment que chez les gens qui ont passé des heures à scruter le paysage sous le soleil, ou à vérifier des instruments de vol pour la millième fois... Lorsqu'on se retrouve sous le feu de ce regard, on a un peu l'impression d'être mentalement disséqué.

# Horreur des idiots

Ensuite, Jerry, je le cite, a "horreur des idiots". Une fois, j'ai vu deux adeptes du pacifisme le chapitrer au sujet des armes nucléaires. Ils avaient l'air sincère et dévoué de ces gens qui font du prosélytisme en porte-à-porte pour leur religion.

Ces gens ont tenu le coup environ une minute. C'est que si vous apportez la contradiction à Jerry, vous avez vraiment intérêt à vérifier vos arguments, à les étayer de faits irréfutables, et à les appuyer par un raisonnement sans faille. Sinon, vous vous faites descendre en flammes. Car Pournelle écrit des romans de science fiction "réaliste", plutôt de l'anticipation donc, et les fanas du genre se font un malin plaisir de souligner les failles d'une histoire mal ficelée et d'éreinter son auteur. Tenez, à propos de son

livre "Le marteau de Lucifer", un critique écrivait "Le problème avec la planète où se passe ce roman est qu'un monde de cette taille, à faible gravité, ne pourrait pas retenir une atmosphère à base d'oxygène et d'azote.") On trouve beaucoup de ces fanas intransigeants dans le monde des bidouilleurs et des techno centristes, entre autres.

Avec un public aussi exigeant, Jerry est habitué à vérifier très soigneusement ses arguments. Ni les mordus de SF, ni les lecteurs de Byte ne l'autoriseraient à se laisser aller à ses sentiments ou à des raisonnements bancals. Face à cet esprit méthodique, il est dangereux de se lancer dans de grandes envolées emphatiques. Sinon, le regard vous transperce, la voix vous indique en phrases nettes et tranchantes les déficiences de vos arguments, et vous vous sentez comme un gosse qui vient de cafouiller dans la table de multiplication devant la maîtresse.

A vrai dire, ce n'est pas tout à fait ce que l'on enseigne dans les cours de relations publiques ou dans les livres "comment se faire des amis". En théorie, il devrait poliment écouter ses contradicteurs, exposer son avis, écouter le leur patiemment, les remercier d'avoir pris le temps de le sermonner, et les quitter sur un sourire. Alors qu'en pratique, il se contente de leur poser quelques questions précises pour voir s'ils maîtrisent leur sujet, et dans le cas contraire, il les laisse en plan sur-le-champ.

Cette brusquerie est cependant nécessaire pour quelqu'un qui, comme lui, est constamment sollicité à tout propos. Il reçoit toutes sortes d'appels téléphoniques d'attachés de presse de firmes informatiques bien décidé à lui vanter leur nouveau produit et à lui arracher quelques paragraphes élogieux dans son prochain article. Si vous avez déjà été dérangé par un coup de fil d'un vendeur de cuisines ou de polices d'assurances; vous connaissez ce genre d'importuns. Jerry est sans cesse pris à parti par tout un tas de dévots qu'il a offensé en n'encensant pas leur produit favori, et qui sont bien décidés à lui faire entendre raison. Or, toutes ses activités informatiques ne sont qu'une occupation secondaire : Pournelle est avant tout un écrivain d'anticipation, qui a besoin de longues heures de calme, sans être dérangé par la sonnerie du téléphone.

J'ai pour ma part subi quelques appels à sept heures du matin ("Comment, vous n'êtes pas debout, M. Small? Mais il est neuf heures sur la côte est !"), de préférence après de longues sessions de bidouille nocturne, ce qui m'a conduit à me faire inscrire sur liste rouge. Je ne peux donc qu'approuver Jerry.

C'est pourquoi la brusquerie est pour lui une indispensable défense naturelle. Si vous prenez le temps d'apprendre à le connaître, Si vous lui montrez que vous n'êtes pas un idiot, et que vous n'essayez pas de le convertir à quelque noble cause, sa vraie nature se révèlera : celle d'une personne charmante. [NdT un tuyau, appelez-le "docteur Pournelle" - il est docteur es psychologie - pour l'amadouer.]

N'allez pas croire qu'il réserve ses bons côtés à ceux qui partagent ses opinions politiques. Jerry est rédacteur-en-chef d'un magazine qui parait sous forme d'un livre de poche plusieurs fois par an, et qui contient des articles de gens de toutes opinions, y compris opposées aux siennes. Mais ces gens ont en commun de connaître leur sujet à fond et de savoir raisonner.

## Les causes sacrées

Jerry n'aime ni le battage publicitaire, ni les causes "religieuses". Un sain réflexe pour un journaliste, car notre industrie abonde en produits et firmes sur médiatisées ! Prenons par exemple le Macinstosh. Sa sortie s'est accompagnée du plus énorme tapage commercial, de la plus gigantesque campagne de "relations publiques" - qui est l'art d'obtenir de la pub gratuite - jamais menés pour un microordinateur. Cette campagne a payé quantité de gens sont allés acheter un ordinateur avec à peine 128 Ko de RAM, un écran minuscule, et un lecteur de disquette quasiment occupé à 100% par les fichiers système. Des magazines ont été lancés rien que pour le Mac. La pression commerciale était énorme Apple a même fait passer des pubs frappantes à la télé durant la coupe du monde de football américain.

J'avoue avoir été l'un de ces idiots utiles, comme les propagandistes appellent ceux qui suivent leurs campagnes. J'ai acheté mon Mac au printemps 1984, et j'ai passé l'été, comme tout le monde, à

attendre que des logiciels daignent se matérialiser entre deux campagnes de promotion. Celles-ci étaient menées de main de maître. Je me rappelle par exemple que dans le magazine MacWorld paraissaient des tests de produits absolument inexistants. Finalement, la carte d'extension mémoire de 512 Ko est sortie, ainsi que le logiciel Excel de Microsoft, et la machine a enfin commencé à être utilisable.

Jerry s'était à l'époque attiré les foudres de nombreux mystiques du Mac, en les appelant la "tribu du Mac", et en remarquant qu'il n'y avait aucun logiciel pour la machine, que l'écran était petit, qu'elle manquait de mémoire et ainsi de suite.

Mais il est plus facile de convaincre quelqu'un de changer de religion que de convertir un véritable adepte de tel ou tel ordinateur. Et Apple a toujours été suivi par une large masse de partisans aveuglément loyaux, qui achètent (notez le mot-clé, "achètent") la contre-culture qui est associée à l'image de la firme. Ils achètent également ("achètent", encore, notez) les magazines sur le Mac, achètent (vous notez, hein 7) les derniers périphériques pour le Mac, etc. Du coup, vendre de la contre-culture, cela n'a plus rien à voir avec le négoce de quelques colifichets psychédéliques c'est une affaire extrêmement juteuse.

Quand le Mac eut finalement assez de mémoire et d'espace disque pour être utilisable, Jerry se laissa d'ailleurs fléchir et en fit usage volontiers. Néanmoins, quand il parla dans un article de notre premier émulateur Mac, il fut inondé de lettres d'insultes de la tribu du Mac, qui s'était offusquée de notre sacrilège. Comment avait-il seulement osé mentionner cette cartouche blasphématoire ? Mais vous devez commencer à vous douter de la façon dont Jerry "ose" faire quoi que ce soit.

## Pas peur des controverses

Sachez donc que (chose rare pour un journaliste) Pournelle n'hésite jamais à soutenir des causes impopulaires ou à susciter des controverses. Il méprise le terrorisme intellectuel. Son sac "Soldier of Fortune" (SOF) en est l'illustration parfaite. Il faut du courage j'ai osé une fois porter un T-shirt SOF à la Foire informatique de la côte Ouest, salon informatique branché s'il en est, qui se tenait à San Francisco. C'était juste histoire de rire le lendemain, je portais bien mon T-shirt "Gary Hart président" que m'avais donné mon associé d'alors [NdT: Gary Hart était un candidat démocrate à la présidence.] Qu'ai-je entendu pour avoir porté ce T-shirt SOF! On m'a fait de grandes leçons de politique. En plein salon, un type maugréait: "Assassins! Tous des assassins!" Étrange notion de la liberté d'opinion.

Parmi ces fanatiques court la rumeur que Jerry Pournelle écrivait jadis pour Soldier of Fortune. Or, contribuer à une revue maudite par les "faiseurs d'opinion", c'est pire qu'être un auteur mis à l'index par l'inquisition. En tout cas, cela vous attire la réprobation immédiate des zélotes, qui arborent alors des mines indignées et vous traitent de "meurtrier", ou de l'infamant "mercenaire", comme j'ai pu en faire l'expérience moi-même.

Eh bien, qu'ils soient rassurés : la rumeur est absolument exacte ! Mais avant qu'ils ne défaillent d'horreur, je précise que Jerry est grand amateur de simulations guerrières, d'analyse tactique et autres wargames, comme le savent les lecteurs de ses articles dans Byte. Ce qui est tout à fait dans les sujets couverts par S0F. Ne nous y trompons pas si cette revue se veut "le magazine des aventuriers professionnels", comme le clame son slogan de couverture, ses lecteurs sont souvent des aventuriers en chambre. Sous cette couverture racoleuse, que trouve-t-on ? Des articles plutôt anodins, comme "Comment nettoyer votre fusil", "Comment aiguiser votre couteau", ou encore "Les souvenirs d'un vétéran du Vietnam". Rien qui le démarque franchement des myriades d'autres journaux tournant autour des armes ou des militaires. Jerry leur a fourni des articles parlant de stratégie militaire. La belle affaire. Alors, ces glapissements effarouchés de sectateurs indignés sont-ils justifiés ? Sont-ils seulement sincères ? (Si ça peut les rassurer, Jerry a depuis longtemps cessé d'écrire pour eux, il trop occupé par ailleurs).

## Loyauté

S'il y a un comportement qui est considéré comme démodé, c'est bien la loyauté. Et pourtant, si vous prenez le temps de le connaître, et si vous en faites un ami, Jerry sera toujours loyal envers vous. Cette attitude surannée, vous en conviendrez, ne peut lui attirer qu'un mépris hautain dans les milieux "branchés" où il faut être vu au dernier cocktail en compagnie des faiseurs de prêt-à-penser dernière mode. Dans ces cercles élégants, la modernité semble nécessairement devoir s'accompagner d'une mentalité de rat d'égout où le grand chic est de piétiner des gens qui vous ont fait confiance. A cela s'ajoute, dans notre industrie, un pesant conformisme et un étouffant terrorisme intellectuel qui se révèlent lors de chaque débat, que ce soit la guerre des langages (Pascal contre C contre Basic), la guerre des machines (Mac contre PC) ou les guerres politiques (vous avez l'embarras du choix).

Dans ce panier de crabes, Jerry surprend agréablement, et j'ai moi-même été témoin d'un exemple de sa loyauté. La femme de Jerry, Roberta, avait en 1986 un projet de programme d'apprentissage de la lecture. Jerry avait passé un accord avec un bidouilleur sur ST, un nommé Alex Leavens, pour réaliser ce programme. Alex prit un bon départ, puis s'enlisa dans un énorme retard. Il avait de gros problèmes personnels qui lui prenaient presque tout son temps. Alex accumula des mois et des mois de retard, et j'étais quasiment certain que Jerry finirait par se lasser et embaucherait quelqu'un d'autre pour terminer le programme. Il était parfaitement en droit de faire, et cela aurait fait sortir le programme bien plus tôt. Mais à ma grande surprise, Pournelle ne le fit pas, et Alex parvint finalement à finir le projet. L'aide de Jerry lui fut précieuse en une période critique de sa vie, et c'est sans doute ce qui lui a permis de s'en tirer.

Autre exemple, Jerry fait partie de ces auteurs qui savent témoigner de leur reconnaissance à leurs assistants pour le gros travail de relecture qu'imposent leurs romans. Dans son dernier livre, la page de garde comporte ainsi des remerciements pour certaines de ces petites mains de l'édition, qu'ignorent superbement la plupart des écrivains. Après tout, cela fait partie du travail de ces gens, n est-ce pas...

Puisque nous parlions de Mme Roberta Pournelle, permettez-moi de vous la présenter. Roberta est une rousse exubérante (et qui ne s'offense pas d'être ainsi décrite, sans quoi je ne me le permettrais pas !) qui a à son actif une brillante carrière d'institutrice dans la région de Los Angeles. Les élèves qu'on lui envoie sont ceux que le système juge irrécupérables. Ils lui arrivent avec des kilos de paperasses expliquant pourquoi ils ne peuvent apprendre à lire "dyslexique", "enfant à problème", etc...

L'approche de Roberta est de mettre la paperasse au panier et d'apprendre à lire aux gosses. Elle a jusqu'à présent un taux de réussite de 100%. Cela demande beaucoup de patience et de dévouement, et c'est bien grâce à elle que des dizaines de gosses ne sont plus analphabètes.

## Un fan du st bon vivant

C'est cette technique d'enseignement éprouvée qu'Alex a transcrite sous forme de programme (qui est à présent disponible). Et devinez quel ordinateur a choisi pour implémenter ce programme ? Mais voyons, dites-vous, ça ne peut être qu'un Apple II, celui que les écoles choisissent toujours ! Perdu, Jerry et Roberta ont choisi le ST.

Le ST a toujours été un ordinateur dont le potentiel n'a jamais été vraiment exploité, mais ses capacités sont évidentes pour les connaisseurs. Il dispose de suffisamment de mémoire, de puissance de calcul, de résolution graphique pour faire du travail sérieux. Or, à l'époque, en 1988, la plupart des écoles étaient en effet équipées d'Apple II (la vague des PC bon marché est venue bien plus tard). Mais si le programme avait dû être réalisé sur Apple II, cela aurait exigé de le faire en mode monochrome (rares étaient les écoles dotées de moniteurs couleur), de le faire tenir dans les malheureux 48 Ko de la machine, et de se passer de souris. En bref, cela aurait sérieusement compromis les chances de succès du programme. C'est pourquoi les Pournelle ont choisi le ST. Et de plus, Jerry apprécie beaucoup cette machine.

Par ailleurs, Jerry est quelqu'un qui sait s'amuser et abandonner tout sérieux lors de certaines surprises-parties. Il vous faut savoir que la micro-informatique abonde en réception ennuyeuses, où

des journalistes s'entourent de petits groupes et pontifient interminablement. Peu importe le sujet sur lequel ils pérorent, que ce soit la vie secrète de Bill Gates ou les philosophies comparées du Mac et de l'OS/2, pourvu qu'ils satisfassent leur ego. Et la soirée se dilue dans des flots de paroles insipides.

Par contre, il est difficile de s'ennuyer dans une soirée où Jerry se déchaîne. "Comment, un Monsieur à l'air si sérieux ? Dave exagère", pensez-vous. Ah, j'exagère ? D'accord, prenons un exemple. Tenez, la fois où Philippe Kahn, patron de Borland, organisa à San Francisco une soirée "romaine" [NdT: où les invités portent des toges.] Celle-ci se tenait dans la cour intérieure d'un hôtel. Un ami et moi y fîmes un saut. Il y avait certes un orchestre, un bar somptueux, et des toges partout. Mais surtout, dans les galeries qui surplombaient la cour, des gens commençaient à remplir des préservatifs avec de la mousse à raser pour pouvoir les balancer sur la foule depuis le troisième étage "quand il commencerait à y avoir de l'ambiance".

J'avoue m'être dégonflé et avoir déguerpi. Mais Jerry, lui, resta, bien qu'il eut la Foire informatique à inaugurer le lendemain matin. Il acquit ce jour-là une certaine gloire car il parvint non seulement à être présentable pour le discours inaugural, mais en plus à se faire applaudir par 3000 personnes dans une salle bourrée à craquer, malgré une mémorable gueule de bois (qu'il soignait à la bière).

### Le manoir du chaos

Enfin, la maison de Jerry, qu'il appelle "le manoir du chaos", est vraiment surprenante. On a beau savoir qu'il s'intéresse à la stratégie et aux jeux de guerre, on reste quand même stupéfait en découvrant une demeure remplie d'armes moyenâgeuses authentiques. Je me souviens d'une soirée donnée lors de la convention HackerCon 2.0, où Jerry avait amené une hache de bataille à double tranchant, qui fut bientôt baptisée "double face, double densité!"

L'apothéose, c'est le bureau de Jerry, au fond duquel trône une cible sur laquelle il tire au pistolet à plombs. Quand un attaché de presse mal inspiré lui envoie un produit qu'il n'aime pas, il colle celui-ci sur la cible et tire dessus... pendant qu'il téléphone à l'attaché. J'ai d'ailleurs récupéré l'idée pour mon propre bureau, et j'aime me défouler ainsi après un appel téléphonique particulièrement pénible. Les disquettes 3,5 pouces font d'ailleurs d'excellentes cibles!

### Vaches sacrées

Voilà donc qui est pour moi Jerry Pournelle. Dans l'univers de la micro, on entend parfois des gens qui le condamnent. Mais le plus souvent, ce sont plutôt des gens creux, qui désapprouvent tous ceux qui ne partagent pas totalement leurs opinions. Ou bien ce sont des membres de quelque secte dont Jerry a malmené les vaches sacrées dans un article.

Son image publique assez rébarbative est nécessaire pour décourager les importuns, mais je puis le juger par ce qu'il a fait, et non par son apparence. Et je l'ai vu faire preuve d'une bonté qu'on ne s'attend certes pas à trouver sous une tenue camouflée ! Enfin, ce qui ne gâte rien, c'est un auteur à succès, et ses livres font la fortune des libraires. Comme exemple à suivre, croyez-moi, on peut trouver bien pire.

"So, what's Pournelle really like?"
Traduction et adaptation Password 90



# Tesla, le plus grand des bidouilleurs

Le nom de Nikola Tesla ne dit rien à la plupart d'entre nous. Tout au plus se rappelle-t-on vaguement qu'une unité de mesure de champ magnétique porte son nom. Et pourtant, Tesla était un être exceptionnel, un théoricien de génie doublé l'un ingénieur de grand talent, qui conçu et réalisa de ses mains les prototypes de machines qui allaient révolutionner le monde et que nous utilisons toujours. Dave Small nous dit ce mois-ci pourquoi il admire Tesla et le considère comme le plus grand des bidouilleurs.

### Nikola Tesla

Vous savez que je considère l'éthique du bidouilleur comme un exemple à suivre pour celui qui s'intéresse à la haute technologie. Pour moi, l'important est de garder la volonté d'innover, d'essayer l'impossible.

Il convient aussi de partager l'information autant que possible. Autant dire que la tâche est rude. On me demande parfois si j'ai des exemples de grands hommes ayant vécu suivant cette éthique. J'ai déjà cité des contemporains, des gens qu'il m'a été donné de côtoyer et qui m'ont beaucoup apporté. Mais si on me demande quel est le plus grand bidouilleur de tous les temps, je réponds sans hésiter "Nikola Tesla". A ce moment, le regard vitreux de mon interlocuteur m'apprend que ce nom lui est inconnu, et qu'il est temps de présenter cet archétype du bidouilleur.

Tesla est né en 1856 en Croatie. Très tôt, il a fait preuve d'un talent scientifique surprenant. Il a émigré vers la France en 1882, où il travailla pour la filiale française de la Compagnie du téléphone Edison. A Strasbourg, il mit au point en 1883 son premier moteur à courant alternatif. Mais la Compagnie le floua : une prime promise suite à un excellent travail ne vint jamais. Charles Batchelor, directeur de la filiale, l'encouragea à aller aux Etats-Unis en 1884. Et là, Tesla rencontra Edison, fréquenta le gratin, et surtout, il mit au point le courant alternatif (CA), et nombre de machines importantes qui l'utilisent, comme le moteur électrique à induction, cœur des innombrables appareils électroménagers que nous utilisons quotidiennement.

## **Tesla contre Edison**

Mais ne croyez pas qu'il ait suffi à Tesla de montrer les plans de son moteur pour que celui-ci soit immédiatement adopté dans l'enthousiasme général. Bien au contraire : Edison, le grand Edison, combattit Tesla et son courant alternatif de toutes ses forces. Au temps pour la réputation de précurseur et d , innovateur d'Edison.

Il faut dire que quand Tesla débarqua à New York, Edison avait déjà installé toute une industrie basée sur le courant continu (CC). L'éclairage, les moteurs CC, tout était basé sur les brevets d'Edison. Lequel avait bien des soucis. Ses installations étaient peu fiables, et les moteurs à CC tombaient souvent en panne, notamment à cause de l'usure de leurs contacteurs rotatifs (les balais). Mais l'électrotechnique était alors la science de l'avenir, la haute technologie, un nouvel Eldorado encore inexploré vers lequel se précipitaient tous ceux qui étaient dotés d'un esprit inventif. Tesla se présenta à Edison muni d'une lettre de recommandation de Batchelor. Cette lettre disait : "Je connais deux grands hommes, vous êtes l'un d'eux; l'autre est ce jeune homme!" Mais dès que Tesla voulu parler à Edison des bienfaits du courant alternatif, celui-ci le rabroua et l'envoya travailler sur ses installations à CC.

Quelques temps après, Tesla, fin théoricien, s'aperçut qu'on pouvait augmenter le rendement des

rustiques dynamos d'Edison, qui produisaient le courant de New York. Edison lui promit 50 000 dollars si Tesla y parvenait. Tesla s'y attela, sua sang et eau pendant des mois, et finalement, perfectionna les installations, les rendant plus rentables et plus fiables. Fièrement, il vint réclamer sa prime à Edison, qui s'était bien gardé de faire sa promesse par écrit. Edison lui répondit froidement : "Tesla, vous n'avez décidément rien compris à l'humour américain!".

Furieux, Tesla démissionna. L'aveuglement d'Edison lui coûta ce jour-là des milliards de dollars... Tesla créa la Tesla Electric Light Company, concurrente déclarée de celle d'Edison. Puis il inventa une nouvelle lampe à arc, première étape des bienfaits qu'il pensait pouvoir désormais déverser sur l'humanité reconnaissante. Mais bientôt, les actionnaires le licencièrent de sa propre société! Or, l'Amérique était en pleine crise suite au krach bancaire de 1884, et Tesla survécu péniblement grâce à des emplois précaires. Mais la chance lui sourit : le contremaître de l'équipe de terrassiers où travaillait Tesla connaissait quelqu'un à la Western Union Telegraph Company (grande rivale l'Edison), qui le fit embaucher en 1887. En quelques mois, Tesla mit au point et breveta de nombreux appareils à courant alternatif utilisant une, deux ou trois phases (notre courant moderne est le triphasé). Immédiatement, le magnat George Westinghouse vit l'importance de l'alternatif de Tesla et acheta ses brevets.

# Les avantages du courant alternatif

Car il faut savoir que le courant continu était une impasse du point de vue de la production industrielle. Les dynamos à CC d'Edison produisaient une tension assez faible, ce qui impliquait que l'intensité devait être importante. D'où de gros câbles et des pertes importantes (la résistance des fils convertit en chaleur une partie du courant, pertes qui croissent comme le carré de l'intensité). Mais l'alternatif, lui, permet de concevoir le transformateur, qui élève et abaisse la tension. Grâce à un transformateur, on peut faire passer des milliers de volts dans un fil, avec un faible ampérage (et donc de faibles pertes). Donc on peut distribuer le courant sur de grandes distances, et on n'est plus obligé d'habiter à côté d'une centrale pour avoir l'électricité. C'est la base de notre système électrique! De plus, le moteur "synchrone", à courant alternatif, est plus économique et plus fiable que le moteur à CC et ses fragiles balais.

Naturellement, Edison et sa General Electric ne se laissèrent pas faire. Ils capturaient des animaux pour les électrocuter en public avec du courant alternatif afin de "prouver" les danger de cette technologie. Ils s'arrangèrent pour faire installer à la prison de Sing-Sing la première chaise électrique qui fonctionnait au courant alternatif. Un meurtrier, un certain Kemmler, fut le premier condamné électrocuté, en 1890. La tension était trop basse, et l'on dut s'y reprendre à plusieurs reprises. Edison en tira une immonde propagande, baptisant ce procédé d'exécution la "westinghousation", et affirmant que l'électrocution guettait inévitablement tous ses utilisateurs. (En réalité, à tension donnée, le CC et le CA de basse fréquence, comme notre 50 Hz - ou le 60 Hz des USA et du Japon - sont tous deux également dangereux.)

Tesla répliqua en acceptant de donner une série de conférences sur le courant alternatif aux Etats-Unis et en Europe. Ces conférences eurent une succès incroyable. Tesla avait des talents d'orateur. Sur son estrade, il y avait des bobinages, des lampes à incandescences, et surtout, d'étonnants tubes de verre emplis de gaz à très basse pression. Tesla saisissait d'une main un fil conducteur provenant d'une de ses bobines, et où circulait un courant alternatif à haute tension. De l'autre main, il prenait un tube et celui-ci s'illuminait, à la stupéfaction de la salle!

Le secret? Tesla employait un courant à très haute fréquence. Par "effet de peau", celui-ci ne pénètre pas dans les conducteurs, comme le corps humain, mais circule à leur périphérie. De grâce, n'allez pas mettre vos doigts dans les prises électriques, où la fréquence est trop basse pour engendrer un tel effet! C'était de la triche, car Tesla "prouvait" l'innocuité du courant alternatif. Edison sentit le vent tourner, mais refusa pendant vingt ans d'admettre la supériorité de l'alternatif. Certes, Edison était un personnage très populaire, et Tesla n'était qu'un immigrant inconnu, mais finalement, le panache de Tesla conquit les foules. Quant à ces tubes, ancêtres des tubes fluorescents modernes, Tesla ne les breveta ni ne les commercialisa jamais, et il ne furent redécouverts que cinquante ans plus tard.

J'ai moi-même refait les expériences de Tesla en public, lors du salon Atari de Washington. J'ai utilisé

une bobine de Tesla, un transformateur résonnant à haute fréquence et haute tension, et devant un public enthousiaste, j'ai fait jaillir de mon doigt des éclairs de 15 cm sans ressentir la moindre décharge (le courant à haute fréquence ne pénètre pas dans le corps). Une chaîne de quatre personnes se tenant par la main illuminait des tubes fluorescents simplement en les prenant en main, sans le moindre fil. Ce fut la présentation la plus populaire que je fis jamais!

### Le cadeau

Cependant, George Westinghouse se battait financièrement contre le banquier Morgan et la General Electric. Les banquiers de Westinghouse, horrifiés, constatèrent que celui-ci avait signé en 1888 un contrat avec Tesla : ce dernier recevait 2,50 dollars de redevances pour chaque cheval-vapeur (736 W) de puissance électrique vendu. Les droits accumulés impayés dépassaient à présent 12 millions de dollars, et représentaient une charge insupportable pour la firme de Westinghouse. Les banquiers de ce dernier lui conseillèrent de se débarrasser de ce contrat. Westinghouse expliqua la situation à Tesla. L'inventeur écouta gravement, sortit son contrat de son coffre, puis répondit quelque chose comme : "M. Westinghouse, vous avez été mon ami, vous seul avez cru en moi... Faites profiter le monde de mon courant alternatif." Et il déchira le contrat. En 1897, la compagnie Westhinghouse versa à Tesla 216 000 dollars pour l'achat complet de tous ses droits, ce qui est une somme ridicule par rapport à ce qu'il aurait pu exiger. Les équipements électriques actuels représentent des centaines de milliards de dollars, et Tesla aurait pu légitimement demander un pourcentage...

C'est grâce à ce geste, d'une folle générosité, que la compagnie Westinghouse survécut et parvint à imposer le courant alternatif et toutes les technologies dérivées. Mais Tesla connut la gêne financière plus tard. Il dut renoncer à poursuivre de nombreux travaux par manque d'argent, et le monde y perdit sans doute maintes inventions. Tesla gagna donc, mais au prix d'un terrible sacrifice. Il aurait pu négocier avec Edison, mais il avait le sens de l'honneur.

En fait, Tesla a pratiquement fait don à l'humanité du courant alternatif, et celle-ci, ingrate, l'a oublié! Quant à Edison, sa General Electric adopta bien plus tard l'alternatif et survécut. Il est aujourd'hui considéré comme un héros, et on lui a même consacré des films... (A vrai dire, il y en a aussi eu un sur Tesla, mais je l'ai hélas trouvé assez décevant.) Je vous encourage à lire les livres qui retracent la vie de Tesla [NdT: voir bibliographie en fin d'article], car je ne fais qu'effleurer un sujet passionnant.

Voilà pourquoi, à mes yeux, Nikola Tesla est l'un des personnages-clés méconnus les plus marquants de l'histoire. Tesla n'avait cependant pas que des qualités. Il était doué d'une grande acuité auditive : des bruits forts mais tolérables pour le commun des mortels lui étaient insupportable. Il pouvait aussi littéralement visualiser en trois dimensions un appareil avant de le construire, dans tous ses détails, et méprisait superbement la planche à dessin, au grand dam de ses collaborateurs. Les ingénieurs qui travaillaient sous ses ordres avaient à subir ses sautes d'humeurs lorsqu'ils ne comprenaient pas assez vite. Il préférait d'ailleurs travailler seul. Il était narcissique, maniaque de propreté, et célibataire endurci - bien qu'il plût beaucoup aux femmes.

### Quelques inventions mineures...

Tesla ne s'est d'ailleurs pas arrêté là. Il avait cerné la nature de l'électricité et ses liens avec le magnétisme, et explorait le champ alors inconnu des hautes fréquences. Parmi ses applications, la radio, où TSF, comme on disait alors. Vous ai-je dis que Tesla avait inventé la radio? La plupart des gens croient que c'est l'œuvre de Marconi, mais en 1943, un arrêt de la Cour Suprême a attribué à Tesla l'antériorité des travaux. Cette même année, Tesla était mort dans son sommeil, le 7 janvier, à l'âge de 86 ans. Ce qui en dit long sur la rapidité de la justice. [NdT: chaque nation revendique néanmoins son inventeur de la radio, Herz en Allemagne, Branly en France, Marconi en Italie...] Tesla fit en 1893 une démonstration public de transmission d'onde radio au moyen d'un éclateur, deux ans

avant les premiers essais de Marconi. Quant à ce dernier, Tesla déclara une fois : "Marconi est un brave garçon. Il est bien parti. Il utilise dix-sept de mes brevets."

Au début des années 1900, Tesla avait fait la démonstration qu'on pouvait contrôler des navires par radio - l'ancêtre du téléguidage. Et ce, uniquement avec des moyens électrotechnique, puisque le tube à vide électronique n'existait pas encore.

De 1901 à 1903, Tesla fit construire une tour octogonale à Wardenclyffe. Cette tour entièrement en poutrelles métalliques était destinée à des expériences de transmission d'énergie à distance. Le rêve de Tesla était qu'il suffise de planter un poteau métallique dans le sol pour recevoir du courant, en utilisant le sol comme conducteur. Pour cela, il se proposait de créer un champ électrique alternatif à la fréquence de résonance du globe terrestre (qui est de quelques Herz). Mais des problèmes financiers l'empêchèrent d'achever sa tour. Aujourd'hui, une telle installation serait un désastre total pour tous les appareils électroniques, puisqu'elle provoquerait des courants induits dans leurs prises de terre. D'ailleurs, des rumeurs circulaient jadis sur de mystérieuses expériences soviétique en Sibérie, où une série de tours de ce genre semblaient parfois émettre de puissants champs électriques de très basse fréquence...

Tesla inventa aussi la porte ET. Il avait présenté ses bateaux radioguidés à la Navy dans le but d'en faire des torpilleurs. Pour se protéger d'éventuels brouillages, ses circuits accordés ne réagissaient qu'à l'émission simultanée de plusieurs fréquences. Mais la marine américaine trouva le concept trop audacieux. Tesla, idéaliste, voulait créer une arme si efficace qu'elle rendrait toute guerre impossible.

Il faut noter que maintes fois, dans la seconde moitié de ce siècle, des inventeurs voulant breveter des circuits d'ordinateurs sont tombés sur des brevets de Tesla. Le concept de porte logique est l'une des bases de la cybernétique, et bien que Tesla ne puisse être considéré comme un de ses pères, l'informatique moderne a redécouvert des idées qu'il avait déjà brevetées.

Naturellement, Tesla avait besoin de manipuler des tensions très hautes, et devait isoler soigneusement ses conducteurs. Il inventa donc le conducteur à paire torsadée et l'isolation des appareils dans un bain d'huile, méthodes aujourd'hui universellement utilisées.

Au passage, il constata que les courant à haute fréquence pouvaient provoquer un réchauffement interne lorsqu'ils traversaient le corps humain. Il expérimenta en 1890 les effets thérapeutiques de ce réchauffement, connu alors sous le nom de diathermie. On l'a récemment remis à l'honneur en cancérologie.

Par ailleurs, il inventa des tubes à très haute tension qui avaient l'étrange propriété de voiler des plaques photographiques, et émettaient un rayonnement inconnu qui provoquait la phosphorescence de certains corps. Peu après, en 1895, Röntgen découvrit les rayons X. Tesla fut aussitôt capable de reproduire les expériences du savant allemand : il n'avait pas réellement envisagé l'existence de ce rayonnement, mais il l'avait produit au cours de ses recherches. Tesla se soumit en toute inconscience à des doses massives de rayons X, particulièrement en prenant des radiographies de sa boîte crânienne et de celle de son ami Mark Twain, constatant même un échauffement de sa tête! En radiographiant ses mains, il contracta une splendide dermite avec cloques et pigmentation. Edison, également engagé dans ce type de recherches, qui fascinait alors le monde, s'abîma un œil, et l'un de ses assistants mourut d'un cancer de la peau. C'est pourquoi Tesla testa différents matériaux de protection, et recommanda de se protéger par des écrans en plomb.

Ajoutons que dès 1921, le génie touche-à-tout déposa le brevet d'un curieux appareil à hélice à atterrissage et décollage verticaux, qu'il nommait son "fourneau volant". Le concept réapparut dans les années 50 mais fut abandonné car les atterrissages étaient trop risqués. Le Harrier et le V-22 Osprey prouvent aujourd'hui que le concept de VTOL (Vertical Take-Off and Landing) est viable.

### Le savant-fou

Si Tesla avait tant rompu avec son époque, c'est qu'il maîtrisait la notion de dynamique, de vibrations, dans un univers où la mentalité des physiciens restait essentiellement tournée vers le statique, le

continu. Tesla s'intéressa ainsi aux phénomènes de résonance, encore inexplorés. Pour les étudier, il avait mis au point de petits vibrateurs électromécaniques d'une puissance étonnante.

Un jour, en 1898, il attacha un petit oscillateur au pilier de fonte central soutenant l'immeuble où se trouvait son laboratoire new-yorkais. Tesla observa la mise en résonance successive de tous les objets de la pièce, un à un, au fur et à mesure que la fréquence augmentait. Mais peu à peu, à l'insu du savant, la vibration de très basse fréquence s'était communiquée au pilier, puis au sous- sol et aux immeubles avoisinants. Ceux-ci se mirent à trembler, des vitres explosèrent, les habitants affolés se ruaient hors des immeubles.

Au commissariat voisin, on ne tarda pas à constater que ce curieux tremblement de terre n'affectait pas les autres quartiers de la ville. Et comme Tesla avait déjà une solide réputation de savant fou, deux agents furent envoyés vérifier s'il n'était pas responsable de ce mini-séisme. Curieusement, l'immeuble de Tesla tremblait moins que les autres, et le savant commençait tout juste à ressentir une forte vibration. Inquiet, il détruisit l'oscillateur d'un coup de marteau, alors même que les policiers faisaient intrusion dans son laboratoire! Le savant éconduisit les agents, et quand les journalistes arrivèrent, Tesla leur déclara pouvoir ainsi détruire le pont de Brooklyn ou l'Empire State Building, ce qui fit leurs choux gras et n'arrangea pas la réputation de l'inventeur! Mais Tesla, il est vrai, ne recherchait pas précisément la discrétion, loin s'en faut.

# Le Jupiter tonnant du Colorado

Les expériences de Tesla devenaient trop dangereuses pour être poursuivies une grande ville. Ses bobines produisaient des tensions de plusieurs millions de volts, projetant d'énormes arcs longs de plusieurs mètres. Il demanda à son avocat de lui trouver un laboratoire plus isolé. Celui-ci était actionnaire de la compagnie d'électricité de la ville de Colorado Spring, et lui proposa donc d'aller s'installer là-bas, sa compagnie lui fournissant gratuitement l'électricité.

L'inventeur déménagea donc, et s'installa dans une bâtisse carrée en bois, sur le toit de laquelle se dressait un mât métallique de 37 mètres de haut surmonté d'une sphère de cuivre. Le mât était connecté à un puissant oscillateur à très haute tension et à haute fréquence, avec lequel l'inventeur pouvait simuler des orages : quand son appareil était en marche, des feux de Saint-Elme apparaissaient sur les paratonnerres à 30 km à la ronde, et le tonnerre des arcs électriques s'entendait à la même distance. A cent mètres du laboratoire, des étincelles de 3 cm de long jaillissaient d'objets métalliques reliés au sol. Des chevaux broutant à 500 mètres de là recevaient des décharges électriques dans leurs fers et devenaient furieux!

Au cours d'une expérience mémorable, il tenta d'émettre une onde électrique qui ferait entrer la terre en résonance. Les tensions et les courants nécessaires étaient tous deux très élevés. Tesla faisait ses expériences de nuit, à l'heure où la demande de courant était la plus faible. Lorsque son assistant abaissa e levier, une forte odeur d'ozone se répandit, et des éclairs de 40 mètres de haut jaillirent du mât! Mais soudain, tout s'arrêta : plus de courant. Furieux, Tesla appela la compagnie d'électricité, mais s'entendit répondre que leur générateur avait été surchargé et avait pris feu... Colorado Springs était plongée dans l'obscurité! Tesla dut accepter de faire réparer à ses frais le générateur avant que la compagnie accepte à nouveau de lui fournir du courant.

## Le prince des bidouilleurs

Toute sa vie, Tesla a donc innové, surpris le public (et ses financiers!), et ne s'est jamais intéressé à l'argent autrement que comme une ressource indispensable à ses expériences. Il écrivit de nombreux articles, multiplia les conférences, et fut un excellent vulgarisateur, exposant en termes clairs et simples les principes de base de l'électricité à un public ravi. Sans compter, bien sûr, son fameux don du courant alternatif au monde... On peut donc vraiment dire qu'il vécut selon l'éthique du bidouilleur,

même s'il est douteux qu'il eut apprécié ce qualificatif! Il y a un musée Tesla à Belgrade, et mon vœu le plus cher est de pouvoir un jour le visiter.

Bibliographie : "Tesla, la passion d'inventer", par Margaret Cheney, éd. Belin (1987)

**Traduction et adaptation : Password 90** 

# Des ordinateurs qui doutent

Les performances des ordinateurs augmentent sans arrêt. Les constructeurs rivalisent de prouesses, et battent régulièrement leurs propres records de vitesse et de taille. Mais leur ingéniosité est-elle canalisée dans la bonne direction ? D'abord, les limites physiques de la technologie commencent à se faire sentir. Ensuite, à quoi bon ces performances étourdissantes, Si c'est pour s'en tenir aux mêmes approches depuis des lustres ? Il est temps d'inventer une nouvelle façon de programmer. Dave Small en a imaginé une. Voulez-vous l'implémenter?

## La nouvelle génération

Attention : je me mets en roue libre pour cet article. Le lecteur attentif y décèlera d'évidents manquements à la logique. Mais après tout, je suis doté d'une personnalité de type NF, type pour lequel, je cite les psychologues, "la logique est optionnelle".

C'est pourquoi je vais discuter d'un sujet a priori assez abstrus : comment la prochaine génération d'ordinateur devrait-elle être construite ? Prenons la peine de nous y pencher. Pour me faire pardonner de vous soumettre mes idées bizarres, je vais commencer par brosser le tableau de la situation actuelle.

### **Causons matos**

À bien y regarder, la technologie informatique semble avoir abouti dans une impasse. Oh, bien sûr, on peut continuer à ajouter de la mémoire, à augmenter les vitesses d'horloge, et à entasser des flopées de transistors dans des circuits minuscules. Mais nous nous heurtons aux lois de la physique. Les circuits intégrés deviennent de plus en plus complexes à fabriquer, et les usines de production nécessitent des investissements de plus en plus colossaux, avec une rentabilité décroissante. Examinons le marché actuel : déjà, la mémoire est suffisamment bon marché pour que des appareils grand public en comportent plusieurs mégaoctets. Les vitesses des processeurs sont désormais ahurissantes. On peut aujourd'hui se procurer un 68040 à 66 MHz (mégahertz, millions de cycles par seconde), avec une horloge à 33 MHz, mais comportant l'équivalent d'un doubleur de fréquence d'horloge. Ou bien un 80486 DX-2 à 66 MHz. Les premiers exemplaires du nouveau processeur de chez Intel, le 80586 rebaptisé Pentium, sont disponibles au prix de 900 dollars pièce (en faibles quantités, mais le prix descendra en flèche sous quelques mois), et les premiers tests indiquent qu'il est beaucoup plus rapide que le 80486 à 66 MHz. Les fidèles de Motorola ne doivent pas désespérer, car un 68060 est en préparation.

Mais Si vous voulez vraiment parler vitesse, jetez donc un œil à l'architecture RISC (Reduced Instruction Set Computer, ordinateur à jeu d'instructions réduit). Il s'agit de processeurs comportant moins d'instructions que les processeurs classiques, dits CISC (Complex Instruction Set Computer, ordinateur à jeu d'instructions complexe), mais capables d'exécuter chaque instruction de manière optimale, très rapidement. Motorola, justement, est en train de révolutionner le marché en proposant un processeur RISC bon marché et très rapide conçu en collaboration avec IBM, le PowerpC.

A la convention HackerCon, qui est une réunion annuelle de bidouilleurs, j'ai pu voir une machine RISÇ, une Silicon Graphics, laisser sur place un Mac ultra-rapide, un Quadra 950. C'est la machine la plus rapide que j'aie jamais vue, à l'exception d'une machine expérimentale basée sur des composants optiques utilisant des lasers au lieu de signaux électriques, et qui opère à des fréquences de plusieurs

milliards de mégahertz (ce n'est pas une coquille !), mais qui est encore quelque peu éloignée du stade commercial.

Peut-être vous rappelez-vous les processeurs 6502 des Atari 8 bits et des Apple II. Le 6502 est très proche d'un processeur RISC : il a peu d'instructions, mais grâce à un pipe-line, il exécute une instruction pendant qu'il décode la suivante, ce qui lui permet d'exécuter pratiquement une instruction par cycle d'horloge, au lieu des multiples cycles par instruction que demandent un 68000 et un 80486. Et vous souvenez-vous de ce qu'on pouvait faire avec un petit 6502 à 1,79 MHz seulement ? Le jeu "Star Raiders", par exemple. Si ses graphiques élaborés étaient possibles, c'est parce qu'on tirait le maximum de chacun des 1 790 000 cycles d'horloge par seconde.

Au fait, le 6502 n'est pas mort. De nos jours, on continue à produire des 6502, cadencés à (tenezvous bien) 16 MHz, soit 9 fois plus vite que celui des Atari 8 bits, et aussi rapide que le 68000 des Méga/STE. Pas mal pour un processeur "obsolète", non ? Par exemple, nos deux Mac IIfx ont chacun deux 6502 dédiés à la gestion des entrées/sorties série et SCSI, pour soulager le 68030. Un fabricant de modems à 14 400 bauds utilise des 6502 rapides de Rockwell, qui fournissent la puissance de calcul nécessaire au traitement des algorithmes complexes du mode télécopieur et de la compression au vol. Or, l'architecture du 6502 est très loin d'être aussi optimisée que celle d'un véritable processeur à jeu d'instructions réduit. C'est dire quelle puissance recèle l'architecture RISC et les améliorations de performances qu'on peut en attendre encore.

(Petit aparté: étant nostalgique de l'Atari 8 bits, j'ai une requête à formuler. Je souhaiterais qu'un bon bidouilleur mette un 6502 à 16 MHz dans un Atari 8 bits, avec sa propre mémoire de façon à ne pas interférer avec le balayage vidéo. L'intervalle d'adresse à utiliser pour cette RAM devrait logiquement être le même que celui des ROM, avec une sélection de page, comme sur le 130XE. Le balayage vidéo de la RAM n'a de toute façon rien à faire dans cette zone d'adresse, vu que personne ne s'amuserait à y mettre une image. Je sais, ce n'est pas trivial, mais je sais que c'est faisable. Si quelqu'un y parvient et que le montage est raisonnablement rapide, qu'il me le fasse savoir : je lui offre un Spectre GCR tout neuf, à la condition qu'il explique comment il a fait, selon l'éthique de la bidouille. J'irais même jusqu'à rédiger la documentation moi-même si c'est nécessaire. Inutile que ce soit bien soigné, j'improvise moi-même souvent des bricolages assez sordides. Je le ferais volontiers moi-même si j'avais le temps...)

### Les pièges de la haute vitesse

On continue donc à augmenter les cadences d'horloge. Mais on ne pourra pas continuer indéfiniment. Déjà, des barrières insurmontables apparaissent. En particulier, on se heurte à la vitesse de la lumière. Selon Einstein, rien ne peut dépasser 299 792 km/s, et ce postulat n'a jamais été contredit. (Beaucoup de choses confirment sa théorie. Des particules subatomiques qui ne devraient exister qu'une infime fraction de seconde durent beaucoup plus longtemps si elles approchent la vitesse de la lumière, car le temps ralentit à ces vitesses. La physique quantique recèle d'intéressants paradoxes, comme celui des deux photons corrélés - émis par la même source - dont chacun réagit instantanément à une modification subie par l'autre, mais même ces expériences ne contredisent pas l'existence de cette limite absolue.)

Donc, les informations qui circulent dans nos ordinateurs ne peuvent dépasser la vitesse de la lumière, notée c. Et encore dans un câble coaxial, les signaux électriques atteignent bien cette vitesse, mais sur un circuit imprimé, les électrons n'en dépassent pas le tiers. Or, les intervalles de temps dans les ordinateurs sont à présent mesurés en nanosecondes (ns), ou milliardièmes de seconde. Et en une nanoseconde, la lumière ne parcourt que 30 centimètres. Et il n'est pas rare de voir des circuits logiques demander 15 ns uniquement pour qu'un signal les traverse - ce qu'on appelle le délai de propagation. C'est par exemple le cas du circuit logique principal de la carte 68030 SST. C'est pour cela qu'à haute fréquence, une bonne synchronisation est absolument cruciale.

Des choses étranges se produisent à haute fréquence. Les électrons ne peuvent se déplacer assez vite. Des signaux bien propres émis simultanément à un bout d'une carte arrivent déformés et décalés à l'autre bout. Un nombre affolant de concepteurs de circuits électroniques ont tenté de créer des ordinateurs fonctionnant à 50 MHz et ont échoué à cause de ce genre de pièges de synchro.

D'innombrables ingénieurs se sont arraché les cheveux sur ces problèmes, faisant la fortune des marchands de lotion capillaire. Il faut recourir à des techniques spéciales de construction de circuit imprimé pour éviter l'apparition de "zones d'ombre" sur les cartes, c'est-à-dire d'endroit où la tension d'alimentation tombe en dessous de la valeur normale, et où la masse ne se maintient pas à zéro volt.

Prenons un peu de recul et examinons ces phénomènes. Les électrons ont beau se déplacer à 100 000 km/s dans des pistes de cuivre très conductrices, ils ne peuvent arriver à rassasier assez vite les circuits intégrés. Les puces pompent le courant si vite (et le recrachent, bien sûr) que la tension d'alimentation à leurs bornes chute parfois pendant quelques fractions de seconde, bien que le cuivre soit à la bonne tension sur les connecteurs d'alimentation ! Et le fil de la puce relié à la masse, au lieu d'être en permanence à une tension nulle, monte à une tension non négligeable jusqu'à ce que les charges aient eu le temps de se répartir. Le problème est que les intervalles de temps sont si courts que même la vitesse phénoménale de l'électricité est insuffisante, et que des phénomènes négligeables à plus basse fréquence - comme les capacités parasites des connexions - deviennent prépondérants.

Pour pallier ces problèmes, il faut monter les puces sur des circuits imprimés multicouches, avec, par exemple, deux couches externes où circulent les signaux, et deux couches internes d'alimentation à O V et +5 V respectivement, alimentées par de très larges connexions. Les couches d'alimentations apportent le courant en tous les points de la carte, et la quadrillent complètement, sauf aux endroits où les pattes des circuits imprimés la traversent, et là où un trou métallisé fait passer un signal d'une face à l'autre de la carte. C'est ainsi qu'est faite la carte SST.

Anecdote significative: à Gadgets By Small, nous arrivons à faire monter à 46 MHz les SST normalement prévues pour 33 MHz, mais seulement en refroidissant énergiquement les circuits intégrés. (Par contre, je ne suis pas encore fixé quant à la vitesse maximum de ma Camaro, dont les limitations restent à atteindre). J'ai un jour assemblé et soudé à la main une carte SST (je suis devenu très fort en soudage). Elle marchait très bien à 33 MHz, mais refusait obstinément de fonctionner à 46 MHz. Nous avons retiré de cette carte les circuits intégrés montés sur support et les avons placés sur une SST soudée à la vague. (Le soudage à la vague consiste à mettre le côté soudure de la carte juste au dessus d'un bain d'étain en fusion ou un vibreur crée des vaguelettes qui vont venir lécher le cuivre nu et déposer de la soudure aux endroits voulus, de façon très uni forme et précise. C'est la technique de soudage ordinaire de cartes complexes comme la SST). Eh bien, sur cette carte soudée à la vague, les composants tournaient parfaitement à 46 MHz. Il y avait apparemment dans mes soudures manuelles quelque chose qui coinçait à cette fréquence, mais pas à 33 MHz. La limite est diablement fine!

On comprend dès lors pourquoi les compatibles PC ultra-rapides à base de 80486DX-2 tournent en réalité à la moitié de la fréquence du processeur. Ainsi, dans mon PC DX-2 à 50 MHz, seuls le processeur et l'antémémoire statique tournent à cette cadence. Le reste est en réalité une machine à 25 MHz, ce qui est beaucoup plus simple à concevoir et à faire certifier conforme par la FCC. [NdT: Commission fédérale des communications, qui exigent que les appareils aient des taux d'émission de parasites très bas. Pour les circuits numériques, ce taux augmente avec la fréquence, ce qui oblige à blinder les câbles et boîtiers, et complique la conception.] Il est en fait assez difficile de trouver des PC tournant entièrement à 50 MHz. De même, seul le processeur de l'Atari TT tourne à 32 Mhz, le reste de la machine est à 16 Mhz.

### Gardons la tête froide

Les ingénieurs se font des cheveux blancs avec l'arrivée des processeurs à plus de 60 Mhz, comme le Pentium. Les circuits imprimés devront être physiquement plus petits et plus denses, juste pour que les électrons n'aient pas trop de chemin à parcourir. Cela signifie que davantage de chaleur sera engendrée dans un espace plus petit. Or, les puces craignent énormément la chaleur, et grillent si elles chauffent trop.

D'ailleurs, dès ce soir, j'installe un refroidisseur dans mon PC, ainsi qu'un ventilateur supplémentaire. Le refroidisseur utilise une jonction de Peltier. C'est un étrange petit carré de métal de 3 cm de côté, avec un connecteur d'alimentation que l'on branche sur celle du disque dur. Cela fonctionne comme un thermocouple inversé. Un thermocouple fournit de l'électricité quand on le soumet à une différence de température, et une jonction de Peltier, elle, crée une différence de température quand on l'alimente en électricité. On la branche, et hop! Un côté devient glacé, l'autre devient brûlant.

On place donc le côté froid contre le circuit intégré à refroidir, avec un peu de graisse au silicone pour mieux conduire la chaleur, et le carré magique transfère vers son côté chaud la chaleur engendrée par la puce. Celle-ci est actuellement brûlante au toucher quand l'ordinateur marche, et le refroidisseur la ramène au-dessous de la température ambiante. Le ventilateur supplémentaire sert à évacuer la chaleur transférée par le refroidisseur.

Si vous avez un PC équipé d'un DX-2, achetez au minimum un ventilateur supplémentaire pour le processeur, ou mieux, un refroidisseur à jonction de Peltier. Selon les techniciens d'un magasin de PC d'à côté, les DX-2 à 66 MHz dégagent tant de chaleur qu'ils grillent fréquemment, même avec les gros radiateurs dont ils sont dotés d'origine. Sans radiateur, un DX-2 grille en quelques secondes.

Bien sûr, si ces circuits intégrés chauffent, c'est à cause du nombre toujours plus élevé de transistors qu'ils contiennent dans un volume à peu prés constant. Vous avez sûrement déjà vu ces diagrammes où l'on représente le nombre de composants par circuit en fonction du temps, ce nombre doublant à peu près tous les deux ans. La raison en est simple : comme la vitesse de la lumière est une limite absolue, il faut rapprocher les composants toujours davantage.

C'est également ce qui a conduit les concepteurs de cartes à dédaigner les circuits intégrés logiques standards (les circuits classiques série 74xx contenant chacun 4 ou 6 portes logiques ET, OU, NON-ET, etc.) pour les remplacer par des "circuits logiques programmables". Ce sont des circuits contenant des portes logiques en vrac, et que l'on connecte les unes aux autres en fonction des besoins, en usine, afin de remplacer de nombreux 74xx. Certes, ces circuits logiques programmables chauffent, mais tiennent cent fois moins de place sur les cartes que les CI logiques standards qu'ils remplacent. Les puces GLUE et MCU de nos Atari ST en sont de bons exemples. Cette technologie, très largement employée, permet de condenser en une seule puce des cartes grand format (60 x 90 cm) bourrées de circuits 74xx. Mais la consommation de cette seule puce est à l'avenant du nombre de circuits élémentaires qu'elle contient.

### La mémoire

La vitesse et le prix de la mémoire, qui ont longtemps été les facteurs limitant traditionnels de l'informatique, s'améliorent rapidement. Gadgets By Small livre beaucoup de cartes SST équipées de mémoires haute performance à 70 ns. Apple et les constructeurs de compatibles PC recommandent à présent de mettre des mémoires à 70, voire à 60 ns sur leurs machines les plus rapides. Les ST utilisent le la RAM à 120 ns. J'aurais honte de vous dire quelle était la vitesse des RAM équipant les Atari 8 bits... Quant à la taille de ces mémoires, on trouve des clones PC dotés de 8 Mo (mégaoctets) sur tous les bureaux. Et souvent, cette mémoire est constituée de 8 barrettes SIMM d'un Mo chacune. Avec l'arrivée des SIMM de 4 Mo, on peut facilement transformer ces PC en machines de 32 Mo. C'est une énorme capacité mémoire. Et les SIMM de 16 Mo débarquent, Si votre compte bancaire est bien approvisionne...

(Tuyau pour les possesseurs de TT GESoft fabrique une carte d'extension RAM doté de 8 emplacements SIMM qui acceptent les SIMM d'1 Mo bon marché en mode page, ainsi que les SIMM 4 Mo. J'ai un TT doté d'une carte GESoft et de 8 Mo de RAM-TT, et elle n'a jamais failli. Le coût des RAM en mode page est très bas comparé à la carte de 16 Mo de RAM en mode "nybble" vendue par Atari, pour une faible différence en vitesse.)

### Les limites du matériel

Soupir nostalgique... Je me rappelle qu'à l'époque où je suis entré dans le monde Atari, en 1980, une carte mémoire de 16 Ko (16 384 octets très exactement) coûtait 200 dollars. Aujourd'hui, nous semblons être entré dans une spirale ascendante de vitesse et de densité. Mais là encore surgissent des problèmes sournois qui nous rappellent brutalement qu'à ces vitesses, nous jonglons sur une corde raide.

Un petit exemple : je suis parvenu à faire fonctionner sur la SST des SIMM d'un mégaoctet à 53 ns, puis à 40 ns. Mais je ne parviens pas encore à les faire aller à leur pleine vitesse un faible nombre de cycles d'attentes se produisent encore. (Un cycle d'attente survient lorsque la RAM ne parvient pas à suivre la cadence du processeur. Un cycle d'horloge est alors gaspillé en attendant que la RAM le rattrape.) Les RAM se comportent normalement à 70 ns, mais à 40, un problème d'adaptation d'impédance survient. Les amplis de ligne 74A5245 deviennent brûlant, ce qui indique un rebond lors des transitions de signal, donc un problème d'impédance.

Ce qui nous mène au problème des instruments de mesure capables d'indiquer ce qui se produit vraiment sur des cartes à haute vitesse. "Déverminer" des cartes, comme on dit, n'est possible que si l'on dispose d'instruments qui ne faussent pas les mesures. Et même avec des instruments ruineux adaptés aux hautes vitesses, faire une mesure est loin d'être simple. Pour vous en donner une idée, je me rappelle le dépannage de la carte d'extension mémoire d'un 5205T. Avec la sonde de mon très bon oscilloscope, j'ai touché la ligne CAS (adressage colonne) de la mémoire, et le ST a planté, rien qu'à cause de cette petite capacité introduite sur cet important signal. Effarant, non ?

Une fois, un technicien d'AT&T me montra une carte et me dit qu'il venait de passer un an à en éliminer les plantages intempestifs. Cette année de travail pouvait se résumer à un seul fil, traversant la carte de part en part, mis pour remplacer une piste de la carte qui virevoltait entre les circuits intégrés. Il s'était avéré que cette piste était trop longue et retardait suffisamment le signal pour désynchroniser et planter la carte par intermittence. Un fil bien droit corrigeait le problème. Vous voyez comme les problèmes deviennent vicieux ? Pourrons-nous encore augmenter les vitesses au rythme actuel ?

En résumé, nous avons des vitesses de processeurs qui changent les cartes en casse-tête chinois. Les temps d'accès des RAM approchent le zéro, ce qui serait parfait (actuellement, 40 ns est une RAM dynamique très rapide, mais une RAM statique lente. Pour une RAM statique, une bonne vitesse est 20 ns...). Nous avons des techniques d'assemblage qui empilent toujours plus de composants en moins d'espace, comme le montage en surface ou le montage direct des puces retournées (développé par IBM, je crois - corrigez-moi si je me trompe). Mais les limites approchent. La vitesse de la lumière, un temps d'accès nul, des cartes entièrement remplies de puces. Et puis après?

## **Parlons programmation**

Côté logiciel, nous ne réussissons guère mieux à augmenter l'efficacité. Les programmes modernes sont lourds, encombrants et lents. Il leur faut absolument des machines ultra-rapides à 50 MHz juste pour être tolérables. La conception des compilateurs n'a guère évolué, et n'a pas connu de percée révolutionnaire.

Le langage assembleur offre toujours au programmeur les programmes les plus rapides et les plus optimisés, mais est toujours aussi difficile. Les langages de plus haut niveau (et plus ennuyeux) permettent au programmeur d'être plus productif en échange d'une vitesse moindre et d'une perte de contrôle. Je ne m'y suis toujours pas résigné, mais la plupart des autres programmeurs l'ont fait. Toujours est-il qu'on peut maintenant faire des tableurs plus gros, cependant leur conception est toujours la même. On peut éditer de plus gros documents. Ouais, super. On peut avoir des bases de données plus grandes. (Bâillement) Et alors?

Il n'y a rien de vraiment neuf dans tout cela. Il y a bien de grands discours sur la programmation par objets. Mais personne n'est capable de me dire ce qu'est un objet! Vous voulez rire un bon coup? Mettez trois diplômés en informatique dans une salle, et demandez-leur "Qu'est-ce qu'un objet ?". Otez de la salle tous les objets pointus, ou vous aurez des morts.

### Du vraiment neuf

Pour progresser, nous avons besoin d'idées radicalement neuves, d'un saut quantique. Mais avant que de nous perdre dans une jungle de concepts abasourdissants, faisons une pause et prenons derechef un peu de recul. Rappelons-nous cette citation célèbre : "Un fanatique est quelqu'un qui redouble d'efforts après s'être égaré."

Reconnaissons que les idées actuelles de l'informatique ont été presque entièrement explorées. On ne pourra plus très longtemps continuer à pousser les performances des matériels et des logiciels actuels. Nous sommes dans une véritable impasse.

Il nous faut en sortir et adopter une approche entièrement nouvelle, suivre une piste inexplorée. Permettez-moi donc, avec mon habituelle modestie, de vous soumettre quelques pensées sur ladite "nouvelle approche". Commençons par un peu de métaphysique. Non, attendez, ne tournez pas la page! La métaphysique, c'est l'avant-garde du progrès. Les physiciens quantiques utilisent la métaphysique et les métaphores bouddhistes pour décrire le monde des électrons, et les puces de nos ordinateurs utilisent la physique quantique.

## Je pense...

Vous vous rappelez de "cogit ergo sum"? On le traduit souvent (mal) par "je pense, donc je suis". Mais l'axiome de Descartes gagne à être traduit plus précisément. Descartes essayait de déterminer ce dont l'existence pouvait être prouvée. L'univers existe-t-il vraiment ? Ou bien n'est-ce qu'un produit de mon imagination ? Vieille question, et si l'on réfléchit bien, il est impossible de se prouver réellement que quoi que ce soit existe. Tout comme il est impossible à quelqu'un sous l'emprise d'hallucinogènes de discerner la réalité de l'illusion.

Nous avons cependant une certitude (c'est là tout ce que la philosophie est parvenue à établir en 4000 ans). Nous existons. Comment en sommes-nous sûr ? Parce que nous pouvons douter de notre propre existence. Nous savons que nous doutons - donc, nous devons exister pour pouvoir douter. Quel soulagement, hein?

Prouver que le reste de l'univers existe n'a pas encore été fait. L'univers entier risque peut-être de s'évaporer si vous cessez d'y penser... La véritable signification de "cogito ergo sum" est "je doute, donc j'existe". La clé, c'est le doute. Voilà qui me semble être une piste toute neuve

Défrichons-là.

### La machine à douter

Imaginez maintenant, si vous le voulez bien, un ordinateur qui doute, au lieu des machines actuelles qui suivent aveuglément une procédure pas-à-pas appelée "programme" (et dont l'aveuglement se double d'obstination, surtout quand les choses vont de travers). Je pense sincèrement que cela pourrait être la prochaine génération de recherches informatiques. Je lance cette idée pour qu'elle puisse germer dans vos cerveaux fertiles, car, premièrement, je n'ai pas actuellement le temps de la pousser plus loin que le stade intuitif, et deuxièmement, je suis partisan du partage de l'information selon l'éthique des bidouilleurs. Vous êtes libres de faire fortune grâce à elle.

Difficile d'imaginer un ordinateur qui doute? Mais peut-être vous représentez-vous une machine qui vous aide à déboguer vos programmes ? Non, non, NON! Ce ne serait que du réchauffé, l'amélioration

du développement des programmes actuels. Pour expliquer ce que j'envisage, prenons l'exemple d'une statue ou d'une sculpture.

Il y a deux façons de faire une statue. La première est de la créer par additions successives. En collant les uns aux autres des morceaux de glaise, on bâtit une statue. Il faut savoir où l'on va, et prévoir le support des structures les plus fines. C'est en quelque sorte l'analogue de l'approche actuelle de la programmation, qui consiste à donner des instructions détaillées en vue de la construction d'une structure, via un programme.

Mais il y a une autre méthode, une voie qui ne s'est ouverte aux créateurs d'ordinateurs que très récemment. Nous pouvons également créer et définir une statue en prenant un gros bloc d'argile et en y soustrayant progressivement des morceaux jusqu'à ce que nous obtenions la forme finale. Nous commençons par une infinité de possibilités, et élaguons des morceaux jusqu'à la forme voulue.

Connaissez-vous un seul ordinateur qui fonctionne ainsi ? Connaissez-vous des programmes qui disent à l'ordinateur ce qu'il faut ne PAS faire, et définissent ainsi la tâche à accomplir ? N'espérez pas me river mon clou en disant que définir ce que doit faire un ordinateur, de manière classique, suffit à définir ce qu'il doit ne pas faire. Tous les programmeurs peuvent raconter des histoires horrifiques d'ordinateur échappant à tout contrôle, exécutant des données en guise d'instructions, se plantant de mille manières. Non, nous avons là défini un nouvel idéal, hors d'atteinte de la technologie actuelle.

Cette méthode - appelons-la technologie négative - recèle de nombreux avantages. Par exemple, un embranchement de programme ne pourrait pas prendre une direction inattendue Si vous élaguez cette possibilité en sculptant le programme.

Cette technologie devient désormais accessible grâce à l'accroissement des vitesses et des capacités des machines. Il y a encore peu, les ordinateurs étaient si limités qu'ils avaient déjà grandmal à seulement suivre rigoureusement un programme strict. Ils ne pouvaient guère se permettre de zigzaguer et trébucher le long d'un chemin balisé par des interdictions, comme je l'entrevois. En laissant mon imagination vagabonder, j'en arrive même au sentiment confus que c'est la véritable manière d'appréhender la programmation parallèle, seule voie d'expansion prometteuse dans l'informatique, et qui s'accommode mal, par nature, des méthodes classiques de programmation séquentielle. Bien sûr, si mon esprit invoque ces visions de processeurs multiples louvoyant entre des murs leur interdisant de s'égarer dans des voies aberrantes, c'est peut-être par réaction envers ces ignobles programmes Mac qui plantent et que je dois corriger à contrecœur.

Mesdames et Messieurs du jury, permettez-moi de vous dire pourquoi je pense que cette approche, cette technologie négative expliquant aux ordinateurs ce qu'il ne faut pas taire, mérite d'être examinée. C'est parce que c'est ainsi que les humains apprennent ! On ne programme pas les gens (et encore moins les enfants, comme le savent tous les parents - et j'en ai trois). En fait, les enfants se programment tout seul, et c'est ce que nous recherchons, crénom ! Par exemple, dès son plus jeune âge, un gosse apprend à ne PAS toucher une casserole chaude, ou à ne PAS jouer avec les allumettes. L'outil qui sculpte ces règles dans l'esprit humain est la douleur, ressentie directement suite à une bêtise, ou appliquée par la main parentale. On apprend de préférence ce qu'il ne faut pas faire, et bien peu ce qu'il faut faire (et encore, en traînant les pieds).

Naturellement, en suivant cette approche, vous obtiendriez tout à fait autre chose qu'un esclave borné. Vous pourriez fort bien vous retrouver avec un ordinateur bien décidé à rester debout toute la nuit pour lire un livre, et se moquant éperdument des tables de multiplication (je me demande bien où j'ai pu aller chercher cet exemple, tiens.) Mais cela pourrait bien être la solution de nos problèmes. Des ordinateurs qui apprennent... Les machines à base de réseaux neuronaux s'orientent vers cette direction.

### **Digression constitutionnelle**

C'est d'ailleurs ainsi que fonctionne notre gouvernement, et c'est ce qui le limite. Il est intéressant de noter que la Constitution américaine est remplie de choses que le gouvernement ne doit PAS faire. Notre gouvernement est divisé en trois pouvoirs - exécutif, législatif, judiciaire - qui passent leur temps

à se dire mutuellement ce qu'ils ne doivent pas faire. [NdT la Constitution française est basée sur un esprit différent et dit au contraire ce que le gouvernement peut faire et ce que le peuple doit faire.] En tant que nation, les États-Unis ont plutôt bien réussi comparés à d'autres systèmes de gouvernement. En d'autres termes, un système de règles en technologie négative s'est révélé convivial.

En fait, notre gouvernement ne semble rencontrer des problèmes que lorsqu'il commence vraiment à faire quelque chose. On se rappelle par exemple le cas de Steve Jackson Games ou du fanzine Phrack Newsletter, entre autres. [NdT dans ces deux cas, le gouvernement US -FBI et services secrets - s'est attaqué à des innocents dans l'espoir de pincer des pirates de l'informatique, et a employé sans discernement des méthodes illégales et dévastatrices.] Ceux qui s'y intéressent peuvent contacter l'Electronic Frontier Foundation.

### Pour finir

Ceci termine ce petit exposé. Je me rends compte qu'il y a encore beaucoup de travail dans cette voie, d'idées à explorer, de programmes à écrire. Peut-être un nouveau langage de programmation, destiné à expliquer ce qu'il ne faut pas faire à un ordinateur curieux et touche-à-tout. Un ordinateur qui doute. Je crois que si jamais nous y parvenons un jour, nous aurons franchi le second pas sur la voie des machines pensantes.

**Traduction et adaptation Password 90** 

# Le sorcier blanc et le spectre couleur

Ce mois-ci, Dave musarde un peu. il nous parle de ses projets, en particulier celui d'un livre d'hemic-fantasy. Ce genre est très populaire parmi les ST-istes, et Dave cherche tout simplement à se faire éditer! Et il fournit quelques conseils précieux pour ceux qui rêvent de voir leur nom sur une couverture. Autre projet qui occupe beaucoup Dave, le Spectre couleur, où comment émuler le Mac couleur sur Falcon. Enfin, pour les malheureux qui ont été martyrisés par leur disque dur, Dave nous fournit une recette extrêmement personnelle pour que ces satanés disques se tiennent à carreau...

### Unix, je ne te hais point!

Il y a quelques temps, j'ai mentionné le langage C et le système d'exploitation Unix dans un article. Pour mon malheur, je n'ai pas usé en cette occasion de métaphores grandioses comme "l'invention la plus démente depuis le fil à couper le beurre", ou encore "le plus génial des systèmes de développement, qui a pris la place de Sandy dans ma vie." Car depuis que je me suis mis à Unix, j'ai souffert d'avoir à ingurgiter l'énorme masse de connaissances que cela nécessite. L'Atari TT, par exemple, est livrable avec Unix, et ce système d'exploitation est plutôt dur à aborder pour un débutant (même le bureau du ST est dur à manier pour un débutant complet!).

Manque de pot, la plupart des utilisateurs d'Unix sont connectés au réseau Internet (voir mes adresses en fin d'article). Beaucoup cessèrent immédiatement de lire l'article parce que la fumée qui sortait de leur naseaux leur obscurcissait la vue, et sautèrent sur leur console pour m'expédier un message incendiaire. Ayant reçu quelques gigaoctets de prose véhémente, je tiens à rassurer les fans d'Unix pourquoi aurais-je acheté deux machines Unix fort coûteuses si je détestais tant ce système ? Pourquoi aurais-je englouti 300 dollars rien que pour des manuels X-Window ? Sans compter un ou deux wagons de livres sur System V, BSD, UUCP, Usenet et j'en passe, que j'ai dénichés en fouinant dans tout ce que Denver compte comme librairies.

Mais je tiens à préciser qu'Unix n'est pas fait pour être lâché dans la nature sans précaution. Le grand public n'est pas prêt. C'est ce que AT&T avait tenté de faire avec le 7300, le "PC sous Unix", qui a été un échec commercial. Le 7300 était joli, avait une belle interface graphique avec une souris, était plutôt convivial... Rien n'y a fait. Je persiste cependant à croire qu'Unix a de l'avenir (notez bien, je n'ai pas dit "c'est l'avenir", j'ai dit qu'Unix y a sa place).

On assiste actuellement à une convergence des systèmes d'exploitation. Steve Jobs, à qui revient le mérite de la conception du Macintosh, a coutume de dire que "les bons artistes créent, les grands artistes empruntent". Cela s'applique à l'informatique. Pour les concepts de gestion de fichiers, nous avons par exemple le système de fichiers hiérarchique (HFS) d'Apple, qui ressemble beaucoup à celui de MS-DOS 2.0 avec ses répertoires, lequel ressemble furieusement aux arborescences de fichiers d'Unix. MSDOS, quant à lui, permet une redirection des entrées-sorties, encore que partiellement implémentée seulement, qui est évidemment inspirée d'Unix. Sans parler du langage C, qui est indissociable d'Unix. Autre exemple, l'interface graphique de Présentation Manager d'OS/2, qui s'inspire de celle du Mac, laquelle a un air de famille avec les machines développées au PARC, le centre de recherche de Palo Alto de Xerox. Bien que, je m'empresse de le dire, Apple ait apporté beaucoup d'idées originales au Mac. (Vous vous dites que j'ai l'air de quelqu'un qui écrit quelque chose par crainte d'un procès, hmmm ? Gagné!) Notre industrie fourmille de bonnes idées empruntées. [NdT: à l'appui des dires de Dave, relire les annonces récentes de systèmes d'exploitation permettant de faire tourner indifféremment des applications PC, Mac ou Unix sur une même machine.]

### Temps partagé

Ceci dit, je n'ai pas eu le temps de m'ennuyer ces dernières années. J'ai plus de projets sur le gaz que je n'ai de temps à leur consacrer. Je vous parlerai de mes projets les plus avouables tout à l'heure. Mais ma vie ne se résume pas à un circuit imprimé et à quelques milliers de lignes d'assembleur 68000. J'ai un vieux projet nommé Hyperweb, très ambitieux, que je décrirais en gros comme un nouveau concept de système d'exploitation. J'avoue avoir été souvent tenté de lâcher mes débogages de Spectre GCR pour me consacrer à Hyperweb. Mais les clients d'abord.

Quand vous concevez quelque chose de neuf, vous devez commencer par regarder ce qui se fait déjà. Certes, mon intuition est loin d'être infaillible je ne croyais pas qu'il y aurait de la demande pour un émulateur Mac avec des ROM de 128K), mais il m'est arrivé de voir juste je pensais bien que les gens s'intéresseraient à un émulateur Mac). Et je pense que Hyperweb changerait complètement la façon d'utiliser les ordinateurs, de même que Sidekick a complètement changé la façon d'utiliser les PC.

Bon, d'accord, pas complètement. Ceux qui ne savent que taper "123" pour lancer Lotus à partir de l'invite "C:>" sur leur PC n'auront pas besoin de Hyperweb. Mais les fonctions que celui-ci fournira sont Si essentielles qu'elles auraient déjà dû être incluses dans les ROM des machines.

Mais pour ne pas réinventer la roue, j'ai dû me familiariser avec ce qui se fait dans ce métier. Je me suis donc successivement frotté au Macintosh, au GEM du ST, au PC. J'ai acheté un Amiga. (Eh oui Allez-y, envoyez les tomates!) J'ai apprécié le concept de multitâche de cette machine, j'ai moins aimé les atrocités de son langage de commandes. Et j'ai deux machines Unix.

Curieusement, j'ai découvert que je n'ai pas à emprunter des idées pour Hyperweb. Toutes ces machines auraient bien besoin de la boîte à outils d'Hyperweb. C'est rassurant, car si j'avais par exemple trouvé une idée à emprunter dans Unix, quinze personnes auraient inévitablement été en train d'en préparer une version pour PC. Hyperweb est donc un territoire à défricher. Et de temps à autre, un programme comprenant certains des concepts de Hyperweb apparaît sur le marché, et est salué de bancs d'essais flatteurs. Ce qui montre que l'idée est bonne et que d'autres développeurs y pensent également.

Je suppose que vous désirez savoir si la première version d'Hyperweb sera écrite sur ST. Franchement, je ne sais pas encore. J'ai défini le concept, je sais ce que ce code doit faire, mais je n'ai pas encore défini la disposition des écrans, ni l'interface utilisateur, qui est primordiale. (J'en ferai probablement une maquette sous HyperCard, qui est aussi un excellent générateur de démos d'applications.)

Mais le vrai problème, c'est que Hyperweb doit être écrit de manière portable. Ce qui signifie qu'il me faut l'écrire en C, un langage à l'encontre duquel j'ai un blocage mental. Après, Si je veux le porter sur différentes machines, je n'aurais à réécrire que les habituelles adaptations spécifiques aux systèmes.

Je ne vous dit pas cela pour me faire mousser, mais pour vous donner un aperçu de la façon dont on développe un programme novateur. Quelqu'un qui travaille sur plusieurs types de machines commence à rêver et à imaginer ce qu'il aimerait trouver dans les systèmes d'exploitation de ces bécanes. Si ce rêve lui plaît, il l'implémente. Et si l'implémentation lui plaît tant qu'il pense que d'autres l'aimeront aussi, il le peaufine avant de le commercialiser. La plupart des bons programmes du marché ont vu le jour sous forme de bidouilles amoureusement fignolées. Alors que beaucoup de programmes froidement conçus par des comités comme des montages commerciaux se sont soldés par des échecs cuisants.

En fait, ma grande ambition est de voir Hyperweb finir par être un morceau des ordinateurs du futur, et d'être plus connu pour ce morceau que pour mes émulateurs Mac. Pour citer deux auteurs de SF "Presque aucun exploit n'est impossible à qui n'en revendique pas le mérite." (Roland Green et Jerry Pournelle).

## Le sorcier blanc

J'ai un autre projet qui lutte avec ses rivaux dans mon cerveau pour obtenir un peu plus de temps de processeur. Il s'agit de deux romans d'heroic-fantasy. Le premier, Le sorcier blanc, a été achevé en 1984, et fièrement soumis à un éditeur. A ma grande surprise, j'ai reçu une lettre de refus Il faut dire que d'habitude, je suis plutôt gâté. Tenez, le tout premier article que j'aie écrit, qui expliquait les modes graphiques de l'Atari 800, a été accepté par la revue Creative Computing, et depuis, les lettres de refus ont été rares. Par contre, la fiction est un domaine où je n'ai jamais fait mes preuves. Un roman est beaucoup plus dur à écrire qu'un article, mais créer quelque chose d'entièrement neuf est très gratifiant, et j'aime beaucoup écrire.

Créer un univers de papier est parfois pénible. Une veille blague, qui n'est sans doute drôle que pour les auteurs, dit que "Dieu a inventé la page blanche pour prouver aux écrivains que créer un univers n'est après tout pas si simple".

Le livre "Le sorcier blanc" trouve son origine dans le morceau Foreplay du groupe Boston. C'est le troisième morceau de leur premier album, et à mon humble avis, c'est le meilleur album de rock de la Création. (Et, bien sûr, chacun ou presque a son meilleur album. Je ne suis pas en train d'essayer de vous convertir!). Le livre s'esquissa pendant que je l'écoutais. Et peu à peu, il prit forme dans mon esprit, pendant que je conduisais, ou que j'attendais que se termine un assemblage... En fait, le livre tournait en tâche de fond dans ma tête et revenait à l'avant-plan chaque fois que mon cerveau avait du temps libre à lui consacrer.

Mettre mes activités créatrices en tâche de fond est une vieille habitude. Mes professeurs appelaient cela "rêvasser". Vingt ans après, je réalise qu'ils se trompaient l'important, c'est de concrétiser ces rêvasseries. Après tout, il y a bien des gens qui sont payés pour interpréter les rêves nocturnes, alors pourquoi ne mettrais-je pas à profit mes rêves diurnes ?

J'ai donc couché ce projet de livre sur le papier. Puis je l'ai fait réviser par un correcteur indépendant, qui m'a dit tout ce qui n'allait pas et m'a rendu une liasse de papier couverte de marques rouges. J'ai corrigé mon brouillon, et ai envoyé le manuscrit à un éditeur. En retour, celui-ci m'a envoyé une magnifique lettre de deux pages. Attention, pas un de ces formulaires de rejet succincts, mais une vraie lettre, dans laquelle l'éditeur disait qu'il n'était pas sûr de faire le bon choix, qu'il avait lu et relu le manuscrit avant de décider de ne pas le publier, et qu'il souhaitait que je lui envoie mon prochain.

En discutant avec d'autres auteurs, je me suis aperçu que ce n'était pas exactement le style usuel des lettres de refus. C'est pourquoi je persévère. J'ai donc lu beaucoup, demandé moult opinions, et je me sens à présent mieux armé qu'en 1984. Il est donc temps de refondre et réécrire ce bouquin.

C'est ainsi que les personnages du Sorcier blanc sont de retour à l'arrière-plan de mon cerveau. Certaines choses n'ont pas changé. Nimue, l'apprenti de Merlin, se bat toujours en 1342 pour éviter que la peste noire n'anéantisse la race humaine. Kevin perd toujours le contrôle d'un démon qu'il avait invoqué pour combattre le Sorcier noir, à son grand dam. Et chaque fois que j'ai un peu de temps, entre deux bogues ou deux routines, ces êtres s'animent et évoluent. L'interruption qui suit immanquablement les fige à nouveau jusqu'à la tranche de temps suivante (changement de contexte, pour parler en unixien!), et il me faut une bonne seconde de temps de latence pour revenir à la réalité.

Quant au second livre, c'est en patientant à un feu rouge que j'y pense le plus. Il faut dire que je déteste les feux rouges. Je parierais que l'enfer en est truffé. Alors plutôt que de les maudire, je m'occupe en rêvassant utilement...

Et puis d'abord, je ne rêvasse pas, je pense, môssieu! Ou plus exactement, je me pose des questions. Et je m'en pose à longueur de journée. Poser des questions inédites, c'est la clé des découvertes. "Pourquoi ne peut-on pas faire tourner des programmes Mac sur le 68000 d'un ST ?" Réponse : on peut, voir le Spectre. Pourquoi ne peut on pas travailler directement sur les fichiers en fonction de leur contenu et non pas, comme actuellement, en fonction de ce qui est commode pour les programmeurs ?" Réponse : Hyperweb. "Pourquoi la magie, qui semblait marcher jadis si l'on en croit les légendes, ne marche-t-elle plus ?" Réponse dans Le sorcier blanc. "

Bien sûr, 90% des questions ne mènent nulle part, et ne font qu'irriter mon entourage. Heureusement, ma femme est d'une infinie patience.

Tout le problème est donc de trouver les bonnes questions. Comme un prospecteur d'or, il me faut passer au crible beaucoup de boue avec de trouver des paillettes. Jusqu'à présent, cela n'a pas trop

mal marché, puisque j'ai même un certain renom dans le monde Atari.

### Les servitudes de la gloire

Ne me prenez pas pour un poseur si je vous confie qu'être connu n'a pas que des avantages. Il y en a quelques-uns : en particulier, il est amusant d'avoir sa photo dans des magazines d'informatique, voire en couverture.

Mais avant de commencer à m'envier, sachez qu'il m'est quasiment impossible de visiter normalement un salon informatique. J'aimerais pouvoir me promener entre les stands, poser des questions, et profiter des démonstrations. Mais le plus souvent, je suis soumis à un feu roulant de questions, et je m'enroue à force de m'égosiller dans le micro du stand. Et encore, j'arrive tout juste à me faire entendre. Il me faut ériger une barricade de tables pour avoir une chance d'échapper à la foule.

Ensuite, si par malheur, je fait trois pas dans une allée, une demi-douzaine de personnes surgissent pour me poser "juste une question rapide sur le GCR", et je ne peux rien visiter.

Et si un produit exposé me plaît, le revendeur veut me l'échanger contre un Spectre, veut l'autorisation de me citer comme utilisateur de son produit, et veut savoir si j'en ferai un banc d'essai, et pour quel journal. Pas moyen de lui tendre ma carte de crédit et d'en terminer.

Le pire fut un salon où il me fallut aller aux sanitaires, espérant pouvoir au moins échapper à la foule un instant. Mais quatre personnes m'y suivirent, sans interrompre leur discussion! J'étais au bord de l'agoraphobie. J'aimerais quand même pouvoir visiter un salon tranquillement! Peut-être qu'en portant un faux nez et des lunettes façon Groucho Marx, la prochaine fois ?...

### Spectre couleur

À propos de questions, l'une de celles qu'on me pose le plus souvent concerne le Spectre couleur. On me demande fréquemment quand il sortira. Il est toujours appréciable de constater qu'il y a une demande pour un appareil que vous pouvez fabriquer. Même si concevoir cet appareil peut mettre en péril un équilibre mental déjà précaire...

Car le Spectre couleur pose de réels problèmes techniques. Dans les ROM des Mac couleur se trouvent une série de routines graphiques, nommées Color Quickdraw, qui utilise les instructions à champ de bits du microprocesseur. Ces instructions comprennent une partie variable dans laquelle on place des bits en fonction de ce que l'on veut faire. Et elles ne se trouvent que dans le 68020 ou le 68030. Cela implique que les ST, contenant des 68000, ne pourront pas faire tourner les routines Color Quickdraw. Point final. Il faudrait ajouter un 68030 à un ST, ce qui est une extension plutôt coûteuse. De plus, le GCR devrait être modifié pour s'accommoder des nouvelles ROM de 256 Ko ou 512 Ko, voire des barrettes SIMM contenant de la ROM. Encore une extension, qui réclamera une refonte totale du circuit imprimé du Spectre, sachant que le port cartouche du ST ne permet d'adresser que 128 Ko.

Car sachez qu'en ouvrant des Mac couleur, on trouve une grande variété de ROM différentes (je ne parle pas du SE qui n'a pas de ROM couleur)

(b) Mac II: 68020, 4 EPROM de 64 Ko

Mac lix 68030,1 barrette SIMM de ROM

Mac SE-30 : idem Mac Ilcx: idem

Il faut y ajouter les llfx, lîci lîsi et tous les modèles que les commerciaux de l'équipe de John Sculley ont cru bon de sortir. Certains ont des ROM de 512 K! C'est pourquoi il ne faut pas acheter un jeu de

ROM à la légère. Le prix de ces ROM varie énormément, et certains ne marcheront pas sur le futur Spectre couleur, s'il sort jamais.

Oui, j'étudie la question. J'ai cessé de dire non, tout au moins. Jusqu'à la sortie du Falcon, il était déjà difficile de savoir sur quel Atari on pouvait faire tourner QuickDraw. Le TT, le seul Atari à base de 68030, n'affichait que 16 couleurs en haute résolution. Certes, à sa sortie d'usine, le Mac II n'affiche également que 16 couleurs, et la première chose que font ses utilisateurs est de commander une extension de RAM vidéo, afin de pouvoir afficher 256 couleurs. Des cartes graphiques comme la Crazy Dots permettent d'augmenter la résolution du TT, mais la plupart sont basées sur des générateurs vidéo conçus pour les PC et qui exigent (ô abomination !) de la mémoire découpée en segments de 64 Ko! Le 68030 utilise une mémoire normale, linéaire, sans ces découpages aberrants hérités d'un temps reculé où les touches des claviers des ordinateurs étaient en ivoire de mammouth. Il est quasiment impossible de faire fonctionner ces générateurs sur un 68030. Sauf peut-être avec un ignoble bricolage où la MMU ferait une pagination par 64 Ko, et qui serait d'une lenteur effroyable ! (A moins qu'avec une carte Chromax sur le port VME du TT... Hmmm...)

Bref, le seul ordinateur qui convienne est le Falcon 030. Il a un 68030 à 16 MHz et un bus de 16 bits, avec d'excellents modes vidéo, et peut faire du 640 x 480 en 256 couleurs. Je ne désespère pas de parvenir à faire marcher l'affichage du Falcon comme celui de la carte vidéo standard du Mac II. (Pour une discussion détaillée du rôle de la mémoire vidéo et de la haute définition dans les performances d'un ordinateur, voir "Philosophie et conception du SST 68030".) [NdT: paru dans ST-Mag n°55.]

Le Falcon est également doté d'un connecteur interne, qui conviendrait à merveille pour y enficher une carte Spectre dotée des ROM couleur. Atari s'y intéresse, et nous a donné un Falcon dès mai 1992!

C'est pourquoi je crois que nous pourrions arriver à mettre au point un Spectre couleur, au prix cependant d'un travail titanesque. Au vu de l'expérience acquise avec les ROM 64 K et 128 K, je peux vous assurer que Color Quickdraw ne se laissera pas dompter simplement. Notez bien que j'étais également très intimidé par les ROM 128 K, et j'ai commencé à les faire fonctionner en à peine trois mois. Ma chance tiendra-t-elle une troisième fois ? C'est un sacré pari!

Oui, nous pouvons y arriver, mais... Y aura-t-il assez de Falcons vendus pour constituer un marché? Presque tous ceux qui ont vu un Falcon admettent que c'est une machine fantastique, et beaucoup en veulent un. Les gens craignent cependant qu'Atari ne fasse pas suffisamment la promotion du Falcon ("comme d'habitude", ai-je souvent entendu), en particulier aux USA. Atari a des ressources limitées. Le magazine financier Forbes a passé Atari en revue dans un article paru en 1992, et le résultat incite à la prudence. L'article concluait cependant qu'Atari a suffisamment de fonds pour lancer une campagne de promotion sur le Falcon. Il me vient une image mentale d'un oisillon jeté hors du nid à grands coups de pied par un publiciste lui criant "Vas-tu voler, crénom !" [NdT: Atari a depuis décidé que ses ressources publicitaires seraient concentrées sur la console Jaguar jusqu'à ce que redémarrent les ventes d'ordinateurs, que la récession a écornées.]

Cela pose à Gadgets By Small un cas de conscience. Nous ne savons pas combien de Falcons seront vendus, donc nous ne savons pas s'il y a un marché suffisant pour un émulateur Mac couleur sur cette machine. Ou bien les utilisateurs du ST ou du TT voudraient-ils d'une extension Mac pour carte SST 68030 ? Votre avis est le bienvenu. Vous avez nos coordonnées en fin d'article. Vous pouvez nous joindre par télécopie ou par courrier électronique.

Quant au nom du produit... Spectre GCR est déjà le nom le plus mal prononcé de l'histoire de l'informatique. Les gens nous posent des questions sur le "Specter GRC" ou autres déformations. Si jamais nous sortions un Spectre CGCR (Color Group Coded Recording), le taux de déformation linguistique ambiant pourrait bien croître jusqu'à mettre en danger la civilisation ! Si vous avez une proposition pour un beau nom sonnant bien, faites-nous le savoir.

## Faire un exemple

Puisque nous parlons de projets de développement, permettez-moi de vous raconter comment j'ai fait

pour me débarrasser de mes problèmes de disques durs. Il était une fois deux disques durs... Ça commence comme un conte de fées; mais ça tourne rapidement au cauchemar. Dieu sait que les disques durs en ont causées, des histoires d'horreur, mais il semble que je les collectionne. Il faut dire que je développe des routines de pilotage de disque, ce qui explique beaucoup de défaillances durant un test, au moindre bogue d'une routine d'écriture disque, le disque dur entier est corrompu, et il faut tout reconstituer. Le développement du Spectre s'est d'ailleurs accéléré lorsque j'ai acquis une unité de sauvegarde sur bande ICD. Chaque fois que je rajoutais quelque chose à mes disques, je les sauvegardais, et en cas de pépin, je pouvais les restaurer plus rapidement.

Mais ces deux disques-là m'ont rendu cinglé. Sur l'un d'eux, j'avais réuni le contenu de nombreuses disquettes, passant des heures à cataloguer et vérifier les fichiers. Je m'apprêtais à tout sauvegarder lorsque le disque dur planta, et perdit ses données d'alignement en piste 0. Rideau, irrécupérable.

Sur un autre, je perdis des données importantes et des heures de travail sur Spectre, dont de nombreuses corrections de bogues. Et RIEN N'EST PLUS ENERVANT QUE DE CORRIGER DES BOGUES POUR LA DEUXIEME FOIS !... Ahem. Excusez-moi, je m'emporte. Deux disques de 20 mégaoctets, vous vous rendez compte ?

Alors, un beau jour d'août, mon voisin et moi sommes allés à un stand de tir. J'avais amené mon petit bijou, un fusil AR-15 à lunette de visée télescopique. (Avec quelques éléments en moins, l'AR-15 est identique au M-16.) Ceux qui détestent les armes parmi vous ne vont sans doute pas comprendre, mais j'aime le tir à la cible, parce que c'est l'un des sports les plus intenses qui soit. Il n'y a que vous, la cible, et votre aptitude à maintenir une visée parfaite. La plus minuscule erreur, et c'est un manqué irrémédiable. Par contre, le style Dirty Harry qui défouraille dans tous les coins, ce n'est pas mon genre.

Hormis le fusil, j'avais aussi amené ces deux satanées "gamelles", harr harr harr !... (Rire sadique.) Nous avons pris deux batteries de 6 volts. Le négatif de l'une à la masse, et son positif sur le +5 V. L'autre batterie entre le +5 V et le +12 V. Ce n'était pas la tension exacte, mais c'était suffisant pour lancer les moteurs des disques. Ceux-ci entrèrent en rotation avec un sifflement crescendo, se moquant probablement de l'humain à qui ils avaient joué ces bons tours.

Nous avons reculé de deux cents mètres et mis l'AR-15 en position. Avec sa lunette grossissant neuf fois, je pouvais presque lire les étiquettes de garantie sur les disques. Et j'ai fait un beau trou dans chaque disque. Et même plusieurs. Bon, d'accord, je l'avoue, j'ai criblé de balles ces sales mécaniques!..

Ce fut très réjouissant. À la première balle, les plateaux, tournant à 3600 tours par minute (60 tours par seconde!) s'immobilisèrent presque instantanément. L'énergie de rotation emmagasinée se libéra, envoyant les disques tournoyer dans les airs, et éparpillant leurs morceaux aux quatre coins de la planète avant de retomber. Sans doute ce qu'on appelle un atterrissage, hin hin hin...

D'après Sandy, je souriais jusqu'aux oreilles en rentrant chez moi, avec mes deux disques réduits à l'état d'épaves. En descendant l'escalier qui mène à mon labo, je les entendais cliqueter d'un bruit de ferraille, à cause des morceaux qui se baladaient à l'intérieur et des trous de calibre 5,56 mm dans leur carrosserie.

Sur les disques durs que j'utilise, j'ai posé les épaves, et je leur ai dit "Vous voyez ces disques ? Essayez donc de me contrarier, et vous subirez le même sort."

À ce jour, mes autres disques n'ont pas bronché.

Traduction et adaptation Password 90

# Le principe d'incertitude d'heisenberg

Dave nous a déjà initié aux problèmes de l'électronique à haute vitesse (cf. "Des ordinateurs qui doutent" dans <u>ST-Mag n°76</u>). La démonstration continue ici. Les concepteurs de systèmes numériques, en particulier d'ordinateurs, doivent désormais se recycler pour appréhender les phénomènes complexes survenant dans les circuits électroniques actuels, qui fonctionnent à plus 100 mégahertz. Mais même en se dotant d'outils théoriquement adéquats, ces concepteurs doivent s'attendre à affronter des situations apparemment irrationnelles. Et qu'y a-t-il de pire que l'irrationnel pour un technicien?

### **Miracles**

Si vous êtes versé dans la physique, vous avez sûrement entendu parler d'Heisenberg. L'une de ses lois établit que l'on ne peut pas prendre réellement connaissance des phénomènes que l'on cherche à étudier parce qu'en les observant, on les modifie. Du moins au niveau des particules subatomiques. L'exemple le plus connu est celui de l'atome : quand vous essayez d'observer un atome, vous faites des observations fantaisistes parce que les photons (particules de la lumière) bousculent l'atome, modifiant ainsi son niveau d'énergie, ses orbites et son spin. Vous ne pouvez pas savoir quel était l'état de cet atome avant que vous ne l'observiez.

D'une façon similaire, développer ma carte accélératrice 68030 SST fut pour moi une expérience intéressante, qui a vu triompher le principe d'Heisenberg à l'échelle humaine. J'y ai attrapé des cheveux blancs.

Je crois qu'à force d'entendre parler de machines de plus en plus performantes, les gens sont blasés, et plus personne ne s'étonne de voir des ordinateurs tourner à des vitesses ahurissantes. Et pourtant, c'est un miracle que ces machines puissent seulement fonctionner

Sachez, bonnes gens, que tous ces ordinateurs à 50 mégahertz et 8 mégaoctets représentent à mes yeux un vrai miracle technologique. Je sais que chez des centaines de fabricants, des ingénieurs versent de la sueur et des larmes pour les faire marcher, afin de ne pas faire mentir les brochures déjà imprimées par les commerciaux. Je le sais car je suis passé par là, moi aussi. Je vous assure que c'est terriblement difficile. (Cela ne constitue en aucun cas une excuse lorsque nos machines ne tournent pas. D'ailleurs, le SST est bien conçu, rapide et peu coûteux. Non, mais.)

Je voudrais vous parler un peu de ce prodige que constitue une carte tournant à haute vitesse. Souvenez-vous que cela s'applique à tous les ordinateurs rapides que l'on trouve sur le marché, et non uniquement à la SST.

Prenez l'électricité, qui constitue l'alimentation de tous ces ordinateurs. Représentez-vous la sous la forme d'un liquide baignant les composants, un peu comme de l'eau qui s'écoule d'un tuyau. La pression s'appelle ici voltage, le nombre de litres qui s'écoulent par minute représente l'ampérage. C'est une assez bonne analogie.

### Vitesse

Cette électricité se déplace à une certaine vitesse : 300 000 km/s (ou encore 30 cm/nanoseconde). C'est le facteur "c" (célérité) de la fameuse équation d'Einstein E=M.c² . C'est également la vitesse de la lumière, que rien ne peut dépasser dans l'univers. (Notez que l'électricité se propage à une vitesse moindre dans certains types de fils; elle circule plus rapidement dans les câbles, plus lentement dans

les pistes conductrices des circuits imprimés. De même, la vitesse exacte est 299 792 km/s. [NdT : et on a même redéfini la valeur de la seconde pour que le chiffre tombe rond.] Mais vous me permettrez cette approximation.)

Cela revient à dire qu'en une nanoseconde, soit un milliardième de seconde, l'électricité parcourt pratiquement 30 cm. Maintenant, prenez par exemple la carte mère d'un quelconque clone PC bas de gamme, un circuit imprimé de 60 cm de côté. Cela signifie que le courant met environ deux nanosecondes pour la traverser de bout en bout. Et croyez-moi les p'tits gars, c'est limite. Parce que les signaux se déplacent en groupe, et sont étroitement synchronisés entre eux (on appelle ces groupes des "bus").

Prenons par exemple les 32 fils véhiculant une adresse. Ces 32 bits sont accompagnés d'un autre signal qui indique que les fils sont à la valeur correcte à un instant donné. Lorsque ce signal (qu'on appelle une ligne de contrôle) change d'état, c'est-à-dire lorsqu'une impulsion électrique y est envoyée, les circuits auxquels cette adresse de 32 bits est destinée savent qu'ils peuvent lire sur ces fils une valeur binaire correcte. Mais Si par malheur, en concevant le circuit imprimé, vous avez fait faire trop de zigzags au fil véhiculant cette ligne de contrôle, vous pouvez très bien retarder le signal de 3 ou 4 ns (nanosecondes), et Si la synchro est vraiment serrée, votre adresse ne sera plus valide lorsque l'impulsion arrivera sur la ligne de contrôle. Et votre carte ne marchera pas, même Si le câblage est rigoureusement correct.

Vous essayez alors de découvrir pourquoi votre carte parfaitement câblée ne marche pas. Vous branchez vos appareils de mesure. C'est bien, mais vos instruments utilisent des câbles qui transmettent le courant à la vitesse de trente centimètres/seconde également ! En plus, ils y ajoutent une chose appelée "capacité" qui libère progressivement les charges électriques, tout comme un ballon en caoutchouc intercalé dans un tuyau d'eau se dilate et se dégonfle lorsque la pression varie. Et quand des signaux électriques sont déformés par une capacité, ils passent progressivement de O à 5 V ou l'inverse, au lieu de passer instantanément d'une valeur à l'autre comme il se doit.

Hélas, pour un ordinateur, l'univers n'est composé que de zéros et de uns, représentés respectivement par O et 5 V. Entre les deux se situe une zone interdite où nul signal ne doit s'aventurer I Les signaux déformés modifient le fonctionnement de la bécane que l'on observe, et une fois encore, Heisenberg triomphe ! (Très souvent, brancher un appareil de mesure, comme un oscilloscope, empêche un ordinateur de fonctionner, et l'ingénieur doit trouver une autre façon de se rendre compte de ce qui se passe à l'intérieur de sa machine sans esquinter les fragiles signaux.)

## Les bouts de fil de cray

Lorsque Seymour Cray, le concepteur des fameux ordinateurs du même nom, construisit son premier super-ordinateur, le Cray 1, il utilisa des fils de couleurs différentes pour câbler les différents sous-ensembles. Selon sa couleur, chaque fil avait une longueur précise qui était un multiple de 30 cm. Le Cray 1 était ultra-rapide pour les composants de l'époque, et la synchronisation des différents signaux était absolument cruciale. Cray synchronisa soigneusement sa machine on ajoutant une nanoseconde ici, une autre là, pour que tous les signaux d'un bus arrivent en même temps. Pour retarder un signal de N nanosecondes, il le faisait passer dans un fil de N fois 30 cm. A la moindre erreur (deux couleurs interverties, par exemple), l'ordinateur plantait. On pouvait aussi provoquer des pannes en soudant les fils au lieu de les "wrapper". Le "wrapping" consiste à dénuder l'extrémité d'un fil isolé et à l'enrouler autour de la patte d'un connecteur. Si c'est bien fait, cela donne, curieusement, une propagation du signal plus rapide qu'avec des soudures.

Le Cray 1 utilisait des circuits intégrés bipolaires, d'une technologie dite ECL (Emittor-Coupled Logic), qui était difficile à miniaturiser et très gourmande en énergie. Ces circuits intégrés étaient donc très nombreux, et leur consommation électrique était digne d'un four à arc. La véritable prouesse technologique du Cray 1 était en fait son système de refroidissement, sans lequel la machine se serait liquéfiée en une flaque de métal fondu!

Seymour Cray découvrit également qu'en dessinant les pistes de cuivre (ou "traces") d'un circuit

imprimé, il fallait éviter de leur faire décrire des angles droits. Sinon, les électrons avaient en quelque sorte tendance à rebondir au passage de l'angle, et le front du signal était déformé et arrivait en capilotade, avec moult oscillations parasites. Ce qui plantait la machine. Cray dut remplacer chacun des angles droits de ses traces par deux virages à 45 degrés, ce qui préservait le front du signal.

Le plus incroyable est qu'aucun des instruments de mesure de l'époque ne lui permettait de visualiser le phénomène. Je pense que l'idée de la cause du problème et la solution lui sont venues intuitivement. C'est grâce à ce genre d'intuitions qu'il a pu donner son nom à des ordinateurs. En fait, pour reprendre une citation de Conan Doyle :"Lorsque vous avez éliminé l'impossible, ce qui reste, même Si c'est improbable, est forcément la vérité." Certes, mon cher Sherlock, mais permettez-moi d'y répondre par une citation de mon cru "Quand on débogue, rien n'est impossible, et c'est bien le problème !"

#### Comparaison explosive

A l'échelle de l'électronique moderne, la nanoseconde est une unité de temps banale. Tenez, dans la SST, nous utilisons des circuits logiques relativement rapides dont le délai de propagation (le temps qu'il faut aux signaux pour les traverser) est de 15 ns. Rien de spécial, ce sont des circuits tout à fait classiques. Pourtant, une petite comparaison physique nous aidera à comprendre pourquoi ce délai, considéré comme normal de nos jours, est en fait diablement court. En 15 ns, la lumière se propage de seulement 5 mètres!

Ou encore, prenons le phénomène physique le plus destructeur jamais créé par l'homme, l'explosion d'une bombe à hydrogène. La raison pour laquelle une bombe H est Si puissante est que toute son énergie est libérée en un délai extrêmement court, 30 ns. Pendant quelques secondes, la sphère autour de l'explosion est plus chaude que la surface du soleil, bien que cette sphère soit en expansion très rapide. Et pourtant, cette terrifiante explosion est déjà deux fois plus lente que ce banal circuit!

#### Société anonyme puskiplante

Le débogage d'une carte est une épreuve nerveuse quotidienne. Et quand le prototype marche, n'allez pas croire que vous êtes tirés d'affaire, oh, que non ! Il vous faut commander des composants, il faut les assembler, les tester... Limitons-nous aujourd'hui au premier point. Donc, vous concevez une carte, votre prototype marche, vous devez en faire quelques centaines d'exemplaires. Vous a-t-on déjà livré les bons composants 7 Quand je faisais les premiers Spectre GCR, j'étais tombé sur des distributeurs de circuits intégrés qui étaient particulièrement bornés. J'ai failli devenir cinglé. Mais j'ai appris la leçon.

Du coup, plutôt que d'avoir à commander et assembler une flopée de circuits logiques pour la SST, j'ai préféré les regrouper dans un circuit logique programmable (PAL, Programmable Array Logic) commandé chez un certain fabricant.

Nous nous apprêtions à expédier un lot de cartes SST, mais certaines cartes, pourtant réputées bonnes, s'obstinaient à ne pas marcher, et nous nous frappions la tête contre les murs à essayer de comprendre pourquoi. C'est alors que le fabricant nous envoya une note nous avertissant que certains de ces PAL étaient défectueux s'ils émettaient certaines combinaisons de signaux, cela consommait trop de courant, et une brève impulsion parasite apparaissait deux nanosecondes plus tard.

Il est particulièrement difficile de détecter un parasite de ce genre. Si vous placez une sonde d'oscilloscope sur le fil suspect, celle-ci va y introduire une faible capacité qui va absorber le parasite. Et vous ne le verrez pas sur votre écran. En fait, dans notre cas, la sonde l'absorbait Si bien que la SST défectueuse se mettait mystérieusement à fonctionner! Ce qui est encore un magnifique exemple d'application du principe d'Heisenberg: en observant le phénomène, nous le modifiions.

C'était bien triste pour nous, parce que jusqu'à ce que nous recevions cet avertissement du fabricant,

nous n'avions qu'un seul moyen de livrer une carte SST à un client : c'était de l'expédier avec un oscilloscope et une notice du genre "branchez la sonde ici." Je suis sûr que cela nous aurait valu un franc succès!

Nous avons finalement commandé de nouveaux PAL chez un autre fabricant. Ce qui a occasionné un délai supplémentaire de 2 à 3 mois, vu que la conception de ce circuit a dû être revue. Le plus pénible nous attendait encore, puisqu'il fallut remplacer les PAL et retester les cartes. Mais au moins, la SST marche à présent même sans oscilloscope. Et ce n'est qu'un des bogues que nous avons dû résoudre pour cette carte relativement simple.

A une moindre échelle, nous retrouvons là le problème des instruments de mesure qui ne sont pas assez rapides. Certes, les instruments évoluent. Mais les vitesses des circuits aussi. Et les concepteurs, surtout dans les petites entreprises, n'ont pas toujours les moyens de s'offrir des instruments au moins dix fois plus rapides que les circuits qu'ils conçoivent, ce qui est un minimum pour pouvoir repérer tous les parasites. Il existe aussi à présent des logiciels d'aide à la mise au point de circuits imprimés à haute fréquence, mais ils donnent souvent des cartes peu compétitives qu'il faut retoucher et optimiser à la main avant la mise en production. Dans tous les cas, les outils sont hélas en retard par rapport aux besoins des ingénieurs les plus pointus. Et ceux-ci doivent souvent recourir à des méthodes empiriques qui s'apparentent plus à l'art qu'à la science. Que dis-je, de l'art ? C'est parfois carrément du vaudou!

#### Rebond

Un autre tour que l'électricité peut nous jouer à ces fréquences s'appelle le rebond. Cela survient lorsqu'on fait changer d'état un signal sur un fil (par exemple, quand il passe de 0 à 5 V). Au lieu de monter sagement à la bonne tension, le signal dépasse cette valeur, puis redescend et oscille un moment avant de se stabiliser enfin à 5 V. Résultat, le signal est vu pendant un court instant comme une rapide succession de 0 et de 1, ce qui sème une épouvantable pagaïe dans tous les circuits dont la synchronisation dépendaient de ce signal.

Heureusement, le rebond persiste lorsque l'on connecte un oscilloscope, ce qui permet au moins de le voir. On peut alors soit ajouter des circuits amortissant ces rebonds (ce qu'on appelle des terminaisons de bus), soit tenter de "secouer" ce pauvre fil moins fort, en y envoyant des signaux moins énergiques, aux transitions plus douces. Comme quoi, le mieux est l'ennemi du bien : Si vous prenez des composants trop rapides qui délivrent des fronts de signaux aussi nets que les plis de pantalons d'un lord anglais, vous allez vous retrouver avec des rebonds partout sur votre carte.

La cause principale du rebond est l'inductance. C'est en quelque sorte l'inertie des conducteurs, leur tendance naturelle à s'opposer à toute variation du courant qui les traverse. Lorsque la tension sur un fil passe brutalement de 0 à 5 V, lors du passage d'une impulsion (dans ce cas-là, un 1 logique), cette tension va croître à partir de zéro, atteindre 5 volts, les dépasser de quelques fractions de volts, redescendre, et osciller un peu avant de se stabiliser. Mais Si le signal repasse maintenant à 0 volts (un 0 logique), la tension va de même décroître, atteindre zéro volts, continuer à décroître, devenir négative (jusqu'à, disons, -2 V), remonter et osciller. Le rebond se traduit ici par ce qu'on appelle un dépassement négatif. Or, les circuits logiques ne supporte pas les tensions négatives, qui les détruisent très vite. Les mémoires, en particulier, en ont une sainte horreur, et ont toujours exigé une attention particulière. Eviter de détruire les composants d'une carte à coups de dépassement négatifs est presque devenu une science à part entière. Et je puis vous assurer qu'il est très difficile de déverminer une carte qui s'autodétruit en 100 milliardièmes de seconde ! (Enfoncée, la bande magnétique de Monsieur Phelps dans "Mission impossible"!)

#### Soudage

Je vous avais parlé de ce prototype de SST que j'avais moi-même soudé à la main et qui plantait de temps en temps, alors que les cartes soudées à la vague, par un procédé automatique, marchaient parfaitement. Irrité par ces plantages intermittents, j'empoignai mon fidèle oscilloscope et branchai une sonde sur la piste d'alimentation des puces mémoires de cette carte soudé à la main. Et je fus horrifié d'y découvrir des dépassements négatifs de -2 volts ! L'alimentation des mémoires doit être ultra stable. La fiche technique de l'un des circuits mémoires précise même qu'une variation de 0,2 V peut empêcher la puce de fonctionner. Et là, j'étais à 10 fois plus I Le miracle était que cela ne plante que de temps en temps. Des sueurs froides m'envahirent la conception de ma carte était-elle donc viciée à ce point?

Puis je fis la même mesure sur une carte soudée à la vague. Les rebonds n'y dépassaient pas 0,05 V. Conclusion ce n'était pas la conception de la carte, seulement une carte défectueuse. Mais pourquoi ? La seule différence était la méthode de soudage. C'est ce qui provoquait ce rebond.

Oui, bon, je sais ce que vous pensez :"ce gars sucre les fraises et s'amuse à tenir un fer à souder." Eh non, mes soudures étaient excellentes, et j'ai même vérifié qu'il n'y avait pas de soudure sèche (c'est-à-dire des faux contacts) au niveau des alimentations. Je n'ai aucune certitude quant à la cause de ce problème. Peut-être la forme des soudure influe-t-elle 7 Les soudures faites à la main sont plutôt pointues, alors que les soudures à la vague sont bien rondes. Et je sais que les formes des conducteurs influent sur la répartition des charges, tout spécialement à haute fréquence, où les charges se concentrent à la surface des conducteurs et sur les pointes.

En tout cas, c'est l'une des raisons pour lesquelles nous testons chacune de nos cartes avant de les expédier.

#### Plantages mystérieux

Notez bien que certains types de problèmes furent longtemps considérés comme mystérieux avant d'être résolus de manière routinière. Par exemple, lorsqu'un circuit intégré fait changer d'état un fil, et le fait passer de 0 à 5 V ou l'inverse, il se produit une commutation dans un transistor à l'intérieur de la puce. Quand ce transistor commute, il draine du courant électrique depuis l'alimentation de la puce. Et les pattes d'alimentation de la puce, à leur tour, drainent du courant depuis le circuit imprimé.

La vitesse avec laquelle les puces modernes commutent est telle que les pistes de cuivre ne peuvent fournir du courant assez vite. C'est pourquoi on met près des puces des petits condensateurs (nommés capacités tampons), placés entre le 0 et le +5 V, qui stockent des charges électriques et peuvent les fournir à la puce instantanément lors d'une commutation. Le condensateur se recharge entre deux commutations. Et Si on ne met pas assez de ces capacités tampons, on obtient des plantages "inexplicables". Du moins avait-on du mal à se les expliquer dans les premiers ordinateurs à circuits intégrés, dans les années soixante, mais maintenant, c'est un problème tout à fait banal.

Autre type de plantages "mystérieux" les rayons cosmiques. J'ai déjà eu l'occasion de dire que les rayons cosmiques qui nous frappent peuvent modifier l'état d'un bit lorsqu'ils frappent une puce mémoire. Ce sont des rayons dits "ionisants", pour la simple raison qu'ils arrachent sur leur passage les électrons des atomes qu'ils frôlent (et un atome avec une charge en plus ou en moins est dit ionisé). Or, comment un bit est-il représenté dans une puce mémoire ? Par une charge électrique infime stockée dans l'équivalent d'un minuscule condensateur, à l'intérieur d'une des cellules élémentaires qui composent la RAM. Et cette charge devient de plus en plus réduite à chaque nouvelle génération de RAM, puisque les cellules mémoires deviennent de plus en plus petites. Si bien qu'un rayon cosmique peut facilement apporter suffisamment de charges pour faire changer d'état quelques cellules s'il frappe une puce.

Et notre univers marine dans un bruit de fond de rayons cosmiques. Ici, à Denver, c'est pire, puisque l'altitude de 1600 m nous prive d'une bonne partie de la protection de l'atmosphère. Par exemple, alors que j'utilisais un éditeur de texte, j'ai vu un caractère changer comme par magie.

C'est précisément pourquoi les IBM PC et compatibles (et certains Macs) sont dotés d'un dispositif de vérification, nommé contrôle de parité, qui détecte l'inversion d'un bit en mémoire, et arrête tout Si cela

se produit. En fait, leurs barrettes mémoire sont dotées de 9 bits au lieu de 8 par octet, et ce bit surnuméraire stocke ce qu'on appelle la parité. C'est un bit qui vaut 1 Si l'octet contient un nombre pair de bits à 1. Si l'un des bits de l'octet ou le bit de parité bascule intempestivement à cause d'un rayon cosmique, le changement est détecté. Le problème des rayons cosmiques affecte tous les ordinateurs. Certaines stations de travail sous Unix ont même plusieurs bits de contrôle par octet, et peuvent rectifier automatiquement un bit qui a basculé.

Mais ne croyez pas qu'il suffise de vivre en plaine pour être protégé contre ces problèmes les sols émettent des radiations naturelles, en particulier les sols granitiques, qui sont plus radioactifs que les autres.

#### Et le futur?

Les ingénieurs qui conçoivent les ordinateurs se battent contre la barrière que constitue la vitesse de la lumière. Et cette barrière est infranchissable, Si le postulat d'Einstein est correct.

Mais on trouve désormais des machines contenant des processeurs cadencés à 250 MHz, qui ont d'ailleurs besoin d'être refroidis presque aussi énergiquement que le Cray 1 de jadis! Les ingénieurs, au pied du mur, n'ont pas d'autre choix que de miniaturiser sans cesse davantage les composants pour que les signaux aient moins de distance à parcourir, ce qui permet d'augmenter les fréquences de fonctionnement. A quelque 150 km d'ici, Seymour Cray est au travail et perfectionne sa dernière création, encore plus rapide et plus puissante.

Je rend hommage à l'esprit humain qui, même confronté à une limite physique absolue, permet à des gens comme Cray, et à des milliers d'ingénieurs moins célèbres, de continuer à repousser les limites de vitesse et de puissance de leurs machines.

Et ce n'est encore que le matériel I

Traduction et adaptation:
Password 90
Titre original : Heisenberg's uncertainty principle

# CYBER par Dave Small

Changement de sujet complet ce mois-ci, puisque Dave nous propose une nouvelle de science-fiction. Fans d'anticipation, réjouissez-vous! Et tous ceux qui ont déjà passé des heures à chasser des bogues récalcitrants sont invités à réfléchir aux hypothèses imaginées dans ce petit récit de fiction. Hypothèses? Hum... Et si... et si, à partir d'un certain niveau de complexité, les machines ne pouvaient plus être comprises selon les seules lois de la physique?...

#### **MESSAGE CONFIDENTIEL**

- Destinataire: John Waite, Conseiller
- ② **Expéditeur**: Axton Davies, chercheur adjoint, Groupe de Recherches Cyber
- ① Sujet : Ultra-secret
- ① Niveau de priorité : Urgent
- ② Routage : Porteur spécial, accusé de réception

Mon cher John,

Si je t'adresse ce message avec une telle priorité, c'est parce que je viens de faire une importante découverte. Découverte qui réclame ton attention immédiate et celle du Conseil.

Et dire que tu t'attendais sans doute à lire le bavardage distrayant de ton vieux copain! Je voudrais bien pouvoir papoter. Mais tu vois, si j'ai pris soin que tu lises ce message le plus tôt possible, ce n'est pas pour te faire part des derniers potins.

Comme tu le sais, le Conseil a financé nos recherches visant à produire un chasseur spatial autonome piloté par ordinateur, capable de se défendre contre les attaques des vaisseaux des Rebelles, voire de les anticiper. Tu sais également déjà que nos efforts ont porté leurs fruits : le vaisseau Cyber modèle 270 entrera en test opérationnel dans deux jours à peine.

Des tests préliminaires ont déjà été effectués sur les systèmes d'armes et l'intelligence artificielle du prototype, et se sont révélés très satisfaisants. Tu as probablement noté que l'opinion publique et les milieux militaires montrent un grand enthousiasme pour ce nouveau vaisseau-robot. Pour la première fois, nous allons disposer d'un appareil capable d'engager un Rebelle, à un contre un, avec une bonne probabilité de succès. Ce qui représentera une nette amélioration par rapport aux attaques de harcèlement de faible envergure que nous sommes encore contraints d'employer.

Mais cet enthousiasme pour le Cyber 270 est aussi la raison pour laquelle je ne t'envoie pas ce message par les canaux normaux, mais sur papier, par porteur spécial. Étant donnée mon poste de chercheur adjoint, mon courrier électronique ordinaire fait l'objet d'une surveillance de routine. L'opinion publique exige vengeance, et place tous ses espoirs dans le projet Cyber. Le sujet est presque devenu religieux, et un message ordinaire pourrait bien être effacé par mégarde, ou bien retenu ``pour vérifications".

Il faut que je te parle du travail que j'ai accompli et de ses conséquences pour le projet tout entier. Je vais m'efforcer d'éviter les termes trop techniques. Ne m'en veux pas si certaines explications te semblent superflues. Le sujet est si grave que je ne veux pas courir le moindre risque d'être mal compris, ou, pis, de sembler parvenir à des conclusions erronées.

Te souviens-tu de notre dernière année à l'université de Colo-Wy, il y a vingt ans ? J'ai poursuivi dans la voie scientifique et suis entré chez CSI. Toi, tu as bifurqué vers la politique, et je me suis souvent demandé si tu t'en tirerais. Maintenant, je ne me pose plus la question : tu es au Conseil, alors que moi, je fais toujours mumuse avec mes neutronoscopes

Ces saletés d'analgésiques qu'ils me donnent m'empêchent de rassembler mes souvenirs! Mais sans

eux, c'est pire, je ne peux aligner deux mots. As-tu jamais eu des migraines juste derrière les yeux ? Cela me fait tout le temps mal, j'y vois à peine.

Donc, quand j'ai débuté chez CSI, nous avons reçu une subvention pour le programme de recherches qui devint par la suite projet Cyber. Il s'agissait de concevoir une machine dotée d'une intelligence égale à celle d'un cerveau humain. L'accent était mis sur les capacités de "pensée associative" et de généralisation. Ces techniques se sont en effet révélées être les seuls à pouvoir battre régulièrement les vaisseaux-robots des Rebelles. L'ennemi nous surpasse apparemment toujours en termes de puissance de calcul brute.

Tu te souviens de l'arrêt du projet, qualifié d'"échec ruineux" ? Je ne sais pas si son arrêt brutal t'a mis la puce à l'oreille, ni si tu as eu la curiosité d'investiguer un peu et de chercher ce qui s'était réellement passé. Moi, je sais. Dieu fasse que les gens de la sécurité n'entendent jamais parler de ce message, car naturellement, cette histoire est si secrète que tu n'aurais sans doute jamais été autorisé à en prendre connaissance.

John, le projet n'a pas échoué. Nous avons au contraire fait une importante découverte: le cerveau humain ne peut pas fonctionner.

Non, je ne délire pas sous l'effet des drogues. Alors! ravale ce sourire condescendant! Je me souviens parfaitement de ce visage narquois que tu arborais pendant les cours de certains profs que tu n'appréciais guère. Laisse-moi t'expliquer, et tu comprendras pourquoi tout ceci n'a jamais été divulgué. "Sujet sensible", disaient-il avec un art consommé de la litote. Peut-être même regretteras-tu de savoir, c'est dire.

Prenons un grand ordinateur, comme nous l'avons fait. Programmons-y le nombre de neurones d'un cerveau humain moyen. La plupart de ces neurones sont d'ailleurs non connectés. Puis ajoutons-y le taux d'interconnexions entre neurones. Mélangeons avec la vitesse de propagation des signaux neuronaux, qui est à peu près celle du son dans l'air à la surface de la Terre (330 m/s). Toutes ces données figurent dans les manuels de neurologie, nous nous sommes contentés de les y piocher. Si j'ai l'air de me couvrir, c'est parce que les conclusions paraissent si extravagantes.

Maintenant, jetons ces chiffres en pâture à l'algorithme de Degaunse pour en déduire la puissance de calcul du réseau résultant. Son théorème est prouvé, donc les résultats sont indiscutables. Nous obtenons un résultat représentant la vitesse maximale de traitement de l'information du cerveau humain. C'est ce que nous avons fait.

Puis nous avons déterminé la vitesse du cerveau empiriquement. Nous avons pris un certain nombre de volontaires et les avons soumis à des épreuves de stress. A un certain moment de l'épreuve, les volontaires étaient amenés à croire que leur vie était en danger avant de perdre connaissance. La plupart ont réagi d'une manière classique, ce qu'un non-scientifique peut décrire comme "leur vie a défilé devant leurs yeux". Après qu'ils se soient remis de leurs émotions, nous leur avons demandé de nous décrire les souvenirs qu'ils avaient évoqués et les pensées qu'ils avaient eues durant les quelques secondes d'angoisse intense précédant leur perte de conscience.

Le nombre des souvenirs évoqués et leur niveau de détail était extraordinaire. D'énormes volumes de mémoire étaient balayés en quelques millièmes de secondes. Ensuite, nous avons pris la manière la plus efficace de coder ces informations et de les emmagasiner sous forme de réseaux neuronaux. Nous en avons déduit, en terme de vitesse de traitement, quelle était la rapidité d'accès à la mémoire humaine. Le taux que nous avons trouvé excède d,ailleurs très largement ce que nos meilleurs ordinateurs sont capables de faire en mode rafale.

Le programme donne les vitesses de transfert dans toute leur sécheresse. Permet-moi de te les traduire en mots. John, il s'en faut de très loin (d'un facteur de 2,61 fois 10 puissance 8) que le cerveau puisse accomplir ce qu'il fait ! Cette enchevêtrement de neurones ridiculement lents ne peut absolument pas penser, ou traiter les données, à la vitesse que nous connaissons. Nous avons tout vérifié deux fois. Et deux équipes travaillant avec des algorithmes différents ont abouti à ces mêmes conclusions. Si les puissances de traitement mesurée et théorique différaient d'un facteur 1 00, voire 1000, je soupçonnerait une erreur statistique ou bien de mauvaises données, mais pas avec une différence de deux cent soixante millions de fois !

Nos calculs semblent donc prouver que le cerveau n'est pas capable de pensée abstraite, mais

seulement de fonctions réflexes et de bas niveau (par exemple, si on touche une surface brûlante, on retire la main par un réflexe qui ne met pas en jeu le cerveau). En outre, même avec nos techniques médicales modernes, personne ne sait pourquoi un neurone choisit tel ou tel autre neurone comme destinataire d'un influx donné, parmi les milliers auxquels il est connecté par ses synapses. Certes, je pourrais te dire le transfert du signal fait intervenir de l'acétylcholine, de la sérotonine et des changement de polarisation, c'est-à-dire comment se fait le transfert. Mais je ne peux pas te dire pourquoi il se fait entre deux neurones donnés.

Nous avons donc abandonné le projet : nous ne pouvions pas construire un ordinateur qui pense aussi vite que le cerveau, parce que le cerveau ne peut pas penser!

J'avance une explication personnelle : le cerveau est un récepteur extrêmement sensible et délicat. La moindre influence externe affecte certains neurones qui envoient des stimuli selon certains chemins synaptiques, ce qui gouverne la pensée et permet d'agir sur les muscles. Une défaillance de ce système de communication conduit à ce qu'on appelle un dysfonctionnement cérébral.

Cette influence externe est inconnue et impossible à identifier. A défaut d'un mot mieux approprié, appelons-là l'âme. C'est la seule explication. Sans elle, le cerveau ne peut raisonnablement fonctionner. Et pourtant, nous avons été incapable de la détecter ou de l'isoler du cerveau, malgré des analyses poussées sur toute l'étendue du spectre électromagnétique.

Imagine un instant l'impact retentissant de la première preuve "scientifique" que l'âme existe. Imagine les guerres de religion qui pourraient en découler, et tu comprendras pourquoi le sujet a été abandonné et le projet enterré en toute hâte et dans le plus grand secret. Les guerres de religion ont toujours été les plus cruelles et les plus dévastatrices.

Pour m'occuper sur ce lit d'hôpital, je me demande souvent où se situent physiquement les ``âmes" qui communiquent avec les cerveaux-récepteurs. C'est pratiquement ma seule distraction.

Donc, notre tentative d'imiter le cerveau humain fut furtivement étouffée. L'équipe fut dissoute. Mes collègues et moi dûmes choisir entre changer de travail ou subir un effacement mémoriel sous hypnose. Changer d'activité ne me plaisait guère après dix ans de carrière, mais je ne voulais pas oublier que j'avais une âme. Abandonnerais-tu ce savoir ? (Beaucoup de mes amis l'ont fait. Tu te souviens de Kevet ? C'était un membre de l'équipe. inutile d'espérer qu'il s'en souvienne, l'effacement mémoriel est parfaitement au point.)

C'est ainsi que j'ai été transféré à la section du projet Cyber qui s'occupait de mettre au point les unités de mémoire (pour garder la mienne - ô ironie de l'administration !). Il y a cinq ans, j'ai commencé à développer la dernière génération de puces mémoires pour les vaisseaux Cyber. Les circuits que j'ai inventés stockent les données sous forme de résonances d'orbitales électroniques autour d'atomes individuels. Ces résonances peuvent être induites par des circuits de commande et sont détectables, ce qui permet d'écrire et de lire les données. Cette conception nouvelle nous permet d'atteindre une densité un billion de fois supérieure à celle de la technologie à base de transistors que nos utilisions jusque là - une technique vieillotte datant du XXe siècle. [NdT: un billion du système international vaut 10 puissance 12, ou un million de millions. Équivaut au trillion américain.]

Si nous n'avions pas été en guerre, mon invention de la mémoire à résonance orbitale m'aurait sans doute valu les honneurs d'un prix. Mais le secret autour de ce projet est tel que je ne serait sans doute jamais reconnu. Cela te prouve au moins que je ne suis pas complètement gâteux.

C'est cette densité de mémoire qui autorise les fonctions intelligentes et cognitives de la série Cyber. Sans cette percée, il aurait été impossible de faire tenir les programmes nécessaires dans un vaisseau spatial. En fait, nous n'aurions même pas pu bâtir une machine assez grande avec l'ancienne technologie, car la chaleur dégagée aurait détruit les puces. (Nous ne savons toujours pas quel type de mémoires les Rebelles utilisent, mais je soupçonne une variante de mes mémoires à résonance.)

Bien sûr, en accroissant la densité de données au millimètre carré, nous avons également accru la sensibilité aux rayons cosmiques. Ce n'est pas neuf. Dès la fin du XXe siècle, les ordinateurs personnels rustiques qui apparurent alors présentaient une erreur mémoire par jour en moyenne à cause des interférences dues aux rayons cosmiques, surtout en haute altitude, là où la couche atmosphérique plus mince absorbe moins les rayons. Par conséquent, les recherches se concentrèrent bientôt sur les "parasites" engendrés dans les mémoires par les rayons cosmiques. Certaines

particules ont une si grande énergie qu'il était très difficile de prémunir totalement les circuits en les blindant. Tout ce que pouvaient faire les concepteurs du XXe, c'était détecter, et, si possible, corriger les erreurs engendrées par ces particules. Ils inventèrent entre autres le système CESDED (Correction des Erreurs Simples, Détection des Erreurs Doubles) pour détecter les erreurs de parité portant sur un ou deux bits et les corriger automatiquement s'il ne s'agissait que d'un seul bit inversé.

De nos jours, nous utilisons toujours les circuits CESDED. Et les rayons cosmiques sont un problème bien plus grave qu'en ces temps héroïques! Les mémoires de mon invention y sont si sensibles que des blocs de données entiers sont effacés par une seule particule là où un seul bit était affecté auparavant.

L'an dernier, j'ai entamé des recherches sur le blindage contre les rayons cosmiques. Elles ont abouti à la catastrophe du Cyber 240. Je suis sûr que tu t'en souviens. Je t'ai même vu à la LD pendant un reportage sur les auditions que tenait le Conseil sur cette affaire. Les faits sont là : nous avions installé un ordinateur Cyber 240 aux commandes d'une navette de maintenance. Il a émis un ordre qui a dépressurisé la navette et tué les nombreux techniciens qui étaient à bord. Nous avons par la suite découvert que la cause en était des interférences dues à des rayons cosmiques qui avaient affecté les mémoires des systèmes de maintien de la vie et de pressurisation. Les interférences avaient aléatoirement créé une séquence d'instruction qui avaient ouvert un sas, en outrepassant les sécurités.

Nous n'avions jamais testé le 240 configuré avec sa mémoire maximale, et nous ne savions pas quel taux d'erreurs mémoire nous pourrions avoir. En fait, nos estimations prédisaient un taux d'erreurs mille fois plus faible que celui observé lors de cette catastrophe. De deux choses l'une : soit les nouvelles mémoires souffraient d'une erreur de conception, soit elles étaient bien plus sensibles aux rayons cosmiques que je le pensais. Il me fallait savoir.

J'ai contacté le Dr Anderson, de New Stockholm, pour qu'il me fournisse un blindage à métal compacté. La trouvaille d'Anderson est très intéressante. Il commence par tasser les atomes de métal les uns contre les autres grâce à un puissant champ électrique. Les atomes atteignent presque la densité de la matière qui a subi un effondrement gravitationnel. Les vides entre les noyaux des atomes sont considérablement réduits, et les particules cosmiques ont une très faible probabilité de passer entre les noyaux, ce qui fait de ce matériau un bon blindage. Naturellement, les atomes reprennent très vite leurs positions initiales et libèrent de l'énergie. Mais le système d'Anderson récupère cette énergie et la réutilise pour recompresser les atomes, ce qui commence un nouveau cycle compression-expansion. On obtient donc un bouclier qui n'est efficace que pendant un demi-cycle. Pendant 50% du temps, il arrête ou réfléchit très efficacement les rayons cosmiques à haute énergie (dans le spectre des rayons X ou gamma). Si on prend deux de ces boucliers et qu'on les met en opposition de phase, l'un des deux est toujours en phase compressée, ce qui fournit une protection permanente quasi totale. C'est ce que nous avons fait.

Dans notre labo de Great Falls, nous installâmes un double blindage d'Anderson dans lequel fut placé un nouveau Cyber 270 doté d'une petite mémoire à résonance d'un téraoctet. Nous y fîmes tourner des programmes de test, et, à notre grande satisfaction, nous constatâmes que le niveau d'interférences était ramené à un niveau acceptable.

Mais les interférences avaient un aspect organisé.

Il se formait des arrangements qui n'avaient rien d'aléatoire. On aurait dit une interférence délibérée avec les cellules mémoire. Ce phénomène avait toujours été là, mais il avait jusque là été noyé dans le bruit de fond des rayons cosmiques, comme un murmure recouvert par le grondement d'une cascade. Notre première hypothèse fut que ce phénomène provenaient d'une diaphonie entre cellules mémoires voisines. Nous décidâmes de faire passer un test au Cyber 270. Le test consistait à faire tourner le programme de poursuite SCIP-1 2, qui détecte les petits objets dans l'espace, prédit leur trajectoire, et les détruit à l'aide d'un petit laser (c'est le moyen de défense classique contre les missiles à fusion Shon des Rebelles).

John, ce que je vais décrire à présent pourrait fort bien être une hallucination. Je le dis pour te faire comprendre que cela m'a fait douter de ma santé mentale. Je suis persuadé que les gens sains d'esprit sont ceux qui sont capable d'en douter, et que se croire absolument inébranlable est un signe qu'on bascule dans la folie.

Le programme SCIP, que nous utilisions comme test, détecte les objets mobiles à l'aide de radars et de

capteurs visuels. Il dirige alors une tourelle laser vers la position prédite de l'objet, et tire une salve pour le détruire.

J'avais pris place à l'holoconsole de surveillance mémoire dans le labo. Mike et Angela étaient avec moi ce jour-là. Déjà, à l'université, nous étions dans la même équipe de travaux pratiques. Nous avions travaillé ensemble pendant tant d'années que nous pouvions presque deviner nos pensées. Mike envoyait des objets dans le champ de tir du laser, et Angela était à la console de programmation.

Mon holoconsole montrait un diagramme fonctionnel en trois dimensions de la mémoire du Cyber 270, méthode classique de débogage. Quand une erreur de parité survenait dans la mémoire, un pixel s,illuminait brièvement à l'endroit concerné. J'espérais pouvoir reproduire le phénomène, déceler une conformation pouvant nous fournir un indice. Nous désespérions de trouver la cause de l'interférence organisée, et parfois, un examen visuel révèle ce qu'une analyse informatisée a laissé passer. Je ne voyais que le diagramme en vert sombre, et un petit éclair blanc occasionnel là où une cellule mémoire avait encaissé un parasite.

Mike commença à lancer des projectiles en tôle avec le canon à accélérateur magnétique, et Angela fit démarrer le programme. Pendant environ cinq minutes, tout marcha bien. Les éclairs blancs ne révélaient que des parasites aléatoires que le CESDED corrigeait. Le phénomène ne se reproduisait pas. Sur ma console, les formes géométriques symbolisant les structures de données se déformaient, se déplaçaient et évoluaient au rythme normal du programme SCIP-12.

Et soudain, la console entière s'illumina de blanc. La mémoire commença à se modifier sous mes yeux, d'une manière sciemment organisée. Je vis des blocs se déplacer et des changements s'opérer. C'était une réorganisation intelligente de la mémoire. Un peu comme un programme de tri.

J'essayai de déceler un indice, de voir si des blocs adjacents interféraient par fuite de courant. J'avais enfin reproduit mon phénomène, et je l'examinais intensément. C'était le but du test. C'est pourquoi je n'ai pas coupé immédiatement le courant. Et c'est parce que je n'ai pas réfléchi que c'est arrivé.

Le laser cessa de suivre la trajectoire de la cible en tôle que Mike venait de propulser dans son champ de tir. La tourelle du laser pivota à fond vers la droite, et arriva en arrêt sur ses butées de sécurité. J'entendis gémir les moteurs selsyn qui continuaient à forcer, et les butées cassèrent presque instantanément. La tourelle pivota encore, braquée à présent vers l'intérieur du labo. Et le laser tira à pleine puissance. Pas le petit rayon faiblard qui servait à détruire les cibles, mais un éblouissant cylindre massif de lumière jaune. Les lumières baissèrent tant le laser consommait d'énergie.

John, le laser perfora littéralement le cœur de Mike. Je sais que c'est horrible, mais il faut que j'en parle à guelqu'un. Et tu es le seul ami qui me reste.

Je suis resté figé d'horreur, je l'avoue. Les yeux de Mike se fixèrent sur les miens, son regard se riva au mien comme pour se retenir, me lançant un appel muet. Et je ne pouvais rien faire d'autre que le regarder. Puis il est tombé. La tourelle pivota et revint vers la gauche, brisant l'autre butée de sécurité. Angela n'avait pas été aussi lente que moi. Peut-être grâce à l'escrime qu'elle pratiquait au lycée ? Elle avait fait basculer en arrière la chaise sur laquelle elle se trouvait, pour tenter de s'abriter derrière la console de programmation. La chaise n'avait pas terminé sa chute lorsque le laser atteignit Angela au travers de la console.

Tout ceci prit peut-être une seconde. Aurais-je eu le temps de réagir ? Je me demande encore. Je n'en sais rien. Le psy me dit que je m'accuse. Mais je suis sûr qu'à ma place, Angela ou toi ne seriez pas restés plantés là' inertes. Il dit aussi que n'importe qui aurait hésité, aurait été choqué par l'irruption de ce spectacle atroce dans la vie monotone du labo, surtout pendant un test de routine. Selon lui, même des soldats surentraînés, gonflés à bloc, en situation de combat, restent un instant paralysés lorsqu'ils voient quelqu'un mourir. Peut-être bien. Mais j'avais le temps de couper le courant et je ne l'ai pas fait.

La tourelle pivota de nouveau et s'aligna instantanément vers moi. Une lumière du jaune le plus pur que j'ai jamais vu m'aveugla. C'était un jaune magnifique, comme un feu. Je le vois encore. Des tâches vertes (la couleur complémentaire) me brouillent parfois la vue là où ma rétine est brûlée, et il me faut déplacer le regard pour voir les mots sur cette feuille entre les lésions rétiniennes.

Soudain, tout s'arrêta. La chaise d'Angela avait rebondi et avait heurté les bornes de connexion au 220 V du laser. Nous ne les avions pas isolées parce que c'était juste un test, et il faut si souvent refaire les câblages pour venir à bout des boucles de masse... Le cadre métallique de la chaise a donc heurté les

bornes, et dans une gerbe d'étincelles, les disjoncteurs magnétiques ont coupé les circuits. Le courant du laser a été coupé juste alors qu'il venait de me prendre pour cible, et je n'ai été atteint que pendant quelques centaines de microsecondes, pas assez pour me tuer.

Et tout fut soudain silencieux. Le seul bruit était le cliquètement du capot du laser dont la tôle refroidissait. C'est de l'odeur que je me souviens le mieux. Une odeur de chair brûlée. Celle de mon front. J'ai senti cette odeur pendant des jours. Le laser avait tenté de me perforer le front, et s'il avait disposé d'une fraction de seconde supplémentaire, il y serait parvenu. Angela m'avait sauvé.

John, je sais que tu es au courant. Les rapports officiels parlent d'une erreur de programmation accompagnée de défaillances matérielles, et disent que j'ai perdu les pédales en voyant de vieux amis mourir. Ils affirment que j'essaie de trouver une raison à une erreur de programmation qui a tué un ami et une ex-compagne.

Mais le laser n'a pas tiré "au hasard", pour reprendre les termes de ce cher colonel Seiers. Il a tiré exactement trois fois, et les trois fois, la visée était parfaitement alignée. Quelque chose s'est emparé de ce laser et l'a utilisé pour tuer trois personnes. Le laser est sorti de son champ de tir programmé, la tourelle a brisé ses butées de sécurité, et le tube laser lui-même est allé bien au-delà de sa limite de puissance programmée. C'est délibérément que ce ``quelque chose" a tué deux personnes, et ne m'a manqué que par chance.

La mémoire s'est effacée lorsque le courant a disjoncté, et je ne peux donc pas prouver les changements que j'avais remarqués. Nous n'avons pas non plus d'enregistrements 3DV de la console - nous n'allions pas brancher des enregistreurs pour un simple test de la mémoire.

Mais réfléchis. S'il y a réellement des âmes qui communiquent avec nos cerveaux et commandent nos corps, autant en accepter toutes les implications...

Il pourrait également y avoir des démons. Je pense que les Rebelles sont des démons, John.

Et ces démons n'ont besoin que d'une mémoire suffisamment sensible pour qu'ils puissent l'influencer, la contrôler. Les vieilles mémoires à transistors y étaient quasiment insensibles. Tout au plus pouvaient-ils sans doute y causer des altérations ponctuelles, mais elles n'étaient pas assez sensibles pour qu'ils y causent des perturbations organisées. Pour les informaticiens, cela devait au pire donner un de ces jours que nous avons tous connus, où l'on regrette de s'être levé et où rien ne marche dans les ordinateurs, sans aucune raison apparente.

Mais les nouvelles mémoires à résonances orbitales sont si délicates qu'elles peuvent être influencées de façon cohérente. En les abritant derrière un excellent blindage, nous supprimons presque tous les parasites aléatoires d'origine cosmique qui pourraient venir couvrir l'influence de ces "choses". Celles-ci disposent alors d'un excellent récepteur à démons. Et on offre à quelque chose de très malfaisant la possibilité d'utiliser les appareils pilotés par l'ordinateur contenant ces mémoires.

Un Rebelle pourrait très bien être le résultat d'une mémoire trop réceptive livrée en pâture à une influence externe inconnue. Ce pourrait être le produit des recherches informatiques de n'importe quelle civilisation avancée. Cela me semble plus que probable.

Bientôt, je serai aveugle. Les toubibs me disent que mes rétines saignent maintenant sans arrêt et se détériorent progressivement. Le laser a endommagé tous les vaisseaux sanguins rétiniens et on ne sait pas les réparer. Je dois aujourd'hui subir une autre opération. Ils pensent que des vaisseaux sanguins proches de mon cerveau ont également été endommagés par la chaleur. Ils disent que le choc opératoire pourrait les faire éclater et qu'en ce cas, je serais mort avant qu'ils puissent faire quoi que ce soit.

Raison de plus pour t'avertir tout de suite. Près de cette station, au point de Lagrange, se trouve le centre spatial de test qui procèdera aux essais du Cyber 270. Cet engin possède les meilleurs systèmes d'armes que nous ayons jamais construits, et est piloté par une intelligence artificielle dotée de 4000 téraoctets de mémoire à résonance. Et cette mémoire est protégée par un double blindage d'Anderson. Comme chacun sait, c'est notre "meilleure arme défensive", notre "meilleur ordinateur jamais construit". J'ai vérifié sa conception. Avec ses alimentations redondantes, il n'y a aucun moyen de le débrancher. Et il commence ses tests dans deux jours.

John... Je me demande vraiment de quel côté se battra ce vaisseau.

Eh bien, ils ont amené le chariot. Je dois dormir à présent.

Amicalement,

Axton.

Traduction et adaptation: Password90

### Les quatre lois de Small 2- La loi de l'effort maximum

Si vous vous êtes déjà approché d'un ordinateur, ne serait-ce que de loin et avec des gants, vous devez avoir remarqué que l'informatique semble régie par une loi fondamentale. Les anglicistes et les snobs la baptisent parfois Loi de Murphy, mais en bon français, il s'agit de la LEM (Loi de l'Emm... Maximum). Dave nous propose ce mois-ci la première partie d'une étude de LEMologie, science de l'étude de la LEM, dans laquelle il est passé maître. La LEM a quatre conséquences principales, qui ont été identifiées et formalisées par Dave, sous le nom des Quatre Lois de Small. Croyez-en son expérience...

#### Les quatre lois

J'ai passé au moins 60% de mon existence à m'occuper d'ordinateurs. En fait, comme mes cinq ou dix premières années n'ont pas été informatisées, elles font baisser la moyenne : actuellement, c'est plutôt 75% de mon temps que je consacre à l'informatique. Durant tout ce temps, j'ai remarqué que ces machines ont une certaine propension à n'en faire qu'à leur tête, et de ces observations sont nées les Première, Seconde, Troisième et Quatrième Lois de Small.

En bon bidouilleur, je voudrais partager l'information et vous épargner d'avoir à les redécouvrir par de tragiques expériences où vous pourriez perdre votre candeur. Si vous voulez bien me suivre...

Première loi de Small (dite "Loi de l'Effort Maximum")

"Quelle que soit la façon dont vous vous y prenez pour résoudre un problème donné, il vous faudra faire le plus grand effort possible"

Permettez-moi d'illustrer cet énoncé de quelques exemples. Naturellement, je vous fais confiance pour en fournir plein d'autres quand vous aurez vu où je veux en venir.

Supposons que vous utilisiez OUTIL 1.0, un programme qui accomplit une tâche donnée. Un beau jour, OUTIL 2.0 sort, et la publicité affirme qu'il est cinq fois plus rapide. Vous avez un projet qui nécessite l'emploi d'OUTIL, version 1.0 ou 2.0.

En achetant la mise à jour, vous espérez donc pouvoir effectuer ce travail 5 fois plus rapidement, libérant ainsi 80% de ce temps pour d'autres tâches - que vous pourrez consacrer à faire avancer vos travaux informatiques, ou à votre famille et vos amis.

Mais la Loi de l'Effort Maximum est là pour anéantir tout espoir concernant ces 80% de temps libre, quoi que vous fassiez. Il se produira quelque catastrophe. J'en ai fait la cruelle expérience tant de fois que c'en est devenu une règle pour moi (je m'attends à l'inévitable, mais ne croyez pas que j'aime ça !).

J'en vois qui hochent la tête en disant : "Ce pauvre Dave a été traumatisé par ses bogues, il est trop pessimiste." Hélas non ! Par exemple...

#### Catastrophe de type 1

OUTIL 2.0 aura fatalement quelque incompatibilité avec la version 1.0. Tenez, ce boque bien connu de

la 1.0, vous aviez appris à le contourner, et vous aviez mis au point une bidouille pour l'éviter. Eh bien, le bogue a été corrigé, et désormais, votre bidouille plante. Ou encore, alléché par les nouvelles fonctionnalités de la 2.0, vous vous hâtez d'en tirer parti. Mais comme tout nouveau code, ces fonctionnalités ont introduit de nouveaux bogues, qu'il vous faudra découvrir et apprendre à contourner, au prix d'un temps considérable. C'est la raison pour laquelle je m'en tiens obstinément à un seul jeu d'outils, tout en tenant une liste des bogues identifiés pour chaque outil. Ce n'est qu'en maugréant et en traînant des pieds que je change d'outil.

Ce type-là de catastrophe m'est arrivé de nombreuses fois, et j'ai un wagon entier d'histoires vécues pour le prouver.

#### **AS68**

Tenez, essayez donc de mettre deux labels sans opération entre eux sur deux lignes consécutives en AS68, l'assembleur livré avec le kit de développement original d'Atari dès 1985. C'est parfaitement licite. Vous pouvez par exemple avoir besoin de mettre un label à une routine dont le début est aussi celui d'une boucle, et vous avez deux labels ROUTINE1 et BOUCLE consécutifs.

Eh bien, AS68 n'aime pas cela. Mais attention, il ne va pas cracher un message d'erreur, que nenni, ce serait trop facile! Non, ce que va faire AS68, c'est décaler de deux octets chaque adresse de saut relatif assemblée après ces deux lignes (une adresse de saut relatif est en fait un nombre d'octets en plus ou en moins à sauter dans le code pour atteindre la destination du branchement). Vous avez bien lu : chaque adresse stockée dans le programme exécutable sera fausse à partir de ce point! Le décalage est cumulatif. Si vous avez à nouveau un double label plus loin dans le code, le décalage sera de 4 octets, puis de 6, et ainsi de suite, à chaque BRA ( branchement), JMP (saut) ou JSR (appel de sous-programme). Et il est très difficile de s'en apercevoir, parce qu'on soupçonne son code ou sa machine avant de penser qu'un assembleur est capable de bêtises pareilles. Cela produit des plantages vraiment curieux. J'ai ainsi découvert, à mon grand dam, que le second mot d'une instruction en langage machine 68000 est habituellement une donnée. Quand un saut aboutit sur celle-ci, on obtient un plantage "instruction illégale"... dans le meilleur des cas. Le pire survient lorsque la donnée, par un malheureux hasard, est interprétée comme une instruction valide, ce qui fera planter le programme plus loin, lançant le malheureux programmeur sur une fausse piste.

Et après "La vengeance du double label", voici "La malédiction de l'assemblage conditionnel". l'assemblage conditionnel consiste à prendre en compte ou non certaines parties du code d'après la valeur de certains symboles. Eh bien, lui aussi provoque le fameux décalage de deux octets des sauts. Nous continuons notre festival des horreurs avec "L'écraseur de fichiers fou contre- attaque". Il est 5 h du matin... La nuit sans lune n'est éclairée que par la lampe du bureau d'un pauvre programmeur (la caméra s'approche lentement de celui-ci, vu de dos)... La future victime, fatiguée, a un instant d'absence et tape innocemment AS68 -L MONPROG au lieu de AS68 -L MONPROG.S. Froidement, AS68 écrit son fichier de sortie par dessus le code source, le fameux fichier de suffixe ".S" amoureusement peaufiné, l'écrasant irrémédiablement ! (Violons suraigus, la victime s'arrache les cheveux et s'effondre en sanglotant...) Ahurissant, non ? J'ai massacré mon code source un bon nombre de fois, perdant à chaque fois des heures de travail, et je suis devenu totalement paranoïaque quant à mes sauvegardes. Dan Moore a finalement résolu le problème par un petit utilitaire appelé à la place de AS68. Si le nom du fichier ne se termine pas par ".S", tout s'arrête, sinon, AS68 est effectivement exécuté. Cela m'a sauvé maintes fois.

Il y a encore d'autres histoires d'horreur sur AS68, LO68 et RELMOD, les outils du kit de développement initial d'Atari, mais je vous les épargnerai, sinon, il faudra rebaptiser ce journal "Freddy Krueger Magazine".

#### Même avec des bogues...

Et pourtant, j'ai utilisé ces outils à longueur de journée, que ce soit pour écrire Spectre ou des

programmes publiés dans le magazine américain START. Car je connaissais leurs bogues. Avant de commencer un nouveau projet, je relisais rapidement la liste de bogues et me remémorais qu'il fallait éviter telle et telle manip.

J'ai laissé tomber AS68, qui était abominablement lent, en faveur de DevPak ST et de DevPak TT de la firme anglaise HiSoft. Malheureusement, HiSoft change sans arrêt de distributeurs, ce qui fait que j'ignore à qui vous pouvez les commander à l'heure où vous lisez cette saga. Si vous cherchez un assembleur et savez où vous procurer DevPak, n'hésitez pas, c'est un bon produit. Rapide de surcroît : il laisse sur place AS68. Mais il m'a fallu pas mal de temps pour convertir mes sources du format AS68 au format DevPak (il n'v a hélas pas de format standard pour le code assembleur). Par exemple, la déclaration l'une variable longue d'un mot s'écrit "VAR1 .dc.w \$0" en AS68, mais "VAR1 dc.w \$0" en DevPak. Notez la différence, le point devant le mot-clé "dc". Cela n'a l'air de rien, et faire cette modification est d'ailleurs automatisable par un Cherche/Remplace, mais il y a d'autres différences plus ou moins importantes. Et dans un programme de 25000 lignes comme Spectre, ces modifications mineures, mises bout à bout, m'ont pris plusieurs jours ! En passant à DevPak, nous avons dû abandonner LO68 et Relmod pour ALN, le nouvel éditeur de liens d'Atari. ALN est beaucoup plus rapide que les programmes qu'il remplace, mais très délicat à paramétrer. Il nous a fallu un temps fou pour trouver la bonne combinaison d'options et de paramètres afin qu'ALN se comporte comme nous le souhaitions. Niveau de débogage, sensibilité aux majuscules/minuscules, longueur des labels... Il devait y avoir 64 combinaisons possibles, c'est naturellement la 64ème qui était la bonne.

Ce fut donc une parfaite illustration de la Première Loi de Small. DevPak nous a sans doute fait gagner un facteur 10 en temps d'assemblage, mais ce gain de temps a été totalement anéanti par un temps de débogage et de paramétrage dix fois plus long.

#### Catastrophe de type 2

Reprenons notre utilitaire OUTIL. Admettons que la version 2.0 soit parfaitement compatible avec la 1 . 0 (ce que, personnellement, je n'ai jamais vu). Eh bien, un autre type de catastrophe se produira. C'est obligatoire : il se produira fatalement quelque chose pour vous faire utiliser ce temps que vous pensiez pouvoir gagner. Tenez, une fois, Dan Moore a cru pouvoir gagner du temps sur la maintenance du code de Spectre en en réécrivant une partie en C ce qui fut long et pénible. Hélas, lorsqu'une structure contenait des déclarations de variables longues d'un octet, le compilateur alignait celle-ci sur les mots. Autrement dit, il insérait des zéros dans la structure entre les octets pour aligner ceux-ci sur des adresses paires. Inutile de dire que cela faisait planter tout ce qui utilisait ces structures. Dan n'a pas pu trouver un moyen d'éviter cet alignement automatique, et a dû se résoudre à abandonner ce compilateur.

Et si ce n'est pas un outil qui vous fait faux bond, c'est le système d'exploitation. En transférant un fichier source de 600000 octets entre un lecteur de disquettes externe Toshiba 2000SX et le ST, les 51 2 premiers octets du fichiers sont répétés ad nauseam jusqu'à remplir les 600 Ko. Inutile de dire qu'en éditant le fichier, on a la surprise de sa vie. C'est un bogue qui n'est toujours pas corrigé en TOS 1 .4. Je me suis laissé dire que le problème provient des tables d'allocations de fichiers (FAT) d'un kilo-octet de ces lecteurs externes, que le TOS ne supporte pas. Il lui faut des FAT de 2 Ko. Le format du Toshiba n'a donc pas l'heur de plaire au TOS, que ce soit en simple ou en double face. Il faut savoir que le DOS, la partie du TOS qui gère les disquettes, a été réécrit, parce que l'ancienne version devenait affreusement lente lors de la création d'un fichier sur un disque dur presque plein... entre autres bogues. Et cette nouvelle version, au lieu de lire sur le disque la taille de la FAT, suppose bêtement que la FAT fait toujours 2 Ko. Je sais même qui a fait cette bourde (mais il ne travaille plus chez Atari). Merci, gars, c'est sympa, on se souviendra de toi.

Croyez-moi, il se produira fatalement quelque chose. Et encore, si c'est un bon gros cataclysme bien franc, qui anéantit proprement tous vos espoirs, estimez-vous heureux. Parce que dans le pire des cas, vous aurez en fait droit à une multitude de petits incidents insignifiants. Vous serez submergés par une infinité d'anicroches qui, prises une à une, n'ont rien d'effrayant, mais qui vous pourriront la vie en survenant toutes en même temps. Comme devoir renommer 36 sous-répertoires pour faire plaisir à un utilitaire. Ou ajouter une ligne de directives d'édition de lien à chacun des 58 fichiers de votre code. Cela prend un temps fou, comme j'ai pu m'en rendre compte : c'est exactement ce que j'ai dû faire pour

#### Spectre 3.1.

Et finalement, tout le temps que vous aviez cru pouvoir économiser avec OUTIL 2.0 sera gaspillé en préparations et corrections pour adapter la nouvelle version à vos besoins. Conclusion : au lieu de vous jeter sur la version 2.0, vous feriez aussi bien d'aller jouer au golf.

La Loi de l'Effort Maximum marche aussi dans les activités non informatiques. En cherchant bien, je suis persuadé que vous vous souviendrez de moult occasions où vous auriez pu gagner du temps si... si un minuscule incident n'était pas survenu. Vous étudiez soigneusement un itinéraire et découvrez qu'on peut emprunter un raccourci ? Paf! Une déviation ou un accident viennent l'obstruer. Ça marche à tous les coups. La possibilité de trouver un raccourci dégagé n'existe pas, sauf dans les films.

#### Précédent quantique

A mon grand regret, je dois vous dire qu'il existe une raison physique sous-jacente à cette folie apparente, et que tout espoir d'échapper à cette loi est donc vain. Cette raison, on la trouve dans la physique quantique, qui remonte à 1 926. Il s'agit l'un ensemble de lois régissant le comportement des particules subatomiques, et expliquant des événements incompréhensibles pour la physique newtonienne. Et vous allez voir que la physique quantique s'accorde parfaitement avec la Première Loi de Small.

Tout d'abord, ne croyez pas que seul les savants atomistes soient concernés par la physique quantique. Les transistors, briques de base des circuits intégrés de nos Atari (entre autres), fonctionnent selon ses principes. Et de nombreux physiciens de haut niveau pensent que la physique quantique ne se contente pas de décrire ce qui se passe réellement au plus profond de la matière, mais qu'elle décrit aussi notre univers quotidien.

Petit exemple : dans tout le Colorado, on peut ramasser des rochers de pechblende, qui contient de l'uranium naturel. De temps à autre, un de ces atomes fissionne, c'est-à-dire qu'il se casse en deux tout en émettant deux neutrons (ou plus). Comme l'uranium n'est pas concentré, il ne se produit nulle réaction en chaîne ou explosion. Mais quel est l'atome qui va éclater ? On ne peut pas le prédire. Et tout semble indiquer que c'est impossible.

Mais... Et là, je vais vous demander de me croire sur parole (et d'aller lire un bouquin d'initiation à la physique quantique, qui est ce qui se rapproche le plus de la magie)...

Ce qui se produit est totalement différent selon que cette fission est observée ou non. l'observation pouvant d'ailleurs être soit directe (par détection des neutrons émis grâce à un compteur Geiger), soit indirecte par détection de phénomènes induits par cet événement. Selon que l'événement quantique (la fission) est observé ou non, il y aura littéralement deux histoires possibles pour cet atome ! Ces deux histoires surviennent en parallèle dans le temps. Dans l'une, l'atome fissionne, dans l'autre, il reste intact. Chacune de ces histoires forme un "continuum d'espace- temps", ce qui signifie "région d'espace dans un temps donné". Il n'y a rien de sorcier à ce sujet, même si la science-fiction abuse du terme. Mais la Nature ne fait son choix entre ces deux continuums possibles que si l'événement est observé ! Sinon, elle s'en contrefiche, et les deux possibilités coexistent. Car la Nature fait rarement plus que le strict nécessaire (tout comme moi d'ailleurs). Et si un observateur est présent et voit l'atome éclater (ou non), Maman Nature va "effacer" le mauvais continuum (celui qui correspond à l'événement contraire), qui cessera d'exister. Mais seulement dans ce cas.

Ce concept l'histoire parallèles a été développé par Richard Feynman, brillant physicien plein d'humour et prix Nobel de physique [NdT : Feynman est mort en 1989.] C'est également lui qui a compris pourquoi la navette Challenger a explosé. Il faisait partie de la commission d'enquête. Durant une réunion, il a pris un échantillon du matériau dont étaient faits les joints toriques des boosters à poudre de la navette, et l'a laissé tremper dans de l'eau avec des glaçons durant la pause de midi. Il a pu alors montrer que le matériau du joint n'était plus souple, mais au contraire dur et cassant, ce qui explique la défaillance du joint le jour du lancement, jour où, précisément, il gelait. Bref, ce n'est pas un idiot.

C'est donc avec la caution d'un éminent scientifique que j'ai formulé ma Première Loi. Il y a deux continuums d'espace-temps, l'un dans lequel vous ne prenez pas le raccourci et où il est dégagé, l'autre dans lequel vous prenez le raccourci et où il est barré par une déviation. Vous en choisissez un et l'autre cesse d'exister. Pas d'autre possibilité. Certes, c'est un peu déprimant, mais c'est ainsi que marche la Nature...

Dans n'importe quelle réunion, on peut entendre des histoires prouvant la Loi de l'Effort Maximum, venant en général de gens qui ont essayé une astuce quelconque pour gagner du temps et ont échoué pour quelque raison bizarre. Je l'affirme haut et fort : ce n'est pas là l'effet d'un mauvais sort, mais bien la façon dont les choses se passe naturellement. Par analogie, considérons à nouveau un rocher de pechblende. Il est impossible de savoir quel atome d'uranium va fissionner à un instant donné, mais en revanche, en connaissant la quantité l'uranium dans le rocher, on peut statistiquement prédire, avec une grande précision, combien de désintégrations atomiques se produiront. l'incertitude quantique débouche sur une certitude statistique : un certain nombre de fissions doit avoir lieu. Eh bien, en informatique, c'est pareil. Vous ne savez pas précisément quelles calamités vont s'abattre sur vous, mais c'est mathématique, elles vont survenir. Le temps que vous comptiez gagner doit être dépensé.

#### Bidouilleur subatomique

Pour rester encore un peu dans ce domaine fascinant, permettez-moi de revenir sur un postulat majeur de la physique moderne, à savoir que rien ne peut se déplacer plus vite que la lumière, y compris l'information. C'est précisément ce qui pousse les ingénieurs à miniaturiser toujours davantage les circuits pour pouvoir déplacer davantage d'information dans des temps de plus en plus brefs. Depuis qu'Einstein l'a formulé en 1915, ce postulat a fini par être universellement accepté. Mais peut-être est-ce aller un peu vite en besogne. Steven Hawking, dans son nouveau livre (excellent !) intitulé "Trous noirs et bébés- univers", dit que les particules plus rapides que la lumière sont probablement nécessaire pour que l'univers fonctionne correctement ! Attention, il ne s'agit pas d'un mystique du Nouvel Age, mais de Steven Hawking, l'homme qui a récupéré la théorie du Big Bang dans la poubelle où ses contradictions l'avaient amenée, et l'a si bien peaufinée qu'elle est à présent presque universellement admise. C'est aussi l'auteur d",Une brève histoire du temps", un incroyable best-seller expliquant comment l'on a pu déterminer l'état de l'univers une infime fraction de seconde après le Big Bang.

L'argument d'Hawking est simple. Comme chacun sait, les trous noirs sont des étoiles qui se sont effondrées sur elles-mêmes, concentrant toute leur masse sur un infime diamètre, ce qui engendre une gravité si intense que rien, même pas la lumière, ne peut s'en échapper. De nombreux vulgarisateurs ont présenté les trous noirs comme le destin ultime de l'univers, car ces astres dévorent toute la matière qui passe à leur portée. (Et il semble que dans la constellation du Cygne, nous en ayons un bon exemple). Mais, souligne Hawking, une particule entrant dans un trou noir a en fait deux histoires distinctes (ça ne vous rappelle rien ?). Car pour une telle particule, le temps ralenti de plus en plus jusqu'à s'arrêter quand elle entre dans le trou noir, en raison de l'intense champ gravitationnel qui y règne. Mais alors, elle viole le principe d'incertitude de Heisenberg, dont j'ai parlé dans un précédent article. [NdT: voir ST-Mag n° 78.] En effet, si le temps se fige pour cette particule, on peut prédire à la fois sa position et le moment où elle l'occupera. Or, le principe d'Heisenberg a été maintes fois validé, et rien n'indique qu'il cesse d'être valable dans ces conditions. Donc, selon Hawking, une particule se sent moralement obligée de sortir du trou noir tôt ou tard, et la seule façon d'y parvenir est d'aller plus vite que la lumière. Cela implique qu'un trou noir laisse graduellement s'échapper des particules, perdant progressivement de sa masse... ce qui peut équilibrer la matière absorbée par l'astre. Les trous noirs ne sont donc peut-être pas le destin ultime de l'univers.

C'est cette bidouille mathématique d'Hawking, simple mais géniale, qui fait de lui LE bidouilleur subatomique, par excellence [NdT : en français dans le texte.]

Cela pourrait signifier que le siècle qui a vu naître la limite absolue de la vitesse de la lumière pourrait bien aussi voir cette limite s'évanouir. Les astronomes et les physiciens du monde entier discutent fébrilement cette théorie dont les implications sont ahurissantes.

#### Voilà, voilà

Voilà donc ma Loi de l'Effort Maximum. Et c'est sans grand effort qu'on en trouve des exemples dans la

vie de tous les jours. Je me suis contentée de lui trouver un nom, je ne prétends pas l'avoir découverte. J'espère que vous êtes soulagé, maintenant que vous savez qu'il est impossible d'y échapper!

Traduction et adaptation: Password90

## Les quatre lois de Small 1- La loi du chaos grandissant

Les ordinateurs sont l'image même de la rigueur, n'est-ce pas ? De fidèles serviteurs suivant à la lettre les instructions qui leur sont fournies. L'image qu'un informaticien aimerait pouvoir donner de son métier est celle d'un horloger mouvant d'invisible rouages logiciels dans un ordre et une précision absolue... Alors que dans la réalité, l'informaticien se sent plutôt dans la peau du directeur de l'annexe psychiatrique d'un zoo! Pourquoi l'ordre numérique dégénère-t-il si souvent en chaos ? C'est ce que nous apprend la Seconde loi de Small...

#### Machines, mes compagnes

Afin de vous expliquer pourquoi j'ai formulé ma Seconde loi, je dois vous ennuyer un moment et vous raconter un peu ma vie. Sachez que la Seconde loi n'est pas née de grandes théories fumeuses, mais bien de mes observations directes et de mon expérience avec les ordinateurs.

Je fréquente les ordinateurs depuis... voyons... 1972, depuis ma Quatrième. Cela fait donc 22 ans que j'ai rencontré pour la première fois un terminal Télétype doté d'une "mémoire de masse" constituée d'un lecteur et d'un perforateur de bandes de papier. J'ai à présent 35 ans, ce qui fait que j'ai passé 60% de ma vie en compagnie des ordinateurs.

Depuis 1980, la fréquentation est passée du passe-temps au gagne-pain (gagne-miettes étant hélas parfois un terme plus approprié). Pendant ces vingt-et-un ans, j'ai écrit moult programmes. forçant souvent les machines à faire des choses pour lesquelles elles n'étaient pas conçues, ou encore contournant des restrictions qui me déplaisaient. J'ai dû apprendre de nombreux langages. La plupart du temps, je programme en assembleur et en BASIC, mais l'ai touché à différents autres langages : APL, C, Fortran (toujours un bon choix pour faire mouliner un Cray), voire Pascal (beuh !). Sans compter qu'il m'a fallu m'imprégner des spécificités des matériels sur lesquels je travaillais.

#### Des monceaux de données

Tous ces programmes représentent des monceaux de données brutes. Certains d'entre eux sont très longs. Spectre GCR compte quelque 25 000 lignes de code, mais j'en ai écrit en réalité beaucoup plus. puisqu'il faut tenir compte des très nombreuses lignes supprimées ou remplacées, des "échafaudages" de test (c'est-à-dire les routines que l'on écrit pour pouvoir tester un programme en cours d'écriture). En outre, certaines parties du code ont été éliminées pour cause d'obsolescence : lorsque j'ai supprimé le support des ROM 64 k, qui n'intéressent plus personne, ce sont au moins 10 000 lignes qui ont sautées. Au total, Spectre représente au bas mot 50 000 lignes. Et ce n'est qu'un seul des produits que j'ai réalisés au cours de toutes ces années!

Pour mieux voir ce que représente ce chiffre, disons que ces lignes sont du code source (lisible par un humain), ce qui n'est guère différent d'un texte (mettons une lettre quelconque) saisi à l'aide d'un éditeur. À raison de 55 lignes par page, cela fait 909 pages. Ce qui ne semble pas énorme, sauf lorsque je l'imprime. Pour produire un programme exécutable, je fais passer ce code source dans un programme appelé assembleur, qui engendre des instructions en langage-machine 68000 (pour ST) ou 68030 (pour TT ou Falcon). Centes, je ne prétends pas que mes programmes battent les records de taille, mais enfin, ça fait tout de même un nombre impressionnant de frappes au clavier (avec deux

#### doigts!)

Toutes ces lignes de code source se sont donc accumulées au fil des années. Et le code source est un fichier de texte, c'est-à-dire des données. Il n'y a pas de différence fondamentale entre ces fichiers et des fichiers provenant d'une base de données ou d'un compilateur. Toutes ces données sont en fait des alignements d'octets mis bout à bout dans des fichiers. (Il y a même une autre loi de l'informatique qui dit que les données s'accumulent toujours jusqu'à emplir la totalité de l'espace disque disponible...) Et on aboutit toujours à un tas de bits, des zéros et des uns arrangés dans un certain ordre. Le binaire n'est d'ailleurs pas la meilleure façon de manipuler des données numérique, et on aura un jour des ordinateurs qui emploient plus de deux états pour représenter l'information.

J'estime avoir écrit au moins 50 mégaoctets (Mo) de code source durant ces 21 ans. Le source de Spectre GCR fait à lui seul 4 Mo, et j'ai réalisé plusieurs projets de cette ampleur: la ROM du lecteur de disque dur d'un système nommé L. E., écrit en langage machine 780: le code du système UDC, constitué de huit 780 en parallèle (extrêmement difficiles à synchroniser!), et quelques tonnes de code en assembleur Cyber pour. . . euh. . . pouvoir accéder à des choses qui m'intéressaient, du temps où j'étais étudiant. (Je suis prudent car j'ignore quels sont les délais de prescription pour les délits informatiques.)

#### Monceaux, certes, mais organisés

Donc, Si vous utilisez une application quelconque, que ce soit un éditeur, une base de données, un tableur ou un programme de PAO, vous engendrez vous aussi, à chaque usage, une grande quantité de données. Oui sont stockées comme une suite d'octets dans des fichiers. L'un des critères de performance des ordinateurs est précisément leur aptitude à manipuler des volumes sans cesse croissants de données. Le disque dur des clones PC bon marché de milieu de gamme passe actuellement de 170 Mo à 330 Mo.

Tout dépend de l'usage qu'on en fait. Moi, j'ai dû ajouter à mon PC un second disque dur de 170 Mo (j'ai pris un Conner à interface IDE) parce que mon fils avait sauvé trop de "films de cinémitrailleuse" dans le jeu X-Wing de Lucas Art (excellent jeu, mérite largement le détour). Mais peut-être avez-vous des usages plus sérieux pour vos disques.

Quand on débute en informatique, il faut toujours se rappeler que les ordinateurs manipulent toutes les données comme des nombres. Ce sont les gens qui donnent à ces nombres des significations. Ainsi, les micro-ordinateurs utilisent tous le code ASCII (American Standard Code for Information Interchange, code américain standard pour l'échange d'informations). Ce code spécifie par exemple que la lettre A est représentée par le chiffre 65. Vous appuyez sur la touche A, l'ordinateur "voit" 65. Pour chaque type d'information, un codage, normalisé ou non, entre enjeu.

Et c'est là qu'intervient la Seconde loi de Small. Si, comme c'est le cas pour beaucoup d'entre vous, vous manipulez beaucoup de données, vous la connaissez déjà intuitivement. Si vous n'avez pas encore trop bidouillé, vous ne tarderez pas à la voir se manifester...

# Deuxième Loi De Small (dite "Loi du chaos grandissant")

"Dans un ensemble de données informatique, le désordre va toujours en augmentant. Toute tentative de réparation ne fait qu'augmenter encore le désordre."

En fait, cette loi affirme que les données informatiques ont toujours tendance à dégénérer et à se corrompre spontanément. Au bout d'un certain temps, vos bits bien ordonnés se sont mués en un chaos parfaitement aléatoire : vos données sont alors perdues. J'en vois qui sourient et qui haussent

les épaules. Allez-y, moquez-vous ! Mais croyez-moi, en 21 ans, j'ai vu ce phénomène se manifester si souvent que je ne crois pas à un simple hasard. Comme dit le vieux dicton: "La première fois, c'est une coïncidence, la deuxième, ça pourrait encore l'être, mais la troisième, c'est manifestement du sabotage."

La Seconde loi se manifeste en particulier dans de très vastes volumes de données. Même dans des disques ou des bandes certifiées, des défauts apparaissent sous forme de "trous" dans la couche d'oxyde du média, trous dans lesquels "tombent" des bits malchanceux. Et parfois, le logiciel ne détecte pas ces bits manquants. Quand vous accumulez des téraoctets de données, ce qui arrive assez vite dans les entreprises, vous marchez sur la corde raide.

Tenez, le travail que j'ai fait la nuit dernière sur Spectre 3.1 en est un parfait exemple. À un certain moment, j'ai examiné le fichier principal (qui contenait jadis la totalité de Spectre, jusqu'à ce qu'il devienne beaucoup trop long, et que j'ai dû éclater en une vingtaine de petits fichiers et un gros fichier principal). Ledit fichier faisait 80 Ko. Impossible, me suis-je immédiatement dit, ce fichier devrait faire 400 Ko. J'ai chargé le fichier dans un éditeur, et naturellement, il se terminait au milieu d'une ligne. Les quatre cinquièmes du fichier avaient disparu, et impossible de savoir où et quand. Il me faut à présent fouiller dans mes sauvegardes, et en trouver une valide de ce fichier. Mais c'est un énorme problème: comment savez-vous si vos fichiers sont devenus défectueux ? Quand vous avez dix disques répartis sur de nombreuses machines, sans compter de nombreuses cartouches Syquest de 44 Mo, c'est un problème qui est loin d'être trivial. Je ne sais plus où se trouvent la moitié des fichiers que je suis sûr d'avoir "quelque part".

Cette corruption d'un fichier vital, est-ce un accident ? Pas vraiment. En créant une structure de données informatiques, comme un fichier, vous allez à l'encontre des tendances profondes de l'univers : vous créez une structure non aléatoire. C'est jeter un défi à l'univers, qui se vengera tôt ou tard.

#### Thermodynamique

La Seconde loi de Small dérive en fait de la seconde loi de la thermodynamique, qui dit que "l'entropie de l'univers ne peut que croître. " La thermodynamique est (plus ou moins> l'étude de la chaleur et des transferts de chaleurs et d'énergie. Quant à l'entropie, c'est une mesure du chaos, du caractère aléatoire, d'un système. La seconde loi de la thermodynamique dit que l'énergie utilisable dans l'univers ne peut que décroître. Qu'est-ce que l'énergie utilisable ? C'est celle qui découle d'un certain ordre, et que l'on peut utiliser en anéantissant cet ordre, donc en augmentant l'entropie.

Prenons un exemple bien connu : il s'agit d'une expérience prouvant l'équivalence entre travail et chaleur. Vous prenez un baquet d'eau sur une table. Vous y plongez une roue à aube d'axe vertical, axe qui dépasse en haut du baquet. Sur l'axe de cette roue à aube, vous enroulez une ficelle qui est relié à un poids pouvant tomber de la table. Vous calfeutrez soigneusement le baquet et vous y mettez un thermomètre. Bon, maintenant, faites tomber le poids. Il tire la ficelle, qui fait tourner la roue à aube. En tournant, les aubes agitent l'eau, y créent des frictions et la réchauffent. Le thermomètre montre une élévation de température. Le travail fourni par la chute du poids échauffe l'eau, d'où travail égale chaleur, c'est-y pas beau la science, hmmm ?

Mais si vous faites chauffer l'eau, les pales ne se mettent pas à tourner. Tiens donc, l'équivalence est donc à sens unique ! Pourquoi ? Parce qu'en consommant du travail pour faire de la chaleur, vous augmentez l'entropie (vous ajoutez du chaos a l'univers), ce qui est permis. Mais prendre de la chaleur et en faire du travail, cela ferait diminuer l'entropie, et ça, c'est Niet! Sinon, les bateaux pourraient naviguer en extrayant leur énergie de la chaleur de l'océan et en laissant derrière eux un sillage de glaçons...

Ce qui est par contre autorisé, c'est d'utiliser une différence de température pour produire un travail. Les pales immergées dans l'eau sont toutes à la même température, et donc, pas de différence à utiliser. Mais si maintenant, vous les mettez dans un jet de vapeur, avec de l'eau chaude d'un côté et un refroidisseur de l'autre, ca marche, les pales tournent. En augmentant l'entropie. Car là aussi, vous

détruisez de l'ordre. Vous aviez de l'eau chaude d'un côté (la source de vapeur), de l'eau froide de l'autre (le refroidisseur), et vous faites tendre les deux vers une tiédeur uniforme et insipide. Et cette eau tiède ne se scindera pas spontanément en une masse d'eau chaude et une d'eau froide : quand le désordre s'installe, on ne le chasse pas Si facilement. Si vous prenez un litre d'eau a 100 degrés et un à 0 degrés, vous avez la même énergie que dans deux litres d'eau à 50 degrés. Mais dans le premier cas, cette énergie est utilisable en partie, grâce à la différence de température. Dans l'autre, elle ne l'est pas.

Ça va, vous suivez ? Bravo, vous venez juste d'avaler votre première leçon de thermodynamique.

Le destin ultime de l'univers sera peut-être, dans 15 milliards d'années, une température uniforme partout, sans source de chaleur, sans la moindre différence de température à utiliser, une entropie maximale. Heureusement, des gens comme Stephen Hawking affirme que l'univers est suffisamment dense pour que ce jour n'arrive jamais : d'ici là, il se sera de nouveau contracté jusqu'à ce que survienne un nouveau Big Bang. Et il y a aussi la théorie de la création continue de matière, qui dérive de la physique quantique, et qui rend suffisamment bien compte de l'état actuel de l'univers pour être prise en considération, et qui dit que l'univers est en expansion et en création continue. (Votre choix dépend en grande partie de votre religion.)

#### **Barrage**

Le concept de différence de température pouvant produire de l'énergie est parfois délicat a appréhender. Aussi, permettez-moi de vous proposer une analogie, celle du barrage hydro-électrique. Un barrage, c'est un mur de béton retenant de l'eau. Au pied du barrage se situe le lit original du cours d'eau barre. Les ingénieurs installent un énorme tuyau entre le lac artificiel et le lit de la rivière, et ils y intercalent une turbine. L'eau s'engouffre dans cette "conduite forcée" à une pression énorme, et fournit une énergie considérable à une turbine, qui fait tourner un alternateur produisant de l'électricité. On parle souvent de "production d'énergie", mais jamais de création. La première loi de la thermodynamique dit qu'on ne peut créer de l'énergie, et qu'on ne peut qu'en transférer (ici, entre l'eau et le générateur électrique). La pression est d'autant plus forte que le niveau d'eau est plus haut dans la retenue. On exploite en fait la différence de hauteur entre le lac artificiel et le lit du cours d'eau pour extraire de l'énergie.

Mais imaginez qu'une sécheresse fasse baisser le niveau du lac à un point tel qu'il n'y ait plus de différence de niveau entre le lac et le lit original. Plus de différence de hauteur, plus moyen d'extraire l'énergie de l'eau.

À ce propos, Si vous avez l'occasion de visiter un barrage hydroélectrique, ne vous en privez pas. J'ai eu l'occasion de visiter le Hoover Dam, pas très loin de chez moi. Le clou de la visite est le moment où l'on monte sur l'énorme conduite amenant l'eau aux turbines, où rugissent les torrents d'eau qui s'y engouffrent depuis le sommet du barrage. On éprouve alors la mémé sensation que dans une fusée qui décolle, avec toute cette énergie brute qui circule sous vos pieds... C'est le genre d'effets spéciaux que les cinéastes voudraient bien arriver à créer lorsque l'entreprise fonce à vitesse maximum. mais ils sont loin du compte comparés à cette force naturelle. Le Hoover Dam est, a mon avis, une des merveilles du monde, car il montre ce que les hommes sont capables de créer quand ils ont le courage de le faire et qu'on ne les enquiquine pas à leur faire remplir des formulaires de l'Agence 'pour la protection de l'environnement. (De nos jours, les lois sur l'impact sur l'environnement interdiraient ce genre de projet.)

#### Réserve d'énergie

Autre exemple: une pierre chauffée au feu représente une réserve d'énergie parce qu'elle est plus

chaude que son entourage, et peut donc rayonner de l'énergie sous forme de chaleur. Cette différence est ce que les scientifiques nomment parfois "un delta", symbolisé par un triangle. Si vous laissez la pierre se refroidir (en réchauffant la pièce), la pièce et la pierre se trouvent à la même température, et l'entropie est maximale. La situation n'évoluera plus. Le delta est devenu nul.

Tout ceci s'applique parfaitement à l'informatique. Prenez un disque dur de ST absolument parfait. La table des partitions et les deux tables d'allocations de fichiers de chaque partition sont parfaitement en ordre, et, pour peu que le disque ait été défragmenté par un utilitaire adéquat, les fichiers sont tous sur des secteurs consécutifs, sans intervalle vide disgracieux. Enfin, tous les fichiers sont lisibles, et contiennent bien ce qu'ils sont censée contenir. (Je ne suis même pas sûr d'avoir un jour entrevu pareil spectacle édénique. Mais c'est assurément la façon dont les choses devraient être.)

Toutes ces données sur votre disque dur sont, pour reprendre mon analogie, similaire à un barrage rempli à ras bord. Le potentiel (le delta> est à son maximum. et l'énergie accumulée est énorme.

Si la seconde loi de la thermodynamique s appliquait à ce disque, vous auriez une situation très instable. Vous auriez un haut niveau d'énergie (le disque bien ordonné) dans un environnement à bas niveau (l'état naturel chaotique de l'oxyde sur le disque). Et de même que l'eau n'attend que la moindre fissure pour s'écouler, en détruisant le barrage si nécessaire, vos données n'attendraient que la moindre occasion pour retourner au chaos primitif, en bousillant votre disque au passage.

Car l'état naturel de l'univers est le chaos. Et l'état naturel des données est le hasard total. Un disque à peine fabriqué n'est pas naturellement formaté. Les particules magnétiques y sont orientées dans tous les sens, sans qu'on puisse y distinguer le moindre ordonnancement. Or, le disque dur soigneusement rempli est à un état minimal d'entropie. ce que l'univers déteste, comme l'explique la seconde loi de la thermodynamique.

Et je vous garantie que dès que vous accédez à votre beau disque parfait bien propre, celui-ci s efforcera de retourner à l'état sauvage au plus vite, nivelant le delta, par un plantage si nécessaire. C'est ce comportement, maintes fois observé empiriquement, qui est formalisé dans ma Seconde loi.

Ne croyez pas que l'analogie entre un haut niveau d'organisation des données et un haut niveau d'énergie soit tirée par les cheveux. En fait, c'est à peu près ce que dit la troisième loi de la thermodynamique. Appliquée aux structures cristallines (c'est-à-dire des molécules organisées en grilles, comme le diamant), elle dit que l'univers ne les apprécie guère, et préférerait voir ces molécules retourner à l'état de bouillie informe. Si vous prenez un cristal et que vous le chauffez, vous allez obtenir cette bouillie en disloquant les grilles, et vous allez augmenter l'entropie de la substance en question. Or, qu'est-ce qu'une structure de données, sinon un agencement de particules magnétiques sur un support, particules qui sont naturellement en désordre ? C'est l'équivalent logiciel d'un cristal. L'entropie ne peut que croître, le cristal ne peut que se disloquer, les données ne peuvent que se corrompre. Ce qui est une façon scientifique de dire: "La vie est une chiennerie, et en plus, quand vous crevez, la nature applaudit."

#### J'ai voulu violer la loi

Je vous donne des conseils, je vous dis que ma Seconde loi est intuitivement évidente, que je l'ai toujours su... En fait, en un instant d'égarement et de naïveté, j'ai tenté de la violer, avec le brillant succès qu'on imagine. Que je vous raconte.

Il y a quelque temps, j'ai décidé que les différentes versions de la vingtaine de fichiers de Spectre 3. 1 commençaient à devenir ingérables. L'horodatage des fichiers par l'horloge interne du ST ne marche pas très bien pour moi (pour une raison que j'ai mis un bon bout de temps à découvrir). Et souvent, il me fallait aller compulser les différentes versions d'un fichier pour savoir laquelle était la bonne, la dernière! J'ai alors décidé, bêtement, sans réfléchir, de créer le disque dur parfait bien propre. J'ai donc pris un disque neuf mais déjà rodé, qui avait assez tourné pour avoir dépassé le stade de la mortalité infantile. Et j'ai commencé à mettre chaque fichier à l'endroit approprié, accompagné de commentaires et de documentations. J'ai créé des dossiers, un par version de Spectre: "1.51", "1 .75","1.9F","2 .3K", "2.65C", "3.0", "3.1Dev", plus toutes les versions intermédiaires que seuls ont vues les bèta-

testeurs. (Chaque saut de numéro de version constitue autant de sueur et de larmes passées en test et en débogage).

Pendant des jours et des jours, j'ai fouillé dans mes disquettes et mes cartouches Syquest. J'ai exhumé de vieilles versions, les ai vérifiées, copiées dans les bons dossiers. Un boulot fastidieux et rébarbatif où j'ai dépensé beaucoup d'énergie et de Pepsi.

Dans chaque dossier, j'ai tout vérifié en assemblant les fichiers et en recréant la version correspondante de Spectre, que j'ai ensuite comparée aux disques de productions, dont Sandy a été assez maligne pour garder un exemplaire pour chaque version, me faisant menacer des pires châtiments si je ne les lui rendais pas promptement. Et chaque dossier a ensuite reçu un fichier de documentation.

Étais-je sot : j'ai même poussé le vice jusqu'à inclure sur ce disque les 19 versions bêta de Spectre 3. 0. Puisque j'étais en train de faire LE disque parfait, autant les y mettre, n'est-ce pas ? Après tout, certaines applications Apple avaient montré une fâcheuse propension à tourner sur une version bêta mais pas sur la suivante (comme Pagemaker qui s'était mis à planter sur l'avant-dernière version bêta de Spectre 3. 0 -un mauvais souvenir, il nous a fallu supprimer ce bogue en un temps record).

J'ai aussi récupéré ici et là des fichiers divers, comme des docs sur le clavier du Macintosh et les codes qu'il émettait. Comme les autres, il aboutirent dans des dossiers soigneusement documentés. Inutile de dire que tout cela a pris un grand nombre de mégaoctets.

Pour être sûr d'éviter les corruptions spontanées de fichiers, j'ai passé les fichiers à l'utitaire ARC. ARC compresse les fichiers, mais surtout, il calcule un CRC (Cyclic Reduncant Check, somme de contrôle redondant cyclique) pour chaque fichier. Il s'agit d'une sorte de signature du fichier, obtenue en calculant un polynôme avec chaque octet du fichier. Modifier un simple bit ou intervertir deux octet modifie le CRC, et il est presque impossible de modifier accidentellement le fichier en conservant le même CRC. J'ai donc fait une liste de tous les fichiers avec leur CRC. En cas de doute sur l'intégrité d'un fichier, je pouvais recalculer son CRC et voir s'il correspondait à celui de la liste.

Enfin, j'ai relancé un programme qui recalculait le CRC pour chacun des 1500 fichiers et le revérifiait par rapport à la liste. J'ai imprimé les fichiers de documentation pour en conserver une version sur papier.

Poussant un soupir satisfait, je me suis étiré. empli de la béatitude du devoir accompli. Je savais où se trouvait chaque fichier à cette seconde précise, et était certain de son intégrité. Il ne restait plus qu'a faire une sauvegarde de ce petit bijou d'ordre et de rigueur, qu'il m'avait fallu un bon mois pour fignoler...

En organisant ainsi tant de données, j'avais naturellement rempli mon barrage à ras bord, créant une situation d'entropie minimale et d'organisation maximale. J'avais défié les lois du chaos et violé la seconde loi de la thermodynamique appliquée à l'informatique. J'avais construit un château de cartes de vingt mètres de haut.

L'univers n'attendait que l'occasion de m'apprendre à vivre. Notez bien qu'avec l'astuce des CRC, le coup classique de la corruption sournoise des fichiers devenait impossible, car j'aurais pu le détecter. Il ne restait plus qu'une possibilité.

Comme vous le savez si vous avez lu mon article précédent, les lois de l'univers qui le vouent au chaos se mirent en œuvre par l'intermédiaire de la mécanique quantique, et engendrèrent un continuum spatio-temporel dans lequel était inscrit mon tragique destin. Le destin en question consistant bien sûr à recevoir le fameux château de cartes sur la tête afin de niveler cet arrogant delta de haute organisation.

Innocemment, j'ai éteint le système pour y connecter un lecteur de bande magnétique. J'ai mis le lecteur et j'ai rallumé.

Le disque dur n'a pas réagit l'allumage.

Mes cheveux se sont dressés sur ma tête et j'ai été pris de sueurs froides.

Pendant une semaine, j'ai tout essayé pour ressusciter ce disque. J'ai remplacé son circuit imprimé interne, son alimentation, je l'ai fait tourner à la main, je l'ai secoué pour décoller les têtes, enfin tout.

En vain. Il était mort. Et je n'avais pas de sauvegarde. C'était la seule possibilité, elle s'était réalisée. Le delta avait été nivelé d'un coup. Paf.

Meuh non, dites-vous, j'ai malencontreusement envoyé une décharge d'électricité statique a ce pauvre disque, ou alors c'est le câble SCSI qui n'était pas bon, ou encore l'alimentation qui a claqué a l'allumage et a bousillé le disque... Ben voyons. Non, désolé, ça ne prend plus, les coïncidences, j'en ai trop vues. C'étaient les lois inexorables de l'univers qui venaient de frapper.

Depuis lors, j'ai pris l'habitude de ne jamais faire d'effort pour organiser mon disque dur. Oh, certes, je sais plus ou moins où sont mes fichiers, mais je me garde bien de trop augmenter mon niveau d'organisation, et d'attirer l'attention des implacables gardiens des lois de l'entropie. En outre, je laisse toujours une partition a l'état de chaos complet sur mon disque. Ce sacrifice aux dieux du désordre m'a jusqu'à présent évité leurs foudres.

Je sais, je sais, ça a l'air idiot. Mais en tout cas, ça marche. Mon taux de pannes de disques est a son niveau historique le plus bas. Un de mes lecteurs, un Syquest souffrant d'un problème de moteur de rotation, a même eu l'extrême obligeance d'avoir une embellie finale et de se remettre a fonctionner, ce qui m'a permis d'y récupérer des mégaoctets de données prises sur des serveurs télématiques, au prix d'innombrables heures de téléchargement. Le Syquest a ensuite définitivement rendu l'âme, cinq minutes après que j'y ai récupéré le dernier fichier. De quoi se poser des questions, non ?

Tout ceci a des implications qui m'effraient. Je constate que les concentrations de données souffrent. C'est peut-être dû en partie aux limites de la technologie que l'on repousse sans cesse. Même avec un taux de fiabilité de 99,9%, il y a un gars sur mille (moi!) à qui échoient les rebuts. Certes, Atari ne teste pas ses machines à 100%, ce qui est reflété par leur prix modique. Mais j'ai aussi trouvé des fichiers corrompus sans raison apparente sur des 486 ou des Macintosh fort coûteux. Tenez, prés de chaque Mac, nous gardons un kit d'installation, parce que nous devons régulièrement recharger le Système et le Finder...

#### Stockage visible

Mon seul espoir est notre lecteur de disque WORM (Write Once, Read Many) connecté à une vieille machine AT&T. C'est un disque sur lequel tout ce qu'on écrit est définitivement enregistré, littéralement gravé au laser. Mais tôt ou tard, ce lecteur tombera en panne, ou bien ce sera l'AT&T. Les circuits imprimés ont une limite d'âge de dix ans. Passé ce temps, des fissures commencent à y apparaître en raison des trop nombreux cycles thermiques (échauffement t l'allumage, refroidissement à l'extinction) qui dilatent et contractent les pistes de cuivre dans les cartes. Je suis persuadé que si cela devient un jour nécessaire, un élément irremplaçable tombera en panne, me privant du disque WORM.

J'en viens à considérer des médias où l'information est stockée sous forme visible, macroscopique, comme les bandes de papier perforées. Les bits sur les surfaces magnétiques sont sujets aux aléas de la physique quantique, dont les trous des bandes de papiers n'ont cure. Les lecteurs de bandes à haute vitesse sont assez rapides, et se trouvent dans les surplus pour un prix dérisoire.

En tout cas, évitez comme la peste de bâtir comme moi des châteaux de cartes informa tiques, des disques durs hyper-organisés vous attireriez immanquablement quelque catastrophe. Ne hissez pas votre potentiel d'ordre trop haut, vous vous feriez remarquer. Ne soyez pas le cristal que l'univers brûle de faire fondre. Et n'oubliez pas que les Seigneurs du Chaos aiment qu'on leur consacre une zone délibérément laissée à l'état sauvage sur un disque dur.

Il faut également que vous sachiez qu'à partir de 20 Mo de données, vous avez au moins un fichier vérolé. Pouvez-vous trouver lequel ? Non, hein ? J'en étais sûr.

Rassurez-vous, vous n'êtes pas les seuls. Les publicités d'utilitaires pour Macintosh affirment que 500A des Macs souffrent de répertoires vérolés. Les Macs utilisent non pas une simple table de répertoire, mais une structure arborescente complexe, un Bqree, qui peut facilement être endommagé. Si vous avez un Spectre, procurez-vous les Norton Utilities pour Mac, vous en aurez besoin.

Quant aux PC... Une fois, je rassemblais d'anciens articles pour une disquette destinée à ST-Mag,

quand soudain, au milieu d'un article que je relisais, je vis apparaître des données binaires venues de Dieu sait où. L'original avait disparu. Heureusement, un ami avait gardé une copie de cet article, et j'ai pu finalement envoyer cette disquette. [NdT: Rassure-toi, Dave, les Postes nous l'ont paumée, tu n'as plus qu'a la refaire. Eh Si.]

Bref n'allez pas à l'encontre des lois de l'univers, ou il vous en cuira Écoutez vos anciens, jeunes présomptueux. Ils ont souffert pour vous. Faites comme moi: suivez le courant, prenez vos dispositions, sinon vous vous en mordrez les doigts

**Traduction et adaptation Password 90**