



**HAL**  
open science

## Le projet ABREGÉ

Monique Chabre-Peccoud

► **To cite this version:**

Monique Chabre-Peccoud. Le projet ABREGÉ. Modélisation et simulation. Institut National Polytechnique de Grenoble - INPG; Université Joseph-Fourier - Grenoble I, 1976. Français. NNT : . tel-00286504

**HAL Id: tel-00286504**

**<https://theses.hal.science/tel-00286504>**

Submitted on 9 Jun 2008

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

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

**THESE**

**présentée à**

**UNIVERSITE SCIENTIFIQUE ET MEDICALE DE GRENOBLE**

**INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE**

POUR OBTENIR LE GRADE DE  
DOCTEUR DE 3ème CYCLE  
Spécialité Informatique

**Monique CHABRE-PECCOUD**

**LE PROJET ABREGÉ**

Thèse soutenue le 4 juin 1976 devant la Commission d'Examen : \_\_\_\_\_

Président : B. VAUQUOIS

Examineurs : J.C. BOUSSARD  
C. DELOBEL  
J.C. DERNIAME

Rapporteur : F. PECCOUD



UNIVERSITE SCIENTIFIQUE ET MEDICALE DE GRENOBLE

Monsieur Michel SOUTIF : Président

Monsieur Gabriel CAU : Vice-Président

---

MEMBRES DU CORPS ENSEIGNANTS DE L'U.S.M.G.

PROFESSEURS TITULAIRES

MM.	ANGLES D'AURIAC Paul	Mécanique des Fluides
	ARNAUD Paul	Chimie
	AUBERT Guy	Physique
	AYANT Yves	Physique Approfondie
Mme	BARBIER Marie-Jeanne	Electrochimie
MM.	BARBIER Jean-Claude	Physique Expérimentale
	BARBIER Reynold	Géologie Appliquée
	BARJON Robert	Physique Nucléaire
	BARNOUD Fernand	Biosynthèse de la Cellulose
	BARRA Jean-René	Statistiques
	BARRIE Joseph	Clinique Chirurgicale
	BEAUDOING André	Clinique de Pédiatrie et Puériculture
	BERNARD Alain	Mathématiques Pures
Mme	BERTRANDIAS Françoise	Mathématiques Pures
MM.	BERTRANDIAS Jean-Paul	Mathématiques Pures
	BEZES Henri	Pathologie Chirurgicale
	BLAMBERT Maurice	Mathématiques Pures
	BOLLIET Louis	Informatique (I.U.T. B)
	BONNET Georges	Electrotechnique
	BONNET Jean-Louis	Clinique Ophtalmologique
	BONNET-EYMARD Joseph	Clinique Gastro-entérologique
Mme	BONNIER Marie-Jeanne	Chimie Générale
MM.	BOUCHERLE André	Chimie et Toxicologie
	BOUCHEZ Robert	Physique nucléaire
	BOUSSARD Jean-Claude	Mathématiques Appliquées
	BRAVARD Yves	Géographie
	CABANEL Guy	Clinique Rhumatologique et Hydrologique
	CALAS François	Anatomie
	CARLIER Georges	Biologie Végétale
	CARRAZ Gilbert	Biologie Animale et Pharmacodynamie
	CAU Gabriel	Médecine Légale et Toxicologie
	CAUQUIS Georges	Chimie Organique
	CHABAUTY Claude	Mathématiques Pures
	CHARACHON Robert	Clinique Oto-Rhino-Laryngologique
	CHATEAU Robert	Clinique de Neurologie
	CHIBON Pierre	Biologie Animale
	COEUR André	Pharmacie Chimique et Chimie Analytique
	CONTAMIN Robert	Clinique Gynécologique
	COUDERC Pierre	Anatomie Pathologique
	CRAYA Antoine	Mécanique
Mme	DEBELMAS Anne-Marie	Matière Médicale
MM.	DEBELMAS Jacques	Géologie Générale
	DEGRANGE Charles	Zoologie
	DELORMAS Pierre	Pneumo-Phtisiologie
	DEPORTES Charles	Chimie Minérale
	DESRE Pierre	Métallurgie



MM.	DESSAUX Georges	Physiologie Animale
	DODU Jacques	Mécanique Appliquée (I.U.T. A)
	DOLIQUE Jean-Michel	Physique des Plasmas
	DREYFUS Bernard	Thermodynamique
	DUCROS Pierre	Cristallographie
	DUGOIS Pierre	Clinique de Dermatologie et Syphiligra
	GAGNAIRE Didier	Chimie Physique
	GALLISSOT François	Mathématiques Pures
	GALVANI Octave	Mathématiques Pures
	GASTINEL Noël	Analyse Numérique
	GAVEND Michel	Pharmacologie
	GEINDRE Michel	Electroradiologie
	GERBER Robert	Mathématiques Pures
	GERMAIN Jean-Pierre	Mécanique
	GIRAUD Pierre	Géologie
	JANIN Bernard	Géographie
	KAHANE André	Physique Générale
	KLEIN Joseph	Mathématiques Pures
	KOSZUL Jean-Louis	Mathématiques Pures
	KRAVTCHENKO Julien	Mécanique
	KUNTZMANN Jean	Mathématiques Appliquées
	LACAZE Albert	Thermodynamique
	LACHARME Jean	Biologie Végétale
Mme	LAJZEROWICZ Janine	Physique
MM.	LAJZEROWICZ Joseph	Physique
	LATREILLE René	Chirurgie Générale
	LATURAZE Jean	Biochimie Pharmaceutique
	LAURENT Pierre-Jean	Mathématiques Appliquées
	LEDRU Jean	Clinique Médicale B
	LLIBOUTRY Louis	Géophysique
	LOISEAUX Pierre	Sciences Nucléaires
	LONGEQUEUE Jean-Pierre	Physique Nucléaires
	LOUP Jean	Géographie
Melle	LUTZ Elisabeth	Mathématiques Pures
	MALGRANGE Bernard	Mathématiques Pures
	BOUTET DE MONVEL Louis	Mathématiques Pures
	MALINAS Yves	Clinique Obstétricale
	MARTIN-NOEL Pierre	Seméiologie médicale
	MAZARE Yves	Clinique Médicale A
	MICHEL Robert	Minéralogie et Pétrographie
	MICOUD Max	Clinique Maladies Infectieuses
	MOURIQUAND Claude	Histologie
	MOUSSA André	Chimie Nucléaire
	MULLER Jean-Michel	Thérapeutique (Néphrologie)
	NEEL Louis	Physique du solide
	OZENDA Paul	Botanique
	PAYAN Jean-Jacques	Mathématiques Pures
	PEBAY-PEYROULA Jean-Claude	Physique
	RASSAT André	Chimie Systématique
	RENARD Michel	Thermodynamique
	RINALDI Renaud	Physique
	DE ROUGEMONT Jacques	Neuro-Chirurgie
	SEIGNEURIN Raymond	Microbiologie et Hygiène
	SENGEL Philippe	Zoologie
	SIBILLE Robert	Construction Mécanique (I.U.T. A)
	SOUTIF Michel	Physique Générale
	TANCHE Maurice	Physiologie

MM.	TRAYNARD Philippe	Chimie Générale
	VAILLANT François	Zoologie
	VALENTIN Jacques	Physique Nucléaire
	VAUQUOIS Bernard	Calcul Electronique
Mme	VERAIN Alice	Pharmacie Galénique
MM.	VERAIN André	Physique
	VEYRET Paul	Géographie
	VIGNAIS Pierre	Biochimie médicale
	YOCOZ Jean	Physique Nucléaire Théorique

#### PROFESSEURS ASSOCIES

MM.	CLARK Gilbert	Spectrométrie Physique
	CRABBE Pierre	CERMO
	ENGLMAN Robert	Spectrométrie Physique
	HOLTZBERG Frédéric	Basses Températures
	ROST Ernest	Sciences Nucléaires

#### PROFESSEURS SANS CHAIRE

Melle	AGNIUS-DELDORD Claudine	Physique Pharmaceutique
	ALARY Josette	Chimie Analytique
MM.	AMBROISE-THOMAS Pierre	Parasitologie
	BELORIZKY Elie	Physique
	BENZAKEN Claude	Mathématiques Appliquées
	BIAREZ Jean-Pierre	Mécanique
	BILLET Jean	Géographie
	BOUCHET Yves	Anatomie
	BRUGEL Lucien	Energétique (I.U.T. A)
	BUISSON René	Physique (I.U.T. A)
	CONTE René	Physique (I.U.T. A)
	DEPASSEL Roger	Mécanique des Fluides
	GAUTHIER Yves	Sciences Biologiques
	GAUTRON René	Chimie
	GIDON Paul	Géologie et Minéralogie
	GLENAT René	Chimie organique
	GROULADE Joseph	Biochimie Médicale
	HACQUES Gérard	Calcul Numérique
	HOLLARD Daniel	Hématologie
	HUGONOT Robert	Hygiène et Médecine Préventive
	IDELMAN Simon	Physiologie Animale
	JOLY Jean-René	Mathématiques Pures
	JULLIEN Pierre	Mathématiques Appliquées
Mme	KAHANE Josette	Physique
MM.	KUHN Gérard	Physique (I.U.T. A)
	LE ROY Philippe	Mécanique (I.U.T. A)
	LUU DUC Cuong	Chimie Organique
	MAYNARD Roger	Physique du Solide
	PELMONT Jean	Biochimie
	PERRIAUX Jean-Jacques	Géologie et Minéralogie
	PSISTER Jean-Claude	Physique du Solide
Melle	PIERY Yvette	Physiologie Animale
MM.	RAYNAUD Hervé	M.I.A.G.
	REBECQ Jacques	Biologie (CUS)
	REVOL Michel	Urologie
	REYMOND Jean-Charles	Chirurgie Générale
	RICHARD Lucien	Biologie Végétale
Mme	RINAUDO Marguerite	Chimie Macromoléculaire
MM.	ROBERT André	Chimie Papetière

MM.	SARRAZIN Roger	Anatomie et Chirurgie
	SARROT-REYNAUD Jean	Géologie
	SIROT Louis	Chirurgie Générale
Mme	SOUTIF Jeanne	Physique Générale
MM.	STREGLITZ Paul	Anesthésiologie
	VIALON Pierre	Géologie
	VAN CUTSEM Bernard	Mathématiques Appliquées

MAITRES DE CONFERENCES ET MAITRES DE CONFERENCES AGREGES

MM.	AMBLARD Pierre	Dermatologie
	ARMAND Gilbert	Géographie
	ARMAND Yves	Chimie (I.U.T. A)
	BACHELOT Yvan	Endocrinologie
	BARGE Michel	Neuro chirurgie
	BARJOLLE Michel	MIAG
	BEGUIN Claude	Chimie organique
Mme	BERIEL Hélène	Pharmacodynamie
MM.	BOST Michel	Pédiatrie
	BOUCHARLAT Jacques	Psychiatrie adultes
Mme	BOUCHE Liane	Mathématiques (C.U.S.)
MM.	BRODEAU François	Mathématiques (I.U.T. B)
	BUTEL Jean	Orthopédie
	CHAMBAZ Edmond	Biochimie médicale
	CHAMPETIER Jean	Anatomie et Organogénèse
	CHARDON Michel	Géographie
	CHERADAME Hervé	Chimie Papetière
	CHIAVERINA Jean	Biologie Appliquée (EFP)
	COHEN-ADDAD Jean-Pierre	Spectrométrie Physique
	COLOMB Maurice	Biochimie Médicale
	CONTAMIN Charles	Chirurgie thoracique et cardio-vasculaire
	CORDONNIER Daniel	Néphrologie
	COULOMB Max	Radiologie
	CROUZET Guy	Radiologie
	CYROT Michel	Physique du solide
	DELOBEL Claude	M.I.A.G.
	DENIS Bernard	Cardiologie
	DOUCE Roland	Physiologie Végétale
	DUSSAUD René	Mathématiques (C.U.S.)
Mme	ETERRADOSSI Jacqueline	Physiologie
MM.	FAURE Jacques	Médecine Légale
	FAURE Gilbert	Urologie
	FONTAINE Jean-Marc	Mathématiques Pures
	GAUTIER Robert	Chirurgie Générale
	GENSAC Pierre	Botanique
	GIDON Maurice	Géologie
	GROS Yves	Physique (I.U.T. A)
	GUITTON Jacques	Chimie
	HICTER Pierre	Chimie
	IVANES Marcel	Electricité
	JALBERT Pierre	Histologie
	KOLODIE Lucien	Hématologie
	KRAKOWIAK Sacha	Mathématiques Appliquées
	LE NOC Pierre	Bactériologie-virologie
	LEROY Philippe	I.U.T. A
	MACHE Régis	Physiologie Végétale
	MAGNIN Robert	Hygiène et Médecine Préventive
	MALLION Jean-Michel	Médecine du Travail
	MARECHAL Jean	Mécanique (I.U.T. A)
	MARTIN-BOUYER Michel	Chimie (C.U.S.)

M.	MICHOULIER Jean	Physique (I.U.T. A)	
Mme	MINIER Colette	Physique (I.U.T. A)	
MM.	NEGRE Robert	Mécanique (I.U.T. A)	
	NEMOZ Alain	Thermodynamique	
	NOUGARET Marcel	Automatique (I.U.T.A)	
	PARAMELLE Bernard	Pneumologie	
	PECCOUD François	Analyse (I.U.T. B)	
	PEFFEN René	Métallurgie (I.U.T. A)	
	PERRET Jean	Neurologie	
	PERRIER Guy	Géophysique - Glaciologie	
	PHELIP Xavier	Rhumatologie	
	RACHAIL Michel	Médecine Interne	
	RACINET Claude	Gynécologie et Obstétrique	
	RAMBAUD André	Hygiène et Hydrologie	
	RAMBAUD Pierre	Pédiatrie	
	Mme	RENAUDET Jacqueline	Bactériologie
	MM.	ROBERT Jean-Bernard	Chimie-Physique
ROMIER Guy		Mathématiques (I.U.T. B)	
SHOM Jean-Claude		Chimie générale	
STOEBNER Pierre		Anatomie pathologique	
	VROUSOS Constantin	Radiologie	

MAITRE DE CONFERENCES ASSOCIES

M.	COLE Antony	Sciences Nucléaires
----	-------------	---------------------

CHARGE DE FONCTIONS DE MAITRE DE CONFERENCES

M.	JUNIEN-LAVILLAVROY Paul	O.R.L.
----	-------------------------	--------

Fait à SAINT MARTIN D'HERES,  
DECEMBRE 1975.

Président : M. NEEL Louis  
 Vice-Présidents : M. BENOIT Jean  
 M. BONNETAIN Lucien

PROFESSEURS TITULAIRES

MM.	BENOIT Jean	Radioélectricité
	BESSON Jean	Electrochimie
	BLOCH Daniel	Physique du solide
	BONNETAIN Lucien	Chimie Minérale
	BONNIER Etienne	Electrochimie et Electrometallurgie
	BRISSONNEAU Pierre	Physique du solide
	BUYLE-BODIN Maurice	Electronique
	COUMES André	Radioélectricité
	FELICI Noël	Electrostatique
	LESPINARD Georges	Mécanique
	MOREAU René	Mécanique
	PARIAUD Jean-Charles	Chimie-Physique
	PAUTHENET René	Physique du solide
	PERRET René	Servomécanismes
	POLOUJADOFF Michel	Electrotechnique
	SILBERT Robert	Mécanique des Fluides

PROFESSEURS ASSOCIES

MM.	RUPPERSBERG Albert, Henner	Chimie
	ROUXEL Roland	Automatique

PROFESSEURS SANS CHAIRE

MM.	BLIMAN Samuel	Electronique
	BOUVARD Maurice	Génie Mécanique
	COHEN Joseph	Electrotechnique
	DURAND Francis	Métallurgie
	FOULARD Claude	Automatique
	LACOUME Jean-Louis	Géophysique
	LANCIA Roland	Electronique
	VEILLON Gérard	Informatique Fondamentale & Appliquée
	ZADWORNY François	Electronique

MAITRES DE CONFERENCES

MM.	ANCEAU François	Mathématiques Appliquées
	BOUDOURIS Georges	Radioélectricité
	CHARTIER Germain	Electronique
	GUYOT Pierre	Chimie Minérale
	IVANES Marcel	Electrotechnique
	JOUBERT Jean-Claude	Physique du solide
	MORET Roger	Electrotechnique Nucléaire
	PIERRARD Jean-Marie	Hydraulique
	ROBERT François	Analyse Numérique
	SABONNADIÈRE Jean-Claude	Informatique Fondamentale & Appliquée
Mme	SAUCIER Gabrièle	Informatique Fondamentale & Appliquée

MAITRE DE CONFERENCES ASSOCIE

M.       LANDAU Ioan                   Automatique

CHERCHEURS DU C.N.R.S. (Directeurs et Maîtres de Recherche)

MM.       FRUCHART Robert                Directeur de Recherche  
          ANSARA Ibrahim                   Maître de Recherche  
          CARRE René                     Maître de Recherche  
          DRIOLE Jean                    Maître de Recherche  
          MATHIEU Jean-Claude            Maître de Recherche  
          MUNIER Jacques                Maître de Recherche



*Qu'il me soit donné de remercier ici*

*Monsieur le Professeur VAUQUOIS, pour l'accueil qu'il a fait à cette thèse en acceptant d'en présider le jury.*

*Messieurs Claude DELOBEL, Maître de Conférences à Grenoble I et Jean-Claude DERNIAME, Maître de Conférences à Nancy I, pour l'intérêt qu'ils ont porté à mon travail, les idées qu'ils m'ont suggérées et surtout pour la confiance qu'ils m'ont ainsi témoignée.*

*A François PECCOUD, Maître de Conférences à Grenoble II, qui m'a proposé l'idée de ce travail et les moyens matériels de le réaliser, je dirai combien j'ai apprécié la constance de son engagement et les critiques de l'utilisateur intransigeant ; qu'il soit très affectueusement remercié d'avoir couru le risque de faire travailler une femme, et de plus sa soeur.*

*Je ne saurai remercier Jean-Claude BOUSSARD, Professeur à Nice, sans l'associer à sa femme Anne. Leur amitié et leur hospitalité ont ajouté aux critiques constructives une force rafraichissante sans laquelle je ne serais peut-être pas parvenue au terme de ce travail.*

*Que mes collègues de l'équipe "Algol 68", ainsi qu'Y. CHIARAMELLA, J. COURTIN et A. STIERS, acceptent ces mêmes remerciements pour les informations partagées, et à travers eux tous ceux qui ont un jour témoigné leur intérêt pour ce travail.*

*Arlette NADEAU et Philippe GRANDJEAN, en assurant la programmation de ce travail, lui ont donné forme et consistance. Leur participation a été déterminante pour la réalisation d'Abrège. Qu'ils en soient très sincèrement remerciés.*



*Je voudrais aussi exprimer à toute l'équipe "Socrate" d'Eca-Automation ma reconnaissance pour leur assistance technique et la disponibilité amicale rencontrée auprès d'eux.*

*Marie-José DOREL a assuré la frappe de cette thèse avec une compétence souriante dont je lui sais gré chaque jour.*

*Surtout, la concentration nécessaire à l'achèvement d'une thèse suppose que d'autres vous déchargent de certains travaux quotidiens. A mes collègues de l'I.U.T. Informatique, je voudrais donc dire ici ma reconnaissance. Et je voudrais exprimer tout particulièrement mon affection à Yves, mon mari, et à travers lui à tous ceux qui ont partagé nos soucis de tous les jours.*

O mes amis, vous tous, je ne renie  
aucun de vous ; ni même ce passant  
qui n'était de l'inconcevable vie  
qu'un doux regard ouvert et hésitant.

Combien de fois un être, malgré lui,  
arrête de son oeil ou de son geste  
l'imperceptible fuite d'autrui,  
en lui rendant un instant manifeste.

Les inconnus. Ils ont leur large part  
à notre sort que chaque jour complète.  
Précise bien, ô inconnue discrète,  
mon coeur distrait, en levant ton regard.

R.M. RILKE

à l'amitié.



## SOMMAIRE

### CHAPITRE 1 - INTRODUCTION ET ANALYSE DES OBJECTIFS

1.1. Analyse du problème initial -----	I.1
1.2. Production automatisée d'algorithmes d'abréviation. Justifications -----	I.3
1.3. Les conditions spécifiques d'Abrège par rapport aux méthodes de codage "standards" et de compactage de texte -----	I.8
1.4. Les objectifs de réalisation spécifiques d'Abrège --	I.10
1.4.1. Prise en compte des mécanismes spontanés d'abréviation -----	I.10
1.4.2. Première ébauche de structuration du problème	I.11
1.5. Choix de production du "logiciel" Abrège -----	I.14
1.5.1. Réalisation d'un prototype d'Abrège -----	I.14
1.5.2. Choix du logiciel d'écriture -----	I.14
1.6. Schéma fonctionnel d'Abrège -----	I.17
1.6.1. La fonction de production d'Abrège -----	I.17
1.6.2. La fonction de saisie -----	I.19

### CHAPITRE 2 - LES ABREVIATIONS PAR LES CARACTERES

2.1. Les fonctions à réaliser -----	II.1
2.1.1. Les textes -----	II.2
2.1.2. Les mots -----	II.3
2.2. La structuration des propriétés -----	II.4
2.2.1. Génération de la base -----	II.7
2.2.2. Les dictionnaires -----	II.8
2.3. Réalisation des fonctions de base -----	II.10
2.4. Les méthodes d'abréviation ou la composition des fonctions -----	II.11

2.4.1. Propriétés générales du résultat -----	II.11
2.4.2. Propriétés des libellés -----	II.12
2.4.3. Propriétés des mots -----	II.12
2.4.4. Traitements du mot : dictionnaire ou calcul -----	II.14
2.4.5. Propriétés des caractères ou lettres prises en compte dans un mode de calcul -----	II.16
2.4.6. Les actions de sélection -----	II.17
2.4.7. Un premier schéma de la grammaire descriptive d'une méthode -----	II.21
2.5. Expression des méthodes dans la structure précédente. Traduction d'abréviations manuelles -----	II.21
2.5.1. G3, Méthode d'abréviation devant donner des noms différenciés de cinq caractères au plus -----	II.22
2.5.2. Méthode 2 : "DM", Dépenses de matériel de l'Etat -	II.27
2.6. Résultats de deux méthodes imitant le processus manuel. Discussion -----	II.30
2.6.1. Adéquation dans l'imitation -----	II.30
2.6.2. Evaluation par rapport aux objectifs. Autres méthodes -----	II.34
Conclusion -----	II.43

### CHAPITRE 3 - LA DEMARCHE PHONETIQUE ET SYLLABIQUE

3.1. Nécessité d'une analyse phonétique et syllabique -----	III.1
3.2. Recherche et choix d'un outil pour la traduction grapho- phonémique -----	III.2
3.3. Implantation de l'analyse phonétique -----	III.5
3.3.1. L'automate. Son implantation -----	III.5
3.3.2. Stockage des résultats -----	III.10
3.4. Analyse syllabique. Automate. Implantation -----	III.11
3.5. Extensions des méthodes d'abréviations -----	III.14
3.5.1. Extension du filtre-mot -----	III.14
3.5.2. Extensions des contextes de calcul -----	III.15
3.5.3. Extensions des actions et des traitements -----	III.17
3.5.4. Schéma étendu de la grammaire descriptive d'une méthode -----	III.20

3.6. Nouvelle expression de méthode utilisant les extensions ---	III.21
3.7. Une méthode d'abréviation de texte produisant un texte et non un identificateur -----	III.25
3.8. Considérations sur les résultats -----	III.28
3.8.1. Les résultats de la méthode 4 -----	III.28
3.8.2. Les résultats d'une méthode "mot à mot". La méthode 5 -----	III.28
3.8.3. L'abréviation mot à mot par Abrège -----	III.29

#### CHAPITRE 4 - LES CARACTERISTIQUES MORPHOLOGIQUES DES ABREVIATIONS

4.1. Pourquoi une analyse morphologique ? -----	IV.1
4.2. Le système PIAF. Programme interactif d'analyse du français -----	IV.3
4.3. Problèmes liés à l'éventuelle utilisation de PIAF dans le système Abrège -----	IV.5
4.3.1. Difficultés contingentes de réalisation -----	IV.5
4.3.2. Réflexions sur l'opportunité d'une analyse mor- logique -----	IV.6
4.4. Conclusion -----	IV.7

#### CHAPITRE 5 - EVALUATION ET VALIDATION DE METHODES

5.1. Les validations irréalisables -----	V.1
5.2. Ce qu'il est réaliste d'envisager. Les objectifs -----	V.2
5.3. Le code obtenu. Mesures -----	V.3
5.4. L'exécution d'une méthode. Utilisation des propriétés, des règles : bilan d'exploitation -----	V.5
5.5. Implantation de l'évaluation -----	V.7

## CHAPITRE 6 - SPECIFICATIONS EXTERNES DU SYSTEME ABREGE

6.1. Lancement des différentes phases -----	VI.2
6.1.1. La grammaire -----	VI.2
6.1.2. Interprétation d'une tâche -----	VI.3
6.1.3. Erreurs de syntaxe dans l'enchaînement des phases	VI.3
6.2. Langage de description de méthode -----	VI.5
6.2.1. La grammaire de description des méthodes -----	VI.5
6.2.2. Sémantique d'une description -----	VI.6
6.2.3. Interface d'une demande d'évaluation -----	VI.34
6.3. Expression en Abrège des méthodes citées en exemple -----	VI.36
6.4. La saisie des dictionnaires -----	VI.41
6.5. La saisie des données -----	VI.46
6.5.1. La grammaire pour la saisie des libellés -----	VI.46
6.5.2. Sémantique d'une description de saisie -----	VI.47
6.5.3. Informations complémentaires sur la saisie des données -----	VI.49
6.6. La demande d'exécution d'une méthode -----	VI.52

## CHAPITRE 7 - AMELIORATION ET EVOLUTION DU SYSTEME ABREGE.

### CONCLUSION

7.1. Réalisation conversationnelle d'Abrège -----	VII.1
7.1.1. Saisie conversationnelle. Points-clefs. Limites --	VII.1
7.1.2. Exécution ponctuelle. Interrogation de la base ---	VII.2
7.2. Extension d'Abrège à d'autres classes de méthodes -----	VII.3
7.2.1. Prise en considération de "groupes de mots" -----	VII.3
7.2.2. Reconnaître la syntaxe de "qualifications en cascade" -----	VII.4
7.2.3. Quelques modifications dans l'écriture des mé- thodes -----	VII.5
7.2.4. Quelques extensions des classes de méthodes -----	VII.6
7.3. Conclusion -----	VII.9

## ANNEXES

### [A1] ENSEMBLE DES LIBELLES. PROJET G3

- 1.1. Données fournies par l'utilisateur
- 1.2. Résultat par abréviation suivant la Méthode 1, sans dictionnaire
- 1.3. Dictionnaire SPECIAL. Nouveaux résultats : Résultat2 - Résultat3.

### [A2] ENSEMBLE DES LIBELLES. PROJET DM

- 2.1. Données proposées par l'utilisateur
- 2.2. Résultat de la Méthode 2
- 2.3. Résultats de la Méthode 3.

### [A3] ENSEMBLE DES LIBELLES. PROJET (GESTION DES CONGRES)

- 3.1. Données proposées par l'utilisateur
- 3.2. Résultats de la Méthode 3 / Dictionnaire
- 3.3. Résultats de la Méthode 4 / Dictionnaires
- 3.4. Méthode 5. Quelques résultats.

### [A4] DICTIONNAIRES STANDARD D'ABREGE

- . Antidic
- . Standic.

### [A5] L'ANALYSEUR PHONETIQUE.

### [A6] NORME D'ABREVIATION DES TITRES DE PERIODIQUES.

### [A7] LA STRUCTURE DE LA BASE D'ABREGE.

### [A8] AUTOMATES DE RECONNAISSANCE DU LANGAGE DE DESCRIPTION DE METHODES.

Note au lecteur :

*Les exemples présentés dans cette thèse sont écrits avec cette typographie.*





## CHAPITRE 1

INTRODUCTION ET ANALYSE DES OBJECTIFS

### 1.1. ANALYSE DU PROBLEME INITIAL

L'écriture de programmes, comme l'enseignement de la programmation et de l'analyse, nous a depuis plusieurs années montré la difficulté que l'on rencontre lorsque l'on cherche à définir des normes permettant de créer des codes d'identification des données, des programmes ou des fichiers qui soient respectés par l'ensemble des participants à une même application. Tantôt les analystes ou programmeurs choisissent des identificateurs sans aucune liaison avec le libellé qui définit la signification de l'identificateur, tantôt ils essaient de produire une chaîne de caractères qui plus ou moins, mnémotechniquement parlant, rappelle la signification de cette donnée, de ce programme ou de ce fichier. Dans d'autres domaines techniques, il existe le même souci qu'en informatique de définir des nomenclatures pour les objets manipulés : par exemple, nomenclatures de produits dans un catalogue, dans un stock.

Deux positions se développent, ici aussi, dans la création de nomenclatures, soit la création d'identifiants n'ayant aucune signification par eux-mêmes et qu'il faut rapprocher du libellé ou de leur définition pour connaître l'objet qu'ils repèrent, ou d'identifiants partiellement codés dont il faut connaître le mode de codification pour pouvoir analyser les informations qu'ils portent (numéro de Sécurité Sociale), soit un souci de définir un code mnémonique direct qui essaie d'attirer l'attention de ceux qui l'utiliseront sur la dénotation de l'objet qu'il représente.

*Ainsi, en programmation, sans parler des identificateurs du type JOJO et TOTO qu'il faut bannir, on peut appeler les programmes de P1 à P100 les données de D1 à D1000, les fichiers de F1 à F100, les étiquettes ..., ce qui oblige à avoir accès à un dictionnaire de correspondance avec un libellé de définition pour connaître la signification de P37, F56, D612, etc .*

*On peut parler, par contre, de PAYE pour un programme de paye et NOSS pour un numéro de sécurité sociale, COPRO pour code produit.*

*Nous nous situons par rapport au problème qui consiste à concevoir un outil permettant de définir et d'exécuter des méthodes d'abréviations ayant toutes pour objectif de condenser un libellé en produisant dans la mesure du possible un identificateur, une abréviation qui rappelle le contenu du libellé.*

Plus généralement, nous pouvons étendre ce problème en voulant abréger des textes en conservant autant que faire se peut leur sémantique, mais sans forcément faire correspondre à un texte un seul identifiant. Il existe en effet plusieurs circonstances où, partant d'un texte, il faut produire un groupe de plusieurs chaînes de caractères condensant le texte initial mais produisant ainsi un "texte" immédiatement lisible, sans décodage autre que celui que peut faire facilement un esprit prévenu.

*C'est le cas par exemple des catalogues de ventes par correspondance où certains descriptifs sont abrégés, le cas de certains fichiers bibliographiques où les titres des périodiques de diverses nationalités sont représentés par des abréviations standardisées et uniques, le cas enfin de petites annonces matrimoniales ou de demande et d'offre d'emploi où il s'agit d'exprimer sous la forme la plus condensée possible un message sans qu'aucune ambiguïté ne se produise.*

Nous nous proposons donc de fournir les moyens de décrire et d'interpréter automatiquement des algorithmes d'abréviation répondant soit au premier besoin de création automatique d'identificateurs mnémoniques, soit au deuxième besoin d'abréviation de textes pour qu'ils conservent leur signification.

## 1.2. PRODUCTION AUTOMATISEE D'ALGORITHMES D'ABREVIATION. JUSTIFICATIONS

Nous devons justifier l'automatisation de cette production d'algorithmes d'abréviation conservant une partie de la sémantique des textes.

Deux questions se posent en effet. Pourquoi rechercher des abréviations mnémoniques et pourquoi rechercher l'automatisation à la fois de la description des algorithmes et de leur exécution ?

Nous ne voulons pas répondre à la première question en prenant part au débat sur l'intérêt de codes d'identification signifiants ou non signifiants. Nous voulons simplement constater que le choix de produire des identifications non signifiantes est souvent provoqué par l'absence de procédures normalisées pouvant donner lieu à des abréviations signifiantes. Quelques exemples illustrent ce constat :

*Certes, il est très difficile d'envisager de réduire les noms et prénoms de tous les français en une codification qui les distinguent les uns des autres et le système de codification de l'INSEE est très satisfaisant à cet égard, pour autant qu'il puisse être mis en place, ce qui suppose déjà une structure administrative considérable. Mais il est des situations socio-politiques dans certains états où la production d'un numéro de Sécurité Sociale n'est pas possible, alors que tout individu même dans ces pays a un nom et plusieurs prénoms. Ne pourrait-on pas par exemple envisager, si on savait le faire, de remplacer l'identifiant numéro de Sécurité Sociale, ou plutôt numéro INSEE propre aux français, par une abréviation discriminante permettant de distinguer chaque citoyen de tous les autres. Il suffirait de disposer d'un algorithme permettant, à partir des noms et prénoms comme texte de base, de créer un code discriminant. C'est un problème qui se pose actuellement dans certains pays du tiers-monde.*

*Citons, par ailleurs, l'attitude naturelle des analystes et programmeurs d'un projet informatique. Ils cherchent en général à créer des identificateurs signifiants. Mais à partir du moment où plusieurs personnes participent au même projet, on s'aperçoit, dans un délai très bref, que le même code identifie, suivant son utilisation, des objets très différents, car si les participants ont produit des abréviations mnémoniques, ils ne les ont pas produites avec assez de méthode pour qu'elles soient discriminantes entre elles.*

*Enfin, il est hors de question, lors de la rédaction de petites annonces, de coder les différents mots du vocabulaire avec des codes non signifiants et standards, ne serait-ce que parce que le résumé que représente une petite annonce doit être non seulement produit mais lu et compris, sans décodage supplémentaire et sans ambiguïté, par ceux auxquels elle est destinée.*

Il nous paraît donc nécessaire de pouvoir produire des abréviations qui conservent une partie de la sémantique du texte d'origine, ce qui fait apparaître deux difficultés.

La première est l'impossibilité dans laquelle on se trouve d'assurer, pour une entité donnée, la production d'un identifiant unique.

*L'absence d'ambiguïté des codes est une caractéristique fondamentale d'un certain nombre de situations.*

*Sans parler des codes produits pour décrire les articles d'un stock, parlons plutôt des applications informatiques où nous trouvons couramment plusieurs noms pour définir le même objet, et encore plus le même code pour désigner des objets différents. Dans les petites annonces aussi, nous trouvons couramment plusieurs abréviations pour le même mot, "cher" et "cherche" indifféremment employés, ou "mouv" et "mvt" pour "mouvement", "jne f." et "j.f" pour "jeune femme" ou "jeune fille".*

Pourtant, si une abréviation est satisfaisante, une de ses caractéristiques essentielles doit être la génération, par abréviation, du minimum d'ambiguïté sémantique.

C'est là notre deuxième difficulté : nous constatons, en effet, au niveau de cette ambiguïté sémantique, l'ignorance presque totale où nous nous trouvons des mécanismes de conservation de la sémantique dans les processus d'abréviation. Pourquoi adopter tantôt "mouv", tantôt "mvt" ? Quelle est de ces deux abréviations celle qui, au niveau d'un individu, garde le maximum de signification ? Quelle est celle qui est la plus signifiante pour un groupe social homogène ?

Nous avons tenté un inventaire des recherches dans ce domaine, mais nous nous sommes retrouvés sans beaucoup de résultats à l'issue de l'analyse bibliographique. Nous avons orienté celle-ci dans deux directions. La psychologie cognitive et les modèles d'apprentissage ainsi que les modèles automatisés d'apprentissage sur ordinateur [P2] [H1] [R1] ne nous ont rien apporté si ce n'est la connaissance des difficultés que l'on rencontre lorsque l'on veut mesurer ou simuler les éléments qui entrent en jeu dans la mémorisation.

Les techniques de documentation et de recherche d'informations codées, par ailleurs, ainsi que les techniques de compression de code [N1][SK1][K2] [LG1][V2] nous ont montré certains éléments à prendre en compte pour une codification de textes. Cependant, un titre aussi prometteur que "Codification sémantique des noms de minéraux" [K1] conduit à des résultats où la sémantique est analysée, puis codée de manière non lisible directement.

Il nous est donc apparu nécessaire, avant même de vouloir fournir des produits opérationnels pour abrégé des textes en gardant une partie de leur sémantique, de donner les moyens d'étudier différentes possibilités d'abréviations à partir des mêmes textes, de façon à examiner ensuite celles qui semblent avoir le plus d'intérêt au niveau de leur utilisation.

Enfin, les tendances actuelles en informatique incitent les responsables à fournir des nomenclatures significatives et discriminantes pour les données, les programmes, les fichiers. Nous citerons simplement pour témoin la nouvelle fonction de "gestionnaire des données" dans un centre mettant en place des systèmes de gestion de bases de données. Lorsqu'on se penche sur le problème de la réalisation de ces nomenclatures, on s'aperçoit que la principale difficulté n'est pas de produire des abréviations, mais bien de définir un libellé qui dénote sans ambiguïté l'objet, la personne, la chose ou le lieu concernés. Des travaux très importants abordent ce problème, que ce soit pour définir des nomenclatures "produits de Bruxelles" pour identifier le commerce extérieur d'un état, ou pour fabriquer des nomenclatures de produits en stock ; l'objet de ces travaux est bien plus un effort de définition qu'un effort d'abréviation.

*Nous avons nous-même constaté que, lorsque nous demandions aux étudiants de produire un libellé de définition associé aux identificateurs des variables utilisées dans un projet informatique, il existait non un trop-plein, mais un vide de définition.*

Il nous est donc apparu dans bon nombre de situations qu'il était méthodologiquement très intéressant de pouvoir proposer la possibilité de définir la chose, la personne, le lieu dont ils parlent, par une définition aussi complète que possible en clair, plutôt que de leur laisser la liberté de fournir une abréviation qui masque l'absence totale de définition.

*Ainsi sommes-nous convaincus que, lorsque nous demandons à des étudiants de faire un dictionnaire des données de leur projet informatique, leur effort doit porter bien plus sur la définition de celles-ci. Supprimer à leur niveau la responsabilité de définir des codes, mais les obliger à fournir une définition précise de ces données c'est les alléger d'un travail secondaire pour les laisser se consacrer à l'essentiel.*

Aussi abrégé des textes en gardant leur sémantique, c'est d'abord reconnaître qu'il est nécessaire d'avoir un texte et une sémantique avant le mécanisme d'abréviation.

Ainsi à partir du moment où nous suggérons, pour étudier les mécanismes de mémorisation et de conservation de la sémantique, de pouvoir définir, créer, utiliser plusieurs méthodes, et où nous proposons de consentir un effort pour définir les objets avant de leur donner un code, il paraît tout naturel de créer des outils automatiques permettant de décrire facilement, puis d'exécuter des mécanismes d'abréviations. D'autant plus qu'on ne sait pas encore les éléments qui sont à prendre en compte lors de la mise en oeuvre de ces mécanismes et qu'une partie de la recherche doit être orientée vers cette définition.

Une dernière justification à la production automatisée d'abréviations conservant la sémantique de textes est liée aux conditions même d'emploi de ces abréviations. Dans bien des cas, des personnes différentes contribuent séparément à définir les parties d'un ensemble d'abréviations que chaque utilisateur doit pouvoir ensuite comprendre dans sa totalité.



*Dans un projet informatique, plusieurs personnes peuvent créer des abréviations par sous-ensembles, mais le dossier doit être lisible dans son entier par le chef de projet.*

*Dans les petites annonces, si on laisse à chacun la liberté de concevoir ses abréviations, alors que la rubrique est destinée dans son ensemble à être lue par une même personne, on risque de créer entre les annonces une hétérogénéité de présentation qui les rende incompréhensibles.*

Or, ce qui est commun à l'ensemble des émetteurs, ce ne sont pas les mécanismes d'abréviations, mais les informations qu'ils désirent transmettre (un sexe, un âge, une situation familiale, des goûts et les conditions d'une "vie commune" pour une annonce matrimoniale par exemple) et la langue naturelle qu'ils parlent. Prenons un journal de petites annonces. Ce que les uns ont abrégé ne l'a pas été chez d'autres ; pourtant les mots ont la même place, le même rôle, et les textes sont très proches.

### 1.3. LES CONDITIONS SPECIFIQUES D'ABREGE PAR RAPPORT AUX METHODES DE CODAGE "STANDARDS" ET DE COMPACTAGE DE TEXTE

Nous avons déjà signalé que le problème posé par Abrège nous avait conduit à étudier les techniques de codage employées en documentation. En particulier, nous avons pensé que les problèmes posés par la recherche de document en fonction de leur contenu avaient dû produire une réflexion spécialement orientée vers la définition de la sémantique des textes et éventuellement le codage de celle-ci. Nous avons constaté que cette sémantique est plus ou moins conservée soit par présence de "mots-clés" dans les textes ou par analyse de fréquence "anormalement" élevée de certains termes dans le texte par rapport à la fréquence "normale" de ces mêmes termes dans le langage, soit enfin par donnée "a priori" des mots-clés. Cette sémantique cependant n'est pas traduite sans codage dans les "descripteurs" permettant la recherche. Nous nous sommes servis, pour établir ce bilan, du livre de Vickery "Techniques modernes de documentation" [V3] et nous avons ensuite contrôlé ces conclusions dans les procédures de l'ASIS [AS1] (American Society for Information Systems). La préoccupation sémantique de l'indexation de texte pour la recherche documentaire est actuellement jumelée à des techniques automatiques de recherche qui permettent un décryptage et conduit à la production de codes très condensés mais "illisibles".

Cela nous a cependant montré que la fréquence des termes semble être un élément à prendre en compte dans la conservation de la sémantique.

Nous avons parallèlement recherché ce que nous pouvions apprendre des recherches en matière de compactage de chaînes de caractères. Mais là aussi la préoccupation principale est la non-ambiguïté du code produit, vérifiée éventuellement par des mesures faisant appel à la théorie de l'information [SK1][LG1], ainsi que la recherche d'un compactage maximum pour une conservation totale de l'information restituée encore une fois par décryptage automatique [N1] [MS1].

Ce bilan négatif, en quelque sorte, a mis en évidence cinq attitudes qui distinguent nos préoccupations de celles que nous avons étudiées :

- Nous voulons conserver une sémantique sur des textes courts auxquels nous ne voulons pas faire subir de traitement préalable à l'abréviation automatique (pas d' "indexation" comme en documentation automatique).

- Nous voulons produire des codes "discriminants ... dans une certaine mesure", nous estimant éventuellement satisfaits de procédures artificielles (comme la numérotation) pour distinguer des abréviations identiques de textes initiaux différents.

- Nous ne cherchons pas tant à "compacter des textes" en conservant toute l'information qu'ils contiennent qu'à conserver autant que possible une bonne partie de la signification, mais ceci étant éventuellement annexe (par exemple, lorsque le code produit doit être très court)

- Nous ne sommes pas tenus de restituer le texte initial.

- Nous devons, par contre, dans presque tous les cas, conserver une signification telle que la simple connaissance de procédés standards d'abréviation permette une lecture immédiate du code produit.

Cette spécificité de notre problème et le manque de connaissance sur les moyens de le résoudre nous ont conduit à la démarche suivante : tenter d'analyser les mécanismes fondamentaux d'abréviations tels qu'on peut les voir fonctionner "in situ" (analyser les journaux de petites annonces, les codes produits dans les applications informatiques) pour dégager les caractéristiques de ces mécanismes ; en fonction de celles-ci, tenter de définir un outil paramétré permettant des mises en oeuvre combinatoires de ces mécanismes de manière à évaluer quels sont les mécanismes les plus "fructueux", les plus "signifiants" suivant les conditions de leur emploi.

#### 1.4. LES OBJECTIFS DE REALISATION SPECIFIQUES D'ABREGE

##### 1.4.1. Prise en compte des mécanismes spontanés d'abréviation

Ainsi que nous l'avons dit ci-dessus, nous voulons permettre de décrire des méthodes d'abréviation qui, soit à partir d'un texte assez court, donnent une chaîne de caractères unique (identificateurs), soit à partir d'un texte un peu plus long, donnent une abréviation mot à mot lisible de ce texte (petites annonces, catalogues).

Dans ces deux cas, il faut que les méthodes permettent l'utilisation de caractéristiques des textes ou des mots qui correspondent aux démarches "naturelles" d'abréviation lorsque celles-ci cherchent à conserver une signification.

Ainsi avons-nous analysé quelques fonctions de base dont il faudra assurer la mise en oeuvre.

D'après l'analyse des techniques documentaires, il semble que la signification réside plus particulièrement dans certains mots connus soit par leur fréquence d'apparition élevée (utilisée pour l'indexation automatique), soit par une insistance spéciale des utilisateurs à leur égard (indexation manuelle avec mise en évidence de mots-clés). La traduction fonctionnelle de cette attitude sera pour nous d'une part la mesure et la prise en compte dans les méthodes de la fréquence des mots, d'autre part la mise en évidence des mots "remarquables" de l'utilisateur, par utilisation de dictionnaires dans lesquels celui-ci peut éventuellement proposer une abréviation standard, qui sera prise en compte prioritairement. Par ailleurs pour tenir compte de mots fréquents, il faut supprimer les mots toujours très fréquents que sont les articles, les conjonctions, les prépositions. Cela rejoint la constatation que l'on peut faire en lisant des petites annonces où ces particules ont généralement été omises. Nous réaliserons cette fonction par enregistrement de ces mots dans des "antidictionnaires", dictionnaires de mots non significatifs dont on ne veut pas.

D'autre part il semble que les processus d'abréviation s'appuient, d'une façon générale, sur des caractéristiques acoustiques pour respecter une certaine syllabisation :

"urg." pour "urgent", "cher." pour "cherche", "entr." pour "entreprise", etc.

Cette approche de nature acoustique, qui cherche à conserver la syllabisation, rejoint les réflexions de l'étude d'IBM France pour la génération de noms de marques à partir d'une décomposition syllabique des textes [V1].

Par ailleurs, seuls les mots les plus longs seront abrégés : *secrétaire*, *restaurant*, *entreprise*, à moins que ce ne soit des mots dont la présence est connue sans ambiguïté comme "cc" pour "code-carte" en informatique, "j.h" et "j.f" pour "jeune homme" et "jeune femme" dans des annonces matrimoniales, "mec." pour "mécanique" dans les annonces automobiles.

Enfin la place du mot joue un rôle dans les choix d'abréviation faits lors de la création d'identificateurs. Le premier mot est soit considérablement réduit, comme "c" pour code quand le mot est fréquent, soit au contraire beaucoup moins abrégé que les autres lorsqu'il est porteur d'une part importante de la signification du texte, les mots suivants servant alors souvent à le qualifier comme dans "enregistrement de l'arrivée du bon d'engagement".

Nous voyons donc s'ébaucher le schéma de ce que sera une méthode d'abréviation, mise en oeuvre combinatoire de diverses propriétés pour prendre en compte telle ou telle part de la signification du texte en portant l'attention des lecteurs sur des mots ou sur certains sons des mots, en fonction de leur fréquence, de leur longueur, de leur position.

#### 1.4.2. Première ébauche de structuration du problème

Nous venons de déterminer certains éléments à mettre en oeuvre à l'exécution de méthodes d'abréviation. Cela montre qu'Abrège doit être un outil permettant de mesurer la fréquence des mots dans les textes de définition donnés, de mesurer la longueur, de connaître la position, de savoir si les mots sont significatifs ou non, de savoir décomposer les mots en syllabes, etc.

En outre, lorsque nous disons qu'une méthode doit être une mise en oeuvre combinatoire de ces divers mécanismes de base, cela suppose l'existence des moyens nécessaires à la description et à l'interprétation de ceux-ci et de leur combinaisons. Non seulement pour permettre de reconnaître les fonctions et les propriétés dont on a fait état ci-dessus, mais pour permettre en fonction de ces propriétés de choisir tel ou tel mode d'abréviation, c'est-à-dire telle ou telle sous-chaîne de caractères dans la chaîne initiale.

En effet, si nous proposons comme exemple de méthode le texte suivant :

*"Pour tous les libellés proposés, garder les mots significatifs et abrégés ceux-ci en prenant pour le premier mot la première syllabe concaténée à la première lettre des syllabes suivantes, puis pour les autres mots les trois premières lettres. Concaténer les abréviations entre elles en les séparant par des points"*

nous voyons aussitôt que d'autres méthodes peuvent nous venir à l'esprit, qui font appel aux mêmes propriétés : mots significatifs, position du mot, position de la syllabe, des caractères, etc ..., auxquelles s'ajoutent des insertions de séparateurs. Aussi lorsque nous en aurons écrit dix, quinze, ou vingt, un utilisateur imaginaire nous reprocherait de ne pas lui en donner une vingt-et-unième, si notre outil est limité à l'écriture non paramétrée de quelques méthodes.

Interpréter, pour les exécuter, des descriptions de méthodes suppose donc l'existence d'un langage de description.

Quelques contraintes s'imposent d'emblée à ce langage.

- Il doit permettre une interprétation déterministe des méthodes, de manière à assurer à l'utilisateur l'interprétation et les résultats qu'il attend, ainsi que l'identité des abréviations pour des textes identiques.

- Il doit être simple d'emploi soit parce qu'il est raisonnable d'envisager son utilisation par des non-informaticiens, soit parce que, ainsi que nous l'avons déjà dit, il faut que pour les informaticiens le travail d'abréviation soit une tâche simple, secondaire par rapport au travail de définition.

Abrège doit donc *proposer un langage de description de méthodes, assurer l'exécution de ces méthodes* sur des textes fournis par l'utilisateur, avec toutes les fonctions intermédiaires que nous avons signalées, *pour assurer la production des abréviations*. Ceci suppose donc, outre les fonctions signalées (syllabisation, mesures de fréquence, etc ...), que le système assure *la saisie des dictionnaires* dont nous avons proposé l'usage, ainsi que *la saisie des données*. Enfin, nous rappelant que nous ignorons toujours ce qu'est une bonne méthode d'abréviation, nous devons ajouter à cela *des fonctions d'évaluation et de trace de l'exécution des diverses méthodes* qui permettent à l'utilisateur d'orienter son choix vers telle ou telle méthode, améliorée s'il le faut pour prendre en compte des critères supplémentaires.

Enfin, nous ajouterons à notre langage la possibilité d'accepter des fonctions spécifiques à certains utilisateurs de manière à *assurer une certaine extensibilité du langage*. Ceci permettra de pallier les lacunes éventuelles de notre analyse concernant les mécanismes fondamentaux qui entrent en jeu dans les processus d'abréviation conservant la signification des textes.

## 1.5. CHOIX DE PRODUCTION DU "LOGICIEL" ABREGE

### 1.5.1. Réalisation d'un prototype d'Abrège

Lorsque nous définissons les objectifs d'Abrège, nous envisageons implicitement non de créer un logiciel "clés en main" permettant de définir, d'exécuter et de comparer des méthodes d'abréviation produisant un texte "significatif", mais bien de proposer un prototype avec le caractère "ouvert" d'une telle démarche.

En effet, nous savons parfaitement que le coût d'un logiciel "clés en main" est bien plus élevé que celui d'une première version d'un logiciel qui peut ensuite être modifié et les moyens limités dont nous disposons suffisent à eux seuls à justifier cette attitude. Cependant, même avec des moyens plus importants, il aurait été vain de vouloir immédiatement produire un logiciel "clés en main". Car ainsi que nous le verrons dans les chapitres qui suivent, nous nous apercevons que le langage proposé ne couvre qu'une classe de méthodes d'abréviation et le lecteur trouvera certainement des exemples de méthodes qui lui sont suggérés par le langage et que celui-ci ne permet pas d'exprimer.

Il ne s'agit donc que de faire une proposition de langage, que de définir un premier outil de recherche de "bonnes méthodes", qui suscite des extensions du langage, des améliorations de l'outil, de nouvelles idées à partir d'une pratique importante.

C'est la raison de la création d'un prototype réalisé autant que possible de manière à permettre assez facilement des extensions du langage, avec les modifications de l'interpréteur qu'elles impliquent.

### 1.5.2. Choix du logiciel d'écriture

C'est à ce niveau de précision de nos objectifs que nous avons rencontré le problème de choisir un logiciel pour écrire ce prototype. Nous avons deux contraintes essentielles : l'obligation de simplicité pour l'utilisateur dans l'interface d'utilisation - langage de description de méthode - et celle de flexibilité du logiciel Abrège, lui-même considéré comme prototype, pour permettre des extensions et des modifications faciles. Nous avons ajouté à cela une volonté de portabilité du produit, même à l'état de prototype, pour permettre l'étude des méthodes d'abréviations.



dans des cadres divers.

Le prototype d'Abrège comporte deux caractéristiques fonctionnelles essentielles : offrir un langage interprétable et être, lors de l'interprétation, un logiciel de transduction d'une chaîne de caractères (texte initial de définition) en une autre chaîne de caractères (texte abrégé) par mise en oeuvre de certaines règles (méthode d'abréviation).

Ceci suggère d'utiliser pour l'interprétation des techniques de "Compilateur de Compilateur" de manière à produire du code "performant" de chaque méthode, et dans ce cadre-là un langage comme CDL (Compiler Description Language) [CD1] permet de produire, à partir d'une description "LL1" de notre langage, un "compilateur" écrit soit en Algol W, soit en PL1. Mais cela nous oblige alors à écrire tous nos mécanismes de "transduction" dans l'un ou l'autre de ces deux langages en gérant nous-même les dictionnaires, les données et leurs fonctions de saisie.

Une autre attitude nous a conduit à rechercher ce qui existe dans notre entourage en matière de mécanismes de "transduction". Dans ce cadre-là, le logiciel écrit pour décrire le système PIAF [C2] poursuit des buts analogues mais plusieurs raisons nous ont conduit à renoncer à son emploi. Ce logiciel était encore un prototype en cours d'écriture lors de l'analyse de notre projet. Cela impliquait donc des risques de modifications du logiciel employé durant la définition de notre propre produit. Écrit, en PL1 sous CP-CMS, il n'est pas transportable immédiatement sur un autre système. Son emploi suppose, en outre, l'écriture d'un interpréteur de notre langage, qui produise comme "pseudo-code" le texte descriptif de ses propres règles de transduction. Les outils d'écriture de compilateur, comme CDL, ne peuvent alors plus nous servir et la gestion de toutes les données est encore à notre charge, ainsi par ailleurs que toutes les fonctions d'évaluation. Son principal intérêt est l'existence d'analyseurs phonétiques et morphologiques écrits avec ce logiciel et qui nous donnent ainsi les résultats de certaines fonctions complexes, dont nous avons besoin. Mais son emploi reste assez complexe, en particulier si des extensions d'Abrège écrit avec ce logiciel supposent qu'un utilisateur informaticien maîtrise à la fois le langage de réalisation de l'interpréteur et celui du logiciel de transduction.

Nous avons finalement choisi pour gérer nos données (descriptions de méthodes, dictionnaires, textes de définitions, abréviations, décompositions phonétiques et syllabiques de mots, etc ...) un logiciel de base de données. Et nous avons employé le langage Socrate pour décrire aussi bien l'interpréteur du langage que la saisie des diverses données de l'utilisateur.

Un logiciel de base de données a comme avantage principal la souplesse offerte dans la gestion des données qui permet de ne pas avoir conçu l'ensemble du projet dans tous ses détails dès le début, mais d'étudier librement l'implantation de chaque nouvelle relation entre des données sans être lié par la "raideur" de la notion de fichier avec les organisations et les accès "classiques".

Nous avons choisi le langage de requête Socrate dans sa version industrielle proposée par ERIA-ECA Automation. En effet, le langage ainsi proposé est volontairement défini pour être compatible sur matériel CII-IRIS 50, IBM 360, SIEMENS.

Nous avons nous-même expérimenté la transportabilité de nos programmes commencés sur IRIS 45-CII, puis développés sur IBM 360. Enfin, ce langage est assez simple, ce qui permet à l'utilisateur à la fois d'écrire ses propres fonctions après assimilation de la structure de la base descriptive d'Abrège, mais aussi facilite la documentation des programmes écrits pour la première version du prototype.

## 1.6. SCHEMA FONCTIONNEL D'ABREGE

Nous présentons, au terme de ce chapitre d'introduction, le schéma fonctionnel d'Abrège de manière à faciliter le cheminement du lecteur dans les chapitres suivants.

Abrège va comporter des fonctions d'abréviation paramétrées permettant la création d'abréviations, connaissant la méthode demandée et les textes donnés, ainsi que des fonctions d'évaluation des résultats et de la méthode. Ceci est en quelque sorte la fonction production d'Abrège.

Préalablement à la tâche de production, est réalisée la fonction de saisie. Pour cette fonction de saisie qui est en fait l'interface entre Abrège et l'utilisateur, nous définissons un langage simple qui permet de décrire les différentes phases de saisie. Ce langage est interprété et si la syntaxe de la demande de l'utilisateur est correcte, cette demande est exécutée.

### 1.6.1. La fonction de production d'Abrège

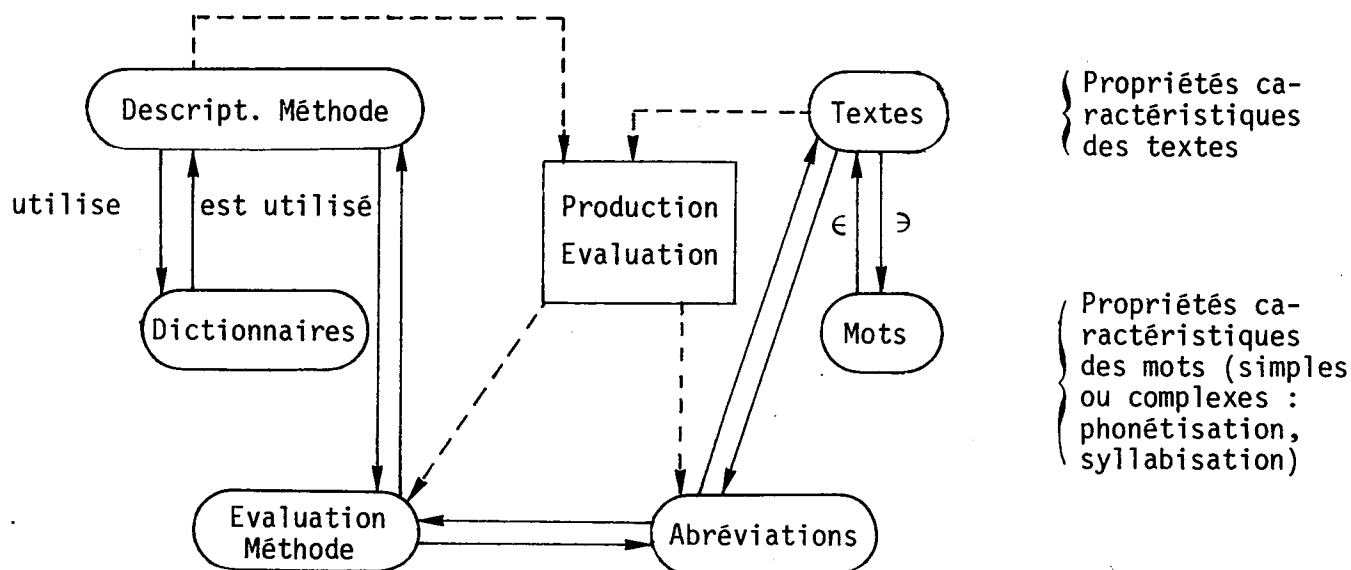
Pour produire des abréviations, le système doit disposer :

- . de la description d'une méthode de la classe de celles qui sont exécutables par Abrège
- . de dictionnaires si la méthode en utilise
- . de textes à abrégés.

La tâche de production d'abréviation consiste à évaluer les propriétés mise en jeu par la méthode pour chaque texte à abrégés et à choisir un sous-ensemble des caractères du texte en fonction de ces propriétés. Lors de ce travail, le système procède par abréviation du texte mot à mot, ce qui introduit un nouvel ensemble de données à prendre en compte : les mots. Cet ensemble, avec toutes les propriétés qui le caractérisent, est inconnu de l'utilisateur. Il est créé à la saisie, par traitement automatique des textes.

La fonction de production est réalisée par un programme qui interprète la description de la méthode pour chaque texte et chaque mot de ce texte, de manière à produire une abréviation cohérente avec la méthode. A la fin de la production d'abréviations, le système fournit aussi les évaluations

demandées de certains résultats de la méthode



Le schéma ci-dessus reflète d'une part les ensembles existant dans la base Abrège qui servent à la fonction de production, d'autre part ceux qui sont produits.

Les flèches continues indiquent l'existence de relations qui sont enregistrées de manière permanente dans la base, les flèches en pointillé indiquent quels ensembles de données caractérisées sont nécessaires à la fonction de production-évaluation et quels sont les ensembles résultants.

Des caractéristiques des mots, comme leur syllabisation ou leur décomposition en phonèmes, peuvent être mises en oeuvre après un travail supplémentaire que nous ne décrivons pas ici.

### 1.6.2. La fonction de saisie

La fonction de saisie comporte trois phases :

a) Saisie de description de méthode, avec les contrôles syntaxiques et sémantiques souhaitables pour assurer l'appartenance de la méthode décrite à l'ensemble des méthodes exécutables par le système Abrège.

b) Saisie des dictionnaires, avec création automatique d'une relation avec les mots des textes, dans la mesure où l'utilisateur envisage l'usage de dictionnaires et d'antidictionnaires dans une méthode.

c) La saisie des textes avec caractérisation automatique des données par décomposition du texte en mots ; analyse et enregistrement dans la base de certaines propriétés des textes et des mots dont l'usage est presque systématique : nombre de caractères, de mots, etc.

L'analyse de ces différentes phases suggère une démarche pédagogique qu'il paraît souhaitable de développer avec Abrège. Lorsque nous proposerons la base/système Abrège à un utilisateur pour lui permettre d'analyser les méthodes qui répondent correctement à son problème d'abréviation, nous envisageons de lui "livrer" simultanément un dictionnaire et un antidictionnaire, ainsi que le texte de description de deux ou trois méthodes répondant à des conditions d'emploi assez différentes pour guider son choix.

## CHAPITRE 2

LES ABREVIATIONS PAR LES CARACTERES

## 2.1. LES FONCTIONS A REALISER

Nous avons déjà vu que les abréviations peuvent être obtenues par choix de certains caractères de la chaîne initiale en fonction de critères de position, de fréquence, de type.

*Par exemple, nous pouvons choisir :*

- les premières lettres du premier mot,
- la lettre "centrale" du mot dont la fréquence d'occurrence dans le jeu d'essai est minimum lorsque le mot est le domaine de définition de cette fréquence,
- les deux premières voyelles (type de lettre) et les trois premières consonnes du mot.

Nous analysons dans ce chapitre les propriétés de mots et des lettres qui nous paraissent utiles pour définir les lettres retenues.

Nous voulons, avant d'en entamer la discussion, définir ce que nous appelons  
*fonction* (à réaliser)  
*propriété* d'un mot ou d'une lettre  
et *prédicat*.

La fonction est l'outil qui permet d'évaluer la propriété en tout point où nous en avons besoin, c'est-à-dire sur l'ensemble de définition de cette fonction.

*Par exemple, la fonction "position d'un mot" permet d'évaluer la propriété "premier mot d'un texte", qui est la réalisation de la fonction dont la valeur est 1.*

Nous employons aussi par la suite la notion de prédicat. Les prédicats définissent des domaines de valeur de la fonction pour lesquels certaines propriétés sont réalisées.

*Par exemple, le prédicat "Position du mot > 3 et Position du mot < 5" est vrai pour tous les mots ayant la propriété d'être en 3ème ou 4ème position. Les prédicats ont pour rôle d'annoncer le contexte de travail où l'on se situe par les valeurs des propriétés qui les vérifient.*

### 2.1.1. Les textes

Il semble normal de considérer que l'on ne choisira pas le même nombre de lettres pour abrégé un mot long et un mot court.

- a) Il faut donc pouvoir utiliser la *longueur des mots* comme paramètre d'une méthode.
- b) Les premiers mots d'un libellé, lorsqu'on a éliminé les articles, servent souvent de *pivot* au texte.

Nous appelons "*pivot*" le mot qui donne son sens principal au libellé, le reste du texte servant à le qualifier. Ce mot est souvent le premier ou le deuxième du texte et il est parfois possible de le reconnaître par une fréquence plus élevée d'occurrence dans le texte.

Il faut, pour reconnaître les pivots, éliminer des mots fréquents, mais porteurs de peu de sens comme les articles ou les conjonctions. Nous considérons donc que les *mots significatifs*, qui sont souvent les seuls pris en compte, sont ceux qui ne sont pas présents dans certains *antidictionnaires*. L'antidictionnaire joue le rôle inverse du dictionnaire, il contient des mots dont on ne veut pas. Nous revenons plus loin sur le rôle des dictionnaires.

Exemple : Ainsi, parmi les notions de comptabilité du jeu d'essai [A1] G3, nous trouvons les libellés :

*compte*

*compte-aboutissement*

*compte-contrepartie*

*compte particulier*

*sous-compte interne*

*où compte joue le rôle de pivot*

*qualifié dans un texte où tous les*

*mots peu significatifs ont déjà été*

*supprimés.*

Exemple :

Le jeu d'essai "Congrès" [A3], que nous proposons par ailleurs, contient lui, des mots non significatifs. Ainsi

"Adresse de l'hôtel ou de la résidence" peut être à peu près aussi clairement exprimé par

"Adresse hôtel ou résidence" ou même "adresse hôtel résidence", ce qui évite d'avoir trop d'éléments à prendre en compte.



Certains mots très fréquents dans un même ensemble de libellés gagnent à être abrégés toujours de la même manière.

- c) Nous voyons aussi que cette notion de pivot fait intervenir la *position du mot* dans le libellé.
- d) Il semble utile d'avoir une idée de la *fréquence des mots* et peut-être aussi de pouvoir introduire la notion d'*abréviation standard* de certains mots.
- e) Dans le cas où le libellé doit être abrégé en un seul mot, il paraît important d'adopter une politique différente d'abréviation suivant *le nombre de mots à traiter* (longueur du libellé) et de condenser davantage lorsqu'il y a plus de mots à prendre en compte.
- f) En outre, un élément externe qui faciliterait peut-être la mémorisation est l'*introduction de séparateurs* entre les abréviations de chaque mot pour donner à l'identificateur une structure de mot composé, proche du texte.
- g) Enfin, très extérieurement à toute structure du texte, on peut imposer une *longueur maximale* aux abréviations obtenues. Dans le jeu d'essais G3 [A1] par exemple la longueur imposée par l'utilisateur est de cinq caractères.

### 2.1.2. Les mots

Nous avons établi une liste des propriétés de mots comme éléments des textes, nous étudions maintenant les propriétés de constituants des mots, *les caractères*, du moins celles qui nous paraissent importantes pour le mécanisme d'abréviation.

- A. Les caractères ont quelques *propriétés extrinsèques* au jeu d'essai, comme
  - leur *fréquence reconnue statistiquement sur une langue donnée*ou
  - leur *type*, voyelle, consonne (labiale, dentale) lié à des propriétés phonétiques.
- B. Dans l'ensemble des textes traités, ils ont surtout une *fréquence* mesurable en *absolu* ou en *relatif*, intrinsèque au jeu d'essai, au texte. La fréquence absolue est celle mesurée par exemple sur le langage ou le jeu d'essai, la fréquence relative étant la fréquence absolue comparée entre les lettres, par exemple, relativement au mot considéré.

Dans les mots, ils ont :

- une *position*,
- une *fréquence* dans ce mot.

Enfin, on peut envisager de considérer différemment certaines suites de caractères, les diphtongues (2 voyelles) par exemple, ou deux lettres répétées.

Si nous devons considérer toutes ces propriétés, il faut maintenant bien préciser comment elles sont évaluées, et sous quelle forme elles peuvent entrer dans le choix fait dans l'abréviation.

## 2.2. LA STRUCTURATION DES PROPRIETES

Le principe général d'implantation de Abrège sous Socrate consiste à mettre en évidence les propriétés permanentes des *ensembles* sur lesquels travaillent les méthodes, à travers une structure Socrate où sont enregistrées ces caractéristiques. Suivant l'utilisation que l'on veut en faire, celles-ci sont classées en deux étapes. Celles d'un usage très courant sont évaluées lors du stockage des jeux d'essai ; celles plus spécialisées ne sont calculées que lorsqu'une méthode y fait explicitement appel.

D'où la structure de la base.

/\* Libellés et mots des jeux d'essai \*/

entité 2000 LIBELLE /\* Libellés \*/

début

entité 6 TXT

début

LIB-CONT mot (30)

fin

LIB-LONG de 1 à 180

NB-MOIS de 1 à 40

/\* Liste non redondante des mots du texte \*/

entité 40 REF-MOTS

début

NO-MOTS de 1 à 5000

OOO-LIB de 1 à 40

fin

/\* Synonymies créées par abréviation-liaison par arneau \*/

/\* Conservation des résultats obtenus de la méthode en cours d'exécution \*/

REF-ABREV réfère ABR-SYN de une ABREVIATION

fin

entité 5000 MOTS-LIB /\* Mots des jeux d'essai \*/

début

MOT-CONT mot (30) avec clé unique fin

FREQ-GEN de 1 à 1000 /\* Nbre occurrences \*/

/\* Fréquence des lettres du mot \*/

FREQ-LITT de 0 à 100 /\* Pourcentage \*/

entité 26 DEC-LETTRE

début

FREQ-LET de 1 à 10 /\* Nbre occurrences \*/

fin

/\* Référence vers des Abrev-non-calculées \*/

REF-MOT-DIC de 1 à 2000

LIST-ABR inverse tout ABRELIST de un MOTS-DIC

/\* Références vers des décompositions \*/

/\* Même Num pour Syll ou Phon \*/

NO-DECPHON de 1 à 5000

fin

/\* Abréviations calculées \*/

entité 2000 ABREVIATION /\* Un libellé donne un mot \*/

début

ABR-SYN anneau

/\* Les synonymies ne sont référées que pour une méthode donnée \*/

ABR-CONTENU mot (30) avec clé unique fin

fin

A. Reprenons terme à terme la structure pour introduire les raisons qui nous ont amenés à choisir ce mode de représentation, mettre en évidence ce qui est un choix délibéré et ce qui est une contrainte du SGBD (Système de Gestion de Base de Donnée) Socrate [S1].

Les *entités* en Socrate sont des ensembles de données répétitives que l'on nomme, et auxquels on peut se référer lorsqu'ils sont au niveau le plus haut. Ces entités contiennent des caractéristiques de type *mot* ou *nombre* (entier seulement) ou *liste de valeurs*. Les caractéristiques "mot" ou "nombre" peuvent parfois servir de clé de recherche pour l'entité qui les contient. Ainsi, le "dictionnaire" MOTS-LIB qui contient tous les mots utilisés dans les libellés, admet pour clé de recherche unique et alphabétique MOT-CONT qui est le contenu du mot stocké dans l'entité. L'annonce que nous faisons par la phrase "avec clé unique fin" de l'utilisation de cette clé permet, en outre, d'utiliser automatiquement l'adressage dispersé très performant fourni par Socrate. Néanmoins, lorsque certaines informations dans la base ont une implantation fixe, comme ici le dictionnaire des mots, l'accès le plus rapide est donné en produisant le numéro d'occurrence de la réalisation de l'entité référée. C'est la raison pour laquelle, pour les libellés, nous avons adopté la structure en deux niveaux : l'entité "LIBELLE" (1er niveau) contient les références vers les mots du texte par NO-MOTS de REF-MOT, qui contient à son tour le numéro de l'entité MOT-LIB (2ème niveau) correspondant au mot du texte.

La position du mot dans le libellé est donnée par le numéro d'occurrence de la réalisation dans l'entité REF-MOT correspondante.

Enfin, en stockant dans l'entité ABREV les chaînes abrégées images des libellés par la transformation des méthodes, nous utilisons cette chaîne de caractères abrégée comme clé, ce qui permet de repérer très facilement les libellés pour lesquels une méthode donnée produit une même abréviation. Nous construisons alors un anneau chaînant entre eux ces libellés, ce qui évite de nouvelles recherches lorsqu'on veut détecter des collisions d'abréviation ; cet anneau est lui aussi géré par Socrate.

B. Après ces quelques remarques sur la structuration de la base, quelques explications sur l'écriture d'une structure en Socrate :  
il est nécessaire de fournir, lors de la définition d'une base Socrate, une borne supérieure absolue du nombre de réalisation de chaque entité. Seules existent, dans l'espace réel de la base, les entités effectivement

engendrées. Pour une entité, sa portée est définie par les mots *début*, *fin*. Une entité peut contenir plusieurs autres entités.

Les caractéristiques des entités sont soit de type "*mot*" (avec sa longueur), soit de type "*entier*" compris entre une valeur minimum et maximum, soit de type "*référence*", soit enfin de type "*liste de valeurs*", chaque "*valeur*" étant considérée comme une chaîne de caractères. L'écriture est celle donnée dans notre structure.

Il est nécessaire de bien faire la différence dans la structure que nous proposons, entre le texte, sous forme libre, du libellé qui est considéré comme au maximum une chaîne de 180 caractères stockés (pour pouvoir être manipulés) dans 6 mots Socrate de 30 caractères chacun - limite imposée par le système Socrate - et les mots du libellé référencés par REF-MOTS et stockés en tant que chaîne de caractères (de 30 caractères au plus) dans MOT-CONT de MOTS-LIB.

#### ✓ 2.2.2. Génération de la base

Dans cette structure, on voit qu'un texte à traiter appelé libellé est actuellement limité à une chaîne de caractères composée au plus de 40 mots. Les deux caractéristiques de longueur en caractères du texte et de longueur en mots sont d'ailleurs stockées de manière à n'être calculées que lors de la saisie.

Le découpage du texte en mots est effectué lors de la saisie. Les mots ne sont pas stockés dans l'entité "LIBELLE" mais référencés par elle. Le stockage se fait dans l'entité "MOTS-LIB" où chaque mot n'est présent qu'une seule fois avec toutes ses caractéristiques. Ceci évite la redondance des traitements des mots, facilite les mesures de fréquence, mais augmente un peu le temps de saisie. Tout mot doit d'abord être cherché dans ce dictionnaire des mots avant d'être stocké. Cette recherche par clé (différenciation alphabétique) n'est pas reproduite trop souvent car nous stockons dans le libellé le numéro de réalisation de l'entité "MOTS-LIB" comme moyen ultérieur de recherche.

Ce début de description du processus de création dans la base des éléments utiles au travail ultérieur met en évidence certains chaînages qui font appel à la notion de réalisation physique d'une entité, plutôt qu'aux chaînages implicites réalisés par le système. Nous avons souvent choisi cette méthode pour augmenter la rapidité des recherches sur une base qui, par ailleurs, dans son état actuel présente une certaine permanence. Cela nous oblige à gérer nous-mêmes bien des modifications, mais elles sont peu nombreuses par rapport au grand nombre d'accès ou de recherches auxquels aboutissent les méthodes. Nous revenons plus tard sur ce point.

En outre, le nombre d'occurrences des lettres de l'alphabet dans le mot est définitivement stocké à l'introduction d'un mot nouveau dans "MOTS-LIB" pour éviter plusieurs balayages du mot, caractère par caractère.

### 2.2.3. Les dictionnaires

A cette structure, il faut ajouter, ainsi que nous l'avons vu en 2.1, des possibilités de suppression de mots inutiles et d'enregistrement d'abréviations spéciales ou standards. Il semble logique de prévoir pour ces deux caractéristiques divers dictionnaires et antidictionnaires que l'utilisateur peut utiliser, créer, modifier et auxquels il peut accéder suivant des jeux de priorité dont il est seul maître.

Les antidictionnaires sont des listes de mots à ne pas prendre en compte. Cela est équivalent à des listes de mots dont l'abréviation est vide. C'est ainsi qu'ils vont être stockés et reconnus comme "antidictionnaires" dans la base. Mais au niveau du traitement, ils jouent bien ce rôle opposé à celui d'un dictionnaire puisque nous supprimons des mots du libellé traité, les mots appartenant aux antidictionnaires utilisés par la méthode, et que nous réduisons le texte sur lequel s'applique la méthode aux seuls mots significatifs. Un *dictionnaire*, lui, assure la correspondance entre un mot et son abréviation. Cette abréviation peut être occasionnellement réduite au mot vide.

Nous avons décidé d'enregistrer, à titre d'ensemble fourni avec la base, un antidictionnaire ANTIDIC et un dictionnaire d'abréviations spéciales STANDIC (Annexe [A4]). Par ailleurs, en incluant dans la base des dictionnaires que l'utilisateur peut nommer, créer, modifier à son gré, nous pensons lui laisser une liberté suffisante de ne pas calculer certaines abréviations, mais de les stocker et de les introduire en fonction de propriétés (position du mot ...)

qu'il choisit. Le détail de cette saisie est donnée au chapitre VI, nous explicitons ici seulement la structure qui reflète les options suivantes d'implantation :

```
/* Dictionnaires des abréviations non calculées */
entité 10 CAR-DICO      /* Nom dictionnaire */
début
  NOM-DIC MOT (10)
  ANTIDIC MOT (1)      /* A N */
fin

entité 2000 MOTS-DIC   /* Mots et abr. des dico */
début
  MOT-CONT MOT (30) avec clé unique fin
/* Abréviations */
entité 10 ABRELIST
début
  ABRE-DON MOT (10)
fin
fin
```

Un dictionnaire est connu par son nom. A ce nom, l'entité CAR-DICO permet d'affecter un numéro. Si un mot ayant une valeur donnée dans MOT-CONT de MOTS-DIC (structure unique) appartient au n<sup>ième</sup> dictionnaire, la caractéristique ABRE-DON de la n<sup>ième</sup> entité de ABRE-LIST sera définie. Cette méthode permet de ne conserver qu'une seule fois le mot entier et de disposer dans la même entité de tous ses équivalents.

Nous n'étudions ici que l'implantation des dictionnaires dans la structure de base et nous expliquons le jeu des priorités au niveau de la description des méthodes (2.4.4. et 6.2.2., Règles 2 et 6).

### 2.3. REALISATION DES FONCTIONS DE BASE

Nous avons déjà dit que nous voulions composer diverses propriétés pour choisir les dictionnaires ou le mode de calcul par lesquels nous allions abrégé un libellé composé de mots. Pour cela, il faut d'abord disposer de tout un jeu de fonctions élémentaires qui évaluent ces propriétés et de certaines fonctions paramétrées de traitement de caractères. Nous les reprenons une à une.

Les fonctions évaluées lors de la saisie et qui sont stockées sont :

- la longueur des libellés en mots et en caractères
- la longueur des mots en caractères, leur position dans le libellé
- la fréquence des lettres dans un mot.

Les *fonctions enregistrées* dans la base, lors de la création de celle-ci, sont en fait des tables de données permettant d'initialiser le contenu de certaines entités. Elles sont évidemment extrinsèques aux jeux d'essai :

- la fréquence des lettres dans la langue de référence
- le type des lettres.

Les dictionnaires sont fournis à part par l'utilisateur en une procédure de saisie spéciale qui n'est pas obligatoire s'il n'en a pas besoin (6.3.). Nous considérons comme des paramètres de la méthode exprimée par l'utilisateur les priorités respectives des dictionnaires pour le cas où un même mot appartient à plusieurs dictionnaires pris en compte en même temps, l'insertion de séparateurs entre les mots abrégés lorsqu'ils sont concaténés pour former un identificateur, la prise en compte d'une longueur maximale et les suites de lettres qui jouent un rôle spécial.



C'est aussi dans les descriptions des méthodes que l'utilisateur du système exprime comment, en fonction de certaines propriétés, il retient certaines lettres pour en faire une abréviation.

#### 2.4. LES METHODES D'ABREVIATION OU LA COMPOSITION DES FONCTIONS

Nous venons de définir les fonctions élémentaires dont nous nous servons pour choisir un mode d'abréviation. Montrons maintenant ce qu'est une méthode.

Une méthode travaille mot à mot en associant à chaque mot, caractérisé par composition de plusieurs prédicats, une manière d'abrégé ce mot liée à ces prédicats. Par ailleurs, l'ensemble des mots d'un libellé ayant été abrégé, le résultat - qu'il soit une abréviation unique ou un texte abrégé - peut être remanié par un travail global sur le texte intermédiaire.

##### 2.4.1. Propriétés générales du résultat

Nous verrons plus tard que les propriétés globales que nous proposons en option d'une méthode sont déclarées en tête de celle-ci (chapitre 6).

Ce sont par exemple :

*la longueur maximum admise lorsque l'abréviation du texte du libellé est concentré en un mot unique.*

C'est aussi

*la possibilité de séparer les abréviations de chaque mot du libellé par des tirets pour former un mot abrégé composé.*

Ce sont enfin

*des traitements spéciaux de caractères, suppression de doublets, ou de collisions lorsque deux textes ont même abréviation résultant d'une même méthode.*

Nous offrons aussi à l'utilisateur

*la possibilité de se définir des traitements très spécifiques et de les appeler sous forme de fonction.*

### 2.4.2. Propriétés des libellés

Montrons maintenant comment travaille une méthode dans sa tâche d'abréviation mot par mot.

Un premier prédicat sert de *filtre pour caractériser le libellé* lui-même ou *filtre-libellé*, de manière à travailler différemment suivant cette propriété du texte. Nous n'avons retenu que la longueur, en nombre de mots.

*Ainsi, une méthode peut comporter la description d'un mode de calcul spécifique aux textes de moins de trois mots, et d'un autre pour les textes de 3 à 10 mots, un autre enfin pour les textes plus longs.*

Cette longueur n'est cependant pas la donnée initiale que nous avons stockée dans la base. Nous avons, en effet, pensé que la suppression de mots non significatifs par usage d'antidictionnaire devait conduire à ne considérer que les mots significatifs restant après "transformation-réduction" du texte initial pour mesurer la longueur intéressante, en nombre de mots significatifs. Elle est implicitement utilisée dès qu'une méthode utilise des antidictionnaires. Il est bien évident que l'usage de ce filtre est facultatif, la méthode n'étant alors pas décomposée en plusieurs *filtres-libellés*.

*Par exemple, la méthode suivante : pour tout libellé, prendre les deux premières lettres du premier mot significatif et la première lettre de chacun des suivants.*

### 2.4.3. Propriétés des mots

Dans la portée d'un même *filtre-libellé*, nous caractérisons ensuite le mot par plusieurs prédicats évalués successivement par une *intersection logique* entre eux. Lorsque cette suite de prédicats a donné une réponse vraie, sont choisis soit les dictionnaires à utiliser pour rechercher une abréviation au mot, soit le ou les calculs permettant de choisir certains caractères dans le mot pour obtenir ainsi une autre abréviation.

Ce *filtre-mot* est construit à partir des prédicats suivants, évalués sur les fonctions proposées au paragraphe précédent.

1) Le premier est le *prédicat de position* qui fournit l'intervalle de position dans lequel doit se trouver le mot pour que le calcul s'applique.

2) Nous considérons ensuite une *expression de fréquence unique*. Elle est complexe car, à ce niveau, le mot peut être caractérisé par trois fréquences :

- a - la fréquence générale par rapport à une langue déterminée,
- b - la fréquence de ce mot par rapport à tout le jeu d'essai traité,
- c - enfin, la fréquence dans le texte du libellé.

Une expression de fréquence permet de prendre en compte l'une seule des trois mesures ci-dessus : nous pensons en effet que, lorsqu'on s'intéresse à l'une d'elles, les autres ne sont plus simultanément très significatives.

Cette expression permet de spécifier des calculs pour des mots ayant une fréquence comprise dans un intervalle absolu, ou pour une fréquence dans un intervalle proche des maximum ou minimum absolus atteints sur tout le jeu d'essai par la fréquence considérée. Enfin, elle donne aussi le droit de demander le traitement qui suit pour le ou les mots du texte pour lesquels la fréquence considérée est la plus élevée ou la plus faible.

*Par exemple, l'expression peut prendre les formes suivantes :*

- fréquence générale du mot  $\leq 3 \%$ ,
- fréquence relative du mot dans le jeu  $\geq 90 \%$  du maximum atteint sur le texte
- fréquence relative du mot dans le jeu = MAX (fréquence de tout mot du jeu).

3) Ayant situé les mots dans les textes, les ayant ensuite caractérisés par leur fréquence, nous pouvons encore différencier le mécanisme d'abréviation par application d'un *prédicat de longueur* donnant une borne inférieure et supérieure de la longueur du mot pour lequel le mécanisme sera défini. A la différence des deux précédents, ce prédicat est intrinsèque au mot.

Ces prédicats sont, lors de l'exécution de la méthode, évalués dans l'ordre où ils ont été énoncés, constituant ainsi une partie du *filtre-mot*. Nous avons figé cet ordre pour refléter une certaine logique entre les propriétés du mot. Mais le filtre ne fait pas nécessairement appel à toutes ces propriétés ; ainsi les trois filtres suivants sont possibles :

- a - Position (mot) entre 3 et 6
- b - Position (mot)  $\leq 4$  et Longueur  $< 10$
- c - Fréquence (mot)  $\geq 0.9$  Max (fréquence-jeu) et Longueur  $< 8$ .

Ces trois propriétés sont les seules considérées comme propriétés propres du mot devant éventuellement figurer pour caractériser un traitement spécifique du mot. Nous avons, cependant, voulu permettre de contrôler le traitement d'un mot en fonction des traitements précédemment exécutés sur les autres mots

du libellé.

- Ainsi, peut-on par exemple conserver dans X1 trace de la nature de la dernière lettre retenue du mot précédent, X1=1 si consonne, X1=2 si voyelle, et appliquer ensuite des traitements différents au mot en cours pour tenter d'obtenir une abréviation prononçable.
- On peut, autre exemple, garder dans X2 la longueur intermédiaire de l'abréviation globale du libellé au niveau courant de manière à modifier la politique d'abréviation suivant l'évolution de cette longueur par rapport à la position du mot dans le texte.  
Nous avons décidé d'enregistrer ces traces des exécutions antérieures dans des compteurs. Ils sont modifiés sur demande explicite de l'utilisateur au moment de l'exécution de l'action dont on veut garder la trace. Puis ils sont testés, au niveau d'un filtre-mot par exemple dans un prédicat de compteur.

Ces deux exemples ne sont pas particulièrement simples, mais ils veulent rappeler que notre propos est de pouvoir assez facilement définir de bonnes méthodes. Et l'usage montre qu'une bonne méthode n'est pas nécessairement une méthode très simple à décrire.

#### 2.4.4. Traitements du mot : dictionnaire ou calcul

A. Revenons à la composition de fonctions. Pour choisir un mode d'action, nous avons jusqu'à maintenant pris en compte :

- la longueur du texte du libellé mesurée en mots significatifs
- la position du mot dans le texte
- une relation de la fréquence du mot dans un certain contexte
- des compteurs reflétant la trace des choix antérieurs.

Pour un mot répondant au contexte ainsi cerné, une méthode propose alors un *traitement*. Ce contexte est obtenu par recherche dans les filtres-libellés du premier filtre vrai pour le libellé considéré, puis, pour un mot du libellé par la recherche du premier filtre-mot lié au filtre-libellé retournant une valeur vraie pour le mot.

Exemple : Antidictionnaire ANTIDIC

filtre-libellé : longueur  $\leq$  3

⋮  
filtre-libellé : longueur  $>$  4 et longueur  $\leq$  7

⋮  
filtre-libellé : longueur = 4

    filtre-mot : Position = 1 : traitement (31)

    filtre-mot : Position  $\leq$  3 et longueur  $\leq$  4 : traitement (32)

    filtre-mot : Position = 3 et longueur  $>$  4 : traitement (33)

    ⋮

Le filtre qui s'applique au mot "mange" dans le libellé

"le petit chien mange son os" est le suivant :

Par antidictionnaire, le texte devient "petit chien mange os"

texte de 4 mots et "mange" de longueur = 5  $>$  4 en troisième position  
se voit appliquer le traitement 33.

On voit que cette recherche est séquentielle. Elle est détaillée au Chapitre 6.

B. Le traitement est défini à partir de deux modes de recherche d'abréviation. Le premier consiste à avoir prédéfini des abréviations dans des dictionnaires saisis avant exécution de la méthode (Chapitre 6), et à rechercher pour chaque mot son abréviation dans un ou plusieurs dictionnaires. L'ordre d'appel des dictionnaires est une loi de priorité qui provoque l'arrêt de la recherche quand l'abréviation est trouvée. On peut, à ce propos, noter que la structure Socrate que nous avons définie et l'existence d'une chaîne d'inverses donnant la correspondance entre un mot et les dictionnaires qui le contiennent accélèrent grandement cette recherche.

Si une abréviation prédéfinie existe dans les dictionnaires pris en considération dans le contexte de traitement, elle est utilisée. Si aucune abréviation prédéfinie n'est considérée, c'est-à-dire si dans ce contexte de traitement aucun dictionnaire n'est utilisé, c'est que le mode de traitement est de type calcul, et dans ce cas, un calcul est défini dans la méthode utilisée.

Lors de l'utilisation des dictionnaires, lorsqu'aucune abréviation prédéfinie n'existe pour le mot considéré, il se peut qu'aucun mode de calcul n'ait été défini dans ce contexte. Le mot est alors laissé pour compte et son abréviation est vide ; on passe à la recherche du contexte de traitement du mot suivant du libellé.

Par contre, si un mode de traitement de type calcul a été défini dans la méthode, on évalue celui-ci pour ce mot courant.

On voit donc que, dans la mesure où l'on fait appel à eux, les dictionnaires ont une priorité absolue sur les calculs. Maintenant, précisons cette notion de calcul.

#### 2.4.5. Propriétés des caractères ou lettres prises en compte dans un mode de calcul

Un calcul est l'exécution d'un certain nombre d'actions dont le contexte est défini par évaluation de quelques prédicats portant sur les lettres du mot.

Le calcul s'exécute de la manière suivante. Jusqu'à ce qu'on rencontre éventuellement une indication explicite contraire, on se déplace dans le mot de gauche à droite.

Pour la lettre courante, qui au début est la première du mot, on recherche le contexte d'action qui lui correspond parmi les combinaisons de prédicats définissant les *filtres-lettres liés au filtre-mot* dans lequel nous sommes.

Ces prédicats sont d'abord :

1) Un *prédicat de position* qui exprime l'intervalle de position des lettres dans le mot, pour lesquelles ce qui suit est à considérer. Ainsi, peut-on faire jouer des rôles particuliers aux premières ou aux dernières lettres du mot.

2) On peut ensuite définir un *prédicat de type*. Nous n'avons pas encore rencontré ce prédicat, car nous ne possédons pas sur les mots de caractérisation intrinsèque. Nous pourrions envisager de définir un mot comme nom, ou sujet, verbe, adjectif, etc ... en mettant en évidence une caractéristique syntaxique, mais nous n'en disposons pas actuellement. Par contre, nous savons reconnaître une voyelle d'une consonne et nous envisageons même d'affiner ces caractérisations de type phonétique en : labiale, dentale, gutturale, sourde, muette ... C'est l'utilisation de ces caractérisations des lettres qu'utilise le prédicat de type par simple citation.

3) Le *prédicat de fréquence* défini pour les lettres, évalue au niveau de la lettre courante le même genre de propriété que celle du mot dans le texte. On dispose seulement d'une mesure supplémentaire, celle de la fréquence dans le mot. L'expression de fréquence a la même structure que celle des mots.

4) Enfin, comme dans le filtre-mot, des *prédicats de compteurs* permettent de garder une trace de situations antérieures.

*Par exemple, le type de la dernière lettre choisie pour appartenir à l'abréviation ou la longueur de l'abréviation par rapport au mot initial, etc .*

Un filtre-lettre est donc composé de l'un ou de plusieurs de ces prédicats qui sont évalués dans l'ordre de l'exposition ci-dessus pour savoir quelles actions effectuer en fonction des caractéristiques de la lettre courante. Ces actions étant celles qui forment une abréviation, nous les exposons maintenant.

#### 2.4.6. Les actions de sélection

A ce niveau, il est possible d'exécuter cinq types d'actions :

1) La première consiste à *prendre*, en fonction des caractéristiques de la lettre courante, cette lettre si elle a telle ou telle nouvelle caractéristique.

*On a par exemple la 4ème lettre et on ne veut la prendre que si c'est une voyelle.*

*Ou bien la lettre courante étant une des moins fréquentes du jeu d'essai, on ne veut la prendre que si c'est une consonne.*

*Ou bien encore, la lettre courante étant une voyelle, on accepte de la prendre si c'est une muette (par exemple, pour finir une abréviation).*

PRENDRE une lettre, cela veut dire qu'on l'ajoute par concaténation aux lettres déjà retenues dans le mot ou le libellé pour faire partie de l'abréviation. Cela veut dire aussi qu'on la place dans l'abréviation créée par sélection de caractères en respectant l'ordre des caractères dans le texte initial.

Il est aussi possible que cette lettre ne réponde pas à la caractérisation du verbe PRENDRE signalée plus haut. On peut alors soit considérer "qu'elle a eu ses chances" et passer à la suite, soit vouloir lui appliquer une autre action, et dans ce cas il ne faut pas avancer.

Le verbe PRENDRE est donc précisé et conditionné par l'action d'avancement dans la chaîne de caractères qui dépend du type de lettre cherché. Si tout type est accepté dès que l'on a exécuté PRENDRE non conditionnel, il faut avancer dans la chaîne de caractères car toute lettre du mot initial n'intervient qu'une fois dans l'abréviation.

Lorsqu'on a pris une lettre effectivement, il est possible de lier à cette action des modifications de valeur de compteurs, contrôlés par l'utilisateur qui veut garder pour la suite une trace de cette action.

2) La deuxième action que nous nous sommes proposés de réaliser ensuite consiste dans la démarche suivante. La lettre courante étant définie comme précédemment, nous voulons *trouver* parmi les lettres qui suivent un certain nombre d'entre elles répondant à certaines conditions et les prendre pour l'abréviation.

TROUVER va donc être caractérisé par le champ d'action dans lequel nous voulons qu'il agisse puis par les caractéristiques des lettres que nous cherchons. Ce champ se situe à droite du caractère sélectionné par filtre.

*Exemple : A partir de la lettre en cours de traitement, nous désirons trouver trois consonnes si elles existent. Mais nous pouvons aussi sur le même champ trouver deux voyelles en ne sachant pas dans quel ordre les unes et les autres apparaissent.*

*Par ailleurs, ces cinq lettres étant trouvées sur le champ donné, nous ne voulons plus repasser pour une recherche de caractère sur les lettres retenues.*

Le déplacement de l'index de position dans le verbe TROUVER doit être assez contrôlé, pour permettre les deux recherches précédentes sur un même champ. Le verbe TROUVER doit donc en outre être qualifié pour spécifier si le balayage d'une partie du mot auquel il conduit se fait de manière irréversible avec déplacement de l'index de position dans la chaîne de caractères, ou si ce balayage ne provoque aucun déplacement effectif dans la chaîne jusqu'à ce que tous les choix aient été faits sur le champ de recherche.



Comme précédemment enfin l'action de TROUVER des caractères qui sont retenus pour l'abréviation peut nécessiter des modifications de compteur pour en avoir une trace.

Nous voyons que TROUVER peut provoquer de grands déplacements dans le mot ; dans tous les cas, le contexte de filtre-lettre suivant sera choisi parmi ceux proposés pour la lettre dont la position est définie par l'index courant.

*Dans l'exemple ci-dessus, la position de l'index à la fin de la recherche et du calcul est sur la lettre qui suit la dernière lettre sélectionnée, c'est-à-dire celle de plus haut index parmi les cinq. S'il n'y en a pas cinq, c'est pareil.*

*Par contre, si pour les voyelles, sachant que nous avons déjà trouvé les consonnes, nous acceptons de nous déplacer, le déplacement a lieu jusqu'au bout du champ prédéfini, éventuellement donc jusqu'au bout du mot s'il ne contient plus qu'une voyelle. Les lettres trouvées sont prises en compte pour l'abréviation dans l'ordre de leur apparition dans le mot.*

Le système ainsi proposé offre des possibilités assez souples de sélection de lettres parmi les chaînes de caractères variées des divers mots.

3) Mais il peut être souhaitable à certains moments de reconnaître l'existence de lettres ayant certains caractères définis dans le filtre-lettre sans exécuter ensuite d'autre action que d'en garder la trace dans un compteur ou même plus simplement se contenter d'avancer l'index de position dans le mot.

L'action "vide" a pour rôle de provoquer le simple déplacement d'index, car ce déplacement, sauf quand il est nié comme précédemment, est transparent à l'utilisateur.

Une modification de compteur isolée est exécutée sur demande à chaque passage sur certains filtres-lettres, en même temps que s'effectue le déplacement d'index.

Nous avons ainsi vu les actions que l'on propose comme conséquence de l'application d'un filtre-lettre. Elles permettent d'utiliser le contexte créé ou un contexte plus vaste pour proposer la sélection de caractères.

4) Nous avons signalé que le balayage du mot se fait de gauche à droite dans la situation normale. Mais, si pour terminer une abréviation on veut récupérer les dernières consonnes ou voyelles du mot, il paraît souhaitable de pouvoir reprendre les choix précédents en procédant en sens inverse pour le balayage à partir de la fin du mot. Nous proposons cette action qui conditionne ensuite, à la fois le champ d'application des filtres-lettres évalués après elle et celui des choix et des fonctions. Cette action de *saut au bout du mot avec inversion de sens* n'est exécutée qu'une fois sur un mot donné et le balayage est limité à la partie du mot non encore utilisée.

5) Nous avons enfin ajouté la possibilité de demander des *actions spécifiques* à l'utilisateur par *appel de fonction*. Cela permet une extensibilité des choix par réalisation de programmes de traitement spéciaux sur la base de l'utilisateur.

Ces actions spécifiques ne peuvent pas être en effet toutes prévues dans l'interprétation que nous faisons des descriptions de méthodes écrites dans le langage décrit au Chapitre 6.

Citons un exemple :

*Un éventuel utilisateur a demandé que l'on essaye de faire des abréviations prononçables. Pour qu'il en soit ainsi, nous devrions avoir des fonctions complexes étudiant à chaque niveau si la séquence de lettres retenues correspond à quelque chose de prononçable en français, fonction elle-même très difficile à définir. Nous ne le faisons pas, mais supposant que les filtres et les actions conséquentes correspondent à des alternances de voyelles et de consonnes, ce qui est déjà un pas vers quelque prononciation, il reste que d'un mot au suivant l'alternance n'est pas donnée. On peut envisager alors une fonction qui "mette des lettres en réserve" pour l'abréviation en fonction du mot suivant, et qui ainsi réalise cette alternance.*

#### 2.4.7. Un premier schéma de la grammaire descriptive d'une méthode

Nous avons vu comment les fonctions sont composées pour définir des contextes de travail pour des choix de caractères. Nous donnons pour clore ce paragraphe la syntaxe de la grammaire qui correspond au mode de description et de conception de méthode que nous avons proposé.

*Méthode* = options-globales\* antidiCTIONNAIRES\* filtre-libellé<sup>+</sup>.

*Filtre-libellé* = prédicat-longueur<sup>0/1</sup> filtre-mot<sup>+</sup>.

*Filtre-mot* = prédicat-position<sup>0/1</sup> prédicat-fréquence<sup>0/1</sup> prédicat-longueur<sup>0/1</sup>  
prédicat-compteur<sup>0/n</sup> traitement.

*Traitement* = dictionnaires | dictionnaires calculs | calculs.

*Calculs* = [filtre-lettre actions]<sup>+</sup>.

*Filtre-lettre* = prédicat-position<sup>0/1</sup> prédicat-type<sup>0/1</sup> prédicat-fréquence<sup>0/1</sup>  
prédicat-compteur<sup>0/n</sup>.

*Actions* = [choix-prendre | choix-trouve]<sup>\*</sup> modif-compteur\* fonctions\*  
inversion-sens<sup>0/1</sup>.

A chaque terme de la partie droite d'une règle est associé le nombre d'occurrences possibles de celui-ci, écrit sous la forme  $a^n$

$$a^{n/m} = \{a^k, \forall k \in [n, m]\}$$

$$a^* = \{a^n \forall n \in [0, +\infty]\}$$

$$a^+ = \{a^n \forall n \in [1, +\infty]\}$$

#### 2.5. EXPRESSION DES METHODES DANS LA STRUCTURE PRECEDENTE. TRADUCTION D'ABREVIATIONS MANUELLES

Au schéma proposé doit maintenant correspondre un lot d'exemples pour évaluer l'adéquation du schéma à l'expression de méthode. Nous disposons de divers jeux d'essai, comportant d'une part des libellés, d'autre part les identificateurs que des analystes avaient créés manuellement par abréviation. Nous avons décidé dans les premiers exemples d'essayer d'obtenir automatiquement les mêmes résultats que ceux qu'ils ont définis manuellement, ou du moins des résultats proches des leurs.

Dans le projet G3 développé par E.D.F., toutes les notions sont nommées par abréviation de la "désignation de l'information" (Annexe [A1]), le mode d'abréviation est très bref, pas très mnémonique et limité à cinq caractères. Nous l'avons pris comme premier exemple ; le deuxième est extrait du projet DM, Dépense de matériel de l'Etat ; nous disposons d'une partie des identificateurs définis et répertoriés dans le dossier d'analyse, ainsi que des libellés qui leur correspondent.

Le nombre d'identificateurs véhiculés par DM est très élevé (400 environ dans le prototype) et l'attitude des analystes qui les ont baptisés proche de la nôtre : recherche d'unicité et de mnémonicité.

De ces deux exemples, nous essayons de déduire quelques pistes de définition de "bonnes méthodes" que nous appliquons à ces jeux d'essai. Nous ne discutons les résultats qu'au paragraphe 2.6, nous intéressant surtout ici à l'expression des méthodes sous forme des prédicats proposés et non à leur exécution.

#### 2.5.1. G3, Méthode d'abréviation devant donner des noms différenciés de cinq caractères au plus

Pour trouver la méthode d'après le jeu d'essai manuellement défini qui nous a été donné, nous analysons deux groupes donnés : les libellés et leurs abréviations (Annexe [A1]).

Les libellés : Ils n'excèdent pas cinq mots avec une majorité à trois ou quatre mots.

Ils comportent quelques groupes de mots très répétitifs qui servent de pivot.

Ce sont :

*organisme*

*unité technique*

*compte*

*code et code élémentaire et code taxe*

*acte élémentaire*

*taxe*

*majoration*

*test fiscal, test fiscalité et test.*

Nous pouvons alors constater que ce pivot a été généralement abrégé sur trois lettres s'il s'agit d'un mot long et isolé (sauf code = c et taxe = taxe) et sur une seule lettre par mot s'il s'agit d'un pivot composé de deux mots.

Certains mots sont systématiquement négligés. C'est surtout parce qu'ils n'apportent guère d'information comme "nombre" dans "nombre d'heures", "numéro" dans "numéro de pièce", l'objet étant désigné par son contenu, ou parce qu'ils sont moins importants que d'autres informations comme "code" dans "code rubrique" ou "code heure". Pouvons-nous traduire toutes ces variantes dans l'expression algorithmique d'une méthode d'abréviation ? Il est certain que non car, à l'étude, la procédure suivie par celui qui a défini les noms abrégés tient compte d'habitudes peu déterministes. Comment exprimer en effet que nous abrégions "mouvement" par "mvt" et "employeur" par "emp", à moins de beaucoup développer les dictionnaires spéciaux au détriment du calcul, surtout quand le même "mouvement" est abrégé "mouv" ailleurs.

Il ne nous est pas non plus possible de réaliser l'interversion de termes qui apparaît par exemple entre :

*"test fiscal travaux atelier imposition"*  
et *"tfiat"* qui est donc l'abréviation de *"test fiscal imposition travaux atelier"*. Cette inversion de position réalisée systématiquement peut être supprimée par l'utilisateur sans que la contrainte imposée soit trop lourde.

Nous proposons la méthode suivante :

#### Méthode 1 : Définition en clair

Pour les libellés d'un seul mot, prendre leur abréviation, si elle existe dans le dictionnaire spécial, ou le standard, sinon effectuer le calcul suivant :

Conserver les mots de quatre lettres au plus, sinon prendre les trois premières lettres des mots plus longs (plus de 4 lettres) et la quatrième si c'est une consonne.

Pour les libellés de plusieurs mots, ne conserver que les mots significatifs. Abréger le premier mot grâce à son abréviation spéciale si elle existe, sinon en prenant la première lettre des mots de moins de 8 caractères ou les trois premières pour ceux qui sont plus longs.

Dans les libellés de quatre mots, pour le deuxième mot, prendre sa première lettre si le mot est un pivot fréquent, sinon son abréviation spéciale si elle existe, ou effectuer le calcul suivant :

Prendre la première lettre des mots de moins de 8 caractères, les deux premières des mots plus longs. Prendre la première lettre des autres mots.

Dans les libellés de plus de 4 mots, prendre la première lettre de chacun.

Dans tous les cas, ne pas excéder 5 caractères.

Méthode 1 : Définition avec ABREGE

Commentaires

option	: longueur-abréviation $\leq$ 5	
antidictionnaire	: antidi	<i>seuls les mots "significatifs" entrent en jeu</i>
filtre-libellé	: longueur-libellé = 1	<i>un seul prédicat</i>
filtre-mot	: longueur-mot $\leq$ 4	<i>un seul prédicat</i>
dictionnaires	: spécial, antidi	
calcul : filtre-lettre :		<i>sans aucune élimination de lettre</i>
action	: PRENDRE	
filtre-mot	:	<i>pour les mots de plus de 4 caractères</i>
dictionnaires	: spécial, antidi	
calcul : filtre-lettre :	position-lettre $\leq$ 3	
action	: PRENDRE	
filtre-lettre	: position-lettre = 4 et type = cons	<i>double prédicat</i>
action	: PRENDRE	
filtre-libellé	: longueur-libellé $\leq$ 4	<i>libellés de 2 à 4 mots</i>
filtre-mot	: position-mot = 1 longueur-mot $\leq$ 8	<i>premier mot "court"</i>
dictionnaires	: spécial, standi	
calcul : filtre-lettre :		
action	: PRENDRE	
filtre-mot	: position-mot = 1	<i>"long"</i>
dictionnaires	: spécial, standi	
calcul : filtre-lettre :	position-lettre $\leq$ 3	
action	: PRENDRE	

filtre-mot : position-mot = 2 *deuxième mot*  
freq-jeu (mot)  $\geq$  90 % (Max (freq-jeu de tout mot))  
longueur-mot  $\leq$  8 *"court", pivot*

calcul : filtre-lettre : position-lettre = 1  
action : PRENDRE

filtre-mot : position-mot = 2  
freq-jeu (mot)  $\geq$  90 % (Max (freq-jeu de tout mot))

calcul : filtre-lettre : position-lettre  $\leq$  3 *"long", pivot*  
action : PRENDRE

filtre-mot : position-mot = 2 *non pivot,*  
longueur-mot  $\leq$  8 *"court"*

dictionnaires : spécial

calcul : filtre-lettre : position-lettre = 1  
action : PRENDRE

filtre-mot : position-mot = 2

dictionnaires : spécial *non pivot,*  
calcul : filtre-lettre : position-lettre  $\leq$  2 *"long"*  
action : PRENDRE

filtre-mot : *tout autre mot, soit 3e et 4e*

filtre-lettre : position-lettre = 1  
action : PRENDRE

filtre-libellé : *tout libellé de plus de 4 mots*

filtre-mot : position-mot = 1  
longueur-mot  $<$  8

dictionnaires : spécial, standic

calcul : filtre-lettre : position-lettre = 1  
action : PRENDRE

filtre-mot : position-mot = 1

dictionnaires : spécial, standic

filtre-lettre : position-lettre  $\leq$  3  
action : PRENDRE

filtre-mot :  
filtre-lettre : position-lettre = 1 *pour tout autre mot*  
action : PRENDRE *que le premier*

Nous voyons dans cet exemple la décomposition qu'il nous faut faire. Le groupe sélectionné par les prédicats est souvent celui que l'on recherche.

Nous avons indiqué en annexe [A1] les mots qui sont supposés être présents dans les divers dictionnaires. Ces dictionnaires sont un des avantages et un des inconvénients des méthodes. Avantage car ils permettent de créer des exceptions, inconvénient car il faut les définir en même temps que la méthode. Dans le cas de l'exemple, ils servent bien leur propos car les mots enregistrés ne sont pas trop nombreux.

Avant de pouvoir exécuter cette méthode sur le jeu d'essai correspondant, il faut prévoir une saisie des textes de libellés et des mots des dictionnaires suivant la procédure expliquée au chapitre 6 : interface avec l'utilisateur.

C'est au paragraphe 2.6. que nous discutons de la validité des résultats obtenus par exécution de la méthode en comparant ceux-ci aux abréviations manuelles qui avaient été proposées dans ce cas et que nous avons cherché à imiter.

Nous proposons maintenant un deuxième exemple, en tentant toujours d' "imiter" une abréviation manuelle.



### 2.5.2. Méthode 2 : "DM", Dépenses de matériel de l'Etat

Les données sur lesquelles nous travaillons sont présentées en Annexe [A2]. Ici encore, nous analysons les données constituées d'un texte en clair, le libellé, et d'une abréviation fixée manuellement.

Nous en dégageons quelques difficultés pour trouver une méthode et quelques éléments constructifs, à la suite de quoi une méthode est définie en clair. Elle est ensuite remplacée par les prédicats et traitements décrits en Abrège.

Les difficultés rencontrées viennent de cas particuliers dus à l'attitude des analystes qui ont défini les identificateurs.

Ce sont surtout les cas suivants :

- *Deux libellés identiques ont des codes différents [n°2], [n° 3]. Ceci provient de ce que l'analyste qui a écrit les textes des libellés n'a pas jugé nécessaire d'indiquer que l'action explicitée dans cette notion pouvait avoir des contextes différents et significatifs. Le code donné par l'analyste a donc une sémantique plus riche que celle du libellé initial. Nous ne pouvons évidemment l'admettre.*
- *Tous les codes contiennent un préfixe dont la signification est liée au rôle de la notion dans le projet "DM". Nous avons systématiquement ignoré ce préfixe. Au chapitre 7, nous proposons quelques réflexions liées à ce sujet.*
- *Certains mots des libellés sont systématiquement remplacés par un synonyme dans le code ("retour" dans le code, "renvoi" dans le texte). Bien que cette attitude paraisse gratuite, il est facile de créer cette équivalence par l'usage d'un dictionnaire.*
- *Les méthodes permettent de traiter les mots un à un dans un calcul mais pas les mots composés (liés par des tirets dans le texte initial) ou les groupes de mots. Or, dans le jeu d'essai proposé, un groupe de mots comme "bons d'engagement" est présent sous trois formes. En clair, dans quelques libellés, en abréviation standard dans les codes "beng", en abrégé*

*très bref dans certains textes de libellés "be".*

- *La difficulté peut être tournée assez facilement si la méthode n'utilise aucun prédicat de position du mot et si la chaîne "bon" est toujours abrégée "b" (par un dictionnaire spécial par exemple, et "engagement" réduit à "eng" par calcul ou par dictionnaire, la chaîne "be" qui, par bonheur, ne correspond à aucun mot courant non significatif pouvant être elle aussi sans difficulté dans un dictionnaire.*

*Mais ceci risque de nous amener à créer un nombre élevé de mots à traitements spéciaux, ce qui nécessite une définition manuelle de dictionnaires divers. Une telle situation, si elle est trop fréquente, risque de rendre inutiles les tâches automatiques.*

- *Un autre palliatif peut être un changement d'attitude de l'utilisateur qui, sachant qu'il va utiliser un procédé automatique, se contraindra, sauf dans le cas où "bon d'engagement" est une notion solitaire, à employer toujours la même terminologie (be ou beng par exemple).*

Partant de là, nous proposons la méthode suivante, dont les résultats sont discutés au paragraphe 2.6.

### Méthode 2 : Définition en clair

Pour les libellés de moins de 4 mots significatifs, prendre l'abréviation du dictionnaire spécial ou la calculer ainsi :

pour chaque mot, prendre les trois premières lettres et la quatrième si celle-ci est une consonne, sauf si cette consonne est une répétition de la troisième.

Pour les libellés de 4 mots significatifs ou plus, prendre l'abréviation du dictionnaire spécial-court, sinon la calculer ainsi :

pour le premier mot, appliquer la règle précédente isolant 4 lettres au maximum ; pour les suivants, cette règle mais en isolant un maximum de 3 lettres, deux si la troisième est une voyelle.

Cette expression condensée n'est évidemment pas possible dans le mode d'expression très décomposé que nous avons choisi pour notre présentation.

Nous traduisons donc celle-ci sous forme de filtres, prédicats, actions.

Méthode 2 : Définition avec Abrège

Commentaires

options	: doublets éliminés	
antidictionnaires	: antidi	<i>antidictionnaire standard de Abrège</i>
filtre-libellé	: longueur-libellé < 4	<i>mesurée sur les mots significatifs seulement</i>
filtre-mot	:	<i>pour chaque mot</i>
dictionnaire	: spécial	
calcul : filtre-lettre	: position-lettre < 4	
action	: PRENDRE	
filtre-lettre	: position-lettre = 4	
action	: PRENDRE consonne	
filtre-libellé	: longueur-libellé <u>&gt;</u> 4	<i>prédicat facultatif. Complément du précédent</i>
filtre-mot	: position-mot = 1	
dictionnaire	: standic	
calcul : filtre-lettre	: position-lettre <u>≤</u> 3	
action	: PRENDRE	
filtre-lettre	: position-lettre = 4	
action	: PRENDRE consonne	
filtre-mot	:	<i>pour tout mot dont la position ≠ 1</i>
dictionnaire	: speccourt, standic	<i>speccourt est déclarée par l'utilisateur et plus prioritaire que <u>standic</u></i>
calcul : filtre-lettre	: position-lettre < 3	
action	: PRENDRE	
filtre-lettre	: position-lettre = 3	
action	: PRENDRE consonne	

Dans cette méthode, nous faisons usage de trois dictionnaires et plutôt que de supprimer les doublets lors de la seule concaténation de deux mots abrégés, nous les supprimons partout.

Un utilisateur non satisfait par ce procédé standard peut demander la programmation en Socrate d'une fonction doublet qui sera appliquée lors des seules concaténations de 2 mots entre eux.

En Annexe [A2], nous pouvons voir le nombre d'éléments contenus dans les dictionnaires et antidictionnaires.

Il est à noter que les libellés de ce jeu d'essai sont beaucoup plus longs que les textes du jeu G3, ceci parce qu'ils comportent certaines conjonctions qui ont disparu dans la définition du premier jeu, mais le nombre de mots significatifs est à peu près le même.

## 2.6. RESULTATS DE DEUX METHODES IMITANT LE PROCESSUS MANUEL. DISCUSSION

Les difficultés rencontrées dans la description de méthode sous forme de prédicat ne semblent pas insurmontables. Nous étudions maintenant les résultats.

Nous regardons tout d'abord jusqu'à quel point les deux méthodes ont donné des résultats proches du procédé manuel.

Ensuite, nous regardons en quoi ces résultats sont satisfaisants ou insuffisants d'après nos objectifs d'unicité, de prononçabilité et de mnémonie.

### 2.6.1. Adéquation dans l'imitation

#### Méthode 1 :

Nous avons trois versions différentes des résultats (Annexe [A1]) pour une même méthode. Ceci montre le rôle important que jouent les dictionnaires.

Dans les *Résultats*, les dictionnaires sont réduits le plus possible. On voit alors que l'identité entre l'abréviation proposée dans le jeu d'essai et l'abréviation calculée n'est pas souvent réalisée. Cela n'atteint pas 50 %.

Par ailleurs, ces résultats présentent une ambiguïté sur un jeu de 80 libellés, ambiguïté créée par la troncature de l'abréviation à 6 caractères qui empêche de prendre dans une séquence de chiffres (conservée intégralement par option standard) le dernier chiffre seul significatif (75 et 76èmes libellés).

Qu'en dire sinon ? Que les abréviations ne paraissent pas beaucoup plus longues que celles faites manuellement, pas beaucoup plus lisibles non plus. Elles sont un peu moins bonnes sur quelques points :

- Dans les libellés courts (2 mots) les mots courts sont encore réduits à 2 lettres, alors qu'une autre méthode ou un "pivot" dans un dictionnaire améliorerait les choses. La procédure manuelle choisit plus librement les caractères significatifs (Lib n° 68, 69, 71 ... 77 ...).

- Dans les libellés longs dont un mot habituellement considéré comme significatif, se trouve négligé par la méthode manuelle. L'information choisie est alors concentrée au profit de mots plus significatifs (libellés n° 86, 83, 84, 85). Pour nous, mettre ces mots dans un antidictionnaire réservé aux libellés longs est possible, mais un peu coûteux.

Dans les *Résultats2*, nous constituons des dictionnaires pour améliorer l'adéquation au jeu manuel. Sous réserve de modifier les textes pour faire correspondre dans tous les libellés où ils sont présents TFI et TFD à "test fiscal imposition" et "test fiscal déduction", ce qui donne un début correct à 13 libellés sur 80, nous améliorons surtout la méthode en constituant un dictionnaire de "pivots" qui vient contredire la règle d'abréviation des libellés longs.

Dans ce cas, se pose la question des mots en même position, mais abrégés très fortement dans un texte long et non dans un texte court comme "élémentaire" en 18 à 23. Il semble qu'il manque une règle pour les textes de deux mots.

Il est évident que, sachant la règle "une lettre retenue par mot pour les textes de quatre à cinq mots ou plus", il est difficile d'identifier la signification de TFDEI [68] ou TFDPI [66], même si on ne les rencontre que dans des applications de comptabilité.

Néanmoins, l'équivalence est en général assurée au prix d'une saisie pas trop importante de dictionnaires.

*Résultat3* est simplement une amélioration légère de *Résultat2* par accroissement des dictionnaires. On peut se demander si le gain est suffisant pour faire ainsi grandir les dictionnaires, et augmenter le travail de l'utilisateur. Ceci serait à considérer dans un cas où le jeu d'essai est un sous-ensemble du jeu final.

Méthode 2 :

Ici, la situation est très différente de la précédente. Nous avons déjà émis des réserves avant même de tenter une imitation automatique. Les résultats sont convainquants dans leur inadéquation. Nous analysons les raisons de celle-ci, et proposons quelques améliorations, sous la forme d'autres méthodes jointes à leurs dictionnaires et de quelques recommandations à faire à l'utilisateur qui définit les libellés.

L'inadéquation dans l'imitation provient de quatre attitudes non automatisables de l'analyste qui a défini les abréviations manuelles. En annexe [A2] nous avons indiqué par A1, A2, A3, A4 les interventions de ces démarches dans l'abréviation manuelle.

A1 : Un libellé est traité deux fois et différemment, produisant deux abréviations différentes. L'analyste a négligé de compléter la sémantique du libellé et a ajouté implicitement un sens à l'abréviation. Notre démarche étant déterministe, il est logique d'exiger que le libellé soit explicite et que les libellés soient différents entre eux, au moins par une numérotation. Cependant, comme nous ne testons pas à la saisie la différence ou l'identité entre les textes, nous pouvons aussi appliquer une fonction générale de numérotation des synonymes puisque nous en proposons une.

La difficulté que nous avons rencontrée ici vient d'attitudes contradictoires entre deux analystes. L'un a défini l'identificateur par son contenu, l'autre a voulu différencier les noms pour une même notion parce qu'elle avait des rôles différents. Nous pensons que le système Abrège, en vérifiant l'unicité du nom pour un même libellé, devrait permettre d'éviter ce genre d'incohérence dans le système en obligeant à une différenciation explicite des libellés s'il y a lieu de la faire.

A2 : La deuxième attitude est plutôt inverse de la précédente. Un libellé très explicite comme "contrôle ordonnateur sur les bons d'engagement" est abrégé "contord1", alors que "Bons d'engagement" est plus souvent abrégé "BE" ou "BENG". Là, l'analyste a implicitement considéré un "contrôle de type 1" comme étant toujours sur les Bons. Mais cela nous paraît porter atteinte à la lisibilité de l'abréviation surtout lorsqu'on ne pénalise pas plus la longueur.

Nous avons aussi classé dans la même catégorie d'anomalie une autre attitude qui consiste à remplacer le terme employé dans le texte par un autre terme dans l'abréviation en faisant ainsi "glisser" le sens du texte. Par exemple

*Le texte "Diffusion des BE et des PSP à leurs destinataires" est abrégé DIFF1BUCO avec BUCO pour "bureau courrier", soit que l'on veuille dire que c'est le bureau courrier qui s'en charge, soit qu'il soit le destinataire.*

Il semble indispensable d'exiger de l'analyste qu'il unifie l'abréviation et le texte explicatif d'une notion. Nous avons analysé cette démarche sous le nom suppression-remplacement. Elle est à proscrire.

- A3 : L'anomalie répertoriée ici est la suivante. Un terme, utilisé tantôt en clair, tantôt en abrégé dans le texte du libellé, est ensuite abrégé de diverses manières dans le nom de la notion. L'exemple type est fourni par le groupe de mots "Bons d'engagement" qui est systématiquement abrégé "BE" dans les textes des libellés, mais qui, dans l'abréviation, apparaît sous la forme BE et sous la forme BENG. Il en est de même pour "titre de créance" et TC, administrateur de crédit et ADMCREDIT et AC ... Ce mode "d'abréviation-désabréviation" est remplacé dans la méthode par l'abréviation systématique du "groupe de mots" dans le texte même du libellé, et application dans la méthode d'un équivalent par dictionnaire.
- A4 : Enfin lorsqu'il abrège les textes qu'il propose, l'analyste supprime systématiquement certaines qualifications. Plutôt que de créer un antidictionnaire spécial, nous donnons comme équivalents de certains qualificatifs la chaîne vide dans le dictionnaire spécifique à ce jeu d'essai. C'est ainsi que nous supprimons exemplaires [58], duplicata [63] [76], original [75], qui qualifient l'exemplaire utilisé dans la notion. Cela revient à établir une hiérarchie dans les descriptions données dans les libellés et à négliger certains éléments intéressants pour le texte mais secondaires pour l'abréviation. Lors d'une création de méthode sur un jeu d'essai, ceci peut se faire dans un deuxième temps par extension de dictionnaires.

### 2.6.2. Evaluation par rapport aux objectifs. Autres méthodes

A la suite de cette analyse des anomalies ou des incohérences entre la méthode manuelle et la méthode automatique, nous voyons qu'il ne faut pas, dans les circonstances actuelles, demander une trop grande identité des résultats. Il nous paraît plus constructif de reprendre cet exemple et de considérer en quoi les résultats répondent à nos objectifs de lisibilité et de diversité et de les étudier aussi pour évaluer s'ils sont prononçables et mnémoniques.

Nous n'appliquons pas cette deuxième étude aux résultats de la première méthode, car nous avons vu qu'une contrainte très forte sur la longueur des abréviations ne permet pas de considérer d'autre critère d'évaluation que le nombre de collisions.

Par contre, cette étude nous conduit à différencier davantage les prédicats utilisés pour améliorer les résultats par rapport à nos objectifs.

Une première analyse des résultats montre que bien des libellés du jeu d'essai comportent quatre ou cinq mots significatifs. Or, la méthode entraîne pour bien des mots la prise en compte de trois ou quatre lettres. Pour des libellés de quatre mots significatifs, on arrive ainsi souvent à des abréviations de 12 caractères et plus dans 20 % des cas sur tous les libellés et 50 % sur ceux de plus de 4 mots. Or, il semble qu'une abréviation trop longue et non entrecoupée de séparateurs explicites comme le tiret ne soit pas facile à retenir et que son emploi soit lassant pour le programmeur à qui elle est imposée. Ainsi apparaît peut-être une limite supérieure de longueur, pour laquelle nous devons affiner la méthode en fonction de la longueur des textes de manière à ne pas provoquer de synonymies par troncature brutale au 12ème ou au plus au 14ème caractère. En fait, il semble bien que cette troncature ne provoquerait pas de synonymies-collisions sur le jeu d'essai que nous possédons. Mais en ne diversifiant pas les règles suivant la longueur, nous enlevons tout rôle sémantique aux derniers mots, sans aucune justification.

Une autre remarque est liée à la "prononçabilité" de nos abréviations. La politique de Méthode2 consiste à garder des consonnes comme fin d'abréviation des mots pris un à un et à éliminer les voyelles. Ceci nous conduit à des successions imprononçables de consonnes du type des identificateurs



REJTTITRCRE, RELOBTCERT ... où ces consonnes ne sont même pas séparées par des tirets, pour nous aider à les vocaliser.

Nous allons donc rechercher une méthode faisant une meilleure part aux voyelles. Peut-être pouvons-nous aussi faire précéder des abréviations spéciales, courtes, comme BE, DI pour DCE-DI (lui-même abréviation de "direction centrale ... "), AC, TC, par un tiret pour bien isoler ces pivots-clés qui sont inchangeables et pas toujours prononçables, utilisant ainsi comme abréviation -TC, -BE, -DI, etc ... Ceci est très simplement réalisable par les dictionnaires.

Nous n'avons pas grand chose à ajouter comme critiques à la Méthode<sup>2</sup> dans le contexte précis où nous l'avons définie. Nous proposons donc Méthode<sup>3</sup> que nous appliquons ensuite à deux jeux d'essai. Celui du jeu DM pour lequel elle a été pensée et celui d'un jeu proposé pour une autre application informatique (de gestion de Congrès) [A3].

Le contexte est donc différent pour ce qui est du vocabulaire, mais nous pouvons alors regarder quelle est la généralité et l'adéquation de la méthode testée en évaluant plus précisément les problèmes de saisie de dictionnaire.

### Méthode 3 : Définition en clair

Nous proposons les traitements plus différenciés suivants :

Ne travailler que sur les mots significatifs, déterminés par l'anti-dictionnaire standard. Limiter la longueur maximale à 14 caractères.

*Libellés d'un mot* : prendre son abréviation spéciale ou ses huit premières lettres.

*Libellé de 2 mots* : pour chaque mot, prendre les cinq premières lettres à condition que la 4<sup>ème</sup> et la 5<sup>ème</sup> ne soient pas toutes deux des consonnes, et seulement s'il n'y a pas d'abréviation spéciale.

*Libellé de 3 mots* : abréviation spéciale ou 4 lettres, sauf si la 3<sup>ème</sup> et la 4<sup>ème</sup> sont des consonnes. La 4<sup>ème</sup> est alors négligée.

*Libellé de 4 mots* : abréviation spéciale ou standard, ou sinon, pour le 1<sup>er</sup> mot, la même règle de 4 lettres que ci-dessus ; pour le 2<sup>ème</sup> mot, les 2 premières lettres et la 3<sup>ème</sup> si elle est nécessaire à une alternance de type avec la 1<sup>ère</sup> lettre du mot suivant ; pour le 3<sup>ème</sup> et le 4<sup>ème</sup>, 2 lettres seulement, en gardant la 3<sup>ème</sup> pour l'euphonie

*Libellés de 5 mots* : prendre les 3 premières lettres du premier mot, s'il  
*ou plus* n'a pas d'abréviation spéciale. Pour les autres  
mots, prendre les abréviations ou 2 ou 3 lettres en  
assurant l'alternance début-fin entre 2 mots.

Supprimer les collisions par numérotation successive de celles-ci à la  
fin du travail.

La méthode ainsi définie est améliorée par divers éléments contenus dans le  
dictionnaire spécial au jeu d'essai.

L'abréviation de "BE", bon d'engagement, est dans le dictionnaire précédé  
d'un tiret, ainsi que pour quelques autres abréviations spéciales.

Nous avons un peu remanié les textes du jeu d'essai dans une optique  
d'abréviation automatique de ceux-ci par références aux remarques faites en  
2.5.2. (Annexe [A2]).

Remarquons que certains adjectifs peu signifiants ont été stockés dans le  
dictionnaire avec une abréviation équivalente vide. Cela évite de les  
éliminer définitivement pour un autre jeu d'essai par intervention d'un  
antidictionnaire supplémentaire et ils interviennent ainsi dans le calcul de  
la longueur du texte sans intervenir dans l'abréviation.

Traduisons la méthode, puis discutons nos divers résultats.

### Méthode 3 :

options	:	doublets éliminés longueur maximum de l'abréviation = 14 collisions numérotées.
antidictionnaires	:	antidic
filtre-libellé	:	longueur-libellé = 1
filtre-mot	:	
dictionnaire	:	spécial, standic
calcul : filtre-lettre	:	position-lettre $\leq$ 8
action	:	prendre
filtre-libellé	:	longueur-libellé = 2
filtre-mot	:	
dictionnaire	:	spécial, standic

calcul : filtre-lettre : position-lettre  $\leq$  3  
action : prendre  
filtre-lettre : position-lettre = 4  
action : prendre sans déplacement  
type consonne, compteur  $\leftarrow$  1  
prendre (sinon)  
type consonne, compteur  $\leftarrow$  0  
  
filtre-lettre : position-lettre = 5  
type-lettre = consonne  
compteur = 0  
  
action : prendre  
filtre-lettre : position-lettre = 5  
action : prendre  
compteur  $\leftarrow$  0  
  
filtre-libellé : longueur-libellé = 3  
filtre-mot :  
dictionnaire : spécial  
calcul : filtre-lettre : position-lettre  $<$  3  
action : prendre  
filtre-lettre : position-lettre = 3  
type-lettre = consonne  
  
action : prendre  
augmenter compteur 1  
  
filtre-lettre : position-lettre = 4  
type-lettre = consonne  
compteur = 1  
  
action : compteur  $\leftarrow$  0  
filtre-lettre : position-lettre = 4  
action : prendre  
compteur  $\leftarrow$  0  
  
filtre-libellé : longueur-libellé = 4  
filtre-mot : position-mot = 1  
dictionnaire : spécial  
calcul : filtre-lettre : position-lettre  $<$  3  
action : prendre  
filtre-lettre : position-lettre = 3  
type-lettre = consonne  
  
action : prendre  
augmenter compteur 1  
  
filtre-lettre : position-lettre = 3  
action : prendre  
compteur  $\leftarrow$  0  
  
filtre-lettre : position-lettre = 4  
type-lettre = consonne  
compteur = 1  
  
action : compteur  $\leftarrow$  0

filtre-lettre	: position-lettre = 4
action	: prendre compteur ← 0
filtre-mot	:
dictionnaire	: spécial
calcul : filtre-lettre	: position ≤ 2
action	: prendre
filtre-lettre	: position = 3
action	: fonction locale : euphonie
filtre-libellé	: longueur-libellé ≥ 5
filtre-mot	: position = 1
dictionnaire	: spécial
calcul : filtre-lettre	: position-lettre ≤ 3
action	: prendre
filtre-mot	: position > 1
dictionnaire	: spécial
calcul : filtre-lettre	: position-lettre ≤ 2
action	: prendre
filtre-lettre	: position-lettre = 3
action	: fonction locale : euphonie

Nous remarquons tout de suite la lourdeur de cette écriture qui est due en partie aux explications un peu détaillées que nous donnons. On peut juger de la forme définitive en se reportant à la traduction Abrège donnée au Chapitre 6, § 6.2.3.

Par ailleurs, la répétition de certaines successions de filtres-lettres pour des calculs identiques mais liés à des prédicats antérieurs différents ne peut être actuellement évitée, même lors de l'écriture dans le langage qui réalise l'interface entre Abrège et l'utilisateur. Il faudrait simplement pour cela pouvoir nommer le filtre-lettre et le calcul dont fait partie ce prédicat, et le rappeler alors par son nom au lieu de réécrire toute la description.

Etudions maintenant les résultats de la méthode sur le premier jeu d'essai, celui de départ DM. (Annexe [A2]).

La méthode choisie nous assure des abréviations ayant une longueur moyenne de 10 caractères, et ne présente pas de collisions au sens du terme où deux libellés distincts donneraient la même abréviation.

En limitant la longueur à 14 caractères, on se donne la possibilité de numérotter les collisions sans troncature de l'abréviation et aussi d'accepter les abréviations spéciales précédées d'un tiret qui sont un peu plus longues que les deux caractères choisis, dans la plupart des cas, sur le jeu d'essai DM.

Le nombre de troncatures imposées par la limitation de longueur à 14 caractères est très faible (3 % dans l'exemple). Nous pouvons à ce propos faire une remarque. Les cas de troncature ont eu lieu évidemment dans des libellés longs et pour des textes où le nombre d'abréviations spéciales précédées de tirets est élevé.

Les alternances choisies entre les voyelles et les consonnes donnent en moyenne des "mots" assez lisibles, et nous pouvons constater que l'introduction du tiret devant certaines abréviations spéciales très condensées provoquent une césure qui aide beaucoup à la lisibilité du texte.

Avec 10 à 12 caractères par identificateur, nous pensons que la mémorisation ne sera pas immédiate, mais la connaissance de la signification du mot, de son mode de "construction" allié à la lisibilité devraient permettre une reproduction fidèle et un emploi assez facile.

Il nous faudrait pour en dire plus l'expérience des utilisateurs sur une assez longue période, ce que nous n'avons encore pu obtenir.

Il semble important de souligner le rôle que jouent les dictionnaires dans cet exemple. Ils contribuent en effet très largement à la qualité du résultat en fournissant des "points fixes" aux utilisateurs sur des éléments répétitifs qui auraient, sinon, considérablement augmenté la longueur des textes initiaux.

Le vocabulaire utile pour un tel jeu est finalement assez réduit et surtout très spécifique. Le dictionnaire apporte un élément supplémentaire de spécificité des termes que l'utilisateur peut exprimer lui-même indépendamment de toute fréquence.

Nous avons voulu dans l'exemple suivant étudier une nouvelle fois cet aspect. C'est la raison pour laquelle, repartant d'une méthode plutôt satisfaisante dans l'exemple ci-dessus, nous l'avons exécutée en donnant une attention minimale aux dictionnaires de manière à étudier un résultat assez brut. Nous le retravaillons ensuite simplement grâce aux dictionnaires.

Ce jeu d'essai (Annexe [A3]) nouveau est plus vaste que le précédent. Il comporte toutes les notions, environ 200, du système de gestion de Congrès mis au point par Grenoble-Accueil. Le vocabulaire est très disjoint du vocabulaire précédent, les textes sont relativement longs avec beaucoup de libellés comportant quatre à cinq mots significatifs. Nous avons réalisé des essais en deux temps, avec très peu de dictionnaires autres que l'antidictionnaire, puis en augmentant le dictionnaire spécial et en utilisant à nouveau des abréviations précédées de tirets.

Considérant les résultats de la méthode 3 en Annexe [A3] sur le jeu d'essai des notions du système de gestion des congrès de Grenoble-Accueil, nous pouvons d'abord facilement constater que toutes les abréviations sont différentes. Nous avons un premier résultat obtenu en utilisant un minimum de mots dans des dictionnaires. Nous constatons ainsi que la méthode donne des abréviations de longueur à peu près homogène de dix caractères.

Ces abréviations sont en fait assez prononçables en général, mais elles ont un air d'étrangeté qui est un peu gênant pour la mémorisation. Essayons d'en déterminer l'origine.

- 1 - Il semble que la méthode fasse apparaître un certain nombre de séquences de caractères inhabituelles en français qui contribuent à l'étrangeté des résultats.
- 2 - La décision que nous avons prise dans cet exemple de faire un usage réduit de dictionnaires spéciaux empêche de trouver des "points fixes" dans les abréviations par reconnaissance d'éléments inchangés, les abréviations données de certains mots.

- 3 - Dans les règles appliquées au libellé de 4 à 5 mots, nous tronquons les mots très tôt (2 lettres), ce qui nous amène à rencontrer l'abréviation "con" ou même "co" pour congrès, congressiste ... Le contenu du texte abrégé est alors difficilement décodable.

Nous reprenons donc notre exemple en acceptant un allongement de l'abréviation pour employer tout d'abord plus de dictionnaires. Il faut au préalable effectuer un travail de sélection des mots qui vont posséder une abréviation spéciale et supposer que ces choix sont assez bons pour que la méthode associée à ses dictionnaires reste correcte même sur d'autres éléments du même système. Il est à noter que la modification du contenu des dictionnaires étant indépendante de la définition de la méthode, l'adaptation du système "par les dictionnaires" en est facilitée, comme nous le verrons mieux au Chapitre 6.

Revenons maintenant à l'étrangeté des mots créés. Elle provient de suites de lettres que nous savons vocaliser (prononciation) mais qui, telles quelles, sont inhabituelles en français du fait d'une coupure abrupte sur la position de la lettre sans tenir compte de ce qui habituellement était l'expression d'un phonème avec éventuellement des lettres muettes que nous sommes habitués à rencontrer. Cet aspect phonétique est sensible particulièrement lorsque nous rencontrons des dyptongues que nous coupons comme "oi", "ou"

Un autre aspect qui est plus évident encore est la multitude d' "anomalies orthographiques" que nous rencontrons en fin de mot. Ces "anomalies orthographiques" sont des séquences de lettres anormales en français, c'est-à-dire peu fréquentes ou inconnues bien que prononçables. Cela vient peut-être de ce que la méthode ne tient jamais compte des dernières lettres des mots, mais coupe au milieu de ceux-ci. Il faudrait essayer d'autres méthodes tenant compte de ces remarques. Notre propos n'est cependant pas de multiplier les exemples. Les prédicats et les actions que nous avons proposés permettraient d'utiliser les dernières lettres comme nous avons utilisé les premières.

Nous voudrions maintenant revenir encore une fois sur nos objectifs initiaux.

Nous n'avons pas, dans l'exemple ci-dessus, utilisé la longueur des mots. La mise au pluriel de certains mots ne nous pose donc de problème que lorsque nous créons l'abréviation de "résidence" ou "inscription" dans le dictionnaire spécial pour nous apercevoir quelques lignes plus loin que ces mots sont employés au pluriel. Nous sommes obligés dans le dictionnaire de stocker les 2 mots au singulier comme au pluriel pour assurer une seule abréviation.

Par ailleurs, nous pouvons aussi remarquer que les mots que nous remplaçons par une abréviation vide dans le dictionnaire spécial sont souvent des adjectifs : la plupart du temps, nous ne gardons pas ceux-ci. Pour reconnaître et traiter automatiquement ces deux "maladresses" du système actuel, il nous faudrait disposer de caractéristiques des mots issues d'une analyse morphologique et syntaxique. Nous verrons au Chapitre 4 ce qu'il faut en penser.

Nous avons donc dressé un premier bilan. Dans le contexte de libellés définissant des identificateurs pour des applications informatiques du type de celles que nous avons présentées, nous arrivons à définir des méthodes à peu près satisfaisantes quant à la prononçabilité.

Les résultats obtenus ne sont pas encore très mnémoniques. Il semble que la seule alternance consonne-voyelle qui aide à la prononciation ne soit pas suffisante pour assurer une bonne "reconnaissance" des mots composant l'abréviation. Parce que plus qu'un apprentissage "par coeur" des abréviations, il nous semble que la mémorisation se fera par codage-décodage.

C'est dans cet esprit que nous avons voulu étudier quels résultats donnerait l'emploi de propriétés plus phonétiques comme la décomposition en phonèmes et même en syllabes que nous proposons au chapitre suivant.



## CONCLUSION

Avant de clore ce chapitre, nous voudrions consacrer un peu plus de temps à un objectif fonctionnel que nous avons signalé au Chapitre 1 et dont nous n'avons pas reparlé jusqu'à maintenant. Il s'agit d'abréger des textes non pour obtenir un identificateur mais pour conserver la structure de phrase en abrégeant mot à mot et en gardant donc à chaque mot son identité sémantique.

Dans ce cas, le problème de longueur de l'abréviation qui était jusqu'à maintenant une de nos principales préoccupations ne se pose plus de la même façon. Ce qui compte ici c'est que chaque mot soit identifiable. Nous ne présentons pas ici d'exemple où nous prenons un nombre donné de lettres en fonction de la longueur du mot, des caractéristiques des lettres rencontrées en balayant le mot (consonne ou voyelle).

Il nous semble, en effet, encore plus que précédemment que la prise en considération des syllabes, des phonèmes et si nous le pouvons du rôle syntaxique et de la morphologie, s'impose.

La syntaxe de titres de livres par exemple est souvent élémentaire et proche de celle de nos libellés de notion. Mais ce n'est plus vrai dans des textes de petites annonces pour lesquels cependant ce système semblerait utilisable. Les catalogues eux aussi ont une structure textuelle proche des exemples qui ont été traités jusqu'à maintenant, mais sur un vocabulaire souvent plus large. Il ne faut pas alors que l'emploi des dictionnaires devienne une contrainte irréalisable pour un utilisateur. Il faut donc supprimer les difficultés signalées au-dessus qui obligent à introduire des singuliers et des pluriels dans les dictionnaires.

Pour reconnaître les mots traités un à un, il paraît indispensable de tenir compte de leur décomposition syllabique. On ne peut donc prévoir la position de la césure de manière fixe, car, par exemple, un mot de 5 lettres peut comporter deux ou une syllabe : pains, beaux, parti, envie ... et si l'on décide de ne prendre qu'une syllabe, la chaîne retenue variera beaucoup en longueur.

Le jeu d'essai "Congrès" se prêterait assez bien par la nature de ses textes à un tel mode d'abréviation pour nous fournir un exemple simple. Mais nous attendons d'avoir élargi notre système de prédicats pour traiter un exemple de ce type.

### CHAPITRE 3

LA DEMARCHE PHONETIQUE ET SYLLABIQUE

### 3.1. NECESSITE D'UNE ANALYSE PHONETIQUE ET SYLLABIQUE

Nous avons vu dans le chapitre précédent que l'alternance voyelle-consonne ne donne pas une trace des mots suffisamment claire pour que l'abréviation obtenue soit facilement mémorisable.

Par ailleurs, lorsque nous ne voulons plus former un identificateur unique à partir du texte du libellé, mais conserver la structure de texte par abréviation mot à mot, les mots doivent alors être reconnaissables individuellement dans leur abréviation. Il semble, dans ce cas, que la plupart des mots sont abrégés par conservation d'un certain nombre de syllabes ; c'est du moins l'impression que l'on a en lisant des catalogues ou des petites annonces.

*Ainsi, on peut lire : "pers" pour personne, "affect." pour affectueux, "enf." pour enfant, etc ....*

Bien souvent, ce qui remplace le mot est une partie tronquée sur la première ou la deuxième syllabe, suivant la longueur de celui-ci.

Cette première impression est fondée sur un concept de syllabe très approximatif, un peu celui qui permet de découper en 12 'pieds' les alexandrins des auteurs classiques.

La conclusion que l'on peut en tirer est que cette troncature ne peut pas être définie par la seule position des caractères. La deuxième est la certitude que les abréviations spontanées sont de nature 'acoustique', respectant un certain nombre de sons du mot ou de nature 'morphologique' ne conservant que la racine de celui-ci.

Tentons maintenant de définir précisément les termes de phonèmes et syllabes. Les "*phonèmes*" sont les sons simples significatifs que l'on peut isoler dans une langue donnée. Ils permettent d'identifier la chaîne parlée par opposition ou distinction des sons entre eux. Notre problème est d'associer un son déterminé à une chaîne de caractères ou graphème, sachant que, suivant le contexte, les caractères ne sont pas prononcés de la même façon, certains d'entre eux pouvant même s'amuir /\* e, s, r \*/. Nous référant à l'ouvrage d'A. Martinet "*Eléments de Linguistique Générale*" [M1], nous pouvons constater que cette correspondance n'est pas facile à définir, qu'elle comporte des règles de base aussitôt infirmées par toutes sortes d'exceptions.

Par ailleurs, la langue française comporte de nombreux phonèmes (38 environ) et l'analyse des traductions d'un mot en phonèmes varie suivant le contexte d'emploi du mot ou la personne qui le prononce.

*Par exemple : dans "les animaux" et "les troupeaux", le s à la fin de "les" n'est pas prononcé lorsqu'aucune liaison ne s'impose.*

Nous savons que l'analyse et la correspondance grapho-phonémique que nous voulons établir n'a pas besoin d'être très fine et d'entrer dans toutes les subtilités de la prononciation. Mais il nous a paru plus simple de rechercher l'emploi de solutions existantes plutôt que de nous plonger dans l'étude de cette traduction, à partir des multiples informations du livre d'A. Martinet.

Les *syllabes* ont une définition bien moins précise que les phonèmes car elles ne sont pas des unités simples, mais elles sont définies comme "l'ensemble des phonèmes qui, dans un contexte déterminé, peuvent se prononcer d'une seule émission de voix". Il nous paraît très souhaitable de disposer d'un outil capable de découper la chaîne phonétique en syllabes. Etant donné les conditions dans lesquelles nous voulons l'employer -chaque mot étant traité en dehors de tout contexte- nous recherchons une solution simple et pratique, même si elle est phonétiquement élémentaire.

### 3.2. RECHERCHE ET CHOIX D'UN OUTIL POUR LA TRADUCTION GRAPHO-PHONÉMIQUE

Avant de rechercher un outil réalisant la traduction grapho-phonémique, nous devons remarquer que celle-ci est intimement liée à la langue employée. Dans notre cas, nous nous intéressons évidemment au français, mais ceci est une limite linguistique de notre système. En particulier, nous ne pourrions pas utiliser cette analyse pour l'abréviation des "Titres de périodiques analysés au bulletin signalétique du CNRS", alors que le problème qui se pose à ceux qui réalisent ce document est précisément un de ceux que nous aurions aimé résoudre. Les "Titres" sont en effet écrits dans des langues très diverses et le mode d'abréviation est à la fois phonétique et morphologique.

Parmi les différentes possibilités que nous avons recensées, nous avons retenu celles présentées par J. Vaissière [V1], par notre collègue Y. Chiaramella [C1], celle enfin de Coulon et Kayser [CK1]. Essayons de situer ces propositions pour expliquer notre choix.

La thèse de J. Vaissière "Contribution à la synthèse par règles du français" présente en deuxième partie une étude approfondie de ce problème de traduction grapho-phonémique. L'auteur cherche, en effet, à traduire un texte écrit sous forme phonique dans le but de faire prononcer le texte par vocodeur. L'ensemble des règles et cas particuliers de la prononciation du français y sont présentés. Nous ne jugeons pas utile de reproduire ici cette étude. Nous voulons seulement exposer ici les raisons pour lesquelles nous n'utilisons pas les résultats qu'elle propose.

Après étude du problème de traduction grapho-phonémique, J. Vaissière a décidé, parce qu'elle avait en outre besoin de l'analyse morphologique de textes, de faire correspondre à chaque mot du vocabulaire sa traduction phonémique dans un dictionnaire. Elle utilise pour cela le dictionnaire de base de son analyseur morphologique dans lequel tout mot doit être indexé pour être reconnu. Or, cet emploi de dictionnaire ne convient pas à un système comme Abrège où les vocabulaires des divers utilisateurs ne sont pas prévisibles. Nous ne pouvons par avance créer ce dictionnaire, et nous ne pouvons pas non plus demander ce travail à l'utilisateur, à qui nous voulons fournir des abréviations. Ce serait contraire à l'objectif de simplicité de l'interface entre Abrège et ses usagers.

En outre, l'analyseur morphologique du français, qui nous était le plus accessible [C2] et dont nous reparlons au chapitre suivant, est un logiciel PIAF réalisé sous CP-CMS sur le 360-65 du CIG. Notre prototype, lui, est réalisé en Socrate que nous avons choisi (Chap. 1, § 3) pour sa transportabilité entre IRIS 50 et l'IBM 360 sous OS, et pour l'aide efficace qu'un logiciel pour bases de données offre à la création de prototypes. Les deux systèmes 360-OS et 360-CP-CMS ne peuvent communiquer que par échange de fichier-données sur bande (si l'on fait abstraction des toutes récentes possibilités du réseau Cyclade). La contrainte d'un dictionnaire complet du vocabulaire et les difficultés de communication entre les systèmes auraient beaucoup pénalisé le prototype d'Abrège à l'exécution. Nous avons donc cherché d'autres solutions.

Dans sa thèse, J. Vaissière signale (Partie II - Introduction - B) que deux méthodes peuvent être utilisées pour convertir la suite de graphèmes en phonèmes. La *méthode du dictionnaire*, qui donne pour chaque mot du texte sa traduction phonémique, est la méthode qu'elle a utilisée. L'autre méthode est celle d'un programme de conversion grapho-phonémique, dit *programme orthoépique* qui reconnaît la suite des "graphèmes" et les convertit selon certaines règles en suites de phonèmes. Nous nous sommes orientés vers un tel logiciel pour simplifier, comme nous le signalions ci-dessus, la tâche de l'utilisateur d'Abrège.

Avant de présenter le logiciel que nous avons retenu, signalons que, pour résoudre ce même problème de conversion, un de nos collègues, Y. Chiaramella a (dans le même temps où nous adoptons une solution sous Socrate) réalisé un "transducteur de conversion ortho-épique" sous le système PIAF mentionné ci-dessus. C'est le programme PIAFPHO. Il pourrait être intéressant de l'utiliser dans un contexte de réalisation où nous voudrions aussi disposer de l'analyse morphologique du français puisque les deux outils sont compatibles. Mais les incompatibilités avec Socrate demeurent actuellement.

Finalement, nous avons retenu la solution proposée par D. Coulon et D. Kayser dans "Analyse de réponses rédigées en français courant pour une réalisation d'Enseignement programmé" [CK1] .

Dans le projet d'enseignement assisté par ordinateur fonctionnant en langage naturel qu'ils ont défini et qui était déjà opérationnel en 1972, les auteurs ont été confrontés au problème suivant : éliminer dans les réponses des étudiants une partie des anomalies orthographiques pour reconnaître des phrases sensées, mais mal orthographiées. Pour aboutir à ce résultat, ils ont décidé d'utiliser un codage phonétique de la phrase alors reconnue comme suite de ses phonèmes [K1]. Pour cela, ils analysent donc chaque texte, caractère par caractère, pour le traduire dans la chaîne des phonèmes équivalents.

Des hypothèses simplificatrices réduisent à 26 le nombre des phonèmes "reconnus" et permettent de confondre en un seul certains phonèmes très différents, comme s et x, ai et aille, etc ... [CK1] [K1].

Nous avons constaté que l'analyseur ainsi obtenu est bien suffisant pour nous qui cherchons essentiellement à découper la chaîne des caractères du mot en groupes de lettres décrivant un même son, sans que la différenciation fine des phonèmes ne nous soit nécessaire (comme elle l'est dans le cas de J. Vaissière).

Pour nous, cet outil est donc directement utilisable et fournit un automate complètement spécifié. Nous devons toutefois traduire les programmes originels (écrits pour NCR4130 en langage d'assemblage) en Socrate, sous forme de la base dont nous discutons l'implantation au paragraphe suivant.

### 3.3. IMPLANTATION DE L'ANALYSE PHONETIQUE

#### 3.3.1. L'automate. Son implantation

La thèse de D. Kayser [K1] et l'article précédemment cité de la RAIRO [CK1] donnent les détails d'implantation de l'automate phonétique tel qu'il fut réalisé par D. Coulon et D. Kayser. Dans la mesure où nous ne pouvions pas nous en servir tel quel, nous avons surtout utilisé comme document de travail la table des phonèmes et celle de l'automate présentées en pages 74 à 78 de l'article cité. Nous les avons reproduites en annexe, jugeant inutile de reproduire ici des résultats que nous n'avons pas modifiés.

Nous expliquons le mode d'utilisation de cette table d'automate pour l'analyse phonétique et l'implantation Socrate que nous avons réalisées.

a) Dans la table de l'automate correspond, à chaque ligne, l'état dans lequel se trouve l'automate à un instant donné. Cet état est défini par la chaîne de caractères reconnue mais non encore associée à un phonème. A chaque colonne correspond le caractère lu par l'automate. Dans chaque case, à l'intersection d'une ligne et d'une colonne, se trouve la règle de transition de l'automate donnant le nouvel état, l'action de progression dans la chaîne d'entrée et l'écriture éventuelle sur la chaîne de sortie (phonème). Il y a 38 états (c'est-à-dire 38 lignes dans la table), 27 caractères (auxquels correspondent 18 colonnes seulement, voir exemple ci-dessous) et 26 phonèmes de sortie. En fait, le triplet (nouvel état, phonème de sortie, progression de l'index) définissant la transition de l'automate, n'est pas toujours explicité dans la table :

- le nouvel état est implicite lorsqu'il est défini par le seul caractère d'entrée repéré par l'index de position sur la chaîne d'entrée

- le phonème de sortie n'est défini que lorsqu'il est effectivement reconnu

- la progression de l'index n'est pas indiquée lorsqu'elle consiste simplement en un déplacement d'une position vers la droite dans la chaîne d'entrée.

Exemple : La ligne donnant les transformations de l'automate pour l'état courant A est la suivante :

	␣	a	b f j v w	d g p q	c	e	h	i y	k	l	m	n	o	r	s x	t	u	z
A	FIN	A						AI			AN	AN						
	a		a	a	a	a	a		a	a			a	a	a	a	o	a
		+	*	*			+	+			*	*					+	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18

Dans les colonnes 1, 4, 5, ..., le phonème a est définitivement reconnu (inscrit sur la chaîne de sortie), de même que o en colonne 17.

L'action + correspond à la lecture des 2 caractères suivants sur la chaîne d'entrée. Le premier sert à définir le nouvel état, le deuxième détermine le caractère courant.

Dans certains cas (colonnes 8, 11, 12), le nouvel état est obtenu par concaténation des caractères de l'état courant avec la lettre courante, ou par prise en compte d'une seule parmi deux lettres A en colonne 2).

Certaines lettres (f, m, n, p, ...) peuvent être gémées. N'en tenant pas compte au niveau de la différenciation phonétique, nous testons cependant cette situation qui provoque éventuellement l'avancement d'un caractère de plus en entrée (action \*).

Lorsque le nouvel état n'est pas précisé, il est constitué par le caractère lu pris indépendamment de celui qui l'a précédé, et l'action d'avancement donne le nouveau caractère lu.

Certains caractères sont regroupés en une seule colonne (3, 4) : b, f, j ... Ceci permet de condenser l'écriture de l'automate quand plusieurs lettres provoquent les mêmes reconnaissances de phonème. Lorsque l'état suivant n'est pas explicite comme pour P, G, D, Q, on passe dans l'état global "/" (3ème ligne de l'automate) où est testé le redoublement du caractère. Celui-ci est ensuite reconnu, comme donnant dans tout voisinage et sans ambiguïté le phonème de sa "lettre". Il y a, dans ces cas, correspondance triviale :



*graphème (simple ou géminé) - phonème.*

b) Pour l'implémentation sous Socrate, nous avons choisi une structure reflétant ce tableau, mais qui traduit économiquement le contenu des transitions : puisque le nouvel état est la plupart du temps déduit du caractère lu en entrée et non explicité dans le triplet, nous pouvons choisir une expression du tableau "par les exceptions" qui conduit à un très petit nombre de contenus de case différents.

Pour cela, nous enregistrons ce contenu dans l'entité PH-CONF. Chacune des réalisations de cette entité détermine la transition de l'automate lorsqu'on est dans une configuration qui admet cette réalisation comme contenu de case.

La structure est complétée par :

- 38 réalisations de l'entité PH-ETAT dont chacune correspond à une ligne de la table
- 26 réalisations de l'entité T-PHON dont chacune correspond à un phonème de sortie
- 27 réalisations de l'entité ALPHA dont chacune correspond à un caractère possible de la chaîne d'entrée.

Chaque réalisation de PH-ETAT comprend la chaîne d'entrée définissant l'état (qui sert de clé de recherche de l'état) et les 18 références aux contenus des cases de la ligne par le numéro correspondant de réalisation de PH-CONF.

Les réalisations de T-PHON et ALPHA contiennent, outre les identifications qui permettent de les retrouver, des caractéristiques intrinsèques que nous utiliserons dans les méthodes d'abréviation (type, fréquence générale ...) D'où les éléments de la structure Socrate ci-dessous :

```
entité 38 PH-ETAT      /* Ligne de l'automate
début
  PH-CHAINE mot (3)    /* Chaîne d'entrée définissant l'état */
entité 24 LET-CONF     /* Une case par colonne
début
  MEM-CONF de 1 à 100 /* Numéro du contenu de la case */
fin
fin
```

entité 100 PH-CONF

début

NV-PH-ETAT mot (3) /\* Nouvel état, indéfini s'il est standard \*/

NV-PHON de 1 à 30 /\* Numéro dans T-PHON du phonème reconnu \*/

PH-ACT (6 2) (L1 L2 R1 R2 NE SO) /\* Action à effectuer \*/

fin

entité 27 T-PHON

début

N-PH mot (2) /\* Graphème explicitant le phonème \*/

TYP-PH mot (3) /\* SOU LAB DEN ... \*/ /\* Type du phonème \*/

FREQ-PH de 1 à 10000 /\* Nombre d'occurrences du phonème dans le jeu d'essai \*/

CODE-TAB-SYLL de 2 à 4 /\* 2=VO 3=SOURD 4=EXPLOSIVE \*/

fin /\* Codification pour l'analyseur syllabique. Numéro de colonne \*/

entité 27 ALPHA /\* Alphabet assurant la correspondance \*/

/\* Lettre-colonne de la table phonétique \*/

début /\* Le stockage des caractéristiques des lettres \*/

LETTRE mot (1) Avec clé unique ordonnée fin /\* Recherche alphabétique \*/

TYPE (2 1) (V C) /\* Consonne-voyelle \*/

PRONONC (5 2) (SR SO RA MU ...) /\* Prononciation \*/

FR-D-R de 1 à 1000 /\* Fréquence dans le langage \*/ /\* 1000 pour exprimer un % \*/

FR-G-R de 1 à 10000 /\* Fréquence dans le jeu d'essai \*/

CODE-TAB-PHON de 1 à 24 /\* Correspondance colonne de l'automate phonétique \*/

fin

Reprenons l'exemple de la ligne A.

*La réalisation correspondante de PH-ETAT serait la suivante :*

PH-ETAT (2) /\* A est la deuxième ligne de la table \*/

PH-CHAINE : A

LET-CONF(1) MEM-CONF : 19

LET-CONF(2) MEM-CONF : 2

(3) : 14

(4) : 14

(5) : 20

(6) : 20

(7) : 15

(8) : 16

<i>LET-CONF</i> (9)	<i>MEM-CONF</i> : 20
(10)	: 20
(11)	: 17
(12)	: 17
(13)	: 20
(14)	: 20
(15)	: 20
(16)	: 20
(17)	: 18
(18)	: 20

*PH-CONF*(14)

<i>NV-PH-ETAT</i>	: U	
<i>NV-PHON</i>	: 17	/* Numéro du phonème T-PHON : a */
<i>PH-ACT</i>	: 'NE'	/* Redoublement */

*PH-CONF*(16)

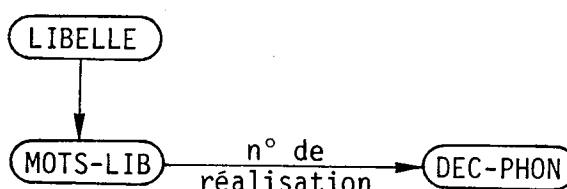
<i>NV-PH-ETAT</i>	: 'AI'	
<i>NV-PHON</i>	: U	
<i>PH-ACT</i>	: L2	/* Lecture de 2 caractères */
	:	

L'algorithme d'utilisation de cette table d'automate est le suivant :

```
si NV-PH-ETAT de Ph-CONF = U
alors
  NV-ETAT ← CAR-COUR (MOT)
sinon
  NV-ETAT ← NV-PH-ETAT de PH-CONF
finsi
si NV-PHON ≠ U
alors
  Ajouter dans DEC-PHON du mot courant NV-PHON de PH-CONF
finsi
Exécuter ACTION (PH-ACT de DEC-PHON)
```

### 3.3.2. Stockage des résultats

Les résultats de l'analyse phonétique sont constitués par les *décompositions phonétiques* respectives de chacun des mots d'un ensemble de libellés à abrégé. Cette décomposition (suite des phonèmes équivalente au mot) est une donnée statique, mais dont nous pouvons avoir besoin plusieurs fois pour un ensemble donné, soit que le même mot apparaisse plusieurs fois dans l'ensemble, soit que plusieurs méthodes soient essayées sur le même ensemble. C'est pourquoi, nous avons choisi de stocker ces résultats dans la structure même de l'ensemble des libellés. Ainsi, l'analyse phonétique est effectuée la première fois qu'une méthode fait appel à des propriétés phonétiques sur les libellés donnés. La présence dans la base des résultats de cette analyse est alors mémorisée de telle sorte que l'analyse ne soit pas effectuée à nouveau lors d'un autre essai sur les mêmes libellés. Les décompositions phonétiques sont rattachées aux "MOTS-LIB" de la structure Abrège par une référence dans l'entité MOTS-LIB donnant le numéro de réalisation de l'entité DEC-PHON qui correspond à la décomposition phonétique du MOT-LIB.



L'entité DEC-PHON est la suivante :

entité 5000 DEC-PHON

début

entité 20 SUITE-PHON /\* Suite des phonèmes du mot \*/

début

NO-PHON de 1 à 26 Numéro du phonème dans la table des phonèmes

NB-LTR de 1 à 5 Longueur en lettres du phonème

FREQ-PHON de 1 à 10 Nombre d'occurrence de ce phonème dans le mot

fin

fin

Dans cette implantation, nous avons donc fait les hypothèses suivantes :

- un mot est composé au plus de 20 phonèmes
- il n'existe pas de phonème traduisant une chaîne de plus de 5 lettres (eaux = 0 nous paraît même un maximum)

- il n'existe pas de mot contenant plus de 10 fois le même phonème.

Remarques :

1) Nous conservons la *longueur du phonème* en lettres de manière à disposer effectivement de la décomposition de la chaîne de caractères du mot en phonèmes. Cette longueur est liée à l'orthographe du mot.

2) Nous calculons le nombre d'occurrences des phonèmes dans le mot de manière à pouvoir éventuellement tenir compte de la *fréquence du phonème relative au mot* et de celle du phonème dans le jeu d'essai.

3) Par ailleurs, dans les "données permanentes" de la base Abrège fournies avec l'initialisation de celle-ci, se trouve, pour chaque phonème, une *propriété de type* liée à sa prononciation.

La manière d'utiliser ces propriétés des décompositions phonétiques des mots dans une méthode est expliquée au § 5 de ce chapitre.

### 3.4. ANALYSE SYLLABIQUE. AUTOMATE. IMPLANTATION

L'analyse phonétique constitue le point de départ de l'analyse syllabique (rappelons que la syllabe est une suite de phonèmes qui peuvent se prononcer d'une seule émission de voix).

L'automate que nous utilisons est déduit de la définition suivante ; très réduite, vis-à-vis des diverses situations possibles [R1] :

"Deux consonnes différentes se séparent (ac-teur) (ex|'z'emple)  
mais que deux consonnes dont la deuxième est une liquide (l, r)  
forment un groupe explosif avant lequel se fait la division".

Nous en avons tenté une traduction moins contextuelle :

nous avons, pour respecter la règle de séparation des phonèmes consonnes, séparé ceux-ci en deux classes, les phonèmes consonnes sourds, CS, et les phonèmes consonnes explosifs, CE.

Les phonèmes sourds terminent une syllabe s'ils sont suivis d'un phonème-consonne. Les phonèmes explosifs provoquent la rupture de l'émission vocale, ils commencent une syllabe nouvelle.

Les consonnes sourdes sont :

t, k, s, ʃ, z, l, r, m, n, ñ

les phonèmes-consonnes explosifs :

b, p, d, f, v, g.

Nous en déduisons les règles d'analyse suivantes, en fonction du type de phonème rencontré :

	Ancien état	Phonème courant	Nouvel état	Actions reconnues
1	VO	VO	VO	
2	VO	CE	CE	SYLLABE
3	VO	CS	VO CS	
4	VO	fin		SYLLABE
5	VO CS	VO	CS	SYLLABE - ARRIERE1
6	VO CS	CE	CE	SYLLABE
7	VO CS	CS	CS	SYLLABE
8	VO CS	fin	fin	SYLLABE
9	CE	VO	VO	
10	CE	CE	CE	
11	CE	CS	CS	
12	CE	fin	fin	SYLLABE
13	CS	VO	VO	
14	CS	CE	CE	
15	CS	CS	CS	
16	CS	fin	fin	SYLLABE

Seule, la cinquième configuration de l'automate provoque un retour arrière dans le déplacement de gauche à droite dans la chaîne des phonétiques.

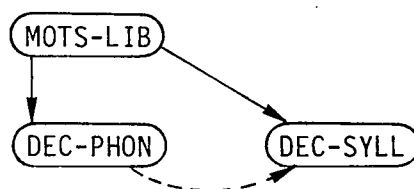
Nous ne donnons ici qu'un seul exemple. Quelques autres sont fournis en annexe [A8] comme exemples d'utilisation de cet analyseur.

Exemple : Mot            W H I S K Y ◡  
                   Phonèmes     v i s k i  
                   TYPES-PHON   CE   VO   CS   CS   VO  
 Configurations            8   3   7   12   4  
                   SYLLABES    WHIS            KY

L'implantation de cet automate est l'analogue de celle du précédent. Il y a 4 états et 12 configurations.

Remarques :

1) L'analyse syllabique fournit la *chaîne des phonèmes équivalents à la syllabe, le nombre de ceux-ci, la longueur de la syllabe en phonèmes et en caractères*. La même raison qui nous a incité à n'exécuter qu'un minimum de fois l'analyse phonétique nous conduit à profiter de la souplesse qu'offre un système de gestion de base de données, pour conserver ces propriétés de chaque syllabe des mots; des libellés donnés dans la structure de la base. D'où la structure :



entité 5000 DEC-SYL

début

entité 15 SUIT-SYLL

début

NB-PH de 1 à 5 /\* nombre de phonèmes \*/

NB-LTR de 1 à 10 /\* nombre de lettres de la syllabe \*/

fin

fin

Comme auparavant pour les décompositions phonétiques, nous avons ici évalué une approximation des maxima de longueur d'un mot en syllabe (15 SUIT-SYLL), d'une syllabe en phonème ( $NB-PHON \leq 5$ ) et d'une syllabe en lettres ( $NB-LTR \leq 10$ ).

2) La définition que nous avons donnée de la syllabe ne nous permet de mettre en évidence aucune propriété non triviale de celle-ci.

Nous ne définissons en particulier aucune relation d'équivalence sur l'ensemble des syllabes, ce qui exclut de tenir compte de la notion de fréquence et d'un type donné de syllabe.

3) Enfin, nous voudrions signaler une caractéristique d'implantation que nous avons adoptée pour disposer simultanément de l'accès à la décomposition phonétique et la décomposition syllabique d'un mot. Nous avons traduit le lien entre un MOTS-LIB et sa décomposition phonétique par le numéro de réalisation de l'entité DEC-PHON stocké en NO-DECPHON dans MOTS-LIB. Nous avons engendré la décomposition syllabique DEC-SYLL sur le même numéro de réalisation de telle sorte qu'ainsi, ce NO-DECPHON soit aussi un NO-DECSYLL (implicite).

### 3.5. EXTENSIONS DES METHODES D'ABREVIATIONS

Après avoir présenté les moyens par lesquels nous réalisons les analyses phonétiques et syllabiques, nous montrons comment peuvent être utilisées, dans les méthodes d'abréviations, les nouvelles caractéristiques des mots ou des parties de mots dégagées par ces analyses.

Ces propriétés définissent dans le mot des champs autres que la lettre, plus vastes, eux-mêmes décomposables. Ainsi, un mot n'est plus considéré seulement comme une suite de lettres, mais aussi comme suite de syllabes décomposables en phonèmes composés de lettres.

Nous définissons donc de nouveaux prédicats utilisables pour caractériser les mots et de nouveaux moyens de calculs, le champ d'action du calcul n'étant plus seulement donné par la position et le type de chaque caractère.

L'extension de la définition de méthodes d'abréviations se situe donc tout d'abord au niveau du *filtre-mot*. Et nous pouvons ensuite introduire des prédicats de syllabes ou *filtre-syllabe*, des *filtres-phonèmes* et étendre en conséquence le schéma des *calculs*.

#### 3.5.1. Extension du filtre-mot

Pour le *filtre-mot*, nous avons précédemment considéré qu'il était le résultat d'une intersection logique entre les prédicats suivants pris dans l'ordre de citation :

- prédicat de position du mot dans le libellé
- prédicat de fréquence du mot relativement à l'ensemble des libellés ou au langage
- prédicat de longueur du mot mesurée en caractères



- prédicat de compteur rendant compte d'actions antérieures.

En considérant ces prédicats d'une part et les propriétés nouvelles que nous avons mises en évidence d'autre part, nous voyons que seule la longueur peut être évaluée différemment

longueur du mot en syllabes  
                  en phonèmes  
                  en lettres.

Nous précisons donc l'unité que nous prenons en considération dans l'expression de ce prédicat. Un prédicat de longueur ne s'exprimera qu'avec une seule unité. Ceci exclut un prédicat faisant référence à plusieurs longueurs, comme par exemple :

*Mot de plus de 3 syllabes et de moins de 15 lettres.*

Un tel prédicat serait évidemment utilisable mais, en fait, nous pensons que lorsque nous nous intéressons au nombre de syllabes du mot (par exemple), le champ qui nous intéresse est l'unité prononçable "syllabe" et non une combinaison indistincte de toutes les propriétés.

Nous pourrions toujours réviser cette analyse lorsque notre expérience de fabrication de "bonnes méthodes" sera plus grande. Notre option fondamentale pour l'instant est de garder au raisonnement une certaine simplicité.

Le prédicat de *compteur* ne se trouve pas explicitement modifié. Mais il peut maintenant tenir compte d'actions nouvelles, de niveau syllabique par exemple.

### 3.5.2. Extensions des contextes de calcul

De la même manière que nous avons délimité précédemment le *champ d'une action de choix* par un *filtre-lettre*, nous pouvons maintenant le faire au niveau des phonèmes ou des syllabes par un *filtre-phonème* ou un *filtre-syllabe*, composés de prédicats caractérisant phonèmes ou syllabes.

A ce niveau, nous avons décidé d'imposer une *contrainte de simplicité* cohérente avec celle du prédicat de longueur d'un filtre-mot.

En II.5, nous avons montré qu'un calcul est composé d'actions dont le champ est défini par des prédicats caractérisant les lettres dans des filtres-lettres. Ici, nous imposons que tout calcul travaille sur des champs de même nature, soit syllabiques, soit phonétiques, soit littéraux, mais exclusivement les uns des autres.

Les filtres exprimés dans un calcul sont donc tous de même nature filtres-syllabes ou filtres-phonèmes ou filtres-lettres ; ceci n'exclut pas que les actions de choix isolent seulement quelques lettres d'une syllabe ou un phonème de celle-ci.

Filtre-syllabe - Il comporte seulement trois prédicats.

1) Prédicat de longueur : il exprime la longueur *en lettres* ou en *phonèmes* des syllabes pour lesquelles les actions de calculs qui suivent sont définies. Il comporte donc une unité exprimée en lettres ou phonèmes et une expression de longueur.

2) Prédicat de position : comme pour les lettres, il exprime l'intervalle de position des syllabes dans le mot. La propriété de position qui sert à l'évaluation du prédicat est le numéro de réalisation de l'entité SUIIT-SYLL dans l'entité DEC-SYLL décrivant le mot puisque les syllabes, lors de l'analyse, sont stockées consécutivement dans leur ordre d'apparition.

3) Prédicat de compteur jouant le même rôle que dans un filtre-mot ou filtre-lettre.

4) Aucun type de syllabe, aucune fréquence n'ayant été dégagés, ces prédicats n'existent pas (3.4., Remarque 2).

Comme pour les prédicats des filtres-mots ou filtres-lettres, l'évaluation du filtre est le résultat d'une intersection logique entre ces trois filtres. Nos conventions imposent, en outre, cet ordre d'écriture des prédicats.

Nous pouvons, par ces règles, exprimer un filtre-syllabe traduisant la phrase suivante :

*"Si les 2ème et 3ème syllabes du mot comportent chacune moins de 4 lettres, et si le contexte transmis par le compteur X1 est 1, faire ..."*

### Filtre-phonème

1) Prédicat de longueur : celle-ci ne peut être exprimée qu'en caractères le prédicat joue le même rôle que précédemment.

2) Le prédicat de position, défini comme précédemment, est évalué grâce au numéro de réalisation de l'entité courante SUIT-PHON de DEC-PHON.

3) Prédicat de type : chaque phonème admet un type pour l'automate d'analyse syllabique et un type qui exprime sa prononciation réelle (labiale sourde, sonore ...), cf. Martinet [M1] p. 72-73. Nous avons pensé que, pour le développement ultérieur des méthodes d'abréviations, un tel critère pouvait être souhaitable. Nous voulons remarquer, cependant, que les simplifications que nous avons introduites dans la différenciation des phonèmes nous obligent à certains choix arbitraires au niveau des types ; ceux-ci ne reflètent pas ainsi toutes les propriétés des sons. Malgré les insuffisances phonétiques de ce prédicat, nous pensons qu'il pourrait aider à la construction prononçable d'abréviations de meilleure qualité.

4) Prédicat de fréquence : nous ne possédons aucune étude statistique nous permettant de stocker, pour chaque phonème, une fréquence relative au langage. Les propriétés de fréquence pour lesquelles nous exprimons un prédicat sont donc celle de fréquence relative à un mot et celle relative à l'ensemble des libellés par cumul des résultats sur les mots. Ce prédicat exprime les mêmes relations de fréquences que le prédicat équivalent pour une lettre, en permettant d'associer un calcul particulier aux phonèmes les plus fréquents ou les moins fréquents du jeu d'essai ou du mot.

5) Le prédicat de compteur joue le même rôle que les autres prédicats de ce type.

Le résultat de l'évaluation du filtre-phonème est toujours l'intersection des prédicats exprimés dans l'ordre ci-dessus.

### 3.5.3. Extensions des actions et des traitements

Nous avons construit les traitements autour des trois actions de base que nous avons définies en 2.4.6 : PRENDRE, TROUVER et "VIDE". Simplement, les contextes de chacune d'entre elles s'étant élargis, nous les avons étendues

en donnant à l'utilisateur les moyens de les définir plus précisément.

1) PRENDRE est toujours une action de choix dont le contexte est celui défini par le filtre qui la précède. Le contexte n'est plus maintenant limité à un caractère. Il peut être formé d'une syllabe dans le cas d'un filtre-syllabe ou d'un phonème. Nous conservons à cette action de choix la possibilité de provoquer un avancement conditionnel dans la chaîne des caractères du mot et d'enregistrer les résultats du choix par des modifications de compteur.

Nous lui ajoutons la possibilité de PRENDRE dans une syllabe le deuxième phonème par exemple, en exprimant, en complément du verbe-action PRENDRE que l'élément choisi est situé en un certain point de l'unité (2ème phonème) et que, depuis ce point, nous cherchons à retenir n éléments d'un certain type (phonème dans cet exemple). Le résultat du choix, quel qu'il soit, nous fait avancer dans la chaîne des caractères, sauf spécification explicite contraire, jusqu'à la fin de l'unité syllabique ou phonétique du contexte.

*"(pour une syllabe de trois phonèmes) PRENDRE un phonème, et PRENDRE un phonème consonne après le premier phonème".*

*Le résultat d'une telle action sur les syllabes soulignées des mots suivants serait :*

*HUISSIER = ui/sié ; s est le résultat du premier prendre. Le deuxième balaye la syllabe sans trouver de phonème consonne et le résultat final est seulement -ss-, traduction du premier phonème dans ses caractères.*

*PATE = p a t (le e est muet) ; le premier p est pris, puis la deuxième action rencontre t, phonème qui est transcrit -te-. Le résultat de ces deux PRENDRE est donc -pte-.*

2) TROUVER est toujours une action dont le contexte initial est précisé par les filtres. La recherche et le choix qu'elle effectue se situe sur ce contexte (syllabe, phonème ou lettre courant du mot) et à droite de ce contexte dans le mot sur la partie caractérisée "par les compléments de TROUVER". La progression se fait sur des limites de syllabes, si le contexte de calcul est syllabique, sur des limites de phonèmes pour un contexte phonétique.

Exemple : (Etant positionnés sur la deuxième syllabe du mot)  
TROUVER un phonème voyelle sans modifier l'index de position,  
et TROUVER deux phonèmes consonnes en partant du dernier phonème  
de la syllabe.

Nous montrons sur divers mots le résultat d'une telle action et  
la manière dont est ensuite déterminée la position dans le mot  
pour le filtre suivant

a) ACTIONNER : a c/s i o/n é

Le contexte de l'action est s i o.

Le premier TROUVER sélectionne i , phonème, et la lettre -i-.  
Le deuxième TROUVER part du phonème o et ne rencontre qu'un seul  
phonème consonne n qui se traduit -nn-. Le résultat final est la  
chaîne -inn-, et l'index est positionné à la fin du mot, puisque  
les deux dernières syllabes ont été balayées.

b) HABILITATION : a/b i/l i/t a/s i ò

Le premier TROUVER sélectionne -i-, le phonème de la syllabe bi.  
Le deuxième TROUVER part de ce phonème -i- et retient les phonèmes  
-l- et -t- des troisième et quatrième syllabes.  
Le résultat est -ilt-, l'index de position étant au début de la  
cinquième syllabe -si<sup>ò</sup>-.

L'extension essentielle de l'action TROUVER est donc liée à la plus grande  
richesse des compléments possibles qui définissent le choix. Celui-ci  
ne se fait plus seulement caractère par caractère, mais par groupe de  
caractères liés par leur appartenance à un phonème ou une syllabe. L'autre  
volet de cette extension est une progression dans le mot par bonds liés  
à la nature du contexte.

Avant de traiter un exemple qui montre la nouvelle puissance de ces actions  
et leur intérêt, nous voulons faire remarquer que notre but n'a pas été de  
permettre l'accès à n'importe quelle combinaison de phonèmes, syllabes  
ou lettres par une action de choix mais de donner certaines actions à  
contexte relativement précis. C'est au travers d'exemples que la signification  
de ces actions s'illustrera le mieux.

3) L'action "VIDE" conserve la même sémantique qu'auparavant : un filtre ayant isolé un caractère ou une suite de caractères, nous voulons seulement savoir que nous "sautons" ce contexte en enregistrant éventuellement cela par modification d'un compteur.

Le déplacement de l'index de position dans le mot est d'une syllabe, d'un phonème ou d'une lettre, suivant la nature du filtre. La seule modification est donc l'unité de déplacement de cet index.

4) L'action BOUT se trouve inchangée, jouant toujours le rôle d'inverseur du sens de balayage.

5) Les fonctions spécifiques demandées et écrites par l'utilisateur existent toujours. Elles sont enrichies du fait que l'on peut maintenant faire appel, pour les écrire, aux propriétés syllabiques et phonétiques qui conservent mieux les "sons".

### 3.5.4. Schéma étendu de la grammaire descriptive d'une méthode

Pas plus que précédemment au Chapitre 2, nous ne précisons le mode d'expression exact des divers prédicats et des actions complexes TROUVER et PRENDRE. Nous en donnons le détail au Chapitre 6 par la syntaxe et la sémantique du langage Abrègé qui sert à l'utilisateur d'interface avec le système pour décrire les méthodes qu'il veut exécuter.

La grammaire suivante très réduite explicite seulement l'enchaînement des prédicats et des actions.

*Méthode* = options-globales\* antidictionnaires\* filtre-libellé<sup>+</sup>.

*Filtre-libellé* = prédicat-longueur<sup>0/1</sup> filtre-mot<sup>+</sup>.

*Filtre-mot* = prédicat-position<sup>0/1</sup> prédicat-fréquence<sup>0/1</sup>  
prédicat-unité-longueur<sup>0/1</sup> prédicat-compteur\* traitement.

*Traitement* = dictionnaires | dictionnaires<sup>0/1</sup> calculs | calculs.

*Calculs* = [filtre-syllabe actions]<sup>+</sup> | [filtre-phonème actions]<sup>+</sup> |  
[filtre-lettre actions]<sup>+</sup>.

*Filtre-syllabe* = prédicat-unité-longueur<sup>0/1</sup> prédicat-position<sup>0/1</sup>  
prédicat-compteur\*.

*Filtre-phonème* = prédicat-longueur<sup>0/1</sup> prédicat-position<sup>0/1</sup> prédicat-type<sup>0/1</sup>  
prédicat-fréquence<sup>0/1</sup> prédicat-compteur\*.

*Filtre-lettre* = *prédicat-position*<sup>0/1</sup> *prédicat-type*<sup>0/1</sup> *prédicat-fréquence*<sup>0/1</sup>  
*prédicat-compteur*<sup>\*</sup>.

*Actions* = [*choix-prendre* | *choix-trouve*]<sup>\*</sup> *modif-compteur*<sup>\*</sup> *fonctions*<sup>\*</sup>  
*inversion-sens*<sup>0/1</sup>.

### 3.6. NOUVELLE EXPRESSION DE METHODE UTILISANT LES EXTENSIONS

L'exemple que nous proposons ici reprend le jeu d'essai CONGRES et propose une méthode d'abréviation sur ce jeu de nature beaucoup plus syllabique que la méthode 3 du Chapitre 2 (2.6.2.).

Cette méthode exprimée en clair est la suivante :

*METHODE 4 : Utiliser l'antidictionnaire standard et celui appelé ANTISPE propre à ce jeu d'essai pour supprimer les mots non significatifs.*

*Sur les textes ainsi transformés, appliquer les règles suivantes en limitant la longueur des abréviations à 14 caractères.*

- Pour les libellés d'un mot, si le mot a moins de 8 caractères, le prendre tout entier, sinon, s'il a 4 syllabes ou plus, prendre les 3 premières et la première lettre de la quatrième.*
- Pour les libellés de 2 mots, prendre soit l'abréviation spéciale de chaque mot si elle existe (dictionnaire), soit les deux premières syllabes de chacun. Si la deuxième syllabe du premier mot se termine par un phonème consonne, prendre le premier phonème voyelle rencontré ensuite.*
- Pour les libellés de 3 mots, prendre ou l'abréviation spéciale aux mots ou la première syllabe de chaque mot et le premier phonème de la deuxième syllabe.*
- Pour les libellés de 4 mots, prendre l'abréviation spéciale aux mots ou la première syllabe de chacun.*
- Pour les libellés de 5 mots et plus, prendre l'abréviation spéciale aux mots ou le premier phonème-consonne et le premier phonème-voyelle de chaque mot.*

Nous voyons que, dans toute cette méthode, les choix proposés s'attachent à ne pas détruire les sons. Cependant, ce qui nous importe maintenant, ce n'est pas d'en faire un bilan, mais d'en proposer une expression cohérente avec notre analyse des méthodes.

Méthode 4 : Expression schématique

options : doublets éliminés  
longueur maximum = 14  
collisions numérotées

antidictionnaires : ANTIDIC et ANTISPE

filtre-libellé : longueur-libellé = 1

filtre-mot : longueur-mot-lettre  $\leq$  8

calcul : filtre-lettre :

action : PRENDRE

filtre-mot : longueur-mot-syll  $\geq$  4

calcul : filtre-syllabe : position  $\leq$  3

action : PRENDRE

filtre-syllabe : position = 4

action : PRENDRE 1 lettre en position 1

filtre-libellé : longueur-libellé = 2

filtre-mot :

dictionnaires : SPECIAL

calcul : filtre-syllabe : position = 1

action : PRENDRE

filtre-syllabe : position = 2

actions : PRENDRE phonèmes position < longueur-syllabe (phon),  
INDEX fixe  
PRENDRE phonème position = longueur-syllabe (phon),  
compteur  $\leftarrow$  1 si phonème-consonne  
PRENDRE et compteur  $\leftarrow$  1 pour phonème-consonne avec  
position = longueur-syllabe (phon) ; index fixe  
PRENDRE et compteur  $\leftarrow$  0 pour phonème-voyelle avec  
position = longueur-syllabe

filtre-syllabe : position = 3  
compteur = 1

action : TROUVER 1 phonème-voyelle  
compteur  $\leftarrow$  0

*Cette méthode ne permet pas de traiter tous les mots (plus de 8 lettres, moins de 4 syllabes comme "hébergées") qui seront négligés à l'exécution.*



filtre-libellé : longueur-libellé = 3  
filtre-mot :  
dictionnaires : SPECIAL  
calcul : filtre-syllabe : position = 1  
action : PRENDRE  
filtre-syllabe : position = 2  
action : PRENDRE 1 phonème

filtre-libellé : longueur-libellé = 4  
filtre-mot :  
dictionnaires : SPECIAL  
calcul : filtre-syllabe : position = 1  
action : PRENDRE

filtre-libellé : longueur-libellé ≥ 5  
filtre-mot :  
dictionnaires : SPECIAL  
calcul : filtre-phon : position-phonème = 1  
TROUVER 1 phonème-consonne depuis 1, INDEX fixe  
TROUVER 1 phonème-voyelle depuis 1.

Remarques :

Nous pouvons constater que ce mode d'expression ne travaille pas sur des bornes fixes de position des caractères, mais permet simplement d'exprimer nos choix. Dans le mode antérieur d'expression, nous aurions dû, pour obtenir un résultat équivalent, travailler sur des combinaisons de valeurs de compteurs susceptibles d'exprimer les résultats de l'analyse syllabique et phonétique qui auraient été très complexes à maîtriser.

Si nous considérons les abréviations (Annexe [A3]) obtenues sur le jeu d'essai CONGRES par cette méthode, nous constatons que la longueur moyenne est proche de la méthode 3. La facilité de prononciation n'est pas fondamentalement améliorée. Mais, lorsque l'on sait que les abréviations ont été reconstituées à partir de syllabes, nous sommes tentés de lire syllabe par syllabe comme si l'abréviation individuelle de chaque mot était séparée de la suivante par un point ou un tiret.

Par exemple :

*GROUPFONCTAUT lu comme GROUP-FONCT-AUT*  
*ATCARTRAIT lu comme AT-CAR-TRAIT*  
*ENSCHAMBDISP lu comme ENS-CHAMB-DISP*  
*ANALERREURS lu comme ANAL-ERREURS*  
*GROUAGTOUHOT lu comme GROU-AG-TOU-HOT.*

Les exemples ci-dessus ont été volontairement pris sur des libellés de longueur variable. Nous voyons que, pour les libellés de 2 ou 3 mots, ce découpage permet de reconstituer pas mal d'information. Pour ceux de 4 mots, la "cesure" mot par mot se fait intuitivement sur les anomalies dans les séquences de lettres rencontrées "OUA", "GT", "OUHO". On peut se demander s'il ne vaudrait pas mieux introduire des tirets entre tous les mots grâce à une option de la méthode. Mais nous devons alors remarquer que les libellés les plus difficiles à traiter ceux de 4 mots et plus sont aussi ceux qui sont les plus pénalisés par l'introduction de séparateurs. C'est la raison pour laquelle nous y avons renoncé.

### 3.7. UNE METHODE D'ABREVIATION DE TEXTE PRODUISANT UN TEXTE ET NON UN IDENTIFICATEUR

Nous avons souhaité, dans l'analyse fonctionnelle du système Abrège pouvoir profiter des méthodes d'abréviations pour abrégé des textes tout en conservant suffisamment d'information pour que ceux-ci restent lisibles.

L'exemple que nous proposons ici est issu du système Gestion des Congrès déjà utilisé. Dans ce système, la liste des notions est assortie d'un dictionnaire du projet destiné à expliciter le contenu sémantique de chaque notion. Il nous semblerait pratique, si ce dictionnaire doit être stocké dans le système, d'en avoir une version explicite abrégée. Nous proposons une méthode susceptible de donner cette version.

#### Méthode 5 : Enoncé des principes

Dans une telle méthode, dont *le but est l'abréviation mot à mot*, la place du mot dans le texte n'a plus grande importance. Chaque mot va être abrégé pour lui-même et ce qui importe est que son abréviation soit discriminante par rapport à celle des mots proches.

*administrateurs, administration, administratif*

*ne peuvent être tous abrégés*

*adm. ou admin.. On préférera adminrs, adminon et adminif, par exemple.*

Nous avons donc ici à construire de bonnes règles à partir de filtres-mots et de calculs sur les mots.

Par ailleurs, dans un tel mode d'abréviation, la structure du texte a une grande importance. S'il semble possible de supprimer un certain nombre d'articles, il paraît souhaitable de conserver des prépositions significatives comme *auprès, pendant ...* Cela exclut pour nous l'utilisation de l'antidictionnaire standard du système, qui est très exhaustif ; nous faisons usage d'un antidictionnaire plus réduit. Il semble, en outre, souhaitable de ne pas traiter ces mots comme nous traitons verbes, adjectifs ou substantifs et d'en proposer plutôt une abréviation par dictionnaire.

Méthode 5 : Expression schématique

Traduction en clair

options	: mot à mot	<i>Abréger mot à mot sans limite de longueur</i>
antidictionnaires	: ANTIDICRED	<i>Antidictionnaire réduit</i>
filtre-libellé	:	<i>Pour tout libellé, faire</i>
filtre-mot	: longueur-mot-lettre $\leq$ 6	<i>1) Pour les mots de 6 caractères</i>
dictionnaire	: SPECPREP SPECDIC	<i>au plus, prendre leur abréviation</i>
calcul : filtre-lettre	:	<i>si elle existe ou les</i>
action	: PRENDRE	<i>conserver.</i>
filtre-mot	: longueur-mot-syll=2	<i>2) Pour les mots de 2 syllabes</i>
dictionnaire	: SPECPREP SPECDIC	<i>(et + de 6 lettres)</i>
calcul : filtre-syllabe	: position = 1	<i>Prendre leur abréviation si elle</i>
action	: PRENDRE	<i>existe ou prendre la première</i>
filtre-syllabe	: position = 2	<i>syllabe.</i>
action	: PRENDRE 2 lettres	<i>Dans la 2ème syllabe</i>
	PRENDRE 1 lettre depuis L	<i>Prendre les 2 premières lettres et la dernière.</i>
filtre-mot	: long-mot-syll = 3	<i>3) Pour les mots de 3 syllabes</i>
dictionnaire	: SPECPREP SPECDIC	<i>Prendre leur abréviation si elle</i>
calcul : filtre-syllabe	: position = 1	<i>existe ou</i>
action	: PRENDRE	<i>Prendre la 1ère syllabe</i>
filtre-syllabe	: position = 2	<i>(Prendre la 1ère lettre de la</i>
action	: PRENDRE 1 lettre	<i>2ème syllabe)</i>
filtre-syllabe	: position = 3	
action	: PRENDRE 2 lettres depuis L-1	<i>Les 2 dernières lettres de la troisième.</i>
filtre-mot	: long-mot-syll $\geq$ 4	<i>4) Pour les mots de 4 syllabes et</i>
dictionnaire	: SPECPREP SPECDIC	<i>plus, prendre l'abréviation</i>
calcul : filtre-syllabe	: position $\leq$ 2	<i>existante ou</i>
action	: PRENDRE	<i>les deux premières syllabes</i>
filtre-syllabe	: position = 3	
action	: PRENDRE 1 phonème	<i>Le premier phonème de la 3ème</i>
	INVERSION de Balayage	
filtre-syllabe	: position = L	
	PRENDRE 1 phonème	<i>le dernier phonème.</i>

Remarques :

1) Considérons les résultats en Annexe [A3]. Il est certain tout d'abord que la longueur de l'abréviation de chaque mot croît avec la longueur de celui-ci. Elle se stabilise sur les mots de 4 syllabes et plus.

2) L'exemple met en évidence *l'intervention de l'exécution sur l'expression de la logique*. Le premier filtre-mot qui s'adresse aux mots de moins de 7 lettres peut avoir une intersection non vide avec les 2 suivants, mais a priorité sur eux à l'exécution.

Ainsi

*"erreur" sera conservé comme tel et non abrégé "errrr" par le 2ème filtre-mot, alors qu'à l'exécution "accueil" est abrégé "accl" par le jeu du 2ème filtre, puisqu'il est refusé par le premier.*

*De même, "anime" n'est pas modifié, alors que "animer" est abrégé en "aner".*

3) Remarquons ce mauvais résultat ; il vient de la brièveté de la 1ère syllabe et nous tentons d'améliorer cette règle de la manière suivante :

filtre-mot	: long-mot-syll = 3	3 bis) Pour les mots de 3
dictionnaire	: SPECPREP SPECDIC	syllabes, prendre l'abréviation
calcul : filtre-syllabe	: POSITION = 1,	existante ou
	long-syll-lettre $\leq$ 2	prendre la 1ère syllabe et si
action	: PRENDRE	elle est courte, incrémenter un
	AJOUTER 1 COMPTEUR	compteur
filtre-syllabe	: POSITION = 1	sinon
action	: PRENDRE	la prendre sans modification du
		compteur.
filtre-syllabe	: POSITION = 2,	Si la 1ère syllabe était courte,
	long-syll-lettre $\leq$ 2	prendre la 2ème syllabe si elle
	compteur = 1	est courte (moins de 3 lettres)
action	: PRENDRE FIXE	et le premier phonème de la 3ème
action	: TROUVER 1 phonème de-	sinon
	puis dernier phonème	:
filtre-syllabe	: POSITION = 2	:
	:	:

Ce qui, alors, donne pour "animer" l'abréviation "animer", pour "groupement" toujours "groupnt".

4) Il semblerait souhaitable, étant donné nos habitudes par ailleurs, de mettre un point à la fin de chaque mot abrégé pour le différencier de ceux qui n'ont pas été modifiés, ce qui peut se faire par une fonction générale et peut-être faudrait-il inclure un tiret entre ce qui est extrait de la partie gauche du mot et ce qui vient de la partie droite "ani-er." et "group-nt".

### 3.8. CONSIDERATIONS SUR LES RESULTATS

Nous ne voulons pas ici trop approfondir les divers résultats des deux méthodes 4 et 5 proposées ci-dessus à titre d'exemple d'utilisation de Abrège.

#### 3.8.1. Les résultats de la méthode 4

La première, plus facile à exprimer que la méthode 3, donne des résultats équivalents quant à la longueur. Elle ne donne pas plus de collisions que la méthode 3, mais cela vient essentiellement de la structure des libellés. Nous ne pensons pas que de telles méthodes risquent de produire de nombreuses collisions parce que les collisions ont des chances d'être rares lorsque l'on combine des mots différents en tenant compte, en outre, de leur position et de leur longueur.

La prononciation des abréviations résultantes n'est pas tellement améliorée puisque nous nous sommes moins attachés à l'alternance consonnes-voyelles, mais il nous semble que le respect des syllabes donne des résultats plus mnémoniques. Néanmoins, ceci est encore une opinion personnelle que, seul, l'usage peut permettre de vérifier.

#### 3.8.2. Les résultats d'une méthode "mot à mot". La méthode 5

La méthode 5, qui travaille en mot à mot, donne des résultats relativement satisfaisants. Les textes abrégés [A3] nous paraissent, pour qui en connaît le contexte, assez facilement décodables à la première lecture. Il semble parfois regrettable que la méthode ne respecte pas la "racine" des mots. Mais cette notion d'origine morphologique ne peut être introduite actuellement. Nous en discutons au Chapitre 4 suivant. Nous ne disposons pas encore malheureusement d'un module d'évaluation statistique des résultats

pour savoir quel taux de compression du code nous atteignons ainsi. Ce résultat serait pourtant très intéressant, ainsi que la reconnaissance des mots que l'abréviation individuelle confond.

L'évaluation fait l'objet du Chapitre 5, mais nous voudrions terminer celui-ci sur quelques réflexions liées à cette dernière remarque.

### 3.8.3. L'abréviation mot à mot par Abrège

L'abréviation mot à mot est différente de l'abréviation unique de plusieurs mots par concaténation. Nous avons vu dans la méthode 5 que le mot est ici abrégé pour lui-même.

Le mode d'exécution et surtout de stockage des résultats nous paraît, dans ces conditions de travail, coûteux car le prototype que nous avons réalisé travaille toujours comme s'il formait, à partir d'un libellé, une abréviation unique. C'est ainsi que le mode d'abréviation mot à mot est simplement obtenu par insertion d'un espace entre les abréviations individuelles de chaque mot. Ceci n'est pas très important. Nous avons réservé dans la structure des libellés la place pour un texte en entrée de 180 caractères, ce qui représente à peu près une phrase simple de définition, il suffit de modifier la structure de l'abréviation elle-même pour stocker dans 4 mots Socrate, soit 120 caractères, la phrase résultante et réserver l'emploi de cette zone au cas où l'option *mot à mot* est déclarée.

Ceci modifie seulement l'entité ABREV-CALC de la manière suivante :

```
entité 20.000 ABREV-CALC /* contenu de l'abréviation */
```

```
début
```

```
  ABR-CONTENU Mot (30) avec clé unique fin /* premier mot qui sert à la */  
                                     /* recherche des abréviations homonymes pour une */  
                                     /* méthode donnée */
```

```
entité 4 ABR-TXT-CONT
```

```
début
```

```
  ABRT-CON mot (30) /* dans le cas de mot à mot, on stocke ici la */  
                   /* suite du texte abrégé du libellé. Les */  
                   /* collisions n'ont plus de sens */
```

```
fin
```

```
fin
```

L'action de "concaténation d'abréviation de mot" étant étendue pour satisfaire à ce mode de stockage.

Ce qui nous paraît inadapté à cet emploi, c'est le mode de calcul de l'abréviation de chaque mot. Ce calcul est, dans ce cas, indépendant de tout contexte du mot dans le libellé. Or, ceci n'était pas vrai dans le calcul d'identificateur. A cause de cela, l'exécution d'une méthode recalcule à chaque occurrence d'un mot dans un libellé l'abréviation de celui-ci dans le contexte où il se trouve. Pour abrégier des textes, il semblerait plus efficace d'abrégier d'abord un à un par calcul cohérent avec la méthode chacun des mots stockés dans MOT-CONT de MOTS-LIB, de garder l'abréviation dans une caractéristique supplémentaire de MOTS-LIB, puis d'abrégier les textes en recherchant soit l'abréviation dans un dictionnaire, soit l'abréviation calculée du mot. La structure pourrait alors être :

entité 5000 MOTS-LIB

début

    MOT-CONT mot (30) avec clé unique fin

    :

    entité 10 MOT-ABR /\* on peut ainsi stocker une abréviation par méthode \*/

    début

        MOT-ABR-CONT mot (20)

    fin

fin

La modification essentielle étant ici dans la logique des phases d'exécution.

Notons seulement au passage la simplicité relative de la modification, essentiellement due au fait que nous n'avons pas à gérer des fichiers puisque nous utilisons un SGBD.





## CHAPITRE 4

LES CARACTERISTIQUES MORPHOLOGIQUES  
DES ABREVIATIONS

#### 4.1. POURQUOI UNE ANALYSE MORPHOLOGIQUE ?

Nous avons remarqué, dans le chapitre précédent, et en particulier lors de l'abréviation de textes mot à mot, que nous aimerions parfois disposer de propriétés de mots non spécifiquement orthographiques ou phonétiques.

Le livre de A. Martinet [M1] étudie l'importance d'une bonne connaissance du rôle et de la structure des mots pour la reconnaissance du langage. Nous voudrions, pour en montrer l'intérêt, revenir sur deux exemples. Le premier est le jeu d'essai dont nous avons abrégé les libellés descriptifs dans la fin du chapitre précédent. Nous avons alors remarqué comment le hasard de nos choix syllabiques donnait un mauvais résultat dans le cas de syllabes initiales courtes comme dans le mot "animer." → "anier".

Nous pouvons évidemment rajouter des règles gérant de plus en plus de cas particuliers pour une méthode syllabique comme nous l'avons fait antérieurement pour les méthodes purement alphabétiques. Mais nous sommes tentés de dire que les mots adjectifs, substantifs, verbes ou adverbes sont généralement formés

- d'une racine portant une part importante du sens du mot.
- de préfixes ou suffixes modifiant ou précisant cette sémantique, de désinances significatives du rôle et du contexte du mot.

Et comme lorsque nous avons introduit l'analyse phonétique et syllabique du mot pour tenir compte de propriétés plus globales que celles des caractères - ce qui a eu un effet simplificateur sur l'expression des méthodes -, nous souhaiterions introduire des propriétés morphologiques et syntaxiques, espérant gagner ainsi une puissance supérieure pour Abrège.

Le deuxième exemple que nous avons peu cité jusqu'à maintenant est la documentation sur la "Norme pour la création d'un code international pour l'abréviation des titres de périodiques" (Annexe [A6]). Les abréviations dans ce contexte se font soit par contraction de la partie désinence-suffixe après conservation de la racine, soit par troncature de la racine elle-même. Les choix de conservation des parties de mots, mais aussi des articles, sont liés à la reconnaissance du rôle morphologique et syntaxique du mot comme critères de choix de règles d'abréviation.

Par exemple :

- *certaines règles s'appliquent aux adjectifs*
- *le pluriel est censé être reconnu et supprimé.*

Nous n'avons pas cherché à traduire toutes les règles de la Norme. Il est certain qu'elles supposent l'usage de nombreux dictionnaires. Il est aussi très difficile de réaliser un analyseur valable pour toutes les langues rencontrées. Mais nous pouvons nous inspirer de cette documentation pour comprendre les besoins des auteurs de la Norme : la préoccupation de ces derniers est, en effet, proche de la nôtre, à savoir obtenir des équivalents de textes assez courts : les titres et les références de publications scientifiques périodiques ou occasionnelles.

Nous avons, pour cela, recherché les outils d'analyse morphologique et syntaxique existants (comme nous l'avons fait pour la phonétique) afin de ne pas affronter des problèmes aussi complexes en partant de rien.

Le développement, dans une équipe proche de la nôtre, d'un système destiné à automatiser ces analyses nous a conduits à étudier plus précisément ce qu'elle proposait. Le Système PIAF [C2] est un programme interactif d'analyse du Français dont le but est de permettre de réaliser ces analyses morphologiques et syntaxiques en tenant compte des progrès faits par la linguistique dans ce domaine. Il nous semble bien refléter les difficultés propres à l'analyse du français, même si son implantation apporte d'autres difficultés quand on cherche à réaliser une interface PIAF-Abrège. Ce système fait donc l'objet de l'étude qui suit.

#### 4.2. LE SYSTEME PIAF. PROGRAMME INTERACTIF D'ANALYSE DU FRANCAIS [C2]

Nous voulons seulement donner les grandes lignes du fonctionnement de PIAF pour expliquer nos décisions concernant l'analyse morphologique.

Le rôle de ce programme est bien, ainsi que nous le désirions, de donner pour une occurrence de mot *tous les découpages possibles* de celui-ci en :

bases,  
préfixes et suffixes,  
désinences ou groupes de désinences.

Ainsi, PARTIE sera reconnu comme

la base PART

IE désinence d'un substantif féminin

I E désinences de participe passé féminin du verbe.

Cet exemple montre qu'un mot peut appartenir à des classes différentes : classe substantif, classe verbale.

Le découpage est ici parti de la même base mais à la même chaîne de caractères peuvent parfois s'appliquer plusieurs bases. L'exemple habituel est le substantif "COUVENT" et le verbe "COUVENT".

##### *a) Les constituants du Système Piaf : dictionnaire, modèle, règles de transition.*

Le système d'analyse comporte un *dictionnaire* grâce auquel on cherche la plus longue chaîne du mot se superposant à un élément du dictionnaire. Celui-ci donne alors les autres "possibles" plus courts.

Ce dictionnaire donne donc les "graphies" qui servent de base à l'analyse morphologique (les bases, les préfixes, suffixes et désinences).

Comme à une base donnée ne peuvent être associés tous les préfixes suffixes et désinences possibles en français, le système comporte des *modèles morphologiques* décrivant les associations permises pour une base donnée, et les règles de passage d'un modèle à un autre pour une base donnée. Ces modèles sont interprétés par un *automate d'état fini*.

Les états successifs de l'automate, ainsi que les règles de transduction d'une graphie en morphèmes, sont exprimés par des *règles* présentées dans un format déterminé qui permet leur interprétation.

b) *Le fonctionnement général.*

Le dictionnaire est un outil de travail conçu pour minimiser le temps de recherche des éléments qui valident ou invalident un modèle. Tout mot est étudié par le système soit parce que ses règles de transduction sont données dans les modèles (par exemple : "*aimer*"), soit parce qu'il est indexé dans le dictionnaire par une relation d'équivalence avec un "mot de base"

(par exemple : *mang-er / aim-er* ou  
*après / avant* ).

Cette indexation est facilitée par le caractère *interactif* du système qui permet une prise en compte dynamique des mots non encore reconnus.

Des modèles nouveaux peuvent aussi être introduits de manière interactive (voir par exemple *l'analyse phonétique PIAFPHO* [C1]).

L'ensemble de ce système est écrit sous CP/CMS, le dialogue pouvant être réalisé sur télécype ou écran de visualisation. Une saisie en traitement par lot est néanmoins possible.

c) *Les résultats.*

L'analyse morphologique est réalisée mot à mot pour les textes donnés. Pour chaque mot, le système produit toutes les décompositions valides du mot avec, pour chacune d'elles, les divers constituants morphémiques qui ont été successivement validés - bases-préfixes-désinences - ainsi que les classifications auxquelles ont conduit les dérivations. Par exemple :

*aimera : aim er a verbe indicatif futur 3ème pers. singulier.*

Dans cet exemple, le résultat est non ambigu, mais il arrive souvent que plusieurs décompositions aient été ainsi également validées. C'est alors à l'utilisateur d'indiquer quelle est la bonne dérivation, à moins qu'un programme d'analyse syntaxique ne soit également appliqué. Toutefois, les travaux correspondants sur l'analyse syntaxique du français sont encore en pleine évolution à l'heure actuelle.

### 4.3. PROBLEMES LIES A L'EVENTUELLE UTILISATION DE PIAF DANS LE SYSTEME ABREGE

#### 4.3.1. Difficultés contingentes de réalisation

Un certain nombre de caractéristiques du système PIAF font que la réalisation d'une interface PIAF-Abrège présenterait de graves difficultés.

1) Le système PIAF est réalisé sous CP/CMS, en grande partie en PL1. Le logiciel que nous avons écrit est réalisé sous Socrate "ERIA", compatible IRIS 50-SIRIS 2 et 360-OS. Nous ne pouvions travailler sous les 2 systèmes simultanément car le système à machine virtuelle CP/CMS ne transmet pas de fichiers compatibles sous OS.

Actuellement, une version Socrate "sous CP" étant à l'étude au C.I.C.G., en travail commun avec ERIA, nous pourrions peut-être envisager de communiquer entre deux machines virtuelles pour transmettre textes et résultats de l'une à l'autre, ceci dans une autre version d'Abrège, évidemment.

2) Le système PIAF est interactif, Abrège ne l'est pas. Nous avons, en effet, réduit le système Abrège dans cette version à un système à traitement par lot (nous discutons de la transformation d'Abrège en système interactif dans nos conclusions au Chapitre 7).

L'interface PIAF-Abrège, dans ces conditions, aurait dû nécessairement être réalisée par transport de l'information via un fichier classique sur bande (les disques ne sont pas organisés de la même manière dans les deux systèmes).

Nous pouvons faire un bilan des informations que nous devons transmettre dans une telle perspective.

Il faut : - tous les textes ou tous les mots  
- l'indexation donnant pour chaque mot les modèles à appliquer.

En échange de quoi, nous obtenons la ou les décompositions morphologiques et les classifications de chaque mot.

L'absence de possibilité d'*interaction* et de *communication entre systèmes* alourdit ou complique l'exploitation de ces résultats, en raison, en outre, de la nature de ceux-ci.

- a - en introduisant un délai de traitement donc d'obtention des résultats définitifs (passage d'un système à l'autre, car ils ne "cohabitent" pas sur le 360).
- b - en introduisant une spécificité de système (emploi de CP/CMS) qui ne peut être levée que par emploi du réseau Cyclades pour le transfert, ce qui, même actuellement, n'est pas une procédure simple. Nous voulons réaliser un prototype simple, pas "un monstre".

#### 4.3.2. Réflexions sur l'opportunité d'une analyse morphologique

Après les considérations ci-dessus liées aux logiciels et à leur emploi, et qui reflètent surtout notre attitude vis-à-vis de la création d'un système "mixte", nous devons surtout étudier la nature même des informations sur lesquelles nous travaillons et les conséquences de l'emploi éventuel de l'analyse morphologique sur les phases de saisie des données et d'exécution des méthodes.

1) L'obligation où nous nous trouvons d'indexer chaque mot du vocabulaire employé pour permettre sa reconnaissance suppose soit un gros travail après la saisie des données, soit la donnée préalable d'un dictionnaire dont l'analyse risque de ne pas satisfaire l'utilisateur. En outre, ceci ne peut être transparent à l'utilisateur et alourdit considérablement la saisie et les modifications de jeu d'essai pour une amélioration du système qui n'est tout de même pas un des points clé de nos objectifs.

2) Les résultats mettent en évidence des homomorphismes de mots (plusieurs décompositions pour une même graphie) que l'on ne peut lever qu'en remplaçant chaque mot dans son contexte. Ceci, devant être fait au retour du traitement des données par le système PIAF, allonge les délais avant l'exécution d'une méthode de type morphologique et suppose, au niveau des mots de chaque texte, la saisie d'informations complémentaires pour choisir la décomposition morphémique spécifique de chaque occurrence.

Remarquons que cette pluralité existe aussi dans l'analyse phonétique où, seul, l'emploi d'un analyseur simplifié permet de la masquer. Mais elle est alors moindre et joue un faible rôle au niveau du découpage du mot suivant ses phonèmes. C'est surtout l'interprétation précise du graphème en phonème qui change. Les propriétés que nous utilisons pour caractériser les phonèmes n'en sont pas fondamentalement modifiées. Nous avons vu, dans le premier exemple concernant le mot "partie" que ceci n'est plus vrai lors de l'analyse et de la décomposition morphologique.



#### 4.4. CONCLUSION

Au terme de ces réflexions, nous avons renoncé à ajouter définitivement au système Abrège, lors de l'abréviation, la possibilité de prendre en compte des propriétés morphologiques des mots.

Cela pourrait, toutefois, être utile dans la réalisation de certaines applications spécifiques où l'utilisateur serait motivé pour réaliser l'indexation des textes, ou, dans le cas de vocabulaires spécifiques bien connus et réduits. Cela est à peu près inutile lors de la réalisation d'identificateurs. Les dictionnaires que nous proposons peuvent servir à donner, comme abréviation, la racine de certains mots usuels ou spécifiques.

Dans le cas d'abréviation de textes, nous pouvons encore proposer aux utilisateurs de faire un large usage de dictionnaires. Pour reprendre alors une remarque que nous avons faite (fin du § 2.6.2) concernant la nécessité où nous nous trouvons de stocker à la fois dans les dictionnaires les formes singulier et pluriel des mots, peut-être pourrions-nous modifier les méthodes. On peut essayer d'employer des règles de troncature de désinences (par dictionnaire de celles-ci) avant abréviation des mots lorsque leur "racine" est contenue dans un dictionnaire des racines à reconnaître.

Ceci modifierait un peu la grammaire des méthodes (application de fonctions générales *avant* le traitement individuel du mot) et augmenterait les contraintes de préparation de méthode par ajout de dictionnaires. Mais un tel résultat intermédiaire peut être obtenu sans provoquer de résultat ambigü, comme dans l'analyse morphologique, et sans compliquer le traitement propre des textes, donc l'exploitation.

**CHAPITRE 5**

**EVALUATION ET VALIDATION DE METHODES**

L'un de nos objectifs fondamentaux, lors de la définition d'Abrège, a été de permettre la création de "bonnes méthodes" d'abréviation. Nous avons bien montré la nécessité de celles-ci, mais nous n'avons jamais précisément défini en quoi une méthode pouvait être considérée comme bonne ou mauvaise, dans quelle mesure ce caractère était permanent ou plus ou moins passager.

Dans ce chapitre, nous voudrions donc nous attacher à la définition de quelques critères permettant l'évaluation de méthodes. Nous voulons aussi montrer les limites de nos possibilités de validation. Enfin, puisque rien n'a été implémenté à ce jour répondant à cet objectif, nous donnons les grandes lignes de ce que doit être la phase de validation ainsi que des éléments organiques de sa réalisation.

#### 5.1. LES VALIDATIONS IRREALISABLES

Nous avons déjà dit plusieurs fois l'impossibilité où nous nous trouvons de mesurer scientifiquement la plus ou moins bonne mémorisation des abréviations obtenues. Ceci ne peut se faire que par l'intermédiaire des méthodes de validation de la psychologie cognitive. Celles-ci supposent des tests nombreux, sur des populations variées, durant des périodes de durée variable pour mesurer la mémorisation à court et à long terme, toutes conditions irréalisables pour un produit développé avec des moyens assez simples. Nous avons donc renoncé, pour cet objectif de mémorisation, à rechercher plus que l'analyse subjective de la satisfaction ou l'insatisfaction de quelques utilisateurs d'Abrège.

Il est aussi malaisé de mesurer une caractéristique de prononçabilité des identificateurs. Il semble pourtant que les identificateurs imprononçables seront difficiles d'emploi et risquent de provoquer des erreurs d'orthographe, signes d'une mauvaise mémorisation. L'étude des corrélations entre la mémorisation et la prononçabilité semble intéressante mais difficile à réaliser. Nous nous contentons encore une fois de l'opinion de l'utilisateur.

Il n'est pas davantage possible de mesurer la valeur sémantique de l'identificateur ou de la comparer à la sémantique du libellé. Celle-ci ne serait sûrement pas la même pour les analystes-programmeurs d'un projet en cours de réalisation, pour les utilisateurs non impliqués dans le projet, mais qui sont familiers avec son contenu, pour toutes ces personnes "2 ans après" et pour un étranger, lors du premier contact avec la documentation de ce projet. Il y a certainement une corrélation entre cette valeur sémantique

et les caractères de "décodabilité immédiate" des identificateurs, de mémorisation à court et à long terme.

La validation de "prononçabilité" est de type phonétique, les deux autres nous renvoient à la psychologie cognitive. Toutes trois font appel à une physique du comportement trop complexe pour que nous tentions de l'étudier dans un contexte d'emploi aussi imprécis que celui d'Abrège.

## 5.2. CE QU'IL EST REALISTE D'ENVISAGER. LES OBJECTIFS

Abrège est fondamentalement un outil de compactage de texte. A ce titre, il semble important de mesurer un certain nombre de variables qui pourraient permettre de comparer les résultats respectifs de diverses méthodes appliquées au même jeu d'essai.

Par ailleurs, lors de la définition d'une méthode, l'utilisateur a, au mieux, une idée globale du genre des libellés sur lesquels va travailler la méthode : il ne peut pas, avant de définir une méthode, mesurer tous les paramètres possibles pour savoir que considérer lors de l'abréviation (par exemple

*la fréquence des mots*

*le dénombrement du vocabulaire*

*les longueurs des textes en mots, caractères ...).*

Nous n'avons pas non plus prévu un premier passage de "caractérisation statistique" de jeu d'essai. Cela pourrait, cependant, s'avérer utile comme "sous produit" de la phase de saisie, pour guider l'utilisateur vers un type de méthode plutôt qu'une autre. Mais cela n'est pas.

Nous envisageons, en fait, une évaluation a posteriori, de la manière dont l'exécution a utilisé les propriétés prises en compte dans une méthode. Ce pourra être par exemple :

*- l'utilisation globale des dictionnaires*

*- l'utilisation individuelle des mots des dictionnaires*

*- l'ordre et le nombre de passages sur les filtres*

*- le nombre de validation des filtres*

*- les choix, leur utilisation positivement et négativement.*

Une telle évaluation peut mettre éventuellement en évidence des règles jamais atteintes, des dictionnaires inutiles, des règles très employées, mais insuffisamment détaillées pour la satisfaction de l'utilisateur quant au résultat final.

Nous avons donc essentiellement deux objectifs :

- le premier est de caractériser les résultats les uns par rapport aux autres,
- le deuxième est de mesurer l'exploitation qui est faite de la base et d'une méthode.

Nous tentons maintenant de préciser pour chacun de ces objectifs les paramètres que nous pensons pouvoir mesurer.

### 5.3. LE CODE OBTENU. MESURES

Les abréviations sont considérées comme des codes de libellés, obtenus par compactage. Nous leur appliquons donc quelques mesures classiques pour de tels procédés.

- a) - Evaluation de la *longueur des mots-abréviations* obtenus.
  - Mesure de cette longueur pour chaque mot (c'est une donnée de la base).
  - Mesure d'une longueur moyenne, et de son écart type.
  - Présentation graphique des longueurs des mots relativement les uns aux autres.
- b) - Evaluation de la longueur obtenue par rapport à la longueur du libellé initial.
  - Taux de *compactage*.

Comme précédemment, nous proposons de calculer d'abord un résultat absolu pour chaque libellé, puis des écarts statistiques permettant une certaine analyse des données.

Il est à remarquer que cette mesure paraît particulièrement intéressante dans le cas d'abréviation du libellé mot à mot lorsque le texte conserve sa structure.

c) *Mesures de la différence entre les abréviations.*

La première mesure dont nous avons signalé l'importance à plusieurs reprises est celle du *nombre et du taux de collisions*. Il semble important, pour faciliter à l'utilisateur l'analyse des éléments de la méthode qui ont provoqué ces collisions, d'ajouter à ces résultats une édition des collisions et des textes initiaux associés. Il paraît, cependant, superflu d'ajouter en outre une trace de l'exécution de la méthode sur tous les libellés dans le but d'en éditer une dans les cas de collision.

La deuxième mesure est celle d'une *collision partielle* sur les débuts de mots que nous baptisons "*début-collision*". Cela paraît être une information dont l'utilisateur pourrait se servir, pour évaluer d'une part s'il peut raccourcir les identificateurs sans risque de collision, d'autre part pour étudier dans quelle mesure des débuts identiques servent ou desservent la mémorisation.

Comment mesurer ces "*débuts-collisions*" ? En donnant pour des longueurs de chaînes de caractères différentes le nombre d'identificateurs ayant même début et le nombre de chaînes de cette longueur correspondant à ces collisions. Mais cela n'est pas très défini et peut être une information partiellement erronée. En effet, si deux identificateurs sont en début-collision sur cinq caractères par exemple, mais pas sur 6, il paraît peu significatif de les considérer à nouveau en début-collision sur toutes les chaînes plus courtes de 1, 2, 3 et 4 caractères. Il faut donc mesurer seulement les débuts-collisions sur la chaîne la plus longue possible.

Ceci risque d'être long et pas très performant si le jeu d'essai comporte beaucoup de libellés. Nous n'avons pas encore étudié comment optimiser cet algorithme.

Quant aux informations fournies lors de cette mesure, elles peuvent être de nature graphique (répartition en fonction de la longueur de la chaîne) ou explicite par édition de groupes d'identificateurs comportant un début-collision. Cette forme de résultats moins statistique est probablement plus utile pour une comparaison subjective des résultats.

d) *Mesures de l'information perdue totalement à l'abréviation.*

Il y a d'abord la *perte d'informations peu significatives par utilisation des antidictionnaires*. A ce titre, il serait probablement intéressant de donner des résultats concernant le nombre de mots initiaux des libellés et le nombre de mots utilisés dans l'abréviation, après la suppression des mots non

significatifs. Nous pouvons facilement disposer de cette information, le problème à ce niveau étant celui de savoir sous quelle forme transmettre cette information à l'utilisateur pour qu'il puisse en faire usage quand il définit des libellés.

Par ailleurs, nous sommes parfois conduits, en exécutant la méthode, à donner à certains mots une abréviation vide non parce que les dictionnaires en décident ainsi - ce qui est connu de l'utilisateur - mais parce qu'aucun filtre ne s'applique au mot considéré. Il paraît important que l'utilisateur puisse clairement savoir dans quels cas la méthode qu'il a définie produit ce résultat.

Enfin, il semble souhaitable de considérer une mesure "d'écrêtage" lorsque la méthode comporte une limite de longueur pour l'identificateur. On est alors conduit à supprimer la partie droite de l'abréviation qui est exédentaire. Une connaissance explicite des libellés "écrétés", ainsi que des mesures du taux d'écrêtage, de la répartition de ceux-ci en fonction du nombre de mots significatifs du libellé, peut orienter l'utilisateur vers une amélioration de la méthode.

Ce sont, pour l'instant, les mesures essentielles qu'il nous semblerait utile d'implanter dans Abrège, dans le but de guider l'utilisateur qui recherche la ou les bonnes méthodes d'abréviation. Nous voyons que ces résultats sont fondamentalement de deux types :

- des informations éditées de listes de valeurs : listes de collisions, de libellés écrétés, de mots négligés
- des informations statistiques : dénombrements, taux, écart type, graphe de résultats numériques.

Nous étudions en 5.5 les problèmes d'implantation.

#### 5.4. L'EXECUTION D'UNE METHODE. UTILISATION DES PROPRIETES, DES REGLES : BILAN D'EXPLOITATION

Lorsqu'un utilisateur définit une méthode ou en emploie une prédéfinie, il ne sait pas très bien suivant quels critères vont être abrégés les libellés. Ceux-ci peuvent en effet être traités essentiellement en fonction des dictionnaires ou de certaines règles, le passage à travers la méthode étant conditionné par la nature des données, bien entendu. Nous envisageons donc d'ajouter à l'évaluation des résultats une *évaluation d'exploitation* des méthodes.

Dressons donc un bilan des accès qui semblent importants ; nous suivons pour cela la structure même d'une méthode. Il nous faut connaître le rôle joué par :

- . les filtres aux divers niveaux
- . les dictionnaires
- . les choix
- . les fonctions spéciales à l'utilisateur.

a) Les filtres - C'est par une connaissance de nature statistique sur la consultation des filtres et la réalisation effective des prédicats que nous pourrons d'une part analyser les sections de méthode "inutiles" car non consultées, d'autre part tenter des corrélations avec les résultats des mesures de qualité des abréviations, enfin connaître le fonctionnement réel de la méthode.

b) Les dictionnaires - L'importance de leur rôle dans l'abréviation conduit à envisager deux niveaux d'analyse. Dans la mesure où les dictionnaires ne sont consultés que lorsqu'ils contiennent une abréviation du mot, le taux global de consultation semble déjà une information intéressante. Il paraît, cependant, encore plus fructueux de prévoir une analyse détaillée de cette consultation au niveau du mot. Si cette trace ne pénalise pas trop l'exploitation de la méthode, nous disposerons alors d'informations exhaustives quant au rôle des dictionnaires.

c) Les choix - Comme pour les dictionnaires, il est important d'évaluer au moins partiellement le résultat des actions de choix définies dans les méthodes. Celles-ci ont cependant un rôle moins simple que celui des dictionnaires. Deux informations paraissent importantes : dénombrer, pour chaque choix, d'une part le nombre de fois où il a été envisagé (à l'exécution), d'autre part le nombre de fois où il a effectivement conduit à la saisie de caractères pour l'abréviation.

d) Les fonctions spéciales - Lorsque celles-ci ne sont pas appliquées à tout coup, il faut connaître leur contexte d'emploi. Ce sera, par exemple, une trace des cas où une deuxième méthode a été exécutée pour lever des collisions, ou bien une trace des caractères insérés pour l'euphonie dans la chaîne retenue par les choix, etc. Dans la mesure où ces fonctions sont définies et programmées en Socrate par l'utilisateur, nous lui laissons aussi la charge d'analyser lui-même ses résultats en lui proposant seulement de dénombrer et éventuellement de situer par rapport aux



libellés l'intervention de ces fonctions.

Avant de passer aux éléments directeurs de l'implantation d'une évaluation dans Abrège, nous tenons à signaler un aspect du problème d'évaluation : c'est celui de la présentation adéquate des résultats pour que la masse d'informations que nous pourrions ainsi donner soit perçue par l'utilisateur non comme un agrégat amorphe, mais comme un ensemble structuré facilitant les corrélations. Dans cet esprit, cependant, nous avons décidé de ne pas faire toute l'évaluation d'un seul coup. Nous la présenterons à l'utilisateur comme une partie de la méthode qu'il lui faut demander explicitement après la partie algorithmique de la description de méthode. Il peut ainsi s'attacher séparément à différents aspects et ne pas pénaliser l'exécution par le coût de phases de validation qui ne l'intéressent pas. Nous revenons sur ce point au Chapitre 6, en donnant un schéma de ce que sera cette description dans le langage proposé à l'utilisateur.

#### 5.5. IMPLANTATION DE L'EVALUATION

Lorsque nous avons divisé l'évaluation en deux parties, l'une concernant les résultats, l'autre l'exploitation d'une méthode, il semble que nous ayons mis en évidence deux types de cette évaluation : un temps d'évaluation ou plutôt de stockage d'informations pour l'évaluation en cours de travail, puis un temps de mesures a posteriori sur les résultats. Le troisième temps est celui de l'édition et des calculs statistiques sur les informations brutes.

- En effet, c'est en cours d'exploitation que nous devons garder trace des accès aux filtres, aux choix, aux dictionnaires et que nous devons conserver une trace du résultat positif ou négatif des actions.

- C'est apparemment plutôt quand une méthode a été entièrement exécutée qu'il faut mesurer la longueur moyenne des abréviations et dénombrer collisions et début collision.

- Pour obtenir les résultats cherchés, il faut ensuite, dans une troisième phase, effectuer des calculs plus statistiques sur ces résultats premiers qui ne sont que des dénombrements d'ensembles ayant certaines propriétés.

Puisque nous avons dit au paragraphe précédent que nous ne traitons pas ici de la manière dont nous interprétons l'expression des désirs de l'utilisateur

en langage Abrège, nous ne précisons donc pas non plus comment nous sélectionnerons les éléments à réaliser pour chaque phase. Nous nous intéressons ici à la réalisation globale de toute évaluation pour les trois phases précédentes.

#### a) Evaluation en cours d'exploitation

Nous n'avons pas encore parlé de la manière dont nous réalisons l'exécution d'une méthode. Disons simplement que cette exécution est obtenue par stockage dans une structure Socrate de la description de la méthode proposée par l'utilisateur (phase de transcription du langage en une sorte de pseudo-code qui est l'entité DESCR-METHODE créée), puis l'exécution proprement dite est l'interprétation du contenu de cette entité par un programme qui évalue les filtres et active les choix et fonctions spécifiques.

Dans ce contexte, il suffit de rajouter, dans l'entité DESCR-METHODE, les caractéristiques d'évaluation des accès qui nous intéressent et de réaliser aux divers niveaux de l'interprétation la mise à jour de ces caractéristiques.

Celles-ci étant de nature simple (*par exemple, compteur de filtre*), il est avantageux de les mettre à jour à tout coup plutôt que de se demander si l'action est à faire ou non. C'est en phase 3 (édition) que nous choisirons d'exploiter les résultats intéressant l'utilisateur.

#### b) Evaluation des résultats

Notons que certains résultats (comme le nombre de mots "significatifs" d'un libellé pris en compte par une méthode) demandent un stockage intermédiaire en cours d'exécution. Nous proposons pour les résultats de ce type de nous ramener au cas a).

Les autres éléments d'évaluation sont donnés par analyse des résultats d'exécution de la méthode. Nous réaliserons donc des programmes qui calculent chaque type de dénombrement indépendamment des autres par balayage des résultats enregistrés dans la base de donnée. Il serait peut-être plus performant d'étudier s'il existe une hiérarchie entre les critères demandés, mais ceci ne peut être que le résultat d'une analyse fine du problème.

Notons simplement au passage la facilité que nous donne le SGBD Socrate d'accéder à toutes ces caractéristiques, de telle sorte que nous n'avons pas été obligés de réaliser toute cette analyse en même temps que celle de l'exécution de la méthode comme nous aurions probablement été obligés de le faire avec un système de gestion par fichiers classiques. Pour réaliser notre prototype, Socrate nous donne une souplesse d'accès à tous les niveaux qui

permet, en outre, d'étudier ce que devrait être une réalisation classique correcte après avoir mis en évidence toutes les informations et tous les accès nécessaires.

c) Les statistiques et l'édition

Pour donner à l'utilisateur des résultats statistiques dans une forme satisfaisante (édition), il faut préalablement avoir réalisé l'analyse des données que nous avons enregistrées dans la base. Nombreux sont les logiciels d'analyse de données (souvent des "utilitaires" fournis par le constructeur). Il est donc inutile d'en réaliser la programmation à notre tour.

Nous réaliserons donc simplement à l'exécution des bandes de données prêtes à être employées par le logiciel choisi d'analyse de données, pour toutes les données dont l'utilisateur aura exprimé le désir d'une analyse statistique. Une phase spéciale de traitement en réalisera ensuite l'analyse et l'édition.

Les seules données directement éditées seront à la demande les libellés et leur abréviation ou les abréviations seules.

**CHAPITRE 6**

.....  
.....  
**SPECIFICATIONS EXTERNES DU SYSTEME ABREGE**  
.....  
.....

L'objet de ce chapitre, au terme de la description de ce que réalise Abrège, est de montrer comment se présente Abrège à un utilisateur. A la lecture des chapitres précédents nous avons constaté que l'utilisateur a principalement trois tâches à réaliser, lorsqu'il veut tester des méthodes.

- 1 - Définir une ou plusieurs méthodes qu'il désire essayer
- 2 - Définir les dictionnaires dont il compte faire usage
- enfin 3 - Définir les données sur lesquelles il veut travailler.

Après cela il peut enfin demander l'exécution d'une ou plusieurs méthodes, sur tout ou partie des données.

Nous supposons évidemment que l'utilisateur dispose d'une base Abrège avec tous ses programmes car l'initialisation de celle-ci n'est pas de son ressort. Dans ces conditions nous lui proposons de s'exprimer dans un langage simple que nous décrivons dans ce chapitre. Ce langage est interprété et sert à la création de tous les éléments nécessaires à l'exécution.

Nous proposons ce langage dans le cadre que nous nous sommes fixé d'exécution de chaque phase en mode de traitement par lot. L'interprétation du langage devrait être un peu différente dans le cas d'un traitement conversationnel interactif (il faudrait rajouter des primitives d'interaction pour changer de niveau d'interprétation à la demande), mais le langage pourrait rester fondamentalement le même.

Le langage actuel comporte déjà une interprétation à deux niveaux.

- Au premier niveau, l'utilisateur exprime la nature de la tâche qu'il veut réaliser - définition de méthode, de dictionnaire, saisie de donnée ou exécution -
- Au deuxième niveau, il exprime le contenu de la phase de traitement.

Le langage est défini successivement à ces deux niveaux et pour chaque phase. Les parties syntaxiques de la description servant à spécifier la forme extérieure de ce que l'utilisateur a besoin d'exprimer. Les parties sémantiques explicitent les résultats que l'utilisateur peut attendre de ce qu'il a écrit et précisent la conduite tenue par le système dans la plupart des cas d'erreur.

## 6.1. LANCEMENT DES DIFFERENTES PHASES

### 6.1.1. La Grammaire

Les diverses phases de travail sont demandées dans un ordre quelconque à l'exception de l'exécution qui doit avoir été précédée des trois phases de définition. Elles peuvent en outre être demandées indépendamment les unes des autres au cours de tâches - au sens système - différentes. Le début et la fin de chacune d'elles sont marqués par un mot clé.

L'emplacement de la phase d'exécution après les trois définitions n'est qu'une contrainte sémantique que la syntaxe n'a pas à prendre en compte : il peut arriver qu'une tâche se limite à l'exécution sur des données et des méthodes déjà définies. On peut même, dans une tâche unique, faire suivre l'exécution par des modifications des données ou des dictionnaires en vue d'une autre exécution. Bien entendu nous vérifions à l'interprétation qu'aucune exécution de méthode n'est demandée sans que la base correspondante (identifiée par un numéro), les dictionnaires et les données nécessaires, soient présents.

A ce niveau, la grammaire se limite donc à :

```
tâche-abrège = définition - exécution+ ***
définition-exécution = def-méthode | defmodif-dico |
                        defmodif-données | demande-exécution
```

Au niveau 1, chaque phase est encadrée de mots-clés spécifiques précédés de deux étoiles. D'où :

```
définition-méthode = **METHODE description-de-méthode **FMETHODE
def-modif-dictionnaire = **DICTION definit-modifie-supprime-des-dictionnaires
                        **FDICTION
def-modif-données = **LIBELLE definit-modifie-supprime-des-libellés **FLIBELLE
demande-exécution = **EXECUTE description-de-demande-d'exécution **FEXECUTE
```

Les règles "description-de-méthode" etc sont des règles qui font uniquement intervenir le deuxième niveau du langage. Nous les explicitons donc par la suite.

### 6.1.2. Interprétation d'une tâche

Les deux niveaux de description d'une tâche sont reflétés par la structure de l'interprétation. L'interprétation de niveau I est gérée par un automate d'états finis - que nous avons baptisé AUTO-AUTO, qui appelle lors de la reconnaissance du mot clé correspondant l'automate-interpréteur spécifique de chaque phase.

AUTO-AUTO gère donc l'appel de quatre automates d'analyse de niveau II

- un automate de METHODE
- un automate de DICTIONNAIRE
- un automate de LIBELLE
- un automate d'EXECUTION.

Cet automate principal reprend le contrôle chaque fois que l'on rencontre en entrée un mot-clé commençant par deux étoiles.

### 6.1.3. Erreurs de syntaxe dans l'enchaînement des phases

Les erreurs que nous considérons ici sont uniquement celles détectées par l'automate principal. Elles sont de deux natures :

- oubli de fermeture d'une phase
- oubli de définition de la nature d'une phase.

La première erreur - oubli de \*\*FMETHODE, \*\*FLIBELLE ... - est détectée si l'on rencontre l'introduction d'une nouvelle phase avant la clôture de la précédente. Dans ce cas la phase en cours est terminée, autant que possible, et la nouvelle phase est lancée après édition d'un message.

La deuxième erreur est fatale à la phase non identifiée. Cependant ceci peut se produire dans des situations d'interprétation différentes.

Dans la première situation, c'est au début de la tâche ou après la marque de fin d'une phase que l'on détecte l'absence du mot-clé d'identification de la phase. Dans ces conditions, le texte est balayé jusqu'à l'apparition d'une marque de fin de phase (sans retour en arrière une fois que le contexte a pu être précisé) et un message est édité.

Dans la deuxième situation, l'absence d'identification de phase est masquée par l'absence simultanée d'une marque de fin de la phase précédente. C'est la situation dans l'exemple suivant.

```

**FMETHODE
      | description de méthode
double erreur ---- |
                   +
                   + éléments de création de dictionnaire
                   +
**FDICTION
```

Nous nous trouvons alors en cours d'interprétation dans un automate spécifique de la première phase. Cette interprétation ne sait pas se poursuivre sur la suite "création de dictionnaire" et détecte une anomalie à chaque phrase du langage qui décrit une action sur les dictionnaires. Le balayage avec message pour chaque anomalie continue jusqu'à retour du contrôle à l'automate principal.

En fait dans ce cas, cela revient à dire que les automates spécifiques n'apprennent pas d'une erreur à l'autre qu'ils ont "quitté" leur domaine de définition. Ce n'est pas très important car une telle erreur tient du paradoxe ou bien est le fruit d'une destruction anormale des éléments décrivant une tâche.

Nous décrivons maintenant chaque phase de niveau II dans l'ordre logique de construction d'un essai

- définition de méthode
- définition de dictionnaires
- saisie des données
- demande d'exécution.

Une dernière remarque s'impose, liée au format d'entrée d'une description de méthode. Celui-ci est libre, à la seule condition de respecter les contraintes suivantes :

- ne pas écrire de ligne de description de plus de 60 caractères
- ne pas couper de mot ou de nombre en allant à la ligne.



Nous avons imposé cette règle pour prévoir la saisie de descriptions en mode conversationnel par télécype ou console à écran. La saisie actuelle a cependant lieu par perforation de cartes d'après des bordereaux de format libre.

## 6.2. LANGAGE DE DESCRIPTION DE METHODE

Si nous avons vu aux chapitres 2 et 3 quelles serait pour nous la structure de description d'une méthode, nous n'y avons pas décrit le langage qui permet à l'utilisateur de s'exprimer. Nous présentons ici la grammaire totale qui reflète à la fois les éléments spécifiques des caractères (chapitre 2) et ceux des propriétés phonétiques des mots (chapitre 3). Nous en explicitons la sémantique par des exemples ponctuels, puis en traduisant en Abrégé les méthodes que nous avons décrites dans les chapitres précédents.

### 6.2.1. La grammaire de description des méthodes

(Les symboles spéciaux, les mots-clés écrits en majuscules et les variables numéro, nombre et identificateur sont considérés comme terminaux).

- 1 description-de-méthode = \*numéro options antidicos<sup>0/1</sup> filtre-libellé<sup>+</sup>
- 2 options = [\*TIRET | \*ESPACE]<sup>0/1</sup> \*DOUBLET<sup>0/1</sup>  
[\*COLLISION [numéro|CHIFFRE]]<sup>0/1</sup> [\*FG identificateur<sup>+</sup>]<sup>0/1</sup>  
[\*LONG ≤ nombre]<sup>0/1</sup>.
- 3 antidicos = \*ANTIDIC identificateur<sup>+</sup>.
- 4 filtre-libellé = \*LIB [\*NBMOT longueur]<sup>0/1</sup> filtre-mot<sup>+</sup>.
- 5 filtre-mot = \*MOT position<sup>0/1</sup> fréquence<sup>0/1</sup> taille<sup>0/1</sup> compteur\*  
[dicos|dicos calculs|calculs].
- 6 dicos = \*DIC identificateur<sup>+</sup>.
- 7 calculs = filtre-syllabe<sup>+</sup> | filtre-phonème<sup>+</sup> | filtre-lettre<sup>+</sup>.
- 8 position = \*POS expression.
- 9 expression = ope expr-simple | DE expr-simple A expr-simple.
- 10 fréquence = Natfreq [ope valfreq | DE valfreq A valfreq | \*MIN | \*MAX]

- 11 natfreq = \*FREQG | \*FREQR | \*FREQL | \*FREQM.  
12 valfreq = nombre [% | % MIN | % MAX]<sup>0/1</sup>.  
13 taille = unité longueur.  
14 unité = \*NBSYLL | \*NBPHON | \*NBLETR.  
15 longueur = ope nombre | DE nombre A nombre.  
16 expr-simple = nombre | L/nombre | L-nombre | L.  
17 ope = < | ≤ | = | ≥ | >.  
18 compteur = \*X nombre expression.  
19 filtre-syllabe = \*SYLL taille<sup>0/1</sup> position<sup>0/1</sup> compteur\* actions<sup>0/1</sup>.  
20 filtre-phonème = \*PHON taille<sup>0/1</sup> position<sup>0/1</sup> type<sup>0/1</sup> fréquence<sup>0/1</sup>  
compteur\* actions<sup>0/1</sup>.  
21 filtre-lettre = \*LETR position<sup>0/1</sup> type<sup>0/1</sup> fréquence<sup>0/1</sup> compteur\* actions<sup>0/1</sup>.
- Remarque : Le choix entre ces trois types de filtre est déterminant pour le niveau de travail de la méthode : syllabique, phonétique ou littéral, ainsi que l'exprime la règle 7.*
- 22 actions = choix\* modif-compteur\* fonction\* \*BOUT<sup>0/1</sup>.  
23 choix = [\*TROUVER | \*PRENDRE] \*FIXE<sup>0/1</sup> [\*N ≤ expr-simple]<sup>0/1</sup>  
[\*DEPUIS expr-simple]<sup>0/1</sup> modif-compteur\*  
[type | \*SY | \*PH | \*LT | \*( xxxxx )\*].  
24 type = \*PV | \*PC | \*PCS | \*PCE  
\*SOU | \*SON | \*NAS  
\*VO | \*CS  
25 modif-compteur = [\*A | \*Z | \*D] Xnombre.  
26 fonction = \*F identificateur.

### 6.2.2. Sémantique d'une description

Règle 1 : Une méthode est identifiée par son numéro - \*numéro -

Nous avons limité à 10 le nombre de méthodes pouvant être présentes simultanément dans la base.

En tête d'une méthode viennent ensuite les 'options' globales à celles-ci, puis la liste des antidictionnaires utilisés, enfin les divers filtres portant sur les libellés, qui vont différencier les traitements des libellés entre eux.

L'ordre de définition des filtres-libellés est un ordre de priorité pour l'exécution de la méthode. Le premier "bon" est celui pris en compte. Dans ce cas, les suivants sont inutilisés pour le "libellé courant".

Exemple règle 1 :

```
**METHODE *4 *TIRET *LONG  $\leq$  12 *ANTIDIC ANTIDIC ANTISPE
  *LIB ... (traitement des libellés satisfaisant le premier filtre)
  *LIB ... (traitement des libellés satisfaisant le 2ème filtre et
            pas le premier)
  *LIB ... (traitement des libellés satisfaisant le 3ème filtre et
            ni le 1er ni le 2ème)

**FMETHODE
```

Règle 2 : Options

Les options globales sont de deux types : les options standard implémentées en permanence dans le système et celles (\*FG) spécifiques à un utilisateur et implémentées à sa demande. Toutes sont significatives de traitements exécutés lors de la création de l'abréviation finale d'un libellé à partir des abréviations de chaque mot dont on a retenu des caractères.

- \*TIRET = un tiret '-' est introduit entre les abréviations élémentaires des mots. Ainsi
- "règle de gestion" (libellé) abrégée après traitement sur les mots en 'RGL' et 'GEST' (abréviations élémentaires) donne comme abréviation finale 'RGL-GEST' au lieu de 'RGLGEST'.*
- \*ESPACE = un espace est introduit entre les abréviations élémentaires des mots. Cette option, ainsi que l'indique la grammaire, est exclusive de la précédente.
- \*DOUBLET = lors de la concaténation des abréviations élémentaires des mots entre eux, la chaîne de caractères finale est balayée de manière à supprimer tous les doublets de caractères identiques.
- Exemple : "Récapitulation des principes de calculs" abrégé en 'RECAP' 'PRINC' 'CALC' donnerait avec cette option 'RECAPRINCIPALC' et non 'RECAPPRINCCALC'.*
- Les doublets internes aux abréviations élémentaires des mots sont eux-mêmes éliminés.*
- 'Apparition' abrégé en 'APPARIT' donnerait 'APARIT'.*

\*COLLISION numéro (a) ou \*COLLISION CHIFFRE (b) au terme du calcul des abréviations sur le jeu d'essai, les collisions d'abréviation pour la méthode exécutée sont connues grâce au chaînage des libellés initiaux. Cette option tente la suppression des collisions. C'est en fait une fonction générale qu'un utilisateur a demandé et que nous avons implantée de manière permanente dans le système.

- a) \*COLLISION numéro : le numéro indiqué est celui d'une méthode de calcul qui va servir à *recalculer* les abréviations litigieuses pour les seuls libellés concernés. Un nouveau bilan de collision est effectué au terme de cette démarche sans qu'une autre action soit tentée, en cas de nouvelles collisions.
- b) \*COLLISION CHIFFRE : les libellés qui ont produit des collisions sont chaînés entre eux. Les collisions sont supprimées par balayage de cette chaîne et ajout d'un chiffre à chaque abréviation dans l'ordre de balayage.

*Exemple : 'Liste des administrateurs'  
et 'Liste des administrations' ayant produit tous deux  
'LIST-ADM' sont abrégés  
'LIST-ADM1' pour le premier rencontré  
'LIST-ADM2' pour le deuxième.*

#### \*FG identificateur

L'identificateur est celui d'une fonction spéciale à l'utilisateur. Lors de l'exécution, la présence dans la base des fonctions citées est vérifiée (3 au maximum). En cas d'absence l'exécution n'est pas entreprise.

Ces fonctions peuvent être de deux natures :

- a) A même titre que COLLISION, ce peut être un travail supplémentaire lors de la formation ultime de l'abréviation. Elles s'intègrent alors au niveau du calcul final des abréviations.

*Citons pour exemple une fonction EUPHONIE qui, prenant les abréviations calculées de chaque mot du libellé en cours de traitement, irait voir si d'un mot à l'autre il y a alternance consonne-voyelle. Lorsque cette alternance n'est pas réalisée, la fonction tenterait de le faire par ajout (en prenant la dernière lettre du mot) par suppression de lettre dans l'abréviation ou par insertion de tiret en cas d'impossibilité.*

b) Ce peut être aussi une fonction de prétraitement de libellé, par exemple pour ajouter un traitement spécial de sigle ou de chiffres et éviter d'exécuter la méthode pour ces mots, sans pour autant faire appel à des dictionnaires.

L'insertion de ces fonctions suppose une bonne connaissance du programme qui interprète les méthodes dans lequel nous prévoyons donc quelques points d'appels de fonctions. Elles sont évidemment à écrire en Socrate. L'intérêt de leur déclaration explicite dans la méthode est de mettre en évidence les ajouts au noyau central.

\*LONG  $\leq$  nombre : limite au nombre indiqué la longueur ultime de l'abréviation par troncature de celle-ci.

*La longueur maximum actuellement retenue est de 30 caractères. Nous n'avons pas implémenté les conséquences de la remarque 3.8.3 sur le traitement mot à mot au Chapitre 3. Les caractères tronqués ne sont pas conservés.*

La longueur de l'abréviation définitive est à la fois déterminée par les abréviations élémentaires des mots et par les fonctions spéciales, les options tirets ou espace. La troncature n'est donc effectuée que lorsqu'elles ont été effectuées.

De même, lors de la suppression des collisions par ajout de chiffres, les chiffres ne doivent pas être tronqués.

*Ainsi, si l'abréviation est limitée à 8 caractères et que l'on a collision (éventuellement après troncature) sur des mots de 8 caractères, on supprimera un ou deux caractères de plus pour permettre l'ajout des chiffres.*

*Alors*

*ADMICENT pour 'administration centrale'*

*ADMICENT pour 'administrateur central'*

*sont transformés par le jeu des collisions chiffrées en*

*ADMICEN1*

*ADMICEN2.*

Règle 3 : antidicos = \*ANTIDIC identificateur<sup>+</sup>

En tête de la méthode sont annoncés les antidictionnaires qui vont être utilisés sur tous les libellés. Lors de la déclaration descriptive de la méthode, ils sont cités par leur nom sans aucune vérification d'existence au niveau de la base Abrège (conséquence de la liberté que nous avons laissée dans l'écriture des phases préparatoires de l'exécution).

Leur existence en tant qu'antidictionnaires (car dans la base ils sont partie constituante de l'ensemble des dictionnaires) est contrôlée au début de l'exécution de la méthode. En cas d'absence, l'exécution n'est pas entreprise.

Leur emploi est le suivant : Pour chaque libellé, supprimer les mots non significatifs, c'est-à-dire ceux qui appartiennent à l'un des antidictionnaires cités, pris dans l'ordre d'écriture (qui est une priorité). Au niveau de l'exécution, le libellé initial est alors recopié sans les mots non significatifs dans un libellé "de travail". La longueur du libellé alors considérée est celle du libellé de travail limité aux seuls mots significatifs, dont la position dans le texte se trouve aussi modifiée. Ces dernières valeurs sont celles prises en compte dans le calcul des prédicats.

Le libellé initial est inchangé (il y a conservation des données).

*Exemple* : Dans l'exemple que nous proposons pour la règle 1, nous citons deux antidictionnaires ANTIDIC et ANTISPE.

*Prenons le libellé de 7 mots suivants*

"CONTROLE PAR LE TRESOR, VISA, PRECOMPTE EVENTUEL"

1 2 3 4 5 6 7

*Si 'PAR' et 'LE' appartiennent à ANTIDIC et que EVENTUEL a été stocké dans ANTISPE par un utilisateur désireux de supprimer les adjectifs, le libellé de travail devient*

"CONTROLE TRESOR VISA PRECOMPTE"

1 2 3 4

*avec les déplacements de position que nous avons notés. C'est à partir de ce libellé réduit que s'effectuent ensuite tous les calculs de la méthode, y compris l'exécution des options citées ci-dessus.*

ANTIDIC est un nom "réservé" à l'antidictionnaire standard que nous proposons avec Abrège. Il contient les articles, pronoms et prépositions, ainsi que les locutions usuelles comme "en vue" qui devront être écrites d'un seul

tenant lors de la saisie des données "envue" pour éviter la suppression de mots significatifs "vue". Voir Annexe [A4] pour plus de détails.

Nous avons limité à 10 le nombre total des dictionnaires et antidictionnaires résidant simultanément dans la base (ANTIDIC est compris dans ce chiffre).

La règle pourrait donc se réécrire :

antidicos : \*ANTIDIC identificateurs<sup>1/10</sup>

mais ceci est une simple limite d'implémentation, qui n'est pas fondamentale pour la grammaire.

Règle 4 : filtre-libellé = \*LIB [\*NBMOT longueur]<sup>0/1</sup> filtre-mot<sup>+</sup>.

Cette règle exprime ce que nous avons dit au chapitre 2. Le seul prédicat caractérisant le libellé est un prédicat de longueur. Ce prédicat n'est pas obligatoire : s'il est absent, le traitement qui suit est toujours appliqué.

Ainsi la méthode suivante :

```
**METHODE*2*LIB  *NBMOT < 5 (*MOT ... *SYLL ... filtre-mot)
                  *LIB  *NBMOT DE 5 A 10 (*MOT ... *SYLL ...)
                  *LIB                               (*MOT ... *SYLL ...)
```

*contient des règles pour les libellés de moins de 5 mots,  
de 5 à 10 mots et pour tous les libellés de longueur supérieure à 10.*

L'ordre d'exécution n'est pas indifférent car les filtres-libellés sont testés dans leur ordre d'écriture. Aussi la méthode suivante serait syntaxiquement exacte mais sémantiquement bien pauvre.

```
**METHODE*3*LIB  (*MOT ... *PHON ... filtre-mot)
                  *LIB  *NBMOT < 5 ...
                  *LIB  *NBMOT DE 5 A 10 (...).
```

*En effet seul le premier filtre est toujours utilisé puisque son prédicat est vide, donc vrai pour tout libellé. Le traitement qui correspond à ce filtre est alors utilisé sur tous les libellés, et les autres ne servent jamais.*

Aucune vérification n'est effectuée à la déclaration d'une méthode pour reconnaître la disjonction des prédicats et leur "bon ordre" pour l'interprétation (ce qui explique la possibilité du type d'erreur sémantique ci-dessus).

Expression du prédicat de longueur : Après rappel de l'unité qui est la seule reconnue pour la longueur d'un libellé -le mot-, "longueur" est une règle qui exprime une ou deux bornes dont les valeurs sont des entiers.

Règle 15 : longueur = ope nombre | DE nombre A nombre

Nombre est un entier dont la valeur minimale est 1 et la valeur maximale est 40 d'après l'implémentation actuelle de la base.

Par ailleurs dans la deuxième forme de l'expression de longueur vérifie, lors de son interprétation, que la première borne exprimée est inférieure à la deuxième.

*Les contraintes d'implantation limitent actuellement à 10 le nombre de filtres-libellés présents dans la description d'une méthode.* Ce nombre choisi comme une borne "raisonnable" peut être facilement modifié par recompilation de la base si l'usage nous conduit à le considérer comme insuffisant. C'est une limite dans le stockage des descriptions de méthode qui n'apparaît qu'au niveau "structure" de la base.

-Règle 8-      -Règle 10-      -Règle 13-      -Règle 18-

Règle 5 : Filtre-mot = \*MOT position<sup>0/1</sup> fréquence<sup>0/1</sup> taille<sup>0/1</sup> compteur<sup>0/n</sup>  
[dicos | dicos calculs | calculs]

a) Le filtre comporte les 4 prédicats annoncés aux chapitres 2 et 3 (taille, position, ...). Cependant lorsqu'ils sont toujours vrais (donc "indifférents") ils sont omis dans l'écriture du filtre. Le filtre-mot est alors général à tout mot et le calcul qui suit ou les dictionnaires dont l'emploi est demandé sont utilisés pour tout "mot" consultant ce "filtre".

Exemple :

*"Pour tout mot prendre la première syllabe" s'exprimerait*  
\*MOT \*SYLL \*POS=1 \*PRENDRE \*SY  
calcul

b) Par ailleurs, la partie traitement différente des filtres [dicos | dicos calculs | calculs] ne peut être vide. Il peut y avoir définition seulement de dictionnaires, ou uniquement de calculs, mais l'une des deux options de traitement doit être présente. Elles sont reconnues par la présence des mots-clés



\*DICO

ou \*LETR ou \*SYLL ou \*PHON

Le détail des filtres et des traitements est explicité au niveau des règles correspondantes.

Lorsqu'à la suite d'un filtre-libellé plusieurs filtres-mots sont présents, l'exécution se déroule de la manière suivante :

- pour tout mot du libellé en cours,
- rechercher le premier filtre-mot qui passe (suite des prédicats vraie) et exécuter le traitement défini :
- par recherche dans les dictionnaires s'il y en a
- par calcul si la recherche a été négative ou n'était pas définie.

De même que nous avons limité la description d'une méthode à 10 *filtres-libellés*, nous avons considéré que *chaque filtre-libellé ne comporte pas plus de 15 filtres-mots*.

Règle 6 : dicos = \*DIC identificateur<sup>+</sup>

L'interprétation immédiate est équivalente à celle de la *règle 3*. On veut utiliser des dictionnaires. On les cite par leur nom. La vérification d'existence des dictionnaires cités n'a lieu qu'à l'exécution, avant l'entrée dans la méthode. Ainsi ne commence-t-on pas inutilement les passages dans les divers filtres.

10 dictionnaires et antidictionnaires seulement peuvent coexister dans la base.

A l'exécution, ayant pour un mot donné d'un libellé trouvé le premier filtre-mot "qui passe", si la *règle 6* a été utilisée pour ce filtre-mot, on recherche la présence du mot dans les dictionnaires cités.

Ce mot, comme tous les autres, a été à la saisie rangé dans MOTS-LIB, ensemble des mots contenus dans les libellés. Parmi les caractéristiques de l'entité qui lui correspond, se trouve une "chaîne d'inverses", chaîne de 10 bits qui indique la présence ou l'absence du mot dans les dictionnaires des utilisateurs. (mise à jour lors de la saisie des libellés et des dictionnaires).

En cas de présence dans le dictionnaire cité dans la règle, celui-ci est consulté. Il fournit pour le mot la chaîne de caractères que l'utilisateur veut voir utilisée comme équivalent dans l'abréviation. Cette chaîne peut éventuellement être réduite à "un blanc", ce qui revient à supprimer la représentation du mot dans l'abréviation finale (abréviation vide).

Les dictionnaires ont pour ordre de priorité d'emploi l'ordre de citation. Si le mot n'est pas présent dans le premier dictionnaire cité, on le recherche dans les suivants jusqu'à épuisement de la liste.

Lorsque la *règle 5* (filtre-mot) présente la forme

**\*MOTS ... filtres ... dicos calculs**

La partie calcul est inutilisée si le mot est présent dans l'un des dictionnaires cités : on passe au mot suivant du libellé.

La partie calcul est effectuée si aucun des dictionnaires cités ne contenait le mot en cours.

Lorsque la *règle 5* ne propose que des dictionnaires, mais qu'aucun d'eux ne contient le mot courant, celui-ci est abandonné et le traitement passe au mot suivant du libellé courant.

Règle 7 : Calculs = filtre-syllabe<sup>+</sup> | filtre-phonème<sup>+</sup> | filtre-lettre<sup>+</sup>

Nous voyons ici l'expression de ce que nous avons dit au chapitre 3. Pour un mot donné, le mode de calcul est de type "unique" parmi les trois proposés.

En effet lorsque le filtre-libellé est trouvé, lorsque le filtre-mot du mot courant est obtenu, un seul calcul est possible et il s'exprime par des suites d'actions, dont l'emploi est conditionné par des filtres tous de même nature : syllabe ou phonème ou lettre, exclusivement les uns des autres.

Les *règles 19, 20 et 21* expriment ce que nous avons dit aux chapitres 2 et 3 des prédicats propres à chaque type de calcul. Au niveau où nous nous trouvons, jonction entre la description et l'évaluation de prédicats pour caractériser l'élément clé de l'action, nous avons décidé d'approfondir l'expression propre de ces prédicats avant d'étudier les actions possibles (déjà citées en 2 et 3).

Nous avons donc présenté la grammaire dans cet esprit. Les règles 8, 9, 10, 13, 18 expriment la syntaxe des prédicats précisée par les règles intermédiaires. Les contraintes d'implantation limitent à 20 les variantes de calcul liées à un même *filtre-mot*, c'est-à-dire le nombre de *filtres-syllabe*, *filtres-phonèmes* ou *filtres-lettres* associés.

La règle 7 est donc implantée suivant cette restriction :

Calculs = filtre-syllabe<sup>1/20</sup> | filtre-phonème<sup>1/20</sup> | filtre-lettre<sup>1/20</sup>

Règle 8 : (ainsi que les règles 9 et 16) prédicat de position

\*POS expression

Suivant la nature du filtre dans lequel il est présent, ce prédicat caractérise la position du mot, de la syllabe, du phonème ou de la lettre.

Cette position peut être inférieure, supérieure ou égale [opel] à la borne exprimée dans expression ou comprise entre les bornes données - Règle 9 - Expression - qui fixent un intervalle fermé (la position égale à la valeur de l'une des bornes met le prédicat à vrai).

L'expression des bornes est simple. C'est un entier ou l'expression d'un déplacement entier relatif à la longueur de l'unité du filtre définie par le contexte.

*Ainsi, dans un filtre-mot*

*\*POS  $\leq$  10 est explicite, ainsi que*

*\*POS  $\geq$  L-5. L est alors sémantiquement interprété comme la longueur du libellé courant exprimée en mots.*

Dans un filtre-syllabe, la sémantique du prédicat "*\*POS DE 1 A L/2*" est : position d'une syllabe située dans la première moitié du mot, la longueur de celui-ci étant mesurée en syllabes. La division ici est en fait une division entière de la longueur.

On voit que cette sémantique est simple mais permet d'exprimer des prédicats de position variables avec la longueur.

Nous effectuons sur les bornes proposées quelques vérifications simples dans la mesure où elles sont possibles, c'est-à-dire seulement sur les valeurs de bornes entières, les autres n'étant interprétables qu'à l'exécution.

Nous vérifions que :

- lorsque deux bornes sont exprimées (forme DE ... A ...), la plus petite est la première
- une borne ne dépasse pas le champ des valeurs possibles (par implémentation) de la position :
  - dans le contexte mot, cette valeur est de 40 (mots par libellé)
  - dans le contexte syllabe, c'est 10 (syllabes pour un mot)
  - dans le contexte phonème, c'est 20 (phonèmes pour un mot)
  - et au maximum 30 caractères pour un mot.

Aucune vérification n'est effectuée quant à la disjonction des prédicats de position pour des filtres de même niveau syntaxique (filtres-mots ou autre) et sémantique. En effet, les prédicats de position peuvent être utilisés conjointement à d'autres types de prédicat (longueur, etc). Nous respectons définitivement la règle d'exécution "Premier vrai, premier utilisé".

Règle 10 : Fréquence = Natfreq [lope valfreq | DE valfreq A valfreq | \*MIN | \*MAX]

La forme extérieure de l'expression d'un prédicat de fréquence est apparemment proche de celle des prédicats de position.

Nous sommes ici obligés de préciser la nature de la fréquence prise en considération. Nous avons retenu quatre types de fréquence - Règle 11 -

- \*FREQG : une fréquence générale extrinsèque aux données. C'est la fréquence :
  - . d'un mot en français, choisie par référence littéraire
  - . d'une lettre en français.
- \*FREQR : fréquence d'occurrence de l'unité considérée (mot, phonème, lettre) relative à l'ensemble des libellés stockés. Elle est évaluée à la saisie des jeux d'essai.
- \*FREQL : la fréquence d'occurrence est exprimée ici relativement au libellé courant (pour mots, phonème ou caractères).
- \*FREQM : fréquence d'un phonème ou d'une lettre relative au mot.

Le type de l'élément pour lequel on veut utiliser la fréquence est défini contextuellement.

Lors de la définition d'une méthode, nous vérifions uniquement que le prédicat comporte une nature de fréquence cohérente avec le contexte :

\*FREQG n'existe que pour les lettres et les mots  
{ \*FREQR pour les mots, les phonèmes, les lettres  
  \*FREQL  
\*FREQM pour les phonèmes, les lettres.

Aucun prédicat de fréquence ne caractérise les syllabes, ainsi que l'indique la syntaxe de la règle 19, pour les raisons que nous avons données au chapitre 3.

Comme pour la position, le prédicat de fréquence peut être défini en comparant la valeur de la fréquence de la variable courante à une ou deux bornes.

A ce schéma simple, nous avons ajouté deux fonctions, l'une de minimum, l'autre de maximum.

*Dans le contexte mot, l'expression suivante "\*MOT ... \*FREQG \*MIN" signifie : "Mots du libellé courant dont la fréquence d'utilisation en français est la plus faible".*

Le champ d'évaluation de ces deux fonctions est celui du contexte englobant l'unité en cours de traitement.

Pour un mot, \*MIN (ou \*MAX) signifie minimum (ou maximum) de la fonction de fréquence parmi toutes les valeurs prises sur les mots du libellé.

Pour un phonème ou une lettre, c'est le minimum (ou maximum) de la fonction de fréquence parmi toutes les valeurs prises sur le mot courant.

L'expression des bornes qui définit le champ de valeur des fréquences pour le prédicat est très simple. C'est une valeur donnée en absolu ou en relatif, *valfreq - Règle 12 -*

Fréquence absolue : c'est, en fait, le nombre réel d'occurrences de l'élément considéré, non relativisé par les autres occurrences des autres éléments. Il est évidemment impossible de parler de fréquence absolue dans la langue française. Nous disposons de ce nombre d'occurrences pour les mots, les phonèmes et les lettres sur l'ensemble des libellés, sur le libellé courant et pour les phonèmes et les lettres sur le mot courant. Néanmoins, nous pensons que cette variable n'a de sens que pour :

- le nombre d'occurrences d'un mot dans un libellé,
- le nombre d'occurrences de lettres ou de phonèmes dans un mot.

Nous n'avons cependant pas enregistré de contrainte sémantique correspondant à cette remarque. Aucune "erreur" n'est donc signalée pour une citation peu significative mais que nous savons évaluer.

Fréquence relative : nous proposons de considérer trois expressions de cette fréquence. L'une, directe, est exprimée comme un pourcentage. Pour des raisons de simplicité, nous avons limité ce pourcentage à une expression entière (1 % ou 30 %, mais pas 1,5 % ou 0,5 %).

Les deux autres expressions sont données relativement aux minimum et maximum de toutes les valeurs prises par la fréquence sur l'ensemble considéré.

Nous acceptons ainsi les diverses expressions suivantes, dont nous détaillons le sens, en les resituant dans leur contexte de *filtre-mot*, *filtre-phonème* ou *filtre-lettre*.

*\*MOT ... \*FREQG > 10 % .. : pour les mots dont la fréquence en français est supérieure à 10 %*

*\*MOT ... \*FREQL DE 3 A 10 ... : pour les mots présents 3 à 10 fois dans le libellé courant*

*\*MOT ... \*PHON ... \*FREQL  $\geq$  90 % MAX ... : pour les phonèmes du mot courant*

*dont la fréquence est supérieure à 90 % du maximum de fréquence atteint par un phonème sur le libellé courant.*

*\*MOT ... \*LETR ... \*FREQM \*MIN : pour la ou les lettres les moins fréquentes du mot courant.*

### Règle 13 : Taille : unité longueur

Cette règle exprime le prédicat de longueur. L'unité est en syllabes phonèmes ou lettres - Règle 14 -

La longueur d'un mot peut s'exprimer suivant les trois unités, tandis que celle d'une syllabe n'a de sens qu'en phonèmes ou lettres et celle d'un phonème en lettre.

La vérification de cohérence de l'unité avec la nature du filtre est effectuée sémantiquement.

L'expression des bornes auxquelles la longueur de l'élément courant est comparée est simplement un entier - Règle 15 -. Nous vérifions la cohérence entre la valeur de la borne et les limites liées à l'implantation de la base : mots de 30 caractères au plus, de 15 syllabes, de 20 phonèmes, syllabes de 5 phonèmes au plus et 5 lettres, phonèmes de 5 lettres au plus.

Les prédicats de longueur sont donc par exemple :

- \*MOT ... \*NBSYLL DE 3 A 6 ... : mot de 3 à 6 syllabes*
- \*MOT ... \*NBPHON < 5 ... : mot de moins de 5 phonèmes*
- \*MOT ... \*SYLL ... \*NBPHON  $\geq$  3 ... : syllabe du mot courant de 3 phonèmes et plus*
- \*MOT ... \*PHON ... \*NBLETR > 2 ... : phonème de plus de 2 lettres.*

#### Règle 18 : Compteur = \*xnombre expression

Nous avons étudié dans la règle 8 la syntaxe des expressions - règle 9 -. La sémantique de l'expression de compteur est très élémentaire car nous ne pouvons prévoir les valeurs prises ; nous avons simplement pensé que celles-ci seraient toujours positives.

Dans la base, nous avons limité à 10 le nombre de compteurs utilisés dans une méthode pour conserver une trace des actions. Les compteurs prennent leurs valeurs dans l'intervalle [0,100].

Si l'usage prouve la nécessité de compteurs négatifs, nous modifierons cet intervalle (car cette contrainte est encore une fois simplement implantée dans la structure de la base).

Enfin, remarquons que l'indépendance des traitements d'un libellé à un autre conduit à remettre tous les compteurs à zéro au début du traitement d'un libellé. Dans tous les autres cas, la gestion des compteurs est assurée par l'utilisateur lorsqu'il exécute des actions "modif-compteur", c'est-à-dire dire lors du passage d'un mot à un autre dans le libellé en particulier.

#### Règles 19 - 20 - 21 : Filtre-syllabe, filtre-phonème, filtre-lettre

La syntaxe de chacun de ces filtres reflète l'analyse des divers prédicats possibles suivant la nature du filtre faite en 3.5. Les prédicats de position, fréquence, longueur et compteur viennent d'être précisés pour tous les contextes de filtre où ils sont utilisés. Seul, reste à

préciser le prédicat de type - *règle 24* - qui est la citation d'un mot de type dont nous vérifions qu'il est cohérent avec la nature du filtre : type pour un phonème dans un *filtre-phonème* ou type pour une lettre dans un *filtre-lettre*.

Nous ne donnons pas d'exemple d'application de ces règles, car l'expression est très similaire de celle des filtres-mots et les exemples complets de méthodes traités dans la suite de ce chapitre proposent plusieurs filtres correspondant à ces règles.

Règle 21 : Actions = choix\* modif-compteur\* fonction\* \*BOUT<sup>0/1</sup>

Les actions consécutives à la reconnaissance d'une "unité de travail" (syllabe, phonème, lettre) pour laquelle les filtres passent doivent fondamentalement conduire à choisir des caractères pour l'abréviation. Elles sont exécutées dans l'ordre de description, c'est-à-dire suivant l'ordre correspondant aux termes de la *règle 21*.

Nous détaillons au niveau de la *règle 22* la syntaxe et la sémantique des "choix".

Signalons simplement que nous limitons à 4 les choix possibles dans un même contexte de filtre.

Que des choix aient été décrits ou pas, les "modif-compteur" témoignent du passage par les filtres qui ont conduit à ce niveau. Les modifications peuvent concerner, par contrainte d'implantation, 3 compteurs seulement.

Les "fonctions" permettent l'insertion dans la base standard de programmes spécifiques à l'utilisateur pour le traitement des mots - *Règle 26*.

Comme pour les fonctions générales - *Règle 2* -, leur existence dans la base n'est vérifiée qu'au moment de l'exécution.

A ce niveau, l'utilisateur peut mettre en oeuvre au plus 3 fonctions parmi les 10 dont nous lui autorisons la définition.

Si l'action BOUT est présente, elle correspond à une demande d'inversion du sens de balayage avec positionnement de l'index de l'unité courante à la fin du mot.



*Par exemple :*

*Si l'exécution de toutes les actions qui ont précédé \*BOUT a placé l'index courant sur la troisième syllabe d'un mot de 10 syllabes, l'exécution de \*BOUT place l'index sur la dixième syllabe et spécifie que le balayage du mot se fait dorénavant pour toute évaluation de filtre et toute action, de droite à gauche jusqu'à la troisième syllabe incluse.*

Cependant, lorsque l'exécution de la méthode conduit à rencontrer plusieurs fois \*BOUT sur un même mot, nous vérifions simplement que les conditions de travail témoignent déjà de la prise en compte de cette action.

*Dans le contexte de l'exemple ci-dessus, si nous rencontrons à nouveau \*BOUT alors que l'index est placé sur la 5ème syllabe par un balayage de droite à gauche depuis la 10ème, nous poursuivons l'évaluation des filtres et l'exécution des actions dans l'ordre 5ème, 4ème, puis 3ème syllabe qui est ici la syllabe terminale de l'exécution, sans modifier le sens de balayage.*

Dans la sémantique de cette règle, nous voyons que les "actions" efficaces relativement à la sélection de caractères sont les "choix" et éventuellement les "fonctions". Dans le cas où la réalisation de la règle ne comporte ni choix ni fonction active relativement à la progression de l'index de position de l'unité courante, nous considérons que l'action est l'action VIDE (elle est vide relativement à la sélection de caractères) et nous faisons à l'exécution progresser l'index de l'unité courante (syllabe, phonème, lettre) de un dans le sens de balayage en cours.

Une telle action modifie donc le contexte qui a été défini par les filtres, en faisant progresser l'index de telle sorte que la même syllabe ou le même phonème ne soient pas "filtres" deux fois de suite.

Nous donnons ici trois exemples d'enchaînement d'actions efficaces et d'actions conduisant à l'action "vide".

Exemple 1 :

\*PRENDRE \*SY \*Z X1 \*A X2 \*F EUPHONIE \*BOUT

Dans ce cas, toutes les actions sont citées. On prend la syllabe courante. Le compteur X1 est remis à zéro, tandis que X2 est incrémenté de 1. La fonction EUPHONIE spéciale à l'utilisateur est appliquée. On inverse le sens de balayage en conservant comme syllabe terminale la syllabe qui suit celle qui vient d'être entièrement retenue par l'action prendre, sauf si le sens de balayage a déjà été inversé : dans ce cas, la syllabe voisine à gauche de la syllabe retenue est le nouveau contexte d'évaluation de filtre. A moins évidemment que le calcul sur le mot ne soit terminé.

Exemple 2 : Action VIDE avec modification de compteur

\*A X3

Dans ce cas, le seul témoignage du passage dans le filtre courant est l'incrément du compteur X3.

L'index de position est déplacé vers l'unité voisine (le contexte n'est pas précisé dans cet exemple).

Le voisinage est "droit" si aucune action \*BOUT n'a été demandée dans les phases précédentes sur le mot courant.

Le voisinage est "gauche" sinon.

Exemple 3 : Action VIDE

Enfin, il est possible que l'action demandée soit uniquement l'action vide. C'est ce qui est détecté ici.

\*MOT \*PHON ...

\*PHON \*NBLETR  $\geq$  3 \*X1 > 2 \*PRENDRE \*PH \*Z X1 \*BOUT

\*PHON \*NBLETR = 1 \*X1 = 0

\*PHON \*NBLETR  $\geq$  2 \*POS = L \*X1 = 0 \*PRENDRE \*PH

Le troisième filtre-phonème conduit simplement à "passer" les phonèmes d'une lettre quand le compteur X1 est nul.

Nous avons donc pour la règle 21 précisé l'enchaînement des actions citées et la définition du mode de progression dans les cas les plus simples. La règle 23 définit les actions standard de recherche de caractères telles que nous les avons étudiées aux chapitres 2 et 3. Nous les re précisons partiellement ici.

Règle 23 : Choix = [ $*\text{PRENDRE} \mid * \text{TROUVER}$ ]  $*\text{FIXE}^{0/1}$  [ $*\text{N} \leq \text{expr-simple}$ ] $^{0/1}$   
[ $*\text{DEPUIS expr-simple}$ ] $^{0/1}$   $\text{modif-compteur}^*$   
[ $\text{type} \mid * \text{SY} \mid * \text{PH} \mid * \text{LT} \mid *(xxxxx)^*$ ]

Nous voyons qu'un choix est soit l'action PRENDRE, soit l'action TROUVER. Le contexte de recherche des caractères, précisé par les "compléments" du verbe, est pour PRENDRE l'unité courante - syllabe, phonème ou lettre dans un filtre-syllabe, filtre-phonème ou filtre-lettre respectivement -, l'unité courante et son voisinage, défini par le sens de balayage du mot pour TROUVER. Nous avons déjà partiellement analysé ceci aux chapitres 2 et 3. Nous reprenons ici le détail des compléments des choix, ainsi que la politique de sélection qui leur correspond.

Il y a cinq compléments :

- 1) =  $\text{FIXE}$  indique si l'action provoque ou non le déplacement de l'index de position.
- 2) = " $*\text{N} \leq \text{expr-simple}$ " exprime combien d'éléments sont recherchés.
- 3) = " $*\text{DEPUIS expr-simple}$ " après que l'on en ait rencontré  $k = \text{expr-simple}$ .
- 4) = " $\text{modif-compteur}^*$ " est la liste des modifications de compteurs provoquées par la sélection d'un élément et répétées à chaque nouvelle sélection.
- 5) = " $\text{type} \mid * \text{SY} \dots$ " définit la nature de ce que nous avons appelé un élément. Dans "type", le nom cité indique si l'élément "typé" est un phonème ou une lettre d'un type déterminé, tandis que  $* \text{SY}$ ,  $* \text{PH}$ ,  $* \text{LT}$  correspondent à n'importe quel syllabe, phonème ou lettre rencontré quel que soit son type. Par ailleurs, la chaîne "xxxxx" (par exemple "w x z y k") correspond à la recherche explicite d'un exemplaire de l'un de ces caractères dans l'élément. Ce complément du choix est obligatoirement présent, ainsi que le montre la grammaire. Il nous sert, en fait, d'indicateur de fin de définition d'action et "caractérise" tous les compléments antérieurs. Pour le type d'élément demandé, il est souhaitable que soit respectée la compatibilité avec le filtre qui a déterminé l'action. Compatibilité large, c'est-à-dire qu'un filtre-syllabe permet le choix de n'importe quel type d'élément, mais qu'il paraît incohérent de demander le choix d'une syllabe quand on a

analysé les caractéristiques d'une lettre par un filtre-lettre.

Nous avons donné la signification principale des divers termes. Nous les reprenons maintenant dans le contexte spécifique des verbes PRENDRE et TROUVER pour montrer les problèmes que peut poser leur emploi.

PRENDRE - Le filtre qui précède cette action a positionné l'index de début d'unité sur le  $i^{\text{ème}}$  caractère du mot.

a) Dans le contexte d'un *filtre-syllabe*, cela veut dire, en outre, que nous sommes au début de la  $k^{\text{ième}}$  syllabe du mot, dont la longueur est connue (dans la décomposition syllabique).

Dans ce cas, si le type est seul précisé, nous cherchons depuis le début de cette syllabe un *élément du type voulu*.

\*Si le type est \*SY, nous prenons toutes les lettres de la syllabe.

\*Si le type est \*PH, nous retenons les lettres du premier phonème.

\*Si le type est \*LT, nous retenons la première lettre de la syllabe.

\*Si le type est celui d'un phonème qualifié (\*PV par exemple, phonème voyelle), nous recherchons le premier de la syllabe et retenons ses lettres. S'il est absent, nous passons au choix suivant.

*Par exemple :*

*Si la syllabe est 'VAIL' de 'travail', le premier phonème-voyelle rencontré est A dans 'AIL' que nous retenons.*

*Si la syllabe est 'beaux', nous retenons les quatre lettres 'eaux' pour ô, en conservant même les muettes.*

*Dans le mot antre, les phonèmes reconnus seraient ã tr, et dans la deuxième syllabe 'tre' aucun phonème-voyelle ne serait reconnu.*

\*Si le type est une chaîne de 1 à 5 caractères \*(xxxxx)\*, on recherche dans la syllabe un caractère correspondant à l'un de ceux proposés.

*Pour les caractères \*(x y z k w)\* par exemple, le mot 'relaxez' verrait les lettres x et z sélectionnées pour sa dernière syllabe 'rez', si \*N=2, sinon seul le x est retenu.*

Par contre, pour un choix :

\*PRENDRE \*(f g h q o)\*

Sur le mot 'aggraver', dont on considère la deuxième syllabe 'ggra', un seul 'g' est retenu pour l'abréviation.

Dans le cas où des compléments \*N ... \*DEPUIS ... (k) sont exprimés, cela veut dire que l'on recherche n éléments lorsqu'on en a déjà trouvé (k). Ceci est essentiellement possible pour l'action PRENDRE dans le contexte actuel de filtre-syllabe.

*On peut, par exemple, rechercher deux phonèmes consonnes après le premier. Ceci s'exprime par :*

*\*PRENDRE \*N=2 \*DEPUIS 1 \*PC*

*Dans la deuxième syllabe du mot a gres te, traduit phonétiquement a | g r e s | t, les phonèmes r s sont sélectionnés et traduits en 'r' et 's', après avoir "passé le g".*

*Dans la deuxième syllabe du mot a ver ti, a | v e r | t i, seul le 'r' sera retenu car on rencontre la fin de la syllabe, contexte de l'action PRENDRE avant d'avoir trouvé deux phonèmes consonnes.*

En reprenant ces deux exemples, signalons que les modifications de compteur éventuellement demandées sont exécutées deux fois dans le premier exemple et une fois dans le deuxième.

*L'action \*PRENDRE \*N=2 \*DEPUIS 1 \*A X1 \*Z X2 \*PC  
aurait donc comme résultat*

*X1 ← X1+2 dans le premier exemple et X2 ← 0 (agreste)*

*X1 ← X1+1 dans le deuxième exemple et X2 ← 0 (averti)*

*Pour un mot comme anion a/n i ÷, la deuxième syllabe donnerait un résultat négatif pour ce choix. Les compteurs X1 et X2 dans ce cas conserveraient leur valeur initiale.*

Si \*FIXE est précisé, l'index de positionnement est conservé au début de la syllabe, sinon il est déplacé au début de la syllabe suivante si elle existe.

Le contexte de départ, donc le contexte du filtre-syllabe, est néanmoins conservé jusqu'à la fin de l'exécution de toutes les actions. Nous expliquons par la suite la sémantique propre à l'enchaînement des actions d'un même filtre.

b) Précisons le rôle des compléments de PRENDRE dans le contexte d'un filtre-phonème.

Ce qui a été isolé par les filtres est donc un phonème unique, éventuellement décomposable en lettres.

On ne peut donc pas demander de choisir une syllabe sur ce contexte même si la syllabe est réduite à ce phonème.

Si le type demandé est \*PH, tout le phonème, toutes les lettres qui le composent sont retenus pour l'abréviation.

Si le type est phonème qualifié, celui-ci est retenu s'il correspond à cette qualification.

Dans ces deux cas, les modifications de compteurs du choix sont effectuées une fois si le phonème est retenu, et inopérantes sinon.

Dans ces deux cas, il est impossible de préciser ce choix de compléments \*N ... et \*DEPUIS, car un seul élément peut être retenu.

On peut enfin rencontrer tous les compléments en recherchant certaines lettres dans un phonème, comme cela s'est présenté dans une syllabe.

Le contexte est simplement un peu plus réduit. Le rôle d'un tel détail de choix (lettre dans un phonème) peut parfois être de mettre en évidence des lettres muettes.

*On peut, par exemple, sur le dernier phonème d'un mot, effectuer le choix suivant*

*\*PRENDRE \*(n s r x z)\**

*sur un phonème filtré comme phonème-voyelle.*

Les conditions de sélection, de modification de compteur, d'avancement pour l'action suivante sont, au contexte près, prédéfinies en a).

c) Dans un cas de filtre-lettre, l'action PRENDRE se réduit à prendre ou négliger la lettre définie par le filtre.

La lettre est prise si elle est du type demandé, et dans ce cas, les modifications de compteur exprimées en complément de ce choix sont exécutées une fois chacune. La position en fin d'action est sur la lettre suivante, sauf si \*FIXE a été spécifié.

TROUVER - La différence de sémantique de cette action provient essentiellement du changement de contexte. Deux cas se présentent suivant le sens de balayage, c'est-à-dire suivant que dans l'exécution d'autres actions sur le même mot, on a rencontré la commande \*BOUJ ou pas.

Si le balayage est normal, c'est-à-dire de gauche à droite, le contexte de TROUVER est ainsi défini.

Dans la syllabe ou le phonème où se trouve l'index courant après l'évaluation du filtre, on tente la recherche des éléments demandés par les compléments de TROUVER. Si l'unité courante, syllabe, phonème ou lettre, ne suffit pas à répondre à la demande, on poursuit cette recherche sur le voisinage à droite de l'unité. On s'arrête lorsque la demande est satisfaite ou lorsque l'on rencontre la fin du mot.

Ceci a déjà été exprimé particulièrement dans les exemples du § 3.5.3.

Nous explicitons encore une fois les divers résultats possibles en exprimant quelques actions et leurs résultats dans divers cas :

- 1 - ... \*TROUVER \*LT à la suite d'un filtre-syllabe  
équivalait à prendre la première lettre de la syllabe. L'emploi de trouver est abusif car on n'a jamais besoin de sortir du contexte immédiat. L'action \*PRENDRE donne le même résultat.
- 2 - ... \*TROUVER \*SY à la suite d'un filtre-lettre par contre est faux car la notion de syllabe est inconnue lorsque le travail s'effectue pour les filtres au niveau des lettres.
- 3 - ... \*TROUVER \*N<3 \*DEPUIS 2 \*A X1 \*VO dans un contexte de filtre-syllabe demande la recherche de 3 voyelles (lettres) après qu'on en ait rencontré déjà 2 en incrémentant à chaque sélection le compteur X1. Pour les exemples suivants, nous le supposons initialement nul.

- Dans BEAUTE alors que le filtre a placé l'index au début de la première syllabe

*les lettres retenues sont U et E (le dernier)*

*à la fin X1 = 2*

*l'index de position pour l'action suivante est à la fin du mot.*

*Dans ce cas, lorsqu'on aura effectué toutes les actions qui suivaient le même filtre-mot, l'index sera effectivement à la fin du mot et le calcul terminé.*

- Dans DIMINUTION découpé en syllabes DI-MI-NU-TION, lorsqu'on est placé au début du mot

les 2 premiers I sont reconnus et négligés à cause de la clause \*DEPUIS 2, puis sont retenues les lettres U, I, O, avec  $X1 = 3$  et l'index de position à la fin du mot malgré l'existence de 4 syllabes car ce n'est pas avant d'être entré dans la quatrième que l'on a trouvé les voyelles cherchées.

- Dans AUXILIAIRE découpé en AU-XI-LIAI-RE par l'analyse syllabique les lettres retenues seraient -IIA- respectivement en quatrième, sixième et septième position dans le mot.

XI=3 à la fin de cette sélection.

l'index de position pour l'action suivante est sur le R du début de la dernière syllabe.

- Nous donnons ici l'exemple d'une action complexe qui suit un filtre-lettre, dont le contexte initial est donc très réduit

\*TROUVER \*N=2 \*DEPUIS 1 \*A X0 \*Z X1 \*CS

Trouver 2 consonnes après la première rencontrée. Si l'action a eu un résultat effectif, incrémenter X0 du nombre d'éléments trouvés, remettre X1 à zéro.

Dans BEAU, si nous supposons que nous sommes sur la première lettre au départ

aucun caractère ne peut être sélectionné

X0 n'est pas modifié

X1 non plus

l'index de position est à la fin du mot pour la fin de cette action.

Dans CONGRES, après avoir admis que le G sur lequel on est positionné du fait du filtre et d'actions antérieures, est une consonne

on retient R et S

X0 est augmenté de 2

X1 = 0

l'index de position est la fin du mot pour cette action donc pour les conditions de passage au filtre suivant.



*Dans CONGRESSISTE, dans les mêmes conditions que pour CONGRES, les résultats seraient les mêmes, sauf pour l'index de position qui serait sur le deuxième S pour la suite du travail.*

Nous désirons seulement répéter une condition d'exécution des actions qui est liée à leur contexte initial :

l'action porte sur des unités de même nature que le filtre ou de nature "plus petite", c'est-à-dire pour un filtre-syllabe sur des syllabes, phonèmes ou lettres mais pour un filtre-lettre sur des lettres seulement.

Nous étudions maintenant les conditions d'enchaînement des actions PRENDRE et TROUVER suivant que l'utilisateur a bien spécifié \*FIXE ou pas.

Il nous importe de savoir où est l'index indiquant la position de début d'une action, de savoir si cette position est toujours considérée comme "correcte", enfin de connaître la position de cet index à la fin de diverses séquences d'actions.

Nous appliquons quatre principes d'exécution :

1 - Le contexte de départ défini par la nature du filtre et l'index de position au début des actions est connu pendant l'exécution de toutes les actions dépendant d'un même filtre. Il est, en particulier, utilisable pour une fonction spécifique de l'utilisateur quelle que soit la séquence de PRENDRE et TROUVER qui a précédé cette action.

2 - Le contexte d'exécution effectif de PRENDRE est toujours le contexte initial. Si une séquence PRENDRE-TROUVER est telle que le contexte a été déplacé (\*FIXE non exprimé), l'action PRENDRE est inopérante.

3 - Le contexte de TROUVER est "libre", l'action est exécutée même si l'index de position n'est pas en position initiale.

4 - La position finale de l'index à la fin d'une séquence est celle du plus grand déplacement de l'index effectué. Elle est *toujours* au début d'une unité de même type que le filtre (syllabe-phonème ou lettre) qui n'a pas été consultée.

Les exemples suivants explicitent les conséquences de ces options.

\*SYLL \*POS=1 \*PRENDRE \*FIXE \*N=2 \*VO  
\*PRENDRE \*FIXE \*CS  
\*PRENDRE \*DEPUIS L \*CS

*Ceci équivaut à rechercher et prendre dans la première syllabe, si elles existent, 2 voyelles et une consonne, puis après la fin de la syllabe, trouver une consonne.*

*Pour le mot AN/TI/QUE*

- . la première action sélectionne le A. Index toujours au début du mot*
- . la deuxième action sélectionne le N. Index toujours au début du mot*
- . la troisième action sélectionne le T. La position pour le filtre suivant (indépendamment de fonctions spécifiques éventuelles) est, au début de la première syllabe, non consultée : Q de QUE. Résultat du quatrième principe.*

*Pour le mot NAUTIQUE*

- . A, U sont sélectionnés*
- . N l'est à son tour et sa position relativement à A et U est conservée :*
  - le résultat des deux actions est la chaîne des caractères "NAU", index au début du mot*
- . T est sélectionné*
  - Le résultat final est NAUT, index pour le filtre suivant sur Q.*

*Pour le mot BAUX, d'une seule syllabe, les caractères retenus seraient BAU, l'action TROUVER ayant un champ de sélection vide et l'index indicatif serait en fin de mot.*

\*SYLL \*POS=1 \*PRENDRE \*N=2 \*VO  
\*PRENDRE \*CS  
\*PRENDRE \*DEPUIS L \*CS

Cet exemple illustre surtout par comparaison avec le précédent le résultat des options 2 et 3.

Dans un cas comme celui-ci, un message signalerait à l'utilisateur les risques de l'exécution d'une telle méthode lors de l'analyse de la description de méthode.

Pour le mot ANTIQUE

- la 1ère action sélectionne le "A", index de position sur le "T"
- la 2ème action trouvant un "contexte décalé" n'est pas exécutée
- la 3ème action s'effectue à partir du "T" et sélectionne "Q".

L'index de position est à la fin du mot. Le calcul pour ce mot se termine (sauf fonction spéciale) et propose comme abréviation "AQ".

Pour le mot NAUTIQUE, l'index est à la fin, la chaîne de caractères sélectionnées est "AUQ".

Pour BAUX, la chaîne sélectionnée est "AU".

Nous avons encore à illustrer le cas de séquences \*TROUVER \*FIXE ... dont les contextes d'analyse sont variables pour déterminer la position finale de l'index, afin d'éclairer la 4ème règle ci-dessus.

\*SYLL \*POS=1 \*TROUVER \*FIXE \*N=2 \*VO  
\*TROUVER \*FIXE \*N=2 \*CS  
\*TROUVER \*DEPUIS L \*VO

trouver deux voyelles et deux consonnes, puis au-delà de la syllabe courante une voyelle.

Pour le mot NAU / TI / QUE

1ère action : sélectionne AU, index au début de la syllabe "NAU", mais la position maximale atteinte est le U.

2ème action : sélectionne N et T, index au début de la syllabe "NAU", mais position maximale atteinte est le T de la 2ème syllabe.

3ème action : sélectionne I, index de position à la 3ème syllabe "Q".

Dans ce cas, la dernière position atteinte est celle qui détermine la position finale.

Pour le mot A / NI / CRO / CHE

1ère action : sélectionne A et I, index au début du mot, mais position maximale atteinte sur I.

2ème action : sélectionne N et le premier C, index au début du mot et position maximale atteinte sur "C" de la 3ème syllabe.

*3ème action : sélectionne à nouveau le I et désigne la fin de la 2ème syllabe comme élément atteint.*

*Mais la position finale atteinte ayant été le C de la 3ème syllabe, la position terminale est sur le C de début de la 4ème syllabe CHE avec comme Chaîne retenue 'ANIC'.*

Ceci montre comment nous tenons compte des unités déjà partiellement balayées.

Une dernière possibilité d'erreur est à signaler. C'est l'enchaînement d'actions contenant toutes \*FIXE.

Pour éviter que l'exécution de telles actions conduisent à reconsidérer deux fois dans les filtres la même unité, le complément \*FIXE de la dernière action PRENDRE ou TROUVER est arbitrairement supprimé lors de la génération de la méthode, un message signalant cette attitude à l'utilisateur.

Il reste enfin trois règles de la grammaire à préciser, règles auxquelles nous avons déjà souvent fait référence dans nos exemples.

Règle 24 : type = \*PV | \*PC | \*PCS | \*PCE  
                  \*SOU | \*SON | \*NAS  
                  \*VO | \*CS

Les premières options de type caractérisent les phonèmes ; nous avons actuellement retenu :

*PV	pour phonème-voyelle	
*PC	pour phonème-consonne	
*PCS	pour phonème-consonne sourd	} au sens de l'analyse syllabique
*PCE	pour phonème-consonne explosif	
	et encore pour les phonèmes	
*SOU	sourd	} d'après une première classification de MARTINET, p. 73
*SON	sonore	
*NAS	nasal	

Nous ne pouvons conserver de caractéristiques plus complexes, car alors la différenciation des phonèmes serait trop fine.

Pour les lettres, nous avons seulement retenu consonne \*CS et voyelle \*VO après avoir envisagé de conserver des caractéristiques de prononciation car celle-ci est trop contextuelle et est davantage révélée par les phonèmes.

Règle 25 : modif-compteur : [\*A | \*Z | \*D] Xnombre

\*A ajouter 1 au compteur cité Xn ( $n \in [0,9]$ )

\*Z remise à zéro du compteur X, ( $n \in [0,9]$ )

\*D décrémenter de 1 le compteur Xn ( $n \in [0,9]$ )

Nous rappelons une contrainte déjà citée dans la *règle 23* :

au même niveau d'une action, on ne peut enregistrer plus de trois modifications de compteur.

Par ailleurs, aucune vérification n'est faite pour savoir si deux modifications consécutives ne sont pas contradictoires en portant sur le même compteur :

\*A X1 \*Z X1 par exemple.

Nous avons étudié extensivement la grammaire descriptive de méthodes telle qu'elle est reconnue actuellement. A la suite du Chapitre 5, nous envisageons de la compléter par des règles permettant de reconnaître les demandes d'évaluation de l'utilisateur.

### 6.2.3. Interface d'une demande d'évaluation

Nous considérons que la demande d'évaluation est une partie de la description d'une méthode qui s'ajoutera en fin de description. Nous voulons rapidement donner ici la grammaire de cette description et expliquer quel sera, à notre avis, le fonctionnement correspondant d'Abrège.

Après la dernière action de la méthode avant \*\*FMETHODE, nous proposons donc d'ajouter la demande d'évaluation.

R26 demande évaluation = \*EVAL (liste abréviation-texte | liste-abréviation)<sup>0/1</sup>  
eval-méthode<sup>0/1</sup> eval-acces<sup>0/1</sup>

R27 liste abréviation-texte = \*LIB-ABR nombre, nombre

R28 liste abréviation = \*ABR nombre, nombre  
donne la liste des abréviations avec ou sans leur libellé initial depuis le libellé n1 jusqu'au libellé n2, où n1 et n2 sont les numéros de réalisation d'entité

R29 eval-méthode = collisions<sup>0/1</sup> compactage<sup>0/1</sup> longueur<sup>0/1</sup> écrétage<sup>0/1</sup>  
début-collision<sup>0/1</sup>

R30 eval-acces = dictionnaire<sup>0/12</sup> phonétique<sup>0/1</sup> syllabique<sup>0/1</sup> filtre-lib<sup>0/1</sup>  
filtre-mot<sup>0/1</sup> calcul<sup>0/1</sup> choix<sup>0/1</sup>

La structure générale de ces règles sera la donnée du mot-clé significatif de la demande, puis de la nature des résultats souhaités par exemple

R31 collision = \*COLLISION \*NI RE<sup>0/1</sup> \*LISTE<sup>0/1</sup> \*TAUX<sup>0/1</sup> ou

R33 longueur = \*LONGUEUR \*MOYEN<sup>0/1</sup> \*STAT<sup>0/1</sup> \*GRAPH<sup>0/1</sup>  
de même pour les accès

R36 dictionnaire = \*ACCES-DIC nom<sup>+</sup> \*NOMBRE<sup>0/1</sup> \*REPARTITION<sup>0/1</sup> \*REPART-STAT<sup>0/1</sup>  
où l'on demande soit un dénombrement simple et global, soit la répartition de ceux-ci par mot, soit des statistiques sur cette répartition.

Il est important d'expliquer à ce niveau la démarche que nous adoptons dans la réalisation de cette interprétation.

La reconnaissance d'une demande d'évaluation entraîne à l'exécution la mise à jour de compteurs témoignant des accès aux filtres, aux règles, aux choix, aux dictionnaires, ainsi que le calcul de variables numériques décrivant les variations du compactage, de la longueur des abréviations, etc.

La reconnaissance de "la forme de sorties" demandée entraîne la création de "fichiers de données" préparés dans le format adéquat pour l'emploi de programmes utilitaires de traitement de données qui soient susceptibles de fournir les résultats dans la forme demandée (statistique, graphique). A la suite de la création des fichiers utiles, a lieu une édition complète de consignes à l'opérateur lui indiquant comment exécuter la phase de traitement de données à partir des fichiers créés.

Nous n'entrons pas plus dans le détail de cette phase qui reste globalement à implanter.

### 6.3. EXPRESSION EN ABREGÉ DES METHODES CITEES EN EXEMPLE

Nous avons, aux Chapitres 2 et 3, conduit cinq méthodes d'abréviation. Nous les écrivons dans la forme reconnue par Abrège, en laissant au lecteur le soin de se rapporter à l'expression en clair donnée précédemment.

Méthode 1 (Chapitre 2, page II.23)

```
**METHODE 1 *LONG < 5 *ANTIDIC ANTIDIC
*LIB *NBMOT = 1
  *MOT *NBLETR < 4 *DIC SPECIAL STANDIC
    *LETR *PRENDRE *LT
  *MOT *DIC SPECIAL STANDIC
    *LETR *POS = 1 *TROUVER *N = 3 LT
    *LETR *POS = 4 *PRENDRE *CS

*LIB *NBMOT < 4
  *MOT *POS = 1 *NBLETR < 8
    *DIC SPECIAL STANDIC
    *LETR *POS = 1 *PRENDRE *LT
  *MOT *POS = 1 *NBLETR >= 8
    *DIC SPECIAL STANDIC
    *LETR *POS = 1 *TROUVER *N = 3 *LT
  *MOT *POS = 2 *FREQR >= 90 % MAX *NBLETR < 8
    *DIC SPECIAL
    *LETR *POS = 1 *PRENDRE *LT
  *MOT *POS = 2 *FREQR >= 90 % MAX *NBLETR >= 8
    *DIC SPECIAL
    *LETR *POS = 1 *TROUVER *N = 2 *LT
  *MOT *LETR *POS = 1 *PRENDRE *LT

*LIB
  *MOT *POS = 1 *NBLETR < 8
    *DIC SPECIAL STANDIC
    *LETR *POS = 1 *PRENDRE *LT
  *MOT *POS = 1 *NBLETR >= 8
    *DIC SPECIAL STANDIC
    *LETR *POS = 1 *TROUVER *N = 3 *LT
  *MOT *LETR *POS = 1 *PRENDRE *LT

** FMETHODE
```



Nous remarquons ici que nous avons traduit par l'action TROUVER la recherche des trois premières lettres du mot, ce qui économise l'usage répétitif des filtres-lettres pour prendre à chaque fois un seul caractère jusqu'au troisième.

Méthode 2 (Chapitre 2, page II.28)

\*\*METHODE \*DOUBLET \*ANTIDIC ANTIDIC

\*LIB \*NBMOT  $\leq$  4

\*MOT \*DIC SPECIAL

\*LETR \*POS = 1 \*TROUVER \*N = 3 \*LT

\*LETR \*POS = 4 \*PRENDRE \*CS

\*LIB \*NBMOT  $\geq$  4

\*MOT \*POS = 1 \*DIC STANDIC

\*LETR \*POS = 1 \*TROUVER \*N = 3 \*LT

\*LETR \*POS = 4 \*PRENDRE \*CS

\*MOT \*DIC SPECCOURT STANDIC

\*LETR \*POS = 1 \*TROUVER \*N = 2 \*LT

\*LETR \*POS = 3 \*PRENDRE \*CS

\*\*FMETHODE

Aucune remarque à ajouter sur cette écriture proche de celle de la méthode précédente.

Dans cet exemple, aucune limite de longueur n'est imposée, mais tous les doublets des abréviations sont supprimés.

Méthode 3 (Chapitre 2, page II.36 )

\*\*METHODE 3 \*DOUBLET \*COLLISION CHIFFRE \*LONG  $\leq$  14

\*ANTIDIC ANTIDIC

\*LIB \*NBMOT = 1 \*MOT \*DIC SPECIAL3

\*LETR \*POS = 1 \*TROUVER \*N = 8 \*LT

\*LIB \*NBMOT = 2 \*MOT \*DIC SPECIAL3

\*LETR \*POS = 1 \*TROUVER \*N = 3 \*LT

\*LETR \*POS = 4 \*PRENDRE \*FIXE \*A X1 \*CS

\*PRENDRE \*Z X1 \*VO

\*LETR \*POS = 5 \*CS \*X1 = 0 \*PRENDRE \*LT

\*LETR \*POS = 5 \*PRENDRE \*LT \*Z X1

\*LIB \*NBMOT = 3 \*MOT \*DIC SPECIAL3

\*LETR \*POS = 1 \*TROUVER \*N = 2 \*LT

\*LETR \*POS = 3 \*PRENDRE \*FIXE \*A X1 \*CS

\*PRENDRE \*Z X1 \*VO

\*LETR \*POS = 4 \*CS \*X1 = 0 \*PRENDRE \*LT

\*LETR \*POS = 4 \*PRENDRE \*LT \*Z X1

\*LIB \*NBMOT = 4 \*MOT \*POS = 1

\*DIC SPECIAL3 STANDIC

\*LETR \*POS = 1 \*TROUVER \*N = 2 \*LT

\*LETR \*POS = 3 \*PRENDRE \*FIXE \*A X1 \*CS

\*PRENDRE \*Z X1 \*VO

\*LETR \*POS = 4 \*CS \*X1 = 0 \*PRENDRE \*LT

\*LETR \*POS = 4 \*PRENDRE \*LT \*Z X1

\*MOT \*POS  $\geq$  2 \*DIC SPECIAL3

\*LETR \*POS = 1 \*TROUVER \*N = 2 \*LT

\*F EUPHONIE

\*LIB \*NBMOT  $\geq$  5

\*MOT \*POS = 1 \*DIC SPECIAL3

\*LETR \*POS = 1 \*TROUVER \*N = 3 \*LT

\*MOT \*DIC SPECIAL3

\*LETR \*POS = 1 \*TROUVER \*N = 2 \*LT \*F EUPHONIE

\*\*FMETHODE

Méthode 4 (Chapitre 3, page III.22)

\*\*METHODE \*DOUBLET \*COLLISION CHIFFRE \*ANTIDIC ANTIDIC ANTISPE

\*LONG  $\leq$  14

\*LIB \*NBMOT = 1

\*MOT \*NLETR  $\leq$  8 \*LETR \*TROUVER \*N = 8 \*LT

\*MOT \*NBSYLL  $\geq$  4 \*SYLL \*POS = 1 \*TROUVER \*N = 3 \*SY

\*SYLL \*POS = 4 \*PRENDRE \*LT

\*LIB \*NBMOT = 2

\*MOT \*DIC SPECIAL4

\*SYLL \*POS = 1 \*PRENDRE \*SY

\*SYLL \*POS = 2 \*PRENDRE \*FIXE \*N < L \*PH

\*PRENDRE \*FIXE \*N = 1 \*DEPUIS L-1 \*A X1 \*PC

\*PRENDRE \*N = 1 \*DEPUIS L-1 \*Z X1 \*PV

\*SYLL \*POS = 3 \*X1 = 1 \*TROUVER \*N = 1 \*PV

\*LIB \*NBMOT = 3

\*MOT \*DIC SPECIAL4

\*SYLL \*POS = 1 \*PRENDRE \*SY

\*SYLL \*POS = 2 \*PRENDRE \*N = 1 \*PH

\*LIB \*NBMOT = 4

\*MOT \*DIC SPECIAL4

\*SYLL \*POS = 1 \*PRENDRE \*SY

\*LIB \*NBMOT = 5

\*MOT \*DIC SPECIAL4

\*SYLL \*POS = 1 \*TROUVER \*FIXE \*N = 1 \*PC

\*TROUVER \*N = 1 \*FV

\*\*FMETHODE

Méthode 5 (Chapitre 3, page III.26)

Cette méthode est celle dont le but est de fournir une abréviation mot à mot.

\*\*METHODE \*ESPACE \*ANTIDIC ANTIDICRED

\*LIB \*MOT \*NBLETR  $\leq$  6

\*DIC SPECPREP SPECDIC

\*LETR \*PRENDRE \*N = L \*LT

peut aussi se traduire

\*LETR \*TROUVER \*N = L \*LT

\*MOT \*NBSYLL = 2

\*DIC SPECPREP SPECDIC

\*SYLL \*POS = 1 \*PRENDRE \*SY

\*SYLL \*POS = 2 \*PRENDRE \*FIXE \*N = 2 \*LT

\*PRENDRE \*N = 1 \*DEPUIS L-1 \*LT

\*MOT \*NBSYLL = 3

\*DIC SPECPREP SPECDIC

\*SYLL \*POS = 1 \*PRENDRE \*FIXE \*N = L \*LT

\*TROUVER \*N = 1 \*DEPUIS L \*LT

\*SYLL \*POS = 3 \*PRENDRE \*N = 2 \*DEPUIS L-2 \*LT

\*MOT \*NBSYLL  $\geq$  4

\*DIC SPECPREP SPECDIC

\*SYLL \*POS = 1 \*TROUVER \*N = 2 \*SY

\*SYLL \*POS = 3 \*PRENDRE \*N = 1 \*PH \*BOUT

\*SYLL \*PRENDRE \*N = 1 \*PH

\*\*FMETHODE

Nous pouvons remarquer dans ces quatre descriptions que nous avons souvent remplacé une action initialement (Chap. 2 et 3) définie par PRENDRE en une action définie par TROUVER, parce qu'à partir d'un certain point on recherche n éléments identiques.

L'intérêt d'une telle modification est que l'on évite ainsi de réévaluer des filtres dont on connaît parfaitement le résultat pour prendre les n éléments que l'on recherche.

Les nombreux dictionnaires cités doivent maintenant être saisis. Nous avons cité :

antidic antispé antidicred  
spécial standic speccourt spécial3 spécial4  
specprep et specdic.

Soit 10 dictionnaires qui peuvent tous "cohabiter" dans la base.

#### 6.4. LA SAISIE DES DICTIONNAIRES

Pour la saisie des dictionnaires destinés à jour le rôle que nous avons défini précédemment, nous avons défini trois actions de base :

- La définition d'un dictionnaire et de ses éléments
- La modification de caractéristiques existantes
- La suppression d'éléments.

La grammaire d'une définition de dictionnaires est donc la suivante :

Règle D1 - def-modif de dictionnaire = [ $*DICO$  identificateur actions-dicos<sup>+</sup>]<sup>+</sup>.  
actions-dicos = définition-dic | modification-dic | suppression-dic.

Règle D2 - définition-dic =  $*DEF$  ( $*M$  mot  $*A$  mot)<sup>+</sup>.

Règle D3 - modification-dic =  $*MOD$  ( $*M$  mot ( $*M$  mot |  $*A$  mot) |  
 $*A$  mot ( $*M$  mot |  $*A$  mot))<sup>+</sup> |  
 $*DE$   $*M$  mot,  $*M$  mot ( $*A$  mot)<sup>+</sup> |  
 $*DE$   $*A$  mot,  $*A$  mot ( $*A$  mot)<sup>+</sup>.

Règle D4 - suppression-dic =  $*SUP$  ( $*M$  mot)<sup>\*</sup> | ( $*DE$   $*M$  mot,  $*M$  mot)<sup>\*</sup> |  
( $*A$  mot)<sup>\*</sup> | ( $*DE$   $*A$  mot,  $*A$  mot)<sup>\*</sup>.

Nous reprenons cette grammaire terme à terme pour en expliciter la sémantique.

Règle D1 - Lors d'une même saisie, il est possible de "travailler" sur plusieurs dictionnaires.

Si l'identificateur est déjà présent dans la base, les actions suivantes sont définies comme des mises à jour ; la définition d'éléments nouveaux correspond à des ajouts.

Si l'identificateur est un nom nouveau, nous vérifions que la base prévue (comprenant 10 dictionnaires et antidictionnaires) permet effectivement la création d'un dictionnaire nouveau. En cas d'impossibilité, un message est édité et la phase de saisie négligée jusqu'au dictionnaire cité suivant.

L'identificateur d'un dictionnaire ne peut excéder 10 caractères (sinon son nom est tronqué).

Toute citation de nom de dictionnaire est suivie d'au moins une action sur celui-ci.

Règle D2 - DEF (\*M mot \*A mot<sup>0/1</sup>)<sup>+</sup>

La définition des éléments nouveaux d'un dictionnaire se fait par donnée simultanée d'un mot et de son abréviation correspondante.

Il est possible qu'aucune abréviation ne soit donnée. Dans ce cas, le mot du texte correspondant à celui du dictionnaire a une abréviation vide dans la chaîne de caractères créée par la méthode.

Si le dictionnaire ne comporte que des abréviations vides, il s'agit d'un antidictionnaire.

Nous vérifions la validité de cette caractéristique "antidictionnaire" par balayage de tous les éléments du dictionnaire et vérification de l'existence en tout point d'abréviation vide.

Lorsqu'un élément est défini dans le dictionnaire, mais qu'il existe déjà dans celui-ci, la nouvelle définition n'est pas prise en compte (puisque'il existe par ailleurs une fonction de modification), mais un message est édité.

Règle D3 - modification-dic = \*MOD ...

Cette règle présente deux formes de base, l'un de modification élément par élément, l'autre de modification de tout un ensemble d'éléments repéré par deux bornes.

a) Modification élément par élément

Il existe deux cas de modifications : le cas où l'on veut modifier l'élément servant à la référence, c'est-à-dire le mot du dictionnaire et le cas où l'on change une abréviation donnée. Dans les deux cas, en outre, la recherche de l'élément dans le dictionnaire peut se faire soit par le mot, soit par son abréviation.

Ceci conduit à 4 formes de modification mot à mot

- 1) \*M mot      \*M mot    = signification avec recherche par mot et changement de celui-ci
  - 2) \*M mot      \*A mot    = modification avec recherche par le mot et changement de l'abréviation
  - 3) \*A mot      \*M mot    = recherche du mot dont l'abréviation est donnée et modification du mot ainsi abrégé
  - 4) \*A mot      \*A mot    = recherche et modification d'abréviation.
- terme                    terme  
de                        de  
recherche                modification

Il est à noter que si la recherche par mot conduit à une recherche par clé unique, nous n'avons aucune hypothèse équivalente pour la recherche par abréviation, car deux mots du dictionnaire peuvent avoir même abréviation.

Nous avons donc supposé que l'utilisateur qui emploie une recherche par l'abréviation "sait ce qu'il fait" et nous nous arrêtons à la première abréviation cohérente trouvée.

A ce propos, il paraît utile d'introduire une remarque concernant l'exécution de telles modifications.

Les modifications avec recherche par mot mettent en oeuvre une utilisation de l'unicité des mots du dictionnaire dans une recherche avec hash-code. Celle-ci est donc supérieure (surtout dans les "gros" dictionnaires) à la recherche séquentielle que nous sommes obligés de faire pour les abréviations.

b) Modification par blocs d'éléments

*DE *M mot , *M mot	*A mot      *A mot ...
*DE *A mot , *A mot	*A mot      *A mot ...
<u>                    </u>	<u>                    </u>
terme de recherche	éléments de modification

Partant de l'hypothèse que la modification globale d'éléments d'un dictionnaire n'intéressait que la modification d'abréviations, nous n'avons ici que deux formes liées au mode de recherche par mot ou par abréviation, en prenant les blocs tels qu'ils sont "rangés" au niveau des réalisations d'entité

MOTS-DIC ; leur liste est connue de l'utilisateur car nous éditons les dictionnaires créés à la fin de chaque saisie. Les abréviations de tous les mots compris dans la réalisation du dictionnaire entre les deux bornes citées (celles-ci étant incluses) sont modifiées par la liste qui suit employée élément par élément.

Ceci suppose évidemment que l'utilisateur dispose d'une "liste valide du dictionnaire" au moment il écrit cette modification et qu'elle est encore valide au moment où il l'exécute. Nous vérifions simplement qu'il y a autant d'éléments dans l'intervalle que dans la liste de citation, avant de faire la modification.

#### Règle D4 - Suppressions

Elle peut concerner un dictionnaire entier, c'est le cas de la règle

\*SUP  $\phi$ , c'est-à-dire où ce mot-clé est aussitôt suivi d'un autre mot-clé.

En général, on rencontrera \*DICO nom \*SUP \*DICO ... ou \*\*FDICTION car il est absurde de faire des modifications sur un dictionnaire que l'on supprime.

Le dictionnaire et ses abréviations, ainsi que les liens avec les mots des libellés sont supprimés au terme de l'exécution.

L'autre version supprime des mots dans le dictionnaire cité avant la modification. La référence comme précédemment dans les modifications de valeur peut se faire par mot ou abréviation avec les contraintes d'exploitation déjà décrites.

Celle-ci est décrite par simple citation du terme ou du bloc à supprimer.

La suppression est une suppression "physique" dans la base.

Enfin, signalons que lors de la réalisation de toutes ces actions, nous mettons à jour, dans la mesure où cela est possible, les liens entre les mots des dictionnaires et l'ensemble des mots des libellés (entité MOTS-LIB).

Nous n'effectuons évidemment que les actions syntaxiquement correctes et nous éditons une liste des actions exécutées et de celles refusées. Il est



raisonnable de supposer que les actions demandées sont indépendantes les unes des autres.

Donnons maintenant un unique exemple de définition, puis de modification de dictionnaire en prenant comme base le dictionnaire de la méthode 1, SPECIAL qui contient surtout les "pivots" du texte.

Saisie de SPECIAL

\*\*DICTION

```
*DICO SPECIAL *DEF *M TECHNIQUE *A T
      *M IMPUTATION *A I
      *M COMPTE *A CPT
      *M CONTRE *A C
      *M ELEMENTAIRE *A E
      *M MOUVEMENT *A MVT
      *M FISCALITE *A F
```

\*\*FDICTION

Ceci correspond à une première saisie du dictionnaire. Lors d'un autre passage, on a envie de le modifier un peu, d'où la nouvelle phase

\*\*DICTION \*DICO SPECIAL

```
*DEF *M NOMBRE *A
      *M TAXE *A TAXE
      *M MONTANT *A
      *M TEST *A T
      *M CODE *A C
      *M 1% *A 1%

*MOD *M MOUVEMENT *A MOUV
*DEF *M AGENT *A
      *M CMVT *A *M GDF *A GDF
      *M 991 *A 1
      *M 994 *A 4
```

\*\*FDICTION

On voit que SPECIAL va "éliminer" certains mots sans que pour autant ceux-ci soient considérés comme non significatifs pour l'abréviation.

## 6.5. LA SAISIE DES DONNEES

La base est maintenant prête à "travailler" puisque nous avons tous les éléments fonctionnels, les méthodes, les dictionnaires.

Reste maintenant à saisir les données sur lesquelles l'utilisateur veut étudier des méthodes.

Pour cela, il va évidemment donner les textes des libellés sur lesquels il veut que travaillent les méthodes, un à un. Nous supposons que peuvent ainsi s'introduire des erreurs et nous avons prévu la *suppression* et la *modification* de textes. Et puisque la suppression introduit des "trous" dans l'ensemble des libellés, nous avons aussi envisagé une fonction d'*insertion*.

Notons dès maintenant que cette saisie avec les possibilités de modifications n'a rien d'un "éditeur de texte". Nous avons décidé de simplifier les modifications au maximum en définissant essentiellement une fonction de *remplacement* de texte par un autre.

Pour ces textes, nous respectons le format des entrées standard en SOCRATE en les limitant à des lignes de 60 caractères. L'analyse de l'existant semble montrer qu'en limitant à 180 caractères, soit 3 lignes, la longueur des textes, nous n'introduisons pas de limite contraignante, même au niveau du traitement des textes mot à mot car ceux-ci sont faits de phrases courtes.

### 6.5.1. La grammaire pour la saisie des libellés

\*\*LIBELLE déf-modif-données \*\*FLIBELLE.

Règle L1 - déf-modif-données = (Définitions|Modifications|Suppressions|Insertio

L2 - définitions = \*DEF descr-texte<sup>+</sup>.

L3 - descr-texte = \*" texte60 (\*1 texte60 (\*2 texte60)<sup>0/1</sup>)<sup>0/1</sup> "\*.

L4 - modifications = \*MOD ((\*UN (descr-texte|num-texte) \*PAR descr-texte)  
|(\*DE (descr-texte|num-texte)(descr-texte|num-te  
\*PAR descr-texte<sup>+</sup>))<sup>+</sup>.

Règle L5 - suppressions = \*SUP ((\*UN (descr-texte | num-texte)) | (\*DE (descr-texte|num-texte)(descr-texte|num-texte)))

L6 - insertions = \*INS ((\*UN num-texte descr-texte) | (\*DE num-texte num-texte (descr-texte)<sup>+</sup>))<sup>+</sup>.

L7 - num-texte = nombre de 0 à 2000.

L8 - texte60 = texte de 1 à 60 caractères.

### 6.5.2. Sémantique d'une description de saisie

Règle L1 : Cette règle montre essentiellement que la définition d'éléments nouveaux, les modifications et les suppressions peuvent apparaître dans un ordre quelconque.

Règle L2 - Règle L3 : Les définitions se font texte par texte sans répétition du nom de l'action définition \*DEF.

Ils sont, en fait, stockés dans l'ensemble des libellés à la première place disponible. Nous n'avons pas permis à l'utilisateur d'imposer en plus une numérotation des textes.

Le texte lui-même est donné sur une à trois lignes suivant sa longueur.

#### FORMAT :

La saisie étant prévue sur carte, nous imposons la contrainte suivante : Les caractères \*, \*1, \*2 de début de ligne de texte doivent se trouver perforés en colonne 1 et 2.

Le caractère de fin de texte "\*" doit se trouver sur la même carte que la "dernière ligne" entre les colonnes 3 et 64.

Le format interne du texte est libre à l'intérieur d'une ligne, mais il n'est pas permis de couper un mot sur deux lignes.

A ce niveau, il paraît souhaitable de donner quelques précisions sur l'usage qui est fait du texte donné.

### Décomposition du texte

Tout d'abord, ce texte est stocké tel quel dans la base avec seulement une "condensation" de plusieurs espaces successifs en un seul. Ensuite, il est décomposé mot à mot, les mots étant séparés les uns des autres par des espaces, des signes de ponctuation ou des tirets. Les tirets étant en français une marque de mots composés, ceux-ci sont découpés suivant leurs termes (pour éviter de les tronquer à 30 caractères), et le tiret est conservé comme premier caractère du deuxième mot.

Les *chiffres* ne sont pas traités séparément et nous avons seulement tenté de reconnaître les *sigles* de la forme 'x.x.x.'... comme un seul mot. Les sigles présentant une autre forme, 'xx.xx..' par exemple, devront être écrits sans point séparateur si l'on ne désire pas les découper.

Comme nous pouvons le noter, les traitements effectués actuellement sont simples car nous avons pour premier but de traiter des textes simples. Nous pourrions facilement modifier ce module de traitement des textes.

### Règle L4 : Modifications

Nous avons adopté deux modes d'expression des modifications, soit élément par élément, soit par modification de tout un groupe de libellés.

La première forme \*UN ... permet d'une part la recherche par citation du texte (dont les blancs sont "condensés"), d'autre part la citation du numéro d'entité libellé correspondant. Celui-ci est facilement connu de l'utilisateur car à la fin de chaque "séance" de traitement (définition, suppression ...) de libellés, nous éditons les textes stockés dans l'ordre des numéros d'entité de la base.

Il semble de ce fait que ce mode de citation, plus rapide et plus sûr, soit à préconiser, car la moindre erreur dans la citation du texte empêche la reconnaissance de celui-ci. Ceci est évidemment une sécurité mais pénalise lourdement le traitement.

La deuxième forme \*DE ... définit un sous-ensemble de l'ensemble des libellés par les bornes d'un intervalle (fermé), exprimé sur l'ensemble des réalisations de l'entité LIBELLES.

Les bornes de cet intervalle sont repérées (comme ci-dessus) soit par citation du texte et recherche séquentielle sur l'ensemble, soit par numéro de libellé et accès direct par le numéro de réalisation de l'entité.

Avant d'effectuer une modification, nous vérifions qu'à la suite du mot-clé \*PAR figurent autant de textes que d'éléments dans l'intervalle (bornes comprises). La modification n'est pas réalisée, pour éviter d'avoir ensuite de nouvelles modifications à reprendre.

#### Règle L6 : Les Suppressions

Les suppressions se font par citation. Il est certain que la citation du texte entier accroît la sécurité de l'action, mais ici encore pour la rapidité de l'accès fourni et la simplicité d'écriture, nous proposons l'emploi du numéro d'entité. La suppression provoque une destruction "physique" du libellé dans la base.

#### Règle L7 : Insertion

Dans la mesure où une insertion ne peut se faire qu'au niveau d'une entité libre - nous n'allons pas faire des insertions qui modifieraient toute la base - la définition de cette action demande la citation explicite de la "place" que l'on désire occuper, ou de l'intervalle d'entités encore indéfinies que l'on désire remplir.

Le contexte d'écriture est sinon le même que pour les règles précédentes.

#### 6.5.3. Informations complémentaires sur la saisie des données

Bien qu'il ne soit pas nécessaire de connaître le détail de l'exécution de cette saisie, nous explicitons un peu la manière dont elle se déroule.

La saisie comporte plusieurs phases :

- 1 - La reconnaissance des actions demandées et de l'exactitude syntaxique de leur expression.
- 2 - L'exécution effective si la phase 1 est correcte de toutes les actions demandées.
- 3 - La mise à jour de la base Abrège pour tous les nouveaux éléments introduits : décomptes pour les fréquences, liens avec les dictionnaires, mots nouveaux, etc ...

Avant d'en détailler le contenu plus précisément, notons que cette saisie est donc a priori une action assez longue à exécuter.

Dans le mode de traitement par lot utilisé, ceci ne pose pas de problème particulier si ce n'est celui du coût.

Dans l'hypothèse de la transformation d'Abrège pour permettre son emploi en mode conversationnel, il paraîtrait souhaitable de séparer la troisième phase des précédentes pour ne pas provoquer lors de chaque saisie une attente longue et irritante pour l'utilisateur assis devant son terminal. Notons, cependant, que la saisie doit raisonnablement toujours être exécutée en mode de traitement par lot, sauf lorsque l'on veut introduire quelques corrections et modifications ponctuelles, dans les données déjà enregistrées.

#### Phase 1 - Analyse syntaxique et validation sémantique -

L'analyse syntaxique est réalisée par un automate résidant dans la base. Il procède du même principe que l'analyse phonétique.

Lorsque les mots-clés ne sont pas ceux prévus, un message est édité et aucune action n'est effectuée.

Tout texte rencontré est analysé pour vérifier la cohérence de sa description avec la syntaxe (séquence des cartes correctes ...).

Si le texte doit agir comme référence, nous vérifions en outre sa présence dans la base.

Cette analyse conduit à la création d'un fichier intermédiaire où la saisie est en fait traduite en un "pseudo-code" plus condensé. Ainsi, les textes de référence "trouvés" sont remplacés par le numéro de réalisation de l'entité LIBELLES correspondante, ce qui évite de recommencer deux fois la recherche séquentielle de celle-ci.

Sur ce fichier, sont aussi stockés les erreurs détectées et les messages associés. L'emploi de ce fichier intermédiaire est évidemment "transparent" à l'utilisateur.

Phase 2 - En cas de détection d'erreur, cette phase est réduite à l'édition de toutes les anomalies détectées dans la saisie.

Si la description de saisie est correcte, les actions demandées sont reprises une à une ; les textes sont conservés dans l'entité libellé, et décomposés mot à mot avec mise à jour du dictionnaire des mots utilisés MOTS-LIB au niveau des occurrences des mots ou par ajout des mots nouveaux.

Phase 3 - Au terme de cette saisie, on met à jour les références qui lient les mots des libellés de MOT-LIB aux dictionnaires (et les chaînes de bit des inverses). On recalcule la fréquence d'occurrence des caractères.

Enfin, on édite un nouvel état des libellés pour l'utilisateur.

Nous ne donnons pas ici d'exemple de saisie. Il suffit de se reporter aux bordereaux donnés en annexe [A9] pour en avoir un échantillon. Nous n'avons pas prévu de numérotter les lignes de définition d'une saisie pour pallier un éventuel désordre dans les cartes. Les colonnes 73 à 80 étant libres il serait bien facile à l'utilisateur de le faire pour remettre facilement ses données en état lors d'un incident. Par ailleurs, l'emploi actuel des cartes peut être remplacé à l'usage par celui de bandes, ou même de disque (traité par SOCRATE comme une bande). Il suffit pour cela de changer l'instruction de lecture du programme de saisie. Cette relative maladresse vient de ce que nous avons écrit tous nos traitements en SOCRATE afin que ceux-ci soient totalement "transparents" à l'utilisateur. Il ne pouvait en être ainsi si nous utilisions l'interface SOCRATE-COBOL pour les données et SOCRATE seulement pour la mise à jour de la base et l'analyse syntaxique. L'avantage de la version actuelle est de permettre à l'utilisateur d'écrire en Abrège seulement et d'ignorer les diverses phases du travail.

## 6.6. LA DEMANDE D'EXECUTION D'UNE METHODE

Cette dernière phase consiste simplement à préciser quelle méthode l'utilisateur veut exécuter et sur quelles données. On peut en effet avoir stocké dix méthodes et 2000 libellés. Le résultat de cette demande est l'exécution de tout ce qui a été décrit dans la méthode en fonction des données, libellé par libellé.

Cependant, avant exécution, le système vérifie que tous les dictionnaires cités dans la méthode sont présents dans la base, que toutes les fonctions sont définies. En cas d'absence de l'un de ces éléments, l'exécution est différée et un message d'erreur est édité.

La grammaire de cette demande est la suivante :

demande-exécution : \*\*EXECUTE description-de-demande-d'exécution \*\*FEXECUTE.

Règle X1 : description-de-demande-d'exécution : \*METH numéro  
\*LIB ( \*TOUT|(\*DE num \*A num)\* ).

Nous voyons que cette demande ne concerne qu'une seule méthode, mais la grammaire de "niveau 1" permet d'enchaîner plusieurs exécutions.

Nous avons décidé de simplifier cette demande en repérant les libellés par leur numéro de réalisation d'entité. Une référence par texte serait possible mais les risques d'erreur de citations sont plus élevés et la référence est plus complexe à implanter.

Lorsque l'utilisateur demande une exécution, il a déjà saisi les données et le système lui a donc fourni une liste complète des libellés avec leur numéro de réalisation.





**CHAPITRE 7**

AMELIORATION ET EVOLUTION DU SYSTEME ABREGE. CONCLUSION

## 7.1. REALISATION CONVERSATIONNELLE D'ABREGE

Nous avons présenté un prototype conçu entièrement en système de traitement par lot essentiellement en raison du logiciel dont nous disposions en 1975, soit sur l'IRIS 45 de l'I.U.T. de Grenoble dont la configuration (absence de virgule flottante cablée, partition mémoire centrale disponible) entraîne des dégradations de performances importantes du système Socrate, soit sur l'IBM 360 sur lequel la première version de Socrate implantée en OS sous ASP-MVT est uniquement prévue pour le traitement par lot.

Nous disposons actuellement d'une version conversationnelle de Socrate sous OS implantée sur IBM 360 sous CP. Ces nouvelles conditions d'exploitation permettent donc d'envisager une version conversationnelle d'Abrège, dont les spécifications sont facilitées par le choix du système Socrate. Les améliorations porteraient surtout sur la saisie ; elles porteront aussi sur les possibilités accrues d'interrogation de la base Abrège, par exemple pour obtenir l'abréviation de certains libellés seulement, ou pour demander l'exécution d'une méthode "point par point" sur des libellés "intéressants".

### 7.1.1. Saisie conversationnelle. Points-clefs. Limites

Lors de la définition des méthodes, il paraît surtout intéressant de faire usage de la conception modulaire de l'analyse syntaxique par "automates enchaînés" (Annexe [A8]) pour éditer des erreurs détectées en analysant les filtres-mots, syllabes, phonèmes ou lettres, ou les choix, de manière à permettre la correction immédiate de ces erreurs.

Si la saisie des données paraît raisonnablement devoir être traitée toujours en temps différé, la correction des erreurs qui ont pu s'y insérer, la modification de certains textes par l'utilisateur, dans le but de modifier l'abréviation résultante pour une meilleure conservation de la signification, sont des fonctions pour lesquelles un système interactif paraît naturel.

Les dictionnaires quant à eux ne sont justiciables d'un traitement par lot que lorsque ce sont de "gros dictionnaires", sinon leur saisie dans la base est plus "agréable" sous un système interactif. Dans tous les cas, leurs modifications justifient un tel système.

Ces diverses fonctions interactives peuvent aisément être ajoutées au système existant, les modules écrits pouvant être réutilisés directement, avec cependant la limite actuelle d'une conception de la base pour une exploitation "monoutilisateur".

#### 7.1.2. Exécution ponctuelle. Interrogation de la base

La base ayant été conçue pour permettre de stocker plusieurs méthodes, une interrogation sur les résultats ponctuels de celles-ci peut aider à déterminer quelles améliorations apporter aux points difficiles et quelles conclusions tirer des résultats.

Dans un tel contexte, il paraît important de revoir les conditions d'emploi de l'évaluation. Nous avons en effet inclu celle-ci dans la description des méthodes. Mais nous avons aussi précisé au chapitre 5 que certains éléments, ceux de traces en particulier, étaient probablement à mesurer à tout coup même si l'on n'édite le résultat que conditionnellement. Les autres cas ne sont pas évalués systématiquement parce que les données déjà présentes dans la base permettent l'évaluation a posteriori. Les conditions d'une évaluation différée sont réalisées donc dans chaque cas. Il est logique de permettre l'expression et l'exécution de cette demande, seulement lorsque cela est demandé par l'utilisateur, c'est-à-dire lorsqu'il a pris connaissance des abréviations d'une méthode et qu'il se pose des questions précises sur les résultats.

Puisque nous disposons à tout instant des résultats bruts nécessaires à l'évaluation, la seule difficulté est la mise en oeuvre, à la demande, des programmes statistiques dont nous avons supposé qu'ils n'étaient pas inclus dans la base sous Socrate.

## 7.2. EXTENSION D'ABREGE A D'AUTRES CLASSES DE METHODES

Notre expérience dans l'écriture de méthodes d'abréviations ainsi que diverses remarques qui nous ont été formulées à propos de la version actuelle du système Abrège nous ont déjà suggéré des extensions possibles à des classes de méthodes non encore descriptibles.

### 7.2.1. Prise en considération de "groupes de mots"

La première extension suggérée est celle de trouver un moyen de traiter des "groupes de mots", unité à abrégé globalement, comme "compte courant", "bon d'engagement", "service direction et méthodes" rencontrés dans les exemples traités. Les groupes de mots doivent être reconnus comme une entité insécable pour le traitement de la méthode et abrégés comme tels. C'est donc l'option inverse de ce que nous avons choisi de faire lorsque nous découpons les mots composés au niveau des séparateurs, et que nous les abrégeons mot à mot.

Les groupes de mots intéressant une application particulière ne sont pas une donnée intrinsèque du langage. En fait, il faut qu'ils aient été désignés d'une manière quelconque par l'utilisateur pour que l'on puisse en repérer la nature particulière.

Dans ces conditions, on peut envisager plusieurs techniques :

- Les transformer artificiellement en "mots composés" reconnus par la présence de tirets et modifier le programme qui reconnaît les mots des textes en ne coupant pas les mots composés. La seule difficulté est alors le risque de dépassement de longueur du groupe de mot lorsque nous voudrions le stocker dans un "mot" Socrate de 30 caractères, comme la structure actuelle le prévoit. Sous réserve de résoudre ce problème, nous pouvons soit lui appliquer les règles existantes de calcul d'abréviation ou d'utilisation de dictionnaires, soit ajouter une caractéristique "groupe de mot" et ajouter au système des règles spéciales de même nature que les filtres-mots et les traitements qui les suivent.

- On pourrait aussi reconnaître les groupes de mots, en prévoyant entre les mots des liens de chaînages qui seraient créés par des "déclarations de groupes de mots". Ensuite les traitements seraient modifiés suivant le même principe que ci-dessus.

### 7.2.2. Reconnaître la syntaxe de "qualifications en cascade"

Nous avons dit au chapitre 4 que nous renoncions à l'analyse morphologique aussi bien qu'à l'analyse syntaxique des textes en raison des ambiguïtés de leurs résultats. Mais il faut aussi constater que les textes de définition pour créer des identificateurs ont souvent une structure de qualifications en cascade, témoin dans l'exemple suivant "Rejet de la proposition de dépense". Il serait intéressant de savoir reconnaître cette structure particulière des textes. Cela suppose une analyse qui fait usage des mots outils significatifs de relations comme "de", "du", "en", "par", ainsi qu'un classement entre ces relations pour permettre de déterminer un "parenthésage" de la phrase. Pour cela, il n'est pas nécessaire de reconnaître toute la syntaxe de la langue naturelle, mais un sous-ensemble extrêmement limité.

L'analyse syntaxique des relations de qualification serait particulièrement intéressante s'il en résultait un parenthésage unique et non ambigu du texte. On peut cependant se rendre compte facilement que cela n'est pas possible ; prenons un exemple :

*"Codification des bons d'engagement du service direction-moyens (SERDM) par le service informatique".*

*Le "par ..." qualifie tout ce qui précède mais doit-on parenthésier la première partie suivant cette forme :*

*"codification (des bons d'engagement) (du SERDM)"*

*ou celle-là :*

*"codification (des bons) (d'engagement du SERDM)".*

Ce choix est sémantique. Peut-être faut-il proposer à l'utilisateur un "Guide de la définition" qui, en définissant une syntaxe de compléments de noms assez figée, permette cette analyse sans ambiguïté.

Nous n'avons pas approfondi cette question. Remarquons simplement que l'emploi d'une analyse syntaxique de ce type doit être préalable à l'utilisation des antidictionnaires tels que nous les avons décrits. Il faut aussi noter que le parenthésage reintroduit la notion de "groupes de mots" que nous n'avons pas conservée dans la version actuelle du prototype. Répétons enfin que la prise en compte de cette structure spécifique du texte est une amélioration d'Abrège pour la production d'un mot unique et n'a aucun sens dans des petites annonces par exemple où la structure est plus une juxtaposition d'éléments sémantiquement différents séparés par des virgules.

Nous n'avons pas approfondi davantage les problèmes introduits par la prise en compte de ces nouvelles caractéristiques du texte. Nous pensons qu'il faut avant cela analyser les moyens et les limites du système dans sa version actuelle pour mieux préciser comment interviendraient ces propriétés nouvelles des textes dans les méthodes d'abréviation.

### 7.2.3. Quelques modifications dans l'écriture des méthodes

Nous avons déjà remarqué au chapitre 6, page VI.38 que lorsque nous décrivons des méthodes, nous sommes parfois amené à décrire le même filtre mot suivi des mêmes actions dans deux contextes de libellés identiques, ou la même suite d'actions dans des contextes filtrés différemment.

Dans un tel cas, on est tenté de faire référence à la première description sans la réécrire. Pour cela on pourrait autoriser un mode de déclaration donnant un nom à un filtre mot puis ensuite faire référence à ce nom au lieu de réécrire le texte. Prenons un exemple :

*Dans Méthode2, page VI.37, la politique de sélection des caractères pour les mots des libellés de 3 mots et pour le premier mot des libellés de quatre mots est la même et elle est assez longue à exprimer. On pourrait remplacer tous les filtres-lettres décrits par des citations de filtre-lettre, ainsi :*

```
⋮
*lib *nbmot = 3 *mot *dic spécial3
      *letr letr1 *pos = 1 ...
      *letr letr2 *pos = 3 ...
      *letr letr4 *pos = 4 ...
```

```
*lib *nbmot = 4 *mot *pos = 1 *dic spécial3 standic
  *letr *ref letr1
  *letr *ref letr2
  *letr *ref letr4
```

*où letr1, letr2, letr4 sont des noms donnés aux filtres-letr-actions qui sont au niveau de la déclaration.*

Une telle notion est équivalente à celle de macro-instruction dans bien des langages. On pourrait par ailleurs, si les méthodes se compliquent, introduire même des macro instructions paramétrées.

Cette modification intervient uniquement au niveau de la grammaire de description et de l'interprétation du langage de description lors de la phase de contrôle et de stockage de la méthode dans une structure Socrate, mais ne change rien quant à la classe des méthodes interprétées.

#### 7.2.4. Quelques extensions des classes de méthodes

Nous avons déjà rencontré au cours de nos essais quelques limites d'expression de méthodes qui sont liées au mode d'exécution de celles-ci.

Citons deux exemples :

- 1 - *Le premier est simplement lié à la définition du champ des actions : nous avons dit combien nous voulions d'éléments lors d'une recherche de caractères (\*N ≤ ...), à partir de quel point nous les prenions en compte (\*DEPUIS ...) mais nous n'avons pas donné à l'utilisateur la possibilité de limiter son balayage à deux ou trois syllabes du mot sans aller jusqu'au bout si à l'exécution la caractéristique n'est pas trouvée. Cela reviendrait à ajouter un complément \*JUSQUA du verbe TROUVER pour lequel on signalerait le nombre de syllabes, phonèmes ou lettres consultés.*
- 2 - *Le deuxième exemple est lié au mode d'exécution séquentiel que nous avons choisi. Celui-ci intervient de manière très contraignante tant au plan de l'écriture que dans la définition des méthodes reconnues.*



Ainsi nous avons choisi d'interpréter une description de méthode en nous arrêtant sur le premier filtre libellé qui retourne la valeur vraie pour le libellé en cours de traitement et en utilisant exclusivement les filtres mots et les calculs qui lui sont associés comme méthode d'abréviation des mots de ce libellé.

Rappelons que la sémantique actuelle d'exécution est la suivante :

- A1 *pour un libellé trouver le premier filtre libellé  $fl(i)$  qui retourne la valeur vraie*
  - A12 *faire : pour un mot du libellé, trouver le premier filtre  $flm(i,j)$  qui retourne la valeur vraie*
    - alors A1.2.1 si le mot a une abréviation par les dictionnaires cités, en faire usage*
    - A1.2.2 trouver le filtre-unité (syll, phon, letr)  $flmu(i,j,k)$  qui retourne la valeur vraie*
    - A1.2.3 faire le calcul*
- sinon passer au mot suivant, refaire A12.*

Nous aurions pu adopter une autre sémantique où les filtres-libellés et les filtres-mots ne soient pas ainsi hiérarchisés, mais où ils soient au même niveau. Dans ce cas, notre recherche du traitement à appliquer serait la suivante :

*pour tout mot du libellé en cours, trouver le premier filtre libellé qui retourne la valeur vraie et qui contienne un filtre-mot qui retourne la valeur vraie.*

Cette interprétation met l'accent sur l'importance du mot comme base de calcul d'une abréviation et diminue le nombre des cas où un mot pris dans un libellé se trouve sans filtre-mot satisfaisant (comme dans la méthode<sup>3</sup>, page VI.39). Par ailleurs, cette écriture plus coûteuse à l'exécution puisqu'elle peut amener à évaluer plus de filtres libellés et de filtres mots, est plus économique d'écriture. Ainsi par exemple, si la méthode est la même pour l'abréviation de tous les premiers mots de tous les libellés, il suffit dans ce cas d'écrire un filtre libellé global contenant un seul filtre-mot concernant le premier mot. La sémantique d'exécution est alors :

*Pour un libellé en cours de traitement, pour le mot du libellé courant,*

*A.1 trouver un filtre-libellé  $fl(i)$  qui passe pour le libellé courant*

*A1.2 trouver un filtre-mot qui passe pour le mot courant  
alors utiliser dictionnaires ou calculs correspondants,  
puis passer au mot suivant,  $i \leftarrow 1$ , refaire A1  
sinon  $i \leftarrow i+1$ , refaire A1.*

Il est certain que la classe de méthodes ainsi reconnue est plus vaste que celle que nous avons admises en simplifiant au maximum l'algorithme d'évaluation du prototype. Nous nous contenterons ici de montrer les simplifications que cela apporte dans l'écriture de la méthode 3 par rapport à l'expression au chapitre 6 (page VI.38) sans chercher à délimiter plus précisément les "bornes" de cette nouvelle classe.

### Méthode 3

```
**METHODE 3 *DOUBLET *COLLISION CHIFFRE *LONG  $\leq$  14
      *ANTIDIC ANTIDIC
*LIB *NBMOT = 1 *MOT *DIC SPECIAL3
      *LETR *POS = 1 *TROUVER *N = 8 *LT
*LIB *NBMOT = 2 *MOT *DIC SPECIAL3
      *LETR *POS = 1 *TROUVER *N = 3 *LT
      *LETR *POS = 4 *PRENDRE *FIXE *A X1 *CS
      *PRENDRE *Z X1 *VO
      *LETR *POS = 5 *CS *X1 = 0 *PRENDRE *LT
      *LETR *POS = 5 *PRENDRE *LT *Z X1
*LIB *NBMOT = 4 *MOT *POS  $\geq$  2 *DIC SPECIAL3
      *LETR *POS = 1 *TROUVER *N = 2 *LT
      *F EUPHONIE
*LIB *NBMOT  $\geq$  5
      *MOT *POS = 1 *DIC SPECIAL3
      *LETR *POS = 1 *TROUVER *N = 3 *LT
      *MOT *DIC SPECIAL3
      *LETR *POS = 1 *TROUVER *N = 2 *LT *F EUPHONIE
```

```
*LIB *NBMOT > 3 *MOT *DIC SPECIAL3
      *LETR *POS = 1 *TROUVER *N = 2 *LT
      *LETR *POS = 3 *PRENDRE *FIXE *A X1 *CS
                *PRENDRE *Z X1 *VO
      *LETR *POS = 4 *CS *X1 = 0 *PRENDRE *LT
      *LETR *POS = 4 *PRENDRE *LT *Z X1

**FMETHODE
```

Le dernier filtre-libellé est accédé seulement pour les libellés de trois mots et le premier mot des libellés de quatre mots car sinon l'ensemble des mots des libellés est recouvert.

Nous voulons insister sur le fait que dans cet exemple la syntaxe du langage n'est pas modifiée et que seule la sémantique en a été modifiée.

### 7.3. CONCLUSION

Le risque encouru lors de la réalisation d'un produit informatique est de lui apporter des améliorations ou soit disant telles par stratification successive, sans marquer une pause entre les différentes versions, pause consacrée à une validation de l'outil informatique dans des conditions normales d'exploitation. Quel que puisse être l'intérêt a priori des extensions proposées, il nous paraît prioritaire de consacrer nos efforts à venir à une évaluation précise de l'efficacité d'utilisation d'Abrège dans des conditions normales d'exploitation.

Nous voulons en souligner plus particulièrement trois :

1) - Il nous paraît intéressant de voir quel est le degré de conscience des utilisateurs que nous considérons comme intéressés par la réalisation de méthodes automatiques d'abréviations, à l'égard des mécanismes que ces méthodes mettent en oeuvre.

En effet, ce travail a supposé une analyse assez fine de mécanismes phonétiques ou syllabiques qui sont le fondement de la classe de méthode que nous avons proposée. Nous ne connaissons pas encore l'étendue du travail qu'il nous faudra non pour apprendre le langage de description aux utilisateurs, mais pour leur apprendre la définition d'une méthode.

Tout comme la connaissance de la syntaxe et de la sémantique d'un langage de programmation n'est pas la garantie d'une bonne programmation, la connaissance des mécanismes, de la syntaxe et de la sémantique d'Abrège n'est pas une garantie de définition de "bonnes méthodes", si ceux qui s'en servent n'ont pas une méthodologie de mise en oeuvre. La disponibilité d'un langage de haut niveau pour exprimer un algorithme ne diminue pas le travail de conception nécessaire pour définir cet algorithme.

Quels que soient la souplesse et le niveau de synthèse que nous avons voulu donner à Abrège, nous ne connaissons pas encore le travail nécessaire et la formation requise de la part des utilisateurs d'Abrège pour concevoir une méthode avant de la formuler en Abrège.

2) - Nous avons vu que nous envisagions dans des projets informatique d'engendrer automatiquement les identificateurs par abréviation des libellés associés. Mais nous pouvons aussi pour évaluer le degré de compréhension et de respect des méthodes automatiques procéder autrement : nous pouvons définir en français une méthode d'abréviation (de la classe de celles proposées) et la donner comme norme de construction des identificateurs, par exemple dans les projets que réalisent les étudiants de 2ème année de l'I.U.T., en demandant qu'à chaque identificateur soit associé le libellé correspondant. Il serait alors intéressant d'utiliser Abrège comme un processeur de contrôle, a posteriori, des normes de travail qui ont été fixées, et d'évaluer selon la nature des normes celles qui sont le plus facilement respectées.

3) - Il est possible aussi d'envisager l'effet de rétro-action de l'outil informatique Abrège sur le processus de définition des libellés. Il serait intéressant, dans un cadre qui reste à définir, de monter le protocole expérimental suivant : fabrication d'une centaine de libellés avec obligation pour ceux qui les ont écrits de fabriquer le code d'identification correspondant, sans imposer aucune méthode particulière. Puis dans une deuxième étape, de proposer aux intéressés des codes d'identification produits par une certaine méthode à partir de ces libellés, en leur demandant de revoir alors les libellés de définition qu'ils ont écrits et éventuellement la méthode utilisée. Il est probable que dans ces conditions les libellés changeront légèrement pour que leur traitement

par la méthode soit le plus efficace possible. Nous aurions ainsi non une mesure, mais une information sur la manière dont les personnes qui prennent en compte une méthode d'abréviation réagissent à celle-ci.

## BIBLIOGRAPHIE

[C1] CHIARAMELLA, Y.

Définition d'un transducteur morpho-phonétique interactif.  
COLING 76. OTTAWA, Juillet 1976.

[C2] COURTIN, J.

Le système PIAF, programme interactif d'analyse du français.  
Séminaire de Programmation. Grenoble, Mars 1975.

[CK1] COULON, D. et KAYSER, D.

Analyse des réponses rédigées en français courant pour une  
réalisation d'enseignement programmé.  
R.A.I.R.O. (6ème année, Juin 1972, B-2, pp. 61-98).

[H1] HINTZMAN, D.L.

Learning and memory in a Discrimination Net.  
Computer studies in the Humanities and Verbal Behavior.  
1-1968, pp. 191-199.

[K1] KAYSER, D.

Etude d'un analyseur et d'un compilateur conçus pour l'enseignement  
assisté.  
Thèse 3ème Cycle. Université de Paris VI, Juin 1970.

[K2] KREMER, M.

Projet "GEOSEMANTICA". Codification sémantique de noms des minéraux.  
C.R.I. Ecole des Mines de Paris. Mars 1971.

[L1] LENCI, M.

Les bases de la codification sémantique.  
C.R.I. Ecole des Mines de Paris. Mars 1974.

[LG1] LOUIS-GAVET, G.

Compactage de données structurées.  
Thèse présentée à l'Université de Lyon I. Juin 1974.

[M1] MARTINET, A.

Eléments de linguistique générale.  
Armand-Colin, Paris, 1970.

- [MS1] MAINE de, P.A.D., and SPRINGER, G.K.  
Details of the SOLID System July 1968.  
Clearinghouse, US Dept., Commerce, Springfield, Va, PB182, p. 476-511.
- [N1] NUGENT, W.R.  
Compression word coding techniques for information retrieval.  
Journal of library Automation, Vol. 1/4, December 1968.
- [P1] PATRICK, E.A.  
Concepts of an Estimation System, an Adaptive System, and a Network  
of Adaptive estimation Systems.  
IEEE. Transactions on systems science and cibernetics. Vol. SCC-5,  
n° 1, Janvier 1969.
- [R1] REITMANN, W.R.  
Cognition and thought.  
New York, Wiley, 1965.
- [R2] ROBERT, M.  
Le Petit Robert, Dictionnaire de la langue française.  
Société du nouveau Littré, 1967, Page XXV.
- [S1] SOCRATE  
Manuel d'opération / Manuel d'utilisation.  
Eca-Automation, Juillet 1974.
- [SK1] SCHWARZ, E.S., and KLEIBOEMER, A.J.  
A language element for Compression Coding.  
Information and Control, Vol. 10 (pp. 315-333), 1967.
- [V1] VAISSIERE, J.  
Contribution à la synthèse par règle du français.  
Thèse de 3ème Cycle. Université de Grenoble III. Novembre 1971.
- [V2] VALABREGUE, H.  
La transcription phonétique par ordinateur des mots français  
écrits et la génération automatique de noms de marque.  
I.B.M. Service développement scientifique. Etude n° FF2-0073.  
Juin 1968.
- [V3] VICKERY, B.C.  
Techniques modernes de documentation. Analyse des systèmes de  
recherche de document.  
DUNOD. Collection sigma. 1962.

**ANNEXE A1**

**ENSEMBLE DES LIBELLES. PROJET G3**





18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100																	
DESIGNATION DES INFORMATIONS																																																																																																			
LIBELLE																				LIBELLE																				LIBELLE																																																											
APPLICATION																				APPLICATION																				APPLICATION																																																											
DATE																				DATE																				DATE																																																											
AN																				AN																				AN																																																											
MOIS																				MOIS																				MOIS																																																											
NUMERO DE PIECE																				NUMERO DE PIECE																				NUMERO DE PIECE																																																											
SENS																				SENS																				SENS																																																											

28  
30  
31  
32  
33  
34

- A1.2

Code Entrées

Longueur

N° de l'inter.

Nom de l'information

Matrice

Chaine Application	DESIGNATION DES INFORMATIONS										N° de l'informa-tion	N° de l'informa-tion	Lettre				
	1	2	3	4	5	6	7	8	9	0							
55												45	50	60	70	7873	78
56												CQED					
57												ATEL					
58												TAXE					
59												TTAXE					
60												ATAXE					
61												CT					
62												SOLD					
63												ENPAR					
64												CPLAN					
65												REGIN					
66												VDP					
67												MAJFI					
68												MAJAC					
69												MAJAA					
70												TSE					
71												MAJFE					
72												MAJFC					
73												TINT					
74												TFIMP					
75												TFDEP					
76												TFIFA					
77												TFDEA					
78												TFIMA					
79												TFDMA					

DESIGNATION DES INFORMATIONS

CREDIT  
 ATELIER  
 TAXE MONTANT  
 TAXE TAUX  
 TAXE ASSIETTE  
 TAXE NATURE  
 MONTANT SOLDE  
 CODE EN QU PAR ATELIER  
 CODE PLAN  
 REGIME D'EXPLOITATION  
 TEST VILLE DE PARIS  
 MAJORATION FRAIS INDIVIS  
 MAJORATION APPROVIS. COMPTE  
 MAJORATION APPROVIS. ARTICLE  
 TRAVAUX SERVICES FACTURABLES  
 FRAIS EQUIPEMENT  
 MAJORATION FRAIS GENERAUX CENT  
 TEST INTERETS INTERCALAIRE  
 FISCALITE IMPOSITION  
 FISCALITE DEDUCTION  
 FISCAL FACTURE IMPOSITION  
 FACTURE DE DEDUCTION  
 SORTIES MAG. IMPOS  
 SORTIES MAG. DEDUC

Application	DESIGNATION DES INFORMATIONS	2	40	45	N° de l'inf.	Len- gueur	50	60	70	7275	75
63	TEST FISCAL	9		TFIAT		1					
64	TRAVAUX ATE. IMP	9		TFDAT		1					
65	TRAVAUX ATE. DEDUC	9		TFIBA		1					
66	ENTRETIEN BAT. IMP	9		TFDBA		1					
67	ENTRETIEN BAT. DED	9		TFIVE		1					
68	AMENAG. VEHIC. IMP	9		TFDVE		1					
69	AMENAG. VEHIC. DED	9		TFIPI		1					
70	PROD. IMMOB. IMP	9		TFDPI		1					
71	PROD. IMMOB. DEDUC	9		TFIEI		1					
72	EMPLS INTERN. IMP	9		TFDEI		1					
73	EMPLS INTERN. DED	9		CTFAC		1					
74	CODE TAXE SUR FACTURE	8		CTART		1					
75	CODE TAXE SUR PARTICLE	8		MAJ1		1					
76	MALGRATION 17	9		FIX		1					
77	FIXITE	9									

A1.4



\*\* SOCRATE STARTED \*\* V1.6 LE 133.76 A 22.41.14

1 FXFC CAL-EDIT-LIB  
 2 ?  
 3 0001  
 4 0001  
 5 0085

ANNEXE A1.2

RESULTAT SANS EMPLOI  
 DE DICTIONNAIRES

METHODE NUMERO 1

ORGANISME/  
 L 3 ABREVIATION : ORG

ORGANISME#FISCO-COMPTABLE/  
 L 5 ABREVIATION : ORGFC

ORGANISME#ABOUTISSEMENT/  
 L 5 ABREVIATION : ORGAB

UNITE#TECHNIQUE/  
 L 3 ABREVIATION : UTE

UNITE#TECHNIQUE#IMPUTATION/  
 L 5 ABREVIATION : UTEIM

UNITE#TECHNIQUE#ORIGINE/  
 L 4 ABREVIATION : UTEO

UNITE##TECHNIQUE##ABOUTISSEMENT##/  
 L 5 ABREVIATION : UTFAB

COMPTE/  
 L 4 ABREVIATION : COMP

COMPTE#IMPUTATION/  
 L 2 ABREVIATION : CIM

COMPTE#CONTRE#PARTIE/  
 L 2 ABREVIATION : CP

COMPTE#ABOUTISSEMENT/  
 L 2 ABREVIATION : CAB

COMPTE#PARTICULIER/  
 L 3 ABREVIATION : CPA

CODE#MOUVEMENT/  
 L 3 ABREVIATION : CMO

SOUS#COMPTE#INTERNE/  
 L 2 ABREVIATION : CI

CODE#ANALYSE#IMPUTATION/  
 L 4 ABREVIATION : CAIM

CODE#ANALYSE#CONTRE#PARTIE/

L 3 ABREVIATION : CAP  
CODE#ANALYSE#ABOUTISSEMENT/  
L 4 ABREVIATION : CAAB  
ACTE#ELEMENTAIRE#IMPUTATION/  
L 5 ABREVIATION : AELIM  
ACTE#ELEMENTAIRE#CONTRE#PARTIE/  
L 4 ABREVIATION : AELP  
CODE#ELEMENTAIRE/  
L 3 ABREVIATION : CEL  
CODE#ELEMENTAIRE#IMPUTATION/  
L 5 ABREVIATION : CELIM  
CODE#ELEMENTAIRE#CONTRE#PARTIE/  
L 4 ABREVIATION : CELP  
CODE#ELEMENTAIRE#ABOUTISSEMENT/  
L 5 ABREVIATION : CELAB  
CODE#STATISTIQUE/  
L 3 ABREVIATION : CST  
QUANTITE/  
L 4 ABREVIATION : QUAN  
MONTANT/  
L 4 ABREVIATION : MONT  
LIBELLE/  
L 3 ABREVIATION : LIB  
APPLICATION/  
L 4 ABREVIATION : APPL  
DATE/  
L 4 ABREVIATION : DATE  
AN/  
L 2 ABREVIATION : AN  
MOIS/  
L 4 ABREVIATION : MOIS  
NUMERO#DE#PIECE/  
L 2 ABREVIATION : NP  
SENS/  
L 4 ABREVIATION : SENS  
CREDIT/  
L 4 ABREVIATION : CRED  
ATELIER/  
L 4 ABREVIATION : ATEL  
TAXE#MONTANT/

L 2 ABREVIATION : TM  
TAXE#TAUX/  
L 2 ABREVIATION : TT  
TAXE#ASSIETTE/  
L 3 ABREVIATION : TAS  
TAXE#NATURE/  
L 2 ABREVIATION : TN  
MONTANT#SOLDE/  
L 2 ABREVIATION : MS  
CODE#EN#OU#PAR#ATELIER/  
L 2 ABREVIATION : CA  
CODE#PLAN/  
L 2 ABREVIATION : CP  
REGIME#D#EXPLOITATION/  
L 3 ABREVIATION : REX  
TEST#VILLE#DE#PARIS/  
L 3 ABREVIATION : TVP  
MAJORATION#FRAIS#INDIVISIBLE/  
L 5 ABREVIATION : MAJFI  
MAJORATION#APPROVISIONNEMENT#COMPTE/  
L 5 ABREVIATION : MAJAP  
MAJORATION#APPROVISIONNEMENT#ARTICLE/  
L 5 ABREVIATION : MAJAP  
TRAVAUX#SERVICES#FACTURABLES/  
L 5 ABREVIATION : TSEFA  
FRAIS#EQUIPEMENT/  
L 3 ABREVIATION : FEQ  
MAJORATION#FRAIS#GENERAUX#CENT/  
L 5 ABREVIATION : MFGEC  
TEST#INTERETS#INTERCALAIRE/  
L 5 ABREVIATION : TININ  
TEST#FISCALITE#IMPOSITION/  
L 5 ABREVIATION : TFIIM  
TEST#FISCALITE#DEDUCTION/  
L 5 ABREVIATION : TFIDE  
TEST#FISCAL#FACTURE#IMPOSITION/  
L 5 ABREVIATION : TFFIM  
TEST#FISCAL#FACTURE#DEDUCTION#/  
L 5 ABREVIATION : TFFDE  
TEST#FISCAL#SORTIES#MAGASIN#IMPOSITION/



L 5 ABREVIATION : TFSMI  
TEST#FISCAL#SORTIES#MAGASIN#DEDUCTION/  
L 5 ABREVIATION : TFSMD  
TEST#FISCAL#TRAVAUX#ATELIER#IMPOSITION/  
L 5 ABREVIATION : TFTAI  
TEST#FISCAL#TRAVAUX#ATELIER#DEDUCTION/  
L 5 ABREVIATION : TFTAD  
TEST#FISCAL#ENTRETIEN#BATIMENT#IMPOSITION/  
L 5 ABREVIATION : TFENB  
TEST#FISCAL#ENTRETIEN#BATIMENT#DEDUCTION/  
L 5 ABREVIATION : TFENB  
TEST#FISCAL#AMENAGEMENT#VEHICULE#IMPOSITION/  
L 5 ABREVIATION : TFAVI  
TEST#FISCAL#AMENAGEMENT#VEHICULE#DEDUCTION/  
L 5 ABREVIATION : TFAVD  
TEST#FISCAL#PRODUCTION#IMMOBILIERE#IMPOSITION/  
L 5 ABREVIATION : TFPII  
TEST#FISCAL#PRODUCTION#IMMOBILIERE#DEDUCTION/  
L 5 ABREVIATION : TFPID  
TEST#FISCAL#EMPLOIS#INTERNES#IMPOSITION/  
L 5 ABREVIATION : TFEII  
TEST#FISCAL#EMPLOIS#INTERNES#DEDUCTION/  
L 5 ABREVIATION : TFEID  
CODE#TAXE#SUR#FACTURE/  
L 3 ABREVIATION : CTF  
CODE#TAXE#SUR#ARTICLE/  
L 3 ABREVIATION : CTA  
MAJORATION#1%/  
L 4 ABREVIATION : MAJI  
FIXITE/  
L 3 ABREVIATION : FIX  
MATRICULE#AGENT/  
L 4 ABREVIATION : MATA  
NOM#DE#L'AGENT/  
L 2 ABREVIATION : NA  
CODE#IMPUTATION#ANALYTIQUE#GAZ#DE#FRANCE/  
L 5 ABREVIATION : CIMAN  
SERVICE#EMPLOYEUR/  
L 3 ABREVIATION : SEM  
CODE#CARTE/

L 2 ABBREVIATION : CC

CODE#RUBRIQUE/

L 3 ABBREVIATION : CRU

CODE#HEURE/

L 2 ABBREVIATION : CH

CODE#HEURE#CMVT#991/

L 5 ABBREVIATION : CHCMV

CODE#HEURE#CMVT#994/

L 5 ABBREVIATION : CHCMV

NOMBRE#D"HEURES/

L 2 ABBREVIATION : NH

NOMBRE#D"HEURES#CMVT#991/

L 5 ABBREVIATION : NHCMV

NOMBRE#D"HEURES#CMVT#994/

L 5 ABBREVIATION : NHCMV

CODE#RUBRIQUE#OU#CODE#HEURE/

L 5 ABBREVIATION : CRUCH

MONTANT#OU#NOMBRE#D"HEURES/

L 3 ABBREVIATION : MNH

FIN METHODE 1

% LOGOUT

LOGOUT BASE: ABRSOCl

\*\* SOCRATE STARTED \*\* V1.6 LE 133.76 A 23.44.41

1 EXEC AUTOMATE-ANALYSE  
2 ? ANNEXE A1.3

3 \*\*LIBELLE  
MODIFICATION DES DONNEES  
4 \*MOD POUR RESULTAT2

5 \*UN 74 \*PAR

6 \*\*CODE IMPUTATION ANALYTIQUE GDF\*\*

7 \*DE 54 67 \*PAR

8 \*\*"TEST FISCAL IMPOSITION FACTURE\*\*

9 \*\*"TEST FISCAL DEBUCTION FACTURE\*\*

10 \*\*"TEST FISCAL IMPOSITION SORTIE MAGASIN\*\*

11 \*\*"TEST FISCAL DEDUCTION SCRTIE MAGASIN\*\*

12 \*\*"TEST FISCAL IMPOSITION TRAVAUX ATELIER\*\*

13 \*\*"TEST FISCAL DEDUCTION TRAVAUX ATELIER\*\*

14 \*\*"TEST FISCAL IMPOSITION ENTRETIEN BATIMENT\*\*

15 \*\*"TEST FISCAL DEDUCTION ENTRETIEN BATIMENT\*\*

16 \*\*"TEST FISCAL IMPOSITION AMENAGEMENT VEHICULE\*\*

17 \*\*"TEST FISCAL DEDUCTION AMENAGEMENT VEHICULE\*\*

18 \*\*"TEST FISCAL IMPOSITION PRODUCTION IMMOBILIERE\*\*

19 \*\*"TEST FISCAL DEDUCTION PRODUCTION IMMOBILIERE\*\*

20 \*\*"TEST FISCAL IMPOSITION EMPLOIS INTERNES\*\*

21 \*\*"TEST FISCAL DEDUCTION EMPLOIS INTERNES\*\*

\*\* SOCRATE STARTED \*\* V1.6 LE 133.76 A 23.47.18

1 EXEC TRACE-DIC DICTIONNAIRES  
2 ?

Les éléments utiles à G3 sont  
indiqués avec une →

3 SPECIAL

DICTIONNAIRE : SPECIAL

BE	BE
SCC	SCC
ORIGINAL	...
BONS	B
ENGAGEMENT	E
PSP	PSP
EXEMPLAIRES	EX
DUPLICATA	...
SERDM	...
CREDIT	-CR
DECAUDINGEORGES	DECAUDIN
ADMINISTRATEUR	AD
ADMINISTRATEURS	A
TYPE1	T1
DIVERSES	...
CONCERNANT	...
→ TECHNIQUE	T
→ COMPTE	CPT
→ MOUVEMENT	MVT
→ CONTREPARTIE	CP
→ IMPUTATION	I
→ ELEMENTAIRE	E

→	NUMERO	...
→	TAXE	TX
→	CODE	C
→	MAJORATION	MAJ
→	GENERAUX	...
	CMVT	...
→	1%	1%
→	GDF	GDF
→	991	1
→	994	4
	QUANTITE	QUANT
	PIECE	PIECE
→	NOMBRE	...
→	HEURES	HE
→	STATISTIQUE	STAT
→	EXPLOITATION	EXP
→	FISCALITE	F
	DCE	...
	TITRE	T
	CRFANCE	C
	TITRES	T

4  
FIN EDITION  
RUN FN:R

\*\*\*

\*\* SECURITE STARTED \*\* V1.6      LE 133.76      A 23.47.23

1      EXEC CAL-EDIT-LIB  
2      ?  
3      0001  
4      0001  
5      0085

METHODE1

RESULTAT2

METHODE NUMERO 1

ORGANISME/

L 3      ABREVIATION : ORG

ORGANISME#FISCO-COMPTABLE/

L 5      ABREVIATION : ORGFC

ORGANISME#ABOUTISSEMENT/

L 5      ABREVIATION : ORGAB

UNITE#TECHNIQUE/

L 2      ABREVIATION : UT

UNITE#TECHNIQUE#IMPUTATION/

L 3      ABREVIATION : UTI

UNITE#TECHNIQUE#ORIGINE/

L 3      ABREVIATION : UTO

UNITE##TECHNIQUE##ABOUTISSEMENT##/

L 4      ABREVIATION : UTAB

COMPTE/

L 3      ABREVIATION : CPT

COMPTE#IMPUTATION/

L 4      ABREVIATION : CPTI

COMPTE#CONTRE#PARTIE/

L 5      ABREVIATION : CPTCP

COMPTE#ABOUTISSEMENT/

L 5      ABREVIATION : CPTAB

COMPTE#PARTICULIER/

L 5      ABREVIATION : CPTPA

CODE#MOUVEMENT/

L 4      ABREVIATION : CMVT

SOUS#COMPTE#INTERNE/

L 4      ABREVIATION : CPTI

CODE#ANALYSE#IMPUTATION/

L 3      ABREVIATION : CAI

CODE#ANALYSE#CONTRE#PARTIE/

L 4 ABREVIATION : CACP  
CODE#ANALYSE#ABOUTISSEMENT/  
L 4 ABREVIATION : CAAB  
ACTE#ELEMENTAIRE#IMPUTATION/  
L 3 ABREVIATION : AEI  
ACTE#ELEMENTAIRE#CONTRE#PARTIE/  
L 4 ABREVIATION : AECF  
CODE#ELEMENTAIRE/  
L 2 ABREVIATION : CE  
CODE#ELEMENTAIRE#IMPUTATION/  
L 3 ABREVIATION : CEI  
CODE#ELEMENTAIRE#CONTRE#PARTIE/  
L 4 ABREVIATION : CFCF  
CODE#ELEMENTAIRE#ABOUTISSEMENT/  
L 4 ABREVIATION : CEAB  
CODE#STATISTIQUE/  
L 5 ABREVIATION : CSTAT  
QUANTITE/  
L 5 ABREVIATION : QUANT  
MONTANT/  
L 4 ABREVIATION : MONT  
LIBELLE/  
L 3 ABREVIATION : LIB  
APPLICATION/  
L 4 ABREVIATION : APPL  
DATE/  
L 4 ABREVIATION : DATE  
AN/  
L 2 ABREVIATION : AN  
MOIS/  
L 4 ABREVIATION : MOIS  
NUMERO#DE#PIECE/  
L 5 ABREVIATION : PIECE  
SENS/  
L 4 ABREVIATION : SENS  
CREDIT/  
L 4 ABREVIATION : CRED  
ATELIER/  
L 4 ABREVIATION : ATEL  
TAXE#MONTANT/

L 3 ABREVIATION : TXM  
TAXE#TAUX/  
L 3 ABREVIATION : TXT  
TAXE#ASSIETTE/  
L 4 ABREVIATION : TXAS  
TAXE#NATURE/  
L 3 ABREVIATION : TXN  
MONTANT#SOLDE/  
L 2 ABREVIATION : MS  
CODE#EN#OU#PAR#ATELIER/  
L 2 ABREVIATION : CA  
CODE#PLAN/  
L 2 ABREVIATION : CP  
REGIME#D#EXPLOITATION/  
L 4 ABREVIATION : REXP  
TEST#VILLE#DE#PARIS/  
L 3 ABREVIATION : TVP  
MAJORATION#FRAIS#INDIVISIBLE/  
L 5 ABREVIATION : MAJFI  
MAJORATION#APPROVISIONNEMENT#COMPTE/  
L 5 ABREVIATION : MAJAC  
MAJORATION#APPROVISIONNEMENT#ARTICLE/  
L 5 ABREVIATION : MAJAA  
TRAVAUX#SERVICES#FACTURABLES/  
L 5 ABREVIATION : TSEFA  
FRAIS#EQUIPEMENT/  
L 5 ABREVIATION : MAJFE  
MAJORATION#FRAIS#GENERAUX#CENT/  
L 5 ABREVIATION : MAJFC  
TEST#INTERETS#INTERCALAIRE/  
L 5 ABREVIATION : TININ  
TEST#FISCALITE#IMPOSITION/  
L 3 ABREVIATION : TFI  
TEST#FISCALITE#DEDUCTION/  
L 3 ABREVIATION : TFD  
TEST#FISCAL#IMPOSITION#FACTURE/  
L 4 ABREVIATION : TFIF  
TEST#FISCAL##DEDUCTION#FACTURE/  
L 4 ABREVIATION : TDFD  
TEST#FISCAL#IMPOSITION#SORTIE#MAGASIN/



L 5 ABREVIATION : TFISM  
TEST#FISCAL#DEDUCTION#SORTIE#MAGASIN/  
L 5 ABREVIATION : TFD SM  
TEST#FISCAL#IMPOSITION#TRAVAUX#ATELIER/  
L 5 ABREVIATION : TFITA  
TEST#FISCAL#DEDUCTION##TRAVAUX#ATELIER/  
L 5 ABREVIATION : TFDTA  
TEST#FISCAL#IMPOSITION#ENTRETIEN#BATIMENT/  
L 5 ABREVIATION : TFIEB  
TEST#FISCAL#DEDUCTION#ENTRETIEN#BATIMENT/  
L 5 ABREVIATION : TFDEB  
TEST#FISCAL#IMPOSITION#AMENAGEMENT#VEHICULE/  
L 5 ABREVIATION : TFIAV  
TEST#FISCAL#DEDUCTICN#AMENAGEMENT#VEHICULE/  
L 5 ABREVIATION : TFDV  
TEST#FISCAL#IMPOSITION#PRODUCTION#IMMOBILIERE/  
L 5 ABREVIATION : TFIPI  
TEST#FISCAL#DEDUCTION#PRODUCTION#IMMOBILIERE/  
L 5 ABREVIATION : TFDPI  
TEST#FISCAL#IMPOSITION#EMPLOIS#INTERNES/  
L 5 ABREVIATION : TFIEI  
TEST#FISCAL#DEDUCTION#EMPLOIS#INTERNES/  
L 5 ABREVIATION : TFDEI  
CODE#TAXE#SUR#FACTURE/  
L 4 ABREVIATION : CTXF  
CODE#TAXE#SUR#ARTICLE/  
L 4 ABREVIATION : CTXA  
MAJORATION#1%/  
L 5 ABREVIATION : MAJ1% ← Modifier si  
le langage refuse %  
FIXITE/  
L 3 ABREVIATION : FIX  
MATRICULE#AGENT/  
L 5 ABREVIATION : MATAG  
NOM#DE#L\*AGENT/  
L 3 ABREVIATION : NAG  
CODE#IMPUTATION#ANALYTIQUE#GDF/  
L 4 ABREVIATION : CIAG  
SERVICE#EMPLOYEUR/  
L 3 ABREVIATION : SEM  
CODE#CARTE/

L 2 ABREVIATION : CC

CODE#RUBRIQUE/

L 3 ABREVIATION : CRU

CODE#HEURE/

L 3 ABREVIATION : CHE

CODE#HEURE#CMVT#991/

L 4 ABREVIATION : CHE1

CODE#HEURE#CMVT#994/

L 4 ABREVIATION : CHE4

NOMBRE#D#HEURES/

L 2 ABREVIATION : HE

NOMBRE#D#HEURES#CMVT#991/

L 3 ABREVIATION : HE1

NOMBRE#D#HEURES#CMVT#994/

L 3 ABREVIATION : HE4

CODE#RUBRIQUE#OU#CODE#HEURE/

L 5 ABREVIATION : CRUCH

MONTANT#OU#NOMBRE#D#HEURES/

L 3 ABREVIATION : MHE

FIN METHODE 1

? LOGOUT

LOGOUT BASE: ABRSOC1

← A Améliorer ex  
supprimant NOMBRE  
comme mot à abréviation  
vide dans SPECIAL



**ANNEXE A2**

**ENSEMBLE DES LIBELLES. PROJET DM**

AGE	CODE	LIBELLE
***	****	*****
	<b>ANOMALIES ( § 2.6.1 )</b>	
48	AC0DCERT	* CODIFICATION DE LA CERTIFICATION *
49	AC0DFACEQUI	* CODIFICATION FACTURE +TC *
A1 50	AC0DFACF0NC	* CODIFICATION FACTURE +TC *
A1 51	AC0NTCEREQUI	* CONTR0LES 0RD0NNATEUR SUR CERTIFICATION *
A1 52	AC0NTCERF0NC	* CONTR0LES 0RD0NNATEUR SUR CERTIFICATION *
A1 53	AC0NTFACF0NC	* CONTR0LES 0RD0NATEUR SUR FACTURE+TC *
A1 54	AC0NT0RD1	* CONTR0LES 0RD0NNATEUR SUR LESB0NS D'ENGAGEMENT *
A2 A3 55	AC0NTREQUI	* CONTR0LE 0RD0NATEUR SUR FACTURE+TC *
A1 56	AC0NTRES0R	* CONTR0LE TRES0R, VISA, PREC0MPTE EVENTUEL *
57	ADIFF1BUC0	* DIFFUSION DES BE ET DES PSP A LEURS DESTINATAIRES
A2 A3 6	AD0PBUD	* AD0PTION DU BUDGET *
58	AEMISFAC	* EMISSION DE LA FACTURE EN 3 EXEMPLAIRES *
A4 59	AENBENGDB	* ENVOI DES BE A LA DCE_DB *
A3 60	AENBENGDBSCC	* ENVOI DES BE VISES AU SCC *
A1 61	AENBENGDI	* ENVOI DES BE A LA DCE_DI *
A3 62	AENBENGDISCC	* ENVOI DES BE VISES AU SCC *
A1 63	AENBENGSCCSI	* ENVOI DU DUPLICATA DU BE AU SERVICE INFORMATIQUE
A3 64	AENBEPSPBUC0	* ENVOI DES BE ET DES PSP AU BUREAU COURRIER SERDM
A3 A4 65	AENBEPSPSCC	* ENVOI DU BE ET DES PSP AU SCC *
A3 66	AENREGBENG	* ENREGISTREMENT DE L'ARRIVEE DEBE *
A4 67	AENVBENGBUC0	* ENVOI DES BE AU BUREAU COURRIER SERDM *
A3 A4 68	AENVOICERT	* ENVOI CERTIF+FACTURE A SERDM *
A4 69	AENVOIFACSCC	* ENVOI FACTURE+TC+BE AU SERDM *
A2 70	AENVOIFACSER	* ENVOI FACTURE AU SERVICE EMETTEUR *
A4 71	AENVOINF0	* ENVOI AU CENTRE INFORMATIQUE *
A4 72	AENVOITREE	* ENVOI DU DOSSIER AU TRESOR *
A1 73	AENVOITREF	* ENVOI DU DOSSIER AU TRESOR *
A1	A4	

138 20/07/75 \*\* DM \*\* P LISTE ALPHABETIQUE DES CODES :

- AETABBENG \* ETABLISSEMENT DES BE \*
- A3 AGARBENG1SCC \* ARCHIVAGE DU BE ORIGINAL AU SCC \*
- A3 A4 APERFBENG \* PERFORATION DU DUPLICATA DU BE \*
- A3 A4 APERFØ \* PERFORATION TC \*
- APRØPDEP \* PRØPOSITION DE DEPENSE A L'ADMINISTRATEUR DE CREDI
- A3 A4 AREJCEREQUI \* REJET CERTIFICATION \*
- AREJCERFØNC \* REJET CERTIFICATION \*
- AREJFACEQUI \* REJET FACTURE+TC \*
- AREJFACFØNC \* REJET FACTURE+TC \*
- AREJPRØPDEP \* REJET DE LA PRØPOSITION DE DEPENSE \*
- AREJTRESØR \* REJET TRESØR DE LA FACTURE ØUDU TC \*
- A4 ARETØURCERE \* RENVØI DE LA CERTIFICATION ERRØNEE AU SERVICE CØUR
- A4 ARETØURCERF \* RENVØI DE LA CERTIFICATION ERRØNEE AU SERVICE CØUR
- A4 ARETØURFACE \* RENVØI DU DØSSIER ERRØNEE AUSERVICE CØURRIER \*
- A4 ARETØURFACF \* RENVØI DU DØSSIER ERRØNE AU SERVICE CØURRIER \*
- A4 ARETØURFØUR \* RENVØI DU DØSSIER ERRØNE AU BENEFICIAIRE \*
- A4 ARETØURSERDM \* RENVØI AU SERDM DES DOCUMENTSINFORMATIQUES \*
- A4 AUTØ3 \* GROUPE FONCTIONNEL AUTØMATIQUE \*
- AVENTIL \* VENTILATION DES DIFFERENTS DØCUMENTS \*
- A4 AVENTILEQUIP \* ENVØI DU DØSSIER A DCF-DI \*
- A2 A3 AVENTILFØNC \* ENVØI DU DØSSIER A DCF-DB \*
- A2 A3 AVISABENGDB \* VISA DES BE PAR LA DCF-DB \*
- A3 AVISABENGDI \* VISA DU BE PAR LA DCE,DI \*
- A3 AVISABENGSCC \* VISA DES BE PAR LE TRESØR \*
- A2 A3 CCØNTAUTØFØN \* CØNTRØLE EXISTENCE ET RYTHME DE CONSØMMATION DES C
- A2 CCØNTAUTØFØR \* CØNTRØLES AUTØMATIQUES SUR LAFØRME \*
- CENREGTC \* ENREGISTREMENT TC DANS FICHIERENGAGEMENT \*
- A4

138 20/07/75 \*\* DM \*\* P LISTE ALPHABETIQUE DES CODES :

118		CMANDEMENT	* ETABLISSEMENT DES TITRES DE REGLEMENT *
	A2		
119		CONTRCERT	* CONTRÔLE CERTIFICATION *
120		CONTRTC	* CONTRÔLE TITRE DE CREANCE *
121		CREJTC	* REJET TITRE DE CREANCE *
122		CRELANCERCERT	* RELANCE EN VUE D'OBTENIR LA CERTIFICATION *
123		CTRAITENG	* TRAITEMENT DU BE *
	A3		
125		DANG	* LISTE DES ANOMALIES *
111		DECAUDIN	* DECAUDINGEORGES *
7		DM	* DEPENSES DE MATERIEL *
126		DRELANCECERT	* RELANCE CERTIFICATION *
127	A3	DTC	* TITRE DE CREANCE *
9		ELABUD	* ELABORATION DU BUDGET *
10		EXECBUD	* EXECUTION DU BUDGET *
99		GADMCRFD	* ADMINISTRATEUR DE CREDIT *
	A3		
100		GBUCOUSERDM	* BUREAU DU COURRIER DU SERDM *
101		GCENTRINF0	* CENTRE INFORMATIQUE *
102		GCHEFSERV	* CHEF DE SERVICE *
103		GOCEDB	* DIVISION CONTRÔLE DES ENGAGEMENTS DE LA DB *
104		GOCEDI	* DIVISION CONTRÔLE DES ENGAGEMENTS DE LA DI *
105		GF0URNISSEUR	* FOURNISSEUR OU BENEFICIAIRE DEL'ENGAGEMENT *
106		GPERF0	* GROUPE DE PERFORATION *
107		GSCC	* SERVICE COMPTABLE CENTRAL *
108		GSERVINF0	* SERVICE INFORMATIQUE *
109		GTRES0R	* TRESOR *
13		LOIREGLEMENT	* ADOPTION DE LA LOI DE REGLEMENT *
14		MADM CREDIT	* CHOIX DES ADMINISTRATEURS DE CREDIT *
	A3		
15		MANNULENG	* ANNULATION DE L'ENGAGEMENT PAR L'AC *
	A3		

140 20/07/75 \*\* DM \*\* P LISTE ALPHABETIQUE DES CODES :

	MAPURENG	* APPUREMENT DU DOSSIER D'ENGAGEMENT *
	MARCHE	* PASSATION ET REGLEMENT DE MARCHES *
	MCNTORDDI1	* CONTRÔLES ORDONNATEUR DI DE TYPE 1 *
	MCNTSCC1	* CONTRÔLES TRESOR DE TYPE 1 *
A2	MDELEGREDIT	* DELEGATION DE CREDIT *
	MDPS	* PROCEDURE SIMPLIFIEE *
	MENGNORMAL	* ENGAGEMENT DE DEPENSE *
A2	MED	* OPERATIONS DIVERSES CONCERNANT LE BUDGET *
A2	MPAIEMENT	* REGLEMENT DE LA DEPENSE *
A2	MPRONORMALE	* PROCEDURE NORMALE *
	MREJAUT01	* REJET AUTOMATIQUE POUR VICE DEFORME *
A2	MREJAUT02	* REJET ABSENCE DE CREDITS OU CONSOMMATION TROP RAP
A2	MREJDB1	* REJET PAR LA DB DE LA PROPOSITION D'ENGAGEMENT *
A2	MREJDI1	* REJET PAR LA DI DE LA PROPOSITION D'ENGAGEMENT *
A2	MREJSCC1	* REJET PAR LE TRESOR DE LA PROPOSITION D'ENGAGEMENT
A2	MRESCREDIT	* RESERVATION DE CREDITS *
	MSUITEREJ	* SUITE A REJET ABSENCE CREDITS OU CONSOMMATION TROP RAPIDE

Les abréviations codées par l'utilisateur sont systématiquement précédées de préfixes tenant compte du rôle du code dans l'application. Nous n'en tenons pas compte

140 20/07/75 \*\* DM \*\* P LISTE ALPHABETIQUE DES CODES :



\*\* SCCRATE STARTED \*\* V1.6 LE 133.76 A 22.44.16

1 EXEC TRACE-DIC  
2 ?

DICTIONNAIRES

3 SPECCOURT

DICTIONNAIRE : SPECCOURT

BE	-BE
-DI	-DI
-DB	-DB
DI	-DI
DB	-DB
AC	-AC
+TC	-TC
TC	-TC
EVENTUEL	...
VISA	AC
PRECOMPTE	...

4 SPECIAL

DICTIONNAIRE : SPECIAL

BE	BE
SCC	SCC
ORIGINAL	...
BONS	B
ENGAGEMENT	E
PSP	PSP

EXEMPLAIRES	EX
DUPLICATA	...
SERDM	...
CREDIT	-CR
DECAUDINGEORGES	DECAUDIN
ADMINISTRATEUR	AD
ADMINISTRATEURS	A
TYPE1	T1
DIVERSES	...
CONCERNANT	...
TECHNIQUE	T
COMPTE	CPT
MOUVEMENT	MVT
CONTREPARTIE	CP
IMPUTATION	I
ELEMENTAIRE	E
NUMERO	...
TAXE	TX
CODE	C
MAJORATION	MAJ
GENERAUX	...
CMVT	...
12	12
GDF	GDF
991	1
994	4
QUANTITE	QUANT
PIECE	PIECE
NOMBRE	...
HEURES	HE

STATISTIQUE	STAT
EXPLOITATION	EXP
FISCALITE	F
OCE	...
TITRE	T
CREANCE	C
TITRES	T

5  
FIN EDITION  
RUN FN:R

\*\*\*

8

\*\* SCCRATE STARTED \*\* V1.6 LE 133.76 A 22.44.22

1	EXEC CAL-EDIT-LIB	RESULTAT
2	?	METHODE2
3	0002	
4	0086	
5	0172	

METHODE NUMERO 2

ETABLISSEMENT#DES#BE/  
L 5 ABREVIATION : ETABE

ARCHIVAGE#DU#BE#ORIGINAL#AU#SCC/  
L 8 ABREVIATION : ARCHBESC

CODIFICATION#DE#LA#CERTIFICATION/  
L 7 ABREVIATION : CODCERT

CODIFICATION#FACTURE+TC/  
L 7 ABREVIATION : CODFACT

CONTROLES#ORDONNATEUR#SUR#CERTIFICATION/  
L 1 ABREVIATION : CONTORDCERT

CONTROLES#ORDONNATEUR#SUR#LES#BONS#D'ENGAGEMENTS/  
L 9 ABREVIATION : CONTORDBE

CONTROLE#ORDONNATEUR#SUR#FACTURE+TC/  
L 1 ABREVIATION : CONTORDFACT

CONTROLE#TRESOR,VISA,PRECOMPTE#EVENTUEL/  
L 7 ABREVIATION : CONTRAC

DIFFUSION#DES#BE#ET#DES#PSP#A#LEURS#DESTINATAIRES/  
L 1 ABREVIATION : DIFBEPSPDES

ADCEPTION#DU#BUDJET/  
L 8 ABREVIATION : ADOPBUDG

EMISSION#DE#LA#FACTURE#EN#3#EXEMPLAIRES/  
L 8 ABREVIATION : EMISFAC3

ENVOI#DES#BE#A#LA#DCE-DB/  
L 8 ABREVIATION : ENVBE-DB

ENVOI#DES#BE#VISES#AU#SCC/  
L 9 ABREVIATION : ENVBEVISC

ENVOI#DES#BE#A#LA#DCE-DI/  
L 8 ABREVIATION : ENVBE-DI

ENVOI#DU#DUPLICATA#DU#BE#AU#SERVICE#INFORMATIQUE/  
L 1 ABREVIATION : ENVBESERINF

ENVOI#DES#BE#ET#DES#PSP#AU#BUREAU#COURRIER#SERDM/

L 1 ABBREVIATION : ENVBEPSPBURCO

ENVOI#DU#BE#ET#DES#PSP#AU#SCC/  
L 1 ABBREVIATION : ENVBEPSPSC

ENREGISTREMENT#DE#L'ARRIVEE#DE#BE/  
L 7 ABBREVIATION : ENRARBE

ENVOI#DES#BE#AU#BUREAU#COURRIER#SERDM/  
L 1 ABBREVIATION : ENVBEBURCO

ENVOI#CERTIF+FACTURE#A#SERDM/  
L 7 ABBREVIATION : ENVCERT

ENVOI#FACTURE+TC+BE#AU#SERDM/  
L 7 ABBREVIATION : ENVFACT

ENVOI#FACTURE#AU#SERVICE#EMETTEUR/  
L 1 ABBREVIATION : ENVFACSEREM

ENVOI#AU#CENTRE#INFORMATIQUE/  
L 1 ABBREVIATION : ENVCENTINF

ENVOI#DU#DOSSIER#AU#TRESOR/  
L 1 ABBREVIATION : ENVDOSTRES

PERFORATION#DU#DUPLICATA#DU#BE/  
L 6 ABBREVIATION : PERFBE

PERFORATION#TC/  
L 6 ABBREVIATION : PERTTC

PROPOSITION#DE#DEPENSE#A#L'ADMINISTRATEUR#DE#CREDIT/  
L 1 ABBREVIATION : PROPDEPAD-CR

REJET#CERTIFICATION/  
L 7 ABBREVIATION : REJCERT

REJET#FACTURE+TC/  
L 7 ABBREVIATION : REJFACT

REJET#DE#LA#PROPOSITION#DE#DEPENSE/  
L 1 ABBREVIATION : REJPROPDEP

REJET#TRESOR#DE#LA#FACTURE#OU#DU#TC/  
L 1 ABBREVIATION : REJTRFACTC

RENVOI#DE#LA#CERTIFICATION#ERRONEE#AU#SERVICE#COURRIER/  
L 1 ABBREVIATION : RENVCERERSERCC

RENVOI#DU#DOSSIER#ERRONE#AU#SERVICE#COURRIER/  
L 1 ABBREVIATION : RENVDOSERSERCC

RENVOI#DU#DOSSIER#ERRONE#AU#BENEFICIAIRE/  
L 1 ABBREVIATION : RENVDOSERBEN

RENVOI#AU#SERDM#DES#DOCUMENTS#INFORMATIQUES/  
L 9 ABBREVIATION : RENVDOCIN

VENTILATION#DES#DIFFERENTS#DOCUMENTS/

L 1 ABBREVIATION : VENTDIFDOC  
ENVCID#DU#DOSSIER#A#DCE-DI/  
L 9 ABBREVIATION : ENVCOS-CI  
ENVDI#DU#DOSSIER#A#DCE-DB/  
L 9 ABBREVIATION : ENVCOS-DB  
VISA#DES#BE#PAR#LA#DCE-DB/  
L 8 ABBREVIATION : VISBE-DB  
VISA#DU#BE#PAR#LA#DCE-DI/  
L 8 ABBREVIATION : VISBE-DI  
VISA#DES#BE#PAR#LE#TRESOR/  
L 9 ABBREVIATION : VISBETRES  
CONTROLE#EXISTENCE#ET#RYTHME#DE#CONSUMATION#DES#CREDITS/  
L 1 ABBREVIATION : CONTEXRYCOCR  
CONTROLES#AUTOMATIQUES#SUR#LA#FORME/  
L 1 ABBREVIATION : CONTAUTFORM  
ENREGISTREMENT#TC#DANS#FICHER#ENGAGEMENT/  
L 9 ABBREVIATION : ENRTCFIGE  
ETABLISSEMENT#DES#TI TRES#DE#REGLEMENT/  
L 1 ABBREVIATION : ETABTITREGL  
CONTROLE#CERTIFICATION/  
L 8 ABBREVIATION : CONTCERT  
CONTROLE#TITRE#DE#CREANCE/  
L 5 ABBREVIATION : CONTC  
REJET#TITRE#DE#CREANCE/  
L 5 ABBREVIATION : REJTC  
RELANCE#ENVUE#D'OBTENIR#LA#CERTIFICATION/  
L 1 ABBREVIATION : RELOBTCERT  
TRAITEMENT#DU#BE/  
L 7 ABBREVIATION : TRAITBE  
LISTE#DES#ANOMALIES/  
L 8 ABBREVIATION : LISTANOM  
DECAUDINGEORGES/  
L 8 ABBREVIATION : DECAUDIN  
DEPENSES#DE#MATERIEL/  
L 6 ABBREVIATION : DEPMAT  
RELANCE#CERTIFICATION/  
L 7 ABBREVIATION : RELCERT  
TITRE#DE#CREANCES/  
L 2 ABBREVIATION : TC  
ELABCRATION#DU#BUDGET/

L 7 ABREVIATION : ELABUDG  
EXECUTION#DU#BUDGET/  
L 8 ABREVIATION : EXECBUDG  
ADMINISTRATEUR#DE#CREDIT/  
L 5 ABREVIATION : AD-CR  
BUREAU#DU#COURRIER#DU#SERDM/  
L 7 ABREVIATION : BURCOUR  
CENTRE#INFORMATIQUE/  
L 7 ABREVIATION : CENTINF  
CHEF#DE#SERVICE/  
L 8 ABREVIATION : CHEFSERV  
DIVISION#CONTROLE#DES#ENGAGEMENTS#DE#LA#DB/  
L 1 ABREVIATION : DIVCONENG-DB  
DIVISION#CONTROLE#DES#ENGAGEMENTS#DE#LA#DI/  
L 1 ABREVIATION : DIVCONENG-DI  
FOURNISSEUR#OU#BENEFICIAIRE#DE#L'ENGAGEMENT/  
L 1 ABREVIATION : FOURBENEFENG  
GROUPE#DE#PERFORATION/  
L 7 ABREVIATION : GROPERT  
SERVICE#COMPTABLE#CENTRAL/  
L 1 ABREVIATION : SERVCMPCENT  
SERVICE#INFORMATIQUE/  
L 7 ABREVIATION : SERVINF  
TRESOR/  
L 4 ABREVIATION : TRES  
ADOPTION#DE#LA#LOT#DE#REGLEMENT/  
L 1 ABREVIATION : ADOPLOIREGL  
CHCIX#DES#ADMINISTRATEURS#DU#CREDIT/  
L 8 ABREVIATION : CHOAD-CR  
ANNULATION#DE#L'ENGAGEMENT#PAR#L'AC/  
L 5 ABREVIATION : ANEAC  
APPUREMENT#DU#DOSSIER#D'ENGAGEMENT/  
L 6 ABREVIATION : APOCSE  
PASSATION#ET#REGLEMENT#DE#MARCHES/  
L 1 ABREVIATION : PASREGLMARC  
CONTROLES#ORDONNATEUR#DE#TYPE1/  
L 1 ABREVIATION : CONTORD-DIT1  
CONTROLES#TRESOR#DE#TYPE1#/  
L 9 ABREVIATION : CONTREST1  
DELEGATION#DE#CREDIT/

L 6 ABBREVIATION : DEL-CR

PROCEDURE#SIMPLIFIEE/

L 8 ABBREVIATION : PROCSIMP

ENGAGEMENT#DE#DEPENSE/

L 4 ABBREVIATION : EDEP

OPERATIONS#DIVERSES#CONCERNANT#LE#BUDGET/

L 8 ABBREVIATION : OPERBUDG

REGLEMENT#DE#LA#DEPENSE/

L 7 ABBREVIATION : REGLDEP

PROCEDURE#NORMALE/

L 8 ABBREVIATION : PROCNORM

REJET#ABSENCE#DE#CREDIT#OU#CONSUMMATION#TRC#RAPIDE/

L 1 ABBREVIATION : REJABS-CRCONRAF

REJET#DE#LA#DB#DE#LA#PROPOSITION#D'ENGAGEMENT/

L 9 ABBREVIATION : REJ-DBPRE

REJET#DE#LA#DI#DE#LA#PROPOSITION#D'ENGAGEMENT/

L 9 ABBREVIATION : REJ-DIPRE

REJET#PAR#LE#TRESOR#DE#LA#PROPOSITION#D'ENGAGEMENT/

L 8 ABBREVIATION : REJTRPRE

RESERVATION#DE#CREDIT/

L 7 ABBREVIATION : RESCRED

SUITE#A#REJET#ABSENCE#CREDITS#OU#CONSUMMATION#TROP#RAPIDE/

L 1 ABBREVIATION : SUIREJ-CRCONRAF

FIN METHODE 2

? LOGOUT

LOGOUT BASE: ABRSOI



\*\* SCCRATE STARTED \*\* V1.6      LE 133.76      A 22.45.17

1      EXEC TRACE-DIC  
2      ?

ANNEXE A2.3

DICTIONNAIRES

3      SPECIAL3

DICTIONNAIRE : SPECIAL3

BE	-BE
DI	-DI
DB	-DB
TC	-TC
EVENTUEL	...
SCC	-SCC
ORIGINAL	...
FACTURE+TC	FAC+TC
BONS	-B
ENGAGEMENTS	E
ENGAGEMENT	E
PSP	-PSP
BUDGET	EUD
EXEMPLAIRES	...
DCE-	...
VISES	...
DUPLICATA	...
SERDM	...
CREDIT	C
CREDITS	C
DIFFERENTS	...
ORTFNIR	...

DECAUDINGEORGES	DECAUDIN
ADMINISTRATEUR	A
ADMINISTRATEURS	A
TYPE1	T1
DIVERSES	...
CONCERNANT	...
MISE	...
CHAUTARD DENTIS	CHAUTARD
PORTEES	...
NECESSAIRES	...
REFNDUS	...
ADRESSEE	...
COMPLETE	...
COMPLETES	...
CHARGE	...

4

\*\*\*

FIN EDITION

RUN FN:R

8

...

\*\* SCCRATE STARTED \*\* V1.6      LE 133.76      A 22.45.21

1	EXEC CAL-EDIT-LIB	RESULTAT
2	?	
3	0003	METHODE3
4	0086	
5	0172	

METHODE NUMERO 3

...

ETABLISSEMENT#DES#BE/

L 7      ABREVIATION : ETAE-BE

ARCHIVAGE#DU#BE#ORIGINAL#AU#SCC/

L 1      ABREVIATION : ARC-BE-SCC

CODIFICATION#DE#LA#CERTIFICATION/

L 9      ABREVIATION : CODICERTI

CODIFICATION#FACTURE+TC/

L 1      ABREVIATION : CODIFAC+TC

CONTROLES#ORDONNATEUR#SUR#CERTIFICATION/

L 1      ABREVIATION : CONCRDOCER

CONTROLES#ORDONNATEUR#SUR#LES#BONS#D'ENGAGEMENTS/

L 8      ABREVIATION : CONOR-BE

CONTROLE#ORDONNATEUR#SUR#FACTURE+TC/

L 1      ABREVIATION : CONGRFAC+TC

CONTROLE#TRESOR,VISA,PRECOMPTE#EVENTUEL/

L 1      ABREVIATION : CONTREVIPRE

DIFFUSION#DES#BE#ET#DES#PSP#A#LEURS#DESTINATAIRES/

L 1      ABREVIATION : DIF-BE-PSPDE

ADOPCTION#DU#BUDJET/

L 7      ABREVIATION : ADOPBUD

EMISSION#DE#LA#FACTURE#EN#3#EXEMPLAIRES/

L 6      ABREVIATION : EMIFA3

ENVOI#DES#BE#A#LA#DCE-CB/

L 1      ABREVIATION : ENVC-BE-DB

ENVOI#DES#BE#VISES#AU#SCC/

L 1      ABREVIATION : ENVC-BE-SCC

ENVOI#DES#BE#A#LA#DCE-DI/

L 1      ABREVIATION : ENVC-BE-DI

ENVOI#DU#DUPLICATA#DU#BE#AU#SERVICE#INFORMATIQUE/

L 1      ABREVIATION : ENVC-BESERIN

ENVOI#DES#BE#ET#DES#PSP#AU#BUREAU#COURRIER#SEROM/

L 1 ABBREVIATION : ENVC-BE-PSPBUC

ENVOI#DU#BE#ET#DES#PSP#AU#SCC/  
L 1 ABBREVIATION : ENVO-BE-PSP-SCC

ENREGISTREMENT#DE#L'ARRIVEE#DE#BE/  
L 1 ABBREVIATION : ENREARRI-BE

ENVOI#DES#BE#AU#BUREAU#COURRIER#SERDM/  
L 1 ABBREVIATION : ENVC-BEBUCO

ENVOI#CERTIF+FACTURE#A#SERDM/  
L 7 ABBREVIATION : ENVCCER

ENVOI#FACTURE+TC+BE#AU#SERDM/  
L 7 ABBREVIATION : ENVCFAC

ENVOI#FACTURE#AU#SERVICE#EMETTEUR/  
L 1 ABBREVIATION : ENVCFACSEREM

ENVC#AU#CENTRE#INFORMATIQUE/  
L 1 ABBREVIATION : ENVCCENINFO

ENVOI#DU#DOSSIER#AU#TRESOR/  
L 1 ABBREVIATION : ENVOOSTRES

PERFORATION#DU#DUPLICATA#DU#BE/  
L 6 ABBREVIATION : PER-BE

PERFORATION#TC/  
L 6 ABBREVIATION : PER-TC

PROPOSITION#DE#DEPENSE#A#L'ADMINISTRATEUR#DE#CREDIT/  
L 9 ABBREVIATION : PROPDEPAC

REJET#CERTIFICATION/  
L 1 ABBREVIATION : REJETCERTI

REJET#FACTURE+TC/  
L 1 ABBREVIATION : REJETFAC+TC

REJET#DE#LA#PROPOSITION#DE#DEPENSE/  
L 1 ABBREVIATION : REJEPROPDEPE

REJET#TRESOR#DE#LA#FACTURE#OU#DU#TC/  
L 1 ABBREVIATION : REJETREFA-TC

RENVOI#DE#LA#CERTIFICATION#ERRONEE#AU#SERVICE#COURRIER/  
L 1 ABBREVIATION : RENCERERSECC

RENVOI#DU#DOSSIER#ERRONE#AU#SERVICE#COURRIER/  
L 1 ABBREVIATION : RENDOSERSECC

RENVOI#DU#DOSSIER#ERRONE#AU#BENEFICIAIRE/  
L 1 ABBREVIATION : RENDOSERBE

RENVOI#AU#SERDM#DES#DOCUMENTS#INFORMATIQUES/  
L 8 ABBREVIATION : RENCOCIN

VENTILATION#DES#DIFFERENTS#DOCUMENTS/

L 8 ABREVIATION : ELABOBUD

EXECUTION#DU#BUDGET/

L 8 ABREVIATION : EXECUBUD

ADMINISTRATEUR#DE#CREDIT/

L 2 ABREVIATION : AC

BUREAU#DU#COURRIER#DU#SEROM/

L 8 ABREVIATION : BURECOUR

CENTRE#INFORMATIQUE/

L 1 ABREVIATION : CENTRINFOR

CHEF#DE#SERVICE/

L 9 ABREVIATION : CHEFSERVI

DIVISION#CONTROLE#DES#ENGAGEMENTS#DE#LA#DB/

L 1 ABREVIATION : DIVICONEN-DB

DIVISION#CONTROLE#DES#ENGAGEMENTS#DE#LA#DI/

L 1 ABREVIATION : DIVICONEN-DI

FOURNISSEUR#OU#BENEFICIAIRE#DE#L'ENGAGEMENT/

L 9 ABREVIATION : FOURBENEE

GRUPE#DE#PERFORATION/

L 1 ABREVIATION : GROUPPERFO

SERVICE#COMPTABLE#CENTRAL/

L 9 ABREVIATION : SERCOMCEN

SERVICE#INFORMATIQUE/

L 1 ABREVIATION : SERVIINFOR

TRESCR/

L 6 ABREVIATION : TRESOR

ADOPTION#DE#LA#LOI#DE#REGLEMENT/

L 1 ABREVIATION : ADOPLOIREG

CHOIX#DES#ADMINISTRATEURS#DU#CREDIT/

L 6 ABREVIATION : CHOIAC

ANNULATION#DE#L'ENGAGEMENT#PAR#L'AC/

L 7 ABREVIATION : ANNLEAC

APPUREMENT#DU#DOSSIER#D'ENGAGEMENT/

L 8 ABREVIATION : APPUDOSE

PASSATION#ET#REGLEMENT#DE#MARCHES/

L 9 ABREVIATION : PASREGMAR

CONTROLES#ORDONNATEUR#DE#TYPE1/

L 9 ABREVIATION : CONCRDOT1

CONTROLES#TRESOR#DE#TYPE1#/

L 9 ABREVIATION : CONTREST1

DELEGATION#DE#CREDIT/

L 7 ABREVIATION : VENOCU  
ENVOI#DU#DOSSIER#A#DCE-DI/  
L 1 ABREVIATION : ENVODOS-DI  
ENVOI#DU#DOSSIER#A#DCE-DB/  
L 1 ABREVIATION : ENVODOS-DB  
VISA#DES#BE#PAR#LA#DCE-DB/  
L 1 ABREVIATION : VISA-BE-DB  
VISA#DU#BE#PAR#LA#DCE-DI/  
L 1 ABREVIATION : VISA-BE-DI  
VISA#DES#BE#PAR#LE#TRESOR/  
L 1 ABREVIATION : VISA-BETRES  
CONTROLE#EXISTENCE#ET#RYTHME#DE#CONSUMMATION#DES#CREDITS/  
L 1 ABREVIATION : CONEXRYCOC  
CONTROLES#AUTOMATIQUES#SUR#LA#FORME/  
L 9 ABREVIATION : CONAUTFOR  
ENREGISTREMENT#TC#DANS#FICHIER#ENGAGEMENT/  
L 1 ABREVIATION : ENRE-TCFICE  
ETABLISSEMENT#DES#TITRES#DE#REGLEMENT/  
L 1 ABREVIATION : ETABTITREG  
CONTROLE#CERTIFICATION/  
L 9 ABREVIATION : CONTCERTI  
CONTROLE#TITRE#DE#CREANCE/  
L 1 ABREVIATION : CONTITCREA  
REJET#TITRE#DE#CREANCE/  
L 1 ABREVIATION : REJETITCREA  
RELANCE#ENVUE#D'OBTENIR#LA#CERTIFICATION/  
L 7 ABREVIATION : RELACER  
TRAITEMENT#DU#BE/  
L 8 ABREVIATION : TRAIT-BE  
LISTE#DES#ANOMALIES/  
L 1 ABREVIATION : LISTEANOMA  
DECAUDINGEORGES/  
L 8 ABREVIATION : DECAUDIN  
DEPENSES#DE#MATERIEL/  
L 1 ABREVIATION : DEPENMATER  
RELANCE#CERTIFICATION/  
L 1 ABREVIATION : RELANCERTI  
TITRE#DE#CREANCES/  
L 1 ABREVIATION : TITRECREAN  
ELABORATION#DU#BUDGET/

L 6 ABBREVIATION : DELEGC

PROCEDURE#SIMPLIFIEE/

L 9 ABBREVIATION : PROCESIMP

ENGAGEMENT#DE#DEPENSE/

L 6 ABBREVIATION : EDEPEN

OPERATIONS#DIVERSES#CONCERNANT#LE#BUDGET/

L 7 ABBREVIATION : OPERBUD

REGLEMENT#DE#LA#DEPENSE/

L 1 ABBREVIATION : REGLEDEPEN

PROCEDURE#NORMALE/

L 1 ABBREVIATION : PROCENORMA

REJET#ABSENCE#DE#CREDIT#OU#CONSUMMATION#TRCP#RAPIDE/

L 1 ABBREVIATION : REJABCCORA

REJET#DE#LA#DB#DE#LA#PROPOSITION#D'ENGAGEMENT/

L 9 ABBREVIATION : REJ-DBPRE

REJET#DE#LA#DI#DE#LA#PROPOSITION#D'ENGAGEMENT/

L 9 ABBREVIATION : REJ-DIPRE

REJET#PAR#LE#TRESOR#DE#LA#PROPOSITION#D'ENGAGEMENT/

L 9 ABBREVIATION : REJTREPRE

RESERVATION#DE#CREDIT/

L 6 ABBREVIATION : RESERC

SUITE#A#REJET#ABSENCE#CREDITS#OU#CONSUMMATION#TROP#RAPIDE/

L 1 ABBREVIATION : SUIREJABCCORA

FIN METHCDE 3

7 LOGOUT

LOGOUT BASE: ABRSOC1

**ANNEXE A3**

**ENSEMBLE DES LIBELLES. PROJET (GESTION DES CONGRES)**



\*\*\* CONGRES \*\*\*

PAGE ****	CODE ****	LIBELLE *****
207	ACCOMPA	* NOM DES ACCOMPAGNANTS *
7	ACCUEIL	* ACCUEIL, INSCRIPTIONS, HEBERGEMENT *
13	ADMINIST	* PREPARATIFS ADMINISTRATIFS-SECURITE *
208	ADREHOTEL	* ADRESSE DE L'HOTEL OU DE LA RESIDENCE *
209	ADRESSI	* ADRESSE PERSONNELLE OU DE L'ENTREPRISE DU CONGRESSISTE
210	ADRHOTEL	* ADRESSE D'HEBERGEMENT *
155	AGETOUR	* GROUPEMENT DES AGENCES DE TOURISMES ET HOTELIERS *
105	ANALERREURS	* ANALYSE DES ERREURS *
106	ANALRESUL	* ANALYSE DES RESULTATS PARTIELS *
14	ANIMATEUR	* OPIONS CONCERNANT L'ANIMATION ET LE TOURISME *
211	ASSOC	* ASSOCIATION *
155	ATELIER	* ATELIER DE PERFORATION *
107	ATTENTRAIT	* MISE EN ATTENTE DES CARTES ENVOIE DU TRAITEMENT *
108	ATTRIBUTION	* RESERVATION DES CHAMBRES *
175	AUTFACTOR	* FACTURATION AUTOMATIQUE DES CONGRESSISTES *
15	AUTOMATRAIT	* TRAITEMENT AUTOMATIQUE DES INSCRIPTIONS *
157	AUT84	* GROUPE FONCTIONNEL AUTOMATIQUE *
212	BANQUEDFBI	* NOM DE LA BANQUE DU DEBITEUR *
20	BILAN	* BILAN D'UN CONGRES *
24	BILANFI	* ARRETE DU BILAN *
25	BILANS	* BILANS DES CONGRES REALISES DURANT L'ANNEE *
26	BUDGET	* BUDGET PREVISIONNEL *
193	CHAMBRES	* ENSEMBLE DES CHAMBRES DISPONIBLES *
27	CHARGES	* ELABORATION DU CAHIER DES CHARGES *
171	CHAUTARD	* CHAUTARA DENIS *
102	CODHEBER	* CODIFICATION DES DONNEES HEBERGEMENT SUR BORDEREAU *
110	CODIENS	* CODIFICATION DE LA MOTIF *

PAGE	284	01/09/75	** CONGRES **	P LISTE ALPHABETIQUE DES CODES :
111	CODIFVERS		* CODIFICATION DES VERSEMENTS *	
112	CODINSCRI		* CODIFICATION DES INFORMATIONS SUR BORDEREAU *	
153	COMITE		* COMITE D'ORGANISATION *	
28	CONFERENCE		* CHOIX DES CONFERENCES ET DES CONFERENCIERS *	
29	CONFIRM		* CONFIRMATION DES RESERVATIONS DE SALLES ET D'HEBERGEMENT	
113	CONFIRMATION		* LETTRE DE CONFIRMATION ADRESSEE AU CONGRESSISTE *	
213	CONGNAM		* NOM DU CONGRESSISTE QUI A RESERVE *	
30	CONGRES		* GESTION DES CONGRES *	
199	CONGRESSI		* ENSEMBLE DES CONGRESSISTES *	
34	CONTACT		* EBAUCHE DES DESIDERATA DU CLIENT *	
35	CONTENTIEUX		* REGLEMENT DES IMPAYES, SUITES JUDICIAIRES *	
114	CONTRAT		* SIGNATURE DU CONTRAT *	
115	CONTROL-HE		* CONTROL DES ARRHEES ET DES BULLETINS D'HEBERGEMENT *	
116	CONTROL-IN		* CONTROL FINANCIER DES INSCRIPTIONS *	
117	CONTROL-HE		* CONTROL DES DONNEES DU BULLETIN D'HEBERGEMENT *	
118	CONTROLI		* CONTROL DES INFORMATIONS PORTEES SUR LES BULLETINS *	
119	CONTOVERS		* CONTROL DES VERSEMENTS *	
120	COURRIER		* REMISE DU COURRIER AU CONGRESSISTE *	
214	DATARRIV		* DATE D'ARRIVEE *	
215	DATE-ARR		* DATE D'ARRIVEE *	
216	DATE-DEP		* DATE DE DEPART *	
217	DATEPATE		* DATE DU PAIEMENT *	
218	DATINSCRI		* DATE DE L'INSCRIPTION *	
219	DATVERS		* DATE DU VERSEMENT *	
175	DECOMPTES		* EDITION DES DECOMPTES PARKEUNITES , EXCURSIONS *	
35	DEFINITION		* DEFINITION DU CONGRES, DES SERVICES DEMANDES *	
41	DEMANDES		* RECEPTION ET TRAITEMENT DES DEMANDES *	

PAGE	DES	01/1975	** CONGRES **	P LISTE ALPHABETIQUE DES CODES :
150	DIRECTION		* GROUPE DE DIRECTION *	
121	DOC.BOU		* ENVOI DE DOCUMENTATION NOUVEAUX INSCRITS *	
12	DOSSIERS		* REALISATION DES DOSSIERS D'INSCRIPTIONS *	
220	DURESERV		* DUREE DE LA RESERVATION *	
177	HOTEL		* EDITION DES LISTES PAR HOTEL *	
122	ENCAISPAIE		* ENCAISSEMENT DES PAIEMENTS *	
13	ENCAISSEMENT		* ENCAISSEMENT DES PAIEMENTS DES SOLDES *	
123	ENVOI		* ENVOI DES DOSSIERS D'INSCRIPTION *	
178	EPAYS		* EDITION DES LISTES PAR PAYS *	
16	EQUIPEMENT		* MATERIEL-EQUIPEMENT NECESSAIRES *	
221	ETATCHAMBRE		* ETAT DE LA CHAMBRE - RESERVEE, NON-RESERVEE *	
222	ETATFINAN		* ETAT FINANCIER DU CONGRESSISTE *	
179	ETYPHE		* EDITION PAR TYPE D'HEBERGEMENT *	
124	EXPEDI		* EXPEDITION DES FACTURES *	
17	FACTURASSOC		* FACTURATION DES SERVICES RENDUS A L'ASSOCIATION *	
18	FACTURATION		* FACTURATION DES PARTICIPANTS *	
53	FACTURE-SOLD		* FACTURATION DES SOLDES DES COMPTES CONGRESSISTES *	
126	FACTUREF1		* FACTURE ADRESSEE AU CONGRESSISTE *	
185	FACTUREP2		* FACTURE ADRESSEE A L'ENTREPRISE *	
160	GRUPOCD		* GROUPE DE CODIFICATION *	
125	HCOMPLTE		* ACTIVER UNE RESERVATION EN ATTENTE *	
161	HEBERGE		* GROUPE RESPONSABLE DE L'HEBERGEMENT *	
128	HEBERGEMENT		* AFFECTATION DE CHAMBRES AU CONGRESSISTE *	
127	LITIGE		* REGLEMENT DES LITIGES CONCERNANT L'HEBERGEMENT *	
162	HOTESSES		* GROUPE DES HOTESSES A L'ACCUEIL *	
183	HRESLITIGE		* RESPONSABLE DES LITIGES HEBERGEMENT *	
129	HCOMPLTE		* COMPLETER DOSSIER D'INSCRIPTION EN ATTENTE *	

35	205	01/09/75	** CONGRES **	P LISTE ALPHABETIQUE DES CODES :
20		LITIGE	*	REGLEMENT DES PROBLEMES DE LITIGE D'INSCRIPTION *
64		INFORMAT	*	SERVICE INFORMATIQUE *
23		INSCRIBM	*	NOM DU CONGRESSISTE INSCRIT *
4		INSCRIBUV	*	CONTROLE ET RELEVÉ INFORMATIONS INSCRIPTIONS NOUVELLES
30		INSCRIPTION	*	INSCRIPTION DU CONGRESSISTE *
8		INSCRIPTIONS	*	INSCRIPTIONS-HEBERGEMENT-FACTURATION *
02		INSCRIBF	*	INSCRIPTION D'UN CONGRESSISTE *
31		INTERREG	*	RECEPTION DE LA REQUETE D'UN CONGRESSISTE *
65		TRESLITIGE	*	RESPONSABLE DES LITIGES INSCRIPTION *
32		LETTREFUS	*	LETRE DE REFUS *
80		LISTASSOC	*	EDITION DE LA LISTE DES PARTICIPANTS PAR ASSOCIATION *
86		LISTERR	*	LISTE DES ERREURS *
87		LISTE1	*	LISTE PAR ASSOCIATION *
84		LISTE2	*	LISTE COMPLETE *
89		LISTE3	*	LISTE DES DECOMPTES *
90		LISTE4	*	LISTE PAR PAYS *
91		LISTE5	*	LISTE PAR TYPE D'HEBERGEMENT *
92		LISTE6	*	LISTE PAR HOTELS OU RESIDENCES *
93		LISTINS	*	LISTE DES NOUVELLES INSCRIPTIONS *
94		LISTMOD	*	LISTE DES MODIFICATIONS *
81		LISTOUT	*	EDITION DE LA LISTE COMPLETE DES INSCRITS *
5		LOCAUX	*	OPTIONS CONCERNANT LES SALLES DE REUNION, L'HEBERGEMENT
6		MAINTENANCE	*	MAINTENANCE, MODIFICATIONS DES SERVICES DE L'ORGANISAT
7		MODIF-HEBERG	*	TRAITEMENT DES MODIFICATIONS HEBERGEMENT *
8		MODIF-INSCRI	*	TRAITEMENT DES MODIFICATIONS D'INSCRIPTIONS *
9		MODIF-ORGA	*	MODIFICATIONS DE L'ORGANISATION *
33		MODIF-RESERV	*	MODIFICATIONS DE RESERVATIONS *

35 205 01/09/75 \*\* CONGRES \*\* P LISTE ALPHABETIQUE DES CODES :

PAGE	247	01/09/75	** CONGRES **	P LISTE ALPHABETIQUE DES CODES :
224	MONTARRHES		* MONTANT DES ARRHER VERSEES *	
225	MONTINSCRIT		* MONTANT DES DROITS D'INSCRIPTION *	
226	MONTPAIE		* MONTANT DU PAIEMENT *	
227	MOPYPORT		* MOYEN DE TRANSPORT *	
228	NATUFTN		* NATURE DES ACTIVITES FINANCEES *	
229	NBMI		* NOM DU CONGRESSISTE *	
230	NUMARRHES		* NUMERO D'ORDRE DU VERSEMENT *	
231	NUMCHAMBRE		* NUMERO D'ORDRE DE LA CHAMBRE *	
232	NUMCHEQUE		* NUMERO DU CHEQUE *	
233	NUMCOMPTE		* NUMERO DU COMPTE *	
234	NUMERISTE		* NUMERO D'ORDRE DU CONGRESSISTE *	
235	NUMINSCRI		* NUMERO D'ORDRE DE L'INSCRIPTION *	
236	NUMPAIE		* NUMERO D'ORDRE DU PAIEMENT *	
237	NUMREFI		* NUMERO DE LA CHAMBRE DANS L'HOTEL *	
238	NUMRESF		* NUMERO D'ORDRE DE LA RESERVATION *	
239	NUMVERS		* NUMERO D'ORDRE DU VERSEMENT *	
200	PAIEMENTS		* ENSEMBLE DES OPERATIONS COMPTABLES *	
75	PCONTRAT		* PROJET DE CONTRAT *	
134	PERCEPDRBIT		* PERCEPTION DES DROITS D'INSCRIPTION *	
135	PERF0		* PERF0 VERIF DES CARTES *	
76	PLANIFT		* PLANIFICATION DE LA REALISATION DES DIFFERENTS CONGRES	
136	POINTAGE		* VERIFICATION DE L'INSCRIPTION ET POINTAGE DU CONGRESSISTE	
77	POST-CONGRES		* PLANIFICATION DES CHARGES POST-CONGRES *	
78	POST-SECRET		* COMPTES RENDUS-SECRETARIAT POST-CONGRES *	
240	PPAYS		* PAYS D'ORIGINE *	
241	PRENOM		* PRENOM DU CONGRESSISTE *	
242	PRESENCE		* INDICATEUR DE PRESENCE AU CONGRES *	

252 253 01/09/75 \*\* CONGRES \*\* P LISTE ALPHABETIQUE DES CODES :

79 PREVISION \* PREVISION-PLANNING DES CONGRES \*

243 PRIEUR \* PRIX JOURNALIER \*

80 PRISE-CHARG \* PREPARATIFS DU CONGRES \*

137 PRISE-COMPT \* PRISE EN COMPTE DE LA MODIFICATION \*

244 PROSCIEF \* PROGRAMME SCIENTIFIQUE \*

245 PROTOURIS \* PROGRAMME TOURISTIQUE \*

84 PROMOTION \* PROMOTION-PUBLICITE \*

85 PROPRE \* PROJET DE PROGRAMME \*

86 PUBLICITE \* PUBLICITE INFORMATION \*

87 REALISATIONS \* REALISATION SIMULTANEE DE PLUSIEURS CONGRES \*

138 RECEPTION \* RECEPTION, TRI ET REPARTITION DU COURRIER \*

139 RECEPERS \* RECEPTION , MISE EN ATTENTE DES VERSEMENTS \*

90 RECRUTE \* RECRUTEMENT DU PERSONNEL ACCUEIL ET CONFERENCES \*

140 REFUSINSCRI \* REFUS DE L'INSCRIPTION \*

141 REFUSMODIF \* REFUS DE LA MODIFICATION \*

142 REJERREUR \* REFUS DES VERSEMENTS ERRONES \*

143 RELAN-INSCRI \* RELANCE DES INSCRIPTIONS \*

144 RELANCE-H \* RELANCE POUR DOSSIER INCOMPLET \*

145 RELANCE-I \* RELANCE POUR ERREUR DU DOSSIER INSCRIPTION \*

146 REMISE \* REMISE DE LA DOCUMENTATION PARTICIPANTS ET DU BADGE \*

91 RENCONTRE \* RENCONTRE DU COMITE D'ORGANISATION ET PALAIS CONGRES

147 RENSEIGNE \* DEMANDE DE RENSEIGNEMENTS \*

148 RENSEIGNEMEN \* REPONDRE A UNE DEMANDE DE RENSEIGNEMENT \*

149 RENVOI \* RENVOI DES DOSSIERS D'INSCRIPTION \*

150 REPONSE \* REPONSES INDIVIDUALISEES AU COURRIER \*

211 RESERVATION \* RESERVATION DE CHAMBRES POUR LE CONGRESSISTE \*

155 RESPONFI \* RESPONSABLE FINANCIER \*

254 255 01/09/75 \*\* CONGRES \*\* P LISTE ALPHABETIQUE DES CODES :

PAGE	259	01/09/75	** CONGRES **	P LISTE ALPHABETIQUE DES CODES :
92	RESTAU		* OPTIONS CONCERNANT LA RESTAURATION LE TRANSPORT *	
151	REUNION		* REUNION DU COMITE D'ORGANISATION *	
167	SECONGRES		* GROUPE DES SECRETAIRES CHARGE DU SUIVI DU CONGRES *	
93	SECRETARIAT		* SECRETARIAT DU CONGRES *	
168	SECRINSCRI		* SECRETARIAT AUX INSCRIPTIONS *	
169	SECTEUR		* SECRETARIAT DES AGENCES DE TOURISME *	
94	SERVICES		* MISE EN PLACE DES DIVERS SERVICES DU CONGRES *	
95	SUIVI		* SUIVI DU DEROULEMENT *	
98	SUIVI-ARGA		* SUIVI D'ORGANISATION *	
182	TRAITINS		* TRAITEMENT DES INSCRIPTIONS ET MODIFICATIONS *	
152	TRANSMIS		* TRANSMISSION DES MODIFICATIONS *	
246	TYPPIE		* TYPE DE PAIEMENT - CCP , CHEQUE BANCAIRE , LIQUIDE *	
247	TYPCHAMBRE		* TYPE DE LA CHAMBRE *	
248	TYPEC		* TYPE DE CONGRESSISTE *	
249	TYPHERBERG		* TYPE D'HEBERGEMENT *	
250	TYPHOTEL		* TYPE D'HOTEL OU DE RESIDENCE *	
251	TYPCCUP		* TYPE D'OCCUPATION - PENSION - DEMI PENSION - NUIT *	
101	UNCONGRES		* REALISATION D'UN CONGRES *	
153	VERSARRHES		* VERSEMENT DES ARRHES *	
204	VERSEMENT		* VERSEMENT D'UN PARTICIPANT *	
205	VSARRHES		* VERSEMENT DES ARRHES *	

JEU pour Méthode 5

```
*****  
*  
*  
*  
*  
*  
* R DICTIONNAIRE DU PROJET *  
*  
*  
*  
*  
*  
*****
```

Les mots barrés sont ceux qui devront figurer dans ANTIDICRED pour l'application de la METHODE5 qui n'a pas été calculée.



COMPAGNA \* NOM DES ACCOMPAGNANTS \*  
U

ACCUEIL \* ACCUEIL, INSCRIPTIONS, HEBERGEMENT \*  
\* SPECIFICATION DES TACHES DU SERVICE ACCUEIL-RENSEIGNEMENTS  
PENDANT LA DUREE DU CONGRES \* ( 246)

ADMINISTRATIF \* PREPARATIFS ADMINISTRATIFS-SECURITE \*  
\* LES PREPARATIFS DU CONGRES COMPORTENT DES DEMARCHES  
ADMINISTRATIVES AUPRES DES AUTORITES LOCALES ( MAIRIE,  
PREFECTURE, POLICE ), DES SOCIETES DE TRANSPORT ( SNCF,  
COMPAGNIES D'AVIATION ) - LES COMPORTENT AUSSI LA MISE EN  
PLACE DE DISPOSITIFS DE SECURITE ( INCENDIE, CROIX ROUGE .. )  
\* ( 404)

ADRESSE HOTEL \* ADRESSE DE L'HOTEL OU DE LA RESIDENCE \*  
U

ADRESSE \* ADRESSE PERSONNELLE DU CONGRESSISTE \*  
U

ADRESSE HOTEL \* ADRESSE D'HEBERGEMENT \*  
U

AGENT \* GROUPEMENT DES AGENCES DE TOURISMES ET HOTELIERS \*  
\* GROUPEMENT D'INTERET ECONOMIQUE EN MATIERE D'HEBERGEMENT  
HOTEL \* IL COMPTE DES ADMINISTRATEURS DELEGUES PAR LES  
AGENCES DE TOURISME ET LES HOTELIERS - IL PLANIFIE LES  
RESERVATIONS HOTEL ET EFFECTUE LE SUIVI DE LA RESERVATION  
DE CHAQUE CONGRESSISTE \* ( 384)

ANALYSE ERREURS \* ANALYSE DES ERREURS \*  
\* ANALYSE DES RESULTATS DES TRAITEMENTS CONCERNANT LES  
INSCRIPTIONS ET MODIFICATIONS. RELEVÉ DES ERREURS \* ( 343)

ANALYSE \* ANALYSE DES RESULTATS PARTIELS \*  
\* ANALYSE DES LISTES DE NOUVEAUX INSCRITS, DES EDITIONS  
COMPLETES OU PARTIELLES \* ( 350)

284 01/09/75 \*\* CONGRES \*\* R DICTIONNAIRE DU PROJET :

- ATEUR \* OPTIONS CONCERNANT L'ANIMATION ET LE TOURISME \*  
\* ~~EN~~ FONCTION DU PROJET DE PROGRAMME TOURISTIQUE , PRISE DE CONTACT AVEC ~~DES~~ AGENCES DE TOURISME POUR PREVOIR AVEC ELLES LES DIFFERENTES EXCURSIONS - DE MEME AVEC DIVERS ASSOCIATIONS ~~OU~~ ORGANISMES CULTURELS POUR PREVOIR LE PROGRAMME ~~DES~~ SPECTACLES \* ( 405)
- IC \* ASSOCIATION \*  
U
- IER \* ATELIER DE PERFORATION \*  
\* ATELIER DE PERFORATION - GROUPE EXTERIEUR A L'ORGANISATION EN PALAIS DES CONGRES QUI SOUS TRAITA LA SAISIE ~~DES~~ DONNEES \* ( 394)
- INTRAIT \* MISE ~~EN~~ ATTENTE ~~DES~~ CARTES ENVUE ~~DU~~ TRAITEMENT \*  
\* MISE ~~EN~~ ATTENTE ~~DES~~ CARTES AURETOUR ~~DE~~ LA PERFORATION \* ( 341)
- IBUTION \* RESERVATION ~~DES~~ CHAMBRES \*  
\* CETTE ACTION CONSISTE A TRANSMETTRE A UN HOTELIER LE NOM DU CONGRESSISTE , LA DUREE ET LES CARACTERISTIQUES DE SA RESERVATION , ET LE MONTANT ~~DES~~ ARRHEES VERSEES \* ( 377)
- FACTOR \* FACTURATION AUTOMATIQUE ~~DES~~ CONGRESSISTES \*  
\* TRAITEMENT AUTOMATIQUE ~~DE~~ LA FACTURATION ~~EN~~ UTILISANT LES ENSEMBLES PAIEMENTS , CONGRESSISTES ET LES CARACTERISTIQUES TARIFS APPLIQUES PAR ACTIVITE ET PAR TYPE DE CONGRESSISTE , ETAT FINANCIER ~~DU~~ PARTICIPANT ET ACTIVITES AUXQUELLES ~~IL~~ PARTICIPE \* ( 319)
- MATRAIT \* TRAITEMENT AUTOMATIQUE ~~DES~~ INSCRIPTIONS \*  
\* MODULE ~~DE~~ TRAITEMENT AUTOMATIQUE ~~DES~~ INSCRIPTIONS , ~~DES~~ MODIFICATIONS , ~~DES~~ DIFFERENTES EDITIONS A LA DEMANDE ~~DU~~ COMITE ~~D'~~ORGANISATION \* ( 296)
- 4 \* GROUPE FONCTIONNEL AUTOMATIQUE \*  
U
- UFDEBI \* NOM ~~DE~~ LA BANQUE ~~DU~~ DEBITEUR \*  
J

284 01/09/75 \*\* CONGRES \*\* R DICTIONNAIRE DU PROJET :

284 01/09/75 \*\* CONGRES \*\* P LISTE ALPHABETIQUE DES MOTS \*

\*\* SECSTATE STARTED \*\* V1.6

LE 133.76

A 22.47.02

1 EXEC TRACE-DIC  
2 ?

ANNEXE A3.2

DICTIONNAIRE

3 SPECIAL3

DICTIONNAIRE : SPECIAL3

BE	-BE
DI	-DI
CB	-DB
TC	-TC
EVENTUEL	...
SCC	-SCC
ORIGINAL	...
FACTURE+TC	FAC+TC
BONS	-B
ENGAGEMENTS	E
ENGAGEMENT	E
PSP	-PSP
BUDGET	BUD
EXEMPLAIRES	...
DCE-	...
VISES	...
DUPLICATA	...
SFRDM	...
CREDIT	C
CREDITS	C
DIFFERENTS	...
CBTENIR	...

DECAUDINGEORGES	GECAUDIN
ADMINISTRATEUR	A
ADMINISTRATEURS	A
TYPE1	T1
DIVERSES	...
CONCERNANT	...
MISE	...
CHAUTARDDENIS	CHAUTARD
PORTEES	...
NECESSAIRES	...
RENDUS	...
ADRESSEE	...
COMPLETE	...
COMPLETFS	...
CHARGE	...

4

\*\*\*

FIN EDITION  
RUN FN:R

8

\*\* SCCRATE STARTED \*\* V1.6 LE 133.76 A 22.47.07

1	EXEC CAL-EDIT-LIB	RESULTAT
2	?	
3	0003	METHODE3
4	0173	
5	0353	

METHODE NUMERO 3

NOM#DES#ACCOMPAGNANTS/

L 8 ABREVIATION : NOMACCOM

ACCUEIL, INSCRIPTIONS, HEBERGEMENT/

L 1 ABREVIATION : ACCUINSHEBE

PREPARATIFS#ADMINISTRATIFS-SECURITE/

L 1 ABREVIATION : PREPADMISECU

ADRESSE#DE#L'HOTEL#OU#DE#LA#RESIDENCE/

L 1 ABREVIATION : ADREHOTERESI

ADRESSE#PERSONNELLE#CU#DE#L'ENTREPRISE#DU#CONGRESSISTE/

L 1 ABREVIATION : ADREPERENCO

ADRESSE#D'HEBERGEMENT/

L 1 ABREVIATION : ADRESHEBER

GROUPEMENT#DES#AGENCES#DE#TOURISMES#ET#HOTELIERS/

L 1 ABREVIATION : GROLAGTCHC

ANALYSE#DES#ERREURS/

L 1 ABREVIATION : ANALYERREU

ANALYSE#DES#RESULTATS#PARTIELS/

L 1 ABREVIATION : ANALRESUPAR

OPTICONS#CONCERNANT#L'ANIMATION#ET#LE#TOURISME/

L 1 ABREVIATION : OPTIANITOU

ASSOCIATION/

L 8 ABREVIATION : ASSCCIAT

ATELIER#CF#PERFORATION/

L 1 ABREVIATION : ATELIPERFC

MISE#EN#ATTENTE#DES#CARTES#EN#VUE#DU#TRAITEMENT/

L 7 ABREVIATION : ATCATRA

RESERVATION#DES#CHAMBRES/

L 9 ABREVIATION : RESERCHAM

FACTURATION#AUTOMATIQUE#DES#CONGRESSISTES/

L 1 ABREVIATION : FACAUTOCON

TRAITEMENT#AUTOMATIQUE#DES#INSCRIPTIONS/

L 1 ABREVIATION : TRIAUTCINS  
GRUPE#FONCTIONNEL#AUTOMATIQUE/  
L 1 ABREVIATION : GROUFONAUTO  
NOM#DE#LA#BANQUE#DU#DEBITEUR/  
L 1 ABREVIATION : NOMBANDEBI  
BILAN#D'UN#CONGRES/  
L 9 ABREVIATION : BILANCONG  
ARRETE#DU#BILAN/  
L 1 ABREVIATION : ARRETBILAN  
BILANS#DES#CONGRES#REALISES#DANS#L'ANNEE/  
L 1 ABREVIATION : BILACOREAN  
BUDGET#PREVISIONNEL/  
L 8 ABREVIATION : BUDPREVI  
ENSEMBLE#DE#CHAMBRES#DISPONIBLES/  
L 1 ABREVIATION : ENSECHADIS  
ELABORATION#DU#CAHIER#DES#CHARGES/  
L 1 ABREVIATION : ELAECACHICAR  
CHAUTARD#DENIS/  
L 8 ABREVIATION : CHAUTARC  
CODIFICATION#DES#DONNEES#HEBERGEMENT#SUR#BORDEREAU/  
L 9 ABREVIATION : CODCOHEBO  
CODIFICATION#DE#LA#MODIF/  
L 1 ABREVIATION : CODIFMODIF  
CODIFICATION#DES#VERSEMENTS/  
L 1 ABREVIATION : CODIFVERSE  
CODIFICATION#DES#INFORMATIONS#SUR#BORDEREAU/  
L 1 ABREVIATION : CODIINFOBOR  
COMITE#D'ORGANISATION/  
L 1 ABREVIATION : COMITORGAN  
CHOIX#DES#CONFERENCES#ET#DES#CONFERENCIERS/  
L 1 ABREVIATION : CHOICONCON  
CONFIRMATION#DES#RESERVATIONS#DE#SALLES#ET#C'HEBERGEMENT/  
L 9 ABREVIATION : CONRESAHE  
LETTRE#DE#CONFIRMATION#ADRESSE#AU#CONGRESSISTE/  
L 1 ABREVIATION : LETCONADCO  
NOM#DU#CONGRESSISTE#CUI#A#RESERVE/  
L 1 ABREVIATION : NOMCONRESE  
GESTION#DES#CONGRES/  
L 9 ABREVIATION : GESTICONG  
ENSEMBLE#DES#CONGRESSISTES/

L 9 ABREVIATION : ENSEMCONG  
EBAUCHE#DES#DESIDERATA#DU#CLIENT/  
L 1 ABREVIATION : EBAUDESICLIE  
REGLEMENT#DES#IMPAYES, SUITES#JUDICIAIRES/  
L 9 ABREVIATION : REGIMSUJU  
SIGNATURE#DU#CONTRAT/  
L 9 ABREVIATION : SIGNACNT  
CONTROLE#DES#ARRHES#ET#DES#BULLETINS#D'HEBERGEMENT/  
L 9 ABREVIATION : CONARBUHE  
CONTROLE#FINANCIER#DES#INSCRIPTIONS/  
L 1 ABREVIATION : CONFINAINS  
CONTROLE#DES#DONNEES#DU#BULLETIN#D'HEBERGEMENT/  
L 9 ABREVIATION : CONCOBUHE  
CONTROLE#DES#INFORMATIONS#PORTEES#SUR#LES#BULLETINS/  
L 7 ABREVIATION : CONINBU  
CONTROLE#DES#VERSEMENTS/  
L 9 ABREVIATION : CONTVERSE  
REMISE#DU#COURRIER#AU#CONGRESSISTE/  
L 1 ABREVIATION : REMICCURCCN  
DATE#D'ARRIVEE/  
L 9 ABREVIATION : DATEARRIV  
DATE#CE#DEPART/  
L 9 ABREVIATION : DATEDEPAR  
DATE#DU#PAIEMENT/  
L 9 ABREVIATION : DATEPAIEM  
DATE#CE#L'INSCRIPTION/  
L 8 ABREVIATION : DATEINSC  
DATE#DU#VERSEMENT/  
L 9 ABREVIATION : DATEVERSE  
EDITION#DES#DECOMPTES#PAR#REUNIONS, EXCURSIONS/  
L 1 ABREVIATION : EDITDEREEX  
DEFINITION#DU#CONGRES, DES#SERVICES#DEMANDES/  
L 1 ABREVIATION : DEFICSEDE  
RECEPTION#ET#TRAITEMENT#DES#DEMANDES/  
L 1 ABREVIATION : RECETRAIDEMA  
GROUPE#DE#DIRECTION/  
L 1 ABREVIATION : GROLPDIREC  
ENVOI#DE#DOCUMENTATION#NOUVEAUX#INSCRITS/  
L 1 ABREVIATION : ENVCNONCIN  
REALISATION#DES#DOSSIERS#D'INSCRIPTIONS/

L 1 ABBREVIATION : REALDOSINS  
DUREE#DE#LA#RESERVATION/  
L 1 ABBREVIATION : DUREERESER  
EDITION#DES#LISTES#PAR#HOTEL/  
L 1 ABBREVIATION : EDITLISHOTE  
L 0 ABBREVIATION :  
ENCAISSEMENT#DES#PAIEMENTS/  
L 1 ABBREVIATION : ENCAIPAIE  
ENCAISSEMENT#DES#PAIEMENTS#DES#SOLDES/  
L 1 ABBREVIATION : ENCPAIEMSOLD  
ENVOI#DES#DOSSIERS#D'INSCRIPTION/  
L 1 ABBREVIATION : ENVCDOSINS  
EDITION#DES#LISTES#PAR#PAYS/  
L 1 ABBREVIATION : EDITLISPAYS  
MATERIEL-EQUIPEMENT#NECESSAIRES/  
L 8 ABBREVIATION : MATEEQUI  
ETAT#DE#LA#CHAMBRE-RESERVEE, NON#RESERVEE/  
L 1 ABBREVIATION : ETACHRENCRE  
ETAT#FINANCIER#DU#CONGRESSISTE/  
L 1 ABBREVIATION : ETATFINACON  
EDITION#PAR#TYPE#D'HEBERGEMENT/  
L 1 ABBREVIATION : EDITTYPEHEBE  
EXPEDITION#DES#FACTURES/  
L 1 ABBREVIATION : EXPEDFACTU  
FACTURATION#DES#SERVICES#RENDUS##L'ASSOCIATION/  
L 9 ABBREVIATION : FACSERASS  
FACTURATION#DES#PARTICIPANTS/  
L 1 ABBREVIATION : FACTUPARTI  
FACTURATION#DES#SOLDES#DES#COMPTES#CONGRESSISTES/  
L 9 ABBREVIATION : FACSOCOCC  
FACTURE#ADRESSEE#AU#CONGRESSISTE/  
L 6 ABBREVIATION : FACCON  
FACTURE#ADRESSEE##L'ENTREPRISE/  
L 6 ABBREVIATION : FACENT  
GRUPE#DE#CODIFICATION/  
L 1 ABBREVIATION : GROUPECODIF  
ACTIVR#UNE#RESERVATION#EN#ATTENTE/  
L 1 ABBREVIATION : ACTRESEATT  
GRUPE#RESPONSABLE#DE#L'HEBERGEMENT/  
L 1 ABBREVIATION : GROURESHEBE



AFFECTATION#DE#CHAMBRES#AU#CONGRESSISTE/  
L 1 ABREVIATION : AFFECHAMCON

REGLEMENT#DES#LITIGES#CCNCERNANT#L'HEBERGEMENT/  
L 7 ABREVIATION : REGLIHE

GROUPE#DES#HOTESSES#A#L'ACCUEIL/  
L 1 ABREVIATION : GROUHOTEACCU

RESPONSABLE#DES#LITIGES#HEBERGEMENT/  
L 1 ABREVIATION : RESPLITIHEBE

COMPLETER#DOSSIER#D'INSCRIPTION#EN#ATTENTE/  
L 1 ABREVIATION : COMPDOSINSAT

REGLEMENT#DES#PROBLEMES#DE#LITIGE#D'INSCRIPTION/  
L 1 ABREVIATION : REGLPROLITIA

NOM#DU#CONGRESSISTE#INSCRIT/  
L 9 ABREVIATION : NOMCONIAS

CONTRCLE#ET#RELEV#INFORMATICNS#INSCRIPTIGNS#NOUVELLES/  
L 1 ABREVIATION : CONRELINFINNC

INSCRIPTION#DU#CONGRESSISTE/  
L 8 ABREVIATION : INSCCONG

INSCRIPTION-HEBERGEMENT-FACTURATION/  
L 1 ABREVIATION : INSHEBEFAC

INSCRIPTION##D'UN##CONGRESSISTE##/  
L 8 ABREVIATION : INSCCONG

RECEPTION#DE#LA#REQUETE#D'UN#CONGRESSISTE/  
L 1 ABREVIATION : RECEREQUCCN

RESPONSABLE#DES#LITIGES#INSCRIPTION/  
L 1 ABREVIATION : RESLITIINS

LETTRE#DE#REFUS/  
L 9 ABREVIATION : LETTREFUS

EDITION#DE#LA#LISTE#DES#PARTICIPANTS#PAR#ASSOCIATION/  
L 1 ABREVIATION : EDILIPARAS

LISTE#DES#ERREURS/  
L 1 ABREVIATION : LISTEERREU

LISTE#PAR#ASSOCIATION/  
L 1 ABREVIATION : LISTEASSCC

LISTE#COMPLETE/  
L 5 ABREVIATION : LISTE

LISTE#DES#DECOMPTES/  
L 1 ABREVIATION : LISTEDECOM

LISTE#PAR#PAYS/  
L 9 ABREVIATION : LISTEPAYS

LISTE#PAR#TYPE#D'HEBERGEMENT/  
L 1 ABREVIATION : LISTYPEHEBE

LISTE#PAR#HOTELS#OU#RESIDENCES/  
L 1 ABREVIATION : LISHOTRESI

LISTE#DES#NOUVELLES#INSCRIPTIONS/  
L 1 ABREVIATION : LISNOUVINS

LISTE#DES#MODIFICATIONS/  
L 8 ABREVIATION : LISMODIF

EDITION#DE#LA#LISTE#COMPLETE#DES#INSCRITS/  
L 9 ABREVIATION : EDITLISIN

OPTICNS#CCNCERNANT#LES#SALLES#DE#REUNION,L'HEBERGEMENT/  
L 9 ABREVIATION : OPTSAREHE

MAINTENANCE,MODIFICATION#DES#SERVICES,DE#L'ORGANISATION/  
L 1 ABREVIATION : MAINMOSEROR

TRAITEMENT#DES#MODIFICATIONS#HEBERGEMENT/  
L 1 ABREVIATION : TRAIMODIHEBE

TRAITEMENT#DES#MODIFICATIONS#D'INSCRIPTIONS/  
L 1 ABREVIATION : TRAIMODIINS

MODIFICATIONS#DE#L'ORGANISATION/  
L 1 ABREVIATION : MODIFORGAN

MODIFICATIONS#DE#RESERVATION/  
L 1 ABREVIATION : MODIFRESER

MONTANT#DES#ARRHES#VERSES/  
L 1 ABREVIATION : MONTAARRVER

MONTANT#DES#DROITS#D'INSCRIPTION/  
L 1 ABREVIATION : MONTADRCIINS

MONTANT#DU#PAIEMENT/  
L 8 ABREVIATION : MONTPAIE

MOYEN#DE#TRANSPORT/  
L 1 ABREVIATION : MOYENTRANS

NATURE#DES#ACTIVITES#FINANCEES/  
L 1 ABREVIATION : NATUACTIFINA

NOM#DU#CONGRESSISTE/  
L 7 ABREVIATION : NOMCONG

NUMERO#D'ORDRE#DU#VERSEMENT/  
L 1 ABREVIATION : NUMEORDVERS

NUMERO#D'ORDRE#DE#LA#CHAMBRE/  
L 1 ABREVIATION : NUMEORDCHAM

NUMERO#DU#CHEQUE/  
L 1 ABREVIATION : NUMERCHEQU

NUMERO#DU#COMPTE/  
L 9 ABREVIATION : NUMERCOMP

NUMERO#D'ORDRE#DU#CONGRESSISTE/  
L 1 ABREVIATION : NUMEORDCON

NUMERO#D'ORDRE#DE#L'INSCRIPTION/  
L 1 ABREVIATION : NUMEORDINS

NUMERO#D'ORDRE#DU#PAIEMENT/  
L 1 ABREVIATION : NUMEORDPAIE

NUMERO#DE#LA#CHAMBRE#DANS#L'HOTEL/  
L 1 ABREVIATION : NUMECHAMHOTE

NUMERO#D'ORDRE#DE#LA#RESERVATION/  
L 1 ABREVIATION : NUMEORDRESE

ENSEMBLE#DES#OPERATIONS#COMPTABLES/  
L 1 ABREVIATION : ENSEOPERCOM

PROJET#DE#CONTRAT/  
L 7 ABREVIATION : PROJCON

PERCEPTION#DES#DROITS#D'INSCRIPTION/  
L 1 ABREVIATION : PERCROIINS

PERF#VERIF#DES#CARTES/  
L 1 ABREVIATION : PERVERICAR

PLANIFICATION#DE#LA#REALISATION#DES#DIFFERENTS#CONGRES/  
L 1 ABREVIATION : PLANREDICO

VERIFICATION#DE#L'INSCRIPTION#ET#POINTAGE#DU#CONGRESSISTE/  
L 9 ABREVIATION : VERINPOCO

PLANIFICATION#DES#CHARGES#PCST-CCNGRES/  
L 1 ABREVIATION : PLANCHPOCC

COMPTES#RENDUS-SECRETARIAT#POST-CONGRES/  
L 1 ABREVIATION : COMRESEPOCO

PAYS#D'ORIGINE/  
L 9 ABREVIATION : PAYSORIGI

PRENOM#DU#CONGRESSISTE/  
L 9 ABREVIATION : PRENOCONG

INDICATEUR#DE#PRESENCE#AU#CCNGRES/  
L 1 ABREVIATION : INDIPRESCCN

PREVISION-PLANNING#DES#CONGRES/  
L 1 ABREVIATION : PREVPLANCCN

PRIX#JOURNALIER/  
L 8 ABREVIATION : PRIXJOUR

PREPARATIFS#DU#CONGRES/  
L 9 ABREVIATION : PREPACONG

PRISE#EN#COMPTE#DE#LA#MODIFICATION/  
L 1 ABREVIATION : PRISCOMMODI

PROGRAMME#SCIENTIFIQUE/  
L 9 ABREVIATION : PROGSCIEN

PROGRAMME#TOURISTIQUE/  
L 9 ABREVIATION : PROGTOURI

PROMOTION-PUBLICITE/  
L 1 ABREVIATION : PROMOPUBLI

PROJET#DE#PROGRAMME/  
L 9 ABREVIATION : PROJEPRCG

PUBLICITE#INFORMATION/  
L 1 ABREVIATION : PUBLIINFCR

REALISATION#SIMULTANEE#DE#PLUSIEURS#CONGRES/  
L 1 ABREVIATION : REALSIMUCON

RECEPTION,TRI#ET#REPARTITION#DU#COURRIER/  
L 1 ABREVIATION : RECETRIRECO

RECEPTION,MISE#EN#ATTENTE#DES#VERSEMENTS/  
L 8 ABREVIATION : RECEATVE

RECRUTEMENT#DU#PERSONNEL#ACCUEIL#ET#CONFERENCES/  
L 1 ABREVIATION : RECPERACCC

REFUS#DE#L'INSCRIPTION/  
L 9 ABREVIATION : REFUSINSC

REFUS#DE#LA#MODIFICATION/  
L 1 ABREVIATION : REFUSMOCIF

REFUS#DES#VERSEMENTS#ERRONES/  
L 1 ABREVIATION : REFUVERERRO

RELANCE#DES#INSCRIPTIONS/  
L 9 ABREVIATION : RELANINSC

RELANCE#POUR#DOSSIER#INCCMPLET/  
L 1 ABREVIATION : RELADOSINCO

RELANCE#POUR#ERREUR#CU#DOSSIER#INSCRIPTION/  
L 1 ABREVIATION : RELAERDOSIN

REMISE#DE#LA#DOCUMENTATION#PARTICIPANTS#ET#CU#BADGE/  
L 1 ABREVIATION : REMIDOPABA

RENCONTRE#DU#COMITE#D'ORGANISATION#ET#PALAIS#CONGRES/  
L 1 ABREVIATION : RENCOMORPACC

DEMANDE#DE#RENSEIGNEMENTS/  
L 1 ABREVIATION : DEMANRESE

REPONDE#A#UNE#DEMANDE#DE#RENSEIGNEMENT/  
L 1 ABREVIATION : REPCDEMAREN

RENVCI#DES#DOSSIERS#D'INSCRIPTION/  
L 9 ABREVIATION : RENCOSINS

REPONSES#INDIVIDUALISEES#AU#COURRIER/  
L 1 ABREVIATION : REPCINDICUR

RESERVATION#DE#CHAMBRES#POUR#UN#CONGRESSISTE/  
L 1 ABREVIATION : RESECHAMCON

RESPONSABLE#FINANCIER/  
L 1 ABREVIATION : RESPOFINAN

OPTICNS#CONCERNANT#LA#RESTAURATION#LE#TRANSECRT/  
L 9 ABREVIATION : OPTIRETRA

REUNION#DU#COMITE#D'ORGANISATION/  
L 1 ABREVIATION : REUNCOMIORGA

GRUPE#DES#SECRETAIRES#CHARGE#DU#SUIVI#DU#CCNGRES/  
L 1 ABREVIATION : GROSECSUCC

SECRETARIAT#DU#CONGRES/  
L 9 ABREVIATION : SECRECONG

SECRETARIAT#AUX#INSCRIPTIONS/  
L 9 ABREVIATION : SECREINSC

SECRETARIAT#DES#AGENCES#DE#TOURISME/  
L 1 ABREVIATION : SECAGENTOUR

MISE#FN#PLACE#DES#DIVERS#SERVICES#DU#CONGRES/  
L 9 ABREVIATION : PLACISECC

SUIVI#DU#DEROULEMENT/  
L 1 ABREVIATION : SUIVIDEROU

SUIVI#D'ORGANISATION/  
L 1 ABREVIATION : SUIVIORGAN

TRAITEMENT#DES#INSCRIPTIONS#ET#MODIFICATIONS/  
L 1 ABREVIATION : TRAIINSMCCI

TRANSMISSION#DES#MODIFICATIONS/  
L 9 ABREVIATION : TRAMODIF

TYPE#DE#PAIFMENT-CCP,CHEQUE#BANCAIRE,LIQUICE/  
L 1 ABREVIATION : TYPPACCHEBALI

TYPE#DE#LA#CHAMBRE/  
L 8 ABREVIATION : TYPECHAM

TYPE#DE#CONGRESSISTE/  
L 8 ABREVIATION : TYPECONG

TYPE#D'HEBERGEMENT/  
L 9 ABREVIATION : TYPEHEBER

TYPE#D'HOTEL#OU#DE#LA#RESIDENCE/  
L 1 ABREVIATION : TYPEHOTERESI

TYPE#D'OCCUPATION-PENSICN-DEMI#PENSION-NUIT/  
L 1 ABREVIATION : TYOCPECEPENU

REALISATION#D'UN#CONGRES/  
L 9 ABREVIATION : REALICONG

VERSEMENT#DES#ARRHES/  
L 1 ABREVIATION : VERSEARRHE

VERSEMENT#D'UN#PARTICIPANT/  
L 1 ABREVIATION : VERSEPARTI

L 0 ABREVIATION :

FIN METHCDE 3  
2 LOGOUT  
LOGOUT BASE: ABRSOCl

\*\* SOCRATE STARTED \*\* V1.6 LE 133.76 A 23.17.19

1 EXEC TRACE-DIC  
2 ?

ANNEXE A3.3

3 ANTISPE

DICTIONNAIRE

DICTIONNAIRE : ANTISPE

DIFFERENTS	...
CONCERNANT	...
MISE	...
PORTEES	...
NECESSAIRES	...
RENDUS	...
ADRESSEE	...
CHARGE	...
PARTIELS	...
REALISES	...
BORDEREAUX	...
DIVERS	...

4 SPECIAL4

DICTIONNAIRE : SPECIAL4

CHAUTARDDENIS	CHAUTARD
CHOIX	CHX
COMPTESENDUS	CPREN
CCP	CCP

\*\* SOCRATE STARTED \*\* V1.6 LE 133.76 A 23.17.26

1	EXEC CAL-EDIT-LIB	RESULTAT
2	?	
3	0004	METHODE4
4	0173	
5	0352	

METHODE NUMERO 4

- NOM#DES#ACCOMPAGNANTS/  
L 7 ABREVIATION : NOMACOM
  
- ACCUEIL, INSCRIPTIONS, HEBERGEMENT/  
L 8 ABREVIATION : ACINSHEB
  
- PREPARATIFS#ADMINISTRATIFS-SECURITE/  
L 1 ABREVIATION : PREPADMSEC
  
- ADRESSE#DE#L'HOTEL#OU#DE#LA#RESIDENCE/  
L 8 ABREVIATION : ADHCTRES
  
- ADRESSE#PERSONNELLE#CU#DE#L'ENTREPRISE#DU#CCNGRESSISTE/  
L 9 ABREVIATION : APERENCON
  
- ADRESSE#D'HEBERGEMENT/  
L 5 ABREVIATION : ADHEB
  
- GROUPEMENT#DES#AGENCES#DE#TOURISMES#ET#HOTELIERS/  
L 1 ABREVIATION : GROUATOUHO
  
- ANALYSE#DES#ERREURS/  
L 1 ABREVIATION : ANALYEREURS
  
- ANALYSE#DES#RESULTATS#PARTIELS/  
L 1 ABREVIATION : ANALYRESUL
  
- OPTIONS#CONCERNANT#L'ANIMATION#ET#LE#TOURISME/  
L 9 ABREVIATION : OPTANTOUR
  
- ASSOCIATION/  
L 7 ABREVIATION : ASOCIAT
  
- ATELIER#DE#PERFORATION/  
L 1 ABREVIATION : ATELIPERFO
  
- MISE#EN#ATTENTE#DES#CARTES#EN#VUE#DU#TRAITEMENT/  
L 1 ABREVIATION : ATCARTRAIT
  
- RESERVATION#DES#CHAMBRES/  
L 1 ABREVIATION : RESERACHAMBRES
  
- FACTURATION#AUTOMATIQUE#DES#CONGRESSISTES/  
L 1 ABREVIATION : FACTAUTCONG
  
- TRAITEMENT#AUTOMATIQUE#DES#INSCRIPTIONS/



L 1 ABREVIATION : TRAITAUTINSC  
GROUPE#FONCTIONNEL#AUTOMATIQUE/  
L 1 ABREVIATION : GROUPEFONCTAUT  
NOM#DE#LA#BANQUE#DU#DEBITEUR/  
L 1 ABREVIATION : NOMBANQDEB  
BILAN#D'UN#CONGRES/  
L 1 ABREVIATION : BILANCONGRES  
ARRETE#DU#BILAN/  
L 1 ABREVIATION : ARRETBILAN  
BILANS#DES#CONGRES#REALISES#DANS#L'ANNEE/  
L 9 ABREVIATION : BILCONGAN  
BUDGET#PREVISIONNEL/  
L 1 ABREVIATION : BUDGETPREVI  
ENSEMBLE#DE#CHAMBRES#DISPONIBLES/  
L 1 ABREVIATION : ENSCHAMBOISP  
ELABORATION#DU#CAHIER#DES#CHARGES/  
L 1 ABREVIATION : ELCAHICHARG  
CHAUTARD#DENIS/  
L 8 ABREVIATION : CHAUTARD  
CODIFICATION#DES#DONNEES#HEBERGEMENT#SUR#BORDEREAU/  
L 8 ABREVIATION : CODCNHEB  
CODIFICATION#DE#LA#MODIF/  
L 9 ABREVIATION : CODIMODIF  
CODIFICATION#DES#VERSEMENTS/  
L 9 ABREVIATION : CODIVERSE  
CODIFICATION#DES#INFORMATIONS#SUR#BORDEREAU/  
L 9 ABREVIATION : CODINFORA  
COMITE#D'ORGANISATION/  
L 8 ABREVIATION : COMIORGA  
CHCIX#DES#CONFERENCES#ET#DES#CONFERENCIERS/  
L 1 ABREVIATION : CHXCONFCONF  
CONFIRMATION#DES#RESERVATIONS#DE#SALLES#ET#D'HEBERGEMENT/  
L 9 ABREVIATION : CONRESAHE  
LETTRE#DE#CONFIRMATION#ADRESSE#AL#CONGRESSISTE/  
L 1 ABREVIATION : LETCONFCONG  
NOM#DU#CONGRESSISTE#QUI#A#RESERVE/  
L 1 ABREVIATION : NOMCONGRES  
GESTION#DES#CONGRES/  
L 1 ABREVIATION : GESTIONCONGRES  
ENSEMBLE#DES#CONGRESSISTES/

L 1 ABREVIATION : ENSEMCONGRE  
EBAUCHE#DES#DESIDERATA#DU#CLIENT/  
L 9 ABREVIATION : EBDESCLIE  
RECLEMENT#DES#IMPAYES,SUITES#JUDICIAIRES/  
L 9 ABREVIATION : REINSUIJU  
SIGNATURE#DU#CONTRAT/  
L 1 ABREVIATION : SIGNACONTRAT  
CONTROLE#DES#ARRHES#ET#DES#BULLETINS#D'HEBERGEMENT/  
L 1 ABREVIATION : CONARHESBUHE  
CONTROLE#FINANCIER#DES#INSCRIPTIONS/  
L 1 ABREVIATION : CONFININSC  
CONTROLE#DES#DONNEES#DU#BULLETIN#D'HEBERGEMENT/  
L 9 ABREVIATION : CONDOBUHE  
CONTROLE#DES#INFORMATIONS#PORTEES#SUR#LES#BULLETINS/  
L 1 ABREVIATION : CONTINFUL  
CONTROLE#DES#VERSEMENTS/  
L 1 ABREVIATION : CONTROVERSE  
REMISE#DU#COURRIER#AU#CONGRESSISTE/  
L 1 ABREVIATION : REMCOURCONG  
DATE#D'ARRIVEE/  
L 8 ABREVIATION : DATEARRI  
DATE#DE#DEPART/  
L 1 ABREVIATION : DATEDEPART  
DATE#DU#PAIEMENT/  
L 1 ABREVIATION : DATEPAIEMENT  
DATE#DE#L'INSCRIPTION/  
L 1 ABREVIATION : DATEINSCRI  
DATE#DU#VERSEMENT/  
L 9 ABREVIATION : DATEVERSE  
EDITION#DES#DECOMPTES#PAR#REUNIONS,EXCURSIONS/  
L 7 ABREVIATION : EDEREIN  
DEFINITION#DU#CONGRES,DES#SERVICES#DEMANDES/  
L 1 ABREVIATION : DECCNSERDE  
RECEPTION#ET#TRAITEMENT#DES#DEMANDES/  
L 1 ABREVIATION : RECTRAITDEM  
GROUPE#DE#DIRECTION/  
L 1 ABREVIATION : GROUPEDIRICI  
ENVOI#DE#DOCUMENTATION#NOUVEAUX#INSCRITS/  
L 9 ABREVIATION : ENDCNOUIN  
REALISATION#DES#DOSSIERS#D'INSCRIPTIONS/

L 1 ABBREVIATION : REALDOSINS  
DUREE#DE#LA#RESERVATION/  
L 1 ABBREVIATION : DURERESERA  
EDITION#DES#LISTES#PAR#HOTEL/  
L 9 ABBREVIATION : EDLISTHOT  
L 0 ABBREVIATION :  
ENCAISSEMENT#DES#PAIEMENTS/  
L 1 ABBREVIATION : ENCAIPAIEMENTS  
ENCAISSEMENT#DES#PAIEMENTS#DES#SOLDES/  
L 1 ABBREVIATION : ENCPAIEMSOLD  
ENVOI#DES#DOSSIERS#D'INSCRIPTION/  
L 1 ABBREVIATION : ENVDOSSINSC  
EDITION#DES#LISTES#PAR#PAYS/  
L 1 ABBREVIATION : EDLISTPAYS  
MATERIEL-EQUIPEMENT#NECESSAIRES/  
L 7 ABBREVIATION : MATEQUI  
ETAT#DE#LA#CHAMBRE-RESERVEE, NON#RESERVEE/  
L 1 ABBREVIATION : ECHAMRENONRE  
ETAT#FINANCIER#DU#CONGRESSISTE/  
L 9 ABBREVIATION : ETFINCONG  
EDITION#PAR#TYPE#D'HEBERGEMENT/  
L 8 ABBREVIATION : EDTYPHEB  
EXPEDITION#DES#FACTURES/  
L 9 ABBREVIATION : EXPEFACTU  
FACTURATION#DES#SERVICES#RENOUS#A#L'ASSOCIATION/  
L 1 ABBREVIATION : FACTSERVAS  
FACTURATION#DES#PARTICIPANTS/  
L 1 ABBREVIATION : FACTUPARTI  
FACTURATION#DES#SOLDES#DES#COMPTES#CONGRESSISTES/  
L 1 ABBREVIATION : FACSOLCOMCON  
FACTURE#ADRESSEE#AU#CONGRESSISTE/  
L 1 ABBREVIATION : FACTUCONGRE  
FACTURE#ADRESSEE#A#L'ENTREPRISE/  
L 1 ABBREVIATION : FACTUENTRE  
GRUPE#DE#CODIFICATION/  
L 1 ABBREVIATION : GROUPECODI  
ACTIVER#UNE#RESERVATION#EN#ATTENTE/  
L 8 ABBREVIATION : ACTRESAT  
GRUPE#RESPONSABLE#DE#L'HEBERGEMENT/  
L 1 ABBREVIATION : GROURESHEB

NUMERO#DU#COMPT#/  
L 1 ABBREVIATION : NUMECCMPTE

NUMERO#D'ORDRE#DU#CONGRESSISTE/  
L 1 ABBREVIATION : NUMCRDCONG

NUMERO#D'ORDRE#DE#L'INSCRIPTION/  
L 1 ABBREVIATION : NUMCRDINSC

NUMERO#D'ORDRE#DU#PAIEMENT/  
L 1 ABBREVIATION : NUMORDPAIEM

NUMERO#DE#LA#CHAMBRE#DANS#L'HOTEL/  
L 1 ABBREVIATION : NUMCHAMBHOT

NUMERO#D'ORDRE#DE#LA#RESERVATION/  
L 9 ABBREVIATION : NUMCRDRES

ENSEMBLE#DES#OPERATIONS#COMPTABLES/  
L 1 ABBREVIATION : ENSCPCOMPT

PROJET#DE#CONTRAT/  
L 1 ABBREVIATION : PROJETCCNTRAT

PERCEPTION#DES#DROITS#D'INSCRIPTION/  
L 1 ABBREVIATION : PERCDROITSINSC

PERFC#VERIF#DES#CARTES/  
L 1 ABBREVIATION : PERFVERICART

PLANIFICATION#DE#LA#REALISATION#DES#DIFFERENTS#CONGRES/  
L 1 ABBREVIATION : PLANIREALCONG

VERIFICATION#DE#L'INSCRIPTION#ET#POINTAGE#DU#CONGRESSISTE/  
L 1 ABBREVIATION : VEINSPOINCON

PLANIFICATION#DES#CHARGES#PCST-CONGRES/  
L 1 ABBREVIATION : PLACHARPOSTCON

COMPTES#RENDUS-SECRETARIAT#POST-CONGRES/  
L 1 ABBREVIATION : CPRENSEPOSTCON

PAYS#D'ORIGINE/  
L 7 ABBREVIATION : PAYSORI

PRENOM#DU#CONGRESSISTE/  
L 1 ABBREVIATION : PRENOMCONGRE

INDICATEUR#DE#PRESENCE#AU#CONGRES/  
L 1 ABBREVIATION : INDPRESCONG

PREVISION-PLANNING#DES#CONGRES/  
L 1 ABBREVIATION : PREVPLANCCNG

PRIX#JOURNALIER/  
L 1 ABBREVIATION : PRIXJOURNA

PREPARATIFS#DU#CONGRES/  
L 1 ABBREVIATION : PREPACONGR

PRISE#EN#COMPTE#DE#LA#MODIFICATION/  
L 1 ABREVIATION : PRISCOMPTMOD

PROGRAMME#SCIENTIFIQUE/  
L 1 ABREVIATION : PROGRASCIENTI

PROGRAMME#TOURISTIQUE/  
L 1 ABREVIATION : PROGRATCURI

PROMOTION-PUBLICITE/  
L 1 ABREVIATION : PROMOPUBLI

PROJET#DE#PROGRAMME/  
L 1 ABREVIATION : PROJETPROGRA

PUBLICITE#INFORMATION/  
L 8 ABREVIATION : PUBLINFO

REALISATION#SIMULTANEE#DE#PLUSIEURS#CONGRES/  
L 1 ABREVIATION : REALSIMCONG

RECEPTION,TRI#ET#REPARTITION#DU#COURRIER/  
L 1 ABREVIATION : RETRIRECOU

RECEPTION,MISE#EN#ATTENTE#DES#VERSEMENTS/  
L 9 ABREVIATION : RECATVERS

RECRUTEMENT#DU#PERSONNEL#ACCUEIL#ET#CONFERENCES/  
L 1 ABREVIATION : REPERACONF

REFUS#DE#L'INSCRIPTION/  
L 1 ABREVIATION : REFUSINSCRIPI

REFUS#DE#LA#MODIFICATION/  
L 9 ABREVIATION : REFUSMODI

REFUS#DES#VERSEMENTS#ERRONES/  
L 9 ABREVIATION : REFVERSER

RELANCE#DES#INSCRIPTIONS/  
L 1 ABREVIATION : RELANINSCRIPI

RELANCE#POUR#DOSSIER#INCOMPLET/  
L 9 ABREVIATION : RELDOSINC

RELANCE#POUR#ERREUR#DU#DOSSIER#INSCRIPTION/  
L 6 ABREVIATION : REDCIN

REMISE#DE#LA#DOCUMENTATION#PARTICIPANTS#ET#DU#BADGE/  
L 8 ABREVIATION : REDCPABA

RENCONTRE#DU#COMITE#D'ORGANISATION#ET#PALAIS#CONGRES/  
L 1 ABREVIATION : RENCORPACON

DEMANDE#DE#RENSEIGNEMENTS/  
L 1 ABREVIATION : DEMANRENSEI

REPONDE#A#UNE#DEMANDE#DE#RENSEIGNEMENT/  
L 1 ABREVIATION : REPCEMRENS

AFFECTATION#DE#CHAMBRES#AU#CONGRESSISTE/  
L 1 ABREVIATION : AFCHAMBCONG

REGLEMENT#DES#LITIGES#CONCERNANT#L'HEBERGEMENT/  
L 9 ABREVIATION : REGLITHEB

GRUPE#DES#HOTESSES##L'ACCUEIL/  
L 1 ABREVIATION : GROUPHOTAC

RESPONSABLE#DES#LITIGES#HEBERGEMENT/  
L 1 ABREVIATION : RESPLITHEB

COMPLETER#DOSSIER#D'INSCRIPTION#EN#ATTENTE/  
L 9 ABREVIATION : COMCOINSA

REGLEMENT#DES#PROBLEMES#DE#LITIGE#D'INSCRIPTION/  
L 9 ABREVIATION : REPROLINS

NOM#DU#CONGRESSISTE#INSCRIT/  
L 1 ABREVIATION : NOMCONGINS

CONTROLE#ET#RELEVÉ#INFORMATIONS#INSCRIPTIONS#NOUVELLES/  
L 1 ABREVIATION : CONREINFINSNOU

INSCRIPTION#DU#CONGRESSISTE/  
L 1 ABREVIATION : INSCRIPCONGRE1

INSCRIPTION-HEBERGEMENT-FACTURATION/  
L 1 ABREVIATION : INSCHEBFACT

INSCRIPTION##D'UN##CONGRESSISTE##/  
L 1 ABREVIATION : INSCRIPCONGRE2

RECEPTION#DE#LA#REQUETE#D'UN#CONGRESSISTE/  
L 1 ABREVIATION : RECREQCCNG

RESPONSABLE#DES#LITIGES#INSCRIPTION/  
L 1 ABREVIATION : RESPLITINSC

LETTRE#DE#REFUS/  
L 1 ABREVIATION : LETREREFUS

EDITION#DE#LA#LISTE#DES#PARTICIPANTS#PAR#ASSOCIATION/  
L 8 ABREVIATION : ELISPARA

LISTE#DES#ERREURS/  
L 6 ABREVIATION : LISTER

LISTE#PAR#ASSOCIATION/  
L 8 ABREVIATION : LISTEASC

LISTE#COMPLETE/  
L 1 ABREVIATION : LISTECOMPLE

LISTE#DES#DECOMPTES/  
L 1 ABREVIATION : LISTEDECOMP

LISTE#PAR#PAYS/  
L 9 ABREVIATION : LISTEPAYS

LISTE#PAR#TYPE#D'HEBERGEMENT/  
L 1 ABBREVIATION : LISTYPEHEB

LISTE#PAR#HOTELS#OU#RESIDENCES/  
L 1 ABBREVIATION : LISTHOTRES

LISTE#DES#NOUVELLES#INSCRIPTIONS/  
L 1 ABBREVIATION : LISTNOUVINSC

LISTE#DES#MODIFICATIONS/  
L 9 ABBREVIATION : LISTEMODI

EDITION#DE#LA#LISTE#COMPLETE#DES#INSCRITS/  
L 1 ABBREVIATION : ELISCCMINS

OPTIONS#CONCERNANT#LES#SALLES#DE#REUNION,L'HEBERGEMENT/  
L 1 ABBREVIATION : OPSALEREUHE

MAINTENANCE,MODIFICATION#DES#SERVICES,DE#L'ORGANISATION/  
L 1 ABBREVIATION : MAINMOSEROR

TRAITEMENT#DES#MODIFICATIONS#HEBERGEMENT/  
L 1 ABBREVIATION : TRAITMODHEB

TRAITEMENT#DES#MODIFICATIONS#D'INSCRIPTIONS/  
L 1 ABBREVIATION : TRAITMODINSC

MODIFICATIONS#DE#L'ORGANISATION/  
L 8 ABBREVIATION : MODIORGA

MODIFICATIONS#DE#RESERVATION/  
L 9 ABBREVIATION : MODIRESER

MONTANT#DES#ARRHES#VERSES/  
L 0 ABBREVIATION :

MONTANT#DES#DROITS#D'INSCRIPTION/  
L 1 ABBREVIATION : MONTDROITSINSC

MONTANT#DU#PAIEMENT/  
L 1 ABBREVIATION : MONTANTPAIEMEN

MOYEN#DE#TRANSPORT/  
L 1 ABBREVIATION : MOYENTRANSPORT

NATURE#DES#ACTIVITES#FINANCEES/  
L 9 ABBREVIATION : NATACTFIN

NOM#DU#CONGRESSISTE/  
L 8 ABBREVIATION : NOMCONGR

NUMERO#D'ORDRE#DU#VERSEMENT/  
L 1 ABBREVIATION : NUMORDVERS

NUMERO#D'ORDRE#DE#LA#CHAMBRE/  
L 1 ABBREVIATION : NUMORDCHAMB

NUMERO#DU#CHEQUE/  
L 1 ABBREVIATION : NUMECHEQUE

RENCI#DES#DOSSIERS#C'INSCRIPTION/  
L 1 ABREVIATION : RENVDOSINSC

REPONSES#INDIVIDUALISEES#AU#COURRIER/  
L 1 ABREVIATION : REPINDCCUR

RESERVATION#DE#CHAMBRES#POUR#UN#CONGRESSISTE/  
L 1 ABREVIATION : RESCHAMBCONG

RESPONSABLE#FINANCIER/  
L 1 ABREVIATION : RESPONFINAN

OPTIONS#CONCERNANT#LA#RESTAURATION#LE#TRANSFCRT/  
L 1 ABREVIATION : OPTRESTRANS

REUNION#DU#COMITE#D'ORGANISATION/  
L 1 ABREVIATION : REUNCOMORG

GRUPE#DES#SECRETAIRES#CHARGE#DU#SUIVI#DU#CCNGRES/  
L 1 ABREVIATION : GROLSESUICON

SECRETARIAT#DU#CONGRES/  
L 1 ABREVIATION : SECRECONGRES

SECRETARIAT#AUX#INSCRIPTIONS/  
L 1 ABREVIATION : SECREINSCRI

SECRETARIAT#DES#AGENCES#DE#TOURISME/  
L 9 ABREVIATION : SECAGTOUR

MISE#EN#PLACE#DES#DIVERS#SERVICES#DU#CONGRES/  
L 1 ABREVIATION : PLACSERVCONG

SUIVI#DU#DEROULEMENT/  
L 1 ABREVIATION : SUIVIDEROU

SUIVI#D'ORGANISATION/  
L 9 ABREVIATION : SUIVIORGA

TRAITEMENT#DES#INSCRIPTIONS#ET#MODIFICATIONS/  
L 1 ABREVIATION : TRAITINSCMOD

TRANSMISSION#DES#MODIFICATIONS/  
L 1 ABREVIATION : TRANSMIMODI

TYPE#DE#PAIEMENT-CCP,CHEQUE#BANCAIRE,LIQUIDE/  
L 1 ABREVIATION : TYPaicCPCHEBAN

TYPE#DE#LA#CHAMBRE/  
L 1 ABREVIATION : TYPECHAMBRE

TYPE#DE#CONGRESSISTE/  
L 1 ABREVIATION : TYPECONGRE

TYPE#D'HEBERGEMENT/  
L 9 ABREVIATION : TYPEHEBER

TYPE#D'HOTEL#OU#DE#LA#RESIDENCE/  
L 1 ABREVIATION : TYPEHOTRES



TYPE#D'OCCUPATION-PENSION-DEMI#PENSION-NUIT/  
L 1 ABBREVIATION : TYOCPENDEPENUI

REALISATION#D'UN#CONGRES/  
L 1 ABBREVIATION : REALICONGRES

VERSEMENT#DES#ARRHES/  
L 1 ABBREVIATION : VERSEARHES

VERSEMENT#D'UN#PARTICIPANT/  
L 1 ABBREVIATION : VERSEPARTI

FIN METHCDE 4  
\* RUN FN:R

**ANNEXE A4**

**DICTIONNAIRES STANDARD D'ABREGE**

\*\* SOCRATE STARTED \*\* V1.6

LE 133.76

A 23.17.37

1 FXFC TRACE-DIC  
2 ?

Ce dictionnaire ne reflète  
que partiellement le dictionnaire  
ANTIDIC qui sera fourni à  
l'utilisateur.

3 ANTICIC

DICTIONNAIRE : ANTIDIC

LE	...
LA	...
LES	...
DE	...
DU	...
DES	...
DONT	...
QUF	...
QUOI	...
QUI	...
QUEL	...
QUELS	...
QUELLE	...
QUELLES	...
QU'	...
IL	...
ILS	...
ELLE	...
ELLES	...
ON	...
JE	...
TU	...

NOUS	...
VOUS	...
CE	...
CET	...
CES	...
CELLE	...
CELLES	...
CELUI	...
SE	...
ME	...
TE	...
LUI	...
LEUR	...
LEURS	...
UN	...
UNE	...
UNS	...
UNES	...
EN	...
Y	...
A	...
AU	...
AUX	...
CEUX	...
CECI	...
QUELQUE	...
QUELQUES	...
CHAQUE	...
PAR	...
POUR	...

AVEC	...
PENDANT	...
AVANT	...
DURANT	...
APRES	...
ALASUITE	...
ENVUE	...
ENFONCTION	...
PARRAPPORT	...
ET	...
OU	...
NI	...
OR	...
CAR	...
MAIS	...
DONC	...
HORS	...
SUR	...
SOUS	...
DESSOUS	...
DESSUS	...
CHACUN	...
CHACUNE	...
CHACUNS	...
CHACUNES	...
TOUT	...
TOUS	...
TOUTE	...
TOUTES	...
AUSSI	...

AINSI	...
ENCORE	...
TANT	...
AUTANT	...
PUTS	...
PLUS	...
MOINS	...
TROP	...
AUPRES	...
PLUSIEURS	...

4 \*\*\*  
FIN EDITION  
8 LOGOUT  
LOGOUT BASE: ABRSOCl

- à compléter -

LE DICTIONNAIRE STANDIC N'A PAS ETE  
ENREGISTRE DANS LA BASE. IL EST ENCORE A  
ANALYSER



**ANNEXE A5**

**L'ANALYSEUR PHONETIQUE**



## ANALYSE DE REPONSES REDIGES EN FRANÇAIS COURANT POUR UNE REALISATION D'ENSEIGNEMENT PROGRAMME

par Daniel COULON (1) et Daniel KAYSER (1)

• • •

### 4. Le phonème

Nous utilisons le phonème comme intermédiaire afin de négliger les principales fautes d'orthographe commises par les étudiants. Nous allons maintenant discuter l'ensemble des phonèmes utilisés et la méthode employée pour les obtenir.

Il ne s'agit pas pour nous de refléter avec exactitude la prononciation de chaque mot : en effet, notre propos est de confondre en un seul codage les différentes orthographe d'un même mot, quand bien même leur prononciation rigoureuse se révélerait légèrement différente. C'est pourquoi nous ne concentrerons pas notre effort sur une transcription phonétique fidèle, mais sur un codage d'emploi facile et efficace.

Nous allons être amenés à décomposer chaque mot en groupes de lettres, de même qu'on sépare les lettres des mots que les enfants apprennent à lire.

Définissons quelques termes qui vont être utiles : Soit  $M$  une chaîne de caractères :  $M = a_1, a_2, \dots, a_m$ .

Les  $a_i$  représentent soit une lettre, soit le blanc, séparateur entre deux mots. La virgule représente l'opération de concaténation.

Nous admettons l'existence d'une application  $K$ , qui à toute chaîne  $M$  associe une chaîne  $P = A_1, A_2, \dots, A_p$ .  $P$  est la prononciation de  $M$ . Les  $A_i$  seront, par abus de langage, appelés « sons ».

Nous définissons l'ensemble  $G$  des groupes phonétiques par la relation suivante :

$$\sigma = a_i, a_j, \dots, a_k \in G \quad \text{si } K(\sigma) = A_1, A_2, \dots, A_k.$$

étant la prononciation de  $g$ , il existe un caractère  $a'_n$  tel que

$$K(a_1, a_2, \dots, a_n) = A'_1, A'_2, \dots, A'_q \quad \text{avec } A'_1 \neq A_1.$$

C'est-à-dire, en langage clair, qu'une variation de la dernière lettre d'un groupe phonétique peut affecter le premier élément de sa transcription phonétique. Prenons un exemple : le mot ANATOMIE n'est pas un groupe phonétique, car on aura beau remplacer le E final par toutes les lettres de l'alphabet, l'espace y compris, la prononciation du mot commencera toujours par un a. Au contraire, ANA est un groupe phonétique, car la substitution du A final par un S fait passer le premier phonème de a à à.

Un groupe phonétique est dit maximal s'il ne demeure jamais groupe phonétique lorsqu'on lui rajoute un caractère à droite. L'intérêt de cette notion est qu'à partir du moment où un groupe phonétique maximal a été détecté, le premier élément de sa transcription est connu.

On est amené à utiliser un automate représenté par une table : les lignes correspondent aux états de l'automate; les colonnes contiennent les lettres, et, à l'intersection, on trouve, outre le déplacement du pointeur de lecture des caractères, le nouvel état et, s'il y a lieu, l'action à effectuer, c'est-à-dire la génération du phonème.

L'encombrement de chaque case de cette table serait considérablement réduit si ces deux informations étaient exclusives l'une de l'autre, c'est-à-dire si le fait d'avoir découvert un phonème ramenait toujours l'automate dans son état initial.

Considérons la proposition (P) suivante :

Soit un groupe phonétique maximal  $a_1, \dots, a_l$ . Soit  $A_1, \dots, A_m$  sa prononciation. Il existe  $k, 1 \leq k \leq l$  tel que la prononciation de  $a_{k+1}, \dots, a_l$  soit  $A_2, \dots, A_m$ .

Si cette proposition est vraie, il suffit de mettre dans l'automate :

— soit un numéro d'état intermédiaire. Il s'agit alors d'un groupe phonétique non maximal;

— soit un phonème et la valeur de  $k$  qui permet de reprendre l'étude dans l'état initial.

Nous avons choisi un jeu de phonèmes vérifiant la proposition (P) :

On constate dans ce tableau le regroupement de certains phonèmes. Les uns sont dus au matériel utilisé. Les télétypes à notre disposition ne possèdent que les caractères majuscules. Il n'est donc pas possible de tenir compte des nuances apportées par l'usage des accents. D'autres phonèmes ont été regroupés car ils traduisent des nuances de sons peu importantes, comme  $\tilde{e}$  et  $\tilde{a}$ , ou parce que, de plus, ils s'écrivent de façon identique, comme o et ɔ, ø et œ. D'autres regroupements enfin paraissent plus hasardeux (comme s, z et x).

Table des phonèmes consonnes

Phonème	EXEMPLES
1 p	Paix aPPrecher
2 t	Table aTTiser
3 k	Camp QUestion
4 f	Fuir PHare
5 s	Ce aScension maiSon gaZ reSSort action aXe
6 ʃ	CHant SCHEMA
7 b	Beau aBBe
8 d	Droit aDDITION
9 g	Gant GUêpe
10 v	Vin Wagon
11 ʒ	Je Géant
12 l	fouLer ombrellE
13 r	Roi
14 m	Mou feMME
15 n	Nul aNNonce
16 ɲ	coGNer maNier

Table des phonèmes voyelles

Phonème	EXEMPLES
1 a	paAte pAtte
11 ä	tEMple ENcORE
12 e	rEtour nEZ frElner
13 i	scri aILL IYs
14 é	FIN aucUN
15 ø	pot cOLLe AUtre
16 ë	nomBre
17 ø	RIEUR chantEUse
18 u	gOUss
19 y	mUR
20 w	loin bOA nOir

D. COULON et D. KAYSER

Ils sont cependant nécessaires pour que la proposition (P) soit vérifiée. L'expérience montre que ces regroupements ne sont pas dangereux, les cas où deux termes pouvant figurer dans une même réponse donnent la même écriture phonétique demeurent très rares.

On peut constater que le son u ne s'écrit que d'une seule façon. Il n'y a donc pas de raison pour qu'il soit la cause de fautes d'orthographe. Nous le supprimons du tableau : il sera remplacé par la succession oy.

Table de l'automate

	U	A	B	C	E	H	I	Y	K	L	M	N	O	R	SX	T	U	Z
\$	FIN	A		C	E	H	I	Y	K	L	M	N	O	R	SX	T	U	Z
A	A	A		C	E	H	I	Y	K	L	M	N	O	R	SX	T	U	Z
C	R	R	R	R	S	F	S	S	K	H	K	K	K	K	S	CT	K	S
D	\$	d	d	d	d	d	d	d	d	d	d	d	d	d	d	d	d	d
E	\$	EA	e	e	E	E	E	E	e	e	e	e	e	e	ES	ER	ER	ER
G	\$	\$	\$	C	\$	\$	\$	\$	K	S	G	G	G	G	S	\$	\$	\$
I	i	i	i	i	i	i	i	i	i	IL	IN	IN	i	i	i	i	i	i
K	k	k	k	C	k	k	k	k	K	K	K	K	K	K	S	K	K	S
N	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n
O	o	w	o	o	OE	w	w	w	o	o	ON	ON	o	o	o	o	o	o
P	\$	p	p	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
Q	k	k	k	C	k	k	k	k	K	S	K	K	K	K	S	K	K	K
S	\$	S	S	SC	S	S	S	S	S	S	S	S	S	S	S	S	S	S
T	\$	t	t	t	t	t	t	t	t	t	t	t	t	t	t	t	t	t
U	u	u	u	u	u	u	u	u	u	u	u	u	u	u	u	u	u	u
AI	e	a	e	e	e	e	e	e	AI	e	AIL	AIN	a	o	e	e	a	e

ANALYSE DE REPONSES REDIGES EN FRANÇAIS COURANT

	L	A	C	E	H	IY	K	L	M	N	O	R	SX	T	U	Z
AN	ā	ā	ā	ā	ā	ā	ā	ā	ā	ā	ā	ā	ā	ā	ā	ā
CT	h	h	h	h	h	CTI +	h	h	h	h	h	h	h	h	h	k
EA	a	a	a	a	a	AI +	a	a	AN +	a	a	a	a	a	a	a
EI	e	e	e	e	e	EI +	e	e	e	e	e	e	e	e	e	e
EN	ē	ē	ē	ē	ē	ē	ē	ē	ē	ē	ē	ē	ē	ē	ē	ē
ER	e	e	e	e	e	e	e	e	e	e	e	e	e	e	e	e
ES	ɛ	ɛ	ɛ	ɛ	ɛ	ɛ	ɛ	ɛ	ɛ	ɛ	ɛ	ɛ	ɛ	ɛ	ɛ	ɛ
IL	i	i	i	i	i	i	i	i	i	i	i	i	i	i	i	i
IN	ī	ī	ī	ī	ī	ī	ī	ī	ī	ī	ī	ī	ī	ī	ī	ī
NI	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n
OE	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o
ON	ō	ō	ō	ō	ō	ō	ō	ō	ō	ō	ō	ō	ō	ō	ō	ō
OU	u	u	u	u	u	u	u	u	u	u	u	u	u	u	u	u
SC	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s
TI	t	t	t	t	t	t	t	t	t	t	t	t	t	t	t	t
UN	ū	ū	ū	ū	ū	ū	ū	ū	ū	ū	ū	ū	ū	ū	ū	ū
AIL	a	a	a	a	a	a	a	a	a	a	a	a	a	a	a	a
AIN	ē	ē	ē	ē	ē	ē	ē	ē	ē	ē	ē	ē	ē	ē	ē	ē
CTI	h	h	h	h	h	h	h	h	h	h	h	h	h	h	h	h
EIL	e	e	e	e	e	e	e	e	e	e	e	e	e	e	e	e

Les phonèmes i et j sont difficiles à distinguer en cas de fautes d'orthographe. Ainsi TRANQUILLE donne trākjil alors que l'usage le prononce

D. COULON et D. KAYSER

trākil. Certaines orthographe peuvent donner naissance à deux prononciations : par exemple PAYE se transcrit suivant les cas pe ou pej. Ces raisons ont conduit à supprimer le phonème j et à modifier en conséquence la table des phonèmes.

L'automate de prononciation associé au jeu de phonèmes indiqué possède 35 états et a besoin d'environ 2,5 tests pour sortir un son, ce qui représente une performance acceptable à la fois sur le plan de l'encombrement et sur celui de la rapidité du codage.

EXEMPLE

ÉTAT DU TAMPON	CARACTÈRE LU	PHONÈME PRODUIT
\$ A T T E E I A I N T	A T T E I N T _	- a - t - ε -
\$ A C C T I I C E	F A C T I C - C E _	f - a - k t i s -

ATTEINT

FACTICE

REMARQUE : les lettres n'apparaissant pas dans les états de l'automate sont sorties immédiatement sous forme de phonème à moins qu'elles ne soient suivies d'une lettre identique.

Extrait du Chapitre II de la Thèse de  
3<sup>e</sup> cycle de D. Kayser

Les représentations usuelles d'un graphe par sa matrice associée ou sa matrice d'incidence sont très mal adaptées à une arborescence, dont peu de sommets, en définitive, sont reliés. Si nous adoptons une représentation chaînée (listes), il faudrait adjoindre à chaque "case" un ensemble de pointeurs correspondant aux différents successeurs des sommets et indiquant, pour chacun d'eux, la lettre qui provoque son utilisation.

De plus le temps d'exécution serait affecté par la recherche, dans cet ensemble, du pointeur qui convient.

Il nous a paru plus intéressant de construire un automate, tels que ceux habituellement utilisés dans l'analyse des langages artificiels. Ceci illustre d'ailleurs la similitude de certaines méthodes employées dans le traitement des langues naturelles avec celles mises au point en compilation.

Les différents états de l'automate (sommets non terminaux du graphe) sont portés en ligne ; les colonnes représentent les lettres, et, à l'intersection, on trouve la "case" correspondant au sommet auquel on arrive.

6) Que doit-on mettre dans chaque "case" ? Si l'état n'est pas terminal, il suffit d'indiquer le numéro de la ligne que le représente. Si l'état est terminal, les éléments nécessaires à la poursuite du travail sont :

- a. le ou les "sons" dont l'existence est dorénavant certaine ;
- b. l'état - initial ou intermédiaire - où il faut reprendre le travail pour continuer le codage du mot.

Or l'écriture de l'automate montre que, dans presque tous les cas, la proposition P suivante est vraie :

(P) Soit  $G = a_1, a_2, \dots, a_n$  un groupe phonétique indépendant, et soit  $K(G) = A_1, A_2, \dots, A_p$  sa prononciation. Les  $a_i$  représentent des caractères, les  $A_i$  des "sons".

On a soit :

(P<sub>1</sub>) il existe j tel que pour toute chaîne R,

$$K(G, R) = A_1, K(a_j, \dots, a_n, R)$$

soit (P<sub>2</sub>) K(G,R) = A<sub>1</sub>, K(R).

C'est-à-dire, en d'autres termes, que les informations nécessaires a. et b. peuvent se ramener à la forme plus maniable suivante :

a'. le "son" A<sub>1</sub> dont la présence est sûre ;

b'. la quantité dont il faut reculer l'index d'extraction des lettres (1, 2 ou 3 lorsque (P<sub>1</sub>) est vraie, 0 lorsque (P<sub>2</sub>) est vraie) pour pouvoir reprendre le traitement en remplaçant l'automate dans son état initial.

L'avantage de cette représentation est qu'elle permet d'utiliser, contrairement à la précédente, des cases de taille fixe, et que cette taille est faible (nous verrons qu'elle est de 8 bits).

Notons qu'un corollaire de la proposition (P) est qu'à chaque "son" du codage, on peut faire correspondre la chaîne du mot origine qui lui a donné naissance.

7) Toutes ces considérations, et les conséquences que nous en avons tirées, dépendent du choix de l'ensemble des sons et de la valeur sur cet ensemble de la fonction K.

Nous avons déjà dit qu'il est possible d'envisager la confusion de certains sons voisins : quels vont être les critères choisis pour les déterminer ? Les obstacles à éviter sont

- d'une part, la prolifération des sons retenus : deux orthographes de prononciations très voisines seront alors codés différemment, et nous n'aurons rien résolu. De plus, la taille de l'automate (quantité d'états intermédiaires et nombre de bits de chaque "case") sera très importante.

- d'autre part, une trop grande restriction de l'ensemble des sons : des mots par trop différenciés auront le même codage, ce qui risque d'introduire une grande confusion dans leur reconnaissance.

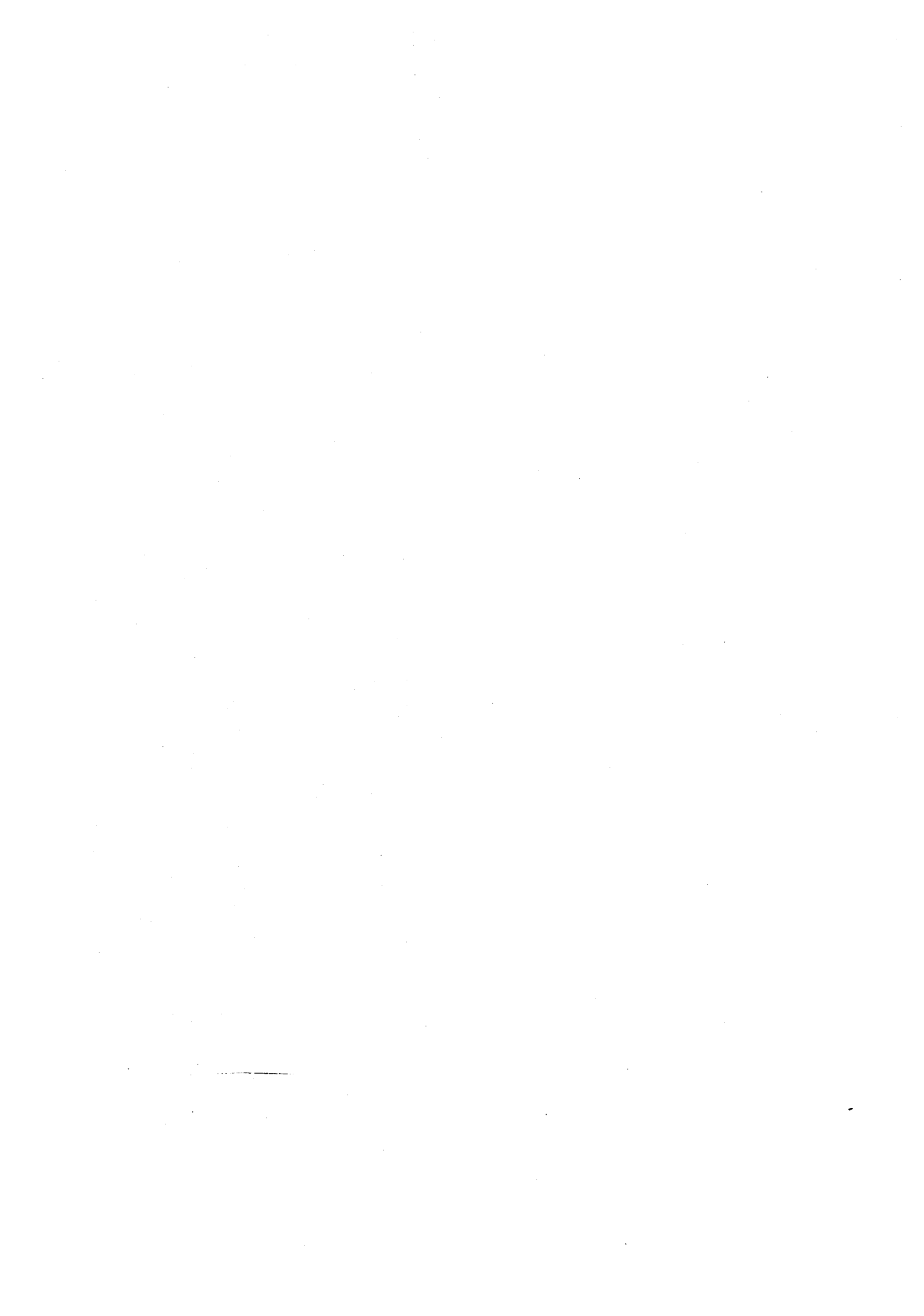
Nous avons choisi un éventail de 26 sons, tels que la propriété (P) soit toujours vraie. Ce sont :

a		i	(i,y,ille)	n	(ni, gn)	s	(s,z,x)
a	(an)	œ	(in, un, ain)	o		ʃ	(ch, sh,sch)
b		ʒ	(j,g doux)	ö	(eu)	t	
d		k	(c dur, qu)	õ	(on)	u	
e	(ai,ei,er)	l		p		v	(v, x)
f	(f,ph)	m		R		wa	(oi, oua)
g	(g dur)	n					

On constate que certains sons assez voisins ont été codés de la manière : s, z, et x par exemple. L'expérience montre que ceci n'engendre pas de contresens.

Nous avons été amenés à prendre cette décision pour éviter que le codage d'une lettre dépende de celles qui précèdent, ce qui rendrait caduque la proposition (P) ; en effet, pour distinguer entre s et z, il est parfois nécessaire de savoir si la lettre précédente n'est pas une voyelle. La confusion des sons s et x s'explique par le souci de ne pas augmenter inutilement le nombre d'états intermédiaires : le double c est, en effet, équivalent, grâce à cet **artifice**, au c simple, quelle que soit la lettre qui le suit, de même que le groupe cs est équivalent au s.

La sélection de ces 26 sons porte le nombre d'états intermédiaires de l'automate à 35. Ceci nous paraît justifier le choix effectué, car l'encombrement est faible pour des performances acceptables.



**ANNEXE A6**

**NORME D'ABREVIATION DES TITRES DE PERIODIQUES**





Les règles contenues dans ce document s'appliquent à l'abréviation des titres de toutes les publications en série et de nombreuses publications non périodiques, y compris les monographies et les comptes rendus de réunions. Ces règles ont pour but de guider et d'aider les auteurs, correcteurs, bibliothécaires et autres personnes travaillant dans le domaine de l'information, lors de la préparation d'abréviations univoques et précises, à l'intérieur d'un cadre de référence donné, pour les titres de publications cités en notes, dans des références ou dans des bibliographies. Ces citations concernent les services d'analyse et d'indexation, les listes de références accompagnant les articles, ainsi que les dossiers publics ou privés, chaque fois qu'une identification de document est importante.

En raison des centaines de milliers de publications en série ou non périodiques citées en abrégé, du nombre de langues dans lesquelles ces textes sont écrits, des formes variées de l'enregistrement des titres de ces publications et de la diversité de formation intellectuelle des personnes utilisant ces abréviations, il est impossible d'établir des règles qui permettent, dans chaque cas, une reconstitution directe du titre original des publications citées sous forme abrégée. Un moyen de faciliter cette identification est d'encourager les auteurs et correcteurs utilisant dans leurs publications des abréviations de titres, à établir fréquemment, et à distribuer à leurs lecteurs des listes de titres abrégés avec les titres non abrégés correspondants.

L'objet de la présente Norme Internationale est de permettre de déterminer pour chaque titre une seule abréviation non ambiguë. Pour ce faire, il faut appliquer les règles énumérées, complétées par une liste d'abréviations normalisées des mots du titre. Grâce à cette méthode, deux titres différents n'auront pas la même abréviation, et une abréviation donnée ne représentera qu'un seul titre.

En vue d'assurer une utilisation effective de la présente Norme Internationale, des listes d'abréviations normalisées de mots contenus dans les titres de publications, en série ou non, doivent être disponibles (voir ISO/R 833 et ANSI/Z 39).

## AVANT-PROPOS

ISO (Organisation Internationale de Normalisation) est une fédération mondiale d'organismes nationaux de normalisation (Comités Membres ISO). L'élaboration de Normes Internationales est confiée aux Comités Techniques ISO. Chaque Comité Membre intéressé par une étude a le droit de faire partie du Comité Technique correspondant. Les organisations internationales, gouvernementales et non gouvernementales, en liaison avec l'ISO, participent également aux travaux.

Les Projets de Normes Internationales adoptés par les Comités Techniques sont soumis aux Comités Membres pour approbation, avant leur acceptation comme Normes Internationales par le Conseil de l'ISO.

La Norme Internationale ISO 4 (précédemment projet N° 1278) a été établie par le Comité Technique ISO/TC 46, *Documentation*.

Elle fut approuvée en juillet 1971 par les Comités Membres des pays suivants :

Afrique du Sud, Rép. d'	Irlande	Royaume-Uni
Allemagne	Israël	Suède
Belgique	Italie	Suisse
Danemark	Japon	Tchécoslovaquie
France	Nouvelle-Zélande	Thaïlande
Hongrie	Pays-Bas	U.S.A.
Inde	Pologne	
Iran	Roumanie	

Le Comité Membre du pays suivant a désapprouvé le document pour des raisons techniques :

Portugal

Cette Norme Internationale annule et remplace la Recommandation ISO/R 4-1953.

# Documentation – Code international pour l'abréviation des titres de périodiques

## 1 OBJET

La présente Norme Internationale a pour objet d'aider à l'établissement des abréviations de titres de publications en série ou non. Elle indique la forme la plus courte autorisée.

Elle n'est pas destinée à servir de guide pour l'établissement des fiches de collectivités auteurs ou de titres.

## 2 DOMAINE D'APPLICATION

2.1 Les règles de la présente Norme Internationale s'appliquent à l'établissement des abréviations de titres de publications en série ou non et éventuellement aux fiches de titres ou de collectivités auteurs utilisées dans les catalogues de bibliothèques.

EXEMPLE :

Titre complet : Bulletin of the Canadian Geological Society

Abréviation apparaissant sur le document :

Bull. Can. Geol. Soc.

Abréviation apparaissant dans les catalogues de bibliothèques :

Can. Geol. Soc., Bull

2.2 Ces règles s'appliquent également à l'abréviation des titres de publications non périodiques, y compris les monographies et les comptes rendus de réunion. Elles peuvent être appliquées, éventuellement, aux collectivités auteurs et aux comptes rendus de conférences pris isolément.

## 3 DÉFINITIONS

3.1 **sigle** : Mot formé de la lettre initiale ou des lettres initiales de chacune des parties successives ou parties principales d'un terme composé.

3.2 **mot composé** : terme dont les éléments constitutifs sont eux-mêmes des mots, des formes combinées ou des affixes.

3.3 **contraction** : Abréviation d'un mot, d'une syllabe ou d'un groupe de mots, obtenue par la suppression d'une ou de plusieurs lettres à l'intérieur du mot.

3.4 **collectivité-auteur** : Organisme ou groupe de personnes responsables du contenu intellectuel ou artistique d'un ouvrage.

3.5 **nom de la collectivité** : Nom de l'organisme qui désigne l'auteur collectif.

3.6 **mot (ou terme) générique** : Mot ou expression décrivant un type de publication, par exemple : Rapport Annuel, Journal, Compte rendu, etc., ou un type d'organisme par exemple : Faculté de Médecine, Institut, Société, etc.

3.7 **monographie** : Traité systématique et complet sur un sujet particulier, comprenant les volumes publiés périodiquement par une université ou une société; un volume peut contenir plusieurs monographies.

3.8 **périodique** : Publication en série paraissant, ou destinée à paraître indéfiniment, à intervalles réguliers ou déterminés, généralement plusieurs fois par an, et dont chaque livraison comprend des articles distincts, des récits ou autres écrits; les journaux qui relatent l'actualité en général, ainsi que les comptes rendus, documents ou autres publications éditées par des collectivités relatives à leurs réunions ne sont pas compris dans ce terme.

3.9 **titre secondaire** : Expression ajoutée au titre principal pour distinguer une partie séparée. Le paragraphe devra être désigné par un chiffre ou une lettre complémentaire.

3.10 **publication en série** : Publication en livraisons successives portant une indication numérique ou chronologique et destinée à paraître indéfiniment. Ces publications en série comprennent les périodiques, journaux, publications annuelles telles que rapports, annuaires, mémoires, comptes rendus, actes, etc., publications de sociétés et collections numérotées.

3.11 **sous-titre** : Phrase explicative ajoutée au titre principal.

3.12 **titre** : Utilisé ici, le mot «titre», désigne le nom de la publication à abrégé, qu'elle soit périodique ou non. L'appellation à considérer est celle qui apparaît sur la couverture, sur la page de titre, au titre courant et au dos de la publication; on peut aussi considérer le nom figurant dans les catalogues de bibliothèques, c'est-à-dire le ou les mots de titres précédés ou suivis du nom de la collectivité-auteur responsable de la publication.

3.13 **troncature** : Abréviation d'un mot par suppression de lettres finales.

## 4 RÈGLES

### 4.1 Abréviations des mots

#### 4.1.1 Méthode

4.1.1.1 La méthode d'abréviation recommandée est la troncature, c'est-à-dire l'abréviation par suppression d'u

groupe continu de lettres (au moins deux) placées à la fin d'un mot. Les mots formés d'une seule syllabe ou de cinq lettres au plus ne doivent pas être abrégés, à moins qu'il ne s'agisse d'articles, de conjonctions, de prépositions ou de termes génériques, fréquemment utilisés et mentionnés dans ISO/R 833.

**4.1.1.2** L'abréviation par contraction, c'est-à-dire par suppression de lettres à l'intérieur d'un mot, est limitée aux mots mentionnés dans ISO/R 833.

**4.1.1.3** Un mot ne doit pas être abrégé en une seule lettre initiale à moins qu'une telle abréviation ne figure dans ISO/R 833.

#### 4.1.2 *Ordre des mots*

L'ordre des mots abrégés doit, sans exception, suivre l'ordre des mots tels qu'ils apparaissent sur la fiche titre choisie, avec toutefois les exceptions suivantes :

**4.1.2.1** Les titres et les noms de collectivités longs peuvent être abrégés, pourvu que le titre reste identifiable et que sa place dans une liste alphabétique complète ne s'en trouve pas modifiée.

**4.1.2.2** Les sous-titres sont supprimés.

#### 4.1.3 *Titres formés d'un seul mot*

Les titres formés d'un seul mot, articles non compris, ne sont jamais abrégés (les règles permettant de différencier les titres identiques formés d'un seul mot sont indiquées en 4.2.2).

#### 4.1.4 *Articles, conjonctions et prépositions*

Les articles, conjonctions et prépositions, sont généralement supprimés dans l'abréviation du titre, à moins que leur maintien sous une forme abrégée suivant l'usage n'aide à l'identification du titre abrégé.

Archiv für Philosophie — Arch f Phil  
Archives of Philosophy — Arch of Phil

**4.1.4.1** Lorsque le premier mot du titre est une préposition, cette dernière est maintenue dans l'abréviation :

EXEMPLE :

Correct : Vom Wasser

Incorrect : Wasser

**4.1.4.2** L'emploi du caractère «perluète» (&) pour «et», ou son équivalent dans d'autres langues, peut rendre plus claires les abréviations de titres.

EXEMPLE :

Titres complets : Journal of Mathematics and Physics  
Journal of Mathematical Physics

Titres abrégés : J. Math. & Phys.  
J. Math. Phys.

#### 4.1.5 *Utilisation des lettres majuscules*

La première lettre du premier élément de l'abréviation doit être une majuscule. Pour le reste de l'abréviation, on utilise des majuscules pour toutes les lettres ou pour la première lettre de chaque élément ou encore des minuscules, suivant l'usage.

EXEMPLE :

Correct :

Archives of Internal Medicine — Arch Intern Med  
— ARCH INTERN MED  
Archiv für klinische Medizin — Arch klin Med  
Archives de Médecine navale — Arch Med nav  
Archivos de medicina interna — Arch med intern

Incorrect :

Archivos de medicina interna — arch med intern

#### 4.1.6 *Ponctuation*

##### 4.1.6.1 *Point*

Un point et un espace, ou un espace seul, peuvent être utilisés entre chaque mot abrégé d'un titre. A moins que l'utilisation du point ne soit complètement abandonnée, celui-ci est nécessaire après chaque abréviation, même si la lettre finale du mot est maintenue.

EXEMPLES :

Canadian Pharmacy Journal

Correct : Can. pharm. j.  
Can Pharm J  
Can pharm j  
CAN. PHARM. J.  
CAN PHARM J

Deutsche Schwesternzeitung

Correct : Dtsch. Schwesternztg.  
Dtsch Schwesternztg  
Incorrect : Dtsch. Schwesternztg

##### 4.1.6.2 *Virgule*

**4.1.6.2.1** Dans l'abréviation d'une fiche collectivité-auteur, une virgule doit être utilisée pour séparer la collectivité-auteur principale de la collectivité secondaire et le ou les mots génériques du titre de la collectivité-auteur. Si l'ordre des mots abrégés suit l'ordre des mots du titre original, les virgules ne sont pas utilisées.

**4.1.6.2.2** Une virgule doit être utilisée pour séparer l'abréviation du titre principal de l'abréviation d'une section, d'une série ou d'un titre secondaire (voir également 4.2.4).

EXEMPLE :

Titre complet : Memoirs of the Faculty of Science,  
Kyushu University, Series D.

Titre abrégé : Mem. Fac. Sci., Kyushu Univ., Ser. D.

#### 4.1.6.3 Autres signes de ponctuation

D'autres signes de ponctuation peuvent être utilisés pour plus de clarté.

#### 4.1.7 Signes diacritiques

? Les signes diacritiques peuvent être supprimés dans les abréviations de titres. Lorsqu'ils sont supprimés, l'orthographe ne doit pas être modifiée pour indiquer leur présence dans le titre original. Si, cependant, on décide de modifier l'orthographe des mots contenant des signes diacritiques, cette pratique doit être clairement indiquée aux utilisateurs des abréviations en question.

#### 4.1.8 Symboles

Les symboles apparaissant dans le titre original doivent être maintenus tels quels dans l'abréviation du titre.

EXEMPLE :

Titre complet : Metall-Reinigung + Vorbehandlung

Titre abrégé : Metall-Reinig. + Vorbehandl.

#### 4.1.9 Pluriels

L'abréviation du singulier d'un mot est également utilisée pour la forme du pluriel, sauf lorsque le pluriel doit être indiqué afin d'éviter une mauvaise interprétation d'une abréviation.

Dans ce cas, la dernière lettre du pluriel du mot est ajoutée à l'abréviation de la forme du singulier.

Les terminaisons du pluriel sont maintenues dans les mots abrégés.

#### 4.1.10 Mot composé

N'abrégé que le dernier élément d'un mot composé, à moins qu'il n'y ait un trait d'union; dans ce cas, abrégé chaque élément selon ISO/R 833, en retenant le trait d'union et en ne laissant pas d'espace après le ou les points à l'intérieur du mot composé.

EXEMPLE : Mot composé ne comprenant pas de trait d'union

Nachrichtenblatt

Correct : Nachrichtenbl.

Incorrect : Nachr.-bl.

EXEMPLE : Mot composé avec trait d'union

Technisch-industrielle Rundschau

Correct : Tech.-ind. Rundsch.

Incorrect : Tech.Ind. Rundsch.

Technisch-ind. Rundsch.

Techind. Rundsch.

## 4.2 Précisions

### 4.2.1 Abréviations de mots

4.2.1.1 La même abréviation ne doit pas être utilisée pour des mots qui n'ont aucun rapport entre eux.

EXEMPLE : Ind.

est correct pour : Industrie ou Industriel, mais

est incorrect pour : Indes, Indiana, Indigency ou Indigo,

sauf lorsqu'une même abréviation, dans l'usage courant, représente deux mots sans liaison et que ces mots font partie d'un nom de personne ou de lieu.

EXEMPLES :

Titres complets : Wall Street Journal

Saint Louis Quarterly

Titres abrégés : Wall St. J.

St. Louis Q.

4.2.1.2 On ne doit pas employer plusieurs abréviations différentes pour représenter le même mot.

EXEMPLE : International

Abréviation correcte : Int.

Abréviation incorrecte : Intern.

Int'l

### 4.2.2 Abréviations de titres

4.2.2.1 On doit distinguer les abréviations de titres identiques ou expliciter les abréviations courtes et ambiguës en indiquant entre parenthèses, après l'abréviation du titre, le lieu de publication abrégé. Le lieu de publication peut être, selon le cas, un pays, une subdivision politique d'un pays (par exemple : état, département, canton, commune) ou une ville. On doit donner la préférence à la subdivision la plus importante. Dans le cas où une même abréviation apparaît pour des titres de périodiques dont la publication se rattache à la même subdivision, on doit indiquer un lieu de publication plus caractéristique.

EXEMPLE :

Annales de Physique

Correct : Ann. Phys. (Fr.)

Incorrect : Ann. Phys.

Annals of Physics

Correct : Ann. Phys. (US)

4.2.2.2 On doit utiliser le nom de l'organisme responsable de la publication, abrégé selon les règles de la présente Norme Internationale, plutôt que le nom de lieu, comme identification supplémentaire si ce dernier ne permet pas de distinguer entre des abréviations contradictoires ou si le nom de l'organisme semble faciliter l'identification.

**4.2.2.3** On ne doit pas différencier les abréviations de titres identiques en utilisant ou non les lettres majuscules et la ponctuation, ou en laissant certains titres ou mots non abrégés lorsque des abréviations pour ces mots sont prévues dans ISO/R 833.

#### 4.2.3 Sigle, groupe d'initiales, groupe de lettres

Un sigle ou un groupe d'initiales ou de lettres dans un titre doivent être conservés intacts dans le titre sous leur forme abrégée et doivent toujours être représentés en majuscules. Lorsqu'un sigle ou un groupe de ce type représente le nom d'un organisme et qu'il est nécessaire d'expliquer ou de distinguer l'abréviation du titre, on doit rétablir la signification du sigle ou du groupe, l'abrégé selon les règles de la présente Norme Internationale et l'ajouter au titre entre parenthèses comme il est indiqué en 4.2.2.2.

EXEMPLE :

AEG Mitteilungen (Allgemeine Elektrizitäts-Gesellschaft)

Correct : AEG Mitt. (Allg. Elek. Ges.)

Incorrect : AEG (Allg. Elek. Ges.) Mitt.

La règle ci-dessus peut ne pas être appliquée lorsque les noms des organismes sont couramment utilisés sous leur forme de sigles, de groupe d'initiales ou de lettres.

#### 4.2.4 Sections et séries

Lorsqu'une publication paraît en plusieurs sections ou séries différenciées par un nom, un numéro ou une lettre, il faut introduire cette indication dans l'abréviation. Les abréviations des termes génériques tels que Partie, Section, Série, etc. n'ont pas à être retenues dans l'abréviation du titre si elles ne sont pas nécessaires à l'identification de ce titre.

EXEMPLES :

Annales Scientifiques de l'Université de Besançon, Géologie

Correct : Ann. Sci. Univ. Besançon, Géol.

Incorrect : Ann. Sci. Univ. Besançon

Annales Scientifiques de l'Université de Besançon, Physique

Correct : Ann. Sci. Univ. Besançon, Phys.

Incorrect : Ann. Sci. Univ. Besançon.

Journal of Botany, Section A

Correct : J. Bot., A

J. Bot., Sect. A

Incorrect : J. Bot.

Journal of Botany, Section B

Correct : J. Bot., B

J. Bot., Sect. B

Incorrect : J. Bot.

Il n'est pas nécessaire d'abrégé un titre secondaire en plus du titre principal lorsque la section en question est différenciée par un numéro ou une lettre.

EXEMPLE :

Titre complet : Journal of Polymer Science, Part A-1  
Polymer Chemistry

Titre abrégé : J. Polym. Sci., A-1

#### 4.2.5 Editions différentes

Lorsqu'une publication paraît en plusieurs éditions, chacune avec un titre abrégé identique, ceci est précisé entre parenthèses, après l'abréviation du titre, à l'aide d'une expression appropriée et abrégée selon les règles de la présente Norme Internationale.

EXEMPLE :

Impact, Science et Société (French Edition)

Impact of Science on Society (English Edition)

Correct : Impact Sci. Soc. (Fr. Ed.)

Impact Sci Soc (Engl Ed)

Incorrect : Impact Sci Soc

Impact Sci. Soc.

#### 4.2.6 Noms propres dans les titres

Lorsque le titre d'une publication contient un nom de personne, ce nom doit être maintenu dans l'abréviation et tous les autres mots du titre doivent être abrégés selon les règles de la présente Norme Internationale. L'utilisation du nom de famille et d'un terme générique dans l'abréviation du titre n'est pas suffisante; l'abréviation doit être faite sur l'ensemble du titre.

EXEMPLE :

Robert A. Welch Foundation Research Bulletin

Correct : Robert A. Welch Found. Res. Bull.

Incorrect : Welch Found. Res. Bull.

Found. Res. Bull.

#### 4.3 Translittération

Les titres écrits dans des alphabets autres que l'alphabet latin doivent être translittérés en caractères latins conformément aux documents énumérés ci-dessous, puis abrégés conformément aux règles de la présente Norme Internationale.

#### RÉFÉRENCES

1) ISO/R 9, *Système international pour la translittération des caractères cyrilliques slaves.*

2) ISO/R 233, *Système international pour la translittération des caractères arabes.*

3) ISO/R 259, *Translittération de l'hébreu.*

4) ISO/R 833, *Abréviations de noms génériques dans les titres de périodiques.* (Revision en préparation.)

5) ISO/R 843, *Système international pour la translittération des caractères grecs en caractères latins.*

6) ANSI/Z 39 : Centre National des Abréviations de Mots Contenus dans les Titres de Périodiques, (NCPTWA), *Liste des abréviations de mots.*





**ANNEXE A7**

**LA STRUCTURE DE LA BASE D'ABREGE**

\* SUCRATE STARTED \*\* V1.6 LE 107.76 A 23.21.13

```
1 DEBUT
2
3 /* ELEMENTS VARIABLES */
4
5
6
7 /* LIBELLES ET MOTS DES JEUX D'ESSAI */
8
9 ENTITE 2000 LIBELLE /* LIBELLES */
10 DEBUT
11 ENTITE 6 TXT
12 DEBUT
13 LIB-CONT MOT (30)
14 FIN
15 LIB-LONG DE 1 A 180
16 NB-MOTS DE 1 A 40
17 /* LISTE NON REDONCANTE DES MOTS DU TEXTE */
18 ENTITE 40 REF-MOTS
19 DEBUT
20 NO-MOTS DE 1 A 5000
21 OCC-LIB DE 1 A 40
22 FIN
23 REF-ABREV REFERE ABR-SYN DE UNE ABREVIATION
24 FIN
25
26 ENTITE 5000 MOTS-LIB /* MOTS DES JEUX D'ESSAI */
27 DEBUT
28 MOT-CONT MOT (30) AVEC CLE UNIQUE FIN
29 FREQ-GEN DE 1 A 1000 /*NBRE OCCURENCES */
30 FREQ-LITT DE 0 A 100 /* POURCENTAGE */
31 /* FREQUENCE LANGAGE */
32 ENTITE 26 DEC-LETTRE
33 DEBUT
34 FREQ-LET DE 1 A 10 /* NBRE OCCURENCES */
35 FIN
36 /* REFERENCE VERS DES ABREV-NON-CALCULEES */
37 REF-MOT-CIC DE 1 A 2000
38 LIST-ABR INVERSE TOUT ABRELIST DE UN MOTS-DIC
39 /* REFERENCES VERS DES DECOMPOSITIONS */
40 /* MEME NUM POUR SYLL OU PHON */
41 NO-DECPHON DE 1 A 5000
42 FIN
43
44
45 /* DECOMPOSITIONS */
46
47 ENTITE 5000 DEC-PHON /* DECOMPOSITION PHONETIQUE */
48 DEBUT
49 ENTITE 20 SLIT-PHON
50 DEBUT
51 NO-PHON DE 1 A 30
52 NB-LTR DE 1 A 5
53 FREQ-PHON DE 1 A 10 /* NBRE OCCURENCES DU PHON */
54 FIN
55 FIN
56
57 ENTITE 5000 DEC-SYL /* CAR. DE DECOMPOSITION SYLLABIQUE */
```

```
58 DEBUT
59 ENTITE 15 SUIT-SYL
60 DEBUT
61 NB-PT DE 1 A 5
62 NB-LTR DE 1 A 10
63 FIN
64 FIN /* FIN DEC-SYL */
65
66
67 /* ABREVIATIONS CALCULEES */
68
69 ENTITE 2000 ABREVIATION /* UN LIBELLE DONNE UN MOT */
70 DEBUT
71 ABR-SYN ANNEAU
72 ABR-CONTFNU MOT (30) AVEC CLE UNIQUE FIN
73 FIN
74
75 /* DICTIONNAIRES DES ABREVIATIONS NON CALCULEES */
76
77
78 ENTITE 10 CAR-DICO /* NOM DICTIONNAIRE */
79 DEBUT
80 NOM-DIC MOT (10)
81 ANTIDIC MOT (1) /* A N */
82 FIN
83
84 ENTITE 2000 MOTS-DIC /* MOTS ET ABR. DES DICO */
85 DEBUT
86 MCT-CONT MOT (30) AVEC CLE UNIQUE FIN
87 /* ABREVIATIONS */
88 ENTITE 10 ABRELIST
89 DEBUT
90 ABRE-CON MOT (10)
91 FIN
92 FIN
93
94
95 /* METHODES D'ABREVIATION */
96
97 ENTITE 11 DESCR-METHODE
98 DEBUT
99 /* DICO ET ANTIDIC UTILISES DANS LA METHODE */
100 ENTITE 10 DIC-METH
101 DEBUT
102 IDENT-DIC MOT (10)
103 FIN
104 CONCATENATION MOT (1) /* T E C */
105 DECOMPOSITION MOT (1) /* S P L */
106 DOUBLET MOT (1)
107 LIMITE DE 1 A 30 /* U SI PAS DE LIMITE */
108 COLLISION DE 0 A 10
109
110 ENTITE 3 FONC1
111 DEBUT
112 IDENT-FONC MOT (10)
113 NUM-FONC DE 1 A 10
114 FIN
115
116 ENTITE 5 ANTIDIC
117 DEBUT
```

178 DEBUT  
179 UNITE MOT (1) /\* P L \*/  
180  
181 ENTITE 2 BL3  
182 DEBUT  
183 BORNE-NUM DE 1 A 8  
184 FIN  
185  
186 ENTITE 2 BP2  
187 DEBUT  
188 BORNE-LITT MOT (2) /\* L/ L- L \*/  
189 BORNE-NUM DE 1 A 30  
190 FIN  
191  
192 TYPE MOT (3) /\* CON VOY LAB GUT DEN MUT \*/  
193 NAT-FREQ MOT (1) /\* G R L M \*/  
194 FREQ-CP MOT (1)  
195  
196 ENTITE 2 BF2  
197 DEBUT  
198 BORNE-LITT MOT (2) /\* U % %N %X \*/  
199 BORNE-NUM DE 0 A 100  
200 FIN  
201  
202 ENTITE 3 COMPTEUR2  
203 DEBUT  
204 NUM-COMPT DE 1 A 10  
205  
206 ENTITE 2 BC2  
207 DEBUT  
208 BORNE-LITT MOT (20)  
209 BORNE-NUM DE 0 A 100  
210 FIN  
211  
212 FIN  
213  
214  
215 ENTITE 4 CHOIX  
216 DEBUT  
217 VERBE MOT (1) /\* T = TROUVER P = PRENDRE \*/  
218 FIXE MOT (1) /\* F OJ U \*/  
219 NOMBRE-LITT MOT (2) /\* L- L/ L \*/  
220 NOMBRE-NUM DE 1 A 30  
221 DEPUIS-LITT MOT (2) /\* L- L/ L \*/  
222 DEPUIS-NUM DE 1 A 30  
223  
224 ENTITE 3 MOD-COMPT1  
225 DEBUT  
226 OPER MOT (1) /\* A Z D \*/  
227 NUM-COMPT DE 1 A 10  
228 FIN  
229  
230 TYPE MOT (3) /\* CUN... + SYL PHO LET \*/  
231 CHAIN MOT (5) /\*XXXXX \*/  
232 FIN  
233  
234 ENTITE 3 MOD-COMPT2  
235 DEBUT  
236 OPER MOT (1) /\* A Z D \*/  
237 NUM-COMPT DE 1 A 10

118 IDENT-DIC MOT (10)  
119 NUM-DIC DE 1 A 10  
120 FIN  
121  
122  
123 ENTITE 10 FILTRE-LIB  
124 DEBUT  
125  
126 ENTITE 2 BL1  
127 DEBUT  
128 BORNE-NUM DE 1 A 32  
129 FIN  
130  
131  
132 ENTITE 15 FILTRE-MOT  
133 DEBUT  
134  
135 ENTITE 2 BP1  
136 DEBUT  
137 BORNE-LITT MOT (2) /\* L- L/ L \*/  
138 BORNE-NUM DE 1 A 32  
139 FIN  
140  
141 NAT-FREQ MOT (1) /\* G R L \*/  
142 FREQ-OP MOT (1) /\* N X \*/ /\* \*MIN OU \*MAX \*/  
143  
144 ENTITE 2 BF1  
145 DEBUT  
146 BORNE-LITT MOT (2) /\* U % %N %X \*/  
147 BORNE-NUM DE 0 A 100  
148 FIN  
149  
150 UNITE MOT (1) /\* S P L \*/  
151  
152 ENTITE 2 BL2  
153 DEBUT  
154 BORNE-NUM DE 1 A 30  
155 FIN  
156  
157 ENTITE 3 COMPTEUR1  
158 DEBUT  
159 NUM-COMPT DE 1 A 10  
160  
161 ENTITE 2 BC1  
162 DEBUT  
163 BORNE-LITT MOT (2) /\* L/ L- L \*/  
164 BORNE-NUM DE 0 A 100  
165 FIN  
166  
167 FIN  
168  
169 ENTITE 10 DICOS  
170 DEBUT  
171 IDENT-DIC MOT (10)  
172 NUM-DIC DE 1 A 10  
173 FIN  
174 TYP-CALCUL MOT (1) /\* S P L \*/  
175  
176  
177 ENTITE 20 CALCUL

238 FIN  
239  
240 ENTITE 3 FONC2  
241 DEBUT  
242 IDENT-FONC MOT (10)  
243 NUM-FONC DE 1 A 10  
244 FIN  
245  
246 BOUT MOT (1) /\* B \*/  
247  
248 FIN /\* CALCUL \*/  
249  
250  
251 FIN /\* FILTRE-MOT \*/  
252  
253  
254 FIN /\* FILTRE-LIB \*/  
255  
256  
257 FIN /\* METHODE \*/  
258  
259  
260  
261 /\* ELEMENTS FIXES \*/  
262  
263  
264  
265 /\* CARACTERISTIQUES DE L'ALPHABET \*/  
266  
267 ENTITE 28 ALPHA  
268 DEBUT  
269 LETTRE MOT (1) AVEC CLE UNIQUE ORDONNE FIN  
270 TYPE (2 1) (V C)  
271 PRONCNC (5 2) (SR SC RA MU)  
272 CODE-TAB-PHON DE 1 A 30  
273 FR-D-R DE 1 A 10000 /\* FREQUENCE LINGUISTIQUE \*/  
274 FR-G-R DE 1 A 10000 /\* FREQUENCE RELATIVE AU JEU \*/  
275 FIN  
276  
277  
278 /\* TABLE DES PHONEMES \*/  
279  
280 ENTITE 30 T-PHON  
281 DEBUT  
282 N-PH MOT (2)  
283 FREC-PHON DE 1 A 10000  
284 TYP-PHON MOT (3)  
285 CODE-TAB-SYL DE 2 A 4  
286 /\* VOYELLE CONSONNE SOURDE OU EXPLOSIVE \*/  
287 FIN  
288  
289  
290 /\* INFORMATIONS POUR AUTCMATE PHONETIQUE \*/  
291  
292 ENTITE 50 PH-ETAT /\* ETATS PHONETIQUES \*/  
293 DEBUT  
294 PH-CHAINE MOT (3)  
295 ENTITE 24 LET-CONF  
296 DEBUT  
297 MEM-CONF DE 1 A 100

```
298     FIN
299     FIN
300
301     ENTITE 100 PH-CONF /* CONFIGURATIONS PHONETIQUES */
302     DEBUT
303         NV-PH-ETAT MOT (3)
304         NV-PHON DE 1 A 30
305         PH-ACT (5 2) (L1 R1 R2 NE S0)
306     FIN
307
308     ENTITE 1 ANALYSEUR /* ANALYSEUR DE L'AUTGMATE */
309     DEBUT
310         MOTBASE MOT (30)
311         INDEX DE 1 A 30
312         CARCCUR MOT (1)
313         ETAT-COUR MOT (3)
314         DEBPHON DE 1 A 30
315     FIN
316
317
318     /* INFORMATIONS POUR AUTOMATE SYLLABIQUE */
319
320     ENTITE 5 SYL-ETAT /* TABLE ETATS SYLLABIQUES */
321     DEBUT
322         CH-PH MOT (3)
323         ENTITE 4 PH-COUR
324         DEBUT
325             NV-SYL-ETAT MOT (3)
326             NV-SYL (2 3) (NUL SY)
327             SYL-ACT (2 2) (NE L1)
328         FIN
329     FIN
330
331
332     /* INFORMATIONS POUR ANALYSE SYNTAXIQUE ET SEMANTIQUE */
333
334     ENTITE 15 AUTOMATE
335     DEBUT
336         ENTITE 30 ENTREE /* LISTE MOTS CLE */
337         DEBUT
338             MOTCLE MOT (10)
339             NUM-COL DE 1 A 50
340         FIN
341         ENTITE 60 ETATS /* TABLE D'ETATS */
342         DEBUT
343             ETAT MOT (3)
344             ENTITE 30 CONF
345             DEBUT
346                 NUM-CONF DE 1 A 999
347             FIN
348         FIN
349     FIN
350
351     ENTITE 999 CONFIGURATION
352     DEBUT
353         ERREUR-SEQ DE 0 A 1
354         NV-ETAT MOT (3)
355         NUM-LIG DE 1 A 60 /* NUMERO ETAT */
356     ENTITE 7 ACTION
357     DEBUT
```

418 S-25 MOT (30)  
419 S-26 MOT (30)  
420 S-27 MOT (30)  
421 FIN  
422  
423  
424  
425 ENTITE 1 XEC  
426 DEBUT  
427 MC DE 1 A 10 /\* DESCR-METHODE COURANT \*/  
428 FLC DE 1 A 10 /\* FILTRE-LIB COURANT \*/  
429 FMC DE 1 A 15 /\* FILTRE-MOT COURANT \*/  
430 CC DE 1 A 20 /\* CALCUL COURANT \*/  
431 HC DE 1 A 4 /\* CHOIX COURANT \*/  
432 CD DE 1 A 32 /\* NUM. CAR. DEPART POUR CHOIX \*/  
433 POSG DE 1 A 32 /\* POINTEUR CHGT DE SENS \*/  
434 LGMOT DE 1 A 30 /\* LONGUEUR DU MOT EN CAR. \*/  
435 SENS MOT (2) /\* GD,DG \*/  
436 L DE 1 A 30 /\* LONGUEUR ELT COURANT \*/  
437 N-M-REF DE 1 A 5000 /\* NO MOT COURANT \*/  
438 NMC DE 1 A 32 /\* M-REF COURANT \*/  
439 CCNC MOT (30)  
440 FLMIN DE 0 A 100 /\* LA + PETITE FREQ-LITT \*/  
441 FLMAX DE 0 A 100 /\* LA + GRDE FREQ-LITT \*/  
442 FGMIN DE 1 A 1000 /\* LA + PETITE FREQ-GEN \*/  
443 FGMAX DE 1 A 1000 /\* LA + GRDE FREQ-GEN \*/  
444 ENTITE 1 TABR /\* CARACTERES RETENUS \*/  
445 DEBUT  
446 ENTITE 30 CAR-RETENU  
447 DEBUT  
448 CARACTERE MOT (1)  
449 FIN  
450 FIN  
451 ENTITE 40 ABRELEM /\* MOTS ABREGES \*/  
452 DEBUT  
453 LG-ABR DE 0 A 20 /\* LGEUR CAR DE L'ABREVIATION \*/  
454 AMOT MOT (20)  
455 FIN  
456 ENTITE 10 COMPTEUR  
457 DEBUT  
458 VALC DE 0 A 100  
459 FIN  
460 FIN /\* XEC \*/  
461 ENTITE 1 LIB  
462 DEBUT  
463 ENTITE 32 M-REF  
464 DEBUT  
465 N-MOT DE 1 A 5000  
466 FIN  
467 REF-ABR DE 1 A 5000  
468 NUM-LIB DE 1 A 2000  
469 LONG-LIB DE 1 A 32  
470 FIN /\*LIB \*/  
471  
472  
473 ENTITE 100 RESERVE  
474 DEBUT  
475 MOT-RESERVE MCT (30)  
476 FIN  
477



358                   CODE-TRAIT MOT (1)  
359                   TRAIT MOT (9)  
360                   FIN  
361                   FIN  
362  
363  
364                   /\* ENTITE DE STOCKAGE D'UN TEXTE \*/  
365  
366                   ENTITE 6 ENTREE-TEXTE  
367                   DEBUT  
368                   CONTENU MOT (30)  
369                   FIN  
370  
371  
372                   /\* ENTITE DE STOCKAGE D'UNE CARTE \*/  
373  
374                   ENTITE 1 CARTE  
375                   DEBUT  
376                   ENTITE 72 SIGNE  
377                   DEBUT  
378                   CARACTERE MOT (1)  
379                   FIN  
380                   FIN  
381  
382  
383                   /\* ENTITE DE SAUVEGARDE \*/  
384  
385                   ENTITE 1 SAUVE  
386                   DEBUT  
387                   S-AUTOMATE REFERE UN AUTOMATE  
388                   S-CONFIGURATION REFERE UN CONFIGURATION  
389                   S-Y1 DE -100000 A +100000  
390                   S-Y2 DE -100000 A +100000  
391                   S-Y3 DE -100000 A +100000  
392                   S-Y4 DE -100000 A +100000  
393                   S-Y5 DE -100000 A +100000  
394                   S-Y6 DE -100000 A +100000  
395                   S-Y7 DE -100000 A +100000  
396                   S-Y8 DE -100000 A +100000  
397                   S-Y9 DE -100000 A +100000  
398                   S-Y10 DE -100000 A 100000  
399                   S-Y11 DE -100000 A 100000  
400                   S-Y12 DE -100000 A 100000  
401                   S-Y13 DE -100000 A 100000  
402                   S-Y14 DE -100000 A 100000  
403                   S-Y15 DE -100000 A 100000  
404                   S-Y16 DE -100000 A 100000  
405                   S-Y17 DE -100000 A 100000  
406                   S-Y18 DE -100000 A 100000  
407                   S-Y19 DE -100000 A 100000  
408                   S-Y20 DE -100000 A 100000  
409                   S-Y21 DE -100000 A +100000  
410                   S-Y22 DE -100000 A +100000  
411                   S-Y23 DE -100000 A +100000  
412                   S-Y24 DE -100000 A +100000  
413                   S-Y25 DE -100000 A +100000  
414                   S-Z1 MOT (30)  
415                   S-Z2 MOT (30)  
416                   S-Z3 MOT (30)  
417                   S-Z4 MOT (30)

538 F-CODE-TAB-SYL DILATE (1)  
539 F-T-FIL2 MOT (1)  
540 FIN  
541 FIN  
542  
543 FORMAL F-PH-ETAT /\* ETATS PHONETIQUES \*/  
544 DEBUT  
545 F-PH-CHAINE MOT (3)  
546 FORMAL 24 F-LET-CONF  
547 DEBUT  
548 F-PH-FIL1 MOT (1)  
549 F-MEM-CONF DILATE (2)  
550 FIN  
551 F-PH-FIL2 MOT (5)  
552 FIN  
553  
554 FORMAL F-CONFI /\* CONFIGURATIONS PHONETIQUES \*/  
555 DEBUT  
556 FORMAL 8 F-PH-CONF  
557 DEBUT  
558 F-NV-PH-ETAT MOT (3)  
559 F-NV-FIL1 MOT (1)  
560 F-NV-PHON DILATE (2)  
561 F-NV-FIL2 MOT (1)  
562 F-PH-ACT MOT (2)  
563 F-NV-FIL3 MOT (1)  
564 FIN  
565 FIN  
566  
567 FORMAL F-SYL-ETAT /\* ETATS SYLLABIQUES \*/  
568 DEBUT  
569 F-CH-PH MOT (3)  
570 FORMAL 4 F-PH-COUR  
571 DEBUT  
572 F-SYL-FIL1 MOT (1)  
573 F-NV-SYL-ETAT MOT (3)  
574 F-SYL-FIL2 MOT (1)  
575 F-NV-SYL MOT (3)  
576 F-SYL-FIL3 MOT (1)  
577 F-SYL-ACT MOT (2)  
578 FIN  
579 F-SYL-FIL4 MOT (33)  
580 FIN  
581  
582 FORMAL F-MOT-CLE /\* MOTS CLES ANALYSE \*/  
583 DEBUT  
584 FORMAL 5 F-ENTREE  
585 DEBUT  
586 F-MOTCLE MOT (10)  
587 F-MOT-FIL1 MOT (2)  
588 F-NUM-COL DILATE (2)  
589 F-MOT-FIL2 MOT (2)  
590 FIN  
591 FIN  
592  
593 FORMAL F-ETATS /\* ETATS ANALYSE \*/  
594 DEBUT  
595 F-ETAT MOT (3)  
596 F-ETAT-FIL1 MOT (1)  
597 FORMAL 19 F-CONF

/\* ENSEMBLE DES FORMALS \*/

478  
479  
480  
481  
482  
483  
484  
485  
486  
487  
488  
489  
490  
491  
492  
493  
494  
495  
496  
497  
498  
499  
500  
501  
502  
503  
504  
505  
506  
507  
508  
509  
510  
511  
512  
513  
514  
515  
516  
517  
518  
519  
520  
521  
522  
523  
524  
525  
526  
527  
528  
529  
530  
531  
532  
533  
534  
535  
536  
537

/\* DESCRIPTION D'UNE CARTE \*/

FORMAL F-CARTE  
DEBUT  
FORMAL 72 F-SIGNE  
DEBUT  
F-CARACTERE MOT (1)  
FIN  
F-SEQUENCE MOT (8)  
FIN

/\* DESCRIPTION D'UN ENR. BANDE \*/

FORMAL BANDE-SCRTIE  
DEBUT  
INFORMATION MOT (30)  
CODE-INF-SUIV MOT (2)  
FIN

/\* ENSEMBLE DES FORMALS POUR LE CHARGEMENT \*/  
/\* BATCH DES ELEMENTS FIXES DE LA BASE \*/

FORMAL FCARTE-NUM  
DEBUT  
FORMAL 10 F-NOMBRE  
DEBUT  
F-NB DILATE (8)  
FIN  
FIN

FORMAL F-ALPHABET /\* CAR LETTRES \*/

DEBUT  
FORMAL 5 F-ALPHA  
DEBUT  
F-LETTRE MOT (1)  
F-AL-FIL1 MOT (1)  
F-TYPE MOT (1)  
F-AL-FIL2 MOT (1)  
F-PRCNONG MOT (2)  
F-AL-FIL3 MOT (1)  
F-CODE-TAB-PHON DILATE (2)  
F-AL-FIL4 MOT (1)  
F-FR-C-R DILATE (4)  
F-AL-FIL5 MOT (2)  
FIN

FIN

FORMAL F-PHONEME /\* TABLE PHONEMES \*/

DEBUT  
FORMAL 16 F-T-PHON  
DEBUT  
F-N-PH MOT (2)  
F-T-FIL1 MOT (1)



**ANNEXE A8**

**AUTOMATES DE RECONNAISSANCE DU LANGAGE  
DE DESCRIPTION DE METHODES**

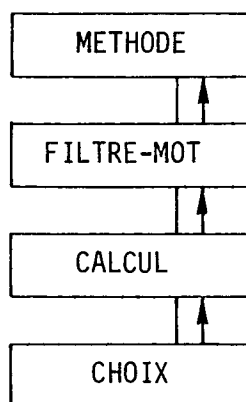
L'analyse syntaxique des méthodes se fait par enchaînement de quatre automates présentés ci-après.

Pour chaque *ligne de l'automate reflétant l'état* dans lequel il est, et pour chaque *colonne indiquant le mot-clé* courant qui vient d'être reconnu, on trouve la nouvelle configuration avec indication du nouvel état dans lequel passe l'automate après exécution de l'action liée à la situation.

Toutes les expressions un peut complexes de calcul sont interprétées dans les actions au niveau sémantique, donc pour une meilleure modularité de l'analyse.

Le rôle des actions est indiqué après chaque automate ; elles sont directement liées à la structure Socrate décrivant une méthode.

En tête, nous donnons la grammaire du langage de description d'une méthode au niveau syntaxique et sémantique, pour la compréhension des automates, ainsi que le schéma d'enchaînement de ceux-ci.



Méthode : \*\*METHODE numéro options antidicos<sup>0/1</sup> filtre-libellé<sup>+</sup> \*\*FMETHODE  
options : [\*TIRET | \*ESPACE]<sup>0/1</sup> [\*LONG ope nombre]<sup>0/1</sup> \*DOUBLET<sup>0/1</sup>  
[\*COLLISION [numéro | CHIFFRE]]<sup>0/1</sup> [\*FG identificateur]<sup>0/3</sup>.  
antidicos : \*ANTIDIC identificateur<sup>+</sup>.  
filtre-libellé : \*LIB [\*NB MOT longueur]<sup>0/1</sup> filtre-mot<sup>+</sup>.  
filtre-mot : \*MOT position<sup>0/1</sup> fréquence<sup>0/1</sup> taille<sup>0/1</sup> compteur<sup>0/1</sup>  
[dicos | dicos calculs | calcul]  
dicos : \*DIC identificateur<sup>+</sup>.  
calculs : filtre-syllabe<sup>+</sup> | filtre-phonème<sup>+</sup> | filtre-lettre<sup>+</sup>.  
position : \*POS [ope expression | DE expression A expression].  
expression : nombre | L/nombre | L-nombre | L.  
fréquence : Natfreq [ope valfreq | DE valfreq A valfreq] |  
[\*MIN | \*MAX] natfreq  
natfreq : \*FREQG | \*FREQR | \*FREQL | \*FREQM.  
valfreq : nombre [% | % MIN | % MAX]<sup>0/1</sup>.  
ope : > | ≥ | = | < | ≤.  
taille : unité longueur.  
unité : \*NBSYLL | \*NBPHON | \*NBLETR.  
longueur : [ope nombre | DE nombre A nombre].  
compteur : \*X nombre [ope expression | DE expression A expression].  
filtre-syllabe : \*SYLL taille<sup>0/1</sup> position<sup>0/1</sup> compteur\* actions<sup>0/1</sup>.  
filtre-phonème : \*PHON taille<sup>0/1</sup> position<sup>0/1</sup> type<sup>0/1</sup> fréquence<sup>0/1</sup> compteur<sup>0/1</sup>  
actions<sup>0/1</sup>.  
filtre-lettre : \*LETR position<sup>0/1</sup> type<sup>0/1</sup> fréquence<sup>0/1</sup> compteur<sup>0/1</sup> actions<sup>0/1</sup>.  
actions : choix\* modif-compt\* fonction\* \*BOUT<sup>0/1</sup>.  
fonction : \*F identificateur.  
modif-compt : [\*A | \*Z | \*D] X nombre.  
choix : [\*TROUVER | \*PRENDRE] \*FIXE<sup>0/1</sup> [\*N ≤ expression]<sup>0/1</sup>  
[DEPUIS expression]<sup>0/1</sup>  
modif-compt\* [type | \*SY | \*PH | \*LT | \*'xxxxx'\*]  
type : \*CONS | \*VOY | \*DENT | \*LAB | \*SOU | \*MUET | \*GUT | .....

M E T H O D E A U T O M A T E

	**METH	*TIRET / *ESPACE	*LONG	*DOUBLET	*COLLI-SION	*FG	*ANTIDIC	*LIB	*NBMOT	*MOT	**FMETH	autre	*NOM
TIR/ESP	PE: ES9	TIR/ESP ES1	LONG M2	DOUB M3	COLL M4	FG M5	ANTIDIC M6	LIB M7	LIB ES2	LIB ES2	FIN ES7	TIR/ESP ES8	TIR/ESP ES4
LONG	FIN ES9	LONG ES4	LONG ES1	DOUB M3	COLL M4	FG M5	ANTIDIC M6	LIB M7	LIB ES2	LIB ES2	FIN ES2	LONG ES8	LONG ES4
DOUB	FIN ES9	DOUB ES4	DOUB ES4	DOUB ES1	COLL M4	FG M5	ANTIDIC M6	LIB M7	LIB ES2	LIB ES2	FIN ES7	DOUB ES8	DOUB ES4
COLL	FIN ES9	COLL ES4	COLL ES4	COLL ES4	COLL ES1	FG M5	ANTIDIC M6	LIB M7	LIB ES2	LIB ES2	FIN ES7	COLL ES8	COLL ES4
FG	FIN ES9	FG ES4	FG ES4	FG ES4	FG ES4	FG M5	ANTIDIC M6	LIB M7	LIB ES2	LIB ES2	FIN ES7	FG ES8	FG ES4
ANTIDIC	FIN ES9	ANTIDIC ES4	ANTIDIC ES4	ANTIDIC ES4	ANTIDIC ES4	ANTIDIC ES4	ANTIDIC ES1	LIB M7	LIB ES2	LIB ES2	FIN ES7	ANTIDIC ES8	ANTIDIC ES4
LIB	FIN ES9	LIB ES4	LIB ES4	LIB ES4	LIB ES4	LIB ES4	LIB ES4	LIB ES1	NBMOT M8	FMOT M9	FIN ES7	LIB ES8	LIB ES4
NBMOT	FIN ES9	NBMOT ES4	NBMOT ES4	NBMOT ES4	NBMOT ES4	NBMOT ES4	NBMOT ES4	NBMOT ES4	NBMOT ES1	FMOT M9	FIN ES7	NBMOT ES8	NBMOT ES4
FMOT	FIN ES9	FMOT ES4	FMOT ES4	FMOT ES4	FMOT ES4	FMOT ES4	FMOT ES4	LIB M8	LIB ES2	FMOT M9	FIN M10	FMOT ES8	FMOT ES4
Début	début ES1	TIR/ESP M1	LONG M2	DOUB M3	COLL M4	FG M5	ANTIDIC M6	LIB M7	LIB ES2	LIB ES2	FIN ES6	début ES8	NUM M11
FIN													
NUM	FIN ES9	TIR/ESP M1	LONG M2	DOUB M3	COLL M4	FG M5	ANTIDIC M6	LIB M7	LIB ES2	LIB ES2	FIN ES6	NUM ES8	NUM ES1



Actions de l'automate METHODE

- M1 : Caractéristique T ou E enregistrée.
- M2 : Interprétation de "ope nombre", stockage du nombre ou du nombre -1 suivant que ope est  $\leq$  ou  $<$ .  
Si erreur de contexte opérateur, mettre  $\leq$ , et message.  
Si erreur nombre, ne pas stocker la caract, et message.
- M3 : Caractéristique D enregistrée.
- M4 : Reconnaître "chiffre" ou numéro.  
Si CHIFFRE, collision de une méthode = 0.  
Si numéro [1, 10], collision de une méthode = numéro.
- M5 : Reconnaissance du nom de fonction impossible à ce niveau.  
Stocker identificateur, ne rien faire si celui-ci manque, message.
- M6 : Isoler et stocker les identificateurs jusqu'au 5ème. Au-delà, isoler avec message de non prise en compte.  
Association par reconnaissance impossible à ce niveau.
- M7 : Génération d'une entité FILTRE-LIB en séquence de celles existant déjà.
- M8 : Interprétation de la borne de longueur et stockage.  
Si valeurs numériques erronées, stocker les minimax et message.
- M9 : Génération d'un filtre-mot.  
Passage à l'automate FMOT dans l'état début.
- M10 : Fin de définition d'une méthode. Préparation de la suite du travail d'exécution : liste des dictionnaires, liste de fonctions.  
Rajouter des vérifications sémantiques : de recouvrement et d'exhaustivité des filtres-lib. Ne pas le faire pour les filtres-mot en raison de leur complexité.  
Editer la méthode stockée pour montrer les résultats des décisions-erreurs par rapport au texte initial édité lui aussi (vérifier son stockage et les diverses erreurs).
- M11 : Stocker le numéro de la méthode à modifier.

## ERREUR-METH1

- ES1 : Le mot-clé courant est du même type que le précédent. Il y a une erreur mais on se contente de ne pas mettre le dernier apparu dans la base et d'éditer un message.
- ES2 : Erreur par oubli probable de \*LIB. On tombe trop tôt sur NBMOT ou MOT. On suppose que cela est la seule erreur  $\Rightarrow$  on crée un nouveau filtre-lib [M8<sup>+</sup>] dont on édite l'existence en erreur puis on se met dans l'état LIB sans avancer.
- ES4 : Le mot-clé que l'on rencontre correspond à un retour arrière non admissible. On reste dans l'état d'où l'on vient. On balaye jusqu'au mot-clé suivant pour avoir le nouveau "mot courant". On édite la caractéristique négligée pour le message d'erreur.
- ES5 : On vient de finir un filtre-mot, donc on est dans un filtre-lib quand on rencontre une "nouvelle définition de méthode". Conserver la position de \*\*METHODE en entrée dans une caractéristique d'interpréteur, puis exécuter toutes les tâches de fin-de-méthode soit M11. Si besoin comme dans M11, redonner le contrôle à l'automate des automates, mais vérifier avant qu'il ne détruise pas la place du mot-clé courant en entrée [\*\*METHODE].
- ES6 : Il n'y a rien dans la méthode Message et sortie avec reprise du contrôle par l'automate des automates.
- ES7 : Fin de définition de méthode intempestive. Suppression de ce qui a été généré. Edition de tout ce qui a été rencontré comme erreur. Contrôle à AUTO-AUTO.
- ES8 : On a un mot-clé inadmissible ici. Même si cela correspond à un filtre-mot on ne reconnaît pas le mot. On reste dans l'état où l'on était et on balaye pour fournir le mot-clé suivant [ce qui peut entraîner la répétition n de ce message d'erreur]. Message [situation  $\pm$  équivalente à ES4]. Tester avant si c'est un mot-clé de l'Automate des Automates.

ES9 : Fin intempestive non déclarée. Même chose que ES7, sauf qu'en plus on a l'annonce d'une nouvelle méthode. Edition des erreurs. Pas de génération de méthode. Contrôle à AUTO-AUTO de telle sorte qu'il accepte \*\*METHODE.

Remarque :

*Pour chaque erreur, un index-erreur mis à 1 dès qu'on a rencontré dans une méthode une erreur de ce type, est augmenté de 1 à chaque nouvelle erreur dans ce type. Bilan erreur = n, erreur type ES1 k ...*

A U T O M A T E F I L T R E - M O T

	*MOT	*POS	*FREQ	*NBSYLL	*NBPHON	*NBLETR	*X nbre	*DIC	*LIB	**FMETH	autre	*PHON *LETR
début	début ER1	POS FM1	FRQ FM2	TAIL FM3	TAIL FM3	TAIL FM3	CPT FM4	DIC FM5	FIN EF1	FIN EF1	début EF2	CALF FM6
POS	FIN EF1	POS ER1	FRQ FM2	TAIL FM3	TAIL FM3	TAIL FM3	CPT FM4	DIC FM5	FIN EF1	FIN EF1	POS EF2	CALF FM6
FRQ	FIN EF1	FRQ EI1	FRQ ER1	TAIL FM3	TAIL FM3	TAIL FM3	CPT FM4	DIC FM5	FIN EF1	FIN EF1	FRQ EF2	CALF FM6
TAIL	FIN EF1	TAIL EI1	TAIL EI1	TAIL ER1	TAIL ER1	TAIL ER1	CPT FM4	DIC FM5	FIN EF1	FIN EF1	TAIL EF2	CALF FM6
CPT	FIN EF1	CPT EI1	CPT EI1	CPT EI1	CPT EI1	CPT EI1	CPT ER1	DIC FM5	FIN EF1	FIN EF1	CPT EF2	CALF FM6
DIC	FIN F1	DIC EI2	DIC EI2	DIC EI2	DIC EI2	DIC EI2	DIC EI2	DIC ER1	FIN F1	FIN F1	FIN EF3	CALF FM6
CALF	FIN F1	CALF EI2	CALF EI2	CALF EI2	CALF EI2	CALF EI2	CALF EI2	CALF EI1	FIN F1	FIN F1	FIN EF3	CALF FM6
FIN	ramène à FMOT du précédent automate											



## ACTIONS DE L'AUTOMATE FILTRE-MOT.

- FM1 : Enregistrer position par calcul des bornes et interprétation des deux types d'expression. Lecture jusqu'au mot-clé suivant.
- FM2 : Interpréter et vérifier la validité du paramètre de fréquence  
\*FREQG            \*FREQR            \*FREQL  
Interpréter expression de fréquence pour le stockage.
- FM3 : Enregistrement de l'unité et interprétation de la longueur jusqu'au mot-clé suivant. Deux bornes.
- FM4 : Expression de compteur. Enregistrement comme au-dessus de deux bornes pour l'expression fournie.
- FM5 : Cf. M7 ; seule différence : la position dans la structure.
- FM6 : Existence d'un calcul : appel de l'automate CALCUL avec positionnement de l'état retour à CALF. Passage du contrôle à l'autre automate dans l'état "début".
- F1 : Fin normale de filtre-mot [dic et/ou calcul]. On fait les éditions concernant les erreurs du filtre-mot, puis on repart dans l'automate méthode dans l'état FMOT avec le mot-clé courant inchangé. On vérifie avant que tous les calculs liés à ce filtre-mot sont du même type.
- ER1 : Erreur de répétition de l'argument courant. Ne pas prendre en compte. Message. Passage au mot-clé suivant.
- EF1 : Erreur de Fin intempestive. On n'a enregistré aucune caractéristique "active" du filtre-mot [dico ou calcul]. On supprime ce filtre-mot après message et retour du contrôle à l'automate appelant Méthode par récupération de l'état FMOT.
- EF2 : Mot-clé inconnu à ce niveau. On lit jusqu'au mot suivant sans changer d'état avec message.
- EF3 : On rencontre un mot-clé inconnu de cet automate alors que le filtre-mot contient déjà des caractéristiques de dictionnaire ou de calcul. On termine le filtre-mot avec en prime un message et on rend le contrôle à l'automate appelant Méthode. Le filtre est donc conservé [F1].

- EI1 : Un prédicat apparaît après un autre qui est censé le précéder. Comme pour EF2, on lit jusqu'au mot-clé suivant sans changer d'état. Seul le message change.
- EI2 : On a pour ce filtre-mot enregistré des dictionnaires quand apparaît un prédicat qui devrait être avant ceux-ci. On ne peut décider s'il s'agit d'une erreur ou du simple oubli de \*MOT pour annoncer un nouveau filtre-mot. Même action que pour EI2 avec seulement la différence de message.

AUTOMATE \*CALCUL

	*MOT *LIB *FMETH	*SYLL	*PHON	*LETR	*NBPHON	*NBLETR	*POS	type	*FREQ	*Xnbre	*PRENDRE *TROUVER	*Fident	*A *Z *D	*BOUT	autre
début		SCAL CO	PCAL CO	LCAL CO											
SCAL	FIN SC1	FIN EGO	FIN EG1	FIN EG1	STAIL FC1	STAIL FC1	SPOS FC2	SCAL ET1	SCAL ET1	CPT FC5	FCHX AC0	FCT AC1	MCPT AC2	BOUT AC3	SCAL ES1
PCAL	FIN SC1	FIN EG1	FIN EGO	FIN EG1	PTAIL EU1	PTAIL FC1	POS FC2	TYPE FC3	FREQ FC4	CPT FC5	FCHX AC0	FCT AC1	MCPT AC2	BOUT AC3	PCAL ES1
LCAL	FIN SC1	FIN EG1	FIN EG1	FIN EGO	LCAL ET1	LCAL ET1	POS FC2	TYPE FC3	FREQ FC4	CPT FC5	FCHX AC0	FCT AC1	MCPT AC2	BOUT AC3	LCAL ES1
STAIL	FIN SC1	FIN SC2	FIN EG1	FIN EG1	STAIL ER1	STAIL ER1	SPOS FC2	STAIL ET1	STAIL ET1	CPT FC5	FCHX AC0	FCT AC1	MCPT AC2	BOUT AC3	STAIL ES1
SPOS	FIN SC1	FIN SC2	FIN EG1	FIN EG1	SPOS EI1	SPOS EI1	SPOS ER1	SPOS ET1	SPOS ET1	CPT FC5	FCHX AC0	FCT AC1	MCPT AC2	BOUT AC3	SPOS ES1
PTAIL	FIN SC1	FIN EG1	FIN SC2	FIN EG1	PTAIL EU1	PTAIL ER1	POS FC2	TYPE FC3	FREQ FC4	CPT FC5	FCHX AC0	FCT AC1	MCPT AC2	BOUT AC3	PTAIL ES1
POS	FIN SC1	FIN EG1	FIN SC2	FIN SC2	POS EI1	POS EI1	POS ER1	TYPE FC3	FREQ FC4	CPT FC5	FCHX AC0	FCT AC1	MCPT AC2	BOUT AC3	POS ES1
TYPE	FIN SC1	FIN EG1	FIN SC2	FIN SC2	TYPE EI1	TYPE EI1	TYPE EI1	TYPE ER1	FREQ FC4	CPT FC5	FCHX AC0	FCT AC1	MCPT AC2	BOUT AC3	TYPE ES1
FREQ	FIN SC1	FIN EG1	FIN SC2	FIN SC2	FREQ EI1	FREQ EI1	FREQ EI1	FREQ EI1	FREQ ER1	CPT FC5	FCHX AC0	FCT AC1	MCPT AC2	BOUT AC3	FREQ ES1
CPT	FIN SC1	FIN SC2	FIN SC2	FIN SC2	CPT EI1	CPT EI1	CPT EI1	CPT EI1	CPT EI1	CPT FC5	FCHX AC0	FCT AC1	MCPT AC2	BOUT AC3	CPT ES1
FCHX	FIN SC1	FIN SC2	FIN SC2	FIN SC2	FIN EF1	FIN EF1	FIN EF1	FIN EF1	FIN EF1	FIN EF1	FCHX AC0	FCT AC1	MCPT AC2	BOUT AC3	FCHX ES1
FCT	FIN SC1	FIN SC2	FIN SC2	FIN SC2	FIN EF1	FIN EF1	FIN EF1	FIN EF1	FIN EF1	FIN EF1	FCT EI1	FCT AC1	FCT EI1	BOUT AC3	IFCT ES1
MCPT	FIN SC1	FIN SC2	FIN SC2	FIN SC2	FIN EF1	FIN EF1	FIN EF1	FIN EF1	FIN EF1	FIN EF1	MCPT EI1	FCT AC1	MCPT AC2	BOUT AC3	MCPT ES1
BOUT	FIN SC1	FIN SC2	FIN SC2	FIN SC2	FIN EF1	FIN EF1	FIN EF1	FIN EF1	FIN EF1	FIN EF1	BOUT EI1	BOUT EI1	BOUT EI1	FIN ER1	FIN ES1



- CO : Initialisation du calcul et enregistrement du type ou contrôle de sa validité (si non fait dans 'mot'). "D" chargé pour prévenir un calcul vide.
- FC1 : Interprétation et enregistrement de "taille". Unité d'abord, puis "longueur" traduite toujours en deux bornes. Contrôle de l'ordre des bornes (cf. filtre-lib).
- FC2 : Position enregistrée (cf. filtre-mot).
- FC3 : (Pas dans calcul syllabique). Interprétation et stockage du mot-clé de type.
- FC4 : Enregistrement de fréquence. Peut-être faut-il prévoir une structure de contrôle donnant les types de fréquence implantés dans la base pour ne pas enregistrer d'impossibilité. Ensuite, interprétation de l'expression de fréquence.
- FC4 : Numéro de compteur, puis interprétation de l'expression (cf. filtre-mot).

### Actions

- AC0 : Reconnaissance d'un "choix" à exécuter. Appel de l'automate. On n'avance pas dans la position courante de la chaîne car on ne traite pas le type dans l'appel mais dans l'automate. Retour à l'état FCHX.
- AC1 : Nom de fonction stockée. Présence non vérifiée.
- AC2 : Modification de compteur par prédicat validé. Type A, Z, D, AX, ZX, DX suivi du numéro  $\epsilon$  [1, 10].
- AC3 : Bout = B de la structure. Seule, l'exécution peut vérifier qu'il n'en exécute qu'un.

### Sorties

- SC1 : Calcul terminant normalement (aux erreurs intermédiaires près). Il peut ne contenir aucune caractéristique. Restitution du contrôle à l'automate mot, sans avancer de mot-clé et de l'état F calcul.
- SC2 : Annonce du début d'un nouveau calcul. Cependant, passage à nouveau dans l'automate MOT qui rappelle calcul ensuite. Peut, peut-être, être confondu avec SC1.

## Erreurs

- EG0 : Répétition du type deux fois. Prise en compte d'un seul. Message.
- EG1 : Autre nom du calcul. Différent du précédent. Prise en compte du seul même s'il doit conclure à un conflit de type. Message.
- ET1 : Prédicat d'un type interdit par le filtre en cours. Passer jusqu'au mot-clé suivant. Message.
- ES1 : Séquence rompue pour mot-clé inconnu. Passer au mot-clé suivant. Message.
- EU1 : Erreur d'unité dans une clause de taille. Forcer la valeur [NB Lettre] mais avec un message très explicite.
- ER1 : Répétition d'un prédicat qui ne doit pas l'être. Passer au mot-clé suivant. Message.
- EI1 : Inversion (qui peut être une répétition) de l'ordre des prédicats. Passer. Message. Cette inversion peut aussi être celle de deux actions.
- EF1 : Rencontre après une définition d'action, d'un nouveau prédicat. On pourrait être au début d'un calcul dont le type aurait été omis. On pourrait "inventer" ce type comme cohérent avec les précédents, mais cela complique le jeu des pointeurs sur le texte. On se contente donc de laisser tomber tout le texte jusqu'au calcul suivant (sauf si on se rend compte qu'un type artificiel est plus facile) ou jusqu'à la détection de fin-filtre-mot.

### ACTIONS DE L'AUTOMATE CHOIX

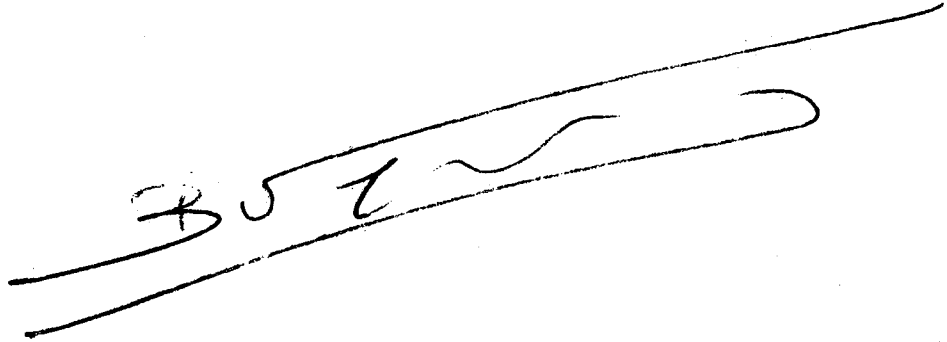
- CH0 : Reconnaissance du type de choix [prendre sa limite au champ défini]  
[trouver va aussi loin qu'il faut].
- CH2 : Mise à "F" de fixe si existe cela veut dire que le pointeur de caractère  
ne se déplace pas dans ce choix jusqu'à ce que sa limite droite ou  
gauche ait été définie .
- CH3 : Enregistrement de la position de départ dans nombre-litt, nombre-num  
expression standard mais une seule borne mass .
- CH4 : Même chose pour la borne de départ dans depuis-litt et depuis-num.
- CH5 : Chaîne ou type suivant la donnée transmise.
- CH1 : Modification de compteur.
- CH6 : Fin normale et retour à l'automate calcul sans changer de mot-clé pour  
l'éventuel début d'un nouveau choix, nouveau calcul ou nouveau filtre.
- EI1 : Cf. calcul.
- ER1 : Cf. calcul.
- EI2 : Rencontre de tous les éléments indiquant qu'on repart peut-être dans un  
nouveau calcul, mais sans avoir indiqué de verbe action. Schifter jusqu'à  
rencontrer un "mot de fin" ou un verbe, ce qui est aussi un mot de fin.
- ES1 : Cf. calcul.
- ES2 : On vient de rencontrer un mot-clé inconnu de l'automate quand le texte  
du choix est "complet". On laisse les autres automates se débrouiller  
en leur rendant le contrôle sur ce mot.
- EF1 : Erreur de fin. On rencontre la fin du choix avant d'avoir défini le type  
du caractère sur lequel on fait ce choix. Cette erreur est fatale mais  
puisque le choix a été généré, on peut toujours continuer à interpréter  
la suite en vue de détecter les erreurs. On n'avance pas et on rend le  
contrôle.
- EF2 : La fin rencontrée, erronée pour les mêmes raisons que EF1 est celle de  
l'entrée dans un nouveau choix. On n'efface pas non plus celui en cours  
et on rend le contrôle. Au fond très équivalent à EF1.

Dernière page d'une thèse

VU

Grenoble, le 14 mai 1976

Le Président de la thèse



Vu, et permis d'imprimer,

Grenoble, le

Le Président de l'Université  
Scientifique et Médicale

