

Préface

Le but de ce livre est de faire comprendre comment on conçoit un système d'exploitation en illustrant notre propos sur un cas concret dont nous commentons les sources complètement. Le choix s'est porté tout naturellement sur le premier noyau Linux, ce que nous justifions au chapitre 3.

En prérequis, nous supposons que le lecteur connaît la notion de système d'exploitation en tant qu'utilisateur, pour des systèmes d'exploitation tels que MS-DOS, UNIX, MacOS ou Windows (95, 98, 2000, NT ou XP), un langage d'assemblage pour les microprocesseurs Intel 80x86 et qu'il sache programmer en langage C.

On peut distinguer cinq niveaux de rapports avec un système d'exploitation :

- le *niveau utilisateur* : le but principal consiste essentiellement à charger les logiciels que l'on veut utiliser et de manipuler quelque peu les fichiers ; on se sert pour cela de l'interpréteur de commandes (et de ses commandes telles que `copy`, `rename`...) ;
- le *niveau administrateur* : cela consiste à paramétrer le système et à le tenir à jour ; il est indispensable pour les systèmes d'exploitation capables d'accueillir plusieurs utilisateurs ;
- le *niveau écriture de scripts* pour automatiser certaines séquences répétitives de commandes ;
- le *niveau programmation système* : cette programmation se fait pour Linux en langage C en utilisant les *appels système* ;
- le *niveau conception du système*, et plus particulièrement du noyau.

Nous allons nous intéresser ici à la conception du système d'exploitation, en illustrant nos propos par Linux, plus particulièrement par le tout premier noyau 0.01. L'intérêt de choisir Linux est que le code est diffusé.

Ce livre n'a d'autre but que de publier en un seul volume les aspects suivants de la conception d'un système d'exploitation :

- les concepts généraux sous-jacents à l'implémentation d'un système d'exploitation, tels qu'on les trouve dans [TAN-87] dont nous nous inspirons fortement ;
- les concepts d'un système d'exploitation de type UNIX, en suivant le plus possible la norme POSIX ;
- de la documentation sur le microprocesseur Intel 80386 ; celle-ci exigeant un ouvrage de la taille de celui-ci, nous en supposons connue au moins une partie, celle qui concerne le mode dit « réel » ;
- la documentation sur les contrôleurs de périphériques et leur implémentation sur un compatible PC, nécessaire à la programmation d'un système d'exploitation ;
- une présentation des choix faits pour l'implémentation de Linux 0.01, suivie d'extraits de fichiers sources, repérables facilement par l'indication `Linux 0.01` située en marge, puis paraphrasés en français ; ces paraphrases, commençant presque toujours par « autrement dit », ne

sont pas théoriquement indispensables mais sont souvent appréciables ; comme nous l'avons déjà dit, tout le source est commenté, même si pour des raisons logiques il est dispersé tout au long de l'ouvrage.

Chemin faisant, nous montrons ainsi une méthode pour étudier les sources d'autres systèmes d'exploitation.

L'index fait références aux concepts mais aussi à tous les noms apparaissant dans les fichiers source, ce qui permet de se rendre directement au commentaire de la partie qui intéresse le lecteur.

Préface à la seconde édition

Dans cette seconde édition, paraissant dix mois après la première, le corps du texte principal n'a pas changé, à part la correction d'une coquille. En revanche, chaque chapitre se conclut désormais par une section « évolution du noyau » renforcée, prenant en compte la version 2.6.0 de ce dernier. Nous conseillons de lire le livre sans tenir compte de ces sections puis d'y revenir dans un deuxième temps.

Nous expliquons au chapitre 3 pourquoi il est préférable, dans un premier temps, de s'attacher au tout premier noyau. Je pense que pour les *aficionados* du tout dernier noyau en date, ces dernières sections seront utiles.

Remerciements

Je tiens à remercier tout particulièrement Mathieu ROPERT, étudiant de l'I.U.T. de Fontainebleau en 2001–2003, pour sa relecture très attentive du manuscrit.