



**HAL**  
open science

# Émergence des systèmes d'exploitation comme discipline

Claude Kaiser

► **To cite this version:**

Claude Kaiser. Émergence des systèmes d'exploitation comme discipline. Cahiers d'histoire du Cnam, 2017, La recherche sur les systèmes : des pivots dans l'histoire de l'informatique, vol.07 - 08 (2), pp.53-76. hal-03027080

**HAL Id: hal-03027080**

**<https://hal.science/hal-03027080>**

Submitted on 27 Nov 2020

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Émergence des systèmes d'exploitation comme discipline

Claude Kaiser  
Cédric, Cnam.

## Résumé

*Nous présentons ici le contexte dans lequel se sont développés les systèmes d'exploitation des ordinateurs et comment une nouvelle discipline a pu émerger à part entière pendant les décennies 1960-1970. Sont abordés successivement l'évolution très rapide et multiforme de l'informatique, les nombreux projets dans l'industrie en intense collaboration avec les centres de recherche, la variété des systèmes d'exploitation développés, le fort engagement académique avec des revues et des thèses, la reconnaissance de cette discipline en 1971 suivie de la publication de livres d'enseignement. L'article présente aussi l'effort français dans ce domaine et l'introduction de son enseignement au Cnam en 1974.*

**Mots-clés :** systèmes d'exploitation, science informatique, industrie, centres de recherche, recherche et développement.

Ce qui suit est la chronique d'années de gloire pour les systèmes d'exploitation, relatée avec la subjectivité d'un témoin proche<sup>1</sup>. Cette vision de l'émergence de la discipline des systèmes d'exploitation des ordinateurs durant les décennies 1960-1970 s'appuie sur mon expérience professionnelle et sur mes archives.

En 1960 un calculateur digital est volumineux, il remplit plusieurs armoires techniques rangées dans une grande salle de calcul. Il est lent et a peu de mémoire. Par exemple l'IBM 7090 a un cycle de

<sup>1</sup> Les sources utilisées pour rédiger cet article ont été essentiellement mes archives personnelles : bibliographies citées dans des thèses d'État, actes de séminaires et colloques organisés à l'IRIA (Institut de Recherche en Informatique et Automatique, devenu en 1979, et jusqu'à ce jour, l'Institut National de Recherche en Informatique et Automatique - INRIA), livres rédigés en collaboration. Elles furent complétées par quelques recherches sur le Web. La liste des archives originales est donnée en annexe.

base de 2 microsecondes et 128 kilooctets de mémoire centrale. Le système d'exploitation qui, à l'époque, est défini comme « *l'intermédiaire entre l'utilisateur d'un ordinateur et le matériel* » ne peut pas être un gros programme, faute de place en mémoire.

En 1975 un ordinateur – il a changé de nom et s'appelle maintenant un ordinateur – occupe toujours une salle de calcul, mais il a gagné en puissance, il est plus rapide, il a plus de mémoire et plus de périphériques. Par exemple le VAX 11/70 de DEC<sup>2</sup> a un cycle de 200 nanosecondes et une mémoire centrale de 4 mégaoctets. L'ordinateur est devenu très complexe et un système d'exploitation est indispensable. Celui-ci est maintenant un programme de très grande taille, parmi les plus gros et les plus complexes réalisés. Il remplit désormais essentiellement deux fonctions complémentaires : fournir aux utilisateurs simultanés une interface plus commode que celle de la machine physique pour que chacun puisse programmer ou exploiter ses applications ; gérer les ressources de cette machine pour les allouer aux multiples activités qui se les partagent (Krakowiak & Mossière 2013 ; Kaiser 2015a).

Selon le dictionnaire *Le Robert*, le terme « discipline » est utilisé depuis le

---

<sup>2</sup> L'article de Paloque-Berges & Petitgirard, ainsi que l'entretien avec Gérard Florin, un collaborateur proche de C. Kaiser au Cnam, reviennent sur l'importance des machines de DEC dans l'histoire de l'informatique et des systèmes (cf. le premier volume de ce double numéro).

xv<sup>e</sup> siècle pour désigner une branche de la connaissance, une matière enseignée. Une discipline scientifique émerge à la suite de nombreuses réalisations et expérimentations techniques et scientifiques, dans un domaine bien délimité, suivies de réflexions pour dégager les abstractions, les lois ou les concepts communs présents dans ces travaux. Quand cet ensemble commun de connaissances a recueilli l'adhésion des principaux protagonistes, il peut faire l'objet d'un enseignement destiné à transmettre le savoir et à former de nouveaux acteurs.

Appliquées au domaine des systèmes d'exploitation, ces étapes vont guider ma présentation. Après avoir indiqué les premiers grands projets qui ont utilisé des ordinateurs dès 1960, on montre le développement de 1956 à 1978 de systèmes d'exploitation de divers genres, tant dans l'industrie que dans les universités, le fort engagement académique qui l'a accompagné et la reconnaissance formelle de cette nouvelle discipline en 1971. On verra comment cette discipline est née aux États-Unis et que la participation des Européens a surtout été britannique. On se penchera néanmoins sur les travaux faits en France et l'introduction d'un enseignement au Cnam.

Pour cette chronique je me suis restreint à ce que je connais le mieux, écartant des aspects importants comme l'évolution de l'architecture matérielle des ordinateurs, les bouleversements du contexte commercial et la mutation des milieux professionnels et sociétaux.

## Dès 1950, des projets industriels et gouvernementaux de très grande taille utilisent des calculateurs digitaux

Aux États-Unis, dès 1950, de grands projets militaires et civils font appel à ce qu'on appelle encore à l'époque des « calculateurs digitaux » (*digital computers*). Des financements privés ou publics via la DARPA (Defense Advanced Research Project Agency), la NSF (National Science Foundation), soutiennent les recherches nécessaires. Participent à ces projets des sociétés comme Univac, IBM, TRW, Burroughs, Control Data, RCA, General Electric. En Europe apparaissent plus tard des projets analogues. Tous ces projets, dont nous donnons quelques

exemples dans l'encadré 1, s'appuient sur des calculateurs spécifiques et dans chaque projet des logiciels spéciaux sont développés pour piloter ces calculateurs et les matériels connectés. Il n'y a pas encore de système d'exploitation bien individualisé.

## Un contexte industriel évoluant rapidement

### Développements constants de nouveaux calculateurs

Pendant ces décennies 1960 et 1970, on assiste à des progrès technologiques rapides et importants (avec en 1958 les circuits intégrés, en 1963

#### QUELQUES PROJETS AMÉRICAINS

1950	Système SAGE, surveillance aérienne USA, avec matériel IBM.
1950	Naval Tactical Data System (NTDS), avec machines Univac.
1960	Système SABRE, pour la réservation aérienne (matériel IBM).
1965	Electronic Switching System de Bell Telephone, par Western Electric.

#### QUELQUES PROJETS EUROPÉENS

1959	CERN, système de contrôle du PS (Proton Synchrotron).
1963	SENIT pour la marine française de surface, HALIOTIS pour les sous-marins.
1967	EUROCONTROL, contrôle aérien en Europe.
1968-73	SNC Système National de Conduite du dispatching EDF (avec CAE 9040).

Encadré 1 - Projets des années 1960-1970 avec des calculateurs digitaux

le CMOS<sup>3</sup>). Selon la « loi de Moore », le nombre de transistors dans une puce double tous les deux ans. Il en résulte une croissance exponentielle de la performance de l'unité centrale et de la taille des mémoires, une complexification de l'architecture des ordinateurs et un accroissement de leur fiabilité. Cela entraîne une extension explosive du champ de l'informatique, notamment de l'informatique de gestion, et une expansion continue du marché des calculateurs. La complexité croissante des ordinateurs et de leurs applications rend indispensable une aide pour leur exploitation.

### **Un flux régulier de systèmes d'exploitation nouveaux et variés dans l'industrie et la recherche à partir de 1956**

Les industriels américains, en forte concurrence commerciale, ont construit des calculateurs dits universels, c'est-à-dire destinés à de nombreux contextes d'utilisation. Les systèmes d'exploitation, que les Anglo-Saxons ont appelés *operating systems*, doivent s'adapter à cette approche. En 1975 un système d'exploitation doit traiter trois aspects majeurs : l'accès à l'information, la gestion des ressources et la synchronisation des processus<sup>4</sup> concurrents. L'optimisation de

l'architecture pour chacun de ces aspects requiert une structuration particulière des éléments du système d'exploitation. La cohabitation de trois hiérarchies peut entraîner des contradictions et il faut éviter que cela soit préjudiciable aux objectifs ou au fonctionnement du système. Selon qu'on donne plus d'importance à l'un des aspects, on aboutit à un type de système plutôt qu'à un autre. Ainsi à partir de 1956 sont développés des systèmes d'exploitation de conceptions diverses et de plus en plus complexes tant dans le domaine privé que public. Les sociétés commerciales comme IBM, Ferranti, Burroughs, DEC, CII, ou Bull demandent au système d'exploitation d'optimiser le flux des programmes à traiter séquentiellement (*batch processing*) tandis que des universitaires, au MIT, à Berkeley, à Michigan, expérimentent l'accès direct au calculateur par des utilisateurs simultanés (*time-sharing*).

Les premiers systèmes d'exploitation considèrent un seul programme à la fois puis de plus en plus de programmes sont traités simultanément, passant donc de la monoprogrammation à la multiprogrammation. Une très étroite coopération entre les concepteurs de systèmes commerciaux et les chercheurs universitaires existe, et cette particularité américaine engendre des résultats significatifs. L'encadré 2 donne un aperçu chronologique de l'apparition des systèmes d'exploitation avec quelques jalons importants et montre la grande variété des approches. On fait apparaître entre parenthèses les noms des principaux concepteurs connus. Cette modalité sera gardée pour la suite des présentations.

---

<sup>3</sup> Complementary metal oxide semi-conductor.

<sup>4</sup> Un programme est une entité passive, décrivant une suite d'instructions. Un processus est son pendant dynamique, une entité active qui représente l'exécution de cette suite d'instructions par l'ordinateur.

1956	le premier moniteur d'enchaînement : GM/NAA est réalisé pour IBM 704, (Patrick, Mock) suivi en 1959 par IBSYS pour IBM 7090/7094.
1961	le premier système en temps partagé, CTSS est réalisé au MIT sur IBM 7094 (Corbató, Dagget, Daley).
1961	Ferranti, à Manchester en Grande-Bretagne, introduit la pagination câblée pour l'ATLAS (Kilburn, Edwards, Lenigan, Sumner).
1962	le premier système multiprocesseur commercialisé l'est par Burroughs avec une machine et un système Algol, le MCP B5000/5500.
1964	la série 360 d'IBM est annoncée avec un ambitieux système OS/360, avec un moniteur d'enchaînement traitant en parallèle la sortie des résultats d'un programme terminé, l'exécution du programme en cours et l'entrée du programme suivant à exécuter (batch with spooling). Cette série va permettre à IBM de conquérir le marché, grâce à un excellent débit pour le traitement séquentiel des programmes et une politique commerciale efficace. Mais le projet, développé entre 1964 et 1967, prend un retard considérable qui doit être comblé par des systèmes moins ambitieux (comme DOS 360) et il a un coût faramineux (cf. Brooks, 1975). Il y aura plusieurs variantes (PCP, MFT, MVT).
1964	le projet GENIE de temps partagé est développé sur SDS 940 à l'université de Berkeley (Lampson, Thacker, Deutsch).
1965-1970	le système MULTICS est réalisé au MIT, avec des innovations majeures qui vont marquer la discipline (Corbató, Dennis, Denning, Saltzer, Schroeder, et de nombreux autres acteurs ; y collaborent les Français Pouzin et Bensoussan).
1967	le MTS, Michigan Time-sharing System, est construit pour un IBM 360-67 à l'université de Michigan (Arden, Galler, Pinkerton).
1967	IBM développe CP/CMS, générateur de machines virtuelles à Cambridge Massachusetts aux États-Unis et au centre scientifique IBM de Grenoble (y participent les Français Auroux, Bellino, Hans).
1969	aux Bell Labs est créé UNICS, mono-utilisateur présenté comme un clin d'œil à MULTICS multi-utilisateur (Ritchie, Thomson). Il sera rapidement suivi par l'UNIX multi-utilisateur.
1970	TOPS-10, puis TENEX (BBN) sont développés pour DEC PDP 10 (Bobrow, Murphy), qui, en dépit de leur faible diffusion commerciale, ont influencé les méthodes de conception et de développement des systèmes.
1970	en France Honeywell BULL annonce la série 60 avec le niveau 64, puis GCOS/64, GCOS7 transactionnel (Lepicard, Carré, Slosberg, Bellec).
1970	en France CII réalise SIRIS 7 pour le CII 10070, ex SDS Sigma 7, puis en 1972 SIRIS 8 pour l'IRIS 80 (réécriture du système par Ichbiah).
1972	IBM annonce VM/370.
1973	au centre de recherche Xerox PARC, l'ALTO OS (Thacker, MacCreight, Lampson, Soltis, Hoffmann, Metcalfe) révolutionne l'interface utilisateur.
1974	Digital Research commercialise CP/M un système d'exploitation pour Intel 8080, l'un des premiers microprocesseurs (Kildall).
1975	l'université Carnegie Mellon étudie le système HYDRA pour multiprocesseur (Wulf, Levin, Pierson).
1978	le système 38 d'IBM (Soltis, Hoffmann) est commercialisé.

#### Encadré 2 - Jalons importants de 1956 à 1978

## Quelques systèmes d'exploitation particuliers pour les applications en temps réel

À côté des calculateurs et des systèmes universels, on a continué à développer des systèmes particuliers conçus pour commander les procédés industriels qui ont des contraintes spécifiques comme le respect des temps de réponse ou des échéances de calcul. Pour ce faire, ces systèmes privilégient la gestion des processus, se restreignent à une allocation statique des ressources et conservent en mémoire centrale les données et programmes de l'application. Ils sont de plus petite taille que les systèmes généraux et peuvent être embarqués dans le matériel de l'application. L'encadré 3 indique quelques réalisations. Certains systèmes comme le SIGMA 7 de XDS ou le VMS de DEC ont aussi comme objec-

tif les applications scientifiques et sont plus proches des systèmes universels.

## Développements de langages de haut niveau utilisables pour l'écriture de systèmes

Les premiers systèmes d'exploitation étaient simples et écrits en langage d'assemblage, représentation symbolique du langage machine binaire. Puis le système d'exploitation est devenu un programme de très grande taille et très complexe (il doit en particulier gérer l'exécution de programmes concurrents et leur synchronisation, aspect jusqu'alors inconnu et qui est l'objet de recherches). Écrire ce programme en langage d'assemblage est devenu un exercice très difficile et sujet à de nombreuses erreurs. Or durant cette période ont été dévelop-

1961	D-17 Autonetics (guidage Minuteman).
1967	RC 4000 de Regnecentralen au Danemark : multiprogrammation avec des processus communiquant par messages (Brinch Hansen).
1968	DEC PDP 8 mini-ordinateur.
1968	CTL (GB) Modular One calculateur à 16 bits, système multitâche, communication par messages, sémaphores... (écrit en assembleur).
1969	XDS 940 puis SIGMA7 pour le temps réel industriel et le calcul scientifique.
1970	Hewlett Packard HP3000.
1972	PLESSEY(GB) System 250 ( <i>capabilities</i> - England).
1972	RTOS Data General.
1972	DEC RT-11 pour PDP11.
1976	RTES pour Solar de Télémécanique (Grenoble).
1977	DEC VAX VMS pour le temps réel industriel et le calcul scientifique.
1981	Wind River VxWorks (utilisé par la NASA dans le robot envoyé sur Mars).

Encadré 3 - Quelques systèmes conçus pour le temps réel

pés des langages informatiques dits de haut niveau, c'est-à-dire plus proches du langage naturel : Algol (1960), BCPL (1960), PL1 (1963), PL360 (1968), Pascal (1970), MESA (1970), C (1972). Les systèmes sont dès lors programmés en langages de haut niveau.

systèmes d'exploitation. Les plus importants, ceux qui sont indispensables tant pour le partage de l'ordinateur que pour la protection entre des utilisateurs multiples, concurrents et concomitants, sont indiqués dans l'encadré 4.

### **Dispositifs spéciaux pour les systèmes d'exploitation**

Le développement de systèmes d'exploitation de plus en plus performants et fiables devient une nécessité incontournable et un facteur de vente. L'architecture des nouveaux calculateurs s'accompagne de dispositifs câblés spécifiques devenus nécessaires pour les

### **Un riche bilan de systèmes de conception et d'architecture différentes**

Parmi les systèmes d'exploitation de conception et de réalisations différentes, réalisés de 1956 à 1978 et cités dans les encadrés 2 et 3, certains ont eu une influence majeure pour la conception des systèmes ultérieurs et pour la définition de la discipline. Sont remar-

Horloge temps réel, programmable, différente du séquenceur de l'unité centrale (UC), pour mesurer le temps écoulé.

Mécanisme d'interruption (1956) et de commutation de contexte, complété par le déroutement, l'appel système, pour intervenir de l'extérieur sur le déroulement d'un programme.

Jeu d'instruction permettant la réentrance des programmes et le partage de leur code

Protection mémoire permettant de traiter séparément le code et les données.

Mode privilégié pour l'utilisation de l'UC par le Système d'exploitation (SE).

Canaux d'entrée-sortie autonomes avec accès direct à la mémoire (DMA).

Disques durs rapides (1956) pour le va-et-vient des programmes avec la mémoire centrale.

Pagination de la mémoire (1962), avec le traitement du défaut de page et la reprise possible et correcte d'une instruction dont l'exécution a été interrompue.

Aide à la gestion de la mémoire virtuelle, avec translation de mémoire ou segmentation, ou utilisation d'une mémoire associative.

Pagination et mémoire virtuelle permettent la cohabitation concomitante de nombreux programmes en mémoire centrale.

**Encadré 4 - Dispositifs câblés inventés pour les systèmes d'exploitation**



quables pour leur originalité le Burroughs MCP B5000/5500<sup>5</sup>, première machine et premier système dont l'architecture est basée sur la sémantique d'un langage, ici Algol, le CP/CMS, première machine virtuelle par IBM, le RC 4000 de Regne-centralen avec une communication par messages entre processus, l'UNIX qui se distingue par la création de processus par copie du contexte du processus créateur, la communication par messages, et l'enchaînement (*pipes*) de processus ou encore l'ALTO qui révolutionne l'interface homme-machine. Le succès commercial de l'OS/360 avec MFT et MVT va permettre à IBM de gagner environ 90 % du marché de l'informatique de gestion et d'en être le leader incontesté tandis que VAX/VMS sera le modèle emblématique de Digital pour conquérir les marchés industriels et scientifiques. TENEX apparaît comme un modèle d'ingénierie et influencera beaucoup VAX/VMS. Enfin certains systèmes (CTSS, GENIE, MULTICS, CP/CMS) sont le résultat de projets de recherche qui comme on le verra dans le chapitre suivant sont accompagnés d'un fort engagement académique.

## **Le contexte académique avec forte activité de recherche et développement scientifiques**

C'est vers 1965 que la thématique des systèmes d'exploitation commence à être intégrée pour elle-même dans les programmes des centres de recherche tant publics que privés. Des journaux scientifiques, des colloques, des écoles de recherche sont consacrés aux systèmes d'exploitation. L'application d'une démarche scientifique fondée sur l'abstraction et la virtualisation, accompagnée par l'expérimentation de nouveaux concepts, permet de dégager des résultats généraux et un fondement commun de connaissances. La diffusion de ces résultats dans la communauté scientifique les rend accessibles à tous, permet la vérification de leur bien-fondé et leur confère une portée générale. Des thèses analysent les problèmes rencontrés et dégagent les aspects majeurs de cette discipline naissante.

### **Une voie originale : « Capability Based Computer Systems »**

En parallèle avec les voies commerciales et académiques qui vont mener à la définition de la discipline, il faut citer une approche originale centrée sur la protection de l'accès aux composants du système d'exploitation. Le système est structuré en éléments coopérants dont l'accès par un autre élément n'est possible que par une clé (*capability*) qui figure dans de la mémoire protégée et qui

---

<sup>5</sup> On lira avec profit les ouvrages d'E. I. Organick, sur Burroughs (1973) et sur MULTICS (1972).

1967	Chicago : Magic Number Machine (Fabry).
1968	Berkeley : CAL-TSS System (Lampson, Sturgis).
1972	Plessey (GB) : System 250 (England).
1973	Cambridge (GB) : CAP Computer (Wilkes, Needham).
1974	Carnegie Mellon University : HYDRA (C.mmp ; Wulf).
1978	IBM : System 38 (puis AS 400).
1981	Intel : IAPX 432.

#### Encadré 5 - Les systèmes d'exploitation à *capabilities*

n'est modifiable qu'en mode spécial. À chaque acteur est associée une liste de clés (*capability list*) qui identifie tous les éléments, matériels ou logiciels, auxquels il peut accéder et pour quelle utilisation.

Regroupés sous le vocable *capability based systems*, apparaissent des notions comme l'adressage uniforme, les objets, les domaines d'exécution et le type d'accès. Les résultats de cette voie de recherche ont été exploités par quelques industriels, comme Plessey, IBM et Intel. Les systèmes réalisés selon cette voie sont présentés dans le livre de Henry Levy, *Capability Based Computer Systems* (1984) qui en donne une description très complète.

#### **Création de centres de recherche industriels travaillant sur les systèmes**

Étant donné l'importance fondamentale du développement de systèmes d'exploitation, des centres de recherche

industriels ont été créés spécialement. Citons par exemple :

- 1960 Centres scientifiques IBM : Yorktown Heights, Cambridge Scientific Center (CSC)
- 1967 Centre scientifique IBM Grenoble (Duby, Hans, Bellino, Auroux)
- 1970 Centre scientifique CII Grenoble (Bolliet, Verjus, Chupin, Balter)
- 1970 PARC, centre Xerox à Palo Alto (Taylor, Lampson, Kay, Thacker)

#### **Prototypes de systèmes en Europe et en France**

Le développement aux États-Unis de la recherche universitaire sur les systèmes d'exploitation a suscité en Europe des réalisations de prototypes.

À de très rares exceptions près, la collaboration exemplaire entre industriels et chercheurs publics aux États-Unis n'a pas eu lieu en France, les industriels français préférant à l'occasion passer des

EN EUROPE	
1968	Eindhoven (Pays-Bas) : THE (Dijkstra, Habermann).
1968	Edinburgh (Grande-Bretagne) : EMAS pour ICL System 4-75 (Whitfield, Shelness, Stephens, Wight).
1971	Manchester (Grande-Bretagne) : MU 5 multiprocesseur (Sumner, Woods, Kilburn, Morris, Rohl).
1973	Cambridge (Grande-Bretagne) : CAP Computer (Wilkes, Needham).
EN FRANCE	
1966	Grenoble : Système en temps partagé 1401/7044 IBM (Auroux, Bellino).
1968	Grenoble : DIAMAG2, Système conversationnel à accès multiple (Bellot, Siret, Verjus).
1967	CERA SupAéro Paris : Système SAM sur CII 10070 (Rossiensky, Tixier).
1968-1972	IRIA Rocquencourt : Système ESOPE sur CII 10070 (Bétourné, Ferrié, Kaiser, Krakowiak, Mossière) <sup>6</sup> .
1969	Paris : Un système de multiprogrammation sur NCR 4130. (Girault).
1972	Rennes : Système SAR (Verjus, Trilling, André, Herman...).
1972-1975	CII Louveciennes : Projet Y (Derville, Chevance, Mansion).
1975	Grenoble : Système GEMAU (Guiboud-Ribaud, Otrage, Briat, Balter, Rousset de Pina) sur IRIS 80.

**Encadré 6 - Quelques prototypes de systèmes d'exploitation en Europe**

contrats avec les universités américaines<sup>7</sup>. L'encadré 6 indique quelques prototypes de systèmes d'exploitation réalisés en Europe pendant cette période.

### **Publications scientifiques, congrès, écoles de recherche**

À partir de 1967, les principales sociétés savantes en informatique ont créé des publications spécialisées ou donné de la place aux systèmes d'exploitation dans les revues déjà existantes.

Aux États-Unis cela concerne l'Association for Computing Machinery (ACM) avec les *Communications of the ACM (CACM)*, *Journal of the ACM (JACM)*, l'Institute of Electrical and Electronics Engineers (IEEE) avec

<sup>6</sup> L'histoire de la réalisation de ce prototype a été publiée dans Bétourné & al., 2004.

<sup>7</sup> Parmi ces rares exceptions, en 1962 la collaboration de la société Mors, devenue ensuite Télémécanique, avec le laboratoire d'automatique de Grenoble et en 1967 et 1970 la collaboration des centres scientifiques IBM et Bull de Grenoble avec l'IMAG (Institut d'informatique et mathématiques appliquées de Grenoble).

les *Transactions on Computers*, puis les *Transactions on Operating Systems*, et IBM qui crée l'*IBM Systems Journal* en 1962. En Grande-Bretagne, la British Computer Society initie *The Computer Journal*.

En France, l'Association française de Calcul et de Traitement de l'Information (AFCALTI) devenue en 1964 l'Association Française d'Informatique et de Recherche opérationnelle (AFIRO), puis en 1970 l'Association Française pour la Cybernétique Économique et Technique (AFCET), voit sa revue *Chiffres* devenir en 1967 *Revue française d'Automatique Informatique et Recherche Opérationnelle (RAIRO)*. Elle

accueille dès 1967 des publications sur les systèmes d'exploitation.

Des colloques spécialisés, des sessions entières de congrès, des écoles de recherche sont désormais dévolus aux systèmes d'exploitation.

## **Thèses sur les systèmes d'exploitation**

Les premières thèses de doctorat (PhD) sur les systèmes d'exploitation sont soutenues aux États-Unis à la suite des prototypes réalisés dans les universités (MIT, Berkeley et Michigan principalement). Quelques thèses parmi les

**CACM en 1967** : Premier SOSP (Symposium on Operating Systems Principles).

**Congrès IFIP** avec forte présence d'articles sur les systèmes dès 1968.

**Grenoble en 1966** : atelier sur « l'utilisation des ordinateurs en temps réel et temps partagé », organisé par Bolliet.

**École d'été CEA-EDF-IRIA en 1969** sur les systèmes d'exploitation (avec Dijkstra, Randell et Whitfield) au Bréau-sans-Nappe (France) (cf. figure 1 ci-après).

**École d'été CEA-EDF-IRIA en 1972** sur les modèles et mesures de systèmes (avec Coffman, Pinkerton et Wescott) au Bréau-sans-Nappe (cf. figure 2 ci-après).

**IRIA Rocquencourt en 1974** : Premier colloque international sur les aspects théoriques et pratiques des systèmes d'exploitation (Arden, Gelenbe, Habermann, Kaiser, Krakowiak, Randell).

**IRIA Rocquencourt en 1974** : Workshop on protection in Operating Systems (Ferrié, Kaiser, Lanciaux, Martin).

**Congrès AFIRO, AFCET** avec forte consonance systèmes dès 1967.

**AFIRO, AFCET** : séminaire Structure des machines, animé par Boucher dès 1964.

**IRIA** : séminaire Structure et programmation des calculateurs de 1971 à 1974.



Figure 1 - École d'été CEA-EDF-IRIA de 1969 sur les systèmes d'exploitation.  
 Conférenciers : E.W.D. Dijkstra, B Randell, H. Whitfield



Figure 2 - École d'été CEA-EDF-IRIA 1972 sur les modèles et mesures de systèmes.  
 Conférenciers : E. Coffman, T.B. Pinkerton, B. Wescott

plus significatives sont indiquées dans l'encadré 8. Elles analysent les principaux problèmes rencontrés et dégagent les concepts majeurs de la discipline. Leurs titres sont expressifs. Elles sont les premiers témoins académiques de l'émergence de la discipline et diffusent les avancées obtenues. Toute l'industrie américaine en profite pour des réalisations futures. Figurent aussi dans cet encadré 8 quelques thèses européennes dans la mouvance des travaux américains.

Les premières thèses d'État en France dans cette discipline portent plutôt sur la conception, la réalisation et l'analyse de systèmes d'exploitation. On y trouve la mise en application des principaux concepts désormais reconnus par la discipline. Elles sont présentées dans les universités de Rennes, Grenoble et Paris.

#### THÈSES DE PHD AUX ÉTATS-UNIS

<b>1964 Saltzer</b>	Traffic control in a multiplexed computer system, MIT.
<b>1965 Scherr</b>	An analysis of time-shared computer systems, MIT.
<b>1967 Lampson</b>	Scheduling and protection on interactive multi-processor systems, Berkeley.
<b>1968 Fabry</b>	List-structured addressing, University of Chicago.
<b>1968 Pinkerton</b>	Program behaviour and control in virtual storage systems, Michigan.
<b>1968 Denning</b>	Resource allocation in multiprocess computer systems, MIT.
<b>1970 Alexander</b>	Time-sharing supervisory programs, Michigan.
<b>1971 Holt</b>	On deadlock in computer systems, Cornell.
<b>1972 Goldberg</b>	Architectural principles for virtual computer systems, Harvard.
<b>1972 Schroeder</b>	Cooperation of mutually suspicious subsystems, MIT.
<b>1974 Redell</b>	Naming and protection in extensible operating systems, Berkeley.
<b>1974 Sturgis</b>	A post mortem for a time-sharing system, Berkeley.

#### THÈSES DE PHD EN EUROPE

<b>1967 Habermann</b>	Harmonious cooperation of abstract machines, THE, Pays-Bas.
<b>1972 Khaja</b>	The implementation of the name store and the associated replacement algorithm in the MU5 computer, Manchester, GB.
<b>1973 Walker</b>	The structure of a well-protected computer, Cambridge GB.

Encadré 8 - Principales thèses de PhD

1973 Verjus	Nature et composition des objets d'un système de programmation. Rennes.
1973 Bellino	Conception et réalisation d'un système à accès multiple. Grenoble.
1973 Hans	Mécanismes pour systèmes générateurs de machine virtuelle. Grenoble.
1973 Kaiser	Conception et réalisation de systèmes à accès multiple : gestion du parallélisme. Paris.
1973 Krakowiak	Conception et réalisation de systèmes à accès multiple : allocation de ressources. Paris.
1973 Gelenbe	Modèles de comportement de systèmes informatiques. Paris.
1974 Anceau	Étude des systèmes hiérarchisés de ressources. Grenoble.
1974 Lenfant	Comportement des programmes dans leur espace d'adresse. Rennes.
1975 Guiboud-Ribaud	Mécanismes d'adressage et de protection dans les systèmes informatiques. Application au noyau Gemau. Grenoble.
1975 Brandwajn	Équivalence et décomposition dans les modèles à files d'attente. Application à l'évaluation des performances des systèmes d'exploitation. Paris.
1975 Ferrié	Contrôle de l'accès aux objets dans les systèmes informatiques. Paris.
1977 Mossière	Méthode pour l'écriture des systèmes d'exploitation. Grenoble.
1977 Leroudier	Systèmes adaptatifs à mémoire virtuelle. Grenoble.
1977 Potier	Modèles à files d'attente et gestion des ressources dans les systèmes. Grenoble.

Encadré 9 - Premières thèses d'État en France sur les systèmes d'exploitation

## Et déjà au Cnam, des mémoires d'ingénieur

Les premiers mémoires d'ingénieur Cnam sur les systèmes d'exploitation sont réalisés dans l'équipe de recherche du Centre scientifique IBM de Grenoble et soutenus au centre régional Cnam associé.

- 1972 Le Heig : Espace virtuel sous CP/CMS, Cnam Grenoble.
- 1974 Dupuy : Working set dispatcher, Cnam Grenoble.

## Définition et création de la discipline en 1971

L'industrie américaine est consciente de l'importance des systèmes d'exploitation. Un bon système est une nécessité et un atout important pour la vente de calculateurs. Il faut donc former de bons ingénieurs pour la conception et la conduite de ces systèmes. Devenu une branche reconnue de l'informatique sur le plan industriel, le monde des systèmes d'exploitation devient une discipline dans les instances

universitaires à la suite du rapport de 1971 de l'Académie nationale américaine d'ingénierie et de la publication de nombreux ouvrages d'enseignement supérieur.

### **An Undergraduate Course on Operating Systems Principles**

L'Académie nationale américaine d'ingénierie a, grâce à un financement de la NSF (National Science Foundation), chargé un comité d'un rapport proposant le programme d'un enseignement supérieur des systèmes d'exploitation. Ce comité, le COSINE Committee<sup>8</sup>, était constitué par d'éminents scientifiques du domaine et son rapport de 37 pages peut être considéré comme le document fondateur de la discipline. Ce comité était composé de six membres :

- Peter J. Denning, Chairman, Princeton University
- Jack B. Dennis, Massachusetts Institute of Technology
- Butler Lampson, Xerox Research Laboratory, Palo Alto
- Nico Haberman, Carnegie-Mellon University
- Richard R. Muntz, University of California, Los Angeles
- Dennis Tsichritzis, University of Toronto

Ils ont commencé par définir de façon plus précise le rôle des systèmes

d'exploitation avant de proposer un enseignement articulé en huit points reprenant les concepts clés dégagés par les travaux académiques et reconnus par tous comme fondamentaux. Ce rapport capital est mis en ligne sur internet et disponible gratuitement (COSINE, 1971). Il situe les rôles du système d'exploitation selon le plan suivant :

1. *The system defines an extended language.*
2. *The system defines an extended machine, e.g., a « virtual machine ».*
3. *The system creates an environment for efficient program execution.*
4. *The system is an information management system.*

Cela définit les fonctions principales du système d'exploitation : étendre le langage machine, fournir à l'utilisateur une machine virtuelle plus commode d'utilisation, créer un environnement d'exécution efficace, gérer l'information du système et des utilisateurs. Le rapport propose en outre un enseignement articulé en huit points :

1. *Introduction*
2. *Procedures*
3. *Processes*
4. *Memory Management*
5. *Name Management*
6. *Protection*
7. *Resource Allocation*
8. *Pragmatic Aspects.*

---

<sup>8</sup> Computer Science in Electrical Engineering Committee.



### AVANT LE COSINE

- 1964 WEGNER. *Introduction to system programming*, Academic Press.  
1968 WILKES. *Time-sharing computer systems*, Nat. Acad. of Engineering.  
1970 ARSAC. *Systèmes de conduite des ordinateurs*, Dunod.  
1970 WATSON. *Time-sharing design concepts*, Mc Graw-Hill.

### APRÈS LA PARUTION DU COSINE

- 1972 HOARE, PERROT. *Operating systems techniques*, Academic Press.  
1973 COFFMAN, DENNING. *Operating systems Theory*, Prentice Hall.  
1973 BRINCH HANSEN. *Operating systems principles*, Prentice Hall.  
1974 MADNICK, DONOVAN. *Operating systems*, Mc Graw Hill.  
1974 TSICHRITZIS, BERNSTEIN. *Operating systems*, Academic Press.  
1974 SHAW. *The logical design of operating systems*, Prentice Hall.  
1975 CROCUS. (nom collectif de Bellino, Bétourné, Briat, Canet, Cleemann, Derniame, Ferrié, Kaiser, Krakowiak, Mossière, Verjus). *Systèmes d'exploitation des ordinateurs. Principes de conception*. Dunod<sup>9</sup>.  
1975 GRAHAM. *Principles of systems programming*, Wiley.  
1975 FREEMAN. *Software systems principles*. A survey, SRA.  
1975 LISTER. *Fundamentals of operating systems*, MacMillan Press.  
1976 HABERMANN. *Introduction to operating system design*, SRA.

#### Encadré 10 - Premiers livres d'enseignement de la discipline

### De nombreux livres d'enseignement

Quelques livres pionniers avaient décrit des réalisations de systèmes. Mais après la publication du rapport du COSINE Committee en 1971 vont paraître désormais des ouvrages présentant les principes des systèmes d'exploitation.

Dans les titres figurent des termes tels que théorie, principes, conception logique, fondamentaux, qui dénotent clairement la volonté d'enseigner une nouvelle discipline et pas seulement un lot de techniques. Ces ouvrages ne se contentent pas d'exposer les principes mais chacun d'eux est accompagné d'exercices qui permettent des mises en pratique de ces principes.

---

<sup>9</sup> L'histoire de l'écriture collective de cet ouvrage est relatée dans CROCUS, 1993.

## 1971 : Une discipline scientifique : principes, concepts, théorie, expérimentations

Dès 1971, dans le COSINE et les ouvrages d'enseignement qui vont suivre, émerge désormais la matière d'un enseignement de la nouvelle discipline. Les principaux chapitres sont consacrés aux points suivants :

- Définition précise des rôles d'un système d'exploitation des ordinateurs.
- Conception hiérarchique ascendante avec architecture en couches au-dessus d'un noyau basique.
- Abstraction et virtualisation du matériel, avec les correspondances entre concret et abstrait :
  - processeur → processus
  - canaux d'entrées-sorties → processus
  - mémoire physique → mémoire virtuelle
  - disques → fichiers, espace virtuel
- Gestion du parallélisme entre des processus s'exécutant en concocomitance et devant se synchroniser entre eux : paradigmes de cette synchronisation (exclusion mutuelle, lecteurs-rédacteurs, producteurs-consommateurs, repas des philosophes) et leur programmation.
- Gestion de l'information : représentation et accès, contrôle de cet accès et du partage de l'information, liaison
  - (passage d'un nom désignant un objet dans un contexte particulier à l'emplacement où l'objet est stocké en mémoire).
- Gestion et optimisation des ressources : ordonnancement des processeurs, partage de la mémoire (pagination à la demande, algorithmes de remplacement de pages en mémoire centrale, va-et-vient entre la mémoire centrale et le disque de pagination), localité du comportement des programmes (analyse de leur utilisation de la mémoire au cours de leur déroulement), allocation d'un espace de travail en mémoire centrale (*Working Set*), choix et réglage du taux de multiprogrammation, optimisation des accès aux disques et aux mémoires secondaires de stockage.
- Permanence et qualité du service : évitement de l'écroulement (à la suite d'une allocation trop optimiste des ressources communes), prévention de l'interblocage (deux processus s'empêchent mutuellement de progresser), protection, tolérance aux pannes, sécurité, performances, mesures, modèles.
- Nécessité d'une démarche de génie logiciel (*Software engineering*) pour obtenir un système fiable, évolutif, adaptable.

## Reconnaissance de la discipline

- *Attribution du prix Turing à des pionniers des systèmes d'exploitation*

Le prix Turing, prix scientifique de haut niveau considéré comme l'équivalent du prix Nobel pour l'informatique, a récompensé certains des acteurs de l'émergence de la discipline des systèmes d'exploitation des ordinateurs.

- 1972 Dijkstra (Algol, science et art des langages de programmation)
- 1980 Hoare (définition et conception des langages de programmation)
- 1983 Thompson, Ritchie (théorie des systèmes génériques et UNIX)
- 1990 Corbató (temps partagé, CTSS et MULTICS)
- 1992 Lampson (systèmes d'exploitation, environnements personnels, sécurité)

- *Classification ACM des systèmes d'exploitation*

L'ACM (Association for Computing Machinery), principale association savante en informatique, a publié en 1964 une première classification pour la *Computer Science*. Dans cette première classification les systèmes d'exploitation n'apparaissent pas sous leur dénomination mais cachés dans une sous-rubrique de la programmation, à savoir « 4. Programming ; 4.3 Supervisory Systems ».

À la suite d'une révision commencée en 1968, la nouvelle classification de 1991 leur attribue désormais une rubrique entière, sous l'intitulé *Operating systems* :

### D.4 OPERATING SYSTEMS (C)

D.4.1 *Process Management*

D.4.2 *Storage Management*

D.4.3 *File Systems Management*

D.4.4 *Communications Management*

D.4.5 *Reliability*

D.4.6 *Security and Protection*

D.4.7 *Organization and Design*

D.4.8 *Performance*

D.4.9 *Systems Programs and Utilities*

## Quel cours système au Cnam en 1974

En 1974 le département de Mathématiques et Informatique du Cnam m'a demandé de créer des cours sur les systèmes d'exploitation pour les cycles B et C hors temps ouvrable, et pour l'Institut d'Informatique d'Entreprise (IIE). Je les ai enseignés en utilisant comme support le livre CROCUS en cours de publication (Crocus, 1975). Les cours portaient sur les concepts des systèmes d'exploitation et les techniques associées et étaient accompagnés de quelques exercices dirigés. Les premiers textes photocopiés de cours comprenaient les points suivants :

- Processus : paradigmes du parallélisme, sémaphores, interblocage.
- Gestion de l'information : nommage, liaison, représentation, mémoire vir-

tuelle, CLICS (version pédagogique du système MULTICS).

- Allocation des ressources : ordonnancement des processeurs, des disques, pagination à la demande, localité, espace de travail, écroulement, taux de multiprogrammation.
- Protection : *Capabilities* et contrôle d'accès aux objets.

Le laboratoire de calcul du Cnam possédait un ordinateur Modular One de la société britannique CTL (Computer Technology Limited) avec un mot de 16 bits, un système d'exploitation multiprocessus (appelé E4) avec une communication entre tâches par message et une synchronisation par sémaphores. C'était un système dans l'esprit du RC 4000, écrit en langage d'assemblage, malheureusement sans documentation du code source (c'était une politique délibérée de CTL). Il n'a donc pas pu être utilisé pour les exercices dirigés<sup>10</sup>.

## Conclusion

On s'est intéressé à l'émergence des systèmes d'exploitation pendant les décennies 1960-1970, en examinant les développements réalisés essentiellement aux États-Unis chez les constructeurs de calculateurs et dans les universités avec, à côté de ce lieu majeur, un fort concours

de la Grande-Bretagne et des Pays-Bas. Après avoir indiqué les premiers grands projets qui ont utilisé des calculateurs dès 1960, on a montré comment le développement exponentiel de calculateurs nouveaux et variés, de 1956 à 1978, a été accompagné par des systèmes d'exploitation de divers genres, tant dans l'industrie que dans les universités. Parmi ces réalisations, certaines relevaient de centres universitaires et ce fut le début d'un fort engagement académique avec des publications et des thèses contribuant au passage vers l'abstraction et les concepts. Quand, vers 1971, la discipline fut jugée assez bien établie, l'adhésion des industriels et des chercheurs s'est concrétisée dans le rapport du COSINE qui installa formellement la discipline et son enseignement. Quelques données françaises, issues principalement de mes archives personnelles, permirent de montrer ce qui se passait sur un mode mineur en France. Enfin, le lecteur aura pu noter l'apparition d'aspects précurseurs, tels les systèmes virtuels, l'interface utilisateur ou les microprocesseurs, qui tous, au-delà de la période décrite, vont participer à l'effervescence continue de la discipline. Celle-ci n'est pas figée !<sup>11</sup>

---

<sup>10</sup> Pour la recherche en systèmes au Cnam à cette époque, on verra l'article de Paloque-Berges & Petitgirard dans le premier volume de ce double numéro.

---

<sup>11</sup> Remerciements : Je tiens à remercier les relecteurs qui m'ont aidé à rendre lisible ce qui était au départ une présentation orale s'appuyant sur des projections. Le lecteur trouvera bon nombre de ces explications sur les concepts de la discipline dans (Krakowiak & Mossière 2013, Kaiser 2015a) ou dans le glossaire ci-après.

## Sources personnelles

### Thèses d'État sur les systèmes d'exploitation

1967 Arie Nicolaas Habermann : « On the harmonious co-operation of abstract machines ». PhD Thesis. Promotor E.W. Dijkstra. Technische Hogeschool Eindhoven. Pays-Bas. 115 pages.

1973 Jean-Pierre Verjus : « Nature et composition des objets d'un système de programmation ». Thèse de doctorat d'État. Jury : M. Métivier (président), L. Bolliet, C. Coatmelec, C. Pair. Université de Rennes. 124 pages.

1973 Jacques Bellino : « Mécanismes de base dans les systèmes superviseurs : Conception et réalisation d'un système à accès multiple ». Thèse de Docteur es-Sciences appliquées Jury : N. Gastinel (président), L. Bolliet, J.-C. Boussard, M. Griffiths, S. Krakowiak. Université de Grenoble (remerciements de l'auteur à Max Peltier, directeur du Centre Scientifique IBM de Grenoble). 156 pages.

1973 Claude Hans : « Contribution à l'architecture de mécanismes élémentaires pour certains systèmes générateurs de machine virtuelle », Thèse de Docteur es-Sciences Jury : J. Kuntzmann (président), N. Gastinel, L. Bolliet, M. Griffiths, L. Nolin. Université de Grenoble (remerciements de l'auteur à Max Peltier, directeur du Centre Scientifique IBM de Grenoble). 123 pages.

1973 Claude Kaiser : « Conception et réalisation de systèmes à accès multiple : gestion du parallélisme ». Thèse de doctorat d'État. Jury : J. Arsac (président), C. Pair, P. Feautrier, H. Boucher. Université de Paris VI. 251 pages.

1973 Sacha Krakowiak : « Conception et réalisation de systèmes à accès multiple : allocation de ressources ». Thèse de doctorat d'État. Jury : J. Arsac (président), C. Pair, P. Feautrier, H. Boucher. Université de Paris VI. 213 pages.

1974 François Anceau : « Contribution à l'étude des systèmes hiérarchisés de ressources dans l'architecture des machines informatiques ». Thèse de Docteur es-Sciences Jury : J. Kuntzmann (président), J. Arsac, L. Bolliet, S. Krakowiak, G. Saucier. Université de Grenoble. 315 pages.

1975 Serge Guiboud-Ribaud : « Mécanismes d'adressage et de protection dans les systèmes informatiques. Application au noyau Gemau ». Thèse de doctorat d'État. Jury : N. Gastinel (président), L. Bolliet, C. Kaiser, S. Krakowiak, F.H. Raymond, J.-P. Verjus. Université de Grenoble. 322 pages.

1975 Alexandre Brandwajn : « Équivalence et décomposition dans les modèles à files d'attente et leur application à l'évaluation des performances des systèmes d'exploitation ». Thèse de doctorat d'État. Jury : J. Arsac (président), C. Bétourné, S. Krakowiak, E. Mourier, M. Nivat. Université de Paris VI. 266 pages.

1975 Jean Ferrié : « Contrôle de l'accès aux objets dans les systèmes informatiques ». Thèse de doctorat d'État. Jury : J. Arsac (président), J.-C. Boussard, P. Feautrier, C. Kaiser, J.-P. Verjus. Université de Paris VI. 134 pages.

1977 Jacques Leroudier : « Systèmes adaptatifs à mémoire virtuelle ». Thèse de doctorat d'État. Jury : N. Gastinel (président), C. Bocksbaum, J.-C. Boussard, P. Denning, S. Krakowiak, J.-P. Verjus. Université de Grenoble. 403 pages.

### Ouvrages concernant les systèmes d'exploitation

1971 Séminaires IRIA. « Structure et programmation des calculateurs ». Éditeurs S. Krakowiak et C. Kaiser. 217 pages.

1972 Séminaires IRIA. « Structure et programmation des calculateurs ». Éditeur C. Kaiser. 275 pages.

1973 Séminaires IRIA. « Structure et programmation des calculateurs ». Éditeur C. Kaiser. 328 pages.

1974 Séminaires IRIA. « Structure et programmation des calculateurs ». Éditeur C. Kaiser. 101 pages.

1974 « Operating Systems. Proceedings of an International Symposium held at Rocquencourt ». *Lecture Notes in Computer Science* 16, Springer Verlag. Edited by E. Gelenbe and C. Kaiser. 310 pages.

1974 « Protection in Operating Systems ». International Workshop. IRIA/LABORIA. Colloques IRIA. Edited by J. Ferrié, C. Kaiser, D. Lanciaux, B. Martin. 250 pages.

1978 J.-S.. Banino, J. Ferrié, C. Kaiser, D. Lanciaux. *Contrôle de l'accès aux objets partagés dans les systèmes informatiques*. Monographie d'informatique de l'AFCEC. Éditions Hommes et Techniques, 104 pages.

## Bibliographie

Bétourné C., Ferrié J., Kaiser C., Krakowiak S. & Mossière J. (2004). « ESOPE : une étape de la recherche française en systèmes d'exploitation (1968-72) ». 7<sup>e</sup> colloque sur l'Histoire de l'Informatique et des Télécommunications. Cesson-Rennes novembre 2004, France. Rennes : Éditions Irisa/Inria-Rennes, pp. 173-198.

COSINE (1971). *An Undergraduate Course on Operating Systems Principles*. Task Force on Operating Systems (VIII) of the National Academy of Engineering (US) Report of the COSINE Committee of the Commission on Education.

CROCUS (1993). « Crocus : une étape dans l'enseignement des systèmes d'exploitation ». Colloque sur l'histoire de l'informatique. Sophia Antipolis, octobre 1993. AFCEC.

CROCUS (collectif de Bellino, Bétourné, Briat, Canet, Cleemann, Derniame, Ferrié, Kaiser, Krakowiak, Mossière, Verjus) (1975). *Systèmes d'exploitation des ordinateurs. Principes de conception*. Paris : Dunod<sup>12</sup>.

Brooks F. (1975). *The Mythical Man-Month*, Addison Wesley.

Levy H. (1984). *Capability Based Computer Systems*. Digital Press.

Organick, E. I. (1973). « Computer System Organisation : The B5700/B6700 series », Academic Press.

Organick, E. I. (1972). « The Multics System: an examination of its structure ». Cambridge, Mass. : MIT Press.

---

<sup>12</sup> Ouvrage disponible gratuitement sous format pdf sur [cnam.cnum.fr](http://cnam.cnum.fr) (grâce à Pierre Cubaud) [URL : <http://cnam.cnum.fr/CGI/redis.cgi?8CA2680>].

## Webographie

Krakowiak S. et Mossière J. (2013). « La naissance des systèmes d'exploitation. » [URL : <https://interstices.info/naissance-systemes>] consulté le 01/08/2017.

Krakowiak S. (2014a). « Les débuts d'une approche scientifique des systèmes d'exploitation. » [URL : [https://interstices.info/jcms/int\\_70839/les-debuts-d-une-approche-scientifique-des-systemes-d-exploitation](https://interstices.info/jcms/int_70839/les-debuts-d-une-approche-scientifique-des-systemes-d-exploitation)] consulté le 01/08/2017.

Krakowiak S. (2014b). « La naissance du génie logiciel. » [URL : [https://interstices.info/jcms/ni\\_79198/la-naissance-du-genie-logiciel](https://interstices.info/jcms/ni_79198/la-naissance-du-genie-logiciel)] consulté le 01/08/2017.

Krakowiak S. (2015) « Quelques aspects de l'histoire des systèmes d'exploitation », Séminaires Histoire de l'informatique et du numérique, au Cnam, conférence de janvier 2015, video [URL : <http://www.musee-informatique-numerique.fr/>] consulté le 01/08/2017.

Kaiser C. (2015a). « À quoi sert un système d'exploitation ? » [URL : [https://interstices.info/jcms/p\\_83884/a-quoi-sert-un-systeme-d-exploitation](https://interstices.info/jcms/p_83884/a-quoi-sert-un-systeme-d-exploitation)] consulté le 01/08/2017.

Kaiser C. (2015b). « Le ballet des processus dans un système d'exploitation. » [URL : [https://interstices.info/jcms/p\\_82112/le-ballet-des-processus-dans-un-systeme-d-exploitation](https://interstices.info/jcms/p_82112/le-ballet-des-processus-dans-un-systeme-d-exploitation)] consulté le 01/08/2017.

## Glossaire

**Code** : un programme comprend du code et des données (synonyme : instructions).

**Désignation** : utilisation de noms pour atteindre des objets du système.

**Écroulement** : effondrement du système suite à de mauvaises allocations de ressource.

**Information** : instruction ou donnée en mémoire.

**Interblocage** : situation où des processus s'empêchent mutuellement de progresser

**Mémoire virtuelle** : espace d'adressage d'un processus, virtualisation de la mémoire physique.

**Multiprocesseur** : calculateur avec plusieurs unités centrales ou processeurs.

**Multiprogrammation** : exécution concomitante de plusieurs processus.

**Nommage** : organisation des noms des objets en mémoire.

**Noyau** : partie centrale du SE, on dit parfois partie enfouie (en anglais kernel).

**Objet** : tout composant logiciel ou matériel, repéré par le système d'exploitation.

**Pagination** : découpage de la mémoire en pages de taille fixe.

**Processeur** : unité centrale, unité active, moteur du calculateur.

**Processus** : déroulement d'un programme séquentiel, abstraction du processeur.

**Ressource** : matériel ou logiciel, nécessaire pour dérouler un processus.

**Sémaphore** : mécanisme de synchronisation, introduit par Dijkstra.

**Synchronisation** : organisation du déroulement concomitant de processus parallèles.

**Système d'exploitation (SE)** : logiciel « chef d'orchestre » indispensable pour la bonne utilisation d'un ordinateur (*Operating System*).

**Système de fichiers** : ensemble ordonné de fichiers et de catalogues.

**Temps réel** : se dit lorsque l'exécution est contrainte par une échéance temporelle.

**Temps partagé (*Time-sharing*)** : accès simultané par plusieurs postes d'utilisateurs.

**Virtualisation** : démarche d'abstraction lors de la construction d'un système.

**Working set ou Espace de travail** : espace minimal de mémoire centrale nécessaire pour la bonne exécution d'un processus à un instant donné.



