

78/82

Sc N 78 / 60 A

**PROJET REMORA :**  
 Modèle de représentation et méthode  
 de construction des composantes statiques d'un S.I.



**THESE**  
 POUR L'OBTENTION DU  
 DOCTORAT DE 3<sup>e</sup> CYCLE  
 INFORMATIQUE, SOUTENUE  
 LE 28 JUIN 1978 PAR  
**Monique KRIEQUER.**

Jury : Mme C. ROLLAND ..... Président  
 MM. G. BENCI  
 J.-C. DERNIAME  
 Mme O. FOUCAUT  
 M. J. MAROLDT } Examineurs

BIBLIOTHEQUE SCIENCES NANCY 1  
  
 D 095 180555 6

**PROJET REMORA :**  
**Modèle de représentation et méthode**  
**de construction des composantes statiques d'un S.I.**

**THESE**

POUR L'OBTENTION DU  
DOCTORAT DE 3<sup>e</sup> CYCLE  
INFORMATIQUE, SOUTENUE  
LE 28 JUIN 1978 PAR  
**Monique KRIEQUER.**

Jury : Mme C. ROLLAND ..... Président  
MM. G. BENCI  
J.-C. DERNIAME  
Mme O. FOUCAUT  
M. J. MAROLDT } Examineurs

## AVANT - P R O P O S

Ce travail de recherche, mené dans le cadre du projet Remora ne peut être un travail strictement personnel, il est le fruit d'une équipe dans le sens le plus large du terme.

Au moment de le présenter, il m'est agréable de remercier tous ceux qui, par leurs conseils, leurs critiques, leur patience et leur présence ont très fortement contribué à sa réalisation.

Ma reconnaissance va également aux membres du Jury qui acceptent aujourd'hui de juger ce travail après en avoir suivi l'élaboration. Par leurs conseils avisés, ils ont su me guider dans cette tâche.

Qu'ils veuillent bien accepter l'expression de ma profonde gratitude.

Je remercie aussi tous ceux qui ont contribué à la réalisation pratique de ce travail.

CHAPITRE I

L'ETUDE DANS LE CADRE DU

PROJET REMORA

Cette étude s'insère dans le cadre du projet REMORA dont nous rappelons l'objectif, les hypothèses de travail et les résultats acquis. Nous la situons dans les propositions faites pour la conception du plus haut niveau (niveau conceptuel).

#### I - L'OBJECTIF DU PROJET REMORA ET SA JUSTIFICATION

L'objectif du projet est de définir et de réaliser un système de pilotage pour la conception et la réalisation assistée des systèmes d'information (SI) dans les organisations.

Cet objectif est justifié par l'absence de moyens satisfaisants pour concevoir et construire le SI. En fait il existe aujourd'hui deux types de conception de SI qui aboutissent à deux classes de SI :

- ceux qui sont conçus à partir d'une analyse ponctuelle et localisée dans le temps d'un problème, par transposition et replâtrages successifs des solutions manuelles. Ces SI sont des bibliothèques de chaînes de programmes inflexibles et incapables de renseigner correctement les gestionnaires.
- ceux qui sont conçus et construits en deux temps : celui d'une base de donnée, analysée de manière intégrée et réellement structurée, celui de chaînes de programmes conçues et réalisées par adjonctions successives au rythme des besoins. Cette approche assimile le SI à une base de données ce qui est incomplet, car alors le SI n'est pas l'image de l'organisation dans son ensemble, mais seulement celle de ses constituants.

De l'analyse et de la critique de ces méthodes découle notre définition du SI qui constitue notre première hypothèse de travail.

## 2-LES HYPOTHESES DE TRAVAIL DU PROJET

Hypothèse 1 - Le SI est un ensemble organisé de données, de programmes et de commandes dynamiques.

. Le SI est un objet artificiel dont l'objectif est d'informer, c'est-à-dire de restituer à ceux qui l'utilisent des images de la réalité sur des situations présentes, passées ou futures, ponctuelles ou panoramiques qu'ils n'ont pu observer directement et sur lesquelles ils ont à faire des analyses pour prendre des décisions.

. Pour assurer cet objectif le SI doit être une représentation correcte de l'organisation en fonctionnement ; correcte signifiant

fidèle (traduction sans biais et sans déformation), complète (de telle sorte que chaque utilisateur retrouve dans la représentation le point de vue qui l'intéresse), cohérente (sans contradiction ni ambiguïté) ; représenter l'organisation dans toutes ses dimensions statiques et dynamiques, c'est représenter ses constituants et leur structure ; c'est aussi représenter ses fonctions et leur interaction sur les constituants ; c'est enfin représenter les conditions dynamiques de l'exécution des fonctions ; c'est à ce prix que le SI peut être une image exacte de l'organisation à tout instant de son fonctionnement.

. Il faut donc que le SI assure trois fonctions :

- la fonction mémorisation si l'on admet que la représentation de la partie statique de l'organisation peut se faire par des données structurées ;
- la fonction traitement si l'on représente les fonctions par des instructions ordonnées ;
- la fonction commande pour qu'il porte en lui les moyens de commander son évolution dans le temps, à l'image de la dynamique de l'organisation.

De ce raisonnement découle notre choix de construire le SI sous forme d'une base de données, d'une base de programmes et d'une structure d'automate dynamique.

Corollairement nous pensons que la conception des trois composantes interdépendantes du SI doit se faire de manière simultanée et par des moyens homogènes.

L'expérimentation des premiers résultats acquis nous a en plus convaincus qu'il fallait donner la primauté à la conception en mettant l'accent sur l'analyse et la représentation de la dynamique de l'organisation.

Hypothèse 2 - Le processus de conception-réalisation est en trois étapes : conceptuelle, logique, physique.

. Par concevoir, nous entendons comme en (17), définir dans sa totalité, la solution technologique et administrative qui va permettre de réaliser le SI.

Pour cela, on part des besoins exprimés par les gestionnaires, on les traduit en fonctions que devra remplir le SI et on définit les moyens pour assurer ces fonctions de telle sorte que, par intégration, ils constituent un système.

. Réaliser le SI c'est créer les moyens, les mettre en place et les rendre aptes à fonctionner. C'est vérifier la qualité du SI par des mesures de ses performances, son adéquation aux objectifs qui lui ont été assignés et assurer son insertion dans l'organisation.

Concevoir le SI est une tâche difficile, complexe qui nécessite que l'on se réfère à une perception d'ensemble de l'organisation et qui implique la prise en compte opportune d'un grand nombre de données et de paramètres. Pour réduire les erreurs de conception-réalisation, il est utile de procéder par étapes et de définir à chaque étape la classe de problèmes qui faut résoudre.

L'analyse diachronique des méthodes actuelles de conception des SI nous conduit à définir un processus de conception en trois étapes successives qui se complètent.

En effet, les méthodes d'analyse (30), (4), (21), (24), ont montré l'intérêt qu'il pourrait y avoir à dissocier l'analyse des problèmes logiques ou fonctionnels de celle des problèmes physiques ou organiques.

Les méthodes de conception des bases de données (26), (3), ont démontré l'erreur qu'avaient commise les approches précédentes en ignorant les problèmes de représentation ou conceptuels.

. L'étape conceptuelle est, dans REMORA, celle de la représentation des phénomènes pris en compte dans le SI dans tous leurs aspects statiques et dynamiques. Elle correspond, d'une part, à l'énoncé des besoins exprimés par les gestionnaires, en termes de résultats attendus, d'autre part, à la définition, à partir de ces énoncés et par des transformations successives, d'une structure conceptuelle abstraite.

. L'étape logique est celle de la prise en compte de l'usage que l'on prévoit de faire du SI. Elle correspond à la structuration logique des données, c'est-à-dire à la définition de la structure d'accès à la base de données, à la structure logique des traitements, c'est-à-dire à la définition de la structure spatiale et temporelle d'exécution des traitements.

. L'étape physique est celle de la prise en compte de l'exploitation que l'on prévoit de faire du SI. Elle correspond à la prise en compte des moyens physiques pour implémenter les structures de données, pour réaliser les structures de traitements et gérer l'automate de fonctionnement.

### Hypothèse 3 - Le processus de conception-réalisation est assisté

Le nombre et la complexité des problèmes à résoudre dans la mise en place des SI a conduit les concepteurs à se demander dans quelle mesure un automate ne pourrait pas les aider, voire les suppléer dans le contrôle de l'ordonnancement des problèmes et de la prise en compte des solutions.

Cette solution, sans doute la plus séduisante, a été retenue dans le projet ISDOS ( 28 ) et dans SCAPFACE ( 20 ). Cependant la réduction des objectifs opérée peu à peu dans ISDOS incite à la prudence. De notre point de vue, trop de problèmes dans le processus de conception réalisation des SI restent non résolus, pour qu'un automate puisse conduire le processus de bout en bout.

Aussi comme Waters ( 31 ), Couger ( 10 ), Peccoud ( 23 ), avons-nous choisi la solution d'une conception assistée.

La conception assistée organisée autour du couple (automate, individu) permet de combiner avantageusement les capacités de résolution heuristique des individus avec les solutions que génère l'automate pour les problèmes qui ont reçu une réponse à la fois théorique et pratique. L'inventaire que nous avons fait de l'ensemble des problèmes qu'il faut traiter dans la conception des SI et de ceux que nous savons désormais résoudre nous a conduit à décider de ce qui peut être confié à l'automate et ce qui reste du ressort du concepteur.

Dans le travail de conception assistée, pour traiter chaque problème, il y a un va et vient permanent et compliqué entre les fonctions remplies par le concepteur et celles qu'assure l'automate. La recherche que nous menons nous a permis de déterminer dans quelles conditions et selon quelle séquence chacune de ces fonctions doit être assurée.

Nous avons retenu, dans notre proposition de conception assistée, de faire assurer et contrôler le déroulement efficace et l'enchaînement des fonctions par un outil que nous désignons PILOTE et qui assure la coordination dans l'exécution des fonctions de la conception assistée du SI. Le Pilote apparaît bien ainsi comme une partie du système de conception assistée.

Nous sommes partis de l'idée que dans la conception assistée d'un SI se trouvaient, à tour de rôle, impliqués pour remplir des fonctions bien définies, un concepteur, et un ensemble de moyens constituant le système de conception assistée.

Le Pilotage consiste alors à régler ou à contrôler les interventions appropriées de ces moyens :

- à gérer la dynamique des moyens de façon qu'en cours d'usage, le système de conception assure efficacement ses fonctions et conserve sa cohérence,
- à gérer les outils qui assurent les fonctions,
- à gérer les communications entre le concepteur et l'automate.

Le PILOTE est le système qui assure ces trois fonctions.

Hypothèse 4 - Les moyens dont doit disposer le concepteur à chacune des étapes du processus sont ceux du quartet (modèles, langages, méthodes, outils)

Cette hypothèse que nous avons émise au départ a été confirmée par les travaux du groupe INFORSID (17).

A priori les méthodes et les outils sont les deux classes de ressources dont a nécessairement besoin tout concepteur.

Par méthode nous entendons tout processus opératoire, qui, agissant à l'image d'un manuel d'utilisation précise, à chacune des étapes du processus de conception réalisation, les entrées utilisées, la répartition des tâches entre les hommes et les automates, les sorties qu'il faut produire et la manière d'y aboutir.

Les outils sont des processus opératoires qui fournissent automatiquement une solution aux problèmes rencontrés dans le déroulement de la méthode.

Cependant, comme l'usage des méthodes d'analyse l'a mis en évidence, une méthode ne peut être réellement satisfaisante si elle ne s'appuie pas sur des concepts précis. Cela est encore plus vrai en matière de conception des SI : concevoir c'est avant tout représenter des phénomènes du monde réel ce qui implique que l'on dispose d'un modèle.

L'approche base de données a confirmé l'intérêt de ce type de ressource que nous considérons, dans l'état actuel de la connaissance, comme indispensable.

Un modèle est un ensemble de concepts et de règles pour les manipuler, permettant de représenter les phénomènes réels.

Les langages sont indispensables pour formuler le résultat de la modélisation.

### 3 - LES DIRECTIONS DE RECHERCHES ET LES RESULTATS ACQUIS

Les quatre hypothèses précédemment citées nous permettent de définir les directions de recherches du projet.

- a) Il s'agit, d'abord, pour chacune des composantes du SI, et à chaque niveau du processus de conception-réalisation, de définir le modèle, la méthode, le langage et les outils appropriés (figure 1).

COMPOSANTS	DONNEES	TRAITEMENTS	COMANDES DYNAMIQUES
C		(1)	
O			
N			
C			
E	LOGIQUE	MODELES	
P		METHODES	LANGAGES
T			OUTILS
I	PHYSIQUE	(2)	
O			
N			
REALISATION			

Figure 1 - Objectifs de recherche

b) Il s'agit, ensuite, de définir le PILOTAGE, de concevoir et réaliser le PILOTE.

Les résultats acquis sont de deux types :

a) Un ensemble de propositions complètes et cohérentes pour la conception du premier niveau (niveau conceptuel) dans lequel s'intègre cette étude et qui comporte d'une part les moyens pour assurer la perception des phénomènes réels de l'organisation et leur description dans un schéma de perception, d'autre part, les moyens de construire le schéma conceptuel du SI à partir de la description du réel perçu ((1)).

Plus précisément, nous avons défini, pour assurer la perception

- un modèle descriptif du réel,
- un langage d'énoncés des problèmes de gestion
- une méthode de perception des problèmes
- des outils de vérification de la cohérence du schéma de perception.

On trouvera une présentation complète de ces propositions dans les thèses de Lamy (19), Perea (23), Thiery (29), Chesseron (7).

Nous avons défini, pour assurer la cohérence du schéma conceptuel à partir du schéma de perception,

- un modèle abstrait de représentation des phénomènes réels,
- un langage de description du schéma conceptuel,
- une méthode et des outils de construction du schéma conceptuel à partir du schéma de perception.

Ces propositions sont en partie présentées dans cette étude, en partie dans la thèse de Chesseron.

b) Une étude complète de la conception et la réalisation des traitements d'un SI ((2)) qui étend les résultats de la programmation structurée à l'analyse conceptuelle d'un problème. Elle propose d'une part, à chacune des étapes du processus de conception, un modèle de structuration des traitements et un langage d'expression associé. Cette étude définit d'autre part les règles de correspondance d'une structure à l'autre et le passage d'une expression à l'autre.

Elle présente enfin l'outil qui assure, dans certaines conditions de gestion des données, le passage automatisé de l'énoncé conceptuel d'un problème au programme correspondant. Elle est l'objet de la thèse de Lamy (19).

Les perspectives de recherche immédiates s'organisent autour de trois points :

- α) L'expérimentation et la mise au point des résultats théoriques acquis au niveau conceptuel par la conception

d'un SI réel dans une entreprise de la région lorraine.

- β) - La poursuite des recherches théoriques au niveau logique du processus de conception des SI pour lequel les solutions disponibles sont incomplètes, insatisfaisantes et même dans certains cas inexistantes.
- γ) - L'achèvement de la réflexion entreprise d'une part sur le PILOTAGE et la conception-réalisation du PILOTE, d'autre part, sur le système de conception assistée dans son ensemble.

#### 4 - LA SITUATION DE L'ETUDE DANS L'ENSEMBLE DES PROPOSITIONS DU NIVEAU CONCEPTUEL

##### 4.1. - Notre attitude au niveau conceptuel

Elle est schématisée sur la figure 2 et reprise sur les deux points clés suivants :

- a) Aboutir, à la fin de l'étape conceptuelle à une structure conceptuelle de SI ou schéma conceptuel de SI qui soit une représentation formelle des phénomènes de l'organisation pris en compte.
- b) Proposer une méthode pour aboutir au schéma conceptuel à partir de l'expression des résultats de l'analyse des problèmes appelée schéma de perception.

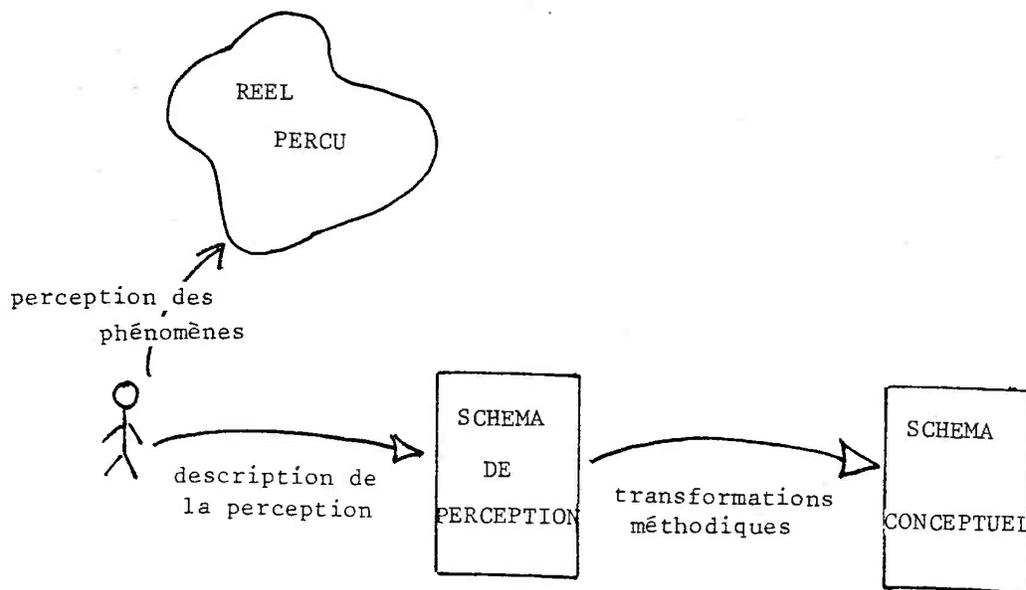


Figure 2 - L'attitude retenue pour la construction du SC

#### 4.2. - Rappels à propos du schéma conceptuel

##### 4.2.1. - Introduction

La difficulté de l'objectif 1 est liée au fait que cette représentation ne doit pas se limiter à une perspective exclusivement statique de ce qui est, à un moment donné, dans le réel organisationnel et doit assurer la prise en compte de la totalité des aspects statiques et dynamiques de l'organisation.

La structure conceptuelle du SI n'est pas seulement une structure de données mais des structures de données, de traitements et de commandes d'évolution.

L'expression de cette représentation que nous appelons par

extension de la terminologie bases de données ( 1 ), schéma conceptuel, est une traduction du monde réel en termes de types. Elle exclut toute considération technique correspondant à des préoccupations d'implantation de telle sorte que soit assuré l'indépendance des solutions d'implémentation et que soient possibles divers choix d'implantation pour un même schéma.

En terme de recherche, la solution repose sur la découverte d'un modèle conceptuel et d'un langage pour formuler le résultat de la modélisation. Les concepts ont été proposés par Foucaut dans ( 15 ). L'un des objets de cette étude est de définir les contraintes d'intégrité relatives à l'un de ces concepts.

#### 4.2.2. - Le modèle conceptuel

C'est un modèle formel, de type relationnel, qui a été défini à partir d'une analyse de l'organisation en termes de catégories de phénomènes qui le caractérisent.

##### 4.2.2.1. - Le modèle descriptif de l'organisation

Il peut se résumer par les deux points suivants :

a) Tous les phénomènes qui se produisent dans une organisation peuvent être classés en trois catégories :

OBJETS, EVENEMENTS, OPERATIONS.

- Un objet est un constituant concret ou abstrait de l'organisation qui peut être particularisé :

Exemple : un client, un produit, une facture.

- Une opération est une action qui peut être exécutée isolément dans l'organisation :

Exemple : livrer, facturer, calculer le salaire.

- Un évènement est ce qui peut survenir à un instant donné :

Exemple : l'arrivée d'une commande, la fin du mois, la rupture du stock.

Tous ces phénomènes peuvent être décrits par des propriétés et des associations.

- Une propriété est la qualité d'un phénomène ou d'une association de phénomènes réels qui permet de l'appréhender et de la distinguer des autres phénomènes ou des autres associations.

Exemple : le nom du client, la date de la livraison, le lieu de l'arrivée de la commande.

- Une association est une combinaison de deux ou plusieurs phénomènes appartenant à la même catégorie dans laquelle chacun d'eux joue un rôle spécifique.

Exemple : l'appartenance d'un employé à un service, la composition de l'opération du calcul du salaire en calcul des gains, des retenues, calcul de la différence.

b) La dynamique de l'organisation se définit par le treillis de relations qu'ont les catégories entre elles et se décrit par des associations.

Nous proposons en effet, une définition de la dynamique de l'organisation de type causale qui consiste à définir les évènements comme les causes de l'exécution d'opérations dont les effets sont les changements d'état d'objets qui peuvent être à leur tour des évènements.

Ceci nous autorise à représenter la dynamique par des associa-

tions entre les catégories. Ainsi :

- Une opération est une action qui provoque le changement d'état de un ou plusieurs objets.  
Exemple : l'opération "analyse des commandes" crée sous certaines conditions l'objet "commande acceptée".
- Un évènement est la constatation du changement d'état de un ou plusieurs objets qui entraîne l'exécution d'opérations.  
Exemple : l'évènement "arrivée de la commande" au courrier du matin déclenche sous certaines conditions les opérations de traitement des commandes.

#### 4.2.2.2. - Le modèle conceptuel

##### 4.2.2.2.1. - Correspondance entre modèle conceptuel et modèle descriptif

Le modèle conceptuel est du type relationnel. Sa correspondance avec le modèle descriptif, illustrée sur la figure 3 est la suivante :

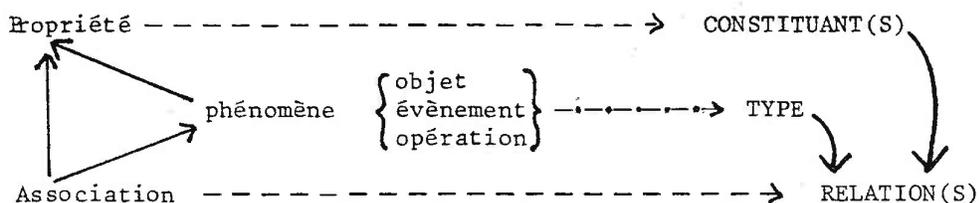


Figure 3 - Correspondance entre concepts des modèles

→ est défini par   -----> est représenté par   -----> est identifié par

Tout phénomène (objet, évènement, opération) ou association

de phénomènes est représentée par une ou plusieurs relations. Les propriétés sont représentées par les constituants de la relation.

Nous désignerons par la suite par c.objet la représentation d'un objet, par c.opération celle d'une opération, par c.évènement la représentation d'un évènement.

#### 4.2.2.2. - Contraintes d'intégrité du modèle

En définissant les relations normalisées ( 9 ), les chercheurs en bases de données ont pu donner des règles de construction d'un SC normalisé qui, par ses qualités, garantit la non redondance de la base des données et sa facilité d'exploitation. En transposant la terminologie BD, pour aboutir à un SC normalisé, nous avons introduit un ensemble de contraintes d'intégrité qui précisent la définition des concepts du modèle et qui garantissent la normalisation, la facilité et l'économie d'emploi du schéma conceptuel. Par ce biais nous sommes assurés de la non redondance du SI non plus seulement pour la partie des données mais globalement pour l'ensemble des données, des traitements et de leur dynamique.

Nous donnons ici l'énoncé de ces contraintes qu'on trouvera plus largement développées dans (15).

Pour faciliter l'écriture, nous notons :

C-OB : l'ensemble des désignations de relations appartenant au type C-objet.

C-OP : l'ensemble des désignations de relations appartenant au type C-opération.

C-EV : l'ensemble des désignations de relations appartenant au type C-évènement.

CHANG-ETAT : l'ensemble des désignations des trois types de changement d'état retenus sur les objets (création, suppression, modification).

et nous utilisons la notation classique de dépendance fonctionnelle (8).

Contrainte 1 - Dépendance fonctionnelle entre C-opération et C-objet

C-OP → C-OB    C-OP → CHANG-ETAT

Signification : Une c-opération modifie l'état d'un et d'un seul c-objet et d'une seule manière. Notons en revanche qu'un c-objet peut être modifié par plusieurs c-opérations.

Exemple : Toutes les opérations du type "adjonction de client" créent un objet supplémentaire du type client.

Contrainte 2 - Dépendance fonctionnelle entre c-événement et c-objet

C-EV → C-OB et C-EV → CHANG-ETAT

Signification : Un c-événement est la constatation d'un certain type de changement d'état d'un seul c-objet.

Exemple : Tous les événements du type "arrivée de commande" sont la constatation de la création dans le SI d'un objet commande.

En revanche un c-objet peut subir plusieurs changements d'état qui ne sont pas tous obligatoirement des événements.

Il est important de bien faire la distinction entre état d'objet et changement d'état. Ainsi par exemple, l'objet stock-produit 222 a, au cours de la vie du SI, un grand nombre d'états (perçus par la quantité en stock) ; un nombre limité d'entre eux, lorsqu'ils surviennent, sont des événements : ce sont ceux qui correspondent à un passage de la quantité en stock en-dessous du seuil de réapprovisionnement.

En outre, l'état rupture de stock-produit 222 subsiste un laps de temps  $t$  ; l'évènement rupture de stock produit 222 est instantané.

Contrainte 3 - Dépendance fonctionnelle multivaluée entre c-évènement et c-opération

C-EV  $\rightarrow$  C-OP

La définition et la description de la notion de dépendance fonctionnelle multivaluée est empruntée à ( 10 ).

Signification : L'ensemble des c-opérations qui peuvent succéder à un c-évènement est toujours le même, quelles que soient les circonstances dans lesquelles l'évènement s'est produit.  
En d'autres termes un c-évènement déclenche une ou plusieurs c-opérations.

Exemple : Tout évènement du type "arrivée de commande" déclenche les opérations "d'analyse de commande" et de "modification du stock virtuel".

Notons que le déclenchement peut être conditionnel.

Exemple : L'évènement "rupture de stock" déclenche l'opération "réapprovisionnement" seulement si l'état de la trésorerie le permet.

Enfin une opération est toujours déclenchée à un instant donné par un seul évènement mais peut l'être, à des instants différents, par plusieurs.

Contrainte 4 - Forme des relations du SC

Toutes les relations du SC sont en troisième forme normale.

### Contrainte 5 - Contrainte relative aux c-objets

Pour disposer d'une représentation correcte du système d'objets d'une organisation à travers la structure des c.objets du schéma conceptuel nous avons été amenés à introduire plusieurs contraintes d'intégrité sur le concept de c.objet qui sont étudiées au chapitre 2 de cette thèse.

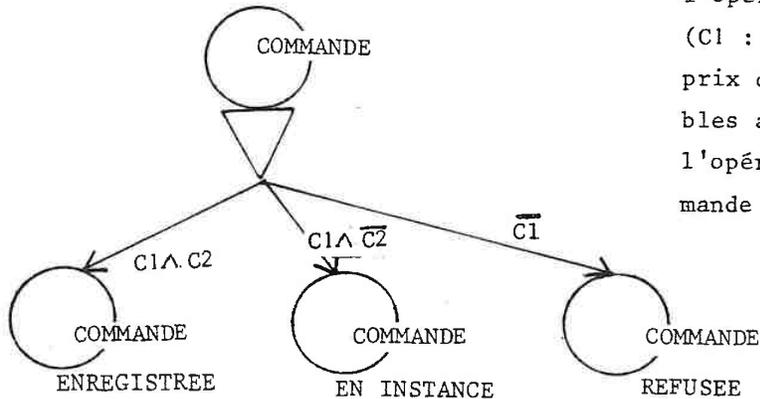
#### 4.2.3. - Le schéma conceptuel

La modélisation d'une organisation au moyen de ce modèle conduit à définir un schéma conceptuel comme une collection de relations que l'on peut analyser comme une double structure.

- Le système de c.classes ou structure des c.objets représentatifs de la structure naturelle de l'organisation. Ce système définit les constituants de l'organisation et leurs relations structurelles qu'on a décidé de prendre en compte dans le SI.  
Son étude fait l'objet du chapitre 2.

- La structure fonctionnelle ou structure des inter-relations entre c.objets, c.opérations et c.événements. Cette structure représente la dynamique de l'organisation à travers le treillis de liens de causalité qu'ont entre elles les catégories de phénomènes qui la composent.

Son étude fait l'objet du travail de M. Chesseron ( 7 ). Nous y ferons référence et utiliserons la représentation graphique de cette structure illustrée sur le schéma 4.



L'arrivée d'une commande est un évènement qui déclenche suivant les cas :  
 l'opération d'enregistrement d'une commande (C1 : le client, C2 : les délais et les prix demandés par le client sont compatibles avec ceux de l'entreprise) ;  
 l'opération de mise en instance de la commande ou le refus de la commande.

Figure 4 - Illustration graphique d'une structure fonctionnelle (partielle)

#### 4.3. - Rappels sur la méthode de construction du schéma conceptuel

La méthode revêt un double aspect :

- a) Celui de la description des résultats de l'analyse des phénomènes de l'organisation qu'on a décidé de prendre en compte dans le SI sous la forme d'un schéma de perception.
- b) Celui du passage du schéma de perception au schéma conceptuel par une suite de transformations respectant des règles méthodiques.

#### 4.3.1. - L'analyse et la représentation de la perception des problèmes

L'option prise par les chercheurs en conception de bases de données consiste à proposer un langage de définition qui permette la description du schéma conceptuel. Cette expression est l'aboutissement de la réflexion conceptuelle. Elle est l'image du résultat de l'étape mais aucune indication n'a été fournie au concepteur sur la manière d'y parvenir.

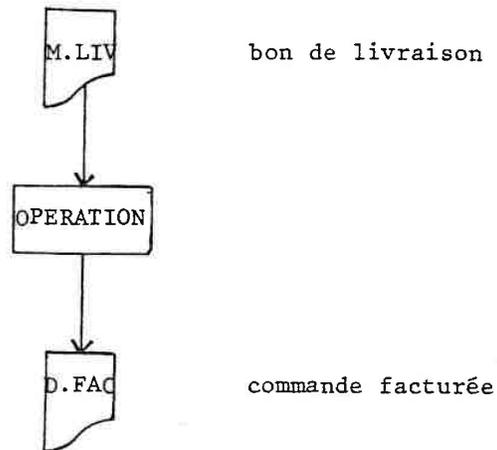
Il est, de notre point de vue, indispensable de proposer une méthode qui aide le concepteur à appréhender les phénomènes réels ; qui lui précise le point de départ de sa réflexion et le guide vers la solution conceptuelle.

Nous avons fait le choix d'obtenir un schéma conceptuel pour une organisation donnée par transformations successives d'un schéma de perception (SP). Il est le résultat de la représentation de perception qu'ont les gestionnaires de la vie de leur organisation.

C'est l'expression de leurs besoins sous forme d'énoncés déclaratifs et indépendants de toute considération technique liée à leur traduction sous forme automatisée. Les moyens assurant cette description sont présentés dans le travail de M. Chesseron ( 7 ). Ils conduisent à décrire le SP sous la forme de procédures indépendantes les unes des autres, et à décrire chaque procédure d'une part, à travers les messages qui lui sont associés, d'autre part les transformations qu'elle assure.

Exemple :

À la procédure de facturation sont associés les messages bons de livraison et commande facturée.



bon de livraison

L'arrivée d'un bon de livraison des marchandises d'une commande provoque le déclenchement de l'opération facturation, concrétisée par l'émission d'une facture.

commande facturée

Cette procédure est décrite par les deux descripteurs de messages "bon de livraison" et "commande facturée" et le descripteur de traitement de la figure 7.

A- Les informations véhiculées par le message sont décrites sur le descripteur de document.

En particulier, les informations du bon de livraison sont portées sur le descripteur identifié par MLIV et se composent :

- . du numéro de livraison (NLIV)
- . du poids de la commande (POIDS)
- . du nom du client ayant commandé (NOM)
- . du numéro de la préfecture correspondant à cette commande (NREF).

IDENTIFICATEUR de DESCRIPTEUR : <u>M-L.I.V.</u>		PROCEDURE	<u>PRGGE</u>
TITRE <u>BON DE LIVRAISON</u>		APPLICATION	<u>GICRM</u>
LIEU D'EMISSION <u>B-123 - LIVRAISON</u>		NATURE	<u>E</u>
DATE D'EMISSION _____ PERIODICITE _____		FREQUENCE	_____
LIGNE	NIVEAU	DESIGNATION des ELEMENTS	IDENTIFICATEUR
1		<u>PREFACTURE</u>	<u>PREF</u>
2		<u>POIDS/COMMANDE</u>	<u>POIDS</u>
3		<u>NOM DU CLIENT</u>	<u>NOM</u>

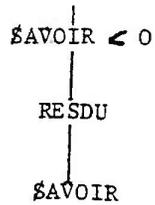
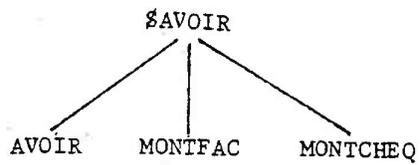
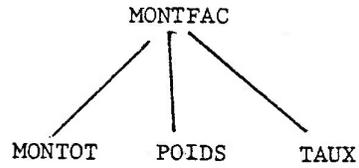
Figure 5

Les informations de la commande facturée sont portées sur le descripteur identifié par DFAC et se composent de deux sous-ensembles d'informations ; le premier étant associé à une facture, le second à chaque ligne de cette facture.

IDENTIFICATEUR de DESCRIPTEUR : <u>D-FAC</u>		PROCEDURE	<u>PRGGE</u>
TITRE <u>FACTURE DE COMMANDE</u>		APPLICATION	<u>GICOM</u>
LIEU D'EMISSION <u>B-01 COMPTABILITE</u>		NATURE	<u>S</u>
DATE D'EMISSION _____ PERIODICITE _____		FREQUENCE	_____
LIGNE	NIVEAU	DESIGNATION des ELEMENTS	IDENTIFICATEUR
01		<u>FACTURE</u>	<u>INFAC</u>
1		<u>DATE D'EMISSION</u>	<u>DATE</u>
2		<u>NOM DU CLIENT</u>	<u>NOM</u>
3		<u>ADRESSE DU CLIENT</u>	<u>ADRESSE</u>
4		<u>MONTANT DE LA FACTURE</u>	<u>MONTANT</u>
5		<u>RESTE DU</u>	<u>RESTE</u>
6		<u>NOUVEL AVANT</u>	<u>SAVOIR</u>
7		<u>LIGNE FACTURE</u>	<u>INFO</u>
8		<u>NUMERO-BREVET</u>	<u>NUMERO</u>
9		<u>PERIODE</u>	<u>PERIODE</u>
10		<u>PRELIMINAIRE</u>	<u>PRELIM</u>
11		<u>MONTANT PRELIMINAIRE</u>	<u>MONTANT</u>
12		<u>QUANTITE LIVREE</u>	<u>QUANTITE</u>

Figure 6

B - Les résultats (MONTFAC), (RESTDU) et (§SAVOIR) de la commande facturée sont définis par les arbres suivants :



Ces résultats sont décrits sur le descripteur de traitement suivant :



#### 4.3.2. - La méthode de transformation du schéma de perception

La méthode développée s'apparente aux méthodes de normalisation d'une collection de relations en forme quelconque ( 9 ) constituant le schéma conceptuel d'une base de données.

Elle consiste à trouver, à partir du schéma de perception, l'ensemble des c.objets, c.opérations et c.événements (concepts de la modélisation) respectant les contraintes du modèle et définissant le schéma conceptuel du SI à conduire sous la forme d'une structure de données (système de c.classes) et d'une structure dynamique (structure fonctionnelle). Elle comporte deux processus imbriqués :

- Le processus de construction du système de données ou système de c.classes, est du type synthétique : il consiste à construire les c.objets et à les structurer en c.classes et système de c.classes à partir d'une liste de constituants et de relations fonctionnelles entre constituants fournis par le concepteur ou déduites du schéma de perception par des automates.

Il fait l'objet du chapitre 3 de cette thèse.

- Le processus de construction de la structure fonctionnelle représentative de la dynamique de l'organisation est du type décomposition. Il consiste à analyser les énoncés du SP, à des décomposer en c.opération et c.événement et à déterminer les inter-relations entre c.objets c.événements et c.opérations.

Il fait l'objet de la thèse de M. Chesseron ( 7 ).

#### 4.4. - Présentation de l'étude

Ce travail définit un modèle de données et une méthode pour aboutir à la structure conceptuelle de données comme partie de la

de la structure conceptuelle du système d'information.

Le chapitre 2 présente le modèle de données choisi dans le cadre du modèle conceptuel de SI proposé par l'équipe Remora. Ce modèle s'apparente aux modèles de données relationnels ( 8 ), ( 11 ), ( 14 )... ou entité -relation ( 6 )... présentés par la communauté scientifique internationale , mais il est plus complet et original parce qu'il permet de prendre en compte le temps et de représenter la dynamique des objets informationnels de l'organisation.

Le chapitre 3 présente la méthode qui permet d'aboutir à la structure conceptuelle de données appelée système de c. classes. Peu de chercheurs se sont efforcés de proposer des moyens méthodiques pour faciliter le raisonnement de modélisation nécessaire à la construction des structures conceptuelles de données. L'ampleur et la difficulté de la tâche justifient pourtant qu'on s'y intéresse. En France, les travaux de Flory ( 14 ) sur lesquels nous nous sommes, en partie, appuyés vont dans ce sens.

C H A P I T R E   I I

MODELE DE DESCRIPTION STATIQUE

DU SYSTEME D'INFORMATION

## I N T R O D U C T I O N

La communauté scientifique des chercheurs et celle des praticiens admettent aujourd'hui que décrire l'aspect statique d'une organisation revient à décrire ses constituants par l'inventaire des objets.

Dans l'analyse et la description du monde réel, le problème posé est celui de la recherche des objets.

Lorsque l'on aborde la conception du SI, le problème devient celui de la représentation des objets du mieux que l'on peut dans une optique de gestion de données. On tente alors de définir les objets compte tenu des opérations de création, de suppression, de modification des renseignements relatifs à ces objets. Il faut tenter aussi de les représenter en respectant l'unité temporelle des différents aspects de leur vie.

C'est dans cette optique que nous avons cherché à définir un modèle de représentation des objets qui permette d'assurer la définition du treillis des objets d'une organisation par un schéma conceptuel (SC en abrégé).

Nous prenons ce terme dans son acception habituelle :

- Celle de l'ANSI-SPARC ( 1 ) en février 1975 : "Le schéma conceptuel représente la vue qu'a l'entreprise de la structure qu'elle tente de modéliser dans la base des données".
- Ou celle de B.B.B.C. ( 2 ). "L'organisation conceptuelle est le résultat de l'analyse d'un système et d'une phase de conception. Ce résultat est l'image complète (aussi complète que possible) des phénomènes présentant un intérêt au monde réel et qui seront représentés par le système d'information".

On recherche un modèle à partir duquel on pourra :

- a) construire un langage qui permettra de décrire le SC correspondant à des phénomènes analysés et modélisés,
- b) définir une méthode.

Nous présentons :

- 1 - Les concepts élémentaires
- 2 - Les concepts agrégés ou composés à partir des concepts élémentaires et nous montrons comment les concepts du modèle servent parce qu'ils permettent de décrire les aspects indispensables du monde réel et parce qu'ils permettent la mise en oeuvre aisée d'une méthode, parce qu'ils facilitent, en l'organisant, le travail du concepteur.

Ce travail rejoint sur de nombreux points les travaux effectués par les chercheurs en matière de base de données sur la prise en compte et la représentation, dans une vision statique, des phénomènes d'organisation. Il s'inspire des travaux de Codd ( 8 ), ( 9 ), Delobel ( 12 ).

Il a été rendu nécessaire par l'insuffisance de tous ces modèles

à assurer une représentation temporelle des objets satisfaisante. Il se différencie par les deux points suivants :

- 1 - Ce modèle, parce qu'il s'intègre dans un modèle plus général de représentation complète des phénomènes dans une double vision statique et dynamique de l'organisation, permet une représentation satisfaisante et originale de la dynamique des objets.
- 2 - Il est associé à une méthode qui permet son emploi aisé ce qui n'a été proposé que dans de très rares cas ( 14 ) par les chercheurs qui se sont intéressés en priorité à la recherche des concepts et ont négligé de proposer parallèlement un savoir faire.

## 1 - LES CONCEPTS ELEMENTAIRES : LE MOT

Nous étudierons les mots, considérés isolément.

### 1.1. - Définition du mot

Le mot est la chaîne de caractères minimum, ayant une signification par elle-même indépendamment des autres chaînes auxquelles on veut l'associer.

La chaîne de caractères est définie dans un système symbolique donné. L'essentiel réside, non pas dans le symbolisme utilisé, mais dans la correspondance permettant d'associer une signification à cette chaîne. A priori, tous les symbolismes sont équivalents. Mais il importe que dans le symbolisme utilisé, à chaque mot soit associée une signification non ambiguë. Ainsi, dans les faits, seuls quelques symbolismes sont acceptables; ceux qui sont familiers aux usagers.

### 1.2. - Etude du mot

#### 1.2.1. - Sémantique du mot

Le mot représente une notion du monde réel.

Exemple : Nom, prénom représentent les propriétés des clients de l'entreprise.

n pro représente le numéro de référence des produits de l'entreprise.

1.2.2. - Le mot est une désignation. Il est censé représenter une notion du monde réel.

La désignation devrait donc respecter la règle d'unicité

suivante : à une désignation n'est associée qu'une et une seule propriété du monde réel.

Tout cas de synonymie ou de polysémie involontaire entraîne le non respect de cette règle et des incohérences dans l'utilisation des mots. La détection des synonymes et des polysèmes est un problème qui a été abordé dans les systèmes d'aide à l'analyse : tels que MACSI (22), SCAPFACE (20). Les méthodes d'analyse se sont quelquefois préoccupées de ces phénomènes, en particulier celles qui ont introduit des outils documentaires, telles que MINOS (21).

Dans REMORA, des contrôles développés par O. Thiery (29) permettent de détecter certaines anomalies dues à l'existence de ces phénomènes.

Afin d'éliminer les cas de polysémie, il peut être intéressant d'obtenir automatiquement des désignations à partir de textes. Monique Chabre-Peccoud propose dans (5) une méthode de production automatique de désignations abrégées.

#### 1.2.3. - Une définition est associée au mot

Pour compléter la définition symbolique du mot, on rajoute une définition. Elle a pour rôle de réduire les ambiguïtés dans le choix de la signification qu'apporte le mot.

#### 1.2.4. - Domaine des valeurs du mot

##### a) Occurrence du type mot

Quand on veut décrire une réalité donnée, on fait usage de plusieurs mots. On parlera des occurrences du mot.

Le type mot est constitué de l'ensemble des mots. Exemple : client, produits sont des occurrences du type mot.

b) Domaine des valeurs du mot

Chaque occurrence de mot prend des valeurs. On appelle le domaine des valeurs du mot l'ensemble des valeurs que peut prendre l'occurrence.

Le domaine des valeurs correspond au contexte de validité.

Exemple : l'âge de personnes peut prendre des valeurs de 1 à 150.

c) Valeur impossible ou inexistante

Il peut arriver que les valeurs de mots ne puissent pas être définies.

Exemple : la situation militaire d'une personne du sexe féminin, ou le nom de jeune fille d'une personne du sexe masculin n'existent pas.

Dans ce cas la valeur du mot est impossible ou inexistante.

d) Valeur inconnue

La valeur du mot peut être inconnue à un instant donné dans le SI.

Ainsi, le montant total d'une facture peut être inconnu à un instant donné, puis valorisé plus tard.

e) Réalisation des mots

L'étude des valeurs prises dans le temps par une occurrence nous conduit à définir la réalisation.

Soit le mot numéro produit, noté npro.

Les occurrences du mot, à différents instants, sont les suivantes :

Instants	Occurrences de npro			
	1ère occ.	2ème occ.	3ème occ.	4ème occ.
$t_0$	10	15	20	
$t_1$	10	15	20	
$t_2$	10			30
$t_3$				30

Dans cet exemple, la valeur de chacune des occurrences est constante au cours du temps. Ce n'est pas toujours le cas, et les occurrences peuvent prendre des valeurs différentes au cours du temps.

Soit le mot quantité du stock. Le tableau des valeurs des différentes occurrences est le suivant :

Instants	Occurrences des qte stock			
	Valeurs 1ère occ.	Valeurs 2ème occ.	Valeurs 3ème occ.	Valeurs 4ème occ.
$t_0$	30	35	40	
$t_1$	40	10	45	
$t_2$	40			5
$t_3$				5

### Définition de la réalisation

Nous appelons réalisation la valeur d'une occurrence à un instant donné.

Exemple : la réalisation de la 1ère occurrence de qte stock à l'instant  $t_1$  est 40.

Les réalisations aux instants  $t_0$ ,  $t_1$  et  $t_2$  de numéro produit ont même valeur, cette valeur est 10.

Les réalisations de qtestock aux instants  $t_0$  et  $t_1$ ,  $t_2$  ont des valeurs différentes, qui sont 30, 40, 40.

#### 1.2.5. - Typologie des mots

Différentes classifications de mots ont été proposées. Reix ( 25) utilise la notion de "rubriques répétitives" et "non répétitives".

Flory ( 14) propose deux classifications ; l'une, par référence au temps, découpe l'ensemble des rubriques en rubriques stables et non stables ; l'autre, fondée sur l'existence ou la non existence d'une règle de gestion, classe les rubriques en rubriques calculées ou non calculées.

Nous proposons également deux typologies : l'une par analyse des valeurs prises par un mot dans le temps, l'autre par référence à l'usage du mot.

##### 1) Typologie des mots par analyse des valeurs qu'ils prennent dans le temps

Mot permanent : dans son contexte de validité, le mot prend une valeur et une seule, qui ne change jamais dans le temps.

Exemple : la date de naissance, le numéro INSEE d'une personne, la date

d'un contrat sont des mots permanents.

Mot non permanent : dans son contexte de validité, le mot prend successivement dans le temps des valeurs qui se substituent les unes ou autres.

Exemple : solde d'un compte ou quantité en stock ; prix d'achat d'un article, taux de changes d'une devise.

## 2) Typologie des mots par référence à l'usage du mot

Le mot peut être soit donnée, soit variable, soit résultat. Ces trois natures, pour un mot, dans un contexte défini s'excluent.

- Un mot est donnée quand les valeurs qu'il prend résultent uniquement d'observations faites dans le monde réel.  
Exemple : nom d'une personne, prix d'achat d'un article.
- Un mot est variable quand les valeurs qu'il prend résultent des valeurs antérieures qu'il a prises; ce qui signifie que la nouvelle valeur est obtenue par un calcul qui fait intervenir la valeur antérieure.  
Exemple : quantité ou stock, avoir d'une personne.
- Un mot est résultat quand les valeurs qu'il prend dépendent des valeurs prises par d'autres mots.  
Exemple : montant d'un salaire, d'une facture.

Nous remarquons que la nature d'un mot dépend de son contexte. Un mot peut prendre plusieurs natures en fonction de son contexte d'usage. Ainsi, dans le schéma de perception, un mot peut être de nature résultat dans une procédure et de nature variable dans une autre procédure ou pour l'ensemble des procédures du SP.

Le contexte que nous considérons ici est le système d'information.

### 3) Combinaison et compatibilité entre les différents types

L'analyse des combinaisons entre les différents types nous conduit à dresser le tableau des compatibilités suivant :

Type/usage Type/temps	Permanent	Non permanent
Résultat	X	
Variable		X
Donnée	X	X

Un mot résultat est un mot permanent.

Exemple : le montant total d'une facture est permanent.

Un mot variable est un mot non permanent.

Exemple : la quantité en stock d'un produit, mot de nature variable, est non permanente.

Un mot donnée peut être permanent ou non permanent.

Exemples: le numéro INSEE d'une personne est permanent.

le prix unitaire d'un article est non permanent.

## 2 - LES LIENS ENTRE LES MOTS

Dans un contexte de représentation défini, les mots sont faits pour être associés à certains autres mots. En effet, en associant plusieurs mots, on transmet plus d'information qu'en maintenant les mots dissociés.

Un mot peut être associé simultanément à plusieurs autres mots. Toutes les associations ne sont pas indifféremment possibles, et toutes ne sont pas porteuses de la même quantité d'information. Enfin, dans chacune d'elles le mot ne joue pas le même rôle.

Nous allons caractériser un mot qui appartient à plusieurs associations, et mettre en évidence les rôles qu'il peut jouer, sachant qu'un même mot peut se retrouver associé à plusieurs autres mots.

Désormais, notre étude des mots se fera par rapport aux autres mots avec lesquels il peut être associé.

Nous étudierons les rôles qu'un mot joue à l'égard d'autres mots qui peuvent lui être associés (identifiant ou composant) et les propriétés des associations ainsi obtenues.

## 2.1. - Les relations fonctionnelles

Quand on considère deux mots, il est intéressant d'analyser les valeurs que prennent les occurrences de l'un par rapport aux valeurs des occurrences de l'autre.

### 2.1.1. - Définition de la relation fonctionnelle entre deux mots

La définition de Codd ( 9 ) : "Un constituant H est dit fonctionnellement dépendant du constituant G d'une même relation si, et seulement si, la valeur du constituant G détermine la valeur du constituant H", a été la source d'autres définitions (12), (14).

Définition de la relation fonctionnelle (RF) ou dépendance fonctionnelle :  
Nous dirons qu'il existe une relation fonctionnelle du mot J par rapport au mot I lorsque la connaissance d'une réalisation  $m_i$  du mot I détermine au plus la connaissance d'une réalisation  $m_j$  du mot J, à un instant donné.

Nous la notons = mot I  $\rightarrow$  mot J

Exemples : les relations client  $\rightarrow$  adresse

numéro commande  $\rightarrow$  client

sont fonctionnelles. A un client n'est associée qu'une seule adresse ; à un numéro de commande ne correspond qu'un client.

Comme d'autres auteurs nous considérons qu'un calcul peut être considéré comme une forme spécifique de la relation fonctionnelle.

Exemple : Le prix d'un article est le produit de son prix d'achat par un taux :

$$PR = PU \times TAUX$$

Le prix de cet article dépend fonctionnellement du couple (prix d'achat, taux) :

$$(PU, TAUX) \rightarrow PR$$

#### 2.1.2. - Typologie des RF

##### Relation fonctionnelle forte :

La RF mot I  $\rightarrow$  mot J est forte si la connaissance d'une réalisation  $m_i$  du mot I détermine la connaissance d'une et une seule réalisation  $m_j$  du mot J.

Exemple : la RF n<sub>fac</sub>  $\rightarrow$  mont<sub>fac</sub> qui, à un numéro de facture, associe le montant de la facture est forte.

Un et un seul montant de facture est toujours présent sur la facture.

##### Relation fonctionnelle faible :

Une relation fonctionnelle faible est une relation qui à une réalisation associe 0 ou une réalisation. Ce qui signifie que  $m_j$  n'existe pas nécessairement quand  $m_i$  existe.

Exemple : dans la gestion d'une agence de voyages, la relation qui, à un client associe le montant de son séjour est faible.

cli  $\rightarrow$  montant

En effet, le montant du séjour est calculé lorsque le séjour souhaité est disponible. Il n'existe pas nécessairement lorsque le client existe.

Remarque : La RF, telle que la définissons, est faible.

#### Relation fonctionnelle directe

La RF mot I  $\rightarrow$  mot J est directe s'il n'existe pas mot K tel que mot I  $\rightarrow$  mot K  $\rightarrow$  mot J.

Exemple : Soit npers le numéro d'une personne

nvoit le numéro de la voiture possédée par cette personne

nbkm le nombre de kilomètres de cette voiture,

La RF npers  $\rightarrow$  nbkm n'est pas directe puisque npers  $\rightarrow$  nvoit  $\rightarrow$  nbkm.

#### 2.1.3. - Typologie des RF compte tenu du temps

Peu d'auteurs se sont intéressés à une typologie des RF dans le temps. Nous définissons une typologie reposant sur l'analyse de la vie des occurrences qui nous permettra une meilleure représentation de la dynamique des objets.

Cette référence au temps et à la vie des occurrences les unes par rapport aux autres rend parfaitement compte des phénomènes qui se produisent dans le monde réel.

##### 2.1.3.1. - Relation fonctionnelle permanente

###### a) Définition

La RF  $m_i \rightarrow m_j$  définie entre deux mots permanents est permanente si, quelle que soit l'occurrence  $m_i$  appartenant à  $m_i$ , l'occurrence  $m_j$  appartenant à  $m_j$  a la même durée de vie que  $m_i$ .

Durée de vie d'une occurrence

Nous appelons durée de vie d'une occurrence la durée comprise entre la date de création et la date de suppression de cette occurrence.

Exemples : dans les exemples de la page

- la durée de vie de la première occurrence numéro produit est  $(t_0 \ t_2)$
- la durée de vie de la quatrième occurrence numéro produit est  $(t_2 \ t_3)$
- la durée de vie de la deuxième occurrence de quantité stock est  $(t_0 \ t_1)$

b) Représentation

mi  
 $\downarrow$  la flèche doublement pointée représente une RF  
 mj entre mi et mj.

On remarque qu'une RF permanente lie deux mots permanents qui naissent aux mêmes instants, et dont les occurrences sont valorisées et détruites à l'occasion de mêmes événements.

c) Exemple

Un accusé de réception d'une commande  $\longrightarrow \dot{\rightarrow}$  lieu de livraison

$\longrightarrow \dot{\rightarrow}$  n° de client

$\longrightarrow \dot{\rightarrow}$  date de la commande

On considère un seul client par commande, et le n° d'accusé de réception ne peut être attribué si le numéro de client n'est pas connu.

d) La RF permanente lie deux mots permanents. Nous allons introduire un type de RF définissant une liaison entre un mot de type permanent et un mot de type non permanent.

### 2.1.3.2.- Relation fonctionnelle semi-permanente

#### a) Définition

La RF mot I  $\rightarrow$  mot J définie entre un mot permanent et un mot non permanent est semi-permanente si, quelle que soit l'occurrence  $m_i$ , l'occurrence  $m_j$  a même durée de vie que  $m_i$ .

#### b) Exemples :

1er exemple : Soit le tableau des réalisations des occurrences du numéro produit (npro) et de la quantité en stock (qtéstock) :

Constituants Instantants	npro	qtéstock
$t_0$	5	10
$t_1$	5	15
$t_2$	5	20
$t_3$	5	25

Les occurrences de npro et qtéstock ont la même durée de vie ( $t_0$   $t_3$ ).  
La RF npro  $\rightarrow$  qtéstock est semi-permanente.

Contre-Exemple : Soit le tableau des réalisations des occurrences du numéro de produit et de la quantité en stock suivant :

Instant	Constituants	
	npro	qtéstock
$t_0$	10	5
$t_1$	10	10
$t_2$	10	15
$t_3$	10	

La durée de vie de npro est  $(t_0, t_2)$ , celle de qtéstock est  $(t_0, t_2)$ .

Les durées de vie sont différentes. La RF n'est pas permanente.

2ème exemple :

n° d'Accusé de réception, n° de ligne  $\longrightarrow$  quantité  
d'une commande commandée

Cette quantité peut changer quand le client fait un avenant à la commande. Cependant, à tout instant de la vie de la commande, on a une quantité commandée.

La relation fonctionnelle n° d'accusé de réception, n° de ligne  $\longrightarrow$  quantité commandée est semi-permanente.

c) Représentation

mi  
↓  
mj

Une flèche pointée représente une RF entre mI et mJ

Une RF semi-permanente lie un mot permanent à un mot non permanent. Elle existe à tout instant de la vie du mot permanent, et fait dépendre, à un instant donné, une réalisation du mot non permanent à une réalisation du mot permanent.

Dans le temps, chaque réalisation du mot non permanent dépend fonctionnellement et successivement de celle du mot permanent.

d) Sémantique

La RF semi-permanente représente une liaison entre deux éléments ayant même durée d'existence, et une succession de liaisons entre les valeurs de ces deux éléments, un élément ayant une valeur constante, l'autre une valeur non permanente au cours du temps.

e) Les RF permanentes et semi-permanentes sont fortes

Les contraintes portant sur les durées de vie des réalisations donnent cette qualité aux R.F. permanentes et semi-permanentes.

Nous allons introduire une RF qui peut être faible liant dans le temps deux mots pouvant naître à des instants différents.

2.1.3.3. - Relation fonctionnelle stable

a) Définition

La RF mot  $i \rightarrow$  mot  $j$  définie entre deux mots permanents est stable si, quelle que soit l'occurrence  $m_i$ , l'occurrence  $m_j$  qui lui est associée est toujours la même.

La notion de stabilité a été définie sous une forme voisine par Flory ( 14 ) par Tardieu ( 27 ) et Hainault ( 16 )

b) Exemple de RF stable

La RF appartement  $\rightarrow$  vente-appartement est stable.  
Une société vend des appartements neufs. A un appartement ne correspond qu'une vente appartement au cours du temps. Supposons que le tableau des

réalisations des mots appartement et vente appartement soit le suivant :

	Appartement	Vente-appartement
$t_0$	10	
$t_1$	10	v 10
$t_2$	10	v 10
$t_3$	supprimé	supprimé

La RF est faible, car un appartement peut ne pas être vendu tout de suite. A l'instant  $t_0$ , appartement existe alors que vente-appartement n'existe pas.

A une occurrence d'appartement correspond toujours la même occurrence de vente-appartement.

Contre-exemple :

La RF appartement  $\rightarrow$  location-appartement n'est pas stable. Une société loue des appartements. A un appartement ne correspond qu'une location à un instant donné, mais plusieurs locations au cours du temps.

Le tableau des réalisations des mots appartement et location appartement est le suivant :

Constituants Instants	Appartement	Location-appartement	
		Occ. 1	Occ. 2
$t_0$	200	$L_1$ 200	
$t_1$	200	$L_1$ 200	
$t_3$	200	suppression	
$t_4$	200		$L_2$ 200

A un instant, à une occurrence d'appartement correspond une occurrence de location appartement, mais cette occurrence n'est pas toujours la même. L'occurrence associée à appartement à l'instant  $t_4$  n'est pas celle qui lui est associée à l'instant  $t_1$ .

c) Sémantique

La RF stable lie deux mots de même nature par rapport au temps, les durées d'existence des deux réalisations en dépendances fonctionnelles peuvent être décalées dans le temps.

d) Nous remarquons que toutes les RF permanentes sont stables.

2.2. - Les relations fonctionnelles sont des associations particulières de mots pris le plus souvent deux à deux.

Nous allons étudier des associations portant sur plusieurs mots, ce qui nous conduit à définir le concept de groupe.

### 3 - LES GROUPES DE MOTS

#### 3.1. - Définition du groupe de mots

Un groupe de mots est un ensemble non vide de mots.

#### 3.2. - Définition d'une réalisation d'un groupe

On définit une réalisation du groupe de mots en attribuant une valeur à chaque mot appartenant au groupe de mots.

#### 3.3. - Les relations fonctionnelles et les groupes

Nous rappelons que Codd définit la RF entre deux

constituants ( 9 ) que ces constituants soient ou non composés. Transposée à nos concepts, la RF garde son sens entre groupes de mots. De même, les notions de types peuvent être transposées.

Deux nouveaux types, ceux de la RF élémentaires et de la RF canonique faisant intervenir la notion de groupe sont définis.

Notion de relation fonctionnelle élémentaire (FULL FONCTIONNAL DEPENDANCY) (Codd, 71).

"Un constituant H est en dépendance fonctionnelle élémentaire par rapport à G :

- . s'il est fonctionnellement dépendant de G,
- . et s'il n'est pas fonctionnellement dépendant d'une quelconque sous liste de G (on suppose dans ce cas que G est un constituant composé".

Notion de RF canonique

On dit qu'une RF entre groupes est canonique si le groupe but de la RF est un singleton, c'est-à-dire un groupe constitué d'un seul mot. Ce terme est introduit par Bernstein (3).

#### 3.4. - Identifiant du groupe

L'identifiant du groupe est un mot ou un groupe de mots dont les réalisations déterminent de façon unique chaque réalisation du groupe. Cette notion correspond à celle de Codd.

. Notion d'identifiant (Codd,71)

"C'est un constituant ou une suite de constituants dont les valeurs déterminent de façon univoque un n-uple d'une relation".

On remarque que l'identifiant est une notion qui caractérise le groupe. On peut ainsi caractériser le groupe par le nombre des mots que

contient l'identifiant. Ainsi on parlera de l'ordre de l'identifiant :

Ordre 1, l'identifiant est d'ordre 1 s'il est composé d'un mot  
Ordre n, l'identifiant est d'ordre n s'il est composé de n mots.

Cette terminologie est introduite par Flory dans (14) ; l'identifiant est encore nommé index.

Nous allons étudier les groupes ayant une structure particulière et définir une typologie des groupes.

### 3.5. - Typologie des groupes

#### 3.5.1. - Groupe en 3ème forme normale

Nous reprenons la définition de Delobel dans (12) : "Une collection de données sera en troisième forme normale si pour tout constituant A relié fonctionnellement à un index I de la relation R, la relation fonctionnelle  $I \rightarrow A$  est élémentaire et directe".

#### 3.5.2. - Groupe primaire GP

La notion de groupe primaire est définie dans les travaux de Flory (14). Cette notion est plus forte que celle de groupe en 3ème FN : tout groupe primaire est en 3ème FN alors que la réciproque n'est pas toujours vérifiée.

#### Définition du groupe primaire (Flory)

Nous disons qu'un groupe de rubriques G est un groupe primaire formé sur C s'il existe un index  $G' \subset G$  ( $G \neq \emptyset$ ) et si, pour toute rubrique de l'ensemble ( $G - G'$ ) la relation  $G' \rightarrow mi$  appartient à F.

F étant ainsi défini :

F : ensemble des sous relations fonctionnelles élémentaires directes et canoniques de la relation n-aire , formée sur l'ensemble des rubriques d'un ensemble C.

Du même auteur nous retiendrons la définition de l'ordre du groupe.

Définition du groupe primaire d'ordre 1

"Un groupe primaire est d'ordre 1 si son index est constitué par une seule rubrique".

Définition du groupe primaire d'ordre n

"Un groupe primaire est d'ordre n si son index est constitué par n rubriques".

Les identifiants de groupes sont des mots permanents.

3.5.3. - Définition du groupe primaire maximal

"Un GP formé sur C est dit maximal sur C si toutes les rubriques de C qui dépendent fonctionnellement (d'une façon élémentaire et directe) de l'index du groupe appartiennent au groupe."

Sémantique du GP maximal

Le GP est la représentation partielle d'un type du monde réel. Le  $GP_1$  représente un type d'entité ; le  $GP_n$  un type d'association.

3.5.4. - Le c.objet

3.5.4.1. - Définition du c.objet

1) Définition du c.objet

Soient M un ensemble de mots,  $M = \{mot_i\}, i \in I$

F un ensemble de sous relations fonctionnelles élémentaires, directes et canoniques de la relation n-aire R formée sur M.

Le c.objet défini sur  $(F, M)$  est un groupe primaire maximal défini sur  $(F, M)$  tel que :

- 1) pour tout  $\text{mot}_i$  appartenant au GP,  $\text{mot}_i$  étant de type permanent et non identifiant de GP, les RF identifiant du  $\text{GP} \rightarrow \text{mot}_i$  sont permanentes.
- 2) pour tout  $\text{mot}_i$  appartenant au GP de type non permanent,
  - les RF identifiant du  $\text{GP} \rightarrow \text{mot}_i$  sont semi-permanentes
  - pour tout  $\text{mot}_j$  appartenant au GP  $j \neq i$ , quelle que soit  $m_i$ , l'occurrence de  $\text{mot}_i$ , les valeurs de l'occurrence  $m_j$  de  $\text{mot}_j$  aient même durée de vie.

Durée de vie de la valeur d'une occurrence d'un mot :

La durée de vie de la valeur d'une occurrence est la durée comprise entre l'instant de création de cette valeur et celui de suppression de cette valeur.

Exemple : Dans l'exemple page 36 , la durée de vie de la valeur 40 de la 1ère occurrence de qtéstock est  $(t_1 t_2)$ .

La définition du c.objet que nous venons de donner exprime que nous n'acceptons comme c.objet que les regroupements homogènes par rapport au temps :

Les RF entre identifiants de c.objet et  $\text{mot}_i$  non identifiants sont : . soit permanentes et tous les  $\text{mot}_i$  sont permanents  
 . soit semi-permanentes et tous les  $\text{mot}_i$  sont non permanents, leurs valeurs ont la même évolution.

Identifiant du c.objet

De même que dans un GP, l'identifiant est un mot ou un groupe de mots permanents .Il est généralement unique.

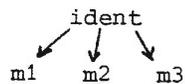
Il découle de la définition du c.objet que lorsqu'un c.objet possède plusieurs identifiants, ceux-ci sont liés par des RF permanentes, et que, réciproquement, lorsque deux identifiants sont liés par des RF permanentes, ils identifient un même c.objet.

Lorsqu'un c.objet possède plusieurs identifiants, le concepteur en choisit un, l'identifiant principal ou identifiant clé.

b) - Représentation du c.objet

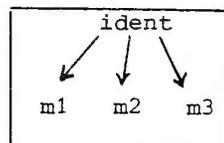
A - Graphe associé au c.objet

Nous adoptons une représentation graphique du c.objet sous forme d'un graphe. Les sommets de ce graphe sont les mots, et les arcs sont les RF.



B - Représentation du c.objet

Tout c.objet sera représenté graphiquement dans un rectangle entourant le graphe, associant l'identifiant du c.objet et les mots mi non identifiants qui le constituent :



c) - Un c.objet vérifie deux hypothèses que nous pouvons qualifier de statique et dynamique.

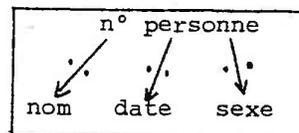
Le c.objet se présente comme le plus grand groupe de mots :

- 1) étant la représentation conceptuelle d'un sous-ensemble d'une classe d'objets du monde réel,
- 2) ce sous-ensemble ayant une homogénéité temporelle.

La seconde hypothèse n'est pas vérifiée par un GP. En effet la représentation, par un GP, d'un type du monde réel ne respecte pas nécessairement l'unité temporelle des différents aspects de ce type.

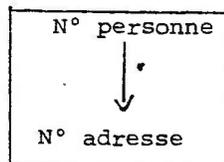
d) Exemples de c.objets

- C.objet dont les RF sont permanentes



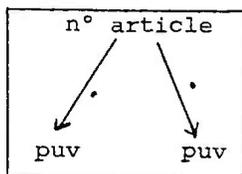
Le nom, la date de naissance, le sexe d'une personne ne sont pas modifiés et sont créés et supprimés en même temps.

- C. Objets dont les RF sont semi-permanentes



L'adresse d'une personne peut être modifiée.

La RF liant n° de personne à adresse est semi-permanente.



Le prix unitaire de vente et le prix unitaire d'achat d'un article sont deux données non permanentes subissant des modifications à l'occasion des mêmes évènements.

### 3.5.4.2. - Nature du c.objet

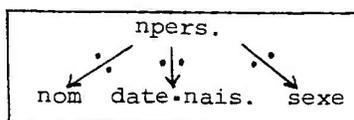
La nature du c.objet est définie à partir de de la nature des mots non identifiant qui le composent.

#### a) Définition.

##### Nature donnée

Un c.objet est de nature donnée si les mots non identifiant lui appartenant sont de nature donnée.

Exemple :

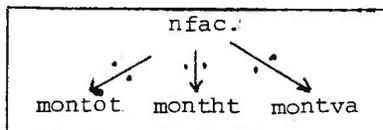


Dans ce c.objet tous les mots non identifiant sont de nature donnée.

##### Nature résultat

Un c.objet est de nature résultat si les mots non identifiant lui appartenant sont de nature résultat.

Exemple :

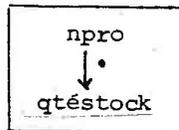


montot, montht, montva sont des mots résultats ; le c.objet facture dont l'identifiant est nfac est de nature résultat.

Nature variable

Un c.objet est de nature variable si les mots non identifiant lui appartenant sont de nature variable.

Exemple :



La quantité en stock d'un produit est un mot variable.

b) Il découle de la définition du c.objet qu'on ne peut pas avoir de c.objet simultanément de nature donnée, de nature résultat ou de nature variable. Ceci est lié à l'homogénéité des RF dans un c.objet et aux contraintes portant sur les durées de vie des occurrences.

Démontrons-le /

Compte-tenu des natures des mots un GP pourrait être composé :

- $\alpha$  - soit de résultats et de données,
- $\beta$  - soit de résultats et de variables,
- $\gamma$  - soit de données et de variables.

Montrons par l'absurde que la définition du c.objet rend "impossible" les compositions  $\alpha$ ,  $\beta$ ,  $\gamma$ .

Composition  $\alpha$  : mots résultats et mots données.

- un mot résultat est un mot dont les valeurs sont calculées.
- un mot donnée est un mot dont les valeurs résultent d'observations.

Les données et les résultats sont donc créés à des instants différents. Il est alors impossible que les RF identifiant donnée et identifiant résultat soient toutes les deux permanentes.

Le GP identifié par nfac n'est pas un c.objet.



np, mot de nature donnée représente le nombre de produits commandés.

monpro, mot de nature résultat représentant le montant total des produit, est évalué à partir du nombre de produits commandés.

Dans ce GP, . soit la RF nfac → np est permanente, et, dans ce cas, la RF nfac → monpro ne l'est pas, car monpro n'existe pas quand nfac existe

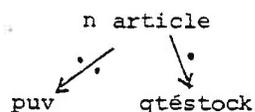
. soit la RF nfac → monpro est permanente, et dans ce cas, la RF nfac → np ne l'est pas car np est créé avant nfac.

Composition β : mots résultats et mots variables.

- Un mot variable est un mot non permanent, qui prend des valeurs se substituant les unes aux autres.
- Un mot résultat est un mot permanent, qui prend une valeur et une seule.

Soit moti, mot variable et motj, mot résultat appartenant à un même GP. Leurs valeurs ont donc des durées de vie différentes. La condition 2β de la définition du c.objet n'est pas vérifiée.

Exemple : le GP identifié par n article n'est pas un c.objet.



La quantité en stock d'un article change au cours du temps indépendamment du prix de vente de l'article.

Les réalisations de puv et de qtéstock n'ont pas la même durée de vie.

Composition γ : mots données et mots variables.

Soient ni mot variable et mj mot donnée appartenant à un même GP.

mi est variable, ce qui signifie qu'une nouvelle valeur de ce mot est obtenue après un calcul faisant intervenir la valeur ancienne.

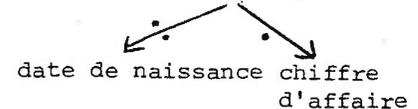
mj est donnée, ce qui signifie que les valeurs résultent d'observations.

Les valeurs sont donc créées à des instants différents. Alors la condition 28 de la définition n'est pas vérifiée.

Exemples : n° personne



n° elient



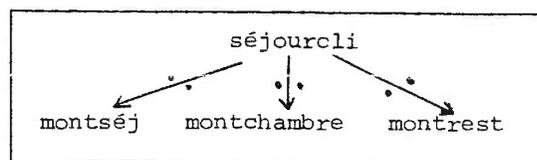
Bien que dans le groupe identifié par n° personne, les RF soient toutes semi-permanentes, ce groupe n'est pas un c.objet. En effet, âge et chiffre d'affaire sont modifiés de façon indépendante.

#### 3.5.4.3. - Intérêt du concept de c.objet

a) Le c.objet est la représentation d'une partie d'objet\* ayant une homogénéité temporelle

En effet, tous les mots non identifiants d'un c.objet jouent le même rôle vis-à-vis des opérations de création, de suppression et de modification.

Exemple : Dans la gestion d'une agence de voyages, nous créons le c.objet séjour-client, dans lequel les mots montant total du séjour, montant chambre, montant restaurant sont créés en même temps.



\* Nous appelons objet le type objet, l'ensemble ou la classe d'objets.

Tous les mots du c.objet identifié par séjourcli. sont créés en même temps.

b) La minimisation des valeurs inconnues est une conséquence de cette unité temporelle

Tous les mots non identifiant du c.objet sont valorisés en même temps que l'identifiant au c.objet.

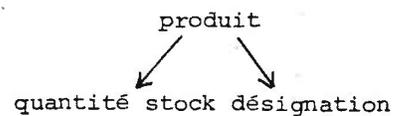
c) Nous verrons que la possibilité de former des c.classes de c.objet présente également un intérêt majeur

Il nous semble important de pouvoir regrouper des c.objets en un ensemble représentatif d'un phénomène du monde réel.

Une autre définition possible du c.objet consisterait à imposer l'existence d'une RF permanente entre l'identifiant du c.objet et tout mot non identifiant.

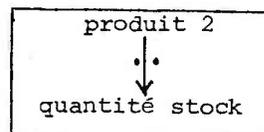
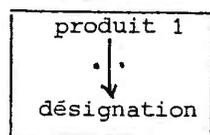
Cette possibilité, qui offrait l'avantage de présenter une plus grande homogénéité temporelle (identifiant et mots non identifiants sont créés et supprimés en même temps) supprimait totalement l'existence de mots non permanents et ne nous permettait pas de constituer des c.classes. C'est pourquoi nous l'avons rejetée.

Exemple : Soit le GP produit



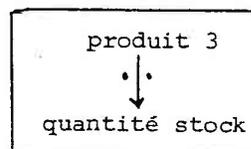
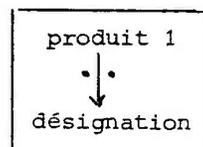
La quantité du stock varie dans le temps, la désignation est constante.

a) D'après la définition du c.objet que nous avons retenue, les c.objets définis seront :



L'existence des 2 RF stables  $\text{produit 1} \rightarrow \text{produit 2}$   
 $\text{produit 2} \rightarrow \text{produit 1}$   
 nous permettra de définir plus tard une c.classe.

β) La possibilité, que nous rejettons, en supprimant les RF semi-permanentes, nous conduirait à définir les c.objets



Il n'existerait plus la RF stable  $\text{produit 1} \rightarrow \text{produit 3}$ .  
 En effet, la RF entre produit 3 et quantité stock étant permanente, à chaque changement de la valeur de quantité stock, une nouvelle occurrence de produit 3 est créée.

Plusieurs occurrences de produit 3 correspondent à une occurrence de produit 1, et nous ne pourrions pas regrouper les c.objets dans une même c.classe.

#### 4 - C. CLASSES

Nous poursuivons notre analyse par l'étude des liens qui existent entre les c.objets. L'agrégation des mots en c.objets nous autorise à reporter l'étude des associations entre mots au niveau des liens qui existent entre les identifiants de c.objets.

#### 4.1. - Collection de c.objets

Une collection de c.objets est un ensemble non vide de c.objets. Dans une collection de c.objets, les liens existant entre c.objets ne respectent aucune contrainte.

#### 4.2. - Groupe de c.objets

Une collection de c.objets forme un groupe quand il existe des relations fonctionnelles stables, élémentaires et réciproques entre les identifiants de c.objets.

##### Définition

Nous disons que A et B sont en RF élémentaire stable et réciproque s'il existe des RF élémentaires  $A \rightarrow B$  et  $B \rightarrow A$ .

Par commodité de langage, on parlera de RF entre c.objets et non plus de RF entre identifiants de c.objets.

##### Exemple de groupe de c.objets

- Dans l'étude de l'application "facturation et réapprovisionnement" on a été conduit à définir les trois c.objets :

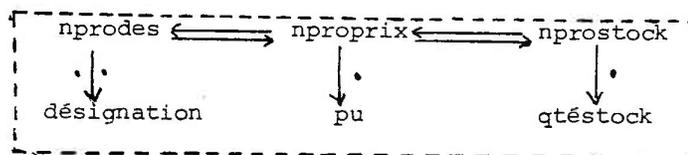
nprodes	
↓	
•	définissant la désignation d'un produit
↓	
désignation	

nproprix	
↓	
•	définissant le prix unitaire du produit
↓	
pu	

nprostock	
↓	
qtéstock	caractérisant l'état en stock du produit

Les relations fonctionnelles élémentaires stables et réciproques entre nprodes, nproprix et nprostock permettent d'associer ces trois

c.objets dans un même groupe.

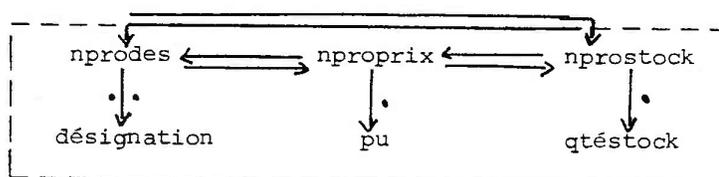


### Représentation

Un groupe de c.objets est encadré d'un rectangle en pointillé.

Remarquons que dans un groupe de c.objets, les RF peuvent ne pas être directes.

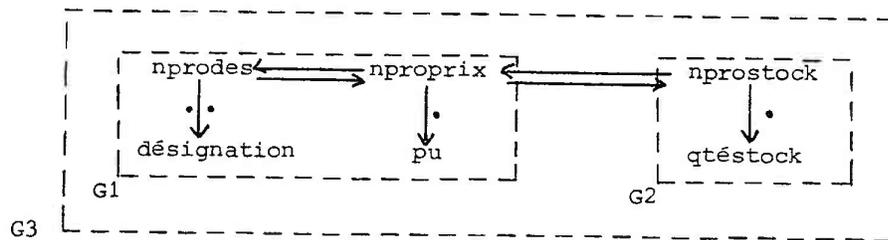
Dans l'exemple cité, il existe une RF entre nprostock et nprodes.



### 4.3. - Groupe maximal de c.objets

Un groupe de c.objets est un groupe maximal si tous les c.objets en RF stable élémentaire et réciproque appartiennent au même groupe.

Supposons que dans l'exemple précédent, on ait été conduit à construire les deux groupes G1 et G2 suivants :



G1 et G2 ne sont pas des groupes maximaux.

G1 ne contient pas nprostock, et il existe une RF stable, élémentaire et réciproque entre nproprix et nprostock et nprostock et nproprix.

Le même raisonnement peut être fait pour G2.

En revanche, le groupe G3 est un groupe maximal de c.objets

#### 4.4. - C.Classe de c.objets

##### 4.4.1. - Définition de la c.classe de c.objets

Un groupe de c.objets  $C_i$  forme une c.classe lorsqu'il est possible de définir un c.objet particulier, le c.objet racine, tel que :

- 1°) il soit en RF élémentaire, stable, directe et réciproque avec tous les autres c.objets  $C_i$
- 2°) Il ne contienne aucun mot non identifiant.
- 3°) La durée de vie d'une occurrence  $r$  du c.objet racine est  $(t_i, t_f)$ ,  $t_i$  et  $t_f$  étant ainsi définis :

Soient  $c_i$  les occurrences des c.objets correspondant à l'occurrence  $r$ ,

- .  $t_i$  l'instant de création de la première occurrence (au sens chronologique)  $c_i$
- .  $t_f$  l'instant de suppression de la dernière occurrence (au sens chronologique)  $c_i$

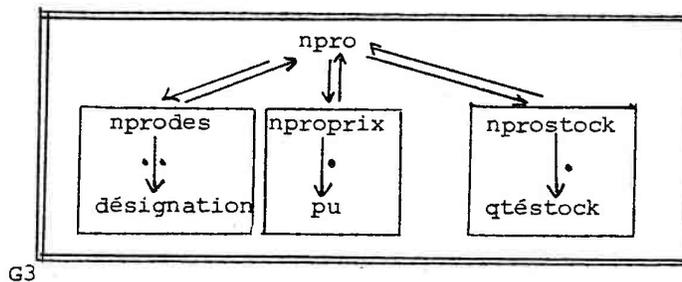
La présence d'un c.objet racine normalise les structures existant entre les c.objets d'un groupe, en créant une hiérarchie.

### Représentation de la c.classe

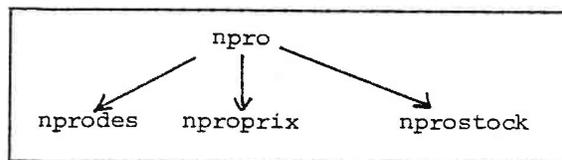
Une c.classe est graphiquement représentée dans un double rectangle encadrant l'ensemble des c.objets de la c.classe.

Chaque c.objet, y compris le c.objet racine y sont représentés par le graphisme habituel.

Exemple : la c.classe obtenue après normalisation du groupe G3 défini page est :



Le c.objet racine, qui confère au groupe la structure de c.classe est :



Remarque : nous admettons qu'un c.objet unique constitue une c.classe.

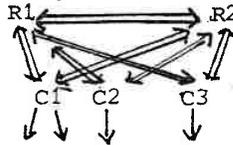
#### 4.4.2. - Unicité du c.objet racine d'une c.classe

Nous démontrons l'unicité par l'absurde :

- Supposons qu'il existe deux c.objets racine d'une c.classe, les c.objets R1 et R2.

Il découle des deux premières parties de la définition que R1 et R2 sont constitués du même ensemble de mots, celui des identifiants de c.objets.

Exemple :



Nous schématisons les RF réciproques par  $\longleftrightarrow$

Les c.objets identifiés par R1 et R2 contiennent les éléments R1, R2, C1, C2, C3

D'après la troisième partie de la définition, les occurrences de R1 et R2 ont même durée de vie ; les identifiants sont donc reliés par des RF permanentes. Il n'y a donc pas deux c.objets distincts, mais un seul c.objet possédant deux identifiants.

#### 4.4.3. - Identifiant de c.classe

Nous appelons identifiant de c.classe, l'identifiant du c.objet racine.

Tout comme l'identifiant du c.objet, l'identifiant de c.classe est en général unique.

#### 4.4.4. - Ordre des c.classes

Nous disons qu'une c.classe est d'ordre 1 si son identifiant est d'ordre 1.

Nous disons qu'une c.classe est d'ordre n si son identifiant est d'ordre n.

#### 4.4.5. - C.objet constituant d'une c.classe

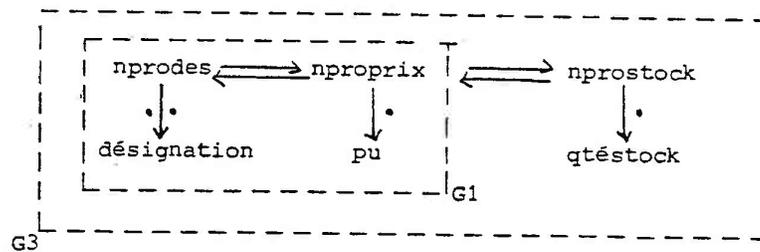
Nous appelons c.objet constituant d'une c.classe tout c.objet appartenant à une c.classe et n'étant pas c.objet racine.

#### 4.5. - C.Classe maximale

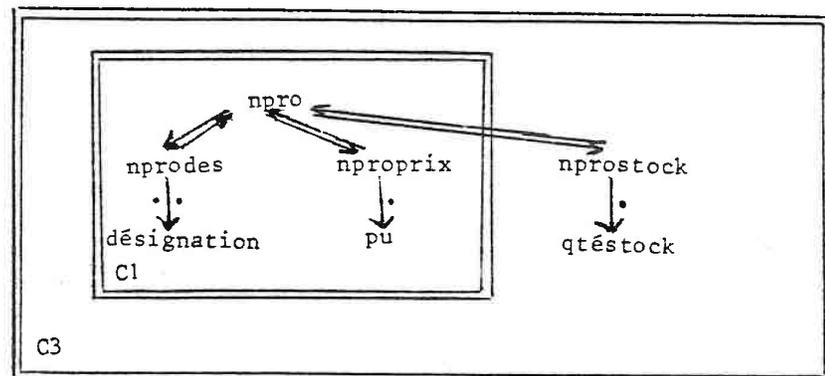
##### 4.5.1. - Définition

La c.classe maximale est une c.classe définie à partir d'un groupe maximal.

Exemple : Soient les groupes de c.objets G1 et G3 :



La c.classe C1, obtenue après la normalisation du groupe G1 n'est pas maximale, alors que la c.classe C3, obtenue après normalisation du groupe G3 l'est.

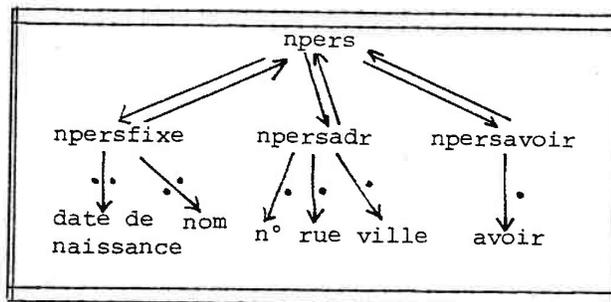


4.5.2. - Sémantique

- a) La c.classe est une représentation conceptuelle d'objets ou d'associations d'objets du monde réel

. La c.classe d'ordre 1 représente des objets. Le produit est représenté, dans l'exemple précédent par la c.classe identifiée par npro.

. la personne est représentée par la c.classe



La représentation est claire, précise. Tous les états de l'objet apparaissent. On crée autant de c.objets qu'un objet de la réalité subit de modifications à la suite d'événements distincts qui ne se produisent pas aux mêmes instants.

Pour un objet qui a été repéré par le concepteur, on crée autant de c.objets appartenant à une même c.classe que l'on peut définir de rythmes temporels distincts dans la vie de l'objet.

Ainsi, dans la c.classe produit, les trois c.objets identifiés par nprodes, nproprix et nprostock caractérisent trois aspects de l'objet produit intervenant à différents moments et dans des usages différents.

- nprodes identifie le produit pendant toute sa vie
- nproprix définit le prix unitaire à tout instant de la vie

du produit

- nprostock caractérise l'état du stock actuel du produit.

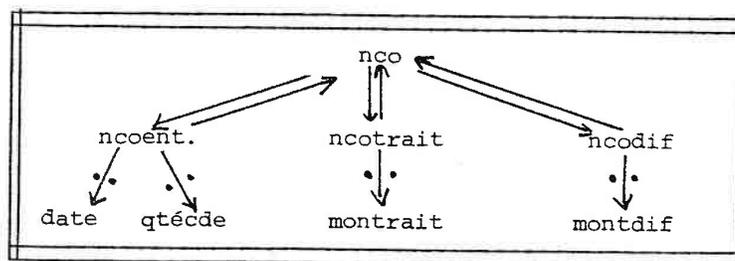
De même, dans la classe personne, les trois c.objets identifiés par npersfixe, npersadr, npersavoir définissent trois aspects distincts d'une personne ; celui qui l'identifie (npersfixe), celui qui caractérise sa localisation actuelle (npersadr), celui qui précise sa situation financière vis-à-vis de l'organisation (npersavoir).

Le rôle joué par les c.objets est encore plus évident lorsqu'on prend le cas de c.objets définis par des mots de nature résultat.

Prenons par exemple le cas du secteur commercial d'une entreprise de vente par correspondance, et cherchons à définir la classe des commandes.

Toute commande passée par un client peut être, soit traitée immédiatement, soit différée si le stock concernant le produit commandé par le client est insuffisant. Dans l'un et l'autre des cas, la commande est valorisée et le montant du produit soit traité, soit différé, est calculé.

La c.classse obtenue dans cet exemple simplifié pourrait être :



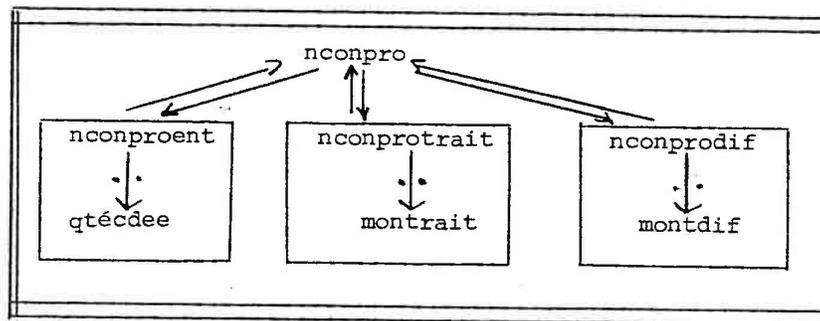
Identifiée par nco, repère habituel de l'objet "commande" perçu par les gestionnaires, la c.classse fait apparaître les trois états que peut avoir, à un instant donné, une commande reçue par l'organisation.

- le c.objet ncoent représente l'état initial de la commande, celui de sa réception par l'entreprise
- le c.objet ncotrait représente l'état "accepté et traité" d'une commande en cours de livraison
- le c.objet ncodif représente une commande valorisée mais en attente.

La c.classe d'ordre n représente des associations entre objets.

Supposons, dans l'exemple précédent, que la commande porte sur plusieurs produits. Dans ce cas, chaque association produit-commande peut être différée ou traitée.

La c.classe obtenue pourrait être



L'association produit-commande est représentée par la c.classe identifiée par le couple (nconpro)

La c.classe fait apparaître les trois états qui peut avoir, à un instant donné, cette association.

- le c.objet nconproent représente l'état initial de l'association,

- le c.objet nconprotrait représente l'état accepté et traité de l'association,

- le c.objet nconprodif représente l'état différé de l'association.

b) Cette structuration va permettre, dans le cadre du modèle conceptuel général, proposé dans Remora, une représentation complète de la dynamique des c.objets et à travers cela, une représentation complète de la dynamique de l'organisation.

En effet, la représentation d'un objet réel par une c.classe exprime que cet objet a en fait plusieurs états temporels possibles et les définit, mais ne dit pas quelles sont les relations qui existent entre eux.

Ces relations existent et il est intéressant de les traduire. Certaines sont en partie définies par les relations fonctionnelles entre les objets.

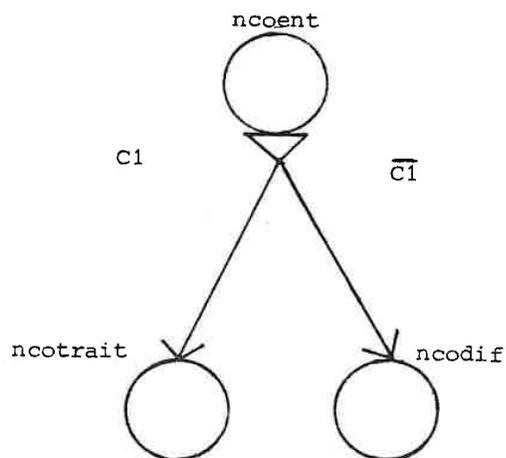
Par exemple, certaines RF entre les c.objets expriment une chronologie la RF ncotrait → ncoent exprime le fait que la commande a été traitée après avoir été prise en compte.

Mais toutes les RF n'expriment pas une chronologie. Par exemple la RF entre ncotrait et ncodif n'en exprime pas.

Pour exprimer toute la dynamique des c.objets de manière précise il faut faire appel à d'autres concepts : ce sont les concepts d'opérations et d'évènements du modèle conceptuel proposé dans l'équipe par O. Foucaut dans (15) et présenté dans (7) •

Ils permettent d'exprimer toutes les causes et toutes les conséquences des changements d'état des c.objets en mettant en évidence, d'une part, les liens de causalité entre les c.objets d'une c.classe, d'autre part (nous le verrons dans la présentation du système de c. classes) les liens de causalité entre c.objets de plusieurs classes en RF.

Ainsi dans l'exemple des commandes, nous pourrions associer à la classe commande, ce que nous appelons la structure fonctionnelle suivante :



L'arrivée de la commande est un évènement qui déclenche conditionnellement :

- soit l'opération conduisant à la production de la commande traitée lorsque la condition C1 (quantité en stock du produit  $\geq$  quantité commandée) est satisfaite.
- soit l'opération conduisant à la production de la commande différée lorsque la condition  $\overline{C1}$  est satisfaite.

Les concepts du c.objet et de c.classe sont indispensables pour disposer d'une structuration précise des constituants du SI. Ils sont nécessaires à une représentation des objets de l'organisation qui prenne en compte le temps et fasse apparaître les rythmes de leurs transformations. Ils sont insuffisants pour traduire seuls ce que sont ces transformations et leurs causes.

Combinés aux concepts de c.opération et de c.évènements, ils autorisent alors une représentation complète des phénomènes réels.

4.5.3. - Unicité de la c.classe maximale définie à partir des c.objets formant un groupe

Un seul groupe maximal peut être constitué à partir d'un groupe. Un unique c.objet racine est défini à partir de ce groupe maximal. On en déduit donc que la c.classe est définie de façon unique.

4.5.4. - Conséquences de la définition de la c.classe

Il est logique, compte tenu de nos concepts, que tout mot d'une c.classe n'appartenant pas à un identifiant de c.objet n'appartienne qu'à un et un seul c.objet. Le non respect de cette contrainte d'intégrité au cours de la construction du SC est dû à une anomalie.

- Les mots appartenant à des c.objets différents d'une même c.classe ont des évolutions et des durées de vie différentes.

- Un mot représente un élément du monde réel, et cet élément a une évolution et une durée de vie unique.

- L'appartenance d'un mot non identifiant à deux c.objets différents est donc due à une anomalie. Cette anomalie étant provoquée :

- . soit par une polysémie, un mot désignant deux éléments différents du monde réel,
- . soit par une redondance dans la description d'opérations; cette redondance se produit lorsque le concepteur décrit plusieurs fois l'opération définissant l'obtention d'un même mot résultat. Ce point est développé par Chesseron.

Cette contrainte comme l'a montré Flory dans (14) est intéressante puisqu'elle évite des erreurs de gestion de la base des données qu'une collection de relations en 3FN laisse subsister.

Exemple : soient les relations en 3° FN :

R1 (n° du service, délégué du personnel)

R2 (n° d'ouvrier, délégué du personnel).

R1 exprime qu'il n'existe qu'un délégué du personnel par service.

R2 exprime que tout ouvrier est représenté par un seul délégué du personnel.

Les occurrences de délégué du personnel peuvent très bien être différentes alors qu'elles devraient être identiques parcequ'on a modifié à des instants différents dans la base des tableaux de valeurs associées à la relation R1 et à la relation R2.

#### 4.5.5. - Structure de la c.classe

- a) Il découle de la définition que toutes les RF entre identifiants de c.objets constituants sont obtenues par transitivité

Pour tout  $\text{ident}_i$ ,  $\text{ident}_j$ ,  $\text{ident}_i$  et  $\text{ident}_j$  étant les identifiants de c.objets constituants distincts d'une même c.classe, la R.F.  $\text{ident}_i$  et  $\text{ident}_j$  s'obtient par la composition suivante :

$$\text{ident}_i \rightarrow \text{identifiant c.classe} \rightarrow \text{ident}_j$$

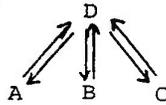
- b) Cette régularisation des RF entre c.objets constituants nous permet de donner un même rôle à chacun des c.objets constituants

Nous ne voulons pas introduire dans le SC de RF non directes. Or, la présence d'un identifiant de c.classe est nécessaire, car la solution qui consiste à admettre des RF directes entre c.objets conduit à en particulariser certains, cette particularisation n'ayant aucune signification dans le monde réel (nous le montrons dans les parties c et d).

Ainsi, soient trois c.objets constituants, identifiés par A, B, C, la représentation  $A \rightleftharpoons B \rightleftharpoons C$  particularise B en le rendant but de 2 RF et source de 2 RF. D'autres représentations auraient pu être retenues . Par exemple :

$$B \rightleftharpoons A \rightleftharpoons C \quad \text{ou} \quad A \rightleftharpoons C \rightleftharpoons B$$

La représentation par la c.classe



évite d'affecter à un des objets un poids plus fort qu'aux autres.

### c) Sémantique

La c.classe permet la représentation d'objets.

Les relations qui permettent d'identifier les états d'un même objet sont représentées dans le SC par des RF directes entre identifiants de c.classe et identifiants de c.objet constituants.

Les RF entre c.objets constituants ne sont pas directes. Nous avons unifié leur représentation. Pourtant, à priori, il semblait normal de particulariser certaines relations temporelles. La chronologie étant une notion intéressante, on pourrait imaginer de représenter les liaisons chronologiques par des RF directes entre c.objets constituants. Mais, comme nous le montrerons plus loin (p. 76 ), cette contrainte soit ne peut pas être respectée, soit entraîne une mauvaise représentation du réel, soit permet la représentation d'une définition donnée du monde réel par plusieurs schémas conceptuels. Aussi, ne pouvant pas particulariser, dans le SC, les RF représentant une chronologie, nous normalisons la représentation des RF entre c.objets constituants.

d) Inconvénients dûs à l'absence de normalisation des RF  
entre c:objets constituants

Nous nous appuyons sur un exemple pour montrer que la contrainte qui consiste à représenter les liaisons chronologiques par des RF directes entre c:objets constituants :

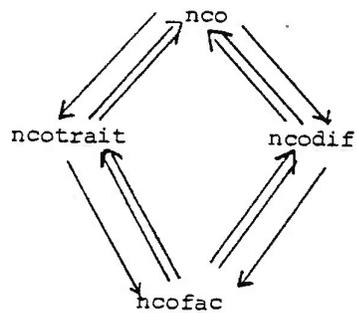
- . soit ne peut pas être respectée,
- . soit entraîne une mauvaise représentation du réel,
- . soit permet la représentation d'une définition donnée du monde réel par plusieurs SC.

Exemple : Dans la facturation d'une entreprise de vente par correspondance, les commandes passées par les clients sont traitées ou différées lorsque le produit commandé est stocké en quantité insuffisante. Une commande peut être partiellement traitée et partiellement différée. La commande différée est traitée après le réapprovisionnement des produits épuisés.

Des remises sont accordées aux clients en fonction des délais d'attente. La facture de la commande, évaluée à partir de la commande traitée et de la commande différée, n'est envoyée au client qu'après le traitement complet de la commande.

Les liaisons entre les différents états de la commande sont représentées dans le SC sous la forme de RF définies uniquement entre c:objets constituants.

Le schéma suivant représente les liaisons entre les différents états de la commande.



$\alpha$  - Toutes les liaisons chronologiques sont représentées par des RF apparaissant avec un double trait.

Par exemple, la RF  $\text{ncotrait} \rightleftarrows \text{nco}$  signifie que la commande a été traitée après avoir été prise en compte.

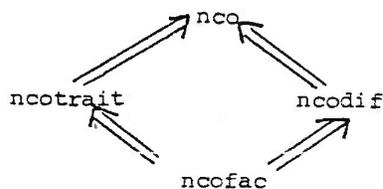
Sur cet exemple, nous voyons qu'aucune RF n'est directe, et en particulier, les RF exprimant la chronologie ne le sont pas. Ainsi, la RF  $\text{ncofac} \rightarrow \text{ncodif}$  ne l'est pas, car il existe  $\text{ncotrait}$  et  $\text{nco}$  tels que  $\text{ncofac} \rightarrow \text{ncotrait} \rightarrow \text{nco} \rightarrow \text{ncodif}$ .

La contrainte, consistant à particulariser les RF exprimant la chronologie en les rendant directes n'est pas respectée.

De plus, il y a une redondance dans la représentation, due à la présence de RF non directes.

$\beta$  - Une possibilité, nous permettant de représenter toutes les liaisons chronologiques par des RF directes, consiste à n'introduire que les RF chronologiques. Mais dans ce cas, le SC donne une représentation incomplète du réel.

Le graphe des relations entre c. objets constituants devient :



Toutes les associations entre objets du monde réel ne sont pas représentées. Ainsi, par exemple, le fait qu'à une commande ne corresponde qu'une commande traitée, n'est pas exprimé.

γ - Une autre possibilité était de choisir arbitrairement l'un ou l'autre des schémas suivants, ce qui particularise arbitrairement certains c.objets constituants.

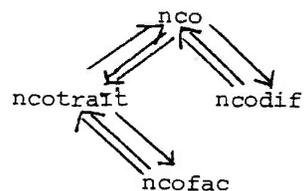


Schéma 1

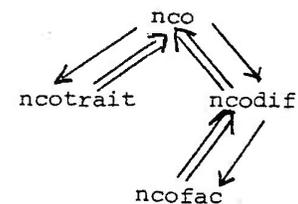


Schéma 2

Toutes les liaisons sont exprimées ; toutes les RF représentées sont directes. Mais, d'une part, une RF exprimant la chronologie n'est pas représentée, ce qui est contraire à la contrainte que nous avons définie ; d'autre part, n'ayant pas de critère pour effectuer notre choix, nous ne pouvons pas faire correspondre de façon unique un schéma conceptuel à une définition du monde réel.

#### Avantages présentés par la notion de c.classe

##### 1) - Facilité de lecture

Dans le SC, le nombre des c.objets est très important. Aussi, il semble naturel de les grouper, ce groupe étant significatif et ayant une structure qui permet de repérer facilement chacun de ses constituants.

2) - Suppression des circuits entre les c.objets constituants et d'un même groupe, ne particularisant aucun c.objet

Dans l'exemple précédent, nous montrons les difficultés soulevées par la présence d'un circuit de RF "réciproques".

En effet, soit nous acceptons le schéma 1 et dans ce cas il y a redondance de RF et la présence de circuits nécessite l'usage d'algorithmes compliqués au niveau de la méthode,

soit nous le refusons, et nous n'avons pas de critères pour choisir un schéma ; les schémas 2, 3, 4 pourraient être retenus.

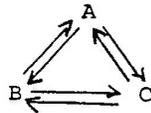


Schéma 1

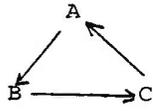


Schéma 2

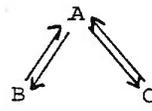


Schéma 3

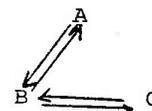


Schéma 4

3) - Facilité de structuration, de modification du SC

Il n'existe pas de relation de dépendance entre les c. objets constituants. L'insertion ou la suppression de c.objets constituants n'entraînera aucune modification des autres c.objets constituants.

Exemples

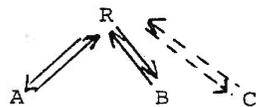


Schéma 1  
c.classe

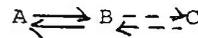


Schéma 2  
Autre solution  
destruction



Schéma 2 bis

Insertion :

- Dans le schéma 1, l'insertion de A modifie le c.objet racine R; elle ne modifie pas les c.objets constituants A ou B.
- Dans le schéma 2, l'insertion de C modifie le c.objet constituant B.

Suppression :

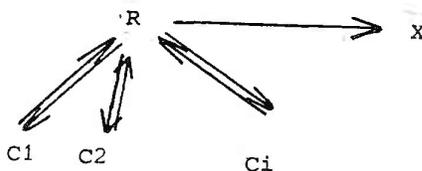
- Dans le schéma 1, la suppression de B modifie R, alors que dans le schéma 2, elle modifie A et C (suppression des RF  $A \rightarrow B$   $C \rightarrow B$  qui sont remplacées par  $A \rightarrow C$  et  $C \rightarrow A$  ; le schéma 2bis représente la structure obtenue).

4)- Absence des valeurs inconnues dans un c.objet constituant

Tous les mots du c.objet constituant sont valorisés en même temps que l'identification au c.objet ; l'identifiant de c.classe qui est connu au moment de la création du c.objet est également valorisé.

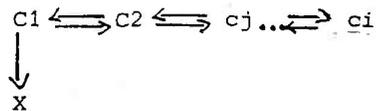
5)- Facilité dans l'exploitation du SI

Dans une c.classe, l'accès, à partir de  $C_i$  à un élément X, extérieur à la c.classe, s'effectue toujours selon le même mode, en passant par un seul intermédiaire, R.



Nous voyons que, dans toute autre structure de groupe, l'accès à X, à partir de  $C_i$ , est moins aisé.

Exemple : L'accès à X, à partir de  $c_i$ , nécessite l'accès successif à chaque élément  $c_j$   $1 \leq j < i$ .



6) - Permet d'exprimer des relations entre objets

Dans le monde réel, nous exprimons des relations entre objets, sans qu'il soit nécessaire de se référer à leurs états temporels.

Les c.classes et leur structuration en système de c.classes, permettent d'exprimer cette liaison.

5 - SYSTEME DE C.CLASSES

Les c.classes ne sont pas indépendantes. A l'image des objets du réel qu'elles représentent, elles ont entre elles des relations que nous allons étudier.

5.1. - Définition

Soit M un ensemble de mots, et E un ensemble de c. classes maximales formées sur (M, F)

E est un système de c.classes si les propriétés suivantes sont vérifiées.

- a) tout mot de M appartient au moins à une c.classe
- b) tout mot de M qui appartient à plusieurs c.classes :
  - . soit forme l'identifiant simple d'un c.objet
  - . soit n'est but d'aucune RF non triviale
- c) il n'existe pas de RF directe entre les identifiants de c.objets constituants.

Les conditions a et b sont semblables à celles que vérifient les groupes primaires d'un système de groupe primaire ( 14 ).

#### 5.2. - Exemple de système de c.classes

##### Gestion et facturation simplifiée d'une commande

Dans cet exemple, la commande passée par un client comporte plusieurs lignes de commandes, chacune de ces lignes étant associée à un produit. Une ligne de commande initiale peut être, soit acceptée, soit différée. Après le traitement de chacune de ses lignes, la commande initiale est préfacturée.

La facture de la commande est calculée après la livraison de la commande préfacturée.



### 5.3. - Sémantique du système de c.classes

Le système de c.classe est l'image de la structure naturelle des objets de l'organisation.

Ce système représente :

- 1) Les objets et associations d'objets de l'organisation
- 2) Les relations structurelles existant entre eux et qu'on a décidé de prendre en compte dans le SI.

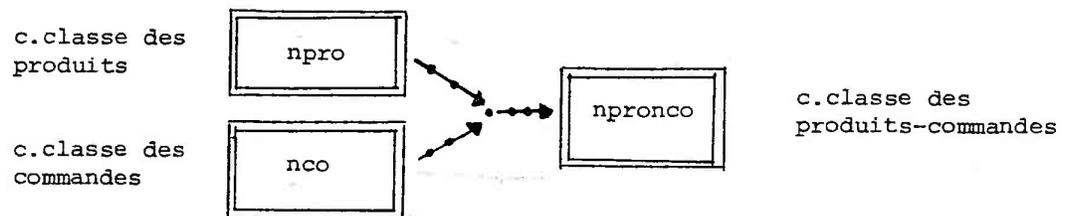
1) Chaque c.classe est la représentation d'objets ou d'associations d'objets du monde réel.

#### Les c.classes d'ordre 1 représentent des objets

- La c.classe produit, identifiée par un numéro de produits (npro) regroupe les trois c.objets caractérisant les trois aspects de l'objet produit.
- La c.classe commande, identifiée par un numéro de commande (nco) regroupe les quatre c.objets représentatifs des états successifs d'une même commande dans l'organisation : la commande initiale (ncoent), la commande acceptée (ncotrait), la commande livrée (ncoliv), la commande facturée (ncofac).
- La c.classe client, identifiée par un numéro de client (ncli) regroupe quatre c.objets définissant les quatre aspects du client.
- La c.classe entreprise, identifiée par un numéro d'entreprise (nent), comporte un seul c.objet dont les constituants représentent les paramètres de la gestion de l'entreprise.

Les c. classes d'ordre n représentent des associations entre objets

- La c. classe produit-commande, identifiée par un numéro de produit et un numéro de commande représente une association entre un produit et une commande.
- L'association entre produit et commande est mise en évidence dans le système de c. classe par le schéma suivant :



La c. classe des produit -commande regroupe trois c. objets significatifs des états possibles d'une même association produits-commandes :

produit -commande initial (nprocoent),  
 produit-commande traité (nproncotrait),  
 produit-commande différé (nproncodif).

2) Les associations entre objets sont représentées par des RF entre identifiants de c. classes

Exemples : la RF nco → ncli définit la liaison entre commande et client.  
 la RF nco → nent définit la liaison entre une commande et l'entreprise  
 la RF npro → nent définit la liaison entre un produit et l'entreprise.

#### 5.4. - Unicité du système dec.c.classes sur un ensemble de mots

Un système de c.classes est un SGP. Flory dans (14) démontre que "si E est sans circuit, il existe une unique manière de construire un SGP sur C".

E est l'ensemble des sous relations fonctionnelles d'une relation R définie sur C, ensemble des mots.

De même nous pouvons dire : E étant sans circuit, il existe une unique manière de construire un système de c.classes sur C.

De plus, nous avons vu que la présence de circuits entre c. objets d'une même c.classe ne supprimait pas l'unicité de la c.classe.

Il découle de la condition c de la définition que seules les RF entre c.classes permettent d'exprimer les relations entre c.objets de différentes c.classes.

Nous en déduisons donc que :

E' étant le sous ensemble de RF définies entre identifiants de c.classes, E'cE, si E' est sans circuit, alors il existe une unique manière de construire un système de c.classes sur C.

#### 5.5. - Avantages du système de c.classes

##### 1) Lisibilité

Le système de c.classes est une représentation claire précise des constituants de l'organisation et de leurs relations structurelles.

## 2) Facilité pour la méthode

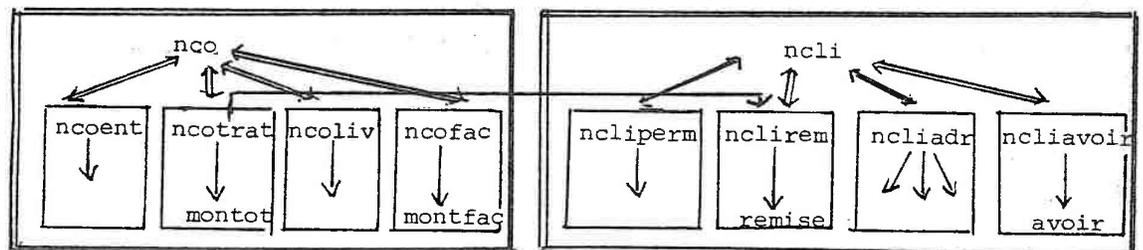
Les relations, entre objets sont représentées par des RF entre identifiants de c.classes.

Dans la méthode, le raisonnement s'effectue au niveau d'agrégats, ce qui permet d'économiser la demande auprès du concepteur des RF qu'il doit fournir pour permettre la structuration. En effet, le nombre de RF pouvant exister entre identifiants de c.classes est beaucoup plus faible que le nombre de RF pouvant exister entre c.objets constituant de différentes c.classes.

De plus, un raisonnement au niveau des c.objets peut conduire pour une même définition du monde réel, à plusieurs structures du SC.

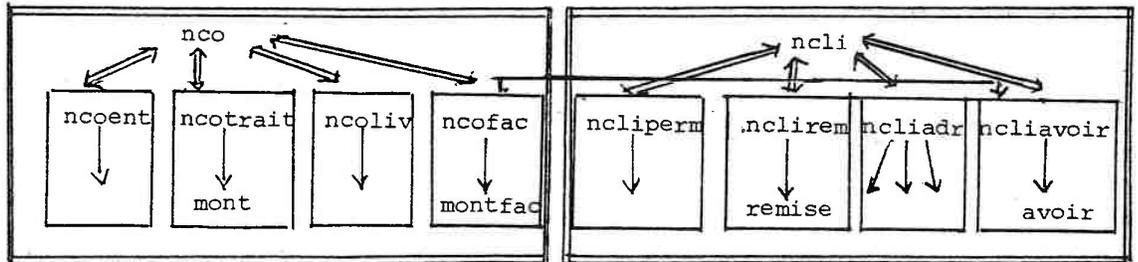
Ainsi, dans l'exemple précédent, le RF  $nco \rightarrow ncli$  exprime la liaison entre une commande et un client. Il existe une seule RF entre les deux c.classes.

Un raisonnement au niveau des c.objets pouvait mener indifféremment à l'une ou l'autre des structures suivantes :



Dans cette structure, le fait qu'à une commande traitée correspond une remise du client est mis en évidence.

Dans la structure suivante, le fait qu'à une facturation correspond de un nouvel avoir du client est mis en évidence.



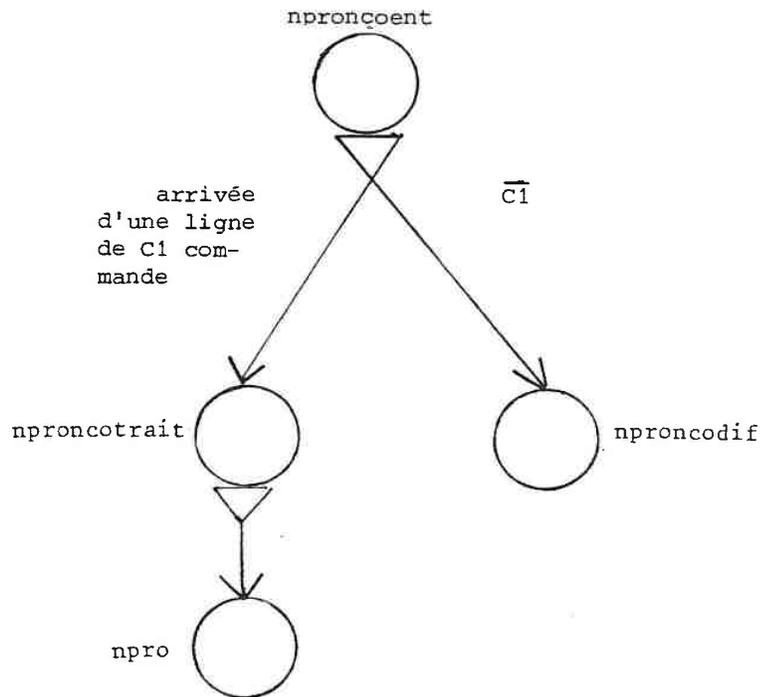
Les structures obtenues

- a) sont moins claires
- b) dépendent des traitements.

3) La sémantique qu'on peut lui associer et l'aspect fonctionnel du système de c.classes

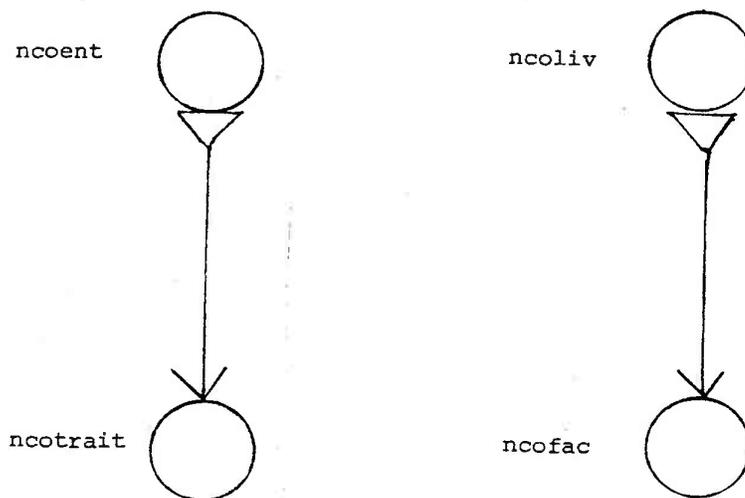
Le système de c.classes est insuffisant pour exprimer seul et complètement la dynamique de l'organisation. Comme nous l'avons fait au niveau de la c.classe, il faut rapprocher le système de c.classes de la structure fonctionnelle ou structure des interrelations entre c.objets, c.opérations et c.événements. Alors, on peut faire apparaître des relations de fonctionnement entre les objets différents.

Une partie de la structure fonctionnelle associée au système de c.classes définie page 82 sera par exemple :



- L'arrivée d'une ligne de commande déclenche l'opération d'acceptation de commande, puis conduit suivant le cas :

- . (C1 : qtécde  $\geq$  qtéstock)  
à l'opération de création d'une ligne commande traitée. La présence du c.objet nproncotrait dans le système est un évènement qui déclenche l'opération de modification du produit.
- . ( $\overline{C1}$  : qtécdee < qtéstock)  
à l'opération de création d'une ligne de commande différée.



- L'arrivée de la commande est un évènement qui déclenche conditionnellement (toutes les lignes de commande ont été traitées) l'opération préfacturation conduisant à la production d'une commande traitée.
- La livraison d'une commande préfacturée déclenche l'opération de facturation et la création de l'objet commande facturée.

Le premier graphe de la structure fonctionnelle exprime les relations de fonctionnement entre les c.objets de la c.classe ligne de commande et ceux de la c.classe produit.

Dans le second graphe, la liaison entre la c.classe commande et la c.classe ligne de commande est exprimée par une condition.

#### 5.6. - Type des mots vis-à-vis du système de c.classes

De la structuration d'un ensemble E de mots proposés, on peut conclure :

Tout mot du système de c.classes a un type vis à vis de ce système. Les trois types que nous définissons correspondent aux types index, propriété et paramètres définis par Flory dans (14). Ils s'excluent mutuellement.

1) Type identifiant

Dans un système de c.classes donné, tout mot identifiant du c.objet d'ordre 1 est du type identifiant.

2) Type constituant

Dans un système de c.classes donné, un mot but d'une et d'une seule RF élémentaire, directe et n'appartenant à aucun identifiant est du type constituant.

3) Type paramètre

Paramètre simple : Dans un système de c.classes donné, tout mot ni but ni source de RF est paramètre simple.

Paramètre d'association : Dans un système de c.classes donné tout mot non but de RF, non identifiant de c.objet d'ordre 1, composant un identifiant d'ordre n est un paramètre d'association.

6 - CONCLUSION

Le modèle que nous proposons a le même objectif que le modèle qui est à l'origine de la plupart des travaux dans ce domaine.

Comme lui, il permet de traduire la structure des objets d'une organisation par un schéma conceptuel, mais il a un contenu sémantique plus grand et permet de mieux traduire un phénomène réel. L'amélioration

porte sur trois points.

1) Dans un système de c.classes un mot non identifiant n'est présent qu'une fois, ce qui n'est pas le cas dans une collection en 3° FN et est une source possible d'erreurs dans la base de données.

2) Dans un système de c.classes, l'ordre de l'identifiant de la c.classe permet de savoir si la c.classe représente un objet du monde réel ou une association d'objets.

3) Dans un système de c.classes, le temps est pris en compte et la dynamique des objets représentée.

Le système de groupes primaires, défini par Flory dans (14) comporte les deux améliorations 1 et 2. La seconde amélioration qui met en évidence les concepts d'entité et d'association a été démontrée dans de nombreux autres modèles CHEN ( 6 ) .

Le troisième aspect permet en le rapprochant des autres propositions de Remora de disposer d'un modèle conceptuel de SI original. La structuration des données que nous proposons est indispensable pour assurer parallèlement la structuration conceptuelle des traitements et la représentation de leur dynamique. Mais elle n'a pu être définie que parce que l'étude des trois composantes du SI qui sont les données, les traitements et leur dynamique, a été menée conjointement.

CHAPITRE III

LA METHODE DE CONSTRUCTION

DU SYSTEME DE C.CLASSES

Le processus méthodique que nous décrivons dans ce chapitre fait partie du processus de construction du SC et permet de construire l'ensemble des c.objets, des c.classes, et de les structurer en système de c.classes.

C'est un processus de structuration par composition qui, à partir d'une liste de mots extraite du SP et d'un ensemble de RF fournies par le concepteur ou déduites du SP par des automates, procède par agrégation des mots en c.objets, des c.objets en c.classes et des c.classes en système de c.classes.

Nous développons :

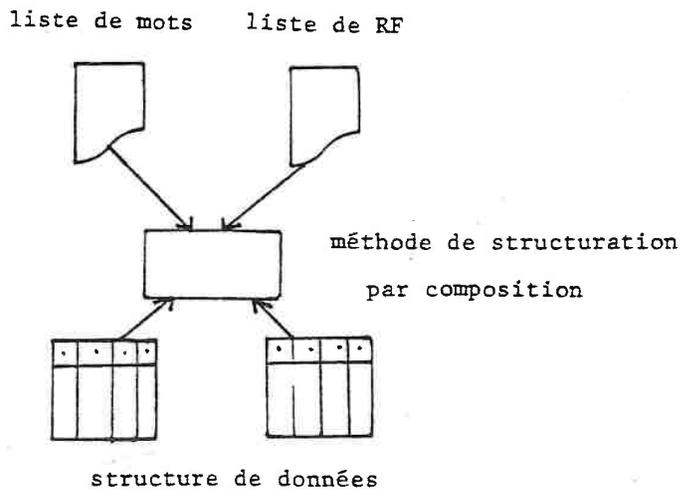
- 3.1. Les principes de la méthode
- 3.2. Les caractéristiques de la méthode
- 3.3. La méthode.

## 1. - PRINCIPES

Deux principes dirigent la structuration : la structuration est effectuée par une méthode de composition ; elle respecte le principe d'économie des échanges concepteur - système.

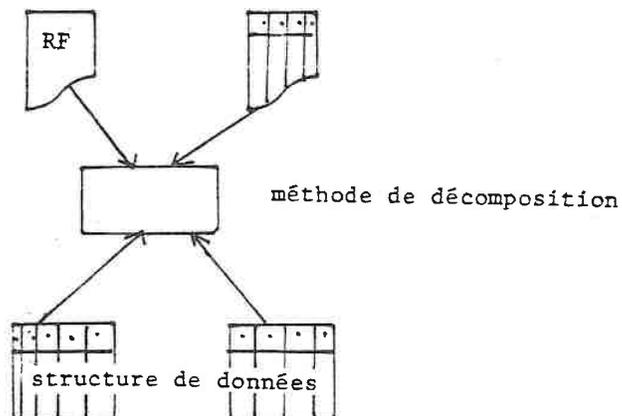
### 1.1. Principe de structuration par composition

Pour systématiser la structuration, nous avons divisé un procédé qualifié par Delobel de synthétique dans ( 13 ), qui, à partir d'une liste de mots et d'une liste de RF permet d'aboutir à une structure de données.



Ce procédé s'oppose aux méthodes de décomposition dérivées du processus de normalisation de Codd ( 9 ) qui, au contraire, prennent comme ensemble de départ un ensemble de relation et un graphe de RF.

Graphe de RF      ensemble de relations



- 1 - La liste des mots est déduite par des automates de la description des résultats de l'analyse de l'organisation sous forme d'énoncés du SP.  
Elle est éventuellement complétée par des identifiants introduits par le concepteur.
- 2 - La liste des RF est en partie fournie par le concepteur, en partie déduite des énoncés par des automates.

La structuration sera effectuée en trois niveaux :

- a - Structuration des mots en c.objets
- b - Structuration des c.objets en c.classes
- c - Structuration des c.classes en système de c.classes.

#### 1.2. Principe d'économie

Dans un procédé de structuration de ce type, il semble logique de s'appuyer, d'une part, sur les  $m^2$  RF existant entre les  $m$  mots à structurer et d'autre part, sur les  $p$  relations fonctionnelles pouvant exister entre entités (agrégat de mots). C'est la solution développée par exemple par Flory dans (14).

. Nous avons pensé qu'il était possible de limiter ce nombre et de réduire ainsi l'effort d'information exigible du concepteur.

. Cette économie de demande d'information est rendue possible car :

- Un SP complet permet la déduction automatique de certaines RF.
- Une exploitation rationnelle du SP (les procédures seront examinées dans l'ordre chronologique) permet de demander uniquement les RF ne pouvant être déduites.
- Une structuration progressive des données permet de raisonner au niveau des agrégats, au lieu de raisonner au niveau atomique.

## 2. - CARACTERISTIQUES

### 2.1. Le processus de structuration dépend de son domaine d'application

Cette caractéristique est une conséquence du principe d'économie.

Afin de minimiser les informations demandées au concepteur, nous différencions le processus de structuration des mots définis dans les opérations de celui des mots définis dans les messages.

Les opérations sont décrites par des modules sur les descripteurs de traitement. Les messages sont décrits sur des descripteurs de documents. Au moment de leur examen dans le processus de structuration, ils ont été les uns et les autres contrôlés et validés, par le biais du système d'aide à la conception et des automates définis dans les travaux d'Odile Thiery (29).

### 2.2. Le processus est sédimentaire

. L'analyse se fait progressivement, par couches successives dans le temps. Les sous-ensembles analysés à un instant donné sont des ensembles  $m$  de mots appartenant, soit à des messages, soit à des modules.

. La sédimentation s'accompagne d'intégration.

A un instant donné, un système de  $c$ .classes temporaire  $S_0$  a été constitué. L'analyse d'un nouvel ensemble  $m$  de mots de  $M$  a conduit à une sous-structure de  $c$ .classes  $s_m$ ; l'intégration de  $s_m$  à  $S_0$  conduit à un nouveau système de  $c$ .classes  $S_1$  qui sert de base à une nouvelle étape de structuration et d'intégration (figure 1 ).

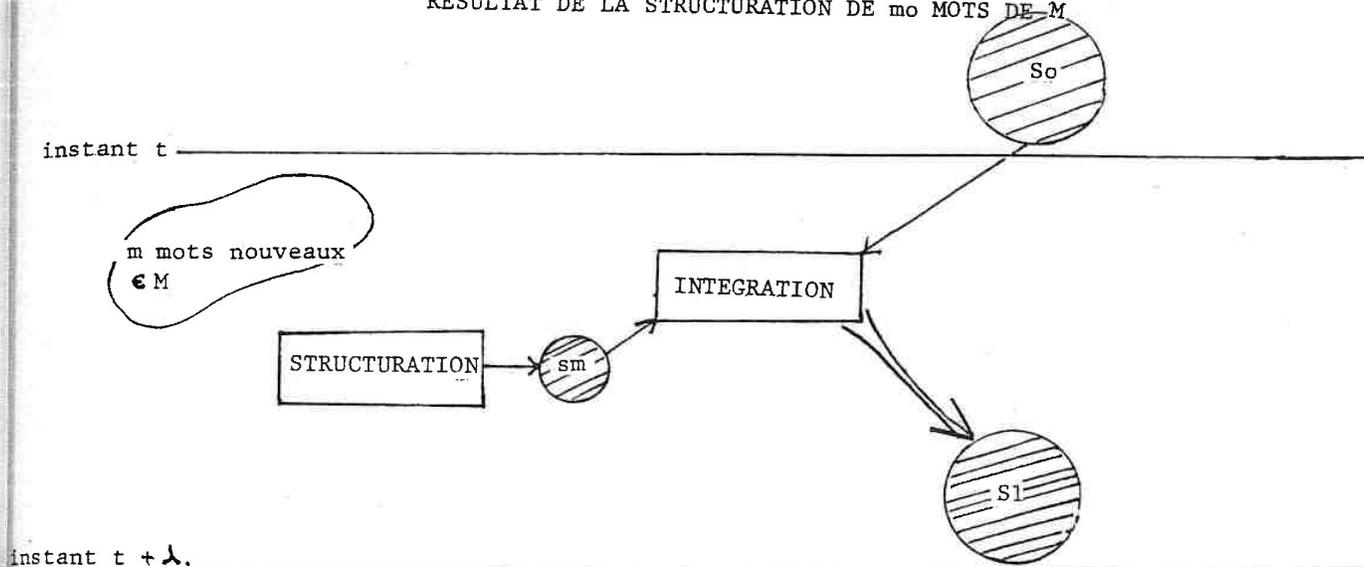
RESULTAT DE LA STRUCTURATION DE  $m_0$  MOTS DE  $M$ 

FIGURE 1

2.3. Le processus de structuration est assisté

Aucun aspect de conception dans Remora n'est extérieur au système d'aide à la conception placé sous le contrôle du Pilote. La conception des données n'y échappe pas.

Nous avons déjà fait mention des automates qui complètent la méthode. Le processus de structuration fait effectivement intervenir en alternance des hommes (les concepteurs) et des automates. Il est dirigé par le Pilote.

- Les hommes apportent des informations nouvelles : les relations fonctionnelles entre mots et effectuent des choix (identifiants, par exemple) que les automates sont incapables d'effectuer de manière rationnelle.

- Ils sont aidés par des règles méthodiques que nous décrivons et l'appel permanent qu'ils peuvent faire au système de documentation.

#### EXEMPLES

- Ecriture par le concepteur : a) des RF entre les  $m$  mots d'un message  
b) des RF réciproques, stables, élémentaires  
entre les identifiants de  $c$ .objets.

Ces écritures se font méthodiquement, à l'aide de matrices fournies par le système.

- Les automates exécutent des algorithmes correspondant, soit à des règles déductives, soit à des règles de cohérence.

#### EXEMPLES : Règles déductives

- Pour tout  $\text{mot}_i$ , mot de nature résultat non évalué à partir d'une liste, les RF identifiant  $c$ .objet  $\rightarrow$   $\text{mot}_i$  sont déduites.
- Pour tout  $c$ .objet de nature résultat, les RF identifiant  $c$ .classe  $\rightarrow$  identifiant  $c$ .objet sont déduites.

#### Règles de cohérence

- Pour tout  $\text{mot}_i$ , mot de nature résultat non évalué à partir d'une liste, les RF identifiant  $\rightarrow$   $\text{mot}_i$  sont contrôlées.
- Contrôle de l'appartenance d'un mot de type constituant à un seul  $c$ .objet, une seule  $c$ .classe.

### 3. - LA METHODE

#### 3.1. Présentation générale

Le problème de structuration de données tel que nous venons de le poser dans l'approche méthodique choisie s'énonce :

STRUCTURER UN ENSEMBLE DE MOTS M, A PARTIR D'UN ENSEMBLE RF DE RELATIONS FONCTIONNELLES SUR M, EN SYSTEME DE C.CLASSES

Pour décrire la solution choisie, il nous faut donc :

- 1 - définir M
- 2 - définir RF
- 3 - définir le raisonnement de structuration.

#### 3.2. Définition et constitution de l'ensemble des mots M

Définition : Il est l'ensemble des mots dont les valeurs constitueront la base de données. Ces mots représentent les propriétés des objets ou des associations d'objets que l'on a décidé de prendre en compte dans le système d'information.

Composition :  $M = MSP \cup MC$

MSP est l'ensemble des mots décrits dans le schéma de perception soit dans les modules soit dans les messages des procédures qui le composent ; MC est l'ensemble des mots identifiants que le concepteur est amené à rajouter au cours du déroulement du processus.

Type de mots de M : Dans M figurent soit des mots constituants, soit des mots identifiants.

Règle d'atomisation des identifiants : Tous les identifiants de M sont d'ordre 1.

Au cours du processus de structuration, le concepteur est amené, dans certains cas, à définir les identifiants. En vue d'alléger sa tâche, nous lui proposons d'introduire systématiquement les identifiants comme des mots simples. En d'autres termes, nous refusons que le concepteur fasse des a priori sur les regroupements d'identifiants à réaliser. Un automate l'aidera à effectuer ultérieurement cette tâche. Cette règle revient à proposer au concepteur d'ignorer au départ la notion d'association d'objets et de ne raisonner qu'en termes d'objets.

Dans le monde réel, la différenciation des objets et des associations d'objets n'est pas toujours évidente. De nombreux auteurs l'ont fait remarquer, par exemple Kent dans (18) ou Flöry dans (14) et ont cité des exemples.

Ainsi, la ligne d'une commande, qui est associée à une commande peut être perçue soit comme un objet, l'objet ligne d'une commande, soit comme une association, l'association produit-commande.

Au premier niveau de perception, nous supposons que les objets appréhendés sont des objets simples. C'est ultérieurement, aidé et documenté, que le concepteur percevra des associations d'objets.

Règle sur le type des mots d'un module : Tout mot intervenant dans une définition de calcul est du type constituant.

Signification : Un mot calculé ou donnée d'un calcul représente une propriété.

Exemple : soit la règle de gestion suivante :  $\text{monpro} = \text{qteliv} \times \text{pu}$ .

Elle définit le montant d'une ligne de commande comme le produit de la quantité livrée de l'article associé à cette ligne par le prix unitaire de cet article. Monpro, qteliv, pu représentent des propriétés d'objets et sont donc constituants de c.objets.

Conséquence de la règle :

Puisque tout mot intervenant dans une définition de calcul est du type constituant, il doit être rattaché à un identifiant.

Ainsi, la RF npro → pu entre le numéro de référence de l'article et son prix unitaire sera définie.

Construction de M :

- Construction de MSP : elle est automatique. Un automate explore la base des objets élémentaires (Implantation du SP) et construit MSP.
- Construction de MC : elle est d'abord manuelle, puisque le concepteur, au fur et à mesure du déroulement du processus, définit MC. Le système répertorie ensuite les nouveaux identifiants.

3.3. Définition et constitution de l'ensemble RF des relations fonctionnelles sur M

- Construction des RF identifiant c.objet → mot non identifiant :
  - . Elle est d'abord manuelle ; le concepteur définit les RF identifiant → mot pour tout mot de nature donnée, variable, ou résultat évalué à partir d'une liste.
  - . Elle est ensuite automatisée ; le système déduit les RF identifiant c.objet → mot, pour tout mot de nature résultat non évalué à partir d'une liste.

- Construction des RF identifiant c.classe → identifiant c.objet

- . Elle est totalement automatisée. A partir de l'exploration des RF entre identifiants de c.objets, le système déduit les RF entre identifiant de c.objet et identifiant de c.classes.

- Construction de RF entre identifiants de c.classes

- . Elle est d'abord manuelle ; le concepteur définit l'ensemble des relations entre identifiants de c.classes.
- . Puis, le système épure cet ensemble, en ne conservant que les RF élémentaires et directes.

3.4. Le raisonnement de structuration

3.4.1. - Présentation générale

La méthode choisie est sédimentaire, cela signifie que le processus de structuration s'applique à des sous-ensembles  $m$  de mots de  $M$  et que pour chaque sous-ensemble alternent une étape de structuration et une étape d'intégration qui peuvent être schématisées comme suit :

$$M = \bigcup_i m_i$$

Etape	Ensemble de départ	Structuration	Résultat; structuration système de c.classes $S_i$	Intégration	Résultat : intégration système de c.classes $SS_i$
1	$m_1$	struct ( $m_1$ )	$s_1$	intg ( $s_1$ )	$ss_1$
2	$m_2$	struct ( $m_2$ )	$s_2$	intg ( $s_1, s_2$ )	$ss_2$
3	$m_3$	struct ( $m_3$ )	$s_3$	intg ( $ss_2, s_3$ )	$ss_3$
$m$	$m_n$	struct ( $m_n$ )	$s_n$	intg ( $ss_{n-1}, s_n$ )	$ss_n$

La structure conceptuelle des données sur M est ssn.

Pour décrire ce raisonnement, nous présentons successivement :

- 1 - Les règles de décomposition de M en ensemble m
- 2 - Les règles de structuration proprement dites
- 3 - Les règles d'intégration
- 4 - Le déroulement du processus méthodique et les algorithmes.

#### 3.4.2. - Règles de décomposition de M en sous-ensembles m de mots

##### 3.4.2.1. - Incidence du principe d'économie et de cohérence : un ordre de structuration des sous-ensembles de mots de M

Le principe d'économie peut être assuré d'une part, parce que certaines RF sont déduites du SP par des automates, d'autre part, parce que les concepts du modèle autorisent de raisonner entre agrégats de mots (c.objets ou c.classes) au lieu de raisonner au niveau des mots qui les composent.

Il est intuitif que le degré d'application de ce principe soit dépendant de l'ordre dans lequel on va tenter de déduire les RF, de l'ordre des déductions et des demandes de RF au concepteur, de l'ordre dans lequel on va construire les agrégats successifs.

Nous définissons dans le paragraphe suivant les règles méthodiques de décomposition de M et donnons leur justification dans l'optique de l'économie maximum.

### 3.4.2.2. - Règles de décomposition de M

#### Règle 1 : Décomposer M en MD et MR et structurer d'abord MD

Décomposons M en deux sous ensembles disjoints MD et MR ;

$$M = MD \cup MR$$

MR =  $\{mr\}$  ensemble des mots de nature résultat non définis à partir d'une liste.

MD =  $\{md\}$  est le complément de MR dans M.

Il est constitué de l'ensemble des mots de nature donnée, variable, ou résultat évalué à partir d'une liste.

#### Justification

Pour assurer l'affectation de tout md à un c.objet, il faudra disposer de la RF  $\text{ident} \rightarrow md$ . Cette RF est demandée au concepteur.

Au contraire, pour associer tout mr à un c.objet, la RF  $\text{ident} \rightarrow mr$  est déduite automatiquement à partir de la connaissance des RF.

- .  $\text{ident} \rightarrow md$  , md étant un mot de MD utilisé dans la définition de mr
- .  $\text{ident} \rightarrow \text{ident}$

Ces déductions sont développées dans le paragraphe "Apport des traitements en matière de sémantique des données".

La connaissance des RF  $\text{ident} \rightarrow md$  étant nécessaire pour effectuer les déductions de RF  $\text{ident} \rightarrow mr$ , il est logique de structurer d'abord l'ensemble MD, puis l'ensemble MR.

Cet ordre peut éviter au concepteur de commettre des erreurs, et d'économiser ainsi le temps nécessaire à la structuration.

Règle 2 : Décomposer MD en MDc et MD $\bar{c}$  et structurer d'abord MD $\bar{c}$ .

Nous décomposons MD en deux sous-ensembles disjoints MDc et MD $\bar{c}$ .

$$MD = MDc \cup MD\bar{c}$$

MDc = ensemble des mots de nature donnée, variable, intervenant dans les définitions de calcul.

MD $\bar{c}$  = ensemble des mots donnée n'intervenant pas dans les définitions de calcul.

Ces mots apparaissent donc uniquement sur les messages.

#### Justification

. Tout mot appartenant à MD $\bar{c}$  peut être l'identifiant d'un mot appartenant à MDc.

Par exemple, npro, mot appartenant à MD $\bar{c}$ , est l'identifiant de pu, qui apparaît dans la définition du résultat de la facture. Il est logique de reconnaître d'abord les identifiants avant de les utiliser et, par voie de conséquence, de structurer MD $\bar{c}$  avant MDc. Cette logique est susceptible de réduire les cas d'erreurs probables du concepteur.

. Cette règle va par ailleurs nous conduire à structurer l'ensemble des mots d'un même message à des instants différents, suivant qu'ils appartiennent soit à MDc, soit à MD $\bar{c}$ . Cette manière de procéder est économique, car elle minimise le nombre de RF demandées.

En effet, pour structurer les  $m$  mots d'un message, il faudrait normalement indiquer  $m^2$  relations fonctionnelles au départ. En réduisant l'ensemble des mots sur lequel sont définies des RF, l'ensemble minimal obtenu, constitué de l'ensemble des mots non rencontrés dans les traitements, étant  $MD\bar{c}$ , nous limitons le nombre de RF nécessaires à la structuration.

Le nombre de mots appartenant à  $MD\bar{c}$  et apparaissant sur un message est très faible ; aussi, nous structurons l'ensemble des mots des messages associées à une même procédure.

Règle 3 : Décomposer MR en MR<sub>i</sub> et structurer les MR<sub>i</sub> dans l'ordre chronologique.

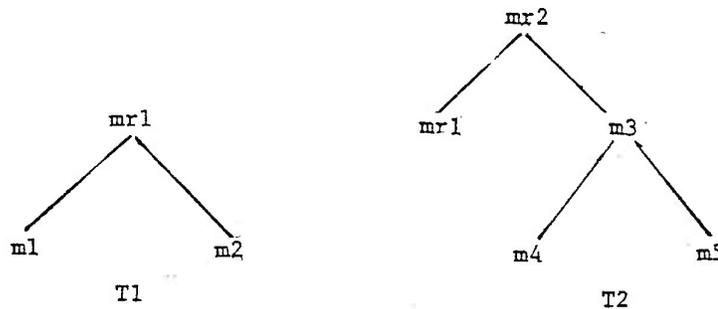
MR<sub>i</sub> = ensemble des mots de nature résultat, non évalués à partir d'une liste d'un même énoncé (formulé sur un descripteur de traitement Ti).

L'ordre chronologique de prise en compte des MR<sub>i</sub> correspond à l'ordre logique d'exécution des traitements décrits dans le SP. Cet ordre a été défini à partir d'un ordonnancement automatique des procédures et des modules d'une procédure pour les besoins de la construction de la structure fonctionnelle ( 7 ).

Il est décrit dans la base des objets élémentaires, et le processus de structuration ne fera que l'exploiter.

#### Justification

Une prise en compte désordonnée des énoncés d'opérations pourrait conduire à demander du concepteur des RF que le système peut déduire par une prise en compte ordonnée des énoncés.

Exemple

Soient :

- . T1 et T2 deux descripteurs de traitement, l'ordre chronologique d'exécution étant T1 puis T2.
- . mr1, mot de nature résultat valorisé dans le traitement T1 et utilisé dans le traitement T2.

Etudier d'abord T2, puis T1, nécessite de demander au concepteur la RF identifiant  $\rightarrow$  mr1, cette RF étant nécessaire pour déduire dans T2 la RF identifiant  $\rightarrow$  mr2.

Au contraire, l'examen chronologique des descripteurs permet, au cours de l'étude du descripteur T1, la déduction de la RF ident  $\rightarrow$  mr1, puis l'utilisation de cette RF dans l'étude du descripteur T2.

#### 3.4.2.3. - Conclusion sur l'ordre de structuration des éléments de M

Les trois règles de décomposition de M nous conduisent à procéder en deux étapes :

- 1 - Structurer  $M\bar{D}c$
- 2 - Structurer le complément de  $M\bar{D}c$  soit  $MDC \cup MR$ .

A - Structuration de  $MDC$ 

On conservera en fait l'unité sémantique des procédures, on analysera successivement l'ensemble des messages d'une même procédure qui alimentent  $MDC$  du SP à considérer. La structuration de  $MDC$  est donc itérative :

Pour toute procédure

- 1 - Définir  $MDC_i$  (ensemble des mots de nature donnée n'intervenant pas dans une définition de calcul)
- 2 - Structurer  $MDC_i$

B - Structuration de  $MDC$  U  $MR$ 

Les règles 1 et 3 déterminent complètement l'ordre de structuration de cet ensemble de mots.

La structure est itérative

pour tout énoncé d'un descripteur de traitement  $T_i$

- 1 - Définir  $MDC_i$  (ensemble des mots de nature donnée, variable ou résultat évalué à partir d'une liste intervenant dans les définitions de  $T_i$  et non structurés auparavant).

et  $MR_i$  (ensemble des mots de nature résultat non évalué à partir d'une liste appartenant à  $T_i$ )

- 2 - Structurer  $MDC_i$

- 3 - Structurer  $MR_i$

### 3.4.3. - Les règles de structuration

#### 3.4.3.1. - Incidence du principe de structuration sur la recherche d'une solution

A un instant donné, le problème posé est celui de la structuration d'un ensemble  $m_i$  de mots ( $m_i \rightarrow M$ ) à partir des relations fonctionnelles sur  $m_i$ .

La connaissance de modèle final et de la structure de  $c$ .classes qui correspond à la solution que l'on doit trouver à ce problème nous conduit à le poser de la manière suivante :

"Structurer  $m_i$  c'est :

- 1 - Trouver les  $c$ .objets c'est-à-dire structurer  $m_i$  en  $c$ .objets.
- 2 - Trouver les  $c$ .classes c'est-à-dire agréger les  $c$ .objets en  $c$ .classes.
- 3 - Trouver le système de  $c$ .classes c'est-à-dire associer les  $c$ .classes dans un système de  $c$ .classes".

C'est effectivement sous cette forme et dans cet ordre que les problèmes seront abordés et résolus.

En outre, à chacun de ces trois problèmes peut être associé un ensemble de règles de structurations qui recouvre la contrainte d'intégrité imposée dans le modèle respectivement au  $c$ .objet, à la  $c$ .classe, au système de  $c$ .classes, et les contraintes de cohérence qui en découlent.

#### 3.4.3.2. - Règles générales de structuration

Nous distinguons deux types de règles : les règles de construction, permettant de grouper les mots ; les règles de contrôle, vérifiant

que la structure obtenue respecte les contraintes imposées par le modèle.

### 3.4.3.2.1. - Règles de structuration de mi en c.objets

Nous distinguons deux cas :

A - La structuration des c.objets de nature donnée, variable.

B - La structuration des c.objets de nature résultat.

#### A - Structuration des c.objets de nature donnée, variable :

##### a) Construction de groupes primaires maximaux

- Soient :  $\cdot mi = \{ \text{mot } j \} \quad j = 1, n$  l'ensemble des mots à structurer

$\cdot \text{ident} = \{ \text{ident } k \} \quad k = 1, \ell$  ;  $\text{ident } k \in mi$  ou  $\notin mi$   
l'ensemble des identifiants.

$\cdot R1$  l'ensemble des RF élémentaires directes  $\text{ident } k \rightarrow$   
 $\text{mot } j$  définies pour tout mot  $j$  de type constituant.

##### - Règle de construction des groupes maximaux

Construire le groupe maximal identifié par  $\text{ident } k$ , c'est rendre constituant de ce groupe tout mot  $j$  tel que la RF  $\text{ident } k \rightarrow \text{mot } j$  appartienne à  $R1$ .

##### - Justification :

Par définition, le résultat de l'action définie dans cette règle est un groupe maximal.

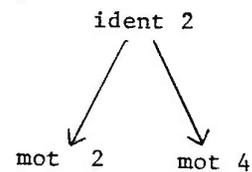
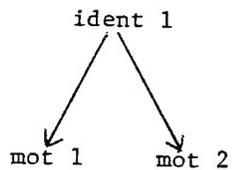
##### - Exemple :

$\cdot mi = \{ \text{mot } 1, \text{mot } 2, \text{mot } 3, \text{mot } 4 \}$

$\cdot \text{Ident} = \{ \text{ident } 1, \text{ident } 2 \}$

$\cdot R1 = \text{ident } 1 \rightarrow \text{mot } 1$  ;  $\text{ident } 2 \rightarrow \text{mot } 4$  ;  
 $\text{ident } 2 \rightarrow \text{mot } 3$  ;  $\text{ident } 1 \rightarrow \text{mot } 2$ .

$\cdot$  Groupes maximaux obtenus après l'application de la règle :



b) Contrôle vérifiant que le groupement obtenu est un c.objet

- Règle de contrôle de l'homogénéité temporelle des RF

Les RF identifiant  $\rightarrow$  constituant d'un c.objet sont toutes de même type par rapport au temps :

- . soit du type permanent
- . soit du type semi-permanent.

- Justification :

Cette règle est la contrainte d'intégrité imposée par la définition du c.objet.

- Règle de contrôle de la nature du c.objet

Tous les constituants du c.objet sont de même type par rapport à l'usage :

- . soit du type donnée
- . soit du type variable

- Justification :

Nous avons vu dans le modèle que cette contrainte est vérifiée par tous les c.objets.

- Règle de contrôle de l'unicité d'un constituant

Un mot de type constituant appartient à un seul c.objet.

- Justification :

Nous avons vu qu'un mot de type constituant appartient à un seul c.objet d'une c.classe. Le non respect de cette contrainte entraîne une polysémie.

De même, de par la définition du système de c.classes, un mot de type

constituant appartient à une seule c.classe et le non respect de cette contrainte entraîne une polysémie.

B - Structuration des c.objets de nature résultat

a) Structuration des c.objets

- Situation

- . Pour tout mot<sub>i</sub> de nature résultat, les RF identifiant de c.classe → mot<sub>i</sub> sont connues.
- . Les groupes associés à une condition définis dans les travaux de M. Chesseron sont connus.

- Règle de contrôle du groupe associé à une condition

Tous les mots d'un groupe associé à une condition doivent appartenir à une même c.classe.

- Justification :

Le non respect de cette règle entraîne la redondance dans l'exécution d'opérations.

- Règle de structuration des c.objets de nature résultat

Structurer les c.objets de nature résultat d'une c.classe c'est découper l'ensemble des mots de nature résultat de cette c.classe en groupes, chacun de ces groupes étant associé à une condition.

. Justification :

La permanence des RF entre identifiant et constituant d'un c.objet impose que tous les constituants d'un c.objet soient créés en même temps. Seuls les mots appartenant à un groupe associé à une même condition peuvent l'être.

b) Contrôles des c.objets

Nous appliquons également dans ce cas la règle de contrôle de l'unicité d'un constituant. En effet, le non respect de cette règle entraîne

à ce niveau soit une redondance dans la description d'opérations, soit une polysémie.

#### 3.4.3.2.2. - Règles de structuration des c.objets en c.classes

Soit  $O = \{O_i\}$   $i = 1, n$  l'ensemble des c.objets à structurer

Soit  $R_2$  l'ensemble des RF élémentaires, stables et réciproques entre identifiants de c.objets.

##### - Règle d'obtention des c.objets constituants d'un groupe maximal de c.objets

Obtenir l'ensemble des c.objets constituants d'un groupe maximal de c.objets, c'est grouper entre eux tous les c.objets reliés par une RF appartenant à  $R_2$ .

##### . Justification :

Par définition du groupe maximal, les ensembles obtenus sont les groupements de c.objets constituants d'un groupe maximal.

##### . Exemple :

$$O = \{O_i\} \quad i = 1, 5$$

$$R_2 = O_1 \longleftrightarrow O_2$$

$$O_1 \longleftrightarrow O_5$$

$$O_5 \longleftrightarrow O_2$$

$$O_2 \longleftrightarrow O_4$$

rep représentation  $\longleftrightarrow$  RF élémentaires,  
stables et réciproques.

Les groupes maximaux obtenus par l'application de la règle ont les composants suivants :

$$G_1 = \{O_1 ; O_2 ; O_5\}$$

$$G_2 = \{O_3 ; O_4\}$$

- Règle de construction d'une c.classe

Construire une c.classe formée sur un groupe maximal de c.objets, c'est :

- créer un mot identifiant de c.classe
- créer les RF entre identifiant de c.classe et identifiant de c.objet
- supprimer les RF directes entre identifiants de c.objets.

. Justification :

- Par construction :

- a) les RF entre identifiant de c.objet et identifiant de c.classe sont élémentaires, stables et réciproques ;
- b) les contraintes portant sur la durée de vie de l'identifiant de c.classe sont respectées.

- La structure obtenue est une c.classe.

- Nous remarquons qu'un identifiant de c.objet ne peut pas être identifiant de c.classe, les contraintes portant sur la durée de vie l'interdisant.

. Exemple :

Soient  $G1 = \{01 ; 02 ; 05\}$   
 $G2 = \{03 ; 04\}$

Les c.classes obtenues après application de la règle sont :



. Remarque : amélioration sémantique

Il se peut que la désignation de l'identifiant de c.classe créée par le système ne soit pas explicite ; en ce cas, le concepteur la transforme.

3.4.3.2.3. - Règles de structuration du système de c.classes

- a) Construction de l'ensemble des RF directes, élémentaires

- entre c.classe d'ordre 1
- Soit  $C = \{C_i\}$   $i = 1, n$  l'ensemble des c.classes d'ordre 1 à structurer en système de c.classes.
  - Soit  $R_3$  l'ensemble des RF entre c.classes.

- Règle de suppression des transitivités

Obtenir un sous ensemble  $R'_3$  de  $R_3$ ,  $R'_3$  étant l'ensemble des RF élémentaires et directes, c'est supprimer de  $R_3$  l'ensemble des RF non directes.

. Justification :

- Par construction,  $R_3$  est un ensemble de RF directes.
- D'après la règle d'atomisation des identifiants, tous les identifiants de c.classe introduits à ce niveau sont des identifiants simples ; les RF sont donc élémentaires.

b) Composition des identifiants de c.classes

Tous les identifiants de c.classes perçus initialement sont simples. La règle de composition indique quelles sont les contraintes devant être respectées pour qu'un identifiant de c.classe soit composé de plusieurs mots.

- Règle de composition

- Soient  $c.iden_i, c.iden_j, c.iden_k$  des identifiants simples de c.classe. S'il existe les RF permanentes, réciproques, élémentaires

$$c.iden_k \longrightarrow c.iden_i, c.iden_j$$

$$c.iden_k \longleftarrow c.iden_i, c.iden_j$$

Alors la c.classe possède deux identifiants  $c.iden_k$  et  $c.iden_i, c.iden_j$  et elle est d'ordre 2.

Cette règle peut être généralisée aux c.classes d'ordre n.

. Justification :

Il découle de l'existence des RF permanentes, réciproques, élémentaires

entre  $c.iden_k$  et  $c.iden_i$ ,  $c.iden_j$  que le c.objet racine de la c.classe a pour identifiant  $c.iden_k$  et  $c.iden_i$ ,  $c.iden_j$  ; (§ 3.5.4.1.1.).

Donc, ces deux identifiants sont identifiants de c.classe.

- Règle de situation

Soient les identifiants simples de c.classe  $c.iden_i$  ;  $c.iden_j$  ;  $c.iden_k$  ; les conditions généralement respectées pour que  $c.iden_k$  et  $c.iden_i$ ,  $c.iden_j$  soient identifiants d'une même c.classe sont les suivantes :

- a) Il existe les RF  $c.iden_k \longrightarrow c.iden_i$   
 $c.iden_k \longrightarrow c.iden_j$

- b) Il n'existe pas de RF entre  $c.iden_i$  et  $c.iden_j$ .

. Démonstration :

- a) démonstration par l'absurde :

Nous montrons que si la RF  $c.iden_k \longrightarrow c.iden_i$  n'existe pas, alors que en général la RF  $c.iden_k \longrightarrow c.iden_i$ ,  $c.iden_j$  n'existe pas.

La non existence de la RF  $c.iden_k \longrightarrow c.iden_i$  implique qu'à une réalisation  $rk$  de  $iden_k$  correspondent plusieurs réalisations  $ri$  de  $c.iden_i$ . Par exemple à  $rkl$  correspond l'ensemble des réalisations  $ril, ri2, ri3$ . S'il existe des réalisations  $rj$  de  $c.iden_j$  que nous notons  $rjx$  et  $rjy$  telles que les réalisations de l'association  $ril, rjx$  et  $ri2, rjy$  existent (condition C1) alors à la réalisation  $rkl$  correspondent les deux réalisations d'association  $ril, rjx$  et  $ri2, rjy$ . La RF  $c.iden_k \longrightarrow c.iden_{i,j}$  n'existe donc pas si la condition C1 est satisfaite. Comme cette condition l'est, dans la pratique, presque toujours, l'ensemble des identifiants susceptibles d'intervenir dans une composition peut être considéré comme inclus dans l'ensemble des identifiants vérifiant la condition a) de la règle.

- b) la démonstration est faite par Flory dans ( 14 ).

c) Contrôles de la structure obtenue

Les types de mots du système de c.classes doivent vérifier certaines règles.

. Mot de type constituant

La règle d'unicité doit être respectée. De plus, un mot de type constituant ne doit pas intervenir dans un identifiant.

. Justification sémantique :

Un mot de type constituant représente une propriété d'un objet du monde réel. Cette propriété n'appartient qu'à un objet, et ne peut permettre l'identification d'autres objets.

. Mot de type paramètre

Paramètre simple

Un paramètre simple n'a qu'une réalisation à un instant donné.

. Justification :

Le paramètre simple n'est ni but, ni source de RF. Aucune relation ne permet de faire correspondre une réalisation d'un paramètre simple à une autre réalisation d'un composant du SI. Nous supposons donc que cette correspondance est implicite. Elle ne peut être faite que si le paramètre a une seule réalisation à un instant donné.

Paramètre d'association

Un paramètre d'association compose l'identifiant d'au moins deux c.objets d'ordre n.

. Justification sémantique

Cette règle permet de distinguer sémantiquement le constituant qui représente une propriété d'un unique objet du monde réel, du paramètre.

Les règles portant sur les mots de type constituant et paramètre simple sont également définies dans les travaux de Flory (14).

- Les règles méthodiques

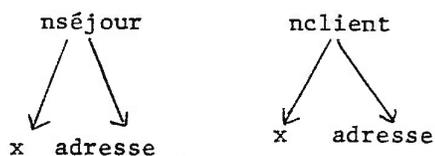
- Règles méthodique 1

Soit un mot apparaissant constituant de plusieurs c.objets de nature donnée ou variable. Le concepteur soit modifie les désignations soit rend ce mot paramètre d'association.

. Justification :

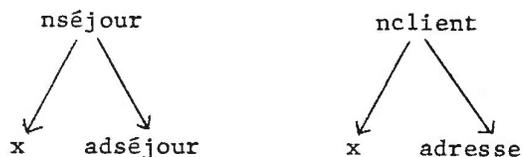
Un mot qui apparaît constituant de plusieurs c.objets est soit constituant, et dans ce cas, il y a polysémie (exemple 1) ; soit paramètre d'association (exemple 2).

Exemple 1



Le constituant, ayant pour désignation adresse représente, lorsqu'il appartient au c.objet identifié par nséjour l'adresse de l'hôtel où s'effectue le séjour, et lorsqu'il appartient au c.objet identifié par c.client l'adresse du client.

Dans ce cas, le concepteur doit modifier les désignations, de la façon suivante par exemple :



Exemple 2



Le mot ayant pour désignation mois représente un même élément du monde réel. Il est paramètre d'association. Le concepteur exprime ceci par les RF élémentaires, réciproques, permanentes

nclimens  $\longleftrightarrow$  ncli ; nmois

npromens  $\longleftrightarrow$  npro ; nmois

- Règle méthodique 2

Soit un mot ni but ni source de RF.

Si ce mot a plusieurs réalosations à un instant donné, le concepteur doit créer un identifiant tel qu'il existe la RF identifiant  $\rightarrow$  mot.

. Justification :

D'après la règle portant sur les paramètres simples, ce mot ne peut être paramètre simple. Nous considérons que ce mot est un constituant dont l'identifiant n'apparaissait pas.

3.4.4. - Les règles d'intégration

Elles interviennent lorsque le processus de structuration a conduit à la situation suivante :

- Un ensemble E de mots est structuré en un système de c. classes  $S_0$ .
- Un sous-ensemble  $M_i$  de mots de M vient d'être structuré en un système de c. classe  $S_i$ .

$S_0$  et  $S_i$  peuvent être intégrés pour définir un système de c. classes  $S_j$  défini sur l'ensemble des mots  $E \cup M_i$ .

L'intégration, comme la structuration s'effectue aux trois niveaux :

- . des c.objets
- . des c.classes
- . du système de c.classes.

Nous distinguons également deux types de règle : les règles de construction et les règles de contrôle.

#### 3.4.4.1. - Règles d'intégration des c.objets

##### a) Reconnaissance des c.objets

Les c.objets de SJ sont, soit ceux de So, soit ceux de Si, soit ceux obtenus par l'intégration d'un c.objet de So et d'un c.Object de Si.

Nous nous intéressons ici aux c.objets obtenus par intégration. La règle de reconnaissance permet de définir les c.objets de Si et de So qui sont intégrés.

##### - Règle de reconnaissance des c.objets à intégrer ;

Soient  $O_o$  et  $O_i$  deux c.objets, appartenant respectivement à So et Si. S'il existe entre les c.Object des RF réciproques, élémentaires et permanentes alors les c.objets sont intégrés.

##### . Remarque :

En général, les identifiants de  $O_o$  et de  $O_i$  ont la même désignation.

##### b) Intégration de $O_o$ et de $O_i$

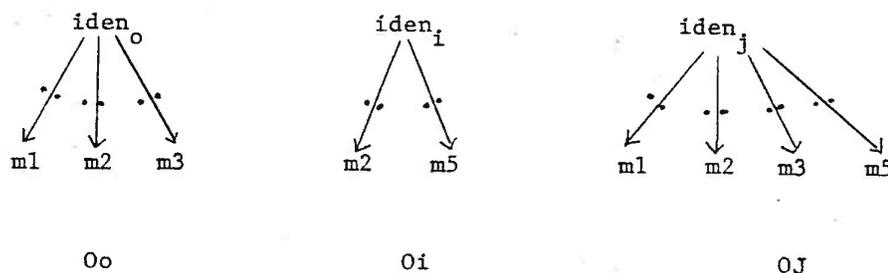
##### - Règle d'intégration

Intégrer  $O_o$  et  $O_i$  c'est créer OJ appartenant à SJ tel que :

- . tout mot constituant de  $O_o$  et de  $O_i$  soit constituant de OJ.
- . l'identifiant de OJ soit  $iden_o$  ou  $iden_i$ ;  $iden_o$  et  $iden_i$  étant respectivement les identifiants des c.objets  $O_o$  et  $O_i$ .

Exemple

Les c.objets  $O_o$ ,  $O_i$  et le c.objet  $O_j$  obtenu par intégration de  $O_o$  et  $O_i$ , sont les suivants :

c) Contrôle du c.objet constitué par intégration

Les règles de contrôles sont les mêmes que celles définies dans la structuration. Ce sont les règles de contrôle de l'homogénéité temporelle des RF et de contrôle de la nature du c.objet.

3.4.4.2. - Règle d'intégration des c.classesa) Reconnaissance des c.classes

De même que pour les c.objets, nous nous intéresserons aux c.classes obtenues par intégration ; la règle de reconnaissance permet de définir les c.classes de  $S_i$  et de  $S_o$  qui sont intégrées.

- Règle de reconnaissance

Soient  $C_i$  et  $C_o$  deux c.classes appartenant respectivement à  $S_i$  et  $S_o$ . S'il existe entre les c.classes des RF élémentaires, stables et réciproques alors les c.classes sont intégrées.

. Remarque :

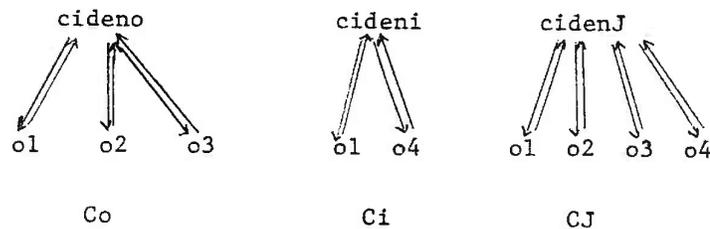
En général, les désignations des identifiants de c.classes qui sont intégrées sont les mêmes.

b) Intégration de Co et Ci- Règle d'intégration

Construire la c.classé CJ obtenue par intégration de Ci et Co  
c'est rendre tout c.objet constituant de Ci ou de Co constituants  
de CJ.

Exemple :

Les c.classés Co et Ci et la c.classé CJ obtenue en intégrant Co et Ci  
sont les suivantes :

. Remarques

- L'identifiant ou la c.classé CJ sera ciden0 ou cideni (selon le choix du concepteur). Implicitement, la durée de vie de ses occurrences respecte les contraintes imposées par le modèle.
- Il découle de la règle d'intégration, que, pour des raisons d'optimisation, nous intégrons d'abord les c.classés, puis les c.objets

3.4.4.3. - Règles d'intégration du système de c.classésa) Construction de l'ensemble des RF directes, élémentaires, entre c.classés

Soit R3o l'ensemble des RF entre c.classés de So

R3i l'ensemble des RF entre c.classés de Si

R3N l'ensemble des RF directes élémentaires entre les  
c.classés de Si et de So.

- La règle de suppression des transitivités définie précédemment permet de construire R3, ensemble des RF directes élémentaires de SJ.

b) Construction de c.classes d'ordre n

Les règles de situation et de composition des identifiants permettent, à partir de la connaissance de R3 et des identifiants de c.classe, de définir s'il y a lieu, de nouvelles c.classes d'ordre n.

c) Contrôle de la structure obtenue

Ce sont les règles définies sur les types de mots dans la structuration.

3.5. - Processus de structuration

3.5.1. - Processus appliqué aux mots définis dans les messages d'une procédure

Ce processus apparaît comme un ensemble alternant structuration et intégration. Il comporte les étapes suivantes :

- Etape 1 : structuration des c.objets
- Etape 2 : Structuration des c.classes ; intégration des c.classes, des c.objets
- Etape 3 : Structuration du système de c.classes ; intégration.

Etape 1 - Structuration des c.objets

- Entrées de l'étape :

- . l'ensemble des  $MD\bar{c}_i$
- . l'ensemble des RF définies sur  $MD\bar{c}_i$

$MD\bar{c}_i$  : ensemble des mots appartenant aux messages d'une procédure de nature donnée, et n'intervenant pas dans les traitements.

Les RF sont définies par le concepteur à l'aide d'une matrice carrée.

mots mots	mot 1	mot 2			mot n
mot 1					
mot 2	SP			S	
			Q		
	SP				
mot n					

L'existence d'une RF et son type par rapport au temps entre  $mot_i$  et  $mot_j$  est signalée par la présence, à l'intersection de la j<sup>ème</sup> ligne et de la i<sup>ème</sup> colonne des caractères :

P si la RF est permanente

SP si la RF est semi-permanente

Q si elle n'est ni permanente, ni semi-permanente.

- Action 1.1. construction de groupes maximaux

Les algorithmes A1 et A2, définis par Flory dans ( 14 ) construisent des GP.

- Action 1.2. contrôle des GP ; construction de c.objets

- . Vérification automatisée (algorithme A3 - règles d'unicité

du constituant) communication de la liste L1 des mots appartenant à plusieurs c.objets, de la liste L2 des mots non identifiants et non rattachés à des identifiants, de la liste I des identifiants.

. Correction par le concepteur

Pour tout mot appartenant à L1, application de la règle méthodique 1

Pour tout mot appartenant à L2, application de la règle méthodique 2

- . Construction des c.objets par éclatement (algorithmes A4.1 ) et contrôle (algorithme A 4.2. -règle d'homogénéité temporelle des RF) - Communication de la liste des groupes ne satisfaisant pas cette condition.

. Correction par le concepteur

Ecriture des RF entre les identifiants nouvellement définis et ceux appartenant à I.

Le processus (construction - contrôle) est itératif, l'itération cessant lorsqu'il n'y a plus d'erreurs.

Etape 2 - Structuration des c.classes

- Entrées de l'étape :

- . L'ensemble I des identifiants de c.objets
- . L'ensemble des RF élémentaires, stables, réciproques définies sur I.

Documentation produite par le système (algorithme A5) : les RF réciproques entre identifiants de c.objets, présentées sous la forme d'une matrice diagonale supérieure.

Les RF réciproques, stables, élémentaires entre I1 et I2 sont indiquées par un 1.

	I1	I2		In
I1		1		
I2				1
				1
In				

Le concepteur indique la stabilité ou non des RF en inscrivant un S entre deux identifiants reliés par des RF stables.

- Action 2.1. - Structuration des c.classes

- . Regroupement des c.classes (algorithmes A6 - règles de construction de groupes de c.objets et de c.classes).
- . Le concepteur donne, si nécessaire, un identifiant plus explicite à la c.classe.

- Action 2.2. - Intégration des c.classes, des c.objets

- . Entrée de l'étape :
  - . L'ensemble des identifiants de c.classes de Si et de So
  - . L'ensemble des RF élémentaires, stables, réciproques définies entre les identifiants de c.classes de Si et de So.

Documentation produite par le système (algorithme A7) : ensemble des identifiants de c.classes de Si et de So.

Les RF sont définies par le concepteur à l'aide d'une matrice diagonale supérieure.

### 2.2.1. - Intégration des c.classes de Si et de So

Reconnaissance des c.classes (algorithmes A8 - règle de reconnaissance) à intégrer.

#### 2.2.1.1. - Pour chaque intégration de c.classes

- . Entrée :
  - . ensemble des identifiants de c.objets de Ci et de Co.
  - . ensemble des RF élémentaires, permanentes, réciproques définies sur ces identifiants.
- . Documentation produite par le système (algorithme A9) :
  - . ensemble des identifiants de c.objets
- . Le concepteur définit les RF à l'aide d'une matrice diagonale supérieure.

#### - Intégration de la c.classe

L'algorithme A 10 effectué

- . pour chaque é.objet de la c.classe la reconnaissance (règle de reconnaissance des c.objets) l'intégration des c.objets (règle d'intégration des c.objets) l'intégration des c.classes.

### Etape 3 - Structuration du système de c.classes

- Entrée de l'étape : ensemble des RF élémentaires, directes entre c.classes d'ordre 1.

. L'algorithme A1 de l'étape 1 supprime les transitivités et construit les RF élémentaires directes entre identifiants de c.objets.

. L'algorithme A 11 construit, à partir des RF, l'ensemble des RF élémentaires, directes entre identifiants de c.classes d'ordre 1.

#### Action 3.1. - Découverte des associations par le concepteur

- Documentation, produite par le système (algorithme A 12 - règle de situation) des identifiants de c.classes susceptibles d'être composés, et intervenant dans la composition.

- Ecriture, par le concepteur, des RF permanentes, élémentaires, réciproques entre identifiants d'ordre 1 et d'ordre n.

- Suppression automatisée de l'ensemble des RF non élémentaires (algorithmes A 13).

Action 3.2. - Intégration des systèmes de c.classes Si et So

- Entrée : ensemble des RF élémentaires et directes entre c.classes de Si et So.

- Documentation produite par le système (algorithme A 7) : ensemble des identifiants de c.classes de Si et de So.

Les RF élémentaires, directes, sont définies par le concepteur.

Action 3.2.1. - Suppression automatisée des transitivités (A 1 partiel)

Action 3.2.2. - Découverte des identifiants de c.classes.

3.5.2. - Processus appliqué aux mots définis dans les traitements d'une procédure

Le processus, alternant structuration et intégration, comporte les étapes suivantes :

Etape 1 : structuration des c.objets de nature donnée, variable

Etape 2 : structuration des c.classes ; intégration des c.classes, des c.objets

Etape 3 : structuration du système de c.classes ; intégration

Etape 4 : structuration des c.objets de nature résultat, intégration.

Etape 1 - Structuration des c.objets de nature donnée, variable

- Entrée de l'étape :

- . l'ensemble des MDci
- . l'ensemble des RF.

MDci : ensemble des mots appartenant aux traitements de nature donnée, variable, ou résultat et non évalués à partir d'une liste.

Pour tout mot appartenant à MDci, le concepteur définit arbitrairement l'identifiant et les RF identifiant → mot, de type permanent ou semi-permanent, exprimées à l'aide d'une matrice.

	I1	I2		Is
mot 1		P		
mot 2			SP	
	P			
mot n				P

$$I = \{ I_j \} \quad j = 1, s \text{ ensemble des identifiants}$$

$$MDci = \{ \text{mot}_i \} \quad i = 1, n$$

L'existence d'une RF et son type par rapport au temps entre Ik et mot l est signalé par la présence, à l'intersection de la kième colonne et de la lième ligne du caractère :

P lorsque la RF est permanente

SP lorsque la RF est semi-permanente.

Action 1.1: Construction des c.objets (algorithmes A 14 - règle de construction des GP maximaux).

Action 1.2: Contrôle des c.objets (algorithmes A 4.2. - règle d'homogénéité temporelle des RF ; algorithme A 3 - règle d'unicité du constituant  
Correction par le concepteur.

### Etape 2 - Structuration de c.classes

- Entrée : l'ensemble des RF élémentaires, stables, réciproques entre identifiant de c.classes.

- Documentation produite par le système (algorithme A 7) : ensemble des identifiants de c.classes.
- Les RF sont définies par le concepteur, à l'aide d'une matrice diagonale supérieure.

Action 2.1. Structuration des c.classes

Action 2.2. Intégration des c.classes, des c.objets.

Ces actions sont identiques à celles définies dans le processus appliqué aux mots appartenant aux messages.

### Etape 3 - Structuration du système de c.classes

- Entrée : Liste des identifiants de c.classes. Cette liste est fournie par le système :

. ensemble des RF entre identifiants de c.classes, défini par le concepteur à l'aide d'une matrice.

Action 3.1. Construction d'un ensemble de RF directes, élémentaires entre c.classes d'ordre 1 (partie de l'algorithme A 1 - règle de suppression des transitivités).

Action 3.2. Découverte des associations par le concepteur.

Action 3.3. Intégration des systèmes de c.classes Si et So . Ces actions sont identiques à celles définies dans le processus appliqué aux mots appartenant aux messages (actions 3.1. et 3.2).

### Etape 4 - Structuration des c.objets de nature résultat

Action 4.1. Création automatisé des c.objets de nature résultat.

Rattachement automatisé des mots de nature résultat à des c. classes (algorithmes A 15 - règles de déduction)  
 Contrôle des groupes associés à une condition (algorithme A 16 - règle de contrôle).

- Correction des erreurs par le concepteur.
- Structuration des c.objets, de nature résultat (algorithme A 16 - règle de structuration des c.objets).
- Contrôle des c.objets (algorithmes A 3 - règle de l'unicité du constituant)
- Correction des erreurs par le concepteur.
- Intégration des c.objets de nature résultat.

### 3.6. - Apport des traitements en matière de sémantique des données

- Les définitions du langage de traitement expriment des relations n.aires entre les mots intervenant dans les définitions et leurs identifiants.

- La nature des définitions, consécutive ou itérative, exprime implicitement des contraintes d'intégrité entre ces mots, qui peuvent être interprétées et décrites dans le SC sous forme de RF.

- Dans ce paragraphe, nous énonçons certaines de ces contraintes, à partir desquelles nous pouvons faire, dans les cas favorables, des déductions concernant l'existence des RF et, dans tous les cas, des contrôles de validité du SC.

#### 3.6.1. Définitions consécutives

Les règles expriment l'interprétation des définitions consécutives en termes de RF, c'est-à-dire l'ensemble des RF entre identifiants qui doivent être respectées pour que la définition ait un sens.

L'utilisation de ces règles permet la déduction des RF identifiant de c.classe → mot. Les déductions se font au cours d'une analyse ascendante de l'arbre des résultats.

La prise en compte des contraintes dynamiques (les conditions définies par M. Chesseron) permet ultérieurement la déduction des RF identifiant c.objet → mot.

Deux cas sont exclus de ces déductions.

- . L'affectation, dans laquelle les mots ne sont pas calculés.
- . La définition consécutive, évaluant un mot de nature variable. Dans ce cas, seuls des contrôles peuvent être faits.

### 3.6.1.1. Contraintes d'intégrité des mots de nature résultat

#### 3.6.1.1.1. - Contrainte 1 :

- Soit  $m_1$  un mot de nature résultat, défini en fonction des mots  $m_2, m_3, \dots, m_n$ ; et pour tout  $i, i = 1, n$ ,  $\text{iden}(m_i)$ , l'identifiant du c. objet de  $m_i$  est un constituant.

- Pour tout  $i, i = 1, n$ , les RF  $\text{iden}(m_1) \longrightarrow \text{iden}(m_i)$  existent.

On trouve dans (14) une règle analogue.

#### Justification

- Soit  $m_1 = f(m_2, \dots, m_n)$  l'énoncé d'une opération, et, pour tout  $i, i = 1, n$ ,  $O_i$  le c. objet dans lequel  $m_i$  est constituant.

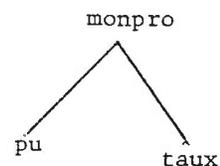
- Supposons connue l'occurrence de l'objet  $O_1$ , au moment de l'exécution d'une occurrence d'un c. opération ; alors, les RF  $O_1 \longrightarrow O_2, \dots, O_1 \longrightarrow O_n$  donnent la connaissance des occurrences des objets  $O_1, \dots, O_n$  qu'il faut utiliser pour calculer la valeur de  $m_1$ .

#### Exemple : "traitement du produit"

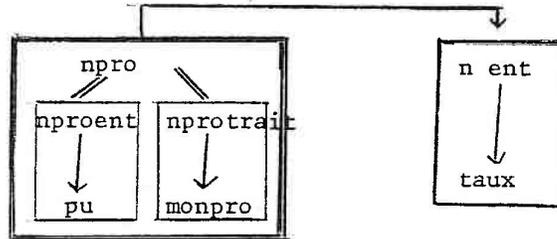
- Soient : . La définition du montant d'un produit

$$\text{monpro} = \text{pu} \times \text{taux}$$

représentée par l'arbre



- et le système des données



Pour effectuer le calcul d'une occurrence de monpro, constituant du c.objet identifié par nprotrait, il est nécessaire de connaître les occurrences des c.objets dont les identifiants sont nproent et nent. L'existence des RF nprotrait → nproent et nprotrait → nent donne cette connaissance.

Si les RF  $O_1 \rightarrow O_2, \dots, O_1 \rightarrow O_n$  sont absentes, l'énoncé seul de la c.opération est insuffisant pour permettre l'exécution d'une de ces occurrences. D'autres informations, permettant de connaître les occurrences des c.objets utilisés, doivent compléter cet énoncé.

#### 3.6.1.1.2. - Contrainte 2

Les c.objets dont les composants apparaissent dans une définition sont : le c.objet créé par la c.opération, les c.objets utilisés dans la c.opération, et parfois le c.objet dont le changement d'état déclenche la c.opération.

La contrainte 2 définit de façon plus précise les contraintes existant entre ce dernier c.objet et le c.objet créé par l'opération.

Contrainte 2 : Le c.objet créé par une c.opération et le c.objet dont le

changement d'état correspond au c.événement qui la déclenche, appartiennent à une même c.classe.

### Justification

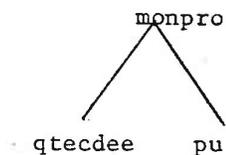
L'arrivée d'une occurrence du c.événement  $e_l$  donne, par définition, la connaissance de l'occurrence du c.objet  $O_e$  dont le changement d'état est constaté par  $e_l$ . La contrainte 2 donne la connaissance de l'objet  $O_c$  qui va être créé.



Si cette contrainte n'est pas respectée, l'énoncé de la c.opération est insuffisant pour permettre l'exécution. D'autres informations doivent le compléter.

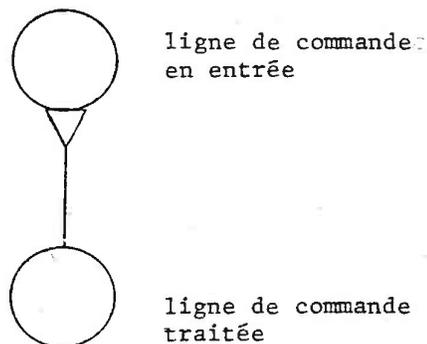
### Exemple : "Traitement de lignes de commandes"

- L'arrivée d'une ligne de commande déclenche l'opération de création d'une ligne de commande traitée.
- Le résultat monpro, constituant du c.objet "ligne de commande traitée" est défini par l'arbre



où qtcdee est un constituant du c.objet "ligne de commande en entrée".

- La structure fonctionnelle est représentée par :



- La contrainte 2 exprime que les c.objets "ligne de commande en entrée" et "ligne de commande traitée" appartiennent à la même c.classe; la c.classe "ligne de commande" identifiée par nlico.

### 3.6.1.1.3. - Contrainte 3

- . Soient  $m_1$ , mot calculé en fonction de  $m_2, m_3, \dots, m_n$  et pour tout  $i, i = 1, n$   $c.iden(m_i)$  l'identifiant de la c.classe dont  $m_i$  est un constituant.
- . Seuls, les mots composant  $c.iden(m_2), \dots, c.iden(m_n)$  peuvent entrer dans la composition de  $c.iden(m_1)$ .

#### Exemple :

Dans le "traitement des lignes de commande", la c.classe où se situe le composant qtecdé est identifiée par nlico, celle où se situe le composant pu est identifié par npro.

La contrainte 3 exprime que l'identifiant de la c.classe où se situe monpro est composé à partir de l'ensemble des identifiants

$$\{nlico ; npro\}$$

#### Justification

Le non respect de cette contrainte ne permet pas à la RF  $c.iden(m_1) \rightarrow m_1$  d'être élémentaire comme cela est imposé par la définition de la c.classe.

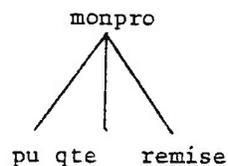
- La valeur de  $m_1$  dépend uniquement de la valeur de  $m_2, \dots, m_n$ . De même, l'identifiant de la c.classe  $c.iden(m_1)$  ne dépend que des identifiants  $c.iden(m_2), \dots, c.iden(m_n)$ . L'introduction d'un autre élément dans  $c.iden(m_1)$  rend la RF  $c.iden(m_1) \rightarrow m_1$  non élémentaire.

Ainsi, supposons, dans l'exemple précédent, qu'un composant

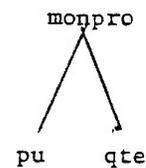
de la c.classe dans laquelle est située monpro soit n four (identifiant du fournisseur).

- Si aucun élément dépendant du fournisseur n'intervient dans la définition de monpro, alors, la RF nlico, nfour  $\rightarrow$  monpro n'est pas élémentaire.

- La RF n'est élémentaire que si un constituant dépendant du fournisseur intervient dans le calcul de monpro, par exemple sous l'une des deux formes suivantes :



Exemple 1



Exemple 2

- . Dans l'exemple 1 remise est un constituant de la c.classe identifiée par nfour.
- . Dans l'exemple 2 pu est un constituant de la c.classe identifiée par nfour, npro.

### 3.6.1.2. Règles de déduction des RF à partir des contraintes précédemment définies

- Soit  $m_1$  mot résultat calculé en fonction de  $m_2, m_3, \dots, m_n$ , pour tout  $i, i = 1, \dots, n$  et c.iden ( $m_i$ ) l'identifiant de la c.classe dont  $m_i$  est un constituant. Alors l'utilisation de la règle de déduction permet de déterminer l'identifiant c.iden ( $m_1$ ).

### 3.6.1.2.1. - Règle de déduction. Cas ou $n=2$

- . Soit  $m_1$ , mot calculé en fonction de  $m_2$ ,  $m_3$  et pour  $i = 1, 2, 3$   $ciden(m_i)$  l'identifiant de la c.classe dont  $m_i$  est un constituant.

1er cas : s'il n'existe pas de RF entre  $ciden(m_2)$  et  $ciden(m_3)$  alors  $ciden(m_1) = ciden(m_2), ciden(m_3)$ .

2ème cas : s'il existe une RF entre  $ciden(m_2)$  et  $ciden(m_3)$  alors  $ciden(m_1) = ciden(m_i)$ ;  $ciden(m_i)$  étant l'identifiant source de cette RF.

#### Exemple :

1er cas :    monpro  
              /  \  
          qteliv  pu

- . pu est constituant de la c.classe identifiée par npro.
- . qteliv est constituant de la c.classe identifiée par nco.
- . Il n'existe pas de RF entre nco et npro.

La règle de déduction (1er cas) permet de déterminer npro, nco comme l'identifiant de la c.classe où est situé monpro.

2ème cas :    monpro  
              /  \  
          pachat  taux

- . taux est constituant de la c.classe identifiée par nent.
- . pachat est constituant de la c.classe identifiée par npro.
- . Il existe la RF  $npro \rightarrow nent$ .

La règle de déduction (2ème cas) permet de déterminer npro comme identifiant de la c.classe où est situé monpro.

Démonstration

1er cas : Il n'existe pas de RF entre ciden (m2) et ciden (m3)

- . D'après la contrainte 3, ciden (m1) est composé à partir de ciden (m2) et ciden (m3).
- . D'après la contrainte 1, ciden (m1) est tel que ciden (m1)  $\rightarrow$  ciden (m2)
- ciden (m1)  $\rightarrow$
- ciden (m3).

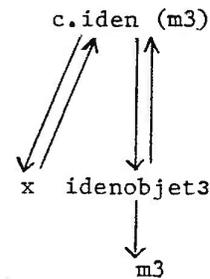
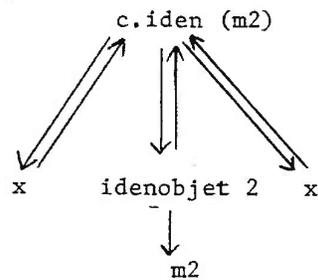
Une solution vérifiant les contraintes est :

ciden (m1) = ciden (m2), ciden (m3).

Vérification de la solution

Il faut vérifier,

- a) que les contraintes définies sont respectées
  - b) que l'indentifiant ainsi déterminé a les propriétés d'un indentifiant de c.classes.
- a) Par construction, ciden (m1) vérifie les contraintes 1 et 3. La contrainte 2 est contrôlée ultérieurement au cours de la méthode appliquée aux composantes dynamiques du SI.
  - b) Montrons que la RF ciden (m2), ciden (m3)  $\rightarrow$  m1 est élémentaire. Nous montrons par l'absurde qu'il n'existe pas les RF ciden (m2)  $\rightarrow$  m1, ou ciden (m3)  $\rightarrow$  m1. D'après la définition du système de c.classes, m3 est situé dans une seule c.classe ; d'après l'hypothèse du cas 1, il n'existe pas de RF ciden (m2)  $\rightarrow$  ciden (m3). Nous pouvons schématiser ceci par :

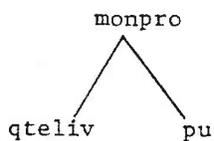


Il en découle qu'il n'existe pas de RF entre ciden (m2) et m3 ; à une réalisation de ciden (m2) correspondent plusieurs réalisations de m3.

m1 étant calculé en fonction de m3., à une réalisation de ciden (m2) correspondent plusieurs réalisations de m1 . La RF ciden (m2)  $\rightarrow$  m1 n'existe pas.

Nous démontrons de façon analogue que la RF ciden (m3)  $\rightarrow$  m1 n'existe pas.

Exemple :



- . ciden (qteliv) = nco identifiant de la c.classe commande
- . ciden (pu) = npro identifiant de la c.classe produit.

Soit, à un instant donné, les réalisations de nco, qteliv, npro, pu.

Composants	nco	qteliv
Réalisations		
réal. 1	C1	10
réal. 2	C2	5

Composants	npro	pu
Réalisations		
réal. 1	10	5
réal. 2	11	2

Nous remarquons qu'à une réalisation de commande correspondent plusieurs réalisations de monpro.

- A la réalisation C1 correspondent les réalisations 50 et 20 de monpro, obtenue en multipliant la réalisation de quantité livrée (10), par les différentes réalisations de pu (5, 2).
- De même, à une réalisation de npro correspondent plusieurs réalisations de monpro.

2ème cas : Il existe une relation fonctionnelle

- . Soit par exemple, la relation fonctionnelle entre les identifiants de c.classe ciden (m2)  $\longrightarrow$  ciden (m3) (R1).
- . La solution obtenue, vérifiant les contraintes 1 et 3, dans le 1er cas est : ciden (m1) = ciden (m2) (R2).
- . Par pseudotransitivité, R2 devient, en utilisant R1 :  
ciden (m2), ciden (m2)  $\longrightarrow$  m1.  
Soit, les réalisations de ciden (m2) étant identiques,  
ciden (m2)  $\longrightarrow$  m1.

Nous avons donc montré que ciden (m1) = ciden (mi); ciden (mi) étant l'identifiant source de RF.

3.6.1.2.2. - Généralisation de la règle de déduction

- . Soit m1 mot calculé en fonction de m2, m3, ..., mn pour tout i, i = 1, ..., n et c.iden<sub>i</sub> l'identifiant de la c.classe dont mi est un constituant.

Les identifiants sont simples ou composés.

c.iden 2 = c.iden 21, ..., c.iden 2 p  
 c.iden 3 = c.iden 31, ..., c.iden 3 q  
 c.iden n = c.iden n1, ..., c.iden n3.

1er cas : Il n'existe aucune RF entre les c.idenij

Alors c.iden 1 = c.iden 21, ... c.iden 2 p, c.iden 31, ..., c.iden 3 q  
 c.iden n1, ..., c.iden n3.

c.iden 1 est constitué de l'ensemble des c.iden ij noté Ciden.

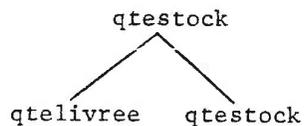
2ème cas : Il existe des RF entre c.idenij

Alors c.iden1 est constitué de l'ensemble C.iden des c.idenij  
 dont on a exclu les c.idenij buts de RF.

### 3.6.1.3. - Contrainte d'intégrité des mots de nature variable

- Les variables ne vérifient ni la règle 1, ni la règle 2. Nous  
 le montrons sur l'exemple suivant :

Dans le "traitement des commandes", pour chaque ligne de commande traitée,  
 la quantité en stock du produit est diminuée de la quantité livrée.



. c.iden (qtestock) = npro identifiant  
 de la c.classe  
 produit

. c.iden (qtelivree) = nlico identifiant  
 de la c.classe  
 ligne de commande.

La règle 1 est mise à défaut, car la RF npro → nlico n'existe pas.

- . Nous énonçons la contrainte suivante, vérifiée par les mots de nature variable, qui permet certains contrôles.

Contrainte 4 - Soient  $m_1$ , mot de nature variable, calculé en fonction de  $m_1, m_2$  et pour  $i = 1, 2$   $c.iden(m_i)$  l'identifiant de la  $c.classe$  dont  $m_i$  est un constituant.

Alors, soit il existe une RF entre  $c.iden(m_1)$  et  $c.iden(m_2)$ ,  
 soit il existe un identifiant de  $c.classe$   $c.iden k$  tel que  
 $c.iden k \longrightarrow c.iden(m_1)$   
 $c.iden k \longrightarrow c.iden(m_2)$ .

#### Justification

Pour effectuer la mise à jour de la variable  $m_1$ , il est nécessaire de faire correspondre une réalisation  $c.iden(m_2)$  à une réalisation  $c.iden(m_1)$ .

Cette correspondance s'exprime :

- soit par une RF entre les identifiants (cas  $a_1$  ou cas  $a_2$ ),

$c.iden(m_1)$   
 $\downarrow$   
 $c.iden(m_2)$

cas  $a_1$

$c.iden(m_1)$   
 $\downarrow$   
 $c.iden(m_2)$

cas  $a_2$

- soit par l'intermédiaire d'un autre identifiant  $c.iden k$

$c.iden(m_1)$  ←  $c.iden k$  →  $c.iden(m_2)$

- Si cette contrainte n'est pas respectée, l'énoncé de l'opération est insuffisant pour permettre l'exécution.

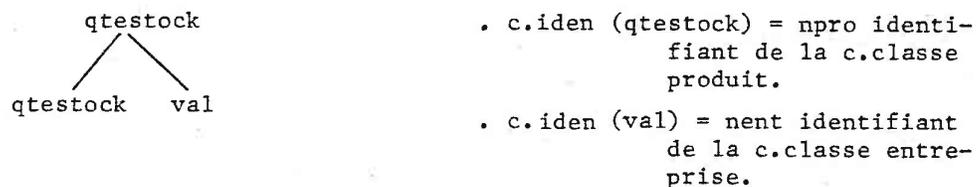
Exemples :

Cas a : Modifications de quantité stock dans le "traitement des commandes"



Il existe la RF nlico  $\longrightarrow$  npro.

La quantité en stock de chaque produit d'une entreprise est augmentée d'une certaine valeur. Cette valeur est constante, quel que soit le produit, et ne dépend que de l'entreprise.



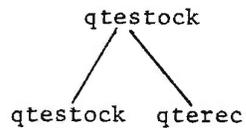
Il existe la RF npro  $\longrightarrow$  nent.

Cas b : Le bon de réception contient les références des produits reçus par l'entreprise, et la quantité reçue. Cette quantité ne dépend que du bon de réception, et est la même quel que soit le produit.

Exemple

Bon de réception:	Qtérée	Produits
1	2	p1 p2

La mise à jour de la quantité en stock est définie par :



- . c.iden (qtestock) = npro identifiant de la c.classe produit
- . c.iden (qterec) = nrec, identifiant de la c.classe bon de réception

Il n'existe pas de RF entre npro et nrec.

La c.classe "ligne bon de réception" identifiée par npro, nrec doit exister.

#### Comparaison avec Flory

Flory dans ( 14 ) a énoncé la règle suivante :

Type de calcul	Fonction	Type de relation nécessairement associée à la réalisation du calcul
Calcul itératif	$ri \leftarrow ri + rk$	Il existe une RF $IN'(rk) \longrightarrow IN(ri)$

. Notation :  $IN(rk)$  : identifiant de la rubrique r

. Définition du calcul itératif ( 14 ).

" Un calcul est itératif si la valeur de la rubrique calculée est obtenue à la fois par l'utilisation d'une unique valeur d'une rubrique intervenant dans le calcul et par la valeur précédemment prise par la rubrique calculée".

Nous ne pouvons pas appliquer cette règle aux définitions consécutives modifiant une variable car elle n'est pas toujours vérifiée. Ainsi, dans l'exemple précédent, la RF  $c.iden (qterec) \rightarrow c.iden (qtestock)$  n'existe pas.

#### 3.6.1.4. - Cas exclus des contraintes d'intégrité : l'affectation

##### Notation :

Nous notons  $c.iden (mi)$  l'identifiant de la  $c$ .classe dans laquelle est située  $mi$ .

L'affectation n'est pas un calcul, les contraintes précédemment définies ne sont pas vérifiées. Nous le montrons sur deux exemples.

##### A - Cas où la contrainte 1 n'est pas vérifiée (exemple 1)

Le concepteur affecte à  $datecli$  - date de la dernière commande du client - la valeur de  $dateco$  - date de la commande. Les commandes étant traitées chronologiquement, l'affectation  $datecli = dateco$  permet d'obtenir la valeur désirée de  $datecli$ .

$datecli$	$c.iden (datecli) = ncli$
$dateco$	$c.iden (dateco) = nco$

La RF  $ncli \rightarrow nco$  n'existe pas. La contrainte 1 est mise en défaut.

##### B - Cas où la contrainte 3 n'est pas vérifiée

Dans l'exemple 1, ainsi que dans l'exemple suivant, la contrainte 3 n'est pas vérifiée.

Le concepteur décide d'affecter à date facture la date du jour

date facture	c.ident (date facture) =
	nfac'
date du jour	c.ident (date du jour) =
	njour

La contrainte 3 n'est pas vérifiée, car njour n'est pas constitué à partir de nfac.

### 3.6.2. - Définitions itératives

Nous étudions les RF entre identifiant de c, classe des mots intervenant dans une définition itérative, et énonçons une contrainte permettant certains contrôles.

La définition itérative exprime qu'une réalisation d'un mot m1 est évaluée à partir de la liste des réalisations d'un mot m2.

$$m1 = \sum_R m2 \quad R : \text{ensemble des réalisations de } m2 \text{ intervenant dans la définition de } m1.$$

Représentation graphique :



Notation : Pour  $i = 1, 2$ , nous appelons c.ident ( $m_i$ ) l'identifiant de la c.classe dans laquelle est situé  $m_i$ .

Nous énonçons la contrainte suivante, analogue à celle citée pour les mots de nature variable, qui permet certains contrôles.

Contrainte 5 : Soient  $m_1$ , mot calculé à partir d'une liste de réalisations d'un mot  $m_2$  et  $c.iden(m_i)$ ,  $i = 1, 2$  l'identifiant de la c.classe dont  $m_i$  est un constituant.

Alors : a) soit il existe la RF  $c.iden(m_2) \longrightarrow c.iden(m_1)$   
 b) soit il existe un identifiant de c.classe  $c.iden k$  tel que :

$c.iden k \longrightarrow c.iden(m_1)$

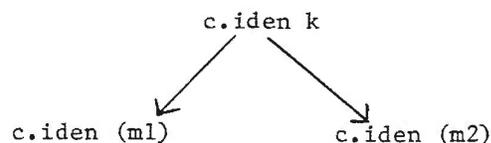
$c.iden k \longrightarrow c.iden(m_2)$ .

#### Justification

Pour calculer  $m_1$ , il est nécessaire de faire correspondre à une réalisation  $c.iden(m_1)$  un ensemble R de réalisations  $c.iden(m_2)$ .

a) Soit il existe la RF  $c.iden(m_2) \longrightarrow c.iden(m_1)$

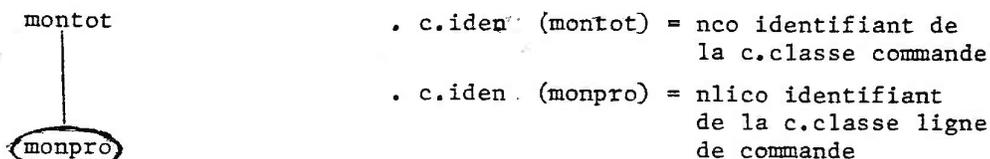
b) soit il existe l'ensemble de RF  $c.iden k \longrightarrow c.iden(m_1)$   
 $c.iden k \longrightarrow c.iden(m_2)$



De plus, la connaissance de ces RF permet, dans les deux cas, de déterminer l'ensemble R des réalisations de  $m_2$  intervenant dans la définition de  $m_1$ .

Exemples :

Cas a) : Dans le traitement d'une commande, le montant total de la commande est évalué à partir de la liste des montants de chaque ligne de commande.

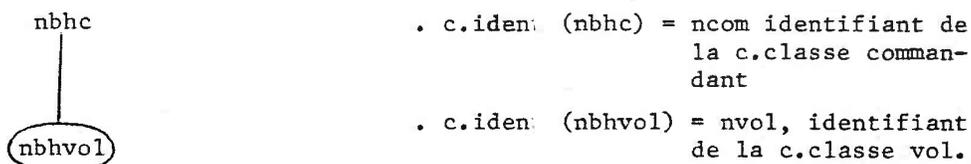


La RF nlico  $\longrightarrow$  nco permet d'associer à une réalisation de nlico une réalisation de nco.

De plus, pour chaque réalisation  $nco_i$  de nco, nous avons :

$$\text{montot} = \sum_R \text{monpro. } R \text{ étant l'ensemble des réalisations de nlico auxquelles correspondent la réalisation } nco_i.$$

Cas b) : Dans une compagnie aérienne, le nombre d'heures de vol du commandant (nbhc) est la somme des heures de vol (nbhvol) de chaque vol effectué par le commandant.



Un vol n'est pas attribué systématiquement à un commandant. Un commandant peut effectuer plusieurs vols. Il n'existe pas de RF entre nco et nvol.

Dans ce cas, il existe un identifiant  $c.iden\ k$  tel que les RF  $c.iden\ k \longrightarrow nvol$ ,  $c.iden\ k \longrightarrow ncom$  existent.

Par exemple,  $c.iden\ k$  peut être  $nvol$ ,  $ncom$  les RF  $nvol$ ,  $ncom \longrightarrow nvol$  ;  $nvol$ ,  $ncom \longrightarrow ncom$  existent.

L'ensemble  $R$ , tel que  $nbhc = \sum_R nbhvol$  est ainsi déterminé : pour chaque réalisation  $ncom_i$  de  $ncom$ ,  $R$  est l'ensemble des réalisations  $nvol_j$  de  $nvol$ , pour lesquelles il existe une réalisation  $ncom_i$ ,  $nvol_j$ .

$c.iden\ k$  peut être  $nvol$ ,  $ndate$ , si la RF  $nvol$ ,  $ndate \longrightarrow ncom$  existe. Dans ce cas,  $R$  est l'ensemble des réalisations  $nvol_j$  de  $nvol$  pour lesquelles il existe une réalisation de l'association  $(nvol_j, ndate)$  liée à  $ncom_i$ .

CONCLUSION

Dans le cadre du modèle conceptuel du SI, proposé par l'équipe Remora, nous avons présenté un modèle conceptuel de données, appelé système de c.classes, et une méthode de structuration de ce système de c.classes, intégrée dans le processus de structuration du schéma conceptuel.

Ce modèle, parce qu'il permet de prendre en compte le temps, est plus complet et original que les modèles relationnels ou entité - relation présentés jusqu'à présent; il serait intéressant de montrer qu'une base de données s'appuyant sur ce modèle contient effectivement l'image de l'évolution de l'organisation au cours du temps.

La méthode, reposant sur le principe de structuration par composition et le principe d'économie, introduit des règles qui pourront être utilisées dans la phase logique de la conception du SI pour compléter, par les fonctions d'accès, les textes des opérations définies dans la phase conceptuelle.

B I B L I O G R A P H I E

- 1 - ANSI/SPARC  
*Report on Data Management Systems.*  
1975
- 2 - BENCI (G.), BODART (F.), BOGAERT (H.), CABANES (A.)  
*Concept for the Design of a Conceptual Schema.*  
pp.378 à 429, IFIP working conference FREUDENSTADT (A11) 1976
- 3 - BERNSTEIN (P.A.)  
*Normalization and functional dependancies in the relational data base model.*  
Ph.D. Thesis. Université de Toronto, octobre 1975
- 4 - BERTINI (M.T.), TALLINEAU (Y.)  
*Cobol structuré : un modèle de programmation.*  
Edition d'informatique, 1973
- 5 - CHABRE-PECCOUD (M.)  
*Le projet ABREGÉ.*  
Thèse de 3ème cycle, Université de Grenoble, 1976
- 6 - CHEN (P.)  
*The entity relationship model toward a unified view of data.*  
ACM transactions on data base systems.  
Vol. 1 n° 1, mars 1976, pp. 9-36
- 7 - CHESSERON (M.)  
*Projet Remora : modèle de représentation et méthode de construction des compositions dynamiques d'un SI.*  
Thèse de 3è cycle, Nancy I, 1978
- 8 - CODD (E.F.)  
*A relational model of data for large shared data banks.*  
Communication ACM, vol 13, n°6, 1970

- 9 - CODD (E.F.)  
*Normalized Data base Structure : a brief tutorial.*  
Sigfider workshop 11-12, 1971
- 10 - COUGER (D.)  
*Evolution of Business System Analysis Techniques*  
*computing surveys.*  
Vol. 5, num.3, 1973
- 11 - DELOBEL (C.)  
*Contributions théoriques à la conception et à*  
*l'évaluation d'un système d'information appliquée*  
*à la gestion.*  
Thèse d'Etat, Grenoble, 1973
- 12 - DELOBEL (C.)  
*Les systèmes de base de données.*  
Polycopié, école d'été, AFCET, 1975
- 13 - DELOBEL (C.)  
*Semantics of relations and decomposition process in the*  
*relational data model.*  
Colloque INFORSID Nice, octobre 1977, publication  
IRIA, pp. 1-27
- 14 - FLORY (A.)  
*Un modèle et une méthode pour la conception logique d'une base*  
*de données.*  
Thèse d'Etat, Lyon, 1977
- 15 - FOUCAUT (O.), ROLLAND (C.)  
*Concepts for design of an information system conceptual*  
*schema and its utilization in Remora Project.*  
VLDB Berlin, 1978, Preprint CRIN, n° 78 PO 14

- 16 - HAINAUT (J.L.)  
*The individual Model.*  
Data Structure Models for Information Systems  
Namur, 1974
- 17 - INFORSID  
*Groupe Méthodes et Outils pour la conception et la réalisation des SI.*  
Colloque Caen, 1976, publ. IRIA, pp. 58-71
- 18 - KENT (W.)  
*Entities and relationship in information,*  
Modelling in data base management systems, IFIP  
TC 2. working conférence, Nice, janvier 1977
- 19 - LAMY (D.)  
*Les traitements : leurs structures et leur génération dans le projet Remora.*  
Thèse de 3ème cycle, Nancy I, 1977
- 20 - LUGUET (J.)  
*SCAPFACE.*  
Thèse d'Etat, Toulouse, 1975
- 21 - Méthode MINOS  
SLIGOS systèmes, 1970
- 22 - PECCOUD  
*MACSI.*  
Thèse d'Etat, Grenoble, 1975
- 23 - PEREA (W.)  
*Méthode d'analyse dans le projet Remora.*  
Thèse de 3ème cycle, Nancy I, 1976

- 24 - PROTEE  
Société d'Informatique et de Système, 1970
- 25 - REIX (R.)  
*L'analyse en informatique de gestion.*  
Dunod, 1971
- 26 - SENKO (M.), ALTMAN (E.), ASTRAHAM (M.), FEHDER (P.)  
*Data structures and accessing in data base systems.*  
IBM, syst. J. 12.1., 1973, pp. 30-93
- 27 - TARDIEU (H.)  
*The Individual Model.*  
Data structure models for Information Systems, Namur, 1974
- 28 - TEICHRGEW (D.), RATAJ (W.), HERSHEY (E.)  
*An introduction to computer aided documentation  
of user requirements for computer bases information processing  
system.*  
ISDOS, Working paper, n° 72, 1973
- 29 - THIERY (O.)  
*L'aide à la conception dans le projet Remora.*  
Thèse 3ème cycle, Nancy I, 1976
- 30 - WARNIER  
*La transformation des programmes.*  
Ed. d'organisation, 1973
- 31 - WATERS  
*Méthodologie assistée par ordinateur.*  
Info. Nouvelle, 1976



T A B L E    d e s    M A T I E R E S

<u>CHAPITRE I - L'ETUDE DANS LE CADRE DU PROJET REMORA.....</u>	1
1. L'objectif du projet Remora et sa justification.	2
2. Les hypothèses de travail du projet .....	3
3. Les directions de recherche et les résultats acquis.....	9
4. La situation de l'étude dans l'ensemble du niveau conceptuel.....	12
41. Notre attitude au niveau conceptuel.....	12
42. Rappels à propos du schéma conceptuel.....	13
421. Introduction.....	13
422. Le modèle conceptuel.....	14
4221. Le modèle descriptif de l'orga- nisation.....	14
4222. Le modèle conceptuel.....	16
42221. Correspondance en modèle conceptuel et modèle des- criptif.....	16
42222. Contraintes d'intégrité du modèle.....	17
423. Le schéma conceptuel.....	20
43. Rappels sur la méthode de construction du schéma conceptuel.....	21
431. L'analyse et la représentation de la perception des problèmes.....	22
432. La méthode de transformation du sché- ma de perception.....	27
44. Présentation de l'étude.....	27

CHAPITRE II - MODELE DE DESCRIPTION STATIQUE DU SYSTEME

<u>D'INFORMATIONS</u> .....	29
Introduction.....	30
1. Les concepts élémentaires : le mot.....	33
11. Définition du mot.....	33
12. Etude du mot.....	33
121. Sémantique.....	33
122. Désignation.....	33
123. Définition.....	34
124. Domaine des valeurs.....	34
125. Typologie des mots.....	37
2. Les liens entre les mots.....	39
21. Les relations fonctionnelles.....	40
212. Typologie des RF.....	41
213. Typologie des RF compte tenu du temps...	42
2131. Relation fonctionnelle permanente.	42
2132. Relation fonctionnelle semi-per-	
manente.....	44
2133. Relation fonctionnelle stable.....	46
22. Associations de mots.....	48
3. Les groupes.....	48
31. Groupes de mots.....	48
32. Définition d'une réalisation d'un groupe.....	48
33. Les RF et les groupes.....	48
34. Identifiant du groupe.....	49
35. Typologie des groupes.....	50
351. Groupe en 3 <sup>e</sup> FN.....	50
352. Groupe primaire GP.....	50
353. Définition du GP maximal.....	51
354. Le c.objet.....	51
3541. Définition.....	51
3542. Nature du c.objet.....	55
3543. Intérêt du concept de c.objet.....	58

4. c.classes.....	60
41. Collection de c.objets.....	61
42. Groupe de c.objets.....	62
43. Groupe maximal de c.objets.....	62
44. c.classe de c.objets.....	63
441. Définition.....	63
442. Unicité du c.objet racine d'une c.classe	64
443. Identifiant de c.classes.....	65
444. Ordre des c.classes.....	65
445. c.objet constituant d'une c.classe.....	65
45. c.classe maximale.....	66
451. Définition.....	67
452. Sémantique.....	72
453. Unicité de la c.classe maximale.....	72
454. Conséquences de la définition de la c.classe.....	73
455. Structure de la c.classe.....	73
5. Systèmes de c.classes.....	80
51. Définition.....	80
52. Exemple de système de c.classes.....	81
53. Sémantique du système de c.classes.....	83
54. Unicité du système de c.classes sur un ensemble de mots.....	85
55. Avantages du système de c.classes.....	85
56. Type des mots vis à vis du système de c.classes.....	89
6. Conclusion.....	90

CHAPITRE III - LA METHODE DE CONSTRUCTION DU SYSTEME DE

<u>C. CLASSES</u> .....	92
1. Les principes de la méthode.....	94
11. Principe de structuration par composition....	94
12. Principe d'économie.....	95
2. Caractéristiques.....	96
21. Le processus de structuration dépend de son domaine d'application.....	96
22. Le processus est sédimentaire.....	96
23. Le processus de structuration est assisté....	97
3. La méthode.....	99
31. Présentation générale.....	99
32. Définition et constitution de l'ensemble M...	99
33. Définition et constitution de l'ensemble RF des relations fonctionnelles sur M.....	101
34. Le raisonnement de structuration.....	102
341. Présentation générale.....	102
342. Règles de décomposition de M en sous- ensembles de mots.....	103
3421. Incidence du principe d'économie et de cohérence : un ordre de structuration des sous-ensembles de mots de M.....	103
3422. Règles de décomposition de M.....	103
3423. Conclusion sur l'ordre de struc- turation des éléments de M.....	107
343. Les règles de structuration.....	109
3431. Incidence du principe de structu- ration sur la recherche d'une solution.....	109
3432. Règles générales de structuration.	109

34321. Règles de structuration de mi en c.objets.....	110
34322. Règles de structuration des c.objets en c.classes	113
34323. Règles de structuration du système de c.classes..	114
344. Les règles d'intégration.....	119
3441. Règles d'intégration des c.objets	120
3442. Règles d'intégration des c.classes	121
3443. Règles d'intégration du système de c.classes.....	122
35. Processus de structuration et d'intégration	123
351. Processus appliqué aux mots définis dans les messages d'une procédure...	123
352. Processus appliqué aux mots définis dans les traitements d'une procédure..	128
36. Apport des traitements en matière de sémantique des données.....	132
361. Définitions consécutives.....	132
3611. Contraintes d'intégrité des mots de nature résultat.....	133
36111. Contrainte 1.....	133
36112. Contrainte 2.....	134
36113. Contrainte 3.....	136
3612. Règles de déduction des RF à partir des contraintes précédemment définies.....	137
36121. Règle de déduction n=2...	138
36122. Généralisation de la règle de déduction.....	141
3613. Contrainte d'intégrité des mots de nature variable.....	142
3614. Cas exclu des contraintes d'intégrité : l'affectation.....	146
362. Définitions itératives.....	147
<u>CONCLUSION</u> .....	151
<u>BIBLIOGRAPHIE</u> .....	154



NOM DE L'ETUDIANT : Mademoiselle KRIEQUER Monique

NATURE DE LA THESE : DOCTORAT DE 3e CYCLE en INFORMATIQUE



VU, APPROUVE

et PERMIS D'IMPRIMER

NANCY LE 13.JUIN 1978 4401

LE PRESIDENT DE L'UNIVERSITE DE NANCY I



M. BOULANGE