

## Le débat Tanenbaum-Torvalds

Vous trouverez dans cette annexe le texte du débat entre Tanenbaum et Linus connu dans la communauté sous le titre « Linux est dépassé ». Andrew Tanenbaum est un chercheur très respecté, qui a fort bien gagné sa vie en étudiant les systèmes d'exploitation et leur conception. Au début de 1992, ayant remarqué que les discussions sur Linux prenaient le pas sur les autres dans comp.os.minix, il décida qu'il était temps de publier ses opinions.

Bien qu'Andrew Tanenbaum fut tourné en ridicule pour sa maladresse et ses mauvais jugements sur le noyau Linux, la réaction qu'il essuya fut injuste. Lorsque Linus lui-même entendit dire que nous allions publier ce débat, il voulut s'assurer que tout le monde comprenne qu'il ne gardait aucune animosité envers Tanenbaum, et n'aurait en fait pas approuvé cette publication si nous n'avions pu le convaincre que cela montrerait comment les systèmes d'exploitation étaient perçus à l'époque.

La publication de cette annexe pourrait donner une bonne idée de l'état des choses dans les milieux universitaires au moment où Linus était sous pression pour avoir abandonné l'idée de micro-noyau. Le premier tiers de la discussion montre cela plus en détail.

Des copies électroniques de cette discussion sont faciles à trouver sur le Web. Il est amusant de lire ceci et de noter que parmi les personnes qui se joignirent à la discussion vous trouverez Ken Thompson (l'un des concepteurs d'Unix) et David Miller (devenu un hacker majeur du noyau Linux) et encore bien d'autres.

Pour replacer cette discussion dans son contexte, lorsqu'elle commença en 1992, le 386 était le microprocesseur dominant et le 486 n'était pas sorti sur le marché. Microsoft était encore une petite entreprise commercialisant MS-DOS et Word pour MS-DOS. Lotus 123 dominait le marché des tableurs, et WordPerfect celui des éditeurs de texte. DBASE était l'éditeur dominant de logiciels de gestion de base de données, et beaucoup d'entreprises dont les noms sont maintenant connus de tous (Netscape, Yahoo, Excite) n'existaient tout simplement pas.

**De: ast@cs.vu.nl (Andy Tanenbaum)**  
**Groupe de discussion: comp.os.minix**  
**Sujet: Linux est dépassé**  
**Date: 29 Jan 92 12:12:50 GMT**

J'ai passé 15 jours aux États-Unis, je n'ai donc pas pu faire beaucoup de remarques sur Linux (non pas que j'en aurais dit plus, aurais-je été dans le coin), mais souhaite à présent publier un ou deux commentaires.

Comme la plupart d'entre vous le savent, MINIX est pour moi un passe-temps, quelque chose que je fais le soir quand j'en ai assez d'écrire des livres et qu'il n'y a pas de guerre exceptionnelle, de révolution, ou d'audience au sénat retransmise sur CNN. Mon vrai travail est professeur et chercheur dans le domaine des systèmes d'exploitation.

Étant donnée ma profession, je pense que je sais à peu près vers où l'exploitation va se diriger dans la décennie à venir. Deux aspects ressortent :

### 1. SYSTÈME MONOLITHIQUE CONTRE MICRO-NOYAU

La plupart des anciens systèmes d'exploitation sont monolithiques, c'est-à-dire que le système d'exploitation entier est un unique fichier a.out qui s'exécute en 'mode noyau'. Ce binaire contient la gestion des processus, la gestion de la mémoire, le système de

fichiers et le reste. Des exemples de tels systèmes sont UNIX, MSDOS, VMS, MVS, OS/360, MULTICS, et beaucoup d'autres.

L'autre option est un système à micro-noyau, dans lequel le plus gros du SE s'exécute en tant que processus séparés, pour la plupart en dehors du noyau. Ils communiquent par passage de messages. Le travail du noyau est de contrôler le passage des messages, le contrôle des interruptions, la gestion des processus bas-niveau, et si possible les E/S. Des exemples de cette conception sont les RC4000, Amoeba, Chorus, Mach, et le pas-encore-réalisé MS-Windows/NT.

Bien que je pourrais me lancer ici dans une longue histoire sur les mérites relatifs des deux concepts, je me contenterai de dire que parmi les personnes concevant à l'heure actuelle des systèmes d'exploitation, la discussion est essentiellement close. Les micro-noyaux ont gagné. Le seul vrai argument pour les systèmes monolithiques était la performance, et il est maintenant assez évident que les systèmes à micro-noyau peuvent être tout aussi rapides que les systèmes monolithiques (par ex., Rick Rashid a publié des notes comparant Mach 3.0 à des systèmes monolithiques), et il ne reste donc maintenant que des détails à régler.

MINIX est un système à micro-noyau. Le système de fichiers et la gestion de la mémoire sont des processus séparés, s'exécutant en dehors du noyau. Les gestionnaires d'E/S sont aussi des processus séparés (dans le noyau, mais seule la nature comateuse des processeurs Intel fait qu'il est difficile de faire autrement).

LINUX est un système de style monolithique. C'est un pas de géant en arrière dans les années 1970. C'est comme prendre un programme C existant et qui marche et le réécrire en BASIC. Pour moi, écrire un système monolithique en 1991 est une idée vraiment médiocre.

## 2. PORTABILITÉ

Il était une fois le microprocesseur 4004. En grandissant, il devint un 8008. Puis il subit une opération de chirurgie plastique et devint le 8080. Il engendra le 8086, qui engendra le 8088, qui engendra le 80286, qui engendra le 80386, qui engendra le 80486, et ainsi de suite jusqu'à la Nième génération. Pendant ce temps, les microprocesseurs RISC apparurent, et certains d'entre eux développant plus de 100 MIPS. Des vitesses supérieures à 200 MIPS seront vraisemblablement atteintes durant les prochaines années. Ces choses ne vont pas disparaître subitement. Ils prennent peu à peu le pas sur la lignée des 80x86 et exécuteront de vieux programmes MS-DOS en simulant le 80386 au niveau logiciel (j'ai même écrit mon propre simulateur IBM PC en C, qui peut être obtenu par FTP depuis ftp.cs.vu.nl = 192.31.231.42 dans le répertoire minix/simulator). Je pense que c'est une erreur grossière que de concevoir un SE pour une architecture spécifique, alors que celle-ci ne sera bientôt plus disponible.

MINIX a été conçu pour être raisonnablement portable, et a été porté de la ligne Intel aux 680x0 (Atari, Amiga, Macintosh), SPARC, et NS32016. Linux est attaché de très près au 80x86. C'est une mauvaise approche.

Ne le prends pas mal, je ne suis pas mécontent de LINUX. Il va amener tous ceux qui veulent faire évoluer MINIX en BSD UNIX derrière moi. Mais en toute honnêteté, je voudrais suggérer à ceux qui veulent un SE "libre" \*\*MODERNE\*\* qu'ils recherchent un SE à micro-noyau, portable, comme peut-être GNU ou quelque chose comme ça.

Andy Tanenbaum (ast@cs.vu.nl)

P.S. Juste en passant, Amoeba a un émulateur UNIX (s'exécutant au niveau utilisateur), mais il est loin d'être complet. S'il y a quelqu'un qui veut travailler dessus, merci de me le faire savoir. Pour exécuter Amoeba, vous devez avoir quelques 386, l'un d'eux nécessitant 16M, et tous équipés d'une carte Ethernet WD.

**De:** torvalds@klaava.Helsinki.FI (Linus Benedict Torvalds)  
**Sujet:** Re : LINUX est dépassé  
**Date:** 29 Jan 92 23:14:26 GMT  
**Organisation:** University of Helsinki

Bien, avec un sujet tel que celui-ci, j'ai bien peur de devoir répondre. Que les utilisateurs de minix qui en ont assez d'entendre parler de linux m'excusent. J'aimerais pouvoir dire "ne tenez pas compte de cette attaque", mais... il est temps d'entamer une sérieuse querelle !

Dans l'article <12595@star.cs.vu.nl> ast@cs.vu.nl (Andy Tanenbaum) écrit :

>  
> J'ai passé 15 jours aux États-Unis, je n'ai donc pas pu faire  
> beaucoup de remarques sur Linux (non pas que j'en aurais dit plus,  
> aurais-je été dans le coin), mais souhaite à présent publier un ou deux  
> commentaires.  
>  
> Comme la plupart d'entre vous le savent, MINIX est pour moi un  
> passe-temps, quelque chose que je fais le soir quand j'en ai assez  
> d'écrire des livres et qu'il n'y a pas de guerre exceptionnelle, de  
> révolution, ou d'audience au sénat retransmise sur CNN. Mon vrai  
> travail est professeur et chercheur dans le domaine des systèmes  
> d'exploitation.

Vous utilisez cela [être un professeur] comme excuse pour les limitations de minix ?  
Désolé mais vous avez perdu : j'ai plus d'excuses que vous, et linux bat toujours minix à plate couture dans la plupart des domaines. Et je n'insisterai pas sur le fait que le plus gros de l'excellent code du minix pour PC semble avoir été écrit par Bruce Evans.

Re 1 : vous considérez minix comme un passe-temps - regardez qui gagne de l'argent avec minix, et qui offre linux. Maintenant parlons passe-temps. Faites en sorte que minix soit librement accessible, et l'un de mes plus gros griefs disparaîtra. Linux a été très certainement un passe-temps (mais du type meilleur type : sérieux) pour moi : je n'ai pas reçu d'argent, et il ne fait même pas partie de mes études universitaires. J'ai tout réalisé sur mon propre temps, et sur ma propre machine.

Re 2 : vous êtes professeur et chercheur : c'est une sacrée bonne excuse expliquant les dommages cérébraux provoqués par minix. J'espère seulement (et je le suppose) qu'Amoeba n'est pas aussi nul que minix.

>1. SYSTÈME MONOLITHIQUE CONTRE MICRO-NOYAU

C'est vrai, linux est monolithique, et je suis d'accord que les micro-noyaux sont mieux. Sur un sujet appelant moins à polémique,

j'aurais été probablement d'accord avec l'essentiel de ce que vous avez dit. D'un point de vue théorique (et esthétique), linux a perdu.

Si le noyau GNU avait été fini au dernier printemps, je ne me serais même pas embêté à démarrer mon projet : le fait est qu'il ne l'était pas et ne l'est pas encore. Linux a l'énorme avantage d'être disponible maintenant.

> MINIX est un système à micro-noyau. [ écourté, mais de  
> façon que vous ne puissiez perdre le fil ]. LINUX est un système  
> monolithique.

Si c'était le seul critère de « bonne qualité » d'un noyau, ce serait exact. Ce que vous ne mentionnez pas, c'est que minix ne fonctionne pas comme un micro-noyau de manière correcte, et pose des problèmes avec le multitâche en temps réel (dans le noyau). Si j'avais fait un SE ayant des problèmes avec un système de fichiers subdivisé en unités d'exécution, je n'aurais pas été aussi prompt à condamner les autres : en fait, j'aurais fait tout mon possible pour que les autres oublient ce fiasco.

[ Oui, je sais qu'il y a des bitouillages sur le multithreading pour minix, mais ce sont des bitouillages, et Bruce Evans m'a dit qu'il existe un tas de contraintes ]

## >2. PORTABILITÉ

« La portabilité, c'est pour les gens qui ne savent pas écrire de nouveaux programmes » - moi, à l'instant (ton ironique).

Le fait est que linux est plus facilement portable que minix. Quoi ? vous entends-je dire. C'est vrai - mais pas dans le sens où l'entend ast : j'ai conçu linux aussi conforme qu'il m'était possible aux normes (sans avoir de copie des normes POSIX sous la main). Porter des choses sous linux est généralement /beaucoup/ plus facile que de les porter sous minix.

Je suis d'accord avec le fait que la portabilité est une bonne chose : mais uniquement lorsque cela a réellement une signification. Cela n'a pas de sens de créer un système d'exploitation totalement portable : coller à une API portable suffit. La véritable /idée/ d'un système d'exploitation est d'utiliser les fonctionnalités du matériel, et de les placer derrière une couche d'appels de haut niveau.

C'est exactement ce que fait linux : il utilise en fait plus complètement les fonctionnalités du 386 que les autres noyaux. Bien sûr cela rend le noyau non portable au sens strict, mais cela conduit à une conception /vraiment/ plus simple. Un compromis acceptable, mais qui a eu pour premier avantage de permettre l'existence de linux.

Je suis d'accord également avec le fait que linux porte la non-portabilité à un extrême : j'ai acquis mon 386 en janvier dernier, et linux faisait partie d'un projet visant à m'apprendre à m'en servir. Les choses auraient été faites de manière plus portable si j'avais eu un projet concret. Je ne cherche pas des tas d'excuses, bien que c'était une décision de conception, et quand j'ai démarré les choses en avril dernier, je ne pensais pas que quelqu'un l'utiliserait réellement. Je suis heureux de dire que j'ai eu tort car mes sources sont librement disponibles et chacun peut essayer de les porter, même si cela ne sera pas facile.

Linus

PS. Excusez-moi pour ce qui peut sembler parfois sévère : minix est suffisant si vous n'avez rien d'autre. Amoeba pourrait être impeccable si vous possédez 5 à 10 386 en secours, mais je ne m'y risquerai pas. Je ne me lance pas habituellement dans les querelles mais suis susceptible quand il s'agit de linux :)

**De:** ast@cs.vu.nl (Andy Tanenbaum)  
**Sujet:** Re : LINUX est dépassé  
**Date:** 30 Jan 92 13:44:34 GMT

Dans l'article

<1992Jan29.231426.20469@klaava.Helsinki.FI> torvalds@klaava.Helsinki.FI (Linus Benedict Torvalds) écrit :

>Vous utilisez cela [être un professeur] comme excuse pour les  
>limitations de minix ?

Les limitations de MINIX sont liées en partie au fait que je suis professeur : j'avais comme objectif précis de le faire fonctionner sur du matériel bon marché afin que les étudiants puissent se l'offrir. En particulier, durant des années il tourna sur un PC ordinaire à 4.77 MHz sans disque dur. Vous pouviez tout faire avec, modifier et recompiler le système inclus. Juste pour mémoire, jusqu'à il y a un an environ, il y avait deux versions, une pour le PC (disquettes 360K) et une pour le 286/386 (1.2M). La version PC se vendait deux fois mieux que la version 286/386. Je n'ai pas de chiffres, mais mon opinion est que la fraction des 60 millions de PC existants qui sont des machines 386/486 opposées aux 8088/286/680x0 etc est faible. Parmi les étudiants elle est encore plus faible. Développer du logiciel gratuit uniquement destiné aux gens ayant assez d'argent pour acheter du matériel de première classe est un concept intéressant.

Bien sûr, dans 5 ans, ça sera différent, mais dans 5 ans tout le monde emploiera le GNU gratuit sur sa station SPARC-5 à 200 MIPS, 64 Mo.

>Re 2 : vous êtes professeur et chercheur : c'est une sacrée bonne  
>excuse expliquant les dommages cérébraux provoqués par minix.  
>J'espère seulement (et je le suppose) qu'Amoeba n'est pas aussi  
>nul que minix.

Amoeba n'a pas été conçu pour fonctionner sur un 8088 sans disque dur.

>Si c'était le seul critère de « bonne qualité » d'un  
>noyau, ce serait exact. Ce que vous ne mentionnez pas, c'est  
>que minix ne fonctionne pas comme un micro-noyau de manière correcte,  
>et pose des problèmes avec le multitâche en temps réel (dans le  
>noyau). Si j'avais fait un SE ayant des problèmes avec un système  
>de fichiers subdivisé en unités d'exécution, je n'aurais pas été  
>aussi prompt à condamner les autres : en fait, j'aurais fait tout  
>mon possible pour que les autres oublient ce fiasco.

Un système de fichiers multi-thread est seulement un bitouillage d'amélioration des

performances. Lorsqu'il y a seulement un processus actif, cas normal sur un petit PC, cela ne vous apporte rien et ajoute de la complexité au code. Sur des machines suffisamment rapides pour servir plusieurs utilisateurs, vous avez probablement assez de mémoire-cache pour assurer un bon cache à un bon taux, auquel cas le multi-thread ne vous apporte rien non plus. Cela ne rapporte vraiment que lorsque plusieurs processus font au même moment des E/S effectives

sur disque. Que ça vaille la peine de rendre le système plus complexe dans ce cas est pour le moins discutable.

Je persiste à penser que concevoir un noyau monolithique en 1991 est une erreur fondamentale. Estime-toi heureux de ne pas être un de mes étudiants. Tu n'obtiendrais pas une bonne note pour une telle conception :-)

>Le fait est que linux est plus facilement portable que minix. Quoi ?  
>vous entends-je dire. C'est vrai - mais pas dans le sens où l'entend  
>ast : j'ai conçu linux aussi conforme qu'il m'était possible aux  
>normes (sans avoir de copie des normes POSIX sous la main).  
>Porter des choses sous linux est généralement /beaucoup/ plus facile  
>que de les porter sous minix.

MINIX fut conçu avant POSIX, et est maintenant en train d'être (lentement) POSIXisé comme quiconque suivant ce groupe le sait. Tout le monde convient que des normes au niveau utilisateur sont une bonne idée. En aparté, je te félicite d'être capable d'écrire un système en conformité avec POSIX sans avoir les normes POSIX en face de toi. Je trouve ceci assez difficile après avoir étudié les normes dans le détail.

Je pense qu'écrire un nouveau système d'exploitation étroitement lié à un matériel particulier, surtout aussi étrange que la ligne Intel, est fondamentalement mauvais. Un SE lui-même doit être facilement portable vers de nouvelles plates-formes matérielles. Le fait qu'OS/360 fut écrit en langage d'assemblage IBM 360 il y a 25 ans peut probablement être excusé. Lorsque MS-DOS fut écrit explicitement pour le 8088 il y a 10 ans, c'était moins que brillant, comme le réalisent trop péniblement IBM et Microsoft maintenant seulement. Écrire un nouveau SE seulement pour le 386 en 1991 te donne une autre mauvaise note pour ce trimestre. Mais si tu fais vraiment mieux à l'examen final, tu peux encore empocher le module.

Prof. Andrew S. Tanenbaum (ast@cs.vu.nl)

**De: feustel@netcom.COM (David Feustel)**  
**Sujet: Re : LINUX est dépassé**  
**Date: 30 Jan 92 18:57:28 GMT**  
**Organisation: DAFCO - An OS/2 Oasis**

ast@cs.vu.nl (Andy Tanenbaum) écrit :

>Je persiste à penser que concevoir un noyau monolithique en  
>1991 est une erreur fondamentale. Estime-toi heureux de ne pas être  
>un de mes étudiants. Tu n'obtiendrais pas une bonne note pour une  
>telle conception :-)

C'est vrai. Einstein a eu des notes déplorables en maths et en physique.

**De:** pete@ohm.york.ac.uk (-Pete French.)  
**Sujet:** Re : LINUX est dépassé  
**Date:** 31 Jan 92 09:49:37 GMT  
**Organisation:** Electronics Department, University of York, UK

dans l'article <1992Jan30.195850.7023@epas.toronto.edu>,  
meggin@epas.utoronto.ca (David Megginson) dit :  
>  
> Dans l'article <1992Jan30.185728.26477feustel@netcom.COM>  
feustel@netcom.COM (David > Feustel) écrit :  
>>  
>>C'est vrai. Einstein a eu des notes déplorables en maths et en  
>>physique.  
>  
> Et Dan Quayle une mauvaise note en sciences politiques. Je pense  
> qu'il y a plus de Dan Quayle que d'Einstein ici bas.. ;-)

Quelle abominable réflexion !  
Mais pour revenir au débat micro-noyau <-> monolithique,  
n'utilise-t-on pas en partie un artifice de langage ? MINIX peut  
très bien être conçu comme un système micro-noyau, mais au bout du  
compte vous finirez toujours avec un gros morceau monolithique  
de données binaires constituant « le système d'exploitation ». N'est-il pas écrit en plusieurs programmes distincts parce que C n'accepte pas l'idée de plusieurs processus inclus dans un seul morceau de code monolithique. Y a-t-il vraiment une différence entre un micro-noyau écrit en plusieurs morceaux de C et un noyau monolithique écrit dans le style d'OCCAM ? Je suis enclin à penser que dans ce cas le concept monolithique serait meilleur que le concept micro-noyau, puisqu'avec l'avantage d'avoir un langage homogène, le noyau peut être conçu de manière encore plus modulaire que celui de MINIX.

MINOX cherche un défenseur :-)

-bat.

**De:** kt4@prism.gatech.EDU (Ken Thompson)  
**Sujet:** Re : LINUX est dépassé  
**Date:** 3 Fév 92 23:07:54 GMT  
**Organisation:** Georgia Institute of Technology

Le point de vue peut être dissocié du côté utilitaire.  
Surtout si ce n'est pas la majorité du logiciel utilisé qui est peut-être dépassé, suivant le tout dernier critère de conception. La plupart des utilisateurs se soucieront peu du fait que la conception interne du système d'exploitation qu'ils utilisent est dépassée. Ils seront beaucoup plus intéressés par les performances et les possibilités au niveau utilisation.

Je serais plutôt d'accord avec le fait que les micro-noyaux vont probablement dans le sens de l'histoire. Cependant, à mon avis, il est plus facile d'implémenter un noyau monolithique. Et il est également plus facile pour lui de tourner rapidement en eau de boudin en cas de modification.

Salutations,  
Ken

**De:** kevin@taronga.taronga.com (Kevin Brown)  
**Sujet:** Re : LINUX est dépassé  
**Date:** 4 Fév 92 08:08:42 GMT  
**Organisation:** University of Houston

Dans l'article <47607@hydra.gatech.EDU> kt4@prism.gatech.EDU  
(Ken Thompson) écrit :

>Le point de vue peut être dissocié du côté utilitaire.  
>Surtout si ce n'est pas la majorité du logiciel utilisé  
>qui est peut-être dépassé, suivant le tout dernier critère  
>de conception. La plupart des utilisateurs se soucieront  
>peu du fait que la conception interne du système d'exploitation  
>qu'ils utilisent est dépassée. Ils seront beaucoup plus intéressés  
>par les performances et les possibilités au niveau utilisation.  
>  
>Je serais plutôt d'accord avec le fait que les micro-noyaux vont  
>probablement dans le sens de l'histoire. Cependant, à mon avis,  
>il est plus facile d'implémenter un noyau monolithique. Et il est  
>également plus facile pour lui de tourner rapidement en eau de  
>boudin en cas de modification.

Quelle difficulté y a-t-il à structurer l'arborescence des  
programmes source d'un noyau monolithique en sorte que  
la plupart des modifications  
n'aient pas de répercussions vraiment négatives sur les sources ?  
Dans quelles sortes de pièges vous précipitez-vous avec ce type  
de comportement, et quelles suggestions pouvez-vous faire pour  
vous en sortir ?

Voilà ce que je demande : qu'y a-t-il de difficile à organiser  
les sources de manière à ce que la plupart des changements  
effectués dans le noyau restent localisés quant à leurs conséquences,  
même si le noyau est lui-même monolithique ?

Je pense que vous avez des années d'expérience sur les noyaux  
monolithiques :-), dès lors je pense que vous essayerez de répondre  
au mieux à de telles interrogations.

Kevin Brown

**De:** rburns@finess.Corp.Sun.COM (Randy Burns)  
**Sujet:** Re : LINUX est dépassé  
**Date:** 30 Jan 92 20:33:07 GMT  
**Organisation:** Sun Microsystems, Mt. View, Ca.

Dans l'article <12615@star.cs.vu.nl> ast@cs.vu.nl (Andy Tanenbaum)  
écrit :

>Dans l'article <1992Jan29.231426.20469@klaava.Helsinki.FI>  
>torvalds@klaava.Helsinki.>FI (Linus Benedict Torvalds) écrit :

>Bien sûr dans 5 ans ça sera différent, mais dans 5 ans tout le monde  
>emploiera le GNU gratuit sur sa station SPARC-5 à 200 MIPS, 64M.

Et bien, je suis de ceux qui aimeraient que cela arrive.

>>Le fait est que linux est plus facilement portable que minix. Quoi ?  
>>vous entends-je dire. C'est vrai - mais pas dans le sens où l'entend  
>>ast : j'ai conçu linux aussi conforme qu'il m'était possible aux

>>normes (sans avoir de copie des normes POSIX sous la main).  
>>Porter des choses sous linux est généralement /beaucoup/ plus facile  
>>que de les porter sous minix.

.....

>Je pense qu'écrire un nouveau système d'exploitation  
>étroitement lié à un matériel particulier, surtout aussi étrange  
>que la ligne Intel, est fondamentalement mauvais.  
En premier lieu, les parties de Linux qui s'adaptent le mieux aux  
80x86 sont le noyau et les pilotes de périphériques.  
Mon sentiment est que même si Linux n'est simplement qu'un pis-aller  
pour nous permettre d'employer des logiciels GNU, cela vaut  
encore la peine d'avoir un noyau bien adapté pour le plus grand  
nombre d'architectures existantes.

>Un SE lui-même doit  
>être facilement portable vers de nouvelles plates-formes matérielles.  
Bien, la seule partie de Linux qui ne soit pas portable est le noyau  
et les pilotes de périphériques. Par rapport aux compilateurs,  
aux utilitaires, aux systèmes de fenêtrage, etc. c'est vraiment une  
petite partie de l'effort. Étant donné que Linux possède une grande  
capacité de compatibilité dans les appels avec les SE portables, je  
ne m'en plaindrais pas. Je suis personnellement très heureux  
d'avoir un SE qui fait que certains d'entre nous pourront  
tirer avantage des logiciels provenant de Berkeley, FSF, CMU, etc.

Peut-être que dans 2 à 3 ans les variantes super économiques BSD et  
Hurd feront des petits, et Linux sera dépassé. Mais jusqu'à présent  
Linux réduit de manière notable le coût d'utilisation des outils  
tels que gcc, bison, bash qui servent pour le développement d'un tel  
SE.

**De:** torvalds@klaava.Helsinki.FI (Linus Benedict Torvalds)  
**Sujet:** Re : LINUX est dépassé  
**Date:** 31 Jan 92 10:33:23 GMT  
**Organisation:** University of Helsinki

Dans l'article <12615@star.cs.vu.nl> ast@cs.vu.nl (Andy Tanenbaum)  
écrit :

>Les limitations de MINIX sont liées en partie au fait que  
>je suis professeur : j'avais comme objectif précis de le faire  
>fonctionner sur du matériel bon marché afin que les étudiants puissent  
>se l'offrir.

D'accord : c'est un point technique indiscutable, ce qui rend certains  
de mes commentaires inexcusables. Mais en même temps vous vous emmêlez  
un peu les pinceaux : maintenant vous admettez que certaines erreurs  
de minix proviennent de sa trop grande portabilité : y compris pour  
des machines qui n'étaient pas vraiment conçues pour  
unix. Cette supposition induit alors que minix ne pourrait  
pas facilement être développé en y incluant des choses telles que  
la pagination, même avec des machines pouvant l'accepter.  
Oui, minix est portable, mais vous pouvez reformuler ceci  
comme « n'utilisant aucune fonctionnalité », et ce serait toujours  
vrai.

>Un système de fichiers multi-thread est seulement un bitouillage  
>d'amélioration des performances.

Ce n'est pas vrai. C'est un bitouillage d'amélioration des  
performances /avec un micro-noyau/, mais cela devient automatique lorsque

vous écrivez un noyau monolithique - c'est un domaine où les micro-noyaux ne sont pas très performants (comme je le faisais remarquer dans ma correspondance privée avec ast). Lorsque vous concevez un unix de manière « dépassée », vous obtenez automatiquement un noyau multithread : chaque processus fait son propre travail, et vous n'avez pas à faire de vilaines choses telles que des files d'attente de messages pour qu'il fonctionne de manière efficace.

De plus, il y a des gens qui considèrent « seulement un bitouillage d'amélioration des performances » comme vital : excepté le cas où vous possédez un cray-3, je pense que tout le monde se laisserait d'attendre le micro-ordinateur tout le temps. Je sais que je l'ai fait avec minix (bien sûr, je l'ai fait avec linux aussi, mais il est /beaucoup/ mieux).

>Je persiste à penser que concevoir un noyau monolithique en >1991 est une erreur fondamentale. Estime-toi heureux de ne pas être >un de mes étudiants. Tu n'obtiendrais pas une bonne note pour une >telle conception :-)

Bien, je n'aurais pas eu de très bonnes notes même avec vous : j'ai eu une discussion (sans rapport aucun - même pas avec les SE) avec la personne qui enseigne ici à l'université la conception des SE. Je me demande quand j'apprendrai :)

>Je pense qu'écrire un nouveau système d'exploitation >étroitement lié à un matériel particulier, surtout aussi étrange >que la ligne Intel, est fondamentalement mauvais. Un SE lui-même doit

Mais /mon/ point de vue est que le système d'exploitation /n'est pas/ lié à un type quelconque de microprocesseur : UNIX fonctionne sur la plupart des processeurs existants. Oui, /l'implémentation/ dépend de l'architecture, mais c'est une ÉNORME différence. Vous citez OS/360 et MS-DOG comme exemples de mauvaise conception car ils sont dépendants du matériel, et je suis d'accord avec vous. Mais il y a une grande différence entre ceux-ci et linux : l'API de linux est portable (non pas parce que je l'ai conçu astucieusement, mais parce que j'ai décidé de faire un unix plutôt-bien-pensé et testé).

Si vous écrivez des programmes pour linux à ce jour, vous ne devriez pas avoir trop de surprises en les recompilant pour Hurd au 21ème siècle. Comme remarqué (et pas seulement par moi), le noyau linux est de toute petite partie d'un système complet : les sources complètes de linux prennent environ 200 Ko sous forme compactée - les sources complètes d'un système en développement prennent au moins 10 Mo sous forme compactée (et facilement beaucoup, beaucoup plus). Et les sources complètes sont portables, sauf pour le minuscule noyau que vous pouvez (pour preuve : je l'ai fait) réécrire totalement à partir de rien en moins d'un an sans avoir /aucune/ connaissance préalable.

De fait, le noyau linux /complet/ est beaucoup plus petit que ce qui concerne le 386 dans mach : i386.tar.Z pour la version actuelle de Mach fait plus de 800 Ko compactés (83391 octets selon nic.funet.fi). Il est vrai que mach est « quelque peu » plus gros et a plus de fonctionnalités, mais cela devrait toujours vous dire quelque chose.

Linus

**De:** kaufman@eecs.nwu.edu (Michael L. Kaufman)  
**Sujet:** Re : LINUX est dépassé  
**Date:** 3 Fév 92 22:27:48 GMT  
**Organisation:** EECS Department, Northwestern University

J'ai essayé d'envoyer ces deux réponses depuis mon travail, mais pense qu'elles ont disparu. Désolé si vous les avez déjà lues.

Andy Tanenbaum écrit un article intéressant (de même, il est intéressant de savoir qu'il lit effectivement ce groupe de discussion) mais je pense qu'il oublie un point important.

Il a écrit :

> Comme la plupart de vous le savent, pour moi MINIX est un  
> passe-temps

Ce qui est probablement vrai pour la plupart des personnes, si ce n'est toutes, qui travaillent sur Linux. Nous ne développons pas un système pour conquérir le marché des systèmes d'exploitation, nous passons seulement un bon moment.

> Ils prennent peu à peu le pas sur la lignée des 80x86 et  
> exécuteront de vieux programmes MS-DOS en simulant le 80386  
> au niveau logiciel

Bien, quand ceci arrivera et si je désire toujours jouer avec Linux, je pourrai le lancer sur mon simulateur de processeur 386.

> MINIX a été conçu pour être raisonnablement portable, et a été porté  
> de la ligne Intel aux 680x0 (Atari, Amiga, Macintosh), SPARC, et  
> NS32016. Linux est attaché de très près au 80x86. C'est une mauvaise  
> approche.

C'est très bien pour les personnes qui possèdent ces machines, mais cela ne s'est pas passé sans dommage. Cette portabilité fut atteinte au détriment des performances et de quelques caractéristiques propres au 386. Avant de décréter que Linux n'est pas sur le bon chemin, vous devriez penser à quoi il va servir. Je vais l'utiliser sur mon 486 pour lancer des applications graphiques très gourmandes en calcul et en mémoire.

>Mais en toute honnêteté, je voudrais suggérer à ceux qui veulent un  
>SE "libre" \*\*MODERNE\*\* qu'ils recherchent un SE à  
>micro-noyau, portable, comme peut-être GNU ou quelque chose comme ça.

Je ne connais pas de système d'exploitation micro-noyau portable et libre. GNU Hurd est toujours dans les limbes et va le rester pour un moment encore. En avez-vous un à recommander, ou bien me taquinez-vous ? ;-)

Dans l'article <12615@star.cs.vu.nl> ast@cs.vu.nl (Andy Tanenbaum) écrit :

>Je pense qu'écrire un nouveau système d'exploitation  
>étroitement lié à un matériel particulier, surtout aussi étrange  
>que la ligne Intel, est fondamentalement mauvais. Un SE lui-même doit  
>être facilement portable vers de nouvelles plates-formes matérielles.

Je pense comprendre nos points de désaccord. Vous regardez la conception de systèmes d'exploitation comme une fin en soi. Minix est bien car il est portable, micro-noyau, etc. Linux est mauvais car il est hermétiquement attaché à Intel. Ce n'est pas une étrange attitude

venant de quelqu'un du monde académique, mais vous ne devriez pas vous attendre à la voir universellement partagée. Linux n'est pas conçu comme un outil pédagogique, ou un exercice abstrait. Il est écrit pour permettre aux gens d'exécuter des logiciels GNU aujourd'hui. Le fait qu'il ne puisse pas être utilisé dans cinq ans est moins important que le fait qu'aujourd'hui, je peux lancer dessus toutes sortes de logiciels que je veux. Vous maintenez que Minix est meilleur, mais s'il ne permet pas d'exécuter les logiciels que je veux, il n'est pas du tout bon (pour moi).

>Le fait qu'OS/360 fut écrit en langage d'assemblage IBM 360 il y a 25  
>ans peut probablement être excusé. Lorsque MS-DOS fut écrit  
>explicitement pour le 8088 il y a 10 ans, c'était moins que brillant,  
>comme le réalisent trop péniblement IBM et Microsoft maintenant  
>seulement.

Même point. Microsoft n'est pas sorti avec DOS pour « explorer les frontières des systèmes d'exploitation ». Ils l'ont écrit pour faire de l'argent. En considérant le fait que MS-DOS se vend en plus grande quantité que tout le reste réuni, vous ne pouvez pas dire qu'ils ont raté leur but. Ce n'est pas que MS-DOS soit le meilleur système d'exploitation suivant d'autres caractéristiques, mais il a servi leur besoin.

Michael

**De:** julien@incal.inria.fr (Julien Maisonneuve)  
**Sujet:** Re : LINUX est dépassé  
**Date:** 3 Fév 92 17:10:14 GMT

J'aimerais apporter de l'eau au moulin de Kevin Brown pour la plupart de ses remarques.

J'ajouterai quelques considérations propres à l'utilisateur :

- Quand ast affirme que le multi-thread pour un système de fichiers est inutile, cela me rappelle toutes les fois où j'ai voulu laisser une tâche en arrière-plan (comme la lecture d'une archive depuis une disquette), c'est tout simplement inutilisable, l'opérateur du shell & aurait pu tout aussi bien être laissé de côté.
  - Les utilitaires les plus intéressants ne sont même pas compilables sous Minix en raison des limitations incroyables du compilateur d'ATK. On pouvait les admettre à la rigueur sur un PC de base, sur un 386, cela devient absurde. N'importe quel bête compilateur DOS est doté d'un modèle mémoire 'large' (plus cher, d'accord). Je déteste le compactage 13 bits !
  - L'absence de gestion de la mémoire virtuelle interdit d'étudier ce domaine à des fins d'expérimentation et d'utiliser des programmes de taille importante.
- La conception étrange du MM le rend également difficile à modifier.

Le problème est que même des travaux d'exploration sont pénibles sous minix. Si vous voulez accomplir une tâche (ou même vous amuser), même DOS devient un meilleur choix (avec des choses telles que DJ GPP). À la base, ce n'est rien d'autre qu'un exemple didactique de SE, un beau jouet, certes, mais un jouet quand même. Récupérer et appliquer des patches est fastidieux et exclut les mises à jour ultérieures.

Domage quand il manque si peu pour en faire quelque chose de vraiment bien. Merci pour le travail Andy, mais Linux ne méritait pas votre réponse. Pour le commun des mortels, il fait beaucoup de choses mieux que Minix.

Julien Maisonneuve.

Ce n'est pas une descente en flamme, seulement mon expérience.

**De:** richard@aiai.ed.ac.uk (Richard Tobin)  
**Sujet:** Re : LINUX est dépassé  
**Date:** 4 Fév 92 14:46:49 GMT  
**Répondre à:** richard@aiai.UUCP (Richard Tobin)  
**Organisation:** AIAI, University of Edinburgh, Scotland

Dans l'article <12615@star.cs.vu.nl> ast@cs.vu.nl (Andy Tanenbaum) écrit :  
>Un système de fichiers multi-thread est seulement un bitouillage  
>d'amélioration des  
>performances. Lorsqu'il y a seulement un processus actif, cas normal  
>sur un petit PC, cela ne vous apporte rien

Je trouve l'utilisation d'un système de fichiers à thread unique particulièrement pénible avec Minix. J'ai souvent besoin de faire autre chose pendant que je lis des fichiers depuis une (atrociement lente) disquette. Je préfère jouer au solitaire pendant les grosses compilations de C ou de Lisp. J'aime bien examiner des fichiers dans le tampon d'un éditeur tout en compilant dans un autre.

(Le problème serait sans doute moindre si le système de fichiers se contentait de servir des fichiers au lieu d'interagir avec les entrées/sorties de terminal).

Évidemment, sur du Minix de base, sans consoles virtuelles ni aucune possibilité de lancer emacs, ce n'est pas vraiment un problème. Mais pour la plupart des gens, c'est un inconvénient, pas un avantage. Ce n'est pas simplement le fait que sur des machines mono-utilisateur il n'y a pas besoin de plus d'un processus actif. Cette idée n'est envisageable que parce que tant de gens sont habitués à des machines médiocres dotées de systèmes d'exploitation médiocres.

Quant à la portabilité, Minix ne l'emporte que grâce à ses prétentions limitées. Si vous vouliez un Unix complet avec pagination, ordonnanceur, système de fenêtrage et ainsi de suite, serait-il plus rapide de partir d'un Minix de base et d'ajouter ces fonctionnalités, ou depuis Linux et de s'occuper des spécificités du 386 ? Je ne crois pas qu'il soit juste de critiquer Linux quand ses objectifs sont si différents de ceux de Minix. Si on veut un système à but pédagogique, Minix est la réponse. Mais s'il s'agit d'avoir un environnement aussi proche que possible de celui d'une Sun (par exemple) sur son ordinateur personnel, il a quelques lacunes.

-- Richard

**De:** ast@cs.vu.nl (Andy Tanenbaum)  
**Sujet:** Re : LINUX est dépassé  
**Date:** 5 Fév 92 14:48:48 GMT  
**Organisation:** Fac. Wiskunde & Informatica, Vrije Universiteit, Amsterdam

Dans l'article <6121@skye.ed.ac.uk> richard@aiai.UUCP (Richard Tobin) écrit :  
> Si vous vouliez un Unix complet avec pagination,  
> ordonnanceur, système de fenêtrage et ainsi de suite, serait-il plus

> rapide de partir d'un Minix de base et d'ajouter ces fonctionnalités,  
> ou depuis Linux et de s'occuper des spécificités du 386 ?

Une autre option qui semble être totalement oubliée ici est d'acheter UNIX ou un clone. Si vous voulez juste UTILISER le système, plutôt que de bitouiller dans ses entrailles, vous n'avez pas besoin du code source. Coherent est à seulement 99 \$, et il y a plusieurs vrais systèmes UNIX avec plus de fonctionnalités pour plus d'argent. Pour le vrai bitouilleur, ne pas avoir le code source est fatal, mais pour ceux qui veulent juste un système UNIX, il y a beaucoup d'autres options (non gratuites).

Andy Tanenbaum ([ast@cs.vu.nl](mailto:ast@cs.vu.nl))

**De:** [ajt@doc.ic.ac.uk](mailto:ajt@doc.ic.ac.uk) (Tony Travis)

**Sujet:** Re : LINUX est dépassé

**Date:** 6 Fév 92 02:17:13 GMT

**Organisation:** Department of Computing, Imperial College, University of London, UK.

[ast@cs.vu.nl](mailto:ast@cs.vu.nl) (Andy Tanenbaum) écrit :

>Une autre option qui semble être totalement oubliée ici est d'acheter  
>UNIX ou un clone. Si vous voulez juste UTILISER le système, plutôt que  
>de bitouiller dans ses entrailles, vous n'avez pas besoin du code  
>source. Coherent est à seulement 99 \$, et il y a plusieurs vrais  
>systèmes UNIX avec plus de fonctionnalités pour plus d'argent. Pour  
>le vrai bitouilleur, ne pas avoir le code source est fatal, mais pour  
>ceux qui veulent juste un système UNIX, il y a beaucoup d'autres options  
>(non gratuites).

Andy, j'ai suivi le développement de Minix depuis que les premiers articles ont été postés dans ce groupe et emploie actuellement la version 1.5.10 avec les correctifs de Bruce Evans pour le 386.

Je veux 'simplement' un Unix sur mon PC et bitouiller ses entrailles ne m'intéresse pas, mais je \*veux\* le code source !

La philosophie qui consiste à créer à partir du travail des autres est un des grands principes sous-jacents qui explique le succès et la popularité d'Unix.

Cette philosophie repose sur la disponibilité du code source afin qu'il puisse être examiné, modifié et réutilisé pour des nouveaux programmes.

Il y a des années de cela, j'étais l'heureux bénéficiaire d'une licence pour le code source Unix AT&T 7ème édition mais, même alors, j'ai accueilli votre décision de rendre celui de Minix disponible comme une libération par rapport aux entraves du copyright AT&T !!

Il me semble que vous devez parfois oublier que votre 'hobby' a eu un effet déterminant sur la disponibilité d'un Unix 'personnel' (c-à-d financièrement accessible) et que le PC 8086 sur lequel tournait Minix 1.2 m'a coûté en fait bien plus que le clone 386/SX que j'utilise à présent.

Il est clair que Minix n'est pas la panacée, mais je vois l'évolution vers des versions 386 et le 68000 ou d'autres

architectures à espace d'adressage linéaire d'une manière assez semblable : c'est une bonne chose pour les gens comme moi qui utilisent Minix et se sentent gênés par l'architecture segmentée de la version PC dans le domaine des applications.

RIEN de ce que vous pouvez dire ne pourrait me convaincre d'utiliser Coherent...

Tony

**De:** richard@aiai.ed.ac.uk (Richard Tobin)  
**Sujet:** Re : LINUX est dépassé  
**Date:** 7 Fév 92 14:58:22 GMT  
**Organisation:** AIAI, University of Edinburgh, Scotland

Dans l'article <12696@star.cs.vu.nl> ast@cs.vu.nl (Andy Tanenbaum) écrit :

> Si vous voulez juste UTILISER le système, plutôt que de bitouiller  
> dans ses entrailles, vous n'avez pas besoin du code source.

Beaucoup d'entre nous souhaitent pouvoir bitouiller dans les entrailles du système... Vous serez débarrassé de la plupart d'entre nous quand BSD-detox ou GNU sortiront, ce qui devrait arriver dans les tous prochains mois (ouais, exactement).

Richard

**De:** comm121@unixg.ubc.ca (Louie)  
**Sujet:** Re : LINUX est dépassé  
**Date:** 30 Jan 92 02:55:22 GMT  
**Organisation:** University of British Columbia, Vancouver, B.C., Canada

Dans <12595@star.cs.vu.nl> ast@cs.vu.nl (Andy Tanenbaum) écrit :

>Mais en toute honnêteté, je voudrais suggérer à ceux qui veulent un  
>SE "libre" \*\*MODERNE\*\* qu'ils recherchent un SE à  
>micro-noyau, portable, comme peut-être GNU ou quelque chose comme ça.

Il n'y a réellement pas d'autre choix que Linux pour les gens qui, comme moi, veulent un SE "libre". Considérant que la majorité des personnes susceptibles d'opter pour un SE "libre" utilisent le 386, la portabilité n'est vraiment pas un si gros problème. Si j'avais une Sparc, j'utiliserais Solaris.

Jusqu'ici, j'ai installé Linux avec gcc, emacs 18.57, kermi et tous les utilitaires GNU sans le moindre ennui. Nul besoin d'appliquer des correctifs. J'ai simplement suivi les instructions d'installation. Je ne peux pas trouver un tel SE \*n'importe où\* au même prix pour mes devoirs d'informatique. Et il semble que la gestion du réseau ainsi que X-Window seront portés sous Linux bien avant Minix. C'est quelque chose qui serait vraiment utile. À mon avis, la portabilité des programmes Unix standards est également importante.

Je sais que le concept du système monolithique n'est pas aussi bon que celui du micro-noyau. Mais à court terme (et je sais que je ne voudrais/pourrais pas faire évoluer mon 386), Linux me convient parfaitement.

Philip Wu  
[pwu@unixg.ubc.ca](mailto:pwu@unixg.ubc.ca)

**De:** [dgraham@bmers30.bnr.ca](mailto:dgraham@bmers30.bnr.ca) (Douglas Graham)  
**Sujet:** Re : LINUX est dépassé  
**Date:** 1 Fév 92 00:26:30 GMT  
**Organisation:** Bell-Northern Research, Ottawa, Canada

Dans l'article <12595@star.cs.vu.nl> [ast@cs.vu.nl](mailto:ast@cs.vu.nl) (Andy Tanenbaum)  
écrit :

> Bien que je pourrais me lancer ici dans une longue histoire sur les  
> mérites relatifs des deux concepts, je me contenterai de dire que  
> parmi les personnes concevant à l'heure actuelle des systèmes  
> d'exploitation, la discussion est essentiellement close. Les  
> micro-noyaux ont gagné.

Pouvez-vous recommander une documentation (sans parti pris)  
qui traite des forces et faiblesses des deux approches ? Je suis  
certain qu'il y a quelque chose à dire sur l'approche  
micro-noyau, mais je m'interroge sur la similitude de Minix vis à vis  
des autres systèmes qui utilisent cette approche. Bien sûr, Minix  
fait appel à beaucoup de tâches et de messages, mais il en faudrait  
plus pour qu'il ait une architecture de micro-noyau. Je suspecte  
que le code Minix ne soit pas découpé en tâches de manière optimale.

> micro-noyaux ont gagné. Le seul vrai argument pour les systèmes  
> monolithiques était la performance, et il est maintenant assez  
> évident que les systèmes à micro-noyau peuvent être tout aussi  
> rapides que les systèmes monolithiques (par ex., Rick Rashid a  
> publié des notes comparant Mach 3.0 à des systèmes monolithiques),  
> et il ne reste donc maintenant que des détails à régler.

Mon principal grief envers Minix ne concerne pas ses performances.  
Ajouter des fonctionnalités est un calvaire - ce que,  
je suppose, une architecture micro-noyau est supposée alléger.

> MINIX est construit autour d'un micro-noyau. Le système de fichiers

Y a-t-il un consensus à ce sujet ?

> LINUX est un système de style monolithique. C'est un pas de géant en  
> arrière dans les années 1970. C'est comme prendre un programme C  
> existant et qui marche et le réécrire en BASIC. Pour moi, écrire un  
> système monolithique en 1991 est une idée vraiment médiocre.

C'est une affirmation pertinente, mais je cherche encore à voir  
une raison d'être pour cela.

Je crois que Linux consiste seulement en 12000 lignes de codes.  
Je ne vois pas en quoi le diviser en tâches et en messages  
à droite et à gauche l'améliorerait.

> Ne le prends pas mal, je ne suis pas mécontent de LINUX. Il va amener  
> tous ceux qui veulent faire évoluer MINIX en BSD UNIX derrière moi.  
> Mais en toute honnêteté, je voudrais suggérer à ceux qui veulent un  
> SE "libre" \*\*MODERNE\*\* qu'ils recherchent un SE à  
> micro-noyau, portable, comme peut-être GNU ou quelque chose comme ça.

Bien, il n'y a pas d'autres choix sur lesquels je puisse me porter  
en ce moment. Mais lorsque le SE de GNU viendra, j'aimerais bien

reprendre ce bateau. Je sens que vous \*êtes\* quelque peu insatisfait de Linux (et cela me surprend un peu). Je suis enclin à penser que la raison pour laquelle tant de personnes s'y intéressent est qu'il offre plus de possibilités. Votre approche sur les gens désirant des fonctionnalités dans Minix a été de dire qu'ils n'en voulaient pas vraiment. Je prétends que l'exode vers Linux prouve que vous avez tort.

Avertissement : je n'ai rien à voir avec le développement de Linux. Je pense simplement que c'est un système plus facile à comprendre que Minix.

--

Doug Graham dgraham@bnr.ca Mes opinions sont les miennes propres.

**De: hedrick@klinzhai.rutgers.edu (Charles Hedrick)**

**Sujet: Re : LINUX est dépassé**

**Date: 1 Fév 92 00:27:04 GMT**

**Organisation: Rutgers Univ., New Brunswick, N.J.**

L'histoire du logiciel montre qu'à chaque fois la disponibilité l'emporte sur la qualité technique. Voilà le principal

avantage de Linux. C'est un petit système pour 386, tout-à-fait compatible avec l'Unix d'origine, et qui est disponible librement. J'ai laissé tombé la communauté Minix il y a quelques années quand il devint évident que (1) Minix ne tirerait profit de rien à part du 8086 dans le futur proche, et (2) que la licence - bien qu'étonnamment bienveillante - le rendait toujours difficile d'accès pour ceux qui voudraient produire une version pour le 386. Apparemment plusieurs personnes ont effectué un bon travail pour le 386. Mais elles ne pouvaient distribuer que des fichiers diffs. Cela rend le travail sur un 386 peu pratique pour un nouvel utilisateur, et de fait je n'étais pas sûr de vouloir le faire.

Excusez-moi si les choses ont changé ces dernières années. S'il est maintenant possible d'avoir une version 386 prête à l'emploi, la communauté a développé un moyen de distribuer les sources Minix, et que des programmes Unix normaux est dorénavant devenu plus facile, alors je reconsidérerai ma position en ce qui concerne Minix. J'aime bien sa conception.

Il est possible que Linux soit rattrapé par Gnu ou un BSD libre. Cependant, si le SE Gnu suit l'exemple de tous les autres logiciels Gnu, il aura besoin de 128 Mo de mémoire et d'un disque de 1 Go. Il y aura toujours de la place pour un petit système. Mon système idéal serait le 4.4 BSD, mais la date de la version du 4.4 a considérablement dérapé dans le temps. Et puis avec toute l'équipe migrant vers BSDI, il est difficile de croire que la situation va aller en s'améliorant. Pour mon usage personnel, le système BSDI sera certainement très bien. Mais même leur prix très attractif est vraiment trop élevé pour la plupart de nos étudiants, et bien que les utilisateurs puissent en acquérir les sources, le fait que certaines soient propriétaires signifie encore que vous ne pourrez que mettre du code modifié sur les sites FTP accessibles au public. De toutes façons Linux existe et tout le reste n'est que littérature.

**De:** tytso@athena.mit.edu (Theodore Y. Ts'o)  
**Sujet:** Re : LINUX est dépassé  
**Date:** 31 Jan 92 21:40:23 GMT  
**Organisation:** Massachusetts Institute of Technology  
**En réponse à:** message d'ast@cs.vu.nl du 29 Jan 92 12: 12:50 GMT

>De : ast@cs.vu.nl (Andy Tanenbaum)

>dans le répertoire minix/simulator). Je pense que c'est une erreur  
>grossière que de concevoir un SE pour une architecture spécifique,  
>alors que celle-ci ne sera bientôt plus disponible.

Ce n'est pas de votre faute si vous croyez que Linux est limité à l'architecture 80386, puisque de nombreux adeptes de Linux (y compris Linus lui-même) sont partis de ce principe. Néanmoins, la quantité de code spécifique 80386 n'est probablement pas plus importante que dans une implémentation Minix, et il n'y a certainement pas plus de code spécifique 80386 dans Linux que de code spécifique Vax dans BSD 4.3.

Ceci dit, le portage sur d'autres architectures n'a pas encore été réalisé. Mais si je commençais à porter un système de type Unix vers une nouvelle architecture, je commencerais probablement par Linux plutôt que Minix, simplement parce que je veux avoir un certain contrôle sur ce que je pourrais faire avec le système résultant une fois que j'aurais fini.

Oui, j'aurais à réécrire de grandes portions du VM et des couches des gestionnaires de périphériques -- mais j'aurais à le faire avec n'importe quel SE. Cela serait peut-être un petit peu plus difficile que pour porter Minix sur la nouvelle architecture, mais cela ne serait probablement vrai que pour la première architecture cible du portage de Linux.

>Bien que je pourrais me lancer ici dans une longue histoire sur les  
>mérites relatifs des deux concepts, je me contenterai de dire que  
>parmi les personnes concevant à l'heure actuelle des systèmes  
>d'exploitation, la discussion est essentiellement close. Les  
>micro-noyaux ont gagné. Le seul vrai argument pour les systèmes  
>monolithiques était la performance, et il est maintenant assez  
>évident que les systèmes à micro-noyau peuvent être tout aussi  
>rapides que les systèmes monolithiques (par ex., Rick Rashid a  
>publié des notes comparant Mach 3.0 à des systèmes monolithiques),  
>et il ne reste donc maintenant que des détails à régler.

Ce n'est pas forcément le cas. Je pense que vous peignez une vue plus manichéenne de l'univers qu'il ne l'est. Je vous renvoie à des articles du genre de celui de Brent Welsh « le système de fichiers est intrinsèque au noyau » dans lequel il estime que le système de fichiers est une abstraction suffisamment mature pour qu'elle réside dans le noyau, et non en-dehors comme ce serait le cas dans une conception micro-noyau stricte.

Un certain nombre de gens sont préoccupés par la rapidité de OSF/1 Mach comparée à celle des systèmes monolithiques ; plus spécialement, le nombre de changements de contexte nécessaires à la gestion du trafic réseau, et des systèmes de fichiers réseau en particulier.

Je connais les bénéfices d'une approche micro-noyau. Néanmoins, le fait reste que Linux est là, et que GNU non - et que les gens ont travaillé sur Hurd beaucoup plus longtemps que Linus n'a travaillé sur Linux.

Minix ne compte pas parce qu'il n'est pas libre. :-)

Je suspecte que le choix entre micro-noyaux et noyaux monolithiques dépend de ce que l'on en fait. Si l'on s'intéresse à la recherche, il est évidemment bien plus facile de retirer et de remplacer des modules dans un micro-noyau, et puisque seuls les chercheurs écrivent des articles sur les systèmes d'exploitation, ipso facto les micro-noyaux doivent être la bonne approche. Néanmoins, je connais des tas de gens qui ne sont pas chercheurs, mais au contraire des programmeurs noyau pratiques, dont les préoccupations principales sont dans le coût de copie et de changement de contexte inhérents à un micro-noyau.

Pour autant, je n'adhère pas à votre argument selon lequel un système de fichiers multi-thread n'est pas nécessaire sur un système mono-utilisateur. Une fois lancé un système de fenêtrage, ainsi qu'une compilation dans une fenêtre, un lecteur de courrier dans une autre et les nouvelles arrivant par UUCP en arrière-plan, il vous faut de bonnes performances du système de fichiers, même sur un système mono-utilisateur. C'est peut-être une optimisation inutile aux yeux du théoricien et un bitouillage de performance (pour reprendre vos propres termes), mais je m'intéresse à un Vrai système d'exploitation - pas à un jouet de chercheur.

Theodore Ts'o bloom-beacon!mit-athena!tytso  
308 High St., Medford, MA 02155 tytso@athena.mit.edu  
Tous jouent au jeu, mais aucun avec les mêmes règles !

**De:** joe@jshark.rn.com  
**Sujet:** Re : LINUX est dépassé  
**Date:** 31 Jan 92 13:21:44 GMT  
**Organisation:** a blip of entropy

Dans l'article <12595@star.cs.vu.nl> ast@cs.vu.nl (Andy Tanenbaum) écrit :

>  
> MINIX a été conçu pour être raisonnablement portable, et a été porté  
> de la ligne Intel aux 680x0 (Atari, Amiga, Macintosh), SPARC, et  
> NS32016. Linux est attaché de très près au 80x86. C'est une mauvaise  
> approche.

Si vous regardiez le source au lieu de croire l'auteur, vous réaliseriez que ce n'est pas vrai !

Il a remplacé "fubyte" par une routine qui utilise explicitement un registre de segment - mais cela peut être facilement modifié. De même, à part à quelques endroits qui présument la présence d'une MMU 386, deux ou trois macros qui cachent les tailles exactes de pages, etc. rendraient triviale la tâche de portage. L'utilisation des TSS 386 simplifient le code, mais le VAX et le WE32000 ont des structures similaires.

Comme il l'a déjà admis, un peu de planification rendrait le système plus propre, mais ce n'est pas un crime de mettre un peu de langage d'assemblage 386 ici ou là !

Et avec tout le respect que je dois :  
- le Livre ne fait pas un but de la portabilité (à part sur quelques "#ifdef M8088") ;  
- au moment de sa sortie, Minix dépendait de certaines

« fonctionnalités » du 8086 qui ont causé l'ire des utilisateurs de 68000.

>Andy Tanenbaum (ast@cs.vu.nl)

joe.

**De: entropy@wintermute.WPI.EDU (Lawrence C. Foard)**

**Sujet: Re : LINUX est dépassé**

**Date: 5 Fév 92 14:56:30 GMT**

**Organisation: Worcester Polytechnic Institute**

Dans l'article <12595@star.cs.vu.nl> ast@cs.vu.nl (Andy Tanenbaum) écrit :

>Ne le prends pas mal, je ne suis pas mécontent de LINUX. Il va amener  
>tous ceux qui veulent faire évoluer MINIX en BSD UNIX derrière moi.  
>Mais en toute honnêteté, je voudrais suggérer à ceux qui veulent un  
>SE "libre" **\*\*MODERNE\*\*** qu'ils recherchent un SE à  
>micro-noyau, portable, comme peut-être GNU ou quelque chose comme ça.

Je crois que certains points de votre position sont valides, bien que je ne sois pas sûr qu'un micro-noyau soit nécessairement meilleur. Il peut y avoir plus de sens de permettre une certaine combinaison des deux. Dans le code d'IPC que je suis en train d'écrire pour Linux, je vais inclure du code qui permettra aux gestionnaires de périphériques et aux systèmes de fichiers tourner en mode utilisateur. Cela sera significativement plus lent malgré tout, et je crois qu'il serait une erreur de tout sortir du noyau (TCP/IP sera interne).

En réalité, mon principal problème avec les théoriciens des SE est qu'ils n'ont jamais testé leurs idées ! Aucune de celles-ci (à l'exception partielle de MACH) n'a jamais vu le jour. Les ordinateurs personnels 32 bits sont disponibles depuis au moins une décennie et Linus a été le premier à écrire un SE qui fonctionne pour eux sans devoir verser 100 000\$ à AT&T. Un morceau de logiciel en mains vaut mieux que dix logiciels fictifs, les théoriciens des SE sautent vite sur un SE mais ne veulent même pas fournir une autre option.

Le consensus général que les micro-noyaux sont la direction à prendre ne veut rien dire tant qu'aucune application réelle n'a fonctionné sur l'un d'entre eux.

La sortie de Linux me permet d'essayer quelques-unes des idées que je souhaite expérimenter depuis des années, mais je n'avais jamais eu l'occasion de travailler avec le code source d'un SE qui fonctionne.

**De: ast@cs.vu.nl (Andy Tanenbaum)**

**Sujet: Re : LINUX est dépassé**

**Date: 5 Fév 92 23:33:23 GMT**

**Organisation: Fac. Wiskunde & Informatica, Vrije Universiteit, Amsterdam**

Dans l'article <1992Feb5.145630.759@wpi.WPI.EDU>

entropy@wintermute.WPI.EDU (Lawrence C. Foard) écrit :

>En réalité, mon principal problème avec les théoriciens des SE est  
>qu'ils n'ont jamais testé leurs idées !

Je suis mortellement injurié. JE NE SUIS PAS UN THÉORICIEN. Demandez à tous ceux qui étaient à la réunion du département hier

(je plaisante).

En fait, ces idées ont été très bien testées en pratique. OSF est en train de jouer son va-tout sur un micro-noyau (Mach 3.0). USL joue son affaire sur un autre (Chorus). Tous deux font tourner de nombreux logiciels et ont été intensivement comparés à des systèmes monolithiques. Amoeba a été complètement implémenté et testé pour un certain nombre d'applications. QNX est bâti autour d'un micro-noyau, et on m'a dit que le parc installé était de 200 000 systèmes. Les micro-noyaux ne sont pas des châteaux en Espagne. Ils reposent sur une approche éprouvée.

Les gars de Mach ont écrit un papier appelé « UNIX comme programme applicatif ». C'était de Golub et al., durant la conférence USENIX de l'été 1990. Les gens de Chorus ont eux aussi un rapport technique sur les performances des micro-noyaux, et je suis co-auteur d'un autre papier sur le sujet, que j'ai mentionné hier (Computing Systems de Déc. 1991). Jetez-y un œil.

Andy Tanenbaum ([ast@cs.vu.nl](mailto:ast@cs.vu.nl))

**De: peter@ferranti.com (peter da silva)**  
**Sujet: Re : LINUX est dépassé**  
**Date: Xenix Support, FICC**  
**Organisation: Jeu, 6 Fév 1992 16:02:47 GMT**

Dans l'article <12747@star.cs.vu.nl> ast@cs.vu.nl (Andy Tanenbaum) écrit :

> QNX est un système à micro-noyau, et on m'a dit que le parc  
> installé était de 200 000 systèmes.

Ah, oui, tant que je suis sur le sujet... il y a plus de 3 millions d'Amiga dans la nature, ce qui veut dire qu'il y en a plus que ce que n'importe quel constructeur d'UNIX a vendu, et probablement plus que tous les systèmes UNIX combinés.

**De: peter@ferranti.com (peter da silva)**  
**Sujet: Re : LINUX est dépassé**  
**Date: Xenix Support, FICC**  
**Organisation: Jeu, 6 Fév 1992 16:00:22 GMT**

Dans l'article <1992Feb5.145630.759@wpi.WPI.EDU> entropy@wintermute.WPI.EDU (Lawrence C. Foard) écrit :  
> En réalité, mon principal problème avec les théoriciens des SE est  
> qu'ils n'ont jamais testé leurs idées !

Je m'inscris en faux... il y a de nombreux systèmes à micro-noyau dans la nature pour tout depuis un 8088 (QNX) jusqu'à de grands systèmes de recherche.

> Aucune de celles-ci (à l'exception partielle de MACH) n'a jamais vu  
> le jour. Les ordinateurs personnels 32 bits sont disponibles depuis  
> au moins une décennie et Linus a été le premier à écrire un SE qui  
> fonctionne pour eux sans devoir verser 100 000\$ à AT&T.

Je dois avoir rêvé AmigaOS, alors. J'ai utilisé un produit de mon imagination durant les 6 dernières années.

AmigaOS a une conception de passage de messages de micro-noyau, avec un temps de réponse et des performances meilleurs que tout autre système d'exploitation pour PC disponible : y compris MINIX, OS/2, MS-Windows, MacOS, Linux, UNIX, et \*certainement\* MS-DOS.

La conception micro-noyau a prouvé sa valeur incalculable. Des choses comme de nouveaux systèmes de fichiers disponibles en principe seulement auprès de leur constructeur sont des produits de l'activité de loisirs sur l'Amiga. Les pilotes de périphériques sont simplement des bibliothèques partagées avec des points d'entrée spécifiques et des ports de messages. Il en est de même pour les systèmes de fichiers, le système de fenêtrage, etc. C'est une conception MERVEILLEUSE, qui valide tout ce que les gens ont pu dire sur les micro-noyaux. Oui, cela représente un plus gros travail de les faire décoller qu'un macro-noyau fondé sur les co-routines comme UNIX, mais la souplesse vous le rembourse plusieurs fois.

Je souhaite vraiment qu'Andy fasse un nouveau MINIX fondé sur ce qui a été appris depuis la première sortie. La construction des responsabilités dans MINIX est assez pauvre, mais le concept de base est bon.

> Le consensus général que les micro-noyaux sont la direction à prendre  
> ne veut rien dire tant qu'aucune application réelle n'a fonctionné  
> sur l'un d'entre eux.

Je rêve encore. J'ai sûrement imaginé que Deluxe Paint, Sculpt 3d, Photon Paint, Manx C, Manx SDB, Perfect Sound, Videoscape 3d, et les autres programmes que j'ai achetés étaient « réels ». Je vais avoir à renvoyer ces sacrés trucs, je pense.

La disponibilité de Linux est une excellente chose. Je suis ravi qu'il existe. Je suis sûr que sa conception macro-noyau est une des raisons de son implémentation si rapide, et c'est une raison valable d'utiliser ceux-ci. MAIS... cela ne veut pas dire que les micro-noyaux soient intrinsèquement lents, ni de simples jouets de chercheurs.

**De: dsmythe@netcom.COM (Dave Smythe)**

**Sujet: Re : LINUX est dépassé**

**Date: 10 Fév 92 07:08:22 GMT**

**Organisation: Netcom - Online Communication Services (408 241-9760 guest)**

Dans l'article <1992Feb5.145630.759@wpi.WPI.EDU>  
entropy@wintermute.WPI.EDU (Lawrence C. Foard) écrit :  
>En réalité, mon principal problème avec les théoriciens des SE est  
>qu'ils n'ont jamais testé leurs idées !  
>Aucune de celles-ci (à l'exception partielle de MACH) n'a jamais vu  
>le jour.

David Cheriton (Prof. à Stanford, et auteur du systemV) disait quelque chose de similaire dans son cours de systèmes distribués. En paraphrasant :

« Il y a deux types de chercheurs : ceux qui ont implémenté quelque chose et ceux qui ne l'ont pas fait. Ces derniers vous diront qu'il y a 142 manières de faire les choses et qu'il n'y a pas de consensus sur la meilleure. Les premiers vous diront simplement que 141 d'entre elles ne fonctionnent pas. »

Il tire aussi sur les ISO-philes, pour des raisons similaires. Les

protocoles Internet ne sont adoptés qu'après avoir été utilisés pendant un certain temps, ce qui évite de standardiser des choses qui ne seront jamais implémentables de manière raisonnable. Les adhérents à l'ISO, d'un autre côté, semblent tenter de standardiser tout ce qui est possible, y compris les « évasions » du standard, avant qu'une implémentation raisonnable de référence existe. En conséquence, on trouve immortalisées des idées dépassées, telles que le paquet de champs de données de niveau infra-octet, ce qui rend difficile de réaliser de bonnes performances lorsque votre ordinateur est alimenté par une lance d'incendie à plus de 10 Gbit/s :-)

Juste mes 0,02 \$

D

**De: torvalds@klaava.Helsinki.FI (Linus Benedict Torvalds)**  
**Sujet: Excuses (était Re : LINUX est dépassé)**  
**Date: 30 Jan 92 15:38:16 GMT**  
**Organisation: University of Helsinki**

Dans l'article <1992Jan29.231426.20469@klaava.Helsinki.FI>  
j'ai écrit :  
>Bien, avec un sujet tel que celui-ci, j'ai bien peur de devoir  
>répondre.

Et j'ai répondu, je me suis laissé aller, sans une pensée pour le bon goût et la netiquette. Pardon à ast, et merci à John Nall pour sa lettre « ce n'est pas comme ça que l'on fait ». Je me suis emporté, et je vais composer maintenant une lettre personnelle (beaucoup moins acerbe) à ast. J'espère que personne ne se détournera de linux parce qu'il est (a) peut-être dépassé (je pense toujours que ce n'est pas le cas, bien que certaines critiques soient valables) et (b) qu'il a été écrit par une tête brûlée :-)

Linus « mon premier, et j'espère mon dernier festival  
incendiaire » Torvalds

**De: pmacdona@sanjuan (Peter MacDonald)**  
**Sujet: Re: Linux est dépassé**  
**Date: 1 Fév 92 02:10:06 GMT**  
**Organisation: University of Victoria, Victoria, BC, CANADA**

Depuis, je pense, que j'ai posté l'un des messages précédents dans toute cette discussion sur Linux contre Minix, je me sens obligé de commenter mes raisons de basculer de Minix vers Linux. Par ordre d'importance, elles sont :

- 1) Linux est gratuit ;
- 2) Linux évolue de manière satisfaisante (car les nouvelles fonctionnalités sont acceptées dans la distribution par Linus).

La première nécessite quelques explications, puisque si j'ai déjà acheté Minix, en quoi le prix peut-il me concerner ? Simple. Si le système d'exploitation est gratuit, beaucoup plus de gens vont l'utiliser/le maintenir/l'améliorer. C'est aussi le raisonnement que j'ai tenu lorsque j'ai acheté mon 386 au lieu d'une SPARC (que j'aurais pu avoir pour seulement 30% de plus). Depuis que les PC sont bon marché et partout disponibles, de plus en plus de gens les achètent/ les utilisent et ainsi, le logiciel bon marché/gratuit est

abondant.

La seconde doit être assez évidente pour quiconque a utilisé Minix durant un certain temps. AST n'accepte généralement pas d'améliorations à Minix. Ce n'est pas destiné à être un challenge, mais seulement un fait. AST a de bonnes et légitimes raisons pour cela, et je ne les discute pas. Mais Minix a quelques limitations avec lesquelles je ne peux plus continuer à vivre, et à cause de cette position, l'hypothèse de les voir résolues dans un temps raisonnable n'est pas satisfaisante. Parmi ces limitations, il y a :

- pas de gestion des instructions 386 ;
- pas de console virtuelle ;
- pas de liens symboliques ;
- pas d'appel select ;
- pas de pseudo-terminal ;
- pas de pagination à la demande/pagination disque/  
texte partagé/bibliothèque partagée (mm efficace)...  
chmem (mm inflexible) ;
- pas de X-Window (pour les mêmes raisons que celles invoquées  
pour Linux et les 386) ;
- pas de TCP/IP ;
- pas d'intégration GNU/SysV (portabilité) ;
- ...

Certaines de ces limitations peuvent être levées par des correctifs (et si vous l'avez réalisé vous-même, je n'ai pas besoin de vous dire la satisfaction que cela procure), mais au moins les cinq derniers points étaient/sont hors de toute attente raisonnable.

Finalement, mon commentaire (ma saillie ?) sur l'architecture en noyau segmenté, ou micro-noyau de Minix était plus une expression de ma frustration/désorientation dans ma tentative d'utilisation des correctifs de pseudo-terminaux de Minix comme un guide sur la manière de le réaliser sous Linux. Cette fonction particulière était une de celles pour lesquelles le passage de messages en rendait l'implémentation vraiment complexe.

J'ai une opinion sur le sujet Monolithique contre passage de messages, mais je ne vais pas l'exprimer maintenant, et je n'ai pas songé à le faire jusqu'à présent. Mes buts sont totalement à court terme (fonctionnalités maximales et temps/coût/effort minimaux), c'est pourquoi mes vues ne sont pas appropriées, et ne doivent pas être mal interprétées. Si le manque des fonctionnalités ci-dessus ne vous gêne pas, alors vous devez prendre Minix en considération (si cela ne vous gêne pas de payer bien sûr :).

**De:** olaf@oski.toppoint.de (Olaf Schlueter)  
**Sujet:** Re : Linux est dépassé  
**Date:** 7 Fév 92 11:41:44 GMT  
**Organisation:** Toppoint Mailbox e.V.

Juste quelques commentaires sur la discussion Linux contre Minix, qui devient en partie une discussion monolithique contre micro-noyau.

Je pense qu'il n'y aura pas d'accords entre les tenants de chacun de ces concepts, s'ils oublient que Linux et Minix ont été conçus pour des utilisations différentes. Si vous voulez un système Unix abordable, puissant et évolutif sur une machine unique, avec la possibilité d'adapter des logiciels Unix sans difficulté, alors Linux est fait pour vous. Si vous vous intéressez aux concepts des systèmes

d'exploitation modernes, et que vous voulez apprendre comment fonctionne un micro-noyau, alors Minix est le meilleur choix.

Ce n'est pas un argument contre le système micro-noyau, que pour l'instant les implémentations monolithiques d'Unix sur PC aient de meilleures performances. Cela veut simplement dire qu'Unix est peut-être mieux implémenté en monolithique, au moins tant qu'il s'exécute sur une seule machine. Du point de vue des utilisateurs, la conception interne du système n'a aucune importance. Jusqu'à ce qu'il soit en réseau. Selon l'approche monolithique, un serveur de fichiers deviendra un processus utilisateur qui s'appuie sur un matériel quelconque comme Ethernet. Les programmes qui souhaitent utiliser ce support devront utiliser des bibliothèques spécifiques qui offrent les appels de communication avec ce serveur. Dans un système micro-noyau il est possible d'incorporer tout le serveur dans le système sans qu'il soit besoin de « nouveaux » appels système. Du point de vue de l'utilisateur c'est un avantage, car rien ne change, il bénéficie juste de meilleures performances (en termes d'augmentation d'espace disque, par exemple). Du point de vue de l'implémenteur, le micro-noyau est plus rapidement adaptable à des modifications de la conception matérielle.

Il a été critiqué qu'AST rejette toute amélioration à Minix. Puisque son intérêt est dans la valeur éducative de Minix, je comprends cet argument de vouloir conserver la simplicité du code, et de ne pas vouloir le surcharger de fonctionnalités. En tant qu'outil éducatif, Minix est écrit comme un système micro-noyau, bien qu'il fonctionne sur des plates-formes qui se comporteraient probablement mieux avec un système monolithique. Mais la zone d'influence des applications réseau croît et des systèmes modernes comme Amoeba ou Plan 9 ne peuvent être écrits sous forme monolithique. Ainsi Minix a-t-il été écrit dans l'intention de donner aux étudiants un exemple pratique d'un système micro-noyau, pour qu'ils puissent jouer avec les tâches et les messages. Ce n'était pas l'idée de fournir à de nombreuses personnes un système économique et puissant pour le dixième du prix d'implémentations SYSV ou BSD.

Résumé : Linux n'est pas meilleur que Minix, ni le contraire. Ils sont différents pour de bonnes raisons.

**De:** meggin@epas.utoronto.ca (David Megginson)

**Sujet:** Mach/Minix/Linux/Gnu etc.

**Date:** 1 Fév 92 17:11:03 GMT

**Organisation:** University of Toronto - EPAS

Bien, c'était une chouette discussion. Je suis totalement convaincu par le professeur Tanenbaum qu'un micro-noyau est la direction à prendre, mais plus je regarde dans le source de Minix, moins je crois que ce soit un micro-noyau. Je ne m'occuperai sans doute pas du portage de Linux sur le M68000, mais je souhaite plus de services que Minix puisse offrir.

Quid d'un micro-noyau compatible MACH au niveau message/appel système ? Il n'a pas en fait besoin de faire tout ce que fait MACH, comme la pagination virtuelle de mémoire, il doit juste avoir l'air de MACH depuis l'extérieur, pour tromper des programmes comme le futur émulateur Unix Gnu, BSD, etc.

Cela prolongerait un peu la vie utile de nos machines sur M68000 ou sur 286. Dans l'intervalle, je vais probablement rester sous Minix avec mon ST plutôt que repasser à MiNT après tout, Minix au moins

ressemble à Unix, alors que MiNT ressemble au TOS en essayant de ressembler à Unix (il est bien obligé d'être compatible TOS).  
David

**De:** peter@ferranti.com (peter da silva)  
**Groupe de discussion:** comp.os.minix  
**Sujet:** Que rapporte cette guerre? (Re : LINUX est dépassé)  
**Date:** 3 Fév 92 16:37:24 GMT  
**Organisation:** Xenix Support, FICC

Allez-vous arrêter de vous descendre en flamme les uns les autres ?  
Je veux dire par là que linux est conçu pour fournir un environnement ayant d'assez bonnes performances sur un matériel handicapé par des années passées à traîner le boulet de la compatibilité ascendante. Minix est conçu comme outil d'enseignement. Il n'est pas bon de faire le travail d'un autre, et pourquoi le ferait-il ? Le fait que Minix s'essouffle rapidement (et c'est vrai) n'est pas un problème dans son milieu de prédilection. Il est sûrement meilleur que le système d'exploitation JOUET. Le fait que Linux ne soit pas transposable sur d'autres plates-formes que le 386/AT ne pose pas problème car on en trouve des millions ici-bas (et vraiment pas chers : vous pouvez trouver un 386/SX en dessous de 1000 dollars).

Un noyau monolithique est assez facile à fabriquer, ce qui vaut le coup si cela permet au système de sortir rapidement. Considérez-le comme une économie de travail pour le temps du programmeur.

L'API est portable.

Vous pouvez remplacer le noyau par un autre ayant la conception micro-noyau (et en matière de conception de micro-noyau MINIX ne représente pas le top du top, même pour des PC d'entrée de gamme... voyez l'AmigaOS) sans perturber les applications.  
C'est tout l'intérêt d'avoir une API portable au départ.

Les micro-noyaux sont certainement ce qu'il y a de mieux pour beaucoup de choses. Cela demande plus de travail pour en améliorer l'efficacité, et dès lors une conception plus simple, ne tirant réellement pas partie des avantages du micro-noyau est préférable, si on veut le faire pour des raisons pédagogiques. Pensez que c'est une économie de temps pour les étudiants. Le concept est toujours bon et lorsqu'il est possible d'avoir une interface de programmation d'applications pour le micro-noyau on peut obtenir une amélioration TRÈS impressionnante (des milliers de commutations par seconde avec un 68000 8 MHz).

**De:** ast@cs.vu.nl (Andy Tanenbaum)  
**Sujet:** Campeurs mécontents  
**Date:** 3 Fév 92 22:46:40 GMT  
**Organisation:** Fac. Wiskunde & Informatica, Vrije Universiteit, Amsterdam

J'ai récemment reçu quelques courriers de campeurs mécontents (en fait 10 messages venant des 43000 lecteurs pourraient paraître beaucoup, mais ça ne l'est pas réellement).  
Il semble en ressortir trois points :

1. Les noyaux monolithiques sont tout aussi bien que les micro-noyaux
2. La portabilité n'est pas si importante
3. Les logiciels devraient être gratuits

Si quelqu'un veut avoir une discussion sérieuse entre micro-noyaux et

noyaux monolithiques, très bien. Nous pouvons le faire sur comp.os.research. Mais par pitié ne venez pas faire de bruit si vous n'avez aucune idée de ce dont vous parlez. J'ai aidé à concevoir et implémenter 3 systèmes d'exploitation, un monolithique et deux micro, et étudié beaucoup d'autres en détail. Beaucoup des arguments proposés ne valent rien (par exemple, les micro-noyaux ne sont pas bons car on ne peut pas faire de pagination au niveau utilisateur - alors que Mach FAIT de la pagination au niveau utilisateur).

Si vous ne connaissez pas grand-chose à propos de la discussion micro-noyaux contre noyaux monolithiques, il existe quelques informations utiles dans un article que j'ai écrit en collaboration avec Fred Ouglis, Frans Kaashoek et John Ousterhout dans le numéro de Déc. 1991 de COMPUTING SYSTEMS, le journal USENIX). Si vous n'avez pas ce journal, vous pouvez FTPer l'article depuis ftp.cs.vu.nl (192.31.231.42) dans le répertoire amoeba/papers sous comp\_sys.tex.Z (source TeX compacté) ou comp\_sys.ps.Z (PostScript compacté). L'article donne des mesures de performances actuelles et appuie la conclusion de Rick Rashid comme quoi les systèmes à micro-noyaux sont tout aussi efficaces que les noyaux monolithiques.

À propos de la portabilité, il n'y a plus de discussion possible. UNIX a été porté sur tout depuis les PCs jusqu'aux Crays. Écrire un SE portable n'est pas beaucoup plus dur qu'un non portable, et tous les systèmes doivent être écrits en gardant à l'esprit la portabilité de nos jours. Le professeur de SE de Linus l'a sûrement signalé. Faire un SE portable n'est pas quelque chose que j'ai inventé en 1987.

Alors que l'on peut rationnellement parler de la conception du noyau et de la portabilité, la question de la gratuité est 100% émotionnelle. Vous ne pourrez pas croire à quel point j'ai été dernièrement [juron supprimé] au sujet de la non gratuité de MINIX. MINIX coûte 169\$, mais la licence permet de faire deux copies de sauvegarde, le prix effectif peut donc être de moins de 60\$. De plus, les professeurs peuvent en faire des copies ILLIMITÉES pour leurs étudiants. Coherent coûte 99\$. FSF facture plus de 100\$ pour les bandes sur lesquelles sont fournies son logiciel « gratuit » si vous n'avez pas d'accès à l'Internet, et je n'ai jamais entendu personne se plaindre. 4.4 BSD coûte 800\$. Je ne pense vraiment pas que l'argent soit la question. D'ailleurs, il est probable que la plupart des lecteurs de ce forum l'ont déjà.

Une opinion, que je pense tout le monde ne partage pas, est que de rendre quelque chose disponible par FTP n'est pas nécessairement le moyen de fournir la plus large distribution. L'Internet est toujours un groupe hautement élitiste. La plupart des utilisateurs d'ordinateurs n'y sont PAS. J'ai compris d'après PH que le pays où MINIX est le plus utilisé est l'Allemagne, et non les USA, principalement parce qu'un des magazines informatiques (commercial) allemand l'a activement appuyé. MINIX est aussi très utilisé en Europe de l'Est, au Japon, Israël, Amérique du Sud, etc. La plupart de ces gens ne l'auraient jamais obtenu s'il n'y avait eu une entreprise pour le vendre.

Pour revenir à ce que « libre » veut dire, qu'en est-il du code source libre ? Coherent est binaire seulement, mais MINIX a le code source, exactement comme LINUX. Vous pouvez le changer comme vous le désirez, et poster ici les changements. Tout le monde l'a fait pendant 5 ans sans problèmes. J'ai aussi donné des mises à jours gratuites pendant des années.

Je pense que la question réelle est quelque chose d'autre. Je me suis vu offrir de façon répétée mémoire virtuelle, pagination, liens symboliques, systèmes de fenêtrage, et toutes sortes de fonctionnalités. J'ai en général refusé car j'essaie toujours de garder le système assez simple pour qu'il soit compris par des étudiants. Vous pouvez mettre toutes ces choses dans votre version, mais je ne les mettrais pas dans la mienne. Je pense que c'est ce point qui ennuie les gens qui disent que « MINIX n'est pas libre », pas les 60\$.

Une question intéressante est de savoir si Linus est prêt à laisser LINUX devenir « libre » de son contrôle. Quelqu'un peut-il le modifier (le ruiner ?) et le vendre ? Souvenez-vous des centaines de messages avec le sujet « Re : Votre logiciel vendu pour de l'argent » lorsqu'il a été découvert que le centre MINIX en Angleterre vendait des disquettes avec des articles de news, plus ou moins au prix coûtant ?

Supposez que Fred van Kempen revienne et veuille prendre le relais, en créant un LINUX Fred et un LINUX Linus, tous deux utiles mais différents. Est-ce ok ? La question se pose lorsqu'un groupe assez grand de personnes veut faire évoluer LINUX d'une manière que Linus ne veut pas. Avant que ça n'arrive, la question reste néanmoins sur le tapis.

Si vous préférez la philosophie de Linus à la mienne, bien sûr, suivez-le, mais par pitié ne prétendez pas que vous faites ça parce que Linux est « libre ». Dites simplement que vous voulez un système avec un tas de fonctionnalités. Très bien. C'est votre choix. Je n'ai pas d'argument contre ça. Dites juste la vérité.

En aparté, pour ces gens qui ne lisent pas les en-têtes des news, Linus est en Finlande et je suis aux Pays-Bas. Atteignons-nous une situation où une autre industrie critique, le logiciel libre, qui a été totalement dominée par les USA est en train de céder la place à la concurrence étrangère ? Verrons-nous bientôt arriver le Président Bush en Europe avec Richard Stallman et Rick Rashid à la remorque, demandant que l'Europe importe plus de logiciels libres Américains ?

Andy Tanenbaum ([ast@cs.vu.nl](mailto:ast@cs.vu.nl))

**De:** [ast@cs.vu.nl](mailto:ast@cs.vu.nl) (Andy Tanenbaum)

**Sujet:** Re : Campeurs mécontents

**Date:** 5 Fév 92 23:23:26 GMT

**Organisation:** Fac. Wiskunde & Informatica, Vrije Universiteit, Amsterdam

Dans l'article <205@fishpond.uucp> fnf@fishpond.uucp (Fred Fish) écrit :

>Si PH ne s'était pas octroyé le monopole de la distribution, il  
>aurait été possible pour tous les bitouilleurs minix intéressés  
>d'organiser et de mettre en place un groupe qui se serait consacré  
>à produire un minix amélioré. Le but de ce groupe aurait été de  
>produire une unique version vivante de minix avec toutes les  
>améliorations couramment demandées. Cela aurait permis à minix  
>d'évoluer un peu de la même manière que gcc a évolué durant ces  
>dernières années.

Ceci EST possible. Si un groupe de personnes veut le faire, c'est très bien. Je pense que coordonner 1000 prima donnas vivant tout autour du monde sera aussi facile que de mener des chats en troupeau, mais il

n'y a pas de problème légal. Lorsqu'une nouvelle version est prête, il suffit de faire un fichier diff par rapport à la 1.5 et le poster ou le rendre disponible par FTP. Bien que cela requiert quelque travail de la part des utilisateurs pour l'installer, ce n'est pas tant de travail. D'ailleurs, j'ai des scripts shell pour créer les diffs et les installer. C'est ce que Fred van Kempen faisait. Ce qu'il a fait de mauvais est d'insister sur le droit de publier la nouvelle version, plutôt que les diffs par rapport à la base de PH. Ceci mettait PH hors course, ce qui, rien d'étonnant, ne les emballait pas. Si des personnes veulent toujours le faire, qu'elles le fassent.

Bien sûr, je ne vais pas nécessairement appliquer tous ces changements à ma version, il y a donc du travail pour garder la version officielle et celles qui sont améliorées en synchro, mais je suis disposé à coopérer pour minimiser le travail. Je l'ai fait pendant longtemps avec Bruce Evans et Frans Meulenbroeks.

Si Linus veut garder le contrôle de la version officielle, et qu'un groupe d'acharnés veulent continuer dans une direction différente, le même problème arrive. Je ne pense pas que le droit de copie soit réellement le problème. Le problème est de coordonner les choses. Des projets comme GNU, MINIX, ou LINUX tiennent debout seulement si une personne en a la charge. Durant les années 1970, lorsque la programmation structurée a été introduite, Harlan Mills signala que l'équipe de programmation devait être organisée comme une équipe chirurgicale (un chirurgien et ses assistants), et non comme une équipe de bouchers (donnez à chacun un morceau et laissez-les hacher seuls).

Quiconque disant que l'on peut avoir un grand nombre de personnes très dispersés bitouillant sur une même partie complexe de code et éviter l'anarchie totale n'a jamais dirigé un projet logiciel.

>Où est le groupe important de personnes désirant faire évoluer gcc  
>d'une manière que rms/FSF n'approuve pas ?

Un compilateur n'est pas une chose pour laquelle les gens ont un grand attachement émotionnel. Si le langage à compiler est une donnée (par exemple, une norme ANSI), il n'y a pas de place pour que quiconque invente de nouvelles fonctionnalités. Un système d'exploitation a des occasions illimitées pour que des personnes implémentent leurs fonctionnalités préférées.

Andy Tanenbaum ([ast@cs.vu.nl](mailto:ast@cs.vu.nl))

**De:** torvalds@klaava.Helsinki.FI (Linus Benedict Torvalds)  
**Sujet:** Re : Campeurs mécontents  
**Date:** 6 Fév 92 10:33:31 GMT  
**Organisation:** University of Helsinki

Dans l'article <12746@star.cs.vu.nl> ast@cs.vu.nl (Andy Tanenbaum) écrit :

>

>Si Linus veut garder le contrôle de la version officielle, et qu'un  
>groupe d'acharnés veulent continuer dans une direction différente, le  
>même problème arrive.

C'est la seconde fois que je vois cette « accusation » de la part de ast, qui trouve malin de faire des commentaires sur un noyau qu'il n'a probablement jamais vu. Ou tout au moins il ne m'a jamais questionné, ou bien lu alt.os.unix sur le sujet. Seulement pour que

personne ne prenne ses suppositions comme vérité première, voici ma position sur « la prise de contrôle », en 2 mots (ou trois ?) :

J'ferai rien.

Le seul contrôle que je garde effectivement sur linux est que je le connais mieux que quiconque, et que j'ai fait en sorte que mes changements soient disponibles sur les sites ftp, etc. Ceux-ci sont devenus des versions officielles et je ne m'attends pas à ce que cela change pour quelque temps : non pas parce que j'estime que j'ai un droit moral sur lui, mais je n'ai pas entendu trop de réclamations et il se passera beaucoup de mois avant que je ne puisse trouver des gens ayant le même « feeling » au sujet du noyau. (Bon, peut-être certains y arrivent : tytso a effectué assurément des changements significatifs même jusqu'au 0.10, et d'autres l'ont bitouillé également).

Dans les faits, j'ai lancé quelques ballons d'essai sur une liste de diffusion « noyau-linux » qui prendraient les décisions sur les mises à jour, et je m'attends à ce que je ne puisse pas apporter toutes les fonctionnalités qui /devront/ être ajoutées : SCSI etc, car je ne dispose pas du matériel. Le résultat fut négatif : les gens ne semblent pas pressés de changer à l'heure actuelle. (oui, quelqu'un a pensé que je devrais demander aux alentours des dons pour pouvoir développer le logiciel nécessaire - et si quelqu'un a du matériel intéressant, je serai heureux de l'accepter :)

La seule chose que le copyright interdise (et je pense que cela est éminemment raisonnable) est que d'autres personnes commencent à se faire de l'argent avec et ne fournissent pas les sources, etc. Ce n'est pas une question de logique, mais je le vivrais mal si quelqu'un venait à vendre mon travail pour de l'argent, alors que je l'avais rendu disponible exprès afin que tout le monde puisse s'amuser comme il l'entend. Je pense que la plupart des gens comprennent mon point de vue.

Ceci mis à part, si Fred van Kempen veut faire un super-linux, il est tout-à-fait le bienvenu. Il ne se fera pas beaucoup d'argent avec (hormis les frais de distribution), et je ne pense pas que ce soit une bonne idée de diviser linux, mais je ne voudrais pas arrêter même si le copyright me le permet.

>Je ne pense pas que le droit de copie soit  
>réellement le problème. Le problème est de coordonner les choses. Des  
>projets comme GNU, MINIX, ou LINUX tiennent debout seulement si une  
>personne en a la charge.

Oui, la coordination est un gros problème, et je ne pense pas que je quitterai le poste de « chirurgien chef » pour linux avant longtemps, en partie parce que la plupart des gens comprennent ces problèmes. Mais le copyright /est/ une solution : si on sent que je fais du mauvais boulot, ils peuvent le faire eux-mêmes. Tout comme pour gcc. Le copyright de minix, cependant, signifie que si quelqu'un a le sentiment qu'il peut faire un meilleur minix, soit il doit faire des patches (qui ne sont pas si gros quoi que vous ayez dit à ce sujet) ou démarrer à partir de rien (et être attaqué car vous avez d'autres idéaux).

Les patches ne sont pas très drôles à diffuser : je n'ai pas encore fait de cdiff pour une seule version de linux (j'espère que ça

changera : bientôt les patches seront beaucoup plus petits que le noyau et faire à la fois les patches et une version complète deviendra une bonne idée - notez que j'ai toujours donné aussi la version complète). Faire des patches sur des patches est simplement impraticable, spécialement pour les personnes pouvant effectuer les changements elles-mêmes.

>>Où est le groupe important de personnes désirant faire évoluer gcc  
>>d'une manière que rms/FSF n'approuve pas ?  
>Un compilateur n'est pas une chose pour laquelle les gens ont un grand  
>attachement émotionnel. Si le langage à compiler est une donnée (par  
>exemple, une norme ANSI), il n'y a pas de place pour que quiconque  
>invente de nouvelles fonctionnalités. Un système d'exploitation a des  
>occasions illimitées pour que des personnes implémentent leurs  
>fonctionnalités préférées.

Bien, il y a GNU emacs... Ne nous dites pas que les gens n'ont pas de relation émotionnelle avec leurs éditeurs :)

Linus

**De: dmiller@acg.uucp (David Miller)**  
**Sujet: Linux est dépassé et messages suivants**  
**Date: 3 Fév 92 01:03:46 GMT**  
**Organisation: AppliedComputerGroup**

En tant qu'observateur intéressé dans la conception de systèmes, je ne peux

résister à cette enfilade. Rendez-vous compte que je ne suis pas réellement

expérimenté avec minux[1] ni linux. Je suis avec unix depuis de nombreuses années. Tout d'abord, quelques observations :

Minix fut écrit pour être un outil éducatif pour les classes d'AST, pas un système d'exploitation commercial. Il n'a jamais été un paramètre de conception de le faire fonctionner en code source librement disponible pour les systèmes unix. Je pense qu'il était aussi une présentation de la manière dont les systèmes d'exploitation doivent être conçus, avec un micro-noyau et des processus séparés couvrant autant de fonctionnalités que possible.

Linux fut écrit essentiellement en tant qu'exercice d'apprentissage de la part de Linus comment programmer la famille 386. La conception du système d'exploitation ultime n'était pas un objectif. Fournir une plate-forme utilisable, libre, qui puisse exécuter toutes sortes de logiciels libres partout disponibles fut une considération, et c'est une de celles qui apparaît avoir été remplie au mieux.

La critique de qui que ce soit considérant que l'un de ces systèmes n'est pas ce que \*lui\* aurait voulu qu'il soit est déplacée. Après tout, quiconque possédant un ordinateur capable d'exécuter l'un ou l'autre de ces systèmes est libre de réaliser lui-même ce que Linus et Andrew ont écrit !

Moi, pour ce que je suis, j'applaudis Linus pour son effort considérable dans le développement de Linux et dans sa décision de le rendre libre pour tous. J'applaudis AST pour son effort pour rendre minix accessible. J'ai un gros problème par rapport aux remarques sur l'aspect non-libre de minix. Si vous pouvez trouver le temps d'explorer minix et un système d'ordinateur de base,

150 dollars ne sont pas grand-chose et vous avez en plus un livre pour l'accompagner.

Ensuite, quelques questions pour le professeur :

Minix est-il supposé être un « vrai système d'exploitation » ou un outil éducatif ? En tant qu'outil éducatif, c'est un excellent travail. En tant que vrai système d'exploitation, il présente quelques manques terriblement gênants (pourquoi pas de malloc() ?, juste pour les débutants). Mon impression à la lecture du Livre et des messages qui passent ici est que vous vouliez un outil pour enseigner en classe, et que de nombreux autres souhaitaient jouer avec un système d'exploitation accessible. Ces autres ont tenté de lui greffer suffisamment de fonctions pour en faire un « vrai système d'exploitation », avec un succès moins éclatant.

Pourquoi séparer les fonctions fondamentales du SE, telles que la gestion mémoire, dans des processus utilisateurs ? Comme le savent tous les bons gourous \*nix, la voie du succès est de diviser pour régner, avec pour but de \*simplifier\* le problème en composants gérables, bien définis. Si le découpage des éléments de base du système d'exploitation en processus de l'espace utilisateur complique la fonction en y introduisant des mécanismes additionnels (passage de messages, signaux compliqués), avons-nous atteint l'objectif de simplifier la conception et l'implémentation ?

Je suis d'accord que \*nix a souffert d'un mauvais niveau de fonctionnalités et particulièrement sysVr4. Peut-être les fonctions que les gens veulent pour telle ou telle fonctionnalité ou compatibilité peuvent-elles être offertes par des modules/bibliothèques chargeables dynamiquement ? Le micro-noyau serait toujours un gestionnaire de ressources de bas niveau capable de router les demandes de fonctions au module ou à la bibliothèque approprié. Les modules peuvent être des threads ou des processus utilisateurs (je pense. Bitouilleurs de systèmes, corrigez-moi :- ) ).

Juste la valeur de mes 0,04 dollars. N'hésitez pas à m'envoyer vos réponses ici ou par la messagerie électronique. Je n'ai pas de formation traditionnelle en informatique, et je pose donc réellement ces questions naïvement. Je suppose qu'un tas d'autres sur le Net ont des questions similaires en tête, mais je me suis déjà trompé.

David

**De:** michael@gandalf.informatik.rwth-aachen.de (Michael Haardt)  
**Sujet:** 1.6.17 résumé et pourquoi je pense qu'AST a raison.  
**Date:** 6 Fév 92 20:07:25 GMT  
**Répondre à:** u31b3hs@messua.informatik.rwth-aachen.de (Michael Haardt)  
**Organisation:** Gandalf - a 386-20 machine

Il y a quelques temps, j'ai demandé des détails sur la prochaine version de MINIX (1.6.17).

J'ai obtenu quelques réponses, mais seulement de gens utilisant la 1.6.16. Les informations qui suivent sont officielles et peuvent être erronées, mais elles sont tout ce que j'en sais actuellement. Corrigez-moi si quelque chose est faux.

- Les correctifs 1.6.17 s'appliqueront sur la 1.5 fournie par PH ;

- Les fichiers d'en-tête sont propres ;
- Les deux types de systèmes de fichiers peuvent être utilisés simultanément ;
- La gestion des signaux est réécrite pour POSIX. Le vieux bogue est supprimé ;
- Le compilateur ANSI (disponible auprès de Transmediar, je crois) est fourni avec les binaires du compilateur et de nouvelles bibliothèques ;
- Le protocole réseau d'Amoeba ne semble pas pris en charge ;
- times(2) renvoie une valeur correcte. termios(2) est implémenté, mais c'est surtout une bitouille. Je ne sais pas si « implémenté » veut dire dans le noyau ou l'émulation actuelle ;
- Le nouveau système de fichiers n'est pas documenté. Il y a de nouveaux fsck et mkfs, pas d'infos sur cela ;
- Le compilateur ANSI travaille mieux sur les flottants ;
- L'ordonnanceur est amélioré, mais pas aussi bien écrit que celui de Kai-Uwe Bloem.

J'ai demandé cela pour appuyer sur des faits ma décision de migrer vers MINIX 1.6.17 ou vers Linux après la fin des examens. Eh bien celle que j'ai prise est de migrer vers Linux à la fin du mois et de retirer MINIX de mon disque lorsque Linux fera tourner tous les logiciels dont j'ai besoin qui tournent actuellement sous MINIX 1.5 avec des correctifs lourds. J'imagine que cela pourra prendre deux mois au plus. Voici les principales raisons de ma décision :

- Il n'y a pas de version « en cours » de MINIX qui puisse être utilisée comme base de correctifs, et personne ne connaît la date d'arrivée de la 1.6.17 ;
- La bibliothèque contient un certain nombre de bogues, et d'après ce que j'ai entendu, ils ne sont pas en cours de correction. Il n'y aura pas de nouveau compilateur, et les utilisateurs en 16 bits continuent à devoir utiliser les ACK bogués ;
- 1.6.17 doit se rapprocher de POSIX, mais il manque encore un termios complet ;
- Je doute qu'il y ait encore beaucoup de développements pour les utilisateurs en 16 bits.

Je pense cesser de maintenir la liste des logiciels MINIX dans quelques mois.

Y a-t-il quelqu'un par ici qui veuille le continuer ? Jusqu'à ce que Linux tourne *\*parfaitement\** sur ma machine, toutes les mises à jour d'Origami continueront à fonctionner en MINIX 16 bits. J'annoncerai lorsque la dernière de ces versions apparaîtra.

À mon avis, AST a raison dans sa décision concernant MINIX. J'ai lu la guerre incendiaire et ne peux m'empêcher de dire que j'aime MINIX comme il est, par rapport à l'état actuel de Linux. MINIX a quelques avantages :

- Vous pouvez commencer à jouer avec sans disque dur, et même compiler des programmes. Je l'ai fait il y a quelques années ;
- Il est si petit qu'il n'est pas nécessaire d'en savoir beaucoup pour avoir un petit système qui fonctionne correctement ;
- Il y a le livre. Ok, seulement pour la version 1.3, mais le plus gros est toujours valide ;
- MINIX est un exemple de noyau non monolithique. Appelez-le micro-noyau ou bitouille pour contourner une architecture débile : Il démontre un concept, avec ses pour et ses contre - un concept documenté.

Pour moi, c'est un bon système pour les premiers pas dans UNIX et la programmation système. J'ai appris l'essentiel de ce que je sais sur UNIX avec MINIX, dans tous les domaines, depuis la programmation en C sous UNIX jusqu'à l'administration système (et les trous de sécurité). MINIX a grandi avec moi : mises à jour 1.5.xx, consoles virtuelles, courrier et nouvelles, traitement de texte, compilation croisée, etc. Maintenant, il est trop petit pour moi. Je n'ai plus besoin d'un système pour l'enseignement, je voudrais un UNIX plus complexe et plus complet, et il y en a un : Linux.

À l'époque, v7 était l' « état de l'art ». Il y avait MINIX qui en offrait le plus gros. En un ou deux ans, POSIX est devenu ce qu'on a l'habitude de voir. Espérons qu'il y aura MINIX, offrant le plus gros, avec un nouveau livre, pour ceux qui veulent un petit système pour jouer et expérimenter.

Cessez de vous incendier, MINIX et Linux sont deux systèmes différents avec des buts différents. L'un est un outil d'enseignement (et un bon, je pense), l'autre est un vrai UNIX pour les vrais bitouilleurs.

Michael

**De:** dingbat@diku.dk (Niels Skov Olsen)  
**Sujet:** Re : 1.6.17 résumé et pourquoi je pense qu'AST a raison.  
**Date:** 10 Fév 92 17:33:39 GMT  
**Organisation:** Department of Computer Science, U of Copenhagen

michael@gandalf.informatik.rwth-aachen.de (Michael Haardt) écrit :

>Cessez de vous incendier, MINIX et Linux sont deux systèmes différents  
 >avec des buts différents. L'un est un outil d'enseignement (et un bon,  
 >je pense), l'autre est un vrai UNIX pour les vrais bitouilleurs.

Oyez, oyez ! Et maintenant, les articles sur Linux sont dans alt.os.linux (ou comp.os.misc si vous ne recevez pas la hiérarchie alt.\*) et ceux sur Minix ici.

fin (festival d'incendies noyé :-)

Niels