



HAL
open science

Gestion et communication d'informations dans un système réparti sur un réseau local

Serge Sasyan

► **To cite this version:**

Serge Sasyan. Gestion et communication d'informations dans un système réparti sur un réseau local. Modélisation et simulation. Institut National Polytechnique de Grenoble - INPG, 1983. Français. NNT: . tel-00306547

HAL Id: tel-00306547

<https://theses.hal.science/tel-00306547>

Submitted on 28 Jul 2008

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.

THESE

présentée à

l'Institut National Polytechnique de Grenoble

pour obtenir le grade de

**DOCTEUR INGENIEUR
«Informatique»**

par

Serge SASYAN



GESTION ET COMMUNICATION D'INFORMATIONS

DANS UN SYSTEME REPARTI SUR UN RESEAU LOCAL.



Thèse soutenue le 7 février 1983 devant la commission d'examen.

C. DELOBEL **Président**

S. GUIBOUD-RIBAUD
J. MOSSIERE
V. QUINT **} Examineurs**

N. NAFFAH **Rapporteur extérieur**



INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE

Année universitaire 1979-1980

Président : M. Philippe TRAYNARD

Vice-Présidents : M. Georges LESPINARD

M. René PAUTHENET

PROFESSEURS DES UNIVERSITES

MM.	ANCEAU François	Informatique fondamentale et appliquée
	BENOIT Jean	Radioélectricité
	BESSON Jean	Chimie Minérale
	BLIMAN Samuel	Electronique
	BLOCH Daniel	Physique du Solide - Cristallographie
	BOIS Philippe	Mécanique
	BONNETAIN Lucien	Génie Chimique
	BONNIER Etienne	Métallurgie
	BOUVARD Maurice	Génie Mécanique
	BRISSONNEAU Pierre	Physique des Matériaux
	BUYLE-BODIN Maurice	Electronique
	CHARTIER Germain	Electronique
	CHERADAME Hervé	Chimie Physique Macromoléculaires
Mme	CHERUY Arlette	Automatique
MM.	CHIAVERINA Jean	Biologie, Biochimie, Agronomie
	COHEN Joseph	Electronique
	COUMES André	Electronique
	DURAND Francis	Métallurgie
	DURAND Jean-Louis	Physique Nucléaire et Corpusculaire
	FELICI Noël	Electrotechnique
	FOULARD Claude	Automatique
	GUYOT Pierre	Métallurgie Physique
	IVANES Marcel	Electrotechnique
	JOUBERT Jean-Claude	Physique du Solide - Cristallographie
	LACOUME Jean-Louis	Géographie - Traitement du Signal
	LANCIA Roland	Electronique - Automatique
	LESIEUR Marcel	Mécanique
	LESPINARD Georges	Mécanique
	LONGEQUEUE Jean-Pierre	Physique Nucléaire Corpusculaire
	MOREAU René	Mécanique
	MORET Roger	Physique Nucléaire Corpusculaire
	PARIAUD Jean-Charles	Chimie - Physique
	PAUTHENET René	Physique du Solide - Cristallographie
	PERRET René	Automatique

.../...

MM.	PERRET Robert	Electrotechnique
	PIAU Jean-Michel	Mécanique
	PIERRARD Jean-Marie	Mécanique
	POLOUJADOFF Michel	Electrotechnique
	POUPOT Christian	Electronique - Automatique
	RAMEAU Jean-Jacques	Chimie
	ROBERT André	Chimie Appliquée et des matériaux
	ROBERT François	Analyse numérique
	SABONNADIÈRE Jean-Claude	Electrotechnique
Mme	SAUCIER Gabrielle	Informatique fondamentale et appliquée
M.	SOHM Jean-Claude	Chimie - Physique
Mme	SCHLENKER Claire	Physique du Solide - Cristallographie
MM.	TRAYNARD Philippe	Chimie - Physique
	VEILLON Gérard	Informatique fondamentale et appliquée
	ZADWORNÝ François	Electronique

CHERCHEURS DU C.N.R.S. (Directeur et Maître de Recherche)

M.	FRUCHART Robert	Directeur de Recherche
MM.	ANSARA Ibrahim	Maître de Recherche
	BRONOEL Guy	Maître de Recherche
	CARRE René	Maître de Recherche
	DAVID René	Maître de Recherche
	DRIOLE Jean	Maître de Recherche
	KAMARINOS Georges	Maître de Recherche
	KLEITZ Michel	Maître de Recherche
	LANDAU Ioan-Doré	Maître de Recherche
	MERMET Jean	Maître de Recherche
	MUNIER Jacques	Maître de Recherche

Personnalités habilitées à diriger des travaux de recherche (décision du Conseil Scientifique)

E.N.S.E.E.G.

MM.	ALLIBERT Michel
	BERNARD Claude
	CAILLET Marcel
Mme	CHATILLON Catherine
MM.	COULON Michel
	HAMMOU Abdelkader
	JOUD Jean-Charles
	RAVAINE Denis
	SAINFORT

C.E.N.G.

.../...

MM. SARRAZIN Pierre
 SOUQUET Jean-Louis
 TOUZAIN Philippe
 URBAIN Georges

Laboratoire des Ultra-Réfractaires ODEILLO

E.N.S.M.E.E.

MM. BISCONDI Michel
 BOOS Jean-Yves
 GUILHOT Bernard
 KOBILANSKI André
 LALAUZE René
 LANCELOT François
 LE COZE Jean
 LESBATS Pierre
 SOUSTELLE Michel
 THEVENOT François
 THOMAS Gérard
 TRAN MINH Canh
 DRIVER Julian
 RIEU Jean

E.N.S.E.R.G.

MM. BOREL Joseph
 CHEHIKIAN Alain
 VIKTOROVITCH Pierre

E.N.S.I.E.G.

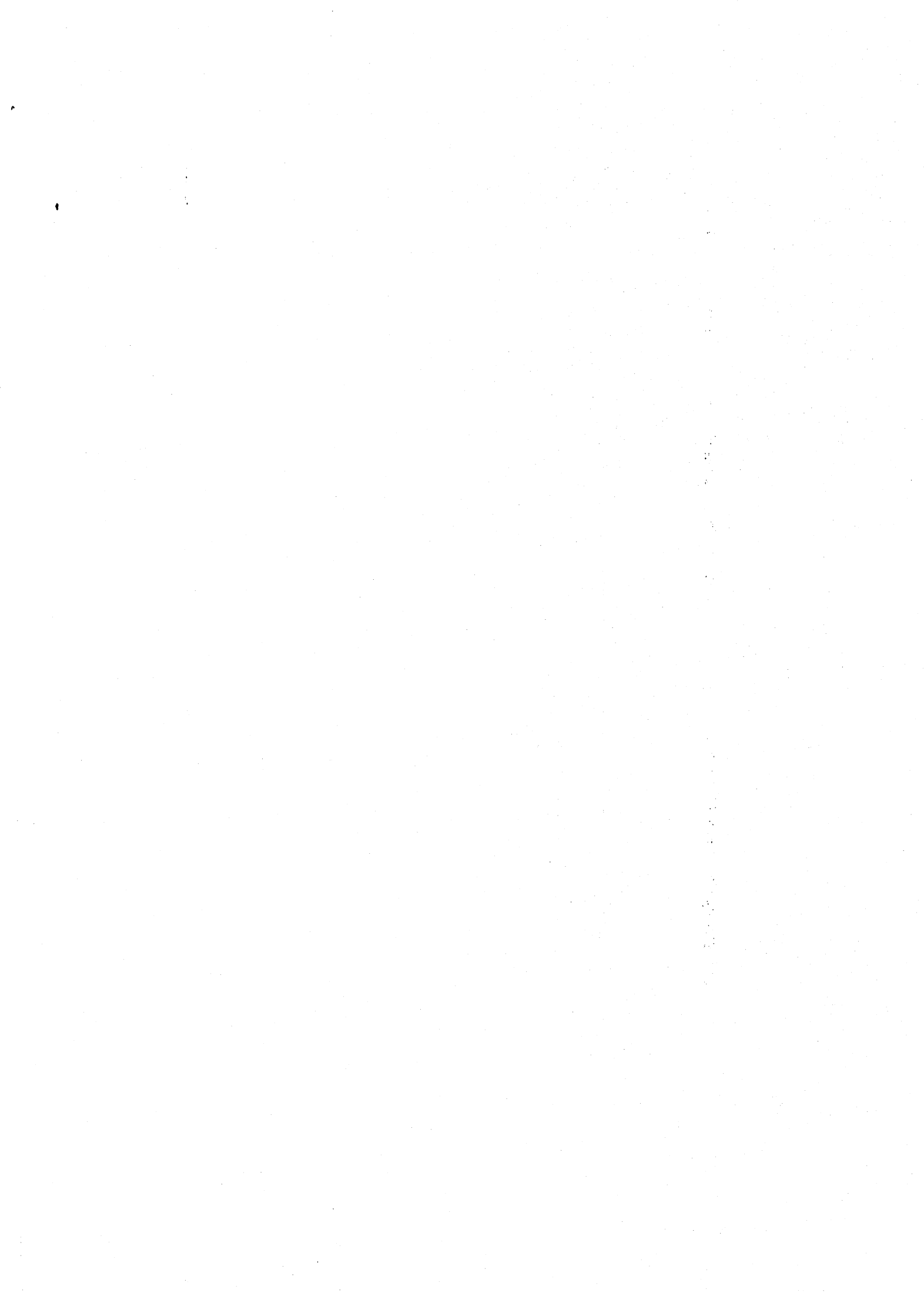
MM. BORNARD Guy
 DESCHIZEAUX Pierre
 GLANGEAUD François
 JAUSSAUD Pierre
 Mme JOURDAIN Geneviève
 MM. LEJEUNE Gérard
 PERARD Jacques

E.N.S.H.G.

M. DELHAYE Jean-Marc

E.N.S.I.M.A.G.

MM. COURTIN Jacques
 LATOMBE Jean-Claude
 LUCAS Michel
 VERDILLON André



Je tiens à remercier de tout coeur,

Monsieur C. DELOBEL, professeur à l'USNG, qui a bien voulu me faire l'honneur de présider le jury de cette thèse,

Monsieur J. MOSSIERE, professeur de l'INPG à l'ENSIHAG, qui a assuré la direction de cette thèse,

Monsieur V. QUINT, responsable de l'équipe "Réseau Local" du laboratoire IMAG, qui a dirigé mon travail au sein de cette équipe.

Monsieur N. NAFFAH, de l'INRIA, directeur du projet pilote KAYAK où il m'a accueilli pour un stage, qui a bien voulu être mon rapporteur pour cette thèse,

Monsieur S. GUIBOUD-RIBAUD, de la société Hewlett-Packard qui apporte à ce jury son point de vue de l'industrie,

Je remercie aussi ces personnes pour les remarques de leur part dont a bénéficié ce rapport, et tout particulièrement Monsieur J. MOSSIERE qui m'a aidé à le corriger tout au long de sa rédaction.

Je témoigne aussi ma reconnaissance à Monsieur S. KRAKOWIAK, professeur à l'USHG, d'avoir proposé le sujet de recherche grâce auquel j'ai obtenu la bourse qui m'a permis de préparer cette thèse.

Je tiens aussi à remercier toutes les personnes de l'équipe "Réseau Local",

Monsieur X. ROUSSET DE PINA qui m'a le premier invité à participer à cette équipe,

Monsieur G. SERGEANT qui jusqu'à son départ m'a incité à rédiger ce rapport,

Mesdames H. RICHY et I. VATTON, indissociables dans mon esprit, qui m'ont aidé dans mon travail et se sont attentivement intéressées à l'évolution de ce rapport,

Madame J. REY qui a contribué à la frappe de ce rapport en installant le système de traitement de texte que j'ai utilisé.

Je salue aussi toutes les personnes de l'étage qui trouveront ici l'expression de ma vive sympathie.

Que toutes les personnes qui ont permis la réalisation matérielle de ce rapport dans les meilleures conditions trouvent ici le témoignage de ma gratitude.

Il me reste à remercier,

Mademoiselle C. ROISIN qui m'a fait l'amitié de m'aider, en lisant ce rapport, à en parfaire la rédaction,

et Traian MUNTEAN qui m'a souvent apporté des avis éclairés et désintéressés au cours de nos discussions et qui m'a aussi initié à l'escalade et à la descente en rappel.

En souvenir de ma grand-mère et de mon père,
à ma famille et mes amis,
pour Anahid et sa "4e génération"!

TABLE DES MATIERES

<u>I</u>	<u>INTRODUCTION</u>	1
<u>II</u>	<u>LES RESEAUX LOCAUX</u>	5
<u>1</u>	<u>Présentation</u>	7
<u>1.1</u>	<u>Ethernet</u>	
<u>1.2</u>	<u>L'anneau de Cambridge</u>	8
<u>2</u>	<u>Nouveaux avantages et problèmes posés</u>	13
<u>2.1</u>	<u>Mise en commun des ressources</u>	14
<u>2.2</u>	<u>Partage du traitement</u>	15
<u>2.3</u>	<u>Communications plus fiables et moins coûteuses</u>	16
<u>2.4</u>	<u>Ordinateurs à capacité plus faible</u>	17
<u>2.5</u>	<u>Réalisation de services par collaboration entre sites</u>	18
<u>3</u>	<u>Le réseau local DANUBE</u>	19
<u>3.1</u>	<u>Le réseau local</u>	20
<u>3.1.1</u>	<u>Description du réseau</u>	
<u>3.1.2</u>	<u>Principe de fonctionnement</u>	21
<u>3.1.3</u>	<u>Réalisation</u>	23
<u>3.2</u>	<u>Configuration du réseau</u>	24
<u>3.3</u>	<u>Le service de transport</u>	25
<u>III</u>	<u>GESTION DES INFORMATIONS DANS SYSTEME REPARTI : GENERALITES</u>	29
<u>1</u>	<u>Les fichiers sur un réseau local</u>	31
<u>1.1</u>	<u>Nature des fichiers</u>	
<u>1.2</u>	<u>Généralités sur les fichiers</u>	
<u>1.2.1</u>	<u>Conservation et représentation des fichiers</u>	
<u>1.2.1.1</u>	<u>Organisation logique</u>	32
<u>1.2.1.2</u>	<u>Organisation physique</u>	
<u>1.2.1.3</u>	<u>Attributs</u>	33
<u>1.2.1.4</u>	<u>Catalogues</u>	
<u>1.2.2</u>	<u>Transfert et accès distant</u>	34

1.2.2.1	Simplicité de l'organisation logique	
1.2.2.2	Finesse de l'accès aux données	
<u>1.3</u>	<u>Gestion répartie des fichiers</u>	35
1.3.1	Systèmes de gestion de fichiers interconnectés	
1.3.2	Serveur de fichiers	37
1.3.2.1	Serveur complet	
1.3.2.2	Serveur élémentaire	39
1.3.3	Gestion globale des ressources de type fichier	
<u>2</u>	<u>Services à offrir sur le réseau</u>	42
<u>2.1</u>	<u>Services existants localement</u>	
2.1.1	Stockage et gestion des informations	43
2.1.2	Echange d'information	
2.1.3	Accès à l'information	
2.1.4	Sortie de l'information	
<u>2.2</u>	<u>Apports du réseau</u>	44
2.2.1	Systèmes interconnectés	45
2.2.1.1	Stockage et gestion	46
2.2.1.2	Echange	47
2.2.1.3	Impression	48
2.2.2	Système réseau	49
2.2.2.1	Stockage et gestion	
2.2.2.2	Echange	51
2.2.2.3	Impression	
2.2.3	Fonction non perturbée : accès aux fichiers	
<u>2.3</u>	<u>Accès aux services étendus</u>	55
2.3.1	Maintenance et accès	
2.3.2	Stockage et impression	56
2.3.3	Transfert	57
<u>2.4</u>	<u>Utilisation des services étendus</u>	
2.4.1	Utilisateur physique	58
2.4.2	Programme d'application	59
2.4.3	Système intégré	
<u>3</u>	<u>Problèmes soulevés par les échanges sur les réseaux locaux</u>	61
<u>3.1</u>	<u>Désignation, localisation et structuration des ressources</u>	62

3.1.1	Désignation et localisation	
3.1.1.1	Désignation	63
3.1.1.1.1	Désignation interne	64
3.1.1.1.2	Désignation externe	65
3.1.1.2	Localisation	66
3.1.1.3	Interférences	
3.1.2	Structure des catalogues réseau	67
3.1.2.1	Racine virtuelle	68
3.1.2.2	Arborescence	69
3.1.3	Recherche des objets et améliorations	70
3.1.3.1	Accélération de la recherche	71
3.1.3.2	Meilleure fiabilité	72
3.1.3.2.1	Informations conservées	73
3.1.3.2.2	Gestion de l'espace de stockage	
<u>3.2</u>	<u>Hétérogénéité</u>	75
3.2.1	Les sites	
3.2.2	Les systèmes d'exploitation	77
<u>3.3</u>	<u>Protection</u>	
3.3.1	Définition des droits d'accès	78
3.3.1.1	Système à capacité	
3.3.1.2	Listes de droits d'accès	79
3.3.2	Lieu de conservation de ces droits	80
3.3.3	Assurer pratiquement la protection	82
<u>3.4</u>	<u>Cohérence des données conservées</u>	84
3.4.1	Violation de l'intégrité des données	85
3.4.1.1	Première cause : accès simultané	
3.4.1.2	Seconde cause : panne imprévue	86
3.4.1.2.1	Retour à l'état antérieur	
3.4.1.2.2	Poursuite des modifications	
3.4.2	Copies multiples	87
3.4.2.1	Support stable	
3.4.2.2	Rapidité d'accès	88

3.4.3	Catalogues multiples	89
3.4.4	Fiabilité proportionnelle à la valeur des données	89
3.5	<u>Fiabilité</u>	90
3.5.1	Fiabilité du matériel	91
3.5.2	Reprise logicielle des pannes	92
<u>IV</u>	<u>SERVICES OFFERTS POUR LA GESTION DES INFORMATIONS</u>	95
<u>1</u>	<u>Nature des services</u>	96
<u>1.1</u>	<u>Utilisation directe</u>	
<u>1.2</u>	<u>Appel aux services par un programme</u>	97
<u>2</u>	<u>Mise en oeuvre des services</u>	98
<u>2.1</u>	<u>Solution Client-Serveur</u>	99
2.1.1	Serveur	100
2.1.1.1	Accès au service	
2.1.1.2	Rôle du serveur	
2.1.2	Client	101
2.1.3	Protocole entre Client et Serveur	
2.1.3.1	Conditions sur le protocole	102
2.1.3.1.1	Contrôle d'erreur	
2.1.3.1.2	Contrôle de flux	
2.1.3.2	Modèles de protocole	
2.1.3.2.1	Caractère répétitif des requêtes	103
2.1.3.2.2	Protocoles à deux ou trois phases	104
<u>2.2</u>	<u>Interface et service partagé</u>	105
2.2.1	Serveur	106
2.2.2	Client	
2.2.2.1	Intégration à l'application	
2.2.2.2	Client multi-usagers	107
<u>3</u>	<u>Exemples d'utilisation par les applications</u>	109
<u>3.1</u>	<u>Transfert de fichiers et gestion de catalogues</u>	
<u>3.2</u>	<u>Service d'imprimerie</u>	111
<u>3.3</u>	<u>Messaagerie</u>	112

<u>V</u>	<u>REALISATIONS</u>	113
<u>1</u>	<u>Applications réalisées</u>	115
<u>1.1</u>	<u>Service de stockage, de gestion, de transfert et d'impression de fichiers</u>	116
<u>1.2</u>	<u>Interface de transfert de fichiers par programme</u>	117
<u>1.3</u>	<u>Serveur d'impression sur le réseau de burovisuels</u>	
<u>2</u>	<u>Le modèle client serveur</u>	118
<u>2.1</u>	<u>Client : interface de dialogue Homme-Machine</u>	119
<u>2.2</u>	<u>Serveur</u>	120
	<u>2.2.1</u> Impression	
	<u>2.2.2</u> Catalogues	121
	<u>2.2.3</u> Accès aux données	
<u>2.3</u>	<u>Localisation du service</u>	122
<u>3</u>	<u>Service de gestion, de transfert et d'impression de fichiers</u>	125
<u>3.1</u>	<u>Protocole de dialogue et de transfert de fichiers</u>	
	<u>3.1.1</u> Transfert du client vers le serveur	126
	<u>3.1.2</u> Transfert du serveur vers le client	127
	<u>3.1.3</u> Cas de mauvais fonctionnement	128
<u>3.2</u>	<u>Compression de données</u>	
	<u>3.2.1</u> Choix du niveau où a lieu la compression	
	<u>3.2.1.1</u> Compression au niveau du transport	129
	<u>3.2.1.2</u> Compression au niveau de l'enregistrement	
	<u>3.2.1.3</u> Comparaison de ces deux niveaux	
<u>3.2.2</u>	<u>Algorithmes de compression</u>	130
	<u>3.2.2.1</u> Compression systématique	131
	<u>3.2.2.2</u> Compression d'octets spécifiques	132
	<u>3.2.2.3</u> Compression optimisée	
<u>3.2.3</u>	<u>Choix de l'algorithme de compression</u>	133
<u>4</u>	<u>Interface de transfert par programme pour la messagerie</u>	134
<u>4.1</u>	<u>Gestion de la liaison de communication</u>	136
<u>4.2</u>	<u>Synchronisation sur la liaison</u>	137
<u>5</u>	<u>Protocole de reprise d'échange</u>	138
<u>5.1</u>	<u>Introduction</u>	
<u>5.2</u>	<u>Contexte de la communication sur un réseau local</u>	140

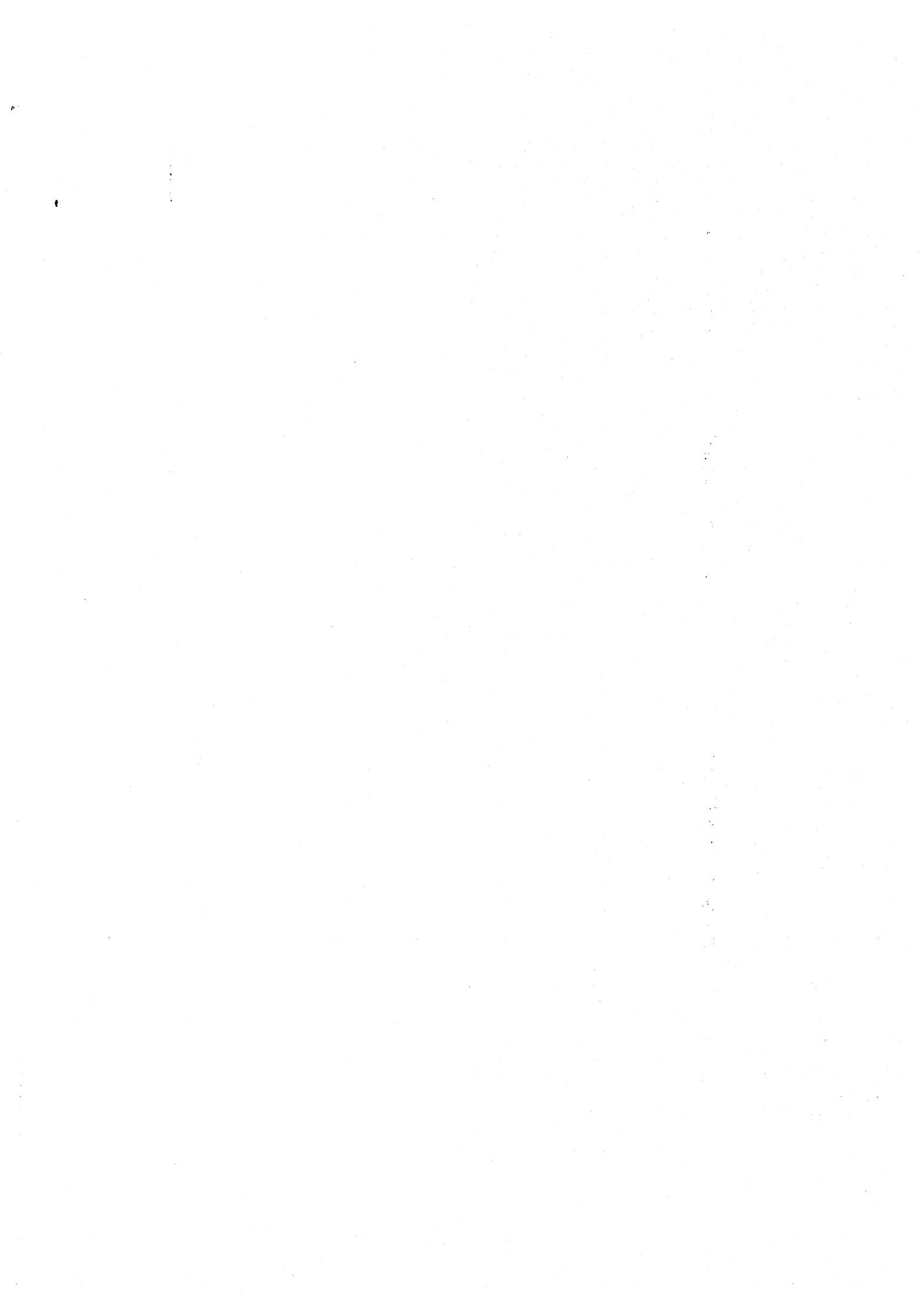
5.2.1	Configuration lors des échanges de documents	141
5.2.2	Principe du transfert	141
5.2.2.1	Emetteur, récepteur	
5.2.2.2	Contrôle de flux : blocage, déblocage	
5.2.3	Conditions sur le service d'accès à l'information	142
5.2.4	Caractéristiques du service de communication	143
5.2.4.1	Fonctionnement normal du service de communication	
5.2.4.2	Comportement en cas de fonctionnement anormal	144
5.2.4.2.1	Aspects positifs	
5.2.4.2.2	Aspects négatifs	145
<u>5.3</u>	<u>Les pannes et leur détection</u>	
5.3.1	<u>Pannes envisagées</u>	146
5.3.1.1	Pannes du matériel	
5.3.1.2	Pannes du logiciel	147
5.3.1.2.1	Pannes logicielles isolées	
5.3.1.2.2	Pannes logicielles couplées	
5.3.1.2.3	Autres pannes logicielles	148
5.3.1.3	Conditions pour une relance automatique	
5.3.2	Détection des pannes	149
5.3.2.1	Absence signalée par la S.T.	
5.3.2.2	Délai de garde en cas de blocage	
5.3.2.2.1	Détection par l'émetteur	150
5.3.2.2.2	Détection par le récepteur	
<u>5.4</u>	<u>Le protocole de reprise automatique</u>	151
5.4.1	Contexte d'un échange en cours	
5.4.1.1	Définition du document échangé	
5.4.1.2	Point d'évolution du transfert	
5.4.2	Gestion du contexte de l'échange	152
5.4.2.1	Définition des points de reprise	
5.4.2.2	Acquittement implicite grâce au contrôle de flux	153
5.4.2.3	Numérotation des lettres et situation	
5.4.2.4	Progression chez l'émetteur	154
5.4.2.5	Progression chez le récepteur	155

5.4.3	Rétablissement de la liaison rompue	
5.4.4	Initiative du rétablissement de la liaison	157
5.4.5	Rétablissement du contexte d'échange	158
5.4.6	Poursuite de l'échange	159

5.5 Conclusion

<u>VI</u>	<u>CONCLUSION</u>	161
-----------	-------------------	-----

	<u>BIBLIOGRAPHIE</u>	167
--	----------------------	-----



I

INTRODUCTION

L'informatique centralisée n'a tout d'abord donné naissance qu'à de gigantesques monstres aux capacités de traitement et de stockage prodigieuses que seuls quelques initiés pouvaient domestiquer et exploiter au bénéfice d'une grande communauté d'usagers.

Depuis une décennie, les progrès techniques ont permis de réaliser à meilleur compte les gros ordinateurs dont la puissance de calcul et les grosses capacités de stockage sont toujours nécessaires. Ils ont aussi permis de disposer d'ordinateurs de puissance et de capacité moyenne, grâce à des "modèles réduits" presque aussi performants que leurs aînés, que l'on peut enfermer dans son bureau et réserver à son usage personnel.

Serait-on passé du "un pour tous" au "chacun pour soi"? Non, car entre ces petites unités de traitement autonomes, la collaboration et les échanges s'avèrent fructueux. La mise en commun et le partage des ressources de chacun permettent à tous et à moindre frais de disposer de moyens réservés auparavant aux seuls utilisateurs des systèmes d'exploitation des grands ordinateurs.

C'est la mise en commun et le partage, la collaboration et les échanges que nous avons voulu étudier dans ce rapport. Nous nous sommes limités toutefois aux seuls échanges entre les unités de traitement et à la collaboration nécessaire à la réalisation de ces échanges.

Parallèlement à la diminution de taille des ordinateurs, nous avons assisté à l'évolution des

techniques d'interconnexion. Après avoir relié de gros ordinateurs à travers de vastes espaces géographiques, on a pu relier toutes sortes d'ordinateurs dans un domaine limité pour obtenir un réseau local.

Pour gérer et échanger des informations sur les réseaux locaux, on peut utiliser les solutions éprouvées dans les réseaux de grande étendue. Mais c'est négliger les avantages propres aux réseaux locaux : leur grand débit d'information et le faible temps d'accès au support de communication.

Les réseaux locaux, leurs avantages nouveaux et les problèmes qu'ils posent sont présentés dans le chapitre II. Ce chapitre situe aussi, par rapport au réseau local Ethernet et à l'anneau de Cambridge, le réseau local DANUBE sur lequel nous avons travaillé.

Le chapitre III analyse les besoins en matière de gestion et de communication d'informations sur un réseau local, ainsi que les difficultés rencontrées. Il présente la conservation et la représentation des fichiers d'information sur les réseaux locaux. Il décrit les services à fournir aux usagers de ce réseau pour qu'ils puissent manipuler l'information, la conserver, l'échanger et l'imprimer. La suite du chapitre soulève les problèmes à traiter et à résoudre pour offrir ces services: la désignation et la localisation des informations, leur protection et la cohérence de leur manipulation, ainsi que l'hétérogénéité des matériels interconnectés et la fiabilité des outils mis à la disposition des usagers.

Le chapitre IV expose les services que l'on doit procurer aux usagers, leur nature et essentiellement leur mise en oeuvre. Pour la plupart, les services sont réalisés par des serveurs répartis sur le réseau. Les serveurs sont

sollicités par des clients qui agissent sous le contrôle des usagers finaux. Trois exemples d'utilisation et d'intégration de ces services, concernant le transfert et la gestion de fichiers, la messagerie et l'impression, sont donnés à titre d'illustration.

Le chapitre V décrit les services réalisés dans le cadre de cette étude menée au sein de l'équipe Réseau local du groupe logiciel de base du Laboratoire IMAG. L'équipe de Grenoble a collaboré au projet KAYAK de l'INRIA en étudiant les nouveaux protocoles et applications adaptés aux réseaux locaux. Pour appréhender ces applications en vraie grandeur des maquettes ont été développées qui recouvrent les nouvelles possibilités des réseaux locaux. A cause de la configuration du réseau, le problème d'hétérogénéité des matériels a été occulté et la protection des documents a été réalisée en se basant uniquement sur les possibilités offertes par les systèmes existants. Les services fournis permettent de stocker, de gérer, d'échanger et d'imprimer les fichiers sur le réseau local. On dispose ainsi dans un système réparti de services équivalents à ceux d'un système centralisé. A la fin de ce chapitre, nous proposons un protocole dont la mise en oeuvre dans les échanges d'information assure la reprise et la poursuite des échanges après des pannes ou des incidents matériels ou logiciels.

Nous concluons en orientant la suite du travail vers des systèmes et des réalisations de réseaux locaux plus performants, vers l'interconnexion de réseaux locaux et la prise en compte de services de transport assurant la diffusion fiable des informations sur un réseau local.

II

LES RESEAUX LOCAUX

Bien qu'apparus récemment, les réseaux locaux ont été suffisamment décrits et commentés pour qu'il ne soit pas indispensable d'y revenir ici longuement.

Sur un ensemble de matériels informatiques interconnectés par un moyen de communication rapide et fiable dans une aire géographique limitée, les solutions techniques matérielles sont en avance sur les outils logiciels informatiques qui permettent de tirer parti de leurs avantages.

Nous ferons ici une brève présentation des réseaux locaux en nous intéressant à deux réalisations marquantes qui sont le réseau Ethernet réalisé par Xerox à Palo Alto en Californie et le système CMDS réalisé sur le réseau en anneau de l'Université de Cambridge en Grande Bretagne.

Ensuite, nous résumerons les avantages apportés par les réseaux locaux et les problèmes qui se posent pour exploiter ces avantages.

Enfin, nous présenterons notre outil de recherche : le réseau local DANUBE, dérivé d'Ethernet, développé par l'INRIA au sein du projet pilote KAYAK, ainsi que les outils de base dont nous disposons pour la communication sur ce réseau, c'est à dire essentiellement le service de transport développé par l'équipe Réseau Local de l'IMAG.

1 PRESENTATION

Apparus à la suite des progrès techniques du début des années 70, les réseaux locaux ont été réalisés sous différentes formes et selon différentes architectures : de l'étoile au bus en passant par les anneaux, avec différents média de communication : ondes radio, paire de fils torsadés, câble coaxial, fibres optiques, etc <Clark 78>.

Au-delà de la configuration matérielle du réseau, interviennent la structure de contrôle et les protocoles de communication mis en place aux différents niveaux de l'architecture. Le réseau local Ethernet et l'anneau de Cambridge sont deux exemples complémentaires: Ethernet est un réseau en bus à contrôle réparti, l'anneau de Cambridge est un réseau en boucle où chaque noeud intervient dans la circulation de l'information.

1.1 Ethernet

Ethernet est un système de communication par diffusion à architecture en bus. Il est destiné à transporter des paquets de données binaires entre des stations localement réparties. Le médium de communication partagé par les stations, l'Ether, est un moyen de communication passif sans contrôle central. Les stations rivales accèdent à l'Ether par un arbitrage contrôlé statistiquement. Un paquet est diffusé sur l'Ether, entendu par toutes les stations et recopié au vol par toutes les stations dont l'adresse correspond à l'adresse du destinataire figurant en tête du paquet.

Le fonctionnement détaillé du système de communication est décrit au paragraphe 3 de ce chapitre (le réseau local DANUBE). Pour les niveaux supérieurs de l'architecture (selon la norme ISO d'interconnexion des systèmes ouverts <Zimmermann 80>) un protocole de transfert de fichiers (EFTP) a été initialement fourni. Il permet le transfert de fichiers entre deux machines du réseau.

A un niveau supérieur, des outils de transfert de fichiers plus performants ont été mis en oeuvre <Shoch 78>. Plus récemment, un serveur de fichiers réseau a été conçu pour servir de support aux recherches en matière de bases de données réparties <Mitchell 82a>.

Par rapport aux travaux menés sur le réseau Ethernet, les services que nous avons étudiés, et les outils mis en oeuvre, se situent à mi-chemin de ces deux dernières réalisations. En effet, nous avons laissé de côté l'étude des outils d'accès aux fichiers ainsi que le sujet connexe de l'accès aux bases de données réparties.

Mais, d'un autre côté, les services que nous proposons représentent plus qu'un simple service de transfert de fichiers sur le réseau. Leur ensemble permet la gestion de catalogues, l'échange et le stockage de fichiers, ainsi que leur impression.

1.2 L'anneau de Cambridge

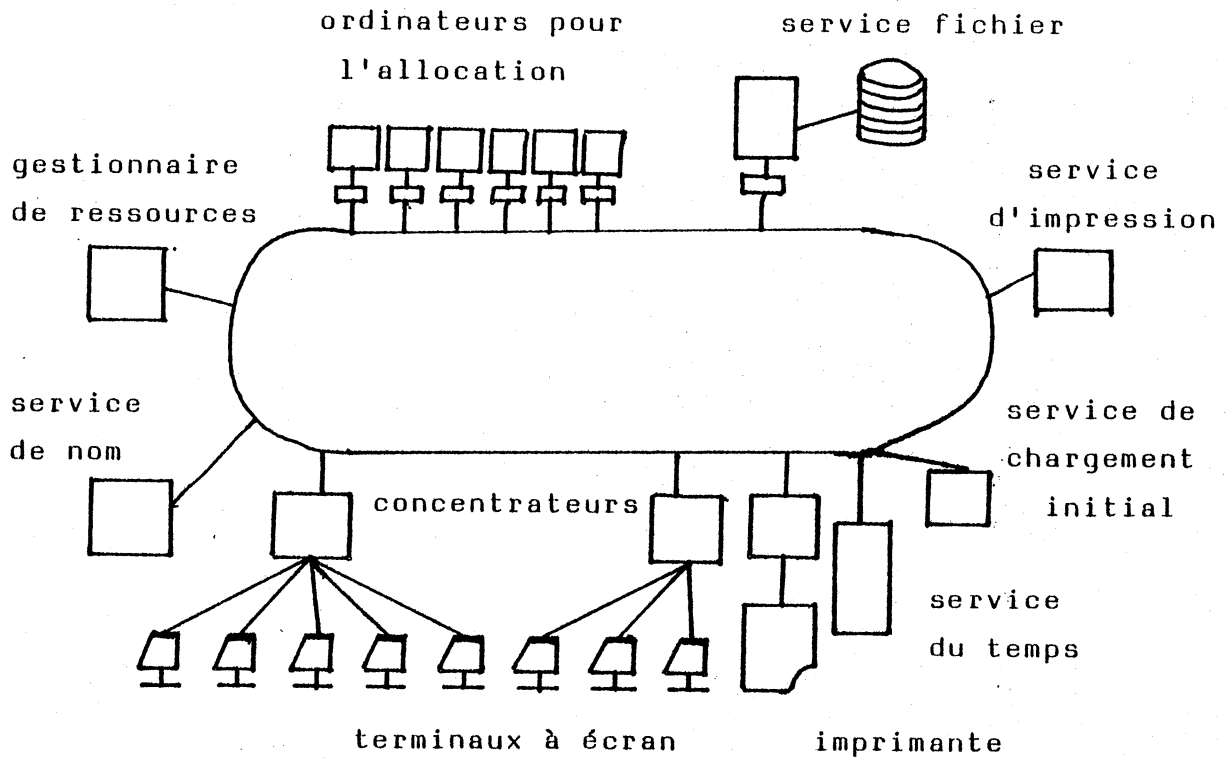
Sur le réseau en anneau de l'Université de Cambridge, des mini- et micro-ordinateurs sont interconnectés au moyen d'une boucle de communication digitale. Les machines sont reliées par une boîte d'accès formée autour d'un microprocesseur réalisant une interface duplex avec accès direct mémoire <Wilkes 80>. Le mode de contrôle choisi

initialement était l'insertion de registre <Wilkes 75>, mais le réseau fonctionne actuellement avec celui dit de la "tranche vide" (empty slot). Il existe un troisième mode de contrôle des réseaux en anneau, fondé sur le passage d'un jeton donnant le droit d'émettre à la station qui le possède. Ce dernier mode est décrit dans le réseau TARO à l'INRIA <Naffah 78>.

Matériellement, la communication est effectuée à l'aide de minipaquets de deux octets de données. Un protocole de communication de paquets est implanté dans les boîtes d'accès: il permet la transmission de paquets de 2k-octets maximum; en cas d'erreur, la retransmission automatique est assurée.

Sur ce réseau, le système distribué selon le modèle de Cambridge (CMDS <Herbert 81>) réalise les fonctions d'un système multi-utilisateurs en temps partagé. Un utilisateur se connecte au système à partir d'un terminal à écran et, au lieu d'un accès partagé aux capacités d'un système central, le système CMDS lui alloue l'un de ses mini-ordinateurs, s'il s'en trouve un de libre. L'utilisateur a alors l'usage exclusif de ce mini-ordinateur ainsi que de différents services fournis par CMDS : service de noms, service de fichiers, service imprimante et service de chargement initial de programme.

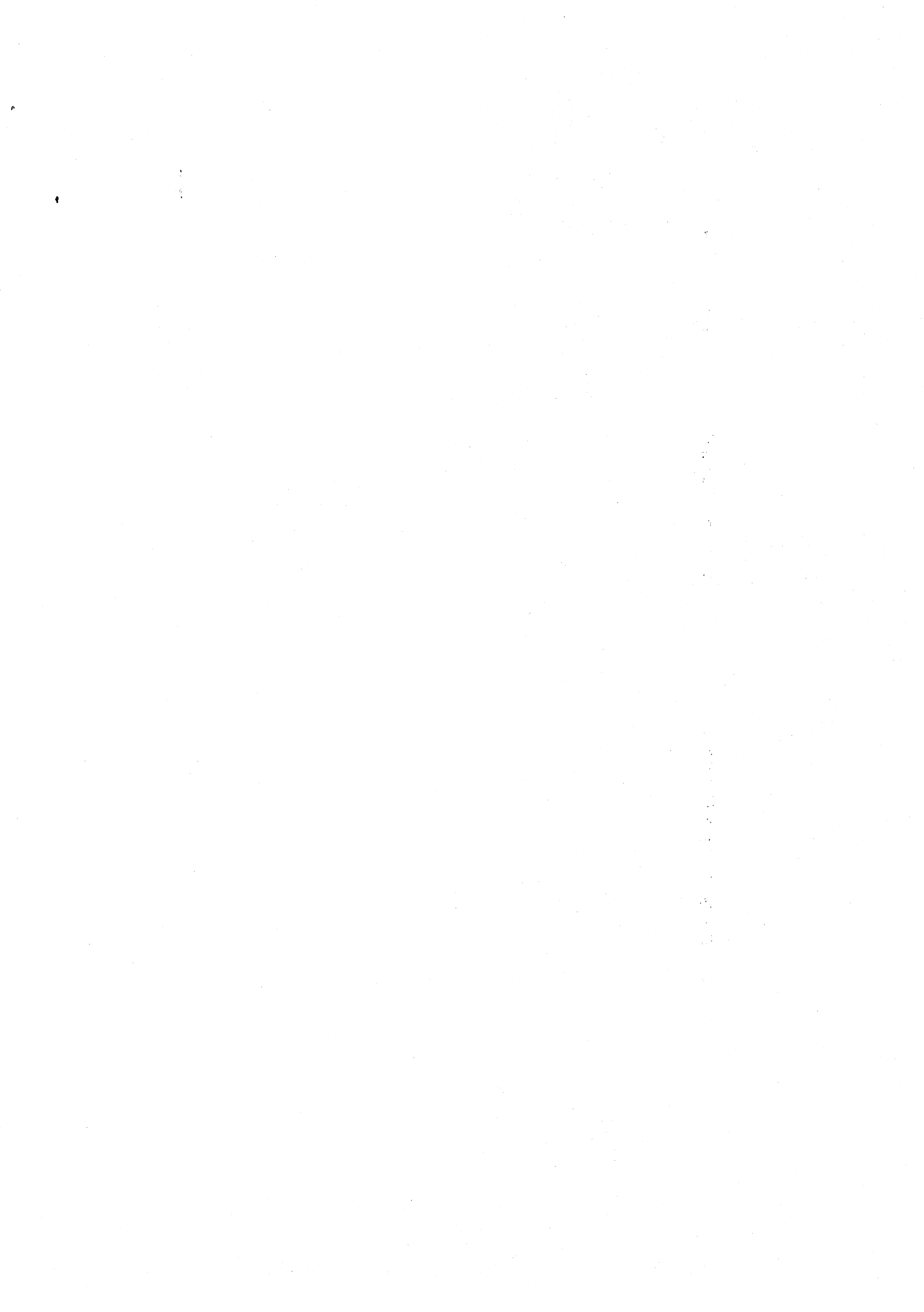
La configuration du système est la suivante:



Les terminaux sont connectés à la boucle de communication à travers des concentrateurs qui utilisent un protocole de flot d'octets pour communiquer avec le système.

La banque d'ordinateurs est composée de 6 LSI-4 de Computer Automation de 64 K-octets de mémoire chacun, cette banque peut être agrandie au fur et à mesure des besoins. Un gestionnaire de ressources permet aux utilisateurs de se connecter au système et leur alloue un ordinateur. Il contrôle les listes d'acceptation de chaque station du réseau (concentrateur, ordinateur); ces listes permettent de filtrer les paquets arrivant dans chaque noeud et de rejeter ceux qui ne leur sont pas destinés. Enfin le gestionnaire de ressources contrôle l'accès par les utilisateurs aux services du système.

CNDS est un essai de réalisation d'un système distribué de ressources à contrôle centralisé sur un réseau local. Il a été constitué à partir de matériels et de systèmes, non prévus pour leur interconnexion, grâce à une bonne conception des boîtes d'accès au réseau.



2 NOUVEAUX AVANTAGES ET PROBLEMES POSES

L'évolution technologique et la chute des coûts ont entraîné la multiplication des matériels à l'intérieur d'un même bâtiment ou groupe de bâtiments. Dans la diversité des matériels apparus, on remarque les postes de travail individuels. Leur coût est comparable à celui du mobilier qui les entoure; la possibilité de les acheter en dehors du budget informatique traditionnel a conduit à leur prolifération dans les grandes organisations. Il est apparu aussi des machines de moyenne puissance permettant d'offrir à un cercle réduit d'utilisateurs les services proposés jadis par les gros systèmes. Les utilisateurs de ces machines individuelles ont perçu le besoin de les faire communiquer pour accéder à des ressources restées rares, car chères, ou pour partager des équipements communs. Les systèmes ainsi constitués ont pour but une autre utilisation que les systèmes informatiques traditionnels; le développement de logiciel ou d'application est délaissé au profit des services de la bureautique <Sweetman 81>.

Grâce aux supports de communication, et aux communicateurs permettant l'accès à ces supports, tout matériel peut être connecté à tout autre. Mais le logiciel et ses développements ne sont pas encore aptes à tirer parti des avantages apparus. Il reste à résoudre les nouveaux problèmes qui se posent pour les exploiter.

Les nouvelles possibilités en matière de communication plus fiable et moins coûteuse permettent la mise en commun des ressources et le partage du traitement. La connexion des équipements entre eux conduit à une utilisation des services de l'ensemble à partir d'ordinateurs de faible puissance de traitement. La réalisation d'un service, demandé par un usager sur un site, nécessite des ressources

sur différents sites, et donc des traitements sur plusieurs sites. Le partage des ressources et du traitement implique de définir la collaboration entre ces divers sites.

En général, on se trouve confronté aux mêmes problèmes que ceux posés par la gestion d'un système d'exploitation classique, mais la répartition y ajoute la gestion des communications à distance, ainsi qu'une autonomie des ressources et des services qui les gèrent.

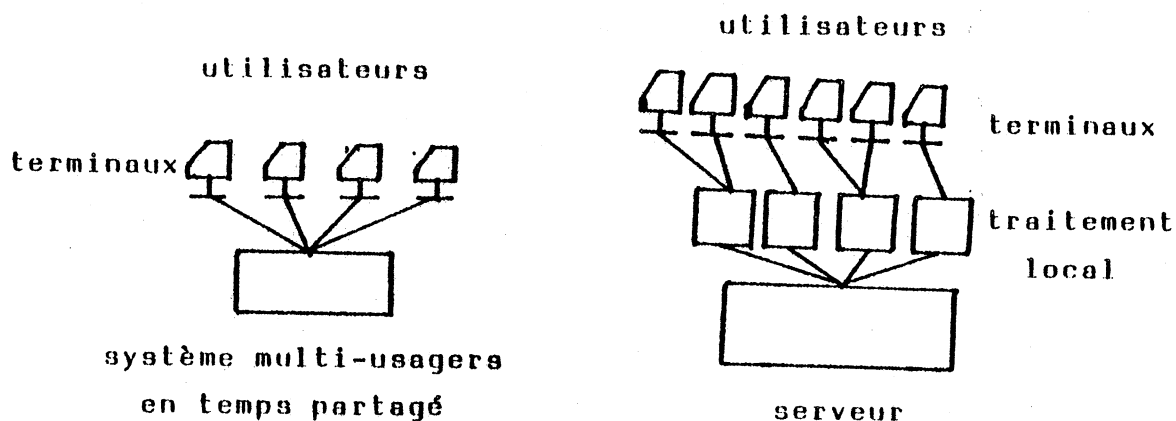
2.1 Mise en commun des ressources

Les connexions établies entre les divers équipements permettent d'envisager l'utilisation par tous de tous ces équipements. Pour répondre à la demande des usagers de l'ensemble du système, il suffit de disposer au total d'assez de ressources pour satisfaire à l'usage normal de pointe des processeurs, des terminaux, des disques et des imprimantes. Ces ressources étant diversifiées et distinctes, elles autorisent une meilleure souplesse pour leur attribution aux usagers.

L'attribution des ressources aux usagers peut se faire à la demande et de manière dynamique <AMOEBA 82>. Cette attribution n'est pas statique car les ressources, non liées à une machine individuelle donnée, ne sont pas monopolisées par un seul usager. Ce dernier n'est plus, par ailleurs, limité par les seules ressources dont il dispose localement sur un site. Il a accès à toutes les ressources du réseau, ce qui le préserve de la dépendance de ressources centralisées ou localisées vulnérables <McLellan 80>. Pour un même investissement en équipement, la mise en commun des ressources permet une meilleure réponse aux demandes des usagers grâce à l'attribution dynamique plutôt que statique des ressources.

2.2 Partage du traitement

L'évolution technologique a permis de passer de l'utilisation d'un terminal distant d'un système multi-usagers en temps partagé, à un système où des machines individuelles font le relais entre l'utilisateur et un gros calculateur, supportant un système en temps partagé, utilisé comme serveur. Les fonctions simples, demandant une puissance raisonnable de calcul, sont réalisées localement; le reste est transmis à distance pour y être traité par un plus gros serveur.



Evolution du partage du traitement

On peut envisager un travail en parallèle sur plusieurs sites, les processeurs nécessaires étant demandés puis rendus par l'utilisateur; l'ensemble des processeurs peut être géré comme un pool commun où chacun prend, utilise et rend ce dont il a besoin.

Cette notion de banque de processeurs, et des services associés, est mise en oeuvre dans le système distribué CMDS à Cambridge (Herbert 81). Elle fournit une alternative au modèle de l'ordinateur personnel dans l'informatique

répartie. Le mécanisme décrit permet l'allocation et la gestion d'ordinateurs de façon similaire à l'allocation des sessions en ligne sur les systèmes en temps partagé. Il se révèle très approprié quand on ne peut pas allouer un ordinateur à chaque usager, ou quand les machines disponibles sont hétérogènes et doivent être accessibles indépendamment de leur localisation grâce à une politique globale de gestion. Les serveurs qui gèrent le partage des ressources mises en commun peuvent être liés à un matériel donné (unité de disque, imprimante ou processeur) ou se trouver n'importe où et avoir pour fonction la gestion des noms, la vérification de droits d'accès ou l'allocation de ressources.

2.3 Communications plus fiables et moins coûteuses

Les supports de transmission sont simples, robustes et peu coûteux si on les compare à ceux des grands réseaux; ils permettent d'assurer une grande vitesse de transfert et un faible délai de propagation. Le moyen d'accès à ce support est aussi moins coûteux. Il autorise la détection rapide des défaillances grâce à des méthodes de contrôle simples, et le remplacement aisé des éléments défaillants à cause de leur faible coût.

Le coût même des communications est réduit pour deux raisons. D'une part, l'accès au support de communication est gratuit, et celles-ci ne sont pas taxées. D'autre part, le volume global des communications peut se trouver réduit grâce à un traitement local plus important avant la soumission du traitement à distance; ceci permet, en outre, de concentrer la communication dans le temps <Manning 76>.

Les communications sur un réseau local étant plus rapides et plus fiables, elles requièrent l'emploi de

protocoles plus simples. En effet, au lieu d'acquitter chaque parcelle de donnée transmise, on peut se contenter de vérifier la bonne transmission de l'ensemble, et de tout retransmettre en cas d'erreur. On aboutit alors à de meilleures performances pour des transferts en masse <Mitchell 82a>.

L'évolution technologique a fait passer de protocoles complexes, mis en oeuvre sur des machines puissantes reliées par des supports peu sûrs, à des protocoles plus simples grâce à des liaisons plus fiables, mais entre des machines moins puissantes.

2.4 Ordinateurs à capacité plus faible

Les ordinateurs individuels, apparus récemment, permettent l'accès aux services sur un réseau local; ils sont caractérisés par leur faible puissance relative et leur maigre capacité de stockage en mémoire secondaire. Bien qu'un usager puisse se contenter de cette faible puissance pour un travail courant, et faire appel à un plus gros serveur si nécessaire, il peut avoir besoin d'une assez grande capacité de stockage. Ceci est d'autant plus vrai que cet ordinateur peut être partagé entre plusieurs usagers et, de plus, ne pas disposer de disques amovibles pour constituer cette mémoire secondaire. Un dispositif du réseau qui étend la capacité locale de stockage est alors nécessaire <Satyanarayanan 81>.

Un ou plusieurs serveurs de fichiers peuvent assurer l'extension des capacités locales de stockage. Ce service lui-même pourra être étendu sur un réseau local de manière plus économique. En cas de saturation de l'espace de stockage, on peut ajouter au serveur de petites unités de stockage au lieu de le remplacer par une plus grosse

configuration. Le serveur peut aussi arriver à saturer son unité de traitement quand le nombre de ses clients augmente. On doit donc ajouter de la puissance de traitement; dans un système réparti, il est possible de créer un nouveau serveur lorsque la puissance de traitement des précédents ne leur permet plus de gérer d'avantage d'unités de stockage.

2.5 Réalisation de services par collaboration entre sites

L'interconnexion des différents matériels va fournir à l'utilisateur un ensemble de moyens disjoints qui sera la somme des services offerts par chacun de ces équipements. Pour coordonner et gérer le partage des moyens de même nature, offerts par les différents équipements, on doit y ajouter une part de logiciel qui contrôle localement l'usage d'un équipement. L'ensemble de ces logiciels, fonctionnant avec autonomie et une certaine collaboration entre eux, constituera, à l'échelle du réseau, un nouveau moyen de même nature que les moyens dont il est constitué.

Selon la nature des systèmes, prévus pour fonctionner sur un ordinateur autonome, il sera plus ou moins aisé de généraliser un service existant vers un contexte comportant plusieurs ordinateurs reliés par un réseau. En outre, il faut tenir compte de l'hétérogénéité éventuelle des systèmes interconnectés; ce qui peut amener des compatibilités diverses entre les services à mettre en commun. Ces diverses dispositions peuvent conduire à conserver, ou à tenter d'éliminer, la distinction entre ressources locales, disponibles sur l'installation de l'utilisateur, et ressources distantes, disponibles sur une autre installation <Wupit 81>.

3 LE RESEAU LOCAL DANUBE

Depuis la fin de 1980, le laboratoire IMAG dispose d'un réseau local qui relie entre eux différents équipements informatiques. Deux objectifs ont conduit à la mise en place de ce réseau : d'une part, fournir un terrain d'expérience aux chercheurs travaillant dans le domaine de l'informatique répartie, d'autre part, offrir aux différents utilisateurs des matériels existants des moyens d'accès souples et efficaces et les outils d'échange nécessaires entre les machines : échange de programmes, de fichiers, de messages, partage de terminaux.

Outre la mise en place de l'outil de communication proprement dit, qui représente un investissement important en temps et en matériel, ces deux objectifs nécessitent des développements de logiciels spécifiques. L'ensemble des services de base offerts par ces logiciels peut être comparé au système d'exploitation d'un ordinateur. En effet, de même qu'un ordinateur sans système n'offre ni service ni moyen de développement, de même un réseau sans ces services de base n'est d'aucune utilité aux usagers des machines connectées et ne permet pas le développement d'applications réparties, même si chacune des machines est dotée de son propre système d'exploitation.

Nous présentons le réseau lui-même, c'est à dire l'outil de communication, les matériels qui lui sont raccordés et sa configuration géographique, puis nous décrivons le service de base pour la communication : le service de transport.

3.1 Le réseau local

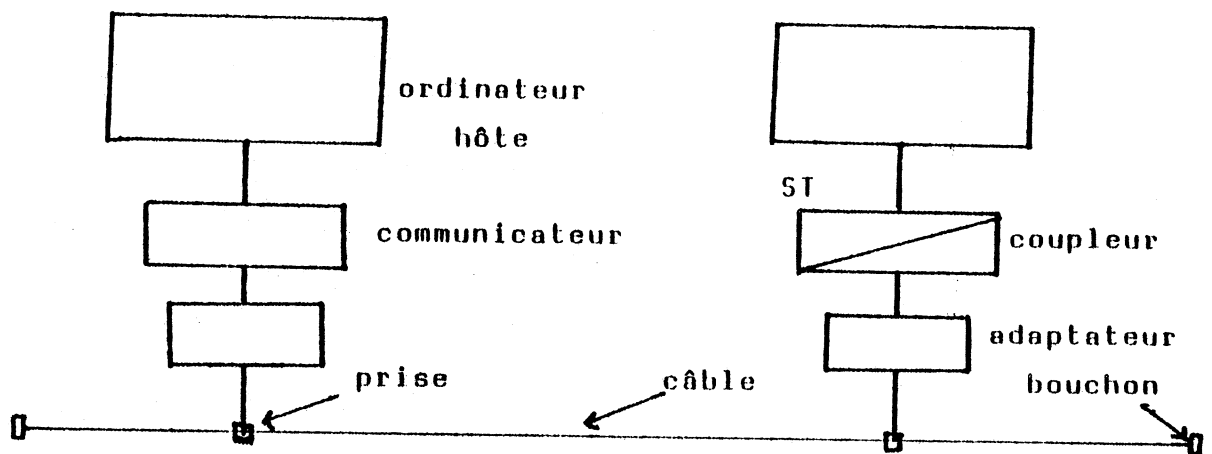
L'étude des moyens de communication répondant aux besoins et aux objectifs de recherche du laboratoire (fiabilité, évolutivité, répartition géographique, hétérogénéité, performance) a conduit à opter pour le réseau local DANUBE développé dans le cadre du projet pilote KAYAK. Ce réseau local à diffusion se distingue par sa souplesse et ses performances.

La souplesse vient de la variété des matériels que l'on peut raccorder et des modes de connexion offerts, mais aussi des possibilités d'évolution du réseau. Le nombre de stations connectables est très élevé (de l'ordre de la centaine). L'adjonction de nouvelles stations au réseau peut se faire en un point quelconque et sans même interrompre son fonctionnement.

Grâce à un débit global de l'ordre d'un Mégabit/seconde, le réseau peut écouler des trafics très importants entre les stations, avec des délais de transmission très courts. De plus son comportement sous forte charge est satisfaisant <Labétouille 80>.

3.1.1 Description du réseau

Actuellement, le réseau local de l'IMAG est constitué autour d'un câble coaxial d'une longueur de 1 km qui parcourt les différents étages de trois bâtiments du laboratoire. Sur ce câble sont branchés des communicateurs qui participent au contrôle du câble et offrent aux équipements qui leur sont connectés (terminaux ou ordinateurs) l'accès au réseau.



Constitution du réseau

3.1.2 Principes de fonctionnement

Les principes de fonctionnement de DANUBE sont ceux de ETHERNET <Boggs 76>. Le contrôle est assuré par l'ensemble des communicateurs, et uniquement par eux. Tous ont exactement le même rôle, et aucun n'est indispensable au fonctionnement de l'ensemble : le contrôle est entièrement réparti.

Le câble est donc le support de transmission partagé par tous les équipements, qui communiquent entre eux par des paquets de taille variable mais limitée à 256 octets. Le support étant unique, à un instant donné un seul paquet peut circuler sur le câble. Aussi, avant d'émettre, chaque communicateur observe l'état du câble et ne déclenche l'émission que si le câble est libre. Si le câble est occupé, un communicateur désirant émettre attend la fin de la transmission en cours. Cette précaution n'élimine pas tous les risques de collision : deux communicateurs (ou plus) peuvent trouver le câble libre au même instant et donc décider d'émettre simultanément. Les paquets se

brouillent alors mutuellement. Ces collisions sont détectées par comparaison du signal émis et du signal reçu (les communicateurs écoutent le câble quand ils émettent). Dès que la collision est détectée, la transmission est abandonnée et une nouvelle tentative d'émission de ce paquet sera faite après un délai aléatoire, donc différent pour chacun des communicateurs impliqués dans la collision.

Chaque communicateur possède une adresse unique sur le câble, c'est son adresse physique. Chaque paquet porte en tête l'adresse du communicateur origine et l'adresse du communicateur destinataire. Les paquets émis sur le câble passent devant tous les communicateurs, et celui dont l'adresse est identique à l'adresse de destination du paquet le recopie au vol dans une mémoire tampon. En plus de cette adresse unique, tous les communicateurs reconnaissent une adresse commune, dite adresse de diffusion. Ainsi, il est possible d'envoyer un paquet à tous les communicateurs actifs du réseau.

Pour les paquets ainsi diffusés, le communicateur permet un deuxième niveau d'adressage en autorisant la déclaration par l'utilisateur externe de noms constitués de chaînes de un à dix caractères, c'est l'adresse logique. Chaque communicateur peut ainsi opérer, à travers la table des noms déclarés, une sélection sur le ou les récepteurs d'une information. Un paquet transmis en diffusion mais portant le nom d'un récepteur déterminé sera lu par tous les communicateurs, mais seul celui qui possède ce nom dans sa table le remettra à la machine connectée: cette technique permet de faire de la localisation dynamique d'utilisateur sans connaître son site d'implantation.

3.1.3 Réalisation

Le communicateur est branché sur le câble par l'intermédiaire d'une prise de dérivation et d'un adaptateur. La prise, qui peut être posée sans couper le câble, permet de raccorder l'adaptateur au câble. L'adaptateur est chargé de l'émission et de la réception des signaux, de la détection d'occupation du câble et de la détection de collision.

Le communicateur proprement dit est formé de deux entités différentes. L'une, dite coupleur, identique pour tous les communicateurs réalise le contrôle d'accès au câble décrit précédemment. L'autre, dite boîte intelligente (BI), est chargée de présenter une interface convenable à l'équipement à connecter et peut donc différer (matériel ou logiciel) suivant l'équipement raccordé. Dans la première phase de développement du réseau une seule réalisation matérielle de la BI est utilisée : une carte microprocesseur (Z80, mémoire vive et morte, porte V24). D'autres sont envisageables pour l'avenir: carte avec interface parallèle ou accès direct mémoire.

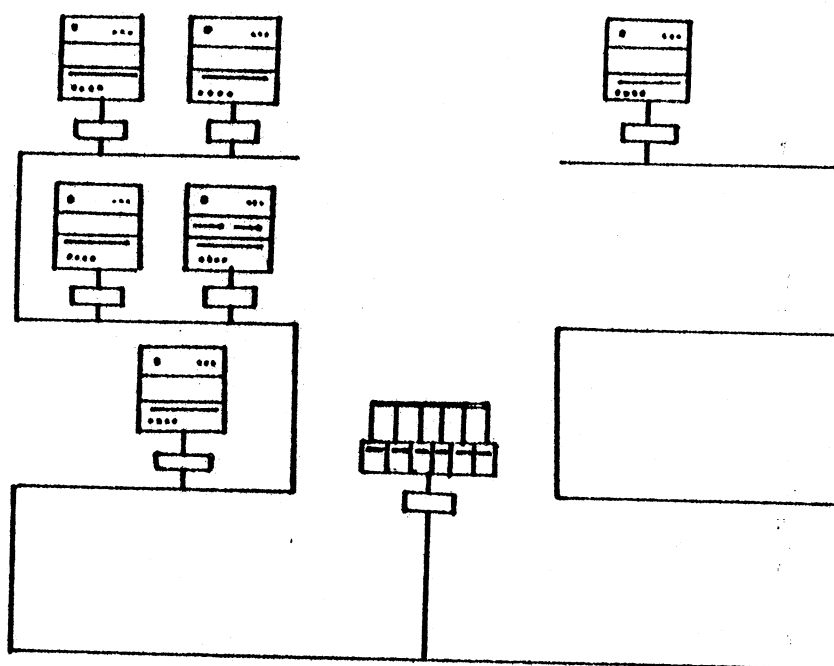
Les conséquences de la réalisation de la BI sur le fonctionnement du réseau sont importantes : toutes les communications passant par le réseau doivent emprunter la porte V24 entre la machine et le communicateur. Leur débit est donc limité aux vitesses maximales de transmission avec ce type d'interface, c'est à dire au plus quelques dizaines de milliers de bits par seconde. Ce débit réel est très faible par rapport au débit maximum du câble (1

Mégabit/seconde). Les services dont la mise en oeuvre nécessite des échanges de grands volumes d'informations souffriront donc d'un ralentissement notable par rapport aux durées théoriques calculées d'après le débit de 1 Mégabit par seconde. C'est le cas par exemple du transfert de fichiers.

3.2 Configuration du réseau

Les possibilités d'extension du nombre de stations et le problème de distance au câble obligent à faire parcourir au câble des trajets très sinueux à travers couloirs et bureaux. En contre partie, l'installation en double plafond permet à l'heure actuelle la connexion d'une nouvelle station depuis n'importe quel bureau des bâtiments traversés.

Le réseau relie sept machines. L'une est connectée par un commutateur comportant sa propre station de transport et émulant un terminal: le système MULTICS du HB-68 est ainsi accessible depuis le réseau. Les six autres machines contiennent leur propre station de transport. Il s'agit de trois LSI/11-2 et de trois LSI/11-23. Ce sont ces machines qui sont plus particulièrement utilisées pour le développement de maquettes de systèmes et d'applications réparties.



Configuration du Réseau

3.3 Le service de transport

Le service de communication offert par l'ensemble des coupleurs est insuffisant pour la plupart des applications: des paquets peuvent être perdus. En effet, les mémoires tampons où les coupleurs recopient les paquets reçus sont en nombre limité, les paquets qui arrivent lorsque toutes ces mémoires sont pleines sont perdus. D'autre part, un contrôle de redondance circulaire placé à la fin du paquet permet au coupleur récepteur de détecter les erreurs de transmission. En cas d'erreur, le coupleur ignore simplement le paquet reçu.

Le rôle du niveau transport est de fournir aux applications un service fiable qui corresponde mieux à leurs besoins. Des programmes, des processus, des tâches

peuvent être mis en communication à travers le réseau et plusieurs de ces entités, appelées abonnés, peuvent profiter simultanément et indépendamment de ce service.

Ce service est réalisé par un ensemble de stations de transport, une pour chaque site du réseau, qui communiquent entre elles suivant un protocole <Martin 80> et qui utilisent le service de communication offert par les coupleurs. Les entités désirant communiquer se déclarent à leur station de transport avec un ou plusieurs noms et désignent les autres entités par leur nom indépendamment du site où elles se trouvent.

Deux modes de transport sont disponibles: le "lettergramme" et la liaison. Le "lettergramme" est un moyen d'accès au réseau sans aucun contrôle supplémentaire. Un abonné peut envoyer des messages de taille variable mais limitée vers un ou plusieurs correspondants. L'arrivée du "lettergramme" n'est pas assurée et les abonnés correspondants doivent avoir une convention entre eux pour détecter les pertes et les récupérer.

La liaison est un mode de communication point à point bidirectionnel avec contrôles d'erreur et de flux. Deux abonnés peuvent établir entre eux un lien virtuel (une liaison), à travers lequel ils peuvent échanger, simultanément dans les deux sens, des lettres de taille variable. Le mécanisme de contrôle d'erreur permet d'assurer que toute lettre parvient au correspondant et le mécanisme de contrôle de flux limite le débit de l'émetteur aux capacités de réception du récepteur. En plus des lettres, les abonnés peuvent échanger des télégrammes, messages courts et prioritaires non soumis au contrôle de flux et d'erreur.

L'établissement d'une liaison utilise la propriété de localisation dynamique des entités concernées. Une fois la liaison établie, les stations de transport impliquées se désignent mutuellement par l'adresse de leur communicateur. Les paquets échangés sont alors directement adressés au communicateur de la station interlocutrice, évitant ainsi une diffusion inutile dans ce cas.



III

GESTION DES INFORMATIONS

DANS UN SYSTEME REPARTI :

GENERALITES

Ce chapitre présente une définition des informations sur un réseau local, introduit les services à offrir pour la manipulation des données et expose les difficultés que l'on peut rencontrer.

Les services à offrir sont ceux qui permettent la gestion et la communication d'informations : stockage et gestion, échange, accès et impression.

Les difficultés sont semblables à celles rencontrées sur des systèmes multi-utilisateurs en temps partagé : désignation et localisation, protection, cohérence et fiabilité. Sur les réseaux, il s'y ajoute l'hétérogénéité des matériels connectés. Sur les réseaux locaux, les solutions sont différentes à cause de la décentralisation des ressources et des unités de traitement.

La communication fait intervenir un service de transport assuré sur l'ensemble du réseau. Il permet le dialogue entre les entités à travers des portes de communication. Les entités peuvent être classées en serveurs et clients : les serveurs fournissent les services à travers leurs portes de communication, les clients utilisent ces services au bénéfice de l'utilisateur final : l'usager humain. La réalisation de ces services fait appel à l'exécution répartie qui consiste pour le client à appeler une fonction distante chez le serveur et à en attendre la réponse.

1 LES FICHIERS SUR UN RESEAU LOCAL

Nous donnons ici une définition des fichiers dans le cadre des réseaux locaux. Puis nous précisons les moyens de les conserver et de les représenter afin de les échanger et d'y accéder dans le cadre d'une gestion répartie.

1.1 Nature des fichiers

On peut considérer un fichier comme le contenant des informations de toutes natures qui peuvent circuler sur un réseau local. On doit envisager la possibilité de conserver ou de véhiculer sous la forme d'un fichier des documents textuels, de type simili-graphique ou graphique, de la voix, ou des données et des programmes de l'informatique classique. Quelle que puisse être la nature des informations qu'il contient, le fichier est aussi le moyen de conserver une information sous une forme qui en permet l'exploitation informatique.

1.2 Généralités sur les fichiers

Les caractéristiques des fichiers peuvent être considérées selon deux points de vue : d'une part pour leur conservation statique, d'autre part en vue d'un transfert ou d'un accès à distance.

1.2.1 Conservation et représentation des fichiers

Dans cette première optique, les fichiers sont caractérisés par l'organisation logique ou physique de leur

contenu, les attributs qui leurs sont attachés, la signification que l'on peut apporter à leur contenu et la façon dont on les répertorie.

1.2.1.1 Organisation logique

Une façon simple de concevoir un fichier, donc un ensemble de données, est de le considérer comme une suite d'octets, numérotés de zéro jusqu'à la longueur du fichier moins un, en offrant des opérations qui permettent de lire ou d'écrire tout octet arbitrairement choisi <AMOEBA 82>. C'est par exemple le modèle adopté dans le système UNIX <Ritchie 74>. Certains systèmes associent à cette notion de séquence d'octets une division en pages de taille fixe. D'autres présentent une structure d'arbre dont les feuilles sont des groupes d'octets, appelés articles, de taille variable <Mitchell 82a>. Les opérations offertes permettent la modification d'un article, son insertion ou sa suppression n'importe où dans l'arbre. Eventuellement, une clé associée à l'article peut en faciliter l'accès.

1.2.1.2 Organisation physique

Les différentes organisations logiques présentées sont réalisées grâce à une représentation physique des données. L'espace de stockage est généralement divisé en pages de taille fixe accessibles individuellement sur l'unité de stockage. L'organisation logique que voit un utilisateur peut, parfois, faire transparaître l'organisation physique sous-jacente des données du fichier. C'est la finalité de cette organisation qui déterminera cette transparence, selon que le système de gestion de l'espace est destiné à un usage immédiat, ou ne sert qu'à bâtir un système plus spécifique.

1.2.1.3 Attributs

Associée à chaque fichier, on trouve le plus souvent une petite masse de données supplémentaires qui ne fait pas partie du fichier lui-même. Cette information est en général distincte du contenu propre du fichier; elle prend la forme d'attributs qui caractérisent le fichier, ou de données qui permettent de le reconstituer si nécessaire. Les attributs que l'on trouve couramment indiquent la longueur du fichier, sa date de création, la date de sa dernière modification ou de sa dernière lecture, ils contiennent les informations de protection ou de vérification des droits d'accès. Une partie de l'information supplémentaire permet aux gestionnaires de catalogues de faire face à des détériorations ou des pertes de fichiers ou de catalogues.

Les attributs peuvent avoir une portée sur la signification des informations ou leur organisation dans le fichier. La sémantique des données constituant un fichier peut apparaître dans leur organisation logique ou dans les attributs qui leur sont associés. Ces informations ne peuvent être interprétées que par les entités qui créent ou manipulent un fichier.

1.2.1.4 Catalogues

Une partie de l'information attachée aux fichiers, et définie comme attribut, peut se trouver dans le catalogue des fichiers. Le catalogue regroupe les informations qui concernent un certain nombre de fichiers, et assure la correspondance entre leur désignation connue par les usagers et leur localisation dans l'espace de stockage. Les informations sur un fichier du catalogue constituent son

descripteur. Le catalogue contient le nom du fichier, l'indication de son propriétaire, les droits d'accès pour tout utilisateur éventuel et la description de l'emplacement de son contenu.

1.2.2 Transfert et accès distant

Dans cette seconde optique, l'accent est mis sur la simplicité de l'organisation logique et physique du contenu, et la finesse de l'accès aux données.

1.2.2.1 Simplicité de l'organisation logique

L'organisation logique des données dans un fichier peut être très élaborée en vue d'une utilisation directe des services qu'elle propose. A l'opposé, elle peut se contenter d'offrir des moyens élémentaires de gestion d'un espace de stockage, laissant aux utilisateurs le soin et la liberté de gérer cet espace, et de donner une signification à son contenu.

Dans la perspective des services de gestion des fichiers sur un réseau local, cela peut permettre à des agents extérieurs ou distants de gérer à leur guise un espace qui leur appartient en partie, et dont ils sont responsables de l'occupation et de l'organisation.

1.2.2.2 Finesse de l'accès aux données

Cette préoccupation majeure dans les systèmes classiques de gestion de fichiers conserve toute son importance dans le cas d'un accès demandant une

communication plus ou moins coûteuse. Elle définit la taille de la plus petite unité d'information que l'on peut manipuler au sein d'un fichier. Elle découle de l'organisation logique des données; on peut, selon les cas, lire, écrire ou modifier un octet, un enregistrement ou une page de données en une seule action <Satyanarayanan 81>. L'efficacité de ces actions dépend de l'organisation physique des données et de l'usage qu'en fait leur utilisateur.

1.3 Gestion répartie des fichiers

L'évolution qui a eu lieu après l'interconnexion de matériels par un réseau local a fait passer de systèmes autonomes reliés entre eux à un système généralisé à l'échelle du réseau. Les systèmes existant sur les ordinateurs connectés ont pu communiquer en gardant la gestion autonome de leurs informations. Puis ont été ajoutés des serveurs supplémentaires partagés entre les sites du réseau. On aboutit à une banalisation de l'ensemble des ressources, et à leur gestion globale sur le réseau.

Nous allons présenter les étapes de cette évolution en ce qui concerne les systèmes de gestion de fichiers. Nous irons de l'interconnexion de systèmes autonomes vers le concept de systèmes étendus à tout un réseau, en passant par l'adjonction de serveurs de fichiers.

1.3.1 Systèmes de gestion de fichiers interconnectés

Sur ce modèle de systèmes autonomes interconnectés, un usager au terminal d'un ordinateur du réseau a accès à tous

les services fournis par cet ordinateur, en pouvant faire appel, occasionnellement, à des ressources situées sur d'autres installations du réseau <Wupit 81>. La répartition des équipements et des ressources est alors apparente pour cet utilisateur. En effet, l'accès à un fichier distant demande de sa part des précisions explicites sur la localisation du fichier, et l'emploi de services supplémentaires qui gèrent les accès hors du système local.

La gestion des fichiers est réalisée en conservant le stockage local des fichiers, et en faisant des accès ou des échanges avec des sites distants sur le réseau. Néanmoins, il peut s'avérer nécessaire d'augmenter la capacité locale de stockage en l'étendant à l'extérieur d'un site. Ce service fourni grâce au réseau ne doit apparaître que comme une prolongation de l'espace de stockage du service local de gestion de fichiers <Satyanarayanan 81>. Ce service extérieur doit donc offrir les mêmes possibilités que le service local pour ne pas présenter de différences sensibles de fonctionnement. Mais en outre, il doit limiter l'éventail de ses services à ceux offerts localement. Le service distant doit se présenter comme une nouvelle unité de stockage ou un catalogue de plus du service local de gestion des fichiers.

Lors de l'interconnexion de matériels sur un réseau, les services de gestion de fichiers de chaque machine connectée au réseau sont mis en relation grâce aux moyens de communication du réseau. Des modifications, ou adaptations mineures peuvent être utiles pour tirer un meilleur parti des possibilités offertes par l'interconnexion. Mais, étant donnée la grande part d'utilisation locale des services préexistants, les modifications apportées aux systèmes de gestion de fichiers existants ne doivent pas modifier leur interface vers l'utilisateur. Dans un premier temps, la connexion au réseau ne

doit pas altérer l'utilisation courante des programmes qui utilisent ces services de gestion de fichiers.

1.3.2 Serveur de fichiers

Les machines connectées au réseau n'ont pas toutes un espace de stockage suffisant en mémoire secondaire. Le principe du serveur de fichiers est de fournir sur le réseau un espace de stockage utilisable par tous les ordinateurs du réseau <Sweetman 81>. Cette fonction particularise un ou plusieurs sites du réseau <Lunn 81>. Le site du serveur peut être entièrement consacré au service, ou bien permettre simultanément le travail normal d'un utilisateur. L'interface offerte à l'extérieur par le serveur peut prendre deux formes extrêmes. Une forme très élaborée permet aux utilisateurs toutes les manipulations de fichiers et de catalogues. L'autre forme, plus élémentaire, ne permet que la gestion de l'espace de stockage mis à la disposition des utilisateurs.

1.3.2.1 Serveur complet

Un serveur de fichiers qui offre aux utilisateurs tous les services possibles en matière de d'accès et de gestion de fichiers réalise les mêmes fonctions que les services de gestion de fichiers d'un système multiutilisateurs tel MULTICS <Saltzer 74> ou UNIX <Ritchie 74>. Ces fonctions consistent à gérer les fichiers de plusieurs utilisateurs, à assurer pour chacun d'eux l'accès aux informations qu'il a confiées au service, à garantir la sécurité de ses informations en contrôlant les droits de ceux qui veulent y accéder.

Les impératifs de protection de l'accès aux données gérées par le serveur peuvent limiter l'espace de stockage utilisé par le serveur, ou le travail simultané du serveur et d'autres utilisateurs sur la même machine. Seul le fonctionnement du serveur sur une machine spécialisée ou en tant qu'environnement de travail d'un utilisateur particulier au sein d'un système multi-utilisateurs en temps partagé peut assurer la sécurité complète des données gérées par le serveur. Sans cette limitation de l'accès, tout utilisateur local pourrait manipuler à l'insu du serveur les fichiers qui sont à sa charge.

Les fichiers conservés par un serveur sont disponibles pour tous les usagers du réseau. Ils ne sont accessibles, après contrôle du serveur, qu'à ceux qui possèdent les droits d'accès exigés. Cela rend possible l'utilisation d'un fichier chez le serveur même si le site qui l'y a déposé est inaccessible. Par ailleurs, la nécessité d'une certaine indépendance vis à vis du serveur impose de pouvoir continuer à travailler localement au cas où le serveur est indisponible, ou le réseau inutilisable.

D'un autre point de vue, on peut vouloir considérer le serveur de fichiers comme un dépositaire de données fiable à très long terme, accessible par tous les ordinateurs du réseau. Il peut aussi constituer un serveur d'archivage où les données stockées restent accessibles même après des défaillances destructrices. Et de plus, il peut offrir aux usagers la capacité de récupérer des erreurs humaines telles que la destruction accidentelle d'un fichier.

Il est à remarquer que de telles possibilités sont réalisables sans difficulté grâce à la technologie des disques optiques. Cette nouveauté permet de disposer d'un espace de stockage de très grande capacité où les données inscrites sont par nature ineffaçables <Satyanarayanan 81>.

1.3.2.2 Serveur élémentaire

Ce type de serveur est apparu pour permettre le contrôle à distance de la gestion et de l'occupation de l'espace de stockage. Il offre donc les primitives qui permettent l'accès aux portions de l'espace de stockage, leur utilisation et leur restitution. La structure du système de fichiers est organisée par les clients du service <FELIX 81>. Ils manipulent les éléments de cette structure qui sont désignés grâce à une numérotation proposée et gérée par le serveur. Le serveur de fichiers ignore alors l'utilisation qui est faite des unités de stockage. Certains fichiers constitués peuvent représenter des catalogues, d'autres des données, le serveur ne connaît pas les relations entre eux. L'ensemble des fichiers ne constitue qu'une collection plate non structurée du point de vue du serveur.

Le défaut de cette approche est d'entraîner la possibilité de fichiers perdus ou oubliés, donc devenus indestructibles, car celui qui les a créés ne les connaît plus. Les clients d'un tel serveur doivent être éprouvés pour détruire explicitement les fichiers qu'ils ont créés quand ils n'ont plus besoin <Mitchell 82a>. Une amélioration de ce fonctionnement consiste à faire gérer par le serveur l'allocation des unités de stockage et à vérifier leur occupation effective à l'aide d'un algorithme de ramasse-miettes <Lunn 81>.

1.3.3 Gestion globale des ressources de type fichier

L'étape suivante de l'évolution des systèmes interconnectés consiste à faire apparaître les nouveaux services de façon que les usagers aient l'impression de

travailler avec un système unique généralisé plutôt qu'un ensemble de systèmes de gestion de fichiers coopérants <LOCUS 81>. Cela rend invisible la répartition des matériels connectés sur le réseau.

Cette notion correspond au fait que l'utilisateur d'un système réparti ne voit pas la répartition lorsqu'il manipule les données sur le réseau. La gestion globale des fichiers consiste à ne plus différencier les sites ou les machines où ils se trouvent pour les désigner indépendamment de leur localisation. L'invisibilité de la répartition doit être effective tant au niveau de la localisation des services qu'au niveau de la localisation des ressources concernant les fichiers. Pour parvenir à cette impression chez l'utilisateur, il faut fournir l'apparence d'un système de gestion de fichiers sous-jacent et uniforme sur un réseau de clients aux exigences diverses <McLellan 80>.

L'utilisateur doit en fait avoir une vue externe du réseau qui ne l'oblige pas à connaître les particularités de chaque système d'exploitation concourant au fonctionnement du réseau <Chambon 81>. On aboutit à un aspect banalisé de l'accès aux ressources distribuées. Un usager au terminal d'un ordinateur du réseau est considéré comme un utilisateur d'un système généralisé, dans lequel les différentes installations se confondent pour former un ensemble unifié de ressources banalisées.

Gestion répartie des fichiers

architecture logique		utilisation	
		locale	distante
SGF interconnectés		comme avant interconnexion la plupart du temps	prolongement des possibilités locales occasionnellement
serveur de fichiers	complet	consultation échange	stockage secondaire archivage
	élémentaire	plus gestion espace	espace partagé
gestion globale		invisibilité de la répartition utilisation indifférente des sites	

2 SERVICES A OFFRIR SUR LE RESEAU

Les utilisateurs ont besoin de services concernant la communication et la manipulation des informations à travers le réseau. Ils doivent pouvoir échanger des documents, des programmes et des données, et avoir la possibilité de gérer l'ensemble de ces informations. On peut concevoir ces services comme une extension au réseau de services existants localement en matière de maintenance, de stockage, d'échange, de méthode d'accès et de possibilité d'impression, ou de sortie physique, de tous les fichiers. Après avoir défini ces services locaux, on montre en quoi consiste leur extension au réseau. Ensuite, on s'intéresse à la façon d'accéder à ces services étendus, et aux caractéristiques de leurs utilisateurs.

2.1 Services existant localement

L'exemple du service de gestion des fichiers d'un système multiutilisateurs permet de présenter les services auxquels nous nous intéressons, et le contexte de leur utilisation. Dans un tel système les usagers disposent d'un espace de travail qui leur est propre; la protection de cet espace et l'accès aux ressources communes sont gérées par le système. Les informations propres d'un usager sont conservées par le système et l'usager peut y accéder à partir de tout terminal. Son déplacement d'un terminal à un autre n'a aucune importance d'un point de vue géographique, les informations de l'usager n'ont pas à être déplacées.

Parmi les services à offrir qui nous intéressent, on peut distinguer quatre classes.

2.1.1 Stockage et gestion des informations

Ces fonctions permettent le stockage de l'information de manière fiable et accessible à tous, et la gestion des catalogues de fichiers et de leur support <Manning 76>. Le fichier contenant l'information est considéré comme un tout indépendamment de la signification de son contenu. L'espace de stockage est considéré comme un ensemble indivisible géré globalement par le système.

2.1.2 Echange d'information

L'échange et le transfert d'informations entre les usagers ou entre les diverses sections de l'espace de travail d'un usager représentent un service de copie des fichiers et des documents sur le réseau.

2.1.3 Accès à l'information

C'est disposer d'une méthode d'accès aux informations pour consulter ou modifier le contenu des fichiers avec une grande finesse d'accès.

2.1.4 Sortie de l'information

Cela représente la possibilité d'imprimer les documents, c'est à dire de sortir les informations sur un support plus classique comme le papier.

2.2 Apports du réseau

En fait sur le réseau, il s'agit de procurer un ensemble de services qui sont en général déjà disponibles sur les sites qui composent le réseau. Donc, on doit envisager une extension sur le réseau des services locaux existants. Cette extension peut prendre deux formes. La première est la collaboration entre les services existants sur les sites. La seconde consiste à concentrer les services sur certains sites qui supportent divers processus système ou qui constituent l'interface de gestion de périphériques partagés <Sweetman 81>. Cette extension doit permettre le mouvement des documents à l'intérieur du réseau : leur création, leur consultation et leur modification, leur gestion au sein de catalogues, leur impression et leur destruction éventuelle.

Précisons la conception que l'on a de l'utilisateur, et, surtout, de sa mobilité. Un usager "immobile" ne changera pas de poste et ses documents personnels pourront rester sur place. Si, par contre, il est mobile on doit lui offrir la possibilité d'atteindre ses fichiers personnels à partir de différents postes de travail. Il y a trois moyens de lui assurer cette disponibilité.

1) L'usager emporte son support avec lui dans ses déplacements : c'est la solution qui prévalait avant la floraison des moyens de communication. Quand son poste de travail est en panne, il peut en changer si un autre poste accepte son support.

2) L'usager dispose en quelque sorte d'un poste d'attache qui conserve ses informations. Il peut y

travailler à distance à partir de n'importe quel autre poste. Il peut aussi l'interroger à distance, c'est à dire simplement transférer les informations. En cas de panne de ce poste d'attache, il ne peut plus travailler.

3) A la fin de chaque session de travail, l'ensemble des documents d'un usager est confié à un serveur qui les conserve. L'utilisateur qui désire travailler se fait restituer ses fichiers le temps de les modifier, puis les confie de nouveau au serveur. Cette solution se rapproche de celle offerte par le travail en temps partagé sur un système multiutilisateurs. Elle permet d'éviter la gêne causée par la panne du poste de travail, mais elle multiplie celle causée par la panne du serveur.

Parmi les services considérés, certains ont subi une évolution liée à la structure du système réseau sur lequel ils étaient réalisés, d'autres sont restés inchangés dans leur principe. On peut discerner deux phases dans l'évolution de la structure des systèmes réseaux : une étape d'interconnexion de systèmes autonomes puis un système généralisé sur le réseau. Le stockage et la gestion d'informations suivent l'évolution de structure; l'échange et l'impression de documents évoluent peu dans leur conception. L'accès au contenu des documents, toujours réalisé localement, n'évolue pas.

2.2.1 Systèmes interconnectés

Dans un premier temps, on se contente d'utiliser l'ensemble des services indépendamment les uns des autres. On travaille sur un site sans se préoccuper des autres. On travaille à partir d'un site sur les autres au moyens de commandes exécutées à distance. Or, pour utiliser les services locaux des sites distants, il faut en connaître

les usages et les particularités. L'hétérogénéité entre les matériels ne peut manquer d'apparaître lors de leur interconnexion. Elle constitue une difficulté supplémentaire de l'utilisation du réseau et de ses ressources dispersées.

2.2.1.1 Stockage et gestion

Les fonctions de stockage et de gestion sont réalisées sur chaque site du réseau qui possède un espace de stockage de l'information; elles gèrent localement cet espace et entretiennent les structures qui décrivent son occupation.

En outre, des sites peuvent se spécialiser, ou des outils apparaître, pour offrir une interface plus homogène aux utilisateurs. Les serveurs de catalogues, qui gèrent toujours un espace limité au site sur lequel ils travaillent, fournissent tous de manière similaire les fonctions qui permettent de créer ou détruire des catalogues, d'introduire ou de supprimer des noms de fichiers, d'attacher des droits à l'accès à ces informations, de vérifier, de restreindre ou de supprimer ces droits, et enfin d'entretenir l'espace de stockage dont ils disposent.

A priori, on peut penser que les moyens de communication sont inutiles pour ces services. En effet sur chaque site il existe un stockage local des documents des utilisateurs locaux. Ceux-ci confient leurs informations au service de stockage qui se charge de les conserver jusqu'à leur prochaine utilisation. Cela n'implique pas ou peu de communication. Cependant, des utilisateurs qui ne travaillent pas habituellement sur le même site peuvent ressentir le besoin d'échanger entre eux des documents stockés. Ils ont le choix entre échanger le support

(disque) des documents en le déplaçant physiquement, dans la mesure où ce support est amovible, ou utiliser un service d'échange de documents, par exemple une messagerie.

2.2.1.2 Echange

Ce service correspond sur un réseau local au service de copie disponible dans les systèmes classiques de gestion de fichiers. A cause de l'interconnexion de matériels différents, il demande la définition d'une convention commune de représentation d'un fichier, ainsi que des primitives pour le transfert de données entre les entités qui échangent ce fichier. De façon classique, on conçoit ces entités comme un serveur et un client, l'information étant détenue par le serveur et réclamée, ou fournie pour conservation, par le client.

Le serveur est supposé pouvoir accepter des documents de toutes tailles, en fonctionnant comme un dépositaire de fichiers manipulés dans leur entier <Birrell 80>. Mais le client peut être limité soit par la taille de son propre espace de stockage, soit par celle de ses tampons de réception ou d'émission. On distingue donc les transferts qui concernent un client de grande capacité qui accepte les fichiers entiers, et les transferts qui font intervenir un client aux faibles moyens qui ne peuvent accéder à l'information qu'au niveau des pages, des enregistrements ou des octets <McLellan 80>.

Dans le premier cas, il s'agit de transfert de données en masse; dans le second, il s'agit beaucoup plus de l'accès à une information distante. Néanmoins, on peut assimiler le second cas au premier dans la mesure où il demande un découpage de l'information lors du transfert.

2.2.1.3 Impression

En général, ce service est aussi fourni localement. Chaque site doit alors disposer d'une imprimante. Cela demande un investissement en matériel qui n'est pas toujours justifié par l'utilisation qui en est faite. En outre, on ne dispose pas toujours localement du matériel qui serait le plus adéquat.

Le regroupement et le partage des ressources, grâce à un serveur qui les gère, n'imposent plus la nécessité d'une imprimante par site et permettent d'offrir à chacun la disposition de plusieurs imprimantes de qualités ou de vitesses différentes. Ces ressources en général sous-employées trouvent alors une meilleure utilisation.

Les ressources en matériel d'impression peuvent être regroupées et partagées de manière physique ou seulement logique. Le regroupement physique demande que tout le matériel soit rassemblé en un même lieu, et que son attribution aux demandeurs soit gérée sur un site spécialisé. Cette concentration du matériel d'impression permet d'assurer l'isolation phonique et de faciliter l'entretien. On rejoint ici le principe des salles machines des grands systèmes en temps partagé.

En laissant le matériel dispersé sur le réseau, on peut partager logiquement son utilisation. L'attribution de chaque équipement sera décidée par la coopération des serveurs qui gèrent chacun une partie du matériel.

Le texte à imprimer, placé dans un fichier, est envoyé sur le site du serveur d'impression. En fait, on peut ne transmettre au serveur que les coordonnées du fichier, le serveur y accédera ensuite à distance. Le texte est imprimé

dès la réception de la commande ou mis en réserve pour son impression ultérieure lorsque le matériel requis sera disponible. Une fois la demande d'impression prise en compte par le serveur, ainsi que les paramètres associés, le service travaille de façon autonome, et le demandeur peut continuer son travail local.

Le serveur d'impression est composé d'une interface de gestion des requêtes (selon un certain ordre de priorité), en liaison avec un serveur de fichiers. Les commandes, qui, en général, ne peuvent pas être satisfaites immédiatement sont conservées en file d'attente. Ce service utilise le mode de travail par appel d'une fonction distante pour traiter les requêtes d'impression, et le mode batch pour traiter les travaux d'impression soumis au fur et à mesure des disponibilités du matériel.

2.2.2 Système réseau

Lorsque d'un ensemble de systèmes autonomes interconnectés on passe à un système étendu au réseau, un changement intervient sur la réalisation du service de stockage et de gestion des informations. Pour les services d'échange et d'impression de documents, le changement, moins important, n'est lié qu'à la désignation différente, uniforme et banalisée, des documents dans le système réseau.

2.2.2.1 Stockage et gestion

L'étape suivante de l'évolution est de considérer la maintenance, non plus comme composée d'un ensemble de services locaux de maintenance, mais comme la gestion de

l'ensemble de l'espace de stockage du réseau au moyen de catalogues ayant une extension sur plusieurs sites. La gestion de ces catalogues répartis pose des problèmes causés par l'hétérogénéité des matériels, ou dus à la nécessité de mise à jour et de maintien de la cohérence de catalogues ou de fichiers dupliqués pour des raisons de sécurité.

Pour partager l'accès aux documents sur une base plus large, et non plus du point de vue d'un seul utilisateur, il devient nécessaire de disposer d'un service de stockage utilisable simultanément par l'ensemble des utilisateurs du réseau. Les documents sont saisis ou modifiés localement puis confiés, sous forme de fichiers, à un service de stockage.

Ce service a évolué pour passer de la notion de stockage à celle d'archivage, c'est à dire à la conservation très fiable et de très longue durée de documents consultés très rarement. Les systèmes de gestion de fichiers étaient utilisés localement pour le stockage. Puis des outils de transfert et de manipulation de fichiers à travers le réseau entre postes de travail ont permis l'accès à un stockage plus généralisé. Ensuite sont apparus les serveurs de fichiers offrant un grand espace de stockage et la possibilité de gérer l'organisation de cet espace. Au stockage, on peut associer une fonction d'archivage, sur un espace de mémoire "tertiaire", qui débarrasse le serveur de stockage des informations qui ne sont pas demandées souvent pour consultation ou modification. A la demande, le document peut être ramené chez le serveur de stockage, ce qui peut entraîner un délai plus long de service.

2.2.2.2 Echange

Etant donnée la désignation générale ou banalisée des documents, leur échange va consister à les copier d'un catalogue dans un autre au lieu d'effectuer un transfert explicite d'un site vers un autre. Sans déplacer le fichier, on peut aussi se contenter de copier simplement sa référence ou ses coordonnées dans le nouveau catalogue auquel il appartient. Dans ce cas, les mises à jour se font automatiquement sur toutes les copies.

2.2.2.3 Impression

De même, pour ce service, c'est plus la désignation des documents à imprimer qui va changer que le principe de réalisation du service. Néanmoins, le service est conçu avec une extension globale sur le réseau. Les points de service, tout comme les documents, sont connus selon une désignation banalisée.

2.2.3 Fonction non perturbée : accès aux fichiers

Pour un client désirant travailler sur un fichier qui se trouve à distance, il y a trois possibilités : 1) le transférer sur son propre site, y travailler à loisir et le renvoyer; 2) y accéder à distance, c'est à dire travailler en déplaçant de petites quantités d'information; ou bien 3) déplacer le processus de traitement et le faire s'exécuter sur le site où se trouve le fichier. La première façon de faire est souvent la simulation de la seconde. Si le client n'a pas la possibilité de garder chez lui le fichier en entier, la deuxième façon est un moyen de l'examiner. Ce

mode d'accès à l'information par petites tranches correspond aussi aux utilisations classiques des mémoires secondaires locales : la fonction de mémoire virtuelle, les échanges de partitions de mémoires lors des commutations de processus ou les chargements de branches en recouvrement <Birrell 80>. Le troisième mode correspond au travail de maintenance à distance des catalogues; on y fait appel aux primitives distantes pour agir localement sur le site éloigné. Dans le premier et le troisième cas, il n'existe qu'un exemplaire du fichier; dans le deuxième, il y a deux copies, à moins que la copie sur place ne soit détruite, ce qui pose un problème de cohérence.

Sans interpréter le contenu des fichiers, on peut séparer trois niveaux dans leur accès. Il s'agit du fichier entier, des pages et des octets.

L'accès au fichier entier se fait de façon séquentielle. Ce mode est acceptable pour le stockage de longue durée et pour le transfert entre des machines possédant assez d'espace disque. On peut s'interroger sur la taille des unités envoyées en une fois entre client et serveur : de grosses unités échangées demandent de gros tampons pour les contenir, des unités trop petites ralentissent le transfert.

L'accès au fichier par pages correspond à l'échange d'éléments de taille fixe, souvent de la taille des unités échangées entre le serveur et le support, c'est à dire de la taille des blocs physiques sur le disque.

L'accès aux octets contenus dans le fichier permet de ne transférer qu'un certain nombre d'octets à partir du rang d'un octet donné. Offerte à un haut niveau de service cette primitive est très utile, mais elle s'avère peu efficace à réaliser à un plus bas niveau. Néanmoins elle

couvre le cas du transfert par pages, et peut être utile selon le débit du réseau et la sophistication du client.

Une méthode d'accès au contenu des fichiers doit permettre, une fois le fichier atteint et ouvert, de travailler et d'intervenir sur l'information qu'il contient. On doit pouvoir écrire ou lire un certain nombre d'octets à partir d'un point donné, se positionner en un point arbitrairement choisi et s'informer sur l'état du fichier. Il faut mettre l'accent sur le caractère répétitif des actions sur les fichiers et des commandes d'accès. Pour permettre qu'une même action ou commande puisse être répétée sans dommage, les écritures et les lectures devront se faire à partir de positions absolues plutôt que relatives. On exclut donc toute progression implicite dans le fichier, résultant d'une action ou commande, pour n'autoriser que des positionnements explicites.

Récapitulation des services
selon l'architecture logique des systèmes

Services	classique	Systemes interconnectés	réseau
Stockage Gestion	gestion des catalogues sur un espace de stockage global et partagé	gestion locale de l'espace de stockage de chaque site serveurs de fichiers	gestion de l'ensemble de l'espace de stockage du réseau catalogues répartis cohérence
Echange	copie de fichiers	transfert de site à site conventions communes client-serveur	copie de catalogue à catalogue à travers le réseau copie des coordonnées
Sortie	impression des fichiers sur du papier	regroupement et partage des imprimantes serveur d'impression	extension globale et désignation banalisée du service d'impression

2.3 Accès aux services étendus

En réduisant au minimum les exigences au sujet des services évoqués, on peut regrouper les services aux vocations similaires et proposer :

- pour la maintenance et l'accès aux fichiers : la possibilité de travailler sur les catalogues ou les fichiers d'un site à partir de tout site,

- pour le stockage et l'impression : la possibilité de ranger, de retrouver et de faire imprimer un fichier à partir d'un site quelconque; donc les serveurs de fichiers ou d'impression doivent être accessibles à partir de n'importe quel site,

- pour le transfert : la possibilité d'échanger des fichiers entre deux sites quelconques.

2.3.1 Maintenance et accès

Pour bénéficier au maximum des avantages du réseau, en matière d'indépendance de ressources centralisées vulnérables, tout fichier du réseau doit pouvoir être accessible uniformément à partir de tout point du réseau. La dispersion des fichiers les met à l'abri d'une panne qui les rendrait tous indisponibles; leur utilisation est aussi aisée car ils restent toujours accessibles. Néanmoins, on a le choix d'offrir cet accès soit à travers un système généralisé soit à travers un ensemble de systèmes autonomes interconnectés.

La gestion des catalogues des fichiers d'un site doit pouvoir se faire à partir d'un autre site sur lequel on dispose d'un service de maintenance à distance. Mais la

gestion locale des catalogues d'un site consistera à utiliser le service de maintenance qui existe localement, ou à faire appel au service de maintenance général qui agira sur place. Ce choix dépend de l'alternative entre le système généralisé et l'ensemble de systèmes autonomes interconnectés, donc de l'architecture logique du réseau.

En ce qui concerne la méthode d'accès aux fichiers, le choix est le même. Puisqu'il existe une méthode d'accès aux fichiers de tous les autres sites, pourquoi ne pas l'employer pour accéder aux fichiers locaux? En fait, il s'agit de déterminer le moment où l'on distingue les fichiers locaux des fichiers distants. Soit c'est fait par le service de gestion et d'accès aux fichiers auquel on s'adresse indifféremment pour tout fichier. Soit, les usagers de ces services feront la distinction, et n'appelleront le service de gestion ou d'accès distant que pour les services qui ne sont pas locaux.

2.3.2 Stockage et impression

L'espace de stockage, tout comme le matériel d'impression, peut être regroupé sur certains sites ou dispersé plus généralement sur le réseau. Des sites peuvent être entièrement dédiés à un serveur de fichiers ou d'impression; ou bien la présence d'un serveur sur un site peut permettre le travail en parallèle d'un usager sur ce site. Dans ce cas, on peut se trouver devant la nécessité de distinguer les possibilités locales et distantes. Cependant tout site offrant un service doit rester accessible aux usagers des autres sites.

2.3.3 Transfert

Le transfert ou échange d'information doit pouvoir se faire entre deux sites quelconques. Néanmoins cette notion a évolué. Au départ, il s'agissait du transfert explicite d'un fichier entre deux sites nommément désignés. Dans un système généralisé, il s'agit plutôt de sa copie d'un catalogue vers un autre : cela peut consister à copier la référence du fichier ou à déplacer le fichier entièrement. Initialement, et en toute généralité, on conçoit un transfert comme ayant lieu entre deux sites émetteur et récepteur, et contrôlé par un troisième site qui en prend l'initiative. L'unité échangée est un fichier entier.

Pour simplifier ce service, on peut imposer l'identité entre le site de contrôle et l'un des sites émetteur ou récepteur. De plus, on peut multiplexer, sur une même liaison entre les sites concernés, la circulation des données et des informations de contrôle. Cela ne diminue en rien les possibilités de transfert d'information entre sites, mais peut amener à des restrictions sur les sites d'où l'on pourra les demander. Dans un système généralisé, où l'on tend à cacher à l'utilisateur la répartition des services et des ressources, le service de transfert doit être le plus général possible et les restrictions ne doivent pas intervenir sur les facilités offertes aux usagers.

2.4 Utilisation des services étendus

La nature des services proposés étant précisée, examinons les bénéficiaires de ces services étendus. Ce sont des utilisateurs physiques, des programmes d'application offrant des services plus évolués, et en

définitive un système généralisé étendu au réseau. Tous ces services sont rarement offerts directement à leur utilisateur final, mais souvent proposés par l'intermédiaire de services plus évolués. On doit donc se rappeler que les programmes d'application, autant et sinon plus que les usagers eux-mêmes, ont besoin de la puissance de calcul et des possibilités d'accès aux services déjà existants.

2.4.1 Utilisateur physique

L'usager humain étant la justification du développement de ces services, leur finalité doit tendre à le satisfaire. Cette satisfaction est mesurée par l'agrément qu'il ressent à se servir de ces nouveaux outils. Elle tient à la facilité qu'offrent ces outils à anticiper les désirs de l'utilisateur, à la démarche faite pour lui rendre les services accessibles par leur présentation, et à l'impression qu'il ressent de pouvoir les maîtriser. Depuis l'usage des langages de commande, la possibilité du travail interactif a permis une meilleure réponse aux vœux des utilisateurs. L'interface entre l'homme et la machine est devenue moins mystérieuse. La machine sait mieux faire connaître ce qu'elle peut réaliser, et mieux "comprendre" ce que désire son utilisateur. Elle permet aussi à l'utilisateur de voir tout de suite ce que la machine a compris pour réagir immédiatement en cas d'erreur.

En fait, la machine n'apparaît plus qu'au travers des services qu'elle permet de rendre. Aussi, les services proposés doivent être ceux que conçoit l'utilisateur même si la machine permet de faire plus. Parmi les quatre services proposés sur un réseau, ceux que l'utilisateur peut appréhender directement sont l'impression, le stockage et la maintenance, et le transfert d'informations. Le

fichier y constitue le volume d'informations que l'utilisateur peut atteindre et manipuler directement.

2.4.2 Programmes d'application

Ce que l'utilisateur humain ne peut faire lui-même, un service informatique doit lui fournir les moyens de le réaliser. Ainsi, l'accès au contenu des fichiers se fait par l'intermédiaire d'outils plus spécialisés (éditeurs ou utilitaires de traitement de texte) ou bien des outils à vocation plus informatique (compilateurs, assembleurs, éditeurs de liens) qui manipulent l'information sous forme de fichiers, mais qui en utilisent le contenu pour le transformer.

Outre l'accès au contenu des fichiers, les programmes d'application doivent disposer de l'accès à tous les autres services du réseau pour les combiner en vue d'en constituer de plus vastes, de plus évolués qui prennent en charge une plus grande part du travail d'un utilisateur non informaticien. Cet accès à des services plus élémentaires peut se faire grâce à des primitives d'appel des services au sein d'un programme, ou prendre la forme d'un protocole de dialogue avec un serveur qui gère certaines ressources.

2.4.3 Système intégré

Les services présentés auparavant peuvent être développés indépendamment, mais ils ne doivent pas être disjoints dans leur finalité. Leur mise en oeuvre se fait dans un esprit de collaboration, soit entre eux, soit pour des services qui les utiliseront pour en fournir de plus évolués. Leur base commune est le service de transport du réseau qui permet aux programmes des différents sites de

communiquer entre eux. Puis le service de transfert et d'échange d'informations peut être utile au service d'imprimerie ou au service de stockage, ainsi qu'à un service plus évolué de messagerie.

APPLICATIONS

imprimerie messagerie

service
fichiers

service de transport

Intégration des services

3 PROBLEMES SOULEVES PAR LES ECHANGES SUR LES RESEAUX LOCAUX

Dans un style "Café du Commerce" on peut exprimer ces problèmes ainsi :

- 1) on sait encore moins où ça se trouve (désignation et localisation)
- 2) c'est pas partout pareil (hétérogénéité)
- 3) ça tire de partout, chut! les murs ont des oreilles (protection)
- 4) mettons nous d'accord une bonne fois (cohérence)
- 5) vous êtes sûrs que ça va toujours marcher (fiabilité)

Du point de vue d'un utilisateur, l'environnement nouveau constitué par le réseau local est aussi complexe qu'un système classique multiutilisateur. Mais l'utilisateur n'a pas un accès aussi commode à toutes les composantes du système ainsi constitué. Cet ensemble de matériels et de systèmes interconnectés devient plus difficile à appréhender par un usager informaticien, et à plus forte raison, par un non informaticien. Ces personnes auront besoin de l'aide que peuvent apporter de nouveaux services mis en place au sein de ce système réseau. La mise en oeuvre de ces services, réalisés par les machines du réseau, devient aussi plus difficile à concevoir. Les possibilités proposées aux utilisateurs seront dépendantes de la structure logique du réseau et de l'organisation des services qui y participent.

Les principaux problèmes qui apparaissent concernent la désignation et la localisation des ressources, l'hétérogénéité des matériels et des systèmes, la protection des informations conservées, la cohérence de la manipulation de ces informations et la fiabilité des services.

3.1 Désignation, localisation et structuration des ressources

Avec la croissance en taille et en diversité des réseaux locaux, il devient difficile à un usager de connaître l'ensemble des sites, les différents serveurs de fichiers et tous les catalogues existants. Le partage, imposé par la répartition et la mise en commun des ressources, doit être réalisé avec un minimum d'actions explicites de la part de l'utilisateur. D'apparence, la répartition des ressources devient transparente ou, plutôt, invisible à l'utilisateur. L'émission et la réception d'information entre les ordinateurs du réseau ne se font plus sur demande de l'utilisateur, et autant que possible lui sont totalement invisibles <Satyanarayanan 81>.

De nouveaux services à l'échelle du réseau se chargent de ce travail à la place de l'utilisateur. Ces nouveaux services du réseau doivent donc pourvoir aux décisions qui ne sont plus prises par les utilisateurs. Leur rôle est d'assurer la localisation des fichiers d'après la désignation qu'en font les utilisateurs et en fonction de l'organisation des ressources. Un choix important est la mise en place d'un mode de désignation et d'une structure d'organisation des ressources qui permettent à ces services de comprendre au mieux les désirs des utilisateurs.

3.1.1 Désignation et localisation

La désignation d'une ressource désirée par un utilisateur, et sa localisation, sont de moins en moins à la charge de l'utilisateur. Ce dernier se rend plus ou moins compte de la répartition des ressources. Le système réparti sur un réseau local tend de plus en plus à cacher sa répartition et à gérer à l'insu de son utilisateur les problèmes qui se posent <Mitchell 82a>.

La recherche d'un document par un usager se situe sur deux niveaux: d'une part, la désignation par l'usager de ce qu'il désire, pour le compte du service de gestion de fichiers; d'autre part, la localisation par ce service du document réclamé par l'usager. On peut admettre plusieurs désignations différentes pour une même information, ce qui permet des voies d'accès multiples à la même information partagée; ou bien, l'existence de plusieurs fichiers répondant à la même désignation, dans le but d'assurer la sécurité par redondance de l'information, et la facilité de l'accès à l'information <Lunn 81>.

3.1.1.1 Désignation

On peut distinguer trois notions dans les relations entre un usager et une ressource qu'il désire employer. D'une part, le nom de la ressource qui permet d'indiquer ce que l'on cherche. D'autre part, l'adresse de la ressource qui dit où elle se trouve. Enfin au milieu, le chemin d'accès qui montre comment on peut aller chercher la ressource.

Dans la conception initiale des réseaux locaux, c'est l'usager qui sait et indique à la machine où se trouve le fichier dont il a besoin. Il précise la machine, le catalogue dans cette machine, et l'emplacement au sein de ce catalogue du fichier qu'il désire utiliser. Selon ce principe d'apparence de la répartition, le stockage et la récupération de fichiers sur des sites différents se font par des commandes explicites.

On peut conserver la trace de la répartition en adoptant une désignation hiérarchisée. Le cheminement vers une information correspond alors à une progression dans un arbre dont les feuilles sont les fichiers, et les noeuds

des étapes dans la localisation. Si l'on veut complètement oublier la répartition, on doit adopter une désignation banalisée. Tout fichier, où qu'il se trouve, sera uniformément désigné d'une façon similaire.

Pour remplir ces deux rôles : désignation et localisation, on peut adopter un nom, textuel ou imprimable, pour désigner les fichiers. Ce nom sera mis en correspondance dans un catalogue avec un identificateur interne qui permettra de localiser le fichier.

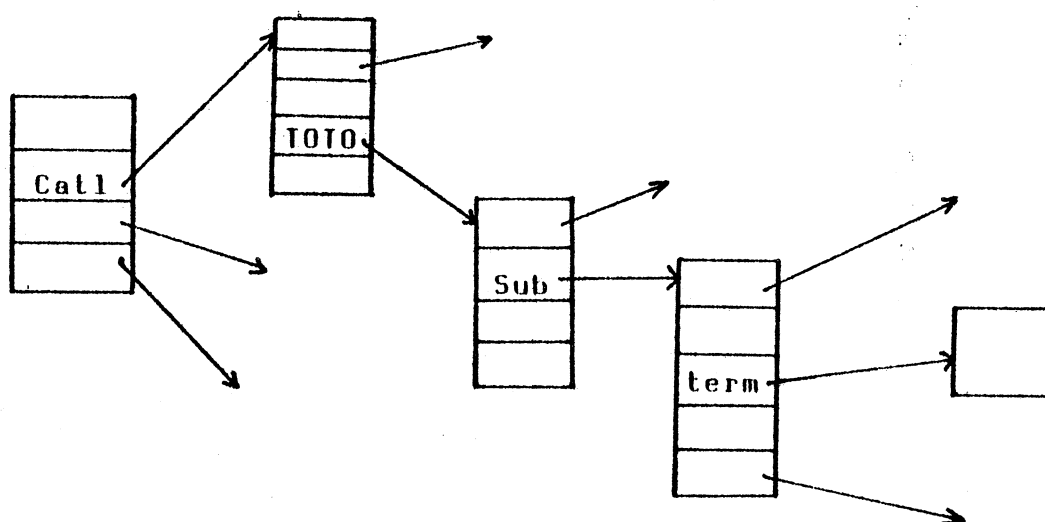
3.1.1.1.1 Désignation externe

On suppose que l'identificateur interne n'est valable que chez le serveur qui gère les fichiers. La forme valide à l'extérieur est le nom textuel. Pour permettre la localisation du fichier, le nom externe doit avoir une portée qui couvre tout le système de fichiers. Il peut même contenir d'une certaine façon une indication sur le site où il se trouve. Mais il est alors impossible de faire des transferts ou de conserver plusieurs copies car les noms seraient différents. Par contre, si le nom ne contient pas d'indication sur le site de résidence, il faut faire une recherche parmi tous les serveurs connus. Cependant, pour limiter ce travail de recherche, il est possible de garder des copies multiples si l'on dispose des mécanismes de mise à jour nécessaires.

Des services de base peuvent gérer une structure de catalogues qui couvre tout le réseau. Les catalogues associent le nom du fichier à son identificateur interne. L'avantage est qu'un fichier n'ayant plus de nom externe est inaccessible, donc il peut être détruit, ce qui permet de récupérer sa place sans se poser de question. L'inconvénient est que, si le nom est trop limité

localement dans sa signification, il ne peut pas être utilisable par d'autres applications ou systèmes de nomenclature. Les catalogues peuvent être organisés sur plusieurs niveaux. Un catalogue associera alors à un nom l'identificateur interne du fichier ou la référence d'un sous-catalogue. Le nom comportera donc différents suffixes qui conduiront de sous-catalogues en sous-catalogues jusqu'à aboutir à l'identificateur interne du fichier.

Ex: Nom = Cat1.TOTO.Sub.term



3.1.1.1.2 Désignation interne

On peut décider que l'identificateur interne est le seul nom valide. Les noms textuels ne sont alors qu'un moyen commode de découvrir les noms internes. Le serveur qui gère les fichiers n'a plus de responsabilité en ce qui concerne le nom textuel. Les clients de ce service peuvent bâtir la nomenclature qu'ils désirent pour les fichiers en adjoignant une sorte de serveur pour gérer la transformation des noms textuels en identificateurs internes <McLellan 80>.

Chaque fichier est repéré par son identificateur. C'est le seul moyen d'identification du fichier, il se réfère toujours à un seul et même fichier malgré les modifications que peut subir le contenu de ce dernier. C'est un entier unique, généré par le système à la création du fichier, qui lui est attaché pour toute son existence et n'est jamais modifié. Cet identificateur doit être unique sur tout le réseau. Le meilleur moyen est qu'il comporte une indication du site sur lequel se trouve le fichier qu'il désigne et qu'il soit unique sur ce site.

3.1.1.2 Localisation

Après le parcours des catalogues et sous catalogues, on aboutit au nom interne et à l'adresse de l'objet recherché. Connaissant ainsi sa localisation, il faut suivre le chemin d'accès pour atteindre l'objet. Souvent dans les réseaux locaux, le chemin d'accès à un matériel découle directement de son adresse car le cheminement revient à un déplacement le long d'un support commun de transport. Dans le cas des fichiers d'information que l'on peut rechercher sur un réseau, la connaissance de l'équipement qui contient ce fichier ne donne pas immédiatement accès au document, il faut aussi rechercher le fichier à l'intérieur de la machine.

3.1.1.3 Interférences

Les services de gestion des informations ne pourront pas être aussi efficaces que les usagers dans la mesure où ils n'auront connaissance que des informations auxquelles ils pourront avoir accès. Tous les fichiers peuvent ne pas être accessibles à la fois : des sites ne sont pas en ligne

par exemple, ou des parties de catalogues ne sont pas présentes car elles se trouvent sur des volumes démontables. Un usager pourra le savoir, mais le système doit gérer des informations supplémentaires pour savoir que ces fichiers existent mais ne sont pas accessibles à un moment donné. Il devra procéder aussi à une mise en cohérence des catalogues lors de l'adjonction d'un site ou du montage d'un volume <Lunn 81>.

Lorsque la gestion des noms n'est pas faite par le serveur de fichiers, ce dernier peut ne gérer que l'espace de stockage des informations. Dans ce cas, les unités de stockage sont repérées par leur numéro d'identification, et ordonnées selon les décisions du client. Le serveur perd connaissance de la structure du graphe de dénomination, et ne peut plus détecter si un fichier devient inaccessible. Il faut que les structures d'organisation et d'occupation de l'espace lui soient accessibles ou même gérées par lui pour qu'il puisse récupérer l'espace occupé par les fichiers inaccessibles à partir de la racine.

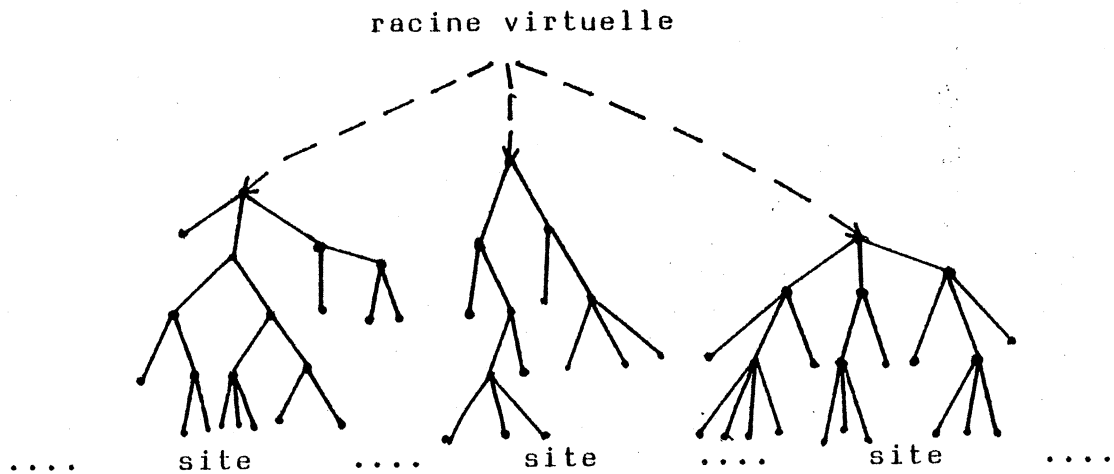
3.1.2 Structure des catalogues réseau

Par construction, chaque site gère son propre catalogue de fichiers. Pour connaître la totalité des fichiers du réseau, on peut réunir logiquement les catalogues dans un même ensemble. Le grand avantage est qu'il n'est pas nécessaire de dupliquer des informations déjà existantes. Le service de gestion des fichiers est alors un système distribué en soi. Il existe un ou plusieurs serveurs séparés contenant chacun son propre système de gestion de fichiers. Un client désirant accéder à un fichier doit connaître le serveur qui le détient, et dialoguer avec lui pour avoir accès au fichier.

Lorsque le système est étendu à tout le réseau, et n'est plus un ensemble de systèmes indépendants interconnectés, la dénomination des fichiers, leur structure d'organisation et de localisation couvrent tout le réseau. Le catalogue des fichiers du réseau possède une structure de graphe orienté. Il peut posséder une racine réelle ou virtuelle et s'étendre sur plusieurs niveaux de hiérarchisation.

3.1.2.1 Racine virtuelle

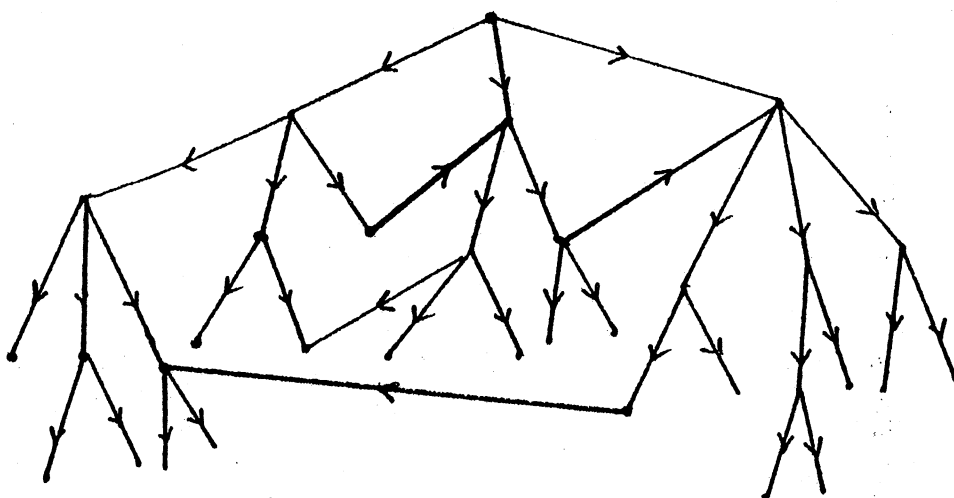
Une façon de constituer un catalogue général des fichiers du réseau, en utilisant les catalogues locaux existants, est de constituer une arborescence à racine virtuelle. Cette racine virtuelle est un catalogue fictif qui contient les racines des arborescences de chacune des installations connectées. L'ajout d'une nouvelle installation est analogue au montage d'un nouveau système de fichiers dans un répertoire de la racine virtuelle. On voit qu'avec cette façon de faire, chaque site garde son indépendance de gestion de ses fichiers, et qu'une fois arrivé sur un site, on ne quitte plus son système de gestion de fichiers <Wupit 81>.



3.1.2.2 Arborescence

On peut entremêler les catalogues de chacun des sites pour aboutir à un graphe orienté dont la racine réelle se trouve sur un site. Cette racine est un catalogue donnant la liste des sous-catalogues et les liens vers eux. Chaque noeud du graphe est un sous-catalogue qui aboutit soit à d'autres sous-catalogues, soit à des fichiers qui constituent les noeuds terminaux du graphe. Les fichiers sont référencés grâce au chemin d'accès qui permet d'aller jusqu'à eux à partir d'un point du graphe. Les catalogues ou sous-catalogues font correspondre un nom et une valeur. Cette valeur peut représenter l'identificateur du fichier, en cas de référence directe, ou le lien symbolique vers un catalogue. Ce lien symbolique est défini par l'utilisateur pour aller vers un catalogue particulier, ou peut correspondre à une porte de communication du réseau.

racine réelle



Au sein de tels graphes, la désignation et la localisation des fichiers sont étroitement mêlées. En même temps que l'on progresse dans le graphe de dénomination, on peut passer de site en site à travers les portes de

communication du réseau. Eventuellement on peut parcourir des cycles, mais on évite une trop grande extension de ces cycles en cachant aux utilisateurs les références aux racines réelles ou virtuelles du graphe orienté <AMOEBA 82>.

Gestion des catalogues et désignation des fichiers
selon la constitution des catalogues réseaux

Catalogues	gestion	désignation
Réunion logique des catalogues des sites	indépendante sur chaque site	nom du site plus nom sur le site
Racine virtuelle	indépendante sur chaque site	globale: un site = un sous-catalogue
Arborescence à racine réelle	entremêlée sans notion de site	globale

3.1.3 Recherche des objets et améliorations

Dans une arborescence qui couvre tous les fichiers du réseau, la recherche des fichiers est évidente car leur nom donne la progression à suivre dans l'arborescence pour y aboutir. Dans les autres modes de fonctionnement, la recherche pratique d'un fichier à la demande d'un client peut prendre plusieurs formes.

Le principe général est de retrouver le site qui contient le fichier, et d'y faire interpréter localement le

nom du fichier. La stratégie consistant à trouver un serveur et à donner le nom d'un fichier sur ce site est en général imposée par la mise en commun de matériels identiques sur le réseau sans une dénomination globale. Cela correspond à la structure logique de réseau où la répartition est apparente à l'utilisateur <Mitchell 82a>.

Pour localiser le serveur qui possède le fichier qu'il désire, le client peut demander à chacun d'eux s'il le possède. Cette stratégie, valable si le nombre de serveurs est réduit, devient prohibitive quand leur nombre croît.

3.1.3.1 Accélération de la recherche

Une solution est que les serveurs aident le client à localiser le fichier. Le client en fait la demande au serveur qui lui semble avoir le plus de chance de détenir le fichier. Si celui-ci le possède, c'est terminé. Sinon ce serveur demande au suivant. En fin de compte, soit le fichier est retrouvé, soit il n'existe pas. L'avantage de cette stratégie est que seuls les serveurs ont à connaître la façon dont il faut utiliser le nom du fichier pour le localiser. Cela évite de surcharger le code des programmes clients avec la partie qui localise un fichier d'après sa désignation.

Selon les possibilités offertes sur le réseau, on peut utiliser aussi la diffusion de la demande d'un fichier. C'est efficace et peu coûteux, dans la mesure où la diffusion est fiable, et que quelqu'un peut répondre que le fichier, le cas échéant, n'existe pas <Mitchell 82a>.

Pour éviter les recherches de fichiers à l'extérieur d'un site, on peut ajouter au catalogue standard un catalogue complémentaire. Il constituera le répertoire des

fichiers accessibles externes au site. Il contiendra les noms et lieux où se trouvent les fichiers. Chaque site pourra ainsi connaître tous les fichiers existants sans avoir à les demander à l'extérieur. Mais lors de la création ou de la destruction de fichiers, la mise à jour des catalogues devra se faire sur tous les sites, avec les problèmes de cohérence que cela pose <Chambon 81>.

La solution qui minimise la diffusion des mises à jour des catalogues consiste à limiter le nombre des catalogues qui contiennent les noms de tous les fichiers, et à placer ces catalogues sur les sites qui ont les plus fortes ressources en matière de fichiers. Ainsi l'accès à ces catalogues généraux est réduit aux cas de création de fichier, pour éviter l'homonymie, de destruction de fichier et de recherche d'un fichier. On peut réduire à un le nombre de ces catalogues généraux, ce qui facilite les opérations de mise à jour.

3.1.3.2 Meilleure fiabilité

La duplication des catalogues généraux permet de faciliter la recherche des fichiers. On peut les trouver plus facilement, et accéder à ces catalogues suivant plusieurs voies. Il en est de même pour les fichiers. En disposant de plusieurs exemplaires d'un fichier, on en facilite l'accès en rapidité et en simultanéité. On permet aussi l'accès même en cas de panne de l'un des sites qui contient le fichier. Par ailleurs, il faut veiller à ne pas saturer l'espace de stockage en y laissant des fichiers qui ne sont plus accessibles.

3.1.3.2.1 Informations conservées

Pour empêcher le fonctionnement, il faut la panne simultanée de tous les sites qui contiennent un catalogue, ce qui est peu probable. Ces sites étant choisis en fonction des ressources déjà existantes en matière de fichiers, la probabilité est très forte de trouver le fichier lui-même sur place quand on cherche à obtenir sa localisation. En outre cela permet de disposer sur le site de la place suffisante pour gérer le catalogue général.

3.1.3.2.2 Gestion de l'espace de stockage

Les arborescences de catalogues représentent la structure logique d'implantation du système de fichiers. La structure physique qui contient ces informations doit aussi être gérée pour conserver toujours disponible l'espace libre, et assurer son organisation. Pour être efficace, cette gestion doit être faite au niveau de chaque volume d'espace de stockage sur disque. Chacun de ces volumes doit être organisé en une structure hiérarchique. Les seuls fichiers accessibles sont ceux répertoriés dans cette hiérarchie. Les blocs non occupés par ces fichiers font partie de l'espace libre. Mais cela ne résoud pas le problème des fichiers créés par un client qui perd ensuite leur identificateur.

Recherche de fichiers sur le réseau
en l'absence de dénomination globale

	serveur	client	efficacité
Interrogation de chaque site	consultation locale	mène la recherche	long mais très efficace
Collaboration en chaîne des serveurs	consultation locale, réponse ou appel au suivant	lance la recherche	aussi long et aussi efficace
Diffusion de la demande	consultation locale et réponse si trouvé	diffuse la demande et attend	rapide mais risqué d'absence de réponse

3.2 Hétérogénéité

Beaucoup d'installations de calcul sont constituées d'une variété d'ordinateurs, différents les uns des autres par leur architecture et par leur interface vers l'utilisateur. Malgré les améliorations en performance et en fonctionnalité apportées par les nouveaux systèmes, l'investissement en logiciel dans les anciens systèmes et l'effort nécessaire à la formation du personnel, aux fortes habitudes acquises, découragent les responsables de telles installations de mettre au rebut leurs vieux systèmes. Par voie de conséquence, l'échange d'information et le partage de ressources entre systèmes non similaires ont une signification particulière dans de telles installations. C'est d'ailleurs l'une des raisons de l'intérêt croissant pour les réseaux locaux ces dernières années <Satyanarayanan 81>.

Les problèmes à surmonter en matière d'hétérogénéité concernent les ordinateurs du réseau, pour le code des programmes et le codage des informations binaires, textuelles ou autres. Ils touchent les systèmes installés sur ces sites pour la gestion des fichiers, les conventions de représentation de l'information et la gestion des tâches exécutables, ainsi que les mécanismes de protection de l'accès aux informations.

3.2.1 Les sites

Les sites du réseau sont des machines informatiques avec leurs particularités en matière de code exécutable et de codage des informations de toute nature.

Les programmes, sous forme de code objet ou de code binaire, ne peuvent pas être échangés directement pour être utilisés sur des sites qui diffèrent par la nature ou la configuration de leurs machines. En cours de fonctionnement d'un service, la migration d'un processus d'un site du réseau vers un autre nécessite une adaptation ou une transformation réalisées au préalable. Un service doit exister sur chaque site pour permettre le chargement du code des programmes grâce à une interface simple et uniforme. L'accès à un nouveau site sur le réseau nécessite l'écriture et l'installation d'un tel contrôleur pour gérer le chargement du code et le transfert d'exécution <Herbert 81>.

En matière de codage des informations, la représentation binaire des nombres pose le problème de leur précision numérique <Manning 76>. Les différents codages des caractères réclament des tables ou des utilitaires de conversion. La représentation des informations au sein des fichiers présente autant de différences et demande les mêmes possibilités de transformation.

Il est nécessaire de disposer d'utilitaires de transcodage qui permettent les transformations entre les différentes représentations. Une représentation normalisée, non encore adoptée par les instances internationales de normalisation, permettrait de faire circuler l'information sous une forme unique. Des utilitaires offriraient la possibilité de passer de cette forme standard à chaque représentation locale. Il paraît plus économique de passer par cette représentation normalisée pour convertir toute représentation locale en toute autre lorsqu'une information est destinée à plusieurs autres sites. Mais si cette information ne doit passer qu'entre deux sites il vaut mieux, si possible, employer un utilitaire qui transforme directement une représentation en une autre.

3.2.2 Les systèmes d'exploitation

L'existence de systèmes différents conduit à des services de gestion des fichiers très peu semblables. Les adaptations nécessaires peuvent être assurées par des traducteurs de langages de commande, ou des émulateurs d'appels aux différents systèmes existants. Ces deux modes de transformation permettent à un usager de travailler sur un site à partir d'un autre site sans avoir à connaître les particularités du site éloigné. On peut aboutir à une uniformisation des services locaux de gestion des fichiers pour donner l'apparence d'un système unique <Mitchell 82>. Les commandes, passées à travers des interfaces semblables, sont converties en des commandes valables localement, ou bien envoyées sur un autre site plus directement concerné par l'exécution de la commande.

3.3 Protection

La protection se manifeste par l'attribution aux utilisateurs de droits d'accès aux informations. Il faut garantir la définition des droits d'accès et leur vérification par les entités qui gèrent les informations et fournir les moyens d'assurer la protection des informations.

Comme on fournit aux utilisateurs les moyens de manipuler des fichiers sur des sites distants, on doit s'assurer qu'ils disposent bien des droits d'accès aux fichiers qu'ils peuvent atteindre. Sur un réseau, on peut rencontrer une hétérogénéité de la protection de l'information d'une part sur la manière d'enregistrer un usager, d'autre part sur la manière de le reconnaître lorsqu'il désire accéder à l'information.

Selon le degré de protection du système où il travaille, un usager pourra voir ou non ses informations utilisées, copiées, modifiées ou détruites, ou son espace de travail envahi par d'autres usagers. Mais ce qui peut se produire sur une machine reliée au réseau n'est que ce qui aurait pu se produire localement sur une machine à système multi-usagers. En effet, un usager distant intervient toujours par l'intermédiaire d'un outil local commandé à distance. Cet outil local subit les contraintes de protection du système local, il peut même imposer des contraintes plus fortes pour réaliser une protection aussi stricte que sur un autre site. C'est à cet outil de vérifier les droits d'un usager distant. Or il peut très bien ne pas assurer une protection supérieure à celle fournie par le système local.

3.3.1 Définition des droits d'accès

Les droits d'accès sont définis par un système à capacité ou par des listes de droits d'accès attachées aux objets.

3.3.1.1 Système à capacité

Le système à capacité est un mécanisme qui fournit et demande une clé pour atteindre un service ou un objet. C'est l'équivalent d'un mot de passe qui serait attaché à un seul objet et connu seulement des utilisateurs autorisés. La capacité a la forme d'une chaîne de bits contenant une grande part d'information aléatoire. Le principe consiste à partir de l'ensemble des identificateurs des services ou des objets. Cet ensemble est initialement dense car toute valeur correspond à une désignation. En ajoutant aux identificateurs des bits à

valeur aléatoire, on va disperser les valeurs valides dans un ensemble plus vaste. Le choix de ces bits par le service qui gère les informations conduit au système à capacité, leur choix par l'utilisateur donne lieu à une protection par mot de passe.

Cette capacité peut être le numéro d'une porte du système de communication qui permet d'accéder à un site, à un autre service, ou à un catalogue de fichiers. Les droits d'accès qui sont codés dans la clé de capacité peuvent être modifiés par celui qui les accorde, ou réduits pour fournir l'accès à une ressource mais avec des droits moins puissants. Il est possible pour le propriétaire d'invalider toutes les clés d'accès à une ressource en modifiant la capacité à montrer pour y accéder; cela revient en quelque sorte à changer la serrure <AMOEBA 82>.

3.3.1.2 Listes de droits d'accès

Grâce aux listes de droits d'accès, on peut indiquer pour chaque objet les droits de chaque utilisateur éventuel de cet objet. Pour ne pas avoir à énumérer chacun des usagers, on introduit la notion de groupe d'accès. C'est un sous-ensemble d'usagers à qui on accorde ou retire collectivement des privilèges <Satyanaranan 81>. Comme un usager peut appartenir à plusieurs groupes d'accès, son environnement de protection est défini, à chaque instant, comme l'union des droits de chaque groupe d'accès dont il est le membre à cet instant. Par rapport à un objet, on peut définir, au sens d'un accroissement des droits, des ensembles concentriques d'utilisateurs formés d'abord par le propriétaire de l'objet, puis les membres du groupe dont fait partie l'utilisateur et ensuite tous les utilisateurs du système <Wupit 81>.

La validation de l'accès se fait soit selon la capacité présentée par le client, soit selon l'identité qu'il présente et le mot de passe qu'il détient. Dans ce dernier cas, le client aura, parmi les droits attachés à l'objet qu'il réclame, ceux qui correspondent au groupe d'utilisateurs dont il fait partie. La liste de droits d'accès fait correspondre à chaque groupe ses droits d'accès à l'objet. On peut y distinguer deux parties : les droits d'accès système, et des droits d'accès auxiliaires. Les premiers sont vérifiés par le serveur qui gère les objets. Les seconds sont interprétés par des sous-systèmes. On peut illustrer la différence entre ces deux parties par l'exemple suivant: un compilateur du langage PASCAL doit, pour accéder au fichier à compiler, avoir des droits en lecture sur ce fichier, ce sont les droits d'accès système; un droit d'accès auxiliaire à vérifier serait que le nom du fichier à compiler comporte le suffixe .PASCAL.

3.3.2 Lieu de conservation de ces droits

Les droits d'accès à une information peuvent être inscrits sur le support physique de cette information, ou sur le catalogue des fichiers. La consultation de ces droits peut être faite par les serveurs qui manipulent les informations ou par un serveur d'authentification. Le serveur d'authentification, associé à un serveur de nom, a pour but de contrôler les droits d'accès d'un usager à une ressource. Le serveur de nom permet, pour une ressource d'associer à son nom sa localisation ainsi que ses propriétés et caractéristiques diverses (protection, attributs,...) et pour un usager d'associer à son identificateur les droits qu'il possède et les particularités de son environnement. Le serveur d'authentification permet de vérifier l'identité d'un

usager et la concordance entre ses droits et la protection des ressources qu'il désire atteindre.

Chaque bloc physique du disque peut contenir l'identification du propriétaire de l'information qui y est inscrite. Ainsi, à la lecture ou à l'écriture, il est possible de vérifier si une requête est autorisée sur un bloc donné. Ces indications de propriété ne feront pas partie des données effectivement contenues sur le bloc <AMOEBA 82>.

Le système de catalogues est un lieu qui convient pour implémenter la protection des fichiers, surtout dans le cas où l'on doit gérer des copies multiples d'un fichier. Par ce moyen, il est possible de protéger des catalogues entiers, ou des fichiers individuellement <Lunn 81>. Le contrôle peut consister à ce niveau à vérifier l'identité des usagers au moyen de leur nom et de leur mot de passe. Il peut être réalisé par un serveur d'authentification, à la demande d'un client qui désire faire vérifier les coordonnées de l'utilisateur qui l'a appelé.

Par ailleurs, si un travail ne peut pas être fait directement par un usager, un service ou programme peut intervenir en agissant au nom de l'utilisateur et avec les droits correspondants. D'un autre côté, on peut envisager un programme qui prene, à un moment précis, des droits supérieurs à ceux de son utilisateur pour manipuler des données spécifiques qui lui sont réservées <Ritchie 74>. On peut illustrer ce genre de programme par l'exemple d'un serveur d'impression en différé qui détruit les fichiers après les avoir imprimés. Les droits d'accès du serveur aux fichiers à imprimer puis détruire sont ceux de l'utilisateur. Par contre pour manipuler sa liste des fichiers à imprimer, le serveur a des droits qu'aucun usager ne doit posséder.

3.3.3 Assurer pratiquement la protection

Les mécanismes pratiques qui permettent la mise en oeuvre de la protection sont le cryptage des indicateurs de droits d'accès et l'absence de désignation absolue des ressources. On précisera aussi une condition élémentaire de validité de la protection des informations.

Le cryptage peut être utilisé pour protéger les capacités dans les systèmes distribués, mais il ne peut servir qu'à empêcher les usagers de changer les droits d'accès indiqués dans la clé. De plus, si l'espace des clés est dense on peut facilement tomber sur de bonnes clés en les fabriquant au hasard et faire des dégâts en aveugle. Par contre si les clés sont bien dispersées, il sera difficile de tomber au hasard sur une clé qui fonctionne. Ce sera plus la dispersion que le cryptage qui assurera la protection <AMOEBA 82>.

L'absence de désignation absolue des ressources signifie que l'utilisateur ne pourra pas atteindre n'importe quel objet de façon directe. Cela évite de laisser un usager accéder à des objets qui lui sont interdits. En effet, le cheminement relatif qu'il devra suivre ne lui permet d'accéder qu'aux ressources qui sont indiquées à partir du point où il se trouve et qu'il a pu atteindre selon ses droits.

Pour assurer effectivement toutes ces protections, les serveurs de fichiers doivent fonctionner sur des machines sur lesquelles on dispose d'un contrôle total. Sinon des programmes hors de contrôle peuvent accéder à l'espace utilisé par le système de fichiers que l'on a mis en oeuvre. Les seuls lieux sûrs sont : une machine spécialisée

comme serveur de fichiers, ou un environnement de travail dans un système multi-usagers en temps partagé. Un tel grand système dispose de beaucoup d'espace de stockage sur disque. Y installer le serveur de fichiers apporte un gain considérable car la plupart des demandes proviendront du système lui-même. Néanmoins, il faut être attentif à assurer la protection de l'espace utilisé par le serveur de fichiers <McLellan 80>.

3.4 Cohérence des données conservées

La principale tâche d'un système de gestion de fichiers consiste à conserver des informations et à assurer leur intégrité. Cela signifie que, tant dans le détail que dans leur ensemble, les données conservées ne subiront pas d'altération à l'insu de leur possesseur. Il faut donc assurer la cohérence des manipulations ponctuelles sur les fichiers et les catalogues. La cohérence apparaît nécessaire car des données peuvent être consultées ou modifiées simultanément par plusieurs usagers, ou parce que les données sont représentées en plusieurs exemplaires dont il faut assurer l'identité au fur et à mesure des modifications.

Un serveur de fichiers partagés qui prétend à une certaine généralité doit posséder les moyens de maintenir la cohérence des données qui lui sont confiées. Il ne peut évidemment empêcher un client de commettre, à l'aide de manipulations correctes, des modifications n'ayant aucune signification. Par contre, il peut contribuer à éviter les incohérences qui peuvent apparaître du fait d'évènements survenant hors du contrôle du client. Ce peuvent être des défauts de communication, ou les conséquences de l'accès concurrent aux mêmes fichiers par d'autres clients, sans précaution. On peut éviter que se produisent de telles incohérences en enfermant une séquence d'opérations élémentaires, demandées par un client, au sein d'une transaction. C'est dans le cadre des transactions que s'appliquent les règles dont l'observation permet d'éviter l'apparition des incohérences <Mitchell 82a>.

3.4.1 Violation de l'intégrité des données

L'intégrité des données dans un serveur de fichiers peut être violée pour deux raisons. D'une part, un client peut obtenir des données invalides en y accédant pendant qu'un autre les met à jour. D'autre part, le système peut s'arrêter au beau milieu d'une série d'écritures sur disque.

3.4.1.1 Première cause : accès simultané

Pour parer à la première cause de violation de l'intégrité des données, le serveur doit assurer que l'accès concurrent aux données ne peut conduire à l'obtention de données invalides. A cette fin, les modifications devront se faire en dehors de tout autre type d'accès. Cette exclusivité de la manipulation des données est obtenue par le blocage de l'accès aux données grâce à un mécanisme de verrou associé à la transaction qui manipule ces données. Le verrouillage est placé implicitement par le client sur les séries d'octets qu'il consulte ou modifie dans les fichiers manipulés.

Le verrou est placé au début d'une transaction, il est ôté quand elle se termine. Le meilleur verrouillage de l'accès est celui qui s'applique durant un temps limité pour éviter que, en cas d'erreur, des fichiers ne restent verrouillés pour toujours. Quand le délai de garde est écoulé avant que la transaction ne se termine, le verrou est dit fragile. Néanmoins, la transaction peut se terminer normalement si aucune autre n'est en compétition pour l'accès aux données verrouillées. Par contre, un autre client, qui désire accéder par le biais d'une autre

transaction aux données verrouillées, peut alors ôter le verrou précédent, avortant la transaction qui l'avait placé, pour poser son propre verrou <Mitchell 82a>.

3.4.1.2 Deuxième cause : panne imprévue

La parade à la seconde cause de violation de l'intégrité des données a entraîné l'introduction de la notion de mise à jour atomique. Cela consiste à effectuer toute modification en une seule opération, la transaction, qui se termine bien ou n'a pas du tout lieu. Evidemment, il paraît difficile de faire en une seule opération la mise à jour d'une page de disque. Donc, il convient de prendre des précautions pour qu'en cas d'échec on puisse revenir dans l'état antérieur, ou continuer les opérations de mise à jour.

3.4.1.2.1 Retour à l'état antérieur

Le principe de la tenue d'un "journal avant" répond à la première possibilité. Chaque page de disque à mettre à jour est auparavant recopiée dans le journal en correspondance avec la transaction responsable de la mise à jour. Les transactions en cours sont aussi répertoriées. En cas d'accident du système, on retrouve les transactions non achevées. Toutes les pages du journal qui sont concernées par des transactions inachevées sont réinstallées avec leurs anciennes valeurs reprises dans le journal.

3.4.1.2.2 Poursuite des modifications

Pour réaliser la seconde possibilité, on peut utiliser les listes d'intentions <McLellan 80>. Au lieu de mettre directement à jour le fichier, on écrit le contenu des pages modifiées à part. Quand la transaction s'achève, on

écrit dans une liste d'intentions les modifications à apporter aux fichiers. Puis on réalise ces intentions. En cas d'accident du système, les listes d'intentions sont redécouvertes et réalisées à nouveau. Cela suppose que ces intentions sont enregistrées de façon à ce qu'elles puissent être exécutées de manière répétitive sans dommage, en cas d'accidents successifs.

3.4.2 Copies multiples

Même si un usager est seul à manipuler les informations qu'il gère, il reste à assurer la cohérence des copies de la même information après un transfert qui crée une nouvelle copie. On peut se demander dans quelle mesure un fichier copié sous le même nom, d'un site sur un autre reste la copie fidèle de l'original, ou bien s'il représente une autre information susceptible d'être modifiée indépendamment. En plus des copies créées à l'occasion d'un transfert, on peut trouver des copies redondantes d'une même information, ainsi gardées par sécurité ou pour accélérer les accès. On peut en distinguer deux formes : la copie systématique en deux exemplaires de chaque page d'un fichier au cours de sa constitution pour sa conservation sur un support stable dans un souci de sécurité, et la copie en deux ou plusieurs exemplaires des fichiers entiers en des points divers du réseau sur un support ordinaire dans un souci de rapidité d'accès.

3.4.2.1 Support stable

Les données sur support stable sont écrites en deux endroits différents du système de fichiers, qui ne risquent pas, en principe, d'être simultanément détruits en cas de

catastrophe. Les deux copies sont toujours écrites dans le même ordre, ce qui permet de résoudre les éventuelles incohérences entre elles après les accidents du système. A chaque fois qu'une page de données doit être écrite sur un disque, son ancien contenu, ou le nouveau, doit pouvoir être reconstruit à partir du contenu d'autres blocs sur disque. En conséquence, pour chaque transfert d'une page, au moins une page de plus doit être écrite pour maintenir la redondance. Pour garder un tel niveau de robustesse aux données conservées, il faut doubler le nombre d'écritures <Mitchell 82a>.

3.4.2.2 Rapidité d'accès

La rapidité et la facilité de l'accès sont d'autres raisons de la redondance des données. La redondance est cachée aux processus utilisateurs, donnant l'apparence d'un seul fichier, alors qu'il peut en exister plusieurs copies. La consultation d'un fichier en plusieurs exemplaires peut se faire sur l'une quelconque des copies. Cela évite les communications pour atteindre le fichier à l'extérieur ou en réduit le coût et la durée, mais pose le problème de cohérence en cas d'accès en écriture sur les copies. La localisation des données redondantes est réalisée dans le système de catalogues qui doit constituer une part intégrante du système global. Le verrouillage, la cohérence et la protection sont assurées à travers les catalogues.

3.4.3 Catalogues multiples

Avant même d'accéder aux fichiers, leur recherche peut être facilitée par l'existence de copies multiples de catalogues. Cela va donner lieu à des problèmes de

cohérence. Pour les éviter, on peut se contenter d'un catalogue en un seul exemplaire. Mais cela va en réduire l'accessibilité et rendre plus difficile la reconstruction d'un catalogue endommagé par une panne.

On peut concilier l'existence d'un catalogue et de copies multiples de la façon suivante. Avec chaque copie d'un fichier on garde son nom et une estampille. On tient un seul catalogue de tous les fichiers. Ce catalogue est utilisé par les processus pour accéder aux fichiers. L'estampille est un simple entier conservé dans le catalogue central et incrémenté à chaque modification ou création de fichiers dans le catalogue. Elle est placée, pour chaque fichier, en regard de son entrée dans le catalogue. Elle se trouve aussi avec chaque version de fichier sur les volumes de stockage appropriés. Les versions identiques d'un même fichier auront la même estampille. En cas de défaillance du catalogue central, on consulte tous les volumes présents pour constituer un nouveau catalogue. La marque de temps fournie par la valeur de l'estampille assure que l'on garde bien la dernière version d'un fichier <Lunn 81>.

Pour améliorer les performances, on peut, au lieu d'un seul catalogue central, concevoir une hiérarchie de catalogues indépendants gérés selon le même principe.

3.4.4 Fiabilité proportionnelle à la valeur des données

Pour assurer la fiabilité des informations, le prix supplémentaire est à payer en espace de stockage pour conserver les copies redondantes, et en temps pour les maintenir à jour. Une particularité intéressante d'un système de gestion de fichiers est la possibilité de pouvoir préciser le niveau de redondance nécessaire et de

piloter le système pour obtenir la meilleure fiabilité au moindre coût <Lunn 81>. En outre, la possibilité de spécifier différents degrés de redondance pour différentes zones de l'espace de stockage, permet de refléter la nature critique de certaines données au sein du système <AMOEBAS 82>.

Pour les services de gestion de fichiers, les descripteurs de fichiers ou leurs entrées dans les catalogues peuvent se trouver en zone protégée, sur un support stable ou en plusieurs exemplaires, alors que les données elles-mêmes sont stockées sans plus sur un support ordinaire <Satyanarayanan 81>.

3.5 Fiabilité

Il est nécessaire d'assurer la fiabilité des informations conservées par un système de gestion de fichiers car ce système est sujet à des pannes matérielles ou logicielles qui peuvent altérer ou détruire les informations conservées.

Le matériel, réputé d'une grande fiabilité, subit néanmoins les agressions de son environnement et l'usure normale du temps. Le logiciel peut présenter plus de risques d'erreurs ou de pannes, surtout pendant une phase de mise au point. C'est cependant le logiciel, plus souple dans son utilisation, qui doit fournir les moyens de tolérer les pannes ou de les détecter et d'y remédier.

3.5.1 Fiabilité du matériel

Le matériel concerné ici est constitué du support passif de la communication, des moyens d'accès à ce support et des unités de stockage des informations. Le support de la communication et ses moyens d'accès peuvent être responsables de coupures fugitives ou permanentes de la communication, provoquant des interruptions du service assuré par les différents serveurs du réseau.

Les unités modernes de stockage d'informations, composées de disques magnétiques, présentent un point fort: elles constituent un moyen très fiable de conservation de l'information. Elles ont cependant deux points faibles : d'une part, il peut se produire, par suite de l'usure des matériaux, un écrasement des têtes de lecture qui détruit en tout ou partie les informations inscrites sur le support; d'autre part, ces unités de stockage sont sensibles aux coupures de courant, accidentelles ou provoquées, lorsqu'une écriture d'information est en cours.

En partant de l'hypothèse qu'une page d'information erronée est détectable, on peut remédier à ces deux faiblesses en gardant deux copies de la même page. Pour éviter la destruction des deux copies lors d'un atterrissage de tête de lecture, il suffit que ces copies se trouvent sur deux supports qui ont de très faibles probabilités de subir simultanément cet incident. Pour éviter la détérioration de l'information lors de sa réécriture durant une modification, il faut réaliser cette écriture comme une opération atomique. C'est à dire que l'écriture doit avoir lieu de manière correcte ou ne pas avoir d'effet. En prenant la précaution de toujours écrire les deux exemplaires d'une page dans le même ordre, on peut

savoir, après un incident, lequel garder pour disposer de l'ancienne ou de la nouvelle version. On peut rencontrer les cas suivants :

- les deux copies sont identiques : il n'y a rien à faire, l'opération est atomique car l'incident est survenu avant ou après les deux écritures,

- les deux copies sont correctes mais différentes, donc l'incident est survenu entre les deux copies : on écrit la première copie (nouvelle) dans la deuxième, l'écriture est terminée,

- l'une des copies est correcte, l'autre est erronée, l'incident est survenu pendant l'une des deux écritures : on écrit la copie correcte dans la copie erronée; si la première copie était erronée on retourne à l'ancienne version, sinon on aboutit à la nouvelle version.

Ce mécanisme doit être réalisé par un logiciel de bas niveau de gestion de l'espace de stockage. La tolérance ou la détection et la réparation des autres pannes est à la charge du logiciel qui gère les communications.

3.5.2 Reprise logicielle des pannes

Les logiciels peuvent présenter des pannes, c'est à dire souvent ne pas réagir correctement aux cas non prévus de comportement de l'environnement. Les logiciels doivent aussi détecter les incidents matériels pour éviter leurs conséquences, y remédier et reprendre leur déroulement.

Il faut exploiter les propriétés d'atomicité des transactions pour que, d'une part, elles s'exécutent avant ou après des transactions concurrentes afin qu'elles

n'induisent pas d'incohérence dans les informations manipulées, et que, d'autre part, en cas d'incident matériel, les transactions aient lieu complètement ou pas du tout.

Pour détecter l'absence d'un serveur ou d'un client, des délais de garde doivent être utilisés. Sur les réseaux locaux, ils sont plus courts et mieux délimités, ce qui représente une modeste performance par rapport aux grands réseaux. Les logiciels doivent être robustes vis à vis des interruptions de services ou de communication pour pouvoir reprendre et terminer une action en cours ou laisser en plan une transaction même longue (transfert de fichier, impression) et la poursuivre dès le retour aux conditions normales <Mitchell 82a>.

Un service ancillaire peut veiller à la bonne marche des machines et des systèmes. Il faut accroître le nombre de serveurs ancillaires de façon à ce que chaque composant veille sur un nombre limité de sites. Cela réduit le nombre de machines inaccessibles lors de la panne d'un serveur ancillaire <Herbert 81>.

Lorsque sur un site, le système d'exploitation entier d'une machine est tombé en panne, il faut pouvoir le redémarrer. Le système CMDS développé à l'université de Cambridge <Wilkes 80> propose la notion de serveur d'initialisation qui réalise le chargement initial de programme ou de système sur des machines disposant d'une mémoire morte contenant un programme de base qui permet l'initialisation.

A l'aide de ces outils, on peut concevoir un service relativement autonome qui peut se passer d'intervention humaine non pas pour les réparations nécessaires mais pour le redémarrage après celles-ci.

IV

SERVICES OFFERTS

POUR LA GESTION DES

INFORMATIONS

Les services, dont la nécessité et les fonctions ont été dégagées dans le chapitre précédent, vont être précisés ici selon leur nature, leur mise en oeuvre et leurs utilisations.

1 NATURE DES SERVICES

La distinction établie entre les services présentés tient surtout compte de leur utilisateur potentiel. L'utilisation directe par un usager peut se concevoir pour les services qui manipulent les fichiers d'information dans leur ensemble. L'utilisation indirecte, au moyen de primitives paramétrées accessibles à partir d'un programme, est envisageable pour tous ces services. Il faut distinguer cependant les services qui sont déjà accessibles directement à l'usager et ceux pour lesquels l'accès doit passer par un programme spécialisé: il s'agit surtout, pour ce dernier point, de l'accès aux informations dans leur détail pour consultation ou modification.

1.1 Utilisation directe

Les services concernés sont le transfert d'information, le stockage et la maintenance des fichiers d'information et l'impression distante. La manipulation de l'information par l'usager se fait au niveau des fichiers entiers: c'est l'entité que gère l'usager. Il peut utiliser directement ces services par l'intermédiaire d'un langage de commande. Le service concerné accepte un certain nombre de commandes et se trouve en mesure d'y répondre.

Selon la mise en oeuvre des services, leur utilisation pour atteindre et manipuler des fichiers à distance peut demander la connaissance des particularités des sites distants auxquels on accède. Dans la mise en oeuvre initiale des services, il fallait avoir une certaine connaissance des conventions distantes pour utiliser des services non locaux. Au contraire, on peut travailler à distance en utilisant une extension ou une adaptation des conventions locales.

Un usager d'un site A désirant travailler sur le site distant D peut à partir de A envoyer des commandes à D selon les conventions locales à D. Ou bien, D peut lui apparaître comme un prolongement du site A sur lequel il se trouve.

D'une manière plus générale, on peut, pour utiliser des services sur le réseau, se servir d'un langage de commande uniforme sur tous les sites. Il ne suffit pas alors de bien connaître les conventions locales et leurs extensions, mais il n'est plus nécessaire de connaître les autres sites.

1.2 Appel aux services par un programme

En fournissant des primitives paramétrées, on permet l'accès à ces services de l'intérieur d'un programme. Cela peut permettre de procurer aux usagers un programme qui donne un accès plus aisé aux données pour les consulter ou les modifier. Cela peut aussi permettre de réaliser un service plus général, et plus sophistiqué, pour simplifier encore le travail des usagers en supprimant les appels directs aux services de base qui seront alors effectués par le service général.

On peut ainsi concevoir un service qui prenne en charge le stockage et la gestion, le transfert et l'impression des informations sans que l'utilisateur ait à se préoccuper d'appeler chacun de ces services. Ceci peut être le rôle d'une application comme la messagerie.

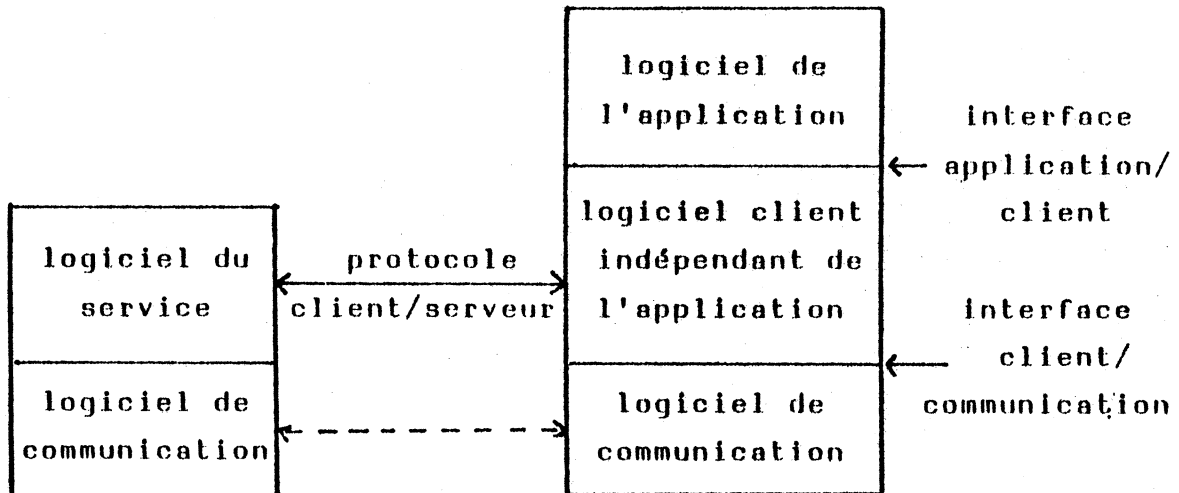
2 MISE EN OEUVRE DES SERVICES

On doit distinguer le service, entité logicielle se déroulant sur une ou plusieurs machines, et le serveur qui est le logiciel du service se déroulant sur une seule machine.

Un service est défini comme un ensemble de commandes et de réponses. Le service est réalisé par un ou plusieurs processus serveurs qui acceptent des messages, effectuent le travail demandé et répondent en conséquence. Le nombre de serveurs par service est déterminé par celui qui fournit le service et ne doit pas être visible des usagers du service.

Un client d'un service est aussi une entité logicielle qui peut opérer sous le contrôle d'un usager humain ou présenter une interface vers une application destinée à des utilisateurs humains.

Une application qui utilise un service de fichiers contient souvent une part de logiciel indépendante de l'application mais dépendante du service de fichiers. C'est ce logiciel que nous entendons par client. Pour atteindre le service, ce logiciel doit passer par l'intermédiaire d'une interface avec le service de communication.



Souvent, ce logiciel client, indépendant de l'application, est fourni par les réalisateurs du service de fichiers pour offrir un moyen d'accès à ce service partagé. Ce logiciel présente alors une interface vers l'application qui l'exploite.

2.1 Solution client-serveur

Dans cette optique on considère que le service est rendu par un ou plusieurs serveurs au bénéfice de processus clients. Les relations entre les serveurs et les clients passent par le logiciel de communication du réseau, elles doivent respecter un protocole de dialogue établi pour le service. La notion hiérarchique de client et de serveur est toute relative car, bien sûr, demander un service à un serveur peut l'amener à demander un service à un autre serveur. Le serveur de l'un est alors le client d'un autre <McLellan 80>.

2.1.1 Serveur

Le serveur doit comporter une partie qui joue un rôle d'interface vers les clients, reçoit les requêtes concernant les fichiers et dialogue avec les autres composantes du service. Une autre partie, qui effectue le travail de serveur, est orientée vers l'environnement local et traduit les requêtes reçues en primitives propres au système de gestion de fichiers local <Chambon 81>.

2.1.1.1 Accès au service

Pour être utilisable un service doit pouvoir être atteint grâce au moyen de communication du réseau. Un service doit donc offrir des portes d'accès : soit à travers les différents serveurs qui le composent, soit pour l'ensemble du service. Quand un processus client A veut communiquer avec un service B, A envoie un message à l'une des portes où B écoute. Pour cela A doit connaître ou être en mesure de trouver les portes en question. Si B est un serveur public, il fera largement connaître les noms de ses portes; mais si c'est un service privé, il gardera ses noms de porte secrets <AMOEBA 82>.

2.1.1.2 Rôle du serveur

Sur chaque site le serveur de fichiers est un moyen d'atteindre le système de gestion de fichiers local. Les serveurs, en tant que processus spécialisés, sont chargés de reconnaître et d'effectuer des opérations sur les fichiers locaux pour le compte de processus s'exécutant sur d'autres installations <Wupit 81>.

2.1.2 Client

Le client présente une interface vers une application ou vers un utilisateur humain. Dans les deux cas, il réalise un traitement local puis fait appel à des fonctions distantes en envoyant des requêtes aux services du réseau. Dans le cas où il constitue une interface homme-machine, il doit faciliter les échanges entre l'utilisateur humain et le service informatique. Le client représente alors le seul moyen d'accès au service. C'est à travers le client que le service sera jugé par ses utilisateurs.

2.1.3 Protocoles entre client et serveur

Lors de chaque échange ou transaction sur le réseau, un processus va se considérer comme client de l'autre processus qui se considère comme serveur. Une transaction peut se définir comme l'ensemble des échanges de messages nécessaires pour que le serveur apporte au client la réponse à sa requête. Le déroulement de la transaction sera soumis à un protocole prédéfini entre ce client et ce serveur.

Le protocole doit constituer un cadre pour les échanges entre client et serveur. Il doit tenir compte des divers facteurs mis en jeu par le client et le serveur. Dans ce qui suit, on décrit deux types courants de protocole de déroulement de transaction, on examine la possibilité de réitérer des requêtes insatisfaites et on définit les informations à conserver, par chacun des partenaires, sur le contexte de la transaction.

2.1.3.1 Conditions sur le protocole

Le modèle de protocole doit tenir compte de deux facteurs : Le contrôle d'erreurs et le contrôle de flux. Il doit permettre de détecter les erreurs de transmission et d'adapter le vitesse d'émission à la vitesse de réception et de traitement des requêtes.

2.1.3.1.1 Contrôle d'erreur

Quels que soient les cas d'erreur le client et le serveur doivent pouvoir les tolérer. Les protocoles doivent aussi être conçus pour supporter les interruptions intermittentes ou fugitives de la communication entre serveur et client.

2.1.3.1.2 Contrôle de flux

Le contrôle de flux doit être exercé par le serveur pour éviter de dépasser ses capacités en ressources partagées. Le serveur peut tout simplement ignorer une requête qu'il ne peut satisfaire ou avorter le traitement d'une requête locale dans la mesure où le client, au bout d'un certain temps, la réitérera <Mitchell 82a>.

2.1.3.2 Modèles de protocole

Pour un type courant de transaction, le déroulement du protocole se réduit à une requête que le client adresse au serveur; celui-ci répond par un seul message. Le client envoie sa requête au serveur, avec l'adresse de retour, et

attend la réponse. Le serveur doit réagir aux commandes qui lui parviennent en lançant les actions demandées et en retournant en réponse les résultats.

Le client peut ne pas recevoir la réponse qu'il attend, c'est à dire qu'au bout d'un certain temps il n'a pas reçu de réponse. Cela peut être dû à des erreurs de transmission, sur la requête ou la réponse, ou bien à la surcharge du serveur qui n'a pas pu répondre à temps à la requête. Pour tenter de résoudre ces problèmes, divers protocoles de dialogues ont été proposés, nous en analysons les trois principaux. Le premier est celui que nous venons de voir, sans plus. Le second admet des actions répétables, c'est à dire qu'une requête peut être réitérée sans dommages secondaires. Dans le troisième, le client doit en outre acquitter la réponse du serveur.

2.1.3.2.1 Caractère répétitif des requêtes

Pour qu'une même requête puisse être répétée sans effet secondaire déplorable, il faut que les opérations réalisées par le serveur soient idempotentes. L'effet d'une requête répétée doit être le même que celui de la requête arrivant seule et la réponse doit être la même.

Par exemple, une action "Ajouter(fichier, données)", qui ajoute des données à un fichier, n'est pas répétable car elle aboutirait à allonger le fichier. Par contre les actions "FixerLongueur(fichier, longueur)" qui attribue une longueur à un fichier et "Ecrire(fichier, position, données)" qui écrit des données dans un fichier à partir d'une position sont répétables et auront toujours le même effet si elles sont répétées. La séquence d'actions "FixerLongueur", "Ecrire" réalise la même chose que l'action "Ajouter" tout en étant répétable.

2.1.3.2.2 Protocoles à deux ou trois phases

Le protocole à deux phases comporte une requête suivie d'une réponse. Dans celui à trois phases, le client doit acquitter la réponse du serveur. On suppose que ces protocoles admettent la répétition des requêtes. On va s'intéresser aux informations que chaque partenaire doit conserver sur les transactions en cours pour éviter de recommencer le travail lorsqu'une requête est répétée.

Dans le protocole à deux phases qui admet la répétition des requêtes, le client attend un délai de garde et renouvelle sa demande s'il n'a pas reçu de réponse. Le serveur n'est donc pas tenu de le prévenir en cas de trouble de son côté. Il n'a même pas à retenir le contexte de la transaction en cours, puisque si le client ne reçoit pas la réponse, il répétera sa question. C'est donc aux clients de conserver le contexte des transactions en cours.

Dans le protocole à trois phases, c'est le serveur qui conserve le contexte de la transaction jusqu'à ce qu'il reçoive l'acquiescement du client. Ainsi, le serveur peut immédiatement répondre à une demande répétée puisqu'il en aura gardé la réponse. Si au bout d'un certain temps, le serveur ne reçoit pas d'acquiescement, il peut oublier le contexte de la transaction. En effet, soit le client a acquiescé la réponse et l'acquiescement s'est perdu, soit le client a abandonné la transaction et ne donne plus signe de vie <Mitchell 82a>.

Du point de vue du serveur, il existe une différence entre ces deux protocoles : dans celui à deux phases, le serveur va employer son temps à recalculer la réponse à une demande réitérée, alors que dans celui à trois phases, il

va employer de l'espace en mémoire pour stocker la réponse et gérer un délai de garde pour pouvoir répondre sans traitement supplémentaire à une question déjà posée.

Le client, pour sa part, doit toujours gérer un délai de garde en attendant la réponse du serveur. Dans le cas du protocole à trois phases, il doit en outre acquitter la réponse reçue pour en décharger le serveur; s'il réitère sa demande, il a aussi l'assurance qu'il recevra plus vite la réponse au cas où c'est la réponse qui a été perdue lors de sa transmission.

Cette organisation entre client et serveur peut être étendue aux transactions comportant plus d'un échange de message dans la mesure où la transaction entière est rendue idempotente. Cette transaction doit aussi être assez rapide pour que le client puisse disposer d'un délai de garde au bout duquel il peut réessayer sans provoquer une situation instable, où de fréquents essais surchargent le serveur entraînant des essais supplémentaires.

2.2 Interface et service partagé

Une façon d'envisager le modèle client serveur est de concevoir le client comme offrant des primitives d'accès au service pour un programme. Le serveur est conçu pour réaliser ces primitives par appel de procédures réentrantes ou d'une procédure avec une file d'attente des requêtes. Lorsque le client est fourni par les responsables du service, il peut être présenté aux applications de deux façons : 1) le code du client est intégré à chaque application, ou 2) le client gère plusieurs applications utilisatrices.

2.2.1 Serveur

Le service peut être réalisé comme un ensemble de processus lancés chacun pour répondre à une commande distante et s'achevant à la fin de l'exécution de cette commande. Ces processus, exécutant des procédures réentrantes, utiliseront le même code avec des données différentes propres à chacune des commandes. Le même service peut aussi, en offrant des primitives à des niveaux différents, employer des primitives d'un niveau plus bas pour réaliser des primitives plus sophistiquées. Le serveur de fichiers, avec les primitives qu'il offre, doit pouvoir être utilisé pour bâtir un autre service de fichiers.

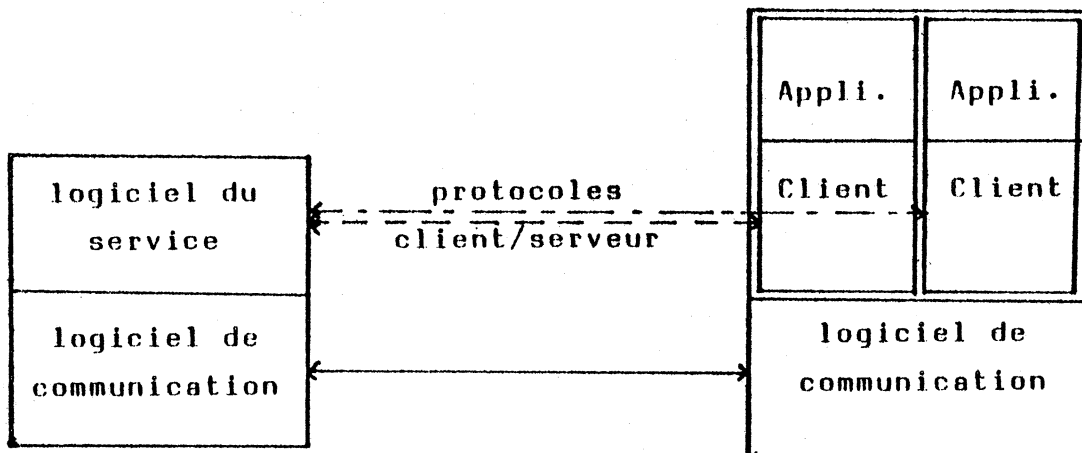
2.2.2 Client

Le client peut se présenter comme une librairie de routines que les usagers ou les applications peuvent appeler pour obtenir toute la gamme de services offerts par les systèmes classiques : ouverture, fermeture, création, destruction de fichier, lecture, écriture, positionnement dans le fichier. En fait, il peut représenter un module de librairie qui émule les appels à différents systèmes d'exploitation. Le client constitue aussi une interface vers l'application qui montre de façon plus abstraite les possibilités offertes à un niveau plus bas.

2.2.2.1 Intégration à l'application

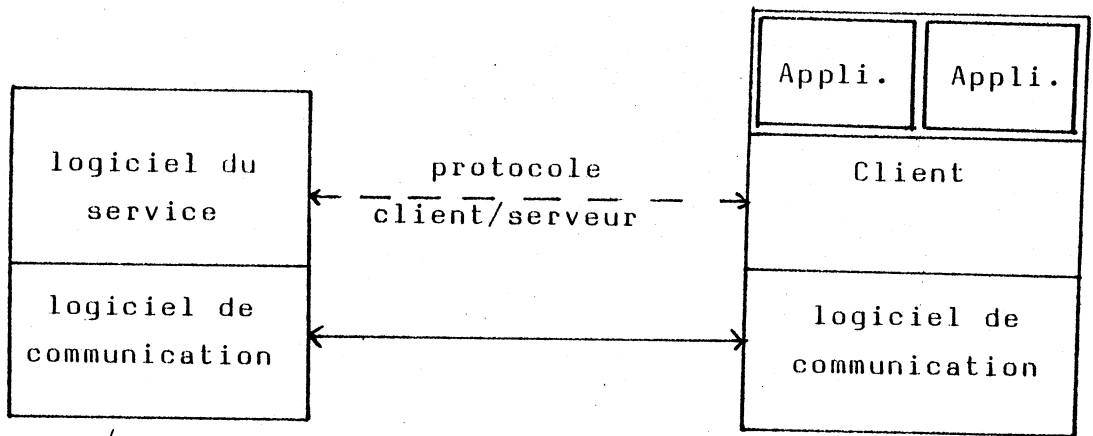
C'est le passage le plus naturel d'un client présentant une interface homme-machine à un client offrant des primitives accessibles par un programme. Les possibilités de l'interface sont réalisées à l'aide de primitives

paramétrées qui rendent le service accessible aux applications. Le code du client se trouve intégré à l'application, même s'il en constitue une partie distincte. Lorsque plusieurs applications ont affaire à des clients, dont le code est intégré au leur et qui dérivent de la même interface homme-machine, il y a répétition dans chacune de ces applications du code de tous ces clients. Sur des sites de petite configuration, l'intégration du code à l'application est une bonne solution car ces sites, le plus souvent, ne peuvent supporter simultanément qu'une ou deux applications.



2.2.2.2 Client multi-usagers

Sur des sites qui permettent une multiprogrammation importante, il devient intéressant d'offrir un service partagé par tous les utilisateurs : le code du client ne sera pas répété, seules les variables propres au client seront répétées pour chaque utilisateur. Mais on devra ajouter à chaque utilisateur un peu de code pour accéder au service partagé, ce qui diminue le gain de place. Pour cette raison, cette solution n'est valable que pour un nombre important d'utilisations simultanées d'un même service à travers le même client.



3 EXEMPLES D'UTILISATION PAR LES APPLICATIONS

Nous allons montrer comment les services de base en matière de gestion des informations sur le réseau local sont utilisés par les différentes applications développées à un plus haut niveau.

Ces applications sont d'abord l'utilisation directe des possibilités de transfert de fichiers et de gestion de catalogues, puis le service d'imprimerie distante et enfin le service de messagerie.

3.1 Transfert de fichiers et gestion de catalogues

Le service de fichiers est conçu à la fois comme un service de base utilisé par les autres applications et comme un service offert directement aux utilisateurs finaux du réseau.

En tant que service de base, sa fonction est de permettre à toute application de transférer des volumes de données relativement importants dans de bonnes conditions. Ces données sont supposées résider, à la source comme à la destination, dans des fichiers. L'application utilisatrice est alors déchargée des tâches d'accès aux fichiers et de transfert de leur contenu.

En tant que service utilisateur, le service de fichiers doit d'abord permettre le transfert de fichiers entre sites du réseau, sous le contrôle de l'utilisateur. Il doit également permettre aux utilisateurs de gérer à distance leurs fichiers, c'est à dire de lister des catalogues, détruire des fichiers, de les renommer, les copier, etc. Le service actuellement mis en place se limite à un service de transfert et de gestion de fichiers. Il n'est pas envisagé

d'offrir une méthode d'accès sur le réseau qui permettrait à un programme se déroulant sur un site de faire des entrées/sorties sur des fichiers résidant sur d'autres sites. Le principal obstacle à cette réalisation sur le réseau actuel est le goulot d'étranglement que constitue le passage par une ligne série V24 pour accéder au support de communication. Le ralentissement qu'il provoque rend déjà le transfert de fichiers particulièrement poussif, la méthode d'accès à des fichiers distants en pâtirait d'autant plus. Aussi, nous avons préféré la solution où les actions de modification des fichiers sont exécutées à distance.

Pour le transfert de fichiers, une solution générale mettant en jeu trois sites aurait consisté à séparer le contrôle du transfert proprement dit. Un site contrôleur, site de l'utilisateur aurait supervisé deux autres sites, le site émetteur sur lequel se trouve le fichier à transmettre, et le site récepteur qui doit recevoir le fichier. L'avantage d'une telle solution est que l'utilisateur peut, depuis n'importe quel site du réseau, demander un transfert de fichier entre deux sites quelconques. Nous avons rejeté cette solution pour deux raisons: elle n'est pas vraiment nécessaire et elle introduit une complexité importante dans les protocoles. De plus elle n'apporte aucun avantage en ce qui concerne la gestion à distance des fichiers.

Chaque utilisateur du service de fichiers met donc deux sites en jeu : le site client et le site serveur. Le transfert ne peut avoir lieu qu'entre ces deux sites, mais il est possible dans les deux sens. Ceci nécessite donc que l'utilisateur soit présent sur l'un de ces deux sites. L'approche que nous avons adoptée est celle où chaque utilisateur dispose de son propre poste de travail (un LSI/11-2 dans notre réseau) où il peut profiter de services

locaux. Le réseau lui permet d'accéder à des serveurs spécialisés (serveur de fichiers, messagerie, imprimerie) et de communiquer avec les autres postes de travail. Dans ces conditions, les transferts de fichiers se font essentiellement entre les postes de travail et les serveurs. L'organisation à deux sites est donc tout à fait suffisante, et si l'on a besoin d'un transfert entre deux sites distincts du site de travail, l'opération est possible grâce à la messagerie.

3.2 Service d'imprimerie

Pour des raisons économiques évidentes, toutes les machines du réseau ne disposent pas de leur propre imprimante; pourtant les utilisateurs de toutes les machines ont besoin de faire imprimer des informations. Le service d'imprimerie permet de répondre à cette nécessité.

Ce service utilise le service de transfert de fichiers. L'opération se déroule en trois temps. Une application, s'exécutant sur un site dépourvu d'imprimante ou ne possédant pas l'imprimante requise, produit la liste à imprimer dans un fichier local. Le fichier est alors transmis vers le site "imprimeur" puis listé sur l'imprimante correspondante.

La première phase ne présente pas de difficulté: le système RSX-11 des LSI-11 permet à toute tâche d'effectuer ses sorties d'informations indifféremment sur terminal, imprimante ou fichier. La deuxième phase, le transfert de fichier, est prise en charge par le service précédemment décrit. Le client du service de transfert de fichiers et de gestion de catalogues a été intégré au client du service d'imprimerie distante dont il constitue un sous-ensemble. Ce client reçoit une commande lui demandant de transférer

le fichier vers le site du serveur d'impression. Il envoie donc le fichier sur ce site.

La troisième phase a demandé un développement plus spécifique. Un service local de gestion des demandes d'impression a été intégré au serveur de fichiers. Le serveur résultant de cette intégration enregistre toutes les demandes d'impression aussi bien locales que provenant d'un site distant. Il gère l'imprimante et imprime les fichiers enregistrés les uns après les autres.

3.3 Messagerie

Un service de messagerie <PAPYRUS 80>, <Quint 80> a été mis en place pour permettre les communications de messages de texte entre utilisateurs du réseau. Le service le plus simple permet d'envoyer immédiatement une ligne de texte qui sera remise au correspondant s'il est sur un terminal du réseau. Mais il est aussi possible d'envoyer des textes plus longs qui sont stockés dans la boîte à lettres du correspondant et ne sont délivrés que sur sa demande.

Ces textes plus longs sont contenus dans des fichiers et leur circulation sur le réseau utilise le service de transfert de fichier directement accessible à cette application. Chacune des composantes du service de messagerie contient un ensemble de procédures lui permettant d'échanger un fichier avec une autre composante sur la liaison qui les relie. L'appel au service de transfert de fichiers se réduit à l'appel d'une procédure paramétrée qui réalise le transfert du fichier indiqué en déchargeant le service de messagerie de l'accès à ce fichier et du contrôle du transfert. Lorsque la procédure de transfert se termine, le service de messagerie reprend le contrôle en sachant que le transfert a eu lieu.

V

REALISATIONS

Après avoir, dans les chapitres précédents, défini le contexte général de la gestion et de la communication d'information dans les systèmes répartis, nous présentons l'étude et la réalisation d'un service de gestion et d'échange d'information sur le réseau, et la façon dont il est utilisé par les applications qui exploitent les services qu'il propose.

Les réalisations que nous présentons ont eu pour but de mettre en oeuvre les services proposés et de permettre la validation des concepts dégagés.

Les maquettes de service ont été réalisées sur le réseau local DANUBE de l'IMAG. Les sites sont constitués essentiellement de LSI-11/02 et /23 dotés du système multi-utilisateur en temps partagé RSX11-M. Les programmes ont été développés en PASCAL. Ils comportent des insertions d'instructions en assembleur ou en macro-assembleur pour réaliser les fonctions d'entrée-sortie et de synchronisation que ne réalise pas le langage PASCAL. L'une de ces maquettes a été transportée sur les burovisseurs <Naffah 79> développés à l'INRIA et adaptée à son environnement en vue des journées de démonstration du projet KAYAK <Naffah 80> en mars 1981.

Dans ce chapitre nous présentons succinctement les applications réalisées : le service de gestion, de transfert et d'impression de fichiers, l'interface de transfert de fichiers pour applications et l'adaptation pour les démonstrations KAYAK du premier service cité. Ensuite nous introduisons la mise en oeuvre du schéma Client-Serveur sur la base duquel ont été bâtis les services présentés. Puis nous présentons un service,

organisé suivant le schéma client-serveur, qui permet aux usagers de gérer les catalogues de fichiers sur le réseau, de transférer des fichiers d'un site à l'autre ou de faire imprimer des fichiers sur une imprimante du réseau <ADI-TIF 81>. Ensuite nous présentons une interface conçue de façon symétrique qui, intégrée aux programmes utilisateurs, leur permet d'appeler des primitives de transfert de fichiers. Enfin nous proposons un protocole permettant la reprise d'un échange d'information après une panne survenue sur le réseau.

1 APPLICATIONS REALISEES

Les services que nous avons mis en oeuvre ne constituent que pour une part des applications. En fait, pour une large part, nos services ont facilité la réalisation d'applications plus générales sur le réseau.

La première application est un service d'imprimerie qui permet à tout utilisateur du réseau de faire imprimer un document sur une imprimante du réseau dans le cas où la machine qu'il utilise ne possède l'imprimante nécessaire.

La deuxième application est un service de messagerie qui offre aux utilisateurs du réseau la possibilité de communiquer entre eux par des messages de texte.

Ces deux applications peuvent donner lieu à des transferts d'information de volume important qui résident pour la plupart, à la source comme à la destination, dans des fichiers. Un service de transfert et de gestion des fichiers a été développé, il peut être utilisé en tant que tel par des usagers humains, mais les deux applications précédentes l'utilisent également.

Pour l'utilisation par des usagers humains, l'application réalisée permet sur le réseau le stockage, la gestion, le transfert et l'impression de fichiers. Pour son utilisation par l'application de messagerie, le service réalisé prend la forme d'une interface qui se charge de l'accès aux fichiers et de leur transfert d'un site à l'autre. L'application adaptée pour les journées KAYAK est constituée d'un serveur d'impression sur le réseau de l'INRIA pour les burovisseurs qui sont en principe dépourvus de moyen d'impression.

1.1 Service de stockage, de gestion, de transfert et d'impression de fichiers

Ce service est réalisé par la collaboration d'un client et d'un serveur. Le serveur fonctionne sur un site du réseau disposant d'espace de stockage et d'une imprimante, et peut servir un client qui a établi une liaison avec lui. Sur cette liaison, le serveur peut recevoir et répondre à trois types de commandes : émission ou réception d'un fichier, impression d'un ou plusieurs fichiers, état du catalogue ou destruction d'un ou plusieurs fichiers. Le premier type de commande entraîne un assez long échange entre le serveur et le client; les deux derniers donnent lieu à une réponse assez courte du serveur. Le client mène le dialogue avec l'utilisateur et traduit ses requêtes en commandes pour le serveur. Le client établit la liaison avec le serveur qu'indique l'usager, lui communique les ordres de l'usager et transmet à ce dernier la réponse du serveur.

1.2 Interface de transfert de fichiers par programme

Ce service est constitué d'un module de programme contenant trois procédures paramétrées. L'appel à ces procédures par un programme d'application lui permet de faire effectuer le transfert d'un fichier en se déchargeant du contrôle du transfert et de l'accès aux fichiers. L'interface doit être intégrée à chaque composante locale d'une application répartie, elle rend alors possible l'échange de fichiers sur une liaison établie entre deux des composantes. Une des procédures de l'interface débute le transfert, la seconde le continue au fur et à mesure puis le termine, la troisième gère les événements qui surviennent sur la liaison durant le transfert.

1.3 Serveur d'impression sur le réseau de burovisseurs

Dans le cadre des journées de présentation du projet KAYAK en mars 1981, une version antérieure du serveur présenté plus haut a été adaptée pour jouer le rôle de serveur d'imprimante sur le réseau de burovisseurs de l'INRIA. Le serveur a été modifié pour gérer une imprimante électrostatique très rapide (39400 Bds). Des clients transportés sur les burovisseurs pouvaient échanger des fichiers avec le serveur, lui demander l'impression de fichiers ou l'état de catalogues. Malgré le fonctionnement correct de l'ensemble, le faible débit de l'interface d'accès au réseau, qui rendait le transfert d'un fichier bien plus lent que son impression, a rendu cette réalisation assez peu démonstrative.

2 LE MODELE CLIENT SERVEUR

La définition de l'architecture des applications ou des services a été guidée par deux principes : la prise en compte des capacités limitées des machines utilisées et le respect du modèle d'architecture des systèmes ouverts proposé par l'ISO. Cela a conduit à simplifier certaines couches de ce modèle d'architecture, conçu au départ pour de grands réseaux et de gros ordinateurs, afin de l'appliquer à un réseau local de mini- et de micro-ordinateurs.

Les applications et les services développés ont été organisés selon le schéma client-serveur. Plusieurs serveurs et clients de services différents peuvent être actifs simultanément sur le réseau, ou sur une même machine. Ceci offre une grande souplesse pour la mise en place des services.

Les protocoles entre client et serveur, le fonctionnement général de ceux-ci et les fonctions offertes par les services sont présentés dans ce qui suit. Puis nous aborderons, lors de la phase initiale de prise de contact du client avec le serveur, les moyens de localiser le service par le client dans le cas où la localisation du serveur a une importance pour le client ou son utilisateur.

Sur un ou plusieurs sites les serveurs sont à l'écoute sur le réseau. Ils attendent les demandes de service que leur adressent les clients. Ces clients sont des programmes qui sont activés à la demande des utilisateurs et s'exécutent sur la machine de chaque utilisateur. Ce sont eux qui prennent en charge le dialogue avec le serveur suivant les directives de l'utilisateur.

Pour chaque application, le client et le serveur ont été définis de sorte que les clients puissent s'exécuter sur les plus petites machines du réseau (LSI-11/2, 64 Koctets de mémoire, disque souple). Les serveurs, eux, s'exécutent sur de plus grosses machines (LSI-11/23, 128 ou 256 Koctets, disques durs); cependant, le serveur fichier, dans une configuration assez simple, est capable de s'exécuter sur les petites machines pour pouvoir être utilisé ultérieurement par d'autres applications.

De même, les protocoles ont été définis pour être implémentables sur des machines de faible capacité. Ils sont donc très simples et tentent d'utiliser au mieux les services offerts par les couches inférieures de l'architecture.

2.1 Client : interface de dialogue Homme-Machine

Le rôle du client est de dialoguer avec son utilisateur et de transmettre ses commandes au serveur qu'il a désigné. Il peut aussi accéder à des fichiers pour les envoyer ou les recevoir à la demande de l'utilisateur. Nous présentons ici ce qui caractérise le client c'est à dire son dialogue avec un utilisateur.

Le programme client est lancé par un usager qui désire travailler sur le réseau. Le client demande à l'usager son identité et s'abonne au service de communication sous ce nom. Ensuite le client lui propose ses services. Selon le choix de l'usager, le client va diffuser une demande de liaison avec un des serveurs concernés par ce service. Le premier serveur qui répond à l'appel rend le service demandé. Selon le service demandé, un serveur caractérisé par sa situation géographique sur le réseau se révélera

plus approprié. A la demande de l'utilisateur, ou de sa propre initiative, le client sélectionnera le serveur parmi ceux qui répondent. Le client mènera ensuite le dialogue en fonction des questions et des réponses attendues par le serveur.

Le client constitue l'intermédiaire entre le service et l'utilisateur. C'est à travers lui que sera jugé le service et les possibilités ou facilités d'usage que le service peut offrir. La présence d'utilisateurs non informaticiens et la nécessité de démontrer un usage simple, efficace et pratique conduisent à soigner la présentation et les moyens d'utilisation des produits développés initialement comme maquettes de recherche. La qualité du service semble ainsi améliorée alors que la base du service reste la même. Il s'agit plus alors du développement de produits finis dans une perspective ergonomique que de recherche de nouvelles idées de service.

2.2 Serveur

Ce programme constitue sur le réseau un point de service pour le transfert de fichiers ainsi que pour l'impression de fichiers, la gestion de catalogues et l'accès aux fichiers.

2.2.1 Impression

Le serveur d'imprimerie distante a été réalisé en intégrant au serveur de transfert de fichiers un service local d'imprimerie en différé. Ce serveur local permettait de constituer une liste de fichiers à imprimer tout en poursuivant l'impression des fichiers déjà rangés dans cette liste. Le serveur résultant de cette fusion peut à la

fois recevoir par transfert des fichiers à imprimer, ranger leurs coordonnées dans la liste et les imprimer au fur et à mesure de la disponibilité du matériel.

2.2.2 Catalogues

Tous ces services de gestion de catalogues peuvent être requis à partir d'un terminal ou par un programme client car ils ne nécessitent pas de capacité particulière de stockage chez leur utilisateur. Ces services permettent de gérer les catalogues de fichiers du site serveur. Ils constituent un ensemble minimal de commandes de gestion de fichiers à distance. Il est possible de lister un catalogue, de détruire un ou plusieurs fichiers, de visualiser le contenu d'un fichier ou d'en changer le nom.

2.2.3 Accès aux données

Le client et le serveur doivent pouvoir échanger n'importe quel type de fichier. L'accès à ces fichiers se fait grâce aux primitives du système RSX. Ces primitives insérées dans des procédures en PASCAL permettent l'ouverture des fichiers en lecture ou en écriture, la lecture ou l'écriture d'enregistrements séquentiellement et la fermeture d'un fichier.

Les fichiers sont lus ou écrits en entier. Il n'est pas possible d'ajouter un fichier à un autre ou d'extraire une partie d'un fichier. La structure d'un fichier est transportée intacte grâce à la récupération des paramètres de son organisation interne dans le descripteur du fichier ouvert et à leur transport dans le descripteur du fichier créé.

2.3 Localisation du service

Le service de transport développé sur le réseau local DANUBE permet de communiquer avec une entité sur le réseau indépendamment de sa localisation. Une entité, un serveur ou autre, s'abonne au service de transport sous un nom; on peut ensuite communiquer avec elle en utilisant ce nom. Plusieurs serveurs équivalents peuvent s'abonner sous le même nom et l'un quelconque d'entre eux pourra rendre le service demandé.

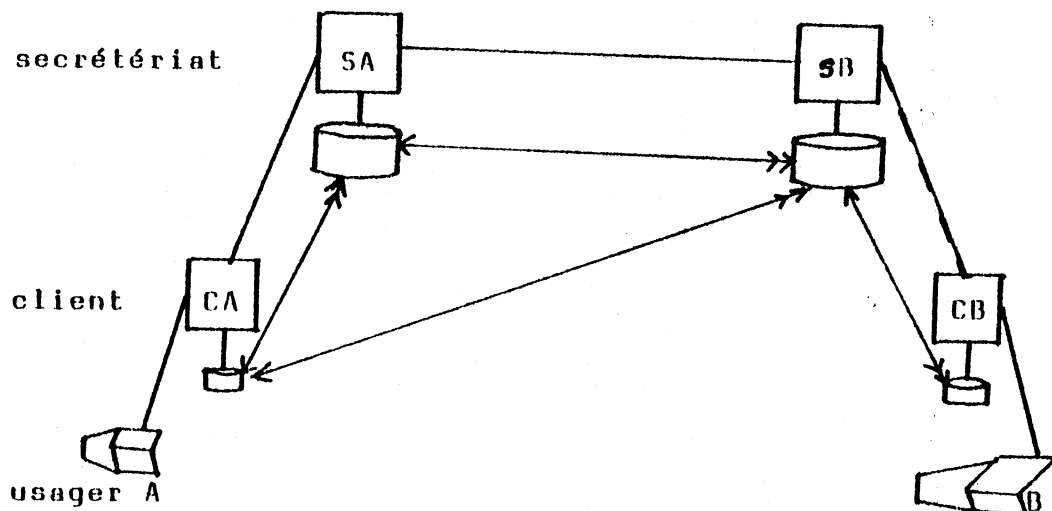
Il est des cas où la connaissance de la localisation géographique d'un serveur ou d'une entité peut être nécessaire, par exemple dans le cas d'un serveur d'impression, pour déterminer où retirer sa liste. Pour cette raison le service de transport offre la possibilité de connaître un nom caractérisant le site sur lequel se trouve l'abonné. Ce nom contient l'indication de la salle ou du bureau où se trouve la machine sur laquelle fonctionne le serveur et une indication sur la machine elle-même si il y en a plusieurs dans le local indiqué. Le nom porte ces indications en clair pour l'utilisateur.

On peut illustrer cette nécessité par un autre exemple qui est l'application de messagerie. Cette application est composée de secrétariats prenant en charge leurs abonnés à partir de postes clients. Chaque secrétariat gère un certain nombre d'abonnés qui lui sont rattachés. Ceux-ci utilisent les clients pour accéder au service de messagerie.

Lorsqu'un abonné A veut faire parvenir un long message à un correspondant B, il fournit ce message dans un fichier au client CA qui l'a pris en charge. Le client CA, pour le

transmettre au correspondant B, remet le fichier à son secrétariat SA. Le secrétariat SA le fait parvenir à travers son client CB à l'abonné B s'il est présent ou à son secrétariat SB s'il est absent.

Dans le cas où les secrétariats SA et SB sont distincts, considérons l'émission d'un message de A vers B. Si B est présent, le message doit aller du client CA de A au client CB de B. En l'absence de B, le message va vers le secrétariat SB de B pour y être conservé. Or CA ne connaît ni CB ni SB, il doit s'adresser à SA. SA indique à CA la localisation de SB ou CB. CA demande alors au service de transfert de fichiers d'y transférer le message.



Transfert de fichiers pour l'application de messagerie

Il se pose le problème de désigner les sites impliqués dans le transfert. Or seuls les noms des entités concernées sont connus. En effet les entités composant le service de

messagerie se connaissent entre elles par leurs noms. Elles s'adressent au service de transfert de fichier pour acheminer les fichiers à transférer. Ce service se base sur les noms des serveurs qui le composent. Il faut donc assurer une concordance entre les noms des serveurs de messagerie et les noms des serveurs de transfert de fichiers. Pour cela on peut se baser : soit sur le moyen de communication et préfixer chaque nom par l'adresse du communicateur, soit définir un nom qui caractérise le site et confier ce nom au service de transport.

C'est cette dernière solution qui a été adoptée. Le service de transport connaît le nom du site sur lequel il exerce et le fournit aux applications abonnées à leur demande. Les serveurs se font connaître sous un nom en rapport avec le service qu'ils fournissent mais peuvent préciser à la demande leur localisation. Ainsi dans le cas de la messagerie, le client CA fournit au service de transfert de fichiers les coordonnées du fichier et le site de CB ou de SB où il doit le transporter.

3 SERVICE DE GESTION, DE TRANSFERT ET D'IMPRESSION DE FICHIERS

La mise en oeuvre de ce service est fondée sur un dialogue entre le client et le serveur. Nous présentons le protocole de dialogue entre le client et le serveur, ainsi qu'une étude des algorithmes de compactage des données échangées.

3.1 Protocole de dialogue et de transfert de fichiers

Sur les LSI-11, des programmes serveurs et clients ont été développés pour mettre en oeuvre un protocole de transfert, de gestion et d'impression de fichiers. Le programme client dialogue avec son utilisateur local et traduit ses requêtes en commandes transmises au serveur désigné par l'utilisateur. Le serveur exécute ces commandes, conformément au protocole. Ces commandes peuvent demander l'envoi ou la réception d'un fichier, la situation d'un catalogue, la destruction d'un fichier ou l'impression d'un fichier en différé sur le site du serveur.

Ces services sont tous offerts par le même serveur qui tourne sur un LSI-11 en accédant au service de transport. Ces services peuvent être requis par des commandes provenant d'un programme client, qui tourne sur un autre LSI-11 en accédant au service de transport, ou un terminal accédant directement au réseau. Ce dernier peut être géré par un communicateur ou par un LSI-11 grâce à un programme de gestion de terminal. Le protocole permet de distinguer les services offerts par le serveur selon la nature du client. Un programme client a accès à l'ensemble des services, un terminal ne peut envoyer que des commandes de gestion de catalogues ou d'impression de fichiers.

L'émission et la réception de fichiers ont lieu entre des systèmes homogènes et des matériels compatibles; ils préservent la structure des fichiers durant le transfert et permettent l'échange de fichiers de toutes sortes. Le transfert a toujours lieu à l'initiative du client qui demande à établir une liaison avec le serveur. Cela implique des différences dans l'exécution du protocole pour les commandes d'émission et de réception. Ce protocole est constitué de trois parties dont la première et la dernière diffèrent selon le sens du transfert.

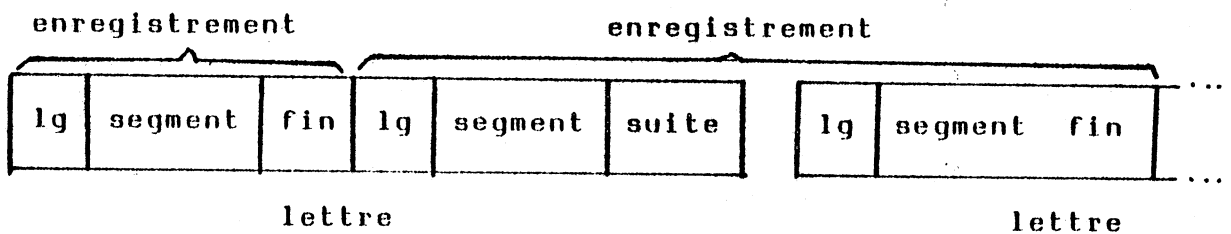
3.1.1 Transfert du client vers le serveur

La première partie du protocole est un dialogue entre le client et le serveur pour échanger des informations sur les caractéristiques et la structure interne du fichier à transférer : le nom et la localisation du fichier, sa taille, le format et les attributs des enregistrements.

Pour satisfaire une requête d'émission, le client ouvre le fichier local, en extrait les paramètres de sa structure interne et envoie la commande d'émission au serveur, lui fournissant le nom et la structure du fichier qu'il doit créer. Le serveur confirme qu'il est prêt dès qu'il a ouvert le fichier.

Durant la deuxième partie, les enregistrements du fichier sont transmis du site source au site destinataire sur la liaison établie entre client et serveur. Le client lit les enregistrements du fichier local comme une suite d'octets, les transmet sous forme de lettres sur la liaison au serveur qui reconstitue et écrit les enregistrements dans son fichier.

Les lettres qui transportent les enregistrements sont remplies au maximum de leur capacité. Elles peuvent, suivant leur taille, contenir des enregistrements entiers ou des segments. L'interprétation de la structure des lettres consécutives permet le transport d'enregistrements de n'importe quelle taille grâce à leur fragmentation et à leur réassemblage. Les lettres sont constituées d'un ou plusieurs segments d'enregistrement, chacun d'eux contient le nombre d'octets du segment, les octets du segment et un code indiquant si ce segment est ou non le dernier de l'enregistrement auquel il appartient. Une troisième valeur de ce code indique la fin du fichier.



Dans la partie finale du protocole, à la réception de la marque de fin de fichier, le serveur ferme son fichier et le confirme au client. Le client informe l'utilisateur de la terminaison de sa requête.

3.1.2 Transfert du serveur vers le client

La requête de réception de fichier implique une phase initiale plus longue car le client ne connaît pas la structure du fichier qu'il doit recevoir. Le client envoie au serveur la commande de réception en lui donnant le nom du fichier sur le site du serveur. Celui-ci ouvre le fichier et en renvoie les paramètres de structure au client qui peut créer le fichier de son côté et le confirmer au

serveur. Le serveur envoie ensuite les enregistrements au client de la même façon. Quand le client reçoit le dernier segment du dernier enregistrement, il ferme le fichier reçu et informe l'utilisateur de la terminaison de sa requête.

3.1.3 Cas de mauvais fonctionnement

Ce protocole de transfert de fichiers ne prévoit pas l'échange de points de reprise et de leurs acquittements. Une difficulté est signalée par le partenaire qui la rencontre et le transfert est annulé de part et d'autre et, de même que dans le cas d'une panne du service de transport, le fichier source est fermé et le fichier destinataire est détruit. L'étude d'une procédure de reprise est donnée à la fin de ce chapitre.

3.2 Compression de données

La compression consiste à regrouper des octets consécutifs identiques pour les représenter de manière condensée. Elle peut intervenir à deux niveaux et être mise en oeuvre de plusieurs façons.

3.2.1 Choix du niveau où a lieu la compression

La compression peut intervenir sur les lettres confiées au service de transport ou sur les enregistrements lus dans le fichier à transférer. Le choix dépend de l'efficacité qui en résulte en volume.

3.2.1.1 Compression au niveau du transport

Un premier niveau d'action est la lettre émise sur la liaison pour le transfert des enregistrements. Une fois constituée par remplissage avec des segments d'enregistrement, la lettre, avant son émission, est analysée pour permettre le tassement des octets consécutifs identiques. Cette facilité pourrait être offerte par le service de transport mais elle ne l'est pas. Néanmoins, on peut considérer qu'elle est offerte au niveau du service de transport. Puisque l'on comprime une lettre juste avant son émission, à la réception d'une lettre on doit procéder à sa décompression avant toute analyse de son contenu.

3.2.1.2 Compression au niveau de l'enregistrement

Le second niveau d'action est plus éloigné du service de transport, il intervient sur l'enregistrement lu dans un fichier. Après lecture d'un enregistrement, on procède au tassement des octets consécutifs identiques. Puis l'enregistrement est livré au découpage en segments placés dans les lettres pour émission. A l'arrivée chez le récepteur, les enregistrements sont reconstitués sous leur forme compressée par regroupement de leurs segments. On procède alors à leur décompression avant de les écrire dans le fichier récepteur.

3.2.1.3 Comparaison de ces deux niveaux

Avant même de considérer les algorithmes de compression à mettre en oeuvre, on peut émettre quelques remarques sur les avantages et les inconvénients des deux niveaux où intervient la compression.

Lorsque l'on comprime le contenu d'une lettre, on diminue (en général) la longueur de la lettre, donc on transmet sur la liaison des lettres qui ne sont pas remplies au maximum de leur capacité. Or on ne comprime pas la partie contenant les informations de contrôle ajoutées par le service de transport. Il y a donc une augmentation de la surcharge de travail en même temps qu'une diminution du volume total transporté puisque l'on transporte autant de lettres plus courtes.

Si l'on agit en comprimant les enregistrements, le mécanisme de découpage remplit toujours les lettres à leur capacité maximale. On aura donc une diminution du volume total transporté sans accroissement de la surcharge de travail puisque le rapport longueur utile sur longueur totale reste inchangé. Le nombre de lettres diminue. On réduit la longueur du transfert sans augmenter la surcharge de travail. Donc effectuer la compression à ce niveau apparaît plus intéressant.

La compression elle-même est-elle plus efficace à un niveau ou à un autre? Elle apparaît moins efficace au niveau des lettres car la présence des délimiteurs de segment d'enregistrement peut nuire à la compression des longues chaînes à cheval sur deux segments.

3.2.2 Algorithmes de compression

Plusieurs algorithmes de compression sont envisagés: comprimer systématiquement tout octet ou n'en comprimer que certains reconnus comme les plus courants.

3.2.2.1 Compression systématique

Pour comprimer tout octet, on peut adopter deux politiques:

1) Faire suivre chaque octet de son facteur de répétition: cela oblige à ajouter à chaque séquence d'octets identiques un octet qui indique le nombre de fois dont il faut répéter l'octet, même si l'octet n'est présent qu'une seule fois. Cela double le volume d'information au cas où aucun octet n'est répété. Cette compression devient efficace quand en moyenne chaque octet est plus que doublé.

A	A	Y	Z	Z	Z	X	X	X	X
---	---	---	---	---	---	---	---	---	---

A	2	Y	1	Z	3	X	4
---	---	---	---	---	---	---	---

2) Faire précéder chaque octet à répéter d'un octet de valeur particulière, et le faire suivre de son facteur de répétition. Chaque séquence d'octets consécutifs identiques sera représentée par un groupe de trois octets: le code particulier, l'octet à répéter et le facteur de répétition. Les octets isolés ou doublés resteront inchangés. L'octet particulier signalant la répétition devra subir la représentation sous forme de trois octets. Cette compression est intéressante si un octet répété l'est en moyenne plus de trois fois.

A	A	Y	Z	Z	Z	X	X	X	X
---	---	---	---	---	---	---	---	---	---

A	A	Y	'R'	Z	3	'R'	X	4
---	---	---	-----	---	---	-----	---	---

3.2.2.2 Compression d'octets spécifiques

Si l'on peut distinguer les valeurs des octets qui sont le plus souvent répétés, on peut ne compresser que ceux-là. Il faut alors repérer ces octets, les compter et ajouter à leur suite leur facteur de répétition. Cette compression est intéressante si les octets répétés le sont en moyenne au moins deux fois, par contre l'algorithme de compression n'est plus systématique. Il faut repérer ces octets à la compression comme à la décompression, avec à chaque fois un nombre de comparaisons égal au nombre d'octets que l'on veut comprimer.

Cet algorithme permet de limiter les octets à comprimer; il est d'autant plus efficace que les octets spécifiques sont peu nombreux et qu'ils sont souvent répétés. Les caractères les plus fréquemment répétés dépendent en grande partie du type de fichier qui les contient; par exemple, un fichier de texte peut contenir plusieurs espaces ou des caractères de tabulation répétés alors qu'un fichier de code binaire présente plus souvent l'octet de valeur nulle abondamment répété.

3.2.2.3 Compression optimisée

Un autre algorithme de compression permet de réunir les avantages des deux précédents: la compression systématique de tout octet sans ajout systématique signalant la répétition. On considère alors les octets non pas individuellement mais en tant que chaîne de plusieurs octets. On distingue des chaînes d'octets tous différents et des chaînes d'octets identiques; parmi ces derniers on peut reconnaître des octets spécifiques dont la répétition peut être indiquée de manière très condensée: la chaîne entière et le facteur de répétition sont représentés sur un octet <Ullmann 70>.

3.2.3 Choix de l'algorithme de compression

Pour choisir l'algorithme de compression, il nous faut trouver un compromis entre la qualité de la compression, qui permet de gagner du temps lors de la transmission des données, et la rapidité de l'algorithme de compression qui introduit une surcharge en temps lors de la préparation des données à transmettre et de leur réception.

Lorsque la vitesse de préparation des données et la vitesse de leur transmission sont égales, il ne faut pas introduire, lors de la compression, de surcharge supérieure au temps gagné à la transmission. Par contre si la vitesse de transmission est largement inférieure à la vitesse de préparation, on peut se permettre de prévoir un algorithme de compression très efficace même s'il prend du temps pour la compression. Il faut néanmoins éviter de consommer trop de temps de traitement au détriment des autres activités locales.

Pour déterminer l'algorithme le plus efficace pour un type donné de fichier, nous nous sommes livrés à une étude sur la fréquence de répétition des valeurs des octets contenus dans un fichier. En lisant les enregistrements des fichiers représentatifs, nous avons calculé la fréquence et la taille des groupes d'octets consécutifs identiques. L'intérêt est de déterminer les valeurs des octets qui sont le plus souvent répétés consécutivement et d'évaluer le gain que nous pouvons obtenir par les différents algorithmes de compression.

Dans un environnement stable, où les fichiers sont bien caractérisés, et avec un moyen de communication rapide, le choix de l'algorithme de compression doit être prédéfini

selon le type de fichier. Mais si le contenu des fichiers n'est pas toujours bien typé et que le moyen de communication est relativement lent, on peut déterminer au vu du fichier à transférer l'algorithme de compression qui lui convient le mieux.

Dans notre environnement, les fichiers sont, pour la plupart, bien typés et reconnaissables d'après leur suffixe. Les communications sur le réseau sont lentes par rapport aux débits d'information avec les unités de disques. Nous avons donc choisi de prédéfinir l'algorithme de compression pour les suffixes connus. Pour les suffixes inconnus, propres à des utilisations particulières et assez rares, nous analysons le contenu du fichier pour déterminer le meilleur algorithme de compression pour son transfert. Cette analyse ne fait que compter les répétitions, donc elle nécessite un temps de traitement relativement faible par rapport à la compression elle-même.

4 INTERFACE DE TRANSFERT PAR PROGRAMME POUR LA MESSAGERIE

Dans sa version précédemment présentée, le service de transfert de fichiers sur le réseau local est employé directement par un utilisateur humain grâce à des commandes lues au terminal par la partie client. Dans l'utilisation courante par des applications comme la messagerie, l'intervention "manuelle" doit disparaître et faire place à une utilisation du transfert de fichiers directement par les applications qui s'en servent. Le service offert à travers un terminal doit devenir accessible par programme.

Il y a plusieurs façons de présenter à un programme d'application une interface d'accès aux services de transfert de fichiers. Il s'agit d'intégrer de manière

cohérente les services de base offerts par le service de transport et le service de transfert de fichiers. Le second utilise le premier et tous deux sont utilisés par l'application. On peut concevoir d'ajouter les services de transfert au service de transport, de l'incorporer à l'application ou d'en faire un ensemble à part.

En fait nous avons envisagé deux types d'interface correspondant à deux classes de programmes d'application. On considère d'une part une application répartie qui désire échanger des fichiers entre deux de ses composantes, et d'autre part une application répartie ou non qui désire échanger des fichiers avec un site de son choix, indépendamment de la présence éventuelle sur ce site d'une composante de cette application.

Le premier type d'interface pourra être finement intégré à l'application pour y constituer un sous-ensemble homogène du protocole de dialogue entre les composantes réparties. Etant destinée uniquement à cette application, cette interface pourra être adaptée aux besoins de l'application, c'est à dire simplifiée si possible.

Initialement l'interface proposée sera complète quant à la nature des services de transfert offerts. Elle sera la même pour chaque composante, offrant par exemple des transferts dans chaque sens et à l'initiative de l'un quelconque des partenaires. Si cela n'est pas nécessaire, on peut réduire les services proposés: ne donner l'initiative du transfert qu'au récepteur ou à l'émetteur du fichier, ou si le transfert n'a toujours lieu que dans un sens, supprimer ce qui permet le transfert dans l'autre sens.

Ce type d'interface adaptée aux besoins et intégrée à l'application, permet de considérer le transfert de fichiers comme un prolongement du protocole de l'application tout en supprimant la gestion du transfert.

Le second type d'interface est obtenu en supprimant la symétrie existant dans la première. On aboutit à une interface qui permet à une application d'échanger des fichiers avec un serveur spécialisé ou de lui passer des commandes. Il reste comme simplification possible la suppression de la partie de l'interface qui assure le transfert dans un sens si celui-ci est inutilisé.

C'est le premier type d'interface que nous allons présenter; elle a été utilisée pour permettre à l'application de messagerie d'échanger des fichiers de messages entre ses composantes. Ces composantes dialoguent entre elles avant de devoir échanger un fichier, elles ont pour cela déjà établi une liaison de communication. Pour des raisons de simplicité des interfaces et d'économie de ressources, cette même liaison est utilisée pour le transfert de fichiers puis rendue à la messagerie. L'interface doit assurer la gestion de la liaison de communication et la synchronisation des échanges sur cette liaison.

4.1 Gestion de la liaison de communication

Puisque l'interface de transfert de fichiers utilise la même liaison que l'application de messagerie, il faut assurer le passage du contrôle de la liaison de l'un à l'autre. La prise en compte des événements qui surviennent sur la liaison est réalisée par l'appel d'une procédure

asynchrone. La messagerie et l'interface fournissent chacune une telle procédure. Selon la valeur d'une variable commune, modifiée lors du transfert de contrôle de la liaison, on choisit la procédure à appeler.

4.2 Synchronisation sur la liaison

Le contrôle du flux de lettres échangées sur la liaison est effectué par un mécanisme de crédit en émission. Lorsque le contrôle à chaque extrémité de liaison est commuté, le crédit doit être nul afin qu'aucune des interfaces ne puisse envoyer de lettre avant que l'autre n'ait aussi le contrôle de la liaison. De même, le crédit est restitué, à la fin du transfert, à une valeur nulle de part et d'autre de la liaison. Pour cela chaque partenaire du transfert épuise le crédit de son vis à vis en émettant des lettres jusqu'à ce qu'il soit bloqué et jusqu'à ce qu'il ait reçu autant de lettres que son propre crédit initial. Cela garantit que le crédit sera nul quand la liaison sera restituée et qu'aucune lettre émise par l'interface n'arrivera après que la liaison ait été rendue à la messagerie.

5 PROTOCOLE DE REPRISE D'ECHANGE

On présente dans un système réparti sur un réseau local un protocole d'échange d'informations qui permet la reprise automatique de l'échange après une interruption causée par une panne matérielle ou logicielle sans demander un trafic supplémentaire d'informations de contrôle pendant un échange sans panne. On part des hypothèses sur le service de communication inter-tâches, on définit les pannes qui permettent une reprise sans intervention manuelle, on expose le principe de la détection des pannes et de la reprise de l'échange, et on aboutit aux conditions à vérifier par le service d'accès à l'information.

5.1 Introduction

Au sein des réseaux généraux comme des réseaux locaux, il est nécessaire d'effectuer des échanges d'informations entre les sites qui les composent. Ces échanges, dont la durée est fonction du volume d'information, ainsi que du débit et du temps de réponse du moyen de communication, présentent des risques d'interruption due à des pannes matérielles ou logicielles. Dans les réseaux locaux, où la distance de transmission a diminué, où le débit est important et le temps de réponse très faible, la durée de l'échange se trouve réduite, diminuant ainsi la probabilité de rencontrer une panne pendant l'échange; une meilleure fiabilité du moyen de communication a aussi contribué à cette baisse de la probabilité de panne pendant l'échange.

Néanmoins, il subsiste des cas d'erreur; des pannes peuvent nécessiter la reprise de l'échange à son début; bien que leur probabilité soit faible, il convient d'éviter

au maximum que leur occurrence ne soit ressentie à l'usage. Aussi, en nous basant sur le service de communication inter-tâches sur le réseau local, nous avons étudié les possibilités de reprendre automatiquement un échange interrompu par des pannes matérielles ou logicielles que nous définirons.

La reprise d'un échange après son interruption imprévue constitue la part ingrate d'un service de gestion des informations dans un système réparti; certaines réalisations passent cette part sous silence, d'autres la laissent pour plus tard <Shoch 78>. Néanmoins son utilité est ressentie, mais l'étude de son implémentation conduit à des solutions nécessitant l'échange de marques de points de reprise, insérés dans le flot d'information échangée, et de leur acquittement. L'attente de ces acquittements explicites fait progresser le transfert par bonds d'un point de reprise à l'autre, ce qui peut nuire à la rapidité et à la régularité de l'échange <FTPSE 80> <FTPIG 81>.

Pour bâtir leur protocole d'échange et de reprise, tous les services d'échange d'information existants s'appuient sur un service de communication qui les décharge du contrôle d'erreur et de flux. Notre originalité consiste à exploiter le contrôle à la source du flux d'information échangée pour éviter l'échange explicite de points de reprise et de leur acquittement.

En outre nous étudierons les moyens de détecter des pannes, et les moyens de reprendre l'échange après leur réparation, sans surcharger un échange sans problème. Nous adopterons, pour conserver le contexte d'évolution de l'échange d'information, une méthode qui ne requiert pas d'échange supplémentaire d'information de contrôle, ni la tenue à jour d'une trace de l'état de l'échange en dehors des entités concernées: documents échangés et émetteur et

récepteur. Ces conditions excluent la possibilité d'une reprise automatique en cas de panne simultanée de l'émetteur et du récepteur à cause de la perte totale du contexte.

Nous allons présenter le contexte dans lequel se pose le problème, les pannes qui peuvent donner lieu à une reprise automatique et le principe de leur détection. Puis nous définirons le protocole d'échange permettant une reprise automatique efficace.

5.2 Contexte de la communication d'informations sur un réseau local

Ce problème se situe dans les réseaux locaux d'ordinateurs. Les entités disposent d'un service de communication qui leur permet d'échanger des messages. Nous allons présenter la configuration du réseau lors d'un échange d'informations, détailler le principe de l'échange d'informations, définir les conditions d'accès à l'information et préciser les caractéristiques du service de communication.

5.2.1 Configuration du réseau lors des échanges de documents

Les composants matériels dépendent de l'architecture du réseau <Clark 78>; mais ils correspondent à ceux décrits ici dans la présentation des réseaux locaux (chapitre II). Les composants logiciels sont l'émetteur et le récepteur du document, et la station de transport qui réalise sur chaque site le service de communication.

5.2.2 Principe du transfert

Le document contenant l'information est transmis d'un correspondant à l'autre. La circulation des données est réglée par le contrôle de flux.

5.2.2.1 Emetteur, récepteur.

L'échange d'un document a lieu entre deux correspondants, récepteur et émetteur du document. La majeure partie du temps, l'émetteur envoie des lettres que le récepteur s'attend à recevoir. Si, à un moment donné, de par le fonctionnement du protocole d'échange, c'est l'émetteur qui doit recevoir une lettre que le récepteur doit lui envoyer, les rôles d'émetteur et de récepteur sont inversés. Néanmoins, à tout moment on peut distinguer une entité qui se prépare à envoyer une lettre, et une entité qui se prépare à la recevoir. C'est dans ce sens que nous utiliserons les termes de récepteur et d'émetteur.

5.2.2.2 Contrôle de flux : blocage, déblocage.

Le contrôle de flux exercé par le service de transport sur la liaison permet au récepteur d'autoriser l'émetteur à n'envoyer que le nombre de lettres qu'il se déclare être en mesure de recevoir. Ce nombre est en général lié à la taille de l'espace de stockage libre chez le récepteur. Le récepteur indique à sa S.T. le crédit initial qu'il donne à l'émetteur; ce crédit se trouve diminué à chaque lettre reçue, il peut être augmenté par le récepteur à sa convenance, dès que, par exemple, il a libéré l'espace de stockage d'une lettre reçue et traitée. L'émetteur verra son envoi refusé par sa S.T. tant qu'elle n'aura pas reçu de crédit pour lui <AMOEBA 82>, <Taft 78>, <TCP 79>.

Nous parlerons de blocage quand l'émetteur sera sans crédit pour émettre, ou le récepteur sans lettre à traiter. Le blocage de l'émetteur ou du récepteur dépend essentiellement de leurs vitesses de traitement respectives des informations transmises.

5.2.3 Conditions sur le service d'accès à l'information

L'émetteur et le récepteur échangent un document en le transférant d'un support chez l'émetteur vers un support chez le récepteur. Ils progressent par unités d'information. Le traitement chez le récepteur consiste à inscrire l'unité d'information sur un support de façon à ne plus avoir à y revenir par la suite. La conservation du document nécessite donc un support stable <Mitchell 82a>.

De plus, il faut que la suite d'unités inscrites sur le support forme à tout moment un ensemble cohérent qui représente, durant le transfert, l'état du document en cours de constitution. En cas d'interruption du récepteur sur une panne, cet état est préservé sur le support. Cette propriété permet de laisser le document dans un état cohérent pour le système de gestion des informations, même si l'entité créatrice du document ne peut, à un moment donné, faire elle-même l'action de clore momentanément l'accès au document car elle subit une panne qui l'en empêche. Ensuite, la même entité peut demander, explicitement cette fois, l'accès au document pour poursuivre sa constitution, d'après l'état existant au moment de la panne et préservé par ce moyen. En fait, cela demande que l'état du document soit fidèlement représenté dans un catalogue et mis à jour à chaque écriture dans le document <DSAI 81>.

5.2.4 Caractéristiques du service de communication

La présentation des caractères généraux de ce service de communication portera d'abord sur l'utilisation par ses abonnés en fonctionnement normal; en cas de fonctionnement anormal, nous décrirons les possibilités de détecter, de manière directe ou indirecte, les défaillances éventuelles.

5.2.4.1 Fonctionnement normal du service de communication

Parmi les possibilités offertes par le service de transport, nous insistons davantage sur les services qui sont utilisés pour l'échange fiable des documents, ceux qui permettent d'intervenir à distance sur le fonctionnement du réseau et ceux qui jouent un rôle pour la reprise automatique d'un échange interrompu. Les caractéristiques intéressantes sont les suivantes :

- la S.T. peut être activée, si elle ne l'est pas déjà, par tout abonné qui désire communiquer, lors de son premier abonnement,

- il existe un service de liaison qui délivre les lettres au destinataire sans erreur et dans l'ordre de leur envoi par l'expéditeur,

- ce même service permet le contrôle par leur destinataire du flux de lettres échangées, grâce à un mécanisme de crédit en émission,

- lorsque la station de transport est active sur un site, on peut demander depuis l'extérieur le lancement d'une tâche existant sur ce site.

Ces caractéristiques sont celles qui vont assurer un échange fiable et permettre la reprise. Nous allons voir maintenant celles qui vont permettre de détecter les défaillances.

5.2.4.2 Comportement de la S.T. en cas de fonctionnement anormal

Lors d'un fonctionnement anormal, le comportement du service de transport présente deux aspects positifs et deux aspects négatifs :

5.2.4.2.1 Aspects positifs :

a) Le service de transport peut détecter l'absence de la station correspondante distante si cette absence survient pendant un échange. En effet pour le service liaison déjà présenté, les S.T. concernées par une liaison bi-point entretiennent cette liaison par un échange régulier de messages de contrôle quand les abonnés correspondants ne communiquent pas. Le défaut d'entretien de la liaison est constaté au bout d'un délai fixé, il peut être dû à l'absence de la station correspondante distante ou bien à une défaillance du réseau.

b) Un abonné peut savoir si sa S.T. est absente. En effet, un abonné communique avec sa S.T. par l'intermédiaire d'une interface qui lui est liée et qui répond à l'abonné que la S.T. n'est pas active, quand celui-ci a demandé un service à sa S.T. défaillante;

5.2.4.2.2 Aspects négatifs :

a) La S.T. ne prévient pas ses abonnés quand elle disparaît, d'une part car la cause de la panne peut être abrupte et ne pas le permettre, d'autre part car il est difficile de le réaliser dans tous les cas. Néanmoins, la possibilité de constater l'absence de la S.T. quand on en a besoin y remédie partiellement.

b) La S.T. ne peut pas être informée de la panne d'un abonné local, quel que soit le moment où cette panne survient, par contre, en cas de besoin, là aussi, elle peut constater l'absence de son abonné.

Un abonné peut donc constater une panne par une action spécifique, mais il n'est jamais prévenu quand la panne survient.

5.3 Les pannes et leur détection

L'interruption d'un échange d'information est due à la rupture de la chaîne de communication qui va de l'émetteur au récepteur. Cette chaîne est constituée de cinq maillons qui sont l'émetteur, sa S.T., le support de communication et ses voies d'accès, la S.T. distante et le récepteur; elle peut donc présenter cinq points de rupture simple qui se ramènent par symétrie à trois points qui sont les correspondants, les S.T. et le support de communication avec ses voies d'accès. Parmi les cas de rupture de la chaîne de communication en deux points, ceux qui nous intéressent par les possibilités de reprise qu'ils offrent sont: la défaillance simultanée des deux S.T., et la défaillance d'un correspondant et de sa S.T. distante.

Nous décrivons dans le détail les pannes envisagées, leur gravité, leur probabilité, pour le matériel et le logiciel, et définissons les conditions d'une reprise automatique. Puis nous présentons la façon de détecter leur occurrence en se plaçant au point de vue des entités émettrice et réceptrice des informations échangées.

5.3.1 Pannes envisagées

La nature des pannes matérielles peut être soit une longue indisponibilité, soit une brève interruption du service causée par des conditions passagères.

Les pannes logicielles envisagées ne sont pas celles propres aux logiciels, mais des pannes qui traduisent l'effet sur le logiciel de mauvaises conditions de fonctionnement du matériel. Nous supposons que le logiciel est au point et nous nous intéressons aux issues anormales dues à un mauvais comportement de l'environnement.

5.3.1.1 Pannes du matériel

Dans l'optique d'une reprise automatique, on peut distinguer des pannes fugitives (se réparant d'elles-mêmes), et des pannes permanentes qui réclament une intervention pour leur réparation. On doit aussi considérer la possibilité de conserver le contexte d'un échange en cours jusqu'à la réparation de la panne. Cela dépend du matériel atteint par la panne, et du temps nécessaire à sa réparation. D'autre part une reprise automatique découle de la capacité des éléments logiciels concernés par la panne à démarrer par eux-mêmes.

Donc, dans le cas d'une panne fugitive atteignant un élément passif ou un élément actif qui ne garde pas le contexte et redémarre tout seul, la reprise est automatique.

5.3.1.2 Pannes du logiciel

Ces pannes peuvent survenir de manière isolée, ou bien concerner plusieurs des éléments logiciels mis en jeu pour l'échange.

5.3.1.2.1 Pannes logicielles isolées

Ces pannes ne concernent qu'un seul élément logiciel : l'émetteur, le récepteur, la S.I. émetteur ou la S.I. récepteur. Ces divers éléments peuvent être relancés par l'un des trois autres, ce qui assure le rétablissement de la chaîne de communication et la poursuite de l'échange.

5.3.1.2.2 Pannes logicielles couplées

Ces pannes vont concerner certains couples d'éléments logiciels : l'émetteur et la S.I. du récepteur, le récepteur et la S.I. de l'émetteur, ou les deux S.I. Les éléments de ces couples peuvent chacun être relancés par l'un des deux qui restent actifs, ce qui assure le rétablissement de la chaîne de communication et la poursuite de l'échange.

5.3.1.2.3 Autres pannes logicielles

Les autres configurations de pannes logicielles ne permettent pas le redémarrage automatique des éléments logiciels inactifs, car ils impliquent la panne simultanée de l'émetteur ou du récepteur et de la S.T. locale, or le matériel ne permet pas de relancer une tâche sans l'intermédiaire de la S.T., qui est absente et ne peut être relancée à distance.

5.3.1.3 Conditions pour une relance automatique

On ne peut disposer d'une reprise automatique que lorsque le matériel permet au logiciel de prendre en compte les pannes. Or on vient de voir qu'on ne peut pas relancer la S.T. à distance et, sans elle, on ne peut lancer de tâche à distance, ce qui interdit une partie des reprises possibles. Une amélioration serait de permettre de lancer la S.T. sur un site donné depuis l'extérieur, ce qui donnerait alors accès aux services offerts par la S.T. Mais maintenir la S.T. toujours active sur chaque site et forcer sa relance au redémarrage du système après panne d'un site demande une surveillance non automatique.

Dans <DECNET-NSP 76>, on définit la procédure d'initialisation d'un site mais sans envisager son démarrage à distance. Dans <Herbert 81>, il est possible de charger et de lancer un programme à distance, mais cela nécessite qu'un processeur ancillaire soit toujours actif.

5.3.2 Détection des pannes

Certaines pannes peuvent être détectées et signalées par le service de transport, d'autres peuvent être détectées grâce à la durée du blocage.

5.3.2.1 Absence signalée par la S.I.

A cause du défaut d'entretien d'une liaison, une S.I. peut détecter l'absence de la S.I. correspondante sur la liaison, ce qui peut correspondre aussi à la rupture du support de communication; cela sera signalé à l'abonné comme une rupture anormale de la liaison. La fermeture normale de liaison sera signalée à un abonné à la suite de la demande de fermeture de liaison à l'initiative de l'abonné correspondant.

Un abonné sera donc averti de la panne d'une S.I. distante, ou de la fermeture de la liaison par le correspondant. En dehors de ces cas de pannes signalées, il reste à la charge de l'émetteur ou du récepteur de repérer tout autre type de panne. Nous allons examiner comment l'émetteur ou le récepteur peuvent constater une panne par dépassement du délai d'attente en cas de blocage.

5.3.2.2 Délai de garde en cas de blocage.

Le délai d'attente avant de recevoir ou de pouvoir envoyer une lettre est fini. Il dépend des vitesses de traitement de l'émetteur et du récepteur et de l'occupation du réseau. On peut fixer un maximum à ce délai et au bout de ce temps limite chercher à localiser l'origine du silence (donc de la panne) et envisager la manière de la

récupérer. Ce délai sera supérieur aux délais de garde de la S.T. qui lui permettent de détecter certains types de pannes et de les signaler à ses abonnés.

5.3.2.2.1 Détection par l'émetteur

Si l'émetteur est bloqué trop longtemps sans pouvoir émettre, c'est que sa station de transport ne reçoit plus de crédit pour lui. Donc : soit sa propre S.T. est en panne, soit le récepteur ne renvoie plus de crédit. L'émetteur peut vérifier la présence de sa S.T. en se voyant refuser l'envoi d'une lettre par celle-ci; en cas d'absence de sa S.T. il ne sait rien sur l'état du site distant. Sinon, comme ce ne peut être un silence de la S.T. distante, qui aurait été signalé par sa S.T., c'est donc le récepteur qui ne renvoie pas de crédit.

5.3.2.2.2 Détection par le récepteur

Si le récepteur ne reçoit plus, il peut y avoir deux causes : soit c'est l'émetteur qui n'envoie plus de lettre, soit c'est la S.T. du récepteur qui est absente. Le récepteur peut s'enquérir de l'état de sa S.T. en tentant une émission; d'après la réponse il connaîtra son état. Si sa S.T. est active, c'est que la S.T. distante est aussi active, donc c'est l'émetteur qui est en défaut. Par contre, si la S.T. locale est absente on ne sait rien sur l'état du site distant.

5.4 Le protocole de reprise automatique

Ce protocole comporte une part qui fonctionne durant tout transfert, pour constituer et entretenir le contexte de l'échange, et une part qui permet de poursuivre l'échange, après son interruption, grâce au contexte conservé.

5.4.1 Contexte d'un échange en cours

Ce contexte définit le document échangé et le point d'évolution du transfert. Ces deux informations devront pouvoir être connues le plus exactement possible par les deux correspondants pour permettre une reprise en cas d'interruption.

5.4.1.1 Définition du document échangé

La coordonnée du document source chez l'émetteur sera aussi connue par le récepteur; de même, la coordonnée du document destination chez le récepteur sera aussi connue par l'émetteur. Ainsi, il suffit qu'un seul des correspondants survive à la panne pour ces informations demeurent connues.

5.4.1.2 Point d'évolution du transfert

Ce point d'évolution se limite à connaître l'état du document à sa destination, c'est à dire chez le récepteur. L'état du document destination sera tenu à jour de part et d'autre, mais il ne sera exact que chez le récepteur. Donc

le contexte conservé exact est l'état du document reçu par le récepteur. En cas de reprise, l'émetteur devra se positionner dans le document source au point indiqué par le récepteur.

5.4.2 Gestion du contexte de l'échange

La tenue à jour de part et d'autre du contexte de l'échange exploite les caractéristiques du service de transport. La définition des points de reprise est liée à la taille des lettres échangées sur une liaison. Leur acquittement est assuré implicitement par le contrôle de flux sur la liaison. L'accès, en lecture ou en écriture, à des unités d'information de taille fixe, par l'émetteur et le récepteur, simplifie le calcul de la progression dans les documents concernés.

5.4.2.1 Définition des points de reprise

Le récepteur et l'émetteur n'accèdent pas chacun à des unités d'information de même taille, on ne peut donc pas se baser là-dessus pour faire progresser des points de reprise communs <FTPIG 81>. Par contre ils ont en commun les lettres qu'ils échangent sur la liaison. En comptant, chacun de leur côté, ces lettres, qui contiennent une information qui a la même signification de part et d'autre, ils constituent une échelle de référence de leur progression dans le document échangé.

5.4.2.2 Acquittement implicite grâce au contrôle de flux

Le contrôle de flux intervient de la façon suivante pour l'acquittement des points de reprise. Le récepteur libère l'espace de stockage d'une lettre après l'avoir traitée, ce qui permet à l'émetteur d'en envoyer une autre. L'émetteur considère une lettre comme acquittée, quand il a pu envoyer par la suite autant de lettres que peut en contenir l'espace de stockage du récepteur. Les lettres étant traitées dans l'ordre de leur arrivée, cette première lettre aura donc été traitée puis retirée de l'espace de stockage du récepteur.

A la fin de l'échange, ce principe ne sera pas applicable, le récepteur pourra alors envoyer à l'émetteur un acquittement explicite des dernières lettres reçues à la fin du document.

5.4.2.3 Numérotation des lettres et situation dans le document

Le récepteur comme l'émetteur doivent pouvoir retrouver le point où l'échange a été interrompu par une panne, au moyen des informations gardées par le correspondant survivant à la panne. Ils disposent des lettres qu'ils échangent, et dont ils numérotent la succession, comme référence commune.

Cependant, ils ont tous deux besoin de passer d'un numéro de lettre à une position dans le document, et réciproquement. La correspondance peut être décrite et entretenue dynamiquement au cours de l'échange dans une table qui associe lettres et unités d'information. Mais en cas de panne cette table est perdue, et il est très

difficile au correspondant de la rétablir chez son vis à vis grâce à sa propre table. Donc, il faut qu'ils puissent interpréter leur seule information commune : le nombre de lettres échangées, pour en déduire la position dans le document et vice versa. Les unités d'information lues ou écrites ont une taille fixe; si l'on convient d'échanger des lettres de taille fixe, il est aisé de déterminer le numéro de la lettre qui correspond à une unité d'information donnée, ou bien le contraire. Il suffit de calculer les résultats de l'algorithme de fragmentation-réassemblage des unités d'informations en lettres sans le réexécuter. Ce calcul est plus long si un algorithme de compactage intervient lors du passage des unités d'information aux lettres.

5.4.2.4 Progression chez l'émetteur

L'émetteur constitue des lettres à partir des informations lues dans le document source. Ces lettres sont envoyées l'une après l'autre au récepteur, et conservées chez l'émetteur jusqu'à ce qu'elles soient acquittées, c'est à dire jusqu'à ce que l'émetteur considère qu'elles ont été traitées par le récepteur. L'émetteur gère donc deux positions dans le document, celle avancée qui est le point où il se trouve dans le document source, et celle qui correspond à la dernière lettre acquittée. Il doit disposer d'un espace de stockage assez vaste pour pouvoir contenir toutes les lettres envoyées mais non acquittées. Ainsi, en cas de panne du récepteur seul, l'émetteur pourra renvoyer à la demande les lettres déjà envoyées mais non acquittées, sans revenir en arrière dans le document pour les reconstituer. L'effet est un gain de temps lors de la reprise.

5.4.2.5 Progression chez le récepteur

Le récepteur réassemble les informations contenues dans les lettres pour reconstituer l'unité à placer sur le support de destination. L'acquiescement des lettres est lié à l'écriture correcte sur le support des informations qu'elles contiennent. En effet, une lettre ne peut être acquiescée au point de vue de l'émetteur que quand la totalité de son contenu est inscrite, par le récepteur, de manière fiable sur le support de destination. Le récepteur doit faire correspondre ses points de reprise internes, qui sont les unités d'information qu'il peut inscrire en une fois sur le support, avec les lettres qui sont l'unité commune de marque de reprise. Il doit donc disposer d'un espace de stockage assez vaste pour contenir toutes les lettres qui forment ensemble une unité à écrire sur le support.

5.4.3 Rétablissement de la liaison rompue

Que ce soit l'émetteur ou le récepteur qui constate la panne de son correspondant, on se trouve face à trois cas :

- 1) panne de leur S.I. locale,
- 2) silence de la S.I. distante signalée par leur S.I. locale,
- 3) panne du correspondant distant, les S.I. restant actives.

Dans le premier cas, S.I. locale absente, on ne connaît rien sur l'état du site distant. On va aboutir à l'un des cas suivants ou au cas de bon fonctionnement.

1) Un correspondant dont la S.T. locale disparaît doit considérer la liaison existante comme rompue, et son abonnement au service de transport comme annulé. Il doit se réabonner, réactivant ainsi sa S.T., grâce à la propriété citée. Puis il cherche à joindre son correspondant. Il y a trois possibilités :

a) il n'y a pas de réponse car : ou bien la S.T. distante est absente (on aboutit au deuxième cas) ou bien le support de communication est rompu,

b) la S.T. répond mais ne retrouve pas le correspondant (on aboutit au troisième cas),

c) la S.T. et le correspondant sont là, et la liaison est rétablie (cas de bon fonctionnement).

2) Silence de la S.T. distante; sur le site distant, il peut coïncider ou non avec le cas précédent. La reprise automatique n'est possible que si le support fonctionne et que le correspondant distant est actif. Dans ce cas, on attend que le correspondant distant relance sa S.T., se réabonne et demande le rétablissement de la liaison.

3) Les S.T. sont actives de part et d'autre; un correspondant constate l'absence du correspondant distant. Il peut demander à la S.T. distante de relancer le correspondant distant. Si celui-ci a effectivement disparu, il est relancé et la liaison peut être rétablie avec lui. Par contre, si le correspondant distant est toujours actif, mais que son silence est dû à son mauvais fonctionnement, il n'y a rien à faire. En général cela peut être évité si les correspondants sont écrits correctement, pour ne pas se mettre par leur faute dans un état irrécupérable, et s'ils réagissent en avertissant l'extérieur lorsqu'ils rencontrent un problème qu'ils ne peuvent résoudre.

5.4.4 Initiative du rétablissement de la liaison

En laissant de côté les cas qui sont irrécupérables dans les conditions que nous avons fixées, il est possible de reconstituer le chemin qui va d'un correspondant à l'autre et de rétablir la liaison sur laquelle ils communiquaient.

L'établissement de la liaison étant demandée par un correspondant et acceptée par l'autre, il convient de préciser lequel doit prendre l'initiative de son rétablissement. On peut distinguer deux cas :

- 1) Une S.T. a été relancée,
- 2) Un correspondant a été réactivé,

La combinaison de ces deux cas correspond à la panne simultanée d'un correspondant et de sa S.T. distante.

1) S.T. relancée : c'est au correspondant qui l'a fait de rétablir la liaison; ainsi, la demande de rétablissement de liaison aura bien lieu après que la communication ait été rétablie. Le correspondant distant aura simplement à attendre la demande de liaison, au lieu de répéter lui-même une demande semblable jusqu'à ce que la S.T ait été relancée par le premier correspondant.

2) Correspondant réactivé à distance : il ignore qu'il y avait un échange en cours, il faut donc l'en avertir, et au préalable établir une liaison pour communiquer avec lui.

5.4.5 Rétablissement du contexte d'échange

Une fois la liaison établie, il s'agit de reconstituer le contexte de l'échange interrompu pour le poursuivre. En distinguant entre le récepteur et l'émetteur, il apparaît trois cas selon l'état des correspondants durant l'interruption :

- 1) l'émetteur seul a été interrompu et relancé,
- 2) le récepteur seul a été interrompu et relancé,
- 3) ils sont tous deux restés actifs, et seules les S.I. ont été interrompues.

Nous excluons le cas où l'émetteur et le récepteur ont été interrompus, il nécessite à l'évidence une intervention manuelle, car le contexte de l'échange est perdu. Dans les cas retenus, le contexte de l'échange est en partie conservé, soit par les deux correspondants, soit par l'émetteur, soit par le récepteur. Les deux correspondants doivent alors se mettre d'accord sur le contexte de l'échange. Ils se basent sur le contexte conservé le plus fidèle, qui est l'état du document reçu chez le récepteur.

1) Les deux correspondants sont restés actifs : la seule action préalable à la reprise est la mise en concordance de leurs contextes.

2) Panne de l'émetteur : le récepteur doit transmettre à l'émetteur les coordonnées du document source, ainsi que la position à partir de laquelle la reprise aura lieu. L'émetteur s'y positionne et poursuit la transmission.

3) Panne du récepteur : l'émetteur connaît les coordonnées du document destination. Il en informe le récepteur et lui demande la position de reprise dans ce document. Il se positionne au point correspondant dans le document source, et reprend la transmission.

5.4.6 Poursuite de l'échange

A partir de là, l'échange pourra se poursuivre. Le contexte rétabli indiquera chez chacun des correspondants les coordonnées des documents source et destination, ainsi que la position courante de l'échange.

5.5 Conclusion

Le protocole décrit n'a pas encore été implémenté sous forme de maquette; on ne peut donc pas évaluer son efficacité réelle. D'autre part, il présente des limitations dans l'éventail des pannes à la suite desquelles il permet une reprise automatique. En effet, une panne qui atteint les deux entités correspondantes concernées par un échange provoque non seulement la perte du contexte de cet échange, mais aussi la connaissance même du déroulement de cet échange.

Seul le demandeur de cet échange peut constater cet oubli, réitérer sa demande, et provoquer, par son intervention, la poursuite de l'échange. Pour permettre une reprise différée de l'échange, ce rôle de surveillance doit être tenu à l'intérieur du système. On peut envisager un contrôle centralisé ou réparti de la terminaison des échanges en cours.

Un contrôleur centralisé doit être prévenu du démarrage et de la terminaison de tout échange. Il a la charge de vérifier que chacun d'eux se termine. Une version répartie peut laisser la responsabilité de la bonne terminaison d'un échange au récepteur, ou à l'émetteur, du document échangé. Celui-ci doit gérer la liste des échanges dont il s'occupe et assurer qu'ils se terminent.

Ainsi, pour une assez faible charge supplémentaire, on peut assurer la reprise automatique de tout échange, immédiate ou en différé, bien que les éléments logiciels concernés par ces échanges nécessitent parfois une intervention manuelle pour leur redémarrage.

VI

CONCLUSION

A la suite de cette étude, quelques réflexions s'imposent; elles portent sur la nature des recherches, les réalisations que nous avons menées à terme et les développements pratiques qui peuvent prolonger ce travail.

* * *

Notre étude a porté sur la manipulation de l'information dans les réseaux locaux. Nous avons considéré le fichier comme unité de représentation de l'information et étudié les moyens d'utiliser les fichiers pour gérer et communiquer l'information.

Les techniques de conservation, de transmission et d'impression étant relativement sûres, notre travail s'est essentiellement penché sur leur mise en oeuvre pour en tirer le meilleur parti.

Quelques lignes générales se dégagent:

La conception de systèmes informatiques complètement répartis à base de matériels et de noyaux de systèmes conçus dans cette perspective est une alternative réaliste aux systèmes classiques centralisés.

Par contre, l'utilisation de serveurs, qui gèrent le partage de ressources communes ou qui réalisent des fonctions d'intérêt général, amène une certaine spécialisation des sites et entraîne la fragilité de l'ensemble en cas de défaillances localisées.

La construction des catalogues de fichiers répartis pose les problèmes de sécurité des données conservées et de mise à jour cohérente en cas de redondance pour assurer cette sécurité. Une redondance simple assurée à la base par la double écriture et réalisée localement par le matériel permet d'assurer la sécurité des données. Leur duplication sur le réseau assure leur disponibilité sans provoquer trop de mises à jour réparties.

En outre, les débits et temps d'accès aux unités périphériques à travers les réseaux locaux étant équivalents à ceux que l'on observe localement, les accès à des catalogues ou données distants ne sont pas pénalisés.

La fiabilité accrue des média de communication et la rapidité de la transmission réduit l'intérêt des reprises d'erreurs. En effet, il devient plus intéressant de laisser l'application recommencer un transfert, ce qui sera peu fréquent, que de développer des mécanismes de reprise au niveau transport.

* * *

Le réseau dont nous avons disposé est conçu comme un support de recherche en même temps qu'un outil de communication au sein du laboratoire. Des besoins se sont manifestés autant pour des moyens permettant la manipulation directe de l'information par les usagers, que pour des services de base nécessaires à l'élaboration d'applications de grande envergure.

Les outils qui ont été mis en oeuvre au cours de cette étude permettent maintenant d'offrir la plupart des services attendus par les usagers du laboratoire. Sur

l'ensemble des matériels connectés au réseau, les usagers disposent de moyens qui leur permettent de stocker et de manipuler des fichiers d'information, de gérer les catalogues répertoriant ces fichiers et de faire imprimer le contenu des fichiers. Un service, qui peut être incorporé aux applications, décharge celles-ci de la réalisation effective du transfert des fichiers.

Cependant les services fournis ont été réalisés sur un ensemble de matériels qui n'a pas été conçu pour former un système réseau. Il nous est apparu qu'il est nettement plus avantageux, pour réaliser un système réparti sur un réseau local, d'utiliser des noyaux de système et des dispositifs de communication conçus dans ce but.

* * *

Aussi, ce travail a pris fin en laissant ouvertes quelques issues :

Le réseau local sur lequel nous avons travaillé, DANUBE, permet un débit de 1 Mégabits par seconde sur le support de communication. Mais les moyens d'accès à ce support (liaison série V24 vers le communicateur) limitent le débit effectif à quelques milliers de bits par seconde. Une réalisation du réseau avec des communicateurs intégrés aux ordinateurs connectés doit assurer un débit convenable et permettre de vérifier la validité des réalisations sur un réseau utilisant réellement les capacités qu'on lui attribue. Ne disposant pas des outils qui permettent de valider les protocoles que nous avons conçus, il nous a été difficile de les affiner car nous n'avons pas pu les tester réellement aux vitesses théoriques possibles.

Un service de transport basé sur la diffusion fiable des informations sur le réseau local a été étudié <IMAG 82>. Son utilisation rendra possible une réalisation plus efficace des services offerts. Un exemple concernant la messagerie est la diffusion à plusieurs abonnés d'une même information. Elle est actuellement réalisée par le service de messagerie en répétant des envois point à point. Lorsqu'un service de transport offrira la diffusion fiable, le service de transfert de fichiers devra exploiter cette possibilité et fournir lui-même ce service.

La diffusion de fichiers s'adapte bien aux services offerts par le protocole de diffusion fiable. Ce protocole rend possibles les rendez-vous de l'émetteur et des destinataires, asservit le débit de l'émetteur à celui du plus lent des récepteurs, permet la définition des points de reprise et détecte la panne d'un récepteur. Le principal problème reste la définition des protocoles de reprise; ces protocoles pourront cependant se baser sur la numérotation unique de tous les messages diffusés. En cas de panne de l'émetteur les protocoles seront similaires à celui présenté dans le chapitre V. En cas de panne d'un récepteur, il faut choisir entre la poursuite de la diffusion vers les récepteurs restants en assurant une reprise particulière pour le récepteur tombé en panne et la suspension du transfert en diffusion suivie de sa reprise lors du retour du récepteur manquant.

La gestion et la communication d'information devront aussi s'adapter à l'interconnexion des réseaux locaux. Les échanges et les accès à l'information pourront se faire d'un réseau à l'autre. Les problèmes à surmonter se posent à différents niveaux du modèle d'architecture pour l'interconnexion des systèmes ouverts. Outre les problèmes posés par l'interconnexion physique des réseaux locaux et

l'hétérogénéité des protocoles de transport de chaque réseau, il faut harmoniser les outils et les services qui existent de part et d'autre de l'interconnexion. Des passerelles (gateway) assurent le rôle de relais de conversion entre les protocoles de transport des réseaux locaux et le protocole du réseau général qui les relie. Ces passerelles contribuent aussi à assurer la collaboration des services interconnectés concernés. Ici encore, la collaboration des services entre réseaux locaux est plus aisée si ces services ont été au départ conçus dans l'optique d'une interconnexion ultérieure.

BIBLIOGRAPHIE

ADI-TIF 81

Quint V., Richy H., Sasyan S.
Etude et mise en oeuvre de protocoles pour des applications bureautiques sur un réseau local, Compte-rendu de fin d'étude d'une recherche financée par l'Agence De l'Informatique
Annexe 1: Transfert et Impression de Fichiers
juin 1981

AMOEBBA 82

Ianenbaum A.S., Mullender S.J.
Amoeba : a capability based distributed operating system
Local Network and Distributed Office Systems, London, May 1981

Anderson 75

Anderson G.A., Jensen E.D.
Computer interconnexion structures : taxonomy, characteristics and examples
Computing surveys Vol 7, n° 4 december 1975

Birrel 80

Birrel A.D., Needham R.M.
A universal file server
IEEE transactions on Software Engineering, Vol SE6 n°5 september 1980

Boggs 76

Boggs D.R., Metcalfe R.M.
Ethernet : distributed packet switching for local computer networks
Communication of ACM, Vol 19, n° 19, july 1976

Chambon 81

Chambon J.F., Jurema M., Niang A.
Banalisation de fichiers dans le cadre d'un centre de calcul réparti en environnement hétérogène
Conception des systèmes télématiques, Nice France
juin 1981

Clark 78

Clark D.D., Pogran K.T., Reed D.P.
An introduction to local area networks
Proceedings of the IEEE, vol 66, n°11 november 1978

DECNET 76

DECNET, Specification for Network Service Protocol
DECNET DIGITAL NETWORK ARCHITECTURE, 15.11.76

DSAI 81

Lampson B.W., Paul M., Siegert H.J.
Distributed Systems, Architecture and Implementation
Lecture notes in computer science, Vol 105
Springer Verlag 1981

FELIX 81

Fridrich M., Older W.
The FELIX file server
Proc. of the eighth Symposium on Operating Systems
Principles, Asilomar, California, December 1981

FTPIG 81

A network independant file transfer protocol, file
transfer protocol implementors group, London GBR
february 1981

FTPSE 80

File Transfer Protocol, Standard ECMA, fifth draft,
october 1980

Gien 77

Gien M.
Proposal for a standard file transfer protocol
EIN/INRIA may 1977

Herbert 81

Dr Herbert A.
The user interface to the Cambridge Model Distributed
System
Proc. 2nd Int. Conf. on Distributed Computing Systems
Paris France april 1981

IMAG 82

Decitre P., Khider A., Vandome G.
Protocole de diffusion fiable pour réseaux locaux
Rapport de recherche IMAG
Décembre 1982

Labétoulle 80

Labétoulle J.
Etude analytique du réseau DANUBE
Doc INRIA, Rel 2.518 Mars 1980

LOCUS 81

Popek G., Walker B., Chow J., Edwards D., Kline C., Rudisin G., Thiel G.
LOCUS, a Network transparent, High Reliability Distributed System
Proc. of The Eighth Symposium on Operating Systems Principles, Asilomar, California, 1981

Lunn 81

Lunn K., Bennett K.H.
A highly reliable distributed filestore directory system
Proc. 2nd Int. Conf. on Distributed Computing Systems, Paris France april 1981

Manning 76

Manning E.G., Howard R., O'Donnell C.G., Pammett K., Chang E.
A UNIX based local processor and network access machine
Computer Networks 1 (1976)

Martin 80

Martin M., Naffah N., Quint V., Scheurer B.
Proposition d'un protocole de transport pour réseaux locaux
Doc KAYAK, Rel 2504.2, Septembre 1980

McDermid 81

McDermid J.A.
Checkpointing and error recovery in distributed systems
Proc. 2nd Int. Conf. on Distributed Computing Systems, Paris France april 1981

McLellan 80

McLellan P.
Network file servers, Locally distributed computing systems
lecture notes, Edinburgh september 1980

Mitchell 81

Mitchell J.G.
the design and implementation of a distributed file system
Design and implementation of complete distributed computing systems, experiences and results
Workshop INRIA avril 1981

Mitchell 82a

Mitchell J.G.
File servers for local area network
Local area network lectures notes
Canterbury april 1982

Mitchell 82b

Mitchell J.G., Dion J.

A comparison of two network based file servers
Communications of the ACM, Vol 25 n° 4, April 82

Naffah 78

Naffah N.

TARO, un système de transmission en boucle pour
réseaux locaux
INRIA Ref. Rel 501, Juin 1978

Naffah 79

Naffah N.

Exemple d'un poste de travail buretique à interface
universelle
Convention informatique, juin 1979

Naffah 80

Naffah N.

KAYAK: a national project for office automation
IKD 1980, International conference for data processing
Berlin octobre 1980

PAPYRUS 80

Quint V., Richy H., Sergeant G.

Propositions pour un système de messagerie et sa mise
en oeuvre sur un réseau local
Congrès AFCET Informatique, Novembre 1980

Popescu-Zeletin 81

Popescu-Zeletin R.

A local computer network in a research environment :
the HMinet
Design and implementation of complete distributed
computing systems, experiences and results
Workshop INRIA avril 1981

Quint 80

Quint V.

Les réseaux locaux : concepts et réalisations
Convention informatique, Paris France septembre 1980

Quint 81

Quint V. et al.

Un réseau local et ses services de base
Conception des systèmes télématiques
Nice France juin 1981

Ritchie 74

Ritchie D.M., Thompson K.

The UNIX time sharing system
Communication of the ACM Vol 17 n° 7 July 74

Satyanarayanan 81

Satyanarayanan M. et al.
Cooperating file systems for a local network
Distributed computing systems
Carnegie Mellon University, march 1981

Shoch 78

Shoch J.
A file transfer protocol using the BSP, 4th edition
Xerox Palo Alto research center july 1978

Shrivastava 81

Shrivastava S.
The newcastle reliability project
Design and implementation of complete distributed
computing systems, experiences and results
Workshop INRIA avril 1981

Spector 82

Spector A.Z.
Performing remote operations efficiently on a local
computer network
Communication of the ACM Vol 25, n° 4 April 82

Sweetman 81

Sweetman D.
A distributed system built with a Cambridge Ring
Local Network & distributed office systems
London may 1981

Taft 78

Taft E., Metcalfe R.M.
PUP specifications
Xerox Palo Alto research center june 1978

TCP 79

INRIA Projets Pilotes, Transmission Control Protocol,
Version 4, February 1979

Ullmann 70

Ullmann P.
Systèmes d'ordinateurs connectés, interface
télétransmission
EMP Fontainebleau, 2 décembre 1970

Wilkes 75

Wilkes M.V.
Communication using a digital ring
Proc. of Pacnet Conf., Sendai, Japan, August 1975

Wupit 81

Wupit A.

Impact de la répartition sur UNIX
Conception des systèmes télématiques
Nice France juin 1981

Zimmerman 80

Zimmerman H.

OSI reference model-The ISO model of architecture for
open systems interconnection
IEEE transactions on Communication COM 28, 4 April 80

A U T O R I S A T I O N D E S O U T E N A N C E

=====

VU les dispositions de l'article 3 de l'arrêté du 16 avril 1974,

VU les rapports de présentation de

Monsieur J. MOSSIERE, Professeur

Monsieur N. NAFFAH

Monsieur SASYAN Serge

est autorisé à présenter une thèse en soutenance pour l'obtention du diplôme de
DOCTEUR-INGENIEUR, spécialité "Informatique".

Fait à Grenoble, le 19 janvier 1983

Le Président de l'I.N.P.-G.

D. BLOCH
Président
de l'Institut National Polytechnique
de Grenoble

P.O. le Vice-Président,

