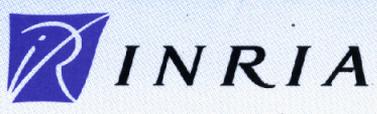


BIBLIOTHEQUE DU CERIST

Construction des systèmes d'exploitation répartis

Editeurs

**R. Balter
J.-P. Banâtre
S. Krakowiak**



COLLECTION DIDACTIQUE

C2303

COLLECTION DIDACTIQUE

**CONSTRUCTION DES SYSTÈMES
D'EXPLOITATION RÉPARTIS**

Éditeurs

R. BALTER

J.- P. BANÂTRE

S. KRAKOWIAK

Édité par



INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE
Domaine de Voluceau - Rocquencourt - B.P. 105 - 78153 Le Chesnay Cedex (France)
Téléphone : + 33 (1) 39 63 55 11 - Télécopie : + 33 (1) 39 63 58 88 - Télex : 697 033 F

6427

Édité par
l'Institut National de Recherche en Informatique et en Automatique
Directeur de publication
Alain Bensoussan
Président de l'INRIA

© INRIA

Toute représentation ou reproduction, intégrale ou partielle, faite sans le consentement du Président de l'Institut est illicite (loi du 11 mars 1957, alinéa 1er de l'article 40).
Cette représentation ou reproduction illicite, par quelques procédés que ce soit, constituerait une contrefaçon sanctionnée par les articles 425 et suivants du Code Pénal.
La loi du 11 mars 1957 n'autorise, aux termes des alinéas 2 et 3 de l'article 41, que les copies ou reproductions strictement réservées à l'usage privé du copiste et non destinées à l'utilisation collective d'une part, et d'autre part, que les analyses et les courtes citations dans un but d'exemple et d'illustration.

Adresses des auteurs

Roland Balter	Bull-IMAG, 2 rue de Vignate 38610 Gières	balter@imag.fr
Jean-Pierre Banâtre	IRISA, Campus de Beaulieu 35042 Rennes Cedex	jpbanatre@irisa.fr
Michel Banâtre	IRISA-INRIA, Campus de Beaulieu 35042 Rennes Cedex	banatre@irisa.fr
Yves Deswarte	LAAS-INRIA, 7 av. du Colonel Roche 31077 Toulouse Cedex	deswarte@laas.fr
Sacha Krakowiak	Bull-IMAG, 2 rue de Vignate 38610 Gières	krakowiak@imag.fr
Marc Rozier	Chorus-Systèmes, 6 rue Gustave Eiffel 78182 Saint-Quentin en Yvelines	mr@chorus.fr
Marc Shapiro	INRIA-Rocquencourt, BP 105 78153 Le Chesnay Cedex	shapiro@inria.fr
Jean-Pierre Verjus	IMAG, 46 av. Félix Viallet 38031 Grenoble Cedex	verjus@imag.fr

BIBLIOTHEQUE DU CERIST

Avant-Propos

Au cours des dernières années, la demande des utilisateurs de systèmes informatiques a mis en évidence les besoins suivants :

- outils pour la communication, le partage de l'information, la prise de décisions coopérative,
- gestion décentralisée des ressources matérielles et de l'information,
- disponibilité permanente du service malgré la possibilité de défaillances,
- capacité d'évolution progressive du matériel et du logiciel pour faire face à l'évolution des spécifications et de la technologie.

Les systèmes informatiques répartis répondent à ces demandes. D'autre part, l'évolution technique a permis de réaliser des systèmes de communication performants, et des postes de travail dont le rapport coût-performances est très favorable par rapport aux grands systèmes partagés. La conjonction de ces facteurs (caractéristiques de la demande, capacités techniques, aspects économiques) explique la très importante croissance actuelle du marché de l'informatique répartie (de l'ordre de 30% par an).

La mise en œuvre d'applications réparties repose sur des outils logiciels de base chargés de gérer la répartition des traitements et des données sur un ensemble de machines reliées par un réseau de communication. Pendant longtemps, ces outils ont été construits en ajoutant des primitives de communication au-dessus de systèmes d'exploitation existants. On a ensuite assisté au développement de systèmes spécifiques intégrant la gestion des communications, des ressources et des informations sur un ensemble de machines interconnectées, constituant ainsi un *système d'exploitation réparti*.

Le domaine des systèmes d'exploitation répartis connaît une évolution rapide : une expérience importante a été acquise au travers de projets de recherche, et on assiste à l'apparition des premiers produits industriels. Il est maintenant possible de dégager des principes de conception et des règles de savoir-faire applicables à la construction de ces systèmes. Cet ouvrage vise à présenter de manière raisonnée l'état actuel de ce domaine.

Notre présentation a été guidée par les choix suivants :

- Démarche descendante, fondée sur la vision d'un système en tant que machine virtuelle fournissant une certaine interface, idéalement définie par un langage de programmation. Cela explique qu'un des premiers chapitres soit consacré à la présentation des modèles et langages pour le parallélisme et la répartition.
- Mise en évidence des concepts de base liés à la solution des divers problèmes des systèmes répartis : désignation, synchronisation, gestion cohérente de l'information, sécurité et protection, etc.

- Orientation vers la construction effective de systèmes. Nous avons cherché à illustrer la présentation des concepts par des exemples tirés de systèmes existants. Un chapitre final présente en détail la technologie des micro-noyaux, composants importants des nouvelles architectures de systèmes répartis.

Les aspects suivants sont traités :

- Rappels sur les systèmes de communication et les architectures multiprocesseurs
- Modèles et langages pour le parallélisme et la répartition
- Principes de la désignation et de la synchronisation dans les systèmes répartis
- Gestion des informations réparties : mémoire virtuelle, fichiers, objets
- Gestion des transactions et cohérence de l'information
- Tolérance aux fautes, sécurité, protection
- Technologie des noyaux de systèmes : exemples de Chorus et Mach

Le choix des thèmes et de leur ordre de présentation résulte des idées directrices formulées ci-dessus, et aussi de l'expérience propre des auteurs. Nous n'avons pas prétendu couvrir tout le domaine des systèmes d'exploitation répartis, et certains aspects ont été laissés de côté, soit parce qu'il nous semblait difficile d'en donner, aujourd'hui, une présentation synthétique, soit en raison de notre manque d'expérience dans le domaine. Ainsi, par exemple, ne sont pas traités les aspects liés à l'administration des systèmes répartis, ni ceux qui touchent spécifiquement aux applications en temps réel et au contrôle de procédés.

Le public auquel s'adresse ce livre comprend les enseignants, chercheurs et ingénieurs intéressés par les systèmes et applications répartis. Les connaissances prérequisées sont des éléments sur les systèmes de communication : téléinformatique, réseaux d'ordinateurs, protocoles de communication ; les principes de base de la conception des systèmes d'exploitation classiques: gestion des processus et synchronisation, mémoire virtuelle, désignation et gestion de l'information ; les principes des langages de programmation.

Cet ouvrage a initialement été préparé comme support de cours pour une Ecole sur la Construction des Systèmes d'Exploitation Répartis, qui a été organisée par l'INRIA en 1990 et sera reprise en 1991. Cette Ecole est issue des travaux de l'Opération «Systèmes Répartis» du Programme de Recherches Coordonnées C³ (Communication, Concurrency, Coopération) du MRT et du CNRS, qui regroupe actuellement quatre projets de recherche expérimentaux sur la construction de systèmes d'exploitation répartis : Gothic (IRISA-INRIA et Bull, Rennes) ; Guide (Bull et IMAG, Grenoble) ; Saturne (LAAS et INRIA, Toulouse) et SOR (INRIA, Rocquencourt). La société Chorus Systèmes est également associée à cette opération.

L'Ecole 1990, qui a eu lieu à la Briantais (Saint-Malo) a réuni 56 participants (industrie et administration : 20, universités et centres de recherche : 36). Un réseau local

installé sur place a permis d'illustrer les cours par des exemples de systèmes répartis : prototypes de recherche (Amoeba, Gothic, Guide, Saturne, SOS) ou produits industriels (Chorus, OSF/1).

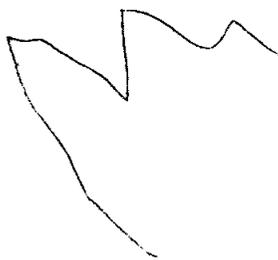
Nous remercions :

- Les organismes qui ont apporté leur soutien à l'organisation de l'Ecole INRIA 1990 sur la Construction des Systèmes d'Exploitation Répartis : Bull, le PRC C³ (CNRS-MRT), l'Open Software Foundation (OSF) Research Institute, ainsi que le Service des Relations Extérieures de l'INRIA qui a assuré la préparation et l'organisation matérielle de ces Ecoles et l'Atelier Informatique de l'IRISA qui a mis en place l'infrastructure du réseau local.
- Les participants à l'Ecole, dont les critiques et suggestions ont permis d'améliorer la forme et le contenu de la présentation.
- Les organismes d'appartenance des auteurs : Bull, Chorus-Systèmes, l'IMAG¹, l'INRIA², l'IRISA³ et le LAAS⁴, qui ont fourni les moyens nécessaires à la préparation des chapitres de ce livre.

Les éditeurs

R. Balter, J.-P. Banâtre, S. Krakowiak

avril 1991



¹ Institut d'Informatique et de Mathématiques Appliquées de Grenoble

² Institut National de Recherche en Informatique et Automatique

³ Institut de Recherche en Informatique et Systèmes Aléatoires (Rennes)

⁴ Laboratoire d'Automatique et d'Analyse des Systèmes du CNRS (Toulouse)

BIBLIOTHEQUE DU CERIST

Chapitre 1 : Introduction aux systèmes répartis

- 1. Problématique et bref historique**
- 2. Prises de décisions réparties et coopératives**
 - 2.1 Synchronisation de processus
 - 2.2 Incertitudes spatiales ou temporelles
 - 2.3 Accès cohérent à une ressource
 - 2.4 Ordonnement d'événements
 - 2.5 Terminaison
 - 2.6 Fiabilité
 - 2.7 Raisonnement et intelligence artificielle
- 3. Objets, programmes... dans un système réparti**
 - 3.1 Gestion de processus
 - 3.2 Protocoles et appel de procédures
 - 3.3 Objets
 - 3.4 Fichiers, bases de données et transactions

Chapitre 2 : Rappels sur les systèmes de communication et les machines parallèles

- 1. Rappels sur les systèmes de communication**
 - 1.1 Interfaces et protocoles
 - 1.2 Classification des systèmes de communication
 - 1.3 Exemples de protocoles
- 2. Notions sur les architectures de machines parallèles**
 - 2.1 Classification des machines parallèles
 - 2.2 Multiprocesseurs à mémoire commune
 - 2.3 Multiprocesseurs à mémoire distribuée
- 3. Résumé**
- 4. Notes bibliographiques**

Chapitre 3 : Modèles et langages pour le parallélisme

1. Introduction

- 1.1 Le concept de processus
- 1.2 Les processus et leurs relations

2. Communication et synchronisation

- 2.1 La communication entre processus : généralités
- 2.2 Communication fondée sur le contrôle procédural
- 2.3 La communication en ADA
- 2.4 Le concept de multiprocédure
- 2.5 Communication de type "rendez-vous"
- 2.6 Communication fondée sur le partage
- 2.7 L'approche LINDA

3. Programmation orientée objet et parallélisme

- 3.1 Introduction à la programmation orientée objet
- 3.2 Objets et parallélisme : l'exemple langage GUIDE

4. Programmation parallèle et tolérance aux fautes

- 4.1 Atomicité
- 4.2 Les transactions atomiques dans ARGUS

5. Développement de programmes sûrs

- 5.1 Preuves de programmes
- 5.2 Construction raisonnée de programmes parallèles

6. Eléments de mise en oeuvre

- 6.1 Mise en oeuvre de l'appel de procédure à distance
- 6.2 Mise en oeuvre du rendez-vous
- 6.3 Discussion

7. Conclusion

Chapitre 4 : Principes et mécanismes de base de la répartition

1. Introduction

- 1.1 Motivation des systèmes répartis
- 1.2 Caractéristiques des systèmes répartis
- 1.3 Exemple introductif
- 1.4 Les problèmes de la répartition

2. Désignation

- 2.1 Introduction
- 2.2 Désignation interne
- 2.3 Désignation symbolique
- 2.4 Serveurs de désignation

3. Synchronisation

- 3.1 Ordonnancement des événements
- 3.2 Exclusion mutuelle
- 3.3 Exemples et applications d'algorithmes répartis

4. Cohérence de l'information répartie

- 4.1 Le problème de la cohérence dans un système réparti
- 4.2 Etat global d'un système réparti
- 4.3 Diffusion fiable

5. Modèles et mécanismes de structuration des systèmes répartis

- 5.1 Un schéma de base : le modèle client-serveur
- 5.2 Appel de procédure à distance
- 5.3 Autres modèles de structuration des systèmes répartis

6. Notes bibliographiques

Chapitre 5 : La gestion répartie de la mémoire

1. Introduction

2. Le partage de la ressource mémoire dans les architectures multiprocesseurs

- 2.1 Le partage de la mémoire dans les architectures fortement couplées
- 2.2 Mise en oeuvre du partage de la mémoire dans les architectures faiblement couplées
- 2.3 Discussion

- 3. Gestion des informations permanentes sur les architectures réparties**
 - 3.1 Les différents modes d'adressage des informations permanentes
 - 3.2 Exemples de mise en oeuvre de systèmes à stockage uniforme des données
- 4. Architectures multiprocesseurs fortement couplées tolérantes aux fautes**
 - 4.1 La machine SEQUOIA
 - 4.2 La machine tc-FTMP
 - 4.3 Discussion
- 5. Sommaire**

Chapitre 6 : Gestion répartie de fichiers

- 1. Introduction aux SGRF**
 - 1.1 Définition d'un SGF
 - 1.2 Systèmes répartis de gestion de fichiers
 - 1.3 Observation empirique de quelques SGF
- 2. Les problèmes de base des SRGF**
 - 2.1 Nomination des fichiers
 - 2.2 Nomination et localisation
 - 2.3 Destruction
 - 2.4 Disponibilité et réplication
 - 2.5 Maintien de la cohérence entre exemplaires
 - 2.6 Caches
- 3. Deux systèmes répartis de gestion de fichiers**
 - 3.1 Le Network File System (NFS)
 - 3.2 Andrew File System (AFS)
- 4. Conclusion**
- 5. Notes bibliographiques**

Chapitre 7 : Gestion répartie d'objets

1. Introduction

2. L'approche de programmation par objets

- 2.1 Concepts de base
- 2.2 Principales propriétés

3. Objets d'un système d'exploitation

- 3.1 Application de l'approche par objets au SE
- 3.2 L'exemple de Choices
- 3.3 La bibliothèque de protocoles de SOS
- 3.4 Conclusion

4. Système d'exploitation d'objets

5. Mécanismes d'un système d'exploitation d'objets

- 5.1 Introduction
- 5.2 Granularité
- 5.3 Objet partagé
- 5.4 Persistance
- 5.5 Objets complexes
- 5.6 Mobilité
- 5.7 Objet fragmenté
- 5.8 Invocation distance
- 5.9 Conclusion : le partage en réparti

6. Glanage (GC), ou ramasse-miettes

- 6.1 Position du problème
- 6.2 Principes du glanage
- 6.3 Comptage de références
- 6.4 Traçage
- 6.5 Le glanage en réparti
- 6.6 Glanage : conclusion

7. Gestion d'objets

- 7.1 Persistance
- 7.2 Accès aux objets
- 7.3 Défaut d'objet
- 7.4 Recherche

8. Quelques systèmes d'objets répartis

- 8.1 SOS et Cool
- 8.2 Guide/Comandos
- 8.3 Gothic
- 8.4 Emerald et Amber

9. SOS, un système d'exploitation réparti d'objets

- 9.1 Principaux concepts
- 9.2 Le prototype
- 9.3 Les objets code
- 9.4 Les objets fragmentés
- 9.5 Migration des objets élémentaires

10. Notes bibliographiques

Chapitre 8 : Gestion répartie des transactions

1. Introduction : cohérence et transactions

- 1.1 Présentation du problème
- 1.2 Le système d'information
- 1.3 Cohérence sémantique - contraintes d'intégrité
- 1.4 Principes du contrôle de la cohérence
- 1.5 Transactions : définition - modélisation
- 1.6 Propriétés d'une transaction
- 1.7 Maintien de la cohérence - les mécanismes transactionnels
- 1.8 Architecture d'un système transactionnel

2. Contrôle des accès concurrents

- 2.1 Principe de fonctionnement
- 2.2 Présentation des techniques de contrôle
- 2.3 Techniques de verrouillage
- 2.4 Interblocages
- 2.5 Traitement des interblocages. Présentation des méthodes
- 2.6 Méthodes de détection-guérison dans un système réparti
- 2.7 Méthodes de prévention dans un système réparti
- 2.8 Conclusion

3. Validation et reprise

- 3.1 Problématique : impact des défaillances
- 3.2 Principe du contrôle d'atomicité : validation et journal
- 3.3 Techniques de validation et reprise
- 3.4 Atomicité d'une transaction répartie
- 3.5 Protocoles de validation
- 3.6 Traitement des défaillances des mémoires secondaires
- 3.7 Conclusion

4. Transactions : développements récents

- 4.1 Introduction
- 4.2 Transactions emboîtées
- 4.3 Objets complexes et transactions de longue durée
- 4.4 Cohérence sémantique
- 4.5 Noyaux transactionnels

5. Bibliographie

Chapitre 9 : Tolérance aux fautes, sécurité et protection

1. Notions de sûreté de fonctionnement

- 1.1 Concepts de base et terminologie
- 1.2 Classifications des fautes, des erreurs et des défaillances

2. Techniques de tolérance aux fautes accidentelles

- 2.1 Traitement d'erreur et traitement de faute
- 2.2 Détection et recouvrement
- 2.3 Compensation d'erreur

3. Sécurité et protection

- 3.1 Classification des attaques contre la sécurité
- 3.2 Contrôles d'accès et protection
- 3.3 Autres aspects de la sécurité

4. Conclusion

5. Notes bibliographiques

Chapitre 10 : Technologie des noyaux de systèmes

1. Introduction

- 1.1 Systèmes d'exploitation répartis
- 1.2 Architecture modulaire : noyau et serveurs
- 1.3 Chorus
- 1.4 Match

2. Noyaux de systèmes

- 2.1 Objets élémentaires
- 2.2 Exécutif
- 2.3 Communications

3. Gestion de mémoire virtuelle

- 3.1 Gestion des contextes
- 3.2 Gestion des segments

4. Construction de sous-systèmes

- 4.1 Chorus/MiX
- 4.2 Principes de réalisation
- 4.3 Extensions à Unix
- 4.4 Avantages de la modularité
- 4.5 Sous-systèmes de Mach

5. Conclusion

6. Notes bibliographiques