

Leçon 1 : Types et caractéristiques des systèmes d'exploitation

Définitions et rôle

Un système d'exploitation (SE) est présent au cœur de l'ordinateur coordonnant les tâches essentielles à la bonne marche du matériel. C'est du système d'exploitation que dépend la qualité de la gestion des ressources (processeur, mémoire, périphériques) et la convivialité de l'utilisation d'un ordinateur. Un SE résout les problèmes relatifs à l'exploitation de l'ordinateur en garantissant :

- Une gestion efficace, fiable et économique des ressources physiques de l'ordinateur (notamment les ressources critiques telles que processeur, mémoire...) : il ordonne et contrôle l'allocation des processeurs, des mémoires, des icônes et fenêtres, des périphériques, des réseaux entre les programmes qui les utilisent. Il assiste les programmes utilisateurs. Il protège les utilisateurs dans le cas d'usage partagé.
- Il propose à l'utilisateur une abstraction plus simple et plus agréable que le matériel : une machine virtuelle permettant l'interaction avec les utilisateurs en leur présentant une machine plus simple à exploiter que la machine réelle

Historique SE

Tout système d'exploitation dépend étroitement de l'architecture de l'ordinateur sur lequel il fonctionne

La première génération : Les tubes à vides et les cartes enfichables (1945__1955)

Il n'existait pas de système d'exploitation. Les utilisateurs travaillaient chacun leur tour sur l'ordinateur qui remplissait une salle entière. [Les cartes perforées, des logiciels de base (assembleur, compilateur) et les sous programmes d'e/s étaient les seuls outils utilisés suivant le modèle porte ouverte (open Shop).]

NB : L'inconvénient était le matériel très cher, la perte de temps, le risque d'erreur et de panne.

La deuxième génération : Les transistors et traitement par lot (1955__1965)

Le passage aux transistors rendait les ordinateurs plus fiables. Le traitement par lots (Batch processing) consiste à rassembler un ensemble de travaux et préparer leur exécution. Ceci est possible grâce à l'automatisation du passage d'un programme à un autre. Ainsi devant les coûts d'équipement élevés le temps a été réduit grâce au traitement par lots et au parallélisme (utilisation de plusieurs canaux : un pour le traitement et l'autre pour les e/s).

NB : L'inconvénient est que l'automatisation des travaux implique une nécessité de protection de l'ensemble des travaux contre les risques de perturbation en cas d'erreur. (Le processeur peut changer de programme avant la fin d'un autre programme en cours d'exécution pour satisfaire au temps de réponse)

La troisième génération : Les circuits intégrés, la multiprogrammation et le temps partagé (1965__1980)

Amélioration des coûts et des performances (circuits intégrés).

La monoprogrammation impose à un processeur d'attendre la fin d'une entrée/sortie avant de procéder.

La multiprogrammation consiste à maintenir plusieurs travaux ou programmes en mémoire à la fois. En effet la mémoire est partitionnée. Chaque partition contient une tâche différente. Lorsque la tâche attend la fin d'une e/s, une autre peut utiliser le processeur. L'inconvénient est la difficulté à gérer le partage de ressource (les travaux de longue durée doivent attendre longtemps.)

Le temps partagé est l'extension logique de la multiprogrammation. Elle consiste à allouer les ressources informatiques à plusieurs utilisateurs (e/s, processeur) dans des tranches de temps fixe. ~~[Allouer un quantum de temps et disposer de toutes les ressources dont il a besoin. Ces tranches de temps sont de l'ordre de quelques millisecondes (ms). A l'expiration du quantum de temps, le programme est sauvegardé sur la mémoire secondaire s'il n'est pas complètement exécuté.]~~

Le premier système à temps partagé est CTSS (Compatible Time Sharing System) qui fut suivi après son succès par les travaux communs au MIT, les laboratoires Bell et General Electric pour donner naissance à MULTICS (MULTiplexed Information and Computing Service). Par la suite K. Thompson écrivit une version simplifiée (mono-utilisateur) de MULTICS ; B. Kernighan l'appela avec humour UNICS (Uniplexed Information and Computer Service) ; ce nom allait devenir UNIX1. D. Ritchie se joignit à K. Thompson pour réécrire UNIX en langage C ; ce système d'exploitation a été le plus porté sur toutes sortes de machines.

NB : L'inconvénient ici est le problème de fiabilité, de sécurité et d'intégrité des données

La quatrième génération : Les ordinateurs personnels, systèmes distribués et systèmes temps réel (1980__1990)

Ils sont dus au développement des circuits VLSI (Very Large Scale Integration) contenant des centaines de transistors au cm². Ils ont la même architecture que les mini-ordinateurs mais leur prix est beaucoup moins élevé. Il existe deux systèmes d'exploitation principaux : MS-DOS (Microsoft Inc.) et UNIX. MS-DOS intègre petit à petit des concepts riches d'UNIX et de MULTICS. Dans le milieu des années 80, on voit l'apparition de réseaux d'ordinateurs individuels qui fonctionnent sous des systèmes d'exploitation en réseau ou des systèmes d'exploitation distribués.

Un système distribué permet le partage des ressources entre les ordinateurs. Un utilisateur d'un ordinateur bon marché peut se servir de ressources coûteuses existant sur un autre ordinateur.

Les systèmes d'exploitation temps-réel sont destinés aux dispositifs devant non seulement donner des résultats corrects, mais les donner dans un délai déterminé. Ces systèmes d'exploitation sont souvent utilisés par des ordinateurs reliés à un appareil externe (pilotes automatiques, robots industriels, applications vidéo et audio) pour lequel un retard de réponse de l'ordinateur entraînerait un échec de l'appareil.

La cinquième génération : Les ordinateurs personnels portables et de poche (1990 jusqu'à présent)

Apparition des PIC (Personal Intelligent Communicator de chez Sony) et des PDA (Personal Digital Assistant, comme le Newton de chez Apple), grâce à l'intégration des composants et l'arrivée des systèmes d'exploitation de type « micro-noyau ». Les systèmes d'exploitation de type « micro-noyau » sont modulaires (un module par fonction) ; ils peuvent être réalisés avec plus ou moins de modules et donc adaptables à des très petites machines (PDA et PIC).

Fonctions SE

Les principales fonctions assurées par un SE sont les suivantes :

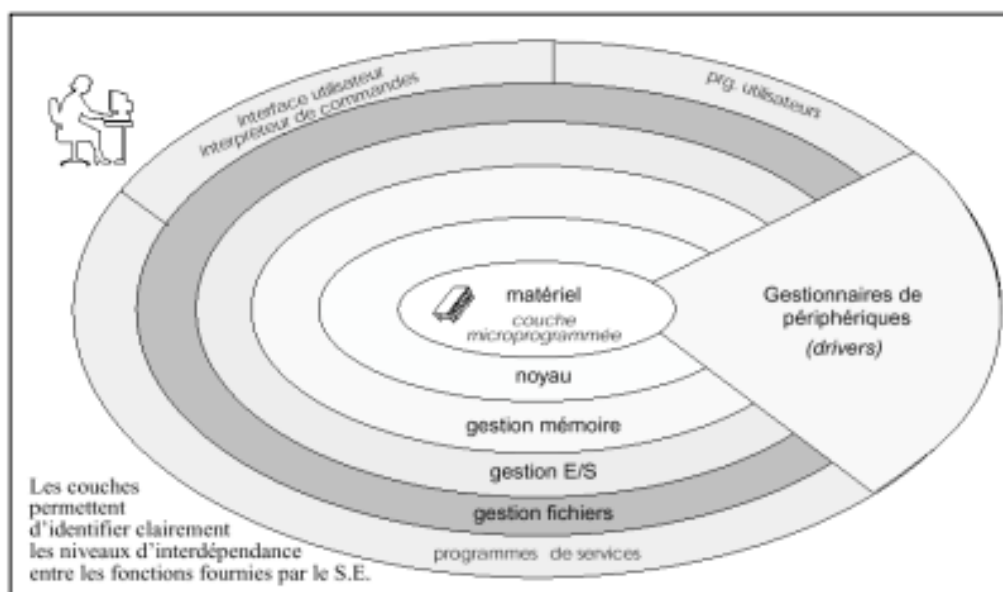
- Gestion de la mémoire principale et des mémoires secondaires,

- Exécution des E/S (périphériques) à faible débit ou haut débit
- Multiprogrammation, temps partagé, parallélisme
- Interruption, ordonnancement, répartition en mémoire, partage des données,
- Lancement des outils du système (compilateurs, environnement utilisateur,)
- Lancement des outils pour l'administrateur du système
- Protection, sécurité ;
- Réseaux

Type de SE

- Mono- tâche ou mono-processus (DOS) : A tout instant, un seul programme est exécuté ; un autre programme ne démarrera, sauf conditions exceptionnelles, que lorsque le premier sera terminé.
- Multi- tâches ou multiprocessus (Windows, Unix, Linux, VMS) : plusieurs processus (i. e. un « programme » en cours d'exécution) peuvent s'exécuter simultanément (systèmes multiprocesseurs) ou en quasi- parallélisme (systèmes à temps partagé)
- Mono- session ou mono-utilisateur (Windows 98,2000) : au plus un utilisateur à la fois sur une machine. Les systèmes réseaux permettent de différencier plusieurs utilisateurs, mais chacun d'eux utilise de manière exclusive la machine (multi- utilisateurs, mono- session)
- Multi- sessions ou multi-utilisateur (Windows XP, Unix, Linux, VMS) : Plusieurs utilisateurs peuvent travailler simultanément sur la même machine

Structures en couches d'un SE moderne



Le noyau

- Réside en mémoire (fréquence élevée des interventions)
- Petite taille
- Gestion du processeur : reposant sur un allocateur (dispatcher) responsable de la répartition du temps processeur entre les différents processus, et un planificateur (scheduler) déterminant les processus à activer, en fonction du contexte.
- Gestion des interruptions : les interruptions sont des signaux envoyés par le matériel, à destination du logiciel, pour signaler un évènement.
- Gestion du multi- tâches : simuler la simultanéité des processus coopératifs (i. e. les processus devant se synchroniser pour échanger des données) et gérer les accès concurrents aux ressources (fichiers, imprimantes, ...)

Le système de gestion de fichiers

Le concept de fichiers est une structure adaptée aux mémoires secondaires et auxiliaires permettant de regrouper des données. Le rôle d'un système d'exploitation est de donner corps au concept de fichiers (les gérer, c'est- à- dire les créer, les détruire, les écrire (modifier) et les lire, en offrant la possibilité de les désigner par des noms symboliques).

Le système de gestion des Entrées/ Sorties

Il s'agit de permettre le dialogue (échange d'informations) avec l'extérieur du système. La tâche est rendue ardue, par la diversité des périphériques d'entrées- sorties et les multiples méthodes de codage des informations (différentes représentations des nombres, des lettres, etc.) Concrètement, la gestion des E/S implique que le SE mette à disposition de l'utilisateur des procédures standard pour l'émission et la réception des données, et qu'il offre des traitements appropriés aux multiples conditions d'erreurs susceptibles de se produire (plus de papier, erreur de disque, débit trop différent, ...)

L'invite des commandes ou shell

Nécessaire pour interagir avec l'utilisateur, il peut être

- Graphique
- Console interpréteur de commandes (langage de commande interprété). Il attend les ordres que l'utilisateur transmet par le biais de l'interface, décode et décompose ces ordres en actions élémentaires, et finalement réalise ces actions en utilisant les services des couches plus profondes du système d'exploitation. Outre l'interaction «directe» (au moyen de terminaux ou de consoles dans le cas d'Unix ou MS DOS), les systèmes offrent le «traitement par lots» (batch). Ce mode de traitement non- interactif est obtenu en regroupant les commandes dans un fichier alors appelé script.

La mémoire virtuelle

La mémoire centrale a toujours été une ressource critique : initialement très coûteuse et peu performante (tapes magnétiques), elle était de très faible capacité. Pour pallier le manque de mémoire centrale, l'idée est venue d'utiliser des mémoires secondaires (de type disque dur), plus lentes, mais de beaucoup plus grandes capacités. La mémoire virtuelle repose sur une corrélation entre la mémoire physique (centrale ou secondaire), présente sur la machine, et l'espace mémoire mis à disposition des programmes par le système d'exploitation (la mémoire virtuelle, ou logique)

Leçon 2 : Coopération et compétition entre processus

Ressources partagées

Ressources critiques

Exécution séquentielle ou parallèle

Blocage et inter blocage

Synchronisation, section critique et outils de synchronisation :

Organisation

Représentation interne

Operations sur les fichiers

Méthode d'accès

Gestion d'espace disque

Leçon 3 : Administration des systèmes informatiques

Configuration d'un système

Évaluation du fonctionnement

Protection, sécurité, classification des systèmes