

## THESIS / THÈSE

### MASTER EN SCIENCES INFORMATIQUES

#### Réalisation d'un prototype d'un système informatique d'aide à l'élaboration d'un jugement

Poncin, Isabelle; Stenne, Pascale

*Award date:*  
1984

*Awarding institution:*  
Universite de Namur

[Link to publication](#)

#### General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

#### Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

FACULTES UNIVERSITAIRES NOTRE DAME DE LA PAIX à NAMUR

INSTITUT D'INFORMATIQUE

REALISATION D'UN PROTOTYPE D'UN  
SYSTEME INFORMATIQUE D'AIDE A  
L'ELABORATION D'UN JUGEMENT

---

Mémoire présenté par  
Isabelle PONCIN et  
Pascale STENNE  
en vue de l'obtention du  
titre de Licencié et  
Maître en Informatique

Année Académique 1983-1984

## Remerciements

Nous tenons à exprimer nos plus vifs remerciements à Monsieur le Professeur F. BODART, promoteur de ce mémoire, qui nous a guidées et encouragées, dans notre travail.

Nous remercions aussi Madame HELLA et Monsieur Yves POULLET de la Faculté de Droit, pour leur collaboration à ce travail.

Nous n'oublierons pas non plus l'aide et l'accueil que nous avons reçus lors de notre stage au laboratoire de l'IRETIJ à Montpellier.

Nous remercions nos parents qui nous ont permis de faire ces études, et qui nous ont encouragées tout au long de celles-ci.

Nous remercions Michel et Yves de leur gentillesse et de leur compréhension.

PLAN DU MEMOIRE

page :

INTRODUCTION

1. Différents aspects associés à l'élaboration d'un jugement	6
2. Utilisation d'un SIAD dans l'élaboration d'un jugement	7
3. Environnement de travail	8
4. Limitation et portée de notre travail	10

PREMIERE PARTIE : LE PROTOTYPE DOCUMENTAIRE

1. <u>Enoncé du problème</u>	11
1.1. Le schéma conceptuel	11
1.1.1. Les informations portant sur le jugement	11
1.1.2. Les informations portant sur les faits et standards	13
1.1.3. Lien sémantique entre les deux groupes d'informations	14
1.2. Conclusion	14

2. <u>L'existant</u>	15
2.1. Structures de données et manipulation	15
2.1.1. Dans les SGBD	15
2.1.1.1. Structures de données	15
2.1.1.2. Manipulation des données	17
2.1.2. Dans les systèmes documentaires	17
2.1.2.1. Structures de données	17
2.1.2.2. Manipulation des données	21
2.2. Langage d'interrogation	21
2.2.1. Dans les SGBD	21
2.2.2. Dans les systèmes documentaires	22
2.2.2.1. Recherche à différents niveaux	22
2.2.2.2. Comment l'utilisateur interroge-t-il?	23
2.3. Utilitaires de gestion de BD	24
3. <u>Proposition de solution</u>	25
3.1. Réalisation d'un prototype	25
3.1.1. Définition d'un prototype	25
3.1.2. Objectifs	25
3.1.2.1. Objectifs généraux	25
3.1.2.2. Objectifs spécifiques	26
3.2. Choix parmi les solutions possibles	26
3.2.1. Ne pas utiliser de logiciel existant	26
3.2.2. Utiliser un système de recherche documentaire	27
3.2.3. Utiliser un SGBD	27
3.3. Choix de l'environnement	28
3.3.1. Micro-ordinateur	28
3.3.1.1. Avantages et limites	28
3.3.1.2. Justification du choix	28
3.3.2. DBASE2	30
3.3.2.1. Particularités de DBASE2	30
3.3.2.2. Justification du choix	33

3.4. Spécification générale de la solution	33
3.4.1. Fonctions	33
3.4.1.1. Fonctions d'ajout	33
3.4.1.2. Fonctions d'interrogation	34
3.4.2. Interactivité	35
3.4.3. Système documentaire	35
4. <u>Spécifications fonctionnelles du prototype</u>	36
4.1. Le modèle de la structuration des informations	36
4.2. Le système de recherche documentaire	36
4.3. Les fonctions du système	38
4.3.1. Au niveau de l'interrogation	38
4.3.1.1. Interrogation sur les données structurées	38
4.3.1.2. Recherche documentaire	41
4.3.2. Au niveau de la mise à jour	43
4.3.2.1. Mise à jour des jugements	43
4.3.2.2. Mise à jour des faits et standards	45
4.3.2.3. Mise à jour des classes	45
4.3.2.4. Mise à jour des relations dans le lexique	46
4.4. L'interface avec l'utilisateur	47
4.4.1. Principes et outils pour la réalisation d'un interface	47
4.4.1.1. Représentation de l'interface	47
4.4.1.2. Principes de réalisation	48
4.4.1.3. Langage d'interface	49
4.4.1.4. Mesures de la facilité d'utilisation de l'interface	51
4.4.2. Réalisation de l'interface utilisateur pour la maquette	51
4.4.2.1. Choix d'un dialogue conduit par le système	52
4.4.2.2. Mise en oeuvre des principes pour la réalisation d'un interface	53
4.4.2.3. Mesures de la facilité d'utilisation	55

5. <u>Implantation du prototype</u>	56
5.1. Règles de transformation du schéma conceptuel entité/association en schéma relationnel	56
5.2. Implantation des faits et standards	57
5.2.1. Le schéma conceptuel	58
5.2.2. Description sémantique de schéma conceptuel	58
5.2.2.1. Les types d'entités	58
5.2.2.2. Les types d'associations	59
5.2.3. Transformation en un schéma relationnel	59
5.3. Traitement de texte	59
5.4. Architecture des modules	61
6. <u>Conclusion de l'implantation du prototype</u>	62
6.1. Critique	62
6.1.1. Outil de dialogue	62
6.1.2. Prototype rapidement disponible	63
6.1.3. Extensibilité du prototype	64
6.1.3.1. Au niveau du prototype réalisé	64
6.1.3.2. Au niveau de DBASE2	64
6.1.4. Gestion de données structurées et non structurées	65
6.1.5. "User-friendliness"	66
6.2. Le prototype vis-à-vis du système à grande échelle	66
6.2.1. Système multi-utilisateurs	66
6.2.2. Volume de données	67
6.2.3. Protection des données	67
6.2.4. Les performances d'exploitation.	67

DEUXIEME PARTIE : LE PROTOTYPE DU TEST DE VALIDITE

	page :
1. <u>Enoncé du problème</u>	68
1.1. Qu'entendons-nous par validité?	68
1.2. Comment réaliser un test de validité	70
2. <u>Outils existants pour réaliser ce test</u>	71
2.1. Langages traditionnels	71
2.2. Langages non-procéduraux	72
3. <u>Choix de solution</u>	73
3.1. Outil de démonstration	73
3.2. Environnement	74
3.3. Implantation	74
3.3.1. Base des connaissances	74
3.3.1.1. Les faits	74
3.3.1.2. Les règles	75
3.3.2. Les questions	75
3.3.2.1. Evaluation : INVALID	75
3.3.2.2. Ajout d'un groupe de validité : AJOUT	75
4. <u>Connexion avec le prototype documentaire</u>	76
4.1. Intégration des deux outils	76
4.2. Implanter tout le système en Prolog	77
5. <u>Conclusion</u>	78
5.1. Critique	78
5.1.1. Outil minimal sans interface	78
5.1.2. Outil performant et rapide	79
5.1.3. Outil isolé	79
5.2. Extensions	80
<u>CONCLUSION</u>	81
<u>BIBLIOGRAPHIE</u>	85
<u>ANNEXES</u>	

INTRODUCTION

Notre travail s'inscrit dans le cadre de la réalisation d'un système d'informations d'aide à la décision (SIAD) relatif à l'élaboration de jugements.

1. Différents aspects associés à l'élaboration d'un jugement

Nous reprenons ici certains aspects liés à la perspective d'un Système de Traitement de l'Information.

1.1. Aspect documentaire

Cet aspect consiste à permettre à toute personne impliquée à l'élaboration d'un jugement de consulter la doctrine et la jurisprudence.

Cette consultation à des fins documentaires peut s'effectuer de différentes façons. D'une part elle donne la possibilité de retrouver des textes bruts à partir de critères de sélection; d'autre part elle donne accès aux textes structurés des jugements (renseignements généraux, personnes impliquées, formalisation de la règle de droit).

1.2. Aspect comparatif

Au moment de l'élaboration d'un jugement, le juge peut rechercher des jugements "similaires" c'est-à-dire faisant intervenir les mêmes faits, soit pour être guidé dans son travail, soit pour tester si la formalisation de la règle de droit qu'il établit est confirmée dans d'autres jugements.

Il désire par exemple voir s'il a accordé la même importance aux mêmes faits, s'il a la même structure logique, ...

Ceci peut donc aller jusqu'à fournir un outil de calcul de distance entre jugements.

Cet aspect comprend donc une composante documentaire, la partie Faits-et-Standards de la base documentaire.

### 1.3. Aspect d'évaluation

#### 1.3.1. Cohérence logique interne

On désire vérifier que la formalisation de la règle de droit d'un jugement est logiquement cohérente.

#### 1.3.2. Cohérence logique externe

On désire vérifier que la formalisation de la règle de droit d'un jugement n'est pas contradictoire par rapport à des éléments extérieurs, relevant de la doctrine et établis par des juristes. En effet, ces éléments sont a priori compatibles avec tous jugements sous peine de non validité du jugement.

### 1.4. Vérité d'un jugement

Nous mentionnons pour mémoire la vérification de la vérité d'un jugement, c'est-à-dire savoir si le jugement est "conforme" à la réalité, car cet aspect relève de l'éthique et n'est donc pas à notre portée.

## 2. Utilisation d'un SIAD, dans l'élaboration d'un jugement

### 2.1. Définition d'un SIAD

" Un SIAD est un système d'information recourant à des technologies basées sur l'ordinateur, utilisé comme support à la décision dans des situations où il n'est ni possible, ni désirable d'avoir un système automatique pour effectuer le processus de décision complet".

C'est la définition donnée par GINZBERG et STOHR.

Nous en retiendrons principalement deux éléments :

- système d'information
- recourant à des technologies basées sur l'ordinateur.

### 2.2. Fonctions à réaliser

2.2.1. Nous assimilons le prototype que nous voulons réaliser à un SIAD qui comporterait des fonctions associées aux différents aspects dégagés pour l'élaboration d'un jugement.

Nous distinguons les fonctions suivantes :

- formalisation du jugement
- fonctions d'interrogation de la Base de Données (BD) (SGBD et/ou système de recherche documentaire)
- fonctions de la mise à jour de la BD
- fonction de comparaison de jugements
- fonction de test de validité interne d'un jugement
- fonction de test de validité externe d'un jugement .

2.2.2. Parmi celles-ci, nous avons réalisé les fonctions d'interrogation de la BD, de mise à jour de la BD, de test de la validité externe d'un jugement.

### 2.3. Caractéristiques d'implantation

Un SIAD doit :

- être construit sur un canevas adaptable c'est-à-dire être capable de suivre l'évolution du problème concerné
- interagir avec l'utilisateur : ce dernier doit apprendre à mieux connaître le système en l'utilisant. C'est ainsi qu'il lui découvrira de nouvelles possibilités à lui ajouter et le rendra dynamique
- être facile à connaître et à utiliser (user-friendliness).

## 3. Environnement du travail

### 3.1. Au niveau juridique

Notre principal acquis est le travail qu'ont réalisé Messieurs HICK et TIRTIAUX dans leur mémoire. Ils ont structuré le problème juridique, de façon à bien distinguer le jugement

dans lequel est développée la formalisation de la règle de droit (représentée dans le formalisme de la logique propositionnelle) et les éléments de droit et factuels qui y interviennent. Ce travail a abouti à la réalisation d'un schéma conceptuel, repris dans l'annexe 1.

On peut y distinguer les différents types d'informations suivants :

- les renseignements généraux portant sur un jugement (date, ville, ...)
- les personnes impliquées
- les compositions qui formalisent la règle de droit
- les éléments de droit (standards) et les faits qui peuvent intervenir dans la démarche juridique.

Messieurs HICK et TIRTIAUX ont aussi jeté les bases de la réalisation manuelle d'un test de validité externe d'un jugement.

### 3.2. Au niveau matériel et logiciel

#### 3.2.1. Aspect documentaire

Nous avons décidé de réaliser le prototype sur un micro. A l'Institut d'Informatique, nous avons la possibilité d'utiliser un Apple ou un PC/IBM; notre choix s'est porté sur ce dernier car il offrait des avantages tels que la capacité d'un disque dur, ...

Au niveau des logiciels, nous voulions utiliser un Système de Gestion de Bases de Données (SGBD).

Or DBASE2 était disponible sur le PC/IBM et avait déjà été utilisé pour réaliser des SIAD. Aussi avons-nous décidé de l'utiliser.

#### 3.2.2. Aspect d'évaluation

Nous avons décidé d'utiliser Prolog pour réaliser le test de validité externe. Car le jugement a été formalisé, en utilisant la logique propositionnelle et Prolog est un langage basé sur la logique des prédicats (clauses de HORN).

Comme matériel, nous avons utilisé le DEC 20.

#### 4. Limitation et portée de notre travail

##### 4.1. Limitation

4.1.1. Nous avons pris comme hypothèse de départ le schéma conceptuel de Messieurs HICK et TIRTIAUX, alors que nous savions qu'il avait des inconvénients :

- le manque de structuration des faits et des standards
- isolement des faits et standards de leur contexte spatial et temporel
- il n'existe pas d'accord parmi les juristes, sur la notion de standard.

4.1.2. Nous n'avons réalisé que les fonctions documentaires au sens large (c'est à dire les fonctions d'interrogation et de mise à jour de la B.D.) et l'aspect de test de la cohérence externe d'un jugement.

4.1.3. Au niveau documentaire, nous avons réalisé une maquette, cela signifie que :

- nous ne nous préoccupons pas de la performance
- la B.D. est réduite à un jeu de données
- la maquette est mono-utilisateur
- il n'y a pas de protection de données.

Au niveau du test de la validité externe, nous nous sommes limitées à un outil de démonstration qui ne comprend pas toutes les caractéristiques d'un SIAD ( ex. userfriendlyness).

##### 4.2. Portée

Dans l'immédiat, notre travail n'a pas l'ambition de servir à l'élaboration d'un jugement, mais il pourra être utilisé à des fins pédagogiques.

Le prototype peut aider des étudiants dans l'apprentissage de la formalisation d'un jugement et du test de sa validité.

Il leur permet aussi d'expérimenter de manière critique les concepts de fait et standard.

---

## PREMIERE PARTIE

---

### LE PROTOTYPE DOCUMENTAIRE

---

#### 1. ENONCE DU PROBLEME

Le domaine de Droit qui est analysé concerne les contrats de cautionnement, c'est-à-dire les contrats par lesquels une personne appelée "caution" promet à un créancier d'exécuter les obligations d'une tierce personne appelée "débiteur-principal" si celui-ci ne tient pas ses engagements.

L'aide apportée par un système informatique se base sur la jurisprudence relative au problème d'extinction du cautionnement. Un modèle sémantique des données de ce problème a été développé dans le mémoire de Messieurs HICK et TIRTIAUX(1).

Le schéma conceptuel qui y a été élaboré se trouve en Annexe 1. Notre travail consiste à l'exploiter à des fins documentaires. Il est donc intéressant d'analyser plus en détail les informations reprises dans ce schéma conceptuel.

#### 1.1. Le schéma conceptuel

Dans le schéma conceptuel, nous distinguons principalement deux groupes d'informations, qui ont une vie relativement autonome, c'est à dire qu'elles sont indépendantes tant au point de vue sémantique, qu'au point de vue de leur gestion.

#### 1.1.1. Les informations portant sur le jugement

Un jugement comprend obligatoirement

- des renseignements généraux

(ville, date, type, résultat, texte)

qui sont représentés comme des propriétés de l'entité "jugement".

- des personnes qui ont un certain rôle dans un jugement. Les personnes sont représentées comme des occurrences d'un type d'entité "personne". Les rôles sont représentés comme des occurrences d'un type d'association (ex. créancier)
  - caractéristiques :
  - . une personne peut intervenir dans différents jugements, avec des rôles différents
  - . une personne ne sera pas reprise dans la Base de Données (B.D.), si elle ne joue pas un rôle dans un jugement
  - . dans un jugement, certains rôles doivent obligatoirement être remplis par des personnes (ex. créancier)
  
- une ou des compositions qui formalisent l'argumentation juridique. Une composition est représentée comme une occurrence d'un type d'entité "composition"
  - caractéristiques :
  - . une composition ne se rapporte qu'à un seul jugement
  - . une composition ne peut avoir qu'un conséquent
  - . une composition comprend des compositions et/ou des faits et standards pondérés; ces derniers sont représentés comme des occurrences d'un type d'entité "fait-et-standard-pondéré".

Ces informations constituent la description au "sens large" d'un jugement.

Elles sont toutes structurées sauf le texte du jugement.

Nous entendons par informations structurées, des informations qui peuvent être modélisées ( ex. modèle entité/association), c'est-à-dire qui correspondent à des classes sémantiques définies a priori.

### 1.1.2. Les informations portant sur les faits et les standards

On a regroupé sous un même type d'entité deux concepts différents, parce qu'ils peuvent intervenir de la même façon dans un jugement.

Un fait est une caractéristique d'une association entre certains éléments (personnes, contrat...) intervenant dans le problème de droit concerné.

ex. : les statuts de la société débiteur principal prévoient un changement de forme de celle-ci.

Un standard est une fonction décrite par l'autorité, par laquelle un juge associe à un certain nombre de faits ou de groupes de faits, un certain effet juridique prédéterminé (Définition donnée par Monsieur Y. POULLET).

ex. : standard de l'acte abstrait :

un acte abstrait est un acte séparé de sa cause. Le standard est basé sur le principe doctrinal qui affirme que l'acte abstrait vaut par lui-même. Peu importe la présence ou l'absence de la cause du contrat, celui-ci est toujours valable.

Un fait-et-standard comprend obligatoirement

- des renseignements (texte, polarisation) qui sont représentés comme des propriétés de l'entité "fait-et-standard",
  - la référence à la classe à laquelle, il appartient.
- Les classes sont représentées par les occurrences d'une entité "classe".

caractéristiques :

- . Un fait ou un standard appartient à une et une seule classe.
- . Tous les standards appartiennent à la même classe.
- . Le concept de classe est défini indépendamment du concept de fait-et-standard.

La plupart des informations relatives aux faits et standards sont non structurées, c'est-à-dire qu'elles n'ont pas une structure sémantique définie a priori, nous ne pouvons donc pas leur donner

un modèle prédéfini.

Les informations sont essentiellement composées de texte.

Un fait-et-standard est donc bien une information non structurée puisqu'il est principalement caractérisé par son texte.

Il en est de même pour les classes.

### 1.1.3. Lien sémantique entre les deux groupes d'informations

Le lien qui existe entre les informations portant sur les jugements et celles portant sur les faits-et-standards est le suivant : un fait ou un standard peut intervenir dans la démarche juridique élaborée dans un jugement, par l'intermédiaire des compositions et avec un certain poids (notion de pondération).

Ce lien est d'un grand intérêt pour l'aspect documentaire.

### 1.2. Conclusion

Le problème consiste à exploiter à des fins documentaires une Base de données constituée d'informations relativement autonomes et de nature différente: les unes ont une structure sémantique définie a priori, les autres pas.

Ces deux types d'informations sont généralement manipulés par des outils distincts : les systèmes de gestion de Bases de Données et les systèmes documentaires.

Même si certains systèmes documentaires (2) permettent des recherches combinées sur des données structurées et textuelles, l'intégration de ces deux types d'outils relève encore du domaine de la recherche. C'est dans cette optique que nous nous sommes fixé comme objectif de notre mémoire la réalisation d'un prototype qui gère les données structurées et textuelles du domaine de droit analysé.

## 2. L' EXISTANT

Les deux types de données, structurées et non structurées, sont gérés actuellement par des systèmes distincts. Aucun ne combine les deux aspects.

Les Systèmes de gestion de Bases de données ( SGBD ) classiques gèrent des données structurées et les Systèmes documentaires permettent de maintenir et d'exploiter des données textuelles, non-structurées.

Nous comparons ces deux familles de système sur base de trois composants, qui sont :

- la structure de données et sa manipulation
- le langage d'interrogation
- les utilitaires de gestion de B.D.

### 2.1. Structure de données et manipulation

#### 2.1.1. Dans les SGBD (3)

On parle de DDL (langage de description de données) et de DML (langage de manipulation de données).

##### 2.1.1.1. Structure de données

Le DDL permet de donner la description du modèle sémantique de structuration des données ainsi que la structure d'accès associée. Le modèle des informations gérées par des SGBD est souvent un modèle sémantique riche.

On distingue deux grandes approches dans les structures de données des SGBD :

##### approche relationnelle :

Les données sont reprises dans des tables appelées relations. Toute information du modèle (entité/association p. ex.) est représentée d'une manière uniforme c'est à dire par une table.

Les éléments de la structure sont :

- domaine ou ensemble de valeurs qui représente un élément du monde réel ou une propriété d'un tel élément.  
ex. : Fournisseur, nom
- type de relation ou ensemble de toutes les associations de même nature entre différents domaines.

Le Système R, INGRES et ORACLE sont des SGBD relationnels.

Exemples :

relation FOURNISSEUR (NOM, NUMERO, VILLE)

NOM	NUMERO	VILLE
Adam	1	NAMUR
Collard	2	BRUXELLES

relation LIVRE (NROFOURN, NROPIECE)

NROFOURN	NROPIECE
1	36
1	48
2	36
2	54

approche réseau :

Les données sont représentées par des types d'articles et des liaisons existant entre ceux-ci.

Les éléments de la structure sont :

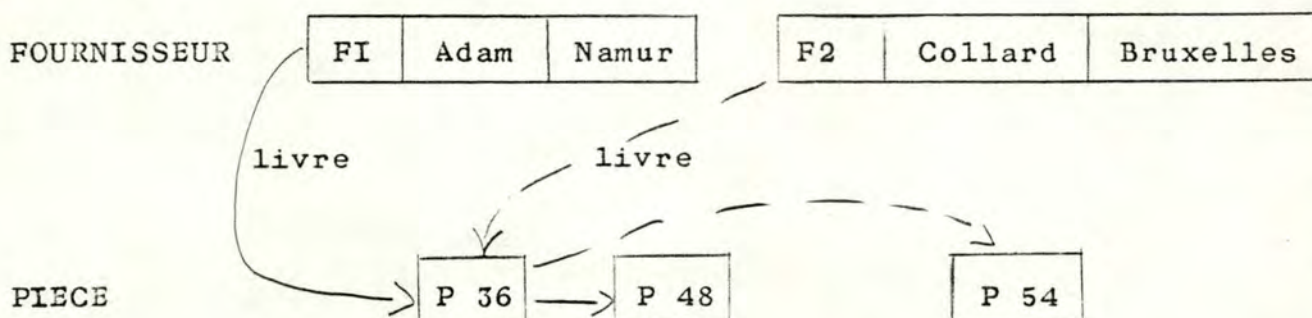
- type d'article : collection de données se rapportant à un concept. ex.: Fournisseur
- propriété : donnée qui peut prendre un ensemble de valeurs et qui se rapporte à un type d'article. ex.: Ville
- type de chemin : collection d'articles reliés entre eux par une relation prédéfinie. ex.: Livrer

CODASYL est fort répandu dans l'approche réseau.

Il existe différentes structures en réseau selon les restrictions que l'on introduit sur les liaisons permises.

La structure hiérarchique est un cas particulier de la structure en réseau, où les données sont représentées par une simple structure en arbre (ex. : TOTAL IMS).

Exemple :



### 2.I.I.2. Manipulation des données

On dispose de primitives de manipulation des données.

Ces primitives s'appliquent à un élément de la structure des données et permettent d'agir ponctuellement sur celle-ci (ex. : quand je suis positionnée sur une ligne d'une relation, je peux modifier la valeur d'un domaine).

Le DML est donc directement fonction du modèle de structuration des données.

La manipulation recouvre des opérations

- de recherche  
ex. : get next FOURNISSEUR
- de calcul  
ex. : maximum, minimum, moyenne, somme des valeurs d'un domaine d'une relation.
- de mise à jour  
ex. : insertion, suppression, modification d'un article

### 2.I.2. Dans les systèmes documentaires

#### 2.I.2.I. Structure de données

Le modèle des données est très pauvre par rapport à celui des données structurées d'un SGBD. Les systèmes de recherche docu-

mentaire gèrent des BD textuelles, constituées d'un seul type de documents qui sont des entités "textes".

Ils peuvent être :

- référence bibliographique :  
pour des documents qui peuvent être résumés et stockés sous forme condensée, les références enregistrées donnent l'information nécessaire pour retrouver le document de base.  
ex. : rapport scientifique, exposé de recherche
- texte intégral :  
pour des documents qui ne tolèrent pas d'être condensés et sont stockés in-extenso; l'enregistrement des documents dans la base documentaire se substitue ici aux textes sur papier.  
ex.: document de type juridique ou littéraire.

Un document se décompose en champs et on distingue deux types de champs :

- champs descripteurs ou mots-clés  
(abstract, auteur, ...)
- champs textes  
(résumé, titre, texte, ...)

Le champ peut dans certains cas être divisé en paragraphes, eux-mêmes comprenant plusieurs phrases composées de mots.

Ainsi, faute d'un modèle sémantique rigoureux des données, les systèmes documentaires ont besoin d'outils spécifiques :

- l'indexation
- les lexiques et thésaurus.

### Indexation

L'indexation consiste à extraire d'un texte les termes significatifs de son contenu.

Elle peut être de trois types différents :

- Indexation manuelle.

Elle est réalisée par un documentaliste spécialisé dans un domaine particulier, avant l'enregistrement du document.

Les descripteurs choisis doivent exprimer le mieux possible

le contenu du document.

Une indexation assez fine est nécessaire, si on veut retrouver le document avec précision.

Il est souhaitable que ce soit la même personne qui indexe les documents d'un même type afin de conserver une uniformité dans le choix du vocabulaire et des formes grammaticales.

- Indexation semi-automatique

Elle nécessite la définition de certains outils par le documentaliste, par exemple un dictionnaire des mots vides, ainsi qu'une syntaxe précise dans la rédaction des champs du document. C'est lors de l'enregistrement du document que le système documentaire l'indexera en caractérisant les champs (résumé, titre, ...) par des mots-clés.

Ce type d'indexation assure une homogénéité dans les descripteurs, mais est seulement satisfaisant pour des textes courts et simples.

Une méthode plus efficace est requise pour le traitement des textes riches et longs.

- Indexation automatique

Elle est entièrement réalisée par le système documentaire. C'est sur base de traitements linguistiques que s'effectue l'indexation automatique; ceux-ci comprennent des fonctions telles que découpage du texte, analyse morphologique, reconnaissance d'expression, normalisation.

Le processus d'indexation nous fournit ainsi des mots-clés. Les fichiers inversés sont constitués à partir de ce traitement et font correspondre à tout descripteur ses localisations dans les différents documents enregistrés. C'est sur base de ces fichiers que se fait la recherche primaire (cfr. infra).

Notons que les différents types d'indexation peuvent être utilisés pour un même document, selon les champs.

### Lexique et thésaurus

Des descripteurs sont extraits des champs du type "mot-clé". Deux possibilités de stockage se présentent selon que l'on établit ou non des relations entre ces mots-clés :

- le lexique est un fichier reprenant par ordre alphabétique les descripteurs retenus; il n'y a aucune relation entre eux.
- le thésaurus est un fichier reprenant les descripteurs qui sont reliés à d'autres par des relations de synonymie, de hiérarchie, de corrélation ou autres.

Un thésaurus ou lexique peut être ouvert ou fermé,

- ouvert : le fichier se constitue progressivement c'est-à-dire que lors du chargement de nouveaux documents dans la BD, les valeurs de descripteurs encore inconnues jusque-là sont incorporées automatiquement dans le fichier sans contrôle d'unicité, ni de permanence des descripteurs utilisés.
- fermé : par rapport au fichier existant, tout nouveau descripteur rencontré est rejeté dans un premier temps. Le fichier est constitué a priori pour une période déterminée et sa mise à jour constitue une procédure spécifique.

Le choix entre ouvert et fermé est difficile à faire.

Un lexique ou thésaurus fermé possède des avantages tels que :

- les fautes d'orthographe ne provoquent pas la présence de plusieurs descripteurs signifiants la même chose, ce qui empêcherait de retrouver les documents désirés, si l'orthographe du mot de la requête diffère de celle du mot qui indexe.
- la taille du fichier est contrôlable, alors que si le documentaliste n'est pas obligé de se cantonner dans un certain vocabulaire pour ses descripteurs, le fichier pourra prendre des dimensions élevées.

Par contre le caractère fermé nécessite une constitution a priori; cela demande une bonne connaissance du domaine afin de déterminer le vocabulaire le plus adéquat pour l'indexation. Des procédures de mise à jour sont requises.

### 2.1.2.2. Manipulation des données

La structure des données étant très pauvre, les systèmes documentaires n'offrent pas non plus beaucoup de primitives de manipulation de ces données.

La manipulation des textes est réduite à des opérations de saisie, modification, suppression de tout ou partie du document.

Notons aussi que des progrès sont nécessaires pour les fonctions de saisie et modification de documents car actuellement elles demandent de constituer des fichiers en utilisant une syntaxe assez rébarbative.

Les outils spécifiques créés nécessitent également des mises à jour : prise en compte de l'indexation, mise à jour des mots du lexique, des relations du thésaurus.

## 2.2. Langage d'interrogation ( QUERY LANGUAGE )

### 2.2.1. Dans les SGBD (3)

Les SGBD fournissent un langage d'interrogation qui leur est propre et qui est fonction du modèle de structuration des données. Ce langage permet d'interroger la BD en combinant des critères de recherche portant sur les différents éléments de la structure de données :

```
ex.  SELECT  nroprièce, prix
      FROM    fournisseur
      WHERE   ville = 'Namur'
      ORDER BY nroprièce
```

Cet exemple illustre l'utilisation de SQL, qui est le langage d'interrogation du système R (IBM).

Notons que la tendance actuelle est de construire des langages d'interrogation de plus haut niveau, qui soient de plus en plus proches du langage naturel.

On voudrait parfois faire entrer ces opérations d'interrogation dans un programme d'application.

On se trouve face à deux possibilités :

- d'une part des langages de SGBD qui recouvrent les structures classiques de programmation ainsi que les primitives de recherche sur les éléments de la BD.

ex. DBASE2

- d'autre part des langages d'interrogation qui peuvent être utilisés par un langage "hôte", normal de programmation.

ex. Embedded SQL dans PL1 . Il faut gérer l'interface entre ces deux langages.

### 2.2.2. Dans les Systèmes Documentaires

Les systèmes de recherche documentaire ayant une structure de données pauvre, ils offrent un langage d'interrogation assez simple qui permet de retrouver rétrospectivement des documents de la BD sur base d'une combinaison de mots-clés.

L'interrogation tient compte des outils spécifiques créés (lexique, thésaurus, indexation).

Voyons - en quoi consiste la recherche

- comment l'utilisateur interroge la BD.

#### 2.2.2.1. Recherche à différents niveaux

Tous les systèmes documentaires étudiés offrent deux niveaux complémentaires d'interrogation :

- la recherche primaire

Elle est basée sur des fichiers inverses qui sont constitués à partir de l'indexation des champs descripteurs.

Ce sont les lexiques ou thésaurus qui contiennent le résultat de cette indexation.

Cette recherche fournit un nombre de documents sélectionnés.

- la recherche secondaire

Elle s'effectue sur les documents sélectionnés par une

recherche primaire et porte sur le texte des différents champs des documents. Elle se fait par balayage du texte. Le résultat revient à sélectionner de manière plus approfondie certains documents. On parle dans les différents systèmes de recherche sur texte, recherche approfondie ou postsélection. Evidemment, elle n'est guère utilisée que pour les champs du type texte, la recherche primaire étant nettement plus performante (fichiers inverses) pour les autres champs.

#### 2.2.2.2. Comment l'utilisateur interroge-t-il?

L'utilisateur doit d'abord faire une recherche au premier niveau. Toute recherche lancée est appelée étape de recherche et est mémorisée.

Une équation de recherche de ce type est constituée d'une combinaison booléenne de descripteurs et de résultats d'étapes précédentes. La solution d'une telle équation est le nombre de documents dont l'indexation satisfait à la requête.

Lorsque ce résultat est trop vaste et que l'utilisateur veut raffiner, il peut faire une recherche sur texte.

Ici, des opérateurs plus ou moins puissants permettent :

- de rechercher la présence ou l'absence d'un mot ou d'un groupe de mots,
- de comparer des valeurs numériques ou alphabétiques dans un ou plusieurs champs spécifiés,
- de rechercher des distances entre mots ou groupes de mots.

Après chaque recherche, l'utilisateur peut visualiser tout ou partie(s) des documents sélectionnés à l'une des étapes.

Toute instruction primaire ou secondaire qui correspond à une étape de recherche est numérotée et mémorisée avec le nombre de documents sélectionnés.

Cela constitue la liste de recherche qui reflète l'historique de la session au terminal.

A tout moment, il est possible de visualiser cette liste ainsi que les documents sélectionnés à une quelconque de ses étapes.

Diverses opérations sont possibles sur la liste de recherche : suppression d'une ou plusieurs étapes, reprise à partir d'une étape donnée, ...

La gestion de cette liste est importante pour l'utilisateur car elle constitue la principale base de communication.

### 2.3. Utilitaires de gestion de B.D.

Les SGBD et les systèmes documentaires fournissent d'une manière semblable des utilitaires de gestion de la concurrence d'accès aux données, des utilitaires de gestion des points de reprise assurant la fiabilité du système, des utilitaires de protection des données, des utilitaires de réorganisation de BD suite à des insertions ou suppressions de données, des utilitaires de gestion de la communication au sein de réseaux, des mécanismes d'optimisation des accès aux données par des fonctions transparentes, ...

#### Remarque :

Nous avons étudié de manière plus approfondie quatre systèmes documentaires :

GOLEM (SIEMENS), MISTRAL (BULL), STAIRS (IBM) et le système développé à l'IRETIJ de Montpellier.

Une étude fonctionnelle de chacun d'eux se trouve en Annexe 2.

### 3. PROPOSITION DE SOLUTION.

Ce chapitre décrit la proposition de solution :

la réalisation d'un prototype, le choix pris parmi les solutions possibles, l'environnement retenu et la spécification générale de la solution considérée.

#### 3.1. Réalisation d'un prototype

Nous avons décidé de réaliser un prototype pour exploiter à des fins documentaires les données relevées lors de l'identification du problème.

##### 3.1.1. Définition d'un prototype (4)

Un prototype est une réalisation à échelle réduite, c'est à dire :

- la BD comprend des fichiers avec un nombre limité de données.
- le système est mono-utilisateur et ne doit donc pas gérer la concurrence d'accès aux données.
- pas de mécanisme assurant la fiabilité, tel que gestion de copies de sécurité des données, gestion de points de reprise.
- pas de préoccupation de performance d'exploitation.

##### 3.1.2. Objectifs

###### 3.1.2.1. Objectifs généraux que nous assignons au prototype :

- outil de dialogue :  
c'est un des objectifs essentiels du prototype. Ce dernier est un outil privilégié de communication entre l'informaticien et l'utilisateur non informaticien.  
L'utilisation du prototype est principalement destinée à susciter les réactions de l'utilisateur et donc à préciser les caractéristiques du SIAD à élaborer.
- outil rapidement disponible :  
c'est par manipulation de cet outil que l'utilisateur réagira en mentionnant les changements qu'il veut voir apparaître

pour le système opérationnel; il est donc important que la première partie c'est-à-dire la réalisation du prototype se fasse dans un délai acceptable.

- outil extensible, évolutif :  
le prototype devra être modifié pour faire face aux nouvelles exigences de l'utilisateur, correspondant à des besoins mal ou non spécifiés au départ. Ces changements doivent pouvoir être faits rapidement et facilement c'est-à-dire sans trop affecter ce qui existe et peut-être par l'utilisateur lui-même.

#### 3.1.2.2. Objectifs spécifiques à notre problème

- deux types de données :  
le prototype exploitera, à des fins documentaires, indifféremment les données structurées et non structurées.  
cela comprend des fonctions d'interrogation et de mise à jour de la BD.
- interface :  
le prototype étant un outil de dialogue avec un utilisateur non informaticien, nous attachons une grande importance à la réalisation d'un interface le plus "user-friendly" possible.

#### 3.2. Choix parmi les solutions possibles

Faisons le panorama des solutions qui s'offraient à nous pour réaliser le prototype :

##### 3.2.1. Créer tout le système, sans utiliser aucun logiciel.

Cela signifie utiliser un langage de programmation tel que BASIC ou PASCAL, puisque nous voulons implanter le prototype sur un micro-ordinateur. Or le BASIC n'offre guère de gestion des données (structures de fichiers assez pauvres...).

Cette solution nous paraît donc assez lourde et peu enrichissante à notre point de vue car elle ne nous permet pas d'utiliser de nouveaux outils.

### 3.2.2. Utiliser un système de recherche documentaire

La solution d'utiliser un système de recherche documentaire pour la réalisation de notre prototype, bien que peut-être intéressante, s'est heurtée à des inconvénients majeurs :

- l'utilisation d'un micro .

Or les systèmes de recherche documentaire que nous connaissons (Mistral, Golem, Stairs, le Système utilisé à l'IRETIJ à Montpellier) ne sont portables que sur des gros ordinateurs. Cela est sans doute dû au fait que ces systèmes offrent des fonctions qui sortent du cadre de la recherche documentaire.  
ex.: tri des documents repris dans les rapports

- un système de recherche documentaire n'admet que des BD avec un seul type de "document".

Or nous avons à gérer dans notre problème deux types d'informations qui sont indépendantes mais qui peuvent entrer en relation. Il n'est donc pas possible d'intégrer une gestion des données structurées dans ces systèmes de recherche documentaire.

### 3.2.3. Utiliser un SGBD

Cette solution est fort différente suivant le type de SGBD qu'on utilise.

#### 3.2.3.1. SGBD avec seulement un "query langage" ( ex. MDBS)

Cela signifie utiliser un langage de programmation "normal" ( PASCAL, COBOL, BASIC ...) dans lequel on peut faire des appels à des fonctions du "query system" du SGBD.

#### 3.2.3.2. SGBD avec un langage de programmation ( ex. DBASE2 )

Choisir un tel SGBD signifie : implanter tout le prototype dans le langage de programmation du SGBD. C'est cette solution que nous avons retenue.

### 3.3. Choix de l'environnement

Les deux composants de l'environnement que nous nous sommes donné pour l'implantation du prototype sont :

- le micro PC/IBM
- le SGDB DBASE2

#### 3.3.1 Micro-ordinateur

Au départ nous nous sommes fixé comme objectif pour notre travail, la réalisation d'un prototype fonctionnel dans le contexte micro-ordinateur. Nous connaissions les principaux avantages et inconvénients du micro et nous pouvons justifier ce choix selon deux points de vue.

##### 3.3.1.1. Avantages et limites

La manipulation d'un micro-ordinateur est relativement aisée et s'acquiert rapidement.

Dans le cadre de notre mémoire, c'est l'avantage essentiel que nous voyons au micro-ordinateur par rapport au gros ordinateur. De même les logiciels développés pour les micros présentent des facilités d'utilisation appréciables; les interfaces-utilisateurs sont plus agréables et les connexions plus directes.

Toutefois le micro-ordinateur a des **limites** intrinsèques.

Nous en relevons essentiellement trois.

- Le volume de l'espace de stockage des informations est réduit, or les documents juridiques que l'on souhaite enregistrer sont généralement des textes nombreux et assez longs.
- La vitesse d'accès aux fichiers et de transmission sur les lignes n'est pas très élevée et induit un temps de réponse qui peut paraître important.
- Le micro est un matériel mono-utilisateur qui ne permet donc pas de partager un même ensemble de données entre plusieurs personnes en même temps.

##### 3.3.1.2. Justification du choix

La justification du choix du micro-ordinateur pour la réalisation du prototype comprend d'une part des éléments extrinsèques

et d'autre part des éléments intrinsèques au micro.

- a) D'un point de vue extrinsèque, nous avons choisi de travailler sur un micro parce que nous n'avions manipulé, au cours de nos études, que du matériel plus gros (PDP 11/45 - DEC 20). Dans le contexte du mémoire nous avons l'opportunité de nous familiariser avec un autre type de matériel, d'en exploiter les possibilités et ainsi de compléter notre formation. C'est pourquoi nous avons préféré utiliser un des micros disponibles à l'Institut.
- b) D'un point de vue intrinsèque, nous tenons à souligner que le prototype à réaliser est un prototype fonctionnel et non un prototype d'exploitation. Notre but était de montrer ce que peut apporter un système documentaire dans le domaine juridique défini et non de fournir un outil d'exploitation.

Pour un utilisateur non informaticien, il est important que le matériel proposé soit facilement et rapidement manipulable. Le choix du micro semble donc indiqué pour réaliser un outil agréable à utiliser et qui devienne familier aux juristes. En effet la connexion au système et ensuite au logiciel DBASE2, support de l'application, se fait par quelques commandes aisées. D'autre part les limites du micro n'entravent pas notre travail puisque la réalisation d'un prototype fonctionnel nous permet d'écarter toute préoccupation de performance. Ainsi il n'est pas nécessaire de disposer d'un grand volume de données pour montrer à l'utilisateur les fonctionnalités du système; notre BD reprend un nombre minimal de jugements. De même la lenteur des accès et la longueur des temps de réponse induits sont acceptables. Et enfin le problème de la partageabilité d'une BD ne se pose pas dans ce contexte.

### 3.3.2. DBASE2

#### 3.3.2.1. Particularités de DBASE2 (5)

DBASE2 est un système de gestion de BD relationnelles qui permet une manipulation assez aisée de petites BD.

Détaillons ses trois principaux composants en les rapprochant de la découpe du chapitre II.

#### Structure des données

. Le DDL permet de décrire l'organisation d'une BD relationnelle.

Toute relation :

- est limitée à 65.500 enregistrements
- comprend au maximum 32 champs ( ou domaines) et 1000 caractères.

Tout champ est de longueur fixe - max 254 caractères

non répétitif

décrit par

- . un nom - max 10 caractères
- . type de données (caractère, numérique, booleen)
- . longueur du champ
- . nombre de positions décimales si type numérique.

Aucun champ ne constitue un identifiant privilégié de la relation.

. DBASE2 donne la possibilité d'organiser les lignes d'une relation dans un ordre différent de celui d'entrée:

- par un tri sur un champ = clé de tri (SORT);  
le tri ne rend pas plus rapide la recherche dans le fichier car elle se fait séquentiellement.
- par une indexation de la relation sur un champ ou une combinaison de champs = clé d'index (INDEXING);  
un fichier index avec la clé est créé.

Toute relation peut être indexée sur différents champs et plusieurs fichiers "index" sont alors conservés.

La recherche ou l'insertion d'une ligne dans une relation indexée devient plus rapide.

Langage de manipulation (DML et QUERY)

- C'est un langage procédural qui recouvre à la fois des commandes de programmation et des commandes de manipulation de données.

Il comprend les quatre structures fondamentales de programmation : séquence

condition

répétition

appel de procédure .

Les manipulations de données sont :

la recherche de données avec critères combinés

les entrées/sorties de données

la modification de données d'un fichier

le calcul automatique de sommation de champs .

Toutes ces instructions peuvent être rassemblées dans un fichier de commandes pour constituer un programme d'application.

exemple de construction : DO WHILE. NOT. EOF

LOCATE FOR VILLE-JUGEMENT=NAMUR

DISPLAY TEXTE-JUGEMENT

CONTINUE

ENDDO.

Le système ne permet que de travailler sur deux relations courantes, une primaire (PRIMARY) et une secondaire (SECONDARY).

Ceci constitue une limite assez contraignante pour la programmation et oblige à de nombreuses ouvertures et fermetures de fichiers de données lorsque les informations nécessaires à un moment sont éclatées sur plus de deux relations. Ces opérations d'ouverture , fermeture et repositionnement entravent énormément la vitesse d'exécution d'un programme. C'est pour cette raison que certaines fonctions de recherche prennent tant de temps.

- C'est un langage interprété. Il n'y a pas de déclaration de variables; elles sont créées lorsqu'on les rencontre et disparaissent à la sortie de DBASE2.

Un sauvetage dans un fichier mémoire est cependant possible et ainsi certaines variables sont gardées d'une session à l'autre.

Son caractère d'interactivité est très intéressant; il permet de tester séparément une ou plusieurs instructions avant de les introduire dans un fichier de commandes.

Toute construction qui peut se trouver dans un programme, peut aussi être exécutée pas à pas à l'écran.

- Le SGBD intègre trois outils qui sont une gestion d'écran, une gestion de BD, une gestion de rapports.

Pour chacun d'eux, il fournit des fonctions de haut niveau facilement utilisables mais ils ne sont toutefois pas très performants (on ne dispose pas, par exemple, d'un système multi-fenêtres dans la gestion d'écran).

#### Gestion de la BD (utilitaires)

Le SGBD ne prévoit aucun système de sauvetage et reprise (RECOVERY) en cas de panne; la sécurité n'étant pas assurée, il est important de prendre soi-même le soin de garder une copie sur disquette de son travail.

Par contre le système offre l'avantage de pouvoir gérer des fichiers en interface c'est à dire en provenance ou à destination de programmes écrits dans un autre langage de programmation.

Le système se caractérise également par une indépendance entre les fichiers de données et les programmes d'application et permet ainsi de modifier la structure de la BD en cours de vie du projet, sans affecter les programmes.

### 3.3.2.2. Justification du choix de DBASE2 pour le prototype

L'approche relationnelle est assez proche du schéma Entité/ Association et sa souplesse favorise l'adaptabilité du produit aux changements dans la structure des informations car on ignore les accès nécessaires aux applications.

Le langage de manipulation a des caractéristiques telles que l'interactivité, les fonctions de haut niveau, les facilités de travail sur les fichiers qui accélèrent la mise-au-point des programmes.

La gestion d'écran et de rapports permettait également de réaliser l'interface dans l'optique "user-friendly" que nous nous sommes fixée.

## 3.4. Spécification générale de solution

### 3.4.1. Fonctions

Pour couvrir l'aspect documentaire, le prototype doit fournir à l'utilisateur des fonctions d'interrogation et de mise à jour. Au niveau de la mise à jour, nous n'avons implémenté aucune fonction de suppression ou modification des données mais nous nous sommes limitées à développer les fonctions d'insertion d'un élément à la BD, à la fois par manque de temps et parce qu'elles nous semblaient les plus importantes.

#### 3.4.1.1. Fonctions d'ajout

- ajout d'un jugement :
  - renseignements généraux
  - personnes impliquées
  - compositions
  - texte
- ajout d'un fait ou standard :
  - renseignements : nroclasse, polarisation
  - textes affirmé et nié
  - descripteurs

- ajout d'une classe :  
libellé de la classe
- mise à jour des outils documentaires

#### remarques

- Les fonctions doivent vérifier que les mises à jour se font dans le respect des contraintes dégagées du schéma conceptuel.
- Les fonctions doivent pouvoir traiter des informations structurées et non structurées.

ex. Une des fonctions est un "mini-éditeur de texte".

#### 3.4.1.2. Fonctions d'interrogation

Elles fournissent deux types d'interrogation.

##### 3.4.1.2.1. Interrogation par listes prédéfinies

Ces fonctions donnent la liste de certains éléments de la BD, sans que l'utilisateur ne donne aucun renseignement au système :

- liste des jugements existants de la BD
- liste des faits-et-standards
- liste des classes
- liste des faits-et-standards d'une classe .

##### 3.4.1.2.2. Interrogation sélective

L'utilisateur fournit au système les critères auxquels devront répondre les informations à sélectionner.

Les fonctions sont organisées en niveaux et à chacun de ceux-ci, l'utilisateur donne de nouveaux renseignements qui affinent son critère d'interrogation.

L'interrogation sélective porte sur les données structurées et sur les données non structurées.

#### Sur les données structurées :

- liste des jugements ayant une caractéristique particulière :

nro, ville, date, type, résultat, personne-impliquée.

- renseignements sur un jugement particulier :  
courte description, personnes impliquées, type, compositions, jugements associés.
  - renseignements sur un fait ou standard particulier, c'est à dire  
intervenant dans une composition comme antécédent et/ou conséquent ;  
intervention dans un groupe de faits-et-standards.
- Cette interrogation exclut la combinaison de critères..

Sur les données non structurées :

- visualisation de tout ou partie du lexique
- recherche documentaire portant sur les faits-et-standards
- recherche documentaire portant sur les faits-et-standards et donnant accès aux jugements qu'ils indexent.

3.4.2. Interactivité

Nous avons décidé que c'était le système qui guidait le dialogue avec l'utilisateur, par l'utilisation des techniques de menu et question/réponse.

Lesdescription et justification de ce choix sont faites au chapitre suivant (cfr 4.4 ).

3.4.3 Système documentaire

Nous avons choisi une solution assez classique pour couvrir la recherche documentaire portant sur les faits-et-standards.

Cela signifie une indexation manuelle,

un lexique ouvert

un thésaurus reprenant deux types de relation sémantique

une interrogation par équation logique combinant les descripteurs .

Le chapitre suivant détaille chacun de ces choix à la section 2 ( cfr 4.2 ).

#### 4. SPECIFICATIONS FONCTIONNELLES DU PROTOTYPE

Les spécifications fonctionnelles portent sur trois points principaux :

- le modèle de la structure des informations
- le système de recherche documentaire
- les fonctions du système
- l'interface avec l'utilisateur .

##### 4.1. Le modèle de la structuration des informations

Nous rappelons simplement que nous avons pris comme hypothèse, le modèle développé dans le mémoire de Messieurs Hick et Tirtiaux (1) dont le schéma conceptuel se trouve en annexe 1.

##### 4.2. Le système de recherche documentaire

Il porte sur les faits-et-standards; leurs textes sont relativement courts et nous avons fait les choix suivants.

###### . Indexation

Nous indexons manuellement les textes des faits et des standards; lors de l'introduction de nouveaux faits ou standards, on saisit séparément leurs descripteurs.

###### . Lexique

Le lexique comprend tous les descripteurs qui indexent des faits et standards, sans aucune restriction sur leur nature; ils peuvent être des mots ou des expressions. Le lexique est ouvert c'est-à-dire étendu à chaque chargement de faits et standards, indexés par de nouveaux descripteurs.

###### . Thésaurus

Deux relations sémantiques sont définies.

La synonymie qui relie un élément du lexique avec une ou plusieurs forme(s) à tout moment équivalente(s) à celui-ci est donc

prise en compte automatiquement dans la recherche.

ex. : débiteur-principal et débiteur principal.

L'analogie relie différents descripteurs du lexique ayant des significations assez proches. Le choix de prise en compte lors de la recherche est laissé à l'utilisateur.

ex. : changement et novation et transformation.

#### . Niveau d'interrogation

Nous nous limitons à la recherche primaire sur base d'un fichier inverse et éliminons un niveau supplémentaire de recherche sur texte car ceux-ci sont trop courts.

#### . Domaines d'interrogation

La recherche documentaire peut se faire sur :

- les faits
- les standards
- les faits et standards confondus.

#### . Fonctions

La recherche documentaire a deux issues possibles :

- soit elle fournit les faits et/ou standards répondant à la requête; on peut visualiser le texte de ceux-ci;
- soit elle fournit les faits et standards solutions ainsi que les jugements dans lesquels ils interviennent; pour ces jugements, on peut visualiser une courte description (renseignements généraux et personnes impliquées) et/ou le texte du jugement et/ou les compositions.

#### . Gestion de la liste de recherche

Une question posée au système est une équation logique reliant des descripteurs. Elle est composée d'une suite de lignes qui sont des couples (opérateur logique / descripteur).

Pour chaque ligne,

on saisit un descripteur (tronqué ou non)

un opérateur logique ET/OU/SAUF reliant le descripteur à l'étape précédente;

on affecte un numéro d'étape;

on donne un résultat qui est le nombre d'éléments du domaine qui vérifient l'équation logique introduite jusque-là.

Les seules opérations possibles sur la liste de recherche établie sont - affiner la question c'est-à-dire lui ajouter des lignes  
- la reprendre à une étape supérieure.

#### . Troncature

Chaque fois que l'utilisateur doit donner au système un descripteur, il peut employer la troncature.

Elle est formalisée par un point en fin de la partie préfixe. Le système prendra en compte toutes les valeurs commençant par ce préfixe.

### 4.3. Les fonctions du système

#### 4.3.1. Au niveau de l'interrogation

Les fonctions se répartissent en deux groupes :

- les fonctions qui portent sur les données structurées c'est-à-dire sur la description au sens large du jugement
- les fonctions qui concernent les données non structurées c'est-à-dire les faits-et-standards.

##### 4.3.1.1. Interrogation sur les données structurées

On distingue ici deux grands types d'opérations :

- interrogation par liste prédéfinie,
- interrogation sélective.

##### 4.3.1.1.1. Listes prédéfinies

Les quatre possibilités sont les suivantes :

- liste des jugements existants

Pour chaque jugement, visualisation d'une courte description :

numéro du jugement dans la BD

ville où le jugement a été rendu

date du jugement

type de juridiction

résultat du jugement

personnes impliquées : nom

rôle joué;

et du texte du jugement, si l'utilisateur le désire.

En fin de visualisation de tous les jugements de la BD, l'utilisateur peut demander à revoir un jugement particulier dont il donne le numéro.

- liste des faits et des standards existants

Cette fonction propose trois possibilités :

- . liste des faits : on visualise les textes de tous les faits à l'état affirmé,
- . liste des standards : on visualise les textes de tous les standards à l'état affirmé,
- . liste des faits-et-standards: même visualisation des textes, en commençant par les standards.

- liste des classes

Pour chaque classe, on visualise son libellé.

- liste des faits-et-standards d'une classe

Cette fonction . demande le nro de la classe dont on désire voir les éléments

- . visualise les textes des faits ou standards de la classe.

retour fonction : (pour les quatre possibilités) retour au menu-interrogation.

#### 4.3.1.1.2. Interrogation sélective

- liste des jugements possédant une caractéristique particulière saisie . de la caractéristique sur laquelle on veut interroger; l'utilisateur choisit entre :

nro de jugement  
ville  
date  
type de juridiction  
résultat  
personne impliquée

- . de la valeur de la caractéristique choisie (et contrôle de la validité de cette valeur si nécessaire).

résultat

Soit aucun jugement ne contient cette caractéristique avec la valeur donnée,  
soit visualisation d'une courte description (cfr supra) des jugements solutions et visualisation de leur texte, si désiré.

retour fonction

Soit réexécution de cette même fonction,  
soit retour au menu-interrogation.

remarque : Il n'est pas possible d'interroger en combinant de façon logique différents critères.

- renseignements sur un jugement particulier

saisie . du renseignement que l'on désire sur un jugement;  
l'utilisateur peut choisir entre :

- une courte description (ville, date, type, résultat)
- les personnes impliquées
- le type de juridiction
- les compositions du jugement
- le(s) jugement(s) associé(s)

remarque : Nous n'avons pas repris ici le texte du jugement car il est possible de l'obtenir par la fonction "liste des jugements possédant une caractéristique particulière" en donnant le numéro du jugement.

- . de l'identifiant du jugement, qui est son numéro ou si l'utilisateur ne le connaît pas, sa ville et sa date.

résultat

rappel de l'identifiant du jugement  
affichage du renseignement demandé .

remarque : Pour les compositions du jugement, nous donnons deux représentations :

- une forme schématique
- une forme textuelle .

retour fonction

Soit rester sur le même jugement et en demander d'autres renseignements,

soit réexécution de cette même fonction pour avoir des renseignements sur un autre jugement,

soit retour au menu - interrogation .

- renseignements sur un fait ou standard particulier

saisie . du choix entre

liste des compositions où un fait ou un standard intervient

liste des compositions où un fait ou un standard intervient comme antécédent

liste des compositions où un fait ou un standard intervient comme conséquent

liste des compositions où un groupe intervient

. du ou des nro(s) de faits et standards

résultat

Visualisation des compositions répondant au critère demandé.

retour fonction

Soit réexécution de cette même fonction,

soit retour menu-interrogation .

4.3.1.2. Recherche documentaire

4.3.1.2.1. Visualisation du lexique

saisie du choix entre

- visualisation de tous les mots du lexique

résultat : tous les mots du lexique par ordre alphabétique

- environnement alphabétique d'un mot  
saisie du mot tronqué ou non  
résultat : cinq mots précédents  
                  mot demandé (si tronqué, le premier répondant)  
                  cinq mots suivants
- mots commençant par une lettre donnée  
saisie de la lettre  
résultat : affichage des mots commençant par cette lettre
- analogues d'un mot  
saisie du mot  
résultat : rappel du mot  
                  tous ses analogues

#### retour fonction

Après chaque fonction, retour au menu des différentes possibilités de visualisation du lexique et/ou retour au menu-recherche documentaire.

#### 4.3.1.2.2. Recherche documentaire sur les faits-et-standards

- saisie . du domaine de recherche : - faits  
  - standards  
  - faits-et-standards
- . de l'équation de recherche, comme une suite de lignes composées d'un opérateur et d'un descripteur et affectation d'un numéro d'étape à chaque ligne.
- . du choix de la prise en compte ou non de l'analogie

#### résultat

- . rappel de l'équation de recherche et du nombre de solutions
- . affichage des opérations possibles sur la question :  
  - visualisation de la question et du nombre de solutions  
  - visualisation des faits et/ou standards solutions  
  - liste des numéros de faits et/ou standards solutions

- affiner la question (ajouter des opérateurs et descripteurs)
- reprendre la question à une étape

. saisie du choix

- . résultat - visualisation pour les deux premières possibilités
  - pour les deux dernières, on reprend la fonction générale au niveau de la saisie de l'équation de recherche mais en tenant compte de l'étape où l'on doit se trouver.

#### retour fonction

retour au menu de la recherche documentaire

#### 4.3.1.2.3. Recherche documentaire sur les jugements

Cette fonction est semblable à la précédente, si ce n'est

- que le domaine est automatiquement pris comme faits-et-standards
- que les résultats comprennent outre les faits-et-standards solutions, pour chacun d'eux les numéros des jugements dans lesquels ils interviennent.

#### 4.3.2. Au niveau de la mise à jour

Nous rappelons ici que nous appelons mise à jour, uniquement l'agrandissement de la BD.

##### 4.3.2.1. Mise à jour des jugements

Elle se fait en quatre étapes :

##### 4.3.2.1.1. Mise à jour des renseignements généraux

saisie de la ville  
 de la date  
 du type  
 du résultat  
 du ou des jugement(s) associé(s)

Durant la saisie, il est permis à l'utilisateur de changer un des renseignements qu'il vient d'entrer.

Il y a vérification automatique de la validité des valeurs introduites sur base des contraintes.

#### 4.3.2.1.2. Mise à jour des personnes impliquées

saisie du débiteur principal  
de la nouvelle forme  
de(s) caution(s) originaire(s)  
de(s) caution(s) nouvelle(s)  
de(s) créancier(s)

Pour chaque personne, on saisit :

le nom et prénom  
la date-début de son rôle  
la date-fin de son rôle.

L'utilisateur doit toujours entrer au moins

un débiteur principal  
une caution originaire  
un créancier.

Après une première saisie de toutes les personnes impliquées, l'utilisateur peut encore en ajouter une ou en modifier une déjà entrée.

#### 4.3.2.1.3. Mise à jour des compositions

Pour chaque composition

saisie . de l'état de la composition (affirmé ou nié)  
. des antécédents

Si l'antécédent est un fait ou un standard :  
numéro du fait ou du standard  
état (affirmé ou nié) du fait ou du standard  
pondération du fait ou du standard.

Si l'antécédent est une composition :  
cfr saisie d'une composition

. du conséquent

Idem que pour les antécédents.

. effet de la composition (libération/maintien)

#### 4.3.2.1.4. Mise à jour du texte du jugement

L'utilisateur entre le texte du jugement dans une fenêtre qui apparaît sur l'écran et a la possibilité de faire voyager le curseur dans celle-ci.

La seule contrainte qui lui est imposée est de taper le mot " FIN " comme dernière ligne et de sortir de la fenêtre quand il a terminé.

Le texte peut être de longueur variable.

retour de la fonction mise-à-jour-jugement

Soit réexécution de la même fonction,  
soit retour au menu-mise-à-jour.

#### 4.3.2.2. Mise à jour des faits-et-standards

saisie du numéro de la classe à laquelle appartient le fait ou le standard;  
de la polarisation,  
des textes affirmé et nié (dans les mêmes conditions que pour le texte d'un jugement);  
des descripteurs du fait ou du standard.

retour fonction

Soit réexécution de cette même fonction,  
soit retour menu-mise-à-jour.

#### 4.3.2.3. Mise à jour des classes

saisie du libellé de la classe (cfr pour le texte du jugement mais dans une zone fixe).

retour fonction

Soit réexécution de cette même fonction,  
soit retour menu-mise-à-jour.

#### 4.3.2.4. Mise à jour des relations dans le lexique

##### 4.3.2.4.1. Mise à jour des relations de synonymie

saisie . du descripteur du lexique  
    . des synonymes que l'on veut définir pour ce  
      descripteur

##### retour fonction

Soit réexécution de la fonction mise à jour synonymie,  
soit retour menu-mise-à-jour-relations,  
soit retour menu-mise-à-jour.

##### 4.3.2.4.2. Mise à jour des relations d'analogie

saisie . du descripteur du lexique  
    . d'autres descripteurs du lexique que l'on veut  
      mettre en relation avec le premier

remarque : la relation est symétrique.

##### retour fonction

Soit réexécution de la fonction mise à jour analogie,  
soit retour menu-mise-à-jour-relations,  
soit retour menu-mise-à-jour.

#### 4.4. L'interface avec l'utilisateur

Dans la réalisation d'un système d'aide à la décision (SIAD), l'interface avec l'utilisateur est une des parties les plus importantes.

En effet du point de vue de l'utilisateur, l'interface est le système. L'utilisation d'un SIAD dépendra donc de la facilité du dialogue fourni par l'interface.

Dans cette partie, nous allons analyser plus en détail, les différentes techniques et alternatives qui existent pour la réalisation d'un interface et ensuite nous expliquerons comment nous avons réalisé l'interface du prototype.

Cette partie est basée sur l'ouvrage (6).

##### 4.4.1. Principes et outils pour la réalisation d'un interface

###### 4.4.1.1. Représentation de l'interface utilisateur

L'information qui est présentée à un terminal, passe par l'interface. En fonction des informations qu'il voit, l'utilisateur conçoit un plan d'action. Ce plan doit être formulable en actions acceptables par l'interface.

On peut représenter l'interface utilisateur par la suite d'actions suivantes :

Utilisateur : - interprète les données affichées  
 - formule les changements qu'il veut y apporter (plan d'actions)  
 - "traduit" son plan d'action, en actions acceptées par le système.

Ordinateur : - interprétation des actions de l'utilisateur  
 - accès à des données, si c'est nécessaire  
 - calcul des valeurs demandées par les actions de l'utilisateur  
 - "traduction" des données dans une forme compréhensible par l'utilisateur.

Ensuite, on recommence la même séquence d'actions.

De cette représentation de l'interface, nous pouvons tirer deux conclusions :

1. L'interface gère deux langages différents :

- un langage de présentation : facile à interpréter par l'utilisateur, dans lequel le système fournit ses informations à l'utilisateur.
- un langage d'action : aussi simple que possible dans lequel l'utilisateur donne ses actions au système.

remarque : Ces deux langages seront très liés dans les dialogues par menu. Car les commandes du langage d'actions sont des éléments du menu, c'est-à-dire du langage de présentation.

2. Dans les interactions utilisateur-système, il faut définir ce que l'utilisateur voit, doit savoir, et peut faire.

- Voir : Quel contexte doit-il voir quand il interagit avec le système ? C'est lié au langage de présentation.
- Savoir : Qu'est-ce que l'utilisateur doit savoir pour l'interpréter ? Il faut définir la base de connaissance de l'utilisateur.
- Faire : Qu'est-ce que l'utilisateur peut faire avec le système pour l'utiliser ? C'est lié au langage d'action.

La réponse à ces questions peut fournir un point de départ pour la conception de l'interface et influencer sa qualité.

#### 4.4.1.2. Principes pour la réalisation d'un interface-utilisateur

##### 4.4.1.2.1. La présentation doit guider l'utilisateur

A tout moment la présentation (ce qui se trouve sur l'écran) doit établir un contexte explicite pour l'action de l'utilisateur. Une bonne conception permettra les actions qu'un utilisateur veut prendre sur une représentation donnée et offrira des indications pour expliquer comment évoquer ces actions.

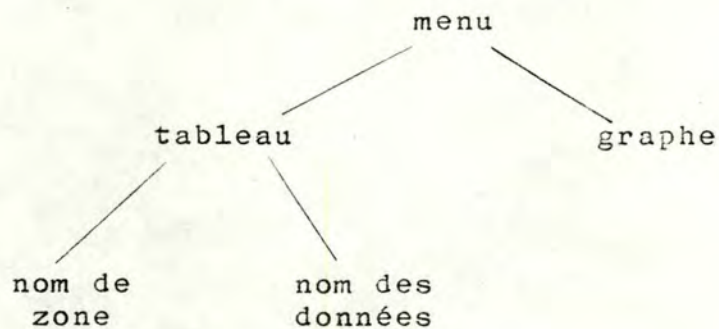
#### 4.4.1.2.2. Les représentations fournissent un "outil" pour la conception

Le processus de décision étant non structuré, le SIAD devra fournir plusieurs représentations pour les mêmes données. Chacune de ces représentations apportera des informations différentes.

#### 4.4.1.2.3. Le système fournit un canevas explicite pour l'utilisation de représentations

Un canevas est ce que l'utilisateur doit savoir sur le système de façon à contrôler l'interaction avec celui-ci. Un canevas peut être défini à différents niveaux de détails.

ex. :



A chaque niveau, l'utilisateur devra utiliser la même procédure pour sélectionner, le niveau suivant.

ex. : Après avoir sélectionné un tableau, l'utilisateur doit toujours donner le nom de la zone et un nom de donnée.

#### 4.4.1.3. Langages d'interface

##### 4.4.1.3.1. Différentes techniques

##### 4.4.1.3.1.1. Interface question/réponse

Le SIAD pose une question à l'utilisateur. Celui-ci y répond et ainsi de suite jusqu'à ce que le SIAD fournisse la réponse adéquate.

L'interface question/réponse peut être assez proche du langage naturel.

#### 4.4.1.3.1.2. Interface avec langage de commandes

L'utilisateur a à sa disposition un ensemble de commandes qui invoquent directement les fonctions du SIAD.

#### 4.4.1.3.1.3. Interface avec menu

Le menu permet à l'utilisateur de sélectionner une des alternatives qu'il propose.

La sélection d'une alternative peut se faire de différentes façons soit en tapant un mot-clé, soit en la pointant (dans le cas d'écran tactile), ...

Dans la plupart des applications, on a des menus "en cascade". Les menus sont alors structurés hiérarchiquement, ou groupés par caractéristiques.

#### 4.4.1.3.1.4. Interface avec formats d'entrée/sortie

Ce type d'interface fournit des formats d'entrées dans lesquels l'utilisateur tape ses commandes ou ses données et des formats de sortie dans lesquels le SIAD met ses réponses.

Après avoir vu une structure de sortie, l'utilisateur peut remplir une structure d'entrée pour continuer le dialogue.

Il est possible de combiner ces différentes techniques dans un même interface.

ex. : utilisation de menus pour sélectionner des commandes dans un interface avec langage de commande.

#### 4.4.1.3.2. Différents types de dialogue

##### 4.4.1.3.2.1. Dialogue conduit par le système

Ce type de dialogue utilise souvent les techniques suivantes : menus, question/réponse, formats d'entrée et de sortie.

##### 4.4.1.3.2.2. Dialogue libre

Ce type de dialogue utilise la technique de langage de commandes, mais aussi les langages d'interrogation de BD, ou le langage naturel.

#### 4.4.1.3.2.3. Dialogue orienté-objet

Dialogue avec manipulation d'objets directe.

#### 4.4.1.4. Quelques mesures de la facilité d'utilisation de l'interface

##### 4.4.1.4.1. Facilité d'utilisation

La facilité d'utilisation se mesure dans la relation personnelle utilisateur-terminal, lors de la réalisation d'une tâche. Elle se rapporte à la qualité de l'interaction qui prend place. Elle est liée aux caractéristiques physiques du terminal et au processus cognitif de l'utilisateur.

##### 4.4.1.4.2. Mesures de la facilité d'utilisation

Ces mesures sont des mesures de performances.

Elles ne peuvent donc être "calculées" que lors de l'utilisation du système.

En voici quelques-unes parmi les plus courantes :

- temps de formation des utilisateurs potentiels
- temps pour qu'un utilisateur exécute les actions "automatiquement"
- temps nécessaire pour se réhabituer au système, quand on ne l'a plus utilisé pendant une certaine période
- types et taux d'erreurs
- attitudes qui suscitent le désir d'utiliser le SIAD.

Ces mesures sont reprises de l'ouvrage (6).

#### 4.4.2. Réalisation de l'interface utilisateur pour notre maquette

Pour cette partie, nous conseillons au lecteur de se rapporter à l'annexe 5 qui reprend un exemple d'utilisation de la maquette.

#### 4.4.2.1. Choix d'un dialogue conduit par le système

##### 4.4.2.1.1. Justification de ce choix

Nous avons opté pour la solution d'un dialogue conduit par le système à la fois pour l'interrogation et la mise à jour.

Il y a principalement deux raisons à ce choix.

La première est le profil de l'utilisateur. Nous avons considéré celui-ci comme étant un juriste qui n'utiliserait le système qu'occasionnellement et pour le problème précis qui l'intéresse.

La seconde raison est basée sur les opérations que l'on désire effectuer sur la BD.

Au niveau de l'interrogation, comme de la mise à jour, nous avons assez facilement dégagé, avec l'utilisateur, les principales opérations qu'il désirait. Nous avons ensuite construit des fonctions prédéfinies pour les réaliser. Il est certain que de nombreuses fonctions pourraient encore être ajoutées au système. Aussi un dialogue conduit par le système ne supprime-t-il pas trop des possibilités à l'utilisateur.

##### 4.4.2.1.2. Techniques d'interface mises en oeuvre

Nous avons utilisé trois techniques : les menus, les questions-réponses et les formats prédéfinis avec quelques différences entre les parties interrogation et mise à jour.

4.4.2.1.2.1. Au niveau de l'interrogation, la technique de menus en cascade est utilisée pour sélectionner une fonction particulière tandis que par la technique question/réponse, le système demande à l'utilisateur les arguments de la fonction sélectionnée. L'affichage des informations, résultat de la fonction, se fait selon un format prédéfini.

4.4.2.1.2.2. Au niveau de la mise à jour, la technique de menus permet de la même manière de sélectionner une mise à jour déterminée et c'est la technique de formats prédéfinis, qui est utilisée pour saisir les informations entrées par l'utilisateur.

Après chaque fonction, par un système de question/réponse, l'utilisateur peut soit rappeler la même fonction avec des arguments différents, soit revenir au menu précédent.

#### 4.4.2.2. Mise en oeuvre des principes pour la réalisation d'un interface-utilisateur

Pour chacun des principes vu précédemment (cfr point 1), nous allons examiner comment ils ont été appliqués.

4.4.2.2.1. La présentation doit guider l'utilisateur et fournir un contexte explicite pour l'interaction.

Les solutions adoptées diffèrent quelque peu entre les fonctions d'interrogation et de mise à jour.

- Au niveau de l'interrogation,

l'utilisateur voit toujours ce qu'il peut faire à un moment donné, mais il n'a pas toujours sous les yeux le rappel de la démarche qu'il a suivie pour sélectionner la fonction. En ce sens, le contexte n'est pas toujours explicite.

Il est intéressant de noter ici qu'un système qui aurait offert une gestion multi-fenêtres nous aurait permis d'atteindre facilement l'objectif visé.

En effet une fenêtre pouvait garder en permanence l'identification de la fonction courante ainsi que ses arguments, alors qu'une autre fenêtre nous servait à faire dérouler les informations s'y rapportant.

Une telle présentation réalisée avec DBASE.2 est fort compliquée et nous l'avons rejetée par manque de temps.

Cependant nous avons choisi à la fin de chaque fonction, de retourner automatiquement au menu précédent afin de rappeler à chaque étape les opérations qui sont possibles.

- Au niveau de la mise à jour,

le contexte est toujours explicite. L'utilisateur sait à tout moment ce qu'il a déjà fait et ce qui lui reste à faire.

Nous avons pris cette option car la mise à jour influence directement l'état de la BD; elle est donc dans ce sens plus dangereuse. Aussi plus de précautions sont-elles nécessaires.

4.4.2.2.2. - Les représentations fournissent un outil pour la conception; plusieurs représentations possibles pour les mêmes informations.

Pour certaines données, nous avons réalisé plusieurs représentations, dans le but de fournir des informations supplémentaires.  
ex. : les compositions d'un jugement peuvent être obtenues sous deux formes.

- Le texte des standards et des faits qui les constituent est affiché; et les différentes compositions défilent les unes après les autres à l'écran.

Cette représentation met en évidence l'impact d'un élément (fait ou standard) dans une composition.

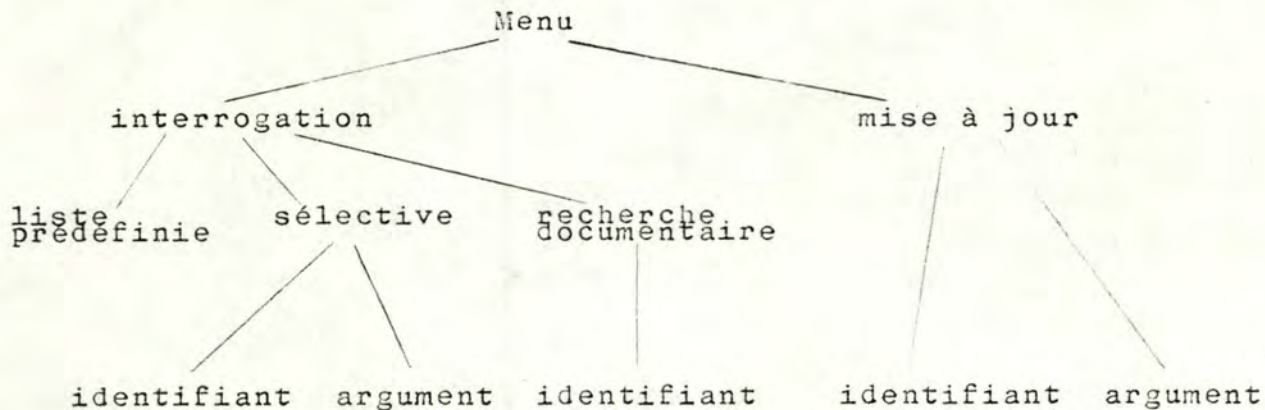
- Les compositions sont représentées sous une forme schématique, qui permet d'avoir toutes les compositions d'un jugement sur un seul écran.

Cette représentation propose donc une vue plus globale du jugement.

4.4.2.2.3. - Le système fournit un canevas explicite pour l'utilisation des représentations.

Nous avons essayé de normaliser les différentes étapes qui mènent à la sélection d'une fonction et à son exécution.

Cela apparaît dans le schéma suivant :



#### 4.4.2.3. Mesures de la facilité d'utilisation

A titre expérimental, nous avons essayé de "mesurer", la facilité d'utilisation de notre système sur un échantillon réduit; composé de deux types d'utilisateur :

- des juristes ayant une connaissance du problème de droit qui concerne les jugements repris dans la base, mais peu habitués à se servir d'un SIAD,
- des utilisateurs informaticiens (copains de cours) ayant peu de connaissance dans le domaine juridique, mais routiniers de l'informatique.

Nous pouvons passer en revue quelques-unes des mesures de la facilité d'utilisation :

- temps de formation : nul dans les deux cas
- temps pour se réhabituer au système : nul dans les deux cas
- temps nécessaire pour exécuter les actions automatiquement : après quelques sessions à l'écran, les utilisateurs informaticiens avaient acquis l'habitude des actions à faire. Mais il faudrait probablement plus de temps pour les utilisateurs juristes.
- types d'erreur : les utilisateurs fournissent des informations qui ne sont pas valides.
  - ex. : - une information qui n'est pas acceptable, pour le format d'entrée prédéfini;
  - une information qui ne constitue pas une alternative du menu proposé.

Ces erreurs sont probablement dues au fait que les utilisateurs ne lisent pas assez attentivement les messages et explications données.

Le système paraît donc fournir toutes les explications nécessaires à son fonctionnement.

Peut-être celles-ci deviendront-elles un peu "lourdes" pour les utilisateurs expérimentés et faudrait-il prévoir une alternative plus rapide (par ex. langage de commandes).

## 5. IMPLANTATION DE PROTOTYPE

Pour réaliser le prototype dans l'environnement retenu, nous avons dû transformer le schéma conceptuel élaboré dans le mémoire de Messieurs HICK et TIRTIAUX (1) en un schéma relationnel DBASE2 et compléter l'entité "fait-et-standard" pour prendre en compte les besoins de la recherche documentaire. Dans ce chapitre nous voyons également la manière dont nous avons traité du texte à l'aide de DBASE2.

Au niveau des traitements, nous donnons une justification de la structure des programmes..

### 5.1. Règles de transformation du schéma E/A en un schéma relationnel (Annexe 3)

Comme nous l'avons dit au chapitre 2 (2.1.1.1. approche relationnelle), toutes les entités et associations du schéma conceptuel sont transformées en relations.

Les domaines de ces relations sont :

- pour une entité:

les attributs de l'entité, présents dans le schéma conceptuel et éventuellement, un attribut supplémentaire, identifiant interne, pour accéder plus rapidement à l'entité.

ex :

JUGEMENT
ville
date
résultat
type

devient JUGEMENT (nrojugement, ville, date, résultat, type ).

- pour une association :

les attributs propres à l'association et les identifiants des entités reliées par l'association.

ex : L'association, sans attribut propre, "CASSATION/INSTANCE" du schéma conceptuel devient CASSINST (nrojugementinstance, ~~nrojugementcassation~~).

### Remarques

1. Lorsque les performances le justifiaient, certaines associations 1-N ont été remplacées par un champ supplémentaire dans une des entités reliées, reprenant l'identifiant de l'autre entité.

ex :

L'association "APPARTENANCE-CLASSE" n'a pas donné lieu à une relation équivalente en DBASE2 mais a été remplacée par le champ NROCLASSE dans la relation FSTD qui correspond à l'entité FAIT-et-STANDARD (NROFST, POLAR, NROCLASSE).

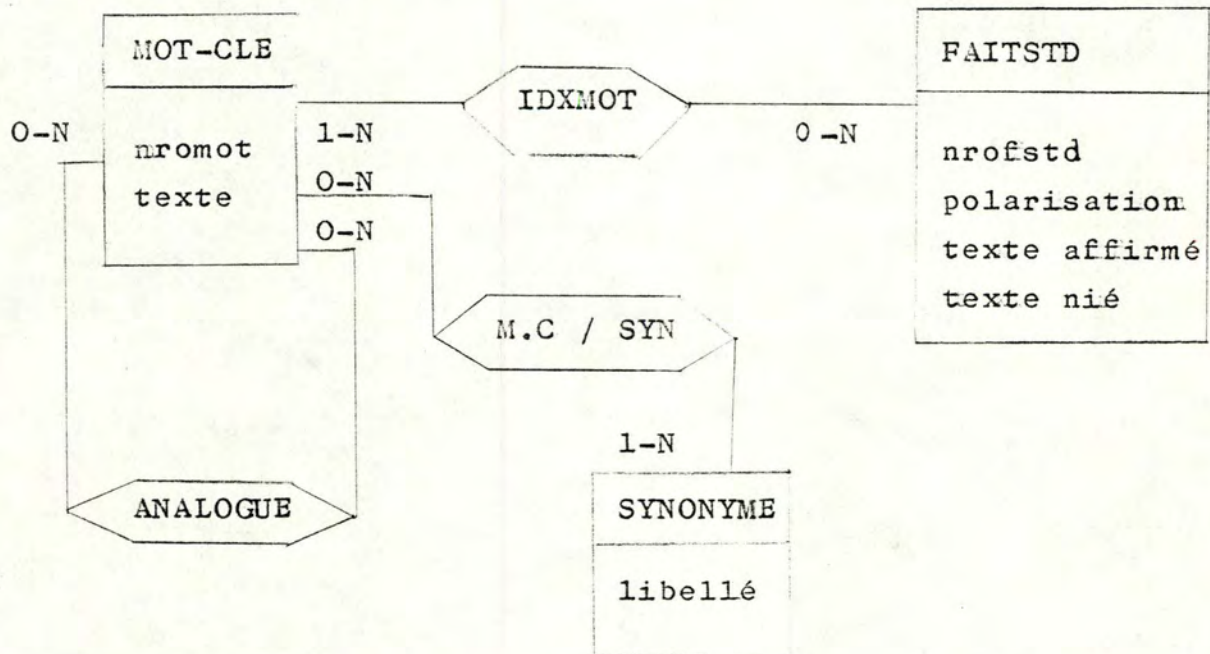
Il en va de même pour les associations "COMPO-INDIRECTE" et "COMPO-DEPENDANTE".

2. Dans le but de minimiser le nombre de relations, il y a eu agrégation de plusieurs associations en une relation générique dont un champ supplémentaire spécifie le rôle initial de l'association.  
C'est le cas pour la relation "PERSONNEIMPLIQUEE" qui recouvre les associations "CREANCIER", "DEBITEUR-PRINCIPAL", "CAUTION ORIGINAIRE", "CAUTION NOUVELLE", "NOUVELLE-FORME". Il en est de même pour les faits et standards antécédents ou conséquents d'une composition.
3. Les attributs dont le domaine de valeurs était binaire ont été introduits comme des champs de type booléen avec le nom qui correspond à une des deux valeurs du domaine. Ceci pour des facilités d'implémentation.

### 5.2. Implantation des faits-et-standards

Pour gérer les données non structurées représentant les faits-et-standards, nous avons étendu le schéma conceptuel en remplaçant l'entité "FAIT-et-STANDARD" par une structure de données qui intègre les éléments intervenant dans la recherche documentaire (cfr. 4.2).

### 5.2.1. Schéma conceptuel



### 5.2.2. Description sémantique du schéma conceptuel

#### 5.2.2.1. Les types d'entités

##### FAITSTD

Un faitstd est un élément qui peut intervenir dans la formulation d'un jugement - Cet élément peut être soit un fait, soit un standard.

"Nrofstd" est l'attribut identifiant du fait ou du standard. Sa valeur est attribuée séquentiellement par le système.

"Polarisation" donne la polarisation du fait ou du standard. Les valeurs possibles sont : POSITIF, NEGATIF, NEUTRE.

MOT-CLE

Un mot-clé est un terme qui indexe au moins un fait ou un standard (FAITSTD).

"nromot" est un attribut dont la valeur est attribuée séquentiellement par le système.

"texte" est le libellé du mot-clé.

SYNONYME

Un synonyme est un terme qui est identique sémantiquement à un mot clé au moins.

"libellé" est le libellé du synonyme

5.2.2.2. Les types d'associations

ANALOGUE (mot-clé, mot-clé) :

exprime la relation symétrique d'analogie entre deux mots-clés.

M.C/SYN ( mot-clé, synonyme):

exprime la relation du synonymie stricte qui existe entre un mot-clé et un synonyme.

IDXMOT ( mot-clé, faitstd):

exprime la relation d'indexation qui existe entre un mot-clé et un fait ou standard.

5.2.3. Transformation en un schéma relationnel

Les transformations se font selon les mêmes règles que celles exprimées à la section 1 de ce même chapitre.

Le schéma relationnel se trouve en annexe 3.

5.3. Traitement du Texte

DBASE2 ne permet pas de gérer des champs de longueur variable; or nous devons manipuler des textes pour les jugements et les faits-et-standards.

La solution technique que nous avons adoptée est de découper ces textes en lignes de longueur fixe, ce qui représente encore une

entité significative. Toutefois cette division est transparente à l'utilisateur; on lui offre une gestion pleine page partout où il manipule du texte c'est-à-dire qu'il le voit comme un autre champ.

Nous avons ainsi les deux relations suivantes :

LIGNEFSTD (nrofstd, état, nroligne, texte)

où "nrofstd" est le numéro du fait ou du standard

"état" donne l'état affirmé ou nié du texte

"nroligne" est le numéro de la ligne du texte

"texte" est le texte d'une ligne du texte du fait ou du standard, à l'un des états.

LIGNEJUG (nrojgement, nroligne, texte)

où "nrojgement" est le numéro du jugement

"nroligne" est le numéro de la ligne du texte du jugement

"texte" est le texte d'une ligne du texte du jugement.

#### 5.4. Architecture des Modules annexe 4

D'une part l'objectif était de réaliser rapidement un prototype fonctionnel.

Il n'était donc pas a priori aussi nécessaire que le système implémenté garantisse les qualités d'architecture (adaptabilité, clarté, portabilité,...) attendues d'une solution opérationnelle. D'autre part, même si l'objectif de réalisation rapide justifiait le choix de DBASE2 dont on a évoqué précédemment les facilités d'écriture, il n'offrait cependant pas toujours les performances souhaitées pour développer une application élaborée.

En effet un de ses principaux inconvénients est l'absence de procédure avec passage de paramètres. Tout morceau de code appelé dans un programme est défini lui-même comme un programme externe et tous les arguments que l'on veut lui fournir doivent être gérés comme variables globales. Cela alourdit la construction des programmes et les appels de procédure allongent les temps d'exécution.

On peut rappeler ici également le fait que DBASE2 ne travaille que sur deux relations courantes, ce qui nécessite des ouvertures et des fermetures de fichiers fréquentes et pèse encore sur le temps d'exécution.

Les facilités de gestion d'un dialogue à l'écran ne sont pas non plus très nombreuses. Nous en faisons une critique en 6. 1.3.2.

C'est dans ce contexte que nous avons fait le choix d'une solution d'implémentation qui calque exactement la structure des programmes sur celle des fonctions identifiées. Ainsi à chaque menu et à chaque fonction de l'architecture fonctionnelle du prototype (cfr 4. 3.) correspond un module DBASE2 tel qu'illustré dans le schéma repris en annexe 4.

Malgré cette approche garantissant une grande indépendance entre les modules, nous nous sommes efforcées de maintenir l'homogénéité dans le dialogue avec l'utilisateur, par l'adoption de standards pour la saisie des informations et l'affichage des résultats.

## 6. CONCLUSION DE L'IMPLANTATION DU PROTOTYPE

Dans ce chapitre, nous regarderons quelles critiques on peut formuler sur l'implantation du prototype et ce qu'il nous apporte pour la réalisation d'un système à grande échelle.

### 6.1. Critique

Nous allons faire une critique du prototype et de l'environnement dans lequel nous l'avons réalisé, en fonction des objectifs que nous lui avons assignés.

Les objectifs sont :

- être un outil de dialogue avec l'utilisateur
- être rapidement disponible
- être évolutif
- être "userfriendly"
- permettre de gérer à la fois des données structurées et non structurées.

#### 6.1.1. Outil de dialogue

Pour l'utilisateur, le prototype est une réponse à ses spécifications fonctionnelles. En effet, les juristes nous ont exposé leurs besoins et après quelques mises au point, nous avons entièrement réalisé le prototype sans les recon tacter. C'est donc pour eux la possibilité de voir si le système que nous leur proposons correspond à leurs spécifications et à leurs besoins peut-être non exprimés.

Les futurs utilisateurs d'un SIAD, ne sont pas toujours convaincus de l'aide que leur procurera l'outil.

Le prototype va leur donner la possibilité de se familiariser avec le système, de voir ce qu'il leur apporte.

Ils en testeront les différentes fonctions sur une BD minimale. Ils réagiront, sans doute, à l'utilisation du système et demanderont certaines modifications.

C'est donc sur base de leurs réactions que le prototype subira des extensions et des améliorations.

En permettant une vérification et un raffinement des spécifications fonctionnelles, ainsi qu'une expérimentation globale du système, le prototype nous paraît réaliser l'objectif d'être un outil de dialogue.

#### 6.1.2. Prototype rapidement disponible

Le prototype a été réalisé en quelques mois.

Mais il nous aurait été possible d'être plus rapides, si certains inconvénients de DBASE2 ne nous avaient pas ralenties dans notre travail.

Ces inconvénients sont les suivants :

6.1.2.1. Le langage d'interrogation de DBASE2 ne permet de manipuler qu'une relation à la fois; et les relations ne peuvent être combinées entre elles que deux à deux.

Si nous avions disposé d'un système qui nous apporte des primitives plus puissantes que celles-là, nous aurions implémenté plus rapidement le prototype.

6.1.2.2. DBASE2 n'offre aucune possibilité graphique.

Cet inconvénient ne nous a guère gênés étant donné la nature juridique du problème que nous traitions.

Dans un cas seulement, nous aurions voulu disposer de primitives graphiques : représentation schématique des compositions d'un jugement.

Cependant, pour d'autres types de problème, (ex. dans le secteur économique), elles s'avèrent nécessaires pour la réalisation d'un prototype.

6.1.2.3. Quand on travaille " en écran plein", DBASE2 ne permet pas de faire dérouler l'écran.

C'est assez contraignant, notamment pour la saisie des textes, qui remplissent souvent plus d'un écran.

Il a donc fallu gérer la liaison d'un écran à l'autre, de façon transparente pour l'utilisateur.

On peut également rappeler ici les facilités qu'un système multi-fenêtres nous aurait apportées. (cfr 4. 4.2.2.1.)

### 6.1.3. Extensibilité du prototype

On peut traiter ce point de deux façons : d'une part examiner le degré d'extensibilité de la structure des données et des programmes du prototype réalisé et d'autre part proposer des extensions DBASE2 pour faciliter la construction du SIAD.

#### 6.1.3.1. Niveau du prototype réalisé

La Base de données en elle-même est facilement extensible; des fonctions sont offertes pour ajouter des éléments de la structure de données.

Quant à cette structure, toute modification ne se fait pas aussi automatiquement. En effet même s'il est possible de changer la longueur d'un champ ou son type de données sans devoir modifier les programmes, on ne peut introduire, par exemple, une nouvelle classification des faits et standards c'est-à-dire modifier la structure sémantique des données sans répercuter ce changement dans les fonctions. Les programmes sont fortement liés à la structure DBASE2 des données et cela rend plus difficile une extension du prototype portant sur la structure des informations.

Toutefois il serait plus aisé d'ajouter une fonction à l'un quelconque des menus; il suffirait de connaître la représentation DBASE2 des données, d'implémenter le programme réalisant cette fonction et de la raccrocher au menu correspondant. Les différentes fonctions offertes sont relativement indépendantes; cela facilite donc leur implémentation.

#### 6.1.3.2. Niveau DBASE2

Les limites de DBASE2 nous amènent à suggérer un éventail d'outils de base qui s'avèreraient très utiles pour celui qui veut réaliser un SIAD.

Rappelons que la partie principale du code d'un SIAD concerne l'interface avec l'utilisateur et c'est à ce niveau que DBASE2 pourrait offrir plus de possibilités.

Ainsi un système de gestion de messages serait apprécié; on pourrait, par exemple, coder les différents messages d'erreurs, commentaires sur l'état de l'interrogation, certaines question/réponse standards,... donner leur localisation sur l'écran une seule fois et pour toute impression de message survenant dans un programme, faire appel à une procédure commune qui gère l'affichage des messages d'après leur code.

Un programme de gestion de menus aurait aussi été pour notre application un outil fort utilisé, puisque c'est la technique du dialogue que nous avons adoptée.

Un tel utilitaire permet pour chaque menu de définir les différentes options et pour chaque option la ou les procédure(s) à exécuter; cet utilitaire génère lui-même l'affichage sur écran du menu souhaité, saisit le choix de l'utilisateur et gère l'enchaînement correspondant des programmes réalisant ce choix. Un troisième outil serait un outil graphique pour restituer à l'écran soit, tout ou partie du schéma conceptuel des données, soit un schéma d'occurrences de ces données.

Dans notre cas, nous avons dû réaliser de toute pièce une présentation schématique des compositions d'un jugement.

Toutefois l'utilité des primitives graphiques s'avère plus grande pour des applications manipulant des données plus structurées que dans le domaine juridique.

Ce sont les principaux compléments que l'on pourrait ajouter à DBASE2 pour le rendre plus efficace dans la construction du SIAD.

#### 6.1.4. Gestion de données structurées et non structurées

Le prototype permet de traiter à la fois des données structurées et non structurées, car nous avons défini un modèle, le plus souple possible, pour les données non structurées (cfr 5.3) Ce modèle est transparent à l'utilisateur.

ex. L'utilisateur ne sait pas que le texte est considéré comme un ensemble d'un nombre variable de lignes.

### 6.1.5. " User-friendliness"

Nous avons fait le choix de créer un dialogue guidé par le système. Nous avons pu nous rendre compte qu'il était apprécié des quelques personnes qui ont utilisé le prototype (cfr. 4.4.)- L'interface paraît donc être assez "userfriendly" pour un utilisateur amateur.

Cependant le dialogue deviendra probablement fastidieux pour un utilisateur habitué.

Aussi serait-il préférable de fournir en option un langage de commandes, en plus du dialogue guidé par le système, afin de mieux rencontrer l'objectif de "user-friendliness".

#### Remarque

Il nous faut signaler ici, que DBASE2 offre de nombreux "gadgets" pour rendre le dialogue avec le terminal plus agréable

ex. impression contrastée, par rapport à l'écran, du champ à remplir par l'utilisateur.

### 6.2. Le prototype vis-à-vis du système à grande échelle

Il est intéressant de voir ce que peut nous apporter le prototype pour la réalisation d'un système à grande échelle.

Les caractéristiques d'un système à grande échelle sont les suivantes :

- il est multi-utilisateurs
- il travaille sur un grand ensemble de données
- il assure la protection des données
- il fournit des utilitaires pour améliorer les performances d'exploitation.

#### 6.2.1. Système multi-utilisateurs

Pour rendre le système multi-utilisateurs, il faudrait créer un réseau de micros avec un serveur unique qui centralise les données, et différents postes qui conservent leur autonomie de travail et leur puissance de traitement.

De plus le logiciel DBASE2 peut s'utiliser sur un tel réseau.

### 6.2.2. Volume de données

L'emploi d'un micro ne pose aucun problème pour traiter de grandes quantités de données car les micros peuvent disposer actuellement d'une capacité de stockage importante (par exemple sur disque dur).

Le logiciel DBASE2 est plus contraignant, car il limite le nombre de fichiers, c'est à dire de relations, et le nombre d'articles par fichier.

On pourrait résoudre ce problème en créant plusieurs BD, chacune correspondant à une période fixe.

ex BD n° 1 : année 1980-81  
 BD n° 2 : année 1982-83 etc...

L'utilisateur aurait la possibilité de choisir la BD qu'il veut interroger. Mais il ne pourrait pas interroger plusieurs BD simultanément.

### 6.2.3. Protection des données

Le logiciel DBASE2 n'offre aucun utilitaire pour la protection des données, tel que gestion des points de reprise, ...

Il faudrait donc écrire ces utilitaires, mais cela ne paraît guère possible, étant donné le langage de programmation fourni par DBASE2.

L'utilisateur devrait donc lui-même faire des copies de ses fichiers, quand il le juge nécessaire. La gestion des profils pourrait peut-être l'y aider.

### 6.2.4. Les performances d'exploitation

Il semble difficile d'améliorer les performances d'exploitation notamment pour réduire le temps de réponse. La longueur de ces derniers proviennent du fait que DBASE2 ne permet pas de combiner plus de deux relations, le logiciel ne peut traiter que deux fichiers simultanément. C'est un des principaux inconvénients de DBASE2, et il nous paraît impossible de le contourner.

---

## DEUXIEME PARTIE

---

### LE PROTOTYPE DU TEST DE VALIDITE

---

#### 1. ENONCE DU PROBLEME

Comme nous l'avons dit dans l'introduction,, la prise en compte de l'aspect d'évaluation associé à l'élaboration d'un jugement, s'est limitée dans notre travail au problème de la cohérence logique externe.

Nous parlerons de la validité d'un jugement et de la manière dont il est possible de la tester.

##### 1.1. Qu'entendons-nous par validité?

On désire vérifier la validité formelle d'un jugement par rapport à des éléments de référence extérieurs, définis par des juristes. Ces éléments se présentent sous la forme d'une relation définie a priori entre deux faits pondérés ou deux standards pondérés. On les appelle des groupes de validité de faits pondérés ou de standards pondérés.

Ces groupes sont repris en annexe 4 du mémoire(1).

Différents types de relations peuvent être pris en compte; ce sont l'implication, l'équivalence et l'incompatibilité.

Par rapport à ces éléments de référence, on peut dire qu'un jugement serait valide si les faits et standards qui interviennent dans ses compositions ne sont pas incompatibles, d'un point de vue logique, avec les groupes de validité.

Exemple

D'une part un groupe de validité de standards pondérés établit l'incompatibilité entre le standard 5 de la renonciation du créancier et le standard 6 du terme extinctif du contrat de cautionnement.

Supposons d'autre part, que dans un jugement, on retrouve dans l'argumentation, le fait que le créancier a lui-même renoncé à la caution originaire en cherchant une nouvelle caution (standard 5) mais aussi le fait que lorsqu'un contrat est arrivé à son terme, la caution originaire est libérée (standard 6).

Alors l'application simultanée des standards 5 et 6 rendra le jugement invalide, puisque ces deux standards ne peuvent s'appliquer que de façon exclusive dans un même jugement.

Remarque

Il est intéressant de rappeler ici la différence qui existe entre la validité et la vérité (cfr (1) au point 4.2.2.1).

La différence entre la validité d'une expression et sa vérité se résume à l'implication suivante.

Si une expression est vraie, alors elle est valide.

Autrement dit la validité d'une expression est la condition nécessaire pour qu'elle soit vraie.

Il est démontré dans (1) que

( si vérité alors validité)<sub>1</sub> est équivalent à  
( si non validité alors non vérité)<sub>2</sub>

et que

( si vérité alors validité)<sub>1</sub> n'est pas équivalent à  
( si validité alors vérité)<sub>3</sub>

Cela signifie donc que si la vérité d'une expression implique sa validité ( )<sub>1</sub>, alors la non validité de l'expression implique sa non vérité ( )<sub>2</sub>, et réciproquement.

De plus l'implication de la validité par la vérité n'est pas équivalente à l'implication de la vérité par la validité ( )<sub>3</sub>. Dès lors, si on peut déterminer la validité d'une expression, on ne peut rien affirmer sur sa vérité; mais si on peut déterminer sa non validité on peut conclure à sa fausseté.

### 1.2. Comment réaliser un test de validité

La réalisation du test de validité d'un jugement repose sur trois hypothèses (cfr (1) au point 4.2.2.3).

- Il existe un accord parmi la communauté des juristes pour définir a priori "un moment de nécessité", pour un domaine juridique donné. Ce "moment de nécessité" correspond au respect de tous les groupes de validité de faits pondérés et de standards pondérés définis pour un domaine précis.
  - Il est possible d'exprimer les relations entre faits pondérés et entre standards pondérés, sous la forme d'équivalences, d'implications ou d'incompatibilités.
  - La définition et la mise à jour des groupes de validité de faits pondérés et de standards pondérés est un travail aisément réalisable par des juristes.
- Sa réalisation n'implique pas de très longues recherches rendant toute mise à jour tardive.

Pour des facilités de manipulation, nous transformerons les différentes relations reprises dans les groupes de validité, c'est-à-dire l'implication, l'équivalence et l'incompatibilité en incompatibilités de la façon suivante.

- 1) A implique B sera transformé en A est incompatible avec NON B
- 2) A équivalent à B sera transformé en A est incompatible avec NON B et B est incompatible avec NON A

Cela a été démontré en (1) au point 4.2.2.4.

Ainsi les groupes de validité sont tous ramenés à un ensemble d'incompatibilités entre deux faits pondérés ou deux standards pondérés.

La validité d'un ensemble de faits-et-standards pondérés sera alors vérifié par comparaison, comme suit.

Un ensemble de faits-et-standards pondérés sera considéré comme valide si pour chaque groupe de validité, soit il ne contient aucun membre du groupe soit il ne contient qu'un seul des deux membres.

Il sera considéré comme non valide lorsqu'il comprend les deux faits ou standards pondérés d'au moins un groupe de validité.

#### Remarque

Nous appelons standard, aussi bien la forme affirmée que la forme niée d'un standard.

Ainsi les standards A et NON A sont deux standards à part entière. Il en est de même pour les faits.

## 2. OUTILS EXISTANTS POUR REALISER CE TEST

Nous avons la possibilité à l'Institut, de réaliser ce test dans un langage traditionnel ( Pascal, Cobol,...) ou dans un langage dit non-procédural ( Prolog, Lisp, ).

Voyons brièvement ce qui a guidé notre choix.

### 2.1. Langages traditionnels

Ces langages manipulent un nombre d'objets assez variés : des variables simples, des tableaux, des sets, des records, des fichiers.

Et sur ces objets sont définis plusieurs primitives qui sont principalement des opérations arithmétiques et quelques opérateurs booléens.

On dispose également de structures de programmation prédéfinies : la séquence, la condition, la répétition, la boucle, la procédure ou sous-routine.

Ces outils de base sont utilisés pour écrire des programmes ayant des fonctions très précises, définies d'avance.

Les instructions d'un tel programme doivent figurer dans un ordre déterminé pour réaliser la fonction qui lui est attribuée et cet ordre a une grande importance.

De plus, on remarquera que les traitements sont séparés des données : certaines données sont extérieures aux programmes (fichiers) et dans un programme, on sépare également les déclarations de données et les instructions de manipulation. Or, pour réaliser le test de validité, nous n'avons pas besoin de structures élaborées de données et de programmation. Le test demande de faire des comparaisons, des opérations logiques et à ce niveau les langages traditionnels ne nous offrent pas beaucoup de primitives puissantes.

La programmation nécessaire serait importante.

## 2.2. Langages non-procéduraux

Dans les langages dits non-procéduraux, on a réduit fortement les types d'objets et les manipulations de base possibles sur ces objets.

On permet une plus grande liaison entre les données et les traitements qui sont souvent représentés par une structure commune.

Leur ordre d'apparition dans un programme a beaucoup moins d'importance que dans les langages traditionnels : les données et les traitements s'intercalent.

Ce sont plutôt des langages déclaratifs ou descriptifs.

Prolog est un langage non-procédural et on y représente les connaissances sous forme de faits (données) et de règles (traitements). Cela constitue la Base de Connaissances.

La représentation externe des connaissances relève de la logique des prédicats (ex. "Tous les oiseaux ont des ailes" est représenté par  $\forall x : \text{oiseau}(x) \Rightarrow \text{ailes}(x)$ ).

Prolog manipule tous les éléments comme une structure de données uniforme (TERME). Un programme qui est une suite de faits et de règles, est lui-même une structure de données. C'est un point fondamental de ce langage.

Le système Prolog comprend en outre un moteur d'inférence, c'est-à-dire un programme de démonstration automatique de formules logiques, par l'exécution d'une chaîne de déductions. Les traitements ne sont pas organisés d'avance, mais on leur fournit les éléments nécessaires c'est-à-dire qu'on déclare des faits (ex. oiseau(pinson)) et qu'on définit des règles (ex. oiseau(x) : -ailes(x) ).

L'utilisateur demande un traitement en posant une question (ex. ?ailes(pinson) ) qui consistera à demander la vérification d'une règle. C'est le moteur d'inférence qui fera celle-ci.

### 3. CHOIX DE SOLUTION

Nous exposons dans ce chapitre les caractéristiques de l'outil de test que nous avons décidé d'implanter et l'environnement dans lequel il a été réalisé.

#### 3.1. Outil de démonstration

Notre but n'était pas de fournir à un utilisateur un système opérationnel pour tester la validité d'un jugement.

Nous voulions voir dans quelle mesure on pouvait réaliser avec Prolog des fonctions d'évaluation et en quoi un langage non-procédural rendait ce travail plus aisé.

C'est pourquoi nous nous sommes limitées à créer un outil minimal de démonstration de la validité d'un ensemble de faits-et-standards pondérés par rapport aux groupes de validité sans nous préoccuper de l'interface avec l'utilisateur.

Cet outil est un noyau de démonstration.

Nous avons greffé à ce noyau une fonction de saisie de nouveaux groupes de validité. Il nous semblait en effet très intéressant d'examiner comment il était possible de saisir des informations venant de l'utilisateur, de les transformer en structures Prolog pour agrandir ainsi la Base de Connaissances.

### 3.2. Environnement

Nous avons choisi le langage Prolog parce que pour réaliser le test de validité, il fallait démontrer des propositions logiques et ces démonstrations pouvaient se faire facilement avec le moteur d'inférence Prolog.

De plus, c'était pour nous une belle occasion de nous rendre compte des possibilités et des limites d'un langage non-procédural par rapport aux langages traditionnels.

Nous avons étudié Prolog à l'aide de l'ouvrage de CLOKSIM et MELLISH (7); la version de Prolog présentée dans ce livre est la plus proche de la version disponible sur le DEC20 de l'Institut et c'est pourquoi nous avons implanté l'outil sur cette machine.

### 3.3. Implantation

Nous avons vu précédemment (2.2) qu'un outil réalisé avec Prolog se compose des parties suivantes : la base de connaissances (faits et règles) et les questions de l'utilisateur (traitements demandés).

#### 3.3.1. Base des connaissances

##### 3.3.1.1. Les Faits

La base de connaissances contient en permanence les différents groupes de validité de faits pondérés et de standards pondérés. Ces groupes sont stockés sous la forme d'une équivalence, d'une implication ou d'une incompatibilité. Ils constituent la base des faits.

ex. équiv ( 1, affirmé, 3, nié )

Cela signifie que le standard 1 à l'état affirmé est équivalent au standard 3 à l'état nié.

### 3.3.1.2. Les Règles

Les règles concernent les fonctions offertes par l'outil. (elles seront décrites en 3.3.2.) et les transformations des trois types de relations en incompatibilités.

ex. incomp (X,E,T,nié) : - équiv (X,E,Y, affirmé).

L'équivalence de X et Y est transformée en incompatibilité entre X et non Y ( cfr. 1.2).

### 3.3.2. Questions

L'outil offre principalement deux fonctions.

#### 3.3.2.1. Evaluation : INVALID (invalidité)

Cette fonction

- . compare une liste entrée de faits ou standards pondérés et leur état avec les groupes de validité définis
- . dit si cette liste est invalide ou non
- (si elle l'est, elle donne les faits ou standards rejetés).

Saisie : liste des couples [ nrofstd, état]

ex [ [1,nié] , [6,affirmé] , [8, nié] ]

Résultat : - si la liste est invalide :

YES + le fait ou standard  $X_1$  à l'état  $E_1$  est incompatible avec le fait ou standard  $X_2$  à l'état  $E_2$

- si la liste est valide

NO

#### 3.3.2.2. Ajout d'un groupe de validité : AJOUT

Cette fonction permet d'ajouter à la base des connaissances un groupe de validité de faits ou de standards.

Elle demande à l'utilisateur la relation qui définit le groupe de validité, et ensuite chacun de ses membres avec son état.

Saisie : . nom de la relation (équivalence, implication, incompatibilité)

- . premier couple [nrofstd, état]
- . deuxième couple [nrofstd, état]

Résultat : YES, quand le groupe a été introduit dans la base de connaissances.

#### 4. CONNEXION AVEC LE PROTOTYPE DOCUMENTAIRE

On peut réaliser une connexion avec le prototype documentaire de trois façons :

réimplanter le système documentaire dans Prolog, implanter le test de validité dans DBASE2 ou intégrer les deux outils. (chacun restant implanté dans le langage choisi initialement).

D'ores et déjà, nous supprimons la solution qui consisterait à implanter tout le système en DBASE2. Car même si cela était possible, cela n'aurait aucun intérêt au niveau du traitement des propositions logiques.

Les deux autres possibilités nous paraissent nettement plus intéressantes, ainsi les examinerons nous plus en détail.

##### 4.1. Intégration des deux outils

Cette solution consiste à intégrer les deux systèmes, chacun de ceux-ci, étant réalisé dans le langage qui lui est le plus adéquat.

Il faut pour cela concevoir un interface entre les deux systèmes, qui extraira les faits et standards de la base documentaire et les retradaira dans le formalisme de Prolog.

Cette solution comporte aussi des inconvénients. En effet, elle résistera difficilement à l'extension du contrôle de validité. Plus celui-ci deviendra complexe, plus il travaillera sur des éléments différents de la BD. En conséquence un plus grand volume d'informations sera dupliqué et le "mapping" d'une représentation à l'autre deviendra lui aussi plus complexe.

Remarquons aussi que cette solution permettrait de réaliser l'interface utilisateur du système, dans DBASE2 qui nous offre des facilités de dialogue.

#### 4.2. Implanter tout le système en Prolog

Cela signifie que le langage Prolog puisse couvrir les trois dimensions suivantes : BD classique, recherche documentaire et représentation des structures logiques.

Au niveau de la dimension BD classique, voici comment nous l'aborderions, pour notre problème particulier.

##### - Les données

La transformation du schéma conceptuel en Prolog est assez aisée.

Une entité du schéma conceptuel devient un prédicat de Prolog.

ex. L'entité "personne" définie comme suit :

PERSONNE	*
- nom	devient le prédicat
- âge	personne(Nom, Age)

##### - Les fonctions

Il est possible de réaliser des fonctions d'interrogation, par critère sélectif, ou par liste prédéfinie, en donnant des valeurs à un, plusieurs ou tous les arguments.

ex: `personne (X, 26) =` liste des personnes ayant 26 ans, se trouvant dans la BD.

Les fonctions des mises à jour consistent souvent à ajouter des faits à la base de connaissances.

Au niveau de la recherche documentaire, le problème nous semble beaucoup plus difficile à résoudre.

Le principal obstacle est la représentation et la manipulation du texte.

Sous quelle forme stocker le texte?

Comment le manipuler?

Ce sont en effet deux questions auxquelles nous n'avons pas de réponse.

C'est d'ailleurs un sujet actuel dans l'étude de Prolog.

Au niveau de la représentation des structures logiques, il est évident que Prolog est approprié à leur traitement.

Remarquons qu'un inconvénient important de cette solution est de réaliser un interface "user-friendly", car Prolog n'offre aucune primitive particulière pour faciliter sa conception. Et le dialogue qu'il propose est insuffisant dans le cadre d'un SIAD, mais il est aussi difficile à cacher.

## 5. CONCLUSION

### 5.1. Critique

Nous voudrions faire une brève critique des points suivants: le test de validité est un outil minimal sans interface, performant et rapide mais isolé.

#### 5.1.1. Outil minimal sans interface

Rappelons que notre but était seulement de voir si le langage Prolog était un outil adéquat pour implanter des fonctions de test de validité, faisant intervenir des éléments de logique propositionnelle.

Dans ce but Prolog nous offre de nombreuses facilités. ex. Prolog transforme facilement des équivalences et implications en incompatibilités.

Nous ne nous sommes pas souciées de l'interface, puisque la facilité d'utilisation était pour nous, par manque de temps, d'un moindre intérêt. C'est en effet un tout autre problème que le test de validité.

De plus le test de validité que nous avons réalisé est très simple. Nous testons seulement la validité d'un ensemble de faits et standards pondérés, par rapport aux groupes de

validité, de faits ou standards pondérés, définis par les juristes (cfr.1.2.).

Nous n'exploitons donc pas totalement dans ce test la notion de composition (que nous pourrions considérer comme un sous-ensemble de faits et de standards dans un jugement).

Le test n'aborde pas non plus l'aspect de cohérence logique interne d'un jugement.

Le système est donc un outil minimal, parce que le test de validité est très simple et que la procédure pour l'utiliser est un peu rébarbative.

#### 5.1.2. Outil performant et rapide

Le test de validité que nous avons défini est élémentaire, mais est cependant basé sur un grand nombre de combinaisons logiques.

Et il devient rapidement impossible à réaliser à la main, quand le nombre de composants de l'ensemble des faits et standards à tester augmente .

Nous avons utilisé plusieurs fois le test dans le contexte suivant: un ensemble de 8 faits et standards à tester par rapport à une dizaine de groupes de validité.

Prolog réalise ce type de test dans un temps "record" (nous entendons par là que la réponse est immédiate) et rigoureusement: il parcourt systématiquement toutes les combinaisons possibles.

#### 5.1.3. Outil isolé

Le test de validité est un outil isolé, en ce sens qu'il est réalisé dans un langage différent et sur une machine différente de l'environnement du système de recherche documentaire.

Or ces deux systèmes sont complémentaires.

ex. On désire avoir de plus amples informations sur des faits et des standards qui interviennent dans un groupe de validité.

Le test de validité ne sera donc vraiment "valable" que remis dans le contexte du système documentaire.

## 5.2. Extensions

Voici les principales extensions que nous voyons pour le test de validité.

- Etendre le test de validité

Comme nous l'avons souligné, le test que nous avons réalisé est très simple. Il vérifie seulement la validité d'un ensemble de faits et standards pondérés, par rapport aux groupes de validité.

L'outil de test de la validité d'un jugement devrait s'étendre notamment à la cohérence logique interne d'un jugement et à la comparaison de jugements entre eux.

- Intégration des deux outils (documentaire et test de validité)

Comme nous l'avons discuté supra, c'est l'extension qui nous paraît la plus nécessaire.

Pour cela, nous avons relevé deux possibilités :

soit réaliser tout le système en Prolog,

soit écrire un interface entre les deux outils, pour gérer la transformation de données de la BD, dans le formalisme demandé par Prolog.

---

## CONCLUSION

---

Le but était de réaliser un prototype d'un système d'aide à l'élaboration d'un jugement qui doit comporter deux aspects, l'un documentaire et l'autre d'évaluation.

Du point de vue documentaire, le prototype devait soulever les problèmes que posait la combinaison des aspects de BD classique et de recherche documentaire.

Du point de vue du contrôle de validité, le prototype devait montrer dans quelle mesure Prolog permettait de réaliser un test de validité.

Par rapport à ces objectifs, nous pouvons tirer les conclusions suivantes.

1. Il est indispensable, dans un outil documentaire, d'intégrer la gestion de données structurées et non structurées. Et c'est à partir d'un système de gestion de BD traditionnel que cette intégration doit se faire.
2. Outre les problèmes d'apprentissage de Prolog, nous pouvons dire qu'il est directement adapté aux fonctions de validité que nous voulions créer. De plus c'est à l'aide de Prolog qu'il faut poursuivre les recherches pour étendre les contrôles de validité.
3. Le prototype réalisé n'est pas d'application dans un environnement réel mais il peut être utilisé à des fins pédagogiques comme outil de support à la formalisation de jugements et à leur contrôle de validité.

Nous voyons principalement trois extensions à envisager :

1. En ce qui concerne l'aspect documentaire, on pourrait chercher à représenter les faits dans leur contexte spatial, temporel, ...

Comme cela a été dit dans le mémoire (1) les faits sont enregistrés indépendamment de leur environnement temporel,...

2. Dans le cadre limité de l'utilisation de Prolog, il faudrait étendre le contrôle de validité; c'est-à-dire faire intervenir les compositions d'un jugement, tester la cohérence d'un jugement, faire des comparaisons entre jugements.
  3. Une des extensions les plus intéressantes reste l'intégration des deux outils et dans cette optique, c'est l'utilisation d'un système tel que Prolog mais qui permettrait d'accéder à des BD traditionnelles qui s'avère la plus adéquate.
-

BIBLIOGRAPHIE

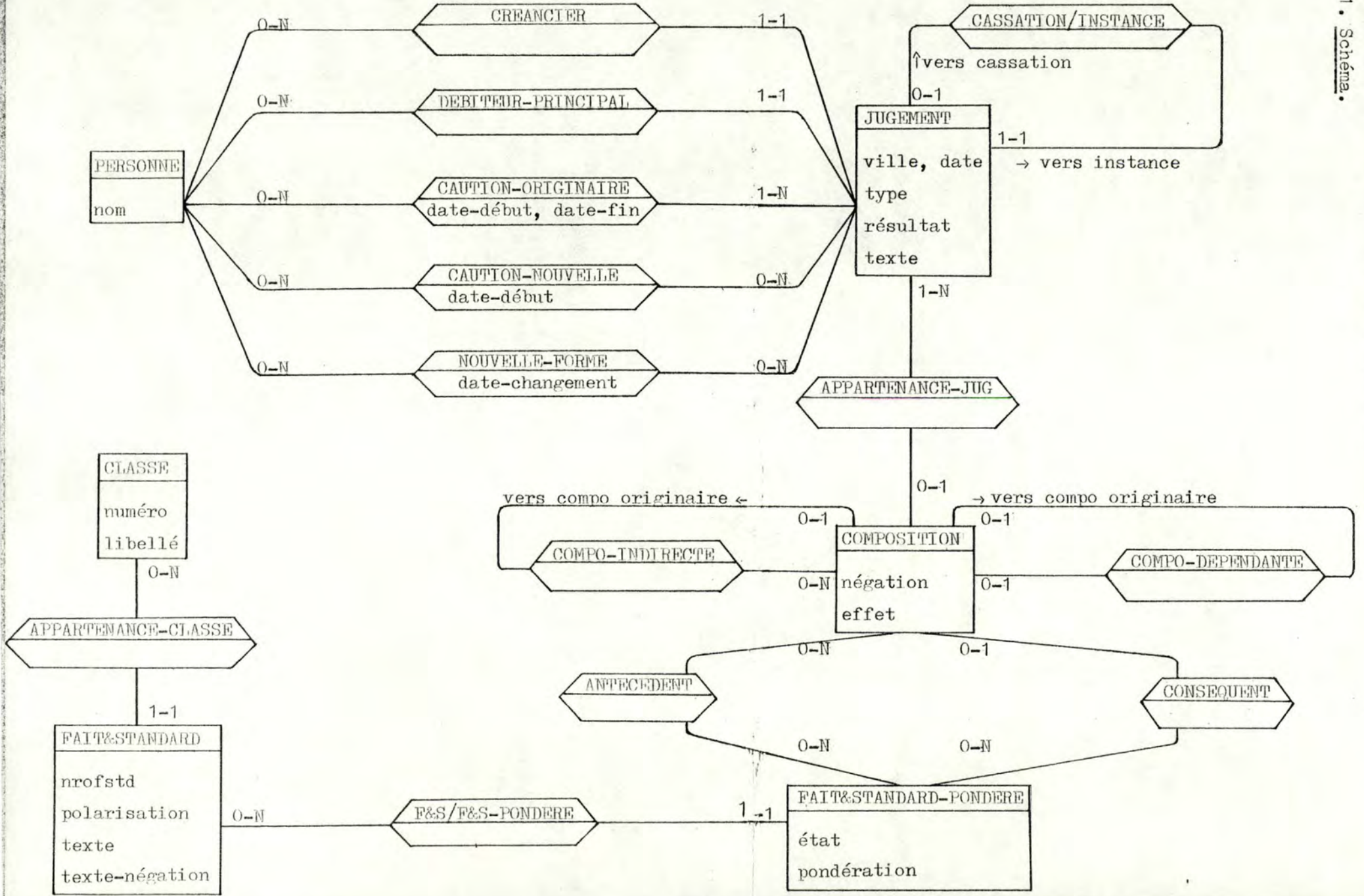
---

- (1) B. HICK et P. TIRTIAUX ;  
Système informatique d'aide au jugement de qualification;  
Mémoire de l'Institut d'Informatique FNDP NAMUR, 1982.
- (2) - ;  
BASIS - Recherche Documentaire et Bases de Données ;  
BATTELLE - Centres de recherche de GENEVE.
- (3) C.J. DATE ;  
An Introduction to Database Systems ;  
ADDISON - WESLEY, 1981.
- (4) F. BODART et Y. PIGNEUR ;  
Conception assistée des applications informatiques :  
1. étude d'opportunité et analyse conceptuelle;  
MASSON, 1983.
- (5) - ;  
DBASE2 - USER MANUAL ;  
ASHTON - TATE.
- (6) John L. BENNETT (ed) ;  
Building Decision Support Systems ;  
ADDISON - WESLEY, 1983.
- (7) W.F. CLOCKSIM et C.S. MELLISH ;  
Programming in Prolog ;  
SPRINGER - VERLAG, 1981.

## ANNEXES

1. Schéma conceptuel tiré du mémoire (1)
2. Systèmes de recherche documentaire :
  - Mistral (BULL)
  - Golem (SIEMENS)
  - Stairs (IBM)
  - Système de l'IRETIJ (Montpellier)
3. Schéma relationnel DBASEII de la base de données du prototype
4. Schéma des Modules DBASEII du prototype
5. Exemples d'utilisation du prototype documentaire
6. Exemple de programmation en DBASEII

1. Schéma.



## 2. Contraintes d'intégrité.

1. Un FAIT&STANDARD PONDERE doit intervenir dans au moins une COMPOSITION soit comme ANTECEDENT , soit comme CONSEQUENT.
2. Une COMPOSITION n'a qu'un seul conséquent. Celui-ci peut être un FAIT&STANDARD PONDERE ou une COMPOSITION qui est dite DEPENDANTE. Les types d'associations CONSEQUENT et COMPO-DEPENDANTE sont donc exclusifs.
3. Une COMPOSITION a au moins un antécédent. Celui-ci peut être un FAIT&STANDARD PONDERE ou une COMPOSITION qui est dite INDIRECTE. Pour chaque COMPOSITION, il faut donc au moins une association ANTECEDENT ou une association COMPO-INDIRECTE.
4. Une PERSONNE intervient au moins dans une des associations suivantes : CREANCIER, DEBITEUR-PRINCIPAL, CAUTION-ORIGINAIRE, CAUTION-NOUVELLE, NOUVELLE-FORME.

### 3. Description sémantique.

#### 3.1 Les types d'entités.

##### PERSONNE :

Une personne est une personne physique ou morale qui intervient dans un jugement.

"nom" est le nom de la personne physique ou la raison sociale de la personne morale.

##### JUGEMENT :

Un jugement est une décision juridique relative au problème de cautionnement restreint au cas décrit dans le mémoire de Hick et Tirtiaux(1) ( 2.1.2 page 8).

"ville" est la ville où a été rendu le jugement.

"date" est la date du jugement.

"type" représente le degré de juridiction. Ses valeurs sont Cassation et Instance.

"résultat" donne le résultat du jugement. Si celui-ci est de type Instance, les valeurs du résultat sont Maintien et Libération suivant que la caution originaire est maintenue ou libérée de ses obligations envers le créancier.

Si le jugement est de type Cassation, les valeurs du résultat sont Cassé et Identique suivant que la décision d'instance correspondante est cassée ou maintenue.

"texte" est le texte de jurisprudence.

##### FAIT&STANDARD :

Un fait&standard est un élément qui peut intervenir dans la formalisation d'un jugement. Cet élément est soit un fait, soit un standard.

"nrofstd" est l'attribut identifiant du fait ou du standard.

"polarisation" donne la polarisation du fait ou du standard. Ses valeurs sont Positive, Négative et Neutre.

"texte" est le libellé du texte du fait ou du standard.

"texte-négation" est le libellé du fait ou du standard à l'état nié.

##### CLASSE :

Une classe regroupe des fait&standard relatifs à un même concept. Tous les standards sont regroupés dans une même classe.

"numéro" est l'attribut identifiant de la classe.

"libellé" est une brève description du contenu de la classe.

**COMPOSITION :**

Une composition représente en tout ou en partie l'argumentation juridique développée par le juge dans un jugement.

"négation" dit si la composition est une négation conditionnelle ou une conditionnelle.

"effet" est l'effet de la composition. Ses valeurs sont + et - suivant que l'effet de la composition est de libérer ou maintenir les obligations de la caution originaire. La propriété n'a pas de valeur s'il s'agit d'un jugement de type Cassation ou d'une composition indirecte.

**FAIT&STANDARD PONDERE :**

Un fait&standard pondéré est un fait ou un standard qualifié auquel le juge a donné dans un jugement, une pondération et un état et qui peut donc intervenir dans une composition.

"état" est l'état du fait ou du standard. Ses valeurs sont Affirmé et Nié suivant que le fait ou le standard est présent ou non.

"pondération" est la pondération du fait ou du standard. Ses valeurs sont Important, Subsidaire et Sans importance suivant le degré d'importance attribué par le juge pour le fait ou le standard dans la composition.

**3.2 Les types d'associations.****CREANCIER ( PERSONNE , JUGEMENT ) :**

exprime le fait qu'une personne joue le rôle de créancier dans un jugement. La personne est une personne morale.

**DEBITEUR-PRINCIPAL ( PERSONNE , JUGEMENT ) :**

exprime le fait qu'une personne joue le rôle de débiteur-principal dans un jugement. La personne est une personne morale.

**CAUTION-ORIGINAIRE ( PERSONNE , JUGEMENT ) :**

exprime le fait qu'une personne joue le rôle de caution-originaire dans un jugement. La personne est une personne physique.

"date-début" est la date du début de l'engagement de la personne comme caution originaire.

"date-fin" est la date de la fin de l'engagement de la personne comme caution originaire.

CAUTION-NOUVELLE ( PERSONNE , JUGEMENT ) :

exprime le fait qu'une personne joue le rôle de caution-nouvelle dans un jugement. La personne est une personne physique.

"date-début" est la date du début de l'engagement de la personne comme caution nouvelle.

NOUVELLE-FORME ( PERSONNE , JUGEMENT ) :

exprime le fait qu'une personne joue le rôle de la nouvelle forme de la société débiteur-principal dans un jugement. La personne est une personne morale.

"date-changement" est la date du changement de forme de la société débiteur-principal.

CASSATION/INSTANCE ( JUGEMENT , JUGEMENT ) :

exprime la relation qui associe un jugement de type Instance au jugement de type Cassation qui lui correspond.

APPARTENANCE-JUG ( JUGEMENT , COMPOSITION ) :

exprime le fait qu'une composition a été développée dans un jugement.

COMPO-INDIRECTE ( COMPOSITION , COMPOSITION ) :

exprime le fait qu'une composition est composition indirecte c-à-d antécédent d'une autre composition.

COMPO-DEPENDANTE ( COMPOSITION , COMPOSITION ) :

exprime le fait qu'une composition est composition dépendante c-à-d le conséquent d'une autre composition.

ANTECEDENT ( FAIT&STANDARD PONDERE , COMPOSITION ) :

exprime le fait qu'un fait ou standard pondéré est un des antécédents d'une composition.

CONSEQUENT ( FAIT&STANDARD PONDERE , COMPOSITION ) :

exprime le fait qu'un fait ou standard pondéré est le conséquent d'une composition.

F&S/F&S-PONDERE ( FAIT&STANDARD , FAIT&STANDARD PONDERE ) :

exprime le fait qu'un fait ou standard a reçu une pondération dans un jugement pour donner un fait ou standard pondéré.

APPARTENANCE-CLASSE ( FAIT&STANDARD , CLASSE ) :

exprime le fait qu'un fait ou standard appartient à une classe.

## ANNEXE 2 : Quelques systèmes de recherche documentaire.

### PLAN :

1. architecture du système documentaire
2. instruments de références documentaires
3. indexation
4. interrogation de la base documentaire
5. protection

### 1. MISTRAL ( BULL ).

#### 1.1 Architecture du système documentaire.

L'élément de base que manipule Mistral est le document. Ces documents sont stockés dans une base documentaire. Dans une même installation, plusieurs bases peuvent être gérées par Mistral.

#### Organisation de la base documentaire.

Un même programme Mistral peut donc gérer plusieurs bases distinctes, pour autant que les caractéristiques de ces bases soient déclarées et que ces dernières soient initialisées.

La base de données est donc le niveau le plus élevé de la structure documentaire.

Pour regrouper les documents, en fonction des applications auxquelles ils se rapportent, la base peut être découpée en sections, appelées domaines. Cette découpe permet à l'utilisateur de ne faire une recherche que dans le domaine qui l'intéresse.

Le domaine constitue le second niveau hiérarchique de la structure documentaire.

Une base peut compter jusqu'à 50 domaines.

On peut encore diviser les domaines, en fonction de la date d'entrée des documents dans la base documentaire.

C'est la période qui constitue le 3ième niveau de la structure documentaire.

Le 4ième niveau est le document. Ce document peut être :

- une description plus ou moins détaillée d'un document (référence bibliographique)
- un texte intégral
- une description et un texte intégral.

### Le document.

Le document est divisé en un certain nombre d'éléments appelés CHAMPS (ex : titre, auteur, référence, texte, date,...).

Un document contient au maximum 150 champs.

Chaque champ contient une succession de DESCRIPTEURS (ex : mots d'un texte, une date,...). Ces descripteurs sont aussi appelés VALEURS.

On distingue 2 types de champs :

champ monoséparateur :

il y a un seul séparateur de valeurs,

ex : champ reprenant les mots-clés

champ multiséparateur :

les séparateurs de valeurs sont multiples,

ex : champs de type texte.

Tous les documents d'une même base de données doivent évidemment avoir la même structure, càd comprendre les mêmes champs.

### 1.2 Instruments de références documentaires.

Un fichier inverse contient pour chaque valeur distincte d'un champ (défini champ "à fichier inverse"), la liste de tous les documents dans lesquels apparaît cette valeur.

Ces fichiers inverses se constituent pour chacun des champs lors de l'initialisation et du chargement de la base documentaire.

C'est ici qu'intervient la découpe du champ en valeurs, en fonction de son attribut monoséparateur ou multiséparateur.

Plusieurs fichiers inverses peuvent donc être associés à une même base documentaire.

A chacun de ces champs "à fichier inverse" est associé un instrument de références, lexique ou thésaurus, constitué à partir de l'indexation.

Les lexique et thésaurus ont pour but de faire correspondre le terme d'indexation et le terme de requête, en normalisant et structurant un langage commun à l'indexation et à l'interrogation. Ainsi, un nombre maximum de documents pertinents est retrouvé lors de l'interrogation par l'utilisateur.

A un champ correspond un lexique ou un thésaurus suivant que l'on définit ou non des relations entre les descripteurs de ce champ.

Lexique.

C'est un dictionnaire de descripteurs (valeurs distinctes contenues dans le champ auquel est associé le lexique) qui sont classés par ordre alphabétique.

Thésaurus.

C'est un dictionnaire de descripteurs (cfr lexique) dont certains de ceux-ci sont reliés entre eux par des relations qui peuvent être de différents types :

synonymie : Mistral considère que les descripteurs synonymes sont équivalents, c'est-à-dire que toute recherche portant sur l'un d'eux entraîne automatiquement la recherche des documents indexés sous le ou les descripteurs équivalents (sauf avis contraire de l'utilisateur).

hiérarchie (générique ou spécifique) :

ex : économie peut se subdiviser en micro-économie et macro-économie; de même macro-économie se subdivise en banque, commerce extérieur,...

association : cette relation est utilisée pour exprimer qu'un descripteur en suggère un autre,  
ex : coiffeur et cheveux.

remarques : l'utilisateur a la possibilité de définir d'autres relations que celles citées supra.

Le thésaurus peut être structuré en thèmes (matières) qui eux-mêmes sont précisés par des niveaux de spécificité. Cette structuration n'est utile que pour l'édition du thésaurus et n'influence pas la recherche.

Caractère ouvert ou fermé du thésaurus ou lexique.

Un thésaurus, ou lexique, peut être ouvert ou fermé.

Ouvert : le thésaurus se constitue progressivement, c'est-à-dire que lors de tout chargement de nouveaux documents dans la base documentaire, les valeurs encore inconnues jusque-là sont incorporées automatiquement dans le thésaurus.  
Il n'y a donc pas de contrôle sur l'unicité et la permanence des descripteurs utilisés.

Fermé : tout nouveau descripteur rencontré lors de l'indexation, qui est différent de ceux qui constituent le thésaurus, est rejeté.

Le thésaurus doit donc être construit a priori et est figé pour un temps déterminé. Pour modifier ou mettre à jour le thésaurus une procédure spécifique sera exécutée.

#### Organisation des lexiques et thésaurus.

A chaque champ déclaré "à fichier inverse" est associé un lexique ou un thésaurus.

De plus, il est possible d'associer à plusieurs champs "à fichier inverse" un même lexique ou thésaurus. On dit alors que le lexique ou thésaurus est multichamp.

lexique monochamp : ce lexique n'est constitué que des descripteurs d'un même champ.

lexique multichamp : ce lexique contient les descripteurs de plusieurs champs.

Deux solutions sont alors possibles :

monosens : on ne sait plus retrouver le champ dont provient un descripteur, c'est-à-dire que l'on mélange dans ce lexique les descripteurs sans noter leur champ origine.

multisens : on sait retrouver le champ dont provient un descripteur, c'est-à-dire que l'on a noté dans le lexique pour chaque descripteur son champ d'origine. Aussi est-il possible qu'un même descripteur apparaisse plusieurs fois dans un même lexique mais avec des champs d'origine différents.

### 1.3 Indexation.

L'indexation se définit comme la traduction d'un document en termes documentaires (descripteurs, termes index ou mots-clés) qui expriment le contenu du document.

On appelle langage documentaire l'ensemble des descripteurs dans un domaine et un système donnés. Le langage documentaire comporte nécessairement un lexique.

Procédés offerts par Mistral.

Le documentaliste qui crée des documents pour la base documentaire doit :

produire des descripteurs ou mots-clés pour les ranger dans les différents champs monoséparateurs (ex : auteur, mots-clés..)

fournir un ou plusieurs textes qui caractérisent mieux le document et qui sont rangés dans des champs "texte" ou multi-séparateurs (ex : résumé, titre,...).

Mistral considère alors comme termes index :

les valeurs des champs monoséparateurs, du type mot-clé

des termes à extraire des textes des champs multiséparateurs.

On distingue ainsi 2 procédés d'indexation :

indexation manuelle réalisée par le documentaliste lui-même

indexation automatique (ou semi-automatique) réalisée par le système au moyen d'algorithmes d'analyse de texte.

Mistral propose en option un système d'indexation entièrement automatique par utilisation de traitements linguistiques, qui s'avère plus efficace pour l'indexation automatique de textes longs et riches tels que le contenu intégral des documents.

Indexation manuelle.

Il s'agit de choisir les mots-clés qui expriment le mieux possible le sujet traité par le document.

Si l'on veut retrouver des documents avec précision, il est nécessaire de les indexer finement, c'est-à-dire de retenir des mots-clés très précis. Afin cependant de ne pas nuire à des questions générales, le documentaliste s'aide du thésaurus dans lequel il définit des relations génériques-spécifiques. Ces relations peuvent être prises en compte lors des recherches.

Les problèmes d'orthographe des différentes formes grammaticales d'un mot peuvent se résoudre,

par convention, c'est-à-dire ramener toute forme au masculin, au singulier, à l'infinitif,...

en mettant les différentes formes en relation de synonymie dans le thésaurus.

Les 2 méthodes se combinent souvent.

Mistral accepte tous les mots composés, munis ou non de blancs (ex : banque de données, machine-outil,...).  
Le documentaliste aura intérêt à ne pas décomposer un mot composé en ses parties.

#### Indexation semi-automatique.

Elle consiste à demander à la machine de caractériser elle-même un texte (résumé, titre,...) par des mots-clés.

La méthode employée par Mistral est la suivante :  
le documentaliste fournit en donnée le texte à analyser,

Mistral lit ce texte,  
en extrait tous les mots,  
supprime les mots vides, c'ad sans aide à la compréhension du texte (articles, prépositions, verbes généraux,...),  
confronte la liste restante au thésaurus;

l'utilisateur a alors le choix entre  
rejeter tout mot n'appartenant pas au thésaurus  
(thésaurus fermé),

incorporer au thésaurus les mots n'y appartenant pas  
(thésaurus ouvert);

les mots retenus sont considérés comme mots-clés et bénéficient donc d'une entrée dans le fichier inverse associé au champ.  
Ils pourront ainsi faire l'objet d'une recherche primaire.

#### Extraction des mots :

on appelle mot toute chaîne de caractères délimitée par des séparateurs, un début ou une fin de champ.  
Les séparateurs sont paramétrables.

#### Fichier des mots vides :

il est possible à l'utilisateur de définir jusqu'à 5 fichiers "mots vides". Ceci permet d'avoir des mots qui seront considérés comme vides pour un champ, mais auront un sens pour un autre champ (suivant le fichier des mots vides qui est associé au champ).

#### Intérêt :

dans ce système, il suffit de fournir une liste des mots vides et le reste est purement automatique. De plus le système d'indexation reste homogène alors que la stabilité de l'indexation manuelle n'est assurée ni dans le temps ni entre les indexeurs.

**Limite :**

si la qualité obtenue par ce système est suffisante pour des textes courts et simples comme les résumés, une méthode plus efficace est toutefois requise pour le traitement des textes longs et riches du document original.

Indexation automatique par traitement linguistique.

Les traitements linguistiques sont réalisés par des modules ayant des fonctions précises. L'ensemble est enchaîné automatiquement par un moniteur.

Pour le traitement du texte intégral, Mistral propose les fonctions linguistiques suivantes :

découpage du texte après la saisie :

le texte est considéré lors de la saisie comme une suite de lignes de caractères formant une page et cet algorithme le transforme en une suite de mots.

analyse morphologique :

chaque mot est analysé; s'il est erroné, le système peut proposer des solutions et si le mot est reconnu, le système lui associe des propriétés syntaxiques et sémantiques qui serviront pour la suite.

reconnaissance d'expressions idiomatiques :

on regroupe des mots formant une expression composée (ex : à concurrence de, table des matières,...); cela permet de lever certaines ambiguïtés de polysémies (ex : à concurrence de et concurrence).

analyse syntaxique locale :

elle lève les ambiguïtés grammaticales (ex : "contre" comme verbe ou préposition); c'est un système à apprentissage.

normalisation :

les mots vides sont éliminés et les autres sont transformés selon les normes choisies

(ex : F-N-D-P et F.N.D.P. deviennent FNDP,  
un verbe est mis à l'infinitif,  
un adjectif au masculin singulier,...).

Les mots normalisés (simples ou composés) qui ont été retenus sont introduits automatiquement dans un champ Mistral "index" qui sera inséré lors de son chargement dans la base documentaire.

Ces algorithmes linguistiques sont aussi utilisés au niveau de l'interrogation. Les mots et expressions introduits en langage naturel par l'utilisateur sont soumis aux mêmes traitements de reconnaissance et normalisation avant d'être confrontés aux fichiers inverses.

Notons que les questions doivent être des expressions où des opérateurs booléens relient des mots ou des morceaux de phrases.

#### 1.4 Interrogation de la base documentaire.

La principale raison d'être d'un système de recherche documentaire est la recherche rétrospective.

Ainsi allons-nous surtout examiner dans le détail les procédures d'interrogation qui permettent d'effectuer une recherche rétrospective.

##### Procédures de service.

Elles permettent la connexion d'un utilisateur à une base (avec vérification des mots de passe,...) avec spécification du domaine et de la période, si nécessaire.

On peut également lister les différents domaines d'une base, les différentes périodes d'un domaine.

Il est aussi possible de changer de base en cours de session.

##### Procédures d'assistance.

A tout moment, il est possible d'obtenir des informations sur la structure de la base documentaire, sur les commandes et sur le déroulement de la session en cours.

Ex : liste des champs associés à un lexique

liste des différents champs avec leur type et le nom du lexique associé s'il y en a un

affichage de la liste de recherche

liste des commandes possibles et leurs fonctions

explications concernant une commande

... .

##### Procédures de recherche.

Mistral permet à la fois d'exécuter des recherches rétrospectives en mode interactif, en traitement par lot.

Deux types de recherche interactive sont possibles :  
recherche sur fichier inverse (ou primaire),  
recherche sur texte.

La recherche sur texte ne peut s'effectuer que sur un ensemble de documents sélectionnés par une recherche sur fichier inverse.

Par contre le champ d'investigation d'une recherche sur fichier inverse peut être la base, limitée ou non à un domaine, limité ou non à une période.

Le résultat d'une instruction de recherche est un résultat chiffré qui représente le nombre de documents répondant à l'équation de recherche.

Si l'utilisateur veut consulter ces documents, en tout ou en partie, il doit exécuter une commande spécifique (visualisation).

#### Liste de recherche.

Toute instruction de recherche (sur fichier inverse ou sur texte) qui obtient un résultat (càd un nombre de documents sélectionnés > 0) est numérotée et mémorisée. On appelle cela une étape de recherche. A tout moment au cours de la session, il est possible de revoir la liste de recherche, de visualiser des documents sélectionnés lors d'une étape précédente, ... . Les résultats de chaque étape de recherche sont donc disponibles pendant toute la durée de la session. De plus, en fin de session, on peut conserver sa liste de recherche, pour qu'elle soit disponible lors de la session suivante. De même, il est possible au cours d'une session d'effacer la liste de recherche ou une partie de celle-ci et d'en recommencer une nouvelle.

#### Recherche sur fichier inverse.

L'instruction de recherche se divise en 3 parties :

1. indication du lexique ou du thésaurus auquel appartient le ou les descripteurs constituant les valeurs de l'instruction de recherche,
2. équation de recherche qui peut se présenter sous différentes formes :
  - descripteur (mot ou groupe de mots)
  - expression booléenne reliant des descripteurs (appartenant au même lexique ou thésaurus); les opérateurs booléens utilisés sont ET, OU, SAUF, avec la logique des parenthèses
  - expression booléenne reliant des résultats obtenus lors d'une étape antérieure (cfr liste de recherche)
  - expression booléenne reliant des descripteurs et des résultats obtenus précédemment,
3. nom du champ d'origine du descripteur, lorsque le lexique est multisens.

La troncature est utilisée comme suit. Les descripteurs peuvent toujours être tronqués à droite. Cette troncature s'opère sur un nombre quelconque de caractères.

Ex : le mot informati+ ("+" étant le caractère de troncature) entraînera la recherche de tous les descripteurs connus dans le lexique indiqué, commençant par informati. Ainsi le système pourrait prendre en compte informaticien,  
 informatique,  
 information,  
 ... .

Pour un mot indéterminé, Mistral procède de la façon suivante. Lorsque pour un mot donné, plusieurs descripteurs peuvent être pris en compte par le système, (cela peut être dû à une troncature par exemple) Mistral demande à l'utilisateur de choisir entre ceux-ci. Les descripteurs que l'utilisateur sélectionnera seront reliés entre eux par un opérateur OU dans la recherche.

#### Recherche sur texte.

La recherche sur texte permet de faire une sélection plus détaillée des documents. Dans cette recherche, n'importe quel type de champ est utilisable. Cela permet de faire de la recherche sur des champs qui ne sont pas "à fichier inverse".

L'instruction de recherche sur texte comprend 4 parties :

1. TXT qui est l'indication de la commande de recherche sur texte,
2. le numéro de l'étape de recherche dont les documents sélectionnés constituent le point de départ de la recherche sur texte,
3. le nom du champ sur lequel porte la recherche,
4. l'opérateur de recherche et les valeurs.  
Les opérateurs de recherche sont basés sur les critères suivants :
  - comparaison de valeurs numériques ou alphabétiques
  - présence ou absence d'une chaîne de caractères ou d'un champ
  - date d'introduction du document dans la base
  - distance entre des mots ou des groupes de mots à l'intérieur d'un champ.

Remarque : dans la recherche sur texte, la fonction de troncature peut aussi être utilisée.

#### Procédures d'aide à la recherche.

Ces procédures ont pour but d'aider l'utilisateur à rédiger de façon optimale ses équations de recherche.

On peut lister l'environnement alphabétique d'un descripteur dans un lexique. Parmi les descripteurs affichés, l'utilisateur peut en sélectionner pour une recherche sur fichier inverse. Les descripteurs ainsi sélectionnés sont reliés entre eux par un opérateur OU.

De même il est possible pour les descripteurs appartenant à un champ auquel est associé un thésaurus, de lister leur environnement sémantique ou une partie de celui-ci en spécifiant une seule relation.

Il est aussi possible de spécifier les relations que Mistral doit prendre en compte automatiquement pour les recherches pendant une session ou même une partie de session.

#### Procédures d'édition.

Le but d'une recherche rétrospective est évidemment de pouvoir consulter les documents recherchés. Aussi est-il possible de les visualiser au terminal ou de les éditer sur papier. On peut visualiser ou n'éditer qu'un des documents sélectionnés, ou un champ de celui-ci, ... . Toutes les étapes intermédiaires entre la visualisation de l'ensemble des documents dans leur entièreté et celle d'un champ d'un des documents, sont possibles.

L'utilisateur dispose également d'une fonction de tri des documents résultats. Les possibilités de tri vont du simple tri croissant sur un champ (numérique ou alphabétique) à des tris sur des champs différents avec des critères variés.

Indépendamment des recherches rétrospectives, des procédures d'édition en traitement par lot peuvent être exécutées. Ce sont les diffusions sélectives ou les gestions de profils.

Lors de la visualisation, après chaque document, l'utilisateur a la possibilité d'arrêter l'édition ou de la poursuivre.

#### Macro-procédures.

Lorsque l'on doit effectuer plusieurs fois les mêmes opérations de recherche, à des périodes différentes, il n'est pas nécessaire de retaper chaque fois les mêmes commandes. En effet, il est possible de définir des macro-procédures. Celles-ci sont constituées par un enchaînement de procédures du système conversationnel (c'est la recherche désirée). Ainsi il suffit à l'utilisateur d'exécuter cette macro-procédure pour réeffectuer la recherche souhaitée.

#### Accès direct au document.

Il est possible d'accéder directement à un document (sans utiliser les fichiers inverses) au moyen de la référence "Mistral" du document. Cette référence constitue un des champs du document; ainsi cette instruction de recherche constitue une étape de recherche et vient s'ajouter à la liste.

### 1.5 Protection.

Mistral permet de limiter l'accès à certains documents, aux seuls utilisateurs autorisés.

La protection fonctionne à différents niveaux :

accès interactif : on stocke une liste des utilisateurs autorisés dans le fichier dialogue,

base documentaire : il y a une association éventuelle d'un mot de passe au nom de la base,

domaine : il y a des associations éventuelles de mots de passe aux noms des domaines,

document : il est possible de spécifier un code de confidentialité lors du stockage du document dans la base,

champ : on peut déclarer un code de confidentialité lors de l'initialisation.

Pour le document et le champ, le code de confidentialité ou clé de secret va de 1 à 50 et chaque mot de passe propre à l'utilisateur comprend un niveau d'habilitation qu'il convient de définir à l'initialisation de la base documentaire. Il peut aller de 1 à 50. L'utilisateur n'accèdera donc qu'aux documents ayant une clé inférieure ou égale à la sienne.

## 2. GOLEM ( SIEMENS ).

### 2.1 Architecture du système documentaire.

Golem permet de gérer une base de données textuelles.  
 L'élément de base qu'il manipule est donc le document.  
 Il est possible de gérer à l'intérieur de la base, un certain nombre de pools d'informations. Chacun est spécialisé  
 soit dans un type d'informations avec une structure et un format propres à chaque pool,  
 soit par un type d'informations par domaine, avec une structure et un format qui peuvent être identiques ou différents d'un pool à l'autre.  
 Chaque pool rassemble un certain nombre de documents.

#### Structure des documents.

Le document est composé d'un ou plusieurs paragraphes qui peuvent être de 3 types :

descripteurs (96 paragraphes au maximum),  
 textes (32 paragraphes au maximum),  
 mixtes (32 paragraphes au maximum).

#### Paragraphes descripteurs.

Ils peuvent contenir :

des descripteurs libres : mot ou expression représentant le contenu du document,

des aspects (intitulés) accompagnés de descripteurs liés (valeurs),

ex :	aspects	descripteurs liés
	auteur	Durand
	éditeur	Dunod.

A ces descripteurs stricts, on peut ajouter :

des informations complémentaires donnant la signification du descripteur,  
 ex : Rome (lieu de rencontre),

indiquant le(s) paragraphe(s) de texte au(x)quel(s) le descripteur se rapporte,

ex : Vienne (3)  
 ↳ 3ième paragraphe de texte;

des informations "utilisateur" donnant par exemple un titre à un paragraphe,

ex : (Résumé)  
 {... texte du résumé...};

des indices qui permettent de relier des descripteurs formant ensemble un sujet du document,

ex : descripteurs libres = créancier(1) / institution financière / Durand(1)

-> Durand est le créancier;

des indicateurs de rôle définissant la fonction d'un descripteur lié à un autre par un indice,

ex : descripteurs libres = exportation(1) / fruits(1,objet) / Belgique(1,destination)

-> exportation de fruits vers la Belgique;

des informations de "sécurité" : clé de protection ajoutée en début de document ou de paragraphe,

ex : (3,Résumé) le paragraphe résumé a le code de protection 3.

#### Paragrapes textes.

Ils contiennent les textes des documents dans un format libre et peuvent être précédés :

d'une clé de protection,

d'une information "utilisateur", par exemple l'intitulé.

#### Paragrapes mixtes.

Ces paragraphes contiennent des textes à l'intérieur desquels certains mots ou expressions sont repérés comme descripteurs libres ou comme aspects avec descripteur(s) lié(s).

Lors du chargement de la notice par Golem, chaque paragraphe mixte est automatiquement supprimé et remplacé par :

un paragraphe descripteur reprenant les descripteurs repérés,

un paragraphe texte.

## 2.2 Instruments de références documentaires.

### Dictionnaire ou thésaurus.

Les descripteurs sont mémorisés dans un dictionnaire qui n'existe pas a priori. Il est constitué des descripteurs des documents au fur et à mesure qu'ils sont enregistrés dans le système.

L'ensemble de ces descripteurs n'est pas spontanément structuré; il s'agit donc d'un dictionnaire plutôt que d'un thésaurus.

Le documentaliste peut cependant introduire jusqu'à 128 types de relations sémantiques et transformer ainsi le dictionnaire en thésaurus.

Trois classes de relations peuvent exister :

relations biunivoques : synonymie, association,...

relations réciproques : hiérarchie,...

relations isolées : définition,...

Golem accepte tous les descripteurs et introduit automatiquement dans le dictionnaire les nouveaux descripteurs rencontrés dans le document; on parle de langage non contrôlé.

Si on désire un contrôle des descripteurs, il faut prévoir un programme en amont de Golem.

### 2.3 Indexation.

L'indexation des textes se fait souvent manuellement par le documentaliste. Celui-ci a le choix entre :

reprendre les mots-clés lui-même dans un paragraphe "descripteurs libres",

signaler les mots-clés dans le texte de paragraphes mixtes et c'est le système qui crée le paragraphe descripteur.

Golem propose un système optionnel PASSAT d'indexation semi-automatique. A l'aide d'un dictionnaire préenregistré, le système réalise une préindexation automatique des textes et le reste de l'indexation sera effectuée par le documentaliste.

Ce dictionnaire comprend les mots et expressions du langage courant et de la terminologie spécialisée de la discipline traitée.

Les mots y sont enregistrés sous la forme de leur racine et sont accompagnés des informations suivantes :

numéro de la liste des désinences à lui appliquer; la première désinence de la liste est celle que doit avoir le mot quand il sert de descripteur (dans le fichier dictionnaire),

indication de la possibilité de faire partie d'une expression,  
ex : rose -> rose des vents

indication de la possibilité d'avoir plusieurs significations,

indication des termes associés à chaque signification du mot,  
ex : bombe -> guerre, révolution,...

indication des termes à utiliser comme descripteur préférentiel,  
ex : verbes et adjectifs renvoient aux formes nominales.

La construction de ce dictionnaire représente un travail considérable; il est toutefois possible de commencer avec un dictionnaire incomplet que l'on enrichit au fur et à mesure de l'enregistrement des textes.

L'indexation PASSAT se fait de la manière suivante :

- enregistrement des textes à traiter,
- tri des mots par ordre alphabétique,
- comparaison de chaque mot avec le dictionnaire,
- pour les mots non trouvés : édition d'une liste qui sert au documentaliste pour compléter manuellement l'indexation, corriger les erreurs, enrichir le dictionnaire,
- pour les mots trouvés : extraction des informations d'accompagnement,
- attribution d'un facteur de pondération à chaque mot, selon sa fréquence dans le texte, la fréquence dans le texte des termes associés,
- calcul de la moyenne des pondérations de chaque mot reconnu,
- choix, à titre de descripteurs, des termes dont la pondération dépasse la moyenne.

#### 2.4 Interrogation de la base documentaire.

##### Procédures de service.

cfr Mistral.

##### Procédures d'assistance.

cfr Mistral.

##### Procédures de recherche.

Deux types de recherche sont possibles :

- recherche primaire par descripteur(s) (fichier inverse),
- recherche secondaire sur texte.

La recherche sur texte ne peut s'effectuer que sur un ensemble de documents sélectionnés par une recherche primaire. Par contre le champ d'investigation de la recherche primaire est le pool entier sur lequel on travaille.

Le résultat d'une instruction de recherche est un chiffre représentant le nombre de documents répondant à l'équation de recherche. Si l'utilisateur veut consulter ces documents, en tout ou en parties, il doit exécuter une commande spécifique.

#### Liste de recherche.

Toute recherche primaire est accomplie à partir de descripteurs. Ceux-ci doivent figurer dans une liste que l'on tient en mémoire (liste de recherche). Ils y sont numérotés et les équations logiques ultérieures seront établies avec les numéros des descripteurs et non les descripteurs eux-mêmes.

Cette liste de recherche est constituée par des descripteurs, libres ou liés, sélectionnés suite à une visualisation du thésaurus, des descripteurs liés à un aspect, ...et pour chacun de ceux-ci, sa fréquence d'apparition dans le pool.

A tout moment de la session, il est possible  
d'ajouter,  
de supprimer,  
de remplacer un élément de la liste de recherche.

Le résultat d'une recherche peut aussi être stocké dans cette liste, sous un nom que lui donnera l'utilisateur.

#### Recherche primaire.

L'instruction de recherche est une équation booléenne reliant des numéros de lignes de la liste de recherche. Les opérateurs booléens sont ET, OU, ET PAS.

Remarque : pour la troncature, voir Mistral.

#### Recherche sur texte.

La recherche sur texte permet de faire une sélection plus détaillée des documents. Cette recherche ne porte que sur les paragraphes de type texte. Elle recherche la présence ou l'absence d'un mot ou d'une phrase dans un ou plusieurs des paragraphes textes, d'un document ou de tous les documents sélectionnés ensemble par une recherche primaire.

### Procédures d'aide à la recherche.

Les visualisations possibles sont les suivantes :

affichage du thésaurus,

affichage des aspects,

affichage des aspects et leurs descripteurs liés,

affichage d'une plage de descripteurs libres  
(environnement alphabétique d'un descripteur libre).

Il est aussi possible de spécifier les relations que Golem doit prendre en compte automatiquement pour ses recherches pendant une session ou même une partie de session.

### Procédures d'édition.

cfr Mistral.

### Accès direct au document.

Le système permet d'accéder directement à un document (sans utiliser de fichier inverse) au moyen de son numéro interne de document. Ce numéro est donné par Golem lorsque le document est entré dans la base documentaire.

## 2.5 Protection.

Des protections existent aux niveaux

des fonctions de chargement des documents,

de maj des documents,

de maj du dictionnaire,

de recherche de documents,

des fichiers des pools : nom identifiant le pool,

des documents : clé de protection,

des paragraphes : clé de protection.

Chaque utilisateur reçoit un numéro de passe qui lui donne accès :

à un ou plusieurs pools,

à une ou plusieurs fonctions pour chaque pool,

aux paragraphes et documents dont la clé de protection est autorisée,

dans ces paragraphes, certains aspects peuvent aussi avoir un niveau de protection.

Si la clé de l'utilisateur ne lui donne pas accès à ces aspects, il pourra quand même visualiser le paragraphe mais sans ces aspects particuliers. De même pour cet utilisateur, ces aspects avec leurs descripteurs liés n'apparaîtront pas lorsqu'il visualisera le thésaurus.

On remarquera qu'il n'y a pas de protection possible sur une valeur (descripteur libre ou lié).

Le niveau de protection est indiqué sous la forme d'information complémentaire du document, du paragraphe et par un code dans le thésaurus pour les aspects.

### 3. STAIRS ( IBM ).

#### 3.1 Architecture du système documentaire.

L'élément de base que Stairs manipule est le document. Ces documents sont stockés dans une base documentaire.

#### Organisation de la base de données.

Le système Stairs permet de créer dans un premier temps, une base de données "maîtresse" et ultérieurement une série de bases de données subordonnées (15 au maximum).

Ces dernières peuvent être concaténées à la première et constituer avec elle une nouvelle base maîtresse; l'ancienne base maîtresse devient alors une base subordonnée comme les autres.

Chaque base subordonnée peut être interrogée séparément, mais on peut aussi interroger l'ensemble des bases comme s'il ne constituait qu'une seule base.

Par exemple, on peut dans une base maîtresse contenant les enregistrements d'une année, grouper les enregistrements d'un mois dans une base subordonnée.

#### Le document.

Il est structuré en 5 niveaux hiérarchiques :

document lui-même, caractérisé par un nom (15 caractères maximum),

paragraphe : un document peut contenir un nombre quelconque de paragraphes; ils sont caractérisés par un code de 3 caractères,

phrase : un paragraphe peut contenir 256 phrases maximum,

ligne : un paragraphe peut en contenir 450 maximum,

mot : une phrase peut en contenir 256 maximum, une ligne peut en contenir 16 maximum; un mot peut avoir une longueur maximale de 63 caractères.

Le format des paragraphes est tout à fait libre, sauf pour les paragraphes "formatés" (cfr infra) dont la liste et la structure sont définies pour chaque base de données Stairs.

### 3.2 Instruments de références documentaires.

L'instrument de références documentaires utilisé par Stairs est le dictionnaire.

A toute base de données subordonnée est associé un dictionnaire. Ce dictionnaire contient tous les mots non vides qui se trouvent dans tous les documents enregistrés dans la base de données correspondante.

Tous ces mots pourront donc servir comme critère pour ce que nous avons appelé précédemment une recherche primaire (càd avec fichier inverse).

Il est possible de définir des relations de synonymie entre les différents mots d'un dictionnaire.

Stairs permet qu'on lui adapte un système optionnel qui l'améliore à 2 points de vue :

extension automatique des mots d'une équation de recherche (search) à toutes leurs formes flexionnées,

consultation et exploitation d'un thésaurus pour aider à constituer une équation de recherche (search).

#### Traitement des formes flexionnées.

Plusieurs étapes sont nécessaires :

génération des formes flexionnées de chacun des mots de l'équation. Les formes flexionnées régulières sont par exemple, le pluriel et le féminin pour les adjectifs, la conjugaison pour les verbes,...

Les formes flexionnées peuvent consister en modification de la racine des mots,

(ex : bon, meilleur)

suffixes,

(ex : bon, bons, bonne,...)

séparations de mots composés;

consultation du dictionnaire Stairs;

élimination de toutes les formes flexionnées qui n'existent pas dans le dictionnaire;

génération d'une équation de recherche étendue comportant toutes les formes flexionnées admises et existantes;

"passe la main" à Stairs pour la recherche.

### Consultation du thésaurus.

Les thésaurus ainsi constitués et mis-à-jour sont indépendants du dictionnaire Stairs.

Le thésaurus comprend des mots simples, composés,  
des expressions,  
des relations sémantiques.

La consultation du thésaurus permet l'accès aux fonctions suivantes :

fixation du ou des thésaurus que l'on souhaite consulter,

fixation des relations sémantiques que l'on souhaite être prises  
en compte,

entrée du descripteur au terminal et affichage par le système  
des termes reliés,

choix par l'utilisateur d'un ou plusieurs descripteurs à mémo-  
riser, parmi ceux qui sont affichés,

construction de l'équation de recherche,

"passe la main" aux fonctions classiques de Stairs.

### 3.3 Indexation.

Deux choses peuvent servir de critère à une recherche primaire.  
Ce sont les mots contenus dans les documents,

les paragraphes définis comme étant "formatés".

Dans le premier cas, l'indexation est semi-automatique, c'est-à-dire qu'après avoir extrait tous les mots des documents, on les compare à un dictionnaire des mots vides, fourni par l'utilisateur. Seuls les mots non vides sont retenus.

Pour les paragraphes "formatés", l'indexation est aussi semi-automatique. L'utilisateur a en effet dû définir un fichier définissant les paragraphes "formatés" dans le texte du document. Ceux-ci sont donc extraits et pris comme critères d'indexation.

### 3.4 Interrogation de la base documentaire.

Le sous-système de Stairs qui s'occupe de l'interrogation de la base documentaire s'appelle AQUARIUS. Il offre 8 fonctions principales dont 3 de recherche proprement dite et 5 autres, de service pour l'utilisateur.

#### Fonctions de recherche.

Liste de recherche.

La gestion de la liste de recherche dans Stairs est en tout point semblable à celle de Mistral.

Fonction Search.

Cette fonction constitue une recherche primaire (sur fichier inverse), qui se compose d'une équation booléenne.

Cette équation booléenne peut relier entre eux :

mots et/ou expressions booléennes; les mots sont ceux repris dans le fichier dictionnaire,

résultats antérieurs (étape de la liste de recherche).

Les opérateurs utilisables sont les suivants :

AND, NOT, OR, XOR, SYN, ADJ, WITH, SAME

ADJ : signifie pour a ADJ b que a et b doivent être adjacents dans l'ordre indiqué, dans la même phrase;

WITH : a WITH b signifie que a et b doivent être dans la même phrase;

SAME : a SAME b signifie que a et b doivent être dans le même paragraphe;

les autres opérateurs ont leur signification habituelle.

Cette recherche peut se faire sur une base ou sur plusieurs mais 16 au maximum.

La troncature est possible.

On peut limiter la recherche sur une ou plusieurs parties du document.

On peut aussi spécifier si la recherche doit porter sur des mots écrits en minuscules ou en majuscules.

### Fonction Select.

Cette fonction constitue une recherche primaire sur les champs déclarés "formatés".

Cette recherche se fait à l'aide d'opérateurs relationnels :

=, < , > , ≠ , ... .

Ces conditions relationnelles peuvent être combinées à l'aide des opérateurs booléens AND et OR.

Remarque : on utilise souvent la fonction Select après une recherche du type Search, pour "affiner" le résultat obtenu.

### Fonction Rank.

Cette fonction combine la recherche sur les mots du texte (Search) et le classement par ordre de pertinence décroissante des documents répondant à la question de recherche.

Ce classement peut se faire selon 5 algorithmes qui diffèrent entre eux par le poids qu'ils donnent

à la fréquence d'un mot dans chacun des documents considérés,

à la fréquence d'un mot dans chacun des documents et aussi dans l'ensemble des documents retrouvés,

aux documents dans lesquels intervient fréquemment un mot intervenant peu dans les autres documents.

### Fonctions de service.

Affichage au terminal, ou impression sur imprimante, en partie ou en tout

soit des documents retrouvés à partir d'une des 3 fonctions de recherche,

soit des documents de l'ensemble de la base de données.

Cette fonction permet aussi de mettre à jour on-line les champs dits "formatés".

Tri des documents retrouvés à partir d'une des 3 fonctions de recherche sur la valeur d'un champ "formaté", dans l'ordre ascendant ou descendant.

Stockage d'une ou plusieurs étapes de la liste de recherche dans un fichier.

Chaque question (étape) reçoit un nom qui sert à la rappeler ultérieurement pour l'effectuer sur d'autres documents.

Reprise et réexécution d'une ou des questions mémorisées.

Obtention des informations nécessaires à la compréhension du système et à l'interprétation des messages d'erreurs.

### 3.5 Protection.

Stairs permet de limiter l'accès à certains documents aux seuls utilisateurs habilités.

La protection fonctionne à différents niveaux :

    système d'exploitation : identification du terminal,

    système de contrôle : nom et mot de passe de l'utilisateur,

    Aquarius : nom de l'utilisateur,

    base de données : nom de la base et mot de passe,

    document : code de confidentialité,

    paragraphe : code de confidentialité,

    information "formatée" : code de confidentialité.

L'accès au document, au paragraphe ou à une zone "formatée" est autorisé ou non suivant le résultat de la comparaison du code de confidentialité de l'utilisateur avec celui associé au document, au paragraphe ou à la zone "formatée".

#### 4. SYSTEME DE RECHERCHE DOCUMENTAIRE DE L'IRETIJ.

L'IRETIJ est l'Institut de Recherche et d'Etude pour le Traitement de l'Information Juridique de Montpellier.

##### 4.1 Architecture du système documentaire.

###### Organisation de la base documentaire.

Les documents stockés dans la base documentaire sont répartis dans différents fichiers, selon le critère temps et ce pour des raisons techniques. Chaque fichier contient 65000 documents au plus.

La base comprend des documents de 3 types :

###### jurisprudence :

inédite : elle est limitée aux Cours d'Appel de Nîmes et de Montpellier. Sur l'ensemble des arrêts, on effectue une sélection selon l'intérêt du document; l'arrêt considéré inintéressant n'est pas retenu, l'arrêt présentant un intérêt régional est enregistré sous forme d'un abstract, l'arrêt d'un intérêt important est enregistré avec un abstract (cfr infra) et un résumé;

publiée dans les revues : concerne les arrêts de tous les tribunaux et toutes les Cours de France. On n'opère aucune sélection et tous les documents seront mémorisés par un abstract et un résumé;

###### doctrine :

elle est tirée des articles publiés dans des revues. On reprend tous les articles parus et on en fait seulement des abstracts;

###### réponses ministérielles :

elles sont également tirées des revues et feront uniquement l'objet d'un abstract.

###### Le document.

Le document de base est un bordereau constitué à partir du texte intégral de l'arrêt, de l'article, de la réponse ministérielle. Il se compose d'un ensemble de lignes dont certaines sont communes et d'autres spécifiques à chaque type de document.

On distingue d'abord une ZONE DE REFERENCE dont

1ère Ligne = clé de tri càd	
année	2 car
code juridiction	2 car
nro de séquence du document par source et par année	6 car

ces 6 caractères ont une signification différente selon le type de document :

pour l'appel, on a 2 car pour la Chambre,  
4 car pour le nro d'arrêt dans la Chambre;

pour la doctrine, on a 2 car pour le code de la revue,  
4 car pour le nro de document dans la revue;

pour une réponse ministérielle, on a  
1 car pour la Chambre ou l'Assemblée,  
5 car pour le nro de réponse ;

Ligne 10 = rappel de la date pour l'appel,  
une réponse ministérielle ;

Ligne 11 = juridiction pour l'appel,  
nom du ministre ou du parlementaire pour une  
réponse ministérielle ;

Ligne 15 = nom de l'auteur pour la doctrine ;

Ligne 20 = titre de l'article pour la doctrine .

Cette zone de référence est suivie d'autres parties :

Lignes 3A à 3Z = abstract,

Notons ici que seul cet abstract est interrogeable par l'intermédiaire d'un Fichier Inverse ;

Lignes 5A à 5Z = résumé ;

Ligne 60 = chaînage,  
on reprendra par exemple sur cette ligne la référence à la décision source, lorsqu'il s'agit d'un arrêt de Cassation ;

Ligne 70 = références bibliographiques  
pour la doctrine,  
une réponse ministérielle,  
càd titre de la revue  
et nombre de pages ;

Ligne 80 = nom des parties pour l'appel.

Remarques : la clé de tri est commune à toutes les lignes du bordereau et sera donc répétée en début de chaque ligne;

les seules parties obligatoires de ce document de base sont

la zone de référence,  
l'abstract.

#### 4.2 Instruments de références documentaires.

On distingue 2 types d'instruments de références documentaires basés sur des critères techniques : le lexique et le thésaurus de mots.

La différence de base entre ces 2 instruments s'explique comme suit :  
à tout élément du lexique, correspond une et une seule entrée dans le Fichier Inverse,

le thésaurus de mots reprend les relations entre éléments du lexique; il n'y a aucun rapport avec le Fichier Inverse.

##### Lexique.

##### Eléments.

A toute forme alphanumérique présente dans les abstracts, est affecté un numéro de notion.

Toutefois des regroupements ont été effectués  
sur base des variations grammaticales/syntaxiques  
(masculin-féminin-pluriel-formes verbales ...),

sur base de parenté sémantique :  
antonymie morphologique, synonymie parfaite.

Le lexique contient donc des couples : forme alphanumérique/numéro de notion.

##### Réalisation technique.

A tout champ sémantique distinct correspond une structure autonome appelée NOTION.

Cette unité structurale doit être décomposée en éléments indépendants mais substituables, constituant les SOUS-NOTIONS.

A chaque notion est attribué un numéro de 5 chiffres, affecté en séquence de 10 en 10. Au sein de chaque notion, toute sous-notion (variante grammaticale, antonyme ou synonyme) reçoit un chiffre en 5ème position.

Notons que l'interrogation peut porter  
sur la notion générale,

sur chaque sous-notion à l'exclusion des autres.

Exemple de structure :

		relation d'équivalence automatique
relation	21410	an, ans, année, années
de	21411	annualité, annuel, annuelle
substitution	21412	annalité, annal, annale
optionnelle ↓	21413	annuité, annuités
	⋮	
	32530	emploi, emplois
	32531	employé, employée, employés, employées
	32532	employer, employait, employant
	32533	employeur, employeurs

Remarques : les mots vides sont repris au lexique avec un numéro de notion NUL,

les valeurs numériques inférieures à 10000 ont leur valeur comme numéro de notion.

Traitement de la polysémie.

Pour les formes alphanumériques polysémiques,  
par exemple affection dans le sens tendresse  
ou

dans le sens maladie,

le lexique isolera le polysème dans une structure sémantique particulière. Les champs sémantiques différents (tendresse et maladie) sont traités séparément; la jonction entre champs distincts n'existe plus. Le bruit inhérent au polysème n'est pas supporté par les unités univoques. Par contre chaque champ sémantique est incomplètement représenté, ce qui fait apparaître un risque de silence certain.

Par exemple, le polysème "échoué" est distingué d'"échec" et d'"échouement" càd qu'ils se trouvent tous trois dans des notions différentes.

Le problème du silence résultant de l'isolement du polysème sera partiellement résolu par l'utilisation de l'autre outil linguistique, le thésaurus de mots.

#### Thésaurus de mots.

Le thésaurus de mots est un outil linguistique complémentaire d'aide à l'interrogation qui, à partir des notions du lexique, recense et complète les représentations formelles des champs sémantiques évoqués dans les notions.

L'objectif du thésaurus de mots est la formalisation de certaines

relations sémantiques pouvant exister entre les mots (du lexique ou autres) :

analogie (auto et camion  
disparition et décès),

antonymie générale (achat et vente),

relations entre polysème et ses allotaxiques :

du polysème vers ses univoques

culture  $\left\{ \begin{array}{l} \text{agricol} \\ \text{culturel,} \end{array} \right.$

de chaque univoque vers le polysème

agricol  $\rightarrow$  culture  
culturel  $\rightarrow$  culture.

Cette orientation particulière du thésaurus de mots est notamment due au fait que certaines relations sont déjà prises en compte dans le lexique :

synonymie parfaite (dommage et préjudice),

antonymie morphologique (créditer et discréditer).

Ce thésaurus de mots retrace 3 types de liaisons caractéristiques du langage orienté constituant le fichier :

des liaisons générales telles que  
accroissement — élévation, croissance  
                                   $\searrow$   
                                  diminution, réduction, baisse

des liaisons spécifiques telles que  
dissimulation — fraude, dol, tromperie, déguise, simulation

des liaisons ambivalentes telles que  
domicile — siège, établissement, résidence, demeure, foyer.

La conception restrictive qui a présidé à l'élaboration du lexique a amené à être très strict du point de vue de la synonymie et de l'antonymie.

Le thésaurus de mots s'applique donc à compléter le lexique de ce double point de vue et peut apparaître comme un dictionnaire analogique.

#### Caractère ouvert du lexique.

Le lexique n'est pas un outil a priori mais il se constitue progressivement. A chaque chargement de nouveaux documents dans la base documentaire, les valeurs alphanumériques encore inconnues jusque-là sont incorporées, sous certaines conditions, dans le lexique.

C'est l'utilisateur lui-même qui à chaque nouveau mot donne, s'il le désire, une entrée particulière dans le lexique :

il attribue un numéro de notion générale  
ou de sous-notion,  
existant ou nouveau selon que le terme est ou non synonyme  
d'un mot du lexique.

#### Caractère plutôt fermé du thésaurus de mots.

L'organisation et l'utilisation du fichier thésaurus étant tout à fait indépendantes de l'exploitation des autres fichiers du système, on peut considérer que le thésaurus de mots est fermé. Cependant il est parfois mis à jour par l'utilisateur quand celui-ci en éprouve la nécessité.

#### 4.3 Indexation.

##### Ce qui est interrogeable en recherche primaire.

Comme nous l'avons vu dans la description du document, seul l'abstract est interrogeable par l'intermédiaire du Fichier Inverse. Les abstracts sont rédigés par des analystes, chacun spécialisé dans un domaine particulier du droit.

L'abstract est un résumé quelque peu particulier; il résulte d'une analyse par synthèse du texte intégral. Il ne s'agit pas d'une réduction d'un document mais d'une décomposition et réorganisation sous une autre forme, du contenu de ce document. Il doit être écrit selon certaines règles, alliant la précision et la concision, reprenant des éléments de droit et des éléments de fait, ... (cfr "L'informatique appliquée à la jurisprudence" M. Bibent).

##### Structure de l'abstract.

L'abstract se subdivise en un certain nombre de paragraphes, chacun d'eux étant déterminé par un seul problème, une seule finalité et distinguant les questions de fond et de procédure.

Un paragraphe comporte un ensemble de phrases qui doivent être pertinentes, cohérentes et réalisent simultanément la fonction de classification et d'information.

Une phrase se compose d'un ou quelques mots exprimant en langage naturel un concept.

Les phrases sont séparées par des "/".

Extraction des descripteurs.

Le système extrait tous les mots de l'abstract et compare chacun d'eux au lexique. Il sort alors un listing reprenant tous les mots inconnus.

Ces mots ne figurent pas au lexique soit parce qu'ils comportent une faute d'orthographe, soit parce qu'ils sont nouveaux. Les premiers seront corrigés dans les textes tandis que les seconds seront introduits au lexique avec un numéro de notion attribué par l'utilisateur (cfr supra).

Le Fichier Inverse est constitué des couples :

numéro de notion (non nul) , ensemble des localisations.

Il est mis à jour à chaque introduction de documents dans la base, soit par une entrée nouvelle pour les mots inconnus jusque-là, soit par mise à jour des localisations pour les mots existants.

Remarque : pour certains fichiers prototypes, le résumé, bien que non structuré, est également interrogeable selon les mêmes principes, en recherche primaire.

4.4 Interrogation de la base documentaire.Procédures de service.

Elles permettent :

la connexion de l'utilisateur au système d'interrogation, avec vérification des mots de passe,

la spécification du fichier sur lequel il désire travailler, (par défaut, le dernier fichier en date sera le fichier courant).

Il est possible aussi de :

changer de fichier après une question, commande /FIC n

reprendre la question courante sur un fichier déterminé ou sur tous les fichiers de la base,

commandes /REP, /AUT, /REP n, /AUT n, /REP T, /AUT T.

Procédures d'assistance.

L'utilisateur peut à tout moment de la session demander l'affichage :  
du fichier courant /FIC

de la liste des fichiers disponibles /FIC ?

de la question en cours avec ses résultats ligne et bloc /QUE .

Procédures de recherche.

Le système permet d'effectuer des recherches en mode interactif.  
Elles sont de 2 types :

recherche sur fichier inverse,

recherche sur texte, appelée post-sélection.

La recherche sur texte ne peut s'effectuer que sur un ensemble de documents sélectionnés par une recherche primaire et porte sur tout le texte du document.

Par contre la recherche primaire s'effectue sur le fichier courant de la base.

Le résultat d'une instruction de recherche est un résultat chiffré représentant le nombre de documents sélectionnés par l'équation de recherche.

## Question- Recherche primaire.

La façon de traiter une question est assez différente des méthodes rencontrées chez Mistral, Golem et Stairs.

Une question de recherche se décompose en lignes.

Une ligne est une équation booléenne combinant des descripteurs devant être contenus dans le lexique.

Les opérateurs booléens utilisés sont ET, OU, SAUF; en cas d'absence entre 2 descripteurs, le ET est pris par défaut.

Le résultat d'une ligne est une valeur chiffrée représentant le nombre de documents dont l'abstract répond à cette équation.

Mais il est possible d'affiner la recherche en spécifiant l'unité de texte sur laquelle doit porter l'équation : la phrase, le paragraphe, l'arrêt. Par défaut, l'opérateur de niveau sera l'arrêt.

Chaque ligne est combinée aux autres lignes par un opérateur booléen pouvant également être le ET, le OU, le SAUF.

A la solution ligne, s'ajoute donc une solution question.

Une ligne de question a alors la structure suivante :

opérateur inter-ligne (ET par défaut)

opérateur de niveau (ARRET par défaut)

équation booléenne { descripteur/opérateur booléen (ET par défaut) }.

Si l'utilisateur désire visualiser les documents sélectionnés, la dernière ligne de sa question doit être le mot "FIN".

Remarque: un seul niveau de parenthèses peut être utilisé et uniquement aux niveaux phrase et paragraphe,

la troncature n'est pas possible en recherche primaire.

### Recherche sur texte.

La recherche sur texte, dite aussi post-sélection, permet de faire une sélection plus fine des documents résultats d'une recherche primaire.

L'ensemble du document est interrogeable en post-sélection et non plus seulement l'abstract.

La recherche sur texte peut se faire avant ou après l'édition des textes des documents sélectionnés par une question primaire.

Elle se caractérise par :

le fait que toutes les formes alphanumériques de la ligne de post-sélection doivent se trouver simultanément, mais dans n'importe quel ordre, dans la même phrase pour qu'un document soit sélectionné,

l'impossibilité de faire des post-sélections en cascade car celles-ci ne constituent pas réellement une ligne de question. Ainsi lorsqu'on désire reprendre la question après avoir fait de la post-sélection, les solutions de cette recherche sur texte sont ignorées.

Remarques : il est possible de tronquer les formes alphanumériques sur lesquelles porte la post-sélection,

une ligne de post-sélection est de la forme :  
FIN\*{forme alphanumérique}.

### Procédures d'aide à la recherche.

Ces procédures ont pour but d'aider l'utilisateur à rédiger de façon optimale ses équations de recherche.

Dans le système de l'IRETIJ, il n'y a qu'une procédure d'aide à la recherche qui est l'interrogation du thésaurus de mots.

A tout moment en cours de session, il est possible d'exploiter le thésaurus sur l'environnement sémantique d'un mot c-à-d ses analogues, antonymes,...

L'utilisateur peut alors décider de les intégrer ou non à sa question.

### Procédures d'édition.

Rappelons que pour visualiser les documents solutions d'une recherche, la dernière ligne de la question doit être le mot "FIN".

Pendant l'édition d'un texte, l'utilisateur peut :

demander le passage au texte suivant TS,

demander la reprise de l'affichage du texte en cours depuis le début. RT,

arrêter l'édition et passer à la question suivante AR.

Après un texte et avant l'affichage du suivant, on peut :  
 revenir à la question, notamment pour la compléter RQ,  
 reprendre la question à partir d'une ligne déterminée RQ n,  
 sauter l'édition de n textes S n.

#### 4.5 Protection.

Le système central se trouve au CNUSC (Centre National Universitaire Sud de Calcul) et est utilisable par plusieurs utilisateurs. Chacun d'eux reçoit un code comptable et un mot de passe associé. L'IRETIJ est un de ces utilisateurs possibles et de ce fait, ne peut interroger que d'un seul terminal à la fois.

Notons que les fichiers permanents du système sont propres à chaque utilisateur et reçoivent un nom qui est :  
 code comptable.identificateur de l'utilisateur.nom du fichier.

#### 4.6 Thésaurus de mots - Automatisation.

Nous avons traité ce problème séparément parce que le thésaurus de mots n'est plus intégré dans le système actuellement et que nous n'avons donc pas pu l'interroger.

Le thésaurus a été construit a priori, reprenant des mots du lexique et d'autres mais il n'a jamais été mis à jour.

C'est un fichier séquentiel indexé variable dont :  
 clé = forme alphanumérique,

structure d'un enregistrement logique =  
 /longueur totale/{longueur des formes alphanumériques/  
 type de relation/formes alphanumériques/} répétitif.

L'article logique est réalisé par plusieurs enregistrements physiques chaînés entre eux.

L'interrogation du thésaurus de mots s'effectue par la commande, THESA forme alphanumérique, qui renvoie l'environnement sémantique du mot. Cet environnement reprend ligne par ligne les analogues interchangeables entre eux.

Il est également possible de renvoyer aux analogues d'une autre notion, en ajoutant une ligne supplémentaire : voir analogues de la notion X.

Voyons sur 2 exemples la représentation de l'environnement sémantique.

Exemple 1 :

- (1) { affectueux  
affectueusement  
affection
- (2) { TH. - tendresse, amical, aimable  
- maladie, trouble, accident, incapable

(1) termes de même notion dans le lexique

(2) termes en relation dans le thésaurus de mots pour les 2 sens du polysème "affection".

Exemple 2 :

- grossier  
grossièreté
- { TH. - fin  
- épais  
- voir analogues de la notion "inconvenant"
- termes en relation dans le thésaurus de mots (antonyme "fin" et analogues).

5. BIBLIOGRAPHIE.

CII HONEYWELL-BULL;  
Mistral Manuel utilisateur,  
Guide de l'implémenteur,  
Guide de l'opérateur,  
Manuel de saisie,  
Documentation sur les applications.

GEORGES VAN SLYPE;  
Documentologie fascicule 4.

SIEMENS;  
Golem Documentation.

IBM;  
Storage and Information Retrieval System,  
Virtual Storage - General Information.

G. MAZET et C. BERNAD, 1976;  
Traitement par ordinateur des variables allotaxiques.

G. MAZET et C. BERNAD, 1978;  
Le thésaurus de mots.

THEMIS, 1976;  
Du rêve à l'instrument.

B. FILLIATRE, 1970;  
Conception et réalisation d'un système de documentation automatique pour  
la jurisprudence.

M. BIBENT, 1976;  
L'informatique appliquée à la jurisprudence.

ANNEXE 3 : Schéma relationnel DBASEII de la base de données du prototype.

Pour chaque relation DBASEII, on donnera le nom de la relation,  
 et pour chaque champ de la relation, on donnera  
 le nom du champ,  
 le type de valeur : C (caractère), N (numérique), L (booléen),  
 la longueur du champ.

La relation DBASEII PERSONNEIMPLIQUEE  
 représente le type d'entité PERSONNE et les types d'associations  
 CREANCIER, DEBITEUR-PRINCIPAL, CAUTION-ORIGINAIRE, CAUTION-NOUVELLE,  
 NOUVELLE-FORME ,

est caractérisée par

NOM	C	40
NROJUG	N	4
ROLE	C	20
DATE:DEB	N	6
DATE:FIN	N	6

La relation JUGEMENT  
 représente le type d'entité JUGEMENT ,

est caractérisée par

NROJUG	N	4
VILLE	C	20
DATE	N	6
RESULTAT	C	10
TYPE	C	9

La relation CASSINST  
 représente le type d'association CASSATION/INSTANCE ,

est caractérisée par

NROJUGINST	N	4
NROJUGCASS	N	4

La relation JUGCOMP  
 représente le type d'association APPARTENANCE-JUG ,

est caractérisée par

NROJUG	N	4
NROCOMP	N	4

La relation LIGNEJUG

représente la propriété TEXTE du type d'entité JUGEMENT ,

est caractérisée par

NROJUG	N	4
NROLIGNE	N	3
TEXTE	C	50

La relation COMPOSIT

représente le type d'entité COMPOSITION et les types d'associations COMPO-INDIRECTE, COMPO-DEPENDANTE ,

est caractérisée par

NROCOMP	N	4
TYPE	C	1 (peut prendre les valeurs 0 : originaire, I : indirecte, D : dépendante)
NROCOMPORIG	N	4 (numéro de composition originaire pour les compositions de types I et D)
EFFET	C	1
AFFIRME	L	1

La relation ELEMCOMP

représente le type d'entité FAIT&STANDARD-PONDERE et les types d'associations ANTECEDENT, CONSEQUENT, F&S/F&S-PONDERE ,

est caractérisée par

NROCOMP	N	4
NROFST	N	4
ANTECEDENT	L	1
ETATAFFIRME	L	1
PONDERATION	C	18

La relation FAITSTD

représente le type d'entité FAIT&STANDARD et le type d'association APPARTENANCE-CLASSE ,

est caractérisée par

NROFST	N	4
POLAR	C	7
NROCLASSE	N	2

La relation LIGNEFST

représente les propriétés TEXTE et TEXTE-NEGATION du type d'entité FAIT&STANDARD ,

est caractérisée par

NROFST	N	4
ETATAFFIRME	L	1
NROLIGNE	N	2
TEXTE	C	50

La relation CLASSE  
représente le type d'entité CLASSE ,

est caractérisée par

NROCLASSE	N	2
LIBELLE	C	150

La relation IDXJUG

a été introduite comme redondance par rapport aux relations ELEMCOMP  
et JUGCOMP pour des raisons de performance d'accès ,

est caractérisée par

NROFST	N	4
NROJUG	N	4

La relation MOTCLE

représente le type d'entité MOTCLE ,

est caractérisée par

NROMOT	N	4
TEXTE	C	40

La relation IDXMOT

représente le type d'association IDXMOT ,

est caractérisée par

NROMOT	N	4
NROFST	N	4

La relation SYNONYME

représente le type d'entité SYNONYME et le type d'association M-C/SYN ,

est caractérisée par

NROMOT1	N	4
MOT2	C	40

La relation ANALOGUE

représente le type d'association ANALOGUE ,

est caractérisée par

NROMOT1	N	4
NROMOT2	N	4

ANNEXE 4 : Schéma des Modules DBASEII du prototype.

Les modules sont de 2 types : les uns réalisent des menus (M:nom du module), les autres réalisent des fonctions (F:nom du module).

- 1 M:Programme principal
  - 2 M:Interrogation de la base JURIS-CAUTIONNEMENT
    - 3 F:Liste des jugements
    - 3 F:Liste des faits-et-standards
    - 3 F:Liste des classes
    - 3 F:Liste des faits-et-standards d'une classe
    - 3 F:Liste des jugements ayant une caractéristique particulière
    - 3 M:Renseignements sur un jugement particulier
      - 4 F:Courte description
      - 4 F:Personnes impliquées
      - 4 F:Type
      - 4 F:Compositions
      - 4 F:Jugement(s) associé(s)
    - 3 M:Renseignements sur un fait-et-standard particulier
      - 4 F:Intervenant dans une composition
      - 4 F:Intervenant dans une composition comme antécédent
      - 4 F:Intervenant dans une composition comme conséquent
      - 4 F:Intervention d'un groupe de faits-et-standards
    - 3 M:Recherche documentaire
      - 4 M:Visualisation du lexique
        - 5 F:Tous les mots du lexique
        - 5 F:Environnement alphabétique d'un mot (5 avant, 5 après)
        - 5 F:Mots commençant par une lettre
        - 5 F:Analogues d'un mot
      - 4 F:Recherche documentaire sur les faits-et-standards
      - 4 F:Recherche documentaire sur les jugements
- 2 M:Mise-à-jour de la base JURIS-CAUTIONNEMENT
  - 3 F:Mise-à-jour des jugements
  - 3 F:Mise-à-jour des faits-et-standards
  - 3 F:Mise-à-jour des classes
  - 3 M:Mise-à-jour des relations dans le lexique
    - 4 F:Relations de synonymie
    - 4 F:Relations d'analogie.

ANNEXE 5 : Exemples d'utilisation du prototype documentaire.

Cette annexe décrit un exemple de dialogue possible entre l'utilisateur et le système.

Elle donne au niveau de l'interrogation,

un exemple de liste prédéfinie :

liste des faits et standards d'une classe 5.5,

quatre exemples de recherches sélectives :

liste des jugements possédant une caractéristique particulière 5.6

recherche documentaire sur les faits et standards 5.8

renseignements sur un jugement particulier 5.13

renseignements sur un fait ou standard particulier 5.16

Elle donne également l'exemple d'ajout d'un jugement à la B.D. 5.18

Notons que dans les écrans qui suivent, les données fournies par l'utilisateur sont soulignées alors que le reste concerne toutes les informations présentées par le système.

Bonjour !

Vous commencez une session de travail avec le systeme ISAPAS ,  
qui gere la base de donnees JURIS-CAUTIONNEMENT .

Cette base de donnees contient deux types d'informations :

- des jugements ( cassation ou instance ) concernant des problemes  
de cautionnement
- des faits et des standards qui interviennent dans les jugements  
( pour la definition de ces concepts , cfr explication du systeme )

Vous pouvez faire deux sortes d'operations sur cette base de donnees :

- interrogation
- elargissement

A vous de jouer maintenant !

Desirez-vous voir l'explicitation du systeme ?

Si oui taper Yes , sinon taper No :N:

## MENU PRINCIPAL

\*\*\*\*\*

- 0 - Arrêt
- 1 - Interrogation de la base JURIS-CAUTIONNEMENT
- 2 - Elargissement de la base JURIS-CAUTIONNEMENT

Taper le chiffre correspondant a votre choix  
( Ensuite taper Return ):

MENU PRINCIPAL : choix 1

MENU D'INTERROGATION

\*\*\*\*\*

- 0- Arret
- 1- Liste des jugements existants
- 2- Liste des faits et standards existants
- 3- Liste des classes
- 4- Liste des faits et standards d'une classe
- 5- Liste des jugements possedant une caracteristique particuliere
- 6- Recherche documentaire sur les faits et standards
- 7- Renseignement(s) sur un jugement particulier
- 8- Renseignement(s) sur un fait ou un standard particulier

Taper le chiffre correspondant a votre choix .  
( Ensuite taper Return ):

MENU D'INTERROGATION : choix 4

Donnez le numero de la classe  
( Ensuite taper Return ! ) : 1

1 Cette classe reprend les faits relatifs au creancier .

FAIT NRO 9 AFFIRME

Le creancier a l'habitude de solliciter le renouvellement de la caution apres un changement de forme de la societe-debiteur principal.

FAIT NRO 9 NIE

Le creancier n'a pas l'habitude de solliciter le renouvellement de la caution apres un changement de forme de la societe-debiteur principal.

FIN

## MENU D'INTERROGATION : choix 5

Quelle est la caractéristique avec laquelle vous voulez interroger?  
Vous avez le choix entre le NUMERO , la VILLE , la DATE , le TYPE ,  
le RESULTAT , une PERSONNE intervenant dans un jugement.  
Taper le mot en majuscule correspondant a votre preference!  
( Ensuite taper return ! ) :type

Quel est le type ? Les deux valeurs possibles sont CASSATION ou INSTANCE .Taper  
l'une des deux ,svp .:instance :

## VOICI LA LISTE DES JUGEMENTS DE TYPE : INSTANCE

numero du jugement : 1  
ville ou le jugement a ete rendu : DOUAI  
date du jugement : 180577  
type du jugement : INSTANCE  
resultat du jugement : LIBERATION

Personnes impliquees :  
caution originaire : GUIDEZ PIERRE  
caution originaire : MAURIN COLETTE  
caution originaire : MALLET LUCIE  
caution nouvelle : LAMENDIN MONSIEUR  
caution nouvelle : LAMENDIN MADAME  
creancier : CREDIT LYONNAIS  
debiteur principal : SOCIETE CAMBRAI-AUTOMOBILE

Voulez-vous voir le texte de ce jugement ?  
Si oui taper Yes , sinon taper No !:y:

## CAUTION - CAUTIONNEMENT

Droits et obligations de la caution . Engagement.  
Duree. Societe . Cautionnement par le president-  
directeur general proprietaire du capital social  
et des membres de sa famille . Cession de l'entre-  
prise . Cautionnement par le nouveau dirigeant .  
Banque creanciere . Acceptation . Acte valant  
renonciation implicite à poursuivre les premieres  
cautions .

FIN

Voulez-vous executer une nouvelle interrogation avec la même caractéristique ou  
avec une autre ? Si oui taper Yes sinon taper No !:N:

MENU D'INTERROGATION : choix 6

MENU DE LA RECHERCHE DOCUMENTAIRE

\*\*\*\*\*

- 0- Sortie du menu
- 1- Visualisation du lexique
- 2- Question pour la recherche de faits et standards
- 3- Question pour la recherche de jugements

Taper le chiffre correspondant a votre choix  
( Ensuite taper Return ! ):

## MENU DE LA RECHERCHE DOCUMENTAIRE : choix 1

MENU DE VISUALISATION DU LEXIQUE  
\*\*\*\*\*

- 0- Sortie du menu
- 1- Visualisation de tous les mots du lexique
- 2- Environnement alphanumérique d'un mot du lexique  
( 5 mots precedents , 5 mots suivants )
- 3- Mots commençant par une lettre donnée
- 4- Analogues d'un mot repris au lexique

Taper le chiffre correspondant a votre choix  
( Ensuite taper Return ! ) : 2

Taper le descripteur tronque ou non : creancier

## VOICI L'ENVIRONNEMENT ALPHABETIQUE DU DESCRIPTEUR : CREANCIER

00024 clair  
00025 clause  
00026 commune  
00027 compte  
00028 contrat

\*\*\*\*\*  
00029 creancier

\*\*\*\*\*  
00030 credit  
00031 debiteur  
00032 demande  
00033 denomination  
00034 depart

## MENU DE LA RECHERCHE DOCUMENTAIRE : choix 2

VOICI LES DOMAINES POSSIBLES DE RECHERCHE :

- 1- les faits
- 2- les standards
- 3- les faits et les standards

DONNEZ VOTRE CHOIX : 3

Vous pouvez entrer votre question en donnant , par ligne :  
un operateur (et/ou/sf) et un descripteur

	OPERATEUR	DESCRIPTEUR	
LIGNE 1		: <u>cause</u>	:
LIGNE 2	: <u>ou</u> :	: <u>creancier</u>	:
LIGNE 3	: <u>sf</u> :	: <u>originaire</u>	:

Voulez-vous prendre en compte l'analogie ? (Y/N) :N:

VOICI LES RESULTATS DE LA RECHERCHE SOUS LA FORME :

etape	operateur	descripteur	nombre de faits et standards
1		cause	2
2	ou	creancier	20
3	sf	originaire	11

Suite a votre recherche, vous pouvez effectuer les operations suivantes :

- 0- quitter la question
- 1- visualiser la question
- 2- visualiser les faits et standards solutions
- 3- obtenir la liste des nros de faits-et-standards et jugements solutions
- 4- affiner la question
- 5- reprendre la question a une etape determinee

Donnez votre choix et ensuite tapez RETURN : 4

1		cause	2
2	ou	creancier	20
3	sf	originaire	11
LIGNE 4	<u>:et:</u>	<u>:dette</u>	:

VOICI LES RESULTATS DE LA RECHERCHE SOUS LA FORME :

etape	operateur	descripteur	nombre de faits et standards
1		cause	2
2	ou	creancier	20
3	sf	originaire	11
4	et	dette	1

Suite a votre recherche, vous pouvez effectuer les operations suivantes :

- 0- quitter la question
- 1- visualiser la question
- 2- visualiser les faits et standards solutions
- 3- obtenir la liste des nros de faits-et-standards et jugements solutions
- 4- affiner la question
- 5- reprendre la question a une etape determinee

Donnez votre choix et ensuite tapez RETURN : 3

Voici les numeros de faits et standards solutions et pour chacun d'eux , les numeros

des jugements qu'il indexe :

FAIT NRO 32 QUI N'INDEXE AUCUN JUGEMENT DE LA BASE DE DONNEES

La visualisation est terminee

## MENU D'INTERROGATION : choix 7

Connaissez-vous le numero du jugement sur lequel vous voulez obtenir des renseignements ?

Si oui taper Yes , sinon taper No !:n:

Connaissez-vous alors la date du jugement et la ville ou il a ete rendu ?

Si oui taper Yes , sinon taper No !:y:

Taper le nom de la ville ou fut rendu le jugement .:douai

Taper la date du jugement .:180577:

QUEL RENSEIGNEMENT DESIREZ-VOUS ?

- 0- Aucun renseignement
- 1- Courte description d'un jugement  
( ville , date , type , resultat )
- 2- Personnes impliquees
- 3- Type du jugement ( instance ou cassation )
- 4- Compositions d'un jugement
- 5- Jugement(s) associe(s)

Taper le chiffre correspondant a votre choix !  
( Ensuite taper return ! ) : 4

La signification des signes employes pour la representation des faits et des standards dans les compositions est la suivante :

F = un fait  
S = un standard

# = le fait ou le standard qu'il precede est a l'etat nie dans la composition ( si le signe est absent , le fait ou le standard est a l'etat affirme ! )

EXEMPLE :

!- F 23 !---->! # S 3 !

Cette expression signifie que :

- l' antecedent de la composition est le fait 23 a l'etat affirme
- le consequent de la composition est le standard 3 a l'etat nie

VOICI VENIR LA LISTE DES COMPOSITIONS !!!

Jugement rendu a DOUAI le :180577

COMPOSITION 1

\*NON\* !- F 23 !

---->! S 5 !

COMPOSITION 2

!- S 5 !

---->! #S 3 !

COMPOSITION 3

!- F 33 !----> F 24

!- F 37

!- F 30

!- F 26

! ---->! S 5 !

!

!

!

Voulez-vous voir le texte des compositions du jugement ?  
Si oui taper Yes sinon taper No , svp . Merci ! :N:

MENU D'INTERROGATION : choix 8

QUEL RENSEIGNEMENT DESIREZ-VOUS ?

- 0- Arrêt
- 1- Liste des compositions ou un fait ou standard intervient
- 2- Liste des compositions avec un fait ou standard comme antecedent
- 3- Liste des compositions avec un fait ou standard comme consequent
- 4- Liste des compositions ou un groupe de faits ou standards intervient

Tapez le chiffre correspondant a votre choix  
Et ensuite tapez RETURN : 2

Donnez le nro du fait ou standard  
Et ensuite tapez RETURN : 33

SI

FAIT NRO 33 AFFIRME

La caution originaire jouait un role important au  
de l'administration de la societe debiteur princi-  
pal avant son changement de role

ponderation : important

ALORS

FAIT NRO 24 AFFIRME

L'intention de la caution etait de maintenir ses  
engagements dans des limites raisonnables.

ponderation : important

---

La visualisation est terminee !

MENU PRINCIPAL : choix 2

MENU DES MISES-A-JOUR

\*\*\*\*\*

- 0- Arrêt
- 1- Mise-a-jour des jugements
- 2- Mise-a-jour des faits-et-standards
- 3- Mise-a-jour des classes
- 4- Mise-a-jour des relations dans le lexique

Taper le chiffre correspondant a votre choix  
(Ensuite taper Return):

MENU DES MISES-A-JOUR : choix 1

Ajout d'un jugement a la base de donnees .

\*\*\*\*\*

1- RENSEIGNEMENTS GENERAUX

ville : NAMUR :

date (Taper dans l'ordre JJMMAA ) : 100984:

type ( Les deux possibilites sont INSTANCE ou CASSATION ) : INSTANCE :

resultat : LIBERATION:

Si vous voulez encore changer quelque chose a ce que vous venez de taper , faites-le ,sinon taper un caractere pour continuer .:/:

Voici les renseignements generaux du jugement que vous ajoutez a la base

numero du jugement : 4

ville ou le jugement a ete rendu : NAMUR

date du jugement : 100984

type du jugement : INSTANCE

resultat du jugement : LIBERATION

Ce jugement est-il associe a un autre jugement ( cassation ou instance)

Si oui taper Yes sinon taper No , svp . Merci !:N:

Ajout d'un jugement a la base de donnees .  
 \*\*\*\*\*

## 2-PERSONNES IMPLIQUEES

## DEBITEUR PRINCIPAL

nom ( suivi du prenom ) : DURAND Paul :

date-debut : 120480:

date-fin : 300983:

## NOUVELLE FORME

nom ( suivi du prenom ) : LAURENT Pierre :

date-debut : 11083:

## CAUTION(S) ORIGINAIRE(S)

nom ( suivi du prenom ) : BRUNSON Jean :

date-debut : 10580:

date-fin : 240883:

## CAUTION(S) NOUVELLE(S)

nom ( suivi du prenom ) : PAQUOT Bernard :

date-debut : 200883:

date-fin : 0:

nom ( suivi du prenom ) : PAQUOT Yves :

date-debut : 200883:

date-fin : 0:

## CREANCIER(S)

nom ( suivi du prenom ) : CGER :

date-debut : 10580:

date-fin : 10585:

Voici les personnes et leurs caracteristiques que vous venez de taper .  
Si vous voulez modifier l'une ou l'autre de celles-ci , retenez en le numero

00020	debiteur principal 120480-300983	DURAND Paul
00021	nouvelle forme 11083- 0	LAURENT Pierre
00022	caution originaire 10580-240883	BRUNSON Jean
00023	caution nouvelle 200883- 0	PAQUOT Bernard
00024	caution nouvelle 200883- 0	PAQUOT Yves
00025	creancier 10580- 10585	CGER

Voulez-vous changer les renseignements portant sur une personne?  
Si oui , taper son numero , sinon taper 0 , svp . Merci !: 0:

Voulez-vous ajouter une personne impliquee dans ce jugement ?  
Si oui taper Yes , sinon taper No , svp . Merci !:N:

Ajout d'un jugement a la base de donnees .

\*\*\*\*\*

3- COMPOSITIONS DU JUGEMENT 4

COMBIEN Y-A-T'IL DE COMPOSITIONS ? : 1:

COMPOSITION 1

NEGATION CONDITIONNELLE (Y:N) : N:

EFFET : LIBERATION:

ANTECEDENT 1

NRO FAIT-ET-STANDARD: 27:

ETAT : NIE :

PONDERATION : IMPORTANT :

Voulez-vous donner un autre antecedent pour la composition originaire ? (Y/N) : N:

Ajout d'un jugement a la base de donnees .

\*\*\*\*\*

3- COMPOSITIONS DU JUGEMENT 4

COMBIEN Y-A-T'IL DE COMPOSITIONS ? : 1:

COMPOSITION 1

NEGATION CONDITIONNELLE (Y:N) : N:

EFFET : LIBERATION:

CONSEQUENT

NRO FAIT-ET-STANDARD: 16:

ETAT : AFFIRME:

PONDERATION : SUBSIDIAIRE :

Le consequent originaire est-il une composition ? (Y/N) : N:

Ajout d'un jugement a la base de donnees .  
\*\*\*\*\*

4- TEXTE DU JUGEMENT

Taper le texte du jugement avec pour derniere ligne le mot FIN !

```

:  CAUTION-CAUTIONNEMENT                               :
:droits de la caution . cautionnement de               :
:la societe par le proprietaire du capi-              :
:tal et ses parents . renonciation impli-            :
:cite .                                               :
:FIN                                                 :
:                                                         :
:                                                         :
:                                                         :
:                                                         :
:                                                         :
:                                                         :
:                                                         :
:                                                         :
:                                                         :

```

Si vous voulez encore modifier ce que vous venez de taper ,faites-le ;  
sinon taper un caractere pour continuer ,svp . Merci !:/:

Voulez-vous entrer un autre jugement ?  
Si oui taper Yes , sinon taper No:N:

ANNEXE 6 : Exemple de programmation en DBASEII.

\*Ce programme introduit une ou plusieurs relations de synonymie dans le lexique

ERASE

?

?"La synonymie est une relation qui est prise en compte automatiquement"  
?" lors de la recherche documentaire."

?

\* boolean qui dit s'il faut encore introduire une relation \*

STORE T TO CONT

DO WHILE CONT

STORE "

" TO SYN1

\* utilitaire qui efface la zone de l'ecran reservee aux messages \*  
DO EFFIN

@ 5,0 SAY "DONNEZ LE MOT DU LEXIQUE AUQUEL VOUS VOULEZ ASSOCIER DES;  
SYNONYMES:"

@ 6,10 GET SYN1  
READ

\* fichier lexique \*  
USE MOTCLE

\* verification de l'existence dans le lexique du mot-cle auquel on veut asso-  
cier des synonymes \*  
LOCATE FOR !(TEXTE)=!(SYN1)

IF EOF

@ 8,0 SAY "Ce mot n'est pas dans le lexique !"  
DO PTPAUSE

ELSE

STORE NROMOT TO NSYN1

\* utilitaire qui saisit les synonymes du mot entre et les introduit dans le  
fichier SYNONYME \*  
DO SAISSYN  
ENDIF

STORE S TO MIN

\* utilitaire qui efface une partie de l'ecran \*  
DO CLEAR

@ 22,0 SAY "Voulez-vous etablir une autre relation de synonymie ? (Y/N) ";  
GET CONT

READ

ENDDO

\* destruction des variables particulieres au programme \*

RELEASE CONT , NSYN1 , SYN1

RETURN