



HAVANE:Un systeme de mise en relation automatique de petites annonces

Patrick Bosc, Michèle Courant, Sophie Robin, Laurent Trilling

► **To cite this version:**

Patrick Bosc, Michèle Courant, Sophie Robin, Laurent Trilling. HAVANE:Un systeme de mise en relation automatique de petites annonces. [Rapport de recherche] RR-0223, INRIA. 1983. inria-00076335

HAL Id: inria-00076335

<https://hal.inria.fr/inria-00076335>

Submitted on 24 May 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

IRIA

CENTRE DE RENNES
IRISA

Institut National
de Recherche
en Informatique
et en Automatique

Domaine de Voluceau
Rocquencourt
BP 105
78153 Le Chesnay Cedex
France
Tél. (3) 954 90 20

Rapports de Recherche

N° 223

HAVANE : UN SYSTÈME DE MISE EN RELATION AUTOMATIQUE DE PETITES ANNONCES

Patrick BOSC
Michèle COURANT
Sophie ROBIN
Laurent TRILLING

Juillet 1983

HAVANE : un système de mise en
relation automatique
de petites annonces

Patrick BOSC
Michèle COURANT
Sophie ROBIN
Laurent TRILLING

RESUME :

HAVANE est un système de mise en relation automatique de petites annonces (P.A.) exprimées en langage naturel. Pour une P.A. donnée, HAVANE recherche dans une banque, les annonces "complémentaires" de celle-ci. De cette application "grand public" se dégagent trois grands axes d'étude :

- la définition du service, débouchant sur des problèmes liés à l'utilisation du langage naturel et le traitement du flou dans les bases de données,
- la paramétrisation du système par un expert es annonces et par suite, la conception d'un langage et d'une méthodologie pour la spécification des connaissances constituant la définition du système
- la réalisation du système en langage PROLOG.

Dans ce document, nous exposons successivement l'état des travaux et des réflexions concernant ces trois aspects.

ABSTRACT :

HAVANE is a system for the automatic processing of classified advertisements expressed in natural language. Its purpose is, for a given ad, to search the matching advertisements in a data base. Three investigation areas are issued from this application :

- the service definition, leading to problems related to natural language use and to fuzzy information processing in data bases,
- the specification of the application by an expert (in ads) and then the design of a language and a method in order to specify the required knowledge,
- the implementation of the system in the programming language PROLOG.

This paper details the work that has been undertaken and gives some ideas and trends concerning these three aspects.

SOMMAIRE

=====

1 - INTRODUCTION

- 1.1 - Cadre de l'étude
- 1.2 - Intérêts scientifiques
- 1.3 - Présentation générale du système
- 1.4 - Exposé

2 - CONCEPTION DU SERVICE

- 2.1 - Fonctions
- 2.2 - Service
- 2.3 - Choix relatifs aux fonctions
 - 2.3.1 - Saisie
 - 2.3.2 - Sélection des annonces complémentaires
- 2.4 - Idées

3 - SPECIFICATION

- 3.1 - Que doit spécifier l'expert?
- 3.2 - Le langage de spécification de la maquette
 - 3.2.1 - Origines
 - 3.2.2 - Le langage
 - 3.2.2.1 - Le descripteur de structure
 - 3.2.2.2 - Les grammaires d'analyse linguistique
 - 3.2.2.3 - Le langage de comparaison
- 3.3 - Idées

4 - REALISATION

- 4.1 - La base de connaissance
- 4.2 - Le système
 - 4.2.1 - L'analyse
 - 4.2.1.1 - Principe
 - 4.2.1.2 - Evaluation
 - 4.2.2 - La comparaison

4.3 - Idées

5 - CONCLUSION

BIBLIOGRAPHIE

ANNEXE 1: Spécifications de la maquette

ANNEXE 2: Exemple d'utilisation de la maquette

1 - I N T R O D U C T I O N

=====

1.1 - CADRE DE L'ETUDE

HAVANE (HAVas-ANnonCE) [1,2,3,4] est un système de compréhension et de consultation automatiques de petites annonces (P.A.) exprimées en langage naturel. Il fait suite au projet ANAGPA [5,6,7,8] conçu et mis en oeuvre par l'IRISA et le CCETT au cours des années 1977 à 1979, et a fait l'objet d'une collaboration IRISA-HAVAS.

Par rapport au service actuel relatif aux petites annonces, à savoir celui qui repose sur le support journal, le service offert par HAVANE présente plusieurs originalités:

- d'une part la mise en relation automatique de P.A., avec obtention non seulement des annonces correspondant exactement à une annonce donnée (cas d'ANAGPA), mais de toutes celles qui en sont "suffisamment proches", à l'aide d'une recherche dite "floue",

- d'autre part la constitution d'un fonds d'annonces, et par conséquent toutes les fonctionnalités offertes au niveau des banques de données (interrogation, stockage, "réactivation"), en temps réel et en temps différé.

De plus HAVANE réunit en un seul service les trois scénarios: "particulier à particulier", "particulier à agence", "agence à particulier".

Cette plus-value d'HAVANE par rapport aux supports classiques de P.A. et aux services existants, ne requiert en outre aucune modification des usages. La formulation des P.A. s'effectue en effet dans le langage usuel des P.A., et les annonces prélevées dans la banque sont restituées sous leur forme originale.

De la nature du service se dégagent plusieurs contraintes.

D'une part il s'avère nécessaire de concevoir un produit hautement évolutif. En effet, outre que l'utilisation d'un nouveau support peut transformer le langage des P.A., il convient de prévoir l'extension du système à plusieurs domaines et l'adaptation aisée d'un domaine à différents publics. Cet impératif d'évolutivité a pour conséquence la nécessité de paramétrer au maximum le système.

D'autre part, cette tâche de paramétrisation ne peut être assumée par le concepteur. Il apparaît donc que la charge en revient à un expert en annonces, disposant seul des connaissances suffisantes sur le domaine traité.

La conception de ce système a donné lieu à la réalisation d'une maquette du domaine immobilier réduit aux locaux à usage d'habitation. Les exemples utilisés dans la suite du document se rapportent à ce domaine.

1.2 - INTERETS SCIENTIFIQUES

Les intérêts scientifiques inhérents à l'application peuvent être répertoriés selon les trois "vues" du système: celles de l'utilisateur, de l'expert, et du réalisateur.

D'un point de vue utilisateur

Ils résident dans l'étude d'une application réelle, en l'occurrence la définition d'un service "grand public". Celui-ci comprend:

- la conception d'une structure de dialogues pour aboutir à la compréhension mutuelle du système et de l'utilisateur,
- un traitement du langage naturel:
 - en analyse, avec notamment la prise en compte de termes "flous" (ex: "grande maison"),
 - en synthèse, pour produire une paraphrase de la compréhension du système relative à la petite annonce fournie.
- pour une annonce donnée, la sélection dans une banque des annonces dites "complémentaires" de celle-ci, à travers une technique de recherche "floue".

D'un point de vue expert

La spécification du domaine d'annonces traité dans HAVANE, sous-tend:

- la définition d'un langage et d'une méthodologie de spécification à l'usage de l'expert, pour qu'il communique au système ses connaissances sur le domaine,
- une conception assistée visant à une spécification cohérente du domaine,
- une étude de la représentation des connaissances.

D'un point de vue réalisation

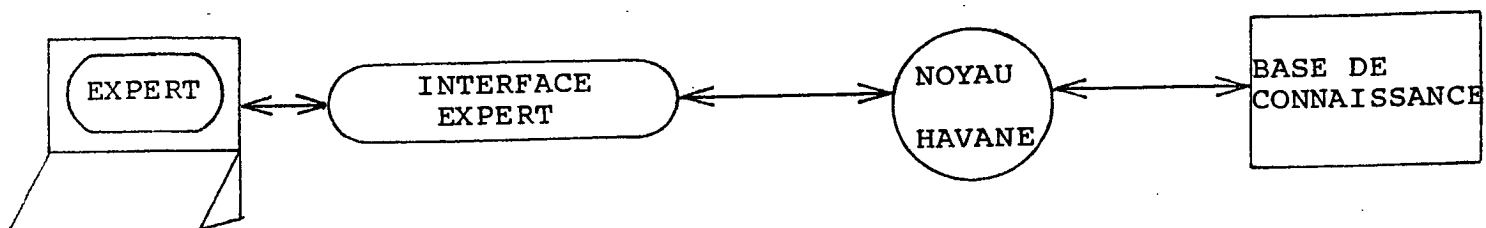
Les intérêts principaux se répartissent sur trois points:

- la compilation du langage de spécification,
- un aspect base données qui suppose des problèmes de structuration et de stockage des informations, avec en plus dans le cas présent, les spécificités suivantes:
 - la dualité des petites annonces, ayant valeur à la fois de question et de réponse (objet de la base),
 - la recherche "floue".
- l'étude de l'architecture du système.

1.3 - PRESENTATION GENERALE DU SYSTEME

Nous donnons maintenant à travers deux schémas une présentation informelle du système sous ses deux aspects, spécification et utilisation.

Définition du système par l'expert

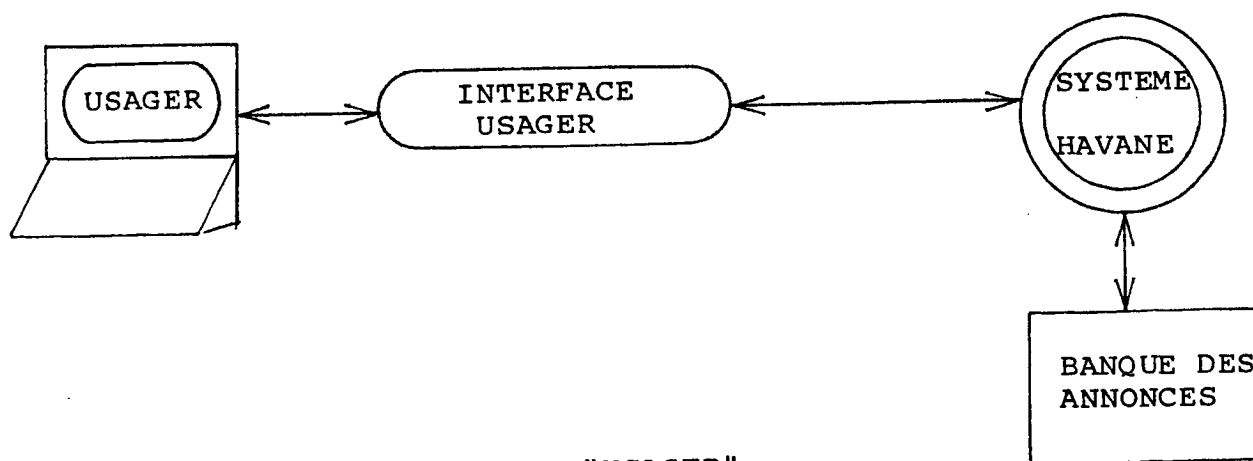


VUE "EXPERT"

Initialement la base de connaissance est vide.

En créant ou modifiant cette base de connaissance, l'expert définit le système qui comprend le noyau et le contenu de la base.

Le schéma général de l'application HAVANE est donc le suivant:



VUE "USAGER"

1.4 - EXPOSE

Le but de ce document est double: d'une part concrétiser les idées précédemment évoquées par une présentation de la maquette, d'autre part, étant données les grandes lignes du service, et la réalisation de la maquette, essayer de dégager les prolongements envisageables et retenus pour l'avenir du projet.

Ces deux points sont examinés dans chacun des trois chapitres qui reprennent successivement les différentes vues du système.

Le premier chapitre décrit les divers aspects de la conception du service. Nous y effectuons tout d'abord une analyse du service en terme de fonctions de base. Nous exposons ensuite un ensemble de scénarios à l'intention de l'utilisateur. Enfin nous reprenons une à une les fonctions, en soulignant pour chaque problème rencontré, les choix possibles.

La spécification est abordée dans le second chapitre, où après avoir introduit une démarche d'analyse d'un domaine d'annonces, nous proposons un langage de spécification pour l'expert.

Le dernier chapitre est consacré à l'aspect réalisation.

2 - C O N C E P T I O N D U S E R V I C E

=====

Etant donnés les objectifs d'HAVANE, à savoir la mise en relation automatique de petites annonces exprimées en langage naturel avec gestion d'une banque d'annonces, une première analyse met en évidence un ensemble de fonctions.

L'aspect conception du service est abordé dans ce chapitre, par la description de ces fonctions. L'articulation des fonctions nous amène ensuite à proposer quelques scénarios à l'intention de l'utilisateur. Puis nous énumérons les choix possibles en notant ceux effectués dans le cadre de la maquette. Enfin nous émettons quelques idées concernant l'extension du service.

2.1 - FONCTIONS

On distingue tout d'abord un ensemble de fonctions de base.

La saisie d'une annonce

La saisie d'une annonce permet son introduction dans le système.

Elle se traduit par la frappe d'une annonce par un utilisateur, suivie d'une proposition de paraphrase de l'annonce exprimant comment le système l'a comprise. La paraphrase doit recevoir l'agrément de l'utilisateur.

La sélection des annonces complémentaires

Cette fonction permet de rechercher dans la banque, les annonces susceptibles de satisfaire la requête d'un usager.

Cette requête étant formulée à travers une annonce dite "annonce courante", le système recherche dans la banque les annonces complémentaires de celle-ci.

Le stockage

Le stockage d'une annonce dans la banque se déroule comme suit:

L'utilisateur déclare ses nom, prénom, adresse, et numéro de téléphone. Le système associe ces informations au texte de l'annonce stockée et attribue à ce texte un numéro qui identifie l'annonce dans la banque.

La suppression

Cette fonction permet de supprimer de la banque d'annonces, un ensemble d'annonces de numéros donnés.

L'ensemble des fonctions de base étant fixé, il est possible de l'étendre à de nouvelles fonctions permettant d'ajouter quelques facilités au service fourni, telle la recherche par numéro.

La recherche par numéro

La recherche par numéro permet l'édition et la réutilisation d'un ensemble de textes d'annonces auparavant enregistrés dans le système. L'utilisateur ayant déclaré son identité, fournit les numéros des annonces et s'en voit éditer les textes respectifs, pour contrôle et utilisation éventuelle d'autres fonctions.

N.B.:

Dans la maquette, seules les trois premières fonctions ont été réalisées, le problème de la gestion de la banque (suppression) n'ayant pas été envisagé dans sa totalité.

2.2 -SERVICE

Entrée d'annonce sans consultation

Un utilisateur (notamment un agent immobilier ou notaire) désire introduire une ou plusieurs annonces dans la banque afin que celles-ci intègrent le marché (des offres et des demandes) et puissent être consultées par d'autres utilisateurs.

Scénario: L'utilisateur saisit une annonce et la stocke. Cette séquence (saisie, stockage) peut être répétée autant de fois qu'il le désire.

Consultation

Un utilisateur souhaitant réaliser une transaction, consulte la banque pour trouver des annonces susceptibles de l'intéresser. Cette consultation s'effectue via une petite annonce dans laquelle l'utilisateur formule ses desiderata.

Scénario: L'utilisateur saisit son annonce, recherche dans la banque les annonces complémentaires de celle qu'il vient de formuler, puis éventuellement stocke son annonce.

Relance d'une consultation

Un utilisateur ayant antérieurement stocké son annonce dans la banque, souhaite reconsulter la banque d'annonces.

Scénario: L'utilisateur fournit le numéro de son annonce pour retrouver l'annonce concernée, puis recherche dans la banque les annonces "complémentaires" du moment, entrées depuis sa dernière consultation.

Suppression d'annonces

Un utilisateur peut vouloir faire disparaître une annonce de la banque, par exemple si celle-ci a fait l'objet d'un accord.

Scénario: L'utilisateur entre le numéro de son annonce, puis commande sa suppression.

L'analyse des scénarios ci-dessus, en termes de fonctions s'énonce donc comme suit:

- "entrée d'annonce sans consultation": "saisie" suivie de "stockage",
- "consultation": "saisie" suivie de "sélection des annonces complémentaires", suivie éventuellement de "stockage",
- "relance d'une consultation": "recherche par numéros" suivie de "sélection des annonces complémentaires",
- "suppression d'annonces": "suppression".

N.B.:

Les services "relance d'une consultation d'annonces" et "suppression d'annonces" n'ont pas été retenus au stade de la maquette.

2.3 - CHOIX RELATIFS AUX FONCTIONS

Dans ce paragraphe sont examinés les problèmes rencontrés lors de la réalisation des fonctions et mentionnés pour chacun d'eux les choix possibles. Seules la saisie et la sélection des annonces complémentaires sont ici étudiées, les autres fonctions n'ayant donné lieu à aucun choix.

2.3.1 - Saisie

Le principe de la saisie des annonces repose sur l'utilisation du langage naturel.

Après avoir formulé son annonce, l'utilisateur s'en voit proposer une paraphrase résumant la sémantique retenue par le système. Ce dernier n'estime avoir bien compris qu'après validation d'une telle paraphrase par l'utilisateur.

L'analyse du langage naturel et la production de paraphrase pose un certain nombre de difficultés successivement considérées dans la suite du paragraphe. Pour chacune d'elles nous exposons les différentes alternatives et indiquons le cas échéant les choix effectués pour la maquette.

A - ANALYSE

1) Fautes d'orthographe

a) Les fautes sont récupérées: le système comprend et corrige automatiquement les mots mal orthographiés.

b) Les mots mal orthographiés ne sont pas compris par le système.

Pour des raisons de robustesse il est indispensable que le problème des fautes d'orthographe soit traité en soi.

Néanmoins il n'a pas été abordé de façon spécifique dans la maquette. L'expert peut toutefois spécifier des synonymes, donc prendre en compte des mots mal orthographiés répertoriés à l'avance.

2) Ambiguïtés dans le texte de l'annonce

L'analyse d'une P.A. exprimée en langage naturel peut se heurter à des problèmes d'ambiguïtés inhérents à l'énoncé. Ces ambiguïtés peuvent notamment se présenter sous forme de chevauchement propre (pas d'inclusion) de chaînes isolément

interprétables par le système (cf. 4.2.1.2); autrement dit les spécifications de l'expert conduisent à interpréter une partie de l'annonce de deux façons différentes.

Exemple:

C'est le cas d'une annonce comprenant:
"...terrain 250 m² habitable...", pour laquelle deux interprétations sont possibles. Il s'agit:
-soit, d'un local d'une superficie habitable de 250 m², avec un terrain,
-soit, d'un local habitable avec un terrain de 250 m².

a) Dès la détection d'un tel chevauchement l'utilisateur est sollicité pour signaler quelle est l'interprétation à retenir: il y a "focalisation" du dialogue.

b) L'utilisateur n'est pas sollicité.

Le but étant d'aboutir rapidement à un accord entre système et utilisateur, dans la maquette nous avons choisi de mettre en oeuvre un dialogue focalisant.

3) Contradictions dans le texte de l'annonce

L'utilisateur a pu introduire dans le texte de l'annonce deux éléments que le système considère comme contradictoires, conformément aux spécifications de l'expert.

Exemple:

C'est le cas d'une annonce comprenant:
"Cherche maisonnette 4 pièces...", si l'expert décide qu'une maisonnette comporte au plus 4 pièces.

a) L'utilisateur doit préciser par dialogue l'information correcte. Cette solution introduit alors un nouveau problème. En effet, la correction (ou l'apport) d'information par l'utilisateur entraîne une non-conformité de la compréhension du système par rapport au texte initialement fourni; ce qui amène la nécessité d'une reformulation automatique de l'annonce par le système.

b) L'utilisateur voit son annonce rejetée. Il est informé de la contradiction et invité à reformuler une nouvelle annonce.

L'idée est toujours de mettre en place un dialogue focalisant.

Cependant la reformulation de l'annonce (cf. 2.4) n'ayant pas été envisagée au stade de la maquette, il s'ensuit que les annonces comportant des contradictions sont rejetées.

B - PARAPHRASAGE

L'idée du paraphrasage est de refléter au mieux la compréhension du système. Il doit par conséquent être le plus complet possible.

De plus il paraît bon de reprendre au maximum les termes de l'utilisateur, fidélité dont on peut attendre qu'elle crée un terrain psychologique favorable entre l'usager et le système.

La soumission d'une paraphrase à l'usager peut donner lieu à plusieurs types de désaccords que nous allons tout d'abord inventorier avant de proposer les alternatives s'y rapportant.

1) Information jugée incorrecte par l'usager

1er cas:

Conformément aux spécifications de l'expert, le système intervient pour compléter une annonce insuffisamment informée. Il formule donc une hypothèse relative à l'élément manquant. L'usager se trouve être en désaccord avec cette hypothèse.

Exemple:

L'expert décide que pour une annonce ne contenant aucune information sur la nature de la transaction, il s'agit d'une transaction de type "achat-vente" (par opposition à "location").

2ème cas:

L'utilisateur s'aperçoit qu'il s'est trompé au sujet d'un élément de l'annonce.

2) Paraphrase jugée incomplète par l'usager

Il manque une information. Deux cas peuvent se produire.

1er cas:

L'expert a décidé de ne pas prendre en compte cette information.

2ème cas:

Il s'agit d'un oubli de la part de l'usager lors de la formulation de son annonce.

Les choix possibles pour la résolution de ces désaccords sont les suivants:

- a) Mise en place d'un dialogue focalisant permettant d'arriver à un accord entre système et usager.
- b) Pas de dialogue focalisant.

Ainsi que pour les problèmes énoncés antérieurement, l'idée d'engager des dialogues focalisants demeure.

Nous avons cependant opté pour la seconde solution dans la maquette; les cas de désaccords de l'utilisateur avec la paraphrase soulèvent en effet également le problème de la reformulation de l'annonce.

2.3.2 - Sélection des annonces complémentaires

Trois principes contribuent à la définition du terme "annonce complémentaire":

- l'origine des critères de recherche,
- la méthode de comparaison,
- le nombre d'annonces fournies.

1) Origines des critères de recherche

Les critères de recherche des annonces complémentaires peuvent avoir différentes origines.

a) L'utilisateur est sollicité pour apporter des précisions sur les annonces qu'il recherche (ex: éléments auxquels il accorde beaucoup d'importance...).

b) Aucune information supplémentaire n'est demandée à l'utilisateur en plus du texte de son annonce.

La deuxième solution a été retenue dans la maquette, ceci pour des raisons ergonomiques (optique de dialogue minimum).

De plus la mise en oeuvre de la première solution suppose la possibilité pour l'utilisateur d'interroger le système quant à l'exploitation qui est faite des informations qu'il donne (cf. 2.4).

2) Comparaison entre annonces

La recherche des annonces complémentaires peut être réalisée selon deux méthodes de sélection, dont les principes sont:

a) Un score global attribué aux annonces contenues dans la banque. Ce score est totalisé à partir de scores élémentaires calculés pour chacun des éléments de l'annonce.

b) Une technique de seuil:

Un seuil se caractérise par un ensemble d'écart maximums tolérés sur chacun des éléments des annonces. Dès qu'un écart est dépassé lors de la mise en relation de deux annonces, les annonces considérées sont dites non complémentaires.

La seconde solution a été préférée pour la maquette. La comparaison des annonces est ainsi envisagée comme un ensemble de comparaisons élémentaires indépendantes sur les différents composants des annonces. La première méthode en revanche, exige de l'expert qu'il spécifie une comparaison globale entre annonces, ce qui suppose qu'il soit capable d'apprécier les

importances relatives des différents composants.

3) Nombre d'annonces complémentaires fournies

Le nombre de réponses délivrées à l'utilisateur dépend de critères divers d'ordre essentiellement psychologique ou commercial.

a) L'utilisateur reçoit de toute façon, un nombre de réponses compris dans une fourchette, donc éventuellement assez éloignées de ses desiderata, s'il n'y a rien de mieux dans la banque. La garantie de ce nombre minimum doit être assurée en vertu d'arguments commerciaux.

b) L'utilisateur se voit restituer des annonces jugées "suffisamment satisfaisantes" par l'expert.

La méthode de comparaison basée sur un calcul de score permet d'ordonner facilement les annonces. Elle se révèle ainsi très adaptée à la première solution. En revanche la méthode du seuil convient mal à cette solution. Il paraît en effet difficile d'assurer la convergence du nombre d'annonces complémentaires obtenues par une telle méthode, vers les limites d'un intervalle prédéfini. Pour pallier cette insuffisance, on peut alors songer à solliciter des compléments d'information de la part de l'utilisateur.

Compte tenu de la taille de la banque, seule la deuxième solution paraissait raisonnable pour la maquette. Toutefois pour éviter de délivrer un nombre trop restreint d'annonces à l'utilisateur, l'expert définit plusieurs seuils. Le processus de sélection des annonces complémentaires est lancé successivement pour chacun de ces seuils, jusqu'à satisfaction de l'utilisateur ou épuisement des seuils.

Lors de l'édition des annonces, l'utilisateur se voit donc proposer les annonces les plus satisfaisantes (seuil le plus restrictif), puis s'il le désire, des annonces un peu plus éloignées de la sienne, etc.

2.4 - IDEES

"Statut d'une annonce"

Différents services pourraient être proposés à l'utilisateur par l'attribution à chaque petite annonce d'un statut composé de deux caractéristiques booléennes ("recherche" et "don") à deux modalités ("fugace" et "pérenne").

"recherche" = "vrai" si l'utilisateur désire obtenir des annonces en réponse à sa P.A..

"don" = "vrai" si l'utilisateur autorise la divulgation de sa P.A. aux autres usagers.

La "fugacité" représente le temps d'une consultation (court terme).

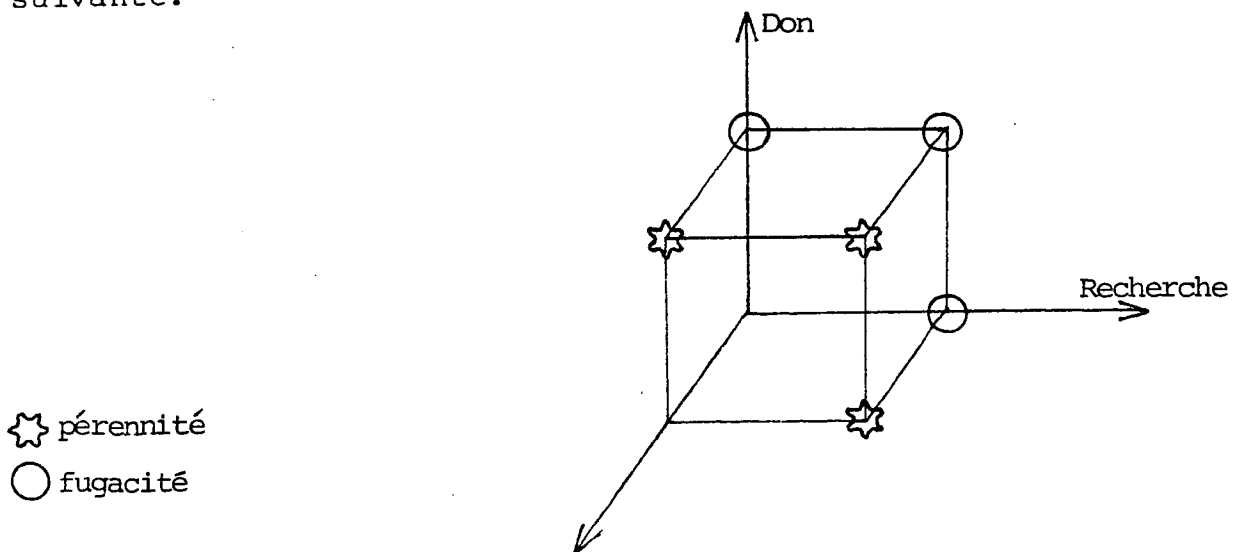
La "pérennité" correspond à une durée de stockage (long terme).

Exemple:

L'utilisateur choisit de faire une recherche pérenne et pas de don. Il recevra donc, lors d'une consultation, les annonces complémentaires de la sienne, et les annonces qui ont obtenu, le temps du stockage, son annonce comme annonce complémentaire (il y a constitution d'un "historique" de la P.A.).

En spécifiant qu'il ne fait pas de don, il s'oppose à la divulgation de son annonce aux annonceurs dont il a obtenu l'annonce comme annonce complémentaire.

La notion de statut peut être représentée de la façon suivante:



Définition d'une interface ergonomique avec l'utilisateur

Cette interface comprend:

- une aide à l'utilisateur pour expliquer à celui-ci les critères (ceux de l'expert) en vertu desquels son annonce a été comprise et comparée, afin qu'il puisse éventuellement la reformuler en connaissance de cause,
- des facilités de composition de texte avec notamment possibilité de correction des fautes d'orthographe et de modification du texte de l'annonce,
- l'utilisation de terminaux vidéotex alpha-géométriques qui offrent une meilleure définition que l'alpha-mosaïque et incluent des possibilités de télédessin,
- la possibilité de saisie et de restitution vocales d'annonces.

3 - S P E C I F I C A T I O N

=====

En un premier temps ce chapitre fait le point sur les spécifications requises par le système vis à vis de l'expert. Il tente également de classer les besoins afin d'aboutir à une expression, encore relativement abstraite, de la démarche de l'expert.

En deuxième partie, il présente le langage mis à la disposition de ce dernier dans la maquette, avant de recenser quelques idées d'amélioration issues de cette expérience.

3.1 - QUE DOIT SPECIFIER L'EXPERT?

Nous exposons ici une démarche de spécification pour l'expert. Celle-ci a pour but d'aider un expert à transmettre au système des connaissances(elle n'a pas valeur de méthodologie d'analyse du domaine traité).

Il apparaît cependant que l'analyse d'un domaine exige une étude lexicographique préliminaire; ceci afin d'extraire du langage usuel des P.A., les éléments linguistiques intéressants, et de regrouper les différentes formulations correspondant à un même élément sémantique.

Le but étant pour une petite annonce, de savoir rechercher dans une banque les annonces complémentaires, l'expert doit:

a) déterminer en premier lieu l'ensemble des unités significatives au sens de la comparaison de deux annonces, appelées UNITES DE COMPARAISON (UC):

- soit EUC cet ensemble,
EUC = { UC_i };

- il associe à chaque UC_i un n-uplet d'attributs:
UC_i(A_{i1}, A_{i2}, ..., A_{ini});

- pour chaque attribut A_{ij}, il précise son domaine de valeurs, soit:

D(A_{ij}) = { valeurs }
V i, j D(A_{ij}) ≠ ∅.

Exemple:

Une annonce immobilière peut comporter des éléments linguistiques tels que " 3 pièces ", "entre 5 et 6 pièces ", qui concrétisent tous une information relative au nombre de pièces. L'expert peut donc définir une UC telle que:
nbpièces = (nbpl, nbp2) avec D(nbpl) = N et D(nbp2) = N, où la présence des deux attributs permet de rendre compte de

toutes les formulations à travers un intervalle.

Remarques:

* Une comparaison élémentaire s'effectuant par définition sur deux U.C., l'expert est amené à rassembler sous forme d'attributs d'une même U.C., plusieurs informations élémentaires, dès lors qu'il souhaite faire intervenir simultanément ces informations dans une comparaison.

Au contraire la scission de ces informations en plusieurs U.C., limite le degré de finesse des comparaisons.

* L'expert peut prévoir pour certains attributs une valeur spéciale signifiant "zéro information" pour le cas où le libellé de l'annonce ne comporte aucune information relative à cet attribut.

b) mettre en évidence l'ensemble des COMPREHENSIONS ADMISSIBLES

Définitions

Unité de comparaison valuée $UCVi = UCi(x_{i1}, x_{i2}, \dots, x_{ini})$
où $x_{ij} \in D(A_{ij})$.

Compréhension: $C = P(EUCV)$ où $EUCV =$ ensemble des U.C. valuées.

Ensemble des compréhensions: $EC = \{ C \}$.

Ensemble des compréhensions admissibles: $ECA \subset EC$

Cet ensemble ne retient d'EC, que les associations d' $UCVi$ représentant la sémantique des annonces jugées cohérentes par l'expert.

Exemple 1:

En analysant le domaine des locaux à usage d'habitation, l'expert dégage les UC suivantes:

-désignation(DO)

où $D(DO) = \{\text{appartement, maison, villa, etc.}\}$,

-terrain(PT) où "PT" (pour "présence terrain") est tel que $D(PT) = \{\text{oui, non, pt-zéro-info}\}$.

Cette dernière UC n'a bien entendu pas de sens pour les annonces concernant les appartements.

Par conséquent dans ce cas une compréhension incluant la combinaison d'UC valuées désignation(appartement) et terrain(oui) n'appartient pas à ECA; en revanche la combinaison d'UCV désignation(maison) et terrain(oui) appartient à ECA.

Remarque:

La présence simultanée de deux informations exclusives au regard des compréhensions admissibles, peut être interprétée comme une contradiction.

Exemple 2:

L'expert souhaite préciser qu'une annonce ne comportant aucune information relative à la désignation de l'objet, n'est pas une annonce admissible. Toute compréhension incluant désignation("zéro information") n'appartient donc pas à ECA.

c) Après avoir caractérisé l'ensemble des compréhensions admissibles, l'expert poursuit sa démarche dans quatre directions: l'analyse, le paraphrasage, les dialogues, et la comparaison.

L'ANALYSE

A partir d'un texte d'annonce, le but est d'arriver à une compréhension (ensemble d'UC valuées) qui constitue la représentation interne qu'a le système de l'annonce.

L'expert doit donc établir le lien entre informations textuelles et valeurs des attributs d'UC. Ceci doit le conduire à exprimer le calcul des attributs des UC à partir des éléments linguistiques figurant dans le texte de l'annonce.

Exemple1:

L'occurrence de la chaîne " 4 à 5 pièces " dans une annonce se traduit par $nbpl=4$ et $nbp2=5$.

Exemple2:

L'expert spécifie que si l'objet de la transaction est une maisonnette, et que le nombre de pièces n'est pas précisé, alors les attributs de nbp sont tels que: $nbpl=2$, $nbp2=3$.

LE PARAPHRASAGE

La spécification de la paraphrase d'un texte d'annonce consiste à en calculer les composants (également modélisés par des attributs) à partir des éléments linguistiques relevés dans le texte de l'annonce et de la représentation interne de l'annonce (ou compréhension).

Exemple:

L'expert décide que l'élément de paraphrasage associé à la chaîne "maison individuelle" est "une maison individuelle" et que celui associé à la chaîne "n pièces" est "comprenant n pièces".

Remarque:

Il semble que lors de la définition du paraphrasage, l'expert ait intérêt:

- d'une part à reprendre au maximum les termes employés par l'utilisateur,
- d'autre part à rendre la paraphrase la plus complète possible.

LES DIALOGUES

Il s'agit d'une part de dialogues visant à lever les ambiguïtés, d'autre part de dialogues mis en oeuvre lorsqu'un texte d'annonce ne cadre avec aucune des compréhensions admissibles décrites par l'expert.

LA COMPARAISON

L'expert construit les couples d'UC comparables; les composants de ces couples appartiennent, l'un à l'ensemble des UC représentant une annonce "offre", l'autre à l'ensemble des UC représentant une annonce "demande". Puis il associe à chaque UC une fonction donnant les distances entre les valeurs prises par les attributs de cette UC, aux valeurs des attributs des UC comparables, ainsi que pour chacun des niveaux de tolérance, l'écart maximum toléré.

Exemple:

Pour le couple d'UC (désignation, désignation), l'expert spécifie que, la distance entre les valeurs de leurs attributs respectifs est:

- pour "maison" et "maison": 0,
- pour "maison" et "villa": 3,
- pour "maison" et "appartement": 8.
- pour "villa" et "villa": 0,
- pour "villa" et "maison": 1,
- pour "villa" et "appartement": 5,
- pour "appartement" et "appartement": 0,
- pour "appartement" et "maison": 8,
- pour "appartement" et "villa": 10,

Les distances maximums tolérées pour ce couple sont:

- au seuil 0: 0,
- au seuil 1: 4,

ce qui implique que deux annonces traitant respectivement d'une offre de maison et d'une demande de villa ne sont pas complémentaires au seuil \emptyset , en revanche elles le sont au seuil 1.

Remarque:

La comparaison est ainsi fondée sur la complémentarité offre/demande, ce qui répond effectivement à la réalité du domaine traité dans la maquette.

Néanmoins des domaines de P.A. tels que les "rencontres" et "divers" (échanges...) ne cadrent pas avec ce schéma.

3.2 - LE LANGAGE DE SPECIFICATION DE LA MAQUETTE

Avant de présenter le langage conçu à l'intention de l'expert au stade de la maquette, ce paragraphe commence par situer le cadre de sa conception, autrement dit les antécédents auxquels se raccroche notre propre réflexion. Les différents outils qui se sont ainsi vus induits de proche en proche, sont ensuite décrits à travers un exemple emprunté au domaine immobilier.

3.2.1 - Origines

Le projet ANAGPA a déjà donné lieu à la réalisation de plusieurs maquettes basées sur des algorithmes différents [6,7].

Pour mettre en oeuvre la méthodologie précédemment décrite il nous apparaît essentiel, au vu de ces précédentes expériences de n'utiliser qu'un seul langage de spécification pour définir les compréhensions admissibles, l'analyse et le paraphrasage. Ce langage "non procédural" est construit sur la notion de grammaire d'attributs [9]. En fait comme nous le verrons dans ce qui suit, nous avons défini des grammaires adaptées au problème en ce sens que:

- les attributs peuvent être à la fois hérités et synthétisés. Les grammaires de métamorphose [10] autorisent ce type d'attributs.
- un petit nombre de conditions élémentaires (égalité d'attributs, appartenance à un ensemble...) sur les attributs a pu être déterminé pour rendre compte de leurs relations.

Le langage proposé possède ainsi deux constituants:

- un constituant grammatical proche des grammaires de métamorphose,
- un lot de prédicats tels que l'appartenance à un ensemble, etc.

A partir d'un texte d'annonce, les grammaires de métamorphose permettent aisément d'en construire une représentation (ou "structure") interne utilisable à des fins de comparaison d'annonces. La représentation ainsi produite consiste en une arborescence dont les noeuds et les feuilles portent des attributs. Certains noeuds sont des unités de comparaison; l'ensemble des U.C. valuées de l'arborescence représente une compréhension admissible.

Nous nous appliquerons dans le paragraphe suivant à retrouver les étapes de la méthodologie décrite précédemment dans cette première version du langage, très inspirée de la représentation arborescente des annonces.

3.2.2 - Le langage

Le langage comprend trois outils que nous étudions successivement:

- le DESCRIPTEUR DE STRUCTURE qui spécifie l'ensemble des structures internes correspondant à des compréhensions admissibles pour le domaine considéré;
- les GRAMMAIRES D'ANALYSE LINGUISTIQUE (AL) qui établissent les liens entre les informations textuelles et les attributs des feuilles de la structure;
- le LANGAGE DE COMPARAISON qui définit les unités de comparaison, les couples d'UC comparables et les fonctions de comparaison associées.

Remarque:

La construction de la paraphrase, également réalisée par du calcul d'attributs est assurée à la fois par le descripteur de structure et les grammaires d'AL. Celle-ci bien qu'introduite dans les règles du descripteur de structure, ne se trouve commentée qu'au paragraphe consacré aux grammaires d'AL.

Afin d'illustrer ces outils et de mettre en évidence la nécessité de cohérence entre les étapes de la spécification, nous construisons à titre d'exemple, la spécification complète d'un sous-ensemble du domaine immobilier. Celui-ci traite les offres et les demandes de "location" ou d'"achat-vente" concernant les maisons et les appartements.

3.2.2.1 - Le descripteur de structure

Pour décrire l'ensemble des représentations internes d'annonces admissibles, l'idée est d'utiliser une grammaire non récursive dotée d'un calcul d'attributs et de contraintes à l'application des règles.
Cette grammaire est définie comme suit:

- * Axiome = annonce
- * Vt = { axiomes des grammaires d'AL }
- * EUC (ensemble des unites de comparaison) \subset Vn U Vt
- * ECA (ensemble des compréhensions admissibles)
= { { x | x \in EUC et x \vdash T
où T est un arbre issu d'une dérivation d'annonce } }

Outre sa fonction de caractérisation des structures admissibles, le descripteur de structure autorise l'expression d'inférence sur les valeurs d'attributs des noeuds de l'arborescence.

Les UC ayant été déterminées préalablement, l'expert commence sa description du domaine en procédant par une décomposition logique des annonces.

Il spécifie que toute annonce est constituée d'un "verbe" (nature de la transaction), d'un "objet" (objet de la transaction) et d'un "échange" (modalités de la transaction) possédant respectivement les attributs SV (sens du verbe) et MV (mode du verbe), DO (désignation de l'objet) et SV, DO, SV et MV grâce à une règle telle que:

```
~~~~~  
R1: <annonce> --->  
      <verbe(SV.MV)><objet(DO.SV)><échange(DO.SV.MV)>
```

L'identité des noms d'attributs rattachés aux différents noeuds explicite le transport de ces attributs d'un noeud à l'autre, et donc leur égalité.

Remarque:

Il convient de noter que l'agencement des sous-chaînes identifiées dans le texte de l'annonce est indépendant de l'ordre des non-terminaux dans les règles; celui-ci n'est absolument pas significatif à ce niveau.

Pour chaque variable terminale ou non-terminale, l'expert spécifie également le domaine de valeurs de ses attributs.

```
~~~~~  
D(verbe.1) = {offre,demande,sv-zéro-info}  
D(verbe.2) = {location,achat-vente,mv-zéro-info}  
D(objet.1) = {maison,appartement}  
D(objet.2) = {offre,demande,sv-zéro-info}  
D(échange.1) = {maison,appartement}  
D(échange.2) = {offre,demande,sv-zéro-info}  
D(échange.3) = {location,achat-vente,mv-zéro-info}
```

où: "N.i" désigne le ième attribut du noeud N.

On remarque que le domaine de "objet.1" par exemple, ne contient aucune valeur correspondant à la non information (zéro-info). L'expert impose ainsi que la désignation de l'objet soit toujours connue.

L'expert continue son analyse des annonces en dérivant successivement chaque non-terminal. Tout d'abord il exprime que les informations recueillies au niveau du noeud verbe sont directement issues d'entités linguistiques. Il introduit pour ce faire deux fils pour "verbe": "sens" et "mode" qui sont des feuilles de l'arborescence, c'est-à-dire des axiomes de grammaires d'analyse linguistique. On note de tels axiomes en les encadrant par "<<" et ">>".

Ayant désigné "verbe" comme UC, l'expert, à l'aide d'un calcul d'attributs, regroupe sur ce noeud, les informations émanant de ses fils. Ce calcul d'attributs matérialisé par une expression "où..." rattachée à la règle, fait apparaître des contraintes telles que des conditions d'appartenance à des ensembles de valeurs...

```
R2: <verbe(SV.MV)> ---> <<sens(SV1)>><<mode(MV1)>>
    OU SV = CAS SV1 DANS {offre,demande}: SV1,
           SV1 DANS {sv-zéro-info}:
           CAS MV1 DANS {vente,offloc}: offre,
           MV1 DANS {achat,demloc}: demande
           FCAS
           FCAS
    MV = CAS MV1 DANS {vente,achat}: vente,
           MV1 DANS {offloc,demloc}: location
           FCAS
```

```
PARAPH(verbe) =
    CAS SV1 DANS {sv-zéro-info}:
    CAS MV1 DANS {mv-zéro-info}:
    CAS SV DANS {offre}:
    'vous avez'%CAS MV DANS {achat-vente}: 'à vendre '
           MV DANS {location}: 'à louer'
           FCAS
    SV DANS {demande}:
    'vous cherchez'%CAS MV DANS {achat-vente}:
           'à acheter'
           MV DANS {location}:
           'à louer'
           FCAS
    SV DANS {demande}:
    'vous cherchez'%CAS MV DANS {achat-vente}:
           'à acheter'
           MV DANS {location}:
           'à louer'
           FCAS
    FCAS
AUTREMENT:
    CAS SV DANS {offre}: 'vous avez'%PARAPH(mode)
    SV DANS {demande}: 'vous cherchez'%PARAPH(mode)
    FCAS
AUTREMENT:
    CAS MV1 DANS {mv-zéro-info}:
    CAS MV DANS {achat-vente}:
    PARAPH(sens)% 'à vendre '
    MV DANS {location}:
    PARAPH(sens)% 'à louer'
    FCAS
    FCAS
    FCAS
```

Avec d'autre part:

```
D(sens.1) = {offre,demande,sv-zéro-info}
D(mode.1) = {vente,achat,offloc,demloc,mv-zéro-info}
```

```
~~~~~  
R3: <objet(DO.SV)> ---> <désignation(DO.PT.ST.NBP1.NBP2)>  
      <caractéristiques(DO.SV.PT.ST.NBP1.NBP2)>
```

Avec:

```
D(désignation.1) = D(caractéristiques.1) = {maison, appartement}  
D(désignation.2) = D(caractéristiques.3) = {oui, non, pt-zéro-info}  
D(désignation.3) = D(caractéristiques.4) = R U {st-zéro-info}  
D(désignation.4) = D(caractéristiques.5)  
      = D(désignation.5) = D(caractéristiques.6)  
      = { x N | 0 < x < 20 } U {nbp-zéro-info}
```

L'expert choisit désignation comme UC; il rassemble donc sous forme d'attributs attachés à ce noeud, toutes les informations susceptibles de moduler la comparaison sur la désignation de l'objet elle-même (DO).

```
~~~~~  
R4: <désignation(DO.PT.ST.NBP1.NBP2)> ---> <<nature(DO)>>
```

Avec:

```
D(nature.1) = {maison, appartement}
```

Les propos d'un annonceur, et donc les éléments sémantiques d'une annonce peuvent être très différents selon qu'il s'agit d'une offre ou d'une demande, d'une maison ou d'un appartement... Le sous-arbre greffé au noeud "caractéristiques" est notamment sujet à de tels aléas. L'expert décrit ce sous-arbre dans tous les cas de figure, lesquels se distinguent par les valeurs des attributs, qui de fait, apparaissent explicitement dans les règles de grammaires.


```
~~~~~  
R5: <caractéristiques(maison.offre.PT.ST.NBP.nbp-zéro-info)>  
----> <<terrain(PT.ST)>><<nbpoff(NBP)>><<nville(X.Y.R)>>
```

```
R6: <caractéristiques(maison.demande.PT.ST.NBPD1.NBPD2)>  
----> <<terrain(PT.ST)>><<nbpdem(NBPD1.NBPD2)>>  
      <<nville(X.Y.R)>>
```

```
R7: <caractéristiques(appartement.offre  
      .pt-zéro-info.st-zéro-info.NBP.nbp-zéro-info)>  
----> <<nbpoff(NBP)>><<nville(X.Y.R)>>
```

```
R8: <caractéristiques(appartement.demande  
      .pt-zéro-info.st-zéro-info.NBPD1.NBPD2)>  
----> <<nbpdem(NBPD1.NBPD2)>><<nville(X.Y.R)>>
```

Où:

D(terrain.1) = {oui,non,pt-zéro-info}

D(terrain.2) = R U {st-zéro-info}

D(nbpoff.1) =D(nbpdem.1) =D(nbpdem.2) =N U {nbp-zéro-info}

D(nville.1) = N U {x-zéro-info}

D(nville.2) = N U {y-zéro-info}

D(nville.3) = N U {r-zéro-info}

Il est important de souligner qu'en faisant apparaître explicitement des valeurs d'attributs dans une règle, l'expert exprime deux choses. En effet, deux cas se présentent:

* La valeur d'un tel attribut est élaborée par ailleurs, soit par reconnaissance linguistique, soit par inférence. Il y a alors succès de la règle si cette valeur est identique à la valeur indiquée ou si la valeur calculée ailleurs est un zéro-information.

Exemple:

Si le premier attribut de "caractéristiques" a pour valeur "appartement", valeur qui a pu être obtenue lors de l'application de R3 par exemple, l'application de la règle R5 provoque un échec quant à la construction de la structure interne.

* La valeur de l'attribut n'est jamais calculée par ailleurs. Le succès d'une règle telle que R5,R6,R7,R8 confère à l'attribut caractéristique du sous-arbre en question la valeur indiquée dans la règle.

Les valeurs de certains attributs peuvent être calculées de plusieurs façons, à partir d'éléments linguistiques de natures

diverses.

Aussi lorsqu'il réalise de l'inférence, l'expert peut souhaiter notifier un contrôle de cohérence de la valeur inférée avec les autres valeurs élaborées pour l'attribut concerné.

```
~~~~~  
R9: <échange(DO.offre.MV)> ---> <<prixoff(P)>><<viager(V)>>  
      OU MV = CAS P DANS ENTIER OU V DANS {oui,non}:  
              achat-vente/CONTRADICTION  
      FCAS  
  
R10: <échange(DO.offre.MV)> ---> <<loyeroff(L)>>  
      OU MV = CAS L DANS ENTIER: location/CONTRADICTION FCAS  
  
R11: <échange(DO.demande.MV)> ---> <<prixdem(P1.P2)>>  
      <<viager(V)>>  
      OU MV = CAS P1 DANS ENTIER OU V DANS {oui,non}:  
              achat-vente/CONTRADICTION  
      FCAS  
  
R12: <échange(DO.demande.MV)> ---> <<loyerdem(L1.L2)>>  
      OU MV = CAS L1 DANS ENTIER: location/CONTRADICTION FCAS  
Avec:  
D(prixoff.1) = N U { p-zéro-info }  
D(viager.1) = { oui,non,v-zéro-info }  
D(loyeroff.1) =D(loyerdem.1) =D(loyerdem.2) =N U {1-zéro-info}
```

Exemple:

L'application de la règle R9, lorsqu'elle associe à MV la valeur "achat-vente", donne lieu à un contrôle de cohérence. Le conflit de cette valeur avec la valeur attribuée d'autre part à MV par la règle R1, traduit l'existence d'une contradiction au sein du texte de l'annonce. Celle-ci est signalée à l'utilisateur par l'édition d'un message.

Remarque:

Un élément de dialogue explicitant la contradiction (non apparent dans notre exemple) peut être intégré à la règle par l'expert.

On pourrait ainsi avoir dans R10:

"... CONTRADICTION('le mode de la transaction','le loyer '%chL%' francs')".

3.2.2.2 - Grammaires d'analyse linguistique

Les grammaires d'analyse linguistiques se raccordent à la description de la structure par le biais des feuilles de l'arborescence, qui en sont les axiomes. Elles ont pour but d'exprimer le lien entre les éléments textuels et les attributs des structures internes des annonces. La définition des éléments de paraphrase associés aux informations retenues du texte de l'annonce s'effectue également à ce niveau. L'intérêt est de permettre ainsi la reprise des termes employés par l'utilisateur dans la paraphrase.

Sur le plan syntaxique, les grammaires d'analyse linguistique sont un outil analogue à la grammaire du descripteur de structure; les terminaux sont cette fois des chaînes de caractères répertoriées dans le langage des P.A..

Remarque:

Le principe de l'analyse linguistique est d'extraire des textes des annonces des informations élémentaires. l'idée est donc de n'avoir pour une grammaire d'AL, qu'un nombre restreint d'attributs. Cependant la dépendance contextuelle oblige quelquefois à la reconnaissance de plusieurs informations par une seule grammaire.

Exemple:

L'expert souhaite disposer à des fins de comparaison, de deux informations relatives au terrain:

- présence (oui, non),
- surface.

Or dans une annonce, la chaîne "1500 m2 de terrain" par exemple, met en évidence l'imbrication de ces deux informations. L'expert est ainsi conduit à définir une seule GAL possédant deux attributs "présence terrain" et "surface terrain".

A chaque feuille de la structure (notée entre "<<" ">>" dans le descripteur de structure) définie dans le paragraphe précédent correspond donc une grammaire d'analyse linguistique. Les grammaires d'AL du domaine traité dans la maquette sont fournies en totalité en annexe. Nous n'en avons sélectionné ici que les formes caractéristiques.

Soit l'ensemble des règles associées à la grammaire "sens":

```
~~~~~  
R1.1: <<sens(offre)('vous avez')>>  
      ---> offre | dispose | à saisir  
R1.2: <<sens(demande)('vous cherchez')>>  
      ---> cherche | recherche | ch  
~~~~~
```

La règle R1.1 exprime que la rencontre dans le texte de l'annonce de l'une des formes indiquées en partie droite (offre, dispose, "à saisir") confère à l'attribut de "sens" la valeur "offre", et que la paraphrase commune à ces trois chaînes est "vous avez". Les chaînes "offre", "dispose", "à saisir" sont appelées synonymes dans la mesure où aucune distinction n'est jamais faite entre elles.

Une partie droite de règle peut également comporter des non-terminaux. C'est le cas de la grammaire "nbpdem".

A tout non-terminal sont attachées deux entités: la première représentant l'information sous forme d'un ou de plusieurs attributs, la seconde la paraphrase correspondant à cette information (respectivement x et chx pour "nombre" dans R6.3). La représentation en termes de chaînes de caractères d'une valeur numérique x est chx. La concaténation est notée "%".

les grammaires d'AL, tout comme le descripteur de structure, permettent de spécifier du calcul d'attributs (dans R6.3, calcul de x1 à partir de x).

```
~~~~~  
R6.1: <<nbpdem(x.Ø)('comportant au moins'%chx%'pièces')>>  
      ---> <<nombre(x)(chx)>> pièces [ [ au ] min | minimum ]  
R6.2: <<nbpdem(x.x)('comportant%chx%'pièces')>>  
      ---> <<nombre(x)(chx)>> pièces |  
           type <<nombre(x)(chx)>> |  
           F <<nombre(x)(chx)>>  
R6.3: <<nbpdem(x1.x1)('comportant'%chx1%'pièces')>>  
      ---> F <<nombre(x)(chx)>> bis  
           OU x1 = x + 1  
R6.4: <<nbpdem(x.y)('comportant entré%chx%'et'%chy%'pièces')>>  
      ---> entre <<nombre(x)(chx)>> et <<nombre(y)(chy)>> pièces  
           | <<nombre(x)(chx)>> [ ou ] <<nombre(y)(chy)>> pièces  
~~~~~
```

Enfin, des contraintes sur les valeurs des attributs peuvent être formulées comme condition au succès de la règle. C'est le cas de R10.1 où le succès de la règle n'est obtenu que si la valeur de x est supérieure à 50000.

```
~~~~~  
R10.1: <<prixoff(x)('à un prix de '%chx%' francs')>>  
      ---> <<nombre(x)(chx)>> [ F | francs ]  
      OU x > 50000  
R10.2: <<prixoff(x)('à un prix de '%chx%' francs')>>  
      ---> prix <<nombre(x)(chx)>> [ F | francs ]  
~~~~~
```

Une seconde famille de règles, dites règles d'exception, a pour but de spécifier l'interprétation du système lorsqu'aucun élément textuel de l'annonce ne permet d'extraire une information. Au sens des grammaires, cette situation correspond à la dérivation dans le vide.

On distingue trois types de sémantiques des règles d'exception.

1er cas

Il se produit pour la grammaire "sens":

Ainsi lorsqu'aucune information concernant le sens de la transaction ne figure dans l'annonce, l'attribut de "sens" prend la valeur "sv-zéro-info", qui signifie zéro information. L'élément de paraphrasage est ici la chaîne vide.

EXCEPTION(sens): <<sens(sv-zéro-info)('')>>

2ème cas

L'expert interdit tout simplement la dérivation dans le vide. C'est le cas pour la grammaire "nature" dont l'exception est:

EXCEPTION(nature): OBLIGATOIRE

3ème cas

Le système émet une hypothèse sous forme de valeur d'attribut. En l'absence de donnée explicite sur le mode de la transaction, le système formule tantôt l'une, tantôt l'autre des hypothèses définies par l'expert.

EXCEPTION(mode): <<mode(location)('')>>

EXCEPTION(mode): <<mode(achat-vente)('')>>

3.2.2.3 - Le langage de comparaison

Muni du langage de comparaison, l'expert détermine les unités de comparaison d'une part, d'autre part les couples d'UC comparables et les comparaisons élémentaires sur ces couples.

Il définit ainsi un ensemble de "termes". A chacun d'eux sera associé un schéma de comparaison.

Un terme est un couple dont chaque composant est une liste d'attributs d'UC typés, analogues à des paramètres formels de procédure. Le premier composant correspond à une UC d'une annonce "offre", le second à une UC d'une annonce "demande".

Exemple: L'expert définit le terme nbpièces dont les deux composants sont des listes d'attributs de type entier:

```
TERME nbpièces* ((ENT nbp0),(ENT nbpd1,npbd2))
```

Pour chacun des termes, l'expert énumère tous les couples d'UC "effectifs" dont la comparaison s'identifie au modèle de comparaison associé au terme.

Exemple:

Pour le terme nbpièces, le seul couple d'U.C. dont la comparaison s'effectue sur ce modèle est:

```
(nbpoff{x},nbpdem{y1,y2})
```

Enfin pour définir le modèle de comparaison, l'expert dispose d'un langage syntaxiquement proche du descripteur de structure en ce qui concerne le calcul d'attributs, et doté d'un certain nombre de fonctions primitives.

Dans notre exemple, on distingue trois sortes de comparaisons selon la constitution des UC en termes d'attributs. Ceux-ci sont de nature:

- discrète (ex: la nature de l'objet, autrement dit l'attribut DO de "désignation"): fonction COMPDIS,
- numérique:
 - intervalle (ex: UC nbpièces, prixoff, prixdem, etc):
fonction COMPSEG,
 - surface (ex: nville) : fonction COMPSURF.

Le langage comprend actuellement les fonctions suivantes:

-COMPSEG(S1,S2):

S1 et S2 étant deux intervalles, la fonction rend vrai s'ils ont une intersection non vide;

-COMPSURF(S1,S2):

S1 et S2 étant deux surfaces, la fonction rend vrai si elles ont une intersection non vide;

-COMPDIS(d)/d1/d2/... :

d étant une distance, la fonction rend vrai au seuil i si $d \leq d_i$.

Exemple:

La comparaison sur le terme nbpièces opère sur un entier nbpo et un intervalle [nbpd1,nbpd2].

Pour le domaine considéré, le seul couple d'UC relevant de ce terme est:

-nbpoff(dont l'attribut est x),
-nbpdem (dont les attributs sont y1 et y2).

La comparaison entre nbpoff et nbpdem associe les paramètres effectifs x,y1,y2 aux paramètres formels nbpo, nbpd1, nbpd2.

Le modèle de comparaison indique que la fonction de comparaison est la fonction COMPSEG (décrite précédemment) dont les arguments sont ensuite définis.

Au vu de la fonction SEGVA dans cet exemple, il apparaît que l'expert a défini ici trois seuils de comparaison.

```
~~~~~  
TERME nbpièces : ((ENT nbpo),(ENT nbpd1,nbpd2)) =
```

```
-----  
(nbpoff{x},nbpdem{y1,y2});
```

```
COMPSEG(S1,S2)
```

```
OU S1=CAS nbpo=nbp-zéro-info:(0,+ ),
```

```
    AUTREMENT: (nbpo,nbpo)
```

```
    FCAS
```

```
    S2=CAS nbpd1=0: SEGVA(nbpd2,nbpd1)/(1,0)/2,1)/(3,1),
```

```
        nbpd2=0: SEGVA(nbpd1,nbpd1)/(0,1)/(1,2)/(1,3),
```

```
        nbpd1=nbp-zéro-info: (0,+ ),
```

```
        nbpd1=nbpd2: SEGVA(nbpd1,nbpd1)/(0,1)/(1,2)/(2,3),
```

```
        nbpd1<nbpd2: SEGVA(nbpd1,nbpd2)/(0,0)/(1,1)/1,2),
```

```
        nbpd1>nbpd2: SEGVA(nbpd2,nbpd1)/(0,0)/(1,1)/(1,2)
```

```
    FCAS;
```

3.3 - IDEES

Coëfficient de vraisemblance sur les informations inférées

Les informations inférées comprennent:

- les informations obtenues par calcul d'attributs,
- les hypothèses établies à partir de dérivations dans le vide de grammaires d'analyse linguistique.

Ce coëfficient permettrait:

- de déterminer parmi plusieurs structures comportant des inférences, les plus vraisemblables;
- d'exiger un seuil de vraisemblance minimum pour le déclenchement d'une contradiction.

Symétrie des contradictions

Le descripteur de structure montre que la contradiction consiste en un conflit entre deux valeurs d'attributs. D'où l'idée de spécifier en dehors des règles la nécessaire identité des valeurs d'attributs concrétisant une même information.

Il convient donc:-soit d'admettre la portée générale d'une spécification de contradiction quel que soit l'ordre dans lequel sont élaborées les deux valeurs d'attributs conflictuelles,

-soit de spécifier les contradictions hors des calculs d'attributs attachés aux règles du descripteur de structure.

Union d'objets

L'étude du domaine traité dans la maquette fait apparaître des éléments qui plutôt qu'à des objets en soi correspondent à des unions d'objets.

Il faudrait donc prévoir dans le langage de spécification certaines facilités permettant de représenter une telle notion.

Exemples:

"logement" : "maison" U "appartement"

"vends ou loue" : "achat-vente" U "location"

Dialogues

Les dialogues à mettre en oeuvre lors du désaccord de l'utilisateur avec la paraphrase n'appartiennent pas à l'heure actuelle au domaine des spécifications. Dans une phase ultérieure du projet, il conviendrait toutefois de donner à l'expert la possibilité d'en spécifier d'une part la structure, d'autre part le contenu. Il serait alors nécessaire d'intégrer au langage de spécification des outils permettant de modéliser un interlocuteur virtuel.

Définition d'un langage de plus haut niveau pour l'expert

L'épreuve de la maquette a confirmé que l'actuel langage répond aux besoins de spécifications d'un expert en annonces. Néanmoins les expérimentations effectuées ne permettent pas d'affirmer qu'il est à la portée d'un expert non familiarisé avec les grammaires et le calcul d'attribut. On doit dès lors songer à un langage de plus haut niveau dont la sémantique serait celle du langage actuel. Celui-ci pourrait par suite, constituer un niveau intermédiaire.

Interface avec l'expert comprenant:

- la compilation des spécifications,
- la définition d'outils de C.A.O.,
(cf. 4.3).

4 - R E A L I S A T I O N

=====

Ce chapitre présente le projet HAVANE, sous son dernier aspect, celui des réalisateurs, et donne les choix de réalisation sous-jacents aux choix effectués quant à la spécification du système par l'expert et à son utilisation.

Le langage utilisé pour la maquette est le langage PROLOG [11], très adapté à la construction des structures arborescentes.

4.1 - LA BASE DE CONNAISSANCE

L'ensemble des spécifications données par l'expert constitue la base de connaissance qui représente la définition du système. La constitution de cette base de connaissance a été réalisée conformément à un schéma de traduction des spécifications en clauses PROLOG.

L'automatisation de la traduction n'a pas été abordée pour l'instant.

4.2 - LE SYSTEME

Il nous faut rappeler que le système HAVANE est bâti autour de la base de connaissance étudiée au paragraphe précédent, et du noyau. Ce sont donc essentiellement les caractéristiques de ce noyau qui sont présentées ici: tout d'abord celles qui concernent l'aspect analyse des petites annonces, puis celles relatives à la comparaison.

4.2.1 L'analyse

4.2.1.1 Principe

L'analyse d'une petite annonce repose sur le principe suivant: soumettre à l'utilisateur la compréhension du système par le biais de la paraphrase, et ce jusqu'à approbation de celui-ci.

4.2.1.2 - Evaluation

1) Priorité à l'information

L'intelligence du système repose notamment sur sa capacité à délivrer en premier lieu, les structures les plus probables, c'est-à-dire susceptibles d'être acceptées par l'utilisateur lors du paraphrasage.

Cette évaluation s'effectue sur la base d'un critère: donner la priorité à l'information par rapport à la non-information.

De plus dans la maquette, pour des raisons d'efficacité, et par opposition à une évaluation globale de l'ensemble des structures admissibles associées à une P.A. donnée, toute structure est évaluée dès sa construction, et ce uniquement par rapport aux structures déjà paraphrasées.

L'efficacité du système réside donc dans sa capacité à construire d'emblée les structures qui auraient été retenues par une évaluation à posteriori.

D'où la mise en oeuvre de techniques visant à respecter ce critère, à différents niveaux: l'utilisation des règles d'analyse et le traitement des exceptions, le traitement des contradictions, des chevauchements et des succès multiples d'une grammaire d'analyse linguistique.

2) Règles d'analyse et traitements des exceptions

Les règles d'exceptions spécifient quelle doit être l'interprétation du système en l'absence d'information explicite dans le texte de l'annonce (cf. 3.2.2.2).

De telles règles, même si elles permettent parfois d'élaborer des informations, ne sont à appliquer qu'en cas d'échec de toutes les règles ramenant des informations basées sur la reconnaissance effective de chaînes dans l'annonce. De plus dès lors qu'une grammaire d'A.L. a mené à la reconnaissance linguistique d'une sous-chaîne ne donnant lieu à aucun conflit avec les autres grammaires (chevauchement, contradiction), les règles d'exception de cette grammaire peuvent être éliminées de l'ensemble des règles applicables.

Parmi les règles élaborant des informations à partir de la reconnaissance de sous-chaînes dans le texte, celles dont les dérivations mènent aux chaînes de caractères les plus longues sont appliquées d'abord, ceci afin d'éviter la reconnaissance de chaînes incluses dans des chaînes également analysables. Les clauses PROLOG associées sont donc à placer en tête.

Exemple:

La grammaire d'A.L. "loyer" reconnaît des sous-chaînes telles que " x francs" et "x francs maximum"; la seconde étant plus informée, il convient de lui donner la priorité par rapport à la première.

De même pour les règles d'exception, le système se doit d'appliquer d'abord les règles déduisant des informations "par défaut".

3) Traitement des contradictions

L'idée est de ne jamais renoncer à une information présente dans le texte d'une annonce; dès la détection d'une contradiction, le système doit la signaler à l'utilisateur et non pas essayer de produire des structures sans contradiction, nécessairement moins informées.

Dans le descripteur de structure, les règles qui notifient des contradictions doivent être appliquées d'abord. Les clauses Prolog associées sont à placer en tête.

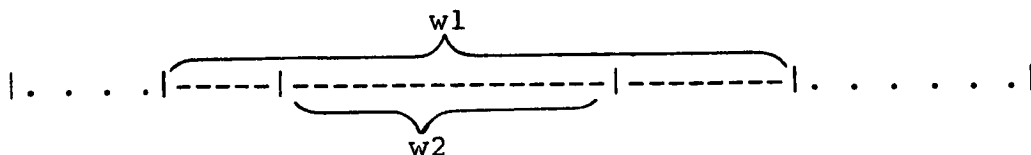
4) Traitement des chevauchements

Dans la maquette, en raison de leur complexité, les problèmes concernant les chevauchements sont résolus localement au cours d'une analyse, tout en visant à respecter le critère de priorité à l'information.

Dans une annonce, deux chaînes w1 et w2 présentant une intersection non vide peuvent être analysées respectivement par des grammaires d'A.L. <<g1>> et <<g2>>.

Deux cas sont possibles:

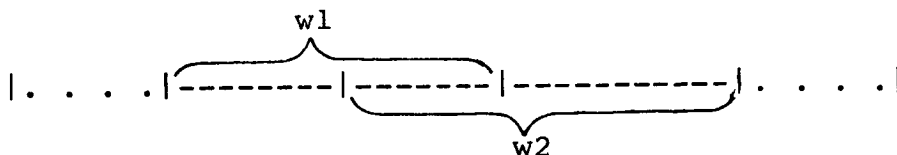
1) w1 inclut w2 (ou vice-versa)
~~~~~



Dans cette configuration il y a succès de la reconnaissance de w1 (plus informée) par <<g1>> et échec de la reconnaissance de w2 par <<g2>>.

2) w1 n'inclut pas w2 (et vice-versa)  
~~~~~

Il y a une ambiguïté dans l'annonce.



Exemple: w1: "terrain 250 m2"
w2: "250 m2 habitable"

La solution mise en oeuvre dans la maquette est alors la suivante:

* Il existe des règles d'exception autorisant la dérivation dans le vide des grammaires <<g1>> et <<g2>>:

D'une part, on ne peut lever pour les deux grammaires, la contrainte en vertu de laquelle la reconnaissance d'une chaîne par une grammaire élimine, pour celle-ci, la dérivation dans le vide; cela pourrait en effet conduire à une structure bâtie sur les dérivations dans le vide de <<g1>> et <<g2>>.

D'autre part il faut lever cette contrainte pour la grammaire mise en échec.

Ces deux arguments nous conduisent à déclencher un dialogue focalisant pour demander à l'utilisateur quelle est entre w1 et w2, la chaîne à retenir.

* La dérivation dans le vide est interdite pour l'une des deux grammaires, par exemple pour <<g1>>.

Dans ce cas, l'appréhension locale du problème conduit à la reconnaissance de w1 par <<g1>>.

* Ni <<g1>>, ni <<g2>> ne peuvent se dériver dans le vide. Havane poursuit son analyse. Grâce au retour arrière de Prolog deux possibilités seront exploitées:

-la reconnaissance de w1 par <<g1>> et la recherche dans l'annonce, d'une nouvelle chaîne w2' telle que <<g2>> se dérive en w2'.

-la reconnaissance de w2 par <<g2>> et la recherche dans l'annonce, d'une nouvelle chaîne w1' telle que <<g1>> se dérive en w1'.

Le système mémorise le chevauchement, afin de pouvoir le signaler à l'utilisateur, si aucune des deux directions ne mène à une structure admissible.

En fait une telle approche se révèle défailante à l'égard de bien des situations. Le critère d'évaluation tend à la reconnaissance et à l'exploitation globales du maximum d'informations, en l'occurrence à la réalisation du maximum des succès de GAL. Le problème des chevauchements doit donc être abordé dans l'optique beaucoup plus large des succès multiples de grammaires. Les succès de plusieurs grammaires sur une même sous-chaîne (chevauchements) interfèrent en effet avec les succès d'une même grammaire sur plusieurs sous-chaînes d'un texte d'annonce. Les choix sous-jacents aux chevauchements et succès multiples relèvent bien entendu d'un traitement global des résultats de l'analyse linguistique.

4.2.2 La comparaison

Ainsi que nous l'avons dit dans le chapitre 2, la comparaison est basée sur un principe de seuils de tolérance.

La sélection des annonces complémentaires s'effectue de la manière suivante:

- Pour chaque seuil, depuis le plus restrictif jusqu'à épuisement des seuils définis par l'expert ou satisfaction de l'utilisateur, est réalisé un parcours séquentiel de la banque.

- Chaque annonce de la banque correspondant à une transaction opposée à celle de l'annonce courante (au sens où "offre" et "demande" sont opposées) donne lieu à une comparaison conforme aux spécifications de l'expert, dans le sens offre-demande. Compte tenu de l'algorithme utilisé, ce sens n'affecte en rien la symétrie de la comparaison.

Cette comparaison est basée sur un parcours de la structure associée à l'annonce "offre". Pour chaque noeud de cette structure correspondant à une unité de comparaison C_1 , HAVANE recherche dans la structure de l'annonce "demande" une UC C_2 , telle qu'il existe un terme dont la liste comporte le couple (C_1, C_2) . S'il n'existe pas de couple (C_1, C_2) tel que C_2 appartient à la structure "demande", il n'y a pas de comparaison élémentaire sur C_1 .

Lors d'une comparaison élémentaire sur la caractéristique C_1 , il y a succès (resp. échec) si le résultat de la comparaison sur (C_1, C_2) a donné un écart inférieur ou égal (resp. supérieur) à l'écart maximum toléré au seuil considéré.

4.3 - IDEES ET PERSPECTIVES

* Réalisation d'une interface ergonomique avec l'utilisateur avec notamment la mise en place d'un système-expert [12,13] pour aider l'utilisateur.

* Reformulation automatique de l'annonce

Le problème de la reformulation de l'annonce se pose lorsque l'utilisateur, en désaccord avec la compréhension du système, y a apporté des modifications entraînant une non-conformité de cette compréhension avec le texte initial de l'annonce.

L'utilisateur a été amené à définir par dialogue:

- un ensemble IE, éventuellement vide, d'informations erronées parmi lesquelles certaines sont à supprimer du texte, d'autres à modifier (resp. les sous-ensembles IES et IEM de IE).
- un ensemble IM, éventuellement vide, d'informations manquantes qu'il faut ajouter au texte de l'annonce.

Dans le cas d'une reformulation automatique, le rôle du système est donc:

- pour chaque élément de IES, de retrouver les chaînes qui lui sont associées et de les supprimer du texte.
- pour chaque élément de IEM, de retrouver les dérivations conduisant aux chaînes concernées et de tenter de les réutiliser pour répercuter la modification dans le texte. Lorsque cela n'est pas possible, l'élément est traité comme un élément de IM.
- pour chaque élément de IM, choisir une (ou plusieurs) dérivations produisant la (ou les) sous-chaîne(s) constituant l'information manquante.

Problèmes

La technique d'analyse par mots-clefs ne permet pas de délimiter et donc de supprimer les contextes lors de la reformulation automatique.

Exemple:

Soit le texte initial:

"vends maison avec jardin et possibilité de garage",
avec IES = { présence garage }.

Si l'information "présence garage" est basée sur la rencontre de la chaîne "garage", la reformulation du texte par le système risque de produire le texte pour le moins incompréhensible et surprenant:

"vends maison avec jardin et possibilité de".

Le problème résulte du fait que le contexte ("avec possibilité de") n'est pas analysé. Cette chaîne est

donc considérée comme du "bruit" par le système.

De même les contextes non analysés par le système, ne permettent pas de savoir où insérer les chaînes manquantes. D'autre part ces chaînes ne peuvent être systématiquement ajoutées en fin de texte d'annonce.

Exemple:

Soit le texte initial:

"Urgent vend cause mutation très beau type 3, état neuf, garage, prix 350000f, quartier calme de Rennes." avec IA = {appartement},

L'information "appartement" étant concrétisée par la chaîne de caractères "appartement".

D'une part l'ajout en fin de texte produit la formulation maladroite:

"Urgent vend cause mutation très beau type 3, état neuf, garage, prix 350000f, quartier calme de Rennes, appartement."

D'autre part, soit en majuscules les expressions-clefs reconnues par le système:

"Urgent VENDS cause mutation très beau TYPE 3, état neuf, GARAGE, PRIX 350000F, quartier calme de RENNES."

Les bruits séparant les chaînes "VENDS" et "TYPE 3" ne permettent pas de désigner l'endroit qui conviendrait à l'insertion de la chaîne "appartement".

* Implémentation des spécifications visant à satisfaire les contraintes de performance inhérentes à tout système à vocation "grand public";

L'idée est de redéfinir l'aspect gestion de la banque d'annonces, en relayant l'actuelle banque de données PROLOG par un appel à un SGBD relationnel classique.

Cette transposition débouche sur une étude de portée en fait très générale, sur l'expression et la manipulation d'informations floues dans les bases de données [14].

Cette optique sous-tend peut-être en outre la nécessité de nouvelles spécifications relatives au stockage des annonces, et par conséquent de nouveaux outils pour l'expert.

* Réalisation d'une interface avec l'expert comprenant:

- la compilation des spécifications pour constitution et mise-à-jour automatiques de la base de connaissance.

La mise-à-jour de cette base induit quant à elle la répercussion des modifications sur la banque d'annonces.

Dans le même ordre d'idées, on peut imaginer une génération automatique de schémas de B.D., de même que la détermination, sinon l'optimisation des chemins d'accès à la base de données.

Ces différentes extensions relèvent bien entendu des compétences d'un système-expert, voire multi-experts, en

matière de spécification de connaissances.

- la mise en oeuvre d'outils de C.A.O., également sous forme de système-expert,
- la production automatique de tests.

* Définition de nouveaux critères d'évaluation permettant d'arriver plus rapidement à la "bonne compréhension".

L'évaluation des structures pourrait par exemple être élargie à un calcul de degré de vraisemblance réalisé à partir des coefficients de vraisemblance définis par l'expert (cf. 3.3).

* Etude d'architectures

Une étude comparée d'architectures multiprocesseurs, spécialisées, réparties, etc, serait à effectuer.

5 - C O N C L U S I O N

=====

HAVANE est un système intelligent assurant la mise en relation automatique de petites annonces exprimées en langage naturel.

Le projet a débuté par la réalisation d'une maquette dont les objectifs et les caractéristiques sont tout d'abord rappelés. Compte tenu de ces acquis, sont ensuite formulées quelques remarques d'ordre général. Une dernière partie dresse les perspectives envisagées à court et à moyen terme.

5.1 - La maquette

Le but de la maquette HAVANE a consisté en la définition d'un service en relation avec des professionnels des P.A.. Cette collaboration a débouché sur la conception d'un système paramétrable, visant à démontrer la faisabilité d'un système construit à partir des spécifications d'un expert en annonces.

L'essentiel des réalisations a donc résidé en la définition et l'implémentation d'un langage de spécification permettant à l'expert de transmettre ses connaissances au système. Le problème des performances n'a pas été abordé à ce stade.

La maquette a été écrite dans le langage de programmation PROLOG (Version CNET, Paris A, d'après GIA-Marseille) sur le système MULTICS. L'environnement PROLOG a permis l'implémentation aisée des grammaires de métamorphose.

5.2 - Appréciations d'ordre général

Cette maquette n'a donné lieu à aucune expérimentation sur des usagers. L'évaluation des réalisations ne permet donc pas de préjuger ni du bien-fondé du service, ni des choix effectués. Une extension de l'aspect "dialogues", quelque peu négligé jusqu'ici, notamment en cas de désaccord de l'utilisateur avec la paraphrase, demeure certainement à prévoir.

En ce qui concerne le langage de spécification, son pouvoir d'expression s'est avéré suffisant; les grammaires de métamorphose et le calcul d'attributs sont adaptés d'une part à la description des structures admissibles, d'autre part à l'établissement du lien entre structures et textes d'annonces. Enfin, le langage de comparaison autorise l'introduction du contexte dans la comparaison des éléments des annonces; il peut donc être jugé comme relativement souple et ad'hoc pour la prise en compte de termes flous dans les annonces.

L'appréhension d'un tel langage par un expert en annonces , sans connaissance particulière sur les grammaires d'attributs, reste toutefois à démontrer par des conditions d'expérimentation réelles. Dès à présent, l'adéquation du langage à ce niveau, peut être mise en doute.

La technique d'analyse par mots-clefs se révèle intéressante dans le cadre d'un tel système, où seul un ensemble limité d'informations intervient pour la mise en relation des annonces. L'absence d'une syntaxe rigoureuse du langage usuel des P.A. corrobore en outre ce choix.

Cette méthode engendre cependant un certain nombre de problèmes spécifiques tels que chevauchements de sous-chaînes, etc, ce qui préconise un traitement global des résultats de l'analyse linguistique.

La généralisation du système à d'autres domaines, en particulier "rencontres" et "divers", sous-tend la remise en question du schéma "offre-demande".

Une traduction immédiate des spécifications a pu être réalisée dans le langage PROLOG, d'utilisation simple et agréable. Les mécanismes d'unification et de retour-arrière permettent en effet une manipulation aisée des arbres.

5.3 - Perspectives

Suite à ces réalisations, une première nécessité s'impose, à savoir la confrontation d'HAVANE avec le grand public. Celle-ci suppose l'apport de quelques aménagements à la maquette actuelle visant principalement à assurer la robustesse du système, notamment par la prise en compte des fautes d'orthographe, mais également son efficacité. La redéfinition de la comparaison en terme de système de gestion de base de données, et par conséquent du stockage des données, contribuera à un gain non négligeable de performance.

De cette mise à l'épreuve se dégageront sans doute quelques améliorations du service.

Par ailleurs, des études à plus long terme seront menées, avec pour but de concevoir sous forme de système-expert:

- une interface avec l'utilisateur qui comprendra une nouvelle structure de dialogues, à la fois plus riche et plus ergonomique, ainsi qu'une aide à l'utilisateur,
- une interface avec l'expert.

Afin de figer le langage intermédiaire et les stratégies d'analyse et de comparaison, une réflexion préalable nécessitant une étude sur la représentation des connaissances, présidera à la définition d'un langage de plus haut niveau doublé d'une méthodologie de spécification

pour l'expert.

Cette interface comprendra outre l'interprétation des spécifications, un ensemble d'outils de C.A.O. mis à la disposition de l'expert.

* BIBLIOGRAPHIE *

- [1] Equipe HAVANE .
Rapport de la convention de recherche HAVAS-IRISA, Juin 1982.
- [2] P. BOSC, A. CHAUFFAUT, L. TRILLING.
Proposition pour une expérimentation, Juin 1982.
- [3] P. BOSC.
HAVANE: un système de gestion automatique de petites annonces, Journées Bases de données, Toulouse, Sept. 1982.
- [4] M. COURANT.
Rapport de DEA, Université de Rennes, Juin 1982.
- [5] M. LEGAULT, R. QUINIOU, L. TRILLING.
Le système ANAGPA (ANalyse et Gestion de Petites Annonces), 2ème congrès AFCET-IRIA Reconnaissance des formes et intelligence artificielle, Toulouse, Sept. 1982.
- [6] M. LEGAULT.
ANAGPA, ANALYSE et Gestion de Petites Annonces, Traitement des annonces d'automobiles, Thèse 3ème cycle, Université de Rennes, Fév. 1978.
- [7] R. QUINIOU.
ANAGPA: un système général d'ANALYSE et de Gestion automatique de Petites Annonces, Thèse Docteur-Ingénieur, Université de Rennes, Nov. 1979.
- [8] R. QUINIOU, L. TRILLING.
Conception et réalisation d'une application grand public: le système ANAGPA, Congrès La conception de systèmes télématiques, Nice, Juin 1981.
- [9] KNUTH.
Semantics of context-free languages, Mathematical System Theory, Vol. 2, No 2, 1968, et Vol. 5, No 1, 1971.
- [10] A. COLMERAUER.
Metamorphosis grammars. Natural communication with

Computer, Lectures Notes in Computer Science, Springer Verlag, 1978.

- [11] A. COLMERAUER.
Programmation en logique du 1er ordre, Actes Journées "La compréhension", IRIA, 1977.
- [12] J.L. LAURIERE.
Représentation et utilisation des connaissances, Les systèmes-experts, Revue TSI-RAIRO, Vol. 1, No 1, 1982, Dunod.
- [13] J.L. LAURIERE.
Représentation et utilisation des connaissances, Représentation de connaissances, Revue TSI-RAIRO, Vol. 1, No 2, 1982, Dunod.
- [14] P. HAFENEDER.
Rapport de DEA, Université de Rennes, Juin 1982.

A1.1

ANNEXE 1

=====

- SPECIFICATIONS DE LA MAQUETTE -

GRAMMAIRE DE STRUCTURE

R0 :

<annonce> -> <verbe(SV.MV)> <objet(DO.SV)> <echange(DO.SV.MV)>

R1 :

<verbe(SV.MV)> -> <<sens(SV1)>><<mode(MV1)>>
 OU SV = CAS SV1 DANS {offre} : offre ,
 SV1 DANS {demande} :
 CAS MV1 DANS {vtenu} : offre,
 AUTREMENT : demande
 FCAS
 SV1 DANS {FANT} :
 CAS MV1 DANS {vtenu,achmou,vente,
 locmou,offloc} : offre,
 MV1 DANS {achat,demloc} : demande
 FCAS
 FCAS
 MV = CAS MV1 DANS {vtenu,achmou,
 vente,achat} : achvente,
 MV1 DANS {locmou,offloc,
 demloc} : location,
 AUTREMENT :MV1
 FCAS

R2 :

<objet(DO.SV)> -> <designation(DO.PT.ST.NBPD1.NBPD2)>
 <caracteristiques(DO.SV.PT.ST.NBPD1.NBPD2)>

R3 :

<designation(DO.PT.ST.NBPD1.NBPD2)> -> <<nature(304)>>
 OU DO = CAS DO1 DANS {maison,villa} : maison
 DO1 DANS {maisonnette} : maisonnette ,
 DO1 DANS {appartement} : appartement,
 DO1 DANS {chambre} : chambre
 FCAS

R4 :

<caracteristiques(DO.offre.PT.ST.NBP.SANSOBJ)> ->
 <<terrain(PT.ST)>> <<nbpoff(NBP)>> <<surface>> <<nville>>
 OU DO DANS {maison,maisonnette}
 NBP = CAS DO DANS {maisonnette} :
 CAS NBP DANS {ENTIER} ET NBP >= 3 :
 CONTRADICTION
 NON (NBP DANS {ENTIER}) : 2
 AUTREMENT : NBP
 FCAS
 AUTREMENT : NBP
 FCAS

R5 :

<caracteristiques(DO.demande.PT.ST.NBPD1.NBPD2)> ->

- <<terrain(PT.ST)>> <<nbpdem(NBPD1.NBPD2)>> <<surface>>
 <<nville>>
 OU DO DANS {maison,maisonnette}
 (NBPD1,NBPD2) = CAS DO DANS {maisonnette} :
 CAS NBPD1 DANS {ENTIER} ET NBPD1 >= 3
 OU NBPD2 >= 3 :
 CONTRADICTION
 NON (NBPD1 DANS {ENTIER}) : (2,3)
 AUTREMENT : (NBPD1,NBPD2)
 FCAS
 AUTREMENT : (NBPD1,NBPD2)
 FCAS
- R6 :
- <caracteristiques(appartement.offre.SANSOBJ.SANSOBJ.NBP.SANSOBJ)> ->
 <<nbpoff(NBP)>> <<surface>> <<nville>>
- R7 :
- <caracteristiques(appartement.demande.SANSOBJ.SANSOBJ.NBPD1.NBPD2)> ->
 <<nbpdem(NBPD1.NBPD2)>> <<surface>> <<nville>>
- R8 :
- <caracteristiques(chambre.SV.SANSOBJ.SANSOBJ.SANSOBJ.SANSOBJ)> ->
 <<surface>> <<nville>>
- R9 :
- <echange(DO.SV.MV)> -> <montant(DO.SV.MV)>
- R10 :
- <montant(DO.offre.MV)> -> <<prixoff(P)>> <<viager(V)>>
 OU DO DANS {maison,maisonnette,appartement}
 MV = CAS P DANS ENTIER OU V DANS {oui,non} : achvente / CONTRADICTION
 FCAS
- R11 :
- <montant(DO.offre.MV)> -> <<loyeroff(L)>>
 OU DO DANS {maison,maisonnette,appartement,chambre}
 MV = CAS L DANS ENTIER : location
 FCAS
- R12 :
- <montant(DO.demande.MV)> -> <<prixdem(P1.P2)>> <<viager(V)>>
 OU DO DANS {maison,maisonnette,appartement}
 MV = CAS P1 DANS ENTIER OU V DANS {oui,non} : achvente / CONTRADICTION
 FCAS
- R13 :
- <montant(DO.demande.MV)> -> <<loyerdem(L1.L2)>>
 OU DO DANS {maison,maisonnette,appartement,chambre}
 MV = CAS L1 DANS ENTIER : location
 FCAS
- R15 :
- <montant(DO.demande.achvente)> -> <<<prixdem>>> <<<viager>>>
 OU DO DANS {maison,maisonnette,appartement}

R16 :

⟨montant(DO.offre.achvente)⟩ -> <<<prixoff>> <<<viager>>
OU DO DANS {maison,maisonnette,appartement}

R17 :

⟨montant(DO.offre.location)⟩ -> <<<loyeroff>>

R19 :

⟨montant(DO.demande.location)⟩ -> <<<loyerdem>>

Commentaires :

La règle R1 permet de synthétiser les valeurs des attributs SV et MV à partir de celles élaborées au niveau des grammaires sens et mode.

Dans la règle R3 seul l'attribut DO est synthétisé, les autres sont hérités des caractéristiques.

Lorsqu'un attribut a la valeur SANSOBJ (exemple règle R4) cela signifie que cet attribut n'a pas de raison d'être. On est cependant tenu de le faire apparaître par souci d'homogénéité. Par exemple le noeud caractéristiques possède toujours 6 attributs, certains d'entre eux étant sans objet (NBPD2 dans R4 ; PR, ST, NBPD2 dans R6, etc.).

Dans R4 on vérifie que le nombre de pièces d'une maisonnette est au plus égal à 2. Si l'annonce ne le spécifie pas, le nombre de pièces est positionné à 2.

Dans R10, la notation achvente/CONTRADICTION signifie que la valeur à affecter à MV est achvente mais que si MV se trouve déjà instantané alors un message de contradiction doit être édité.

Lorsqu'un élément est entre triples chevrons (règles R15 et R19), la dérivation dans le vide de cet élément permet l'application de la règle. Ce type de règle doit apparaître comme possibilité ultime et donc figurer après les règles plus "fiables".

GRAMMAIRE D'ANALYSE LINGUISTIQUE

```

<<sens(offre)('vous avez')>> -> offre | dispose | a saisir
<<sens(demande)('vous cherchez')>> -> cherche | recherche | ch
EXCEPTION (sens) : <<sens(inc)('')>>

<<mode(achat)('a acheter')>> -> achete | acheterait
<<mode(vente)('a vendre')>> -> vends | cede | vds | cederait
<<mode(offloc)('a louer')>> -> dispose | loue
<<mode(demloc)('a louer')>> -> louerait | loue
<<mode(achmou)('a acheter')>> -> a acheter | a acquerir
<<mode(vtemou)('a vendre')>> -> a vendre
<<mode(locmou)('a louer')>> -> a louer | location
EXCEPTION (mode) : <<mode(achvente)('')>> | <<mode(location)('')>>

<<nature(maison)('une maison')>> -> maison
<<nature(maison)('une maison individuelle')>> -> maison individuelle
<<nature(maison)('un pavillon')>> -> pavillon
<<nature(villa)('une villa')>> -> villa
<<nature(maisonnette)('une maisonnette')>> -> maisonnette
<<nature(appartement)('un appartement')>> -> appartement | appt |
    appart
<<nature(chambre)('une chambre')>> -> chambre | une ch | chbre
EXCEPTION (nature) : OBLIGATOIRE

<<terrain(oui.ST)('avec terrain')>> -> terrain
<<terrain(oui.ST)('avec jardin')>> -> jardin
<<terrain(oui.x)('avec %chx% [m2|m] de terrain')>>
    -> <<nombre(x)(chx)>> [m2 | m] de [terrain |
        fardin ]
        [terrain | jardin] <<nombre(x)(chx)>> m2
<<terrain(non.ST)('sans terrain')>> -> sans terrain
<<terrain(non.ST)('sans jardin')>> -> sans jardin
EXCEPTION (terrain) : <<terrain(inc.inc)('')>>

<<nbpoff(x)('comprenant %chx% pieces')>> -> <<nombre(x)(chx)>>
    pieces |
    type <<nombre(x)(chx)>> |
    F <<nombre(x)(chx)>>
EXCEPTION (nbpoff) : <<nbpoff(inc)('')>>

<<nbpdem(x.Ø)('comportant au moins %chx% pieces')>>
    -> <<nombre(x)(chx)>> pieces [au | ] [minimum | min]
<<nbpdem(x.x)('comportant %chx% pieces')>> -> <<nombre(x)(chx)>>
    pieces |

```

```

type <<nombre(x)(chx)>> |
F <<nombre(x)(chx)>>
<<nbpdem(x1.x1)('comportant'%chx%'pieces')>> -> F <<nombre(x)(chx)>>
bis
OU x1 = x + 1
<<nbpdem(x.y)('comportant entre'%chx%'et'%chy%'pieces')>> ->
entre <<nombre(x)(chx)>> et <<nombre(y)(chy)>> pieces|
<<nombre(x)(chx)>> [ou | ] <<nombre(y)(chy)>> pieces
EXCEPTION (nbpdem) : <<nbpdem(inc.inc)('')>>

<<surface(x)('d une surface habitable de'%chx%'m2')>> ->
<<nombre(x)(chx)>> m2 habitable |
surface habitable <<nombre(x)(chx)>>
m2
OU x < 300
EXCEPTION (surface) : <<surface(inc)('')>>

<<nville(x.y.r)('a'%v)>> -> <<ville(x.y.r)(v)>>
<<nville(x.y.r)('aux environs de'%v)>> -> proximite <<ville(x.y.r)(v)
|
<<ville(x.y.r)(v)>>
ou environs
<<nville(x.y.rl)('dans la region de'%v)>> -> region
<<ville(x.y.r)(v)>>
OU rl = r + 20
EXCEPTION (nville) : <<nville(inc.inc.inc)('')>>

<<ville(49.33.0)('acigne')>> -> acigne
<<ville(42.38.0)('betton')>> -> betton
<<ville(44.31.0)('cesson')>> -> cesson
<<ville(52.23.0)('chateaugiron')>> -> chateaugiron
<<ville(37.23.0)('chartres de bretagne')>> -> chartres de bretagne
<<ville(50.25.0)('domloup')>> -> domloup
<<ville(28.15.0)('guichen')>> -> guichen
<<ville(30.30.0)('le rheu')>> -> le rheu
<<ville(52.42.0)('liffre')>> -> liffre
<<ville(25.27.0)('mordelles')>> -> mordelles
<<ville(50.30.0)('noyal sur vilaine')>> -> noyal sur vilaine
<<ville(38.30.3)('rennes')>> -> rennes
<<ville(40.30.0)('rennes est')>> -> rennes est
<<ville(38.32.0)('rennes nord')>> -> rennes nord
<<ville(36.30.0)('rennes ouest')>> -> rennes ouest
<<ville(38.28.0)('rennes sud')>> -> rennes sud
<<ville(45.47.0)('saint aubin d aubigne')>> -> saint aubin d aubigne
<<ville(38.35.0)('saint gregoire')>> -> saint gregoire
<<ville(34.27.0)('saint jacques de la lande')>> -> saint jacques
<<ville(17.92.0)('saint malo')>> -> saint malo

<<prixdem(x.x)('pour un prix de'%chx%'francs')>> -> <<nombre(x)(chx)>>
[f | francs]
OU x > 50000
<<prixdem(x.x)('pour un prix de'%chx%'francs')>> -> prix
<<nombre(x)(chx)>>
[f | francs]
<<prixdem(0.x)('pour un prix maximum de'%chx%'francs')>> ->

```

```

<<nombre(x)(chx)>> [f | francs]
[ max | maximum]
OU x > 50000
<<prixdem(Ø.x)('pour un prix maximum de'%chx%'francs')>> ->
    prix [max | maximum] <<nombre(x)(chx)>>
    [f | francs]
EXCEPTION (prixdem) : <<prixdem(inc.inc)('')>>

<<prixoff(x)('a un prix de'%chx%'francs')>> -> <<nombre(x)(chx)>>
    [f | francs]
OU x > 50000
<<prixoff(x)('a un prix de'%chx%'francs')>> -> prix <<nombre(x)(chx)>>
    [f | francs]
EXCEPTION (prixoff) : <<prixoff(inc)('')>>

<<loyerdem(x.y)('pour un loyer compris entre'%chx%'et'%chy%'francs')
-> loyer entre <<nombre(x)(chx)>> et <<nombre(y)(chy)>>
    [francs | f]
<<loyerdem(x.x)('pour un loyer de'%chx%'francs')>>
-> loyer <<nombre(x)(chx)>> [ F | francs]
<<loyerdem(x.x)('pour un loyer de'%chx%'francs')>> -> <<nombre(x)(chx)>>
    [f | francs]
OU x < 10000
<<loyerdem(Ø.x)('pour un loyer de'%chx%'francs')>> -> loyer
    [maximum | max]
    <<nombre(x)(chx)>>
    [f | francs]
<<loyerdem(Ø.x)('pour un loyer maximum de'%chx%'francs')>>
-> <<nombre(x)(chx)>> [f | francs] [max | maximum]
OU x < 10000
EXCEPTION (loyerdem) : <<loyerdem(inc.inc)('')>>

<<loyeroff(x)('pour un loyer de'%chx%'francs')>> -> <<nombre(x)(chx)>>
    [f | francs]
OU x < 10000
<<loyeroff(x)('pour un loyer de'%chx%'francs')>> -> loyer
    <<nombre(x)(chx)>>
    [f | francs]
EXCEPTION (loyeroff) : <<loyeroff(inc)('')>>

<<viager(oui)('en viager')>> -> viager
<<viager(non)('non en viager')>> -> pas de viager
EXCEPTION (viager) : <<viager(inc)('')>>

```

SPECIFICATIONS DE LA COMPARAISON

 TERME verbe : ((CHAINE senso,modeo),(CHAINE sensd,moded)) =

(verbe{sv,mv},verbe{sv,mv}) ;

COMPDIS(d)/ø/ø/ø/

OU d = CAS (senso,sensd) DANS (offre,demande) ET
 (modeo,moded) DANS {(achvente,achvente),
 (location,location)} : ø

AUTREMENT : 1

FCAS

FCAS ;

 TERME nbpiece : ((ENT nbpo),(ENT nbpd1,nbpd2)) =

(nbpoff{x},nbpdem{y1,y2}) ;

COMPSEG(s1,s2)

OU s1 = CAS nbpo = inc : (ø,99)

AUTREMENT : (nbpo,nbpo)

FCAS

s2 = CAS nbpd1 = ø : SEGVA (nbpd2,nbpd2)/(1,ø)/(2,1)/(3,1) ,
 nbpd2 = ø : SEGVA (nbpd1,nbpd1)/(ø,1)/(1,2)/(1,3) ,
 nbpd1 = inc : (ø,99) ,
 nbpd1 = nbpd2 : SEGVA (nbpd1,nbpd1)/(ø,1)/(1,2)/(2,3) ,
 nbpd1 < nbpd2 : SEGVA (nbpd1,nbpd2)/(ø,ø)/(1,1)/(1,2)
 nbpd1 > nbpd2 : SEGVA (nbpd2,nbpd1)/(ø,ø)/(1,1)/(1,2)

FCAS ;

 TERME lieu : ((ENT xo,yo,ro),(ENT xd,yd,rd)) =

(nville{x,y,r},nville{x,y,r}) ;

COMPSURF(s1,s2)

OU s1 = CAS xo DANS {inc} : CERCLE (ø,ø,9999)

AUTREMENT : CERCLE (xo,yo,ro)

FCAS

```
s2 = CAS xd DANS {inc} : CERCLE (0,0,9999)
AUTREMENT : VCERCLE (xd,yd,rd,0,10,15)
FCAS ;
```

```
*****
TERME prix : ((ENT monto),(ENT montd1,montd2)) =
*****
```

```
(prixoff{x},prixdem{y1,y2}),(loyeroff{x},loyerdem{y1,y2}) ;
```

```
COMPSEG(s1,s2)
```

```
OU s1 = CAS monto = inc : (0,999999999999)
AUTREMENT : (monto,monto)
FCAS
```

```
s2 = CAS montd1 = 0 : SEGPC (montd2,montd2)/(10,0)/(20,10)/(30,20),
montd2 = 0 : SEGPC (montd1,montd1)/(0,10)/(10,20)/(20,30),
montd1 = inc : (0,999999999999),
montd1 = montd2 : SEGPC (montd1,montd1)/(10,10)/
(20,20)/(30,30)/,
montd1 < montd2 : SEGPC (montd1,montd2)/(0,0)/
(10,10)/(20,20)/,
montd2 < montd1 : SEGPC (montd2,montd1)/(0,0)/
(10,10)/(20,20)/,
```

```
FCAS ;
```

```
*****
TERME viager : ((CHAINE vo),(CHAINE vd)) =
*****
```

```
(viager{v},viager{v}) ;
```

```
COMPDIS(d)/2/4/6/
```

```
OU d = CAS (vo,vd) DANS {(oui,oui),{(inc,non},{inc,non)}} : 0
AUTREMENT : 5
FCAS ;
```

```
*****
TERME terrain : ((CHAINE pto,ENT sto),(CHAINE ptd,ENT std)) =
*****
```

```
(terrain{pt,st},terrain{pt,st}) ;
```

```
COMPDIS(d)/2/4/6/
```

```
OU d = CAS (pto,ptd) DANS {(inc,{oui,non}),(oui,non)} : 3 ,
(pto,ptd) DANS (non,oui) : 5 ,
(pto,ptd) DANS {(inc,inc),(non,non)} : 0,
(pto,ptd) DANS (oui,oui) :
CAS sto = inc : 0 ,
std = inc : 0
AUTREMENT : CAS BSEGPC(sto,std,20,20) : 1 ,
BSEGPC(sto,std,30,30) : 3 ,
BSEGPC(sto,std,40,40) : 5
```

AUTREMENT : 9
FCAS

FCAS
AUTREMENT : 0
FCAS ;

CO une variante de la procedure de comparaison est possible par le biais de :

COMPSEG(s1,s2)
OU s1 = intervalle offre
s2 = intervalle demande

FCO

TERME des : ((CHAINE deso,pto,ENT sto,nbpo),(CHAINE desd,
ptd,ENT std,nbpd1,nbpd2))

(designation{nature{dol},pt,st,nbp},designation{nature{dol},
pt,st,nbpl,nbp2});

COMPDIS(d)/2/4/6/

OU d = CAS (deso,desd) DANS ({maison,maisonnette},
{maison,maisonnette}):
0,

(deso,desd) DANS (villa,villa) : 0 ,
(deso,desd) DANS {(appartement,appartement),
(chambre,chambre)} : 0 ,

(deso,desd) DANS (villa,maison) : 1 ,
(deso,desd) DANS ({maison,maisonnette},
appartement) : 3 ,

(deso,desd) DANS (chambre,appartement) :
CAS nbpd2 <= 2 : 5
AUTREMENT : 9
FCAS

(deso,desd) DANS (appartement,chambre) :
CAS nbpo < 2 : 0 ,
nbpo < 3 : 3
AUTREMENT : 9
FCAS

(deso,desd) DANS (appartement,maison) :
CAS ptd = oui : 9
AUTREMENT : 5
FCAS

(deso,desd) DANS (maison,villa) :
CAS std = inc ET nbpd1 = inc :
CAS pto = oui :
CAS nbpo >= 4 :
CAS sto >= 1000 : 3,
sto < 1000 : 5
AUTREMENT : 5
FCAS
nbpo = inc :

Al.11

```

    CAS sto = inc : 9,
      sto >= 1000 : 3,
      sto < 1000 : 5
  FCAS
  nbpo < 4 :
    CAS sto >= 1000 : 5 ,
      sto < 1000 : 9 ,
      sto = inc : 9
  FCAS
  FCAS
  AUTREMENT : 9
  FCAS
  std ^= inc ET nbpd1 ^= inc :
  CAS pto = oui :
    CAS nbpd1 = nbpd2 :
      CAS nbpo >= nbpd1 :
        CAS sto >= std : 0 ,
          sto = inc : 3
        AUTREMENT : 3
      FCAS
    AUTREMENT : 5
  FCAS
  AUTREMENT : 5
  FCAS
  AUTREMENT : 9
  FCAS
  AUTREMENT : CAS pto = oui :
    CAS nbpo >= 4 :
      CAS sto >= 1000 : 2,
        sto < 1000 : 5
      AUTREMENT : 5
    FCAS
    nbpo = inc :
      CAS sto >= 1000 : 2
      AUTREMENT : 5
    FCAS
  AUTREMENT : 9
  FCAS
  AUTREMENT : 9
  FCAS
  FCAS
  AUTREMENT : 9
  FCAS ;

```

A2.1

ANNEXE 2

=====

- EXEMPLE D'UTILISATION DE LA MAQUETTE -

S Y S T E M E H A V A N E

B O N J O U R . . .

Version : Ma, 82

I R I S A

A Entrez votre annonce :

+VDS MAISONNETTE 3 PIECES A BETTON

NOUS AVONS COMPRIS QUE:

VOUS AVEZ A VENDRE UNE MAISONNETTE,
COMPRENANT 3 PIECES-
A BETTON

La comprehension est-elle correcte?
+0

Recherche et Edition
des annonces correspondant
a la votre

Desole,
il n'y a aucune annonce satisfaisante

Voulez-vous que nous recherchions
des annonces un peu moins proches
de la votre
+N

Voulez-vous stocker votre annonce?
+N

Voulez-vous entrer une autre annonce?
+0

B)

Entrez votre annonce :

+VDS MAISONNETTE 4 PIECES A DESSON
 PRIX 150000 F

CONTRADICTION

entre

la designation maisonnette
 et 4 pieces

Voulez-vous entrer une autre annonce?

+D

Entrez votre annonce :

C)

-LOUE MAISON 5 PIECES A RENNES SUD EN
 VIAGER

CONTRADICTION

entre

le mode de la transaction : location
 et le viager

Nous avons compris que:

vous avez a louer une maison,
 comprenant 5 pieces,
 a rennes sud

La comprehension est-elle correcte?

+D

Recherche et Edition
 des annonces correspondant
 a la votre

Desole,

il n'y a aucune annonce satisfaisante

Voulez-vous que nous recherchions
 des annonces un peu moins proches
 de la votre

+N

Voulez-vous stocker votre annonce?

+N

Voulez-vous entrer une autre annonce?

+



Entrez votre annonce :

+ LOUE APPT 5 PIECES PRIX 55000 F A CESSO

CONTRADICTION

erreur

le mode de la transaction : location
et le prix 55000 f

Nous avons comparé que:

vous avez a louer un appartement,
comportant 5 pieces,
a cession

La compréhension est-elle correcte?

+0

Recherche et Edition
des annonces correspondant
a la votre

Desole,

il n'y a aucune annonce satisfaisante

Voulez-vous que nous recherchions
des annonces un peu moins proches
de la votre

+X

Voulez-vous stocker votre annonce?

+X

Voulez-vous entrer une autre annonce?

+0

(E) Entrez votre annonce :

→ LOUE MAISON TYPE 4 A SAINT JACQUES
TERRAIN 200 M2 HABITABLE

Le resultat de la rubrique suivante
est-il correct?

surface : d'une superficie habitable d
e 200 m2 :

+N

Nous avons compris que:

vous avez a louer une maison,
avec 200 m2 de terrain,
comprenant 4 pieces,
a saint Jacques de la lande

La comprehension est-elle correcte?

+D

Recherche et Edition
des annonces correspondant
a la votre

Desole,

il n'y a aucune annonce satisfaisante

Voulez-vous que nous recherchions
des annonces un peu moins proches
de la votre

+N

Voulez-vous stocker votre annonce?

+D

Voulez-vous entrer une autre annonce?

+D

Ⓕ Entrez votre annonce :

+OFFRE APPART 5 PIECES RENNES CENTRE

Nous avons compris que:

vous avez a vendre un appartement,
comprenant 5 pieces,
a rennes

La comprehension est-elle correcte?
+N

Nous avons compris que:

vous avez a louer un appartement,
comprenant 5 pieces,
a rennes

La comprehension est-elle correcte?
+X

Voulez-vous entrer une autre annonce?
+0

Ⓖ Entrez votre annonce :

+OFFRE APPT 5 PIECES 450000 F BETTON

Nous avons compris que:

vous avez a vendre un appartement,
comprenant 5 pieces,
a betton,
a un prix de 450000 francs

La comprehension est-elle correcte?
+N

Nous avons compris que:
vous avez a louer un appartement,
comprenant 5 pieces,
a betton

La comprehension est-elle correcte?
+X

Voulez-vous entrer une autre annonce?
+0

(H) Entrez votre annonce :

+CHERCHE MAISON 4 6 PIECES A RENNES OU E
NVIRONS ENTRE 1500 ET 2300 F

Nous avons compris que:

vous cherchez a louer une maison,
comportant entre 4 et 6 pieces,
aux environs de rennes,
pour un loyer compris entre 1500 et 23
00 francs

La comprehension est-elle correcte?
+N

Nous avons compris que:

vous cherchez a louer une maison,
comportant entre 4 et 6 pieces,
aux environs de rennes,
pour un loyer de 2300 francs

La comprehension est-elle correcte?
+N

Nous avons compris que:

vous cherchez a acheter une maison,
comportant entre 4 et 6 pieces, aux env
irons de rennes

La comprehension est-elle correcte?
+N

Nous avons compris que:

vous cherchez a louer une maison,
comportant entre 4 et 6 pieces,
a rennes,
pour un loyer compris entre 1500 et 23
00 francs

La comprehension est-elle correcte?
+N

Nous avons compris que:

vous cherchez a louer une maison,
comportant entre 4 et 6 pieces,
a rennes,
pour un loyer de 2300 francs

La comprehension est-elle correcte?
+N

Nous avons compris que:

A2.8

vous cherchez a acheter une maison,
comportant entre 4 et 6 pieces,
a rennes

La comprehension est-elle correcte?
+N

Nous avons compris que:

vous cherchez a louer une maison,
comprenant 5 pieces,
aux environs de rennes,
pour un loyer compris entre 1500 et 23
00 francs

La comprehension est-elle correcte?
+N

Voulez-vous entrer une autre annonce?

+0

Entrez votre annonce :

① →CHERCHE A LOUER MAISON 4 6 PIECES RENNE
C 2222 P MAXIMUM

Nous avons compris que:
vous cherchez a louer une maison,
comportant entre 4 et 6 pieces,
a rennes,
pour un loyer maximum de 2000 francs

La comprehension est-elle correcte?
+0

Recherche et Edition
des annonces correspondant
a la votre

Desole,
il n'y a aucune annonce satisfaisante

Voulez-vous que nous recherchions
des annonces un peu moins proches
de la votre

+N

Voulez-vous stocker votre annonce?

+0

Voulez-vous entrer une autre annonce?

+0

①

Entrez votre annonce :

+CHERCHE MAISON A LOUER 5 A 7 PIECES BET
TON OU ENVIRONS

Nous avons compris que:

vous cherchez a louer une maison,
comprenant 7 pieces,
aux environs de betton

La comprehension est-elle correcte?

+0

Recherche et Edition
des annonces correspondant
a la votre

Desole,

il n'y a aucune annonce satisfaisante

Voulez-vous que nous recherchions
des annonces un peu moins proches
de la votre

+N

Voulez-vous stocker votre annonce?

+0

Voulez-vous entrer une autre annonce?

+0

(K) Entrez votre annonce :

→CEDE APPARTEMENT F5 A RENNES 300000 F

Nous avons compris que:

vous avez a vendre un appartement,
comprenant 5 pieces,
a rennes,
a un prix de 300000 francs

La comprehension est-elle correcte?

+0

Recherche et Edition
des annonces correspondant
a la votre

Desole,

il n'y a aucune annonce satisfaisante

Voulez-vous que nous recherchions
des annonces un peu moins proches
de la votre

+N

Voulez-vous stocker votre annonce?

+0

Voulez-vous entrer une autre annonce?

+0

Ⓛ Entrez votre annonce :

+CHERCHE MAISON TYPE 5 AVEC JARDIN
RENNES SUD 500000F MAX

Nous avons compris que:

vous cherchez a acheter une maison,
avec jardin,
comprenant 5 pieces,
a rennes sud,
a un prix maximum de 500000 francs

La comprehension est-elle correcte?

→O

Recherche et Edition
des annonces correspondant
a la votre
vends maison type 5 terrain 500 m 2 48
0000 f

Etes-vous satisfait?

→N

Voulez-vous que nous recherchions
des annonces un peu moins proches
de la votre

→O

vds pavillon type 4 400000 f rennes e

s
t

vds maison type 6 2000 m 2 de terrain
a saint gregoire

vds pavillon type 5 avec 1500 m 2 de t
errain a saint gregoire

vds pavillon 5 pieces terrain cesson s
evigne 500000 f

vds maison type 4 avec jardin rennes o
uest

Etes-vous satisfait?

→O

→O Voulez-vous stocker votre annonce?

(M)

Entrez votre annonce :

A2.12

+CHERCHE PAVILLON AVEC GRAND JARDIN
PROXIMITE RENNES

Nous avons compris que:

vous cherchez a acheter un pavillon,
avec jardin,
aux environs de rennes

La comprehension est-elle correcte?

+N

Nous avons compris que:

vous cherchez a louer un pavillon,
avec jardin,
aux environs de rennes

La comprehension est-elle correcte?

+O

Recherche et Edition
des annonces correspondant
a la votre

Desole,

il n'y a aucune annonce satisfaisante

Voulez-vous que nous recherchions
des annonces un peu moins proches
de la votre

+O

loue maison type 4 a saint jacques ter
rain 200 m 2 habitable

Etes-vous satisfait?

+N

Voulez-vous que nous recherchions
des annonces encore moins proches
de la votre

+O

Desole,

il n'y a aucune annonce satisfaisante

Desole de ne
pouvoir mieux faire

Voulez-vous stocker votre annonce?

+O

(2) Entrez votre annonce :

→ACHETERAIT APPART A RENNES 4 A 7 PIECES

Vous avons compris que:

vous cherchez a acheter un appartement
comprenant 7 pieces,
a rennes

La comprehension est-elle correcte?→0

Recherche et Edition
des annonces correspondant
a la votre

Desole,
il n'y a aucune annonce satisfaisante

Voulez-vous que nous recherchions
des annonces un peu moins proches
de la votre

→0

Desole,
il n'y a aucune annonce satisfaisante

Voulez-vous que nous recherchions
des annonces encore moins proches
de la votre

→0

cede appartement f 5 a rennes 300000 f

Etes-vous satisfait?

→N

Desole de ne
pouvoir mieux faire

Voulez-vous stocker votre annonce?

→0

Voulez-vous entrer une autre annonce?

→0

⊙ Entrez votre annonce :

+ACHETE MAISON 4 6 PIECES

Nous avons compris que:

vous cherchez a acheter une maison,
comportant entre 4 et 6 pieces

La comprehension est-elle correcte?
+O

Recherche et Edition
des annonces correspondant
a la votre

vds pavillon type 4 400000 f rennes es
t

vds maison 6 pieces 1800 m 2 de terrai
n guichen prix 620000 f

vds maison type E 2000 m 2 de terrain
a saint gregoire

vds pavillon type 5 avec 1500 m 2 de t
errain a saint gregoire

vends maison type 5 terrain 500 m 2 48
0000 f

vds pavillon 5 pieces terrain cesson s
eviane 500000 f

vds maison type 4 avec jardin rennes o
uest

Etes-vous satisfait?

+O

Voulez-vous stocker votre annonce?

+N

Voulez-vous entrer une autre annonce?

+N

Commentaires sur les annonces de l'annexe 3 :

L'annonce A montre qu'une maisonnette de 3 pièces est acceptée par le système alors que l'annonce B provoque une contradiction car le nombre de pièces de la maisonnette excède 3.

En C et D, on met en évidence des contradictions dues à la présence d'un viager ou d'un prix dans une location.

L'exemple E illustre un cas d'ambiguïté, car le texte terrain 200 m² habitable peut être interprété ou bien comme "le terrain a 200 m² et la maison est habitable" ou alors comme "il y a un terrain et la surface habitable est de 200 m²".

Les exemples F, G et H mettent en évidence la possibilité d'interprétation du système. En F et G, le verbe offre ne permet pas de savoir s'il s'agit d'une vente ou d'une location et le système propose les deux solutions. Dans l'exemple H, il y a tentative d'interprétation sur le mode (achat-vente ou location), le nombre de pièces, le lieu et le prix.

Les annonces I et J montrent que les écritures "cherche à louer maison" et "cherche maison à louer" sont équivalentes et conduisent à la même interprétation.

Les exemples K à O mettent en lumière les diverses possibilités offertes lors de la recherche des annonces complémentaires.

Liste des Publications Internes IRISA

- PI 150 **Construction automatique et évaluation d'un graphe d'«implication» issu de données binaires, dans le cadre de la didactique des mathématiques**
H. Rostam , 112 pages ; Juin 1981
- PI 151 **Réalisation d'un outil d'évaluation de mécanismes de détection de pannes]-[Projet Pilote SURF**
B. Decouty, G. Michel, C. Wagner, Y. Crouzet , 59 pages ; Juillet 1981
- PI 152 **Règle maximale**
J. Pellaumail , 18 pages ; Septembre 1981
- PI 153 **Corrélation partielle dans le cas « qualitatif »**
I.C. Lerman , 125 pages ; Octobre 1981
- PI 154 **Stability analysis of adaptively controlled not-necessarily minimum phase systems with disturbances**
Cl. Samson , 40 pages ; Octobre 1981
- PI 155 **Analyses d'opinions d'instituteurs à l'égard de l'appropriation des nombres naturels par les élèves de cycle préparatoire**
R. Gras , 37 pages ; Octobre 1981
- PI 156 **Récursion induction principle revisited**
G. Boudol, L. Kott , 49 pages ; Décembre 1981
- PI 157 **Loi d'une variable aléatoire à valeur R^+ réalisant le minimum des moments d'ordre supérieur à deux lorsque les deux premiers sont fixés**
M.Kowalowka, R. Marie , 8 pages ; Décembre 1981
- PI 158 **Réalisations stochastiques de signaux non stationnaires, et identification sur un seul échantillon**
A. Benveniste J.J. Fuchs , 33 pages ; Mars 1982
- PI 159 **Méthode d'interprétation d'une classification hiérarchique d'attributs-modalités pour l'«explication» d'une variable ; application à la recherche de seuil critique de la tension artérielle systolique et des indicateurs de risque cardiovasculaire**
B. Tallur , 34 pages ; Janvier 1982
- PI 160 **Probabilité stationnaire d'un réseau de files d'attente multiclasse à serveur central et à routages dépendant de l'état**
L.M. Le Ny , 18 pages ; Janvier 1982
- PI 161 **Détection séquentielle de changements brusques des caractéristiques spectrales d'un signal numérique**
M. Basseville, A. Benveniste , pages ; Mars 1982
- PI 162 **Actes regroupés des journées de Classification de Toulouse (Mai 1980), et de Nancy (Juin 1981)**
I.C. Lerman , 304 pages ;
- PI 163 **Modélisation et Identification des caractéristiques d'une structure vibratoire : un problème de réalisation stochastique d'un grand système non stationnaire**
M. Prévosto, A. Benveniste, B. Barnouin , 46 pages ; Mars 1982
- PI 164 **An enlarged definition and complete axiomatization of observational congruence of finite processes**
Ph. Darondeau , 45 pages ; Avril 1982
- PI 165 **Accès vidéotex à une banque de données médicales**
A. Chauffaut, M. Dragone, R. Rivoire, J.M. Roger , 25 pages ; Mai 1982
- PI 166 **Comparaison de groupes de variables définies sur le même ensemble d'individus**
B. Escofier, J. Pages , 115 pages ; Mai 1982
- PI 167 **Transport en circuits virtuels internes sur réseau local et connexion Transpac**
M. Tournois, R. Trépos , 90 pages ; Mai 1982
- PI 168 **Impact de l'intégration sur le traitement automatique de la parole**
P. Quinton , 14 pages ; Mai 1982
- PI 169 **A systolic algorithm for connected word recognition**
J.P. Banâtre, P. Frison, P. Quinton , 13 pages ; Mai 1982
- PI 170 **A network for the detection of words in continuous speech**
J.P. Banâtre, P. Frison, P. Quinton , 24 pages ; Mai 1982
- PI 171 **Le langage ADA : Etude bibliographique**
J. André, Y. Jégou, M. Raynal , 12 pages ; Juin 1982
- PI 172 **Comparaison de groupes de variables : 2ème partie : un exemple d'application**
B. Escofier, J. Pajès , 37 pages ; Juillet 1982
- PI 173 **Unfold-fold program transformations**
L. Kott , 29 pages ; Juillet 1982
- PI 174 **Remarques sur les langages de parenthèses**
J.M. Autebert, J. Beauquier, L. Boasson, G. Senizergues , 20 pages ; Juillet 1982
- PI 175 **Langages de parenthèses, langages N.T.S. et homomorphismes inverses**
J.M. Autebert, L. Boasson, G. Senizergues , 26 pages ; Juillet 1982
- PI 176 **Tris pour machines synchrones ou Baudet Stevenson revisited**
R. Rannou , 26 pages ; Juillet 1982
- PI 177 **Un nouvel algorithme de classification hiérarchique des éléments constitutifs de tableau de contingence basé sur la corrélation**
B. Tallur , Juillet 1982 ;
- PI 178 **Programmes d'analyse des résultats d'une classification automatique**
I.C. Lerman et collaborateurs , 79 pages ; Septembre 1982
- PI 179 **Attitude à l'égard des mathématiques des élèves de sixième**
J.Degouys, R. Gras, M. Postic , 29 pages ; Septembre 1982
- PI 180 **Traitements de textes et manipulations de documents : bibliographie analytique**
J. André , 20 pages ; Septembre 1982

- PI 181 **Algorithme assurant l'insertion dynamique d'un processeur autour d'un réseau à diffusion et garantissant la cohérence d'un système de numérotation des paquets global et réparti**
Annick Le Coz, Hervé Le Goff, Michel Ollivier . 31 pages ; Octobre 1982
- PI 182 **Interprétation non linéaire d'un coefficient d'association entre modalités d'une juxtaposition de tables de contingence**
Israël César Lerman . 34 pages ; Novembre 1982
- PI 183 **L'IRISA vu à travers les stages effectués par ses étudiants de DEA (1^è année de thèse)**
Daniel Herman . 41 pages ; Novembre 1982
- PI 184 **Commande non linéaire robuste des robots manipulateurs**
Claude Samson . 52 pages ; Janvier 1983
- PI 185 **Dialogue et représentation des informations dans un système de messagerie intelligent**
Philippe Besnard, René Quiniou, Patrice Quinton, Patrick Saint-Dizier, Jacques Siroux, Laurent Trilling . 45 pages ; Janvier 1983
- PI 186 **Analyse classificatoire d'une correspondance multiple ; typologie et régression**
I.C. Lerman . 54 pages ; Janvier 1983
- PI 187 **Estimation de mouvement dans une sequence d'images de télévision en vue d'un codage avec compensation de mouvement**
Claude Labit . 132 pages ; Janvier 1983
- PI 188 **Conception et réalisation d'un logiciel de saisie et restitution de cartes élémentaires**
Eric Sécher . 45 pages ; Janvier 1983
- PI 189 **Etude comparative d'algorithmes pour l'amélioration de dessins au trait sur surfaces point par point**
M.A. ROY . 96 pages ; Janvier 1983
- PI 190 **Généralisation de l'analyse des correspondances à la comparaison de tableaux de fréquence**
Brigitte Escofier . 35 pages ; Mars 1983
- PI 191 **Association entre variables qualitatives ordinales «nettes» ou «floues»**
Israël-César Lerman . 42 pages ; Mars 1983
- PI 192 **Un processeur intégré pour la reconnaissance de la parole**
Patrice Frison . 80 pages ; Mars 1983
- PI 193 **The Systematic Design of Systolic Arrays**
Patrice Quinton . 39 pages ; Avril 1983
- PI 194 **Régime stationnaire pour une file M/H/1 avec impatience**
Raymond Marie et Jean Pellaumail . 8 pages ; Mars 1983
- PI 195 **SIGNAL : un langage pour le traitement du signal**
Paul Le Guernic, Albert Benveniste, Thierry Gautier . 49 pages ; Mars 1983
- PI 196 **Algorithmes systoliques : de la théorie à la pratique**
Françoise André, Patrice Frison, Patrice Quinton . 19 pages ; Mars 1983
- PI 197 **HAVANE : un système de mise en relation automatique de petites annonces**
Patrick Bosc, Michèle Courant, Sophie Robin, Laurent Trilling . 79 pages ; Mai 1983

Imprimé en France

par

l'Institut National de Recherche en Informatique et en Automatique

