

77/1

Sc N 77 / 113 A

TRANSFORMATION D'UNE BASE DE DONNÉES CENTRALISÉE EN UNE BASE DE DONNÉES RÉPARTIE SUR UN RÉSEAU D'ORDINATEURS



THÈSE

pour l'obtention

du grade de

DOCTEUR-INGÉNIEUR en INFORMATIQUE

soutenue le 16 novembre 1977

par

Daniel MEYER

Ingénieur E.N.S.E.E.I.H.T.

BIBLIOTHEQUE SCIENCES NANCY 1



D 095 181199 1

Membres du Jury :

Président : M. C. PAIR

Examineurs : M. A. DE JAEGER

M. J.C. DERNIAME

M. J. LE BIHAN

Mme C. ROLLAND

**TRANSFORMATION D'UNE BASE
DE DONNÉES CENTRALISÉE
EN UNE BASE DE DONNÉES RÉPARTIE
SUR UN RÉSEAU D'ORDINATEURS**



THÈSE

pour l'obtention

du grade de

DOCTEUR-INGÉNIEUR en INFORMATIQUE

soutenue le 16 novembre 1977

par

Daniel MEYER

Ingénieur E.N.S.E.E.I.H.T.

Membres du Jury :

Président : M. C. PAIR

Examineurs : M. A. DE JAEGER

M. J.C. DERNIAME

M. J. LE BIHAN

Mme C. ROLLAND

Je remercie Monsieur le Professeur C. PAIR, Président de l'Institut National Polytechnique de Lorraine, pour l'honneur qu'il me fait en présidant le jury de cette thèse.

*Je remercie Monsieur A. DE JAEGER, Directeur-adjoint à la SLIGOS,
pour l'honneur qu'il me fait en participant au jury et pour l'intérêt qu'il
porte à mes travaux.*

J'adresse tous mes remerciements à Monsieur J.C. DERNIAME, Maître de Conférences à l'Université de Nancy I, pour l'attention et les conseils qu'il n'a cessé de me prodiguer dans la direction de mes recherches.

C'est avec Monsieur J. LE BIHAN, Chef de Projet à l'Institut de Recherche d'Informatique et d'Automatique, que j'ai fait mes premiers pas dans la Recherche alors que j'effectuais un stage dans l'équipe "CYCLADES". Devenu responsable du projet "SIRIUS", il est à l'origine du travail qui fait l'objet de cette thèse. Je lui exprime toute ma reconnaissance pour la confiance qu'il n'a cessé de m'accorder.

J'exprime toute ma gratitude à Madame C. ROLLAND, Maître de Conférences à l'Université de Nancy II, pour avoir bien voulu participer au jury.

*Je remercie Monsieur J.L. DESEUILLET, Ingénieur Principal à la
SLIGOS, pour sa collaboration qui s'est traduite par de nombreuses et lon-
gues discussions qui ont permis d'orienter mes travaux.*

*Je remercie également Monsieur L. DARGENT, Etudiant de 3ème Cycle
à l'Université de Nancy I, pour sa participation à mes travaux.*

Le travail qui fait l'objet de cette thèse a été réalisé dans le cadre de deux contrats (Nos 77 128 et 77 075) de l'Institut de Recherche d'Informatique et d'Automatique (I.R.I.A.).

AVANT - PROPOS

Deux démarches permettent d'aboutir à l'existence d'une base de données répartie.

La première consiste à disposer d'une base répartie à partir de bases existantes initialement utilisées par des applications indépendantes. La démarche, tout en préservant les applications initiales, permet de définir des applications qui utilisent les données de plusieurs bases ; pour ces applications l'ensemble des bases forme une base répartie obtenue par "association".

La seconde consiste, pour une application répartie donnée, à transformer une base initiale en plusieurs bases géographiquement dispersées ; l'ensemble de ces bases forme, pour la même application, une base répartie résultant d'une "décentralisation".

C'est précisément dans le contexte de cette seconde démarche que se situe notre étude. Cette démarche intéresse les organisations d'administration ou d'entreprise, ayant par définition une vocation répartie et qui utilisent, par exemple pour des applications de gestion, une base de données centralisée sur un réseau de terminaux. L'emploi d'une base centralisée correspond, en fait, à une centralisation de l'ensemble des équipements informatiques. Or, on s'aperçoit, dans de nombreux cas, qu'une telle centralisation n'est pas adaptée à la vocation répartie de l'organisation et qu'elle empêche toute répartition des décisions et des contrôles. La transformation de la base centralisée en une base répartie est une étape qui doit permettre de rendre la structure informatique plus proche de l'organisation répartie.

Avec cette approche de décentralisation, nous nous intéressons à un type précis d'organisation répartie, l'organisation "étoilée" ; celle-ci consiste, pour l'application, à mettre en jeu d'une part un site central, et d'autre part, des sites régionaux indépendants mais identiques dans

leurs fonctions et ayant chacun une liaison fonctionnelle avec le site central. Nous nous intéressons également à un système de gestion de base de données particulier, le système SOCRATE.

L'objet de notre étude est donc la transformation d'une base de données SOCRATE, utilisée de manière centralisée par une application ayant une organisation "étoilée", en une base de données répartie.

Dans un premier chapitre nous décrivons notre cadre de travail "Les bases de données réparties" ; nous considérons d'abord les "Réseaux" puis les "Bases de Données" et enfin les possibilités de faire coopérer les deux technologies "Réseaux" et "Bases de Données".

Le chapitre II est consacré à la "description d'une application répartie" qui correspond aux deux critères "base centralisée SOCRATE" et "organisation étoilée". Cette application nous sert d'exemple tout au long de notre étude.

Dans le chapitre III nous proposons un "modèle de répartition étoilée" ; ce modèle définit et exprime une architecture de base répartie qui correspond au type d'organisation auquel nous nous intéressons.

Le chapitre suivant concerne les problèmes de "mise à jour et cohérence de la base répartie". Nous définissons des opérations permettant aux utilisateurs de mettre à jour la base répartie. Nous proposons également un mécanisme permettant d'assurer la cohérence de la base répartie face aux actions de mise à jour.

Dans le chapitre V nous définissons et décrivons le "logiciel de répartition et de cohérence" ; celui-ci assure l'interface entre les différentes bases locales qui composent la base répartie. Cet interface assure toutes les fonctions de répartition et de mise en cohérence.

Dans le chapitre suivant nous considérons le problème de la "sécurité" de la base répartie.

Le chapitre VII est consacré à la "mise en œuvre du modèle étoilé" ; nous présentons les incidences de la transformation de la base centralisée en base répartie et considérons la mise en œuvre de nos propositions sur l'exemple d'application décrit dans le chapitre II.

Enfin, le chapitre VIII présente les "extensions et conclusions" de notre étude.

CHAPITRE I

INTRODUCTION AUX BASES DE DONNÉES RÉPARTIES

1. LES RESEAUX
2. LES BASES DE DONNEES
3. RESEAUX ET BASES DE DONNEES

Ce chapitre présente le cadre de notre travail ; celui-ci se situe parmi un ensemble de recherches sur les "Bases de Données Réparties". Ces recherches résultent de la conjonction de deux techniques informatiques jusqu'alors très disjointes : les "Réseaux" et les "Bases de Données".

Les Réseaux téléinformatiques servent de support matériel et logiciel à l'utilisation de données pour le compte d'applications très diverses qui mettent en jeu des sites géographiquement dispersés.

Nous présentons les principaux types de réseaux existants en considérant leur architecture, leur fonctionnement, et leur utilisation.

Les Bases de Données sont des collections de données généralement très grandes et ayant une structure interne relativement complexe. Les données sont gérées par un système de gestion de base de données. Nous nous intéressons aux bases de données en général et à un système particulier : SOCRATE. Nos travaux concernent la répartition d'une base SOCRATE et toutes nos propositions seront exprimées dans la terminologie de ce système ; c'est pourquoi il nous semble intéressant de préciser, dans cette partie, les principaux concepts utilisés dans ce système.

Utiliser une base sur un réseau veut généralement dire que l'on dispose d'une base en un site privilégié et qu'elle est accédée depuis d'autres sites où se trouvent les utilisateurs ; c'est une utilisation "centralisée". Cette forme d'utilisation caractérise la situation actuelle en matière de coopération entre les deux technologies "Réseaux" et "Bases de Données". Cependant, d'autres formes d'utilisation apparaissent : certaines sont encore futuristes, tandis que d'autres sont d'ores et déjà possibles.

1. LES RESEAUX

Un réseau peut être défini comme étant un ensemble de composants informatiques géographiquement dispersés et communiquant entre eux grâce à des équipements de transmission assemblés selon une certaine architecture.

Le plus traditionnel des réseaux est le réseau de terminaux reliant un ensemble dispersé de terminaux à un centre de traitement. Mais la nécessité de répartir des centres de traitement a fait naître les réseaux d'ordinateurs. Parmi ces réseaux, il faut distinguer les réseaux spécialisés qui sont conçus pour une application particulière et les réseaux généraux qui offrent à des utilisateurs, variés et indépendants, la possibilité de se partager les accès à tout un ensemble de ressources.

1.1. Les réseaux de terminaux

Le fonctionnement de ces réseaux est étoilé, c'est-à-dire que les échanges se font entre l'ordinateur central et un terminal ; il n'y a pas d'échange entre les terminaux eux-mêmes. Tous les traitements et les contrôles des échanges sont à la charge du site central. A ce principe de fonctionnement peuvent correspondre diverses architectures physiques ; ce sont des variantes dans l'assemblage et la composition des équipements de transmission et de traitement qui ont pour but d'optimiser certains facteurs tels que le coût de fonctionnement du réseau, le débit des informations, les temps de réponse, la sécurité. Les figures I.1 et I.2 représentent quelques variantes.

Ces architectures font utiliser des équipements tels que :

- des multiplexeurs, qui permettent de "mélanger" les voies de transmission sur une même ligne physique et de les séparer aux deux extrémités.

- des concentrateurs, qui sont, le plus souvent, des mini-ordinateurs programmés, dont le principe consiste à enregistrer temporairement caractères ou messages entre terminal et ordinateur afin d'effectuer un partage temporel optimal d'une même ligne.

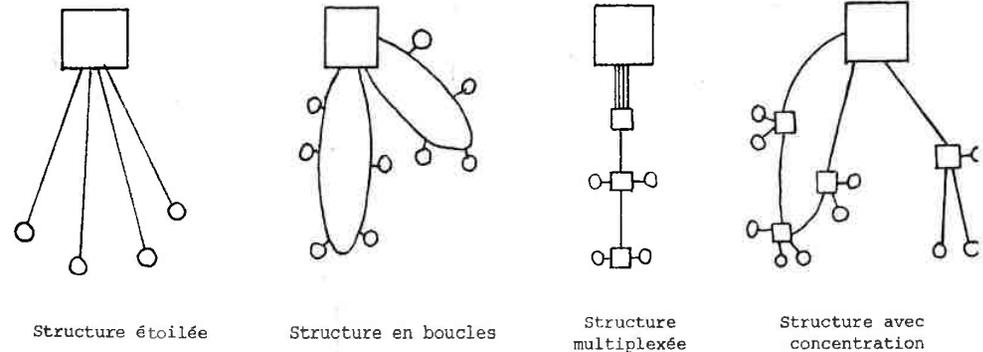


Figure I.1. : Architectures de réseaux de terminaux

L'architecture d'un réseau de terminaux ne concerne pas uniquement la partie "communication" du réseau, c'est-à-dire les lignes et les équipements de transmission ; elle concerne aussi la configuration du site central, c'est-à-dire du centre de traitement qui ne se limite pas toujours à un seul ordinateur comme sur la figure I.1.

Le centre de traitement peut être composé d'un groupe d'ordinateurs interconnectés localement selon les configurations suivantes :

- Ordinateur frontal (figure I.2.a.)

Un ordinateur, séparé de celui qui assure les traitements, assure les contrôles des échanges et toutes les fonctions du réseau ; il communique directement, par un canal, avec l'ordinateur de traitement.

- Tandem (figure I.2.b.)

Le tandem est constitué de deux ordinateurs parallèles, chacun étant susceptible d'assurer les fonctions du réseau ; l'un est en exploitation normale, alors que l'autre est soit en attente, soit occupé à des travaux moins prioritaires.

- Commutateur (figure I.2.c.)

Le commutateur assure la liaison entre les terminaux et l'ensemble des ordinateurs de traitement ; ceux-ci peuvent être identiques et fonctionner tous en même temps, il y a alors équilibrage de charge ; ils peuvent aussi être différents et spécialisés pour certains travaux.

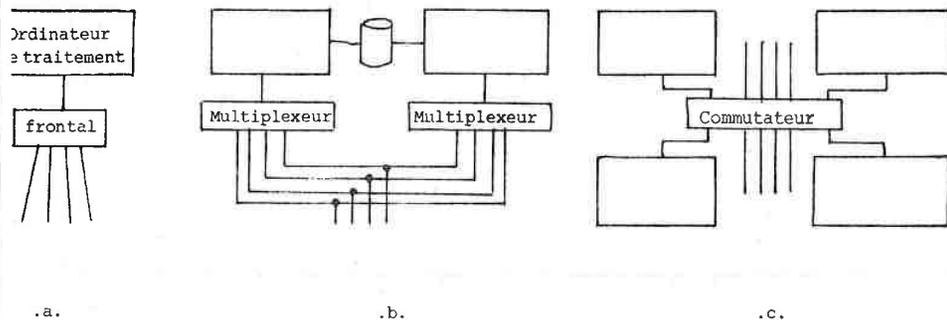


Figure I.2. : Architectures du centre de traitement d'un réseau de terminaux

Ces réseaux peuvent être classés en deux catégories selon leur utilisation :

- les réseaux de terminaux à application spécialisée utilisés dans des domaines comme la commande de processus, les systèmes de réservation de places, les banques, les assurances. (RES-1)
- les réseaux de terminaux de temps partagé qui sont généralement des réseaux commerciaux offrant leurs possibilités de traitement, simultanément, à divers utilisateurs géographiquement très dispersés. (RES-2)

1.2. Les réseaux spécialisés

Les réseaux spécialisés sont des réseaux d'ordinateurs, le plus souvent privés, conçus pour une application particulière.

Le transport des données entre les sites est, le plus souvent, effectué selon le principe de la commutation de messages. A partir de l'émetteur, un message est transmis, de proche en proche à différents ordinateurs, jusqu'à atteindre l'ordinateur destinataire.

Au niveau de chaque ordinateur les fonctions "réseau" (routage des messages, comptabilité et historique du trafic, contrôle de validité des messages entrants) sont exécutés de la même façon que les tâches locales de traitement. La communication entre émetteur et récepteur peut donc être mise "en attente", avec stockage temporaire du message en mémoire secondaire, au niveau de chaque ordinateur intermédiaire. Un tel principe de communication ne se prête donc pas à des applications conversationnelles ou de temps réel. C'est pour cette

raison que la commutation de messages est peu à peu remplacée par la commutation de paquets (RES-3) et que les réseaux spécialisés sont peu à peu transformés en réseaux généraux.

Parmi les réseaux spécialisés les plus connus nous pouvons citer le réseau SINAC de la Société Internationale de Télécommunications (S.I.T.A.), et le réseau de l'Organisation Météorologique Mondiale ; ce sont des réseaux de dimension internationale puisqu'ils "recouvrent" pratiquement tout le globe.

1.3. Les réseaux généraux

Les réseaux généraux sont des réseaux d'ordinateurs qui permettent, dans le cadre d'applications très diverses,

- de faire communiquer des usagers entre eux
- de faire accéder des usagers à des ressources réparties et partagées.

Le transport des données est assuré selon les principes de la commutation de paquets ; elle est une évolution de la commutation de messages (cf. 1.2.) et a été expérimentée sur le réseau ARPA (RES-4) ; elle ne transporte pas de messages complets, au sens où l'entendent l'émetteur et le récepteur de l'information ; mais le transport se fait sous la forme de "paquets" courts de longueur fixe (500 à 2000 bits). Ces paquets résultent, au départ, du découpage d'un message, et sont recomposés, à l'arrivée, pour reformer le message initial.

Un réseau général est généralement composé de deux niveaux reliés par un interface :

- le niveau de communication ou de transport (CYC-1) : il est composé de machines (généralement des mini-ordinateurs) chargées, exclusivement, d'assurer le

transport des données selon la technique de la commutation de paquets ; ce niveau est souvent appelé "réseau de transport" et les machines qui le composent sont appelées des "nœuds" (RES-5).

- le niveau de traitement : il est composé de centres de traitement ou centres participants, chaque centre pouvant être composé de un ou plusieurs ordinateurs et/ou terminaux.

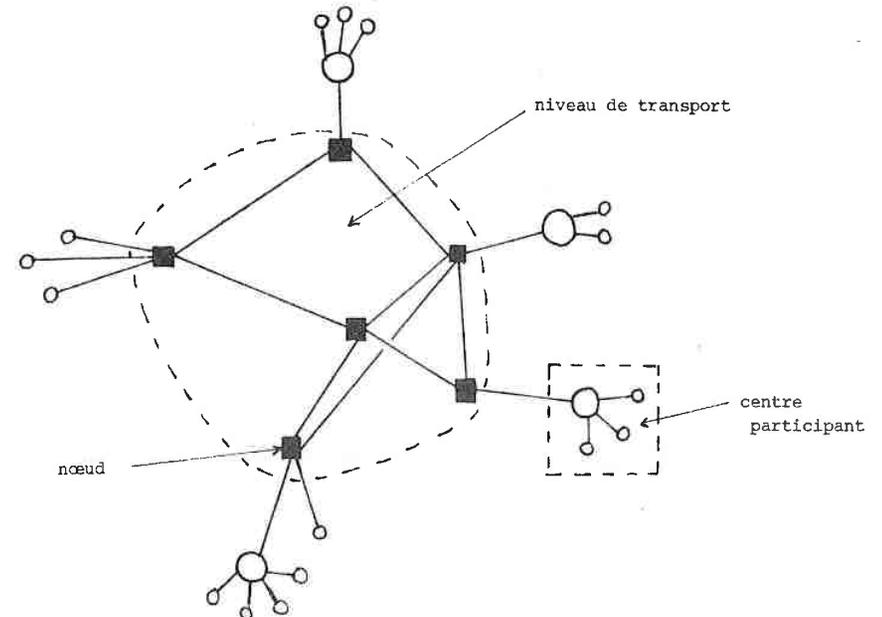


Figure I.3. : Réseau d'ordinateurs

La communication entre les deux niveaux se fait par un interface ; ainsi, sur chaque centre participant, l'ensemble des règles et conventions (CYC-4) (protocoles) qui régissent les communications avec le nœud le plus proche d'une part, et, avec les centres participants distants d'autre part, sont exécutées par un interface appelé "station de transport" dans ARPA et CYCLADES (RES-6, CYC-2, CYC-4) ou "procédure X 25" dans TRANSPAC (RES-7).

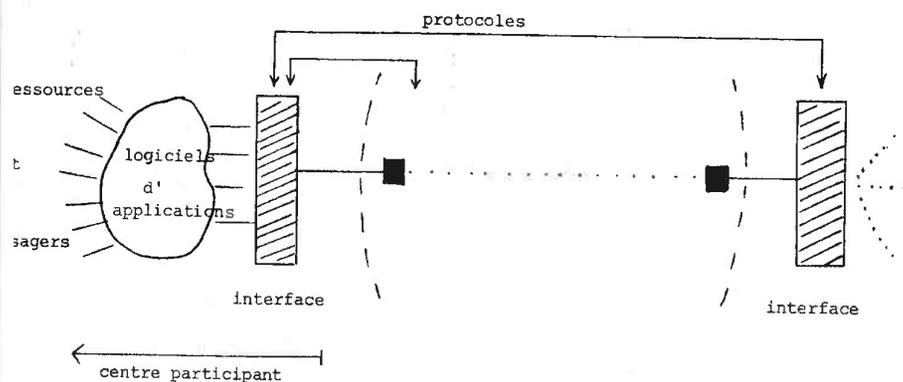


Figure I.4. : Interface de communication entre niveau de traitement et niveau de transport

Les utilisations des réseaux généraux peuvent être regroupées parmi les quatre classes suivantes :

- partage d'équipement physique : périphérique spécialisé, unité centrale, mémoire secondaire.
- partage de logiciel : logiciel spécialisé (compilateur, éditeur de texte, assembleur), sous-système de temps partagé ou de traitement par lots à distance.
- échange de données : de terminal à terminal, de disque à disque.
- bases de données partagées et réparties.

Parmi les réseaux généraux les plus connus, nous pouvons présenter les réseaux ARPANET (RES-4) et CYCLADES (CYC-3).

ARPANET (Advanced Research Project Agency Network) est, depuis 1968, le réseau le plus avancé et le plus important ; il constitue l'exemple type des réseaux hétérogènes de grosses configurations.

Sa partie "communication" se présente sous la forme d'un réseau maillé d'environ quarante nœuds appelés I.M.P. (Interface Message Processor) reliés entre eux par des lignes téléphoniques louées de 50.000 bauds. Les nœuds sont des mini-ordinateurs HONEYWELL DDP-516 ; ils assurent le transport de l'information d'après la technique de la commutation de paquets.

La partie "traitement" est composée d'ordinateurs hétérogènes tels que PDP 10, IBM 370, ILLIAC IV, CDC 7600. Ces machines constituent les centres participants appelés "hosts".

Le succès d'ARPANET l'a conduit récemment à s'étendre en Suède et en Angleterre.

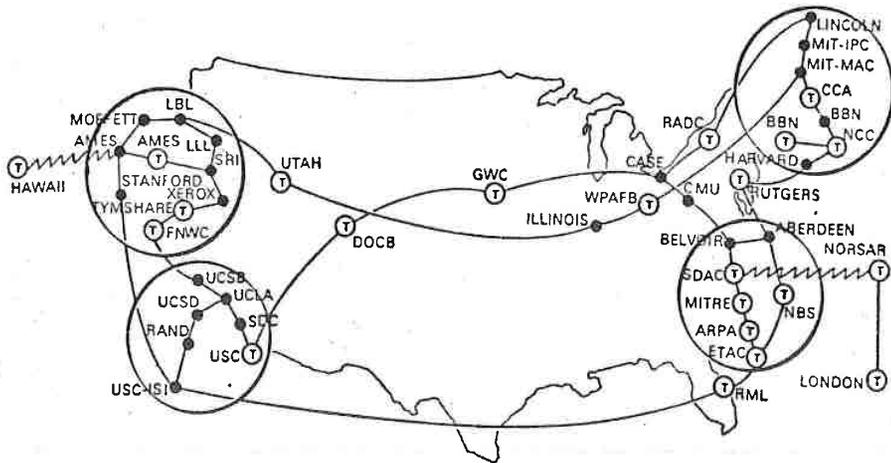
CYCLADES est un réseau général hétérogène, débuté en 1972, faisant l'objet d'un projet pilote français destiné à expérimenter en vraie grandeur l'utilisation et l'exploitation de ce genre de réseau.

Sa partie "communication", appelée CIGALE (CYC-1), est un sous-réseau maillé utilisant, pour le transport des données, la commutation de paquets ; les nœuds sont des mini-ordinateurs MITRA 15 qui sont reliés entre eux par des lignes louées de 4800 ou de 48000 bauds.

Sa partie "traitement" comporte des calculateurs de constructeurs aussi divers que CII, PHILIPS, IBM, SIEMENS.

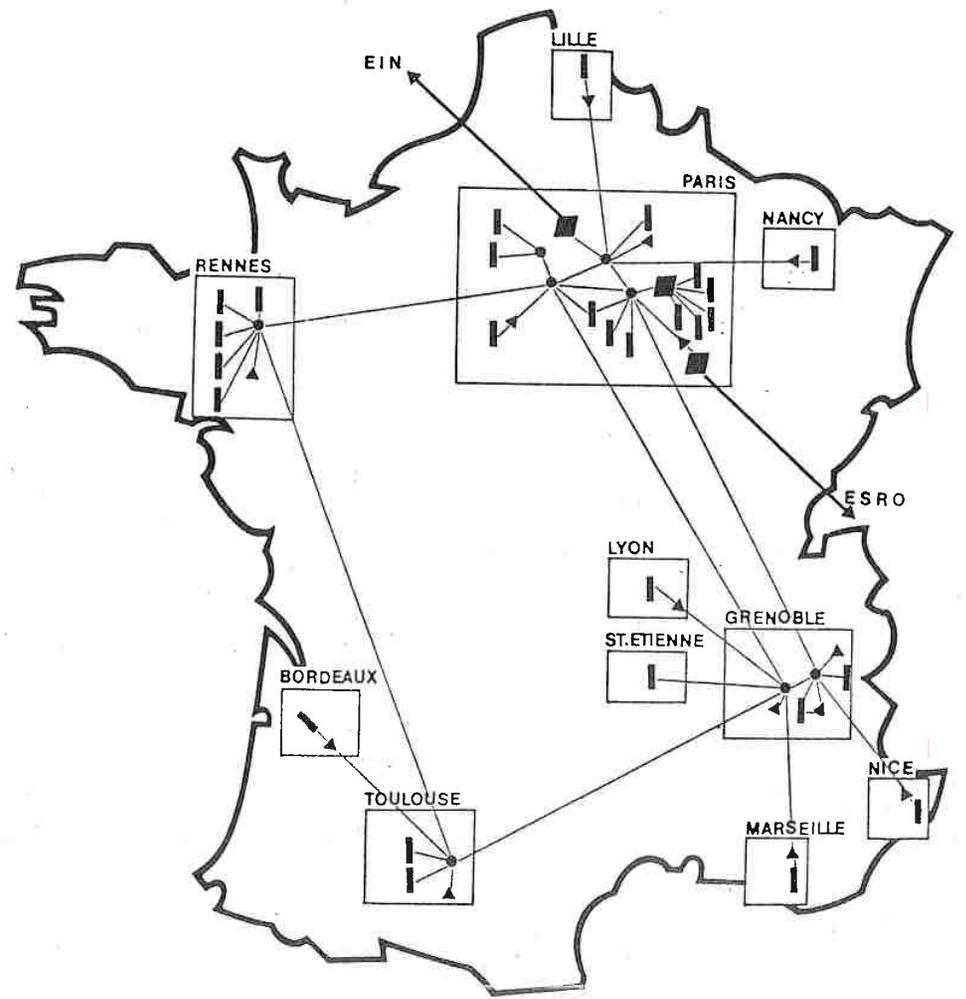
En plus des applications traditionnelles des réseaux d'ordinateurs, CYCLADES veut permettre aux différents usagers l'accès à des bases de données localisées ou réparties.

La maîtrise d'œuvre du projet est à l'I.R.I.A. ; les PTT et la CII, ainsi que les centres participants, contribuent à la réalisation technique.



- lignes téléphoniques louées
- ~~~~ transmission par satellite
- I.M.P.
- Ⓣ T.I.P. (Terminal Interface Processor)

Figure I.5 : Configuration géographique du réseau A.R.P.A. (ARPANET)



- Liaison avec réseau externe
- Liaison téléphonique
- ▶ Concentrateur
- Nœud
- ▬ Centre participant
- Point d'accès à un autre réseau
- E.S.R.O. } réseaux européens
- E.I.N. }

2. LES BASES DE DONNEES

2.1. Généralités

2.1.1. Base de données et système de gestion

Une base de données peut se définir comme la solution technique permettant de mémoriser des ensembles de collections de données intégrées, définies, utiles, fiables et cohérentes, organisées sur le support de stockage "indépendamment" de leurs utilisations et des structures de l'entreprise, accessibles en temps utile, facilement exploitables et satisfaisant aux normes de sécurité et de confidentialité (SYS-1).

Une base de données permet des structures de données plus évoluées que celles offertes par les organisations de fichiers classiques. Avec ces derniers les liens sémantiques entre les données s'expriment de manière externe, au niveau de chaque programme d'application. Une base, par contre, mémorise, elle-même, les liens sémantiques sous forme physique : tables d'index, pointeurs, listes inverses.

Travaillant avec un fichier de données, tout programme d'application doit posséder la description des données contenues dans le fichier (par exemple dans la data division d'un programme CØBØL). Le programme, en fonction de cette description, formule des requêtes d'accès physique aux données.

Avec une base de données, tout utilisateur dispose d'un langage de requêtes pour "converser" avec un système de gestion ; celui-ci possède lui-même la description de la base et réalise, en fonction des requêtes logiques (portant sur des types de données), des accès physiques aux valeurs contenues dans la base .

2.1.2. Fonctions d'un système de gestion

Sachant qu'un grand nombre d'utilisateurs peuvent demander des accès (interrogation, mise à jour) simultanément, le système de gestion est un ensemble coordonné de logiciels dont nous allons présenter les principales fonctions.

. Description de la base

Le système de gestion possède une description de la base qui lui permet de connaître l'ensemble des relations logiques qui existent entre les données de la base. Pour établir cette description, au moment où on veut créer la base, le système de gestion offre aux utilisateurs un langage de description.

. Manipulation des données

Le système de gestion offre aux utilisateurs des possibilités de manipuler des données soit à partir d'un terminal conversationnel, soit à partir d'un programme. Le langage de manipulation (ou langage de requêtes) doit permettre

- d'extraire des données, de manière ponctuelle ou massive
- de déduire de nouvelles données ou des relations
- d'introduire, de supprimer ou de modifier des données.

Le langage de manipulation est généralement complété par un macro-générateur qui permet de disposer d'un ensemble de programmes précompilés qui contiennent des requêtes ou des ensembles de requêtes associés à des applications particulières. Ces programmes permettent à des non-informaticiens d'accéder facilement au système de gestion sans connaître le langage de requêtes.

. Sécurité des données

Du fait du volume des données traitées, de la diversité des utilisateurs, de la complexité des structures de données, le système de gestion doit assurer la sécurité de sa base en prévoyant d'éventuelles destructions maladroites ; ceci peut être fait par la gestion de "journaux" et par la définition de points de contrôle.

. Confidentialité des données

Le système de gestion doit contrôler les autorisations d'accès à la base ; il identifie l'utilisateur par son nom ou par un mot de passe. Puis le système permet ou interdit certaines actions en fonction de règles édictées à l'avance par le responsable de la base.

2.1.3. Fonctionnement et utilisation

Il existe deux modes de fonctionnement :

- le mode "conversationnel"
- le mode "traitement par lots" (ou "batch")

En mode conversationnel, le système de gestion dispose d'un moniteur chargé d'assurer la gestion des terminaux conversationnels connectés.

Avec ce mode, plusieurs utilisateurs peuvent travailler simultanément et indépendamment sur la même base.

Pour des traitements qui s'accrochent mal d'une exploitation conversationnelle, les systèmes peuvent être utilisés par "lots". Dans ce cas, le terminal conversationnel est remplacé par le couple : lecteur de cartes et imprimante. Les requêtes sont formulées sur cartes et les résultats reçus sur l'imprimante. Mais un seul programme peut travailler à la fois sur une base.

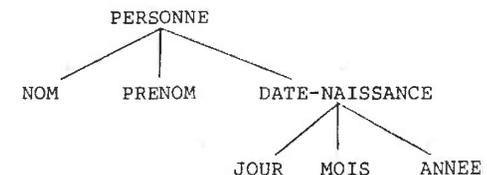
2.2. Les modèles de description

Les systèmes actuels sont caractérisés par les structures de données qu'ils sont capables de gérer et par la représentation de ces structures dans la description de la base ; ainsi, les systèmes sont regroupés en trois classes principales associées à un certain modèle de description qui peut être arborescent, en réseaux, ou relationnel. (SYS-2)

2.2.1. Le modèle arborescent (SYS-3)

Avec ce modèle, les relations logiques s'expriment à l'aide d'un arbre ; les seules relations permises sont des relations hiérarchiques du type "père-fils".

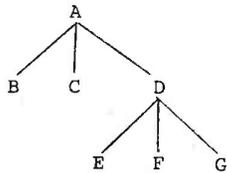
Un arbre représente un groupe dont le nom est porté par la racine de l'arbre tandis que les branches représentent les constituants du groupe ; chaque constituant pouvant, lui-même, être un groupe.



2.2.2. Le modèle réseau (SOC-1, SYS-4, SYS-5)

Ce modèle introduit la notion de référence ; elle permet de disposer de groupes indépendants, non associés dans un même arbre, mais ayant une ou plusieurs relations entre eux.

Supposons que l'on ait l'arborescence suivante :

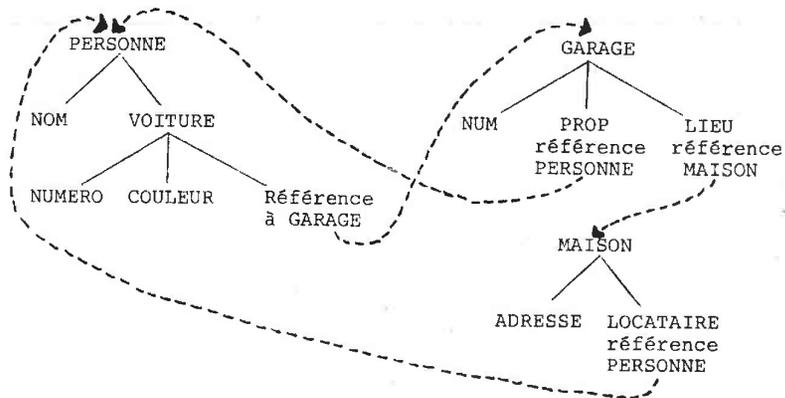


Si l'on veut rendre les deux groupes indépendants, mais en relation entre eux, il faut détruire l'arborescence et introduire une référence :



"----->" est un pointeur logique qui doit avoir une valeur effective dans la base comme toute donnée A, B, C...

Les références permettent ainsi de représenter des structures plus complexes que les arborescences, des graphes :



Ce modèle permet de supprimer un lien défini par référence sans, pour autant, supprimer le groupe référencé.

2.2.3. Le modèle relationnel (SYS-6, SYS-7)

Les modèles de type relationnel sont les plus récents ; parmi les plus connus il faut citer celui proposé par E.F. CODD (I.B.M. San Jose).

La différence essentielle entre ce modèle et le modèle réseau est que dans ce dernier les relations sont représentées explicitement dans la base alors que dans le modèle relationnel elles sont exprimées implicitement par la notion "d'associativité par valeur". Cette notion doit accroître l'indépendance entre les représentations logiques et physiques des données de la base.

Le modèle de CODD représente une relation par une matrice. Supposons des ensembles de données S1, S2, ... Sn (non nécessairement distincts) ; R est une relation entre ces n ensembles si il existe un ensemble de n-uples, chacun ayant son premier élément dans S1, son second élément dans S2, ainsi de suite.

La relation R associe des domaines ; elle est de degré n si elle associe n domaines.

La matrice représentative de R a les propriétés suivantes :

- chaque rangée représente un n-uple de R
- l'ordre des rangées n'a aucune importance
- toutes les rangées sont distinctes
- l'ordre des colonnes est significatif ; il correspond à l'ordre des domaines sur lesquels R est définie

- chaque colonne est libellée par le nom du domaine.

Soit la relation "PIECES-EN-STOCK" de degré 4 qui associe les domaines "NUMERO", "NOM", "COULEUR", "QUANTITE" ; sa matrice sera la suivante :

NUMERO	NOM	COULEUR	QUANTITE
120	ECROU	GRIS	50
165	VIS	BLEU	100
176	RONDELLE	BLANC	25
190	CLOU	GRIS	600
191	PUNAISE	JAUNE	17

← ces 4 valeurs sont en association directe pour la relation "PIECES-EN-STOCK"

Au cours du temps, il est facile d'ajouter ou de supprimer un domaine à une relation.

De même, des n-uples peuvent être ajoutés ou supprimés. Un des domaines possède les valeurs permettant de distinguer et d'ordonner les n-uples : le domaine "NUMERO" dans l'exemple précédent ; ce domaine est appelé "clé-primaire".

2.3. Le système de gestion SOCRATE (SOC-1, SOC-2, SOC-3)

Le système SOCRATE est un système de gestion de bases de données développé à partir d'études menées à l'IMAG de Grenoble sous la direction de J.R. ABRIAL. Il en est résulté un produit industriel dont la réalisation a été assurée par la CII et ECA-AUTOMATION.

2.3.1. Structures de données

Le système SOCRATE permet la description d'arborescences ou de réseaux (cf. 2.2.2.).

. Notion de caractéristique

La caractéristique est la plus petite unité manipulable par le système de gestion ; elle est caractérisée par un nom et elle est représentative de valeurs.

. Notion d'entité

Une entité est un groupe de caractéristiques destiné à représenter un objet dans la base ; l'entité est également caractérisée par un nom.

. Notion de réalisation d'entité

Un ensemble de valeurs représentatives d'un objet représenté par une entité est une réalisation d'entité ; l'ensemble est composé de valeurs, chacune correspondant à une caractéristique de l'entité.

Exemple : NOM, ADRESSE, AGE sont des caractéristiques de l'entité PERSONNE ; DUPONT, NANCY, 36 forment une réalisation de l'entité PERSONNE.

. Type d'une caractéristique

Le type indique la forme de la donnée.

Type mot : chaîne de caractères alphanumériques de longueur inférieure à 30 octets sur laquelle les opérations de comparaison sont seules admises.

Type texte : chaîne de caractères découpée en lignes de texte ; seul peut agir l'éditeur de texte.

Type valeur numérique : valeur comprise dans une certaine plage de valeurs.

Type liste de valeurs : valeur numérique ou alphanumérique prise dans une liste.

Ces quatre types correspondent à des caractéristiques "descriptives" alors que les suivants correspondent à des caractéristiques "structurelles".

Type référence : pointeur vers une réalisation d'entité qu'il permet d'atteindre directement.

Type anneau : lié au type référence, permet à partir d'une réalisation référencée, d'atteindre directement toutes les réalisations d'une entité qui y font référence.

Type inverse : permet d'atteindre une liste de réalisations d'entités préalablement définies par l'utilisateur ; permet la constitution de fichiers inverses sur des entités par rapport à la valeur de certaines de leurs caractéristiques.

. Bloc de caractéristiques

Les caractéristiques peuvent être regroupées en blocs délimités par les mots clés DEBUT et FIN ; un bloc peut contenir d'autres blocs ce qui permet de décrire une hiérarchie.

. Entités et relations entre entités

Une entité est décrite comme un bloc ; elle se compose de caractéristiques simples, de blocs ou d'autres entités. L'imbrication d'entités permet de décrire des structures hiérarchiques linéaires ou arborescentes.

Les caractéristiques de type référence et anneau permettent de relier des entités indépendamment de leurs imbrications.

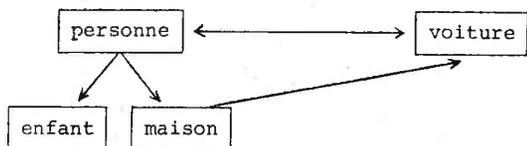
. Caractéristiques du type entité

C'est une caractéristique du type bloc susceptible d'avoir plusieurs réalisations.

2.3.2. Exemple d'une structure de données SOCRATE

```
DEBUT
  ENTITE (3000) personne
    DEBUT
      nom MOT (12)
      véhicule ANNEAU
      numéro-sec-soc MOT (16)
      sexe (2 8)(masculin féminin)
      naissance
        DEBUT
          lieu MOT (12)
          date
            DEBUT
              jour DE 1 A 31
              mois DE 1 A 12
              an DE 1950 A 1999
            FIN
          FIN
        FIN
      FIN
    ENTITE (20) enfant
      DEBUT
        prénom MOT (12)
        âge DE 0 A 21
      FIN
    ENTITE (10) maison
      DEBUT
        adresse MOT (12)
        garage REFERE UNE voiture
      FIN
    FIN
  sans-enfant INVERSE TOUTE personne
  ENTITE (5000) voiture
    DEBUT
      numéro MOT (12)
      marque MOT (8)
      propriétaire REFERE véhicule DE UNE personne
    FIN
  FIN
```

Cette structure peut être représentée par le schéma suivant :



Elle permet à partir d'une voiture de trouver son propriétaire et à partir d'une personne de trouver ses enfants, ses maisons et ses voitures ; pour chaque maison on peut connaître la voiture qu'elle gare.

"Sans-enfant" permet d'accéder directement, sans consultation de la caractéristique "enfant", à toutes les personnes qui n'ont pas d'enfant.

2.3.3. Espace virtuel

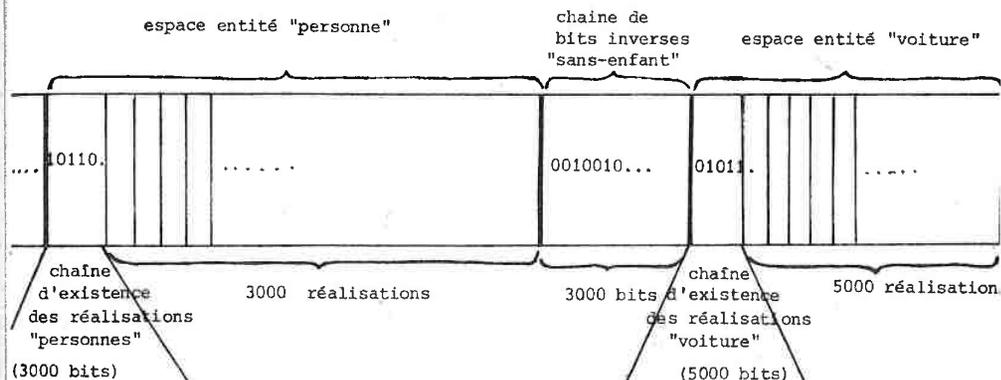
L'exemple précédent représente, telle qu'elle a été définie par l'utilisateur, la description logique d'une structure des données "personne", "voiture", "maison", etc...

Les valeurs associées à ces éléments logiques sont stockées sur les supports physiques de la base.

Le système doit donc passer de la représentation logique à la représentation physique des données. Dans le système SOCRATE, ce passage s'effectue par l'intermédiaire d'un espace virtuel.

L'espace virtuel est un espace linéaire de taille infinie qui est structuré à partir de la description logique des données.

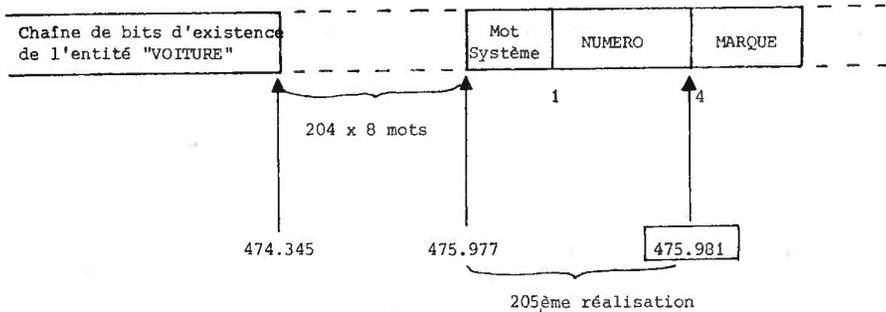
Si l'on considère l'exemple précédent, l'espace virtuel sera divisé en deux zones correspondant aux réalisations des entités principales : personne et voiture ; une zone supplémentaire est consacrée à la chaîne de bits "inverses" : autant de bits que de réalisations "personne" ; tout bit égal à "1" représente une réalisation de personne n'ayant pas d'enfant.



Chaque espace d'entité commence par une chaîne d'existence ; autant de bits que de réalisations prévues. Un bit égal à 1 signifie que la réalisation est existante. Puis la zone contient des espaces prévus pour les différentes réalisations de l'entité ; chacun de ces espaces commence par un mot système pouvant contenir un pointeur ou une "tête" d'anneau.

Soit le calcul de l'adresse virtuelle de la MARQUE de la VOITURE stockée dans la 205ème réalisation :

$$\begin{aligned}
\text{adresse virtuelle} &= \left[\begin{array}{l} \text{adresse virtuelle} \\ \text{de l'entité} \\ \text{VOITURE} \end{array} \right] + (205-1) \times \left[\begin{array}{l} \text{longueur} \\ \text{de l'entité} \\ \text{VOITURE} \end{array} \right] + \left[\begin{array}{l} \text{adresse} \\ \text{relative} \\ \text{de la} \\ \text{caractéristique} \\ \text{MARQUE} \\ \text{dans} \\ \text{l'entité} \\ \text{VOITURE} \end{array} \right] \\
&= 474.345 + 1.632 + 4 \\
&= 475.981
\end{aligned}$$



2.3.4. Espace réel

L'espace réel qui correspond à la mémoire secondaire réelle est très inférieur à l'espace virtuel.

Il est découpé en pages de 256 mots, les pages sont divisées en sous-pages de 8 mots dont 1 mot d'en-tête et 7 mots de données. Il y a correspondance entre sous-page dans

l'espace virtuel et sous-page dans l'espace réel. Une donnée peut s'étaler sur plusieurs sous-pages et une sous-page peut contenir plusieurs données.

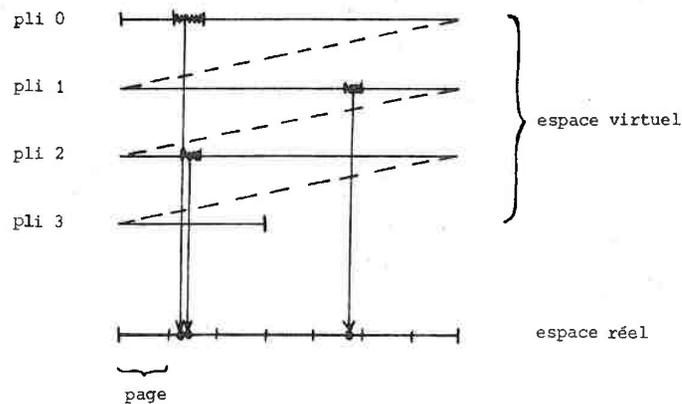
La page correspond à l'enregistrement physique que le système de gestion de la mémoire virtuelle transfère de la mémoire centrale vers la mémoire secondaire et inversement.

2.3.5. Correspondance entre espace virtuel et espace réel

Le système effectue une "application" de la mémoire virtuelle sur la mémoire réelle. La mémoire virtuelle, bien que très grande, étant très "creuse" on peut, en n'envoyant sur la mémoire réelle que les parties occupées de la mémoire virtuelle, effectuer une telle application.

Toute donnée possède une adresse virtuelle et une adresse réelle ; toute adresse réelle est identifiée par le triplet composé du numéro de page, du numéro de sous-page dans une page, et du numéro de mot dans une sous-page.

Schématiquement, l'application se représente par une projection de l'espace virtuel, décomposé en plis égaux, sur l'espace réel :



La projection se fait par sous-page entière.

On s'aperçoit que cette méthode peut entraîner des collisions : deux sous-pages virtuelles peuvent être projetées sur une même adresse réelle ; il faut donc créer deux sous-pages réelles, l'une étant déplacée sur un espace réel vide. Ainsi, le mot d'en-tête des sous-pages permet, entre autre, d'indiquer si le contenu d'une sous-page est déplacé ou non et de chaîner les différentes sous-pages qui, à partir d'une même adresse réelle, ont été déplacées.

3. RESEAUX ET BASES DE DONNEES

3.1. Situation actuelle : base centralisée et utilisateurs répartis

Les utilisations actuelles de bases de données sur les réseaux sont des utilisations centralisées ; elles consistent à implanter une base de données et son système de gestion sur un site privilégié d'un réseau, le site central, et de permettre des accès distants. La base est donc centrale et les utilisateurs sont répartis.

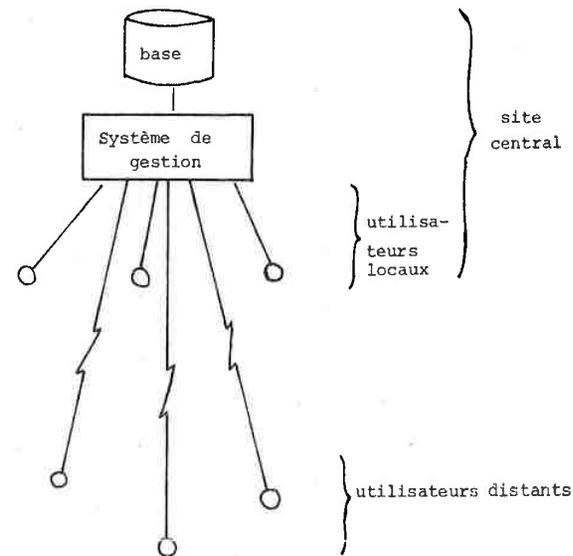


Figure I.7. : Utilisation centralisée d'une base de données

Cette approche, très courante, permet à des utilisateurs distants de ne pas avoir à formuler leurs requêtes par courrier ou téléphone. Le réseau permet un accès direct, à partir de terminaux légers (conversationnels) ou lourds (traitement par lots), soit sous la forme d'interrogation directe, soit sous la forme de dépôt de requêtes avec traitement différé. Généralement, des unités de stockage sont rattachées aux sites distants, ce qui leur permet de recevoir, par le réseau, des ensembles de données, extraits de la base centrale, destinés à des traitements purement locaux. Dans le chapitre II nous décrirons un exemple d'application utilisant une base centralisée.

3.2. Possibilités futures : utilisateurs, systèmes, bases répartis

Avec l'essor des réseaux généraux (cf. I.1.3.), leur fiabilité et les temps de transmission étant satisfaisants, apparaissent des possibilités nouvelles pour l'utilisation des bases de données. Le réseau général permet de faire communiquer des centres de traitement qui sont répartis. De ce fait, il est possible de considérer des utilisateurs, des systèmes de gestion, des bases de données qui soient répartis et que l'on puisse faire communiquer pour les besoins d'applications particulières. Ainsi, un réseau général rend possible des configurations, telles que celles représentées par la figure I.8, qui sont fondées sur le partage des accès à une base par plusieurs systèmes (b, d) et sur la gestion de une ou plusieurs bases distantes par un même système (a, c, d).

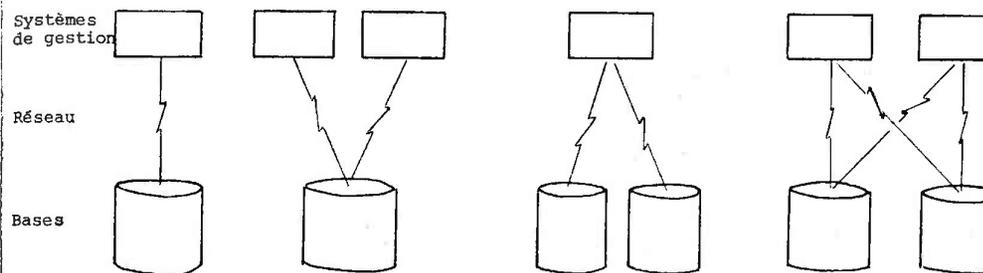


Figure I.8. : Configurations "systèmes et bases répartis"

Evidemment ces configurations permettent toujours les utilisations distantes évoquées dans le paragraphe précédent : le réseau permet toujours de "séparer" utilisateurs et système(s) de gestion.

Mais les systèmes de gestion actuels ne sont pas adaptés à de telles configurations qui détruisent la structure mono-site habituelle de l'ensemble "utilisateurs, système, et base de données". Pour l'instant, ces configurations restent futuristes, mais de nombreuses recherches sont entreprises pour permettre d'orienter les nouveaux systèmes vers ces possibilités de répartition (BDR-4).

3.3. Possibilités actuelles : ensembles "système + base" répartis

Il est possible, dès à présent, de concevoir et de réaliser des configurations réparties qui utilisent les systèmes de gestion actuels, sans les remettre en cause, et qui préservent, sur chaque site, l'unité de l'ensemble "système + base". Ces solutions consistent à faire communiquer de tels ensembles afin de disposer d'un contexte global qui soit réparti tout en préservant le contexte local de chaque site. (BDR-8)

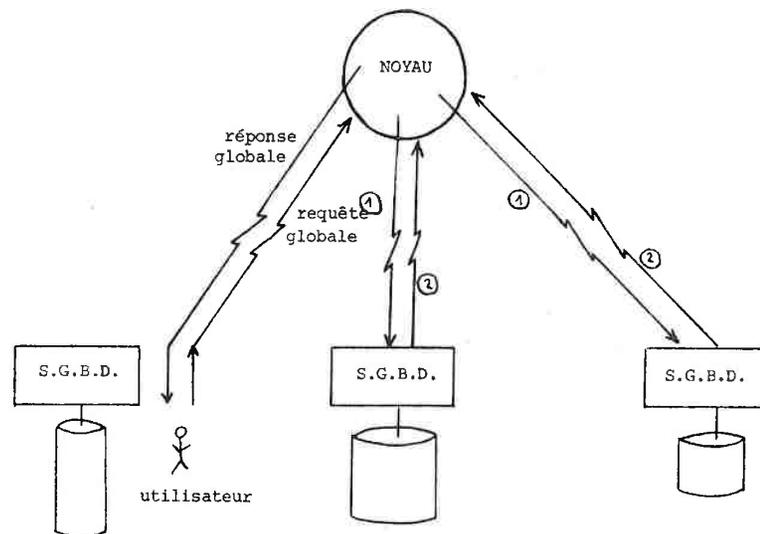
Ces solutions sont réalisables car elles ne nécessitent pas de modification fondamentale ou de refonte des systèmes de gestion locaux ; le contexte global n'est pas assuré par les systèmes locaux eux-mêmes mais par des logiciels d'appoint qui réalisent les interfaces entre les systèmes. Pour le contexte global, l'ensemble des bases de données forment une base répartie.

Le contexte global réparti peut être réalisé de diverses manières ; nous citerons deux exemples d'études et de réalisations actuellement en cours dans le cadre du projet SIRIUS de l'IRIA (BDR-2, BDR-3).

1) L'interface entre les différents ensembles "SGBD + BD" est assuré par un "noyau" central (BDR-5)

Sur chaque site, un utilisateur émet deux types de requêtes : soit des requêtes locales adressées directement au SGBD local et qui concernent la base locale, soit des requêtes globales qu'il transmet, par le réseau, au "noyau". Celui-ci recevant une telle requête vérifie si elle concerne une ou plusieurs bases locales. Ensuite, il décompose la requête en requêtes locales qu'il transmet aux SGBD locaux concernés ; en retour, le noyau reçoit les

différentes réponses et forme une réponse globale qu'il transmet à l'utilisateur initial.



- ① requêtes locales issues de la décomposition de la requête globale
- ② réponses locales destinées à former la réponse globale

Figure I.9. : Interface central assuré par le "noyau"

2) L'interface entre les différents ensembles "SBDB + BD" est assuré, de manière répartie, par des "stations de répartition" (BDR-6)

Chaque site dispose d'une station de répartition ; un utilisateur adresse toutes ses requêtes à ce logiciel local. Celui-ci vérifie si la requête est locale ou si elle est globale c'est-à-dire si elle intéresse une ou plusieurs autres bases locales. Dans ce cas, la station

émet des requêtes aux autres stations qui les font exécuter par leur SGBD local ; de la même façon que le "noyau" de l'exemple précédent, une station récupère des réponses distantes et forme une réponse "globale" qu'elle remet à l'utilisateur.

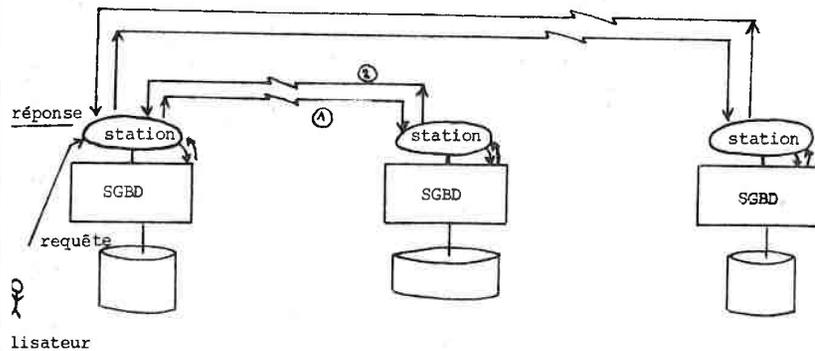


Figure I.10. : Interface réparti assuré par les "stations de répartition"

Dans les deux études une "vue globale" de la base répartie est nécessaire ; dans la première, elle est utilisée par le "noyau", dans la seconde par les "stations de répartition". Cette vue globale est une "synthèse" de l'ensemble des descriptions logiques des différentes bases locales qui composent la base répartie. Des études sont en cours pour définir un modèle capable de représenter une "vue globale" à partir de modèles de description hétérogènes ; un tel modèle de description "globale" permettrait de faire coopérer des ensembles "SGBD + BD" qui soient hétérogènes (BDR-7).

CHAPITRE II

DESCRIPTION D'UNE APPLICATION RÉPARTIE

1. ORGANISATION DE LA GESTION COMPTABLE ET FINANCIERE DU C.N.R.S.
2. LE RESEAU DE COMMUNICATION
3. LA BASE DE DONNEES
4. UTILISATION CENTRALISEE DE LA BASE
5. INCONVENIENTS DE LA CENTRALISATION

Dans ce chapitre, nous nous intéressons à la description de l'application de gestion comptable et financière de l'administration du C.N.R.S. (*).

Cette description a pour but de présenter un exemple d'application répartie qui utilise une base de données SOCRATE centralisée et qui fonctionne selon une organisation "étoilée".

Pour décrire cette application nous considérons successivement l'organisation de la gestion comptable et financière, l'architecture du réseau utilisé, la base de données, et, l'utilisation centralisée de celle-ci sur le réseau. Nous présentons également les principaux inconvénients qui résultent de la centralisation de la base de données ; ces inconvénients ne doivent, cependant, pas être considérés comme propres à l'exemple d'application considéré mais plutôt comme spécifiques à l'utilisation centralisée d'une base de données quelle que soit l'application.

(*) Centre National de la Recherche Scientifique.

1. ORGANISATION DE LA GESTION COMPTABLE ET FINANCIERE DU C.N.R.S.

L'administration du C.N.R.S. se compose d'une administration centrale et d'administrations déléguées, chacune de celles-ci représentant une circonscription régionale.

L'administration centrale coordonne, anime et contrôle, l'ensemble de la gestion du C.N.R.S. C'est elle qui a le pouvoir de prendre des décisions ou d'effectuer des opérations dites "nationales" qui concernent plusieurs ou toutes les administrations déléguées.

Les administrations déléguées n'ont qu'un pouvoir "régional" ; elles sont indépendantes entre elles. Cependant, toute administration déléguée est en liaison avec l'administration centrale : il y a échange de données et de résultats de traitements entre l'administration déléguée et l'administration centrale.

Cette répartition et ces relations fonctionnelles définissent ce que nous appelons une organisation étoilée.

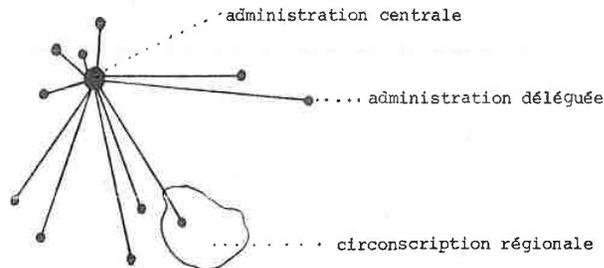


Figure II.1. : Organisation étoilée

Les tâches administratives sont essentiellement de deux ordres :

- la gestion administrative du personnel (G.A.P.)
- la gestion comptable et financière (G.C.F.)

Nous nous intéressons uniquement à l'application constituée par la gestion comptable et financière.

Les principales fonctions comptables et financières qui sont mécanisées, ou en passe de l'être, sont les suivantes :

1) Fonctions d'intérêt régional et central

- Comptabilité budgétaire et analytique
 - . des mandatements
 - . des commandes
 - . des engagements comptables
 - . des dotations

par compte budgétaire, engagements, ou laboratoires et formations, marchés et contrats.

- Régies d'avances (réglement local des fournisseurs et missions) et régies de recettes.
- Inventaires

2) Fonctions d'intérêt régional (circonscriptions et laboratoires)

- Comptabilité de magasins en quantité et prix
- Comptabilité d'ateliers
- Comptabilité des coûts d'entretien (locaux et matériel)
- Comptabilité analytique des laboratoires

Ces fonctions, bien que purement régionales, peuvent nécessiter le transfert de leurs résultats vers l'administration centrale.

La plupart des fonctions mécanisées nécessitent des échanges de données entre administration déléguée et administration centrale, et réciproquement, d'où l'utilité d'un réseau de communication permettant ces liaisons.

Toutes les données nécessaires à ces fonctions comptables sont regroupées dans des bases de données implantées et gérées sur le site central. Le réseau permet des accès distants à ces bases.

2. LE RESEAU DE COMMUNICATION

2.1. Structure logique du réseau

A l'administration centrale correspond le site central du réseau ; aux administrations déléguées correspondent les sites régionaux du réseau.

La structure logique du réseau correspond à l'organisation de l'application ; elle est donc étoilée, c'est-à-dire qu'il n'y a pas de communication entre deux sites régionaux ; il y a uniquement communication entre le site central et chaque site régional

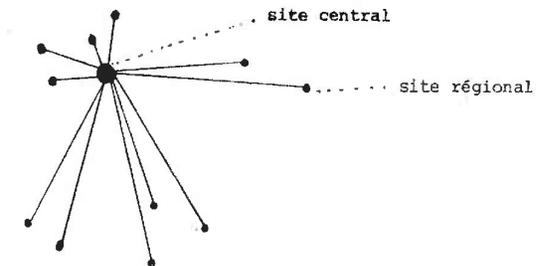


Figure II.2. : Structure logique du réseau

Remarquons que l'architecture du support de transmission est indépendante de la structure logique du réseau ; elle peut être étoilée, c'est le cas actuellement ; mais pour des raisons de coût, de sécurité, de performance, elle peut être différente.

L'ensemble du réseau peut être décomposé en trois parties :

- le site central
- les sites régionaux
- le support de transmission

2.2. Configuration du site central

Le site central est équipé de deux ordinateurs CII-HONEYWELL BULL :

- un ordinateur du type IRIS 60 de 512 K-mots
- un ordinateur du type IRIS 50 de 192 K-mots

Il est également équipé de terminaux de visualisation câblés IRISCOPE 300.

L'interconnexion des périphériques (disques et dérouleurs de bandes, des deux ordinateurs permet d'exploiter en multi-base (cf. II.3) l'application de gestion comptable et financière parallèlement à d'autres applications.

Les exploitations actuelles comportent :

- des exploitations en télétraitement conversationnel dans la journée, et en temps différé de nuit ; les exploitations traitées sur IRIS 50 et IRIS 60, concernant la gestion comptable et financière, nécessitent 118 K-mots de mémoire centrale.
- des exploitations classiques, par lots, traitées sur IRIS 60 de jour et de nuit ; elles nécessitent 84 à 128 K-mots de mémoire centrale.

2.3. Configuration des sites régionaux

La plupart des sites régionaux possèdent un ordinateur CII-HONEYWELL BULL du type GE 58 ; leur taille mémoire variant entre 5 et 10 K-mots selon l'importance des exploitations.

Ces sites sont également équipés de terminaux de visualisation programmables CIT TRANSAC ou HONEYWELL BULL.

Une terminal particulier sert à la fois de frontal à l'ordinateur et de concentrateur/diffuseur aux terminaux de visualisation. Il permet d'assurer les fonctions suivantes :

- routage des messages, assemblage/désassemblage
- adaptation des vitesses des lignes amont et aval
- gestion de la procédure de ligne TMM-VU (mode visualisation)
- gestion de la procédure TMM-RB (remote batch)
- interface entre l'IRIS 60 et les terminaux CIT ou HONEYWELL BULL

La connexion de l'ordinateur régional à ce terminal permet d'effectuer des transferts site central → site régional : fichiers de données stockés sur disques ou données sorties sur l'imprimante régionale.

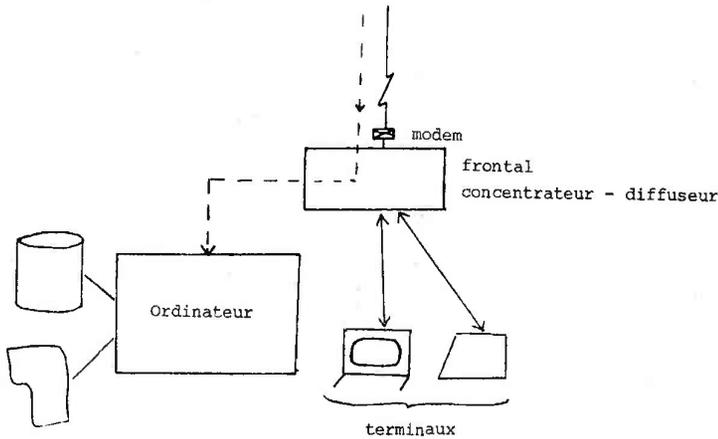


Figure II.3. : Configuration régionale

2.4. Le support de transmission

L'architecture du support de transmission est en cours d'évolution.

Dans la situation actuelle, elle est identique à la structure logique du réseau, c'est-à-dire qu'elle est étoilée, chaque site régional étant directement connecté au site central.

Les lignes permettant ces connexions sont des lignes PTT, louées, à 4 fils, exploitées à 4800 bauds en point à point. La procédure utilisée est un sous-ensemble de TMM-VU, seul le polling aléatoire étant utilisé.

Mais cette architecture n'est plus adaptée aux besoins des utilisateurs et aux applications en cours de développement.

L'architecture doit donc évoluer pour permettre l'intégration de 16 administrations déléguées dans le réseau, c'est-à-dire 100 terminaux de visualisation d'ici 1980.

Cette évolution suppose la mise en place :

- d'un frontal (matériel et logiciel) au niveau des deux ordinateurs centraux.
- de noeuds de commutation situés dans des zones privilégiées et pouvant être accédés, au minimum, par deux chemins différents ; ces noeuds assurent le transport des informations en provenance ou à destination des ordinateurs, des terminaux ou des autres noeuds ; la commutation dans chaque noeud s'effectue en fonction de la destination des informations mais aussi de l'état des lignes et des noeuds voisins.
- de procédures de transmission plus performantes du type "commutation de paquets".
- d'une console RESEAU assurant le contrôle centralisé de l'exploitation du réseau.

3. LA BASE DE DONNEES

L'ensemble des données servant à l'administration du C.N.R.S. est stocké sous forme de bases de données sur le site central. Ces bases sont gérées et exploitées par un système SOCRATE V1-5 qui peut servir simultanément plusieurs applications utilisant les diverses bases.

Les données comptables et financières sont regroupées dans la base de gestion comptable et financière. La figure II.4 en donne la description en représentant les principales entités ainsi que les relations logiques (références, anneaux) qui les lient. Cette base est constituée de sous-ensembles tels que Commande-Facture, Matériel-Inventaire, ou, Fournisseur-Marché.

Parmi ces sous-ensembles, seul le dernier sera considéré dans notre étude ; en effet, nous la limiterons à la "base" constituée par les deux entités "Fournisseur" et "Marché" qui sont reliées par une relation référence-anneau ; cet ensemble logique et physique de données nous est suffisant pour exprimer tous les problèmes à résoudre pour aboutir à une base répartie dans le contexte d'application de type "étoilée". Les tableaux qui suivent la figure II.4 donnent, pour chacune des deux entités, la structure logique et la définition des différentes caractéristiques.

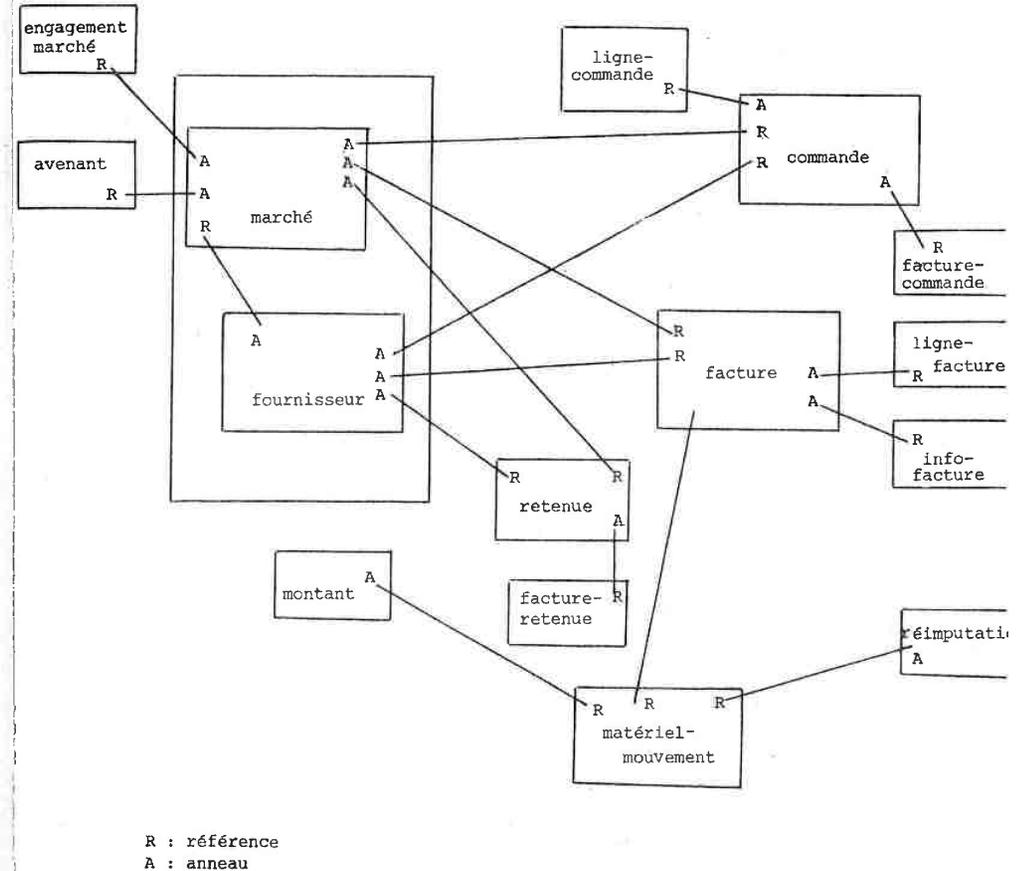


Figure II.4. : Description de la base de gestion comptable et financière

ENTITE (80000) FOURNISSEUR

DEBUT
 A-MARFOU ANNEAU
 A-COMFOU ANNEAU
 A-FACFOU ANNEAU
 COD-FOURNISSEUR DE 1 A 99999999
 AVEC CLE UNIQUE FIN
 N-FOURNISSEUR DE 1 A 999999
 AVEC CLE FIN
 SCOD-FOURNISSEUR DE 1 A 99
 COD-TYPE (11 1) (A R C D G L M N R S X)
 RAISON-SOCIALE MOT (30) AVEC CLE FIN
 LGN-ADRESSF1 MOT (30)
 COD-POSTAL MOT (6)
 LGN-ADRESSE3 MOT (23)
 COD-NATION/REGION DE 0 A 1
 CIRC-CREATRICE MOT (2)
 CIRC-UTIL1 MOT (2)
 CIRC-UTIL2 MOT (2)
 MT-COMMANDE MOT (13)
 COD-PLAFOND DE 0 A 1
 DATE-MAJ DE 101 A 991231
 DATE-C/M DE 101 A 991231
 MOD-PAIEMENT (34 1)
 (0 1 2 3 4 5 6 7 8 A B C D E F G H I J K L M N O P
 Q R S T V W X Y Z)
 COD-DOMICIL (2 1) (B P)
 CLE-DOMICIL MOT (21) AVEC CLE UNIQUE FIN
 BANQUE DE 0 A 99999
 GUICHET DE 0 A 99999
 N-CPTBP MOT (11) AVEC CLE FIN
 LIBELLE-DOMICIL MOT (24)
 CA-HORS-MARCHE MOT (13)
 CA-MARCHE MOT (13)
 CIRC-UTIL3 MOT (2)
 CIRC-UTIL4 MOT (2)
 LGN-ADRESSE2 MOT (30)
 CIRC-UTIL5 MOT (2)
 CIRC-UTIL6 MOT (2)
 CIRC-UTIL7 MOT (2)
 N-SIRET MOT (14) AVEC CLE UNIQUE FIN

FIN

ENTITE (15000) MARCHÉ

DEBUT
 A-ENGMAR ANNEAU
 A-AVMAR ANNEAU
 A-COMMAR ANNEAU
 A-RETMAR ANNEAU
 A-FACMAR ANNEAU

R-MARFOU REFERE A-MARFOU
 DE UN FOURNISSEUR
 N-MARCHE MOT (7) AVEC CLE UNIQUE FIN
 TYP-MARCHE (7 1) (1 2 3 4 5 6)
 ORG-CONTRACTANT MOT (1)
 PARTIE-PRENANTE DE 1 A 999999
 CIRC-CREATRICE MOT (2)
 COD-FOURBENEF DE 1 A 99999999
 MOD-PAIEMENT MOT (1)
 DATE-DEBUT DE 101 A 991231
 DATE-FIN DE 101 A 991231
 QMU-COMMANDE MOT (13)
 QMU-MANDATE MOT (13)
 NP-MANDAT DE 1 A 99999
 DATE-NP-MDT DE 101 A 991231
 DATE-MAJ DE 101 A 991231
 SEUIL-MINI MOT (13)
 SEUIL-MAXI MOT (13)
 MT-MARCHE MOT (13)
 LIB-MARCHE MOT (22)
 PERIODICITE DE 1 A 6
 NB-PA-A-FAIRE DE 0 A 99
 NB-PA-FAITS DE 0 A 99
 DATE-MDT-PA DE 1 A 9912
 MT-PA MOT (13)
 N-COMMANDE MOT (16)
 FILLER MOT (30)
 FIN

ENTITE "FOURNISSEUR"

A-MARFOU anneau des marchés d'un fournisseur
 A-COMFOU anneau des commandes d'un fournisseur
 A-FACFOU anneau des factures d'un fournisseur
 COD-FOURNISSEUR code fournisseur composé de :
 -N-FOURNISSEUR numéro fournisseur (000.000 à 999.999)
 -SCOD-FOURNISSEUR sous-code de domiciliation bancaire
 COD-TYPE code type fournisseur
 - fournisseur français sans déclaration d'honoraires → A
 - fournisseur français avec déclaration d'honoraires → B
 - fournisseur étranger sans déclaration d'honoraires → C
 - fournisseur étranger avec déclaration d'honoraires → D
 - société de nantissement (bénéficiaire) → N
 - laboratoire ou formation C.N.R.S. → L
 - agent missionnaire → M
 - pseudo-fournisseur "paie" → X

	- régisseur français	→ R
	- régisseur étranger	→ S
	- clients	→ G
RAISON-SOCIALE	nom ou raison sociale du fournisseur	
LGN-ADRESSE1	adresse fournisseur (1ère ligne)	
LGN-ADRESSE2	adresse fournisseur (2ème ligne)	
COD-POSTAL	code postal composé du code du département et du code localité	
LGN-ADRESSE3	adresse fournisseur (ville) (3ème ligne)	
COD-NATION/REGION	code national ou régional	
	- fournisseur national : fournisseur utilisé par plus de 7 circonscriptions	
	- fournisseur régional : fournisseur utilisé par 7 circonscriptions maximum	
CIRC-CREATRICE	code de la circonscription créatrice du fournisseur	
CIRC-UTIL1	code de la circonscription utilisatrice du fournisseur	
CIRC-UTIL2	code de la circonscription utilisatrice du fournisseur	
CIRC-UTIL3	code de la circonscription utilisatrice du fournisseur	
CIRC-UTIL4	code de la circonscription utilisatrice du fournisseur	
CIRC-UTIL5	code de la circonscription utilisatrice du fournisseur	
CIRC-UTIL6	code de la circonscription utilisatrice du fournisseur	
CIRC-UTIL7	code de la circonscription utilisatrice du fournisseur	
MT-COMMANDE	montant global des commandes passées à un fournisseur	
COD-PLAFOND	code plafond	
	= 0 : plafond de 100.000 F au-delà duquel le fournisseur a obligation de passer un marché national	
	= 1 : plafond de 100.000 F en-deça duquel le fournisseur n'a pas obligation de passer un marché national	
DATE-MAJ	date de la dernière mise à jour d'un fournisseur	
DATE-C/M	date de création ou modification d'un fournisseur	
MOD-PAIEMENT	code du mode de paiement (caisse, bancaire, CCP, TRESOR)	
COD-DOMICIL	code de domiciliation	
	domiciliation bancaire → B	
	domiciliation postale → P	
CLE-DOMICIL	clé de domiciliation composée de	
- BANQUE	code de la banque	
- GUICHET	code du guichet	
- N-CPTCP	numéro du compte bancaire ou postal	
LIBELLE-DOMICIL	libellé de domiciliation bancaire ou postale	
CA-HORS-MARCHE	chiffre d'affaires hors marché du fournisseur	
CA-MARCHE	chiffre d'affaires sur marché du fournisseur	
N-SIRET	numéro d'identification nationale du fournisseur	

ENTITE "MARCHE"

A-ENGMAR	anneau des engagements relatifs à un marché	
A-AVEMAR	anneau des avenants passés sur un marché	
A-COMMAR	anneau des commandes relatives à un marché	
A-RETMAR	anneau des retenues relatives à un marché	
A-FACMAR	anneau des factures relatives à un marché	
R-MARFOU	référence du fournisseur avec lequel le marché est passé	
N-MARCHE	numéro de marché	
TYP-MARCHE	code du type de marché	
	- contrat d'entretien → 1	
	- marché de clientèle → 2	
	- autre type → 3	
ORG-CONTRACTANT	code de l'organisme contractant	
	- C.N.R.S.	→ A
	- Institut National d'Astronomie et Géographique	→ B
	- Institut National de Physique Nucléaire	→ C
	- Palais de la Découverte	→ D
	- Agence Nationale pour la valorisation de la Recherche	→ E
PARTIE-PRENANTE	code laboratoire ou formation	
CIRC-CREATRICE	code circonscription créatrice du marché	
COD-FOURBENEF	code du bénéficiaire du règlement fournisseur	
MOD-PAIEMENT	code du mode de paiement (caisse, bancaire...)	
DATE-DEBUT	date de début du marché	
DATE-FIN	date de fin du marché	
QMU-COMMANDE	cumul des commandes passées sur un marché	
QMU-MANDATE	montant mandaté relatif à un marché	
NP-MANDAT	numéro du premier mandat	
DATE-NP-MDT	date du premier mandat	
DATE-MAJ	date de mise à jour	
SEUIL-MINI	montant du devis du marché	
SEUIL-MAXI	montant maximal autorisé du marché	
MT-MARCHE	montant du marché	
LIB-MARCHE	libellé du marché	
PERIODICITE	périodicité de paiement automatique	
	- le mandatement se déclenche tous les mois → 1	
	- le mandatement se déclenche tous les 2 mois → 2	
	- le mandatement se déclenche tous les 3 mois → 3	
NB-PA-A-FAIRE	nombre de paiements automatiques à faire	
NB-PA-FAITS	nombre de paiements automatiques effectués	
DATE-MDT-PA	date du premier mandat à effectuer	
MT-PA	montant du paiement automatique à mandater	
N-COMMANDE	numéro de la commande	

4. UTILISATION CENTRALISEE DE LA BASE

4.1. Utilisation centralisée : généralités

L'application considérée fournit un exemple concret de l'utilisation centralisée d'une base au sens où celle-ci a été définie au paragraphe 3.1. du chapitre I.

Le site central est le point de stockage de l'ensemble des données ; il possède la base et son système de gestion, ainsi que les logiciels d'accès aux données (cf. 4.2). Les sites régionaux sont les points d'accès à la base centrale ; les utilisateurs se trouvent sur ces points d'accès, ils sont répartis. Ils utilisent des postes d'accès qui sont des consoles de visualisation, pour effectuer la collecte des données, les mises à jour ainsi que les interrogations. Ces postes sont gérés par le système de gestion de la base centrale auquel ils sont reliés par le réseau.

Remarquons que le site central, en disposant de postes d'accès, remplit comme tout site régional la fonction de point d'accès.

4.2. Manipulation des données de la base centrale

Le personnel gestionnaire, qui constitue l'ensemble des utilisateurs, manipule les données de la base soit "par fournée" soit en mode conversationnel.

1) Manipulation par "fournées" (ou par "lots", ou "batch")

Pour effectuer les traitements, la base de données est manipulée, sur le site central, en mode par lots (cf. I.2.1.3). Un programme de traitement est écrit soit en langage SOCRATE soit en langage évolué ; dans ce cas, il utilise un module de liaison langage évolué-SOCRATE pour accéder à la base (SOC-2, SOC-3).

2) Manipulation en mode conversationnel

La manipulation des données par l'utilisation du langage de requêtes SOCRATE supposant, de la part des utilisateurs, une connaissance du système SOCRATE, une autre forme de manipulation est proposée aux utilisateurs : il s'agit de la forme de manipulation dite "presse-bouton".

Elle permet à l'utilisateur d'appeler depuis sa console, par un simple nom, un programme précompilé (écrit en langage SOCRATE) catalogué dans la base ; un tel programme contient un ensemble de requêtes d'accès et correspond à des transactions conversationnelles ou à des traitements standards. (cf. I.2.1.2)

Ces programmes, dits "macro-requêtes", sont prédéfinis par le macro-générateur du système SOCRATE.

Dans l'approche centralisée de l'application, tous les programmes offerts aux utilisateurs pour accéder aux données de la base sont stockés sur le site central et s'exécutent sur ce même site en liaison avec le système SOCRATE.

Dans la suite pour faire référence à ces macro-requêtes nous parlerons de programmes d'accès aux données.

4.3. Les accès à la base

Nous regroupons les accès à la base centrale en deux types principaux : les interrogations d'une part, les mises à jour d'autre part. Parmi les accès, certains peuvent être conversationnels uniquement, d'autres à la fois conversationnels et par "fournées" ; les accès conversationnels se faisant par l'intermédiaire de programmes d'accès précompilés (cf. 4.2.).

4.3.1. Interrogation de la base centrale

1) Interrogation "par fournées" sur le site central

La base centrale peut être interrogée par des programmes exécutés "par fournées" depuis le site central uniquement. En effet, les programmes de traitements centraux s'exécutent directement à partir des données de la base. Ces programmes peuvent être des programmes SOCRATE, l'accès aux données est donc direct. Mais ils peuvent aussi être des programmes COBOL ; dans ce cas, l'interrogation consiste à faire transiter une réalisation complète d'entité dans un tampon du programme, puis les traitements s'effectuent sur ce tampon ; l'interrogation se fait donc, dans ce cas, de manière séquentielle, réalisation par réalisation.

2) Interrogation conversationnelle

Pour chaque entité, le site central possède un programme SOCRATE d'accès en interrogation. L'utilisateur, d'un point d'accès quelconque, appelle ce programme et celui-ci lui édite alors une réalisation complète d'entité et ce, caractéristique par caractéristique. Au préalable, le programme demande à l'utilisateur de lui fournir le numéro (la clé) de l'objet désiré.

4.3.2. Mise à jour de la base centrale

Par mise à jour nous entendons trois types d'actions sur la base : la création d'une réalisation d'entité, la modification d'une caractéristique, la suppression d'une réalisation.

1) Création d'une réalisation d'entité par saisie

Les créations se font pratiquement uniquement en conversationnel. Pour chaque entité, il existe sur le site central un programme conversationnel de création qui correspond à l'introduction dans la base d'un fournisseur, d'un marché, d'une commande, etc...

L'utilisateur appelle un tel programme et lui fournit une donnée, la clé, de l' "objet" à créer qui permet au programme de vérifier si l'objet n'existe pas déjà dans la base, auquel cas la création est refusée.

Ce contrôle effectué, le programme fournit, un à un, à l'utilisateur les identificateurs des différentes données à saisir ; au fur et à mesure, l'utilisateur transmet la donnée à introduire.

2) Modification d'une réalisation d'entité

Une modification peut être effectuée soit en différé par un programme de traitement soit directement en conversationnel.

a) Modifications centrales "par fournées"

Un programme de traitement exécute des modifications sur un certain nombre de réalisations d'entité.

Si le programme est un programme SOCRATE, il peut accéder directement une caractéristique à modifier.

Si le programme est un programme COBOL alors la modification s'effectue dans un tampon contenant une réalisation complète. Puis, une fois modifiée, l'ensemble de la réalisation est "recopiée" dans la base.

b) Modifications conversationnelles

Sur appel, un programme de modification transmet à l'utilisateur l'ensemble de la réalisation qu'il désire modifier ; pour cela, l'utilisateur a transmis au programme la clé de l'objet désiré.

Puis, l'utilisateur a la possibilité de modifier, une à une, les caractéristiques de la réalisation.

3) Suppression d'une réalisation d'entité

Une suppression peut être exécutée soit par un programme de traitement pour créer des fichiers historiques soit par l'utilisateur lui-même. Dans les deux cas l'ensemble de la réalisation est supprimée.

4.4. Les traitements

Les traitements sont effectués soit par des programmes COBOL sur tous les sites soit par des programmes SOCRATE uniquement sur le site central.

Sur le site central, ils utilisent les données de la base centrale (cf. 4.3.1.1.) et les modifications qui en résultent sont exécutées directement dans la base. Certains résultats de ces traitements intéressent des sites régionaux ; ils peuvent être transmis par le réseau pour être imprimés sur les sites intéressés. Les traitements régionaux s'effectuent à partir de données régionales (cf. 4.5.) et n'intéressent, en général, que le site lui-même. Contrairement aux traitements centraux, les traitements régionaux ne modifient ni les données régionales ni la base centrale.

4.5. Les fichiers de données régionales

Pour permettre aux sites régionaux d'effectuer des traitements qui leur sont propres, des données de la base centrale peuvent être transférées, par le réseau, à un site régional. Ainsi chaque site possède ses fichiers régionaux à partir desquels il exécute ses traitements. Lorsqu'un site régional demande au site central de lui transférer des données, celles-ci sont prétraitées par le site central afin qu'elles puissent être directement utilisées par les programmes COBOL régionaux.

5. INCONVENIENTS DE LA CENTRALISATION (BDR-1)

La centralisation des données entraîne des inconvénients ayant des effets de trois types essentiels.

5.1. Résistance psychologique des utilisateurs à la centralisation

Un premier effet de la centralisation est une certaine résistance psychologique qui s'exprime parmi les utilisateurs régionaux.

En effet, l'administration est décentralisée dans le but de donner aux régions un certain pouvoir de gestion.

Or en même temps que s'effectue une décentralisation des pouvoirs, les utilisateurs régionaux constatent une centralisation des moyens informatiques ; cette centralisation impose des règles de fonctionnement communes à tous ; de ce fait, le pouvoir donné aux régions disparaît sous les contraintes posées par des procédures trop rigides qui ont été conçues par le site central et qui ne tiennent pas compte des problèmes régionaux.

En fait la décentralisation de la gestion devrait s'accompagner d'une plus grande autonomie au niveau informatique ; un premier pas pourrait consister à permettre à chaque administration régionale de stocker ses propres données, qu'elle a saisies, plutôt que de les transmettre au site central pour les lui redemander ultérieurement pour effectuer des traitements régionaux.

5.2. Dégradation du fonctionnement de l'application

La centralisation a des effets "techniques" sur le fonctionnement de l'application de gestion informatique. Ils sont de plusieurs types.

5.2.1. Sécurité

Toutes les données étant stockées sur le site central il en résulte un problème de sécurité à deux niveaux.

- a) La destruction des données de la base centrale entraîne une paralysie de l'ensemble de l'application et de tous les sites.
- b) L'inactivité du site central, pour une raison quelconque, paralyse l'ensemble de la gestion.

5.2.2. Saturation du site central

Pratiquement tous les programmes d'accès conversationnel ou de traitement sont exécutés sur le site central ; d'autre part, ce site "gère" tous les terminaux de l'application, il s'en suit un grand nombre d'accès. Il en résulte que le site central est un point d'accumulation toujours très proche de la saturation. Cette saturation du site central ne permet donc pas d'entreprendre des extensions des traitements ou des opérations actuels, l'application est figée.

5.2.3. Charge du réseau

Les transferts sur le réseau entre les sites régionaux et le site central sont nombreux.

Il faut considérer d'une part tous les accès des terminaux conversationnels régionaux à la base centrale (interrogations et mises à jour) et, d'autre part, les transferts de données dans le sens central-régional pour former les fichiers régionaux.

Le coût d'utilisation du réseau doit donc être relativement élevé.

5.2.4. Dégradation des temps de réponse

Du fait de la saturation du site central et de la charge du réseau, il s'en suit des temps de réponse assez médiocres et une certaine perte de temps dans les attentes sur les sites régionaux.

5.3. Manque de cohérence des données

La centralisation des données provoque des effets sur la cohérence des données.

5.3.1. Incohérence des actions sur la base centrale

Les règles de fonctionnement imposées par le site central ne sont pas toujours interprétées de la même manière sur les différents sites. Il peut en résulter des incohérences au niveau des données de la base centrale.

Par exemple, certains sites traitent avec des fournisseurs locaux pour des achats de faible importance et n'estiment pas utile de transmettre les données correspondantes au système central. Lorsque de telles actions se multiplient, il en résulte des incohérences.

5.3.2. Incohérence entre les données des fichiers régionaux

Les données des fichiers régionaux ne sont pas toutes au même niveau de cohérence par rapport à la base centrale. En effet, les transferts central → régional se font à l'initiative des sites régionaux lorsqu'ils le désirent. Si deux sites demandent des transferts avec une journée d'intervalle il se peut que des mises à jour soient parvenues dans la base centrale

au cours de cette journée. Si elles concernent les données transférées aux deux sites alors le second reçoit les données mises à jour tandis que le premier possède des données non mises à jour. Il en résulte des incohérences au niveau des traitements effectués par les deux sites.

A partir de ces traitements, les sites peuvent "agir" sur la base centrale en fonction des résultats obtenus ; or ceux-ci diffèrent, donc les actions risquent d'être incompatibles et de créer des incohérences du type 5.3.1. dans la base centrale.

CHAPITRE III

LE MODÈLE DE RÉPARTITION ÉTOILÉE

1. APPLICATIONS ET BASES DE DONNEES REPARTIES
2. PRINCIPES DE REPARTITION ETOILEE
3. EXPRESSION ET UTILISATION DU MODELE
4. ARCHITECTURE DE LA BASE REPARTIE ETOILEE

Le but de ce chapitre est de définir l'architecture de la base répartie proposée.

Dans une première partie nous nous intéressons à la correspondance qui peut exister entre une application répartie et une base de données répartie. Une application répartie se caractérise par un certain type d'organisation ; nous présentons brièvement les principaux types possibles. Une base répartie se caractérise par une certaine architecture c'est-à-dire par un certain "assemblage réparti" des données ; nous présentons les diverses possibilités de répartition des données. Construire une base répartie pour une application répartie donnée consiste à faire correspondre une certaine architecture à une organisation donnée ; cette correspondance s'exprime par un "modèle de répartition".

Dans la suite du chapitre nous élaborons un modèle de répartition particulier, le modèle étoilé. Ce modèle permet de définir une architecture de base répartie qui convienne à des applications ayant une organisation "étoilée", en particulier à l'application décrite dans le chapitre précédent. Nous considérons d'abord les principes de répartition mis en œuvre par le modèle étoilé ; ceux-ci conduisent à la définition d'une base centrale et de bases régionales ainsi qu'à la définition d'axes central-régional représentant une liaison entre chaque base régionale et la base centrale. Puis nous considérons l'expression et l'utilisation du modèle ; cinq types de répartition sont définis et sont affectés aux caractéristiques des entités qui composent la base répartie.

1. APPLICATIONS ET BASES DE DONNEES REPARTIES

1.1. Application répartie

1.1.1. Généralités

L'application informatique répartie peut être définie comme étant une application qui met en œuvre plusieurs sites, reliés entre eux par un réseau, sur lesquels sont réalisés le stockage des données (points de stockage), les accès aux données (points d'accès) et les traitements (points de traitement). Une telle application s'oppose à une application locale qui s'exécute sur un seul site qui joue le triple rôle de point de stockage, de point d'accès et de point de traitement.

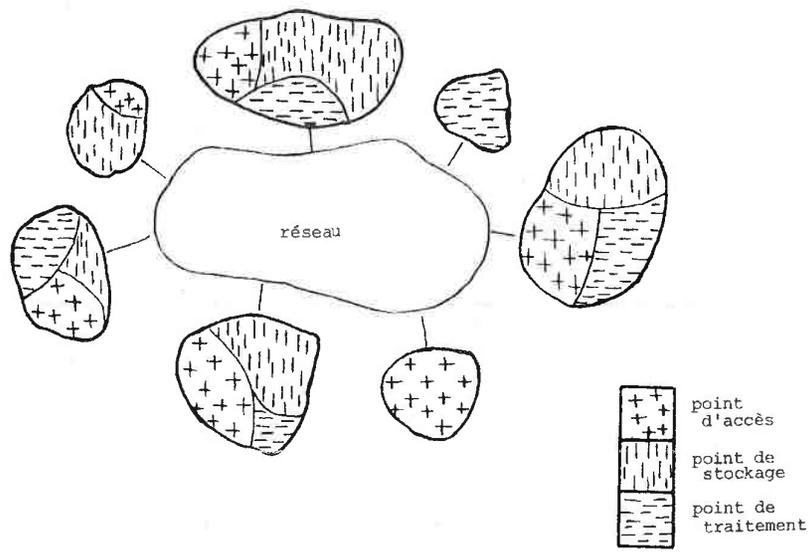


Figure III.1. : Application répartie

Les fonctions de stockage, d'accès ou de traitement peuvent être équitablement réparties entre les différents sites comme c'est le cas sur la figure 1. Par contre il peut arriver que toutes les données soient stockées sur un site, que tous les traitements se fassent sur un site. Dans ce cas l'application répartie peut être qualifiée de centralisée.

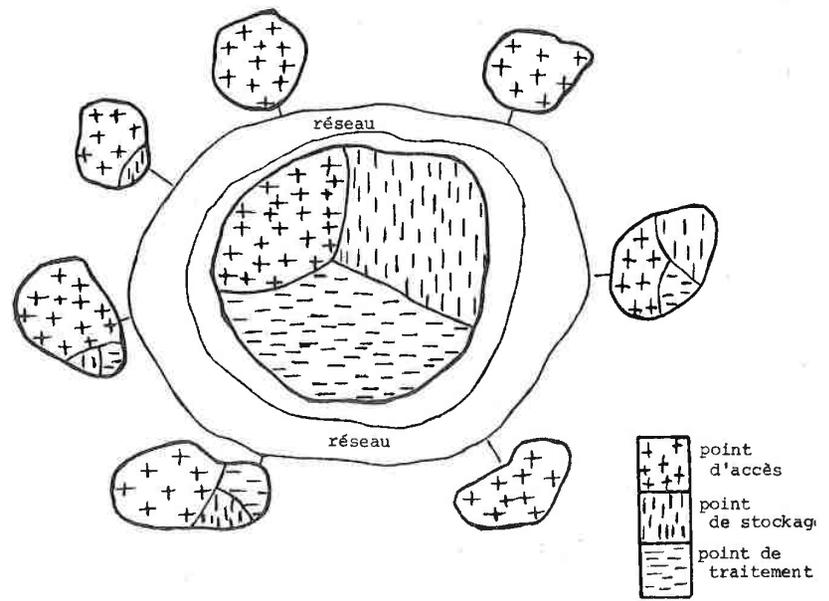


Figure III.2. : Application répartie à stockage et traitements centralisés

L'application de gestion comptable et financière décrite dans le chapitre II constitue un exemple d'application répartie pour laquelle les fonctions de stockage et de traitement sont fortement centralisées ; il existe un déséquilibre important entre le site "central" d'une part et les sites "régionaux" d'autre part.

Maintes applications réparties actuelles, qu'elles soient de dimension régionale ou nationale, sont centralisées par le fait même du regroupement de toutes les données au sein d'une base de données unique gérée en un point privilégié. Dans le paragraphe 5 du chapitre II sont présentés les principaux inconvénients d'une situation centralisée caractérisée par l'unicité du point de stockage et la centralisation des accès.

Une application répartie travaille sur des "objets-réseau" c'est-à-dire des objets qui ont un sens pour tous les sites. Dans l'application du chapitre II "fournisseur", "marché", "commande", "facture" sont des objets-réseau.

Si les données relatives aux objets-réseau sont stockées dans une base de données alors chaque objet est représenté logiquement par un ensemble de caractéristiques formant une entité (dans la terminologie SOCRATE, cf. I.2.3.1.).

Des objets peuvent être imbriqués ; ainsi une entité qui représente un objet peut posséder une caractéristique qui est elle-même du type entité c'est-à-dire qu'elle contient un bloc de caractéristiques et qu'elle représente un objet imbriqué dans le premier. Une telle imbrication peut se faire en plusieurs niveaux et une entité peut posséder plusieurs caractéristiques du type entité.

1.1.2. Organisations d'applications réparties

Au sein d'une application répartie, il existe une certaine organisation (au sens de l'organisation d'une entreprise ou d'une administration) ; celle-ci est caractérisée par une certaine signification relative d'un site par rapport à un autre, par les liaisons et les échanges qui existent entre les différents sites. Le fonctionnement global de l'application répartie repose sur cette organisation.

Il est possible de distinguer différents types d'organisation et donc de classer, selon ceux-ci, les différentes applications réparties et les différents modes de fonctionnement.

. Organisation à un niveau

Les différents sites qui participent à l'application répartie ne sont pas contrôlés et coordonnés par un site "père". Aucun site n'a de responsabilité sur un autre.

Les relations qui existent entre les sites sont des relations "horizontales".



Figure III.3. : Organisation à un niveau

Tout site n'est pas forcément en relation avec tout autre ; il peut y avoir des relations privilégiées.

. Organisation à plusieurs niveaux : organisation hiérarchique

Dans ce cas, il existe une hiérarchie entre les sites qui participent à l'application répartie. Cette hiérarchie décompose l'ensemble des sites en niveaux.

Les relations entre les sites sont "verticales" et sont du type "père-fils".

Un site "père" contrôle et coordonne les actions des sites "fils" dont il a la responsabilité.

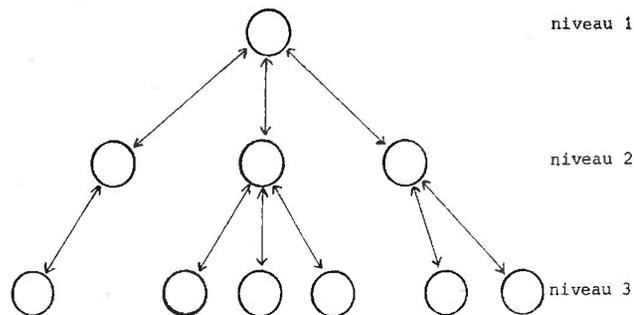


Figure III.4. : Organisation hiérarchique

Une telle organisation peut être représentée, par exemple, par une administration nationale qui possède un siège principal à Paris, des sièges régionaux, des sièges départementaux,...

. Organisation mixte

Certaines applications réparties peuvent avoir une organisation qui permette à la fois des relations du type père-fils (entre deux niveaux) et du type fils-fils ou cousin-cousin (sur un même niveau).

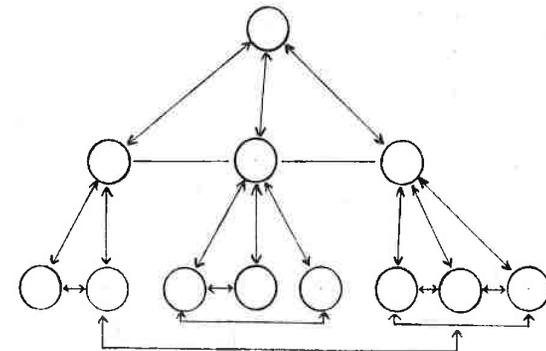


Figure III.5. : Organisation mixte

Si l'on considère, indépendamment du type d'organisation les sites d'un même niveau qui sont "frères" deux cas peuvent se produire :

- soit ces sites sont spécialisés dans leurs fonctions, c'est-à-dire que chacun joue un rôle différent vis à vis du père : ce sont des frères spécialisés.
- soit, ces sites sont équivalents dans leurs fonctions, c'est-à-dire qu'ils jouent le même rôle ; mais ce rôle est distribué à plusieurs sites géographiquement répartis ; ce sont des frères équivalents.

Pour une organisation hiérarchique, les sites "cousins" peuvent également être équivalents, c'est-à-dire que tous les sites d'un même niveau sont équivalents.

1.1.3. Organisation étoilée

L'application que nous avons décrite dans le chapitre II est une application répartie à organisation hiérarchique limitée à deux niveaux : le niveau central et le niveau régional ; il y a des liaisons entre le site central et chaque site régional mais il n'y a pas de relation entre deux sites régionaux. De plus, chaque site régional joue le même rôle vis à vis du site central : les sites régionaux remplissent les mêmes fonctions ; ils sont "équivalents".

Une telle organisation sera qualifiée d'organisation "étoilée".

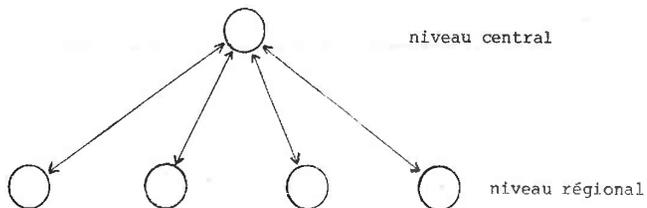


Figure III.6. : Organisation "étoilée"

C'est précisément à ce type particulier d'application répartie que nous nous intéresserons par la suite.

1.2. Base de données répartie

Définir et créer une base de données répartie consiste à proposer une architecture externe et une architecture interne.

1.2.1. Architecture externe

La base répartie est obtenue par la répartition géographique d'ensembles "système de gestion + base". Elle se compose donc de bases locales implantées sur des sites différents et chacune gérée par un système propre.

L'architecture externe de la base répartie est caractérisée par :

- l'implantation des bases locales,
- les liaisons et les échanges qui existent entre ces différentes bases locales.

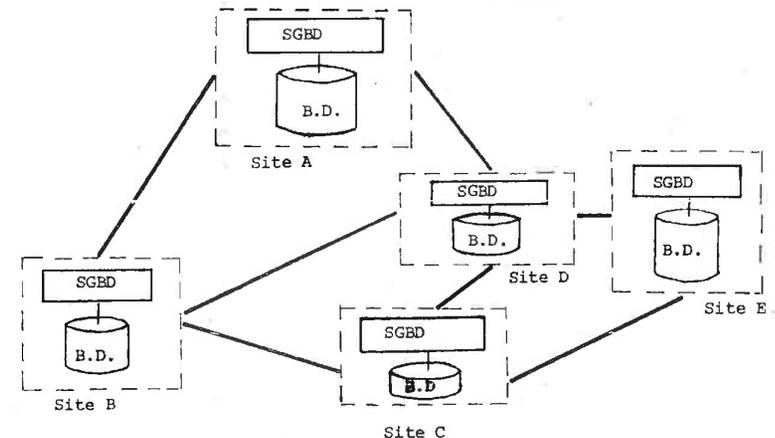


Figure III.7. : Architecture externe de la base répartie

1.2.2. Architecture interne

L'architecture interne de la base répartie concerne les contenus des différentes bases locales ; en effet, la base répartie doit contenir les données relatives aux différents "objets-réseau" (cf.1.1.1.1.) utilisés par l'application répartie, ces données étant réparties dans les différentes bases locales. Or tout système de gestion travaille sur deux niveaux : d'une part sur des entités et des caractéristiques qui sont décrites dans la description de la base, et d'autre part, sur les données contenues dans la base locale ; ceci nous amène à considérer deux niveaux de répartition dans l'architecture interne.

1.2.2.1. Répartition des entités et des caractéristiques

Cette architecture se situe au niveau des descriptions logiques des différentes bases locales et concerne donc la répartition des entités et des caractéristiques qui représentent et décrivent les "objets-réseau". Cette répartition, effectuée pour tous les objets-réseau, permet de définir la description logique de chacune des bases locales et les correspondances entre les différentes descriptions. Ainsi deux bases pourront par exemple posséder la même description logique, ou, deux bases pourront avoir quelques caractéristiques communes dans leurs descriptions respectives.

Chaque objet-réseau est représenté par un ensemble de caractéristiques et celles-ci sont réparties parmi les différentes bases locales dans lesquelles elles sont regroupées pour former une entité locale ; ainsi un même identificateur d'entité peut apparaître dans plusieurs descriptions locales : les différentes caractéristiques sont alors relatives au même objet-réseau. Cet ensemble réparti de caractéristiques constitue une entité virtuelle que nous appelons entité-réseau.

objet-réseau "F"
représenté par une entité-réseau "F"

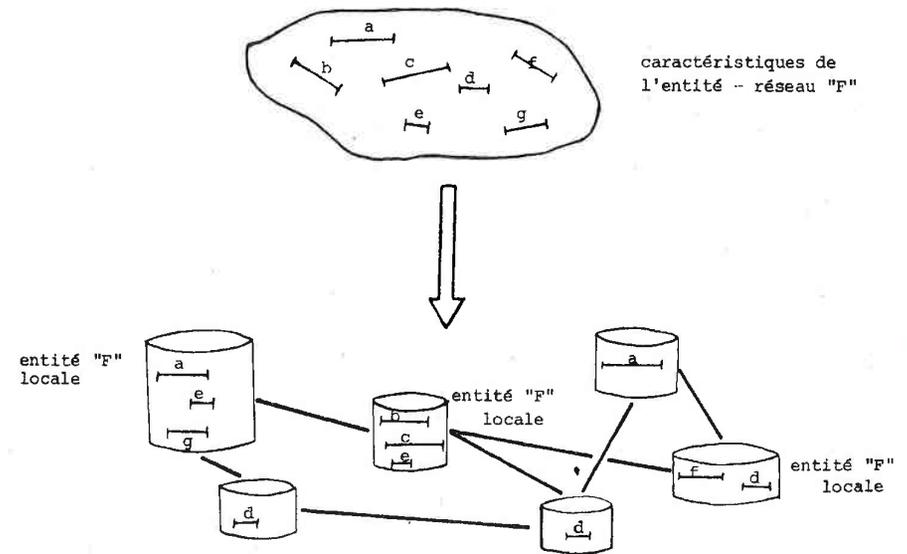


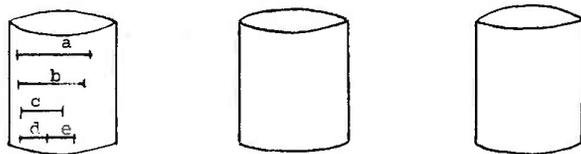
Figure III.8. : Répartition d'une entité-réseau

Si on considère une entité-réseau composée des caractéristiques a, b, c, d, e, plusieurs modes de répartition sont possibles :

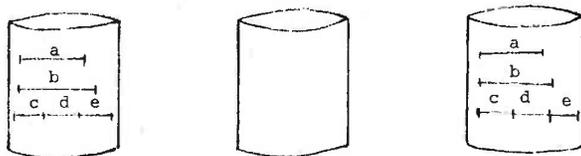
- 1) L'entité-réseau n'est pas décomposée en plusieurs entités différentes
 - 1.1) L'entité-réseau existe et n'est décrite que dans une seule base locale ; cela signifie que toutes les ca-

caractéristiques qui représentent l'objet-réseau sont dans une même base.

Si ce même type de répartition est adopté pour toutes les entités-réseau alors les bases locales auront des descriptions logiques indépendantes et chaque objet-réseau ne sera représenté que dans une seule base.

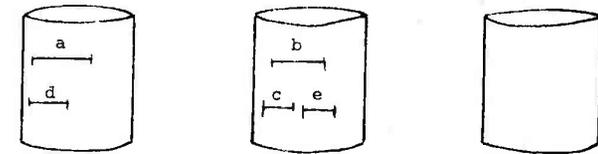


1.2) L'entité-réseau existe et est décrite de façon identique dans plusieurs bases locales ; donc l'objet-réseau est entièrement représenté dans plusieurs bases locales.

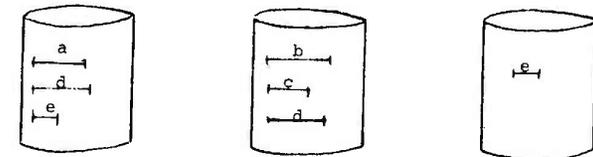


2) L'entité-réseau est décomposée en plusieurs entités différentes ayant un même nom

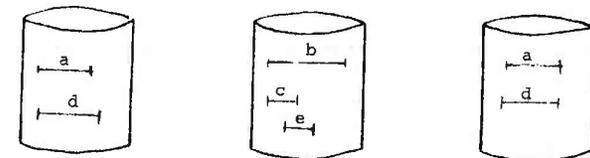
2.1) Les entités n'ont pas de caractéristiques communes (elles sont disjointes) et n'existent et ne sont décrites que dans une seule base locale



2.2) Deux entités peuvent avoir des caractéristiques communes mais n'exister que dans une seule base locale (il y a redondance entre des caractéristiques).



2.3) Deux entités existent et sont décrites dans plusieurs bases locales (il y a deux redondances d'entité et forcément redondance de caractéristique).



Si le même type de répartition est utilisé pour toutes les entités-réseau alors deux bases locales peuvent avoir des descriptions logiques :

- indépendantes (1.1 et 2.1)
- identiques (1.2 et 2.3)
- ayant des redondances entre caractéristique (2.2)

Remarque : Il n'y a pas de relations logiques, référence ou anneau, entre deux bases locales ; de telles relations ne sont possibles qu'entre caractéristiques d'entités d'une même base locale. Cependant, les caractéristiques des types référence ou anneau peuvent suivre les mêmes modes de répartition que les autres caractéristiques : ainsi une référence ou un anneau peuvent, par exemple, être définis et exister dans plusieurs bases locales.

1.2.2.2. Répartition des données

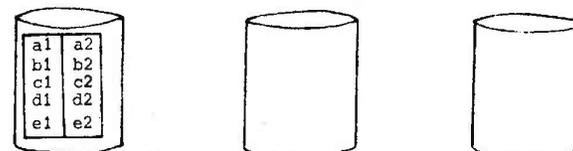
Ce niveau d'architecture concerne les données contenues dans les différentes bases locales.

A tout objet-réseau correspond une description logique d'entité-réseau constituée de caractéristiques. Mais à un objet-réseau correspond aussi une multitude d'ensembles de données qui constituent des réalisations d'objet-réseau. Ainsi si on considère l'objet-réseau "fournisseur", chaque fournisseur physique est représenté par un ensemble (défini par l'entité-réseau "fournisseur"), structuré de données qui constitue une réalisation de l'objet-réseau. Le problème, dans ce paragraphe, est de montrer comment les données qui composent une même réalisation d'objet-réseau doivent être réparties dans les différentes bases locales : dans certains cas la répartition des données suit la répartition des caractéristiques évoquée dans le paragraphe précédent ; mais dans d'autres cas, lorsqu'il y a des redondances logiques entre les bases locales, la répartition des données peut faire l'objet d'un choix.

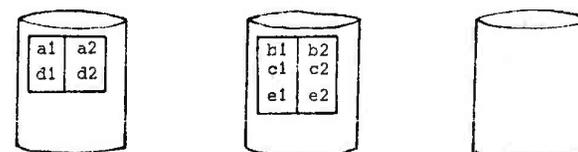
Divers cas de répartition, dépendant essentiellement de la répartition du niveau précédent, sont à considérer :

a) Si l'entité-réseau est décrite entièrement dans une base locale unique (cas 1.1.) (entité-réseau = 1 entité locale), alors les valeurs d'une même réalisation-réseau sont stockées dans la même base et il n'y a pas de choix de répartition des données.

De même, si l'entité-réseau est décomposée de manière disjointe (cas 2.1., pas de redondance de caractéristiques) en plusieurs entités locales, chacune n'existant que dans une base, alors la répartition des données suit la décomposition de l'entité-réseau ; si celle-ci est décomposée en 2 entités locales, alors la réalisation-réseau est décomposée en deux réalisations d'entité locales disjointes.



Cas 1.1.



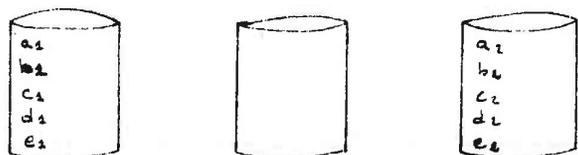
Cas 2.1.

Dans ces deux cas il n'y a ni redondance logique ni redondance physique.

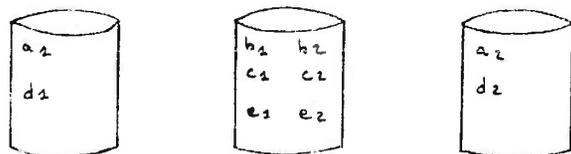
b) Si l'entité-réseau est entièrement (cas 1.2.) ou en partie (cas 2.3.) dupliquée dans plusieurs bases locales alors plusieurs cas de répartition de données sont possibles :

b.1. La répartition des données est unique : c'est-à-dire que si deux bases locales possèdent une même entité, alors elles ne pourront pas posséder chacune une réalisation locale relative à la même réalisation d'objet-réseau ; les deux bases locales posséderont des réalisations locales correspondant à des réalisations d'objet-réseau différentes.

Si par exemple, plusieurs bases possèdent la même description de l'entité fournisseur, alors les données d'un même fournisseur (d'une même réalisation-réseau) n'existent que dans une seule base. Dans ce cas, les bases redondantes au niveau des caractéristiques sont disjointes au niveau des données.



Cas 1.2.

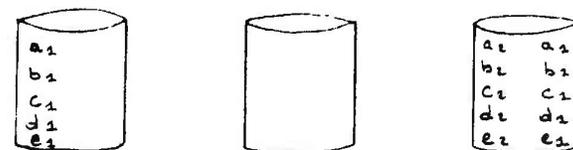


Cas 2.3.

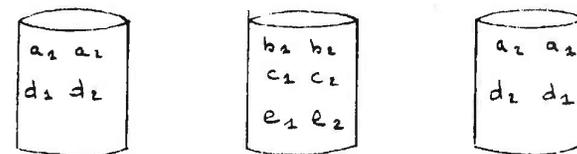
b.2. Deux bases locales possédant une même description d'entité peuvent chacune posséder une réalisation locale relative à la même réalisation d'objet-réseau.

Ainsi, deux bases peuvent posséder chacune une réalisation locale relative au même fournisseur (à la même réalisation d'objet-réseau).

Dans ce cas des redondances de données entre les bases locales sont possibles.

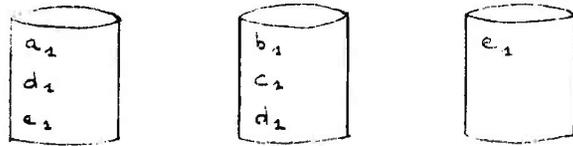


Cas 1.2.



Cas 2.3.

c) Si l'entité-réseau est décomposée de manière non disjointe, c'est-à-dire qu'une caractéristique peut apparaître dans plusieurs bases locales, alors en général il y aura redondance de données au niveau de certaines caractéristiques pour une même réalisation d'objet-réseau.

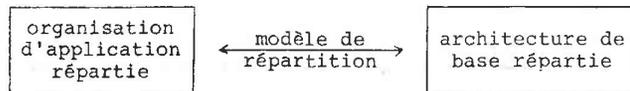


Cas 2.2.

1.3. Modèle de répartition

Nous avons présenté dans les deux paragraphes précédents d'une part des types d'organisation d'application répartie et d'autre part des modes de répartition pouvant être utilisés dans une architecture de base répartie. Il s'agit désormais d'associer type d'organisation et mode de répartition : c'est le but du modèle de répartition.

Le modèle de répartition exprime des choix à faire pour aboutir à une certaine architecture de base répartie qui corresponde à une certaine organisation d'application répartie tout en convenant aux besoins des utilisateurs.



Dans les paragraphes qui suivent, nous allons proposer un modèle particulier, le modèle de répartition étoilée, qui permet de définir et d'exprimer l'architecture d'une base répartie étoilée ; les principes de répartition de ce modèle correspondent aux besoins d'une organisation étoilée telle qu'elle a été définie au paragraphe 1.1.3. Ces principes sont conçus à partir de l'exemple d'organisation étoilée présenté dans le chapitre II.

2. PRINCIPES DE REPARTITION ETOILEE

2.1. Autonomie des sites

Un premier principe de répartition consiste à rendre les sites autonomes en ce sens que chacun d'eux doit posséder toutes et seulement toutes les données dont il peut avoir besoin que ce soit pour des interrogations ou des mises à jour conversationnelles, ou, pour exécuter des traitements locaux.

2.1.1. Indépendance des sites régionaux par rapport au site central

En multipliant les points de stockage et en donnant à chaque site "ses données", ce principe permet d'éviter que les interrogations régionales soient transportées par le réseau et exécutées sur le site central, et il permet d'éviter les transferts périodiques d'ensembles de données du site central vers les sites régionaux pour permettre à ces derniers d'effectuer des traitements locaux (cf. II.4.5.).

Il en résulte un double avantage caractérisé par

- une diminution du nombre des transmissions sur le réseau
- une amélioration des temps de réponse aux interrogations conversationnelles.

et une plus grande indépendance des sites régionaux par rapport au site central pour ce qui concerne la propriété des données.

2.1.2. Indépendance des sites régionaux entre eux

L'organisation étoilée, par définition, confère une indépendance des sites régionaux entre eux. Le fait de donner à chacun de ces sites ses propres données ne va donc pas à l'en-

contre de cette définition. Cependant, les sites régionaux étant "équivalents" (cf. 1.1.3.), ils peuvent avoir des besoins communs ; plusieurs sites régionaux peuvent, par exemple, travailler avec un même fournisseur et il se pose alors le problème du stockage des données qui lui sont relatives. Le présent principe impose que dans un tel cas les données soient stockées sur chacun des sites utilisateurs : ainsi un site régional "possède" tous les fournisseurs avec lesquels il a passé un marché indépendamment du fait que le même fournisseur ait passé des marchés avec d'autres sites. Au stockage unique, nous préférons donc le stockage multiple, c'est-à-dire la duplication.

Considérons un exemple : supposons que le site "Marseille" ait passé un marché le 1.1.76 avec le fournisseur "Martin" ; celui-ci est donc répertorié, depuis cette date, sur le site "Marseille" ; le 15.2.76, le site "Grenoble" passe un marché avec le même fournisseur : le principe d'autonomie impose alors que le fournisseur soit également répertorié sur "Grenoble" ; à compter du 15.2.76, le fournisseur est donc répertorié sur deux sites.

Cet exemple montre que l'on pourrait se contenter d'un stockage unique du fournisseur sur le premier site utilisateur ; le second site dépendrait alors du premier et cela poserait les inconvénients suivants :

- Comment le site "Grenoble" peut-il avoir la connaissance de l'existence de "Martin" à "Marseille" ? C'est un problème de connaissance et de localisation des données.
- Les requêtes d'interrogation devraient passer via le site central puisqu'il n'y a pas de liaisons directes entre les sites régionaux.
- Que se passe-t-il si le site propriétaire n'est pas actif ?

En étendant cette possibilité de stockage unique à l'ensemble des objets, on s'apercevrait alors que le mécanisme deviendrait rapidement plus lourd que le mécanisme centralisé car il lui ajouterait des problèmes de recherche et de localisation des données.

2.2. Répartition des logiciels d'accès aux données ; accès décentralisés

Dans l'application que nous considérons, l'accès aux données de la base ne se fait pas directement par un langage de requêtes adressées au système de gestion. Les accès se font par appel de programmes d'accès aux données qui sont catalogués au niveau du système de gestion (cf. II.4.3.). Ces programmes réalisent d'une part les interrogations de données et d'autre part les mises à jour (créations de réalisations, modifications, suppressions).

Dans une approche centralisée tous les logiciels d'accès sont implantés sur le site central et les différents terminaux répartis sur les points d'accès sont donc logiquement "rattachés" à ces logiciels centraux.

Dans notre approche nous imposons que tout site possède ses propres logiciels d'accès et les terminaux ne sont plus "rattachés" au site central mais "rattachés" à des logiciels régionaux.

a) Logiciels d'interrogation

Un site possède toutes les données qu'il peut interroger ainsi que ses logiciels d'interrogation ; l'interrogation est donc une opération purement locale.

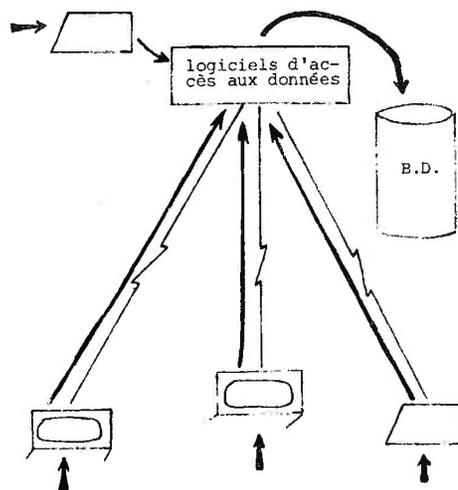


Figure III.9. : Logiciels d'accès centralisés

b) Logiciels de mise à jour

Les données étant réparties dans plusieurs bases locales une opération de mise à jour, déclenchée en un site, peut donc intéresser d'autres sites ; ainsi dans le cas d'une création, les données saisies vont être réparties et la création sera donc transmise à d'autres sites.

Sur un site, les logiciels d'accès en mise à jour reçoivent des opérations saisies localement mais également des opérations distantes ; mais l'accès aux données d'une base locale se fait toujours depuis un logiciel local et non à distance comme dans l'approche centralisée.

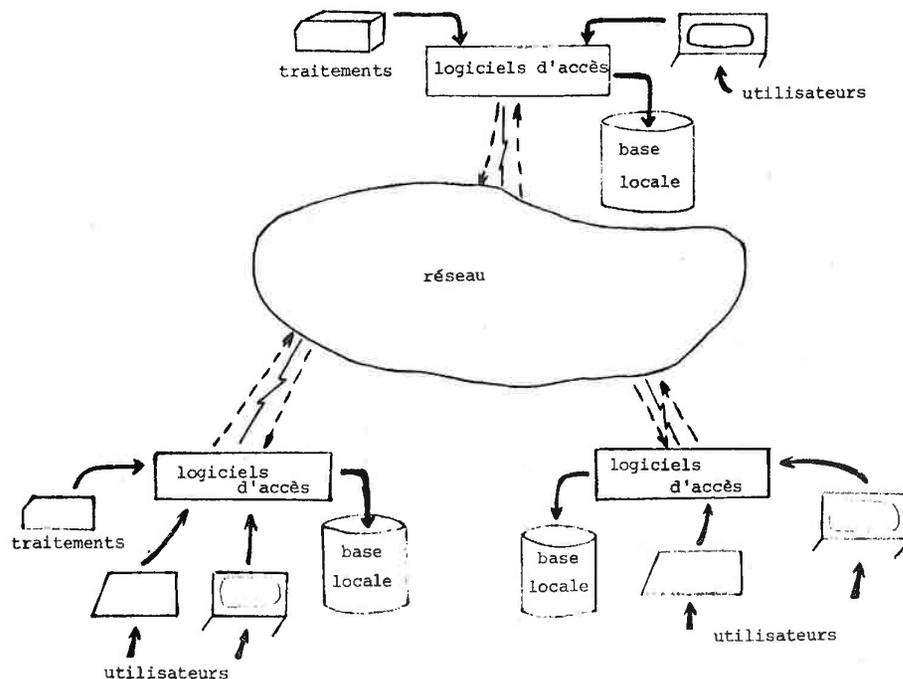


Figure III.10. : Logiciels d'accès répartis

2.3. Répartition des données

Les données reçues par un logiciel d'accès lors d'une saisie doivent être stockées sur les sites intéressés : il s'agit de répartir les données.

2.3.1. Notion d'axe de répartition

L'organisation de l'application est telle que tout site régional n'établit des échanges qu'avec le site central ; nous dirons que les échanges se font, d'un point de vue fonc-

tionnel, sur des axes central-régional, tous convergeant vers le site central. La répartition des données se fait sur ces axes.

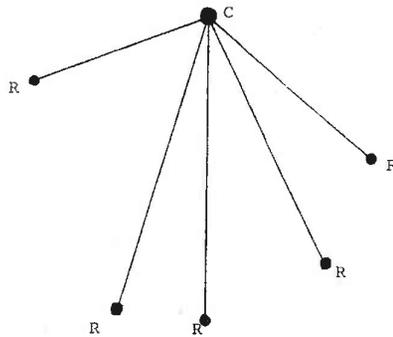


Figure III.11. : Axes central-régional

2.3.2. Répartition des données sur un axe

Si l'on considère une donnée saisie en une extrémité d'un axe, deux cas de répartition peuvent se produire :

- soit, elle n'intéresse pas le site sur lequel elle a été saisie, elle y est alors stockée (a).
- soit, elle intéresse les deux extrémités, elle est stockée localement et est transmise au site symétrique (b).

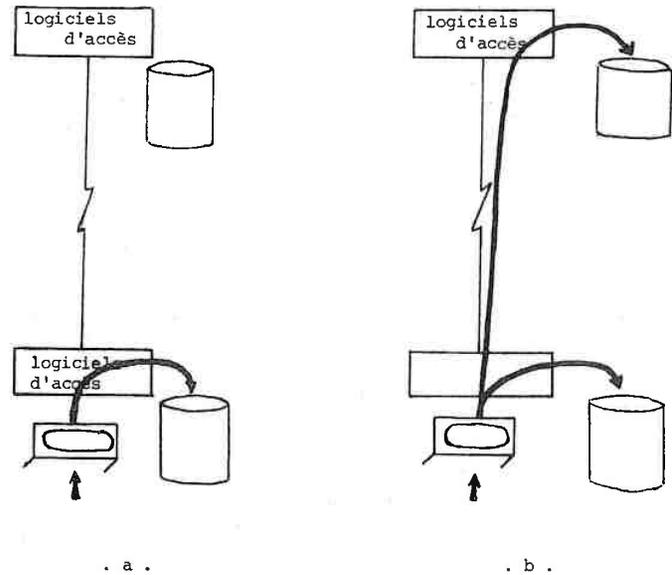


Figure III.12. : Cas de répartition sur un axe

Sur cette figure nous avons représenté les deux cas de répartition possibles d'une donnée sur un axe dans le cas d'une saisie régionale ; les deux mêmes cas sont possibles pour une saisie effectuée sur le site central.

Ces deux cas de répartition sont affectés à des données ponctuelles; ainsi lors d'une saisie d'un ensemble de données celles-ci sont réparties ponctuellement c'est-à-dire que les unes seront réparties selon le cas a, tandis que les autres le seront selon le cas b. La répartition se fait donc au niveau de la donnée ponctuelle.

2.3.3. Répartition sur plusieurs axes

Dans certains cas, le site central peut déclencher une répartition dans le sens central → régional, sur plusieurs axes. La répartition se fait alors ponctuellement et en parallèle sur les différents axes concernés, sur chacun d'eux selon les deux cas précédents.

2.4. Base centrale et bases régionales

Pour respecter le premier principe d'autonomie chaque site disposera d'une base de données gérée par son propre système et chaque site possédera ses propres logiciels d'accès à ses données.

Pour respecter l'organisation générale étoilée, les bases seront implantées aux extrémités de chaque axe : il y aura donc une base centrale et des bases régionales, l'ensemble de ces bases locales formant la base répartie étoilée.

Les sites régionaux étant équivalents (cf. 1.1.3.), les bases régionales seront également équivalentes ; ceci se traduit par le fait que les bases régionales possèdent toutes la même description logique. Il existe donc, d'une part, une description centrale pour la base centrale, et, d'autre part, une description régionale, unique, dans toutes les bases régionales.

2.5. Répartition statique

L'implantation géographique ainsi que les descriptions logiques des différentes bases locales sont définies de manière fixe selon un accord entre les différents sites. Pour chaque objet-réseau, les lieux d'implantation des données sont choisis

de manière définitive lors de la définition des descriptions logiques des différentes bases. Ainsi au cours de l'utilisation de la base répartie, les "lieux" de stockage des données ne changent pas ; la répartition est donc statique : les données sont réparties, au fur et à mesure de leur saisie, en des "lieux" prédéfinis et fixes. Ce principe s'oppose à une répartition dynamique où les contenus des différentes bases locales pourraient changer de site, au cours du temps, selon des critères prédéfinis ; les descriptions des différentes bases locales ne seraient alors plus fixes.

3. EXPRESSION ET UTILISATION DU MODELE ETOILE

L'ensemble des principes de répartition évoqués dans le paragraphe précédent constitue le modèle de répartition étoilé. Dans cette partie nous allons voir comment ce modèle s'exprime sur les données structurées qui doivent être contenues dans la base répartie et comment il est utilisé.

3.1. Expression du modèle

3.1.1. Expression dans la répartition des entités et des caractéristiques

Une entité-réseau, ou ensemble de caractéristiques représentant un objet-réseau, est décomposée en deux entités : une entité centrale décrite dans la base centrale, et une entité régionale décrite dans chaque base régionale.

Cette décomposition en deux entités s'explique par le fait que tout objet est utilisé à la fois au niveau régional et au niveau central ; les besoins des deux niveaux ne sont pas exactement identiques, d'où deux entités différentes ; cependant certaines caractéristiques sont nécessaires aux deux niveaux, de ce fait les deux entités ne sont pas disjointes et certaines caractéristiques existent dans les deux entités.

Du fait que les descriptions régionales sont toutes identiques, il en résulte que la répartition des caractéristiques d'une entité-réseau, est la même sur tous les axes ; il faut distinguer :

- les caractéristiques centrales
- les caractéristiques régionales
- les caractéristiques centrales et régionales

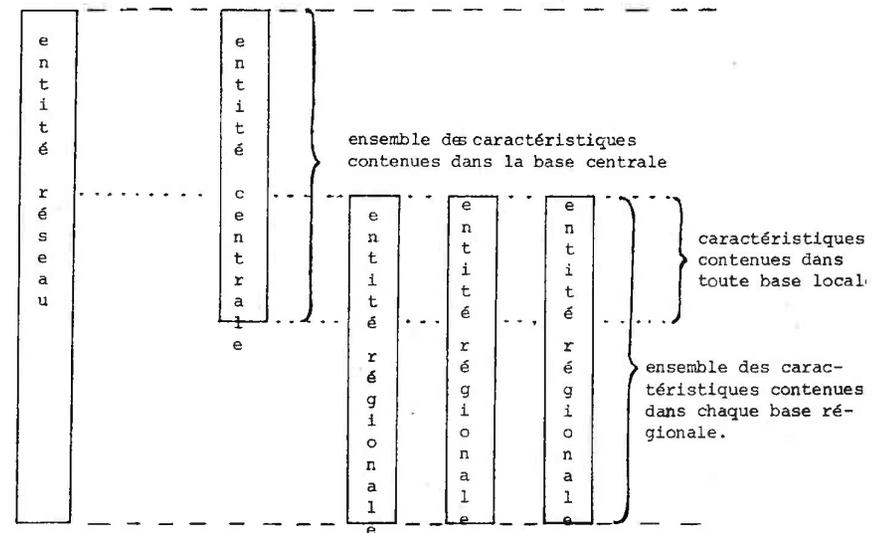


Figure III.13. : Répartition d'une entité-réseau

La répartition proposée correspond aux types 2.2. et 2.3. du paragraphe 1.2.2.1. :

- 2.2. : l'entité-réseau est décomposée de manière non disjointe en une entité centrale et une entité régionale (redondance de caractéristiques).
- 2.3. : l'entité régionale existe dans chaque base régionale (redondance d'entités locales).

3.1.2. Expression dans la répartition des données

En fonction de la répartition des entités des caractéristiques certains choix sont possibles au niveau de la répartition des données d'une réalisation d'objet-réseau.

Pour une même réalisation d'objet-réseau, par exemple pour un marché ou un fournisseur, il existera au moins deux réalisations locales qui correspondent à la répartition sur un axe :

- une réalisation centrale, et,
- une réalisation régionale.

Dans les cas où une réalisation d'objet-réseau est utilisée sur plusieurs axes, le principe d'autonomie impose que les données de cette réalisation figurent sur tous les axes intéressés ; il y aura alors pour une même réalisation d'objet-réseau :

- une réalisation centrale (commune à tous les axes intéressés)
- plusieurs réalisations régionales, étant entendu que chacune se trouve dans une base locale différente.

Exemple : Si l'on considère l'exemple du paragraphe 2.1.2. :

- entre le 1.1.76 et le 15.2.76 : il existe deux réalisations locales pour représenter le fournisseur "Martin" ; l'une dans la base centrale, l'autre dans la base régionale de Marseille.
- à compter du 15.2.76 : il existe trois réalisations ; l'une dans la base centrale, l'autre dans la base de Marseille, la troisième dans la base de Grenoble.

Remarque : si une même réalisation d'objet-réseau est utilisée sur plusieurs axes, le site central ne possède qu'une seule et unique réalisation relative à cette réalisation d'objet.

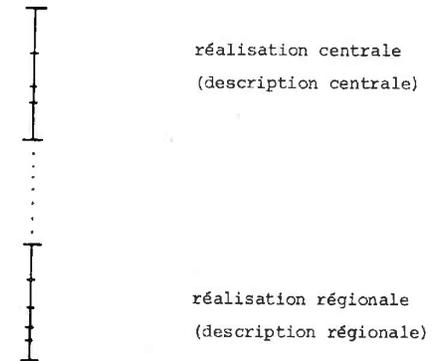


Figure III.14. : Répartition d'une réalisation-réseau sur un seul axe.

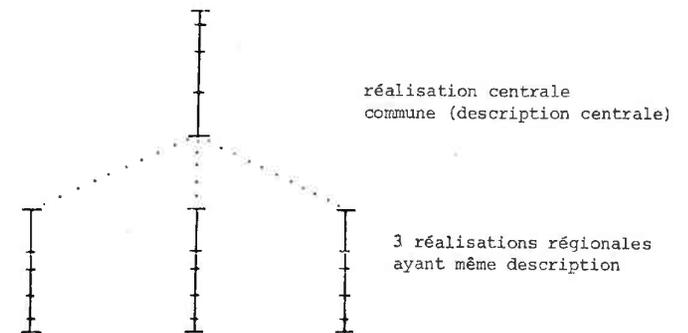


Figure III.15. : Répartition d'une réalisation-réseau sur trois axes

Cette possibilité de stocker des données d'une même réalisation d'objet dans plusieurs bases locales ayant une même description (bases régionales) correspond au cas b.2. de répartition des données et provoque des redondances entre ces différentes bases.

Le fait de répartir les données sur un axe correspond au cas de répartition c ; une entité centrale et une entité régionale n'étant pas disjointes, des redondances, au niveau de certaines caractéristiques, vont apparaître entre la base centrale et une base locale.

3.1.3. Types de données réparties

En tenant compte d'une part de la répartition des entités et caractéristiques et d'autre part de la répartition des données, le modèle introduit cinq types différents de données réparties ; ainsi une donnée peut être stockée

- dans la base centrale uniquement (unicité)
- dans, au plus, une seule base régionale (unicité)
- dans une ou plusieurs bases régionales (redondance horizontale)
- dans la base centrale et, au plus, une seule base régionale (redondance verticale)
- dans la base centrale et dans une ou plusieurs bases régionales (redondance triangulaire)

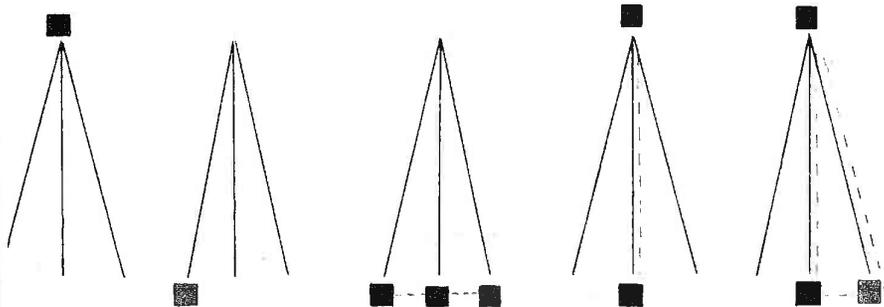


Figure III.16. : Types de données réparties

3.2. Utilisation du modèle

3.2.1. Attribution de types de répartition aux caractéristiques d'entités

Dans le paragraphe précédent ont été présentés les cinq types de données réparties définies par le modèle étoilé.

Cependant, lors de la manipulation de la base répartie les logiciels accèdent, non pas directement aux données, mais à leurs représentants logiques c'est-à-dire à des caractéristiques d'entités. Il est donc important que le modèle puisse être utilisé, non pas lors des accès aux données d'une base, mais lors des accès logiques.

Si on considère une caractéristique d'une entité-réseau elle représente logiquement tout un ensemble de données, une donnée par réalisation d'entité-réseau.

Les principes de répartition étant les mêmes pour toute réalisation d'entité-réseau, on peut donc en déduire que toutes les données représentées logiquement par une même caractéristique sont réparties de la même façon. Le type de répartition de ces données peut donc être attribué à la caractéristique qui les représente. Ainsi, à chaque caractéristique d'entité-réseau est associé un type de répartition.

Par exemple, si la caractéristique NOM de FOURNISSEUR est du type "base centrale uniquement" cela signifie que les noms de tous les fournisseurs répertoriés dans la base répartie sont stockés uniquement et obligatoirement dans la base centrale.

Le type attribué à une caractéristique exprime d'une part, la façon dont les données représentées par la caractéristique sont réparties, et, d'autre part, la façon dont doivent

être exécutées les créations (c), modifications (m), suppressions (s) ; en effet, ces actions sont organisées différemment selon le type attribué à la caractéristique.

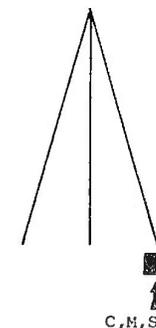
3.2.2. Définition des cinq types affectés aux caractéristiques

DCP : Type affecté à une caractéristique centrale représentant une "Données Centrale Pure". Pour une réalisation-réseau toute donnée représentée par une caractéristique de ce type n'existe et ne peut être utilisée que sur le site central ; les actions sont donc purement locales au site central (accès depuis le site central uniquement).

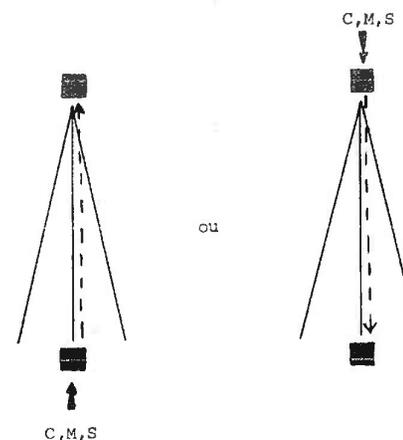


DRP : Type affecté à une caractéristique régionale représentant une "Donnée Régionale Pure" ; pour une réalisation-réseau qui est répartie sur plusieurs axes (cf. 3.3.) une caractéristique DRP a des contenus diffé-

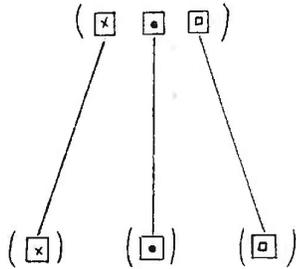
rents selon les axes ; les actions sont donc locales et une DRP ne peut être utilisée que sur le site régional où elle a été saisie.



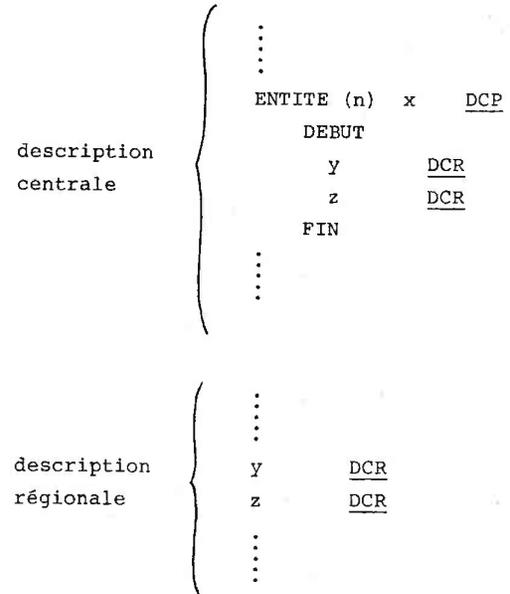
DCR : Type affecté à une caractéristique centrale et régionale représentant une "Donnée Centrale et Régionale" ; une DCR peut être utilisée sur le site central et sur un site régional ; elle peut être créée, modifiée, ou supprimée, par le site central ou par le site régional de l'axe qui l'utilise mais chaque action doit être répartie sur l'axe.



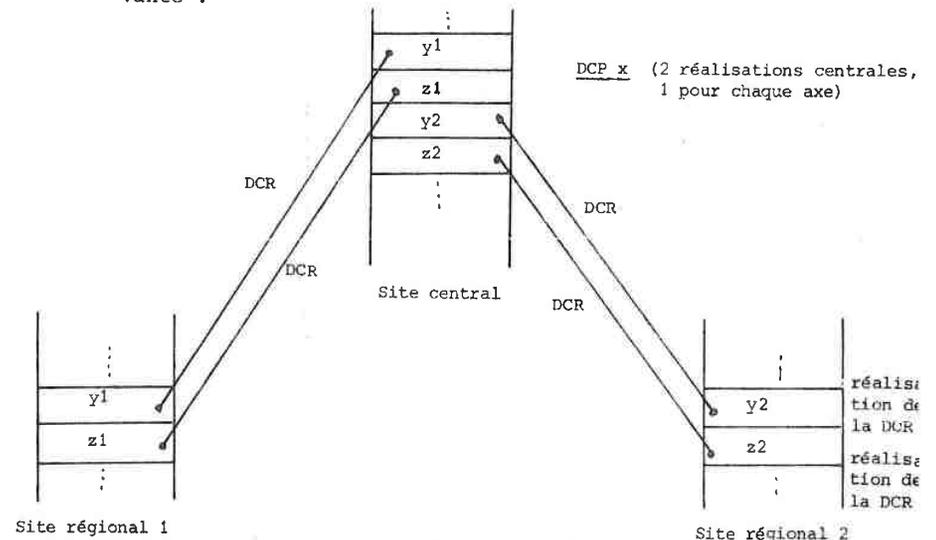
Pour une réalisation-réseau répartie sur plusieurs axes, les contenus d'une même DCR sont différents selon l'axe ; la réalisation centrale, commune à tous les axes (cf. 3.3.), doit contenir, sous la même DCR, les valeurs relatives à tous les axes.



Sur le site central, une DCR doit donc permettre le stockage maximum de n valeurs si n est le nombre d'axes ; les DCR du site central doivent être réunies dans une caractéristique du type ENTITE, selon la terminologie SOCRATE et le nombre de réalisations de cette caractéristique est fixé à n . Cette caractéristique du type ENTITE est une DCP car elle n'est utilisée que sur le site central pour contenir les DCR qui nécessitent plusieurs réalisations centrales.

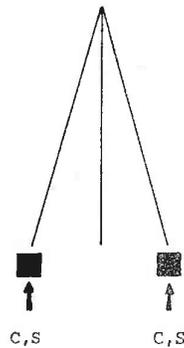


Supposons une réalisation-réseau répartie sur 2 axes l'entité comportant deux caractéristiques DCR, x et y ; la répartition des données centrales et régionales serait la suivante :



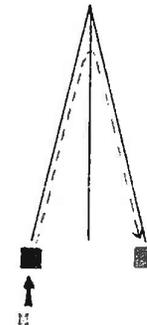
DRR : Type affecté à une caractéristique régionale pour représenter une "Donnée Régionale Redondante". Une DRR est utilisée sur chaque site où elle existe. La création et la suppression sont des opérations locales, propres à chaque axe ; pour une même réalisation-réseau, ces opérations se font de manière indépendante sur les différents axes.

A un instant donné, pour une même réalisation-réseau, il se peut que le contenu d'une DRR n'existe que sur un seul site régional : cela signifie que la réalisation-réseau n'existe que sur un seul axe ; mais le fait que la caractéristique soit DRR signifie que pour la même réalisation-réseau une réalisation peut être créée, ultérieurement, sur un ou plusieurs autres sites régionaux et dans ce cas il y aura redondance entre l'ensemble des sites régionaux répertoriant des données relatives à une même réalisation d'objet-réseau.



Toute modification d'une DRR doit être exécutée sur tout site possédant un exemplaire de la donnée dupliqué horizontalement ; c'est donc une opération à répartir. Cependant, vu l'indépendance des sites régio-

gionaux lors des créations et suppressions, un site régional n'a pas la connaissance de l'existence d'une copie éventuelle d'une de ses données DRR sur un autre site régional. Ainsi, un site régional possédant des données d'un certain fournisseur ignore si celui-ci est répertorié dans d'autres bases régionales ; lorsqu'il reçoit un accès en modification pour une DRR relative à ce fournisseur, il ne sait pas à quel(s) site(s) transmettre l'opération. Le principe admis est que toute modification pour une DRR, reçue sur un site régional, est transmise au site central qui est en mesure de savoir sur quel(s) axe(s) existe une même réalisation d'objet-réseau et donc de savoir si une DRR est réellement dupliquée ; le site central transmet alors la modification aux sites intéressés. Pour toute réalisation-réseau, la réalisation centrale possède des informations de localisation ; ainsi au niveau du bloc "DCR", il est possible de savoir sur combien d'axes et sur lesquels une même réalisation-réseau est répertoriée (cf. "DCR").

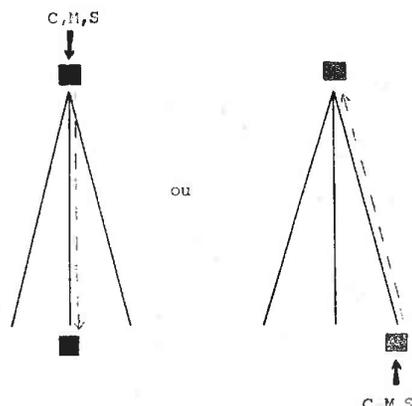


DRT : Type affecté à une caractéristique régionale et centrale représentant une "Donnée à Redondance Triangulaire". La création et la suppression sur une DRT sont des opérations relatives à un seul axe alors que la modification concerne tous les sites sur lesquels une DRT, relative à une réalisation-réseau, existe. A un instant donné, pour une réalisation-réseau, une DRT peut se limiter à une DCR : cela signifie que la réalisation-réseau n'est répertoriée que sur un seul axe : il y a alors redondance verticale ; cependant, pour la même réalisation-réseau, il peut y avoir, ultérieurement des créations sur d'autres axes et il y a alors des redondances triangulaires.

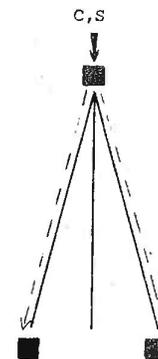
En fait, la DRT est la combinaison des deux types précédents, DCR et DRR.

Comparativement à la DCR, la réalisation centrale d'une DRT ne contient qu'une seule donnée commune à tous les axes concernés par une même réalisation-réseau, et, chaque site régional de ces axes contient un exemplaire de cette donnée.

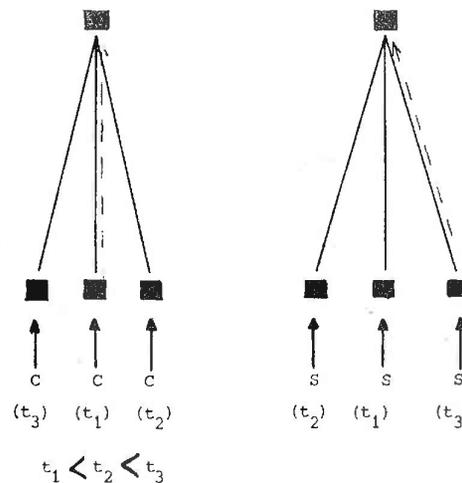
Dans le cas où la donnée représentée par une DRT n'existe que sur un seul axe, alors, création, suppression et modification peuvent être exécutées depuis le site central ou depuis le site régional et doivent être réparties.



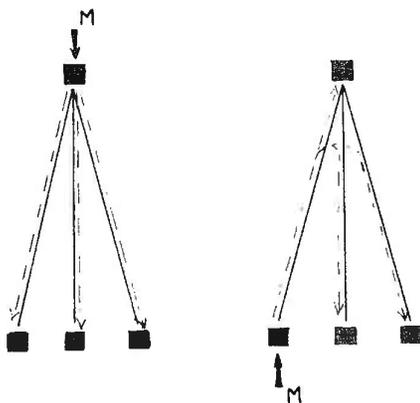
Lorsqu'une création ou suppression de DRT intéresse plusieurs axes alors l'action doit être exécutée depuis le site central.



Une création ou une suppression peut provenir d'un site régional ; la première création doit être répartie (t_1), la dernière suppression également.



Une modification doit toujours être répartie sur tous les axes qui possèdent une même réalisation-réseau ; la localisation des exemplaires d'une DRT est connue dans la base centrale au niveau des DCR contenues dans un "bloc" contenant autant de réalisations que d'axes (cf. "DRR"). Dans tous les cas une modification saisie sur un site régional est transmise au site central qui localise les exemplaires régionaux d'une DRT.



3.2.3. Attribution des types aux références, anneaux et chaînes inverses

Les types qui viennent d'être présentés ont surtout été définis pour les caractéristiques "descriptives" (cf. 1.2.3.1) c'est-à-dire les caractéristiques de type mot, valeur numérique, ou, liste de valeurs.

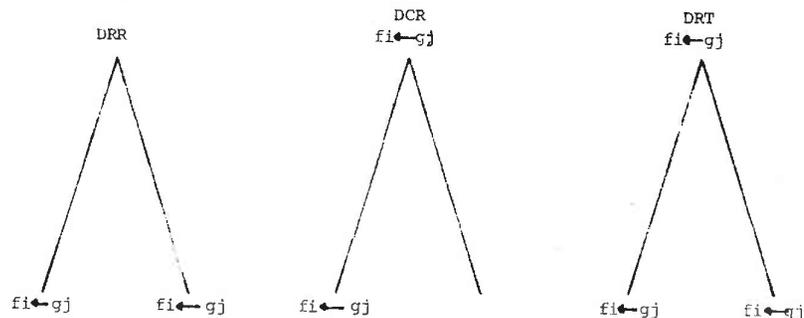
Cependant ils sont également attribués aux caractéristiques "structurelles" c'est-à-dire aux références, aux anneaux et aux chaînes de bits inverses (cf. 1.2.3.1.). Les types DCP et DRP gardent la même signification c'est-à-dire qu'une caracté-

ristique structurelle est "purement locale", soit centrale, soit régionale.

Pour les types "redondants", DRR, DCR, et DRT il semble intéressant de préciser leur signification pour chaque type de caractéristique structurelle.

. Référence

Une référence est redondante, selon l'un des trois cas, si elle existe dans plusieurs bases locales et si elle associe des "éléments" qui sont redondants, deux à deux, dans plusieurs bases. Ainsi deux réalisations d'objet-réseau f_i et g_j , répertoriées dans plusieurs bases locales, se réfèrent dans chacune des bases :

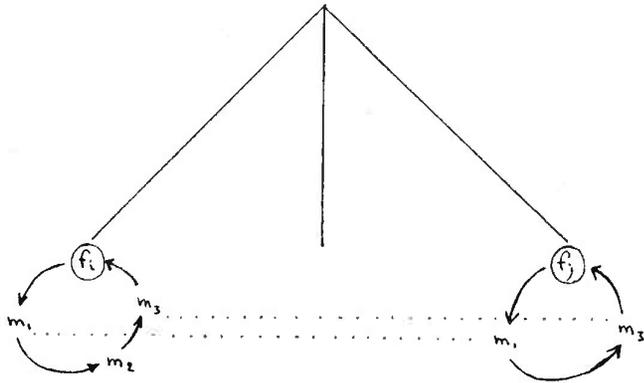


En fait, pour une référence, un type DCR, DRR, DRT indique l'existence de la référence dans plusieurs bases locales.

. Anneau

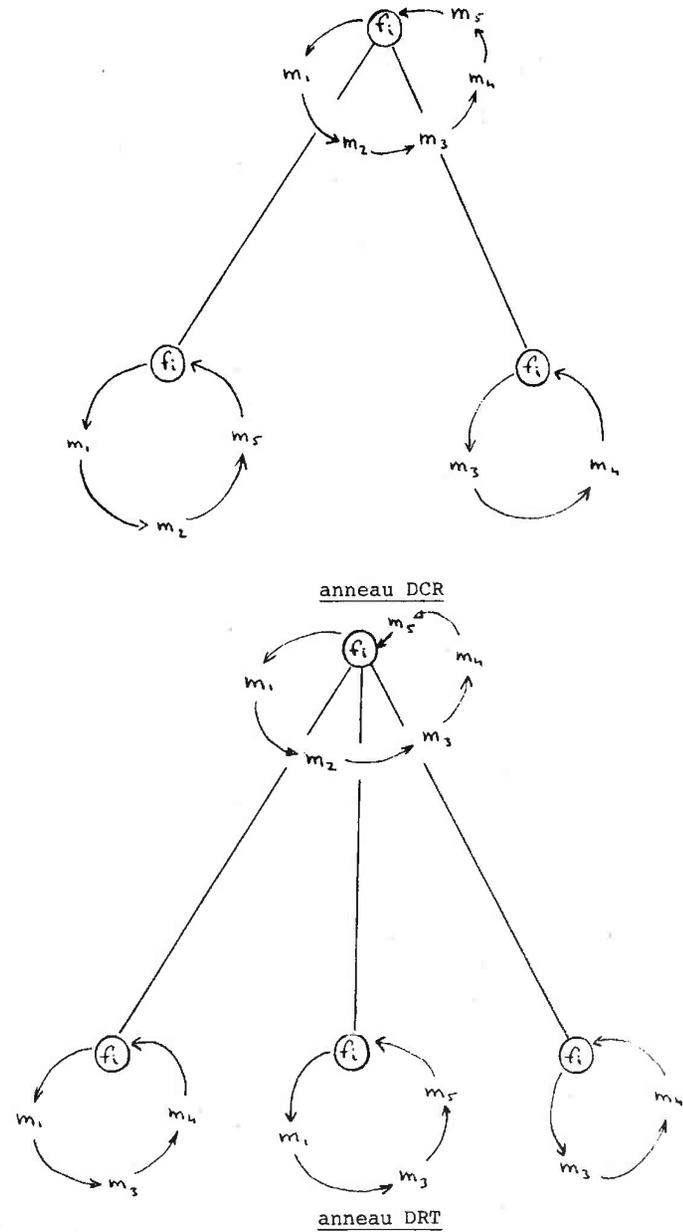
Un anneau est du type DCR, DRR, DRT si il existe dans plusieurs bases locales.

Dire qu'un anneau est du type DRR, signifie qu'il existe dans toutes les bases régionales et que pour une même réalisation f_i , redondante, l'anneau peut contenir des réalisations m_j également redondantes horizontalement.



Un anneau est du type DCR ou DRT, si il existe dans la base centrale et les bases régionales.

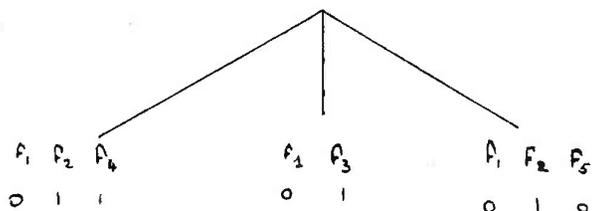
Pour une même réalisation f_i et m_j incluses dans l'anneau existent dans la base centrale et une seule base régionale (anneau DCR) ou dans la base centrale et au moins une base régionale (anneau DRT).



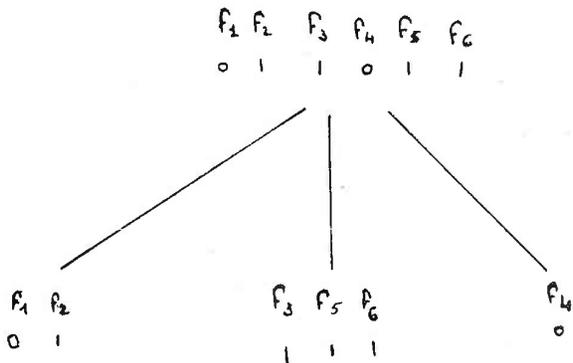
Chaîne de bits inverse

Une file inverse est du type :

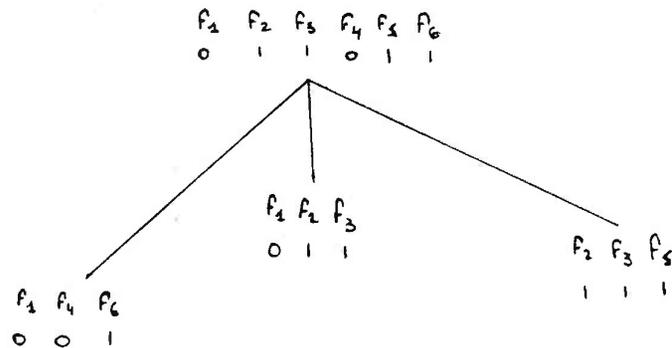
DRR : si elle existe dans les bases régionales et si une réalisation-réseau répertoriée dans plusieurs bases locales peut être "pointée" dans chacune des bases par la chaîne de bits inverses.



DCR : si elle existe dans toutes les bases locales et si une réalisation-réseau peut être "pointée" dans la base centrale et, au plus, une base régionale.



DRT : si elle existe dans toutes les bases locales et si une réalisation-réseau peut être "pointée" dans la base centrale et, au moins, une base régionale.



3.2.4. Notion de logiciel et de table de répartition

Du fait de la répartition, les mises à jour sur certains types de caractéristiques doivent être réparties c'est-à-dire transmises à d'autres sites pour y être exécutées. Ce ne sont pas directement les logiciels d'accès aux données (cf. II.4.3.) qui effectuent les opérations liées à la répartition mais un "Logiciel de répartition" unique sur chaque site possédant une base locale.

Ce logiciel dispose d'une table de répartition qui, pour chaque entité-réseau, indique le type de répartition associé à chaque caractéristique. Ainsi, pour chaque caractéristique qui fait l'objet d'une mise à jour, le logiciel de répartition consulte la table et sait donc, selon le type de répartition, quelle suite donner à la mise à jour : action purement locale (cas de non redondance) ou action locale et distante.

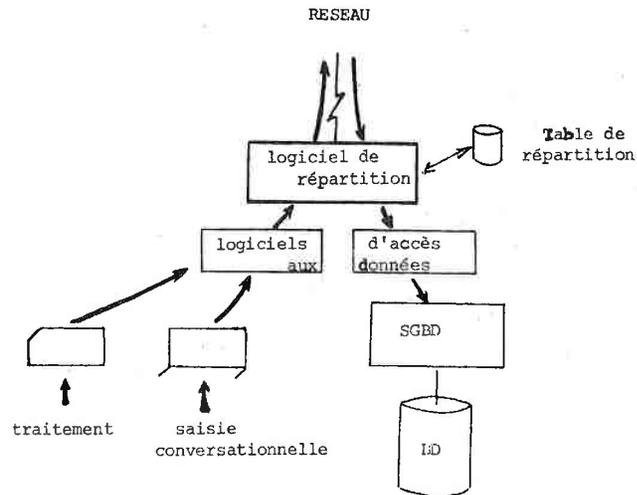


Figure III.17. : Logiciel et table de répartition d'un site

Sur un site, les logiciels d'accès et le logiciel de répartition ne "manipulent" que des caractéristiques figurant dans la description locale ; ainsi, sur un site régional, ne sera jamais manipulée une caractéristique centrale : une telle caractéristique ne pourra pas faire l'objet d'une création ou d'une modification régionale. Les manipulations que font les logiciels d'accès et le logiciel de répartition d'un site proviennent soit, d'actions locales obtenues par saisie ou depuis des programmes de traitement, soit d'actions distantes obtenues par le réseau et transmises par le logiciel de répartition symétrique sur l'axe (cf. 2.3.2.).

Remarquons que les interrogations qui sont des opérations locales (cf. 2.2.) ne sont donc pas soumises à une quel-

conque répartition et ne sont pas prises en considération par le logiciel de répartition ; elles sont traitées directement par les logiciels d'accès.

3.2.5. Reconnaissance des réalisations-réseau

Une réalisation-réseau est composée de plusieurs réalisations locales. Pour reconnaître que plusieurs réalisations locales correspondent à une même réalisation-réseau ou impose que chaque réalisation locale possède une donnée permettant de faire ce lien. Autrement dit, toute entité-réseau doit posséder une caractéristique de type DRT qui contienne un numéro global, c'est-à-dire valable pour tous les sites, qui permette d'identifier la réalisation-réseau répertoriée.

Ce numéro peut être implicite : par exemple pour une entité "PERSONNE" le numéro INSEE peut permettre d'identifier toute personne répertoriée dans la base répartie. Le numéro peut être attribué de manière explicite par le site central lors de la création de la réalisation-réseau.

Dans le cas d'imbrications, chaque objet imbriqué doit également posséder une telle donnée.

3.3. Exemple d'utilisation du modèle de répartition

étoilé

Considérons l'objet-réseau FOURNISSEUR et supposons qu'une réalisation de cet objet-réseau soit utilisée et répartie sur deux des quatre axes existants.

L'entité-réseau est composée des caractéristiques suivantes :

- A-MARFOU : Anneau des marchés relatifs à un même fournisseur ; cet anneau existe sur chaque site, c'est donc une caractéristique DRT .
- NUFOUR : Contient un numéro attribué, par le site central, à chaque fournisseur physique ; c'est donc une donnée qui permet de reconnaître sur chaque site un même fournisseur ; c'est une DRT.
- NOMFOUR : Contient le nom du fournisseur ; n'intéresse que les sites régionaux et comme le nom est unique, cette donnée existe donc dans chaque base régionale qui stocke un même fournisseur : c'est une DRR.
- ADR FOUR : Contient l'adresse du fournisseur ; c'est une DRP.
- SS FOUR : Contient le numéro de sécurité sociale du fournisseur ; donnée nécessaire sur chaque site : c'est donc une DRT.
- DATENTREE : Contient la date à laquelle un fournisseur est créé sur un axe et cette donnée n'intéresse que le site régional : propre à un axe, c'est donc une DRP.
- CAGLOB : Contient le chiffre d'affaires avec l'ensemble des sites ; c'est donc une donnée propre à l'ensemble des axes ; mais elle est centralisée dans la base centrale ; c'est une DCP.

- SITE (*) : Caractéristique introduite dans la base centrale pour contenir les DCR (cf. 3.2.2.) ; c'est une DCP par définition.
- NOSITE : Contient, dans la réalisation centrale, les numéros des sites sur lesquels un fournisseur est répertorié : DCP ayant autant de réalisations que d'axes.
- CAREG : Contient le chiffre d'affaires régional du fournisseur ; donnée propre à un site régional intéressant le site et le site central ; c'est une DCR.

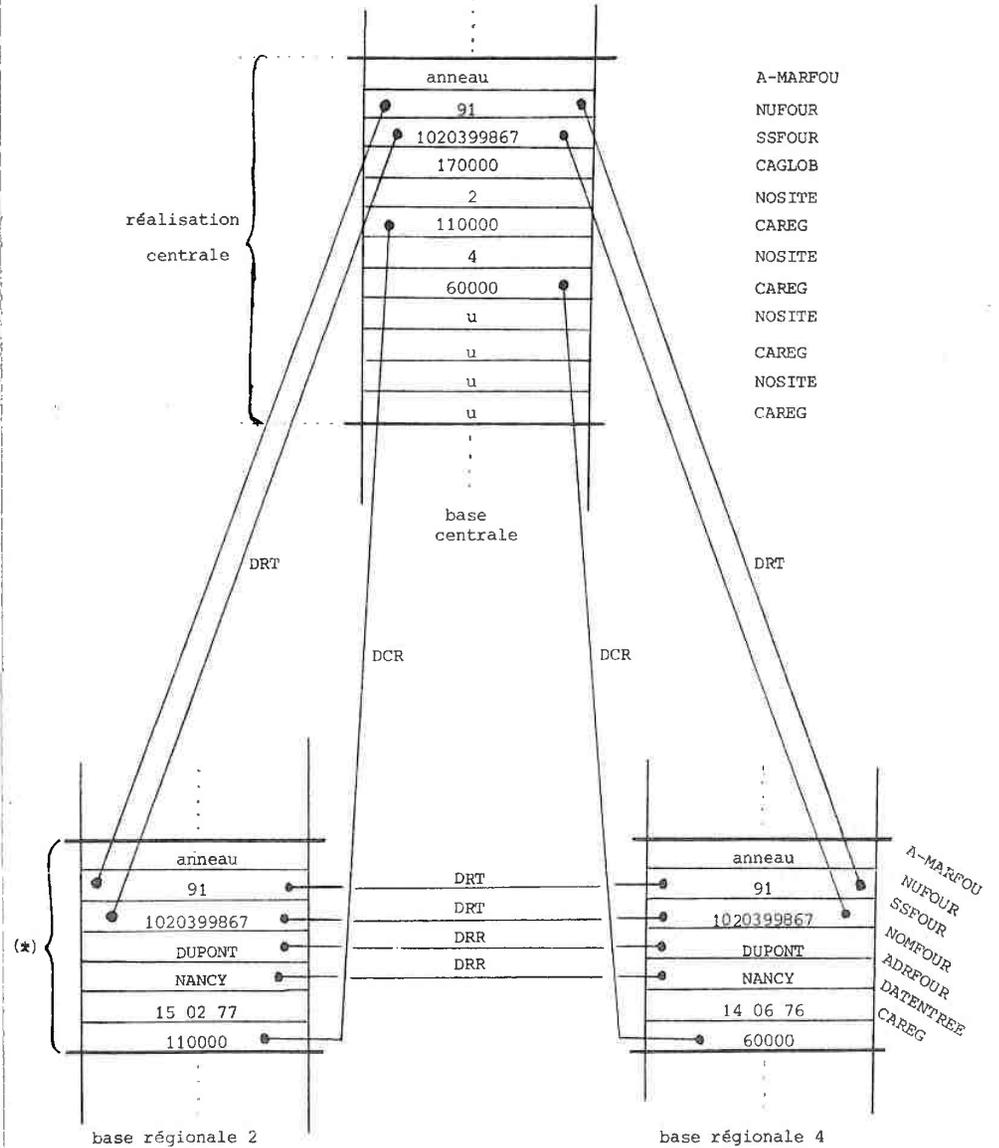
(*) Cette caractéristique centrale permet de localiser les axes sur lesquels, à un instant t, existe une même réalisation-réseau "Fournisseur" ; utilisée pour les modifications de DRR ou DRT.

A partir de cette entité-réseau les descriptions centrale et régionale sont les suivantes :

entité (2000) FOURNISSEUR	centrale
début	
A-MARFOU anneau	DRT
NUFOUR de 1 à 99	DRT
SSFOUR mot (13)	DRT
CAGLOB de 0 à 9999999	DCP
entité (4) SITE	DCP
début	
NOSITE mot (1)	DCP
CAREG de 0 à 9999999	DCR
fin	
fin	

entité (500) FOURNISSEUR	régionale
début	
A-MARFOU anneau	DRT
NUFOUR de 1 à 99	DRT
SSFOUR mot (13)	DRT
NOMFOUR mot (20)	DRR
ADRFOUR texte (5 60)	DRR
DATENTREE mot (6)	DRP
CAREG de 0 à 9999999	DCR
fin	

Si l'on considère la répartition des données d'une réalisation-réseau "fournisseur" sur les axes 2 et 4 de la base répartie on obtient :



4. ARCHITECTURE DE LA BASE REPARTIE ETOILEE

Le modèle de répartition que nous venons de présenter définit et exprime une architecture particulière de base de données répartie, à savoir une base répartie étoilée. Cette architecture correspond au type d'organisation d'application répartie présentée dans le chapitre II.

L'architecture externe de la base étoilée est exprimée par l'existence d'une base locale sur chaque site de l'application répartie et par la définition des axes central-régional sur lesquels s'effectuent les opérations liées à la répartition.

L'architecture interne est exprimée par l'existence de deux descriptions logiques différentes, centrale et régionale, et par la définition de cinq types différents de caractéristiques d'entité, ces types exprimant cinq modes différents de répartition des données.

Les bases locales qui composent la base répartie ont une description logique qui certes est imposée par les besoins globaux de l'application répartie, mais sont indépendantes pour ce qui concerne leur utilisation. Les usagers d'une "région" accèdent leur base locale et effectuent leurs traitements indépendamment de ce qui se passe sur les autres sites. Le contexte global de base répartie n'est pas rendu visible aux utilisateurs ; il est assuré de manière automatique par un logiciel de répartition qui est implanté sur chaque site disposant d'une base locale et qui fonctionne selon le modèle de répartition. Nous verrons plus tard que ce logiciel s'intègre dans un logiciel plus large, l'ensemble de ces logiciels répartis formant alors un interface entre les ensembles "SGBD + BD" répartis.

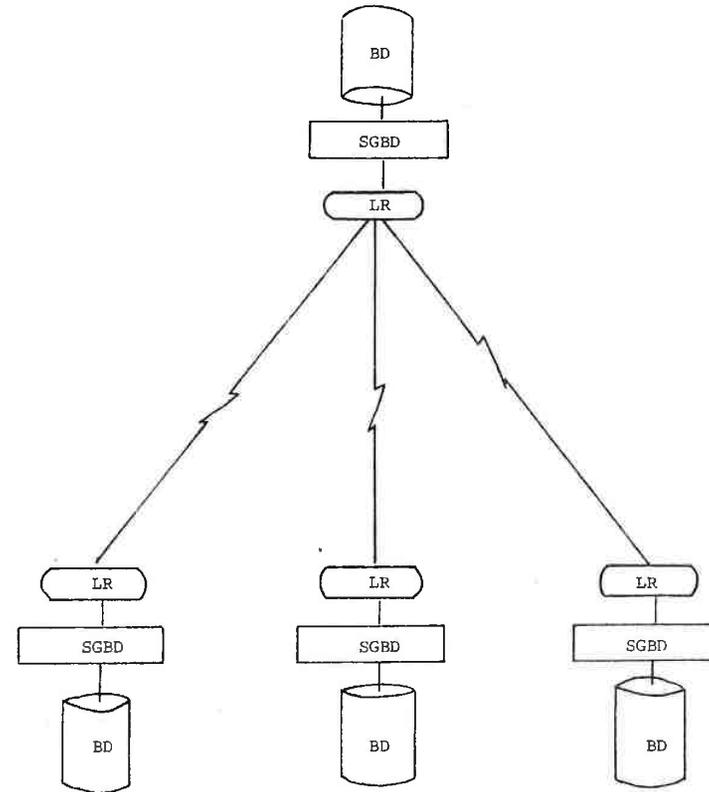


Figure III.18. : Base répartie étoilée

CHAPITRE IV

MISE À JOUR ET COHÉRENCE DE LA BASE RÉPARTIE .

1. COHERENCE LOCALE ET COHERENCE GLOBALE
2. MISES A JOUR DE LA BASE REPARTIE
3. NIVEAU DE COHERENCE DE LA BASE REPARTIE
4. MECANISME DE COHERENCE GLOBALE PROPOSE

Dans ce chapitre nous nous intéressons à la mise à jour de la base répartie et à sa cohérence.

Dans une première partie nous définissons ce qu'il faut entendre par "cohérence de la base répartie". La base répartie se compose de bases locales chacune étant gérée par un système propre. Ceci nous amène à distinguer deux niveaux de cohérence : la "cohérence locale" de chaque base locale qui est assurée par un système de gestion, et la "cohérence globale" qui est associée au fait que les données sont réparties et qui doit être assurée à un niveau supérieur à celui des systèmes de gestion locaux. L'ensemble du chapitre est, en fait, consacré à la "cohérence globale".

Une architecture de base répartie ayant été proposée, il faut donner aux utilisateurs des possibilités de manipuler la base. Cela nous amène à définir, dans une seconde partie, des opérations de mise à jour de la base répartie. Ces opérations sont conçues en fonction du modèle étoilé ; elles sont de trois types : création, modification, suppression. Chaque opération de mise à jour est une "action répartie" exécutée sur la base répartie et se décompose en actions élémentaires ; chaque action élémentaire pouvant exécuter une opération de mise à jour (création, modification, suppression) dans une base locale. Du fait même que les opérations de mise à

jour sont des "actions réparties" elles peuvent altérer la cohérence globale de la base répartie.

Pour assurer la cohérence globale, face à des actions réparties de mise à jour, divers mécanismes sont possibles. Le choix d'un mécanisme dépend des contraintes qu'il pose à l'application et du "niveau de cohérence" qu'il assure. Dans une troisième partie du chapitre nous présentons les types de mécanisme possibles et la notion de "niveau de cohérence".

Une quatrième partie du chapitre est consacrée à la proposition d'un mécanisme assurant un "niveau de cohérence" suffisant pour le type d'application auquel nous nous intéressons. La principale caractéristique de ce mécanisme est l'introduction, dans l'exploitation de la base répartie, de sessions périodiques de mise en cohérence globale.

1. COHERENCE LOCALE ET COHERENCE GLOBALE

1.1. Cohérence locale

Lorsque l'on dispose d'une base de données mise à la disposition d'une communauté d'utilisateurs il est difficile d'éviter que l'ensemble des données ne soit, à un moment ou à un autre, perturbé par une manipulation interdite ou maladroite. D'où la nécessité de faire "gérer" la base par un système de gestion qui assure à tout utilisateur et à tout instant que chaque donnée soit correcte : la base est alors cohérente.

Les manipulations de données sont de deux types : les interrogations et les mises à jour. Susceptibles d'altérer la cohérence de la base ces manipulations doivent être contrôlées par le système de gestion. Le système assure des contrôles à partir de "règles" établies à la conception et à la description de la base ; elles concernent l' "intégrité" des données, la "confidentialité", et l'organisation des accès simultanés.

Des contraintes d'intégrité sont fournies au système à la description de la structure logique de la base ; pour les accès qu'il effectue, le système est chargé de respecter ces contraintes. Si, par exemple, le numéro d'un fournisseur doit être compris entre 1 et 99 alors, à la description de la base, la contrainte s'exprime par : NUFOUR DE 1 A 99 (cf. I.2.3.). Grâce à cette contrainte, un utilisateur ne pourra jamais attribuer un numéro supérieur à 99 à un fournisseur répertorié dans la base.

Une donnée peut ne pas être rendue accessible à certains utilisateurs ; la confidentialité est alors assurée grâce à des mots de passe associés à des macro-instructions utilisables par des utilisateurs ayant le droit d'accès ; lors de l'appel

d'une macro-instruction, le système vérifie les autorisations d'accès et, suivant le résultat, exécute ou non la macro.

Pour éviter que plusieurs mises à jour s'effectuent au même instant sur un même sous-ensemble de données en relation entre elles, le système effectue le blocage de tout sous-ensemble pendant la durée de toute mise à jour ; ainsi tout accès, en interrogation ou en mise à jour, ne peut être fait qu'après que la mise à jour en cours soit terminée.

1.2. Cohérence globale

Dans le cas d'une base répartie, il existe plusieurs bases locales et chaque donnée appartient à l'une d'entre elles ; la cohérence d'une donnée doit donc être assurée par le système de gestion local. Mais cette donnée, comme toutes autres, participe à un contexte global qui est réparti et doit non seulement être correcte localement mais également "globalement" : cela introduit un second niveau de cohérence, la cohérence globale, qui n'est pas assurée par les systèmes de gestion locaux.

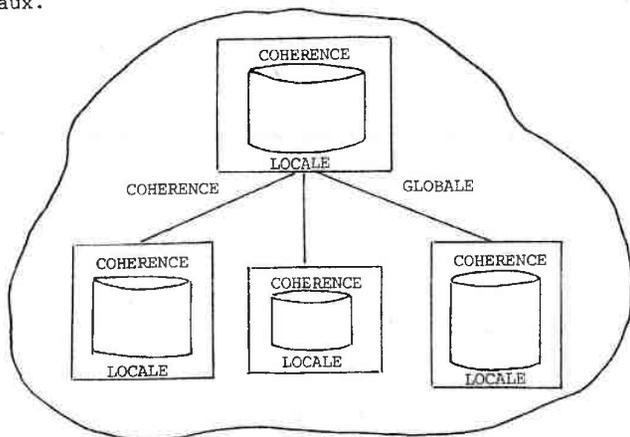


Figure IV.1. : Cohérence locale et cohérence globale

Une incohérence globale affecte toujours une réalisation-réseau entière (cf. I.1.2.2.2.). En effet, l'application répartie dispose d'opérations de mise à jour (cf. 2) qui "agissent" au niveau d'une réalisation-réseau et ce sont ces mêmes opérations qui sont susceptibles d'altérer la cohérence globale de la réalisation-réseau.

L'incohérence globale d'une réalisation-réseau s'exprime alors :

- soit, au niveau d'une réalisation locale faisant partie de la réalisation-réseau ; cette réalisation locale peut être anormalement présente ou absente, voire incomplète, sur un site. Un tel défaut résulte d'une création ou d'une suppression mal transmise à un site ou mal exécutée.
- soit, au niveau d'une donnée ponctuelle qui existe en plusieurs exemplaires dans la réalisation-réseau - c'est le cas des données des types DCR, DRT, DRR (cf. I.3.1.3.) - et cette donnée ne possède pas la même valeur selon l'exemplaire. Un tel défaut résulte d'une modification qui n'a pas été exécutée sur chaque exemplaire.

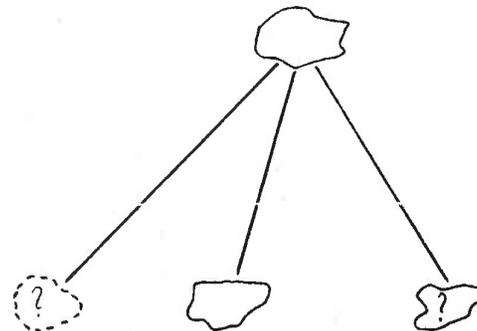


Figure IV.2. : Incohérence globale au niveau de réalisations locales d'une même réalisation-réseau

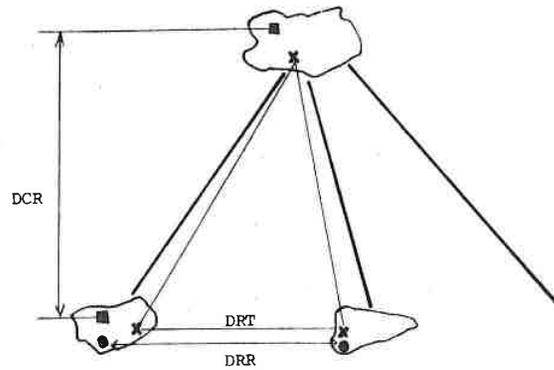


Figure IV.3. : Incohérence globale entre données redondantes d'une même réalisation-réseau

1.3. Le logiciel de cohérence

Les cas d'incohérence globale ne sont pas résolus par les différents systèmes de gestion locaux.

Chaque site est muni d'un logiciel de cohérence et l'ensemble de ces logiciels doit développer un mécanisme de mise en cohérence globale de la base répartie.

Le fonctionnement de ce mécanisme doit tenir compte d'une part du modèle de répartition choisi pour la base répartie et d'autre part des contraintes d'exploitation imposées par l'application répartie.

Sur un site, le logiciel de cohérence se situe au même niveau que le logiciel de répartition (cf. III.3.2.4.). Il est en relation avec le logiciel de cohérence qui lui est symétrique sur un axe, mais il est également en relation avec les logiciels locaux d'accès à la base.

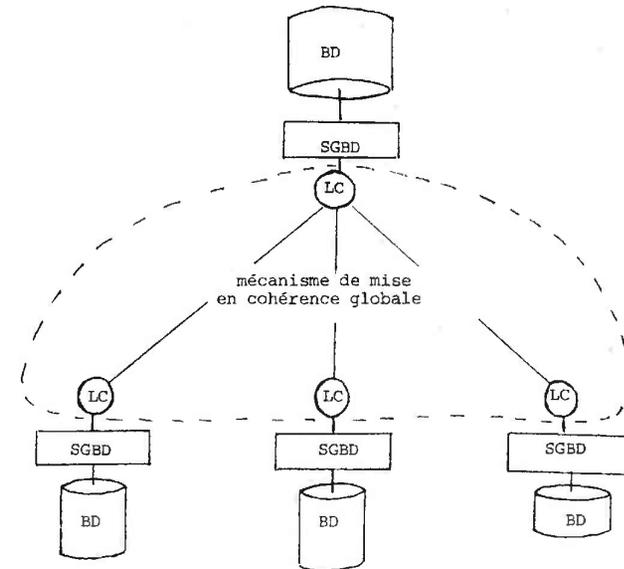


Figure IV.4. : Logiciel de cohérence

1.4. Cohérence de la base répartie

La base répartie est cohérente

- si la cohérence globale est assurée par l'ensemble des logiciels de cohérence et,
- si la cohérence locale est assurée sur chaque site par le système de gestion de la base locale.

$\text{cohérence de la base répartie} = \text{cohérence globale} + (n) \text{ cohérence locale}$
$\begin{matrix} \uparrow & & \uparrow \\ \text{logiciels de cohérence} & & \text{systèmes de gestion locaux} \end{matrix}$

2. LES MISES A JOUR DE LA BASE REPARTIE

2.1. Définition

Pour la manipulation de la base répartie, nous définissons trois opérations de mise à jour qui sont compatibles avec le modèle de répartition étoilée.

- La création, qui introduit ou développe une réalisation-réseau dans la base répartie ; elle permet de générer les réalisations locales qui composent une réalisation-réseau.
- La modification, qui permet de changer certaines valeurs d'une réalisation-réseau.
- La suppression, qui permet, en supprimant des réalisations locales, de réduire ou de supprimer une réalisation-réseau.

Ces opérations, lorsqu'elles s'exécutent, sont relatives à une certaine réalisation-réseau et peuvent, de ce fait, mettre en jeu plusieurs sites et nécessiter des échanges sur le réseau. Sur chaque site où une telle opération doit s'exécuter, elle s'adresse à une certaine réalisation locale ; celle-ci peut subir selon les cas une création, une modification ou une suppression, ces opérations étant définies au sens du système de gestion de la base locale.

Les opérations que nous détaillons dans ce paragraphe sont donc des opérations de manipulation de la base répartie utilisées par l'application répartie ; elles ne doivent pas être confondues avec les opérations élémentaires de mise à jour (création, modification, suppression) offertes par un système de gestion de base de données habituel.

2.2. La création

2.2.1. Création principale et créations secondaires

L'existence d'une réalisation-réseau dans la base répartie peut résulter d'une succession de créations plus ou moins rapprochées dans le temps. La première création, celle qui fait apparaître une réalisation-réseau dans la base répartie, est la création principale ; elle concerne un ou plusieurs axes, ceux qui sont intéressés par la réalisation. Puis, ultérieurement, d'autres sites peuvent être intéressés, et des créations secondaires sont donc exécutées ; celles-ci permettent d' "élargir" la "surface" d'utilisation de la réalisation-réseau.

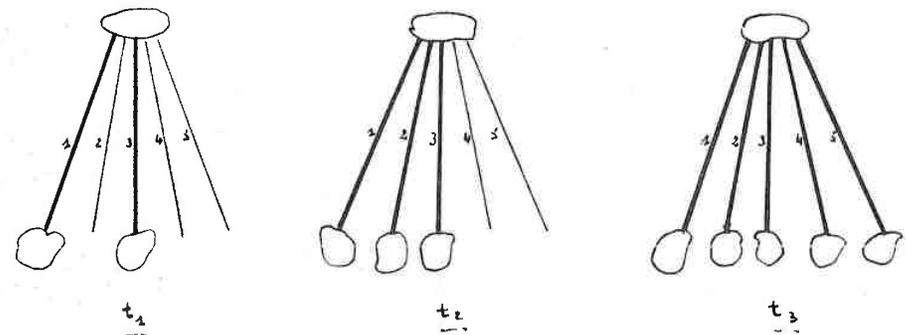
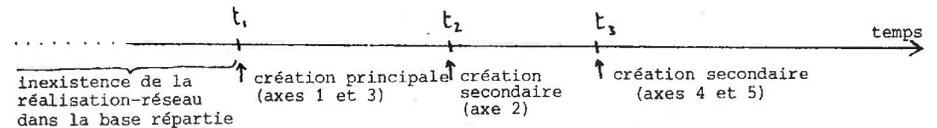


Figure IV.5. : Evolution d'une création de réalisation-réseau

Dans certains cas d'utilisation, une réalisation-réseau peut ne faire l'objet que d'une seule création, la création principale ; elle peut, par exemple, concerner tous les axes.

Relativement à une même réalisation-réseau, une base locale ne possède au maximum qu'une seule réalisation locale. La réalisation centrale peut évoluer au fur et à mesure que se font des créations secondaires, mais elle reste unique : des données centrales peuvent être ajoutées, d'autres modifiées.

2.2.2. Données saisies et données calculées

Lors d'une création des données sont introduites dans la base répartie.

Parmi ces données, certaines sont introduites directement par l'utilisateur ; un programme conversationnel saisit ces données.

D'autres sont calculées par les programmes de création à partir d'autres données déjà répertoriées dans la base et sont introduites dans la nouvelle réalisation générée tout comme les données saisies.

2.2.3. Répartition d'une création

L'action de création se répartit sur les axes central-régional (cf. III.2.3.1.).

Une création, principale ou secondaire, émise sur un site régional est une création régionale et les données, dans ce cas, ne sont réparties que sur un seul axe.

Par contre, une création émise sur le site central, peut répartir des données sur plusieurs axes ; on parle alors de création centrale.

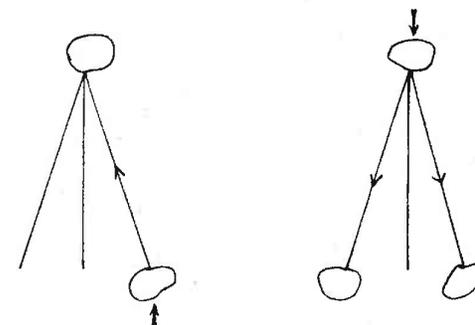


Figure IV.6. : Création régionale et création centrale

Toute action de création, principale ou secondaire, est répartie, c'est-à-dire qu'elle se développe sur un ou plusieurs axes selon les cas. ; elle met donc en jeu au minimum deux sites reliés par le réseau et se déroule donc en plusieurs étapes.

a) Cas d'une création centrale principale

1. Etape centrale

- saisie de données (DCP, DRT, DCR) par un programme conversationnel central
- génération d'une réalisation d'entité centrale
- calcul de données (cf. 2.2.2.)
- stockage des données saisies et calculées
- transmission de données (DRT, DCR) aux sites régionaux concernés

2. Etape régionale (sur chaque site régional concerné)

- réception de données du site central (DRT, DCR) et de l'ordre de création
- génération d'une réalisation d'entité régionale
- stockage des données reçues
- éventuellement, calcul de données
- éventuellement, transfert de données calculées (DRT, DCR) au site central

3. Etape centrale (éventuelle)

- réception de données des sites régionaux destinés à compléter la réalisation centrale
- ajout des données reçues à la réalisation centrale

La troisième étape se déroule en fait dans les cas où la réalisation centrale doit être complétée par des données émises par les sites régionaux ; dans le cas contraire, la création se limite à deux étapes.

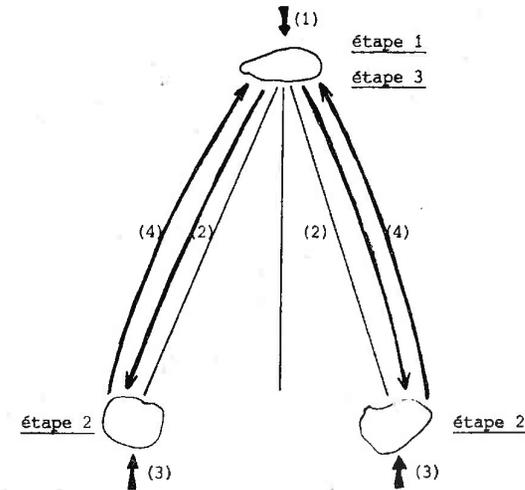


Figure IV.7. : Cas d'une création centrale concernant deux axes

b) Cas d'une création centrale secondaire

L'évolution est la même à l'exception des faits que dans l'étape 1 :

- les données ne sont pas saisies mais récupérées dans la réalisation d'entité centrale déjà existante
- il n'y a pas génération d'une réalisation centrale puisqu'elle a été générée lors de la création principale
- il y a modification de données centrales pour indiquer qu'il y a création secondaire sur tel et tel axe

c) Cas d'une création régionale principale

1. Etape régionale

- saisie de données (DRP, DRR, DCR, DRT)
- génération d'une réalisation d'entité régionale
- calcul de données (cf. 2.2.2.)

- stockage des données saisies et calculées
- transmission de données (DCR, DRT) au site central

2. Etape centrale

- réception de données d'un site régional (DCR, DRT) et de l'ordre de création
- génération d'une réalisation d'entité centrale
- éventuellement calcul de données (DCP, DCR, DRT)
- stockage des données dans la réalisation
- éventuellement transfert de données (DCR, DRT) au site régional initial

3. Etape régionale (éventuelle)

- réception de données calculées sur le site central (DCR, DRT)
- stockage de ces données dans la réalisation régionale

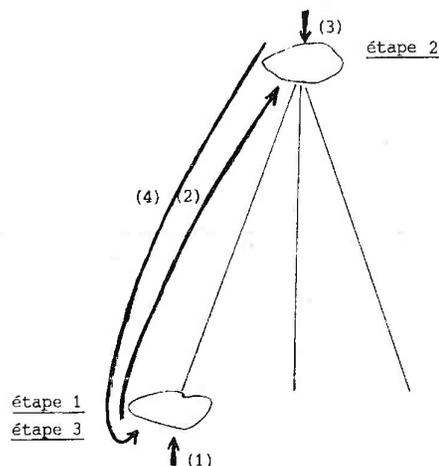


Figure IV.8. : Cas d'une création régionale

d) Cas d'une création régionale secondaire

L'évolution est la même que dans le cas précédent à l'exception des faits que dans l'étape 2 :

- il n'y a pas génération d'une réalisation centrale puisque celle-ci a été générée lors de la création principale.
- il y a modification de données centrales pour indiquer qu'il y a création secondaire sur tel nouvel axe.
- il n'y a pas calcul de nouvelles données centrales ; les données à transmettre au site régional existent dans la réalisation centrale.

2.3. La modification

Sur un site, les modifications saisies ne peuvent concerner que des données stockées dans une réalisation locale. Cependant, toute réalisation locale fait partie d'une réalisation-réseau qui est répartie et les modifications apportées à une réalisation locale peuvent intéresser les autres réalisations locales de la même réalisation-réseau, d'où la nécessité, dans certains cas de répartir l'opération de modification.

2.3.1. Modification centrale

Sur le site central une modification peut concerner les données représentées par les caractéristiques DCP, DRT, DCR.

- Modification des DCP :

Une DCP n'existe que dans la base centrale et sa modification est donc une action purement locale (la modification n'est pas transmise à un autre site).

- Modification des DCR :

Lorsque le site central saisit une modification pour une DCR, celle-ci existe non seulement sur le site central mais aussi sur un certain site régional ; la modification doit donc être transmise et s'effectue donc en deux étapes, l'une centrale, l'autre régionale.

- Modification des DRT :

La DRT existe sur le site central mais aussi dans une ou plusieurs bases régionales ; sa modification s'exécute donc en deux étapes : étape centrale et étape régionale ; l'étape régionale s'exécutant dans chaque base régionale qui possède la DRT.

2.3.2. Modification régionale

Sur un site régional une modification peut concerner les données représentées par les caractéristiques DRP, DRR, DCR, DRT.

- Modification des DRP :

Une DRP n'existe que dans une seule base régionale ; sa modification est une action locale.

- Modification des DRR :

Une DRR est susceptible d'exister dans plusieurs bases régionales ; toute modification doit donc être affectée à chaque exemplaire. Elle est exécutée en trois étapes :

- exécution sur le site régional récepteur et transfert au site central.

- reconnaissance de l'existence d'autre(s) exemplaire(s) et sur quel(s) site(s) régional(s) ? Transfert sur d'autres axes.
- exécution sur chaque site régional qui possède un exemplaire de la DRR (excepté le site initial).

- Modification des DCR :

S'exécute en deux étapes : sur le site régional récepteur et sur le site central.

- Modification des DRT :

Une DRT est susceptible d'exister sur plusieurs axes ; sa modification s'exécute en trois étapes :

- exécution sur le site régional récepteur et transfert au site central.
- exécution sur le site central et reconnaissance de l'existence de la donnée sur d'autres axes ; transfert de la modification aux sites régionaux possesseurs d'un exemplaire.
- exécution sur les sites régionaux.

La modification d'une réalisation locale peut donc provoquer des actions locales (cas des DCP et DPP) sans incidence sur la cohérence globale de la base répartie et des actions réparties qui s'exécutent en deux ou trois étapes sur des sites différents.

2.4. La suppression

La suppression est l'action symétrique de la création. Comme les deux actions précédentes elle est relative à une réalisation-réseau.

2.4.1. Suppression totale et suppression partielle

Comme pour la création, la suppression d'une réalisation-réseau de la base répartie peut être échelonnée dans le temps et se faire en plusieurs suppressions.

Une suppression ordonnée par un site régional ne peut concerner que l'axe sur lequel se trouve le site ; elle peut donc concerner au maximum deux sites. Par contre une suppression ordonnée par le site central peut, selon les cas, concerner un seul axe ou, au contraire, plusieurs axes.

Une suppression qui retire l'existence d'une réalisation-réseau de la base répartie est une suppression totale ; à la suite d'une telle action plus aucune donnée relative à la réalisation-réseau ne figure dans la base.

Si une réalisation-réseau n'est répertoriée que sur un seul axe, alors toute suppression régionale (figure 9a) ou centrale (figure 9b) est une suppression totale. Par contre, si une réalisation-réseau est répertoriée sur plusieurs axes, alors seul le site central peut effectuer une suppression totale (figure 9).

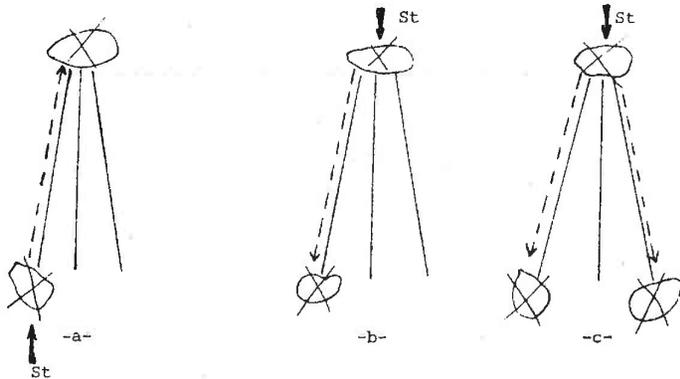


Figure IV. 9. : Les trois cas de suppression totale

Une réalisation-réseau peut également être partiellement supprimée ; c'est-à-dire que la suppression ne concerne qu'un seul axe ou plusieurs sans toutefois concerner tous les axes sur lesquels la réalisation-réseau possède des données.

Le site central peut ordonner une suppression partielle pour un ou plusieurs axes : sur le site central des données de la réalisation centrale sont modifiées et supprimées (certaines DCP et DCR) mais la réalisation centrale demeure ; sur les sites régionaux des axes concernés, les réalisations sont supprimées. Un site régional peut ordonner une suppression partielle sur son axe : sa réalisation est supprimée et dans la réalisation centrale des données sont supprimées et d'autres modifiées (certaines DCP et DCR).

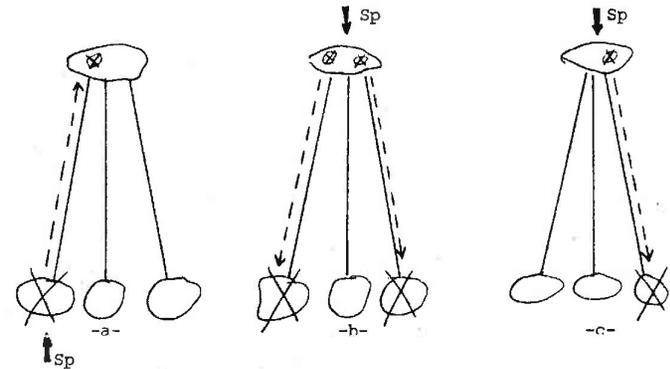


Figure IV.10. : Cas de suppression partielle

Quand le site central reçoit d'un site régional un ordre de suppression alors

- soit la réalisation-réseau n'existe que sur l'axe, la suppression est alors totale et la réalisation centrale est supprimée,
- soit la réalisation-réseau existe sur d'autres axes et ceux-ci ne sont alors pas concernés par la suppression ; seules les données de l'axe initial sont supprimées.

2.4.2. Répartition de la suppression

Toute action de suppression met au minimum deux sites en jeu ; s'est donc une opération qui s'exécute en plusieurs étapes.

. Suppression centrale totale (figures 9b, 9c)

1. Etape centrale

- saisie de l'ordre de suppression totale
- suppression de la réalisation centrale
- transmission de l'ordre de suppression aux sites régionaux concernés

2. Etape régionale (en parallèle sur chaque site régional possesseur de données de la réalisation-réseau)

- réception de l'ordre de suppression
- suppression de la réalisation régionale

. Suppression centrale partielle (figures 10b, 10c)

L'évolution est identique, mais la suppression porte sur certains axes précisés dans la saisie ; la réali-

sation centrale n'est pas détruite. Seules les données représentées par des caractéristiques DCR sont supprimées ainsi que des données DCP relatives aux axes concernés par la suppression.

. Suppression régionale (figures 9a, 10a)

1. Etape régionale

- saisie de l'ordre de suppression
- suppression de la réalisation régionale
- transmission de l'ordre au site central

2. Etape centrale

- réception de l'ordre de suppression
- existence de la réalisation-réseau sur d'autres axes ?
 - oui : suppression des DCR et DCP relatives à l'axe initial.
 - non : suppression de la réalisation centrale.

2.5. Les programmes de mise à jour

2.5.1. Définition d'une "chaîne" de programmes

Chacun des types de mise à jour définis précédemment donne lieu à l'existence d'une chaîne de programmes. Une chaîne est formée d'autant de programmes qu'il y a d'étapes dans la répartition de l'opération de mise à jour qu'elle exécute. Ainsi un programme Pi correspond à l'exécution de l'étape i de l'opération de mise à jour.

Une chaîne peut s'exécuter sur un seul axe ; c'est le cas des chaînes de création régionale , de suppression régionale, de création ou suppression centrale n'engageant qu'un seul axe. Elle peut aussi s'exécuter sur plusieurs axes ; dans ce cas un même programme P_i peut s'exécuter en parallèle sur plusieurs sites régionaux.

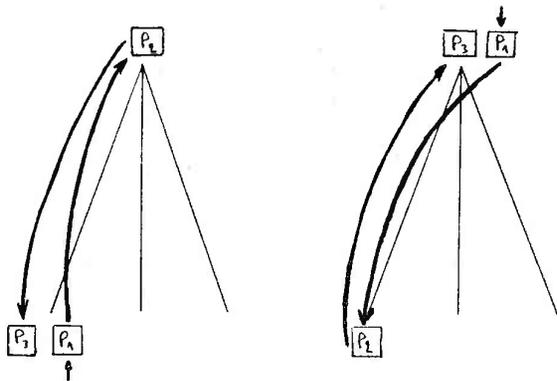


Figure IV.11. : Chaînes de programmes s'exécutant sur un seul axe

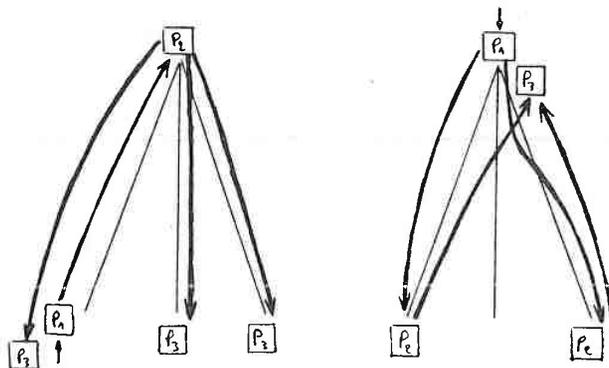


Figure IV.12. : Chaînes de programmes s'exécutant sur plusieurs axes

Un programme P_i exécute des mises à jour (création, modification, suppression) sur une base locale, alors que globalement une chaîne de programmes exécute une mise à jour sur la base répartie.

2.5.2. Les programmes P_i d'une chaîne

Le programme P₁ d'une chaîne est le programme qui entreprend une mise à jour de la base répartie. Il peut être soit un programme "lot", soit un programme conversationnel. Dans les deux cas P₁ s'exécute sur ordre de l'utilisateur.

Les programmes P_i constituent les programmes d'accès, en mise à jour, aux données de la base répartie puisque ce sont eux qui engendrent les actions sur appel externe.

Les autres programmes, P₂, P₃, etc..., s'exécutent après P₁ mais automatiquement (cf. V.3.3.).

Tous les programmes sont écrits en langage SOCRATE et sont précompilés.

Pour respecter le principe du modèle étoilé qui veut que tout site possède ses propres logiciels d'accès, chaque site possède le programme P₁ des chaînes qui correspondent aux opérations que le site peut entreprendre.

Ainsi, tout site régional peut posséder un programme P₁ pour les chaînes de

- création régionale principale
- création régionale secondaire (ces deux chaînes peuvent éventuellement être fondées en une seule)
- modification régionale
- suppression régionale (totale ou partielle)

Les programmes P2 de ces chaînes existent alors sur le site central, les programmes P3 sur les sites régionaux, etc...

Le site central peut posséder un programme P1 pour les chaînes de

- création centrale principale
- création centrale secondaire (ces deux chaînes peuvent éventuellement être fondées en une seule)
- modification centrale
- suppression centrale totale
- suppression centrale partielle (ces deux chaînes peuvent éventuellement être fondées en une seule)

Pour ces chaînes les sites régionaux possèdent les programmes P2, le site central possède les programmes P3, etc...

2.6. Actions réparties et actions élémentaires

2.6.1. Définitions

Les opérations de mise à jour décrites dans le paragraphe 2.4. mettent toutes plusieurs sites en jeu, se déroulent en plusieurs étapes successives, et nécessitent des transferts sur le réseau. Pour ces raisons nous considérons qu'elles constituent des actions réparties sur la base répartie. Chaque action répartie se décomposant en actions élémentaires, chacune s'exécutant sur une base locale par un programme P_i (cf. 2.5.).

Les actions élémentaires, d'une même action répartie, s'exécutent par étapes.

Un même site peut être l'objet de deux (ou plusieurs) étapes ; au cours de chacune il subit une action élémentaire différente. C'est le cas par exemple d'une création régionale ;

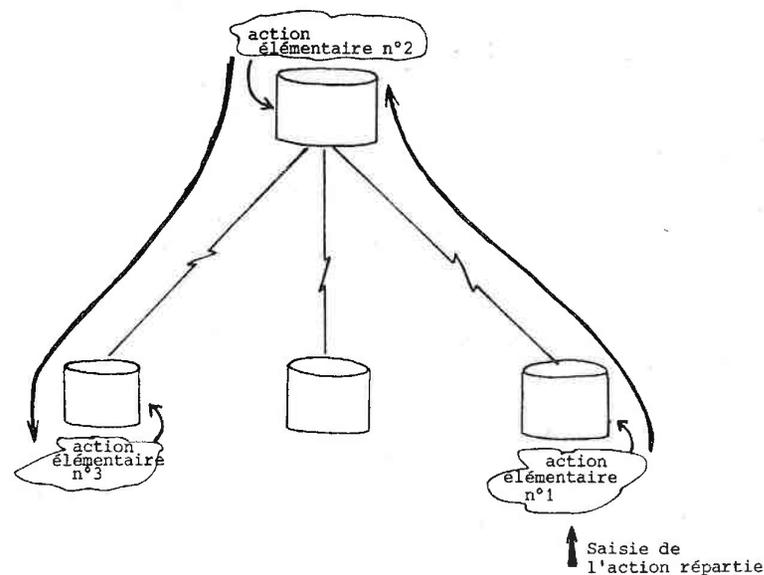


Figure IV.13. : Décomposition d'une action répartie en actions élémentaires

elle s'exécute en trois étapes : la première donne lieu à une action élémentaire sur le site initial, la seconde sur le site central, et enfin la troisième à nouveau sur le site initial. Une telle action répartie nécessite deux transferts sur le même axe.

Deux ou plusieurs sites peuvent être impliqués au cours d'une même étape ; les actions élémentaires qu'ils subissent sont alors généralement identiques. C'est le cas par exemple pour une modification d'une DRT redondante sur trois sites régionaux.

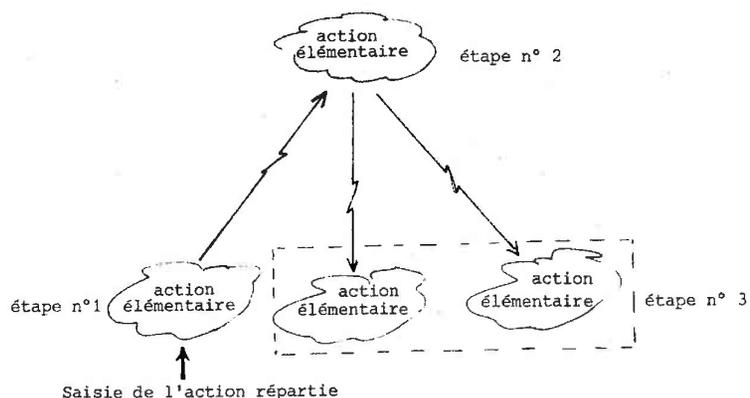


Figure IV.14. : Décomposition de l'action répartie de modification d'une D.R.T.

Du fait de la nécessité d'effectuer un transfert entre chaque étape et de mettre plusieurs sites en œuvre, une action répartie ne peut pas être considérée comme une action instantanée et il est important de considérer sa durée.

2.6.2. Durée d'une action répartie

Le début d'une action répartie correspond à l'instant où l'action est saisie (appel d'un programme P1) ; la fin de l'action répartie correspond à la fin de la dernière action élémentaire exécutée.

Trois facteurs essentiels interviennent dans la durée d'une action répartie :

- les temps de transmission sur le réseau entre les actions élémentaires. Si la connexion entre deux

sites est normale, le temps de transfert reste cependant relativement important par rapport à un temps d'exécution d'une action élémentaire.

- les temps d'attente, sur chaque site, avant l'exécution d'une action élémentaire : ces temps dépendent de la charge des sites au moment où une action élémentaire doit être exécutée.
- les temps d'exécution des différentes actions élémentaires.

Par rapport à une action élémentaire, la durée d'une action répartie est importante surtout si le nombre d'étapes est grand.

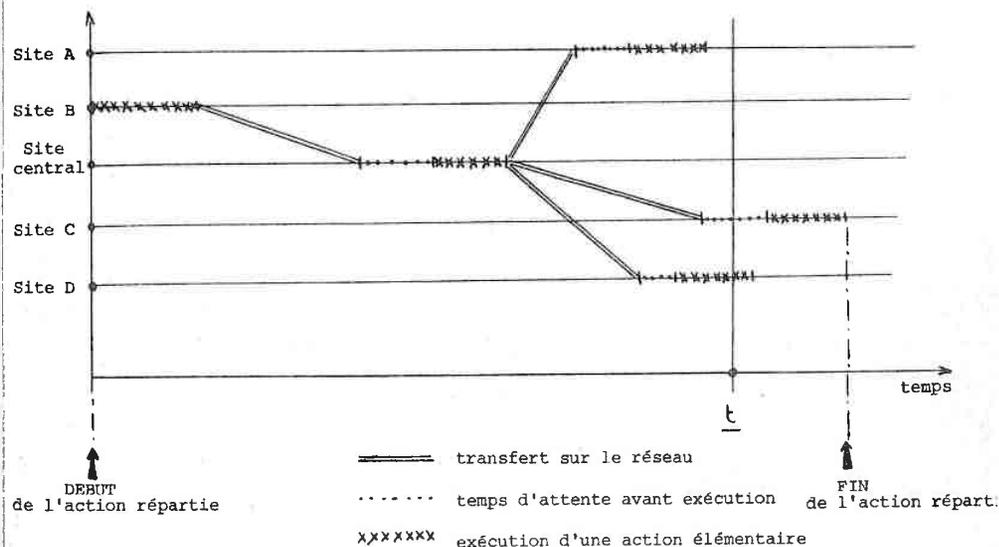


Figure IV.15. : Durée d'une action répartie

2.6.3. Action répartie et incohérences globales

Dans le cas d'une base locale, l'exécution d'une mise à jour provoque le blocage des données concernées et de celles qui sont en relation avec elles ; pendant toute la durée de la mise à jour aucun autre accès n'est possible à cet ensemble de données. Lorsque la mise à jour est terminée, la base a retrouvé sa cohérence et les accès sont à nouveau possibles.

Dans le cas d'une base répartie un blocage des données est à priori plus difficile à obtenir du fait de la durée d'une action répartie et du fait même que les données sont réparties. Il en résulte que l'exécution d'une action répartie sur une réalisation-réseau soulève deux problèmes :

- 1) Au cours de l'exécution d'une action répartie il y a des "décalages" entre les différentes réalisations locales qui composent la réalisation-réseau mise à jour (cf. figure IV.15) ; ces décalages se traduisent par l'absence (en création) ou la présence (en suppression) momentanément anormale d'une réalisation dans une base locale ou par des différences entre les différents exemplaires répartis d'une même donnée. L'ensemble de la réalisation-réseau est donc supposée incohérente pendant toute la durée de l'action répartie. Pendant cet état d'incohérence des interrogations sont possibles sur les différentes bases locales contenant la réalisation-réseau, excepté sur le ou les sites qui sont en train d'exécuter l'action élémentaire qui les concerne ; à un instant t selon le site interrogé, le résultat de l'interrogation risque de ne pas être le même. Supposons que la figure précédente représente la modification d'une DRT répertoriée sur les sites régionaux A, B, C, D et sur le site central. A l'instant t , selon le site, la valeur n'est

pas la même : sur les sites A, B, et le site central, la valeur est supposée exacte, elle a subi la modification ; sur le site D, l'accès à la donnée est impossible, la modification étant en cours (blocage local), et sur le site C la valeur est encore l' "ancienne" valeur.

- 2) Au cours de l'exécution d'une action répartie sur une certaine réalisation-réseau une autre action répartie peut être émise pour la même réalisation-réseau. Dans ce cas il risque d'apparaître une concurrence entre les deux actions. En effet celles-ci vont "générer" des actions élémentaires sur les mêmes sites. L'ordre d'exécution des deux actions élémentaires concurrentes ne sera pas le même sur tous les sites. A la fin des deux actions réparties certains sites auront pour résultat l'effet de la première action répartie ; tandis que les autres auront pour résultat l'effet de la seconde. D'où l'incohérence globale de la réalisation-réseau.

3. NIVEAU DE COHERENCE DE LA BASE REPARTIE

Pour assurer la cohérence globale de la base répartie face à des actions réparties telles que celles décrites dans le paragraphe précédent un mécanisme de mise en cohérence globale est nécessaire. Deux types de mécanismes sont à envisager : l'un synchronisé, l'autre non synchronisé.

3.1. Mécanisme synchronisé (COP-1, COP-2)

Ce type de mécanisme consiste à "synchroniser" l'exécution des différentes actions élémentaires d'une même action répartie. Ainsi dès qu'une action répartie est saisie, le mécanisme assure un blocage global de toute la réalisation-réseau concernée et les actions élémentaires s'exécutent donc pratiquement en même temps ; par blocage global il faut comprendre un blocage local sur chacun des sites mis en œuvre par l'action répartie. Le blocage reste permanent pendant toute la durée de l'action répartie. Une telle synchronisation s'effectue pour toute action répartie et elle est entreprise dès la saisie de l'action.

Le mécanisme synchronisé peut se décomposer en trois phases successives :

1. Phase de synchronisation

Cette phase consiste, à partir de la réception d'une action répartie, à obtenir le blocage sur chacun des sites impliqués dans l'action. Un message de synchronisation est envoyé à chacun des sites et chacun d'eux se met dès que possible en situation de blocage local pour les données concernées ; le blocage obtenu, le site doit transmettre au site initial un message signifiant qu'il est prêt pour l'exécution synchronisée.

2. Phase d'exécution

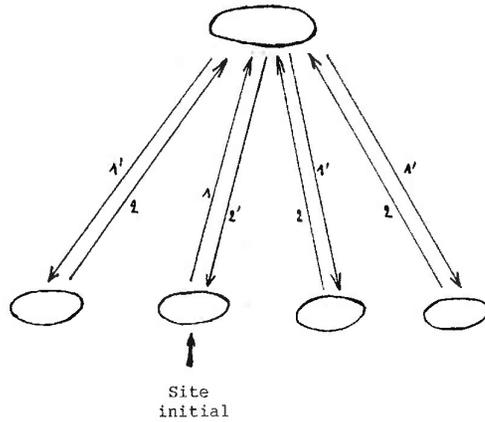
Lorsque le site initial a reçu le message "prêt" de tous les autres, alors il peut engager l'exécution de l'action répartie ; celle-ci s'exécute selon les processus décrits dans le paragraphe 2. L'ensemble des actions élémentaires s'exécute, tous les sites restant "bloqués" pour les données concernées.

3. Phase de désynchronisation

Tout site qui a terminé sa ou ses action(s) élémentaire(s) transmet un message de "fin" au site initial. Celui-ci ayant reçu tous les messages et ayant lui-même terminé ses exécutions envoie à chaque site un message de "FIN" et chaque site peut alors se "débloquer" : l'action répartie est terminée.

Tout algorithme synchronisé fonctionne selon le principe de ces trois phases ; quelques variantes sont cependant possibles selon l'organisation des échanges entre les différents sites qui composent l'application répartie (application étoilée ou non).

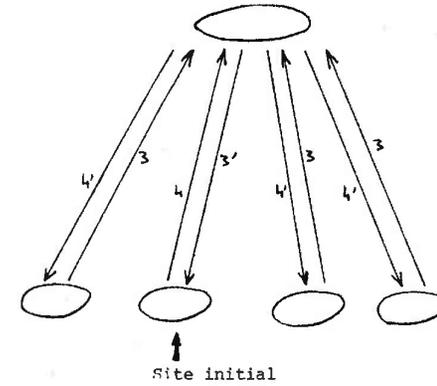
Si on considère le modèle étoilé proposé dans le chapitre III et une action répartie qui intéresse n sites régionaux le mécanisme synchronisé pourrait évoluer comme l'indique la figure IV.16.



a. Phase de synchronisation

- 1 : message de synchronisation du site initial au site central
- 1' : message de synchronisation du site central aux autres sites régionaux concernés
- 2 : message "prêt" envoyé par chaque site régional (excepté le site initial) ou site central lorsque le blocage local est obtenu
- 2' : message "prêt" du site central au site initial lorsque le central a reçu tous les messages "prêt" et obtenu le blocage central

Nombre de messages : $1 + (n-1) + (n-1) + 1 = 2n$



b. Phase de désynchronisation

- 3 : message de "Fin" envoyé par chaque site régional (excepté le site initial) au site central
- 3' : message "fin" envoyé par le site central au site initial dès la fin locale de ses exécutions et réception de (n-1) messages "Fin"
- 4 : message de Fin envoyé par le site initial lorsqu'il a reçu 3' et terminé ses exécutions
- 4' : message de Fin envoyé par le central aux sites régionaux après réception de 4.

Nombre de messages : $(n-1) + 1 + 1 + (n-1) = 2n$

Figure IV.16. : Mécanisme synchronisé pour le modèle étoilé

On s'aperçoit que pour le modèle étoilé, le mécanisme synchronisé nécessiterait 4 n transferts sur un axe lorsqu'une action répartie concerne n sites régionaux. Les transferts nécessaires par l'exécution de l'action répartie (phase d'exécution) ne sont pas représentés sur la figure ; ils dépendent en fait de l'action répartie. Si on considère l'exemple de la figure IV.15, le nombre de transferts pour l'exécution de l'action répartie sur 4 sites régionaux est de 4 ; au total 20 transferts sont donc nécessaires pour exécuter l'action répartie selon un mécanisme synchronisé :

- 8 transferts pour la synchronisation
- 4 transferts pour l'exécution
- 8 transferts pour la désynchronisation

3.2. Mécanisme non synchronisé

Ce type de mécanisme consiste à ne pas synchroniser chaque action répartie mais à établir des sessions synchronisées lors desquelles plusieurs actions réparties sont exécutées. Par session synchronisée, il faut comprendre que tous les sites sont présents pour une durée déterminée et à partir d'un certain instant.

Une action répartie qui est saisie sur un site est exécutée localement, c'est-à-dire que la première action élémentaire, celle qui correspond à la première étape, s'exécute normalement comme toute action locale (programme P1). Le reste de l'action répartie sera exécuté ultérieurement, par des programmes P_i (i > 1), lors de la session synchronisée.

Un tel mécanisme entraîne des incohérences globales momentanées du fait qu'une action répartie n'est que partiellement exécutée lorsqu'elle est saisie. Selon l'application,

les sessions synchronisées, qui permettent de remettre la base répartie en cohérence, sont plus ou moins rapprochées dans le temps. A la fin d'une session synchronisée, il n'y a plus d'action répartie en cours, toutes sont entièrement exécutées. Ainsi la cohérence de la base est obtenue périodiquement à la fin de chaque session synchronisée.

Une interrogation ou un traitement se fait toujours sur un état de cohérence qui correspond au dernier état de cohérence globale obtenu. Cependant, chaque site dispose instantanément de toutes les mises à jour qui apparaissent sur son site : donc entre deux sessions synchronisées une base locale possède la cohérence globale de la dernière session et, toutes les mises à jour survenues sur son site depuis la dernière session (cf. figure IV.17).

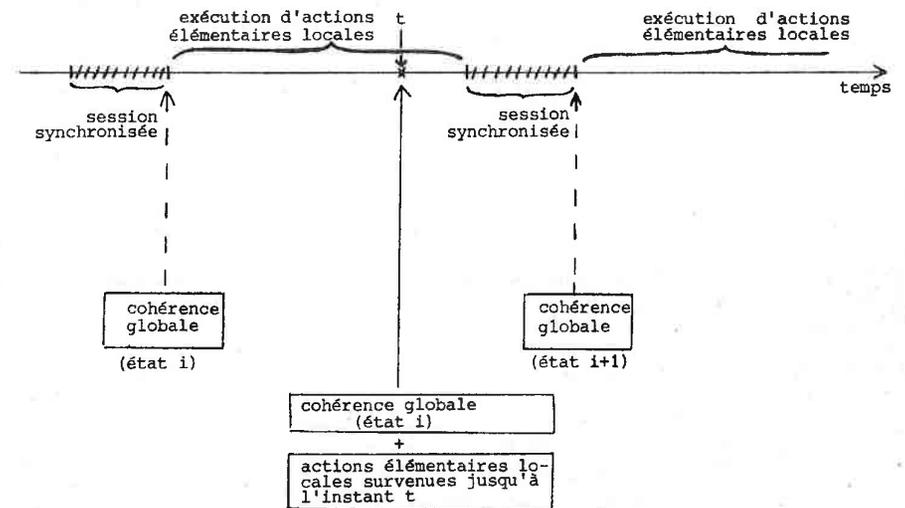


Figure IV.17. : Mécanisme non synchronisé

3.3. Choix d'un mécanisme

Le choix d'un mécanisme dépend essentiellement de l'application répartie qui utilise la base répartie. Une application est-elle en mesure de supporter un mécanisme synchronisé pour assurer la cohérence globale ou peut-elle se contenter d'un mécanisme non synchronisé ?

Bien qu'apportant à priori une meilleure cohérence globale, le mécanisme synchronisé présente quelques inconvénients qui risquent de rendre très "lourd" le fonctionnement de l'application répartie.

Les trois principaux inconvénients du mécanisme synchronisé sont les suivants :

- 1) Il nécessite une connexion quasi-permanente de tous les sites au réseau. Ceci peut être une contrainte d'exploitation très importante pour l'application. Cette connexion permanente est nécessaire du fait qu'une action répartie est entièrement exécutée dès sa réception. Que se passe-t-il si, pour une raison ou pour une autre, un ou plusieurs sites ne sont pas connectés ?
- 2) Il nécessite de nombreux messages sur le réseau, ce qui risque d'augmenter la charge de celui-ci. Le nombre de messages destinés à synchroniser et à désynchroniser les sites est beaucoup plus grand que ceux nécessités par les mises à jour elles-mêmes.
- 3) La durée globale d'une action répartie est longue vu le nombre de transferts nécessités. Il en résulte des temps de blocage très long. Globalement la base répartie risque d'être très souvent bloquée donc inaccessible pour des interrogations.

Malgré ces inconvénients majeurs un mécanisme synchronisé possède l'avantage d'assurer une cohérence globale permanente et quasi-instantanée ; ceci à la condition que le réseau soit très fiable et que tous les sites soient connectés en permanence.

Par contre, le mécanisme non synchronisé donne aux sites une plus grande indépendance en ce sens qu'il ne leur est pas nécessaire d'être constamment connectés au réseau. L'application ne travaille pas, dans ce cas, sur une base globalement cohérente à l'instant t, mais sur un état cohérent antérieur.

Un tel mécanisme nécessite un nombre négligeable de messages de synchronisation et de désynchronisation puisqu'ils sont valables pour tout un ensemble d'actions réparties exécutées de façon groupées.

Ce mécanisme semble plus souple à utiliser par une application répartie mais offre-t-il une cohérence globale suffisante ?

3.4. Notion de "niveau de cohérence"

Dans les deux mécanismes présentés apparaît la notion de "temps au bout duquel" est retrouvée la cohérence globale à partir de la réception d'une action répartie. Il est possible d'associer à ce temps un niveau de cohérence globale.

Ainsi si on parle d'un niveau correspondant à un délai de 12 heures, cela signifie qu'au bout de 12 heures, à partir de sa réception, une action répartie doit être entièrement exécutée. On dispose donc d'un délai de 12 heures pour effectuer entièrement l'action répartie.

Le niveau de cohérence maximum correspond à l'utilisation d'un mécanisme synchronisé puisque celui-ci exécute immédiatement toute action répartie.

Un mécanisme non synchronisé donne un niveau inférieur variant avec la fréquence des sessions de mise en cohérence. Ainsi si la fréquence des sessions est de l'heure, le niveau est supérieur à celui qui correspond à une fréquence de la journée.

En fait, le choix du mécanisme et la fréquence des sessions dans le cas non synchronisé dépend du niveau de cohérence voulu pour la base répartie.

Dans une application du genre "réservation de place" où la base doit être constamment en état cohérent, le niveau de cohérence doit être maximum.

Dans le cas d'une application du genre "gestion" où se font alternativement, par sessions, "mise à jour des données" et "utilisation des données pour traitements" le niveau de cohérence peut être inférieur. Une session de mise en cohérence peut être intercalée entre deux sessions de travail ; le niveau de cohérence correspond alors à la durée de la session de travail précédant une session de mise en cohérence. Ainsi pour toute session de travail la base répartie dispose de toutes les actions réparties reçues au cours de la session de travail précédente.

4. MECANISME DE COHERENCE GLOBALE PROPOSE

4.1. Choix d'un niveau de cohérence

L'application considérée (cf. chapitre II) ne nécessite pas un niveau de cohérence maximum ; un mécanisme synchronisé serait d'un emploi trop lourd pour l'application. Un mécanisme non synchronisé paraît plus adapté à l'application. Le niveau de cohérence choisi, suffisant pour l'application, correspond à la durée des sessions de travail, c'est-à-dire à peu près 12 heures. Ainsi, au "démarrage" d'une session de travail, la base est cohérente et possède toutes les actions réparties survenues au cours des 12 heures précédentes.

4.2. Mécanisme centralisé

Le mécanisme proposé doit tenir compte de l'architecture de la base répartie. Le mécanisme a donc un fonctionnement étoilé.

Il n'y a pas d'échange direct entre les sites régionaux. Toute action répartie donne lieu à au moins une action élémentaire centrale. Entre deux actions élémentaires régionales, il y a nécessairement une transition par une action élémentaire centrale ; rappelons que la base centrale joue un rôle de localisation des données régionales (cf. III.3.3.2.).

Si on considère par exemple la modification d'une DRR existant sur trois sites régionaux, il y a obligatoirement une action élémentaire centrale qui consiste à localiser les sites sur lesquels existe une copie et donc auxquels il faut transmettre les éléments nécessaires pour exécuter l'action élémentaire. Ces éléments ne sont pas transmis directement du site initial aux autres sites régionaux, ils sont transmis du site

4.4. Messages de mise en cohérence

Toute action répartie nécessite des transferts entre les actions réparties. Un message est transmis ; il comporte une entête précisant sur quelle réalisation-réseau porte l'action, de quelle action répartie il s'agit, et d'autres renseignements de "service" ; il comporte aussi des "données" destinées aux actions élémentaires (excepté pour les actions de suppression).

La composition exacte de ces messages sera précisée au paragraphe 3.2. du chapitre V.

4.5. Fichier "émission" et fichier "réception" (FE et FR)

Au cours d'une session de travail, chaque site, pour chaque action répartie, "dépose" un message (cf. 4.4.) dans un fichier "émission" ; tous ces messages sont destinés au site symétrique sur l'axe pour y créer une action élémentaire. Le site symétrique possède également son fichier émission dans lequel il dépose ses messages. Une action élémentaire qui s'exécute peut également donner lieu à un message déposé dans le fichier émission (F.E.).

Inversement, chaque site reçoit des messages, du site symétrique, dans un fichier réception (F.R.). Chaque message retiré de FR crée une action élémentaire locale.

Ces fichiers FE et FR sont exclusivement gérés par le logiciel de cohérence (cf. V.3.).

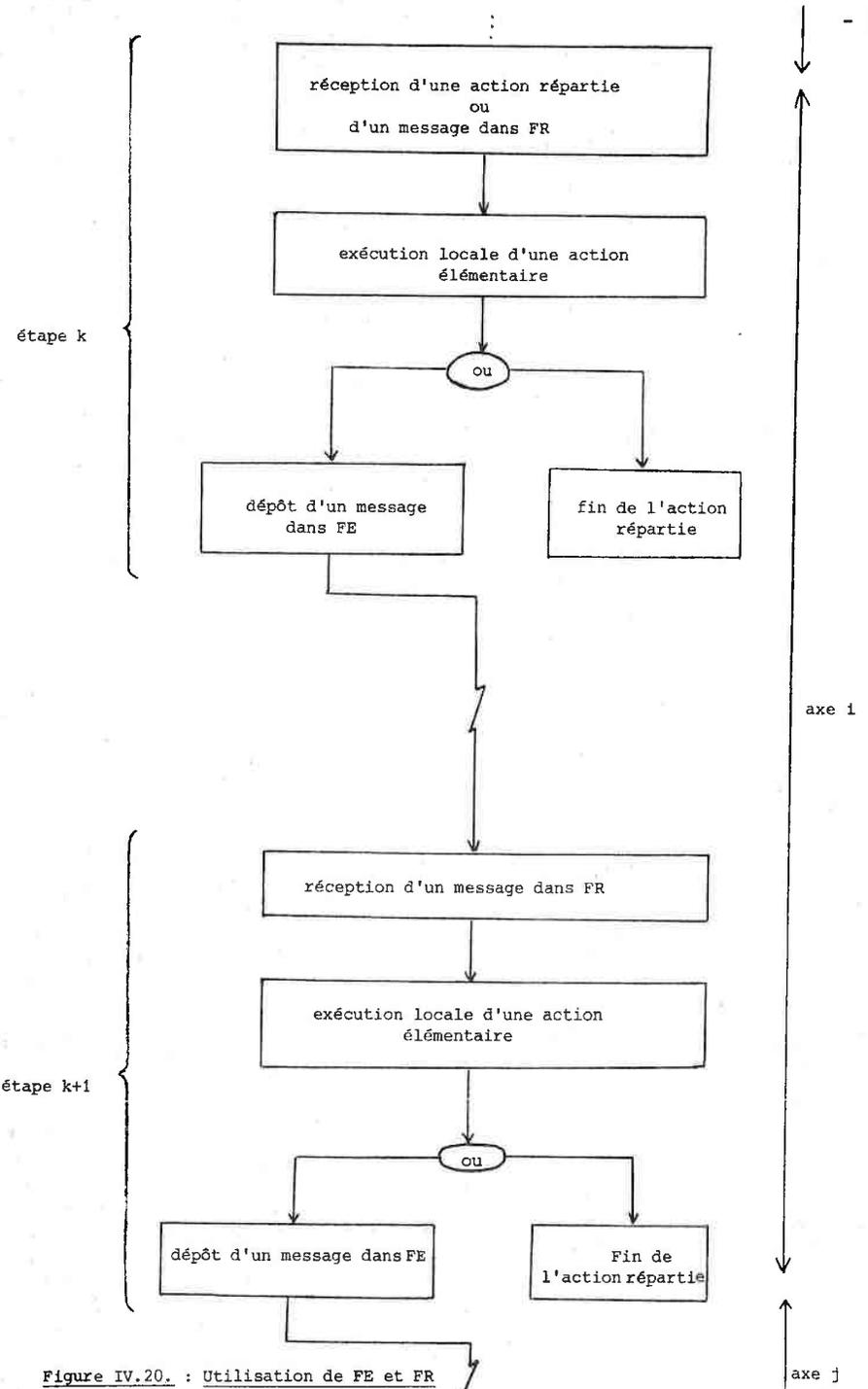


Figure IV.20. : Utilisation de FE et FR

Tout site régional possède donc un fichier émission et un fichier réception. Le site central possède un fichier réception commun à tous les axes, et un fichier émission pour chaque axe.

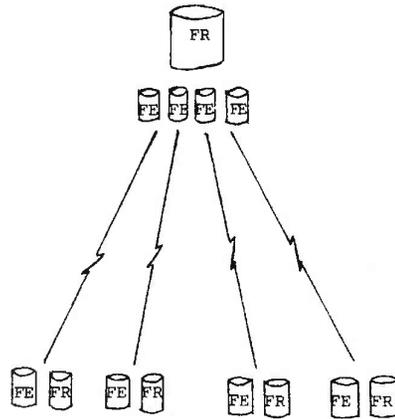


Figure IV.21. : Fichiers émission et réception

Au "démarrage" d'une session de travail tous les fichiers FE et FR doivent être "vides" : cela signifie qu'il n'y a plus d'action répartie en cours, toutes sont exécutées.

4.6. Sessions de mise en cohérence globale

4.6.1. Définition

Pour permettre à chaque session de travail de débiter sur un état cohérent de la base répartie il est nécessaire de la faire précéder d'une session préalable de mise en cohérence globale.

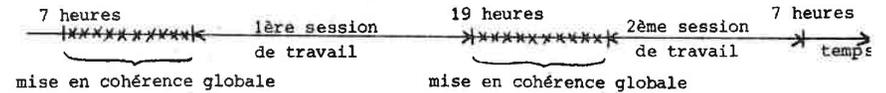


Figure IV.22. : Sessions de mise en cohérence globale

Au cours d'une telle session toutes les actions réparties survenues au cours de la session de travail précédente et dont la première étape a été exécutée sur le site de réception sont poursuivies et terminées.

La durée d'une session ne peut pas être fixée à priori ; elle dépend, en fait, du nombre d'actions réparties en suspens et du temps nécessaire, sur chaque site, pour exécuter toutes les actions élémentaires.

4.6.2. Début d'une session

Tous les sites doivent obligatoirement participer à chaque session de mise en cohérence globale ; tout site peut

à priori être concerné par une action répartie survenue sur un autre site et sa connexion est donc nécessaire pour que l'action puisse être entièrement exécutée.

L'heure approximative de début de chaque session de mise en cohérence doit être négociée, une fois pour toutes, entre les différents administrateurs locaux.

A partir de cette heure, chaque site régional doit établir un "contact-réseau" avec le site central, et, réciproquement le site central doit établir un contact avec chaque site régional. Le contact s'établit axe par axe mais à peu près simultanément.

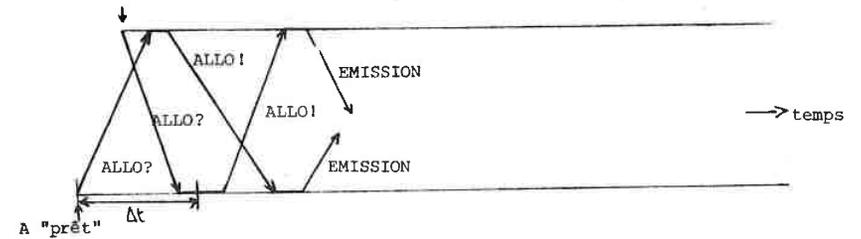
Le principe du contact-réseau entre les deux sites A et B d'un axe est le suivant.

A partir de l'heure de début convenue, le premier des deux sites qui est "prêt", A, adresse à l'autre le message "ALLO ?".

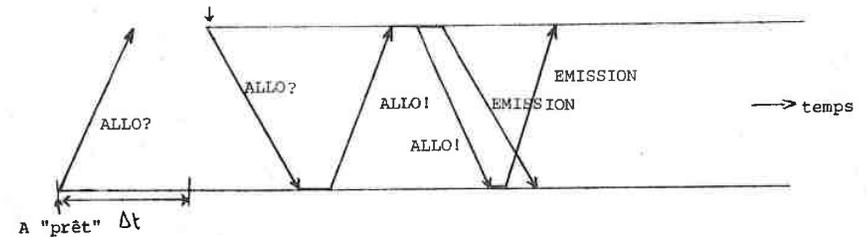
- Si A reçoit un message "ALLO ?" au cours de Δt cela signifie que A et B sont prêts pratiquement simultanément. A attend la réponse "ALLO !" de B, et inversement. Le message "ALLO ?" constitue la réponse au message "ALLO !". Le site ayant émis "ALLO ?" et ayant reçu "ALLO !" sait que la connexion est établie dans les deux sens et peut donc émettre ses messages. (cf. figure IV. 23 a)

- Si A ne reçoit ni un message "ALLO ?", ni un message "ALLO !" au cours de Δt , cela signifie soit que la liaison A-B est impossible, soit que le site B n'est pas prêt. Alors A reste en attente de réception d'un message de la part de B. Lorsque A reçoit, après Δt ,

le message "ALLO ?" alors il renvoie le message "ALLO !" ; le site B, recevant ce message, sait alors que la communication fonctionne dans les deux sens ; il envoie le message "ALLO !" à A ; ceci permet à A de constater que B a bien reçu le message "ALLO !".



a. A et B "prêts" presque simultanément



b. A "prêt" avant "B"

Figure IV.23. : Début d'une session synchronisée sur un axe

4.6.3. Transferts et exécutions locales

Lorsque le contact-réseau est établi un site envoie à son symétrique le contenu de son fichier FE. Il y a donc, sur chaque axe, transferts symétriques de messages : les fichiers FE se "vident", tandis que les fichiers FR se "remplissent".

Chaque site reçoit dans FR des messages qui donnent lieu (cf. 4.4.) à des actions élémentaires locales qui doivent être exécutées pendant la session en cours. Une telle action élémentaire peut, elle-même donner lieu à un message stocké dans FE ; celui-ci doit obligatoirement être transmis au site symétrique : cela correspond au cas où une action répartie est composée de plus de deux étapes ; elle nécessite alors plus d'un seul transfert et d'une seule exécution élémentaire au cours de la même session de mise en cohérence globale.

Sur la figure qui suit nous considérons l'évolution d'une action répartie à 4 étapes successives (E1, E2, E3, E4).

Sur cet exemple, nous avons supposé qu'une étape ne concernait qu'un seul site à la fois. Sur la figure suivante nous considérons l'exemple d'une action élémentaire dont une étape est parallèle sur trois sites régionaux : ce peut être le cas d'une modification de DRT.

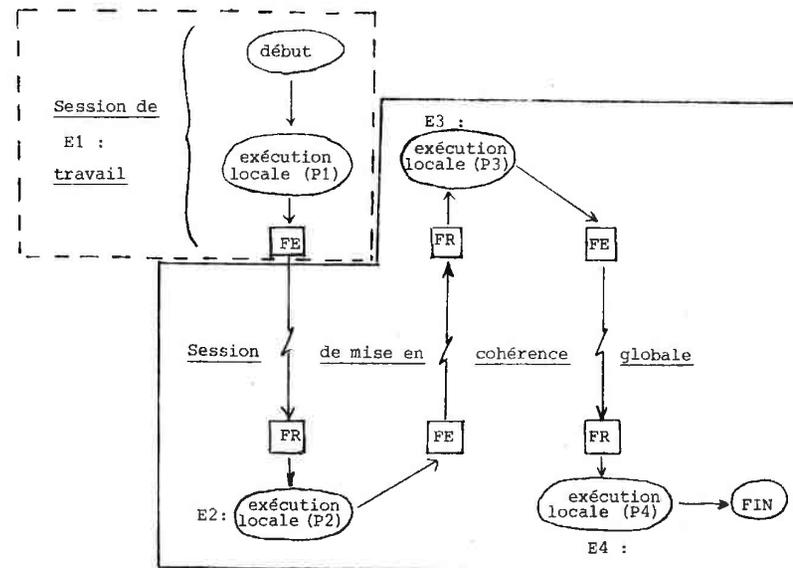


Figure IV.24. : Evolution d'une action répartie à 4 étapes au cours d'une session de mise en cohérence globale

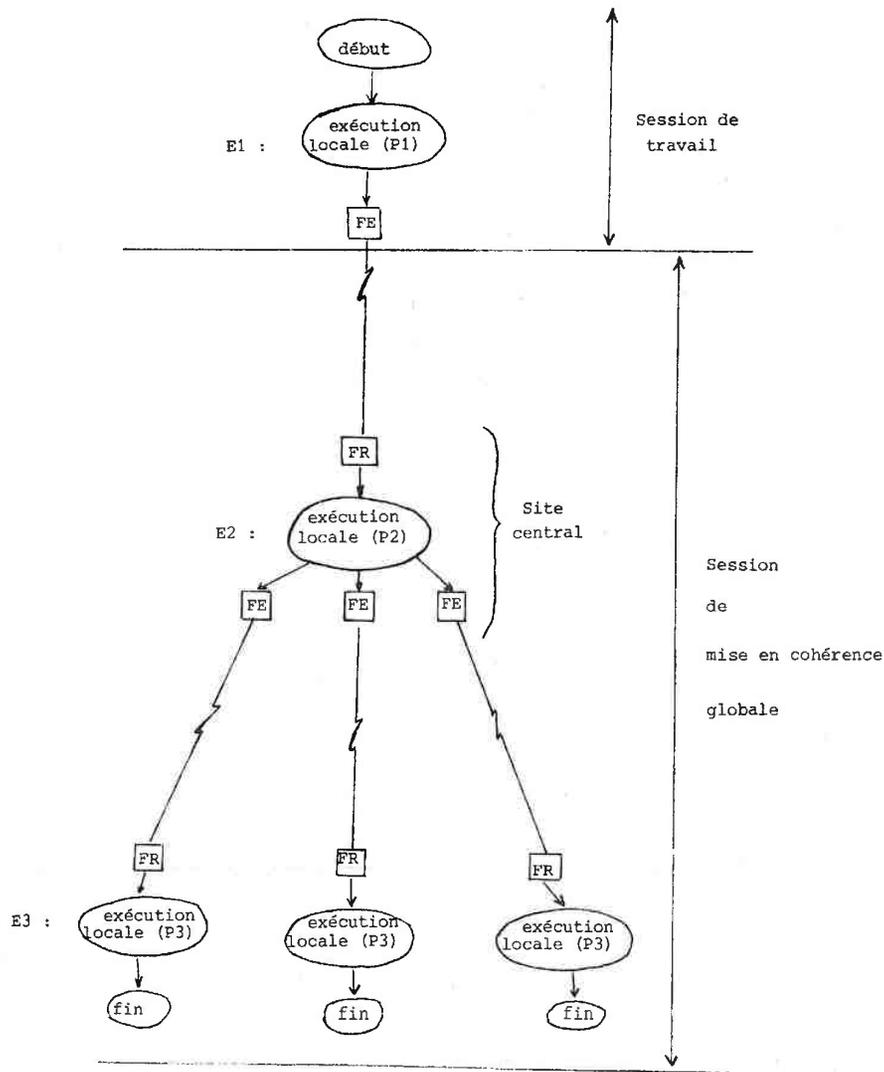


Figure IV.25. : Evolution d'une action répartie au cours d'une session de mise en cohérence globale

Si on considère un site quelconque au cours d'une session de mise en cohérence globale, il va donc recevoir, émettre des messages, et exécuter des actions élémentaires.

4.6.4. Fin d'une session

Lorsqu'un site régional possède ses fichiers FE et FR "vides" et qu'il a terminé toutes ses exécutions d'actions élémentaires alors il envoie, au site central, le message "FIN" suivi de son numéro de site ; il est alors au repos.

Si le site central et d'autres sites régionaux ne sont pas au repos, alors le site régional au repos peut encore recevoir dans FR des messages et avoir des exécutions à entreprendre ; il est alors réactivé ; il devra alors renvoyer un nouveau message "FIN" (cf. figure IV.26) lorsqu'il aura terminé.

Si le site central a reçu le message "FIN" d'un site régional mais qu'il lui envoie des messages ultérieurs, alors le site régional est considéré comme actif et le central annule le message "FIN" reçu et attend un nouveau message "FIN" ultérieur.

Lorsque le site central a reçu un message "FIN" de tous les sites régionaux (ce qui signifie qu'ils sont tous au repos) et que tous ses fichiers FE et que son fichier FR sont vides et qu'il n'a plus d'action élémentaire en cours d'exécution, alors il envoie à chaque site régional un message "Ø K-FIN" qui signifie la fin globale de la session de mise en cohérence globale.

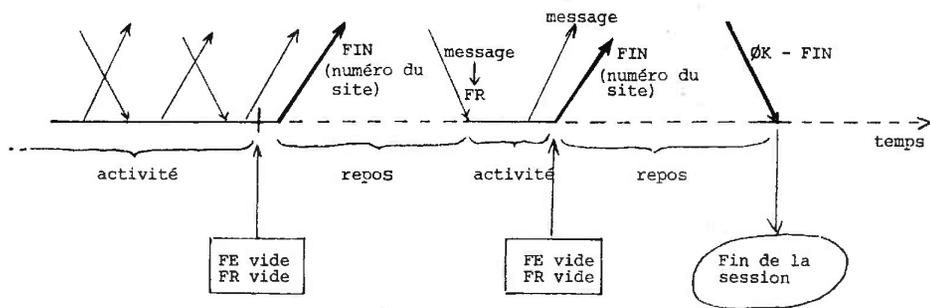


Figure IV.26. : Fin d'une session de mise en cohérence globale sur un site régional

4.7. Concurrence entre actions réparties

Tout site pouvant émettre des actions réparties, il est possible que des actions émises sur des sites différents soient "concurrentes" et provoquent des incohérences globales sur la base répartie. Ces concurrences apparaissent lors des actions de modification. Nous distinguons deux types de modification, le type étant lié à la donnée modifiée : la modification relative et la modification absolue. Mais elles peuvent également apparaître à la suite de créations régionales.

4.7.1. Modification relative

Une donnée subit une modification relative si la modification consiste à exécuter une opération arithmétique sur la valeur initiale : la valeur nouvelle tient compte de la valeur initiale.

Une telle donnée peut être, par exemple, un chiffre d'affaire d'un certain fournisseur. Ce chiffre d'affaire peut être modifié par une addition ou une soustraction. Dans le cas où le chiffre d'affaire est dupliqué dans plusieurs bases et que plusieurs sites peuvent le modifier il est important qu'à la fin d'une session de mise en cohérence globale, le chiffre d'affaire ait subi toutes les modifications et que la valeur soit la même dans toutes les bases.

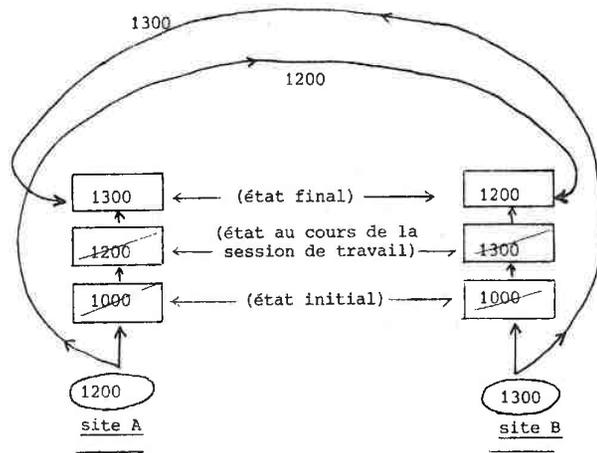
Dans la situation initiale de l'application pour laquelle les données sont répertoriées dans une base centralisée unique une telle modification s'effectue de la manière suivante :

- l'utilisateur lit la valeur : 1000
- l'utilisateur introduit la nouvelle valeur : 1200

En fait la modification effectuée sur la base est le remplacement de la valeur 1000 par la valeur 1200 ; la modification effectuée par l'utilisateur étant l'addition $1000 + 200$.

Si l'on utilise le même principe pour le modèle étoilé et que l'on suppose le chiffre d'affaire répertorié dans deux bases régionales alors on aboutit à une incohérence globale si les deux sites modifient chacun la valeur.

Considérons qu'un site désire augmenter le chiffre d'affaire de 300, l'autre de 200 ; l'état initial étant 1000, l'état final devrait être $1000 + 200 + 300 = 1500$.

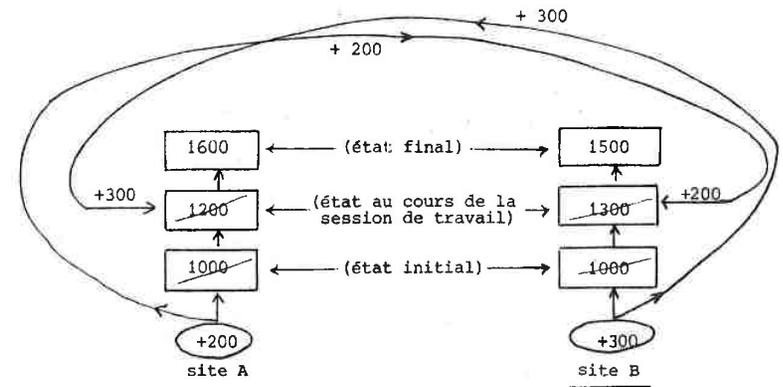


Au cours de la session de travail, chaque site introduit sa nouvelle valeur par une action élémentaire qui consiste à remplacer la valeur initiale par 1200 sur le site A, par 1300 sur le site B.

Au cours de la session de mise en cohérence globale, le site A exécute l'action élémentaire qui consiste à remplacer la valeur initiale 1200 par 1300 qui est la valeur introduite en B ; inversement en B, 1300 est remplacée par 1200. Il en résulte, qu'au lieu de s'ajouter, les modifications se sont "croisées". L'incohérence résulte du fait que la modification élémentaire s'exécute par un remplacement de valeur.

En fait, pour préserver la cohérence globale il est nécessaire que le mécanisme ne transmette pas le résultat local de la modification mais l'opération ainsi que l'opérande de l'opération arithmétique. L'action élémentaire consiste alors

à exécuter l'opération arithmétique à partir de la valeur contenue dans la base locale du site. Deux modifications "concurrentes" peuvent alors se compléter et non se croiser.



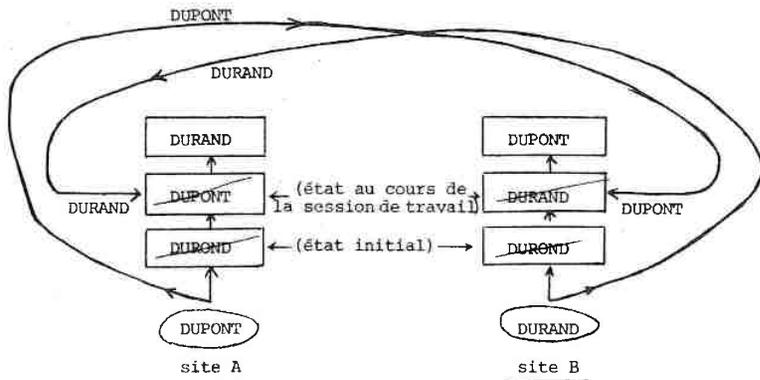
4.7.2. Modification absolue

Une donnée subit une modification absolue si la modification consiste à remplacer une valeur initiale par une nouvelle valeur.

C'est le cas d'une chaîne de caractère comportant une erreur ; on remplace la chaîne par une autre chaîne ne comportant pas d'erreur.

Si elles ne sont pas identiques, deux modifications absolues d'une même donnée, émises sur des sites différents au cours d'une même session de travail, soulèvent une incohérence globale entre les "copies" de la donnée.

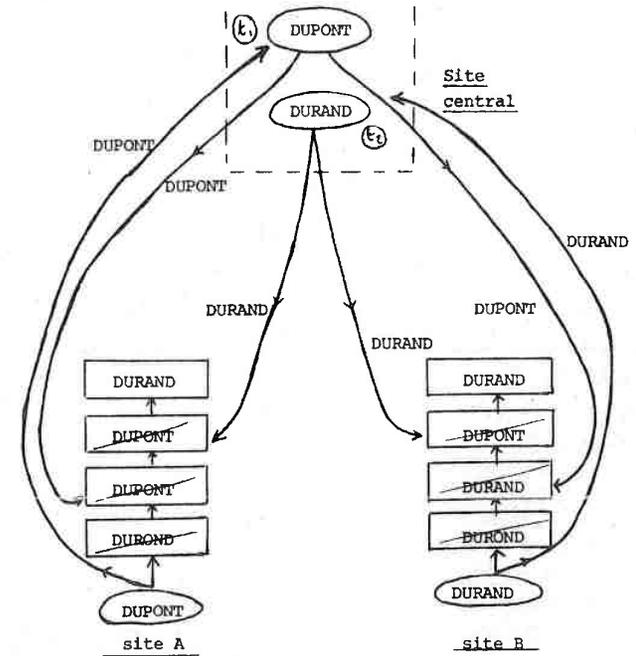
Supposons que deux sites possèdent la chaîne DUROND et que le site A propose la modification DUPONT alors que le site B propose la modification DURAND.



Au cours de la session de mise en cohérence globale chaque site remplace la valeur acquise au cours de sa session de travail par la valeur de l'autre site. Les valeurs sont donc échangées et ne sont pas identiques : il y a donc une incohérence globale.

Pour éviter l'obtention d'une telle incohérence, une solution consiste à ordonnancer l'exécution des modifications sur les différents sites. Cet ordonnancement est régi par le site central.

Ainsi, lors d'une session de travail chaque site exécute localement "sa" modification. Le site A remplace "DUROND" par "DUPONT", le site B remplace "DUROND" par "DURAND". Pendant toute la session de travail, il y a donc une incohérence globale sur la donnée initiale "DUROND". Lors de la session de mise en cohérence globale, le site central reçoit les deux modifications dans un certain ordre et les considère dans le même ordre. A l'instant t_1 , il reçoit la modification émise par le site A ; il la transmet à tous les sites possédant une copie : c'est-à-dire, sur notre exemple, à A et à B ; A, qui est le site émetteur reçoit à nouveau la modification. Celle-ci est alors exécutée sur chacune des deux copies. Il en résulte que



la base est globalement cohérente avec la donnée "DUPONT". Puis, le site central reçoit la modification émise par B. De même, celle-ci est transmise à chacun des deux sites et y est exécutée. Globalement, à la fin de la session, la base est cohérente, la donnée ayant la valeur "DURAND" qui est le résultat de la dernière modification parvenue au site central.

Grâce à cet ordonnancement les modifications ne se "croisent" plus sur le réseau. Cependant sur chaque site qui émet une telle modification, celle-ci est exécutée deux fois : une fois au cours de la session de travail, une fois au cours de la session de mise en cohérence globale. A la fin de cette session, la cohérence globale est obtenue du fait que tous les exemplaires d'une donnée, modifiée en absolu, sont identiques.

Sur le plan "informatique" cette solution est satisfaisante : en effet, elle apporte, à la fin de la session, la cohérence globale et résout le problème de la concurrence de plusieurs modifications absolues "simultanées" d'une donnée dupliquée. Cependant, l'action émise par le site A n'apparaît pas dans la base répartie.

En effet, seule la dernière modification (celle du site B) parvenue au site central est exprimée dans la base répartie. Globalement "DUROND" a été remplacé par "DURAND", mais la modification "DUPONT" émise par A n'apparaît nulle part. Il est donc important que le site A soit avisé de l'écrasement de sa modification par une modification concurrente.

Le site central doit donc mémoriser toutes les modifications reçues et exécutées sur une même donnée au cours d'une session de mise en cohérence. Tous les sites ayant émis une telle modification, doivent recevoir du site central des informations sur les différentes modifications absolues qui ont été exécutées au cours d'une session de travail (cf. V.3.4.2.).

Les solutions que nous venons de présenter dans ces deux paragraphes sont valables pour les trois types de données dupliquées : DRR, DCR, DRT.

4.7.3. Créations régionales (cf. V.3.4.2.)

Une incohérence globale peut provenir d'une concurrence entre plusieurs créations portant sur une même réalisation-réseau. Supposons qu'une réalisation-réseau soit répertoriée sur deux axes et que le site régional d'un troisième axe crée, sur son site, une réalisation locale appartenant à la même réalisation-réseau. Les données introduites peuvent être contradictoires avec celles déjà existantes.

Il est donc nécessaire d'effectuer des comparaisons entre les données introduites et celles déjà stockées dans la base répartie en supposant que ces dernières sont les données correctes. Dans le cas où il y a des contradictions, les données introduites sont remplacées par les anciennes.

La comparaison doit être effectuée pour les données des types DRT et DRR :

- la comparaison des DRT est faite sur le site central
- la comparaison des DRR est faite sur un site régional "possédant" la réalisation-réseau. (cf. V.3.5.)

La comparaison est inutile pour les DCR puisque celles-ci sont propres à un axe.

CHAPITRE V

LE LOGICIEL DE RÉPARTITION ET DE COHÉRENCE (L.R.C.)

1. DEFINITION
2. ENVIRONNEMENT ET ARCHITECTURE
EXTERNE
3. FONCTIONS DU L.R.C.
4. LA TABLE DE REPARTITION
5. ARCHITECTURE INTERNE DU L.R.C.

Dans ce chapitre nous définissons le "logiciel de répartition et de cohérence" et décrivons son fonctionnement.

Chaque site, possédant une base locale, dispose d'un tel logiciel. L'ensemble des logiciels constitue un interface réparti entre les différentes bases locales. Cet interface exécute, au cours des sessions de mise en cohérence définies dans le chapitre précédent, toutes les fonctions liées à la répartition et à la cohérence globale.

Après une première partie consacrée à la définition du logiciel de répartition et de cohérence, une seconde partie est consacrée à la description de son environnement et de son architecture externe. Nous précisons la rôle de ce logiciel dans l'exécution des mises à jour de la base répartie et son rôle d'interface entre une base locale et le reste de la base répartie. Enfin, nous précisons sa position par rapport aux autres logiciels mis en œuvre par chaque site.

La partie suivante est consacrée aux différentes fonctions exécutées par le logiciel.

Le logiciel de répartition et de cohérence est conçu selon le modèle de répartition étoilé et doit fonctionner indépendamment des données contenues dans la base répartie ; à cet effet, le logiciel exécute ses fonctions sur des données "anonymes" dont la description figure dans une table de répartition. Une partie du chapitre est consacrée à cette table.

Enfin, nous terminons le chapitre avec la description de l'architecture interne du logiciel ; celle-ci se compose de différents modules dont nous donnons le principe de fonctionnement.

1. DEFINITION

1.1. Logiciel de répartition et logiciel de cohérence

Les deux chapitres précédents nous ont permis de définir respectivement le logiciel de répartition et le logiciel de cohérence.

Le logiciel de répartition (cf. III.3.2.4.) existant sur chaque site, dispose d'une table de répartition exprimant le mode de répartition de toutes les caractéristiques de la base locale.

Le logiciel de cohérence (cf. IV.1.3.), également sur chaque site, est chargé de faire exécuter un mécanisme permettant d'assurer la cohérence globale de la base répartie. Les deux logiciels sont, en fait, utilisés pour l'exécution des opérations de mise à jour, ces opérations étant associées à la fois à la répartition des données et à la cohérence de la base répartie. Pour cette raison, nous ne considérerons pas l'existence sur chaque site de deux logiciels séparés mais d'un logiciel unique chargé d'assurer à la fois la répartition des données et la cohérence de la base répartie : le logiciel de Répartition et de Cohérence (L.R.C.).

1.2. Contexte local et contexte global

Chaque site disposant d'une base locale dispose d'un logiciel de gestion de cette base (SGBD). Ce logiciel suffit à assurer le contexte local de l'application. Les programmes d'application locaux manipulent la base locale par l'intermédiaire du système de gestion.

L'existence d'une base répartie engendre un contexte global qui associe entre eux divers contextes locaux ; chaque

contexte local fait partie du contexte global. Le contexte global utilise un outil de communications constitué par le réseau et les logiciels d'accès au réseau.

Le LRC permet, sur chaque site, d'intégrer la base locale au contexte global et d'exécuter des actions réparties (cf. IV.2.6.) c'est-à-dire des actions qui agissent sur plusieurs bases.

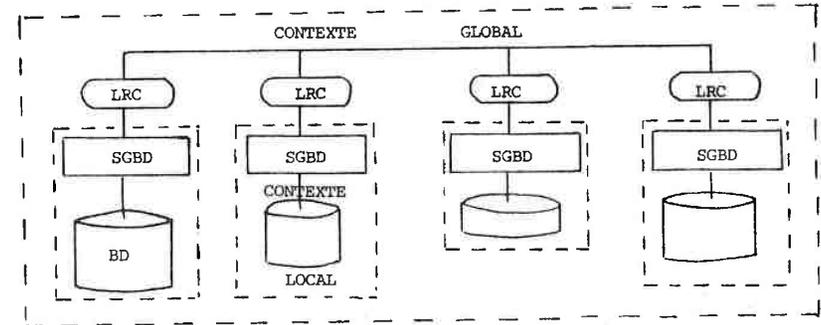


Figure V.1. : Contextes locaux et contexte global

1.3. Interface étoilé

Dans le cadre de notre application, l'ensemble des bases locales étant réparti selon un modèle étoilé, les L.R.C. sont donc répartis selon le même modèle ; il faut distinguer un L.R.C. central (L.R.C.C.) et des L.R.C. régionaux (L.R.C.R.). L'ensemble des L.R.C. constitue pour les contextes locaux un interface étoilé et assure un contexte global étoilé.

Les communications se font selon le modèle proposé c'est-à-dire qu'un L.R.C. régional communique directement et uniquement avec le L.R.C. central, la communication se faisant selon un protocole d'échange LRC-LRC (cf. 3.2.).

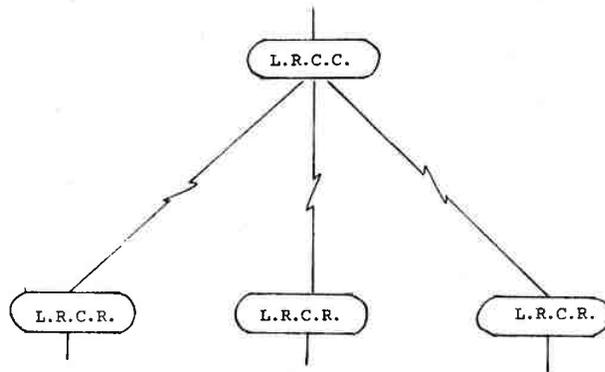


Figure V.2. : Interface étoilé

2. ENVIRONNEMENT ET ARCHITECTURE EXTERNE

2.1. Actions réparties et L.R.C.

Dans le chapitre précédent nous avons défini des actions réparties qui sont en fait, au niveau du contexte global, des opérations de mise à jour de la base répartie.

Une action répartie est au départ, lorsqu'elle est ordonnée par un utilisateur, une action élémentaire exécutée par un programme de type P1 (cf. IV.2.5.2.) au cours d'une session de travail sur une base locale ; cette action élémentaire s'exprime sur la base locale soit par une création de réalisation, soit par une suppression de réalisation, soit par la modification de une ou plusieurs caractéristiques d'une réalisation.

Pour l'utilisateur l'action s'exécute dans le contexte local et il est étranger à toute notion de répartition. Interrogations et mises à jour s'effectuent, pour l'utilisateur, dans le contexte local.

Cependant, la plupart des actions élémentaires de mise à jour ne constituent qu'une première étape d'un ensemble d'actions élémentaires qui globalement forment une action répartie, celle-ci s'exécutant sur la base répartie. Ainsi, si un utilisateur effectue sur son site la création d'un fournisseur, cette action va se poursuivre par une ou plusieurs autres actions sur d'autres sites.

Le L.R.C. a pour tâche de faire exécuter ces actions qui suivent celle ordonnée par l'utilisateur. Elles sont exécutées par des programmes P_i (i > 1) à l'initiative de l'interface constitué par les L.R.C. et ce pendant une session de mise en cohérence globale sans que l'utilisateur initial n'en soit avisé.

2.2. Interface contexte local → contextes locaux distants

Le L.R.C. d'un site est chargé de donner suite à toute action élémentaire de mise à jour qui a été exécutée dans la base locale. Il recupère toutes les mises à jour pour les faire suivre, éventuellement, par d'autres actions élémentaires sur d'autres sites.

Cette récupération s'effectue au cours de la session de mise en cohérence globale. Pour chaque mise à jour récupérée dans la base locale, le L.R.C. forme un message (cf. IV.4.4.) qu'il dépose dans FE (cf. IV.4.5.). Tous les messages déposés dans FE sont ensuite transmis à d'autres L.R.C.

Sur un site, le L.R.C. permet donc d'interfacer le contexte local avec tous les contextes locaux distants.

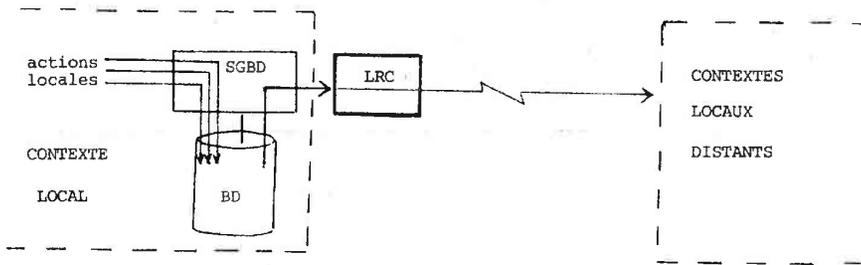


Figure V.3. : Interface contexte local → contextes locaux distants

Les mises à jour que le L.R.C. récupère dans la base locale résultent soit de programmes P1 exécutés au cours de la session de travail qui précède la session de mise en cohérence, soit par des programmes P_i (i > 1) qui s'exécutent au cours de la session de mise en cohérence.

2.3. Interface contextes locaux distants → contexte local

Dans le paragraphe précédent nous avons vu que le L.R.C. reprenait toute action sur la base locale pour la faire poursuivre en action répartie.

Inversement, le L.R.C. d'un site reçoit dans FR, au cours d'une session de mise en cohérence globale, des messages destinés à créer, sur le site, des actions élémentaires qui compléteront des actions déjà entreprises ailleurs ; le L.R.C.

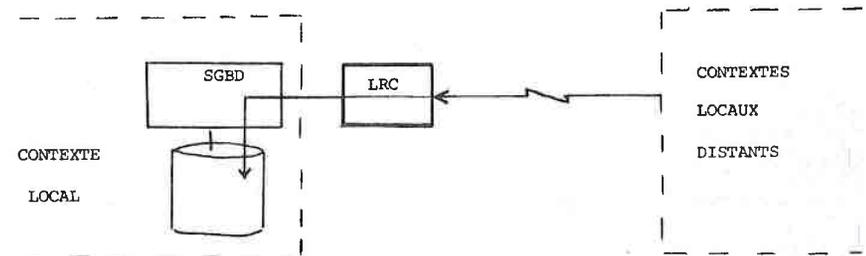


Figure V.4. : Interface contextes locaux distants → contexte local

retire ces messages un à un de FR et, pour chacun, fait exécuter un programme de manipulation de la base ; ce programme (de type P2, P3) exécute une action élémentaire qui correspond au message reçu. Cette action peut ne pas être la dernière phase

de l'action répartie et peut donc donner lieu à la création d'un nouveau message qui sera déposé dans FE.

2.4. Architecture externe du L.R.C.

Le L.R.C. est un logiciel utilisé, uniquement, pendant les sessions de mise en cohérence globale ; ces sessions étant "synchronisées" (cf. IV.4.6.) tous les L.R.C. de l'application doivent être actifs.

Au cours d'une telle session chaque site a donc en machine :

- le logiciel réseau, ce qui permet au L.R.C. d'expédier et de recevoir des messages donc de communiquer avec le ou les L.R.C. distants.
- le L.R.C.
- le système de gestion de la base de données locale qui permet au L.R.C. d'accéder aux données de la base ; à ce système de gestion sont associés tous les logiciels précompilés qui permettent aux utilisateurs et au L.R.C. de manipuler les données.

Pendant une session de mise en cohérence globale, aucun utilisateur ne peut travailler sur la base, seul le L.R.C. y a accès. Le travail s'effectue en mode "par lots", le L.R.C. et les différents modules du système de gestion étant regroupés en une même édition de liens.

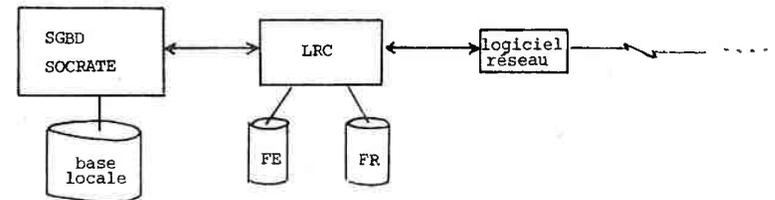


Figure V.5. : Architecture externe du L.R.C.

3. FONCTIONS DU L.R.C.

3.1. Récupération des mises à jour locales

La récupération par le L.R.C. des mises à jour se fait de manière indirecte par l'intermédiaire de la base elle-même. En effet, les programmes qui effectuent des mises à jour, par "lots" ou conversationnelles, les exécutent directement dans la base ; pour récupérer les données mises à jour, le L.R.C. parcourt la base de son site à l'aide de renseignements laissés par les programmes Pi.

3.1.1. Les renseignements de récupération

Au fur et à mesure de leur exécution les programmes Pi déposent des renseignements dans la base locale.

Ces renseignements permettent au L.R.C.

- de localiser et de reconnaître les données qui ont subi localement soit une modification, soit une création, soit une suppression.
- de connaître le type de l'action répartie dans laquelle doit être intégrée l'action locale repérée et donc de savoir quelle suite il doit donner à cette action locale.
- de savoir sur quel(s) axe(s) il doit "prolonger" l'action repérée, dans le cas du L.R.C. central.

3.1.1.1. Définition de l'entité MAJ

Pour le dépôt de ces renseignements, on définit dans chaque base locale une entité "MAJ" (mise à jour) de structure logique donnée.

Un programme Pi de mise à jour génère une réalisation de l'entité MAJ pour chaque réalisation d'entité locale qu'il met à jour.

Dans le cas où une entité possède des caractéristiques du type entité (cf. I.2.3.) le programme de mise à jour doit les considérer comme des entités à part entière et génère une réalisation d'entité MAJ pour chaque réalisation d'une telle caractéristique qu'il met à jour.

3.1.1.2. Structure de l'entité MAJ régionale

Les L.R.C. régionaux disposent tous d'une même structure d'entité MAJ ; cette structure est la suivante :

```
entité (n) MAJ
début
  TYPMAJ (3 1) ( 1 2 3)
  NUMENTITE de 1 à m
  NUMOBJET  de 1 à u
  OBJETPERE de 1 à u
entité (v) CARAC
  début
    NUMCARAC de 1 à w
    OPERATEUR (4 1) (+ - 0 1)
    OPERANDE de 1 à 999999999
  fin
fin
```

n : Nombre maximum de réalisations de l'entité MAJ générées localement (correspond à un nombre maximum de réalisations d'entité ou de caractéristiques-entité mises à jour).

m : Nombre maximum d'entités et de caractéristiques du type entité figurant dans la description locale. Les entités et

caractéristiques possèdent un numéro de 1 à m ; ce numéro est global c'est-à-dire qu'il est identique sur tous les sites pour une même entité (cf. 4).

u : Chaque réalisation-réseau possède un numéro (cf. III.3.2.5.) attribué automatiquement par le site central ; u représente un numéro maximum.

v : Nombre maximum de réalisations de caractéristiques mises à jour.

w : Chaque caractéristique possède un numéro figurant dans la table de répartition ; w représente un numéro maximum.

TYPMAJ : Contient le type de mise à jour effectuée :

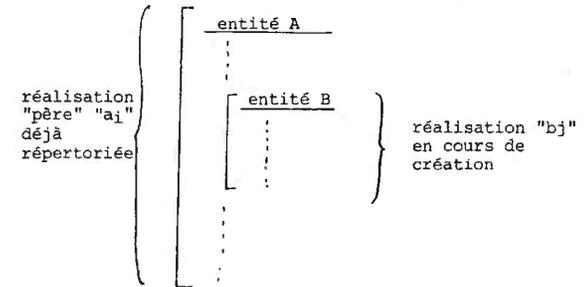
- "1" : indique une suppression de réalisation.
- "2" : indique une création régionale (principale ou secondaire).
- "3" : indique une modification de réalisation-réseau.

NUMENTITE : Contient le numéro de l'entité concernée par la mise à jour : fournisseur, marché, commande ou facture ? (cf. 4).

NUMOBJET : Contient le numéro attribué à la réalisation-réseau (cf. III.3.2.5.). La caractéristique qui contient ce numéro est connue dans la table de répartition et dans le programme de mise à jour. Dans le cas d'une création régionale, le numéro est un numéro temporaire remplacé ultérieurement par le numéro attribué par le site central. Ce numéro permet au L.R.C. de retrouver dans la base la réalisation concernée par la mise à jour.

OBJETPERE : Utilisé pour les réalisations de caractéristiques de type entité lors des créations. La réalisation n'est pas encore répertoriée dans la base et il faut savoir à quelle

réalisation d'objet de niveau supérieur il faut la rattacher. OBJETPERE contient le numéro (cf. III.3.2.5.) attribué à la réalisation de niveau supérieur.



CARAC : Caractéristique-entité de l'entité MAJ dont chaque réalisation représente une caractéristique dont le contenu est modifié par une création ou une modification.

NUMCARAC : Contient le numéro d'une caractéristique modifiée ; chaque caractéristique possède un numéro qui est connu dans les programmes de mise à jour et qui figure également dans les tables de répartition. A l'aide de cette caractéristique, le L.R.C. peut savoir quelles sont les caractéristiques de la base qui ont effectivement été modifiées.

OPERATEUR : Cette caractéristique est utilisée dans deux cas :

- 1) Dans le cas d'une modification relative (cf. IV.4.7.1.) elle contient l'opérateur de modification qui doit être transmis par le L.R.C.
- 2) Pour les chaînes de bits inverses des types DCR, DRT ou DRR (cf. III.3.2.3.).

Un numéro est attribué à toute chaîne de bits qui inverse une entité et ce numéro figure, comme numéro de caractéristique de cette entité, dans la table de répartition (cf. 4.2.). Dans le cas où on modifie un bit d'une chaîne alors :

- le numéro attribué à la chaîne est dans NUMCARAC (comme si la chaîne était une caractéristique de l'entité dont le numéro figure dans NUMENTITE).
- la réalisation-réseau référencée par le bit modifié est indiquée dans NUMOBJET.
- si la réalisation-réseau est inversée alors OPERATEUR contient "1", sinon contient "0".

OPERANDE : Cette caractéristique est utilisée pour :

- Une modification relative : elle contient l'opérande de la modification. Le type "valeur numérique" de OPERANDE est choisi suffisamment grand pour pouvoir contenir une grande valeur.
- Les caractéristiques du type "référence" ; le programme de mise à jour qui insère une réalisation-réseau dans l'anneau d'une autre réalisation dépose dans OPERANDE le numéro de cette dernière réalisation c'est-à-dire celle qui est référencée.

3.1.1.3. Structure de l'entité MAJ centrale

Par rapport à la structure régionale, la structure centrale doit posséder des caractéristiques supplémentaires qui permettent au L.R.C. de savoir sur quel(s) axe(s) il doit prolonger l'action locale effectuée. Cette structure est la suivante :

```
entité (n) MAJ
début
  TYPMAJ (5 1) (3 4 5 6 7)
  NUMENTITE de 1 à m
  NUMOBJET de 1 à u
  OBJETPERE de 1 à u
entité (v) CARAC
début
  NUMCARAC de 1 à w
  AXE de 1 à i
  OPERATEUR (4 1) (+ - 0 1)
  OPERANDE de 1 à 999999999
fin
entité (i) NUMAXE
debut
  NUMERO de 1 à i
fin
fin
```

i : Représente le nombre d'axes.

TYPMAJ : Les types de mises à jour possibles sont les suivants :

- "3" : modification d'une réalisation-réseau
- "4" : création principale d'une réalisation-réseau
- "5" : création secondaire d'une réalisation-réseau
- "6" : suppression totale d'une réalisation-réseau
- "7" : suppression partielle d'une réalisation-réseau

AXE : Contient le numéro de l'axe auquel appartient la donnée modifiée. Certaines caractéristiques possèdent une valeur par axe : c'est le cas des DCR et de certaines DCP (cf.III. 3.2.2.). NUMCARAC indique le numéro de la caractéristique, mais dans ces cas le programme doit préciser, en plus, de quelle réalisation (qui correspond à un des "i" axes) il s'agit.

NUMAXE : Le programme doit préciser, dans le cas des créations et des suppressions, à quel(s) axe(s) s'adresse l'action. NUMAXE contient un nombre maximum de réalisations qui correspond au nombre d'axes.

NUMERO : Contient le numéro d'un axe concerné par l'action de mise à jour entreprise.

3.1.2. Les programmes de mises à jour

Les programmes Pi qui exécutent les mises à jour sont chargés de générer et de valoriser des réalisations de l'entité MAJ.

Le principe et la structure de l'entité MAJ imposent une certaine structure à tout programme de mise à jour. En effet, l'entité MAJ est définie pour la mise à jour d'une réalisation d'entité et pour chaque réalisation considère les caractéristiques une à une. Ceci impose aux programmes de procéder par itération d'une part sur les réalisations d'entité et d'autre part, pour chaque réalisation, sur les caractéristiques ; celles-ci sont considérées une à une, les caractéristiques de type entité étant traitées en dernier. Pour une telle caractéristique, le principe est le même : pour chaque réalisation on traite l'une après l'autre les caractéristiques du bloc.

Remarquons que dans le cas des suppressions l'itération ne se fait qu'au niveau des réalisations d'entité ; les caractéristiques ne sont pas considérées puisqu'une suppression affecte une réalisation d'entité entière.

Pour générer et valoriser les réalisations de l'entité MAJ des macro-requêtes SOCRATE sont définies et utilisées par les programmes de mise à jour. Ces "macros" sont de trois types chacun correspondant à un niveau spécifique d'utilisation à l'intérieur d'un programme.

1) Macro-requête GEN-MAJ :

Pour chaque réalisation d'entité ou de caractéristique du type entité qu'il met à jour le programme fait appel à cette macro-requête et lui donne des paramètres relatifs à la réalisation concernée.

La macro GEN-MAJ génère une réalisation de l'entité MAJ et valorise ses caractéristiques simples.

Les caractéristiques TYPMAJ, NUMENTITE, NUMOBJET sont valorisées (à l'aide des paramètres transmis par le programme) pour une création, modification, ou suppression. La caractéristique OBJETPERE est valorisée dans le seul cas d'une création.

2) Macro-requête GEN-CARAC :

Pour chaque réalisation de caractéristique simple qu'il crée ou modifie, le programme appelle la macro GEN-CARAC.

Cette macro génère, dans la réalisation MAJ générée par la macro précédente, une réalisation de la caractéristique-entité CARAC et valorise ses caractéristiques à l'aide des paramètres donnés par le programme.

Réalisation i de l'entité F

itération sur
es
éalisations
e l'entité F

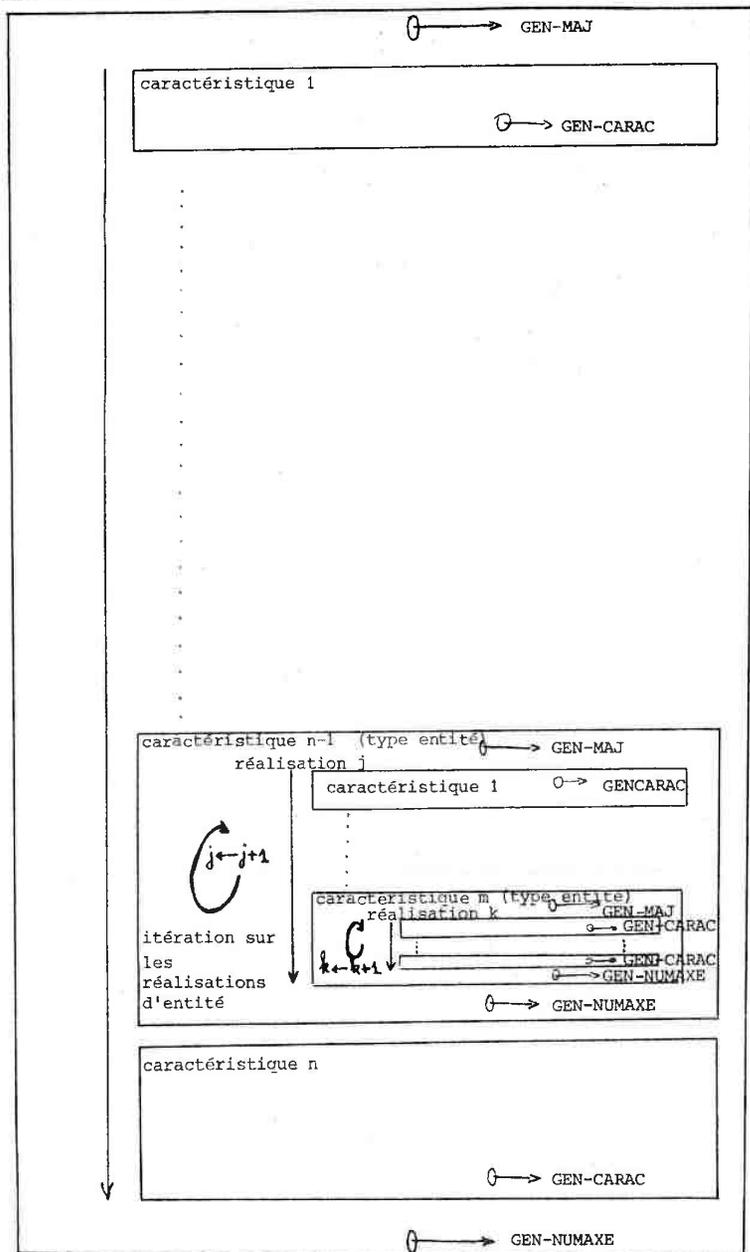


Figure 6 : Structure de programmation pour un programme Pi de mise à jour (cas d'un programme central de création)

La caractéristique NUMCARAC est valorisée dans tous les cas ; AXE est valorisée pour certaines DCP, et pour les DCR dans le cas d'un programme central ; les caractéristiques OPERATEUR et OPERANDE sont valorisées uniquement dans les cas d'une modification relative, d'une caractéristique du type "référence", d'une opération sur un bit d'une chaîne de bits inverses.

On considère que les opérations sur les chaînes de bits qui inversent une certaine entité sont effectuées dans les programmes relatifs à l'entité comme si les chaînes étaient des caractéristiques internes de l'entité.

La macro-requête GEN-CARAC est uniquement appelée par les programmes de création et de modification puisqu'une suppression ne nécessite pas une itération sur les caractéristiques.

3) Macro-requête GEN-NUMAXE

Cette macro est utilisée uniquement par les programmes centraux.

L'utilisation de cette macro est symétrique de celle de la macro GEN-MAJ. Lorsque le programme a terminé la création ou la suppression d'une réalisation de la base il appelle cette macro GEN-NUMAXE.

Celle-ci génère, dans la réalisation MAJ générée par GEN-MAJ, une ou plusieurs réalisations de la caractéristique-entité NUMAXE pour préciser quel(s) axe(s) est concerné par la création ou la suppression de réalisation qui s'achève ; il valorise la caractéristique NUMERO.

3.1.3. Récupération par le L.R.C.

Le L.R.C. a pour tâche de récupérer dans la base locale les données qui ont été mises à jour par les programmes Pi locaux. Pour localiser les réalisations d'entité ou de caractéristique du type entité mises à jour, il utilise les réalisations de l'entité MAJ. Sur un site régional, il crée pour chaque réalisation d'entité MAJ un message unique ; par contre, sur le site central une réalisation d'entité MAJ peut donner lieu à plusieurs messages, au maximum un par axe, si la mise à jour effectuée concerne plusieurs axes : le nombre de messages formés dépend du contenu de NUMAXE.

Le L.R.C. étant un programme écrit en langage évolué il ne lui est pas possible d'accéder directement une réalisation de la base. A cette fin il utilise le module de liaison interface SOCRATE-langage évolué ; cet interface lui permet de faire exécuter des programmes précompilés SOCRATE et de leur transmettre des paramètres.

Pour une réalisation de l'entité MAJ, le L.R.C. crée, en mémoire centrale, un tampon destiné à recevoir toute la réalisation mise à jour. En fonction du contenu de NUMENTITE, il lance un programme d'interrogation qui correspond à l'entité ou à la caractéristique du type entité et lui transmet en paramètre le contenu de NUMOBJET.

Pour chaque entité ou caractéristique-entité, il existe donc un programme d'interrogation qui en fonction du paramètre NUMOBJET dépose toutes les valeurs des caractéristiques de premier niveau dans le tampon du L.R.C. Pour déposer les données, le programme utilise un formal : celui-ci correspond à la description du premier niveau de l'entité ou de la caractéristique-entité. Dans le cas d'imbrications, chaque caractéristique-entité possède son propre programme d'interrogation ainsi que le formal donnant la description de ses caractéristiques de premier niveau.

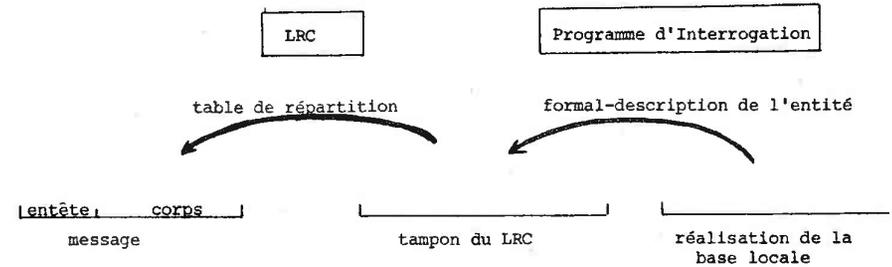


Figure V.7. : Utilisation du formal pour la récupération des mises à jour d'une réalisation

Le tampon rempli, le L.R.C. dispose alors de toute la réalisation mise à jour et de la réalisation MAJ qui lui fournit des renseignements sur les caractéristiques effectivement mises à jour. A l'aide de ces renseignements et de la table de répartition, le L.R.C. peut former le message qu'il dépose ensuite dans le fichier FE. Cependant, il ne dépose pas toutes les données dans le message : en effet, sur un site régional les données qui correspondent à des caractéristiques de type DRP et qui ont été mises à jour n'ont pas à figurer dans le message ; inversement, sur le site central, les données qui correspondent à des caractéristiques de type DCP ne sont pas déposées dans un message.

Pour accéder une à une aux données contenues dans le tampon et les déposer dans le message, le L.R.C. dispose dans la table de répartition de la longueur de chaque caractéristique; en fonction de ces longueurs, il se "déplace" dans le tampon.

Remarquons que dans le cas où la réalisation MAJ indique que la mise à jour effectuée est du type "suppression" alors le L.R.C. peut former immédiatement le message car aucune donnée de la base n'est à transmettre ; il n'y a donc pas appel à un programme d'interrogation.

3.2. Le protocole d'échange LRC-LRC

3.2.1. Définition

Au cours de chaque session de mise en cohérence, les LRC se transmettent les messages contenus dans les fichiers "émission". Pour standardiser les échanges qui ont lieu sur les différents axes un protocole d'échange est nécessaire.

Ce protocole exprime des règles de structuration des messages qui sont imposées à tous les LRC.

Tout message est composé de deux parties : l'entête et le corps.

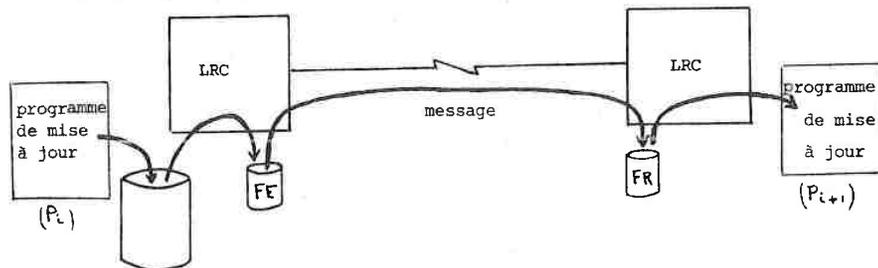


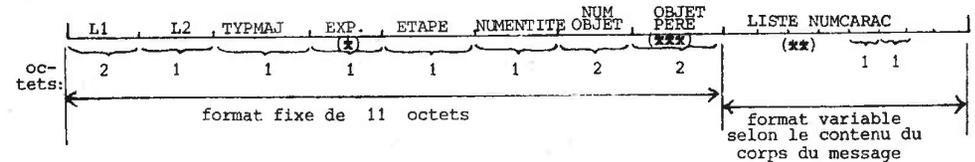
Figure V.8. : Echange de messages entre deux LRC sur un axe

3.2.2. Entête de message

L'entête de message est formée par le LRC à partir des renseignements qu'il trouve dans une réalisation de l'entité MAJ. Les renseignements contenus dans l'entête permettent au LRC destinataire :

- de reconnaître le type de mise à jour et donc de savoir quel programme local faire exécuter.
- de reconnaître les données qui figurent dans le corps du message et qui doivent être données au programme de mise à jour.

La structure de l'entête est la suivante :



- (*) : Dans le sens LRC central → LRC régional, la donnée EXP. n'est pas nécessaire.
- (**) : Dans le cas d'une suppression les données LISTE NUMCARAC n'existent pas car le message ne transmet pas de données ; le message se limite à la partie "format fixe".
- (***) : n'est utilisé que pour les créations.

L1 : Contient la longueur totale du message

L2 : Contient la longueur de l'entête du message

EXP. : Contient le numéro du site régional expéditeur ; nécessaire car tous les messages régionaux sont déposés, sur le site central, dans un même fichier FR.

TYPMAJ: Contient la valeur lue dans TYPMAJ de l'entité MAJ.

ETAPE : Contient le numéro de l'étape de l'action répartie ; permet au LRC destinataire de savoir quel programme faire exécuter.

NUMENTITE : Contient la valeur lue dans NUMENTITE de MAJ.

OBJETPERE : Contient la valeur lue dans OBJETPERE de MAJ.

LISTE NUMCARAC : Contient tous les numéros de caractéristiques ; chaque numéro correspond à un NUMCARAC lu dans MAJ ; le nombre de caractéristiques transmises dans le corps du message est variable : d'où le format variable de cette partie de l'entête.
Seuls les numéros des caractéristiques des types DCR, DRT, DRR sont déposés par le LRC dans le message.

3.2.3. Le corps du message

La longueur du corps du message est définie dans l'entête ; sa structure est variable et dépend de la LISTE NUMCARAC.

En fonction de cette liste, le LRC récepteur procède par déplacements pour accéder aux différentes données contenues dans le corps. Les données y sont rangées dans l'ordre de la LISTE NUMCARAC. Pour chaque NUMCARAC le LRC récepteur consulte sa table de répartition et y trouve la longueur de la caractéristique ce qui permet d'extraire la donnée du corps et de pointer sur la donnée suivante.

Les données contenues dans le corps du message sont des données destinées à être stockées dans la base répartie soit par une action de création soit par une action de modification.

Il existe cependant trois exceptions :

1) Cas des modifications relatives :

Dans le cas d'une caractéristique modifiée de façon relative (cf. IV.4.7.1) ce n'est pas la valeur à stocker qui est transmise mais un opérateur et un opérande. Pour un même NUMCARAC figurant dans la LISTE, le corps contient successivement l'OPERATEUR et l'OPERANDE ; la table de répartition indique pour le NUMCARAC qu'il y a deux données et leur longueur respective : celle de l'opérateur et celle de l'opérande.

2) Cas des caractéristiques du type "référence"

Dans ce cas, la donnée transportée dans le corps est le NUMERO attribué à la réalisation de niveau supérieur qui est référencée par la réalisation qui est en cours de création ou de modification.

3) Cas des chaînes de bit inverse

Le message transporte dans ce cas le contenu de OPERATEUR qui est soit "1" soit "0" ce qui permet au programme récepteur de savoir si la réalisation-réseau dont le numéro figure dans l'entête du message respecte ou non la propriété représentée par la chaîne de bits inverses.

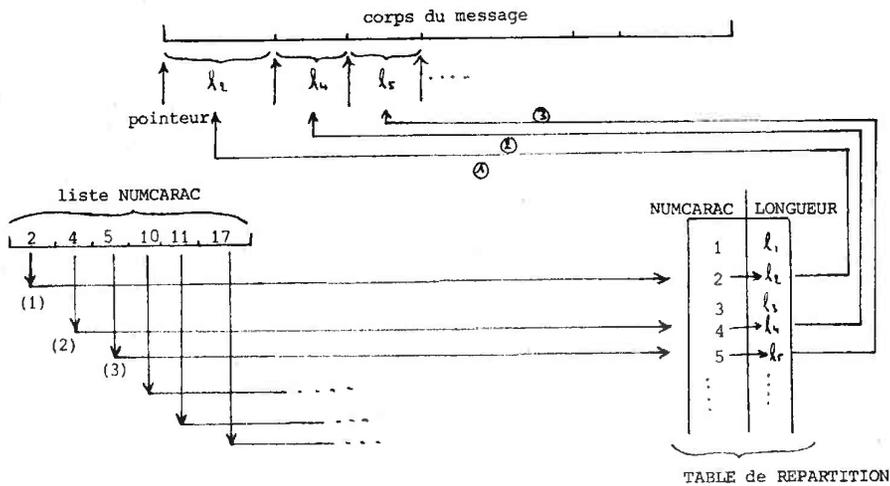


Figure V.9. : Déplacements dans le corps du message

3.3. Exécution des programmes de type P2, P3

3.3.1. Lancement d'un programme

Contrairement aux programmes P1 qui s'exécutent sur ordre des utilisateurs, les programmes des types P2, P3 sont lancés par les LRC et s'exécutent durant les phases de mise en cohérence globale de la base répartie.

Le LRC "lance" un tel programme pour tout message qu'il reçoit dans FR.

Il trouve les renseignements qui lui permettent de sélectionner le programme dans l'entête du message :

- TYPMAJ : permet au LRC de savoir à quelle chaîne correspond le message.
- ETAPE : permet au LRC de savoir si le programme de la chaîne qu'il doit lancer est le programme P2 (étape 2) ou le programme P3 (étape 3).
- NUMENTITE: nécessaire pour savoir sur quelle entité faire exécuter le programme.

Le LRC considère un après l'autre les messages contenus dans FR ; lorsqu'il a sélectionné le programme qui correspond à un message il utilise le module de liaison interface SOCRATE-langage évolué pour "lancer" le programme précompilé ; à la fin de l'exécution du programme, il y a retour au LRC qui peut alors considérer le message suivant contenu dans FR.

3.3.2. Communication LRC → Programme

L'exécution d'un programme de type P2 ou P3 fait suite à celle du programme précédent de la chaîne. Dans la plupart des cas, l'action élémentaire que le programme est chargé d'exécuter sur la base locale se fait à l'aide de données transmises par le programme précédent ; ces données sont contenues dans le message que reçoit le LRC et celui-ci doit les transmettre au programme.

a) Données transmises sous forme de paramètres au lancement du programme :

L'interface SOCRATE-langage évolué permet au LRC de transmettre au programme quelques données. Les données qu'il transmet de cette façon sont celles qui sont nécessaires au programme pour reconnaître la réalisation-réseau sur laquelle il doit exécuter l'action élémentaire.

Dans les cas d'une modification ou d'une suppression le LRC donne au programme le contenu de NUMOBJET. Pour une création qui concerne une réalisation imbriquée dans une autre réalisation il transmet le contenu de OBJETPERE ; cette donnée permet au programme de savoir à quelle réalisation de niveau supérieur il doit "rattacher" la réalisation à créer ; il transmet également le contenu de NUMOBJET : c'est soit un numéro temporaire attribué par un site régional à la réalisation-réseau créée, soit un numéro définitif attribué par le site central ; cette donnée sert de référence à la création et doit être transmise dans le cas où il y a une étape ultérieure. Ces données figurent dans l'entête du message reçu par le LRC.

b) Données déposées dans un tampon

Le corps du message, que possède le LRC, contient des données sous forme concaténée, la liste des numéros de caractéristiques étant contenue dans LISTE NUMCARAC.

Le programme récepteur, s'il doit exécuter une création ou une modification de réalisation d'entité, s'attend à recevoir du LRC des données qui correspondent à des caractéristiques des types DCR, DRT ou DRR. A cet effet, le LRC déconcatène le corps du message et dépose les données dans un tampon dont la structure logique correspond à celle de l'entité mise en cause.

La longueur de chaque élément de la structure du tampon est celle qui est indiquée dans la table de répartition : elle est la même dans le corps du message et dans le tampon. Cette longueur est également celle qui figure dans la description logique de l'entité.

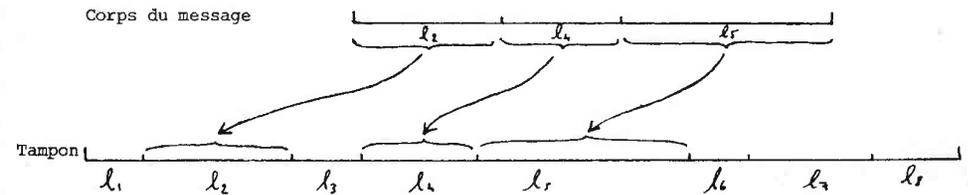


Figure V.10 : Déconcaténation du corps du message

Lorsque le programme est "lancé" celui-ci peut alors accéder aux données du tampon à l'aide d'un formal (SOC-3) ; ce formal représente la description logique du tampon avec les mêmes identificateurs de caractéristiques et les mêmes longueurs que ceux qui figurent dans la description de l'entité.

A l'aide de ce formal, le programme peut accéder une à une les données du tampon et les déposer dans la réalisation d'entité locale.

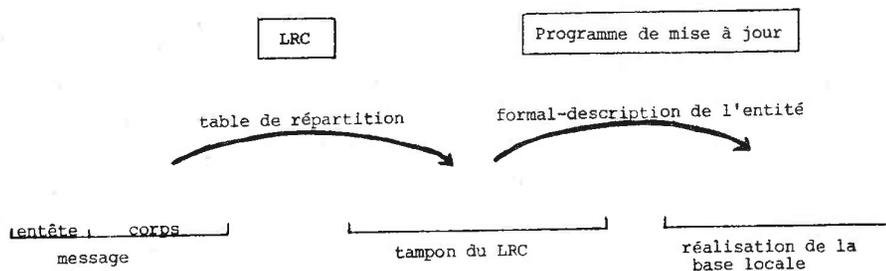


Figure V.11 : Utilisation du formal pour l'exécution d'un programme P2 ou P3 à partir d'un message reçu dans FR

Les remarques mentionnées au paragraphe 3.2.3. restent valables à ce niveau ; en effet, pour les caractéristiques qui subissent des modifications relatives, la structure du tampon et du formal considèrent deux éléments successifs correspondant à l'opérateur et à l'opérande. Pour les caractéristiques du type référence, la donnée contenue dans le tampon est un numéro d'une réalisation de niveau supérieur.

3.3.3. Communication Programme → L.R.C.

Dans certains cas, notamment pour les créations de réalisation-réseau les programmes de type P2 ou P3 ont à transmettre des données à un programme suivant dans la chaîne ; ces données étant stockées dans la base locale.

La communication entre le programme et le LRC local se fait alors exactement comme pour les programmes P1 c'est-à-dire par l'intermédiaire d'une réalisation MAJ.

A la fin de l'exécution du programme, le LRC consulte la réalisation MAJ et récupère les données à transmettre dans la base. Il forme un message et le dépose dans FE. Le message est transmis et exécuté par le programme suivant de la chaîne au cours de la même session de mise en cohérence globale.

Pour générer et valoriser les réalisations MAJ, le programme utilise les principes et les macro-requêtes présentés dans le paragraphe 3.1.2.

3.4. Fonctions propres au LRC central : cohérence globale des DRT, DCR, DRR

Le LRC central possède, par rapport aux LRC régionaux, une fonction supplémentaire qui consiste à exécuter les mécanismes présentés au paragraphe IV.4.7. destinés à éviter les incohérences globales qui résultent des actions concurrentes.

Le LRC central exécute cette fonction à la réception des messages de création ou de modification régionales ; ces messages faisant suite à une action régionale exécutée par un programme P1.

3.4.1. Cas des modifications

Lorsque le LRC central reçoit un message de modification celui-ci contient des modifications relatives ou absolues exécutées sur des données des types DRR, DCR ou DRT. Revenant un tel message, il

- 1) lance un programme P2 (cf. 3.3.) chargé d'exécuter les modifications des DCR et des DRT dans la base centrale.
- 2) lance un programme P'2 qui vérifie si la réalisation-réseau existe sur plusieurs axes :
 - . Si elle n'existe que sur un seul axe, alors le LRC transmet un message au site initial contenant les modifications absolues portant sur les DRT, DCR, et DRR ; celles-ci seront exécutées par un programme P3.
 - . Si elle existe sur d'autre(s) axe(s), alors le LRC transmet un message
 - au site initial, ce message contient les modifications absolues des DRT, DRR, DCR.
 - à chaque autre site régional, un message contenant les modifications absolues et relatives des DRR et des DRT.

Globalement, le site régional émetteur exécute deux fois les modifications absolues portant sur des DRT, DCR ou DRR (cf. IV.4.7.2.).

Pour permettre à un programme P'2 de vérifier sur quel(s) axe(s) est répertoriée une réalisation-réseau, on impose à chaque entité-réseau de posséder dans la réalisation centrale une caractéristique permettant de "localiser" la réalisation-réseau (cf. III.3.2.2.). Si cette caractéristique est la

même pour toutes les entités, alors le programme P'2 est un programme standard capable d'effectuer la vérification quelle que soit l'entité.

Le programme P2, à la fin de son exécution, dépose dans un certain paramètre le numéro de la réalisation centrale qui fait l'objet de la modification ; P'2 peut alors, à l'aide de ce paramètre consulter la caractéristique de localisation.

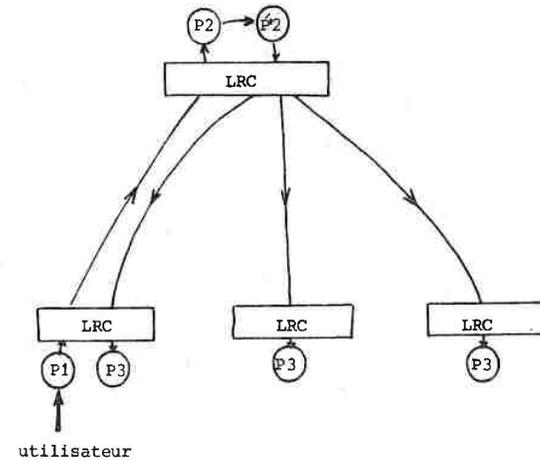


Figure V.12. : Chaîne de modification régionale

3.4.2. Cas des créations (cf. IV.4.7.3.)

Lorsque le LRC central reçoit un message de création régionale celui-ci peut contenir des données des types DCR, DRT, DRR valorisées par l'opération régionale de création.

Si la réalisation-réseau concernée existe déjà sur un ou plusieurs autres axes alors le LRC a pour tâche de vérifier

que les DRT et DRR reçues coïncident avec celles déjà répertoriées sur les autres axes et qui sont supposées exactes et cohérentes entre elles.

Recevant un tel message le LRC central "lance", grâce au module de liaison interface SOCRATE-langage évolué, un programme d'interrogation qui correspond au numéro de l'entité, ce numéro étant contenu dans NUMENTITE (entête de message), et lui transmet en paramètre une donnée CLE qui lui permet de vérifier si la réalisation-réseau n'a pas déjà été créée sur d'autres axes.

a) Si la réponse du programme d'interrogation est négative cela signifie que la réalisation-réseau n'existe encore pas dans la base centrale et le LRC peut alors lancer un programme P2 de création centrale principale en lui transmettant un tampon contenant les DCR et les DRT (cf. 3.3.2.).

b) Si la réponse du programme d'interrogation est positive alors le LRC central doit :

1. Comparer les DRT reçues et les DRT existantes dans la réalisation centrale :

Pour cela, le programme d'interrogation dépose la réalisation centrale dans un tampon du LRC et celui-ci compare les DRT.

En cas d'inégalité, il forme un message de modification qu'il transmet au LRC régional expéditeur ; celui-ci lance alors le programme régional P3 de modification. Les DRT prennent la valeur qui existe dans la réalisation centrale.

2. Former un message contenant les DRR et l'envoyer au LRC d'un des sites régionaux qui possède la réalisation-réseau.

Le message a la structure suivante :

L1 | L2 | TYPMAJ | ORIGINE | NUMENTITE | NUMOBJET | LISTE CARAC |
corps du message

- TYPMAJ : contient une valeur qui signifie une opération de comparaison.
- ORIGINE : contient le numéro du site régional à l'origine de la création.
- Le corps du message contient toutes les DRR.

3.4.3. Le recensement des modifications absolues

Dans le paragraphe 4.7.2. du chapitre IV nous avons vu que les modifications absolues émises par certains sites pouvaient ne pas apparaître dans la base répartie ; dans ce cas elles ont été "écrasées" par une autre modification absolue parvenue ultérieurement au site central.

Il est donc nécessaire que les utilisateurs dont leur modification n'apparaît pas dans la base soient avisés de ce fait.

Au fur et à mesure que le LRC central reçoit des messages contenant des modifications absolues de DRT, DCR, DRR il remplit une table qui lui permet de recenser toutes ces modifications.

NUMENTITE	NUMCARAC	NUMOBJET	SITE
5	18	6742	A
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
5	18	6742	C
-	-	-	-
-	-	-	-
-	-	-	-
5	18	6742	E
-	-	-	-
-	-	-	-

Figure V.13. : Table de recensement des modifications absolue

Pour chaque modification absolue recensée le LRC central indique l'identification ou le numéro du site émetteur ; ce site peut être un site régional ou le site central.

Cette table est transmise à la fin de chaque session de mise en cohérence globale à tous les sites. Ainsi si le site A a modifié la caractéristique n° 18 du fournisseur n° 6742, il s'aperçoit qu'au cours de la même session de travail, les sites C et E ont également modifié la même donnée ; il peut en déduire que la base répartie contient la donnée émise par E puisque celle-ci est la dernière qui ait été remise au LRC central.

Cette table permet, du même coup, à chaque site de connaître les modifications qui ont été introduites par les autres sites dans la base répartie.

3.5. Fonction propre aux LRC régionaux : cohérence globale des DRR

Suite à une création régionale, un LRC régional peut recevoir du LRC central un message contenant les DRR d'une réalisation-réseau, ces DRR étant destinées à être comparées à celles existantes dans la réalisation régionale faisant partie de la réalisation-réseau concernée (cf. 3.4.2.).

Le LRC recevant un tel message lance le programme d'interrogation qui correspond à l'entité en question. Il lui fournit en paramètre le contenu de NUMOBJET qui se trouve dans l'entête du message.

Le programme d'interrogation dépose la réalisation régionale dans un tampon du LRC ; celui-ci compare alors les DRR reçues avec celles déposées dans le tampon.

- En cas d'égalité, les DRR créées sont correctes et le LRC régional ne donne pas suite au message reçu.
- En cas d'inégalité, le LRC forme un message contenant les DRR correctes :

TYPMAJ : contient alors une valeur signifiant pour le LRC central et pour le site initial "modification de DRR".

ORIGINE : contient le numéro du site régional qui doit exécuter la modification.

Le LRC central recevant un tel message dans FR, le dépose directement dans le fichier FE correspondant au site régional spécifié dans ORIGINE.

Un LRC régional recevant un tel message doit lancer le programme P3 de modification régionale ; il remplace les DRR

"créées" par le site par celles qui existent dans les autres bases régionales : la réalisation-réseau est alors globalement cohérente.

3.6. Gestion des fichiers FE et FR

Les fichiers FE et FR sont des fichiers uniquement accessibles par les LRC.

Ce sont des fichiers sur disque à organisation consécutive et à accès séquentiel comportant un nombre déterminé d'enregistrements de longueur fixe. Chaque enregistrement est destiné à contenir un message, celui-ci ayant une longueur variable inférieure à celle des enregistrements.

L'accès à ces enregistrements est cyclique, c'est-à-dire que les messages sont lus dans l'ordre de leur écriture : le premier message introduit dans un fichier FE ou FR sera le premier extrait.

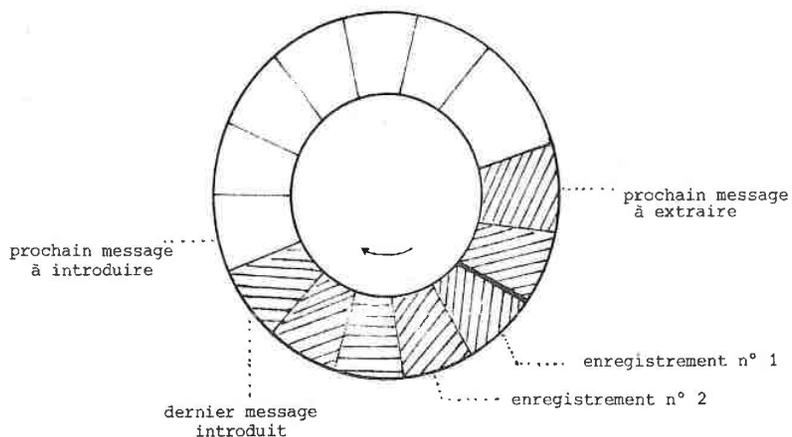


Figure V.14. : Accès cycliques aux fichiers FE et FR

4. LA TABLE DE REPARTITION

4.1. Définition

Nous avons vu dans le paragraphe 3 que pour ses diverses fonctions le LRC nécessite des renseignements sur les données qu'il manipule notamment sur leur type de répartition.

Pour cette raison tous les LRC, central ou régional, possèdent une copie d'une même table constituée à partir de la description logique des différentes entités-réseau qui sont contenues dans la base répartie. Rappelons que la description d'une entité-réseau contient à la fois la description des caractéristiques qui figurent dans l'entité centrale et de celles qui figurent dans l'entité régionale.

Le LRC est un logiciel indépendant des données qui figurent dans la base répartie et doit s'exécuter quelque soit la description logique de celle-ci. Il ne peut "manipuler" les données à partir d'identificateurs d'entités ou de caractéristiques car à chaque changement de description logique le logiciel LRC devrait subir des changements. Ainsi pour assurer la "portabilité" du LRC par rapport à la description logique de la base répartie, la table de répartition ne contient pas d'identificateur d'entités ou de caractéristiques mais des numéros ; chaque entité-réseau introduite dans la base répartie se voit attribuer un numéro ; de même chaque caractéristique de l'entité-réseau possède un numéro propre, relatif à l'entité à laquelle la caractéristique appartient. Ces numérotations sont évidemment globales (la même pour tous les sites) puisque tous les LRC possèdent la même table de répartition.

4.2. Structure de la table

La table de répartition considère séquentiellement la "description" de toutes les entités-réseau et sa structure est la suivante :

	LONGUEUR	TYPE DE REPARTITION	CLE ?	SITE ?	MODIFICATION RELATIVE ?	REFERENCE ?	CHAINE DE BITS INVERSE ?
ité-réseau n°1	l ₁	DCP	0	0	1	0	0
	l ₂	DRT	1	0	1	1	0
	l ₃	DCR	0	1	0	0	0
	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	l _n	DRR	0	0	1	0	0
ité-réseau n°2	l ₁						
	l ₂						
	⋮						
	l _k						
⋮							
	1 octet	3 bits	1 bit	1 bit	1 bit	1 bit	1 bit

caractéristique n°3 de l'entité n°1

LONGUEUR : Contient la longueur prévue dans la description de la caractéristique pour contenir la donnée ; cette longueur est la même que celle qui figure dans la description logique de la caractéristique possédée par le SGBD, sauf pour :

- les caractéristiques du type "référence" : la longueur est égale à celle de la caractéristique OPERANDE de l'entité MAJ (cf. 3.1.1.1.).
- les chaînes de bits qui "inversent" l'entité : la longueur est égale à 1 octet ce qui correspond à la longueur de la caractéristique OPERATEUR de l'entité MAJ (cf. 3.1.1.1.).
- les caractéristiques du type "anneau" : la longueur est égale à zéro.
- les caractéristiques modifiées en "relatif" (cf. IV.4.7. et 3.1.1.1.) : la longueur est égale à celle prévue pour la caractéristique OPERANDE de l'entité MAJ.

TYPE DE REPARTITION : Contient une valeur qui correspond à un des 5 types de répartition :

- DCP = '1'
- DRP = '2'
- DRT = '3'
- DCR = '4'
- DRR = '5'

CLE : Indique si la donnée dans la caractéristique peut permettre de reconnaître la réalisation-réseau par rapport aux autres réalisations-réseau du même objet-réseau ; dans une même entité, plusieurs caractéristiques peuvent servir de clé ; dans une base répertoriant des PERSONNES, le numéro INSEE peut très bien servir de clé ; de même, le numéro d'immatriculation peut servir de clé pour des VOITURES.

SITE : Les caractéristiques DCR et certaines DCP sont contenues dans une "sous-entité" appelée "SITE" dans chaque entité-réseau. Cette colonne permet d'indiquer si la caractéristique appartient (bit = "1") ou non (bit = "0") à la caractéristique de type entité "SITE" (cf.III.3.2.2. et III.3.3.)

MODIFICATION RELATIVE ? : Si le bit est égal à "1", cela indique que la caractéristique subit des modifications relatives.

REFERENCE : Indique si la caractéristique est du type "référence".

CHAINE DE BITS INVERSE ? : Indique s'il s'agit d'une chaîne de bits qui inverse l'entité.

Dans le cas où des entités-réseau sont imbriquées celles-ci sont "décrites" séquentiellement et l'imbrication n'est pas représentée ; ceci est dû au fait que les messages traités par le LRC ne considèrent que les caractéristiques de 1er niveau d'une entité. Une entité imbriquée est considérée comme une entité à part entière par le LRC.

Une exception est cependant faite pour la caractéristique du type entité "SITE" contenue par toute entité-réseau. Les caractéristiques contenues dans le bloc "SITE" sont considérées comme faisant partie du 1er niveau de caractéristiques ; la colonne SITE permet d'indiquer si une caractéristique appartient ou non au bloc SITE.

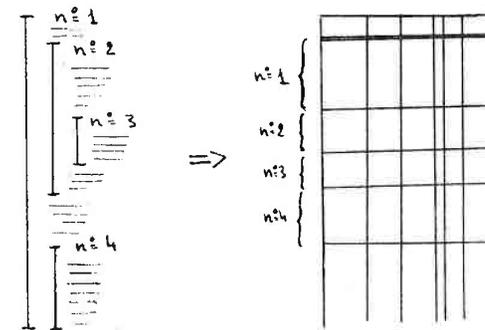


Figure V.15. : Représentation d'entités imbriquées dans la table

Le numéro de chaque caractéristique ne figure pas explicitement dans la table ; le rang dans la table le fixe implicitement.

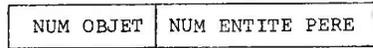
4.3. Stockage et utilisation de la table

Sur chaque site, la table est stockée dans un fichier à organisation partitionnée.

Une première partition contient des numéros d'entités, ces numéros permettant d'identifier les partitions.

A chaque entité-réseau correspond une partition du fichier ; chaque partition comprend des enregistrements de longueur fixe correspondant à une ligne de la table (9 bits). Le nombre d'enregistrements d'une partition est égal au nombre de lignes relatives à l'entité ; chaque enregistrement fournit les renseignements relatifs à une seule caractéristique.

Chaque partition possède un premier enregistrement de structure standard qui contient des renseignements relatifs à l'entité :

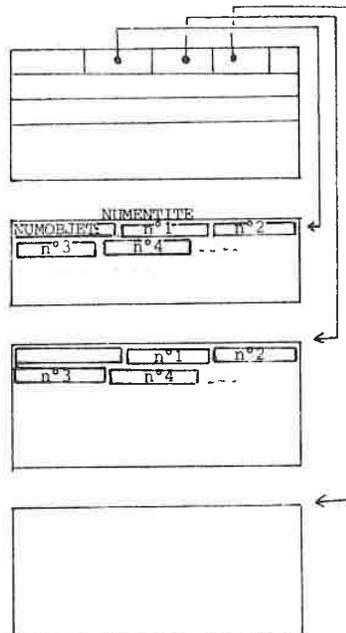


1 octet 1 octet

NUMOBJET : Contient le numéro de la caractéristique qui contient le numéro attribué à la réalisation-réseau ; si le rang de cette caractéristique est implicitement le même dans toutes les entités alors ce renseignement est inutile ; le LRC sait, dans ce cas, trouver, toujours au même rang, le numéro de la réalisation-réseau.

NUM-ENTITE PERE: Contient le numéro de l'entité de niveau supérieur dans le cas d'une entité imbriquée. Si l'entité n'est pas imbriquée dans une autre alors NUM-ENTITE PERE contient "0".

rtition des numéros
entité
dentificateurs de
artitions)



partition relative à
une entité-réseau

Figure V.16. : Stockage de la table de répartition

Ce fichier une fois créé est uniquement accessible en lecture par le seul LRC.

Chaque fois que le LRC travaille sur un message celui-ci est relatif à une certaine entité dont il connaît le numéro (cf. NUM ENTITE) ; il charge alors en mémoire centrale, dans un tampon, la partition qui correspond au numéro d'entité. Chaque enregistrement, excepté le premier, correspond à une caractéristique de l'entité.

4.4. Création de la table de répartition

La table de répartition est créée à partir de la définition de la description des entités régionales et centrales qui composent la base répartie.

Un programme permet de saisir les renseignements à stocker dans la table et ce programme crée le fichier selon la structure de la table. Un exemplaire de ce fichier doit être transmis à chaque site.

La description logique d'une entité SOCRATE ne pouvant être modifiée sans une nouvelle compilation, la table ne peut donc pas être modifiée.

5. ARCHITECTURE INTERNE DU LRC

5.1. La tâche LRC

Sur chaque site, le LRC se présente sous la forme d'une tâche usager ; à l'édition de lien de cette tâche sont inclus les modules du système SOCRATE.

Pour exécuter les fonctions présentées dans le paragraphe 3 la tâche LRC, outre les modules SOCRATE, se décompose en quatre modules :

- un module "moniteur" qui permet d'alterner les différentes fonctions à exécuter au cours d'une session de mise en cohérence.
- un module "interface-réseau" chargé de mettre le LRC en liaison avec le logiciel réseau, d'expédier le contenu de FE et de recevoir des messages dans FR.
- un module "recupération des mises à jour locales" qui consulte l'entité MAJ pour découvrir les mises à jour exécutées dans la base locale et former des messages destinés à d'autres sites.
- un module "exécution" qui, à partir de messages reçus, doit lancer des exécutions de programmes précompilés sur la base locale.

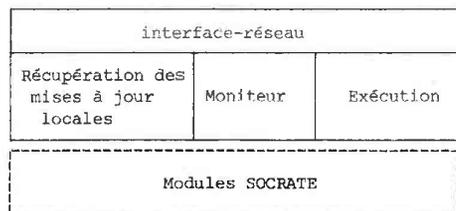


Figure V.17. : Modules de la tâche LRC

5.2. Le module "Interface-réseau"

Le module "interface-réseau" se décompose de quatre sous-modules :

- Le sous-module de "début de session" : lorsque la tâche LRC est activée, ce sous-module a pour fonction d'engager le dialogue avec le LRC symétrique (ou les LRC symétriques dans le cas central) pour établir la connexion. Le principe de la négociation du début de session a été donné dans le chapitre IV au paragraphe 4.6.2. Selon le réseau et le logiciel réseau utilisés, cette négociation nécessite ou non l'échange de messages de type ALLO ? ou ALLO !. Si l'on se place par exemple dans le cas du réseau CYCLADES (CYC-3) l'envoi de tels messages est inutile, mais le sous-module doit demander au logiciel réseau d'établir un "flot" avec le LRC distant ; dans ce cas la négociation de connexion est assurée par le logiciel réseau sur demande du sous-module "début de session".
- Le sous-module "émission de message" : ce sous-module est chargé de prendre un message dans FE, de le remettre au logiciel-réseau en demandant à celui-ci de le transmettre sur le réseau. Dans la terminologie CYCLADES, le sous-module demande au logiciel-réseau (station de transport) de former une "lettre" et de l'expédier.
- Le sous-module "réception de message" : ce sous-module est chargé de demander au logiciel-réseau la réception d'un message ; si un message parvient, alors le sous-module le dépose dans FR. Dans la terminologie CYCLADES, le sous-module adresse au logiciel-réseau (station de transport) une demande de réception de "lettre" ; cette "lettre" contient le message.

- Le sous-module "fin de session" : ce sous-module est chargé de négocier la fin de la session selon le principe évoqué dans le chapitre IV au paragraphe 4.6.4.

La programmation du module "interface-réseau" est en fait très proche de celle du logiciel-réseau utilisé.

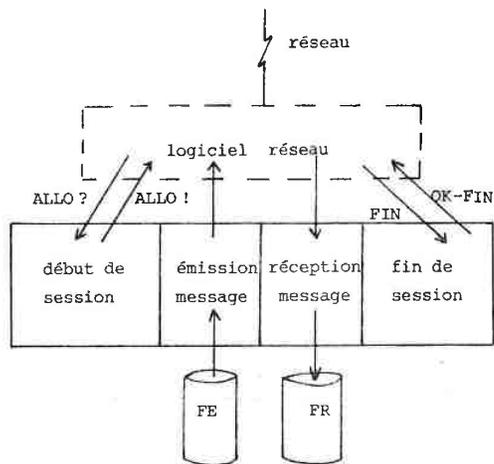


Figure V.18. : Le module "interface-réseau"

5.3. Le module "Récupération des mises à jour locales"

Ce module, chaque fois qu'il est appelé par le module "Moniteur", extrait de l'entité MAJ une réalisation (1) qui lui permet d'extraire de la base locale les mises à jour exécutées sur une réalisation (2). A partir de ces mises à jour, il forme un message qu'il dépose dans le fichier FE (3). Pour consulter MAJ, le module utilise un programme précompilé SOCRATE d'interrogation de l'entité MAJ ; pour extraire les mises à jour de la base, il utilise, selon l'entité, un programme SOCRATE précompilé d'interrogation.

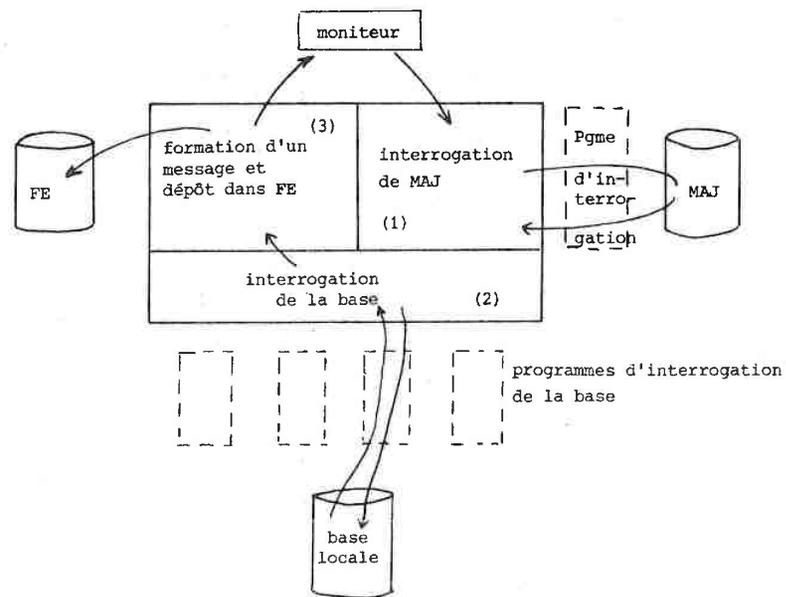


Figure V.19. : Le module "Récupération des mises à jour"

5.4. Le module "Exécution"

Ce module, chaque fois qu'il est appelé par le module "Moniteur" extrait un message de FR et le "déconcatène" ; il lance alors un programme précompilé SOCRATE chargé d'exécuter les mises à jour contenues dans le message.

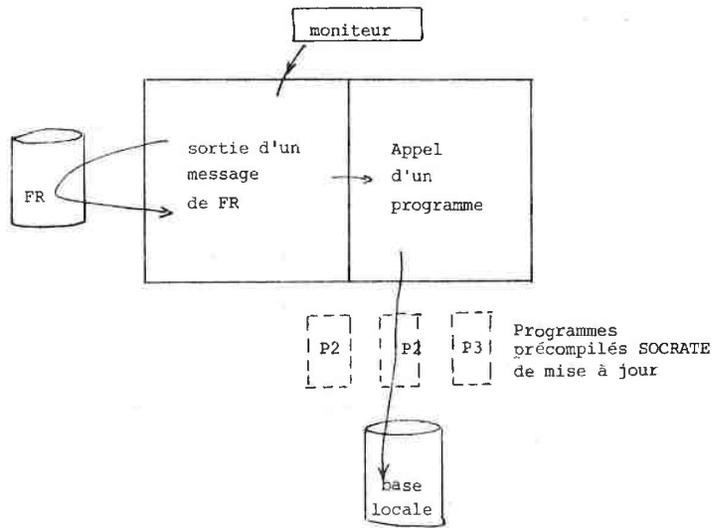


Figure V.20. : Le module "Exécution"

5.5. Le module "Moniteur"

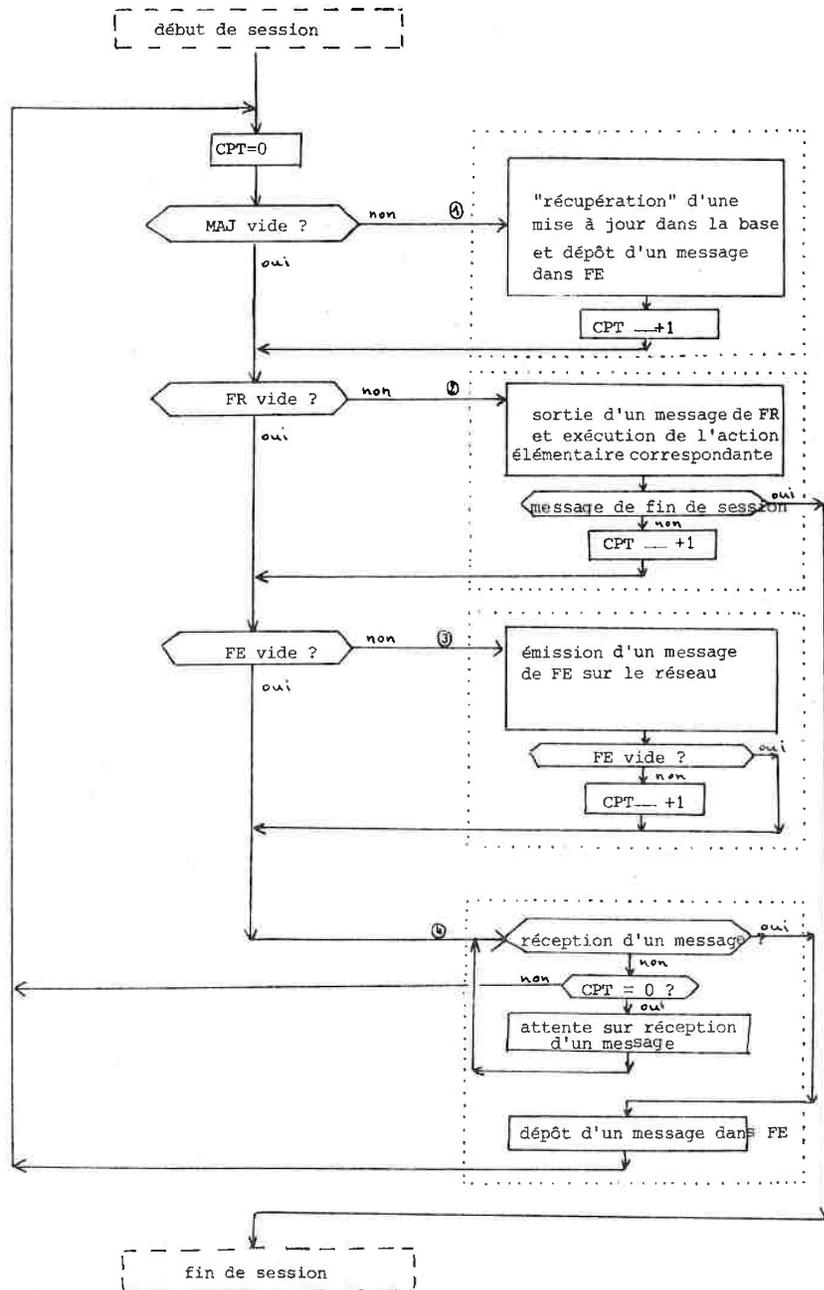
L'organigramme qui suit représente le fonctionnement normal du moniteur c'est-à-dire lorsque la connexion avec le LRC distant est établie ; le moniteur fait exécuter les différents modules ou sous-modules :

1. le module "récupération des mises à jour locales" qui consulte une réalisation MAJ et dépose un message dans FE.
2. le module "exécution" qui extrait un message de FR et fait exécuter l'action élémentaire qui lui correspond.
3. le sous-module "émission" du module "interface-réseau" qui extrait un message de FE et le donne au logiciel-réseau pour qu'il le transmette sur le réseau.
4. le sous-module "réception" du module "interface-réseau" qui demande la réception d'un message au logiciel-réseau.

Le moniteur utilise un compteur CPT qui lui permet de vérifier s'il n'a plus d'activité ; lorsque CPT = 0 cela signifie pour le LRC que

- MAJ est vide, il n'y a plus de mise à jour locale à "récupérer"
- FR est vide
- FE est vide

Dans ce cas le LRC reste en attente de réception d'un message (sous-module "réception") ; lorsqu'un message arrive, il est déposé dans FE et le moniteur reprend son activité.



CHAPITRE VI

SÉCURITÉ DE LA BASE RÉPARTIE

1. DEFINITION DE LA SECURITE POUR LA BASE REPARTIE
2. INACTIVITE D'UN SITE
3. PANNE DE COMMUNICATION
4. PANNE LOCALE AU COURS D'UNE SESSION DE MISE EN COHERENCE

Ce chapitre est consacré à la Sécurité de la base répartie. Comme pour la cohérence (IV), la base répartie impose deux niveaux de sécurité : une "sécurité locale" assurée par les systèmes de gestion, et une "sécurité globale" nécessitée par l'existence d'échanges entre les divers sites possédant une base locale ; c'est précisément ce niveau de sécurité qui fait l'objet du chapitre.

Cette sécurité est à mettre en œuvre au cours des sessions de mise en cohérence lors desquelles toutes les bases locales participent à un contexte global réparti. Trois types de "pannes" sont successivement considérées : l'inactivité d'un site, la panne de communication, et la panne locale survenant au cours d'une session de mise en cohérence.

1. DEFINITION DE LA SECURITE POUR LA BASE REPARTIE (BDR-9)

La base de données répartie dont nous avons proposé l'architecture dans le chapitre III est composée de bases locales "assemblées" selon un modèle étoilé.

Sur chaque site, le système de gestion assure la sécurité de la base dont il a la responsabilité (SOCRATE) ; il préserve les données contre toute panne matérielle ou logicielle : c'est une sécurité locale. Elle est assurée pendant les sessions de travail.

Cependant cette sécurité n'est pas suffisante pour les sessions de mise en cohérence. En effet, lors de ces sessions, toutes les bases locales participent à un contexte global, réparti, mis en œuvre par les logiciels de répartition et de cohérence (L.R.C.) (cf. V.1.) et une panne peut perturber plusieurs voire toutes les bases ; une sécurité de niveau supérieur est donc nécessaire et elle doit être assurée par les L.R.C. ; c'est une sécurité globale.

La sécurité de la base répartie est assurée d'une part, si toutes les sécurités locales sont assurées, et, d'autre part, si la sécurité globale est assurée.

Sécurité de la base répartie	=	sécurité globale	+	(n)	sécurité locale
------------------------------	---	------------------	---	-----	-----------------

Dans la suite du chapitre nous nous intéressons qu'à la seule sécurité globale en montrant comment elle peut être assurée. Pour cela, nous considérons trois classes d' "incidents" pouvant nuire à la sécurité globale :

- L'inactivité d'un site pour une raison humaine, logicielle ou matérielle qui empêche un site de participer à une session de mise en cohérence.
- La panne de communication qui isole un site.
- La panne locale survenant au cours d'une session de mise en cohérence.

2. INACTIVITE D'UN SITE

Par inactivité nous entendons que le site ne participe pas à la session de mise en cohérence.

Dans le cas où il s'agit du site central, alors l'ensemble du contexte global est paralysé : la session de mise en cohérence ne peut pas avoir lieu. Cependant, chaque site peut travailler sur son contexte local jusqu'à la prochaine session de mise en cohérence possible.

Dans le cas où il s'agit d'un site régional son contexte local est isolé du reste du contexte global. Deux possibilités sont à envisager : soit l'inactivité englobe la session de travail qui précède la session de mise en cohérence, dans ce cas il n'y a pas d'action locale à transmettre à d'autres sites; soit l'inactivité débute au cours de la session de travail qui précède, ou, au début de la session de mise en cohérence.

2.1. Inactivité prolongée d'un site régional

Nous considérons dans cette partie le cas où un site régional est inactif pendant toute une (ou plusieurs) session de travail et la session de mise en cohérence. Le site n'a pas travaillé et il n'y a donc pas d'action à transmettre aux autres sites : autrement dit, aucune réalisation d'entité MAJ n'est valorisée dans la base locale. L'isolement du contexte local ne nuit pas aux autres contextes locaux.

Cependant, le site devrait recevoir des messages d'autres sites destinés à créer des actions locales. Globalement, la base répartie est incohérente du fait que les actions réparties qui nécessitent l'exécution d'une action élémentaire sur ce site inactif ne sont pas entièrement exécutées. Le site central, pendant toute la durée de l'inactivité du site, garde dans le fichier "émission" correspondant tous les messages dus au site.

Lorsque le site va reprendre son activité normale, il risque de travailler sur une base incohérente par rapport aux autres bases locales. Pour cette raison, on impose que la reprise de l'activité se fasse obligatoirement après une session de mise en cohérence ; ceci permet à la base locale de se "re-mettre à niveau".

2.2. Inactivité pendant la session de mise en cohérence

Dans ce cas on considère que le site a travaillé et qu'il est inactif pour la session de mise en cohérence. Des actions élémentaires découlent de celles exécutées sur la base locale et devraient être exécutées sur d'autres sites. Mais le transfert n'est pas possible du fait que le site ne participe pas à la session. Inversement, comme dans le cas précédent, le site ne peut recevoir de messages. Lorsque le site va reprendre son activité, alors les actions élémentaires qu'il transmettra seront peut être "démodées" en ce sens qu'elles ne seront plus compatibles avec celles exécutées sur les autres sites.

On impose au site de revenir à un état antérieur qui correspond à la dernière session de mise en cohérence à laquelle il a participé ; toutes les actions locales qu'il a entreprises depuis lors doivent être retirées de la base locale. Eventuellement, les utilisateurs pourront les réexécuter si elles sont encore compatibles avec le contexte global.

Comme dans le cas précédent, cette reprise devra se faire à partir d'une session de mise en cohérence, ce qui permettra à la base locale de se remettre à niveau par rapport au reste du contexte global.

3. PANNE DE COMMUNICATION

Dans ce paragraphe nous considérons les pannes qui sont dues au fait qu'un site ne peut communiquer avec son site symétrique sur l'axe, soit que l'établissement de la liaison est impossible au début de la session, soit qu'elle est interrompue en cours de session.

3.1. Impossibilité d'établir une liaison

Un site désire participer à la session de mise en cohérence et ne parvient pas à établir la connexion-réseau avec le site symétrique sur l'axe :

- Soit, il s'agit d'une défaillance du réseau, mais ce cas est pratiquement impossible ; des solutions de secours sont prévues au niveau du support de transmission pour assurer une fiabilité parfaite.
- Soit, le site distant n'est pas encore connecté :
 - . Il va se connecter ; dans ce cas la liaison sera établie et il n'y a pas de panne.
 - . Il est inactif (cf. 2) et la liaison sera impossible :
 - Si le site distant est le site central, alors tous les sites régionaux sont dans l'impossibilité d'établir la liaison et dans ce cas la session de mise en cohérence n'a pas lieu.
 - Si le site distant est un site régional, alors le site central le considère comme inactif, la session a lieu avec les autres sites régionaux. Le site central peut éventuellement décider d'annuler la session si un nombre maximum de sites régionaux inactifs est atteint.

3.2. Liaison interrompue en cours de session

Si un site fonctionne normalement et que subitement la liaison avec le site symétrique est interrompue alors il s'agit :

- soit, d'une défaillance du réseau, mais on suppose que ce cas est impossible (cf. 3.1.).
- soit, que le site symétrique est tombé en panne (cf. 4).

Dans ce dernier cas, le site symétrique, dès que possible, effectue une session de reprise et reprend contact avec le site. Pendant ce temps, le site peut continuer son activité locale sans émettre ni recevoir de messages en attendant le rétablissement de la liaison. Le site effectuant la reprise se charge de vérifier si aucun message n'a été perdu.

4. PANNE LOCALE AU COURS D'UNE SESSION DE MISE EN COHERENCE

4.1. Définition

Au cours d'une session de mise en cohérence globale, chaque site possède deux tâches en activité : la tâche qui correspond au logiciel-réseau et la tâche qui correspond au LRC et aux modules du système de gestion de la base (cf.V.2.4.).

Dans cette partie, nous considérons le cas d'une panne qui survient au cours de l'exécution normale de ces deux tâches.

L'incident peut être interne à l'une des deux tâches, et celle-ci est détruite. Il peut également être externe (panne de machine, panne du système, ...) et dans ce cas les deux tâches sont détruites.

4.2. Session locale de reprise

La destruction entraîne des perturbations sur les données que la ou les tâches détruites manipulent au moment de l'incident : destruction d'un tampon, d'un fichier, mise à jour non terminée, etc... Les données perturbées par l'incident étant difficilement localisables, il est nécessaire de réexécuter un ensemble d'actions, qui avaient été exécutées, et parmi lesquelles on est sûr de réexécuter celle qui était en cours au moment de l'incident.

A la suite de la panne, lorsque celle-ci a été réparée, le site fait exécuter une reprise par le LRC ; celui-ci reprend son activité à un moment antérieur à celui de l'incident ; il exécute une session locale de reprise. Dès la fin de celle-ci, le fonctionnement normal est automatiquement repris.

4.3. Points de contrôle du LRC

Régulièrement, au cours de la session de mise en cohérence le LRC établit des points de contrôle ; cela lui permet de faire régulièrement un état des actions en cours sur son site.

Au moment où le LRC établit un point de contrôle

- le fichier FE contient des messages
- le fichier FR contient des messages
- l'entité MAJ possède des réalisations

Etablir un point de contrôle consiste pour le LRC :

- A établir un point de contrôle, au sens SOCRATE (SOC-2, SOC-3) sur la base locale ; cela établit une sauvegarde des données de la base, y compris celles de l'entité MAJ.
- A recenser les messages contenus dans FE et FR ; pour cela le LRC possède dans la base une entité qui lui permet de préserver deux paramètres relatifs à FE et FR.

```
entité 50 SECURITE
début
    NUM-FE de 1 à 999
    NUM-FR de 1 à 999
fin
```

Chaque point de contrôle donne lieu à la valorisation d'une réalisation de l'entité SECURITE.

La caractéristique "NUM-FE" contient le numéro du prochain message, contenu dans FE, à transmettre au site distant.

La caractéristique "NUM-FR" contient le numéro du dernier message de FR qui a été entièrement "exécuté" par le LRC local.

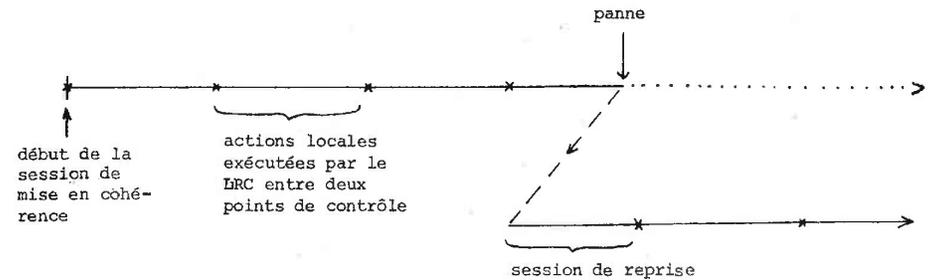


Figure VI.1 : Retour du LRC au dernier point de contrôle

4.4. Principes d'une reprise

4.4.1. Principe général

Les points de contrôle établis par les différents LRC sont indépendants ; il en est de même pour les reprises : celles-ci sont locales et il n'y a aucune synchronisation entre les différents sites. (cf. BDR-9)

Une reprise sur un site, s'établit en liaison avec le site symétrique sur l'axe. En effet, selon l'importance des perturbations, la reprise peut nécessiter des renseignements ou des données de la part du site symétrique.

Dans le cas d'une reprise régionale, le site est aidé par le site central ; dans le cas d'une reprise centrale, le site central doit entrer en liaison avec les différents sites régionaux, mais les principes sont les mêmes sur chaque axe.

Pour décrire une reprise, nous considérons donc un axe, avec un site en fonctionnement normal (le site distant), et un site en reprise (le site local). Les principes de la reprise sont totalement symétriques de part et d'autre d'un axe.

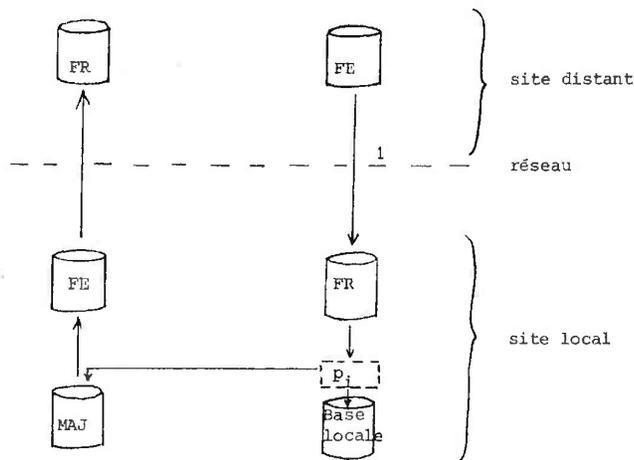


Figure VI.2.: Liaisons entre les fichiers FE et FR sur un axe

La reprise s'exécute à partir du dernier point de contrôle : la base locale est dans un état cohérent ; les exécutions du LRC reprennent à partir des contenus de FR (cf. "exécution") et de MAJ (cf. "récupération") au moment du dernier point de contrôle.

4.4.2. Connexion au site symétrique

La première fonction de la procédure de reprise est de charger le LRC d'établir une nouvelle connexion avec le site symétrique comme pour une participation à une session normale de mise en cohérence.

Dans le cas d'une reprise centrale, cette connexion doit être établie avec tous les sites régionaux.

4.4.3. Vérification de l'état du fichier "réception" local

Selon la panne, le fichier local FR a pu être détruit ou non.

- a) Dans le cas où FR a été détruit, les messages qui doivent être réexécutés sont perdus. Le site local adresse alors un message au LRC distant : "envoyez-moi tous les messages que vous possédez dans FE dont le numéro est supérieur à "NUM-FR"; le site réexécute uniquement les messages parvenus depuis le dernier point de contrôle donc ceux dont le numéro est supérieur à NUM-FR. Il s'en suit donc un transfert de messages de FE distant vers FR local (cf. figure VI.2.1.). Le site local peut ainsi reconstituer son fichier FR.
- b) Dans le cas où FR n'a pas été détruit, le processus de reprise vérifie combien de messages (NB-FR_l) il possède dans FR ; à la fin de la session de reprise, le site demandera au site symétrique de lui transmettre, normalement, les messages dont le numéro est supérieur à NB-FR_l qu'il possède dans FE.

4.4.4. Vérification de l'état du fichier "réception" distant

Il est important de savoir combien de messages, provenant du site en reprise, sont contenus dans le fichier FR distant.

Le message "quel est le numéro du dernier message reçu" est envoyé au site distant. Celui-ci renvoie NB-FR_d, le nombre demandé.

Cela permet au site en reprise de ne pas retransmettre les messages déjà parvenus au site distant ; celui-ci pourrait exécuter deux fois le même message.

4.4.5. Exécution de messages de FR et récupération dans l'entité MAJ

Le LRC considère alternativement un message de FR et une réalisation de l'entité MAJ ; il réalise les fonctions d' "exécution" et de "récupération" définies en V.3. Ces fonctions donnent lieu à la création de messages ; le numéro de ces messages est obtenu par incrémentation à partir de NUM-FE inclus : ces messages ne doivent pas être transmis au site distant car celui-ci les a reçus avant la panne.

4.4.6. Fin de la procédure de reprise

Lorsque le LRC a formé tous les messages jusqu'au numéro NB-FR_d, alors il peut considérer que la reprise est terminée. Tous les messages suivants peuvent être transmis et le LRC reprend son fonctionnement normal.

4.5. Point de contrôle en début de session de mise en cohérence

Dans le paragraphe précédent nous avons vu qu'une reprise sur un site nécessite l'activité normale du site symétrique sur l'axe (ou des sites symétriques dans le cas d'une reprise centrale).

Dans le cas où deux sites symétriques sont en panne en même temps, alors il n'est plus possible de reconstituer les fichiers "émission" et "réception" ; il faut alors entièrement refaire la session de mise en cohérence à partir d'un point de contrôle établi au début de la session de mise en cohérence globale ; dès le début de sa participation à une session, le LRC doit ordonner une sauvegarde de la base locale.

On peut cependant espérer que ce cas de pannes symétriques soit rare.



CHAPITRE VII

MISE EN ŒUVRE DU MODÈLE ÉTOILÉ

1. INCIDENCES DE LA REPARTITION SUR L'APPLICATION
2. LA BASE REPARTIE "FOURNISSEUR-MARCHE"
3. UTILISATION DE LA BASE REPARTIE "FOURNISSEUR-MARCHE"

Nous considérons, dans ce chapitre, les problèmes de mise en œuvre du modèle étoilé. Nous avons présenté dans les chapitres précédents des principes de répartition (III) et de fonctionnement (IV et V). Il s'agit désormais de voir quelles sont leurs incidences sur l'application répartie et comment ils s'appliquent à l'exemple d'application présenté au chapitre II.

Une première partie est consacrée aux incidences de la répartition étoilée sur l'application. Pour cela, nous considérons successivement les incidences logicielles, matérielles, puis les incidences sur les données et sur l'administration de l'application répartie.

Dans la partie suivante nous appliquons les principes de répartition à la base "FOURNISSEUR -MARCHE" de l'exemple d'application du chapitre II. A partir des deux entités initiales nous formons des entités centrales et des entités régionales, à chaque caractéristique nous attribuons un type de répartition en fonction de son utilisation sur tel ou/et tel site. Nous établissons également la table de répartition et les autres descriptions nécessaires au logiciel de répartition et de cohérence.

Dans une dernière partie nous considérons l'utilisation de la base répartie "FOURNISSEUR-MARCHE". A partir des opérations de mise à jour définies au chapitre IV, nous définissons les chaînes de mise à jour effectivement nécessaires à l'application.

1. INCIDENCES DE LA REPARTITION ETOILEE SUR L'APPLICATION

1.1. Incidences logicielles

Le fonctionnement réparti de l'application, tel que nous le proposons nécessite, de la part des sites, un équipement logiciel différent de celui qui existe pour un fonctionnement centralisé.

1.1.1. Les logiciels-réseau

Chaque site doit posséder un logiciel qui lui permette d'expédier et de recevoir des messages.

Ce logiciel dépend essentiellement du support de transmission qui est utilisé ; ce peut être, par exemple, une procédure "X25" dans le cas du réseau TRANSPAC (RES-7) ou une "station de transport" dans le cas du réseau CYCLADES (CYC-2).

Ce logiciel est uniquement actif lors des sessions de mise en cohérence globale, les sites étant "déconnectés" du support de transmission pendant les sessions de travail.

1.1.2. Les systèmes de gestion de base de données

Le modèle de répartition proposé suppose l'existence d'une base de données sur chaque site participant à l'application. Pour cela chaque site doit posséder son propre système de gestion.

Ce système doit pouvoir être utilisé aussi bien en mode conversationnel avec les terminaux d'une région qu'en mode "par lots" avec un interface SOCRATE-langage évolué.

Rappelons que dans l'approche centralisée, seul le site central possède un système SOCRATE.

1.1.3. Les logiciels de répartition et de cohérence (L.R.C.)

Pour pouvoir assurer le contexte global (cf. V.1.) de l'application répartie, chaque site doit posséder un L.R.C. Ce logiciel est uniquement actif en machine pendant la durée des sessions de mise en cohérence globale.

A ce logiciel, programmé en langage d'assemblage, doivent être associés des programmes précompilés dans le langage du système de gestion de la base de données ; ces programmes, stockés en mémoire secondaire, sont appelés par le LRC pour exécuter des fonctions qui nécessitent des accès à des données de la base locale (cf. V.3.).

1.1.4. Les programmes d'accès aux données

Dans la version centralisée, le site central possède tous les logiciels, précompilés dans le langage SOCRATE, qui permettent aux utilisateurs d'accéder aux données de l'application. Dans notre approche, ces logiciels sont répartis. Tous les sites régionaux possèdent les mêmes logiciels (cf.III.2.2.) puisqu'ils sont équivalents dans leurs fonctions (cf.III.1.1.3.). Le site central possède ses logiciels propres.

1.2. Incidences matérielles

1.2.1. Incidences sur le réseau

Dans la version centralisée de l'application répartie, le réseau est essentiellement un réseau de terminaux ; en effet, toutes les transactions se font, de manière étoilée, entre les terminaux implantés dans les régions et les ordinateurs du site central, quelle que soit l'architecture du support de

transmission. Les terminaux ont donc une vocation "réseau".

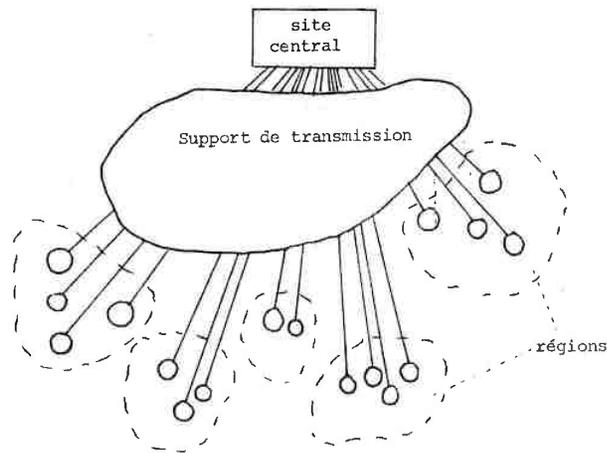


Figure VII.1. : Vocation "réseau" des terminaux

Avec l'approche que nous proposons, les terminaux ont une vocation "régionale" en ce sens qu'ils ne communiquent plus avec le site central mais avec un ordinateur régional. Globalement, le support de transmission et le site central demeurent, mais le réseau est régionalisé : le réseau global se compose de "sous-réseaux" régionaux.

Dans l'approche centralisée, les ordinateurs régionaux ont surtout une vocation "régionale" en ce sens qu'ils ne servent qu'aux traitements locaux ; mais dans l'approche répartie, ils ont une vocation plus orientée vers le réseau : au cours des sessions de travail ils ont surtout une vocation "régionale", mais au cours des sessions de mise en cohérence ils ont surtout une vocation "réseau".

Cette évolution tend à transformer le réseau de terminaux en un réseau d'ordinateurs ou de mini-ordinateurs selon la configuration des sites. Cependant le fonctionnement logique étoilé (cf. II.2.1.) est préservé : un site régional communique toujours uniquement avec le site central.

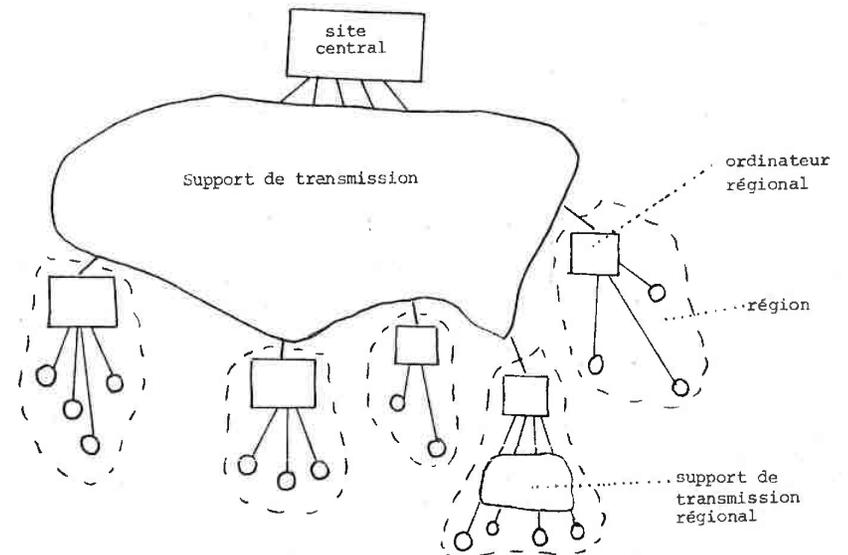


Figure VII.2. : Vocation "régionale" des terminaux

Dans chaque région, l'ordinateur et l'ensemble des terminaux forment un réseau régional de terminaux ; les terminaux peuvent le cas échéant ne pas être directement connectés à l'ordinateur : un support de transmission d'architecture quelconque (cf. I.1.1.) peut être plus judicieux que des liaisons directes.

1.2.2. Incidences sur les sites régionaux

Les sites régionaux doivent être équipés d'une machine capable de supporter les logiciels énumérés en 1. Cette machine doit permettre trois types de sessions (cf. IV.4.3.) :

- une session de travail en mode conversationnel :
Cette session doit permettre aux utilisateurs d'une région d'interroger et de mettre à jour la base de données locale ; la machine doit permettre l'exécution du système de gestion de la base de données locale en mode conversationnel avec l'ensemble des terminaux régionaux.
- une session de travail par lots pour exécuter des traitements :
La machine doit permettre l'exécution d'un programme de traitement écrit en langage évolué, par exemple COBOL. Le programme est édité avec les modules du système de gestion de la base pour permettre l'accès aux données de la base.
- une session de mise en cohérence globale :
Pour cette session, la machine doit supporter
 - . le logiciel-réseau
 - . le L.R.C. édité avec les modules du système de gestion qui lui permet d'accéder aux données de la base.

De plus, la machine régionale doit disposer d'une mémoire secondaire de taille suffisante pour pouvoir y stocker une base de données.

Dans la version centralisée, les configurations régionales existantes ne sont pas forcément suffisantes par rapport à celles exigées pour le fonctionnement réparti. Si on se réfère à l'exemple de l'application décrite dans le chapitre II, la configuration régionale est insuffisante : en effet, elle ne permet pas l'exécution d'un système de gestion de base de données.

Dans le cas d'un changement de machine, le choix doit tenir compte de l'existence d'un système de gestion de base de données : de plus, la taille de la machine doit être suffisante pour permettre l'exécution des trois sessions décrites précédemment.

1.2.3. Incidences sur le site central

Dans la version centralisée, le site central dispose d'un système de gestion de base de données. Le passage à un fonctionnement réparti ne nécessite donc pas de changement ; comme pour les configurations régionales, la configuration centrale doit permettre le déroulement de chacune des trois sessions précédemment décrites.

1.3. Incidences sur les données

Le passage de la base centralisée à la base répartie suppose un certain nombre de modifications sur l'ensemble des données.

1.3.1. Modification des entités initiales

Une entité initiale est décomposée en une entité centrale et une entité régionale, chacune ayant sa description propre.

Dans l'entité centrale, une caractéristique SITE doit être introduite pour contenir les caractéristiques du type DCR et certaines du type DCP (cf. 3.2.2.) ; cette caractéristique est du type "entité" et contient autant de réalisations qu'il existe d'axes central-régional.

1.3.2. Introduction d'une entité "MAJ" et d'une entité "SECURITE"

Pour "récupérer" les mises à jour effectuées par les utilisateurs locaux, chaque LRC dispose d'une entité MAJ. Cette entité est décrite dans chaque base locale (cf. V.3.1.).

Dans chaque base locale doit également figurer une entité "SECURITE" (cf. VI.4.3.) qui permet au LRC de stocker des paramètres lorsqu'il établit des points de reprise.

1.3.3. Introduction de formals

Pour permettre à un LRC d'échanger des données avec un programme en langage SOCRATE, des formals sont introduits dans chaque base locale ; un formal permet de définir une structure à un tampon dans lequel sont déposées des données.

1.3.4. La table de répartition

Chaque site possède, dans un fichier partitionné, une table de répartition ; celle-ci est utilisée par le LRC et contient des renseignements relatifs aux différentes entités-réseau contenues dans la base répartie.

1.3.5. Les fichiers FE et FR

Chaque site régional dispose d'un fichier "émission" et d'un fichier "réception" dans lesquels sont déposés des messages destinés à "créer" des actions élémentaires sur la base répartie.

Le site central possède autant de fichiers "émission" qu'il existe d'axes régional-central.

1.4. Incidences sur l'administration de l'application répartie

Chaque site possède un responsable propre chargé du bon fonctionnement local et des relations avec les autres responsables. L'administration de l'ensemble de l'application répartie nécessite une coordination entre tous les responsables locaux aussi bien du site central que des sites régionaux. La répartition de la base de données a pour avantage d'accroître les responsabilités régionales par rapport aux responsabilités centrales donc de décentraliser les responsabilités de l'administration de l'application.

A la création d'une base répartie, les tâches suivantes incombent aux administrateurs de l'application :

- 1) Définir et décrire les différentes entités-réseau qui composent la base

Cette tâche consiste, pour chaque entité-réseau, à établir sa description logique en fonction des besoins de l'application. Elle établit également les relations entre les différentes entités.

2) Répartir les entité-réseau

Selon les besoins respectifs des différents sites, les administrateurs doivent décider de la répartition des données ; celle-ci s'effectue selon le modèle étoilé décrit au chapitre III. Chaque entité-réseau est décomposée en une entité régionale et une entité centrale. A chaque caractéristique de l'entité-réseau doit être affecté un type de répartition : DCP, DCR, DRT, DRF, DRR.

3) Etablir la table de répartition

En fonction des entité-réseau et de leur répartition, les administrateurs établissent la table de répartition qui correspond à la base créée.

4) Définition des sessions

Les différents administrateurs doivent se mettre d'accord sur les heures de début des différentes sessions et sur la durée des sessions de travail.

5) Définir et faire programmer les chaînes de programmes

A chaque entité-réseau correspondent des chaînes de programmes qui permettent aux utilisateurs de manipuler les données de la base répartie.

Ceci fait, chaque responsable local doit assurer les tâches suivantes :

- 1) Faire compiler la structure de la base locale
- 2) Faire compiler les programmes locaux des différentes chaînes de programmes

- 3) Implanter la table de répartition dans un fichier local.

Dès que ces tâches ont été exécutées sur tous les sites locaux, la base répartie est accessible aux utilisateurs.

2. LA BASE REPARTIE "FOURNISSEUR-MARCHE"

2.1. Les entités-réseau "Fournisseur" et Marché"

Si l'on applique les principes du modèle étoilé aux deux entités "Fournisseur" et "Marché" décrites dans le chapitre II celles-ci deviennent des entités-réseau c'est-à-dire deux entités qui se décomposent chacune en une entité centrale, d'une part, et en une entité régionale, d'autre part.

Les deux tableaux qui suivent donnent pour chaque entité-réseau la liste des caractéristiques qui la composent ainsi que le type de répartition affecté à chaque caractéristique, les raisons du choix du type affecté et le numéro attribué à chaque caractéristique.

ENTITE-RESEAU "FOURNISSEUR"

A-MARFOU	DRT	1	Un marché peut concerner plusieurs axes
A-COMFOU	DCR	2	Une commande est propre à un site régional mais intéresse aussi le site central
A-FACFOU	DCR	3	Une facture suit la répartition de la commande
COD-FOURNISSEUR	DRP	4	Donnée propre à un axe, le sous-code qui compose cette donnée étant une DCR
N-FOURNISSEUR	DRT	5	Numéro global permettant d'identifier un fournisseur sur tous les sites
SCOD-FOURNISSEUR	DCR	6	Donnée propre à un axe
COD-TYPE	DRT	7	Le fournisseur est d'un type unique
RAISON-SOCIALE	DRT	8	Intéresse tous les sites répertoriants le fournisseur
LGN-ADRESSE1	DCP	9	N'intéresse que le site central
LGN-ADRESSE2	DCP	10	N'intéresse que le site central
COD-POSTAL	DCP	11	N'intéresse que le site central
LGN-ADRESSE3	DRT	12	Intéresse tous les sites répertoriants le fournisseur
COD-NATION/REGION	DRT	13	
CIRC-UTIL	DCP	14	Donnée permettant de savoir dans la base centrale sur quel axe est utilisé et répertorié le fournisseur
MT-COMMANDE	DCR	15	Contient le montant relatif à un axe
MT-COM-GLOBAL	DCP	16	Contient la somme des montants ; n'intéresse que le site central

COD-PLAFOND	DCP	17	N'intéresse que le site central
DATE-MAJ	DRT	18	Intéresse tous les sites répertoriants le fournisseur
DATE-C/M	DCR	19	La date de création est propre à un axe
MOD-PAIEMENT	DRT	20	Intéresse tous les sites répertoriants le fournisseur
COD-DOMICIL	DCP	21	N'intéresse que le site central
CLE-DOMICIL	DRT	22	Intéresse tous les sites répertoriants le fournisseur
BANQUE	DRT	23	Intéresse tous les sites répertoriants le fournisseur
GUICHET	DRT	24	Intéresse tous les sites répertoriants le fournisseur
N-CPTCP	DRT	25	Intéresse tous les sites répertoriants le fournisseur
LIBELLE-DOMICIL	DCP	26	N'intéresse que le site central
CA-HORS-MARCHE	DRT	27	Intéresse tous les sites répertoriants le fournisseur
CA-MARCHE	DRT	28	Intéresse tous les sites répertoriants le fournisseur
N-SIRET	DRT	29	Intéresse tous les sites répertoriants le fournisseur

ENTITE-RESEAU "MARCHE"

A-ENGMAR	DCP	1	N'intéresse que le site central
A-AVEMAR	DRT	2	Un avenant peut concerner un marché national donc concerne plusieurs sites
A-COMMAR	DCR	3	Une commande est saisie sur un site régional donc est relative à un axe
A-RETMAR	DCR	4	Idem pour la retenue
A-FACMAR	DCR	5	Idem pour la facture
R-MARFOU	DRT	6	Le fournisseur peut être répertorié sur plusieurs sites
N-MARCHE	DRT	7	Numéro global intéressant tous les sites répertoriants le marché (dans le cas d'un marché national)
TYP-MARCHE	DRT	8	Intéresse tous les sites répertoriants le marché
ORG-CONTRACTANT	DCP	9	N'intéresse que le site central
PARTIE-PRENANTE	DCR	10	N'intéresse qu'un axe car dépend de la connexion
CIRC-CREATRICE	DCP	11	Code "CNRS" si marché national (création centrale) code de la circonscription si marché régional (création régionale)
COD-FOURBENEF	DRT	12	Intéresse tous les sites répertoriants le marché
MOD-PAIEMENT	DRT	13	Intéresse tous les sites répertoriants le marché
DATE-DEBUT	DCP	14	Intéresse uniquement le site central (valorisée uniquement dans le cas d'un marché national)
DATE-FIN	DCP	15	Intéresse uniquement le site central (valorisée uniquement dans le cas d'un marché national)
QMU-COMMANDE	DRT	16	Cumul global des commandes relatives au marché, concerne tous les sites répertoriants le marché
QMU-MANDATE	DRT	17	Intéresse tous les sites répertoriants le marché

NP-MANDAT	DRT	18	Intéresse tous les sites répertoriant le marché
DATE-NP-MDT	DRT	19	Intéresse tous les sites répertoriant le marché
DATE-MAJ	DRT	20	Intéresse tous les sites répertoriant le marché
SEUIL-MINI	DRT	21	Intéresse tous les sites répertoriant le marché
SEUIL-MAXI	DRT	22	Intéresse tous les sites répertoriant le marché
MT-MARCHE	DCR	23	Valorisée uniquement pour un marché régional ; donc donnée relative à un axe
LIB-MARCHE	DRT	24	Intéresse tous les sites répertoriant le marché
PERIODICITE	DCP	25	N'intéresse pas le site central qui effectue le paiement
NB-PA-FAIRE	DCP	26	N'intéresse pas le site central qui effectue le paiement
NB-PA-FAITS	DCP	27	N'intéresse pas le site central qui effectue le paiement
DATE-MDT-PA	DCP	28	N'intéresse pas le site central qui effectue le paiement
MT-PA	DCP	29	N'intéresse pas le site central qui effectue le paiement
N-COMMANDE	DCP	30	N'intéresse pas le site central qui effectue le paiement

Si on compare ces tableaux à ceux présentés dans le chapitre II certaines remarques s'imposent :

1) CIRC-UTIL

Dans la version centralisée de l'application il existe 7 caractéristiques (CIRC-UTIL1, ..., CIRC-UTIL7) pour indiquer sur quel site est utilisé un certain fournisseur. Dans notre cas, il existe une caractéristique centrale unique incluse dans le bloc SITE (cf.III.3.2.2.) de l'entité FOURNISSEUR ; pour chaque site (ou axe) utilisant le fournisseur, une réalisation de cette caractéristique est valorisée (cf. III.3.3.).

2) CIRC-CREATRICE (cf.II.3.)

Cette caractéristique est supprimée dans notre approche, car tout site régional qui contient le fournisseur l'a obligatoirement créé ; la donnée est donc implicite.

3) MT-COMMANDE

Cette caractéristique contient un montant relatif à un axe ; autant de réalisations possibles dans la base centrale que d'axes.

4) MT-COM-GLOBAL

Cette caractéristique est introduite dans notre approche pour totaliser les contenus des différentes réalisations de la caractéristique MT-COMMANDE.

2.2. Les entités centrales et régionales "Fournisseur" et "Marché"

Chaque base possède une description locale des entités "Fournisseur" et "Marché".

2.2.1. Descriptions centrales

	1	ENTITE (80 000)	FOURNISSEUR
		DEBUT	
DRT	1	A-MARFOU	ANNEAU
DRT	5	N-FOURNISSEUR	DE 1 A 999999 AVEC CLE FIN
DRT	7	COD-TYPE	(11 1) (A B C D G L M N R S X)
DRT	8	RAISON-SOCIALE	MOT (30) AVEC CLE FIN
DRT	9	LGN-ADRESSE1	MOT (30)
DCP	10	LGN-ADRESSE2	MOT (30)
DCP	11	COD-POSTAL	MOT (6)
DRT	12	LGN-ADRESSE3	MOT (23)
DRT	13	COD-NATION/REGION	DE 0 A 1
DCP	15	MT-COM-GLOBAL	MOT (13)
DCP	17	COD-PLAFOND	DE 0 A 1
DRT	18	DATE-MAJ	DE 101 A 991231
DRT	20	MOD-PAIEMENT	(34 1) (0 1 2 3 4 5 6 7 8 A B C D E F G H I J K L M N O P Q R S T V W X Y Z)
DCP	21	COD-DOMICIL	(2 1) (B P)
DRT	22	CLE-DOMICIL	MOT (21) AVEC CLE UNIQUE FIN
DRT	23	BANQUE	DE 0 A 99999
DRT	24	GUICHET	DE 0 A 99999
DRT	25	N-CPTCP	MOT (11) AVEC CLE FIN

DCP	26	LIBELLE-DOMICIL	MOT (24)
DRT	27	CA-HORS-MARCHE	MOT (13)
DRT	28	CA-MARCHE	MOT (13)
DRT	29	N-SIRET	MOT (14) AVEC CLE UNIQUE FIN
		ENTITE x SITE	
		DEBUT	
DCR	2	A-COMFOU	ANNEAU
DCR	3	A-FACFOU	ANNEAU
DCR	6	SCOD-FOURNISSEUR	DE 1 A 99
DCP	14	CIRC-UTIL	MOT (2)
DCR	15	MT-COMMANDE	MOT (13)
DCR	19	DATE C/M	DE 101 A 991231
		FIN	

	2	ENTITE (15 000) MARCHE	
		DEBUT	
DCP	1	A-ENGMAR	ANNEAU
DRT	2	A-AVEMAR	ANNEAU
DRT	6	R-MARFOU	REFERE A-MARFOU DE UN FOURNISSEUR
DRT	7	N-MARCHE	MOT (7) AVEC CLE UNIQUE FIN
DRT	8	TYP-MARCHE	(7 1) (1 2 3 4 5 6)
DCP	9	ORG-CONTRACTANT	MOT (1)
DCP	11	CIRC-CREATRICE	MOT (2)
DRT	12	COD-FOURBENEF	DE 1 A 99999999
DRT	13	MOD-PAIEMENT	MOT (1)
DCP	14	DATE-DEBUT	DE 1 A 991231
DCP	15	DATE-FIN	DE 1 A 991231
DRT	16	QMU-COMMANDE	MOT (13)
DRT	17	QMU-MANDAT	MOT (13)
DRT	18	NP-MANDAT	DE 1 A 99999
DRT	19	DATE-NP-MDT	DE 101 A 991231
DRT	20	DATE-MAJ	DE 101 A 991231
DRT	21	SEUIL-MINI	MOT (13)
DRT	22	SEUIL-MAXI	MOT (13)
DRT	24	LIB-MARCHE	MOT (22)
DCP	25	PERIODICITE	DE 1 A 6
DCP	26	NB-PA-A-FAIRE	DE 0 A 99
DCP	27	NB-PA-FAITS	DE 0 A 99
DCP	28	DATE-MDT-PA	DE 1 A 9912
DCP	29	MT-PA	MOT (13)
DCP	30	N-COMMANDE	MOT (6)
		ENTITE x SITE	
		DEBUT	
DCR	3	A-COMMAR	ANNEAU
DCR	4	A-RETMAR	ANNEAU
DCR	5	A-FACMAR	ANNEAU
DCR	10	PARTIE-PRENANTE	DE 1 A 999999
DCR	23	MT-MARCHE	MOT (13)
		FIN	

x : indique le nombre d'axes de l'application

2.2.2. Descriptions régionales

	1	ENTITE (10 000) FOURNISSEUR	
		DEBUT	
DRT	1	A-MARFOU	ANNEAU
DCR	2	A-COMFOU	ANNEAU
DCR	3	A-FACFOU	ANNEAU
DRP	4	COD-FOURNISSEUR	DE 1 A 99999999
DRT	5	N-FOURNISSEUR	DE 1 A 999999
DCR	6	SCOD-FOURNISSEUR	DE 1 A 99
DRT	7	COD-TYPE	(11 1) (A B C D G L M N R S X)
DRT	8	RAISON-SOCIALE	MOT (30) AVEC CLE FIN
DRT	12	LGN-ADRESSE3	MOT (23)
DRT	13	COD-NATION/REGION	DE 0 A 1
DCR	15	MT-COMMANDE	MOT (13)
DRT	18	DATE-MAJ	DE 101 A 991231
DCR	19	DATE-C/M	DE 101 A 991231
DRT	20	MOD-PAIEMENT	(34 1)
			(0 1 2 3 4 5 6 7 8 A B C D E F G H I J
			K L M N O P Q R S T V W X Y Z)
DRT	22	CLE-DOMICIL	MOT (21) AVEC CLE UNIQUE FIN
DRT	23	BANQUE	DE 0 A 99999
DRT	24	GUICHET	DE 0 A 99999
DRT	25	N-CPTCP	MOT (11) AVEC CLE FIN
DRT	27	CA-HORS-MARCHE	MOT (13)
DRT	29	N-SIRET	MOT (14) AVEC CLE UNIQUE FIN
		FIN	

	2	ENTITE (5 000) MARCHE	
		DEBUT	
DRT	2	A-AVEMAR	ANNEAU
DCR	3	A-COMMAR	ANNEAU
DCR	4	A-RETMAR	ANNEAU
DCR	5	A-FACMAR	ANNEAU
DRT	6	R-MARFOU	REFERE A-MARFOU DE UN FOURNISSEUR
DRT	7	N-MARCHE	MOT (7) AVEC CLE UNIQUE FIN
DRT	8	TYP-MARCHE	(7 1) (1 2 3 4 5 6)
DCR	10	PARTIE-PRENANTE	DE 1 A 999999
DRT	12	COD-FOURBENEF	DE 1 A 99999999
DRT	13	MOD-PAIEMENT	MOT (1)
DRT	16	QMU-COMMANDE	MOT (13)
DRT	17	QMU-MANDAT	MOT (13)
DRT	18	NP-MANDAT	DE 1 A 99999
DRT	19	DATE-NP-MDT	DE 101 A 991231
DRT	20	DATE-MAJ	DE 101 A 991231
DRT	21	SEUIL-MINI	MOT (13)
DRT	22	SEUIL-MAXI	MOT (13)
DCR	23	MT-MARCHE	MOT (13)
DRT	24	LIB-MARCHE	MOT (22)
		FIN	

2.3. La table de répartition

En fonction des descriptions que nous venons de donner, la table de répartition a le contenu suivant.

ENTITE : 1 (Fournisseur)

5	0
---	---

numobjet objet père

A-MARFOU
 A-COMFOU
 A-FACFOU
 COD-FOURNISSEUR
 N-FOURNISSEUR
 SCOD-FOURNISSEUR
 CODE-TYPE
 RAISON-SOCIALE
 LGN-ADRESSE1
 LGN-ADRESSE2
 COD-POSTAL
 LGN-ADRESSE3
 COD-NATION/REGION
 CIRC-UTIL
 MT-COMMANDE
 MT-COM-GLOBAL
 COD-PLAFOND
 DATE-MAJ
 DATE-C/M
 MOD-PAIEMENT
 COD-DOMICIL
 CLE-DOMICIL
 BANQUE
 GUICHET
 N-CFTCP
 LIBELLE-DOMICIL
 CA-HORS-MARCHE
 CA-MARCHE
 N-SIRET

LONGUEUR (octets)	TYPE	CLE	SITE	MODIF. RELATIVE	Réf.	CHAINE INVERSE
0	DRT					
0	DCR		1			
0	DCR		1			
4	DRP					
3	DRT	1				
1	DCR		1			
1	DRT					
30	DRT					
30	DCP					
30	DCP					
6	DCP					
23	DRT					
1	DRT					
1	DCP		1			
13	DCR		1	1		
13	DCP			1		
1	DCP					
3	DRT					
3	DCR		1			
1	DRT					
1	DCP					
21	DRT					
3	DRT					
3	DRT					
11	DRT	1				
24	DCP					
13	DRT					
13	DRT					
14	DRT	1				

ENTITE : 2 (Marché)

7	0
---	---

numobjet objet père

A-ENGMAR
 A-AVEMAR
 A-COMMAR
 A-RETMAR
 A-FACMAR
 R-MARFOU
 N-MARCHE
 TYP-MARCHE
 ORG-CONTRACTANT
 PARTIE-PRENANTE
 CIRC-CREATRICE
 COD-FOURBENEF
 MOD-PAIEMENT
 DATE-DEBUT
 DATE-FIN
 QMU-COMMANDE
 QMU-MANDATE
 NP-MANDAT
 DATE-NP-MDT
 DATE-MAJ
 SEUIL-MINI
 SEUIL-MAXI
 MT-MARCHE
 LIB-MARCHE
 PERIODICITE
 NB-PA-FAIRE
 NB-PA-FAITS
 DATE-MDT-PA
 MT-PA
 N-COMMANDE

LONGUEUR (octets)	TYPE	CLE	SITE	MODIF. RELATIVE	Réf.	CHAINE INVERSE
0	DCP					
0	DRT					
0	DCR		1			
0	DCR		1			
0	DCR		1			
	DRT					1
7	DRT		1			
1	DRT					
1	DCP					
3	DCR		1			
2	DCP					
4	DRT					
1	DRT					
3	DCP					
3	DCP					
13	DRT					
13	DRT					
3	DRT					
3	DRT					
3	DRT					
13	DRT					
13	DRT					
13	DCR		1	1		
22	DRT					
1	DCP					
1	DCP					
1	DCP					
2	DCP					
13	DCP					
6	DCP					

2.4. Les formals

Pour récupérer les données mises à jour dans une base locale le LRC utilise des programmes d'interrogation de chaque entité de la base (cf.V.3.1.3); chaque tel programme dépose, grâce à un formal, les données mises à jour dans un tampon du LRC ; le LRC "manipule" alors les données contenues dans le tampon grâce aux longueurs indiquées dans la table de répartition et forme un message ; celui-ci ne contient pas toutes les données du tampon : seules celles susceptibles d'intéresser d'autres sites sont déposées dans le message.

Ces formals sont également utilisés lorsque le LRC lance un programme P2 ou P3 ; le LRC, à partir du message reçu dans FR, dépose les données contenues dans le message dans un tampon ayant la structure du formal ; le programme SOCRATE accède ces données grâce au formal.

Les mêmes formals sont donc utilisés pour la fonction de "récupération des mises à jour locales" (cf. V.3.1.) par des programmes SOCRATE d'interrogation de la base et pour la fonction "exécution des programmes P2 ou P3" (cf. V.3.3.) par ces mêmes programmes.

Remarquons que les formals centraux ne considèrent que les données DCR, DRT, alors que les formals régionaux ne considèrent que les données DRR, DCR, DRT.

2.4.1. Les formals centraux

FORMAL	FOURNISSEUR		
DEBUT			
N-FOURNISSEUR	PACKE	3	
COD-TYPE	MOT	1	
RAISON-SOCIALE	MOT	30	
LGN-ADRESSE1	MOT	30	
LGN-ADRESSE3	MOT	23	
CODE-NATION/REGION	MOT	1	
DATE-MAJ	PACKE	3	
MOD-PAIEMENT	MOT	1	
CLE-DOMICIL	MOT	1	
BANQUE	PACKE	3	
GUICHET	PACKE	3	
N-CPTCP	MOT	11	
CA-HORS-MARCHE	MOT	13	
CA-MARCHE	MOT	13	
N-SIRET	MOT	14	
FORMAL (x) SITE			
DEBUT			
SCOD-FOURNISSEUR	PACKE	1	
MT-COMMANDE	MOT	()	(*) dimension de
OPERATEUR1	MOT	1	OPERANDE
DATE-C/M	PACKE	3	
FIN			
FORMAL			
MARCHÉ			
DEBUT			
R-MARFOU	MOT	()	(*)
N-MARCHE	MOT	7	
TYP-MARCHE	MOT	1	
COD-FOURBENEF	PACKE	4	
MOD-PAIEMENT	MOT	1	
QMU-COMMANDE	MOT	13	
QMU-MANDATE	MOT	13	
NP-MANDAT	PACKE	3	
DATE-NP-MDT	PACKE	3	
DATE-MAJ	PACKE	3	
SEUIL-MINI	MOT	13	
SEUIL-MAXI	MOT	13	
LIB-MARCHE	MOT	22	
FORMAL (x) SITE			
DEBUT			
PARTIE-PRENANTE	PACKE	3	
MT-MARCHE	MOT	()	(*)
OPERATEUR1	MOT	1	
FIN			
FIN			

(*) Les données contenues dans ces caractéristiques sont déposées en fait, dans un message, dans une zone ayant la dimension de la caractéristique OPERANDE de l'entité MAJ (cf. V. 3.1.).

x : Représente le nombre d'axes de l'application répartie.

2.4.2. Les formals régionaux

FORMAL	FOURNISSEUR			
DEBUT				
N-FOURNISSEUR	PACKE	3		
SCOD-FOURNISSEUR	MOT			
COD-TYPE	MOT	1		
RAISON-SOCIALE	MOT	30		
LGN-ADRESSE3	MOT	23		
COD-NATION/REGION	MOT	1		
MT-COMMANDE	MOT	()		(*)
OPERATEUR1	MOT	1		
DATE-MAJ	PACKE	3		
DATE-C/M	PACKE	3		
MOD-PAIEMENT	MOT	1		
CLE-DOMICIL	MOT	21		
BANQUE	PACKE	3		
GUICHET	PACKE	3		
N-CPTBP	MOT	11		
CA-HORS-MARCHE	MOT	13		
CA-MARCHE	MOT	13		
N-SIRET	MOT	14		

FIN

FORMAL	MARCHE			
DEBUT				
R-MARFOU	MOT	()		(*)
N-MARFOU	MOT	7		
TYP-MARCHE	MOT	1		
PARTIE-PRENANTE	PACKE	3		
COD-FOURBENEF	PACKE	4		
MOD-PAIEMENT	MOT	1		
QMU-COMMANDE	MOT	13		
QMU-MANDATE	MOT	13		
NP-MANDAT	PACKE	3		
DATE-NP-MDT	PACKE	3		
DATE-MAJ	PACKE	3		
SEUIL-MINI	MOT	13		
SEUIL-MAXI	MOT	13		
MT-MARCHE	MOT	()		(*)
OPERATEUR1	MOT	1		
LIB-MARCHE	MOT	22		

FIN

3. UTILISATION DE LA BASE REPARTIE

3.1. Les interrogations conversationnelles

Chaque site dispose, pour chaque entité, d'un programme local d'interrogation conversationnelle.

Le site central possède ses programmes propres qui lui permettent d'interroger les données DCP, DRT, DCR des deux entités "Fournisseur" et "Marché".

Les sites régionaux possèdent tous les mêmes programmes qui leurs permettent d'interroger, localement, les données DRP, DRR, DRT et DCR des deux entités.

Ces programmes sont précompilés et sont accessibles depuis des terminaux conversationnels sur simple appel.

Ces programmes ne doivent pas être confondus avec les programmes d'interrogation utilisés par les LRC ; ces programmes ne sont pas conversationnels ; ils sont appelés par le LRC pour interroger une réalisation locale et déposer les données dans un tampon du LRC à l'aide d'un formal (cf. 2.4.).

3.2. Les mises à jour conversationnelles

3.2.1. Les programmes conversationnels P1

Les utilisateurs effectuent des mises à jour de manière conversationnelle c'est-à-dire qu'à partir d'un terminal conversationnel ils peuvent créer, modifier, supprimer un fournisseur ou un marché, ceux-ci étant représentés dans la base répartie par des réalisations-réseau des entités "fournisseur" ou "marché".

Pour réaliser ces mises à jour des programmes conversationnels de type P1 (cf. IV.2.5.) sont mis à la disposition des utilisateurs. Ce sont, comme les programmes d'interrogation conversationnelle, des programmes précompilés, accessibles depuis un terminal sur simple appel. Cependant ces programmes sont intégrés à des chaînes de programmes de mise à jour (cf. IV.2.5.) ; les utilisateurs ayant uniquement accès aux programmes P1 de ces chaînes.

Comme pour les interrogations, le site central possède ses programmes P1 propres, tandis que les sites régionaux possèdent tous les mêmes programmes P1 différents de ceux du site central.

3.2.2. Les chaînes de programmes de mise à jour

Pour créer, modifier, ou supprimer des réalisations-réseau de la base répartie nous avons défini différents types d'opérations possibles, chacune étant exécutée par une chaîne de programmes ; les différentes opérations ont été conçues en fonction du modèle étoilé (cf. IV.2.).

Selon l'application répartie qui utilise une base étoilée toutes ces opérations ne sont pas forcément mises à la disposition des utilisateurs. Dans ce paragraphe, nous considérons parmi les opérations définies, celles qui sont effectivement mises à la disposition des utilisateurs de la base "Fournisseur-Marché".

D'autre part, les opérations effectuées diffèrent selon qu'elles concernent l'entité-réseau "Fournisseur" ou l'entité-réseau "Marché". Pour cette raison, à chaque entité-réseau correspond ses propres chaînes de mise à jour.

3.2.2.1. Les chaînes de mise à jour de l'entité-réseau "Fournisseur"

L'entité-réseau "fournisseur" (cf. 1.1.) contient des données des trois types DCP, DRT, DCR. Pour respecter les principes et les définitions évoqués dans le chapitre III cela suppose que des accès à l'entité-réseau sont possibles sur le site central, pour des DCP, DCR, DRT et sur les sites régionaux pour les DCR et DRT.

Nous dirons que l'entité-réseau "Fournisseur" est essentiellement sous responsabilité régionale ; en effet, c'est un site régional qui décide de l' "introduction" d'un fournisseur, sur son axe, lorsqu'il passe un marché avec lui ; c'est également le site régional qui "retire" le fournisseur de son axe lorsqu'il ne passe plus de marché avec lui.

Pour introduire et retirer des fournisseurs de la base répartie, les sites régionaux disposent tous d'une chaîne régionale de création et d'une chaîne régionale de suppression. Ces chaînes permettent de créer ou de supprimer une réalisation sur un axe. Pour modifier les données d'une réalisation-réseau "fournisseur" il existe deux chaînes de modification : la chaîne centrale qui permet de modifier les DCP et certaines DRT ou DCR et la chaîne régionale qui permet de modifier des DRT et des DCR.

a) La chaîne de création régionale

- Le programme régional P1

Ce programme est conversationnel, il est appelé par un utilisateur régional qui désire introduire un fournisseur sur l'axe central-régional. L'utilisateur transmet au programme une donnée qui permet de vérifier si le fournisseur est déjà répertorié ou non sur l'axe. Si oui, la création est rejetée, sinon elle est acceptée

et le programme génère une réalisation locale et saisit les données fournies par l'utilisateur : des DCR et des DRT ; ce programme peut également introduire des données non saisies qui sont connues par le programme.

- Le programme central P2

Ce programme reçoit un ordre de création d'un site régional ainsi que des données provenant de P1. Sa première tâche est de vérifier si le fournisseur est déjà répertorié dans la base centrale.

- . Si oui, alors la création est une création secondaire, le fournisseur existe déjà sur, au moins, un autre axe. Dans ce cas les DRT reçues sont comparées à celles de la réalisation centrale. Les DCR reçues sont déposées dans la réalisation centrale.
- . Si non, la création est principale, le programme P2 génère une réalisation centrale et y dépose les données DCR et DRT reçues.

P2 peut introduire des DCR, DCP ou DRT qu'il calcule (par exemple le numéro attribué automatiquement au fournisseur) ou qui existent dans le programme ; ces données ne sont pas saisies. Si des DCP doivent être introduites par saisie alors elles le sont ultérieurement par la chaîne centrale de modification.

- Le programme régional P3

Ce programme reçoit les données DCR ou DRT introduites directement par P2 ; elles doivent compléter celles introduites par P1.

b) La chaîne de modification régionale

- Le programme régional P1

C'est un programme conversationnel qui, sur appel d'un utilisateur, saisit les modifications qui portent sur des DCR ou DRT : uniquement sur des données qui ont été introduites dans la base répartie par le programme P1 de création régionale.

- Le programme central P2

Ce programme reçoit de P1 des modifications portant sur des DCR ou DRT.

- Le programme régional P3

Ce programme s'exécute sur le site où s'est exécuté pour toutes les données, DCR ou DRT, modifiées en absolu (cf.IV.4.7.2.).

Il s'exécute également sur tous les sites régionaux qui "possèdent" le fournisseur modifié ; sur chacun de ces sites, un programme P3 exécute aussi bien les modifications absolues que les modifications relatives saisies par P1.

c) La chaîne de modification centrale

- Le programme central P1

Ce programme est conversationnel ; il permet d'introduire par saisie des données centrales (DCP) suite à une création régionale. Il permet aussi de "modifier" des DCP, DCR ou DRT qui doivent obligatoirement avoir été introduites soit par ce même programme soit par le programme central P2 de création régionale.

- Le programme régional P2

Ce programme existe sur tout site régional ; si les modifications effectuées par P1 concernent des DCR ou des

DRT alors celles-ci existent sur un ou plusieurs sites régionaux ; sur ce ou ces sites, P2 exécute alors les modifications transmises par le site central.

d) La chaîne régionale de suppression

- Le programme régional P1

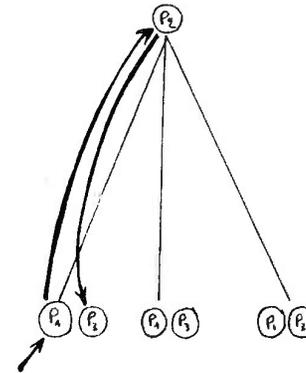
Il est appelé depuis un terminal et l'utilisateur lui transmet le numéro attribué au fournisseur ; le programme supprime alors la réalisation régionale.

- Le programme central P2

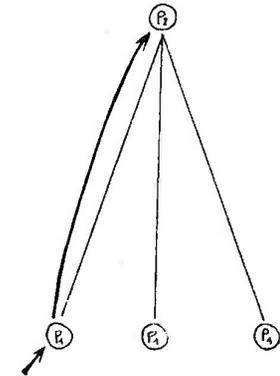
Ce programme reçoit, de P1, l'ordre de supprimer le fournisseur dont il reçoit le numéro.

Si le fournisseur n'est pas répertorié sur un autre axe, alors P2 supprime la réalisation centrale relative au fournisseur : c'est une suppression totale. Si par contre il existe sur un ou plusieurs autres axes, alors il supprime uniquement les données (les DCR et certaines DCP) relatives à l'axe ayant émis l'ordre de suppression ; dans ce cas, il s'agit d'une suppression partielle.

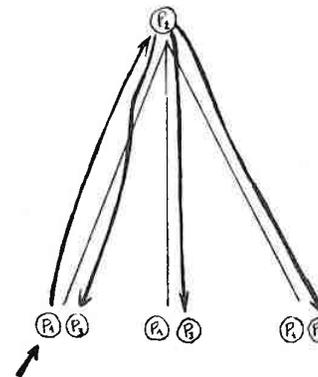
Remarque : Si un fournisseur est supprimé sur tous les axes par les sites régionaux et que le site central désire cependant garder des données DCP, DRT, DCR de la réalisation centrale alors le programme P2 doit déposer ces données dans une base "historique" des fournisseurs.



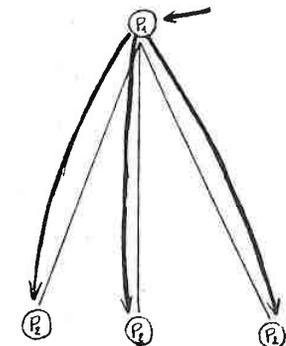
a) Création régionale



b) Suppression régionale



c) Modification régionale



d) Modification centrale

Figure VI.3. : Chaînes de programmes pour l'entité-réseau "Fournisseur"

3.2.2.2. Les chaînes de mise à jour de l'entité-réseau "Marché"

La même entité-réseau "Marché" permet de stocker dans la base répartie des données relatives à deux classes de marché : les marchés "régionaux" et les marchés "nationaux".

Un marché "régional" est propre à un axe unique ; une réalisation-réseau d'un tel marché se décompose en une réalisation centrale et en une seule réalisation régionale.

Le site régional qui passe un marché avec un fournisseur a l'initiative de créer et de supprimer la réalisation-réseau qui lui correspond. Pour un tel marché il n'y a pas de redondance entre les différents axes. Cependant certaines données sont représentées par des caractéristiques du type DRT. Celles-ci ont été définies au sens large pour le cas des marchés "nationaux" qui peuvent concerner plusieurs axes. Comme pour les fournisseurs, chaque site régional possède une chaîne de création et une chaîne de suppression de marché "régional".

Un marché "national" est imposé par le site central à un ou plusieurs axes sur lesquels le site régional traite avec un certain fournisseur ; un tel marché est répertorié sur plusieurs axes en général, voire tous les axes. Le site central a l'initiative de créer ou de supprimer un tel marché. Pour cela il dispose de deux chaînes de création centrale : une chaîne de création principale, utilisée par le site central lorsqu'il désire créer une réalisation-réseau pour un marché "national" ; une chaîne de création secondaire lorsqu'il désire étendre une réalisation-réseau à un axe supplémentaire : un site régional "entre" dans un certain marché national avec un certain fournisseur. Le site central dispose également d'une chaîne de suppression totale qui lui permet de supprimer toute une réalisation-réseau correspondant à un marché "national".

Pour les modifications, le site central et les sites régionaux disposent chacun d'une chaîne de modification qui leur permet de modifier des données qu'ils ont introduites dans la base répartie.

a) Chaîne de création régionale d'un marché "régional"

- Programme régional P1

Ce programme est conversationnel, il est chargé de créer une réalisation-réseau d'un marché passé avec un certain fournisseur.

L'utilisateur doit fournir au programme le numéro du fournisseur ce qui permet au programme de vérifier si le fournisseur est bien répertorié sur l'axe. Dans le cas contraire, la création du marché n'est pas possible. Le programme génère une réalisation locale et saisit les données fournies par l'utilisateur ; il peut également introduire des données qui sont connues par le programme. A la fin de cette création régionale, le programme vérifie si le même marché n'existe pas déjà dans la base auquel cas la nouvelle réalisation est détruite.

- Programme central P2

Celui-ci reçoit, de P1, l'ordre de créer une réalisation centrale et des données DCR et DRT. Il génère une réalisation et introduit les données reçues et celles dont il dispose. Il peut calculer certaines données (par exemple le numéro qu'il attribue au marché) et mettre à jour certaines données du fournisseur de ce marché. Si les modifications des données du fournisseur concernent des DCR ou DRT, alors ces modifications devront être exécutées dans une ou plusieurs bases régionales (si le fournisseur y est répertorié) par le programme P2 de la chaîne de modification centrale de l'entité "Fournisseur".

- Programme régional P3

Les données DCR, DRT introduites par P2 dans la réalisation "marché" sont transmises à P3 ; celui-ci complète alors la réalisation générée par P1.

b) Chaînes de suppression régionale d'un marché "régional"

- Programme régional P1

Ce programme reçoit l'ordre de supprimer une réalisation locale qui représente un certain marché régional.

- Programme central P2

Dans le cas d'un marché "régional", la suppression est totale puisqu'elle élimine entièrement la réalisation-réseau qui représente le marché. La réalisation-réseau n'intéresse qu'un seul axe.

Si le site central désire garder des données DCP, DCR ou DRT de la réalisation centrale alors P2 doit les déposer dans une base "historique des marchés".

c) Chaîne de création principale centrale d'un marché "national"

Cette chaîne permet d'introduire dans la base répartie un nouveau marché "national".

- Programme central P1

Grâce à ce programme conversationnel, l'utilisateur donne l'ordre de générer une réalisation-réseau pour un marché national sur tel ou tel axe.

Le programme génère une réalisation centrale et saisit les données DCP, DCR, DRT fournies par l'utilisateur ; il peut également introduire lui-même des données calculées ou non.

Puis il vérifie si le marché n'a pas déjà été répertorié auquel cas il serait directement supprimé.

Il envoie l'ordre de création à différents sites régionaux qui doivent être "intégrés" à ce marché "national"; il faut que le fournisseur soit répertorié sur chacun de ces axes sinon la création du marché y est impossible ; P1 vérifie donc l'existence du fournisseur sur les différents axes.

- Programme régional P2

Sur chaque site régional mis en cause par le marché "national" un programme P2 est chargé de générer une réalisation régionale et de stocker des DCR et DRT reçues de P1. P2 peut éventuellement introduire des données ; si elles sont des types DCR ou DRT, P2 devra être suivi d'un programme central P3. Si des données DRP ou DRR doivent être introduites par saisie dans les réalisations régionales alors elles le seront par les programmes P1 de modification régionale.

- Programme central P3

Ce programme est chargé de recevoir des programmes P2, des DCR ou DRT et de les intégrer dans la réalisation centrale générée par P1.

d) Chaîne de création secondaire d'un marché "national"

Si ultérieurement à une création principale, un marché "national" intéresse un nouveau site régional, alors une création secondaire permet d'étendre le marché à l'axe de ce nouveau site.

- Programme central P1

Ce programme conversationnel reçoit l'ordre d'étendre un certain marché "national" à un certain axe.

Le programme doit d'abord vérifier si le fournisseur est bien répertorié sur l'axe : vérification dans la réalisation centrale relative au fournisseur.

Le programme saisit des données DCR relatives à l'axe intéressé et complète la réalisation centrale du marché ; les données DCR, DRT sont transmises à un P2 régional.

- Programme régional P2

Ce programme est identique à celui de la chaîne précédente. Il reçoit de P1 les mêmes données et effectue les mêmes opérations.

- Programme central P3

Ce programme est identique à celui de la chaîne précédente.

e) Chaîne de suppression centrale d'un marché "national"

Cette chaîne permet de supprimer un marché "national" sur tous les axes sur lesquels il est répertorié.

- Programme central P1

Reçoit l'ordre de suppression et détruit la réalisation centrale ; puis transmet le même ordre aux sites régionaux qui possèdent ce marché.

- Programme régional P2

Supprime, à la suite d'un ordre de P1, la réalisation régionale relative à un marché national.

Ces programmes doivent éventuellement modifier des données du fournisseur du marché ; ces modifications si elles concernent des DCR ou DRT, DRR doivent être transmises. Elles sont exécutées par des programmes de type P2 ou P3 de modification d'un "Fournisseur".

f) Chaîne de modification centrale d'un marché

Cette chaîne permet d'introduire dans la base centrale, par saisie, des DCP ; elle permet également de modifier des DCP, DCR ou DRT introduites par un programme central.

- Programme central P1

Ce programme permet de modifier des données de la réalisation d'un marché qu'il soit "national" ou "régional"

Dans le cas "régional", les modifications des DCR ou DRT sont transmises, sur un seul axe, à un site régional.

Dans le cas "national", des modifications de DCR peuvent être transmises sur plusieurs axes ; les modifications d'une DRT doivent être transmises à tous les sites régionaux qui ont dans leur base une réalisation relative au marché national modifié par P1.

- Programme régional P2

Ce programme reçoit des modifications pour des DCR ou des DRT qu'il s'agisse d'un marché "national" ou d'un marché "régional".

g) Chaîne de modification régionale d'un marché

Cette chaîne permet d'introduire des DRP ou des DRR et de modifier des DRP, DRR, DCR ou DRT.

- Programme régional P1

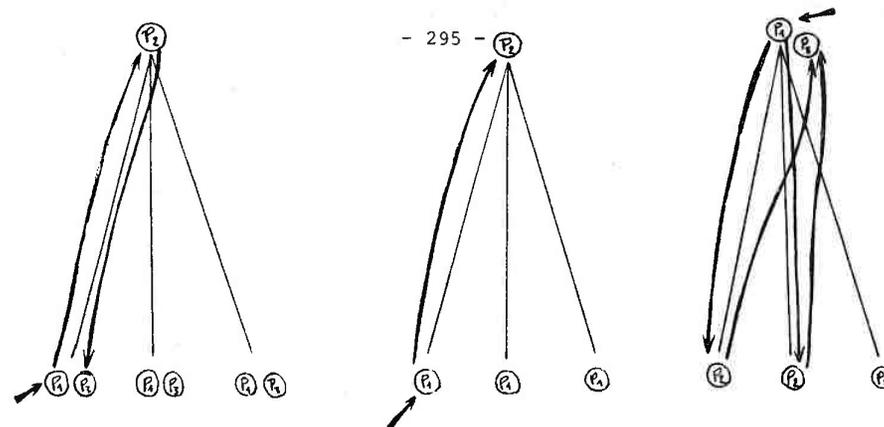
Ce programme introduit ou modifie des données dans la réalisation d'un marché qu'il soit "national" ou "régional". Il transmet les modifications de DCR ou DRT au programme central P2.

- Programme central P2

Modifie les DCR et DRT de la réalisation centrale ; s'il s'agit d'un marché "régional" alors les modifications absolues de DCR ou DRT sont renvoyées au site initial pour y être exécutées par P3 : s'il s'agit d'un marché "national" il renvoie au site initial les modifications absolues de DCR, mais il envoie les modifications des DRT à tous les sites régionaux possédant une réalisation du marché "national".

- Programme régional P3

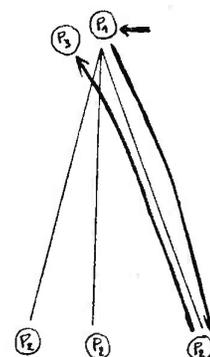
Est chargé d'exécuter des modifications absolues de DCR et des modifications absolues ou relatives de DRT.



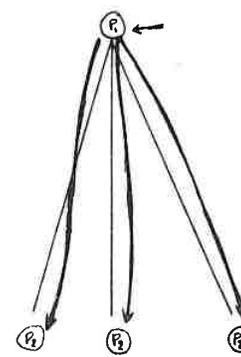
a) Création régionale d'un marché "régional"

b) Suppression régionale d'un marché "régional"

c) Création principale centrale d'un marché "national"



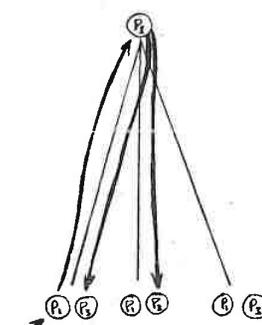
d) Création secondaire centrale d'un marché "national"



e) Suppression centrale d'un marché "national"



f) Modification centrale d'un marché



g) Modification régionale d'un marché

Figure VII.4. : Chânes de programmes pour l'entité-réseau "Marché"

CHAPITRE VIII

EXTENSIONS ET CONCLUSIONS

1. EXTENSIONS
2. CONCLUSIONS

1. EXTENSIONS

1.1. Extension du modèle de répartition étoilée

Dans le chapitre III nous avons présenté trois types d'organisation d'application répartie. Dans notre étude nous nous sommes intéressé au cas particulier d' "organisation hiérarchique" que constitue l'organisation étoilée.

L'ensemble de nos travaux peut en fait être étendu au cas plus général d'organisation hiérarchique ne se limitant pas à 2 niveaux mais constitué de n niveaux.

Pour ce type d'organisation nous pouvons concevoir une architecture de base répartie qui soit un assemblage de plusieurs architectures étoilées élémentaires. Cette architecture comporte des redondances "horizontales" entre sites de même niveau, des redondances "verticales" entre plusieurs niveaux et des redondances "triangulaires".

Dans le modèle étoilé nous considérons un site central et des sites régionaux ; dans un modèle hiérarchique plus général, les termes "central" et "régional" perdent leur signification ; tout site est un site local. Cependant pour localiser

les données, une numérotation des sites est nécessaire ; sur les figures nous avons représenté une numérotation possible.

Si nous supposons que tous les sites d'un même niveau sont équivalents dans leurs fonctions, alors le modèle peut être constitué des 4 types de répartition suivants :

- D.L.P. : donnée locale pure (non redondante)
- D.R.H.j : donnée à redondance horizontale sur le niveau j
- D.R.V.i, j : donnée à redondance verticale sur j niveaux, à partir du site i
- D.R.T.i, j : donnée à redondance triangulaire sur j niveaux, à partir du site i

Si, par contre, seuls les sites de même site "père" sont équivalents dans leurs fonctions alors le modèle est constitué des 4 types de répartition suivants :

- D.L.P. : donnée locale pure (non redondante)
- D.R.H.i : donnée à redondance horizontale et de site "père" i
- D.R.V.i, j : donnée à redondance verticale sur j niveaux, à partir du site i
- D.R.T.i : donnée à redondance triangulaire de site "père" i

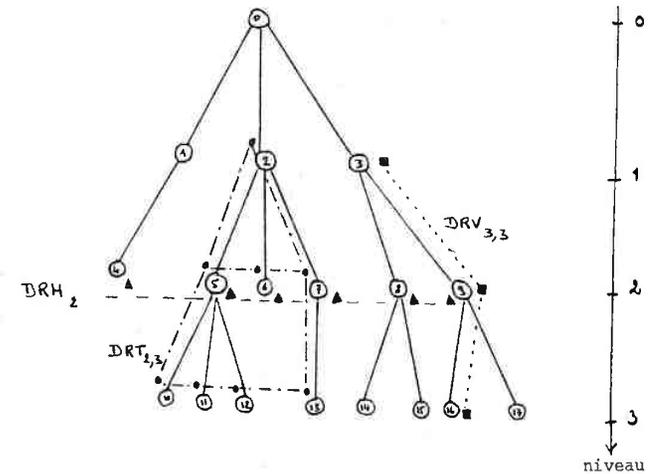


Figure VIII.1. : Modèle hiérarchique pour une organisation hiérarchique dont les sites d'un même niveau sont équivalents

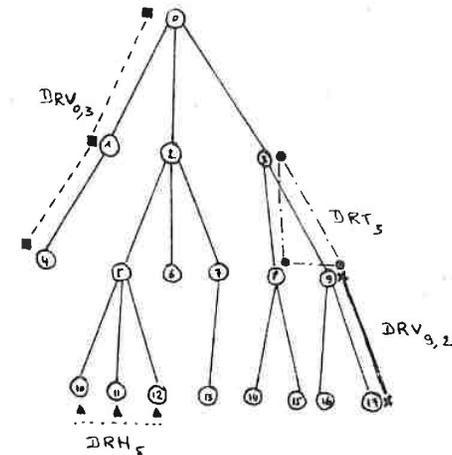


Figure VIII.2. : Modèle hiérarchique pour une organisation hiérarchique dont les sites de même site "père" sont équivalents

1.2. Intégration du LRC au SGBD

Une extension intéressante de nos travaux serait l'intégration des fonctions du logiciel de répartition et de cohérence dans le système de gestion SOCRATE. Ceci permettrait de faire communiquer directement plusieurs systèmes de gestion reliés entre eux par un réseau

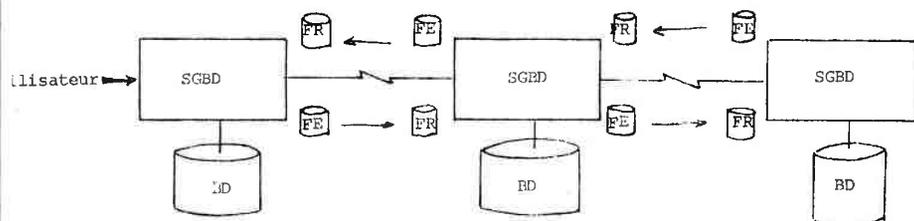


Figure VIII.3. : Communications entre systèmes de gestion

Les deux fonctions essentielles à intégrer seraient

- d'une part, la récupération directe des mises à jour effectuées par un utilisateur local pour les transmettre directement à un autre système de gestion, ou, pour les déposer dans un fichier FE.
- d'autre part, la réception de messages provenant de systèmes distants et destinés à créer des mises à jour sur la base locale.

Cette approche permettrait de ne plus limiter nos travaux à l'utilisation "presse-bouton" des bases de données ; en étant intégrées au système de gestion, les fonctions du LRC s'appliqueraient aussi bien aux programmes précompilés qu'aux requêtes conversationnelles de mise à jour.

Ce genre de communication entre systèmes de gestion fait actuellement l'objet de travaux par la société ECA-AUTOMATION (BDR-10), abstraction faite des problèmes de modèle de répartition et de données dupliquées.

1.3. Extension des notions d'action répartie et de chaîne de programme

Dans notre étude nous avons défini trois classes d'actions réparties chacune représentée par des chaînes de programmes ; la création répartie, la modification répartie et la suppression répartie.

D'autres actions réparties, moins associées aux accès à une base de données, pourraient être définies et caractériser un système d'informations réparties. A partir d'un événement externe constitué par l'accès d'un utilisateur au système, une action répartie pourrait être déclenchée ; celle-ci s'exprimerait par l'exécution d'actions élémentaires successives ou/et parallèles sur différents sites.

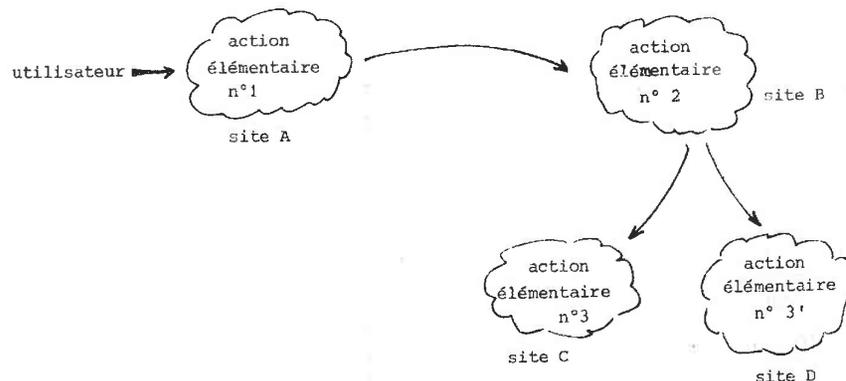


Figure VIII.4. : Notion de système d'informations réparties

2. CONCLUSIONS

2.1. Répartition et duplication des données

Le modèle de répartition que nous avons proposé introduit de nombreuses duplications de données ; ceci est essentiellement dû au fait que les sites ont des besoins communs, entre un site régional et le site central ou entre sites régionaux. Nous pouvons affirmer que le modèle proposé établit une répartition par duplication.

Cependant les duplications que nous introduisons se font au niveau de la donnée élémentaire et nos travaux s'identifient nullement avec certains travaux effectués sur la duplication et qui concernent des duplications de fichiers sur un réseau (COP-4, COP-5, COP-6, COP-7). En effet, ces travaux sont vus sous un aspect mathématique pour établir une fonction coût minimum d'utilisation d'un fichier sur un réseau ; cette fonction, à partir de certains paramètres, détermine si un fichier doit être dupliqué et si oui, en combien d'exemplaires et sur quels sites ; appliqués à une base de données, ces travaux permettent d'obtenir la duplication de la base entière. Or le but recherché dans notre étude, n'est pas de dupliquer une base mais de considérer une base initiale et de la répartir en dupliquant certaines données selon des critères d'organisation et non de coût.

Si on considère l'application du modèle sur l'exemple de la base "fournisseur-marché" (cf. VII.2.), on s'aperçoit que les types de répartition affectés aux caractéristiques sont les types DRT, DCP, DCR. Pratiquement aucune caractéristique n'est du type DRP ou du type DRR. Cela signifie qu'aucune donnée n'intéresse que les sites régionaux ; ceux-ci n'ont pas de données qui leur sont propres.

On peut interpréter ce "résultat", apparemment négatif, par le fait que les types de répartition ont été affectés dans l'état actuel de l'application "répartie" c'est-à-dire dans sa version centralisée. En effet, dans cette version, pratiquement tous les traitements sont exécutés sur le site central ; celui-ci a donc besoin de toutes les données : d'où le fait que celles-ci sont toutes au minimum centrales. Cela signifie qu'une répartition des données telle que nous la proposons doit évidemment s'accompagner d'une répartition des traitements ; ainsi, certains traitements ne seraient plus effectués sur le site central, mais sur les sites régionaux. La répartition des données ne constitue qu'une étape devant mener à une décentralisation des traitements et de ce fait à une décentralisation des fonctions de gestion qui sont traitées par informatique.

Exemple : si on considère la caractéristique "MOD-PAIEMENT" de l'entité "FOURNISSEUR" on s'aperçoit qu'elle est du type DRT ; si la fonction de paiement pouvait être décentralisée, la caractéristique "MOD-PAIEMENT" deviendrait du type DRP ainsi que toutes les caractéristiques de l'entité "MARCHE" associées au paiement telles que "PERIODICITE", "NB-PA-FAIRE", "NB-PA-FAITS"

Une meilleure répartition des fonctions et des traitements permettrait d'obtenir une meilleure répartition des données ; plus de données seraient "locales pures" c'est-à-dire soit centrales, soit régionales, et moins de données seraient dupliquées.

2.2. Coûts

2.2.1. Investissements

Nous avons vu (cf. VII.1.) que la mise en œuvre du modèle étoilé supposait un certain nombre de modifications matérielles ou logicielles sur la situation centralisée de l'application. Ces modifications concernent essentiellement les sites régionaux qu'il faut équiper de mini-ordinateurs devant supporter un logiciel de gestion de base de données ; d'où la nécessité d'un investissement important qui ne peut être justifié que par un coût de "fonctionnement réparti" meilleur que le coût de "fonctionnement centralisé" et par un meilleur "fonctionnement" de l'application répartie.

2.2.2. Coût de "fonctionnement réparti"

Une étude d'évaluation doit nous permettre prochainement à partir du cas concret "Fournisseur-Marché" d'établir un coût approximatif du fonctionnement réparti qui sera comparé au coût actuel de fonctionnement centralisé. Dans cette étude interviendront les paramètres liés à l'utilisation du réseau (temps d'"occupation" du réseau, nombre de bits transmis), au stockage des données et des programmes, aux temps d'exécution.

2.3. Validation des mécanismes proposés

L'ensemble des mécanismes proposés doit donner lieu à la réalisation d'une maquette locale utilisant les logiciels CYCLADES (logiciel-réseau) et SOCRATE (système de gestion de base de données). Cette maquette permettra la réalisation du logiciel de répartition et de cohérence et permettra de tester les mécanismes de mise en cohérence et de sécurité proposés.

Une étape ultérieure pourra consister à mettre cette maquette directement en œuvre, en vraie grandeur, sur le réseau CYCLADES. Cela permettrait d'évaluer les coûts de fonctionnement et de les comparer avec les résultats de l'évaluation "plus théorique" évoquée dans le paragraphe précédent ; cela permettrait également de disposer d'un "prototype" de base de données répartie .

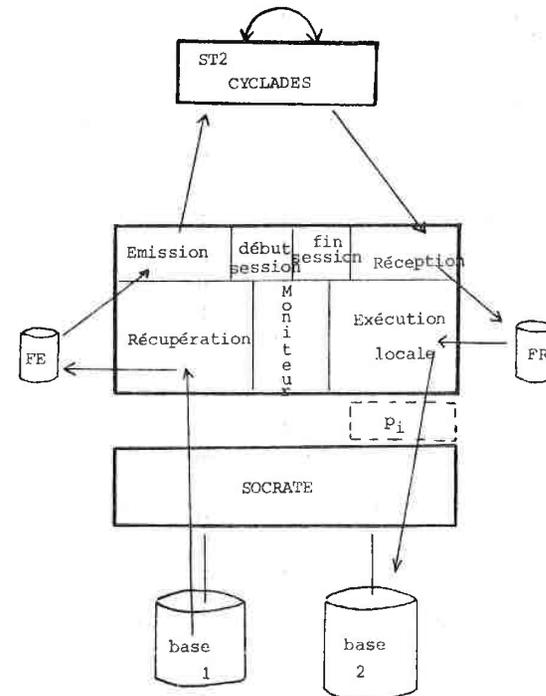


Figure VIII.5. : Maquette de réalisation du L.R.C.

BIBLIOGRAPHIE

1. RESEAUX
2. RESEAU CYCLADES
3. SYSTEMES DE GESTION DE BASES DE DONNEES
4. SYSTEME SOCRATE
5. BASES DE DONNEES REPARTIES
6. COPIES SUR UN RESEAU

RESEAUX

- RES-1 FRANK H. , I. FRISCH
"Testing the NASDAQ system - Traffic and response time"
Computer Communications, Networks and Teletraffic Conf.
Proc.,
Polytechnic Institute of Brooklyn, 1972, pp. 577-586.
- RES-2 BEERE M. and N. SULLIVAN
"TYMNET, a serendipitous evolution"
IEEE Trans. on Communications, vol. 20, n° 3, june 72,
pp. 511-515.
- RES-3 CROWTHER W. and al.
"Issues in Packet Switching Network Design"
National Comp. Conf., 1975, pp. 161-175.
- RES-4 Mc QUILLAN J.M. and al.
"Improvements in the design and performance of the ARPA
network",
Proc. of AFIPS, vol. 41, 1972, pp. 741-753.
- RES-5 HEART F. and al.
"The interface message processor for the ARPA network"
AFIPS Conf. Proc., Vol. 36, SJCC 1970.
- RES-6 CARR S. , CROCKER D. , CERF V.
"Host-host communication protocol in the ARPA network"
Proc. of AFIPS-SJCC, vol. 36, 1970, pp. 589-599
- RES-7 P.T.T.
"TRANSPAC-Réseau public de transmission de données par
paquets"
doc. commercial.

Réseau CYCLADES

- CYC-1 GRANGE J.L. et L. POUZIN
"CIGALE, la machine de commutation de paquets du réseau CYCLADES",
Congrès AFCET Rennes, Nov. 73, pp. 249-263.
- CYC-2 GARCIA C. et al.
"Spécifications de réalisation de la station de transport ST2 portable",
SCH 536.1 (bibliothèque CYCLADES à l'I.R.I.A.), juin 73,
20 pages.
- CYC-3 POUZIN L. et ZIMMERMANN H.
"Présentation du réseau CYCLADES - Introduction générale aux réseaux et à CYCLADES",
GAL 514 (bibliothèque CYCLADES à l'I.R.I.A.), février 74,
10 pages.
- CYC-4 ELIE M. et ZIMMERMANN H.
"Proposed standard HOST-HOST protocol heterogeneous computer network - Transport protocol",
SCH 519.1 (bibliothèque CYCLADES à l'I.R.I.A.), juin 74,
31 pages.

SYSTEMES de gestion de bases de données

- SYS-1 ROLLAND C.
"L'approche base de données : causes - caractéristiques - conséquences",
Université de Nancy II - M.I.A.G.E.
- SYS-2 DELOBEL Cl.
"Les systèmes de base de données",
Ecole d'été de Rabart, juillet 75, 200 pages.
- SYS-3 I.B.M.
"I.M.S./ 360 Applications Description Manual",
White Plains, N.Y., GH-20-07 65.
- SYS-4 "CODASYL, data base task group report",
A.C.M., New-York, 1971.
- SYS-5 "CODASYL, data description language",
Handbook 112, U.S. Department of Commerce, janvier 74.
- SYS-6 E. CODD
"A relational model of data for large shared data banks",
C.A.C.M., june 70.
- SYS-7 E. CODD
"Recent investigations in relational data base systems",
Information processing 74, North-Holland, 1974.

Système SOCRATE

- SOC-1 ABRIAL J.R. et col.
"Projet SOCRATE - Nouvelles spécifications",
Université de Grenoble, sept. 72.
- SOC-2 "Socrate sous SIRIS 7/SIRIS 8- Manuel d'opérations",
document C.I.I.-H.B. (référence 4553 E/FR).
- SOC-3 "Socrate SIRIS 7/SIRIS 8 - Manuel d'utilisation",
document C.I.I.-H.B. (référence 4338 E1/FR).

BASES DE DONNEES REPARTIES

- BDR-1 ROCHFELD A.
"Intérêts des bases de données réparties",
Journées "Bases de données réparties" de l'A.F.C.E.T. -
Mars 77.
- BDR-2 LE BIHAN J.
"La répartition des données dans les réseaux informatiques
A.F.C.E.T., Gif-sur-Yvette, nov. 76, 11 pages.
- BDR-3 LE BIHAN J.
"Projet-pilote sur les bases de données réparties",
Bibliothèque Projet SIRIUS à l'I.R.I.A.
- BDR-4 R.H. CANADAY and al.
"A back-end computer for data base management",
Bell Telephone Laboratories Inc., Piscataway, New Jersey
(octobre 74).
- BDR-5 ADIBA M. et col.
"Présentation générale d'un système de coopération de
bases de données réparties",
Laboratoire d'informatique, Université scientifique et
médicale de Grenoble, B.P. 53, 38041-Grenoble.
- BDR-6 GARDARIN G. et col.
"Analyse de la conception d'un système de gestion de
bases de données réparties",
Institut de Programmation, 4 Place Jussieu, 75230 Paris.
- BDR-7 DEMOLOMBE R. et LEMAITRE M.
"Rôles d'un modèle commun dans la conception et l'utili-
sation d'un S.G.B.D. réparti : analyse des principaux
modèles",
C.E.R.T., 2 av. Edouard Belin, 31055 Toulouse.

- BDR-8 H. KARL
"The distributed data bases of the information system of
the german police (I.N.P.O.L.)",
European Computer Workshop series "Distributed Computer
systems", Darmstadt, 1974.
- BDR-9 JOUVE M.
"Etude des prob èmes de fiabilité (permanence aux pannes)
dans un S.G.B.D. réparti",
Université de PARIS VI - Place Jussieu 75230-PARIS.
- BDR-10 ECA-AUTOMATION
"SOCRATE/CYCLADES - Manuel d'opération et d'utilisation
Mai 1975.

COPIES sur un réseau

- COP-1 MEYER D.
"Cohérence et Algorithmes de Synchronisation des accès
à des copies",
Université de Nancy I - U.E.R. de Mathématiques.
- COP-2 HOLLER E.
"Files in computer networks ",
Congrès "Réseaux d'ordinateurs", Arles 73.
- COP-3 RINJONNEAU F.
"Etude des mécanismes de gestion d'une base de données
à haut degré de fiabilité, répartie et dupliquée sur un
réseau d'ordinateurs",
C.E.R.T., 2 av. Edouard Belin, Toulouse.
- COP-4 MIRANDA S.
"Systèmes d'accès interactif à des fichiers et copies
de fichiers répartis sur un réseau d'ordinateurs",
Université des Sciences Sociales de Toulouse.
- COP-5 K. DAN LEWIN and H.L. MORGAN
"Optimizing distributed data bases - A frameworle for
research",
The Wharton School - Philadelphia, Pemsylvania.
- COP-6 W.W. CHU
"Optimal file allocation in a multiple computer system",
IEEE transactions on computers, oct. 69.
- COP-7 R.G. CASEY
"Allocation of copies of a file in an information
network",
I.B.M. Research Laboratory, San Jose, California.

SOMMAIRE

	page
<u>AVANT-PROPOS</u>	1
<u>CHAPITRE I : INTRODUCTION AUX BASES DE DONNES</u> <u>REPARTIES</u>	5
1 - <u>LES RESEAUX</u>	8
1.1 - Les réseaux de terminaux	8
1.2 - Les réseaux spécialisés	11
1.3 - Les réseaux généraux	12
2 - <u>LES BASES DE DONNEES</u>	18
2.1 - Généralités	18
2.1.1 - Base de données et sys- tème de gestion	18
2.1.2 - Fonctions d'un système de gestion	19
2.1.3 - Fontionnement et utili- sation	20
2.2 - Les modèles de description	21
2.2.1 - Le modèle arborescent ..	21
2.2.2 - Le modèle réseau	21
2.2.3 - Le modèle relationnel ..	23
2.3 - Le système de gestion SOCRATE ..	24
2.3.1 - Structures de données ..	25
2.3.2 - Exemple d'une structure données	27

2.3.3 - Espace virtuel	28	4.3 - Les accès à la base	59
2.3.4 - Espace réel	32	4.3.1 - Interrogation de la base centrale	60
2.3.5 - Correspondance entre espace virtuel et espace réel	33	4.3.2 - Mise à jour de la base centrale	60
3 - <u>RESEAUX ET BASES DE DONNES</u>	35	4.4 - Les traitements	62
3.1 - Situation actuelle : Base centralisée et utilisateurs répartis	35	4.5 - Les fichiers de données régionales	62
3.2 - Possibilités futures : Utilisateurs, systèmes, bases répartis	36	5 - <u>INCONVENIENTS DE LA CENTRALISATION</u> ...	63
3.3 - Possibilités actuelles : Ensembles "système + base" répartis	38	5.1 - Résistance psychologique des utilisateurs à la centralisation	63
<u>CHAPITRE II : DESCRIPTION D'UNE APPLICATION REPARTIE</u>	41	5.2 - Dégradation du fonctionnement de l'application	63
1 - <u>ORGANISATION DE LA GESTION COMPTABLE ET FINANCIERE DU C.N.R.S.</u>	44	5.2.1 - Sécurité	64
2 - <u>LE RESEAU DE COMMUNICATION</u>	47	5.2.2 - Saturation du site central	64
2.1 - Structure logique du réseau	47	5.2.3 - Charge du réseau	64
2.2 - Configuration du site central ..	48	5.2.4 - Dégradation des temps de réponse	65
2.3 - Configuration des sites régionaux	49	5.3 - Manque de cohérence des données	65
2.4 - Le support de transmission	50	5.3.1 - Incohérence des actions sur la base centrale ..	65
3 - <u>LA BASE DE DONNEES</u>	52	5.3.2 - Incohérence entre les données des fichiers régionaux	65
4 - <u>UTILISATION CENTRALISEE DE LA BASE</u> ...	58	<u>CHAPITRE III : LE MODELE DE REPARTITION ETOILEE</u> .	67
4.1 - Utilisation centralisée : généralités	58	1 - <u>APPLICATIONS ET BASES DE DONNEES REPARTIES</u>	70
4.2 - Manipulation des données de la base	58	1.1 - Application répartie	70
		1.1.1 - Généralités	70

1.1.2 - Organisations d'applications réparties	73
1.1.3 - Organisation étoilée ...	76
1.2 - Base de données répartie	77
1.2.1 - Architecture externe ...	77
1.2.2 - Architecture interne ...	78
1.2.2.1 - Répartition des entités et des caractéristiques	78
1.2.2.2 - Répartition des données ..	82
1.3 - Notion de modèle de répartition	86
2 - <u>PRINCIPES DE REPARTITION ETOILEE</u>	87
2.1 - Autonomie des sites	87
2.1.1 - Indépendance des sites régionaux par rapport au site central	87
2.1.2 - Indépendance des sites régionaux entre eux	87
2.2 - Répartition des logiciels d'accès aux données ; accès décentralisé	89
2.3 - Répartition des données	91
2.3.1 - Notion d'axe de répartition	91
2.3.2 - Répartition des données sur un axe	92
2.3.3 - Répartition sur plusieurs axes	94

2.4 - Base centrale et bases régionales	94
2.5 - Répartition statique	94
3 - <u>EXPRESSION ET UTILISATION DU MODELE ETOILE</u>	96
3.1 - Expression du modèle	96
3.1.1 - Expression dans la répartition logique	96
3.1.2 - Expression dans la répartition physique	98
3.1.3 - Types de données réparties	100
3.2 - Utilisation du modèle	101
3.2.1 - Attribution de types de répartition aux caractéristiques d'entités	101
3.2.2 - Définition des cinq types affectés aux caractéristiques	102
3.2.3 - Attribution des types aux références, anneaux, chaînes inverses	110
3.2.4 - Notion de logiciel et de table de répartition	115
3.2.5 - Reconnaissance des réalisations d'objet-réseau	117
3.3 - Exemple d'utilisation du modèle étoilé	118
4 - <u>ARCHITECTURE DE LA BASE REPARTIE ETOILEE</u>	122

<u>CHAPITRE IV : MISE A JOUR ET COHERENCE DE LA BASE</u>	
<u>REPARTIE</u>	125
1 - <u>COHERENCE LOCALE ET COHERENCE GLOBALE</u>	129
1.1 - Cohérence locale	129
1.2 - Cohérence globale	130
1.3 - Logiciel de cohérence	132
1.4 - Cohérence de la base répartie ..	133
2 - <u>MISES A JOUR DE LA BASE REPARTIE</u>	134
2.1 - Définition	134
2.2 - La création	135
2.2.1 - Création principale et créations secondaires ..	135
2.2.2 - Données saisies et données calculées	136
2.2.3 - Répartition d'une création	137
2.3 - La modification	141
2.3.1 - Modification centrale ..	141
2.3.2 - Modification régionale ..	142
2.4 - La suppression	143
2.4.1 - Suppression totale et suppression partielle ..	144
2.4.2 - Répartition de la suppression	146
2.5 - Les programmes de mise à jour ..	147
2.5.1 - Définition d'une chaîne de programmes	147
2.5.2 - Les programmes Pi d'une chaîne	149

2.6 - Actions réparties et actions élémentaires	150
2.6.1 - Définitions	150
2.6.2 - Durée d'une action répartie	152
2.6.3 - Action répartie et incohérences globales	154
3 - <u>NIVEAU DE COHERENCE DE LA BASE REPARTIE</u>	156
3.1 - Mécanisme synchronisé	156
3.2 - Mécanisme non synchronisé	160
3.3 - Choix d'un mécanisme	162
3.4 - Notion de "niveau de cohérence"	163
4 - <u>MECANISME DE COHERENCE GLOBALE PROPOSE</u>	165
4.1 - Choix d'un niveau de cohérence ..	165
4.2 - Mécanisme centralisé	165
4.3 - Sessions de travail sur la base répartie	166
4.4 - Messages de mise en cohérence ..	168
4.5 - Fichier émission et fichier réception	168
4.6 - Sessions de mise en cohérence globale	171
4.6.1 - Définition	171
4.6.2 - Début d'une session	171
4.6.3 - Transferts et exécutions locales	174
4.6.4 - Fin d'une session	177

4.7 - Concurrence entre actions réparties	178
4.7.1 - Modification relative ..	178
4.7.2 - Modification absolue ...	181
4.7.3 - Créations régionales ...	185
CHAPITRE V : LE LOGICIEL DE REPARTITION ET DE COHERENCE	187
1 - DEFINITION	190
1.1 - Logiciel de répartition et logiciel de cohérence	190
1.2 - Contexte local et contexte global	190
1.3 - Interface étoilé	191
2 - ENVIRONNEMENT ET ARCHITECTURE EXTERNE	193
2.1 - Actions réparties et L.R.C.	193
2.2 - Interface contexte local → contextes locaux distants	194
2.3 - Interface contextes locaux distants → contexte local	195
2.4 - Architecture externe du L.R.C. .	196
3 - FONCTIONS DU L.R.C.	198
3.1 - La récupération des mises à jour locales	198
3.1.1 - Les renseignements de récupération	198
3.1.1.1 - Définition de l'entité MAJ	198
3.1.1.2 - Structure de l'entité MAJ régionale	199
3.1.1.3 - Structure de l'entité MAJ centrale	203

3.1.2 - Les programmes de mise à jour	204
3.1.3 - Récupération par le L.R.C.	208
3.2 - Le protocole d'échange L.R.C.-L.R.C.	210
3.2.1 - Définition	210
3.2.2 - L'entête du message	211
3.2.3 - Le corps du message	212
3.3 - Exécution des programmes de type P ₂ , P ₃	215
3.3.1 - Lancement d'un programme	215
3.3.2 - Communication L.R.C.-Programme	216
3.3.3 - Communication Programme-L.R.C.	219
3.4 - Fonctions propres au L.R.C. central : cohérence globale des DRT, DCR, DRR	219
3.4.1 - Cas des modifications ...	220
3.4.2 - Cas des créations	221
3.4.3 - Le recensement des modifications absolues	223
3.5 - Fonction propre aux L.R.C. régionaux : cohérence globale des DRR	225
3.6 - Gestion des fichiers FE et FR ...	226
4 - LA TABLE DE REPARTITION	227
4.1 - Définition	227
4.2 - Structure de la table	228

4.3 - Stockage et utilisation	231
4.4 - Création de la table	233
5 - <u>ARCHITECTURE INTERNE DU L.R.C.</u>	234
5.1 - La tâche L.R.C.	234
5.2 - Le module "interface réseau" ...	235
5.3 - Le module "récupération des mises à jour"	237
5.4 - Le module "exécution"	238
5.5 - Le module "moniteur"	239
<u>CHAPITRE VI : SECURITE DE LA BASE REPARTIE</u>	241
1 - <u>DEFINITION DE LA SECURITE POUR LA BASE REPARTIE</u>	244
2 - <u>INACTIVITE D'UN SITE</u>	245
2.1 - Inactivité prolongée	245
2.2 - Inactivité pendant la session de mise en cohérence	246
3 - <u>PANNE DE COMMUNICATION</u>	247
3.1 - Impossibilité d'établir une liaison	247
3.2 - Liaison interrompue en cours de session	248
4 - <u>PANNE LOCALE AU COURS D'UNE SESSION DE MISE EN COHERENCE</u>	249
4.1 - Définition	249
4.2 - Session locale de reprise	249
4.3 - Points de contrôle du L.R.C. ...	250

4.4 - Principes d'une reprise	251
4.4.1 - Principe général	251
4.4.2 - Connexion au site symétrique	253
4.4.3 - Vérification de l'état du fichier "réception" local	253
4.4.4 - Vérification de l'état du fichier "réception" distant	254
4.4.5 - Exécution des messages de FR et récupération dans l'entité MAJ	254
4.4.6 - Fin de la procédure de reprise	254
4.5 - Point de contrôle en début de session de mise en cohérence ...	255

CHAPITRE VII : MISE EN OEUVRE DU MODELE ETOILE ... 257

1 - <u>INCIDENCE DE LA REPARTITION ETOILEE SUR L'APPLICATION</u>	260
1.1 - Incidences logicielles	260
1.1.1 - Les logiciels-réseau ...	260
1.1.2 - Les systèmes de gestion de base de données	260
1.1.3 - Les logiciels de répartition et de cohérence .	261
1.1.4 - Les programmes d'accès aux données	261
1.2 - Incidences matérielles	261
1.2.1 - Incidences sur le réseau	261
1.2.2 - Incidences sur les sites régionaux	264
1.2.3 - Incidences sur le site central	265

1.3 - Incidences sur les données	266
1.3.1 - Modification des entités initiales	266
1.3.2 - Introduction d'une entité MAJ	266
1.3.3 - Introduction de formals.	266
1.3.4 - La table de répartition.	267
1.3.5 - Les fichiers FE et FR ..	267
1.4 - Incidences sur l'administration de l'application répartie	267
2 - <u>LA BASE REPARTIE "FOURNISSEUR-MARCHE"</u>	270
2.1 - Les entités-réseau "Fournisseur" et "Marché"	270
2.2 - Les entités centrales et régionales "Fournisseur" et "Marché" ..	273
2.2.1 - Descriptions centrales .	273
2.2.2 - Descriptions régionales.	275
2.3 - La table de répartition	276
2.4 - Les formals	278
2.4.1 - Les formals centraux ..	279
2.4.2 - Les formals régionaux .	280
3 - <u>UTILISATION DE LA BASE REPARTIE "FOURNISSEUR-MARCHE"</u>	281
3.1 - Les interrogations conversationnelles	281
3.2 - Les mises à jour conversationnelles	281
3.2.1 - Les programmes conversationnels P ₁	281

3.2.2 - Les chaînes de programmes de mise à jour	282
3.2.2.1 - Les chaînes de mise à jour de l'entité-réseau "Fournisseur" ..	283
3.2.2.2 - Les chaînes de mise à jour de l'entité-réseau "Marché"	288

CHAPITRE VIII : EXTENSIONS ET CONCLUSIONS

1 - <u>EXTENSIONS</u>	299
1.1 - Extension du modèle de répartition étoilée	299
1.2 - Intégration du L.R.C. au SGBD ...	302
1.3 - Extension des notions d'action répartie et de chaîne de programmes	303
2 - <u>CONCLUSIONS</u>	304
2.1 - Répartition et duplication des données	304
2.2 - Coûts	306
2.2.1 - Investissements	306
2.2.2 - Coût de "fonctionnement réparti"	306
2.3 - Validation des mécanismes proposés	306

BIBLIOGRAPHIE



309