



## AVERTISSEMENT

Ce document est le fruit d'un long travail approuvé par le jury de soutenance et mis à disposition de l'ensemble de la communauté universitaire élargie.

Il est soumis à la propriété intellectuelle de l'auteur. Ceci implique une obligation de citation et de référencement lors de l'utilisation de ce document.

D'autre part, toute contrefaçon, plagiat, reproduction illicite encourt une poursuite pénale.

Contact : [ddoc-theses-contact@univ-lorraine.fr](mailto:ddoc-theses-contact@univ-lorraine.fr)

## LIENS

Code de la Propriété Intellectuelle. articles L 122. 4

Code de la Propriété Intellectuelle. articles L 335.2- L 335.10

[http://www.cfcopies.com/V2/leg/leg\\_droi.php](http://www.cfcopies.com/V2/leg/leg_droi.php)

<http://www.culture.gouv.fr/culture/infos-pratiques/droits/protection.htm>

# UN SYSTÈME MULTILINGUAL D'INTERPRÉTATION AUTOMATIQUE

ÉTAPE DU SOUS-LOGICIEL "ANALYSE"  
POUR LES LANGUES GERMANIQUES

1<sup>ère</sup> partie

BIBLIOTHEQUE UNIVERSITAIRE DE METZ



022 420060 4

Thèse

par Pierre DIMON

Lettres

BIBLIOTHEQUE UNIVERSITAIRE LETTRES - METZ -	
N° Inv.	1994032L
Cote	L/M3 94/5
Loc.	Magasin I

## Avant-propos

Il y a plus de quinze ans, le Professeur Jean-Marie ZEMB (Collège de France) nous a donné l'occasion de découvrir le Traitement Automatique des Langues Naturelles sous la conduite de Daniel HERAULT, directeur du Centre de Recherche Jean Favard et de Patrice POGNAN, professeur à l'INALCO. Nous lui devons la découverte d'un domaine passionnant dont la pluridisciplinarité n'est pas le moindre intérêt. Nous lui en sommes particulièrement reconnaissant.

Implanté à Paris et associé au Service de Linguistique de l'Université Pierre et Marie Curie (Paris VI), le Centre de recherche a effectué de nombreux travaux sur le spectre sémantique<sup>1</sup>, une idée de Daniel HERAULT<sup>2</sup> approfondie depuis 1970 sur le français, le bulgare, le tchèque, le russe, l'allemand et le japonais.

- Collaborant avec l'Académie des Sciences de Bulgarie (Institut de Mathématiques) et sous l'égide du CNRS, le Centre a étudié le spectre sémantique slave (A. JUDSKANOV) et précisé la nature d'un spectre sémantique indo-européen, en soulignant l'universalité probable des caractéristiques profondes de sa partie prédicative. Les travaux ont confirmé la possibilité d'obtenir une description convenable du contenu des textes non-littéraires.

- Le soutien de la DRME (Direction des Recherches et Moyens d'Essai) puis de la DRET (Direction des Recherches et Etudes Techniques) pour le Ministère de la Défense a permis d'étendre ces études à d'autres langues avant de concevoir l'Automate de Compréhension Implicite<sup>3</sup> (ACI) qui, partant des propriétés essentielles du spectre sémantique, donnera d'un texte non-littéraire une certaine forme de compréhension.

- Le CNRS<sup>4</sup> a soutenu nos premiers travaux sur l'allemand (module de traitement des mots composés<sup>5</sup> et module verbal).

- Le professeur Jean-Claude LEJOSNE (Université de Metz) n'a ménagé ni son temps ni son aide pour le test des modules d'analyse sur l'anglais et le néerlandais. Nous n'avons pas oublié les séances de travail tardives et fructueuses et l'en remercions vivement.

- Sans la bienveillante attention du Président Jean DAVID, le Traitement Automatique des Langues Naturelles n'aurait pas pu se développer dans des conditions aussi favorables qu'à la Faculté des Lettres de Metz. Nous le remercions avec gratitude pour ses conseils, ses suggestions et l'intérêt qu'il a bien voulu porter à ce travail, tout au long de son déroulement.

(1) Cf. Introduction et chapitre IV

(2) D. HERAULT : *Compréhension automatique et spectre sémantique (russe, bulgare, tchèque, français)*, Documents de Linguistique Quantitative n°18, Editions Jean Favard, Paris, 1981

(3) Cf. Introduction et chapitre IV

(4) Action Thématique Programmée : "Mise au point de procédés de simulation de la lecture humaine, avec appréhension automatique du contenu sémantique de textes non-littéraires en langue naturelle".

(5) P. DIMON : "Aspects de la composition dans les langues germaniques et réalisation d'un algorithme de localisation des composés pour l'allemand", Thèse de 3ème cycle, Paris III, 1978

- **Patrice POGNAN** a guidé nos premiers pas en informatique. La fréquentation du Centre de Calcul du CNRS à Orsay, de l'Ecole Normale Supérieure (rue D'Ulm) et du laboratoire de la Maison des Sciences de l'Homme sont très certainement à l'origine d'une passion immodérée pour l'informatique et le traitement automatique des Langues Naturelles.

- Nous avons travaillé avec **Daniel HERAULT** pendant plus de treize ans et tenons à l'assurer ici de toute notre reconnaissance pour la richesse et la rigueur de son enseignement. Cette thèse lui doit beaucoup. Il convient de rappeler qu'il a programmé l'environnement du système, réécrit et optimisé le module de traitement des mots composés. Nous avons quant à nous défini l'enchaînement des programmes, rassemblé et organisé toutes les données linguistiques, conçu et programmé l'analyseur morphologique (le verbe, l'adverbe, l'adjectif) et l'analyseur syntaxique.

- Le professeur **Henri ZINGLÉ** (université de Nice) a accepté de relire notre travail. Nous lui sommes reconnaissant d'en avoir examiné le moindre détail et le remercions très sincèrement pour ses critiques constructives.

# SOMMAIRE

RÉSUMÉ	12
INTRODUCTION	14
- Traduction littéraire, traduction utilitaire	15
- La traduction utilitaire et ses buts	16
- La Traduction automatique	17
- Informatique, linguistique et industrie des langues	18
- Compréhension automatique et Automate de Compréhension Implicite (ACI)	20
<b>I. LE TRAITEMENT AUTOMATIQUE DES LANGUES NATURELLES</b>	<b>21</b>
1.1 Sens et compréhension	22
1.2 Les obstacles à la compréhension	23
1.3 Les connaissances requises pour la compréhension	27
1.4 Les modèles de la langue	29
1.4.1 Les niveaux	30
1.4.2 Les modèles sans syntaxe	30
1.4.3 Les modèles avec syntaxe	31
1.4.3.1 <u>Grammaires formelles</u>	31
1.4.3.1.1 Les grammaires de type 3	32
1.4.3.1.2 Les grammaires de type 2	32
1.4.3.1.3 Les grammaires de type 1	33
1.4.3.1.4 Les grammaires de type 0	34
1.4.3.2 <u>Grammaires transformationnelles</u>	34
1.4.3.2.1 La théorie standard	35
1.4.3.2.2 La transformation	35
1.4.3.2.3 Les grammaires en chaîne	36
1.4.3.2.4 Extensions de la théorie standard	37
1.4.3.2.4.1 La théorie standard étendue	37
1.4.3.2.4.2 La théorie des traces	37
1.4.3.2.4.3 Les grammaires syntagmatiques généralisées	38
1.4.3.3 <u>Grammaires de cas</u>	39
1.4.3.3.1 La théorie de C. FILLMORE	39
1.4.3.3.2 La théorie de J. GRIMES	40
1.4.3.3.3 La théorie de R. SIMMONS	40

1.4.3.3.4 La théorie de R. SCHANK	40
1.4.3.3.5 La théorie de B.B. BOGURAEV et K. SPARK-JONES	41
<i>1.4.3.4 Grammaires sémantiques</i>	42
<i>1.4.3.5 Grammaires systématiques</i>	42
1.4.3.5.1 Présentation	42
1.4.3.5.2 Notion de système	43
1.4.3.5.3 Le modèle de M. HALLIDAY	44
1.4.3.5.4 T. WINOGRAD et le SHRDLU	44
<i>1.4.3.6 Grammaires et lexique</i>	44
1.4.3.6.1 Les grammaires d'unification	44
1.4.3.6.1.1 Le formalisme	45
1.4.3.6.1.2 Utilisation du formalisme	45
1.4.3.6.1.3 La superposition	48
1.4.3.6.1.4 La notion de chemin	49
1.4.3.6.2 Les grammaires lexicales et fonctionnelles	50
1.4.3.6.2.1 La production d'une phrase	51
1.4.3.6.2.2 La représentation d'une phrase	52
1.4.3.6.2.3 Phénomènes à distance, éléments discontinus	53
1.4.3.6.3 Le lexique grammaire de M. GROSS	55
1.4.3.6.3.1 Les dictionnaires du LADL	55
1.4.3.6.3.2 Les lexiques-grammaires	56
1.4.3.6.4 Le modèle Sens-Texte de I. MEL'CUK	56
1.4.4 Conclusion	57
1.5 Les outils	57
1.5.1 Introduction	57
1.5.2 <i>L'analyse morphologique</i>	57
1.5.2.1 Les techniques	58
1.5.2.2 T. WINOGRAD	59
1.5.2.3 J. PITRAT	61
1.5.2.4 Problèmes annexes	63
1.5.3 <i>L'analyse de la phrase</i>	64
1.5.3.1 La grammaire et la description interne de la phrase	64
1.5.3.1.1 Syntaxe et sémantique	66
1.5.3.2 Traitement automatique	67
1.5.3.2.1 L'architecture des systèmes	67
1.5.3.2.2 L'analyse	69

1.5.3.3 Exemples	72
1.5.4 <u>Les réseaux de transition</u>	75
1.5.4.1 Introduction	75
1.5.4.2 Le RTN (Recursive Transition Network)	78
1.5.4.3 L'ATN (Augmented Transition Network)	81
1.5.4.4 Le CATN (Cascaded Augmented Transition Network)	82
1.5.4.5 Les réseaux à noeuds procéduraux	83
1.5.4.6 Exemples	83
1.5.5 <u>Les analyseurs déterministes</u>	84
1.5.5.1 Présentation	84
1.5.5.2 Exemples	86
1.5.5.2.1 PARSIFAL	86
1.5.5.2.2 ANDI (Analyseur Déterministe Intégré)	87
1.5.6 <u>Les analyseurs à mots clés</u>	87
1.5.7 <u>Les analyseurs conceptuels</u>	88
1.5.8 <u>Les grammaires logiques</u>	89
1.5.8.1 Les grammaires de métamorphose. Les DCG (Definite Clause Grammars)	90
1.5.8.2 Les extensions	91
1.5.9 <u>Les outils intégrés</u>	92
1.5.10 <u>Le traitement des erreurs</u>	92
1.5.10.1 Plan lexical	93
1.5.10.1.1 Types d'erreur	93
1.5.10.1.2 Techniques de correction	93
1.5.10.2 Plan syntaxique	95
1.5.10.2.1 Types d'erreur	95
1.5.10.2.2 Techniques de correction	95
1.5.10.3 Plan sémantique	96
1.5.10.3.1 Types d'erreur	96
1.5.10.3.2 Techniques de correction	96
1.6 Conclusion	97

## II. TRADUCTION AUTOMATIQUE (T.A.), TRADUCTION ASSISTÉE PAR ORDINATEUR (T.A.O.)

2.1 Industrie des langues	102
2.2 La traduction	103

2.2.1 Les outils de la traduction informatisée	103
2.2.2 Les besoins	105
2.2.3 Evolutions théoriques et techniques	105
2.2.4 Architecture générale et catégories	107
2.2.5 Techniques linguistiques	107
2.2.6 Intelligence Artificielle	109
2.3 Les systèmes de T.A. et de T.A.O.	109
2.3.1 Historique	109
2.3.2 <u>Les systèmes de base</u>	114
2.3.2.1 GAT	114
2.3.2.2 CETA	114
2.3.2.3 METAL	115
2.3.2.4 TAUM	115
2.3.2.5 ALP	115
2.3.3 <u>Présentation des systèmes actuels</u>	116
2.3.3.1 Introduction	116
2.3.3.1.1 Contexte	116
2.3.3.1.2 Les utilisateurs	116
2.3.3.1.3 Les problèmes spécifiques	116
2.3.3.1.4 Les critères de choix	118
2.3.3.2 Les systèmes commercialisés actuellement	118
2.3.3.2.1 ATLAS I, ATLAS II	118
2.3.3.2.2 HICATS	122
2.3.3.2.3 LOGOS	122
2.3.3.2.4 LP	130
2.3.3.2.5 METAL	130
2.3.3.2.6 PENSEE	142
2.3.3.2.7 PIVOT	142
2.3.3.2.8 SANYO	142
2.3.3.2.9 SYSTRAN	142
2.3.3.2.10 SYSTRAN (JAPON)	163
2.3.3.2.11 TAURAS	164
2.3.3.2.12 WEIDNER	165
2.3.3.3 Les systèmes dédiés	168
2.3.3.3.1 AMPAR	168
2.3.3.3.2 CHIMKENT	168
2.3.3.3.3 CULT	168
2.3.3.3.4 FRAP	168
2.3.3.3.5 NERPA	168
2.3.3.3.6 SPANAM - ENGSPAN	169
2.3.3.3.7 TAUM	176
2.3.3.3.7.1 METEO	176
2.3.3.3.7.2 AVIATION	178

2.3.3.3.8 TITRAN	183
2.3.3.3.9 TITUS IV	184
2.3.3.3.10 TITUS V	192
2.3.3.4 Aides à la traduction	192
2.3.3.4.1 ALPS	193
2.3.3.4.2 MERCURY	195
2.3.3.4.3 MULTI LINGUA	195
2.3.3.4.4 SITE	195
2.3.3.4.5 SMART	195
2.3.3.4.6 TII	196
2.3.4 <u>Prototypes en voie d'achèvement</u>	196
2.3.4.1 ATAMIRA	196
2.3.4.2 IBM JAPON	196
2.3.4.3 MELTRAN	196
2.3.3.4 RMT	196
2.3.3.5 SHARP	196
2.3.3.6 TOVNA	197
2.3.3.7 TRANSTAR 1	197
2.3.5 <u>Projets</u>	197
2.3.5.1 ATTP	197
2.3.5.2 BRITISH TELECOM	197
2.3.5.3 BYU-TAS	198
2.3.5.4 CAP SOGETI	198
2.3.5.5 DLT	198
2.3.5.6 EDR	199
2.3.5.7 EUROTRA	200
2.3.3.8 GETA	206
2.3.3.8.1 ARIANE	206
2.3.3.8.2 CALLIOPE	208
2.3.3.9 LUTE	212
2.3.3.10 MARIS	213
2.3.3.11 Autres projets	213
2.4 Services de traduction	215
2.5 Banques de données terminologiques	216
2.6 Conclusion	216
<b>III. UN SYSTÈME D'INTERPRÉTATION AUTOMATIQUE : L'AUTOMATE DE COMPRÉHENSION IMPLICITE (ACI)</b>	218
3.1 <u>Le contexte</u>	219
3.1.1 Analyse des systèmes et évolution	219
3.1.1.1 De 1950 à 1970	219
3.1.1.2 De 1970 à nos jours	222

3.1.2 Evolution des conceptions	223
<b><u>3.2 Une autre approche de la compréhension avec l'ACI</u></b>	<b>224</b>
3.2.1 Définition	224
3.2.2 Principes	224
3.2.2.1 Compréhension explicite, compréhension implicite	224
3.2.2.2 Interprétation automatique	225
3.2.2.3 Le système dérivationnel	229
3.2.2.3.1 Les propriétés quantitatives des systèmes dérivationnels	229
3.2.2.3.2 La classification du lexique	231
3.2.2.3.3 La sélection entre plusieurs systèmes dérivationnels	231
3.2.2.4 La compréhension et sa localisation	232
3.2.2.5 Le transfert "global"	232
3.2.2.6 L'absence de dictionnaire	234
3.2.2.7 Analyse de surface	234
3.2.2.8 Le principe des groupes de langues	235
3.2.2.9 Hyperanalyse, hypersémantique et hypersyntaxe	236
3.2.2.9.1 Le module hypersémantique	236
3.2.2.9.2 Le module hypersyntaxique	237
3.2.2.9.2.1 Hypersyntaxe locale	237
3.2.2.9.2.2 Hypersyntaxe globale	237
<b><u>3.3 Le système d'interprétation automatique des langues germaniques</u></b>	<b>238</b>
3.3.1 Caractéristiques générales	238
3.3.2 Langues sources germaniques	238
3.3.3 Le sous-logiciel <i>ANALYSE DE LA LANGUE SOURCE</i>	239
3.3.3.1 Le module verbal	239
3.3.3.2 Le module "mots composés"	240
3.3.3.2.1 Le sous-module "Analyse générale"	240
3.3.3.2.2 Le sous-module "Décomposition Morphématique"	240
3.3.3.2.3 Le sous-module "Segmentation par cohérence"	240
3.3.3.2.4 Le sous-module "Mots composés verbaux"	241
3.3.3.3 Le module "Analyseur syntaxique"	241
3.3.4 Le sous-logiciel <i>GENERATION DE LA LANGUE CIBLE</i>	242
3.3.4.1 Le module d'interprétation des mots composés	242
3.3.4.2 Le module d'interprétation des syntagmes verbaux	243
3.3.4.3 Le module d'interprétation des mots non composés	243
3.3.4.4 Le module du transfert de syntaxe	243
3.3.4.5 Le module de création des SN français	243
3.3.4.6 Le module final	244
3.3.4.7 L'aspect multilingual	244

<b>IV. L'AUTOMATE "GERMANIQUE-ROMAN" :</b>	
<b>STRUCTURE ET FONCTIONNEMENT</b>	245
<b><u>4.1 Caractéristiques générales</u></b>	246
4.1.1 Système	246
4.1.2 Multilinguisme	248
4.1.3 Interprétation	248
4.1.4 Automatique	249
4.1.5 Analyse	250
4.1.6 Langues germaniques	251
<b><u>4.2 Contextes</u></b>	251
4.2.1 Le contexte linguistique	251
4.2.1.1 Les difficultés	251
4.2.1.2 Les solutions de l'ACI	252
4.2.1.2.1 Les niveaux morphologique et lexical	252
4.2.1.2.2 Le niveau syntaxique	256
4.2.2 Le contexte informatique	257
4.2.2.1 Introduction à PROLOG	258
4.2.2.1.1 L'arbre	258
4.2.2.1.2 Les faits	259
4.2.2.1.3 Les questions	260
4.2.2.1.4 Les règles	261
4.2.2.1.5 Le mécanisme d'unification	262
4.2.2.1.6 La récursivité	263
4.2.2.1.7 Mécanismes de contrôle	265
4.2.2.1.8 Les prédicats définis	266
4.2.2.1.9 Le retour en arrière et la coupure	266
4.2.2.1.10 Les listes	266
4.2.2.2 PROLOG et le cadre linguistique	267
4.2.2.2.1 Les grammaires formelles	267
4.2.2.2.2 Les catégories syntaxiques	268
4.2.2.2.3 Les arbres syntaxiques	269
4.2.2.2.4 Les structures du lexique	270
4.2.2.2.5 Les transformations	270
4.2.2.2.6 Conclusion	270
4.2.2.3 PROLOG et l'analyse automatique	271
4.2.2.4 La réalisation de l'ACI	277
4.2.2.4.1 Matériel utilisé	277
4.2.2.4.2 Données	277
4.2.2.4.3 Programmes	278
4.2.2.4.4 Procédures	278



4.3.2.5.3.7 Nettoyage des segments (.NETALL1 et .NETALL2)	458
4.3.2.5.3.8 Les mots romans (.RSELECT)	462
4.3.2.5.3.9 Les suffixes romans (.SUFROM)	465
4.3.2.5.3.10 Nettoyage des segments (.NETROM)	466
4.3.2.5.3.11 Fonctionnement de la procédure	469
4.3.2.5.4 Etape 3 : Critères de cohérence interne (MOTCOMPA)	473
4.3.2.5.4.1 Vue d'ensemble	473
4.3.2.5.4.2 Fonctionnement	473
4.3.2.5.4.3 Comparaison par la droite (.SEGMD)	473
4.3.2.5.4.4 Comparaison par la gauche (.SEGMINT)	474
4.3.2.5.4.5 Comparaison interne (.SEGMINT)	476
4.3.2.5.4.6 Récapitulation des résultats (.RECAPIT)	476
4.3.2.5.4.7 Test du contexte germanique (.ASELECT1)	478
4.3.2.5.4.8 Les suffixes germaniques (.SUFALL1)	479
4.3.2.5.4.9 Nettoyage des segments (.NETALL1)	481
4.3.2.5.4.10 Test en contexte roman (.RSELECT1)	482
4.3.2.5.4.11 Les suffixes romans (.SUFROM1)	483
4.3.2.5.4.12 Nettoyage des segments (.NETROM1)	484
4.3.2.5.5 Etape 4 : Les verbes composés (MOTCOMPB)	488
4.3.2.5.6 Résultats	489
4.3.3 Analyse syntaxique	489
4.3.3.1 Les différentes étapes	489
4.3.3.2 Fonctionnement	490
4.3.3.2.1 Ponctuations, conjonctions délimiteurs (.CRJF(SYNTA1))	492
4.3.3.2.2 Les relatives1 (.CRJF(SYNTA2))	506
4.3.3.2.3 Les relatives2 (.CRJF(SYNTA3))	511
4.3.3.2.4 Les conjonctives (.CRJF(SYNTA4))	518
4.3.3.2.5 Repérage des subordonnées traitement du "als" (.CRJF(SYNTA5))	527
4.3.3.2.6 "zu" et le groupe prépositionnel1 (.CRJF(SYNTA6))	535
4.3.3.2.7 Le groupe prépositionnel2 (.CRJF(SYNTA7))	545
4.3.3.2.8 Le groupe prépositionnel3 (.CRJF(SYNTA8))	548
4.3.3.2.9 Le groupe prépositionnel4 (.CRJF(SYNTA9))	552
4.3.3.2.10 Le groupe nominal (.CRJF(SYNTA10))	558
4.3.3.2.11 Le noyau verbal (.CRJF(SYNTA11))	566
4.4 Les résultats complets	569
V. CONCLUSION	665
- Le problème	666
- Son fondement	666
- Les réponses classiques	666
- Notre réponse : l'automate de compréhension implicite	666
- Ses bases	667
- Ses caractéristiques	667
- Un exemple d'interprétation	668
- L'avenir de la T.A. et de la T.A.O.	675
- Les applications de l'ACI	675

<b>VI. BIBLIOGRAPHIE</b>	677
6.1 Dictionnaires, bibliographies	679
6.2 Grammaires, ouvrages de linguistique, monographies	681
6.3 Modèles linguistiques, traitement automatique des langues	694
6.4 Traduction Assistée par Ordinateur, Traduction Automatique	706
6.5 Systèmes, projets de T.A.O. et de T.A.	714
6.6 Intelligence artificielle, systèmes experts, industrie de la langue, E.A.O., digitalisation	725
6.7 Informatique	730
6.8 Documentation du C.I.R.C.E.	731
6.9 Liste des références classées par auteur et date de parution	732
6.10 Liste des auteurs cités	739
<b>ANNEXE I</b>	743
<b>ANNEXE II</b>	755

# Résumé

## Introduction

Nous définissons le champ de recherche : *la traduction utilitaire*. Liés à l'internationalisation croissante de l'économie, les buts qui lui sont assignés et les besoins afférents appellent de nouveaux moyens parmi lesquels *la Traduction Automatique (TA)* et *la Traduction Assistée par Ordinateur (TAO)*. *Le Traitement Automatique des Langues Naturelles (TALN)* sur un plan plus général, et les progrès de l'Intelligence Artificielle (IA) illustrent le mariage de la linguistique et de l'informatique, moteurs essentiels d'un domaine en pleine expansion : *l'industrie des langues*.

Au coeur de toutes ces activités, la *compréhension automatique*, sous des formes diverses. Une réflexion sur ses mécanismes conduit à cerner les connaissances nécessaires à la compréhension et les choix à opérer pour bâtir un système automatique. Les voies classiques seront explorées et leurs limites soulignées. Nous introduisons ensuite les principes de notre Automate de Compréhension Implicite (ACI), inspiré des travaux du Centre de Recherche Jean Favard. Avec une nouvelle approche de la compréhension, il privilégie l'hypothèse sur les parentés étymologiques des langues européennes.

## Chapitre I

Nous tentons de définir la compréhension (1) et d'en déterminer les caractéristiques dans le contexte du traitement automatique. Les difficultés qu'elle rencontre sont répertoriées et présentées dans l'ordre chronologique de leur apparition (1.2). Les éléments qu'elle requiert sont identifiés (1.3), ce qui nous permet d'aborder les théories nécessaires à la représentation de ces connaissances, les modèles de la langue (1.4) et les moyens mis en oeuvre dans le processus de compréhension, les outils d'analyse (1.5). En conclusion, nous nous demandons s'il est vraiment possible de formaliser clairement les langues.

## Chapitre II

Pour illustrer les modèles et les outils que nous venons de parcourir, le chapitre II développe la notion d'*industrie des langues* (2.1), et présente la T.A., la T.A.O. et leurs applications essentielles (2.3). Après un bilan des évolutions théoriques et techniques, nous soulignons la difficulté de concevoir une classification cohérente des divers systèmes. Nous préférons distinguer *les systèmes de base* (2.3.2), *les systèmes actuels* (2.3.3), *les prototypes* (2.3.4) et *les projets* (2.3.5). En conclusion, tout en citant d'autres outils (2.4 et 2.5), nous soulignons les limites de ces automates et de leurs performances, malgré les recours de plus en plus importants à la sémantique et à l'Intelligence Artificielle.

### **Chapitre III**

L'analyse des systèmes et l'évolution des conceptions précisent le contexte (3.1) dans lequel nous avons choisi une approche nouvelle avec l'ACI (3.2), un automate de compréhension implicite dont nous définirons les principes et la structure générale. L'automate roman-germanique sera décrit dans son ensemble (3.3), y compris les modules qui n'ont pas encore été réalisés.

### **Chapitre IV**

Nous rappelons les caractéristiques générales de l'automate (4.1), les contextes linguistique et informatique (4.2) puis le décrivons dans le détail, niveau d'analyse par niveau d'analyse, (4.3.1, 4.3.2 et 4.3.3). Le chapitre s'achève sur un listing des résultats complets.

### **Conclusion**

Nous rappelons les réponses classiques aux problèmes que pose la traduction utilitaire et proposons une nouvelle perspective avec l'ACI. Nous dégageons les avantages des principes que nous avons retenus et présentons en illustration, une simulation de l'interprétation telle que doit la générer le module final d'interprétation. Nous donnons ensuite quelques exemples d'extensions de l'ACI sous la forme de sous-produits originaux et exploitables en E.A.O. (annexes I et II).

En ce qui concerne la bibliographie et dans un souci de clarté, nous avons tenté de classer les ouvrages et les articles de référence en huit chapitres, qui traduisent la pluridisciplinarité du sujet et l'interdisciplinarité des domaines concernés.

## **INTRODUCTION**

Si nous considérons les langues selon l'expérience quotidienne qu'en a chacun, nous disposons immédiatement d'une première caractérisation, "*impliquée par une activité aussi vieille que les cultures les plus anciennes, attestée jour après jour, indéfiniment reconduite dans sa permanente nécessité au mépris des écueils supposés : la traduction*<sup>1</sup>". Il faut bien admettre, en effet, que tout texte d'une langue peut être traduit approximativement ou parfaitement dans une autre langue. Malgré de nombreux obstacles, chaque langue est "*une sémiotique dans laquelle toutes les autres sémiotiques peuvent être traduites*<sup>2</sup>".

La traduction est l'opération qui consiste à transposer les informations contenues dans un texte source dans un texte cible. Prise dans son sens le plus général, elle pose certains problèmes relatifs à la signification. Le sens se coule dans une grande variété de moules formels, "*le sens est partout. Les traducteurs le savent, d'instinct ou d'expérience, qui choisissent une position pour traduire une forme, ou une forme pour traduire un mot*<sup>3</sup>". Il arrive même que la réalité à cerner ne soit pas toujours nette et parfaitement délimitée...

### Traduction littéraire, traduction utilitaire

Le langage et la pensée entretiennent des relations obscures. A ce propos, la traduction est un objet d'analyse privilégié, mais c'est tardivement qu'elle a suscité l'intérêt des spécialistes du langage. Le philosophe américain W.M. URBAN<sup>4</sup>, les linguistes S.C. GARDINER, O. JESPERSEN, E. SAPIR, VOSSLER et l'ethnologue B. MALINOWSKI<sup>5</sup> lui consacrent une réflexion spécifique. E.A. NIDA<sup>6</sup> rassemble une très grande quantité de problèmes et de solutions formulés du point de vue linguistique. Des besoins de traduction inhérents au statut particulier du Canada ont conduit VINAY et DALBELNET<sup>7</sup> à bâtir une des premières méthodes de traduction fondée sur une base scientifique. Ils mettent en évidence la notion d'unité de traduction, un groupe de mots traduits en bloc parce qu'ils constituent une unité de sens. A partir des années 50, les relations internationales s'amplifient. Les activités de traduction se structurent (Ecoles d'Interprètes, Associations...) et se spécialisent. A.V. FEDOROV<sup>8</sup> et E. CARY<sup>9</sup> étudient les exigences spécifiques de la traduction selon les domaines abordés.

La traduction de la littérature et de la poésie, par exemple, soulève des problèmes considérables. S'il est possible de traduire des structures linguistiques, il est moins évident de traiter des structures métriques, stylistiques ou poétiques, sans parler des connotations culturelles. Le traducteur littéraire n'est pas seulement celui qui réalise l'opération linguistique élémentaire qui consiste à trouver la correspondance la plus exacte des signes d'une langue à l'autre, c'est un amoureux de la communication qui s'efforce de trans-

(1) C. HAGEGE : *L'homme de paroles*, Contribution linguistique aux sciences humaines, Le temps des Sciences, Fayard, Paris, 1985

(2) L. HJELMSLEV : *Prolégomènes à une théorie du langage*, Ed. de Minuit, Paris, 1968, p. 138

(3) J.-M. ZEMB : *Vergleichende Grammatik. Französisch-Deutsch*, Teil 1, (Sonderreihe-Vergleichende Grammatiken), Bibliographisches Institut, Mannheim, Dudenverlag, 1978, p. 27

(4) W.M. URBAN : *Language and Thought*, 1939

(5) B. MALINOWSKI : "The Problem of Meaning in Primitive Languages" in *The Meaning of Meaning*, C.K. OGDEN & I.A. RICHARDS, 1946

(6) E.A. NIDA : *Toward a Science of Translating with Special Reference to Principales and Procedures Involved in Bible Translating*, Brill, Leyde, 1964

(7) J.-P. VILNAY, J. DALBELNET : *La stylistique comparée du français et de l'anglais*, Paris, Montréal, Didier, 1958

(8) A. V. FEDOROV : *Vvedenie v teoriju perevoda*, 1954

(9) E. CARY : *La traduction dans le monde moderne*, 1956

mettre une pensée, des émotions, une culture, le chant profond d'un autre idiome. Un ordinateur ne pourra jamais le remplacer dans cette opération intuitive, et quand bien même, jamais sans son aide.

Nous limiterons donc notre champ d'expérimentation à la traduction de documents techniques et scientifiques, à la traduction utilitaire autrement dit. L'acquisition et la propagation d'informations pertinentes impliquent en effet pour ces textes, une complexité relativement limitée des structures utilisées, ce qui nous conduit à en préciser les buts.

### La traduction utilitaire et ses buts

L'internationalisation de la communication est un phénomène essentiel dans notre société. Si le latin ou l'anglais ont, au cours de notre histoire, permis de véhiculer les informations et de les mettre à la portée d'un grand nombre, la barrière des langues n'en constitue pas moins aujourd'hui un obstacle difficile à surmonter. L'humanité a toujours rêvé d'une société idéale de communication. Elle mesure avec regret la profondeur du fossé qui l'en sépare et ne cesse de croître. Le besoin en information spécialisée souligne les inconvénients d'une disparité que même une institution comme la Communauté Européenne a du mal à résoudre. Le choix d'une langue commune serait une solution. Cependant, les efforts consentis par les organisations internationales (ONU, CEE...) en faveur de la traduction humaine ou automatisée montrent que ce n'est pas une solution à court terme.

Le multilinguisme est une expression de la pluralité et de la richesse de nos civilisations. C'est la seule voie dans laquelle chacun puisse se reconnaître, c'est aussi la chance de maintenir en vie et de continuer à enrichir une civilisation dont nous sommes dépositaires. Au delà des intentions culturelles et "*de la valorisation sans frontière*" d'un "*capital humain considérable*<sup>1</sup>", nous citerons les deux objectifs essentiels que le volume des documents et la multiplicité des langues utilisées assignent à la traduction :

- *Extraire l'information pertinente* : les scientifiques ne pourront jamais plus consacrer suffisamment de temps à la lecture de la littérature spécialisée qui devient trop riche. De plus, il est risqué d'investir un temps précieux pour déchiffrer un texte rédigé dans une langue étrangère que l'on ne maîtrise pas, surtout s'il n'offre en fin de compte que peu d'intérêt.

Les ouvrages scientifiques sont très souvent rédigés en anglais mais la répartition croissante de la technologie dans le monde entame la suprématie d'une langue qui reste tout de même essentielle<sup>2</sup>.

En livrant une traduction de faible qualité, un système de traduction rapide et peu coûteux pourrait souvent suffire, dans un premier temps, à l'acquisition des textes concernant le domaine d'activité du lecteur. Dans le pire des cas, ce dernier serait en mesure d'évaluer la nécessité d'un travail plus soigné, par conséquent, plus onéreux. L'expérience montre qu'une traduction juste mais de mauvaise qualité garantit la compréhension du contenu et n'implique pas d'opération supplémentaire. Précisons que le type de texte obtenu (morphologie et syntaxe imparfaites) illustre la liaison en quelque sorte lâche entre correction grammaticale et intelligibilité. Nous développerons cette idée dans le paragraphe 3.2.

(1) R. PETRELLA : Directeur de l'enseignement, de la culture et du sport, Conseil de l'Europe

(2) G. KINGSCOTT : "*Applications of Machine Translation*". Study for the Commission of the European Communities, septembre 1989, Praetorius Limited, pp. 17-18

- *Assurer la propagation de l'information* : L'industrie, par exemple, qui désire exporter ses produits, doit en général les accompagner d'une documentation rédigée dans la langue de l'acheteur. Les Etats-Unis, dans le passé, ont souvent ignoré cette obligation. Ils commencent à modifier leur politique. D'autres pays, comme l'Allemagne, n'ont pas pu s'offrir ce luxe. La traduction est une pratique qui se répand d'autant plus que les économies s'ouvrent sur l'extérieur. La recherche des marchés étrangers implique, entre autres, une bonne pratique des langues concernées. La difficulté de trouver des traducteurs techniques qualifiés, capables de réaliser rapidement des traductions justes et claires, a suscité un regain d'intérêt pour la traduction automatique, regain dont on peut mesurer l'ampleur lors des grandes manifestations consacrées à la traduction et à ses professionnels<sup>1</sup>.

## La traduction automatique

Aux deux formes de traduction évoquées plus haut, correspondent deux démarches économiques distinctes :

- Confrontée au problème du prix (le prix de fabrication d'un livre doit être environ le sixième de son prix public de vente hors-tax pour qu'un éditeur ne perde pas d'argent en l'éditant)<sup>2</sup> et au problème de la gestion des illustrations (le volume de l'iconographie reste constant, alors que le texte peut augmenter de 15 à 20 %), la traduction littéraire est peu rentable. L'éditeur reste cependant libre de ne pas engager la dépense !
- En ce qui concerne la traduction utilitaire, elle s'impose pour les raisons que nous avons indiquées dans le paragraphe précédent. Les besoins sont à la mesure de l'accroissement des échanges commerciaux, scientifiques, industriels, culturels et diplomatiques<sup>3</sup>.

(1) Parmi les nombreuses manifestations :

- International Conference on the Methodology and Techniques of Machine Translation, Cranfield Institute of Technology, England
- ICCL (International Conference on Computational Linguistics), COLING (Pise/1973, Ottawa/1976, Bergen/1978, Tokyo/1980, Prague/1982, Stanford/1984, Bonn/1986)
- ISSCO Tutorial on Machine Translation (Lugano)
- Annual Meeting of the ACL (Association for Computational Linguistics)
- International Colloquium on Machine Translation, Lexicography and Analysis
- Conference on Theoretical and Methodological Issues in Machine Translation of Natural Languages
- Symposium of the Use of Computers in Literary and Linguistic Research
- FBIS (Foreign Broadcast Information Service, Washington DC) Seminar on Machine Translation
- Annual Convention of the Information Processing Society of Japan
- ASIS (American Society for Information Science) Annual Meeting
- IJCAI International Joint Conference on Artificial Intelligence
- ACL-NRL Conference on Applied Natural Language Processing (Association for Computational Linguistics)

(2) F. GUERARD : Responsable des dictionnaires et industries de la langue chez Hachette

(3) Lors de la conférence IFTT à Tokyo en avril 1989, L. ROLLING (DGXIII) a évalué le marché mondial de la traduction à 200 millions de pages/an (traductions effectuées par des bureaux de traduction pour le compte de grandes entreprises).

Dans "Machine Translation Technology : on the way to market introduction" in Siemens Review, vol.54, n°6, nov./dec. 1987, J.A. ALONSO et T. SCHNEIDER parlent de 100 millions de pages pour l'Europe de l'Ouest en 1986.

Le Translation Bureau of the Canadian Secretary of State traduit 120 000 pages /an, dont 80% de l'anglais vers le français et vice versa. DATAQUEST cite, pour 1986, entre 500 millions et 4,5 milliards de \$ selon les secteurs géographiques concernés. COOPERS et LYBRAND (Canada, septembre 1987) situent la marché mondial entre 20 et 60 millions de mots (6-18 milliards de FF).

La CEE avance le chiffre de 5,6 milliards de \$

Le volume des traductions techniques est écrasant<sup>1</sup>. Il suppose un effort considérable pour la saisie, la maintenance et l'utilisation cohérente d'une terminologie convenable.

SECTEUR	MILLIONS DE PAGE/AN	TAUX DE CROISSANCE
commercial-industriel	200	15 %
gouvernemental	100	10 %
scientifique	70	13 %
juridique	35	9 %
divers	60	3-5 %
EUROPE		en M de % US
Allemagne de l'Ouest		1000
Angleterre		650
France		500
Belgique		180
Espagne		200
Italie		140
Suisse		110
Danemark		40
AMERIQUE		en M de % US
E.U.A.		2000
Canada		250
Amérique du Sud		500
Pays francophones (excluant ceux mentionnés)		60

Observatoire des industries de la langue : Actes du séminaire TAO/mars 1989

Si la traduction était par tradition une activité artisanale, elle doit s'adapter à l'évolution du marché. Il devient alors indispensable d'augmenter la productivité du traducteur. L'informatique représente un espoir certain de solution, dans un domaine où la qualité réside bien plus dans la fidélité que dans le style, ce qui est justement l'apanage des systèmes de TALN. L'intérêt est donc grand pour ces applications qui, de la Traduction Automatique (T.A.) et de la Traduction Assistée par Ordinateur (T.A.O.) aux banques de données terminologiques, consacrent le mariage de l'informatique et de la linguistique.

### Informatique, linguistique et industrie des langues

L'informatique a d'abord automatisé des processus simples mais les capacités de stockage et de calcul ont augmenté sans cesse pour atteindre un niveau de puissance

(1) - Pour M. BORIS dans "Le marché mondial de la traduction" in Décision, n° 246, octobre 1989, p. 66, le marché réel de la France avoisinerait 1,1 milliard de FF et son marché potentiel 1,6 milliard de FF.

- Dans une étude du JEIDA (Japan Electronics Industry Development Association) publiée à l'occasion du MTII (Machine Translation) à Munich en août 1989, le volume total des traductions réalisées au Japon en 1988 est estimé à 200 millions de pages : 65 millions de pages dans les bureaux de traduction, 134 millions de pages dans les entreprises et les administrations. 90% de l'ensemble sont des traductions de l'anglais vers le japonais et vice versa, ce qui représente un montant de 40 milliards de FF pour 1988.

qui suscite d'immenses efforts de recherche. Admirant les performances extraordinaires de l'homme, les techniciens s'intéressent aux sciences humaines et se penchent naturellement sur la manipulation de la langue écrite et orale.

Cet intérêt pour le traitement automatique des langues a donné naissance à une branche particulière de la linguistique. Les recherches en cryptographie imposées par la Seconde Guerre mondiale en sont à l'origine. Elles ont montré que certaines manipulations statistiques dégageaient des propriétés aidant à comprendre des messages en langue naturelle. Certains firent alors le pari très audacieux que l'ordinateur pourrait conduire à la traduction automatique (Memorandum WEAWER)<sup>1</sup>. On s'est aperçu qu'il n'était pas facile de soumettre la langue à une transformation de son matériau pour l'inclure dans des usages automatisables. Les problèmes rencontrés ont stimulé les recherches en linguistique. Une partie des théories et des applications qui en découlent sera l'objet du chapitre I.

- Le traitement de la langue par les machines revêt des formes diverses qui ne peuvent cependant reproduire les compétences langagières de l'homme dans leur étendue, leur raffinement et leur complexité. Les machines chargées de traiter l'information, doivent être capables de manipuler, gérer, conserver, générer et comprendre le langage humain en fonction de structures propres. En dépit des perspectives fascinantes mais à très long terme qu'ouvrent l'Intelligence Artificielle et les Sciences Cognitives, il importe de considérer avant tout les fonctionnalités des traitements linguistiques. Dans le contexte actuel, l'univers techno-économique se développe à un rythme très rapide et appelle une automatisation partielle du premier moyen de communication humain et des applications qui en découlent.

C'est dans ce contexte que l'informatique et la linguistique se rejoignent, on parle d'informatique linguistique, d'automatique linguistique, de linguistique computationnelle. Ces termes voisins ont un objet commun : comprendre et représenter le fonctionnement de la langue parlée et écrite pour en permettre l'interprétation et la génération par des automates. Ce phénomène de convergence concerne également d'autres secteurs scientifiques, l'ergonomie informatique, la logique appliquée, etc. Ces travaux de recherche et de développement pluridisciplinaires ont pris une telle importance qu'ils engendrent une véritable industrie, l'industrie des langues.

Les résultats de la recherche en informatique linguistique trouvent en effet une application parfois lucrative dans un ensemble de produits dérivés qui alimentent une nouvelle industrie.

Les champs privilégiés sont :

- L'élaboration des systèmes de dialogue homme-machine au service des automates qui utilisent la langue écrite ou orale pour communiquer avec l'homme.
- La construction d'interfaces en langage naturel pour assurer la "convivialité" des systèmes experts, des systèmes d'E.A.O. (Enseignement Assisté par Ordinateur) et des S.G.B.D. (Systèmes de Gestion de Base de Données).
- Le développement, dans le secteur tertiaire, de tous les outils capables d'automatiser le plus complètement la manipulation de la langue en multipliant les capacités à rédiger, traduire, archiver, consulter, communiquer, en bref, les techniques qui transformeront la bureautique en bureautique "intelligente".

---

(1) W. WEAVER : in *Machine Translation of Languages*, W.N. LOCKE, A.D. BOOTH (eds), MIT Press, 1955

Des grands secteurs comme les banques et les compagnies d'assurances investissent déjà dans ce type de recherche. Il faut cependant espérer que l'industrie restera consciente des énormes difficultés dont nous pensons qu'elles ne pourront jamais être surmontées à 100%, et que, renonçant à des systèmes totalement automatisés, elle saura se fixer des objectifs moins ambitieux. Les analyseurs morphologiques et syntaxiques ont beaucoup progressé. Encore existe-t-il des cas où les connaissances linguistiques ne suffisent pas à la machine pour mener une analyse à terme et lever toutes les ambiguïtés. On sait depuis longtemps qu'elle aurait besoin, pour ce faire, de connaissances particulières concernant notre univers (les données pragmatiques) et le domaine traité.

## Compréhension automatique et Automate de compréhension Implicite (ACI)

Il est évident que la machine ne peut traduire sans comprendre. Aussi tenterons nous de définir les connaissances nécessaires à la compréhension avant de développer les méthodes utilisées couramment pour les représenter, les organiser et les utiliser (chapitre I). Les nombreuses applications que nous évoquons dans le chapitre II illustrent l'évolution exemplaire de la Traduction Automatique. La leçon tirée des premiers échecs est précisément qu'il est impossible de traduire sans comprendre. Le poids de la sémantique augmente alors et les résultats gagnent en qualité. Il est cependant de plus en plus difficile d'améliorer la performance des systèmes dont on cerne mieux les limites.

L'histoire déjà riche de la T.A. et de la T.A.O. montre que la complexité et le coût des systèmes croissent plus vite que les gains de qualité. Ceci étant, peut-on aborder le problème dans l'autre sens en recherchant les moyens d'ajuster les résultats aux systèmes possibles, techniquement et financièrement ? En d'autres termes, jusqu'où peut-on se résigner à faire moins bien, tout en "en ayant pour son argent" ? D'autres voies ont été exploitées avec les systèmes à sémantique fermée et les systèmes d'aide à la traduction qui constituent une alternative classique à la Traduction Automatique. Nous avons choisi de nous intéresser à la réduction de la correction du résultat, qui ne garantit plus que l'équivalence cognitive (et encore !), en sacrifiant la correction morphologique et grammaticale. Autrement dit : nous allons tenter de cerner les moyens et donc les coûts d'un "petit nègre intelligible" ?

C'est en privilégiant la compréhension implicite et les parentés étymologiques des langues européennes que nous avons voulu construire un système d'interprétation automatique (et non pas traduction) de textes non-littéraires. L'horizon de ces recherches est ambitieux. Il s'agit d'extraire d'un texte technique, rédigé dans une langue étrangère, des informations suffisantes pour que le lecteur, spécialiste du domaine traité mais ne connaissant pas la langue source, puisse en reconstituer le contenu. Nous nous écartons légèrement de la Traduction Automatique et de ses écueils. Nous montrerons dans le chapitre III qu'il est possible de reconnaître les éléments nécessaires à ce type de compréhension et que le processus est automatisable.

L'ensemble du travail repose sur deux notions essentielles :

- **La compréhension implicite** : Distinguant *compréhension explicite* et *compréhension implicite*, nous assignons à notre système une fonction de compréhension implicite qui consiste à extraire une information suffisante, à partir d'un texte donné et par des procédés automatiques, pour que tout lecteur ayant une maîtrise du domaine concerné, puisse en déduire une description exacte de ce dont on parle. C'est la compétence du lecteur qui "ferme" la sémantique. La maîtrise du *système dérivationnel de la langue* permettra d'appréhender les racines et leurs modificateurs pour construire le "*spectre sémantique*" du texte analysé et distinguer deux sous-ensembles, l'ossature du texte pratiquement indépendante du sujet traité et les "objets" représentant ce dont parle le texte. Pour que la relation entre le spectre sémantique et la compréhension implicite devienne évidente, il

faut considérer le *spectre sémantique discursif* (obtenu à partir d'un grand nombre de textes non-littéraires) qui donne une idée précise du "noyau prédicatif" de la langue et du système dérivationnel qui lui est associé. Compte tenu de sa stabilité, ce spectre correspond à une "sémantique fermée". D'un autre côté, on doit lui associer le spectre instable qui correspond à tout texte et dont la sémantique est "ouverte" (racines nominales...). En respectant cette opposition "sémantique ouverte"/"sémantique fermée", des systèmes automatiques indépendants mais compatibles (*le module prédicatif et le module hypersémantique*) traiteront les deux types de données.

- **Les parentés étymologiques des langues européennes** : Constatant que le vocabulaire de toute langue est illimité et qu'il devient par conséquent très difficile d'en concevoir la gestion informatique, nous verrons qu'il est possible de le réduire à une combinatoire. Cette réduction s'appuie en partie sur une décomposition des chaînes de caractères conforme aux mécanismes de formation des mots et révèle leur structure étymologique. L'hypothèse de D. HERAULT est que les langues européennes constituent leur vocabulaire en effectuant de nombreux emprunts par reconstruction. Les règles de correspondance des combinatoires sont restées stables de sorte qu'il est possible de générer des mots compréhensibles dans la langue B à partir des structures étymologiques de la langue A, sur la base d'un système de correspondances souvent bi-univoques entre les ensembles de mots simples et les affixes des deux langues. Associées à d'autres résultats, ces reconstructions, plus souvent compréhensibles que correctes, permettront d'obtenir une *interprétation* plutôt qu'une traduction, la correction formelle de la sortie posant le problème des coûts. C'est la correspondance entre les ensembles finis d'éléments des différentes langues, et la nature des programmes d'analyse, indépendants des données, qui confèrent au système ses propriétés *multilinguales*.

Dans le chapitre III, nous situons les travaux du Centre de Recherche Jean Favard dans leur contexte. Après une description des principes fondamentaux de l'Automate de Compréhension Implicite, nous indiquons rapidement ce que doit être sa structure générale, et présentons l'automate germanique-roman, l'objet de notre thèse étant la réalisation des modules d'analyse et leurs résultats.

Dans le chapitre IV, nous détaillerons tous les programmes qui traitent le verbe, l'adjectif, la décomposition des mots et l'analyse syntaxique. Nous joignons l'ensemble des résultats obtenus sur un texte de 149 phrases : "*Brennstoffzellen-Kraftwerke*".

Le cahier des charges tenait compte des moyens disponibles, humains et financiers. On pourrait parler d'un sain réalisme. Il n'en reste pas moins vrai que si les industriels du TALN reculent souvent devant l'ampleur des investissements financiers et humains à réaliser pour des applications plus ambitieuses, les créneaux qu'ils occupent ont assuré un chiffre d'affaires mondial de 55 millions de dollars pour 1985, et qui progresse, depuis lors, de 100% par an<sup>1</sup>. Cette actualité rehausse l'intérêt de l'ACI (Automate de Compréhension Implicite) dont nous rappellerons les avantages, en conclusion. Le module d'interprétation n'étant pas disponible, nous ne proposerons qu'une simulation de son fonctionnement. Quelques exemples d'applications originales en E.A.O. (Enseignement Assisté par Ordinateur) illustreront une partie des résultats obtenus.

---

(1) A. ABBOU, I. LEFAUCHEUR, T. MEYER : *Les industries de la langue. Les applications industrielles du traitement de la langue par les machines*. (2 volumes), Editions DAICADIF, Paris, septembre 1987

I.

**LE TRAITEMENT AUTOMATIQUE  
DES LANGUES NATURELLES**

Dans ce chapitre, nous aborderons le problème du sens et de la compréhension (1.1) et recenserons, en respectant grossièrement la chronologie de leur apparition, les particularités qui rendent le traitement automatique de la langue si difficile (1.2). Ceci nous permettra de définir les connaissances qu'elle requiert (1.3). Pour réaliser des systèmes de traitement automatique, il convient de choisir des formalismes de représentation pour les connaissances qu'utiliseront les ordinateurs. Nous serons donc amenés à présenter les formalismes grammaticaux (les modèles linguistiques, 1.4) et les mécanismes informatiques responsables du processus de compréhension (les outils, 1.5) en les illustrant par quelques exemples.

### 1.1 Sens et compréhension

Au coeur des problèmes que doit résoudre l'ordinateur, et pour le domaine qui nous intéresse, figurent le sens et la compréhension.

*"Que n'a-t-on tenté, pour éviter, ignorer ou expulser le sens ? On aura beau faire, cette tête de méduse est toujours au centre de la langue, fascinant ceux qui la contemplent".* Comme le laisse entendre E. BENVENISTE<sup>1</sup>, les opérations qu'accomplit l'"énonceur" pour produire et interpréter le sens, sont complexes et très mal connues. Les langues sont d'une diversité typologique considérable et les mécanismes selon lesquels se déploie le sens sont reliés à l'inconscient, inaccessible à l'analyse directe.

---

(1) E. BENVENISTE : "Les niveaux de l'analyse linguistique" in Problèmes de linguistique générale 1, Paris, 1966, Gallimard, Collection Tel, p. 126

- Que signifie "comprendre" ? : Pour l'homme, nous sommes loin d'avoir trouvé une réponse précise. Le développement des Sciences Cognitives est récent. Pour la machine, le problème est encore plus complexe. Il faut de plus pouvoir montrer qu'elle a compris ! La compréhension s'effectue à plusieurs niveaux : le mot, la phrase, le texte et le contexte. Le sens d'une phrase, en effet, n'est pas la juxtaposition des sens de chaque mot, celui d'un texte ne peut être réduit à la somme des sens de chaque phrase. Qu'il s'agisse du mot, de la phrase ou du texte, il est parfois impossible de comprendre sans connaître le contexte.

- Comment comprendre que quelque chose n'a pas de sens ? : Nous pensons à des unités lexicales inconnues (mauvaise orthographe, faute de frappe, néologisme...) ou à des constructions syntaxiques inédites. S'il est difficile de définir cette notion, il nous semble par contre plus simple d'en étudier quelques caractéristiques, reflétant chacune un degré divers de compréhension. Voici quelques critères :

- La capacité de répondre à une question, de façon appropriée.

ex. - *Paul est-il le frère de Michel ?*

- *Non.* (réponse appropriée)

- *Non, c'est Jean.* (réponse plus appropriée)

- *Non. Paul est le frère de Julien.*

- La capacité de paraphraser, d'exprimer autrement une idée.

- La capacité de faire des inférences en tirant des conséquences probables de ce qui a été dit.

- La capacité de traduire d'une langue à l'autre, ce qui suppose une certaine capacité d'abstraction reflétant la possibilité de représenter à un niveau sémantique ou conceptuel les différentes relations causales ou de dépendance entre les constituants d'une phrase ou d'un texte, en tenant le moins compte possible de la "forme de surface".

- La capacité de résoudre des problèmes de référence. Une part importante de nos capacités de compréhension consiste à réaliser qu'un même objet ou un même personnage apparaît sous plusieurs formes (pronominalisation...).

- La réussite du test de TURING<sup>1</sup> : un opérateur communique tantôt avec un ordinateur, tantôt avec un être humain, sans connaître l'identité du destinataire. S'il ne peut reconnaître l'ordinateur au bout d'un moment, d'après les réponses, c'est que le programme est "intelligent".

Le rejet du sens a été poussé à son stade ultime par le structuralisme américain<sup>2</sup>. On conçoit dès lors les espoirs insensés placés dans le développement d'une Traduction Automatique asémantique. Les ennuis sont apparus dès les premiers travaux.

## 1.2 Les obstacles à la compréhension

La compréhension automatique se heurte à un ensemble de difficultés considérables, liées essentiellement aux propriétés des Langues Naturelles. Nous allons les passer en revue, selon une chronologie approximative de leur apparition.

---

(1) Ce thème est repris dans D. HOFSTADTER : *Gödel, Escher, Bach, les Brins d'une Guirlande Éternelle*, Interéditions, 1988, Paris, traduit de l'américain par J. HENRY et R. French

(2) L. BLOOMFIELD : *Language*, New York, 1967, (Traduction française, 1970, Payot, Paris)

## 1946-1955 : La morphologie et les statistiques -> le mot

Dès l'apparition de l'ordinateur, les recherches de W. WEAVER<sup>1</sup> aux Etats-Unis et de W. N. LOCKE en Angleterre accèdent à la faisabilité d'une analyse morphologique automatique qui pourrait mener à la traduction, pour autant que des traitements statistiques permettent d'accéder à un niveau sémantique suffisant. La première conférence sur la Traduction Automatique est organisée par BAR-HILLEL au MIT en 1952. L'optimisme lié aux nombreux travaux sur la recherche dans les dictionnaires sera vite déçu. Considérer la traduction comme une substitution de mots et un éventuel réordonnement est une vision trop simpliste qui oublie les traits essentiels que partagent toutes les langues sous une forme ou sous une autre.

Ces caractéristiques sont telles qu'il n'y a pas de correspondance précise entre l'ensemble des mots ou des phrases et l'ensemble des sens. Au niveau des mots, on peut citer plusieurs écueils :

- L'homonymie : c'est la relation qui existe entre deux ou plusieurs formes ayant le même signifiant, mais des signifiés différents. On distingue les homophones (même son, sens différents) qui ne nous concernent pas ici et les homographes (même graphie, sens différents).

- La polysémie : Le signifiant présente plusieurs signifiés. Les signifiants différents sont perçus comme présentant des traits sémantiques communs.

Homonymie et polysémie sont à l'origine des nombreuses difficultés que le traitement automatique regroupe sous le terme d'*ambiguïtés*.

ex :        *les poules du couvent couvent*  
              *il est uni*

Se fondant sur des exemples tels que :

*The box is in the pen* (la boîte est dans le parc)

*The pen is in the box* (le stylo est dans la boîte)

BAR-HILLEL<sup>2</sup> montrera que des connaissances sur le contexte sont indispensables. Il pose ainsi la question fondamentale de la représentation des connaissances et de leur utilisation et le considère comme insoluble.

Au vu des résultats obtenus, le Rapport ALPAC<sup>3</sup> condamne brutalement les recherches. Il convient cependant de citer les travaux de Mc CARTHY, MINSKY, NEWELL et SIMON qui pensent que l'on pourra écrire des programmes "intelligents" dans la mesure où il sera possible de décrire précisément les aspects de l'intelligence pour qu'une machine puisse la simuler. Tandis que les travaux sur la Traduction Automatique sont pratiquement stoppés, les chercheurs du MIT développent des systèmes (SIR, BASEBALL, STUDENT, ELIZA...) qui relancent les recherches sur la compréhension automatique du langage. Les domaines traités sont très limités et la syntaxe inexistante.

## 1955-1970 : La syntaxe -> la phrase

De son côté, N. CHOMSKY<sup>4</sup> développe la théorie des grammaires formelles qui débouche sur des traitements purements syntaxiques. Les évolutions sont nombreuses mais les interprétations syntaxiques possibles sont trop riches pour qu'elles se prêtent

---

(1) W. WEAVER : "Translation" in W.N. LOCKE, A.D. BOOTH (eds.), *Machine Translation of Languages*, Wiley, New York, pp. 15-23, 1955

(2) Y. BAR-HILLEL : *Language and Information*, Addison Wesley, Reading Mass., 1964

(3) ALPAC : *Language and Machines ; Computers in Translation and Linguistics*, Report by the Automatic Language Processing Advisory National Research Council, National Academy of Sciences, Publication 1416, Washington D.C., 1966

(4) N. CHOMSKY : *Syntactic Structures*, Mouton, La Hague, 1957

au traitement automatique. S. PETRICK<sup>1</sup> cite le cas d'une phrase de 17 mots donnant 572 interprétations syntaxiques possibles.

Les phénomènes que nous rencontrons au niveau des mots se retrouvent ici, au niveau de la phrase :

- L'allotaxie : les formes syntaxiques différentes peuvent s'interpréter de la même façon (une phrase à l'actif a le même sens qu'au passif)

- l'homotaxie : la même forme syntaxique peut avoir des interprétations différentes. La maîtrise de la structure du langage est essentielle. A une période particulièrement féconde succède cependant le désenchantement des chercheurs qui ont placé trop d'espoir dans la recherche syntaxique et surestimé son poids.

Dans la deuxième moitié de cette période, on peut citer, en ce qui concerne les univers limités, la logique des prédicats du premier ordre qui permet d'interpréter la structure syntaxique et donne de bons résultats :

- programme de dialogue avec un robot, de L. S. COLES<sup>2</sup>  
- travaux de T. WINOGRAD<sup>3</sup>, et, plus récemment, de A. COLMERAUER<sup>4</sup> et P. SABATIER<sup>5</sup> avec Prolog.

### 1968 : La sémantique et le contexte

De nombreux systèmes voient le jour, qui éludent les difficultés en limitant les domaines et en réduisant la syntaxe à sa plus simple expression. De nouvelles équipes décident alors d'étudier les aspects sémantiques. Il faut citer les grammaires de cas de FILLMORE<sup>6</sup> qui fondent l'analyse de la phrase sur le sens des rapports verbe-objet plutôt que sur la structure même. QUILLIAN<sup>7</sup> propose de représenter le sens des mots et des phrases à l'aide des réseaux sémantiques. Les approches sont ensuite essentiellement sémantiques (SHANK<sup>8</sup>, WILKS<sup>9</sup>), le rôle de la syntaxe est réduit.

Les limites des programmes alors conçus résident dans les difficultés à relier les phrases d'un texte. On découvre l'importance du contexte et les avantages que procurerait une bonne connaissance du domaine traité. Les problèmes sont :

- 
- (1) S. PETRICK : "Transformational Analysis", in *Natural Language Processing*, Rustin, New York, Academic Press, pp. 27-41
  - (2) L.S. COLES : "An On-line Question-Answering System with Natural Language and Pictorial Input"; *Proc. of the 23rd A.C.M. National Conference*, 1968
  - (3) T. WINOGRAD : "Procedures as a Representation for Data in a Computer Program for Understanding Natural Language", *Cognitive Psychology*, 3(1), pp. 1-191, Jan. 1972,
  - (4) A. COLMERAUER : "Les grammaires de métamorphose" in L. BOLC (ed.) ; *Natural Language Communication with Computer : Lecture notes in Computer Science*, n°63, Springer Verlag, pp. 133-189, 1978
  - (5) P. SABATIER : "Puzzle Grammars" in *Prog. Log. 84*, Actes du colloque "Compréhension du langage naturel et programmation en logique", Rennes, sept. 1984, North Holland, 1985
  - (6) C. FILLMORE : "The Case for Case", in *Universals in Linguistic Theory*, Bach & Harms, Chicago, Holt, Rinehart and Winston, 1968, pp. 1-90
  - (7) R. QUILLIAN : "Semantic Memory", in *Semantic Information Processing*, Minsky, MIT Press, Cambridge Mass, 1968, pp. 227-270
  - (8) R. SCHANK : "Conceptual Dependency : a Theory of Natural Language Understanding", *Cognitive Psychology*, Vol. 3 (4), pp. 552-631
  - (9) Y. WILKS : "An artificial Intelligence Approach to Machine Translation", in *Computer Models of Thought and Language*, Schank & Colby, Freeman, San Francisco, 1973, pp. 114-151

- L'anaphore : c'est la relation entre une forme et une forme à laquelle on renvoie dans le discours. (le pronom - *Le juge se lève, il est calme* -), (la référence définie - *le juge se lève, cet homme calme...* -)

- L'implicite : C'est la partie du message que l'auditeur peut reconstituer. Les recherches qui traitent ce domaine se multiplient en 1974. La logique du premier ordre n'est pas adaptée au traitement des univers complexes. C'est la représentation des connaissances qui focalise les énergies et suscite l'intérêt des informaticiens pour les psychologues. Les recherches intègrent la psychologie cognitive et l'informatique. R.C. SCHANK<sup>1,2,3</sup> identifie les connaissances nécessaires dans un système de compréhension du langage, M. MINSKY<sup>4</sup> propose un cadre général de représentation des connaissances avec les "frames". On peut noter l'apparition d'autres disciplines, telles la neurologie (N. GESCHWIND<sup>5</sup>), la philosophie et la métaphysique (D.C. DENNET<sup>6</sup>, S. TORRANCE<sup>7</sup>).

Les premières tentatives de Traduction Automatique ont échoué parce qu'elles ne reposaient pas sur la compréhension du texte et manipulaient les mots du langage comme n'importe quel autre symbole, autrement dit, sans tenir compte du fait que le langage est un outil de communication entre êtres humains, destiné à véhiculer certaines idées. A la difficulté d'incorporer des dictionnaires complets des langues à traduire, (le russe et l'anglais, initialement), se sont ajoutés les problèmes de polysémie et de syntaxe propres à chaque langue. Les premiers résultats étaient loin du mot à mot, le programme ne proposant qu'une liste de mots possibles. Si Y. BAR HILLEL<sup>8</sup> montre alors que la traduction a besoin de connaissances encyclopédiques, il faut se rappeler qu'à l'époque, on ne savait pas encore comment implanter une grammaire dans un ordinateur, que les systèmes étaient rédigés en langage machine et que la récursivité n'était pas autorisée par la majorité des langages de programmation. Partant du fait qu'il était impossible de traduire sans comprendre, on cherche alors les moyens de représenter la signification d'une phrase ou d'un texte, et de lever les ambiguïtés possibles de certains mots, ce qui nécessite parfois d'utiliser le contexte et de disposer peut-être d'un modèle du monde.

Deux approches illustrent ces préoccupations :

- Le texte d'entrée est traduit en une structure arborescente qui représente la construction syntaxique de la phrase, augmentée éventuellement de quelques informations sémantiques extraites d'un dictionnaire. Cette structure est ensuite manipulée par une

---

(1) R.C. SCHANK (ed.) : *Conceptual Information Processing* ; North Holland, 1975

(2) R.C. SCHANK, R.P. ABELSON : *Scripts, Plans, Goals and Understanding*, Lawrence Erlbaum Ass., 1977

(3) R.C. SCHANK : "Reminding and Memory Organisation : an Introduction to MOP's" in W. LEHNERT, M. RINGLE (eds.), *Strategies for Natural Language Processing*, Lawrence Erlbaum Ass., 1982

(4) M. MINSKY : "A Framework for Representing Knowledge" in P.H. WINSTON (ed.), *The psychology of Computer Vision*, Mc Graw Hill, pp. 211-277, 1975

(5) N. GESCHWIND : "Neurological Knowledge and Complex Behaviour" in D.A. NORMAN (ed.), *Perspectives on Cognitive Science*, Lawrence Erlbaum Ass., 1981

(6) D.C. DENNET : *Brainstorm : Philosophical Essays on Mind and Psychology*, Harvester Press, 1979

(7) S. TORRANCE (ed.) : *The Mind and the Machine : Philosophical Aspects of Artificial Intelligence*, Ellis Horwood, 1984

(8) Y. BAR HILLEL : *Language and Information*, Addison Wesley, Reading Mass, 1964

grammaire de transfert<sup>1,2</sup> pour générer une phrase syntaxiquement correcte dans la langue cible.

- La seconde approche accorde moins d'intérêt à la syntaxe. Elle vise en revanche une représentation conceptuelle des phrases beaucoup plus fine et implique une compréhension plus profonde du texte à traduire (Y. WILKS<sup>3</sup> et R. SCHANK).

### 1.3 Les connaissances requises pour la compréhension

Si l'on considère les divers types de connaissances qui interviennent dans les mécanismes de compréhension, on reconnaît les niveaux classiques de l'analyse avec :

- Le vocabulaire et sa structure (lexique et morphologie)
- L'ordre des mots dans la phrase (syntaxe)
- Les aspects sémantiques
- La structure du discours, les relations entre phrases
- Les données pragmatiques

Il est clair qu'une bonne compréhension dépendra de la maîtrise des éléments que nous venons d'énumérer et que ces éléments ne seront pas seulement utilisés pour l'analyse.

1. Les connaissances morphologiques : Les mots apparaissent dans la phrase sous des formes diverses (accords grammaticaux, conjugaison...). L'analyse doit reconnaître qu'il s'agit d'un même mot. Deux méthodes sont possibles.

- On relève toutes les formes dans le dictionnaire, en répétant certaines informations, (ex. petit, petite, petits, petites). La taille des fichiers devient rédhibitoire pour les langues riches en flexions à moins de limiter les entrées en leur associant un module de génération des formes fléchies.
- La reconstruction de la forme de référence à chaque occurrence du mot.

Dans les deux cas, il faudra utiliser la forme "canonique" du mot (infinitif pour le verbe, masculin singulier pour le nom, l'adjectif...), disposer du système dérivationnel de la langue et des règles qui régissent les combinaisons qu'il engendre. Pour la phrase *le secrétaire vole des livres*, nous obtenons les résultats suivants :

<i>le</i>	article défini masculin singulier, pronom
<i>secrétaire</i>	nom masculin singulier
<i>vole</i>	voler (transitif, 1ère et 3ème personne présent indicatif, ....)
	voler (intransitif, 1ère et 3ème personne présent indicatif, ....)
<i>des</i>	article indéfini (masculin/féminin pluriel), article contracté,...
<i>livres</i>	nom masculin pluriel, nom féminin pluriel

A ce niveau, les ambiguïtés sont nombreuses.

(1) R. KITTREDGE, L. BOURBEAU, P. ISABELLE : "Design and Implementation of an English-French Transfer Grammar" in Coling 76, Ottawa

(2) C. BOITET : "Problèmes actuels en T.A. : un essai de réponse" in Coling 76, Ottawa

(3) Y. WILKS : "An Artificial Intelligence Approach to Machine Translation" in *Computer Models of Thought and Language*, SCHANK et COLBY (Eds.), San Francisco, 1973, Freeman, PP. 114-151

**2. Les connaissances syntaxiques :** Une fois que les entrées dans le dictionnaire ont été reconnues, il faut éclaircir leur agencement dans la phrase, distinguer le ou les sujets, les compléments, le groupe verbal... Ces connaissances qui décrivent des relations formelles entre des catégories de mot, sans tenir compte de leur valeur sémantique constituent la grammaire.

Reprenons l'exemple précédent. L'application des règles de grammaire va nous indiquer, sans tenir compte du sens, que la phrase est correcte si *le* est un article et *vole* transitif. Cette étape nous aura donc permis de réduire le nombre des ambiguïtés.

**3. Les connaissances sémantiques :** Une phrase peut être correcte mais n'avoir aucun sens. Nous avons besoin des connaissances sémantiques qui précisent les relations entre les mots de la langue et le monde (réel ou spécifique d'un domaine traité), sans référence au narrateur. Ce sont les aspects explicites du sens. On dira par exemple que *boire* fait intervenir un agent et un objet, que l'agent peut être animé ou inanimé, que l'objet est liquide.....

Pour notre phrase, le *secrétaire* est un employé ou un rapace, *livres* est une mesure de poids, une monnaie anglaise ou un objet de lecture. *Voler* (transitif) implique un objet concret, ce qui permet d'éliminer *livre* mesure de poids. La levée de cette ambiguïté permet de préciser la nature de *des*.

Il reste quatre possibilités. Cependant, il faut préciser que le nombre d'interprétations dépend des traits retenus par le système. Dans le traitement d'un domaine très spécifique, il serait inutile de conserver *secrétaire* objet inanimé... Il n'en reste pas moins qu'à ce stade de l'"analyse", un homme ou un animal ont dérobé un livre ou des billets de banque.

**4. Les connaissances pragmatiques :** Nous avons maintenant besoin d'informations sur la situation évoquée, pour opérer les choix conformes au contexte et aux réalités. Ce sont les connaissances pragmatiques qui expriment les aspects implicites du sens. La compréhension dépend de la situation de communication et de la façon dont l'énoncé s'intègre dans notre représentation du monde. On retrouve la composante pragmatico-interprétative (influence de la situation sur le sens pour obtenir la signification) que R. MARTIN<sup>1</sup> ajoute à la composante sémantico-logique (sens littéral). H.P. GRICE<sup>2</sup> définit également la signification globale d'un énoncé comme une combinaison du sens des mots qui le composent (sens lexical), avec des implications conventionnelles (pragmatique intentionnelle) et des implications non conventionnelles.

Nous venons d'évoquer les connaissances nécessaires à la compréhension d'un énoncé, partant du principe que cette compréhension était préalable à l'opération de traduction. C'est cette optique qui explique les nombreuses incursions du chapitre dans le domaine de l'Intelligence Artificielle. Pour construire des systèmes de Traduction Automatique ou de Traduction Assistée par Ordinateur basés sur la compréhension, les chercheurs doivent effectuer des choix à différents niveaux. Nous rappelons ces niveaux et les possibilités qui lui correspondent.

- Comme nous venons de le voir dans 1.3, il faut disposer de connaissances externes par rapport aux textes traités. Ces connaissances devront être formalisées pour que l'ordinateur puisse les manipuler. Il reste à choisir le système formel qui sera utilisé.

---

(1) R. MARTIN : *Inférence, antonymie et paraphrase*, Paris, Klincksieck, 1976

(2) H.P. GRICE : *Utterer's Meaning, Sentence Meaning and Word Meaning*, Foundations of Language, 4, 1968, pp. 225-242

- Qu'il s'agisse de l'étape d'analyse ou d'opérations ultérieures affectées à la compréhension, il convient de construire les algorithmes de traitement, c'est à dire les mécanismes informatiques.

Dans ce qui suit, nous étudierons les moyens disponibles pour résoudre les deux types de problème :

Dans 1.4 , nous verrons les formalismes grammaticaux et leur influence importante sur la forme des représentations internes. Les grammaires traditionnelles ne se prêtant pas facilement au traitement informatique, nous parlerons d'abord des grammaires formelles (1.4.3.1) et de leurs évolutions, les grammaires transformationnelles (1.4.3.2) avec la théorie standard, la théorie standard étendue, la théorie des traces, qui privilégient les aspects syntaxiques. Parmi les extensions intéressantes figurent les grammaires syntagmatiques généralisées (elles tentent de traiter les aspects sémantiques) et les grammaires logiques (grammaires de métamorphose et grammaires à clauses définies) qui permettent de construire des représentations internes plus variées et concernent davantage les aspects de mise en oeuvre sur ordinateur. C'est la raison pour laquelle nous avons préféré les traiter avec les outils (1.5.8). Les grammaires de cas (1.4.3.3) accentuent le rôle de la sémantique. Les grammaires systémiques (1.4.3.5) accordent la priorité au contexte d'utilisation du langage et ne le considèrent plus comme un système formel isolé. Nous terminerons par les grammaires fonctionnelles (1.4.3.6) qui intègrent des aspects sémantiques et constituent actuellement, pour cette raison, un axe de recherche important. Elles expriment l'importance accrue du lexique dans les descriptions linguistiques, comme dans les travaux de M. GROSS et de MEL'CUK.

Dans 1.5 , nous aborderons les processus informatiques qui utilisent les représentations basées sur les modèles présentés dans la partie précédente et qui concernent schématiquement l'analyse syntaxico-sémantique de la phrase pour la construction d'une représentation interne de la phrase.

## 1.4 Les modèles de la langue

Nous ne nous intéresserons qu'au traitement du langage naturel écrit. Les applications sont multiples et l'apparition d'une "*industrie de la langue*" confirme l'intérêt économique qu'il suscite. La présentation sommaire de ce vaste domaine, articulée sur les théories linguistiques, les techniques informatiques et leurs applications soulignera la difficulté des problèmes rencontrés et l'aspect pluridisciplinaire de ces recherches. Nous ne tiendrons pas compte des modèles inutilisés par l'informatique et ne mentionnerons des applications distinctes de la T.A. et de la T.A.O. qu'en abordant les outils d'application (1.5).

Que l'on examine les moyens qui permettent au dialogue homme-machine de se rapprocher d'un dialogue normal, ou que l'on considère les réflexions que l'être humain a pu consacrer à son propre langage, depuis l'antiquité déjà, sous les angles de la grammaire, de la philosophie, de la linguistique... sans oublier l'informatique qui introduit une façon de poser les questions et d'exprimer les réponses susceptibles de renouveler notre pensée sur le langage, nous constatons que l'informatique entretient des rapports complexes avec le langage naturel. Les modèles choisis pour l'interprétation de textes en langage naturel sont issus d'autres disciplines que l'informatique. Cette dernière ne doit cependant pas être réduite à un simple outil, elle peut également fournir des modèles.

### 1.4.1 Les niveaux

Le traitement informatique du langage naturel sous-tend un ensemble d'hypothèses sur ce qu'est le langage. Ces hypothèses se traduisent par le choix d'un *modèle* dont nous admettrons que le but est d'expliquer comment on passe d'une chaîne de caractères à sa signification. Il est désormais habituel, en linguistique, de considérer cinq niveaux de la langue écrite :

- le niveau **morphologique** : on reconnaît les mots sous les différentes formes liées à leur rôle dans la phrase (déclinaison, conjugaison...).
- le niveau **lexical** : permet d'associer le mot reconnu aux informations dont on dispose sur ce mot.
- le niveau **syntaxique** : traduit l'agencement des mots dans la phrase.
- le niveau **sémantique** : correspond à la représentation du sens des mots.
- le niveau **pragmatique** : élargit le contexte et concerne les aspects implicites du sens. C'est ce que le locuteur fait du langage.

Les traitements qui sont effectués à chaque niveau sont spécifiques. La conception des systèmes est de plus en plus souvent modulaire, de sorte que l'on retrouve dans les réalisations informatiques une division des tâches identique à celle des linguistes.

- La plupart des systèmes sont conçus pour un enchaînement séquentiel des traitements, du morphologique au pragmatique, avec des possibilités limitées de retour en arrière.
- Une autre conception se développe dans le sens d'une intégration des différents traitements. Les réalisations ne sont pas très nombreuses, nous y reviendrons par la suite.

Si les deux premiers niveaux ne présentent pas de difficultés importantes, il en va autrement pour les niveaux supérieurs (syntaxe, sémantique, et pragmatique), ainsi définis par C. W. MORRIS<sup>1</sup>, dans la lignée des travaux de C. S. PEIRCE.

### 1.4.2 Les modèles sans syntaxe

Une vision réductrice classe les mots d'un texte en deux catégories, ceux qui portent une signification et les autres. Une hypothèse encore plus forte affirme que l'ordre des mots n'influe pas sur la signification d'un texte. Un texte se réduit alors à un ensemble de mots clés. Des équivalences permettent de savoir si le texte répond positivement ou négativement à une requête exprimée comme une expression booléenne.

Ce type d'approche est à proscrire lorsque le traitement s'applique à un domaine sémantique ouvert ou à des constructions complexes. Il convient par contre à certaines applications comme l'interrogation de bases de données de petite taille. Il est en effet possible de substituer à une requête formulée en langage naturel:

*"Pouvez-vous m'indiquer les horaires de train au départ de Paris pour la ville de Metz ?"*  
une liste de mots clés *"Horaire, train, Paris, Metz"*.

---

(1) C.W. MORRIS : "Foundations of the Theory of Signs" in O. NEURATH, R. CARNAP, C.W. MORRIS (eds.), *Encyclopaedia of Unified Science*, University of Chicago Press, 1939

Le choix de ces mots clés n'est pas toujours évident. Il est parfois nécessaire de lever l'ambiguïté de certains termes et d'interpréter correctement l'ordre des mots.

### 1.4.3 Les modèles avec syntaxe

Les autres modèles admettent que les phrases ont une structure et que cette structure joue un rôle important pour la compréhension.

#### 1.4.3.1. Les grammaires formelles

Un langage formel construit sur un alphabet est un sous-ensemble (potentiellement infini) de toutes les chaînes formées des éléments de cet alphabet.

Le premier concept est celui du vocabulaire  $V$ , composé de deux sous-ensembles finis et disjoints:

- L'alphabet  $V_T$  (vocabulaire terminal) avec un nombre fini de symboles qui figurent les mots du texte tels qu'ils ont été identifiés par les traitements morpho-lexicaux.

- L'ensemble  $V_N$  (vocabulaire non terminal) des symboles nécessaires à la description de cette langue ("les variables" ou "les catégories syntaxiques"). La phrase est ainsi mieux structurée que dans les grammaires traditionnelles où l'on ne formule que deux niveaux : les propositions et les catégories lexicales.

-  $V^*$ , le monoïde libre construit sur  $V_T$ , est l'ensemble des chaînes finies formées en combinant les symboles terminaux de toutes les façons possibles. Un langage est défini, par conséquent, comme un sous-ensemble de  $V^*$ .

Pour illustrer la structure interne d'une phrase, on utilise l'opération de réécriture (règles de réécriture ou règles de production). Un ensemble de ces règles substitue à un axiome  $S$  (Sentence) ou  $P$  (Phrase) des chaînes composées d'éléments terminaux et/ou auxiliaires.

La règle  $P \longrightarrow GN + GV$  indique qu'une phrase ( $P$ ) peut être composée d'un syntagme nominal  $GN$  suivi d'un syntagme verbal  $GV$ .

$+$  est le symbole de la concaténation.

$\longrightarrow$  signifie qu'il faut réécrire le symbole de gauche en utilisant les symboles de droite.

L'ensemble de règles de réécriture peut se noter :

$R = \{ X \longrightarrow Y \}$  avec  $X, Y \in V^*$  et  $X \neq \emptyset$

On peut définir une grammaire formelle par le quadruplet :

$G = (V_N, V_T, R, P)$

Le langage engendré par la grammaire  $G$  est l'ensemble de toutes les chaînes terminales que l'on peut dériver de  $S$  ou de  $P$  en appliquant toutes les séquences possibles des règles de réécriture.

On peut classer les grammaires selon plusieurs critères. Nous reprendrons la classification de N. CHOMSKY<sup>1</sup> pour les quatre types de grammaires formelles, numérotées de 0 à 3.

---

(1) M. GROSS, A. LENTIN : *Notions sur les grammaires formelles*, Gauthier-Villars, Paris, 1970

## 1.4.3.1.1 Les grammaires de type 3

Ce sont des grammaires régulières. Si l'on note X et Y les éléments non terminaux et a un élément terminal

( $X, Y \in V_N$ ,  $a \in V_T$ ) on peut définir les règles:

$X \longrightarrow a+Y$  ou  $X \longrightarrow a$  grammaire régulière à gauche  
 $X \longrightarrow Y+a$  ou  $X \longrightarrow a$  grammaire régulière à droite

Les langages décrits par ces grammaires sont caractérisés par la possibilité de rajouter itérativement un nombre arbitraire de fois, une séquence de mots, si la grammaire le prévoit. On peut montrer que le mécanisme de génération correspond à un automate d'états finis. On les appelle pour cette raison grammaires d'états finis. Au cours de la génération, il suffit de connaître l'état dans lequel on se trouve pour terminer la phrase correctement.

Tout langage fini est régulier, il peut être décrit par une grammaire régulière. Dans une application concrète, l'alphabet est fini, tout comme la longueur des phrases à analyser. On peut donc utiliser un modèle régulier mais les modèles de ce genre risquent d'être énormes et reflètent mal la structure des phrases. Ce type de grammaire ne peut pas traiter certains aspects comme les structures imbriquées ou parenthésées, d'où son insuffisance. On peut cependant les utiliser pour des cas simples, car elles se comportent comme des outils d'analyse déterministe qui traitent une phrase dans un temps proportionnel à sa longueur.

Pour tenir compte des régularités sous-jacentes des langues, on est amené à considérer le formalisme suivant dans la hiérarchie définie par N. CHOMSKY.

## 1.4.3.1.2 Les grammaires de type 2

Le membre gauche de la règle ne peut être qu'un symbole non terminal (appartenant à  $V_N$ ). Parce que ces règles n'expriment aucune contrainte sur le membre droit, la grammaire est dite indépendante du contexte, ou à contexte libre. Le membre gauche peut être réécrit sous la forme du membre droit, indépendamment des symboles qui l'entourent.

Soit la grammaire composée des deux règles suivantes :

$P \longrightarrow a+P+b$

$P \longrightarrow a+b$

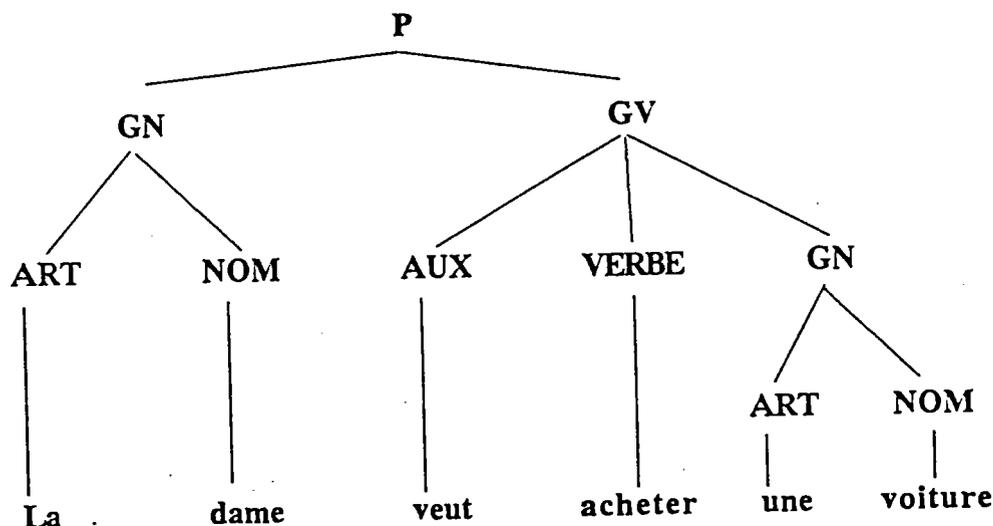
Elle engendre un nombre quelconque de "a" suivi d'un nombre quelconque de "b". On a besoin de connaître le début de la phrase pour la terminer correctement.

exemple :

Règle 1	$P \longrightarrow$	GN+GV
Règle 2	$GN \longrightarrow$	ART+NOM
Règle 3	$GV \longrightarrow$	(AUX)+VERBE+GN
Règle 4	$AUX \longrightarrow$	peut/doit/veut/...
Règle 5	$VERBE \longrightarrow$	lire/manger/acheter/...
Règle 6	$ART \longrightarrow$	le/la/les/un/une/des/...
Règle 7	$NOM \longrightarrow$	homme/femme/garçon/soupe/...

L'application successive de R1, R2, R3, R2, R6, R7, R4, R5 permet de construire la phrase *La dame veut acheter une voiture*.

Cette grammaire permet une représentation graphique de l'application des règles à la manière des schémas parenthésés de R. S. WELLS<sup>1</sup> et des "boîtes" de C. F. HOCKETT<sup>2</sup>.



Ce type de représentation illustre l'impossibilité de traiter des constituants discontinus (il faudrait que les branches se croisent) mais peut rendre compte des phénomènes d'enchâssement. Ce formalisme correspond à la notion d'automate à pile, ce qui est assez simple à transcrire sur un plan informatique (paragraphe 1.5.3.3).

Les grammaires à contexte libre étaient réputées insuffisantes pour décrire les langues naturelles. Les travaux de M. SALKOFF<sup>3</sup> et de G. GAZDAR<sup>4</sup> ont tenté d'y remédier, mais elles mettent alors en œuvre un nombre de règles (de l'ordre du milliard) qui les rend inadéquates au traitement informatique. La prise en compte d'un aspect linguistique nouveau entraîne une croissance exponentielle du nombre de règles.

Bien que des modèles à contexte libre regroupant un petit nombre de règles soient adaptés à des sous-ensembles restreints du langage naturel, le formalisme suivant de N. CHOMSKY semble présenter de nouveaux avantages.

#### 1.4.3.1.3 Les grammaires de type 1

Les règles  $X \rightarrow Y$  ( $X, Y \in V^*$ ,  $X \neq \emptyset$ ) sont telles que la chaîne  $Y$  contient au moins autant de symboles que la chaîne  $X$ .

On peut écrire les règles sous une forme équivalente :  
 $u+x+v \rightarrow u+y+v$  ( $u, v, y \in V^*$ ,  $x \in V_N$ ,  $y \neq \emptyset$ )

(1) R.S. WELLS : "Constituants immédiats" in *Langage*, n°20, pp. 61-100, 1970

(2) C.F. HOCKETT : *A Course in Modern Linguistics*, Macmillan, New York

(3) M. SALKOFF : "Analyse syntaxique du français : grammaire en chaîne", *Etudes en linguistique française et générale*, n°2, John Benjamins, Amsterdam, 1979

(4) G. GAZDAR : "Phase Structure Grammars and Natural Languages", *Proc. of the 8th IJCAI*, Karlsruhe, pp. 536-565, aug. 1983

Ceci signifie que  $x$  peut se réécrire en  $y$  dans le contexte  $u—v$ , autrement dit, s'ils sont tous deux entourés de  $u$  et de  $v$ .

Ces grammaires sont appelées, pour cette raison, *sensibles au contexte* ou *grammaires contextuelles*. La première forme signifie que le nombre de symboles de la chaîne ne peut qu'augmenter. Si nous considérons une phrase de longueur  $n$ , le fait de ne pas l'avoir rencontrée après avoir généré toutes les chaînes de longueur au moins égale à  $n$  implique qu'on ne pourra pas la rencontrer par la suite, elle n'appartient donc pas au langage. On peut avoir recours, par conséquent, à un automate linéaire borné qui utilise une mémoire proportionnelle à la longueur de la phrase à reconnaître.

La difficulté de traiter facilement les allotaxies et les constituants discontinus, la possibilité de ne pas reconnaître toutes les homotaxies, l'extrême complexité dans la description des règles sont autant de raisons qui ont conduit au rejet des grammaires contextuelles. Ce modèle n'est pas utilisé sous cette forme pour l'analyse des langues bien que certains mécanismes employés lui soient en fait équivalents. La mise en oeuvre dans des programmes reste très délicate. Il est souvent préférable d'accroître la puissance générative des grammaires non contextuelles pour les rendre équivalentes aux grammaires contextuelles. C'est l'approche qui a été développée avec les réseaux de transition augmentés (ATN) que nous aborderons dans la présentation des outils d'analyse.

La résolution des problèmes liés au sens, (rejet des phrases ayant une structure correcte mais un sens inadmissible) conduit, pour ces grammaires contextuelles à développer des traits sémantiques, dans le module syntaxique. Il a fallu modifier le modèle pour que ces traits puissent fonctionner en liaison avec la notion de transformation. Dans ce cas, ils sont considérés comme des marqueurs qui indiquent les transformations que subira la structure.

#### 1.4.3.1.4 Les grammaires de type 0

Il s'agit de systèmes non limités et trop peu structurés (systèmes de réécriture généraux ou règles générales de réécriture), puisque l'on n'introduit aucune contrainte sur la forme des règles de réécriture et que les langages correspondants peuvent être analysés par une machine de TURING générale<sup>1</sup>. Cette grammaire est par conséquent difficilement utilisable. Nous citons quelques explications en 1.5.3.3.

#### 1.4.3.2 Les grammaires transformationnelles

Issues de la linguistique structurale américaine, ces grammaires génératives sont connues essentiellement à travers les publications de N. CHOMSKY.

Simple mécanisme génératif syntaxique, le modèle devient une théorie complète du langage avec des visées plus cognitives.

---

(1) A. TURING : "On Compatible Numbers with an Application to the Entscheidungsproblem", *Actes London mathematical society*, Série 2, 42, pp. 230-265, 1936

### 1.4.3.2.1 La théorie standard

N. CHOMSKY<sup>1</sup> intègre une composante sémantique et introduit des transformations. Deux étapes aboutissent à la formulation d'une idée :

- la création d'une structure profonde qui, seule, peut être interprétée sémantiquement.
- la transformation de cette structure profonde en une structure de surface liée directement à la forme de l'énoncé.

Les connaissances syntaxiques s'articulent en trois volets :

- une grammaire formelle qui, complétée par le lexique, constitue la base du module syntaxique et permet d'engendrer des structures abstraites, appelées structures profondes (correspondant pratiquement à des phrases noyaux). On leur associe l'arbre de dérivation correspondant.
- des règles de transformation qui s'appliquent à ces arbres de dérivation et permettent d'engendrer toutes les formes possibles de phrases grâce à différents mécanismes manipulant les chaînes terminales pour produire les structures de surface.
- les règles morpho-phonémiques engendrent à partir de la structure de surface la suite des caractères et des phonèmes qui constituent la phrase.

Un module sémantique interprète la structure profonde et produit le sens de la phrase générée. On constate, dans ce modèle, que tous les éléments nécessaires au module sémantique appartiennent à la structure profonde. On peut douter de l'autonomie de la syntaxe que ce modèle implique. Il n'est pas évident d'autre part qu'il soit possible de décrire la compétence linguistique sans tenir compte des mécanismes de production et de compréhension.

### 1.4.3.2.2 La transformation

Z. HARRIS<sup>2,3</sup> a été le premier à utiliser la transformation de façon opératoire. Une transformation se décompose en :

- une description structurelle DS (forme qui correspond à la partie de l'arbre à laquelle on applique la transformation et qui indique une suite d'éléments voisins, sans préciser leurs relations exactes dans l'arbre).
- des changements structurels CS (les opérations à appliquer sur l'arbre, conformément à une règle).
- des conditions (pour définir les cas dans lesquels les règles s'appliquent).

Une règle s'appliquera à un sous-arbre à condition qu'il contienne la suite de noeuds indiquée par DS. La présence d'éléments avant ou après est sans importance.

---

(1) N. CHOMSKY : *Structures syntaxiques*, Le Seuil, Paris, 1959

(2) Z. HARRIS : "Co-occurrence and Transformation in Linguistic Structure" in *Language*, 33, 3, pp. 283-340, 1957

(3) Z. HARRIS : "Linguistic Transformation for Information Retrieval", *International Conference on Scientific Information*, Vol. 2, p. 158, 1958

La présence d'éléments supplémentaires entre les éléments indiqués est par contre interdite. Quand il y a correspondance avec le sous-arbre, les changements structurels (CS) interviennent comme une série de transformations élémentaires (suppression, substitution, adjonction) pour aboutir à la transformation de la structure profonde en une structure de surface.

On distingue deux catégories de règles :

- les règles obligatoires qui arrangent les éléments terminaux et sans lesquelles on ne peut obtenir une phrase correcte. Elles sont marquées par des traits dans la structure profonde.
- les règles facultatives, qu'il n'est pas indispensable d'appliquer pour obtenir une phrase correcte et qui traduisent les possibilités de paraphrasage. Elles peuvent engendrer d'autres phrases correctes à partir de règles obligatoires (négation, forme interrogative, forme passive...).

Le statut des règles varie plus ou moins selon les traits reconnus par la grammaire et l'influence que peuvent ou ne peuvent pas exercer les transformations sur le sens (théorie standard, théorie standard étendue...). Dans le cas présent, il ne peut y avoir modification du sens puisque la composante sémantique n'agit que sur la structure profonde.

Une série de transformations ne peut pas toujours s'effectuer dans n'importe quel ordre. Une transformation peut, en intervenant prématurément, modifier une structure, la rendre impropre à une autre transformation ou fausser les résultats. On est ainsi contraint de fixer un ordre d'application, par paire de règles. On trouve un excellent exemple pour l'anglais dans A. AKMAJIAN, F. HENY<sup>1</sup>.

#### 1.4.3.2.3 Les grammaires en chaîne

Z. HARRIS<sup>2</sup> a introduit la notion de transformation, mais sans structure profonde et sans traitement sémantique, dans le but principal de décrire les relations de paraphrase entre les formes de surface de phrases attestées.

Basée sur une étude distributionnelle de la langue<sup>3</sup>, sa théorie consiste à considérer tout énoncé comme l'adjonction de chaînes auxiliaires à une chaîne centrale élémentaire (le plus petit segment de phrase acceptable et significatif que l'on peut isoler dans l'énoncé).

Ces chaînes centrales élémentaires constituent le noyau de cette grammaire. Une chaîne centrale se décrit par une quinzaine de formules, du type  $\Sigma t V \Omega$  (sujet, verbe fléchi, objet). La séquence de catégories grammaticales indique une structure où les catégories pourront être remplacées par des mots qui en font partie.

On définit ensuite des chaînes d'ajout (groupes prépositionnels, adverbes, propositions subordonnées) également décrites par des formules, qui pourront s'insérer entre deux éléments de chaîne.

Les phrases de la langue sont ainsi des chaînes dérivées des chaînes élémentaires par insertion de chaînes d'ajout. Il est cependant nécessaire de limiter les combinaisons possibles en imposant des restrictions le plus souvent sémantiques, à la combinaison de catégories syntaxiques autorisées par les formules.

---

(1) A. AKMAJIAN, F. HENY : *An Introduction to the Principles of Transformational Syntax*, Cambridge Mass., MIT press, 1975

(2) Z. HARRIS : *Structures mathématiques du langage*, Paris, Dunod, 1971

(3) Z. HARRIS : *Methods in Structural Linguistics*, Chicago, 1951

Ces restrictions sont souvent complexes. Il n'est pas toujours facile de les formaliser. Pour le verbe *regarder*, on limitera la classe des sujets possibles à celle des êtres vivants, et la classe des objets à celle des éléments concrets. L'étude des relations qu'entretiennent plusieurs phrases entre elles, dans le contexte de l'acceptabilité et de l'ambiguïté conduit à construire une grammaire de listes (travaux de M. GROSS). La grammaire en chaîne apporte une souplesse importante pour exprimer l'ordre relatif des constituants de la phrase. Son rôle essentiel est de décrire les phénomènes de surface sans utiliser la notion de structure profonde. Son intérêt n'est évident que dans le cas d'une analyse purement syntaxique (analyseur automatique de l'anglais de N. SAGER et analyseur d'intitulés en français par J.H. JAYEZ, p. 74).

#### 1.4.3.2.4 Extensions de la théorie standard

Le modèle transformationnel tel que le définit la théorie standard repose sur des hypothèses qui concernent des structures abstraites. Dans la théorie standard étendue et la théorie des traces, Z. HARRIS cherche à maîtriser la surpuissance d'un modèle assimilé à une grammaire de type 0. P.S. PETERS et R.W. RICHTIE<sup>1,2</sup> ont remis en cause son universalité.

##### 1.4.3.2.4.1 La théorie standard étendue

Elle reconnaît l'influence des transformations sur l'interprétation sémantique<sup>3</sup>. Le module sémantique doit s'appliquer à l'ensemble des arbres engendrés par les transformations, à partir de la structure profonde (N. CHOMSKY<sup>4</sup>). Pour traiter les exceptions et les irrégularités en ce qui concerne les mots à insérer dans la structure de surface, N. CHOMSKY propose d'ajouter l'étape d'insertion lexicale avant les processus de transformation, ce qui a pour résultat d'aboutir à deux types de règles :

- la production d'une séquence préterminale de marqueurs grammaticaux, de marqueurs de transformations et de catégories lexicales.
- Ces dernières sont remplacées par des mots grâce aux règles d'insertion lexicale et l'on obtient la séquence terminale, ou structure de surface.

Peu de programmes utilisent actuellement la théorie standard étendue. On peut citer W. PLATH et F. DAMEREAU avec le système REQUEST.

##### 1.4.3.2.4.2 La théorie des traces

Dans cette variation de la théorie précédente, N. CHOMSKY<sup>5</sup> propose que les deux modules sémantique et phonologique n'agissent que sur la structure de surface, qui contient des traces reflétant les informations de la structure profonde et des transformations. L'hypothèse selon laquelle on pourrait engendrer n'importe quel langage à partir d'une grammaire de base ordinaire, avec des transformations appropriées, a engendré des modèles voisins où les transformations ont été remplacées par d'autres mécanismes. Des difficultés théoriques et le fait surtout que la grammaire transformationnelle suppose que, lors de l'analyse, on applique les transformations à l'envers pour retrouver

---

(1) P.S. PETERS, R.W. RICHTIE : "On the Generative Power of Transformational Grammars", *Information Science*, 6, pp. 49-83, 1973

(2) P.S. PETERS, R.W. RICHTIE : "A Note of the Universal Base Hypothesis", *Journal of Linguistics*, 5, pp. 150-152, 1969

(3) N. CHOMSKY : *Aspects de la théorie syntaxique*, Le Seuil, Paris, 1971

(4) N. CHOMSKY : "Deep Structure, Surface Structure and Semantic Interpretation" in Steinberg & Jakobovits, pp. 183-216, 1971

(5) N. CHOMSKY : *Reflexions on Language*, Pantheon, New Jersey, 1975, (traduction française chez Payot, Paris, 1977)

une structure profonde, expliquent, outre la complexité de mise en œuvre, que l'informatique et le modèle n'aient pas fait bon ménage. Il faut bien avouer, également, qu'il est beaucoup plus important de modéliser la compréhension que de reconnaître si une phrase a une structure syntaxique correcte. Un des buts que poursuivent les travaux de Traduction Automatique et de Traduction Assistée par Ordinateur est bien d'interpréter un texte. Les théories qui introduisent toujours plus de notions sémantiques seront probablement mieux adaptées au traitement automatique des langues naturelles. Les travaux les plus significatifs sont ici ceux de M. MARCUS<sup>1</sup> avec son analyseur syntaxique déterministe.

#### 1.4.2.4.3 Les grammaires syntagmatiques généralisées

La complexité croissante des règles et des contraintes nécessaires n'a pas résolu tous les problèmes dont certains ont pu, cependant, trouver une solution, sans recours aux transformations. L'idée de les abandonner a conduit aux grammaires lexicales fonctionnelles avec le modèle de J. BRESNAN<sup>2</sup> et le modèle de G. GAZDAR<sup>3</sup> qui introduisent les notions de traits avec les catégories dérivées, les méta-règles et les règles sémantiques. Si les traits tiennent compte des aspects contextuels dans les grammaires de type 1, ils s'appliquent ici à chaque nœud et sont organisés hiérarchiquement, à la façon des grammaires systémiques (1.5.3.5). La notion de catégorie dérivée permet de réserver une place pour un constituant qui n'est pas précisé et de générer ainsi directement une structure de surface, sans utiliser de transformations mais avec les informations suffisantes pour retrouver l'équivalent d'une structure profonde. Les méta-règles augmentent la puissance de création des règles sans influencer la puissance du formalisme lui-même, en fonctionnant dans une certaine mesure comme les transformations mais en opérant sur les règles et non sur les arbres. Le but des règles sémantiques est de créer une structure logique qui représente le sens de la phrase. La création de cette structure est réalisée en parallèle avec la structure syntaxique.

A chaque règle syntaxique est liée une règle sémantique, l'application de la première dépendant de l'applicabilité de la seconde. Cette méthode est identique au fonctionnement de la grammaire du logicien R. MONTAGUE qui laisse de côté les problèmes syntaxiques des "transformationnalistes" pour s'attacher aux aspects sémantiques. A chaque mot sont associées une catégorie syntaxique et une catégorie sémantique logique. R. MONTAGUE<sup>4</sup> substitue au calcul des prédicats du premier ordre une grammaire de catégories avec des règles syntaxiques et sémantiques interdépendantes. Les avantages de ces règles sémantiques sont évidents pour l'interprétation de certaines ambiguïtés, les problèmes de grammaticalité, de la quantification, des constituants discontinus... La théorie de G. GAZDAR<sup>5</sup> s'écarte des grammaires syntagmatiques au fil des modifications concernant :

- les catégories syntaxiques considérées comme des ensembles non ordonnés de traits décrits par des couples <attribut - valeur>

(1) M. MARCUS : *A Theory of syntactic recognition for natural language*, Cambridge Mass., MIT Press, 1980

(2) J. BRESNAN, R. KAPLAN : "Lexical Functional Grammars ; a Formal System for Grammatical Representation" in *The mental Representation of Grammatical Relations*, J. BRESNAN (ed.), MIT Press, 1981, Cambridge Mass.

(3) G. GAZDAR, G. PULLUM, I. SAG : "A Phrase Structure Grammar of the English Auxiliary System", 1980, in *The Nature of Syntactic Representation*, B. JACOBSON, G. PULLUM, Dordrecht, 1982

(4) R. MONTAGUE : "English as a Formal Language", repris dans *Formal Philosophy*, THOMASON, Yale University Press, p. 188

(5) G. GAZDAR, E. KLEIN, G. PULLUM, I. SAG : *Generalized Phrase Structure Grammar*, Basil Blackwell, Oxford, 1985

- les règles parmi lesquelles il faut distinguer les règles de dominance immédiate (DI) et les déclarations de précedence linéaire (PL).  
Le fonctionnement des méta-règles est de plus en plus limité et les éloigne progressivement des transformations.

On mesurera le poids de la syntaxe et des théories de N. CHOMSKY dans le système SYSTRAN (2.3.3.2.9). Pour une application de grammaire indépendante du contexte, nous renvoyons au système TAUM-AVIATION (3.3.3.3.7.2). En ce qui concerne les grammaires transformationnelles, nous verrons TAUM-METEO (2.3.3.3.7.1), METAL (2.3.2.3) et pour les grammaires syntagmatiques généralisées, ATLAS I (2.3.3.2.1), HICATS (2.3.3.2.2) et METAL (2.3.3.2.5).

### 1.4.3.3 Grammaires de cas

Le modèle transformationnel est capable de traduire les similarités entre les structures profondes de phrases présentant des structures de surface différentes. Il est incapable de résoudre un certain nombre de problèmes comme, par exemple, la conjonction des sujets : (1) *Pierre lave le linge.* (2) *Sophie lave le linge.*

*Pierre et Sophie lavent le linge* serait une transformation correcte.

Considérons une autre phrase :

(3) *la lessive lave le linge.*

Une transformation du même type sur (2) et (3) donnerait :

*Sophie et la lessive lavent le linge.*

Ce qui n'est pas acceptable. Les règles syntaxiques sont équivalentes mais les règles sémantiques sont différentes.

#### 1.4.3.3.1 La théorie de C. FILLMORE

Le modèle casuel que définit C. FILLMORE<sup>1</sup>, plutôt que d'attribuer une structure profonde identique aux deux phrases, distingue les sujets par leurs traits. La structure profonde se décompose en un verbe relié à des groupes syntaxiques par des cas (agent, instrument, datif, factitif, locatif, objet). Ce système est complété par une liste de modalités. Les cas (relations casuelles) décrivent les fonctions sémantiques générales des arguments d'un prédicat. C. FILLMORE pensait réunir un ensemble de cas capables de définir des représentations indépendantes de la langue d'origine et qu'il supposait en nombre restreint. Les diverses versions du modèle<sup>2,3</sup> traduisent les difficultés qu'il a rencontrées. Si nous reprenons l'exemple précédent, la structure de cas résout le problème. Les groupes nominaux ne peuvent être liés par une conjonction que s'ils jouent le même cas sémantique par rapport au verbe.

*Pierre et Sophie* sont les agents de l'action exprimée dans (1) et (2).

*La lessive* est l'instrument.

Parce que les structures profondes font apparaître avec plus de netteté le sens de la phrase, les spécialistes en Intelligence Artificielle et en Traitement Automatique des Langues Naturelles ont manifesté beaucoup d'intérêt pour cette théorie. Les propriétés responsables de son succès en fixent également les limites. S'il est simple de limiter à une dizaine de cas les relations entre un groupe nominal et un verbe, on appauvrit le registre des significations représentables.

---

(1) C. FILLMORE : "The Case for Case" in *Universals in Linguistic Theory*, Bach & Harms, Chicago, Holt, Rinehart and Winston, pp. 1-90, 1968

(2) C. FILLMORE : "Types of Lexical Information" in *Semantics : an Interdisciplinary Reader*, Steinberg & Jakobovits, Cambridge University Press, pp. 370-392, 1971

(3) C. FILLMORE : "Verbs of Judging : an Exercise in Semantic Description" in *Studies in Linguistic Semantics*, C. FILLMORE & LANGENDSEN, Chicago, Holt, Rinehart and Winston, 1971

La notion de "Case-frame" qui correspond au modèle casuel de B. POTTIER<sup>1</sup> permet de traiter correctement certains cas de polysémies et d'homotaxies. Il faut cependant noter des difficultés pour le choix cohérent des cas sémantiques. B. BRUCE<sup>2</sup> a tenté de proposer quelques règles pour définir la notion de cas mais ses indications ne procèdent pas de critères scientifiques indiscutables.

D'autres théories font suite au modèle de C. FILLMORE.

#### 1.4.3.3.2 La théorie de J. GRIMES<sup>3</sup>

Elle a pour but d'atteindre un degré d'abstraction plus élevé en réunissant des cas regroupant des concepts voisins. Un groupe nominal donné peut jouer plusieurs rôles sémantiques distincts, ce qui constitue une différence essentielle avec le modèle de C. FILLMORE. Cette théorie s'applique en fait à l'analyse des discours. Dans le même ordre d'idées, FEUILLET<sup>4</sup> essaie de déterminer un ensemble universel de fonctions sémantiques profondes (*STATIF, AGENTIF, RECEPTIF, OBJECTIF, MEDLATIF, DESCRIPTIF, et SITUATIF*), liées à tout ensemble de cas.

#### 1.4.3.3.3 La théorie de R. SIMMONS<sup>5</sup>

Les réseaux sémantiques de R. SIMMONS utilisent également les cas sémantiques profonds (*ACTEUR, THEME, SOURCE, BUT, INSTRUMENT, LIEU, TEMPS*). Le principe de base du mécanisme de l'analyse qu'induit son modèle est de tester la compatibilité des concepts nominaux avec les restrictions correspondant aux cas du verbe.

#### 1.4.3.3.4 La théorie de R. SCHANK

Avec les "Dépendances conceptuelles", R. SCHANK<sup>6</sup> cherche à représenter le sens en s'appuyant sur des relations conceptuelles entre des objets et des actions, ces actions étant indépendantes de la langue. Il définit 11 primitives à partir desquelles pourront être décrites toutes les actions.

La dépendance conceptuelle implique :

- deux phrases ayant une signification équivalente doivent avoir la même représentation interne, même si elles ont des structures syntaxiques très différentes.
- toute information implicite doit être explicite dans la représentation.
- toute action s'exprime en termes de primitives. Chaque primitive possède un schéma associé qui devra être "instancié" et rempli lors du processus de compréhension.

La signification d'une phrase est représentée par un schéma de dépendance conceptuelle. Les catégories qui interviennent dans le schéma sont :

- les PPs (Picture Production) les noms
- les ACTs (Action) les verbes ou groupes de verbes
- les PA (Picture Aider) les modificateurs des PPs comme les adjectifs
- les AA (Action Aider) les modificateurs des ACTs comme les adverbes

---

(1) B. POTTIER : "Vers une sémantique moderne" in *Travaux de Linguistique et de Littérature, Centre de Philosophie et Littérature romanes de Strasbourg*, tome 2, pp. 107-137, Klincksieck, Paris, 1964

(2) B. BRUCE : "Case Systems for Natural Language", *BBN report n° 3010*, 1975

(3) J. GRIMES : "The Thread of Discourse", *Rapport technique NSF*, Cornell University, 1972

(4) J. FEUILLET : "Les fonctions sémantiques profondes" in *Bulletin de la Société de Linguistique de Paris*, LXXV, 1, pp. 1-37

(5) R. SIMMONS : "On Managing Sentence Meanings", *Report NL-20*, Dept. of Computer Science, University of Texas, Austin, 1974

(6) R. SCHANK : "Conceptual Dependency : a Theory of Natural Language Understanding", *Cognitive Psychology*, vol.3 (4), pp. 552-631, 1972

Les dépendances peuvent être à double sens (le plus souvent entre un PP et un ACT), à sens unique entre un ACT et un PP ou entre un PP et un PA, à sens unique entre deux PPs.

A la différence d'une représentation en grammaire casuelle, la représentation en DC fait apparaître les relations causales. Représenter la signification d'une phrase consiste à trouver les primitives qui correspondent aux actions et à compléter la structure associée.

Il y a une quinzaine de primitives en tout, parmi lesquelles :

- *ATRANS* : désigne le transfert abstrait de quelque chose à quelqu'un (donner), de quelque chose à soi-même (prendre), le transfert simultané de plusieurs objets (acheter).
- *PTRANS* : désigne le transfert physique de soi-même quelque part (aller), d'un objet quelque part (mettre).
- *SPEAK* indique une production de sons (parler, chanter, siffler).
- *ATTEND* indique le fait de consacrer un organe des sens à un stimulus (écouter, voir).

C. RIESBECK, C. RIEGER et N. GOLDMAN ont développé le programme MARGIE<sup>1</sup> sous la direction de R. SCHANK. Ce programme analyse des phrases en langue naturelle pour les transformer en une représentation de dépendance conceptuelle. On trouve des applications des théories de R. SCHANK en T.A. et T.A.O. avec les systèmes HICATS (2.3.3.2.2) pour l'analyse sémantique et ATLAS II (2.3.3.2.1) avec les réseaux sémantiques.

Les travaux de Y. WILKS<sup>2</sup> sont assez voisins, avec la théorie des sémantiques préférentielles.

#### 1.4.3.3.5 Le modèle de B.B. BOGURAEV et K. SPARK-JONES<sup>3</sup>

A partir d'un grand corpus et avec le souci de déterminer un ensemble de cas exhaustif, B.B. BOGURAEV et K. SPARK-JONES s'appuient sur les travaux de F.T. WOOD<sup>4</sup> et proposent une liste de 27 cas.

D. RUMELHART, P. LINDSAY et D. NORMAN<sup>5</sup> ont construit le modèle de mémoire dans lequel la connaissance est codée comme un ensemble de propositions et de concepts liés par des cas sémantiques (notions de réseau sémantique et de schéma).

Parmi les applications informatiques de ce type de grammaire figurent :

- les recherches de B. NASH-WEBBER<sup>6</sup>, qui utilisent les cas sémantiques pour s'assurer que les structures syntaxiques repérées ont bien un sens (projet SPEECHLIS).

---

(1) R.C. SCHANK : *Conceptual Information Processing*, New York, 1975, North Holland

(2) Y. WILKS : "A Preferential Pattern-seeking Semantics for Natural Language Inference", *Artificial Intelligence*, 6, pp. 53-74, 1975

(3) B.B. BOGURAEV, K. SPARK-JONES : "How Drive a Database Front-end Using General Semantic Information", *Conference on Applied Natural Language Processing*, Santa Monica, pp. 81-88, 1983

(4) F.T. WOOD : *English Prepositional Idioms*, MacMillan, 1979, Londres

(5) D. RUMELHART, P. LINDSAY, D. NORMAN : "A Process Model of Long Term Memory" in *Organization of Memory*, Tulving & Donalson, Academic Press, 1975, New York

(6) B. NASH-WEBBER : "Semantic support for a speech understanding system", in *Representation and understanding, studies in cognitive science*, Bobrow & Collins, Academic Press, 1975, New York

- Le programme CHRONOS de G. BROWN, B. BRUCE et M. TRIGOBOFF<sup>1</sup>, qui analyse le langage en traitant des ensembles de cas adaptés au domaine abordé.

Le premier avantage des grammaires de cas est la représentation sémantique que l'on obtient, la notion de verbe et de son modèle casuel rappelant l'application du prédicat logique à des arguments.

Le second avantage est la possibilité, tout en utilisant les contraintes syntaxiques, de se diriger vers une analyse purement sémantique reposant sur les restrictions de sélection apportées par les modèles casuels.

La possibilité de traiter des phrases à syntaxe incorrecte justifie l'emploi du modèle pour le traitement automatique, en particulier pour les systèmes question-réponse destinés au grand public. De nombreux aspects ne peuvent être abordés, les phrases non verbales par exemple.

#### 1.4.3.4. Les grammaires sémantiques

Il est parfois difficile de conduire une analyse syntaxique complète avant tout début d'interprétation. De même, la jonction des niveaux syntaxique et sémantique au cours d'une analyse peut-être laborieuse. R.R. BURTON<sup>2</sup> propose d'étendre les règles d'analyse des grammaires indépendantes du contexte en classant les mots non plus seulement d'après leur nature grammaticale mais aussi d'après leur signification. A. BONNET<sup>3</sup> multiplie par 5 la vitesse d'interprétation de comptes-rendus médicaux, en introduisant les classes d'information qui, à l'inverse des cas, sont spécifiques de l'application. Le champ de l'application doit être limité (cas des dialogues dans les systèmes experts). Pour la résolution des ellipses, A. BONNET suggère de reprendre les dernières dérivations jusqu'à celle qui permettra de compléter les données manquantes. Ces grammaires permettent de traiter des énoncés normés. Très faciles à mettre en œuvre dans des domaines restreints, elles sont très souples mais aboutissent à une compréhension assez limitée.

#### 1.4.3.5 Les grammaires systémiques

##### 1.4.3.5.1 Présentation

Elles ne considèrent plus le langage comme un système isolé mais comme une activité sociale.

B. MALINOWSKI<sup>4</sup> puis B. WHORF<sup>5</sup> focalisent leurs travaux sur les aspects fonctionnels du langage et le rôle des différents constituants par rapport aux intentions du discours.

---

(1) G. BROWN, B. BRUCE, M. TRIGOBOFF : "The CHRONOS Natural Language Understanding System", Computer Science Department, Rutgers, *NIH report CBM-tr-30*, 1974

(2) R.R. BURTON : "Semantic Grammars, an Engineering Technique for Constructing Natural Language Understanding Systems", *Rapport BBN n°3453*, Dec. 1976

(3) A. BONNET : "Les grammaires sémantiques, outil puissant pour interroger les bases de données en langage naturel", *Rairo informatique*, 14(2), pp. 137-148, 1980

(4) B. MALINOWSKI : "Culture" in *International Encyclopedie of the Social Sciences*, Sills, New York, MacMillan, 1968

(5) B. WHORF : *Language, Thought and Reality*, Caroll, Cambridge Mass., MIT Press.

J.R. FIRTH<sup>1</sup> introduit ces idées dans la linguistique avec les premières notions de système. M. HALLIDAY<sup>2</sup> élabore une théorie complète, concentrée sur l'organisation fonctionnelle du langage et les rapports entre la forme du texte et le contexte dans lequel il est produit.

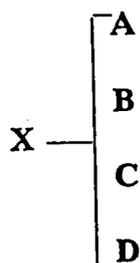
L'avantage de ce modèle descriptif réside, pour l'Intelligence Artificielle, dans l'approche de la sémantique et de la pragmatique, essentielle pour la gestion des dialogues homme-machine. Il est en effet très simple, en informatique, de transférer des traits d'une procédure à une autre, ce qui permet de tenir compte des contextes et de prendre des décisions fondées sur l'interaction de plusieurs processus.

### 1.4.3.5.2 Notion de système

Les fonctions envisagées selon divers points de vue peuvent se recouvrir, il est possible également qu'un même constituant puisse jouer plusieurs rôles.

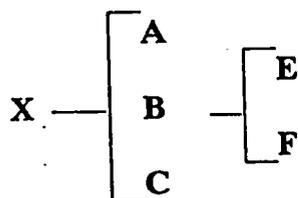
Un mot, un groupe de mots peut représenter le sujet, l'objet, l'agent... Les grammairiens systémiques partent du principe que la combinaison de ces fonctions n'est pas due au hasard, qu'elles appartiennent en fait à différents systèmes qui agissent ensemble pour déterminer la structure et le rôle de la phrase.

La description des systèmes utilise un formalisme simple. Dans le cas d'un choix simple, le schéma est le suivant :



Si un élément a la caractéristique X, il a un des traits A,B,C ou D, (exclusifs les uns des autres).

La représentation peut être récursive. Si une catégorie est retenue, d'autres choix peuvent intervenir, uniquement dans ce cas. On peut, dans le schéma suivant, avoir le choix entre E et F, pour les objets qui ont le trait B, seulement.



On peut également effectuer plusieurs choix indépendants les uns des autres (orthogonaux), ou encore, des choix non marqués.

(1) J.R. FIRTH : "A synopsis of Linguistic Theory (1930-1955)" in *Selected papers of J.R. FIRTH*, Palmer, Longman, Londres, pp. 168-205, 1957  
 (2) M. HALLIDAY : *Categories of the Theories of Grammar* Word, 17, pp. 241-292, 1961

### 1.4.3.5.3 Le modèle de M. HALLIDAY

M. HALLIDAY distingue 4 systèmes qui agissent séparément dans la détermination de la structure de la phrase (*MODE, ACTANTS, THEMATIQUE et INFORMATIF*).

- Le système du mode précise le type d'énonciation de la phrase (question, réponse, ordre...). Ces catégories sont définies a priori, mais, dans la pratique, ramènent aux catégories syntaxiques. Ce système fournit également des indications sur les rôles des constituants pour opérer les choix.
- Le système d'actants détermine les divers participants et les relations qu'ils entretiennent, syntaxiques ou, plus souvent, sémantiques. Le mécanisme est proche de celui des grammaires casuelles et appelle par conséquent les mêmes critiques.
- Le système thématique s'appuie sur le fait qu'une phrase met en évidence l'élément dont on parle (le thème), le reste de la phrase représente une propriété ou un prédicat qu'on lui attribue (le rhème). Le système thématique précise la structure prédicative de la phrase. Il est à quelques exceptions près, indépendant du système d'actants, mais très lié au système de mode.
- Le système informatif souligne ce qui représente une information nouvelle par rapport à ce que l'interlocuteur sait déjà.

### 1.4.3.5.4 T. WINOGRAD et le SHRDLU

Ce programme reste le plus célèbre dans l'histoire de l'Intelligence Artificielle. Il a été développé au M.I.T. par T. WINOGRAD<sup>1</sup> et simule un robot qui peut manipuler divers objets disposés sur une table, en conversant en langue naturelle avec un interlocuteur qui lui donne des ordres ou lui pose des questions sur les objets.

La grammaire utilisée est dérivée des grammaires systémiques de M. HALLIDAY. Un système de réseaux spécifie certaines propriétés des unités syntaxiques comme le mode, le temps et la voix pour les verbes. En cours d'analyse, des informations sémantiques réduisent le nombre de structures potentielles selon des critères purement syntaxiques.

Le module syntaxique, pour représenter le sens de la phrase, construit un arbre syntagmatique aux nœuds duquel sont attachés des traits issus d'une grammaire systémique. Cette méthode est parfois nommée technique des arbres décorés.

Les grammaires systémiques sont essentiellement explicatives et descriptives. Elles peuvent cependant engendrer des phrases si l'on associe à chaque trait un ensemble de règles de réalisation qui se résument à cinq types (*INCLUSION, CONFLATION, GATEGORISATION, ACCORD, ORDONNANCEMENT*). Ces grammaires sont très intéressantes lorsque l'on envisage l'intégration d'aspects syntaxiques et sémantiques.

Les systèmes ENGSPAN et SPANAM (2.3.3.3.6) s'inspirent des grammaires systémiques.

### 1.4.3.6 Grammaires et lexique

#### 1.4.3.6.1 Les grammaires d'unification

Nous avons noté qu'en grammaire générative, le module d'insertion lexicale intervenait avant l'application des règles syntaxiques et constituait une étape délicate. Les grammaires d'unification et les grammaires lexicales fonctionnelles appliquent une analyse fondée sur les caractéristiques syntaxiques et sémantiques des mots de la phrase.

---

(1) T. WINOGRAD : Understanding Natural Language, New York, Academic Press, 1972

Les modèles obtenus sont plus riches et plus souples que celui de N. CHOMSKY. Leur originalité est de considérer de façon uniforme les informations des dictionnaires, les connaissances sur la structure et les règles grammaticales comme des conditions nécessaires pour qu'un élément donné ait une forme correcte.

M. KAY<sup>1</sup> tente de construire un formalisme unique pour traiter ces aspects et introduit les "descriptions fonctionnelles". J. BRESNAN et R. KAPLAN<sup>2</sup> ont développé la notion de description additive, permettant des descriptions partielles complémentaires. Leur grammaire lexicale fonctionnelle accorde un rôle essentiel au lexique et aboutit à la description interne d'une phrase articulée en structure de constituants (aspects syntaxiques) et en structure fonctionnelle (aspects sémantiques).

#### 1.4.3.6.1.1 Le formalisme

Il doit être le même pour représenter les données lexicales, les règles de grammaire et les structures internes des phrases. On utilise le modèle de schéma proposé à l'origine par M. MINSKY<sup>3</sup>.

- un constituant se décrit par un ensemble de couples (chaque couple en donne une description partielle, indépendante des autres).

{(Attribut = Valeur)}

La notation est toujours identique, que l'attribut soit une fonction, la description d'un trait ou une unité lexicale.

- Les schémas (*Frames*) sont présentés dans des "boîtes", symbolisées par des crochets "[ ]", qui peuvent s'emboîter pour traduire la structure à la manière d'un parenthésage. On les réunit par une accolade chaque fois que plusieurs solutions sont possibles à un niveau donné.

- Un attribut est obligatoire, c'est la catégorie. Les réalisations lexicales ont l'attribut LEX. Sa valeur est le mot étudié.

Les autres types de schéma contiennent l'attribut FORME qui, à l'aide de descriptions partielles, indique les contraintes sur l'ordre des constituants concernés.

#### 1.4.3.6.1.2. Utilisation du formalisme

Nous allons présenter des exemples pour chacune des trois applications. Une partie de ces exemples sont empruntés à G. SABAH<sup>4</sup>

---

(1) M. KAY : "Functional Grammars", *Actes 5th Annual Meeting of the Berkeley Linguistic Society*, pp. 142-158, 1979

(2) J. BRESNAN, R. KAPLAN : "Lexical Functional Grammars ; A Formal System for Grammatical Representation", in *The Mental Representation of Grammatical Relations*, J. BRESNAN (ed.), MIT press, Cambridge Mass., 1981

(3) M. MINSKY : "A Framework for Representing Knowledge", *Memo 308*, MIT, Cambridge Mass., 1974

(4) G. SABAH : *L'intelligence artificielle et le langage*, Vol. 1, représentations des connaissances, pp. 128-155, 1988

- exemple de formalisme appliqué au lexique : une description dans le dictionnaire du nom masculin, singulier, *canari* aurait la forme :

Catégorie	=	Nom
Nombre	=	Singulier
Genre	=	Masculin
Lex	=	<i>canari</i>

Si le dictionnaire contient les formes fléchies, on trouvera pour *achètera* :

Catégorie	=	Verbe
Temps	=	Futur
Type	=	Action
Racine	=	<i>acheter</i>
Lex	=	<i>achètera</i>

Dans le cas d'un terme ambigu comme *la* (article défini, pronom personnel, note de musique), on regroupe les trois possibilités :

<table border="1"> <tr> <td>Catégorie</td> <td>=</td> <td>Article</td> </tr> <tr> <td>Type</td> <td>=</td> <td>Défini</td> </tr> <tr> <td>Nombre</td> <td>=</td> <td>Singulier</td> </tr> <tr> <td>Genre</td> <td>=</td> <td>Féminin</td> </tr> <tr> <td>Lex</td> <td>=</td> <td><i>la</i></td> </tr> </table>	Catégorie	=	Article	Type	=	Défini	Nombre	=	Singulier	Genre	=	Féminin	Lex	=	<i>la</i>	<table border="1"> <tr> <td>Catégorie</td> <td>=</td> <td>Pronom</td> </tr> <tr> <td>Type</td> <td>=</td> <td>Personnel</td> </tr> <tr> <td>Nombre</td> <td>=</td> <td>Singulier</td> </tr> <tr> <td>Personne</td> <td>=</td> <td>Troisième</td> </tr> <tr> <td>Genre</td> <td>=</td> <td>Féminin</td> </tr> <tr> <td>Lex</td> <td>=</td> <td><i>la</i></td> </tr> </table>	Catégorie	=	Pronom	Type	=	Personnel	Nombre	=	Singulier	Personne	=	Troisième	Genre	=	Féminin	Lex	=	<i>la</i>
Catégorie	=	Article																																
Type	=	Défini																																
Nombre	=	Singulier																																
Genre	=	Féminin																																
Lex	=	<i>la</i>																																
Catégorie	=	Pronom																																
Type	=	Personnel																																
Nombre	=	Singulier																																
Personne	=	Troisième																																
Genre	=	Féminin																																
Lex	=	<i>la</i>																																
<table border="1"> <tr> <td>Catégorie</td> <td>=</td> <td>Nom</td> </tr> <tr> <td>Nombre</td> <td>=</td> <td>Singulier</td> </tr> <tr> <td>Genre</td> <td>=</td> <td>Masculin</td> </tr> <tr> <td>Lex</td> <td>=</td> <td><i>la</i></td> </tr> </table>	Catégorie	=	Nom	Nombre	=	Singulier	Genre	=	Masculin	Lex	=	<i>la</i>																						
Catégorie	=	Nom																																
Nombre	=	Singulier																																
Genre	=	Masculin																																
Lex	=	<i>la</i>																																

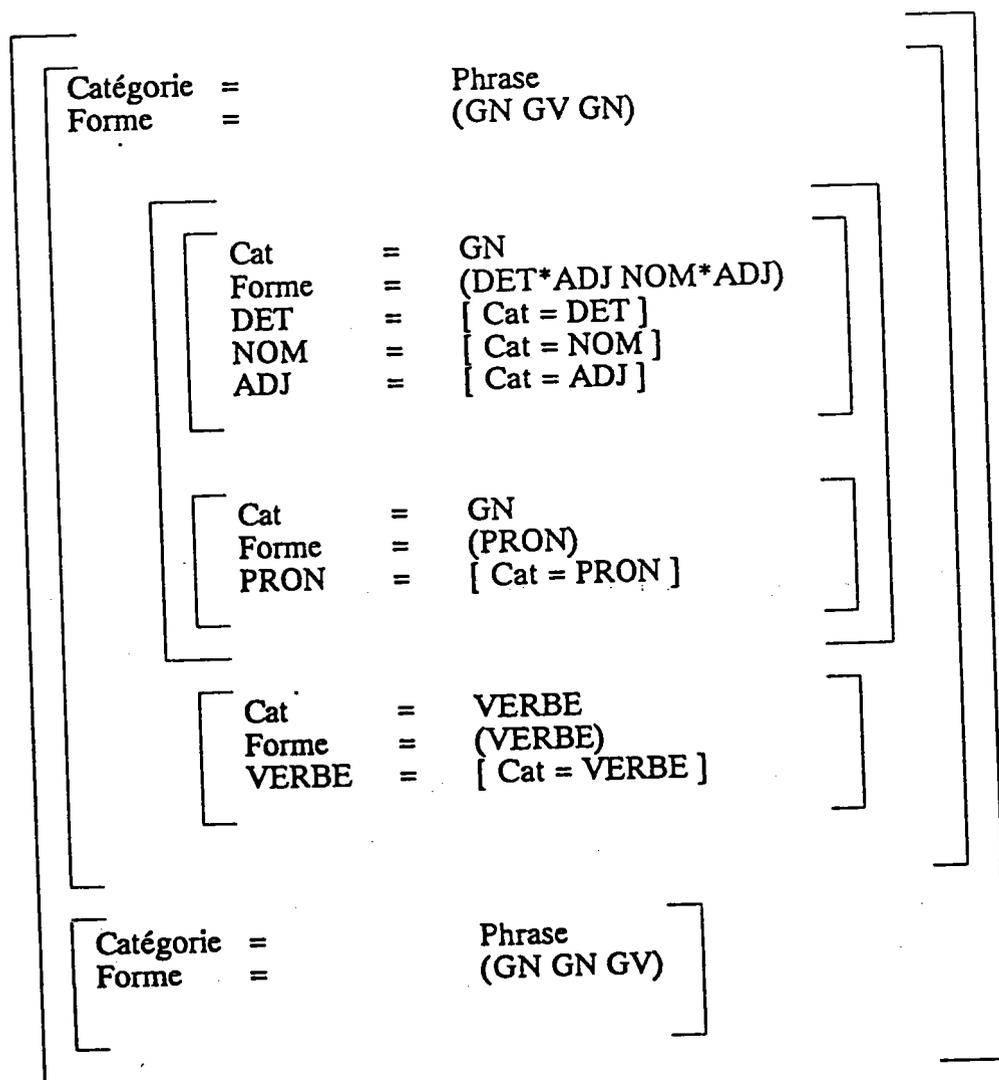
- exemple de formalisme appliqué aux règles grammaticales. Soit une grammaire ne permettant que les deux types de phrase suivants :

GN + GV + GN  
GN + GN + GV

Les GN sont des pronoms ou des chaînes composées d'un déterminant, d'un nombre quelconque d'adjectifs, d'un nom suivi à nouveau d'un nombre quelconque d'adjectifs.

Le GV est formé d'un verbe seul.

En indiquant par \* que l'élément suivant est facultatif ou répété un nombre quelconque de fois, la description fonctionnelle correspondant à la grammaire ci-dessus aura la forme :



- exemple de formalisme appliqué à la représentation interne d'une phrase :  
La représentation de la structure de la phrase qui permettra d'appréhender son sens est construite à partir des règles de grammaire et des représentations lexicales.

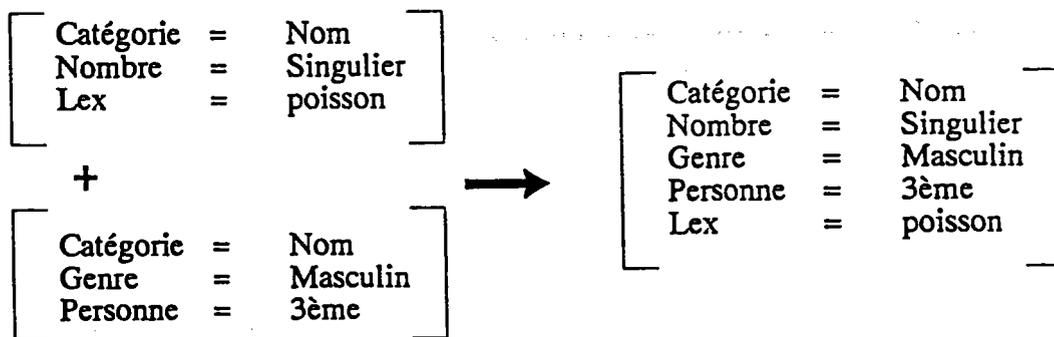
Représentons le groupe nominal *le poisson* :

Cat	=	GN																
Forme	=	(DET NOM)																
DET	=	<table border="1"> <tr> <td>Cat</td> <td>=</td> <td>Article</td> </tr> <tr> <td>Type</td> <td>=</td> <td>Défini</td> </tr> <tr> <td>Nombre</td> <td>=</td> <td>Sing.</td> </tr> <tr> <td>Genre</td> <td>=</td> <td>Masculin</td> </tr> <tr> <td>Lex</td> <td>=</td> <td><i>le</i></td> </tr> </table>	Cat	=	Article	Type	=	Défini	Nombre	=	Sing.	Genre	=	Masculin	Lex	=	<i>le</i>	
Cat	=	Article																
Type	=	Défini																
Nombre	=	Sing.																
Genre	=	Masculin																
Lex	=	<i>le</i>																
NOM	=	<table border="1"> <tr> <td>Cat</td> <td>=</td> <td>NOM</td> </tr> <tr> <td>Nombre</td> <td>=</td> <td>Sing.</td> </tr> <tr> <td>Genre</td> <td>=</td> <td>Masculin</td> </tr> <tr> <td>Lex</td> <td>=</td> <td><i>poisson</i></td> </tr> </table>	Cat	=	NOM	Nombre	=	Sing.	Genre	=	Masculin	Lex	=	<i>poisson</i>				
Cat	=	NOM																
Nombre	=	Sing.																
Genre	=	Masculin																
Lex	=	<i>poisson</i>																

On note que la catégorie syntaxique est GN et qu'il est constitué d'un déterminant et d'un nom. Des schémas en précisent les différents traits avec la lexicalisation correspondante (LEX).

#### 1.4.3.6.1.3 La superposition

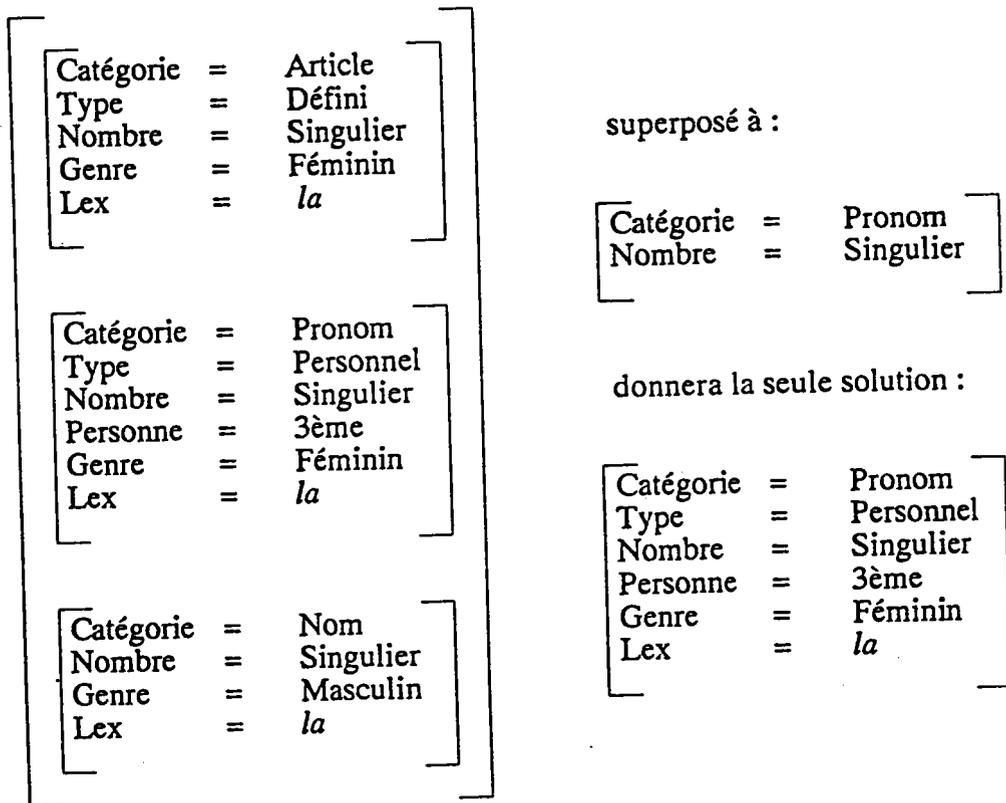
Lors de la construction de la représentation interne, on est parfois amené à utiliser la notion de superposition pour construire une description fonctionnelle à partir de deux autres descriptions fonctionnelles, à condition que des attributs identiques n'aient pas de valeurs incompatibles



S'il permet de compléter une description à partir de descriptions partielles (dans l'exemple, descriptions du concept *poisson* et de la catégorie lexicale nom masculin), le mécanisme de superposition permet de résoudre des ambiguïtés lexicales.

On parvient ainsi à la description d'un langage qui permet d'explicitier les règles de grammaire, de représenter les éléments du dictionnaire et les structures internes de la phrase, ce qu'il est possible d'assimiler à une grammaire indépendante du contexte.

Un exemple de résolution d'une ambiguïté par superposition serait :



#### 1.4.3.6.1.4 La notion de chemin

La notion de chemin qui traduit des références aux divers éléments d'une structure décrite par ce langage, détermine des contraintes qui augmentent la puissance du modèle. On décrit un chemin dans une description fonctionnelle DF1 par la notation  $\langle a_1, a_2, \dots, a_n \rangle$ .

exemple : test d'accord, en genre et en nombre, dans le groupe nominal, du déterminant et des adjectifs avec le nom.

GN =	Cat = GN Forme = (DET*ADJ NOM*ADJ) DET = [ Cat = DET ] NOM = [ Cat = NOM ] ADJ = [ Cat = ADJ ] EGAL = <DET, GENRE>, <NOM, GENRE> EGAL = <ADJ, GENRE>, <NOM, GENRE> EGAL = <DET, NOMBRE>, <NOM, NOMBRE> EGAL = <ADJ, NOMBRE>, <NOM, NOMBRE>
------	--

Les attributs GENRE et NOMBRE ne figurent pas pour DET, NOM et ADJ. Le mécanisme de superposition, appliqué à cette règle et à la description lexicale, les feront apparaître.

L'avantage de ce formalisme tient à l'unité dans les descriptions et à la facilité d'intégration des différents types de connaissances.

### 1.4.3.6.2 Les grammaires lexicales fonctionnelles

Développées par J. BRESNAN et R. KAPLAN<sup>1</sup>, ces grammaires privilégient les lexiques et donnent des descriptions partielles qui, grâce à la notion de description additive, peuvent ensuite se compléter.

La structure de constituants ou *C-structure* (aspects syntaxiques) et la structure fonctionnelle ou *F-structure* (aspects sémantiques) contribuent à la description interne d'une phrase.

La *C-structure* est le résultat d'une analyse de la phrase par une grammaire indépendante du contexte. Cette dernière autorise des structures incorrectes qui seront éliminées par la *F-structure*.

La *F-structure* s'appuie sur la notion de schéma telle que nous venons de l'introduire, débarrassée de la notion de *forme*, désormais inutile puisque les aspects syntaxiques sont traités par la *C-structure*. Associées aux règles de la grammaire, des équations génèrent la structure fonctionnelle. Elles sont écrites selon la notation LISP.

(A,B) représente "l'attribut B de l'élément A".

Ces équations doivent être considérées comme liées aux arcs de l'arbre engendré.

↑ désigne la *F-structure* du nœud père  
↓ désigne la *F-structure* du nœud fils

L'équation ( ↑ Nombre ) = singulier attachée à l'article *un* dans le lexique, signifiera que le nombre de la *F-structure* du déterminant est singulier, (cet élément sera en effet lié à un nœud de type déterminant après l'étape d'insertion lexicale).

De même, pour la règle :

$P \longrightarrow GN \quad GV$

on a les deux équations :

( ↑ Sujet ) = ↓ et ↑ = ↓

↑ désigne "P", ↓ successivement GN et GV.

Cette règle précise que les constituants de la phrase sont un GN et un GV.

( ↑ Sujet ) = ↓ Cette première équation indique que le sujet de la *F-structure* de la phrase est identique à la *F-structure* du 1er groupe nominal.

↑ = ↓ Les structures de P et de GV sont égales.

En matière d'entrée lexicale, pour *un* et *donne*, par exemple, on peut trouver la description suivante :

---

(1) J. BRESNAN, R. KAPLAN : "Lexical Functional Grammars ; A Formal System for Grammatical Representation" in *The Mental Representation of Grammatical Relations*, J. BRESNAN (ed.), MIT press, 1981, Cambridge Mass.

un Déterminant (↑Définition) = indéfini  
(↑Nombre) = singulier

donne Verbe (↑Temps) = présent  
(↑Prédicat) = donner < (↑S), (↑O), (↑O2) >

La structure du prédicat comprend 3 actants, le sujet et les deux objets du verbe. On note ainsi au niveau du lexique les formes que prendra une représentation logique dans laquelle interviendra le mot.

Le même formalisme permet de décrire les règles de grammaire qui entreront par la suite dans le calcul de la C-structure et la résolution des équations menant à la F-structure.

Règle 1 :

Deux équations sont associées à la règle :

$P \rightarrow GN + GV$

La phrase est formée d'un groupe nominal et d'un groupe verbal.

Règle 2 :

$(\uparrow \text{ sujet}) = \downarrow$

Le sujet de la F-structure et le GN ont la même structure.

$\uparrow = \downarrow$

P et GV ont la même structure

Règle 3 :

$GN \rightarrow \text{Déterminant} \quad \text{Nom}$

C-structure de GN

Trois équations sont associées à la règle :

$GV \rightarrow \text{Verbe}$

GN

GP

C-structure de GV

$\uparrow = \downarrow$

Les structures du GV et du verbe sont identiques.

$(\uparrow \text{ Obj.-dir.}) = \downarrow$

La structure du 1er GN est celle de l'objet direct du GV.

$(\uparrow \text{ Obj.-ind.}) = \downarrow$

La structure du 2ème GN est celle de l'objet indirect du GV.

### 1.4.3.6.2.1 La production d'une phrase

Elle s'effectue en trois étapes :

- 1ère étape : La grammaire indépendante du contexte, sans que l'on tienne compte des équations liées aux règles, produit un arbre de dérivation dont toutes les feuilles sont des catégories lexicales.
- 2ème étape : C'est l'insertion lexicale avec l'attribution à chaque feuille d'un mot approprié du dictionnaire. On actualise ensuite les équations en substituant aux flèches les variables qui conviennent.
- 3ème étape : Les équations sont résolues et donnent la structure fonctionnelle de la phrase.

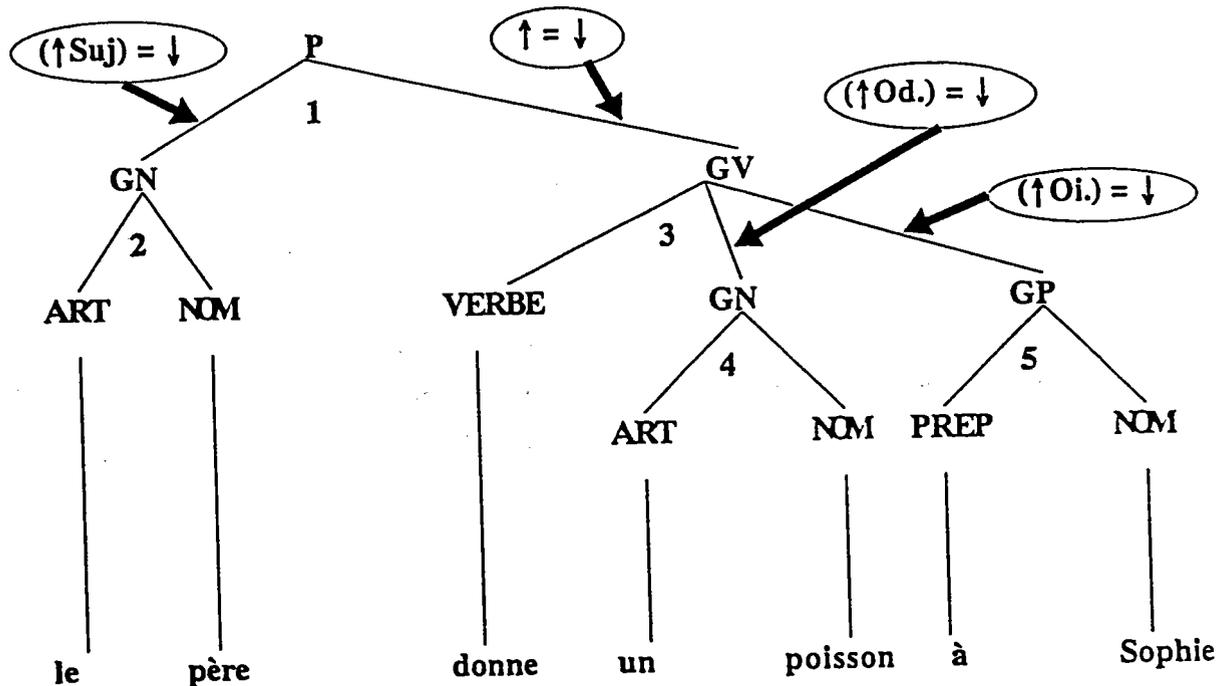
### 1.4.3.6.2.2 La représentation d'une phrase

Elle repose ainsi sur deux éléments : la C-structure et le F-structure.

La C-structure donne une représentation syntaxique de la phrase, on y distingue deux types de nœuds :

- les catégories syntaxiques (P, GN, GV, GP, ...) auxquelles sont associées les variables notées  $x_j$  (1 variable par nœud).
- les nœuds lexicaux (art., nom, prép., verbe, ...) placés juste au dessus des nœuds terminaux (les mots).

A partir de cette structure, des règles de grammaire et des règles associées au lexique, on déduit des équations qui permettront de construire la structure fonctionnelle :



P est sur le nœud  $x_1$

GN et GV sont sur les nœuds  $x_2$  et  $x_3$

GN et GP sont sur les nœuds  $x_4$  et  $x_5$

Règle de grammaire :  $P \longrightarrow GN \quad GV$

En remplaçant les flèches dans les équations correspondantes, on obtient :

$(\uparrow \text{sujet}) = \downarrow \longrightarrow x_1 \text{sujet} = x_2$

$\uparrow = \downarrow \longrightarrow x_1 = x_3$

de même, pour GV :

$$\begin{aligned} (\uparrow \text{Obj.}-\text{dir.}) = \downarrow &\longrightarrow (x_3 \text{Obj.}-\text{dir.}) = x_4 \\ (\uparrow \text{Obj.}-\text{ind.}) = \downarrow &\longrightarrow (x_3 \text{Obj.}-\text{ind.}) = x_5 \end{aligned}$$

Règles du lexique :

$$\begin{aligned} \text{le} \quad \text{Déterminant } (\uparrow \text{Définition}) = \text{défini} &\longrightarrow (x_2 \text{Définition}) = \text{défini} \\ &(\uparrow \text{Nombre}) = \text{singulier} &\longrightarrow (x_2 \text{Nombre}) = \text{singulier} \end{aligned}$$

$$\begin{aligned} \text{père} &\longrightarrow (x_2 \text{Concept}) = \text{père} \\ &\longrightarrow (x_2 \text{Nombre}) = \text{singulier)... \end{aligned}$$

Les équations résolues, on obtient la structure fonctionnelle (représentation sémantique) suivante :

$$x_1 = x_3 = \left[ \begin{array}{l} \text{Sujet} = x_2 \\ \text{Obj.}-\text{dir.} = x_4 \\ \text{Obj.}-\text{ind.} = x_5 \\ \text{Temps} = \text{présent} \\ \text{Concept} = \text{donner} \langle \text{père, poisson, Sophie} \rangle \end{array} \right]$$

$$x_2 = \left[ \begin{array}{l} \text{Définition} = \text{défini} \\ \text{Nombre} = \text{singulier} \\ \text{Concept} = \text{père} \end{array} \right]$$

$$x_4 = \left[ \begin{array}{l} \text{Définition} = \text{indéfini} \\ \text{Nombre} = \text{singulier} \\ \text{Concept} = \text{poisson} \end{array} \right]$$

$$x_5 = \left[ \begin{array}{l} \text{Définition} = \text{défini} \\ \text{Concept} = \text{Sophie} \end{array} \right]$$

### 2.5.3.6.2.3 Phénomènes à distance, éléments discontinus

Les symboles  $\uparrow X$  et  $\downarrow X$  ( $X$  est une catégorie grammaticale) représentent successivement la F-structure de l'élément contrôlé et celle du contrôleur. A chaque  $\uparrow X$  correspondra un  $\downarrow X$ .

$\uparrow = \uparrow X$  et  $\downarrow = \downarrow X$  signifient que les deux F-structures doivent se correspondre. Pour traiter des liaisons à distance comme dans la phrase *Combien de poissons le père donne-t-il à Sophie ?* on ajoute deux règles :

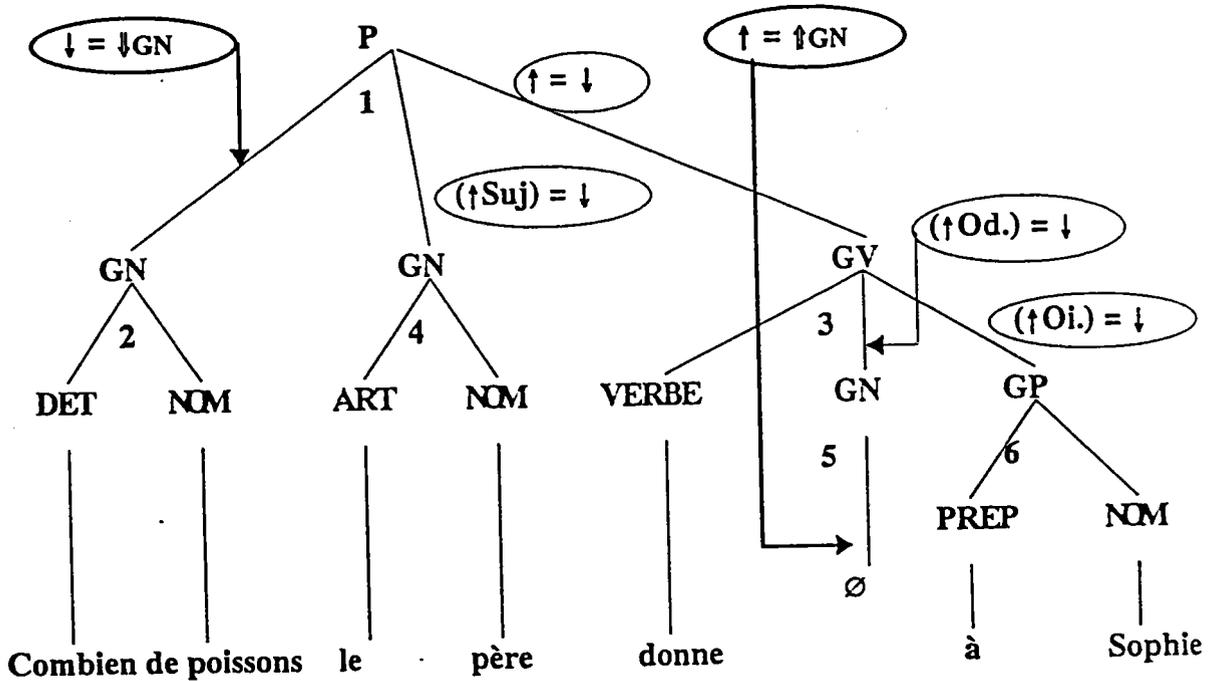
1. P  $\longrightarrow$  GN GN GV avec les équations  $\downarrow = \downarrow X$ ,  $(\uparrow \text{sujet}) = \downarrow$  et  $\uparrow = \downarrow$

2. GN  $\longrightarrow$   $\emptyset$  avec l'équation  $\uparrow = \uparrow \text{GN}$

La règle 2 indique que n'importe quel GN peut se réécrire en un élément nul. Une C-structure pourra du reste contenir un nombre quelconque de tels GN puisque la grammaire est indépendante du contexte.

Les symboles  $\uparrow$  et  $\downarrow$  fonctionnant par paire, l'analyse se soldera par un échec s'il y a un déséquilibre car les équations associées aux règles ne pourront être résolues.

La C-structure sera :



La structure fonctionnelle correspondante :

$x1 = x3 =$ 

Forme	= question
Sujet	= x4
Obj.-dir.	= x5
Obj.-ind.	= x6
Temps	= présent
Concept	= donner <père, poisson, Sophie>

$x4 =$ 

Définition	= défini
Nombre	= singulier
Concept	= père

$x2 = x5 =$ 

Définition	= indéfini
Nombre	= singulier
Concept	= poisson

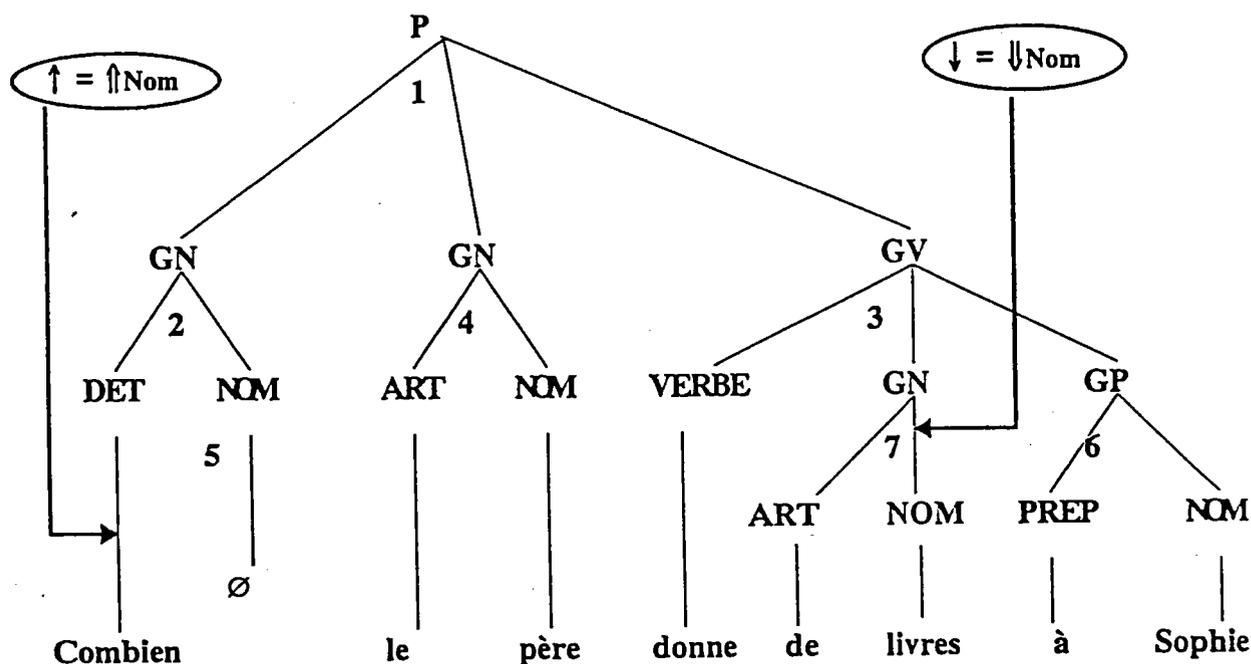
$x6 =$ 

Définition	= défini
Concept	= Sophie

La dépendance à distance est exprimée dans la F-structure par l'égalité  $x_2 = x_5$   
 Un mécanisme identique permet de représenter des constituants discontinus comme dans la phrase *Combien le père donne de livres à Sophie ?* en ajoutant les règles :

1. GN  $\longrightarrow$  DET NOM avec l'équation  $\downarrow = \Downarrow\text{NOM}$
2. GN  $\longrightarrow$   $\emptyset$  avec l'équation  $\uparrow = \Uparrow\text{NOM}$

On obtient le schéma :



#### 1.4.3.6.3 Le lexique-grammaire de M. Gross

En partant comme Z. HARRIS d'un modèle de la langue le moins formalisé possible, M. GROSS dresse un inventaire systématique des diverses constructions syntaxiques du français.

##### 1.4.3.6.3.1 Les dictionnaires du LADL (Laboratoire Documentaire et Linguistique)

DELAS est un dictionnaire de 50 000 mots dans lequel sont stockés les formes canoniques et des codes indiquant les types de flexion des mots, indépendamment des sens possibles.

DELAF réunit les 350 000 formes fléchées associées, générées automatiquement à partir de DELAS.

DELAP donne une représentation phonétique à chaque entrée de DELAS.

Des analyseurs, des générateurs, des programmes de "syllabation" et de classement ainsi que différents outils logiciels sont utilisés pour la construction de programmes de correction orthographique.

Il faut également mentionner le recensement des expressions complexes, de locutions courantes et de noms composés.

#### 1.4.3.6.3.2 Les lexiques-grammaires

Le traitement automatique du langage naturel nécessite une représentation systématique des langues naturelles, utilisable par des automates. Traditionnellement le lexique, la grammaire et la sémantique sont couverts par les dictionnaires et les grammaires classiques. En situation d'apprentissage, les êtres humains s'accommodent d'une part d'implicité. Les ordinateurs, par contre, ont besoin de descriptions exhaustives, répertoriées et formalisées pour une application informatique comme la traduction automatique. Les dictionnaires évoqués précédemment tiennent surtout compte des données morphologiques.

Les aspects syntaxiques élémentaires (relations du verbe avec d'autres catégories lexicales par ex.) sont abordés au cours d'autres travaux qui consistent à décrire systématiquement un mot (selon son sens, pour tenir compte de l'influence du sens sur la syntaxe) par rapport à environ 500 propriétés syntaxiques.

M. GROSS a entrepris d'élaborer un dictionnaire électronique des verbes du français (lexique-grammaire), qu'il conçoit comme une grande base structurée de données lexicales, associant à la description de chaque entrée les règles syntaxiques d'emplois, sous forme de schémas d'insertion de l'item verbal. Achievé pour 12 000 verbes, le dictionnaire morphosyntaxique du Laboratoire Documentaire et Linguistique (LADL) présente plusieurs milliers de schémas descriptifs.

On constate ensuite que ces propriétés syntaxiques utilisées comme critères de classement, mettent en évidence, pour les verbes étudiés, le fait que les verbes sémantiquement apparentés ont souvent des propriétés syntaxiques communes.

#### 1.4.3.6.4 Le modèle Sens-Texte de I. MEL'CUK<sup>1</sup>

Le modèle de I. MEL'CUK et de A.K. ZOLKOVSKIJ appartient aux modèles informatiques du langage. Il est organisé comme un processus qui mettrait en correspondance une représentation du sens (représentation sémantique) avec un texte (énoncé écrit ou oral).

Il a pour but

- d'expliquer une même idée de plusieurs façons
- d'identifier diverses expressions d'apparence différentes mais ayant le même sens.

Le sens n'est ni l'interprétation des expressions décrites par la syntaxe (modèles de N. CHOMSKY et modèles de sémantique formelle), ni étudié indépendamment du texte. Le sens est défini comme "l'invariant des transformations synonymiques (les paraphrases)".

Loin des modèles de N. CHOMSKY dont les structures profondes sont essentiellement des représentations syntaxiques, le modèle est hiérarchisé en 7 niveaux. Ses structures lexico-syntaxiques de base sont en effet des structures qui représentent le sens.

---

(1) I. MEL'CUK, A.K. ZOLKOVSKI : *Vers un modèle "Sens-Texte" du langage*, Documents de Linguistique Quantitative, n° 10, Paris, Dunod, 1971

Des phrases en relation paraphrastique ont une même représentation sémantique sous forme d'une même SLSB mais se distinguent par les SLS différentes. Les FS regroupent des phrases synonymes dont les SLSB sont toutefois distinctes (à cause des variations lexicales).

L'ensemble de ces modèles et de ces théories peut sembler disparate. Le manque d'unité n'est qu'apparent. Après avoir illustré des points de vue essentiellement syntaxiques et formels (grammaires formelles, grammaires transformationnelles, théorie standard, théorie standard étendue, théorie des traces, grammaire en chaîne), nous avons parcouru les techniques plus récentes, orientées vers les aspects sémantiques (grammaire syntagmatique généralisée, grammaire de cas, grammaire lexicale fonctionnelle), les aspects fonctionnels (grammaire systématique, grammaire fonctionnelle) et le rôle du lexique (théorie de M. GROSS et modèle "Sens-Texte" de I. MEL'CUK).

#### 1.4.4 Conclusion

La possibilité d'écrire et d'utiliser conjointement les connaissances lexicales, syntaxiques, sémantiques et même pragmatiques confère un grand avantage aux grammaires d'unification. Issues des grammaires transformationnelles, elles s'en démarquent pourtant nettement, avec le rôle qu'elles accordent à la fonction grammaticale, à la structure et au lexique qui est fondamental. Les grammaires fonctionnelles introduisent des notions de variable qui les disposent au traitement automatique. Elles fournissent l'appareil théorique nécessaire à la TAO. Examinons maintenant les outils informatiques.

### 1.5 Les outils

#### 1.5.1 Introduction

Nous observerons ici les aspects opératoires, autrement dit, les techniques informatiques qui vont utiliser les connaissances représentées selon les modèles que nous avons parcourus dans le paragraphe 1.4. Nous présenterons les modules classiques d'analyse morphologique, syntaxique et morphosyntaxique, puis les analyseurs déterministes, les analyseurs dirigés par le lexique, les outils logiques dans le sens où ils ont un usage spécifiquement lié au langage naturel, les outils intégrés. Nous concluons sur le traitement des différentes erreurs.

Nous laissons de côté les techniques mises en oeuvre pour expliciter les relations entre les phrases (les inférences, les références et les grammaires de récit). Elles appartiennent au traitement automatique des langues mais contrairement aux modèles linguistiques évoqués plus haut, concernent des applications précises, distinctes de la Traduction Automatique. Nous ne nous intéresserons pas non plus aux mécanismes de génération, puisque l'objet de notre travail se limite à l'analyse.

#### 1.5.2 L'analyse morphologique

On peut distinguer plusieurs types d'analyse selon le but poursuivi. Il s'agit en général de reconnaître les mots sous leurs différentes formes, en distinguant la *forme canonique* (une entrée du dictionnaire par exemple) et les *formes fléchies* (selon le rôle qui leur est assigné dans la phrase et les marques qui leur sont attribuées, comme celles du genre, du nombre ...). Cela permet d'effectuer des recherches ultérieures dans les dictionnaires de formes canoniques.

Pour analyser correctement une forme fléchie, il est indispensable de pouvoir la décomposer en repérant la racine, les affixes et les désinences. Des problèmes supplémentaires interviennent lorsque le mot est composé, comme nous le verrons pour l'al-

lemand (3.3.3.2 et 4.3.2.5). Le problème des irrégularités renvoie à la structure complexe du mot, qui ne peut souvent être appréhendée sans un recours à l'étymologie, aux phénomènes d'emprunt et aux mécanismes de formation que sont la composition et la dérivation. Isoler les racines et maîtriser les affixes (préfixes, suffixes) implique que l'on organise ces données en systèmes, selon les buts poursuivis.

La construction de tels systèmes et la gestion des influences des bases<sup>1</sup> et des affixes sur le sens, évitent la construction de dictionnaires qui ne sont jamais exhaustifs, et facilitent la compréhension des mots nouveaux ou inconnus. C'est un des principes que nous avons retenus pour notre système et qui sera développé dans 3.2. Le traitement d'une langue flexionnelle comme l'allemand permet de mesurer les avantages d'une telle démarche. Nous verrons à ce propos qu'il est souvent difficile d'identifier clairement le sens d'un préfixe ou d'un suffixe et qu'il est plus intéressant de raisonner sur une base. Le traitement des composés se heurte au problème de leur sens qui n'est pas toujours égal à la juxtaposition des sens de chaque terme. Dans le même ordre d'idée, le sens de certaines expressions n'est pas forcément déductible des sens des mots qui les composent. Ces expressions figées ou semi-figées sont, de plus, aussi fréquentes que les expressions non spécifiques<sup>2,3</sup>. Il est indispensable de les localiser dès l'étape de l'analyse morphologique, ce qui ne pourra se faire qu'à partir d'une base de recensement.

### 1.5.2.1 Les techniques

Il y a deux types de dictionnaires.

1 - *Le dictionnaire de formes canoniques* qui nécessite un module d'analyse et un module de génération, pour retrouver la forme d'entrée à partir de la forme fléchie. Il est économique en mémoire de stockage puisque les différentes données concernant une entrée ne sont enregistrées qu'une fois. Le temps de calcul est cependant très important, puisqu'il faut découper le mot pour retrouver la forme canonique.

2 - *Le dictionnaire de toutes les formes fléchies* occupe beaucoup plus de place (les différents traits sont répétés pour toutes les variantes d'un même mot) mais le calcul est très rapide, le traitement se bornant à une recherche du mot dans le dictionnaire.

Les avantages et les inconvénients se compensent de sorte que c'est la taille du dictionnaire dont on a besoin qui détermine la méthode appropriée. Pour des petits dictionnaires, dans le contexte d'un système expert ou d'un système d'E.A.O. (Enseignement Assisté par Ordinateur), on préférera stocker toutes les formes fléchies. Pour d'importantes masses de données, il faudra optimiser la vitesse d'accès au disque et les temps de réponse. L'efficacité des algorithmes de recherche dépend de la structuration des données.

Parmi les méthodes, peu nombreuses, nous citerons :

- la recherche dichotomique

*avantages* : Elle ne recourt à aucune structure de recherche, ce qui la rend d'autant

(1) La base est l'association d'un préverbe ou d'un préfixe à une racine.

(2) M. GROSS : "Une classification des phrases "figées" du français". *Revue Québécoise de linguistique*, 11,2, Presses de l'Université du Québec, Montréal, 1982

(3) L. DANLOS : "La morphosyntaxe des expressions figées", *Langage* n° 63, *Formes syntaxiques et prédicats sémantiques*, Larousse, Paris, 1981

rapide en mémoire vive. L'emplacement des données recherchées n'a aucune influence sur la vitesse d'accès.

*inconvénients* : Les informations doivent être triées, ce qui est pénalisant pour la réactualisation du dictionnaire. L'insertion ou le retrait deviennent en effet très coûteux. Les enregistrements (entrées du dictionnaire avec les traits correspondants) doivent être de taille fixe, en pratique, ce qui implique un gaspillage de la mémoire. Dans le cas d'un tri sur une mémoire de masse (disque dur, bande ou cartouche), on n'élimine que la moitié de l'espace précédent à chaque accès disque.

- la méthode de l'adressage dispersé, (on calcule le numéro d'ordre du mot, dans le dictionnaire, et on lui associe une adresse disque dans une table de correspondance)

*avantages* : Les enregistrements peuvent être rangés dans le désordre et être de longueur variable. Il devient alors possible de les classer dans une autre perspective (pour accélérer la recherche de mots liés ou phonétiquement voisins par exemple). L'information est extraite très rapidement, généralement en un accès, parfois deux.

*inconvénients* : Lorsque l'on construit le dictionnaire ou au cours d'une recherche, les calculs peuvent donner le même résultat pour des mots distincts. L'actualisation du dictionnaire est très délicate et la table de correspondance encombre la mémoire vive. Le choix d'enregistrements fixes permet d'en faire l'économie. On peut également limiter les risques de collision en augmentant la marge de sécurité.

- la méthode des arbres de Bayer<sup>1</sup> présente les mêmes avantages avec un accès direct dans le dictionnaire. On représente la table de correspondance entre les mots et leurs adresses sous la forme d'un arbre n-aire. On range sur chaque noeud, une adresse et un pivot, en alternance.

On compare l'élément recherché au premier pivot. S'il est supérieur, on s'intéresse alors à l'adresse de droite (correspondant à un autre pivot dans l'arbre). S'il est inférieur, on s'intéresse à l'adresse de gauche (et on recommence avec le nouveau pivot).

*avantages* : méthode régulière, le temps de recherche est indépendant du mot cherché. La plus grande liberté est laissée au concepteur qui peut choisir les spécifications d'enregistrement (fixe ou variable) et le nombre des couples adresse-pivot attachés aux noeuds. Les ajouts ou les expressions dans le dictionnaire ne posent plus de problèmes et on peut se contenter de stocker un seul noeud dans la mémoire vive.

*inconvénient* : la mise en oeuvre informatique est assez complexe.

### 1.5.2.2 T. WINOGRAD

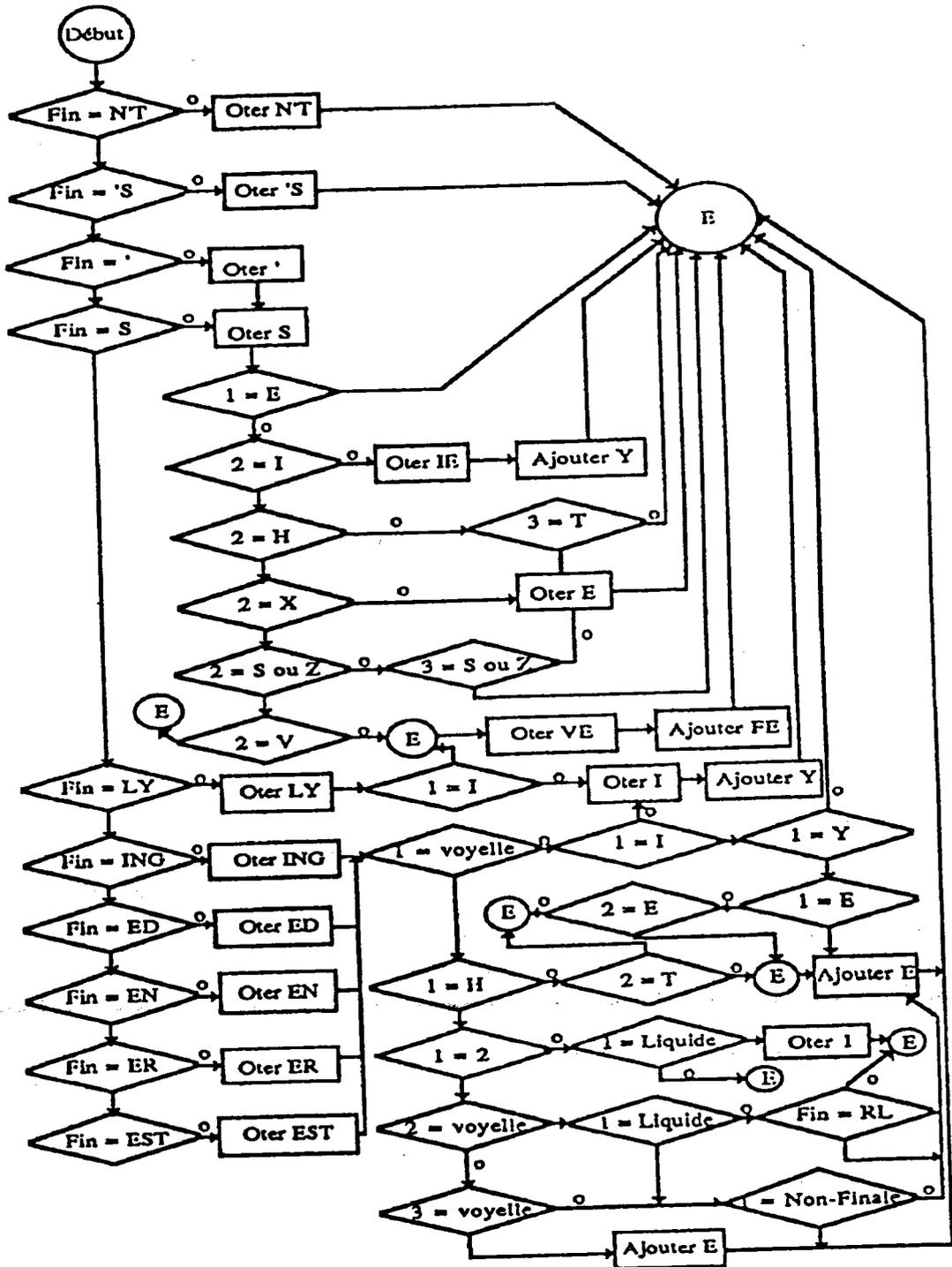
Le programme SHRDLU<sup>2</sup> fonctionne avec un dictionnaire construit sur les formes canoniques. Une méthode procédurale permet de partir d'une forme fléchie pour arriver à la forme canonique, ou inversement.

Le schéma ci-dessous, extrait de "Understanding Natural Language", (WINOGRAD 1972), s'applique à l'analyse de l'anglais. Il repose sur la suppression de certaines fins de mots et le rajout éventuel de séquences pour reconstruire la forme connue.

---

(1) D. KNUTH : *The Art of the Computer Programming*, volume 2, (2ème édition), Addison Wesley, Reading, Mass, 1973

(2) T. WINOGRAD : *Understanding Natural Language*, Academic Press, Edinburgh, 1972



1. L'analyse débute par un test de comparaison entre la fin du mot et le segment spécifié par l'algorithme (Fin = \*). S'il y a coïncidence, la fonction Oter débarrasse le mot analysé du segment prévu.
2. Les fonctions 1, 2, 3, ... indiquent des lettres qui se trouvent à la fin du mot après qu'il ait été raccourci lors de l'étape précédente.

Dans le bas du schéma, à droite, on reconnaît des ensembles de lettres :

voyelles : a, e, i, o, u, y

liquides : l, r, s, v, z

non finales : c, g, s, v, z (que l'on ne peut trouver en fin de racine).

**E** est le test d'essai qui recherche dans le dictionnaire la présence de la chaîne de caractères obtenue.

exemple : *prettily* - *pretty*

1. test de la terminaison. On repère *ly* et on le supprime. *prettily* - *pretti*

2. 1=I est vérifié, le mot se terminant par i.

ôter I : on ôte le i - *Prett*

ajouter y : on ajoute y - *Pretty*

exemple : *kisses* - *kiss*

1. test de la terminaison. On repère s et on le supprime. *kisses* - *kisse*

2. 1=E est vérifié. Le mot se termine par e.

E n'aboutit pas, il n'y a pas de forme canonique *kisse*

2=S ou Z est vérifié. L'avant dernière lettre est bien un s

3=S ou Z également.

ôter E - *kiss*

Pour les exceptions, on ajoute des règles à l'algorithme ou on entre la forme dans le dictionnaire. Les résultats doivent ensuite être validés par des filtres de vérification morphosyntaxiques, ou désambiguïsés lors des étapes suivantes.

### 1.5.2.3 J. PITRAT

J. PITRAT<sup>1</sup> a construit un système qui fonctionne, en analyse et en génération, avec un dictionnaire contenant toutes les données d'une langue. Général dans son fonctionnement, il suffit de modifier les données pour l'adapter à une autre langue, cette propriété ayant été vérifiée sur 10 langues. Les données regroupent l'ensemble des mots (fichier mots), l'ensemble des désinences possibles (fichier terminaisons) et l'ensemble des conjugaisons/déclinaisons (fichier conjugaisons).

En considérant toutes les désinences a priori possibles, de la plus courte à la plus longue, le système examine pour chacune s'il existe une racine dans le lexique, (les racines sont recensées avec leurs variantes *recev*, *reçoi*, *reçoiv*... et les terminaisons qui peuvent leur être associées). Il vérifie que la nature grammaticale du mot ainsi trouvé concorde avec la désinence. Les ambiguïtés seront levées aux niveaux supérieurs.

Fichier des mots :

Mot	Nom de conjugaison	Racines
Tenir	Venir	Tien, ten, tienn, tin, tîn...

(1) J. PITRAT : "Réalisation d'un analyseur-générateur lexicographique général", rapport de recherche n° 79/2, GR22, Institut de programmation, Paris VI, 1983

Fichier des terminaisons :

	Nom des terminaisons(T)	T1, T2, T3...Tn
(1)	VIP	s, s, t, ons, ez, ent
(2)	NMF	-, e, s, es

L'exemple (1) concerne ici les terminaisons (s, s, t, ons, ez, ent), pour certains verbes, à l'indicatif présent (VIP).

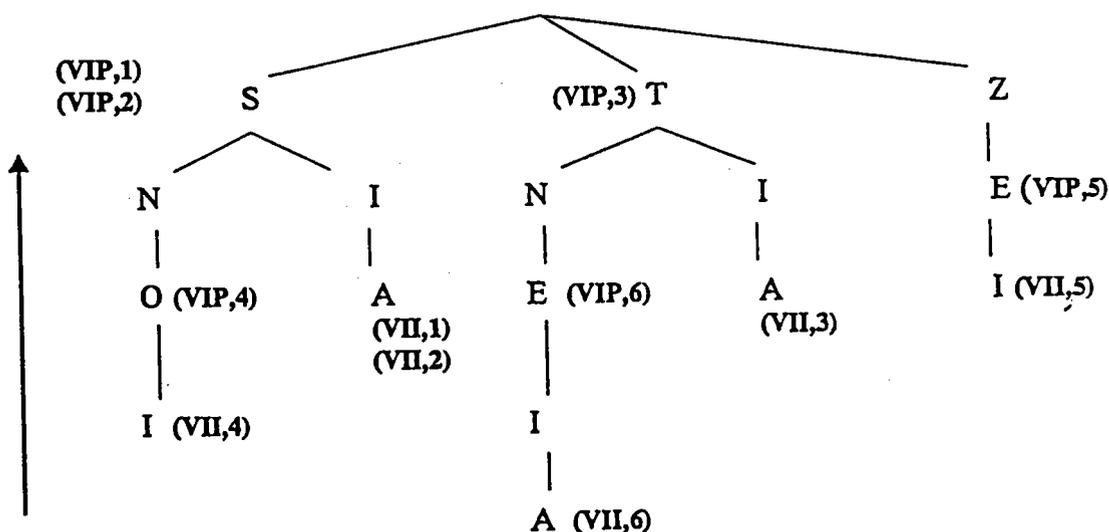
L'exemple (2) concerne les noms et les adjectifs (au masculin et au féminin) avec les désinences masculin sing., féminin sing., masculin plur., féminin plur.

Fichier des conjugaisons :

	conjugaison C	Groupe G	Terminaison T	i1, i2, i3...iq n° des racines
(1)	VENIR	IP	VIP	111223
(2)		II	VII	222222

La colonne 1 renvoie au fichier des mots, la colonne 3 au fichier des terminaisons.  
 La colonne 4 signifie que i1, i2, i3 = racine n°1, i4, i5 = racine n°2, i6 = racine n°3  
 L'entrée (1) de la figure ci-dessus indique que pour le groupe IP (Indicatif Présent), on utilise les terminaisons VIP (Verbe Indicatif Présent) et dans le cas présent (Tenir), on associe les trois premières désinences (T1=s, T2=s, T3=t) aux racines (I1=tien, I2=tien, I3=tien), puis les deux désinences suivantes (T4=ons, T5=ez) aux racines (I4=ten, I5=ten), et enfin T6=ent à I6=tienn.  
 Le chiffre 0 correspondra aux formes défectives.

L'analyse fonctionne de la droite vers la gauche. Les terminaisons du fichier sont utilisées sous une forme arborescente.



Chaque noeud contient le nom des terminaisons T qui correspond à la séquence allant du noeud au sommet, ainsi que sa position dans les terminaisons.

Soit la forme *TENAIENT*  $n=8$   
Le mot s'écrit  $a_1, a_2, a_3, \dots a_n$ .

Lors de l'analyse, on part de la gauche, en incrémentant le pointeur  $i$  d'un caractère.

Pour  $i=3$ , on a deux parties :

$a_1, a_2, \dots a_i = TEN$   
 $a_n \dots a_{i+2}, a_{i+1} = TNELA$

On recherche la présence de *AIENT* dans l'arborescence. Si le test est vrai, on vérifie que la première partie  $a_1, a_2 \dots a_i$  est une racine connue.

L'opération de génération est bien plus directe que l'opération d'analyse car on possède toutes les informations utiles dès le départ.

L'organisation en arborescence accélère le traitement automatique. Il est intéressant, par conséquent, de présenter les racines sous cette même forme.

#### 1.5.2.4 Problèmes annexes

L'analyse morphologique doit également tenir compte de phénomènes divers et essentiels, variables selon les langues.

- la ponctuation nécessite un traitement des ponctuations simples qui suivent immédiatement le mot, avec la reconnaissance des virgules et des points décimaux,
- les nombres composés, les expansions d'éllision, les tirets de composition, les contractions
- la majuscule de début de phrase

En ce qui concerne les dictionnaires électroniques, nous avons cité le DELAS<sup>1</sup> et ses 50 000 entrées de formes canoniques, capables d'engendrer 350 000 formes fléchies. On peut ajouter le BDLEX<sup>2</sup>. Les dictionnaires évoluent sans cesse et ne seront jamais terminés. Les termes techniques, les néologismes, les sigles sont autant de défis.

Nous dirons en conclusion que l'organisation et l'utilisation d'un lexique dépend de l'application visée. L'exhaustivité, dans ce domaine, est impossible. Certaines études<sup>3</sup> ont mis au point des listes limitées, capables de couvrir de très nombreuses formes (4000 formes pour 90% des occurrences d'un texte de français courant)<sup>4</sup>. La taille des lexiques ne pose pourtant pas de problèmes majeurs aux informaticiens qui savent gérer de grandes masses de données.

---

(1) G. GROSS : "Typologie des noms composés", *Rapport ATP CNRS*, Université Paris Nord Villeta-neuse, 1986

(2) G. PERENNOU, M. de CALMES : "BDLEX base de données lexicale du français écrit et parlé", *Rapport du GRECO "Communication parlée"*, 1987

(3) G. GOUGENHEIM : *Dictionnaire fondamental de la langue française*, Didier, Paris, 1958

(4) N. CATACH : *Les listes orthographiques de base*, Collection Recherche, Nathan, Paris, 1984

Dès que la taille des lexiques devient importante, et puisqu'il est illusoire de vouloir recenser l'ensemble des mots utiles pour une application, il convient de prévoir une organisation incrémentielle<sup>1</sup> des lexiques<sup>2</sup>.

La vocation du lexique est de mettre les constituants de base à la disposition des analyseurs syntaxiques et sémantiques. Les véritables difficultés sont de deux ordres.

- L'élément de base ne correspond pas toujours à un mot (locutions, mots composés...).
- La dépendance entre l'élaboration de la structure syntaxique et la composante sémantique. Il est impossible de mener concomitamment l'analyse des différents aspects d'un texte. On est donc souvent contraint de retenir une première hypothèse, en se ménageant des possibilités de retour en arrière.

### 1.5.3. Analyse de la phrase

Nous étudierons dans ce chapitre, les techniques informatiques qui permettront, à partir des constituants d'une phrase et des relations qu'ils entretiennent entre eux, de lui assigner une structure interne censée représenter sa "signification".

Ce mécanisme utilise des connaissances générales, externes, relatives à la langue et à l'univers contextuel. Présentées le plus souvent sous forme déclarative, elles recouvrent :

la grammaire avec les connaissances syntaxiques issues des modèles linguistiques que nous avons évoqués dans le paragraphe 1.4.

les mots : Ce sont les connaissances morphologiques et lexicales

les concepts : pour les connaissances sémantiques

le monde pour les données pragmatiques

Les rapports qu'entretiennent ces connaissances entre elles sont exprimées par des relations.

Ces connaissances, la grammaire plus particulièrement, sont étroitement liées à la forme de description chargée de représenter la phrase étudiée.

#### 1.5.3.1 La grammaire et la description interne de la phrase

Cette influence des modèles linguistiques sur la représentation interne d'une phrase peut être importante lorsqu'il s'agit de systèmes fondés sur une théorie grammaticale. Nous verrons dans les paragraphes 1.5.7 (analyseurs conceptuels) et 1.5.4.3 (L'ATN) des analyseurs indépendants de toute théorie linguistique ou des mécanismes adaptables à diverses représentations.

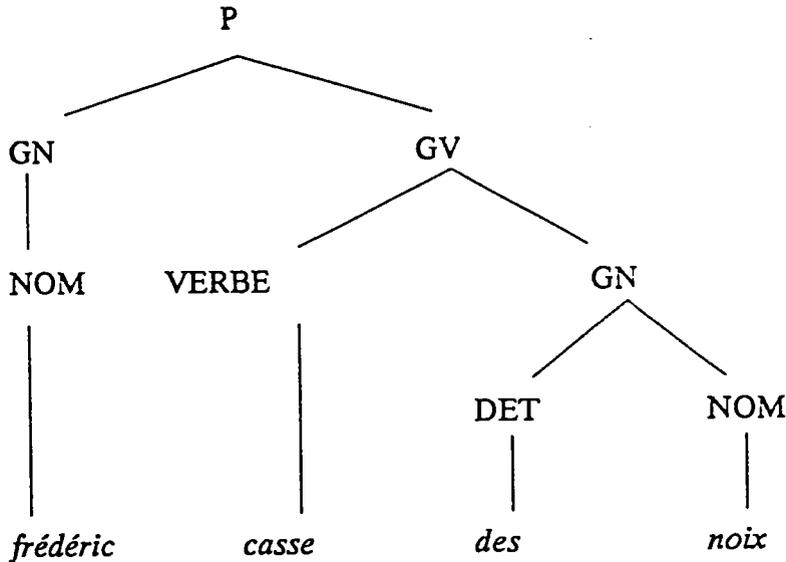
---

(1) L'organisation incrémentielle permet d'étendre les lexiques pas à pas, c'est à dire d'y ajouter des enregistrements, un à un.

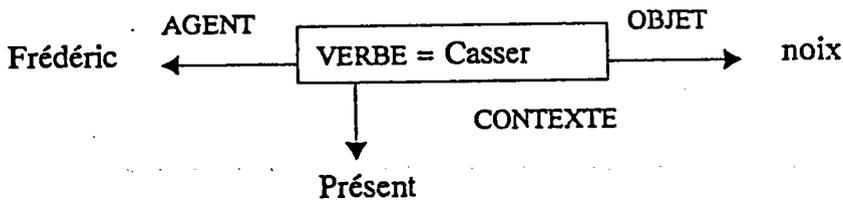
(2) Une méthode générale pour la constitution des lexiques et leur implémentation sur ordinateur est décrite dans : J. GOETSCHALCKX, L. ROLLING (eds.) : *Lexicography in the Electronic Age*, North Holland, 1982

A partir de la phrase *Frédéric casse des noix*, nous allons construire les structures correspondant à quelques modèles cités dans le chapitre 1.4.

La grammaire formelle (1.4.3.1) produit l'arbre ci-dessous :



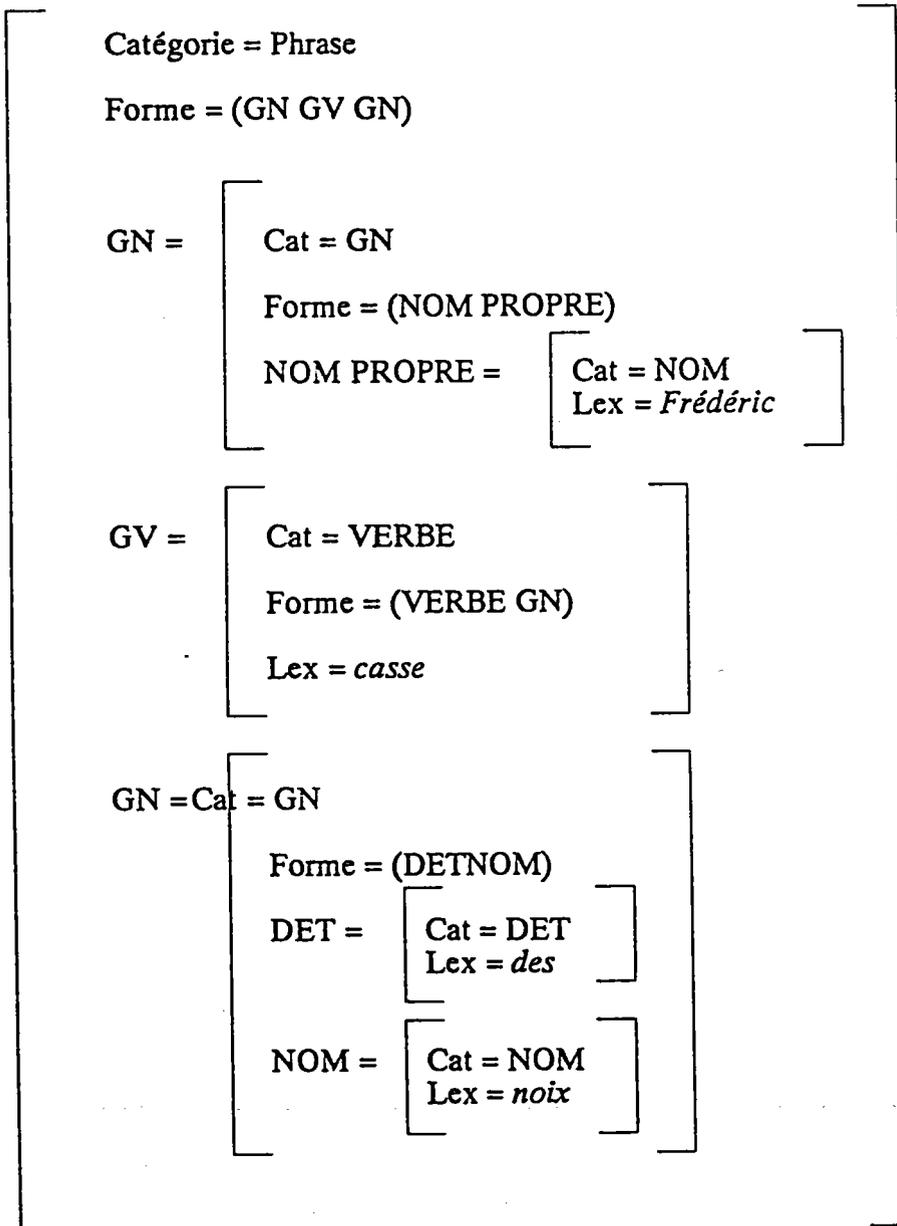
La grammaire de cas (1.4.3.3) construit un graphe avec des arcs étiquetés et introduit des notions sémantiques.



Les grammaires systémiques (2.5.3.5) introduisent des points de vue complémentaires.

Système de MODE	Principale, déclarative, assertion
Système d'ACTANTS	Agent : <i>Frédéric</i> Objet : <i>noix</i> , direct, pluriel
Système THEMATIQUE	Thème <i>Frédéric</i> Prédicat <i>casser(Frédéric, noix)</i>
Système INFORMATIF	connu : <i>Frédéric casse</i> nouveau : <i>ce sont des noix qu'il casse</i>

Les grammaires fonctionnelles (1.4.3.6.2) donnent une représentation plus complète.



On constate que la représentation interne revêt des aspects assez différents selon le modèle dont elle est issue et contient plus ou moins d'informations sémantiques.

### 1.5.3.1.1 Syntaxe et sémantique

De nombreux aspects syntaxiques posent problème. Nous en avons relevés quelques uns déjà, (comparatifs/superlatifs, ellipses, anaphores, références, temps, portée des quantificateurs et de la négation...). Il n'existe pas de système capable de les résoudre tous. Chaque système dépend d'une application. Il correspond au niveau de complexité des phrases que l'on décide d'accepter ou au degré de grammaticalité que l'on tolère.

En construisant un arbre de précedence (N. CHOMSKY) ou un arbre de dépendance (I. MEL'CUK), l'analyseur syntaxique reconnaît la structure de la phrase traitée en iden-

tifiant le verbe, son sujet, et des objets... Les grammaires syntagmatiques généralisées (1.4.3.2.4.3) et les grammaires lexicales fonctionnelles (1.4.3.6.2) sont les plus utilisées actuellement. Parmi les nombreuses applications en T.A. et T.A.O. on peut citer ATLAS I (2.3.3.2.1), HICATS (2.3.3.2.2) et METAL (2.3.3.2.5).

Quel que soit le traitement ultérieur, l'analyse syntaxique doit montrer que des phrases distinctes peuvent avoir des structures identiques, si l'on veut traiter des transformations, des phénomènes particuliers (ellipses, anaphores...) et organiser une compréhension du sens.

On évoque ici la structure sémantique que l'on peut aborder en utilisant :

- des contraintes sémantiques qui, lorsqu'elles ne sont pas respectées, invalident les arbres syntaxiques construits
- des combinaisons de règles syntaxiques et de règles sémantiques qui, par interrétaction, associent une représentation sémantique à la représentation syntaxique (grammaire de MONTAGUE et grammaires syntagmatiques généralisées 1.4.3.2.4.3)

Certains modèles privilégient plutôt la sémantique et intègrent une syntaxe simplifiée<sup>1</sup>. Ils s'appuient sur les grammaires sémantiques. D'autres modèles privilégient la pragmatique (Dialogues).

### 1.5.3.2 Traitement automatique

On peut considérer qu'il y a deux façons de procéder :

- l'analyse se déroule mot à mot, les décisions sont toutes prises au fur et à mesure et prises en compte dans l'élaboration des représentations.
- l'analyse peut revenir en arrière pour supprimer ce que G. SABAH<sup>2</sup> appelle les points d'embarras (état du système où les éléments de décision ne permettent pas d'opérer un choix parmi les traitements possibles). Les avantages sont certains, mais s'accompagnent d'une grosse consommation de mémoire et rallongent le temps de travail. La majorité des systèmes réalisent une analyse mot à mot, de la gauche vers la droite.

#### 1.5.3.2.1 L'architecture des systèmes

Les modules sont liés aux sources de connaissances qu'ils utilisent et reflètent de ce fait les distinctions classiques (morphologie, syntaxe, sémantique).

L'architecture traduit le type des interactions qu'ils entretiennent.

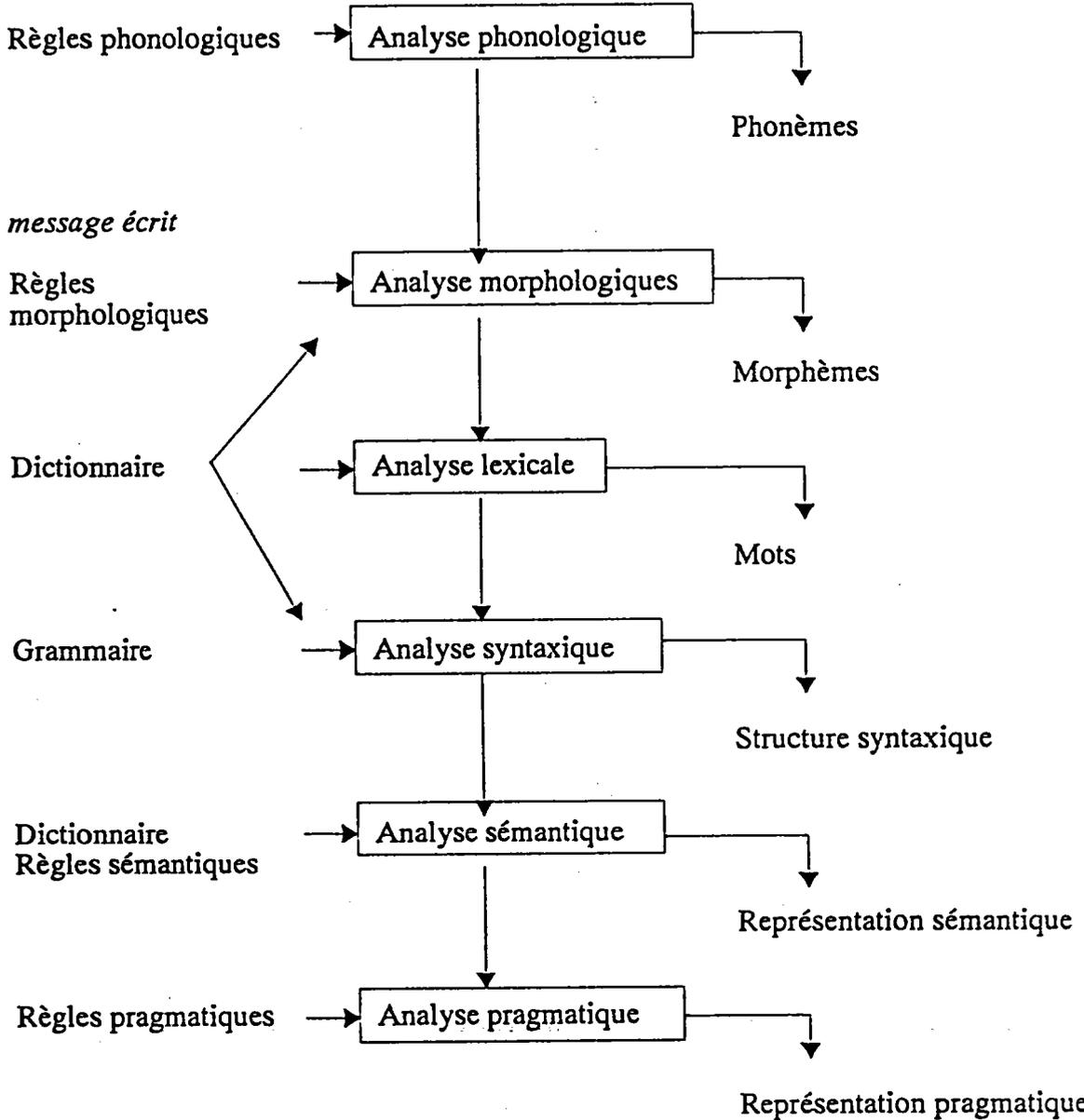
- Dans une architecture en série, les modules s'enchaînent, dans le même sens, le fonctionnement du module  $n$  n'ayant aucune influence sur celui du module  $n-1$ . Ceci peut être source d'ambiguïté artificielle.

---

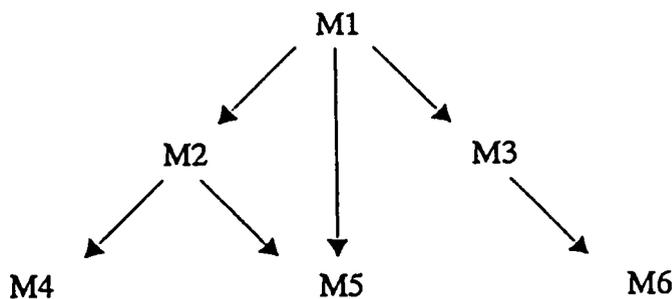
(1) Y. WILKS : "An Artificial Intelligence Approach to Machine Translation", in : Computer Models of Thought and Language, Scank & Colby, Freeman, San Francisco, 1973, pp. 114-151

(2) G. SABAH : L'intelligence artificielle et le langage. Processus de compréhension, vol. n°2, Hermes, Paris, 1989, pp. 46-47

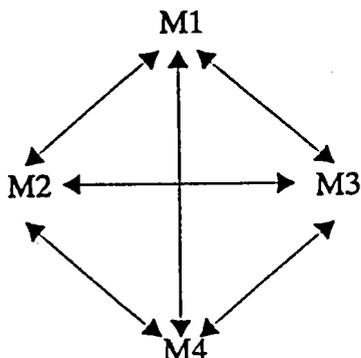
message oral



- *Dans une architecture hiérarchique*, l'analyse dépend d'un module principal. Le contrôle peut échoir à un sous module. Il n'y a jamais de retour en arrière. Si elle apporte quelques remèdes aux insuffisances d'une construction en série, elle est délicate à mettre en oeuvre et entraîne une limitation des interactions entre modules.



- Dans l'architecture libre, chaque module peut contrôler chaque module.



- Le seul moyen d'éliminer tous les points d'embarras est de renoncer à la modularité, au profit d'une intégration des éléments de décision qui rappelle l'organisation de certaines de nos connaissances. Ces dernières sont malheureusement si complexes que certains travaux<sup>1,2</sup> ont pu valider la méthode sur un plan théorique mais se sont heurtés à des problèmes de gestion insurmontables. T. WINOGRAD<sup>3</sup> s'était demandé "comment concevoir un formalisme gardant les avantages de la décomposabilité sans sacrifier l'expression des interactions?" avant de proposer la notion de schéma comme une première tentative de synthèse. Les recherches pour mieux contrôler les différentes sources de connaissances tout en conservant leur modularité ont conduit à la notion de tableau noir, qui permet à des bases de connaissances distinctes et considérées comme indépendantes d'intervenir ensemble, sans communiquer entre elles.

### 1.5.3.2.2 L'analyse

Construisons une grammaire de type génératif capable d'engendrer la phrase :

*Un arbre cache la forêt*

1. P -> GN GV
2. GN -> DET NOM
3. GV -> VERBE GN
4. DET -> un / la
5. NOM -> arbre / forêt
6. VERBE -> cache

Le processus qui consiste, en partant du symbole P, à le remplacer par la partie droite, (Les nouveaux symboles pouvant être à leur tour réécrits par l'application de nouvelles règles qui peuvent intervenir plusieurs fois), est un processus de génération.

(1) M. RADY : "L'ambiguïté du langage naturel est-elle la source du non-déterminisme des procédures de traitement ?", *Thèse de Doctorat ès Sciences*, Paris, Université Pierre et Marie Curie, 1983

(2) L. SELIGMAN : "Intégration de la syntaxe, de la sémantique et de la pragmatique dans un analyseur de textes. Application à l'avionique." *Thèse de l'Université Pierre et Marie Curie*, Paris, 1985

(3) T. WINOGRAD : *Understanding Natural Language*, Academic press, Edinburgh, 1972

Or notre but est de réaliser une analyse. Il nous faut donc, en partant de la phrase à analyser, choisir les règles qui nous mèneront au résultat, en fonction des mots de la phrase.

Nous disposons de deux méthodes :

- L'analyse descendante, dirigée par les hypothèses, part de P et compare avec la phrase. S'il y a correspondance entre les éléments le plus à gauche, on les élimine, sinon on réécrit l'élément en appliquant une règle possible.

P	<i>un arbre cache la forêt</i>
GN/GV	<i>un arbre cache la forêt</i>
DET/NOM/GV	<i>un arbre cache la forêt</i>
<i>un</i> /NOM/GV	<i>un arbre cache la forêt</i>
NOM/GV	<i>arbre cache la forêt</i>
<i>arbre</i> /GV	<i>arbre cache la forêt</i>
GV	<i>cache la forêt</i>
VERBE/GN	<i>cache la forêt</i>
<i>cache</i> /GN	<i>cache la forêt</i>
GN	<i>la forêt</i>
DET/NOM	<i>la forêt</i>
<i>la</i> /NOM	<i>la forêt</i>
NOM	<i>forêt</i>
<i>forêt</i>	<i>forêt</i>

L'application successive des règles 1, 2, 4, 5, 3, 2, 4, 5 permet de construire l'arbre syntaxique de la phrase.

- Avec l'analyse montante, dirigée par les données, on part des mots de la phrase pour arriver aux structures grammaticales possibles.

On lit le mot puis on lui substitue une catégorie. On essaie ensuite de réécrire ces éléments pour arriver à P, en appliquant les règles à l'envers. (on réécrit le membre droit par le membre gauche). Quand on reconnaît une séquence, on la remplace par la catégorie correspondante.

*un arbre cache la forêt*

règle 4	DET <i>arbre cache la forêt</i>
règle 5	DET NOM <i>cache la forêt</i>
règle 2	GN <i>cache la forêt</i>
règle 6	GN VERBE <i>la forêt</i>
règle 4	GN VERBE DET <i>forêt</i>
règle 5	GN VERBE DET NOM
règle 2	GN VERBE GN
règle 3	GN GV
règle 1	P

Les deux méthodes peuvent être à l'origine d'ambiguïtés artificielles. Certains chercheurs<sup>1</sup> prônent l'utilisation conjointe des deux techniques.

T. WINOGRAD a traité ces aspects dans le détail<sup>2</sup>.

Les exemples que nous avons choisis ne présentent aucune difficulté pour les analyseurs. Différentes techniques ont été élaborées afin que le système, comparé à un point

(1) Groupe Langage et Cognition du LIMSI, Université Pierre et Marie Curie, Paris

(2) T. WINOGRAD : *Language as a Cognitive Process, Volume I, Syntax*, Addison Wesley, pp. 93-109, 1983

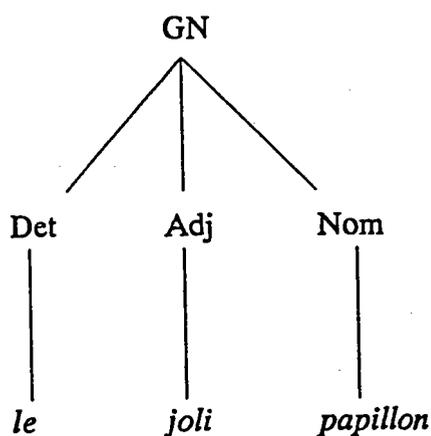
d'embarras, puisse fonctionner malgré l'insuffisance des critères de décision. Le traitement en parallèle consiste à construire toutes les solutions possibles, pour chaque point d'embarras. La production de toutes les analyses, à propos d'un énoncé, exige énormément de temps calcul.

La stratégie de retour en arrière consiste à privilégier arbitrairement une solution parmi les solutions possibles. Celles qui sont rejetées sont stockées dans une pile avec toutes les informations décrivant l'état du système, et ceci, pour chaque point d'embarras. Lorsque l'analyse est dans l'impasse, on repart du sommet de la pile après avoir remis le système dans l'état correspondant.

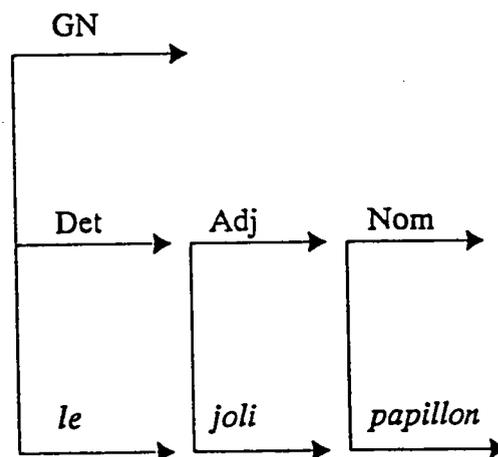
Lorsque le système est bloqué et que la pile est vide, il y a échec de l'analyse.

On peut assouplir cette stratégie en classant les décisions a priori ou en les choisissant selon des heuristiques. Le système placé sur un point d'embarras prendra alors les décisions les plus probables. (Réseaux de Transitions Augmentés, paragraphe 1.5.4.3).

La plupart des analyseurs préfèrent la stratégie du retour en arrière, malgré l'inconvénient d'un temps d'analyse qui peut devenir très long, si les mauvaises solutions sont utilisées d'abord. Ces stratégies non déterministes ne sont pas forcément indispensables, comme nous le verrons dans le paragraphe 1.5.5 à propos des analyseurs déterministes. La notion de "table" cherche à intégrer les avantages des deux méthodes sans en avoir les inconvénients. J. KAPLAN<sup>1</sup> a imaginé une structure de données adaptée à un traitement optimal des possibilités multiples. Son algorithme n'effectue que le traitement approprié à une situation donnée, et ne l'effectue qu'une fois. Il introduit la notion de "chart" pour représenter la grammaire et la structure de la phrase. Les arcs de l'arbre traduisent les relations verticales (dominance) et les relations horizontales (précédence). Pour obtenir une table, on modifie l'arbre selon deux axes. Les arcs sont transformés pour produire un arbre binaire. Les arcs et les noeuds sont échangés : les arcs de l'arbre devenant les sommets du chart et les noeuds de l'arbre devenant ses arêtes. Les sommets représentent alors les séparateurs de mots, et les arêtes les mots ou séquences de mots.



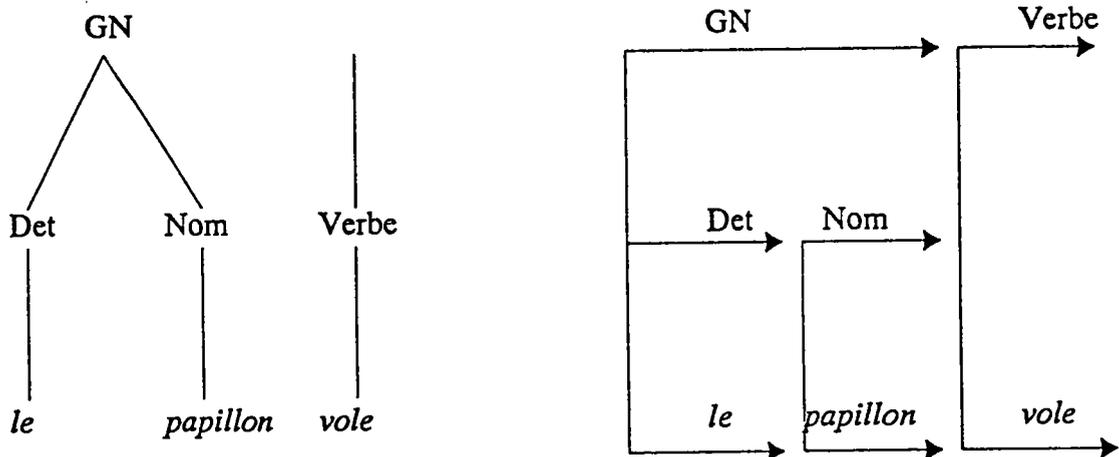
Relations de dominance : explicites  
Relations de précédence : implicites



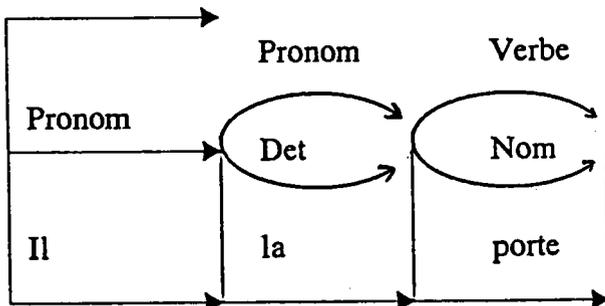
Relations de dominance : implicites  
Relations de précédence : explicites

(1) J. KAPLAN : "A General Syntactic Processor" in *Natural Language Processing*, Rustin, Algorithmic Press, New York, pp. 193-241, 1979

La table peut coder un ensemble d'arbres disjoints :



On peut également représenter les ambiguïtés, en reliant deux mêmes noeuds par plusieurs arêtes. Les étiquettes de ces arêtes indiqueront les interprétations correspondantes.



### 1.5.3.3 Exemples

Pour illustrer les paragraphes précédents, nous présentons des analyseurs qui ne seront pas étudiés dans le détail, par la suite. Nous les classerons en fonction des grammaires utilisées, qui seront, dans le cas présent, des grammaires formelles.

#### Grammaires indépendantes du contexte

- "L'analyseur prédictif" de Harvard<sup>1</sup> : Cet analyseur descendant est basé sur une grammaire non contextuelle (133 classes de mots, 2100 règles)<sup>2</sup>. La taille de cette grammaire s'est vite révélée très insuffisante et les temps de calcul beaucoup trop longs (1 minute pour une phrase de 18 mots sur les ordinateurs de l'époque).

(1) S. KUNO, A. OETTINGER : "Multiple Path Syntactic Analyser" in *Information Processing*, North Holland, Amsterdam, 1962

(2) S. KUNO : "The Multiple Path Syntactic Analyser for English", *Rapport NSF-9, Mathematical Linguistics and Automatic Translation*, Computation Laboratory, Harvard University, Cambridge Mass., 1963

L'utilisation d'une matrice destinée à supprimer les essais inutiles<sup>1</sup> n'a pas réussi à démentir le fait que ce type de formalisme n'était pas vraiment adapté au traitement automatique.

- *L'analyseur en constituants immédiats*<sup>2</sup> : Développé chez RAND, il est basé sur un algorithme montant rapide, qui occupe une place importante en mémoire et utilise des règles de grammaire écrites sous la forme normale de N. CHOMSKY.

### Grammaires transformationnelles

- *Le système de Z. HARRIS*<sup>3</sup> : à stratégie montante, il est basé sur une grammaire en chaîne et reconnaît divers types de chaînes d'une grammaire qui correspond à des règles non contextuelles. Si les analyseurs concernaient les phrases isolées, Z. HARRIS imagine les processus de transformation pour relier des phrases qui ont des éléments en commun.

L'analyse est plus complexe qu'avec une grammaire à structures de phrases.

- *Le système de A. ZWICKY*<sup>4</sup> : Basé sur une grammaire transformationnelle générative (275 règles et 54 transformations pour le module de base), il dispose d'un ensemble de 550 règles et de 134 transformations inverses (36 minutes pour analyser une phrase de 12 mots).

Pour l'améliorer, D. WALKER<sup>5</sup> a proposé la notion de "super-arbre", identique au chart de J. KAPLAN, et de règles de rejet.

- *Le système de S. PETRICK*<sup>6,7</sup> était plus général donc plus lent encore que le système de A. ZWICKY. Il a été complètement remanié<sup>8</sup> et est utilisé comme analyseur dans un système de question-réponse<sup>9</sup> où il analyse des requêtes en moins d'une minute<sup>10</sup>.

### Grammaires en chaîne

Les théories de Z. HARRIS sont peu utilisées mais présentent deux avantages pour l'automatisation.

- La structure en chaîne et la structure transformationnelle d'une phrase sont en correspondance étroite. L'analyse en chaîne est d'autant plus adaptée à une décomposition transformationnelle.

---

(1) S. KUNO : "The Predictive Analyser and a Path Elimination Technique", *Communications ACM* 8, 7, pp. 453-462, 1965

(2) D HAYS : *Introduction to Computational Linguistics*, American Elsevier, New York, 1967

(3) Z. HARRIS : *String Analysis of Sentence Structure*, Mouton and Co, La Hague, 1962

(4) A. ZWICKY, J. FRIEDMAN, B. HALL, D. WALKER : "The MITRE Syntactic Analysis Procedure for Transformational Grammars", *Actes Fall Joint Computer Conference*, Thompson Books, Washington DC, 1965

(5) D. WALKER, P. CHAPLIN, M. GEIS, L. GROSS : "Recent Developments in the MITRE Syntactic Analysis Procedure", *Rapport MITRE MTP.11*, 1965

(6) S. PETRICK : "A Recognition Procedure for Transformational Grammars", *Thèse de PhD*, dep. Modern Languages, MIT, Cambridge, Mass., 1965

(7) S. PETRICK : "A Program for Transformational Syntactic Analysis". *Rapport AFCRL-66-698*, Air Force Cambridge Research Laboratory, 1966

(8) S. PETRICK : *Transformational Analysis in Natural Language Processing*, Rustin, Algorithmics Press, New York, 1973

(9) W. PLATH : REQUEST : "A Natural Language Question-answering System", *IBJ Journal of research and development*, 20, 4, pp. 326-335, 1976

(10) F. DAMERAU : "Operating Statistics for the Transformational Question-answering System", *American Journal of computational linguistics*, 7, 1, pp. 32-40, 1981

- La grammaire en chaîne décrit facilement certaines relations entre constituants, car elle est assez proche des structures de surface.

- *L'analyseur de N. SAGER* a bénéficié des améliorations apportées par R. GRISHMAN<sup>1</sup> avec "le langage de restrictions" et J. HOBBS<sup>2</sup> avec un mécanisme d'analyse transformationnelle.

- *L'analyseur de J.H. JAYEZ*<sup>3</sup> traite des intitulés en français à l'aide d'une grammaire en chaîne du français développée par M. SALKOFF<sup>4</sup>.

### Grammaires contextuelles

- *Le système DEACON* (Direct English Access and Control) a été développé pour l'armée, chez General Electric<sup>5</sup>. Basé sur une grammaire contextuelle, le système est un des premiers à organiser des ponts systématiques entre le composant syntaxique et un composant sémantique. W. WOODS<sup>6</sup> a construit un algorithme pour améliorer les performances de ce type d'analyseur qui produisait des analyses redondantes.

### Grammaires de type 0

- *Le système de M. KAY* utilise les règles de réécriture non restreintes, auxquelles on ajoute des mécanismes de restriction qui n'entament pas le pouvoir génératif de la grammaire mais en facilitent l'écriture. Inclus dans le système REL (Rapidly Extensible Language)<sup>7</sup>, il fait appel à une grammaire de 239 règles.

- *Le système Q de A. COLMERAUER*<sup>8</sup> fonctionne également avec des règles non restreintes. Ce système a été utilisé dans le projet de traduction automatique de l'université de Montréal<sup>9</sup>.

- Hewlett-Packard a développé un analyseur basé sur les GPSG de G. GAZDAR (Generalized Phrase Structure Grammars)<sup>10</sup>. Des métrarègles développent les règles de la grammaire non contextuelle, avant l'analyse de la phrase. (40 règles et 10 métrarègles produisent une grammaire finale de 283 règles). Le système analyse les interrogations d'une base de données relationnelle.

---

(1) R. GRISHMAN : "Implementation of the String Parser of English" in *Natural Language Processing*, Rustin, Algorithmics Press, New York, 1973

(2) J. HOBBS, R. GRISHMAN : "The Automatic Transformational Analysis of English Sentences : an Implementation", *International Journal of Mathematics*, section A, 5, pp. 267-283, 1976

(3) J.H. JAYEZ : "Une approche de la compréhension par machine du langage naturel", *Thèse d'Etat*, Paris VII, 1979

(4) M. SALKOFF : *Une grammaire en chaîne du français. Analyse distributionnelle*, Paris, Dunod, 1973

(5) J.A. CRAIG, S.C. BERENZER, H.C. CARNOY, C.R. LONGYEAR : "DECON : Direct English Access and Control", *Actes Fall Joint Computer Conference*, Thompson Books, Washington DC, 1966

(6) W. WOODS : "Context-sensitive Parsing", *Communications ACM*, 13, 7, pp. 437-445, 1970

(7) F. THOMPSON, P.C. LOCKEMAN, B. DOSTERT, R.S. DEVERILL : "REL : a Rapidly Extensible Language System", *Actes de la 24ème conférence nationale ACM*, 1969

(8) A. COLMERAUER : "Les systèmes Q ou un formalisme pour analyser et synthétiser des phrases sur ordinateur", *publication interne n°43*, Faculté des Sciences, Université de Montréal, 1970

(9) R. KITTREDGE et al. : *TAUM73 Rapport sur le projet de traduction automatique de l'université de Montréal*, 1973

(10) J.M. GAWRON et al. : "Processing English with a Generalized Phrase Structure Grammar", *Actes du 20ème séminaire annuel de l'association Computational Linguistics*, pp. 74-81, 1982

## Grammaires systématiques

- Le programme de T. WINOGRAD est un des rares à avoir utilisé ce type de grammaire. Il intègre sous une forme procédurale, les modules syntaxiques, sémantiques et pragmatiques.

### 1.5.4. Les réseaux de transitions

#### 1.5.4.1 Introduction

Pour mettre en oeuvre les différents modules qui décrivent la structure des phrases, on dispose de certaines procédures qui ne leur sont pas totalement liées.

Les grammaires formelles et transformationnelles donnent des modèles orientés vers la génération. Bien que leur utilisation pour l'analyse soit difficile, nous avons cité quelques réalisations dans le paragraphe 1.5.3.3. Les algorithmes opèrent des transformations inverses et dont le nombre croît exponentiellement.

Ces difficultés ont conduit les chercheurs à substituer aux règles de grammaire la notion de "représentation des formes de phrases acceptables". A la suite des travaux de S. KUNO<sup>1</sup> avec l'élaboration d'une procédure exprimant les transformations inverses sous forme de réseaux, les chercheurs ont tenté d'utiliser directement ces réseaux pour l'analyse de la langue.

Le principe de ces réseaux<sup>2</sup> est de "réunir les parties droites des règles d'une grammaire hors-contexte qui ont la même partie gauche en un diagramme explicitant les transitions correspondantes. Pour décrire des phrases comme :

*La jeune fille chante*  
*Le vilain chat griffe le facteur*

on pourra utiliser les séquences :

DET ADJ NOM VERBE  
DET ADJ NOM VERBE DET ADJ NOM

Une représentation<sup>3</sup> plus économique serait :

(PRON U (DET ADJ \* NOM)) VERBE { (DET ADJ \* NOM) }

U symbole d'alternative (on peut remplacer DET ADJ \* NOM par un pronom)

\* symbole de répétition (*la fille, la jeune fille, la belle jeune fille, la belle grande jeune fille...*)

{...} le segment est optionnel

Le réseau de transitions permet d'éviter ce type de notation très lourde dans le cas de phrases complexes.

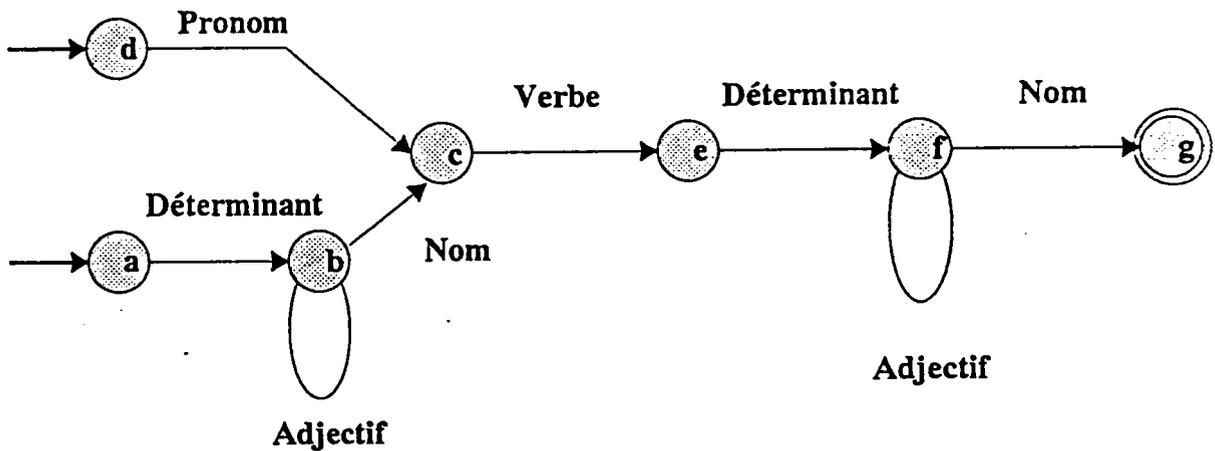
(1) S. KUNO : "A System for Transformational Analysis", *Rapport NSF-15, Mathematical Linguistics and Automatic Translation*, Computation Laboratory, Harvard University, Cambridge, Mass., 1965

(2) M. CONWAY : "Design of a separate transition-diagram compiler" in *Communication ACM* 6, 7, pp. 396-408

(3) Notation développée par S. KLEENE à propos des langages réguliers (*Introduction of mathematics*, Van Nostrand, 1952)

On imagine alors le réseau de transitions, qui a l'allure d'un **graphe** formé d'états (a, b, c) reliés par des arcs orientés. Ces arcs représentent des transitions entre les états.

On distingue, parmi les états, l'état initial (→) et l'état final (⊙). Les arcs sont étiquetés par des mots ou des catégories lexicales. La suite des étiquettes lues de toutes les façons à partir d'un état initial jusqu'à un état final, correspond aux phrases définies par le réseau. Pour analyser la phrase, on pointe les mots, de la gauche vers la droite, en essayant de relier un point initial et un point final.



Analyser une phrase, c'est lire les mots, un par un, en partant de la gauche vers la droite et en cherchant à relier un état initial à un état final. Une méthode non déterministe d'analyse descendante utiliserait un réseau de transitions représentant la grammaire et un dictionnaire (indications sur les mots et les catégories lexicales). A partir d'un état initial, le programme effectue une boucle sur les mots de la phrase à analyser. Pour chaque état, il choisit un arc sortant. S'il n'y a pas d'arc sortant, l'analyse avorte.

Si le mot pointé correspond à l'étiquette, l'état courant est l'état terminal de l'arc parcouru. Sinon, il y a échec. Quand tous les mots sont lus, le dernier état courant doit être un état final pour que l'analyse se solde par un succès. Dans le cas contraire, il y a échec. Il faut préciser, au cours de l'analyse, la manière d'opérer certains choix ou la marche à suivre en cas d'échec, ce qui renvoie aux techniques d'analyse en parallèle et d'analyse avec retour en arrière décrites au paragraphe 1.5.3.2.2.

Pour une analyse en parallèle de *la porte ferme le couloir*<sup>1</sup> nous aurons :

Mot lu : *la*

A partir des noeuds a et d on peut atteindre les noeuds b et c par l'arc a-b (*la* peut être déterminant) et par l'arc d-c (*la* peut être pronom). Nous sommes en b et en c.

Mot lu : *porte*

A partir de b et de c on peut atteindre les noeuds c et e par l'arc b-c (*porte* n'est pas un adjectif mais peut être un nom) et par l'arc c-e (*porte* peut être un verbe). Nous sommes en c et en e.

(1) Exemple emprunté à G. SABAH : L'intelligence artificielle et le langage. Processus de compréhension, Hermès, Paris, 1989, p. 77

Mot lu : *ferme*

A partir de c on peut atteindre e par l'arc c-e (*ferme* est un verbe). L'autre possibilité, c'est à dire l'emprunt de l'arc e-f est impossible. Nous sommes en e.

Mot lu : *le*

On atteint f par l'arc e-f

Mot lu : *couloir*

On atteint g par l'arc fg. L'analyse se termine avec succès puisque g est un noeud terminal.

L'analyse avec retour arrière utilise une pile dans laquelle sont stockés les états du système selon le schéma [Position, {Arcs à essayer}]. Le signe + dans la colonne ACTION indique une correspondance entre le mot lu et la catégorie de l'arc. Le signe - indique qu'ils ne se correspondent pas.

PILE (avant)	MOT LU	ARC COURANT	ACTION	PILE (après)
[1, {d-c, a-b}]	<i>la</i>	d-c	+	[1, {a-b}]
[2, {c-e}] [1, {a-b}]	<i>porte</i>	c-e	+	[2, {φ}] [1, {a-b}]
[3, {e-f}] [2, {φ}] [1, {a-b}]	<i>ferme</i>	e-f	-	[3, {φ}] [2, {φ}] [1, {a-b}]
[3, {φ}] [2, {φ}] [1, {a-b}]	<i>ferme</i>	φ		[2, {j}] [1, {a-b}]
[2, {j}] [1, {a-b}]	<i>porte</i>	φ		[1, {a-b}]
[1, {a-b}]	<i>la</i>	a-b	+	[1, {φ}]
[2, {b-b, b-c}] [1, {φ}]	<i>porte</i>	b-b	-	[2, {b-c}] [1, {φ}]
[2, {b-c}] [1, {φ}]	<i>porte</i>	b-c	+	[2, {φ}] [1, {φ}]
[3, {c-e}] [2, {φ}] [1, {φ}]	<i>ferme</i>	c-e	+	[3, {φ}] [2, {φ}] [1, {φ}]
[4, {e-f}] [3, {φ}] [2, {φ}] [1, {φ}]	<i>le</i>	e-f	+	[4, {φ}] [3, {φ}] [2, {φ}] [1, {φ}]
[5, {f-f, f-g}] [4, {φ}] [3, {φ}] [2, {φ}] [1, {φ}]	<i>couloir</i>	f-f	-	[5, {f-g}] [4, {φ}] [3, {φ}] [2, {φ}] [1, {φ}]
[5, {f-g}] [4, {φ}] [3, {φ}] [2, {φ}] [1, {φ}]	<i>couloir</i>	f-g	+	[5, {φ}] [4, {φ}] [3, {φ}] [2, {φ}] [1, {φ}]

Le noeud *g* est un noeud terminal. Après avoir lu le dernier mot, l'analyse se termine par un succès.

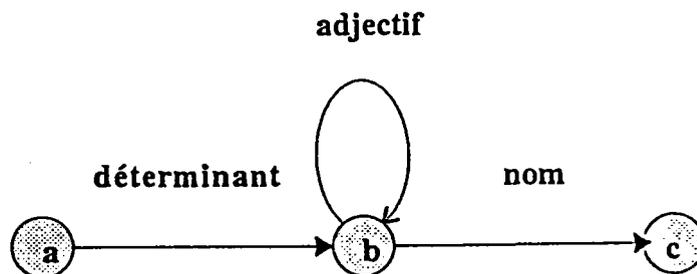
Il est clair, ici, qu'il ne s'agit que d'une procédure de reconnaissance. Cette procédure permet de vérifier si une phrase correspond à un réseau mais ne détermine en aucune façon sa structure. La puissance de ce formalisme est équivalente aux grammaires régulières (grammaires formelles de type 3). Pour atteindre une équivalence avec les grammaires de type 2 (grammaires indépendantes du contexte) et obtenir une analyse de la phrase, il faut étendre le modèle. On réunit alors des réseaux de transitions étiquetés et l'on parle d'un réseau de transitions récursif.

#### 1.5.4.2 Le R.T.N. ("Recursive Transition Network")

Lorsque les étiquettes d'un réseau sont des catégories syntaxiques (P, GN, GV, GP...), on obtient un ensemble de réseaux de transition étiquetés. Cet ensemble constitue un réseau de transitions récursif, dans le sens où, dans un réseau donné, un arc peut appeler ce réseau.

Issus des automates d'états finis, avec quelques ajouts (aspect récursif), ces réseaux sont équivalents, en puissance d'expression, aux grammaires indépendantes du contexte.

Considérons l'automate suivant :

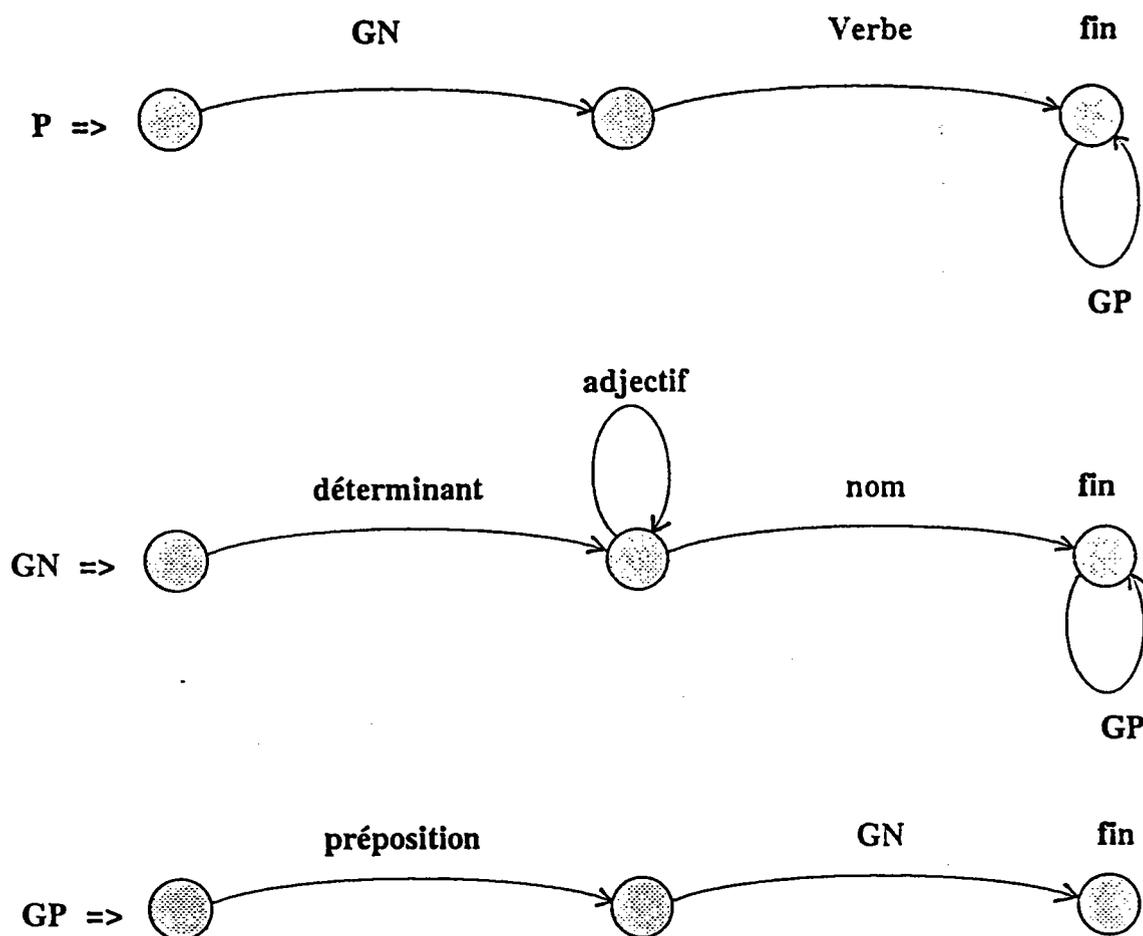


Une phrase est acceptée si, en partant de l'état initial avec le début de la phrase, on peut atteindre un état final à la fin de la phrase. L'automate représenté peut accepter des phrases du type "le joli papillon", "le beau livre", "le joli petit cadeau" mais pas "joli papillon".

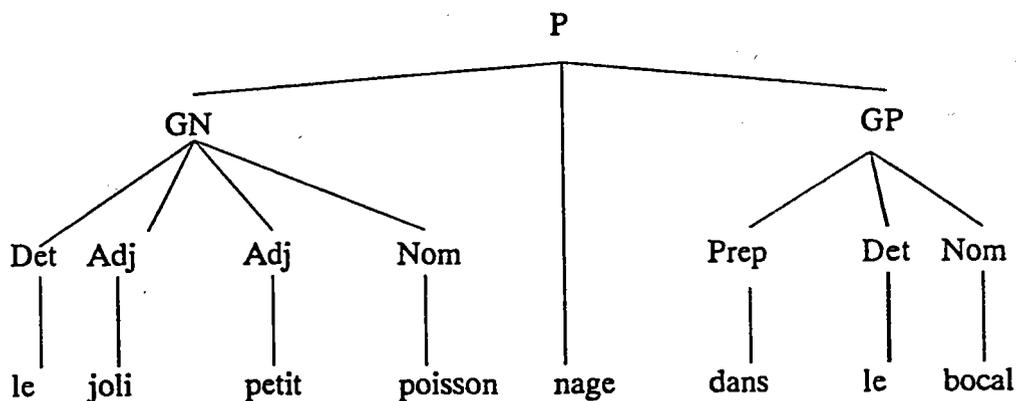
Il n'est pas possible, avec ce modèle, de représenter récursivement des phrases comme "le beau chien de la soeur de l'ami de notre voisin".

En spécifiant sur les arcs, non seulement des symboles terminaux (qui donnent un accès direct aux mots du langage) tels que déterminant, verbe, nom mais aussi des symboles non terminaux tels que GN (groupe nominal) ou GP (groupe prépositionnel) définis eux-mêmes par un autre automate pouvant faire référence à l'automate dont il fait partie, on arrive à la notion de RTN (Recursive Transition Network), dont nous présentons un exemple très simple.

Les catégories terminales sont ici préposition, adjectif, verbe, Déterminant et nom. Les catégories non terminales sont GN (groupe nominal) et GP (groupe prépositionnel). Un GN contient ici un GP et réciproquement. Il est donc essentiel d'employer un langage de programmation qui autorise la récursion.

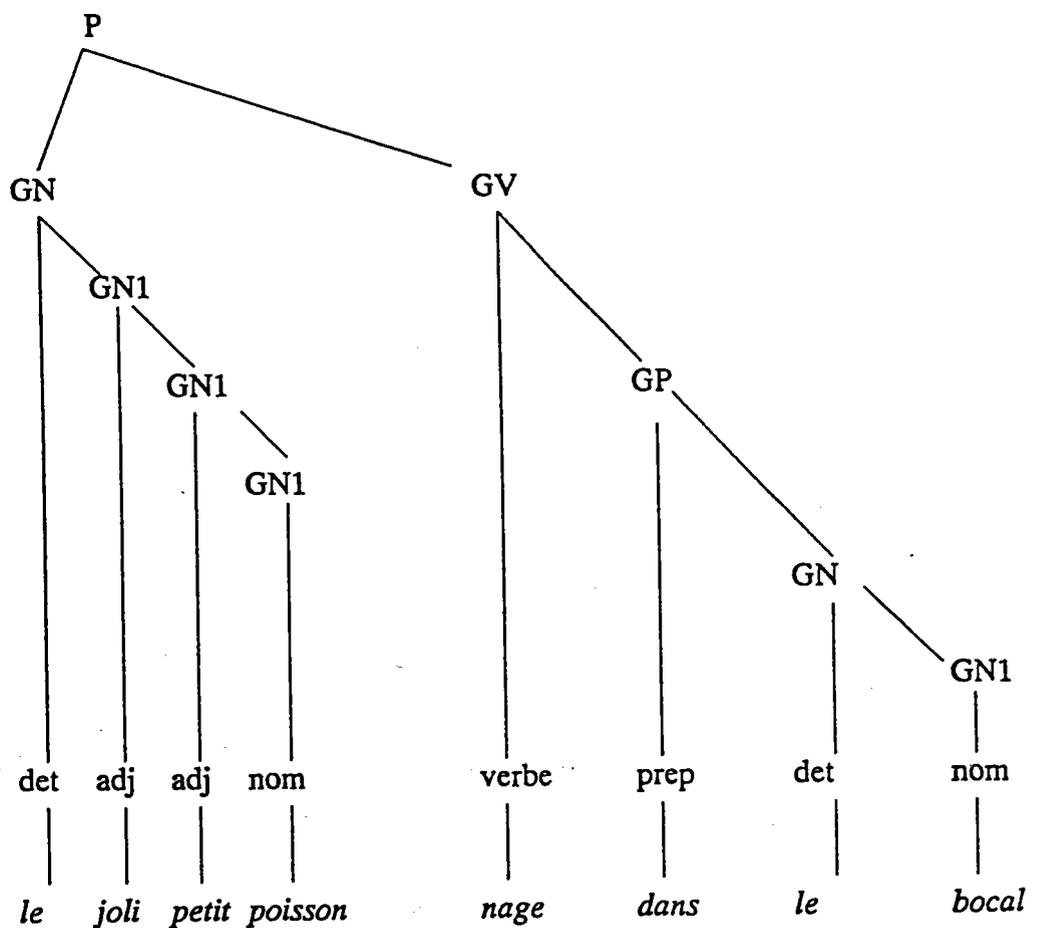


L'automate ainsi défini accepte la phrase *Le joli petit poisson de la soeur de Frédéric nage dans le grand bocal*". On remarque qu'une analyse produite par une grammaire hors contexte équivalente produirait un arbre moins plat. Cette grammaire aurait dû, dans le cas présent, introduire plusieurs symboles pour tenir compte de la répétition des adjectifs. Pour la phrase simplifiée *Le joli petit poisson nage dans le bocal*, nous obtiendrons, avec un RTN :



Avec une grammaire indépendante du contexte :

P->GN GV  
 GN1->GN1+GP  
 GN->Det GN1  
 GP->Prép+GN  
 GN->GN1  
 GV->Verbe  
 GN1->Nom  
 GV->Verbe+GP  
 GN1->Adj+GN1



Le mécanisme d'analyse devra être adapté au fonctionnement en parallèle ou au retour en arrière.

Ce type de réseau, fort pratique, ne permet pas de prendre en compte tous les phénomènes des langues naturelles et plus particulièrement certains aspects contextuels. Lorsque le traitement arrive à un noeud, il se poursuit au delà, indépendamment de la façon dont il est arrivé à ce noeud. Il est alors difficile de tester les accords et, plus généralement, de choisir le sens d'un mot par rapport à son contexte.

Si une grammaire à contexte libre peut gérer l'accord (en nombre par exemple), c'est au prix d'une lourdeur rédhibitoire. Les ATNS offrent un formalisme plus approprié pour lever ces limitations.

## 1.5.4.3 L'ATN ("Augmented Transition Network")

Un certain nombre d'ajouts qui justifient le nom de ces réseaux (réseau de transitions augmenté) va permettre d'intégrer des propriétés des grammaires transformationnelles.

- On peut associer aux arcs des tests à satisfaire pour pouvoir les emprunter. Il faut ajouter à ces conditions, l'appartenance à une classe syntaxique ou sémantique particulière.

ex : on impose que les catégories Nom et Article soient du même genre.

- On peut spécifier des actions à entreprendre dans le cas où l'arc est emprunté. Ces actions servent à construire des descriptions partielles des segments analysés.

ex : conserver une information qui sera réutilisée par la suite.

- On détermine un ensemble de registres qui contiendront ces descriptions ainsi que des indicateurs qui pourront être testés par des conditions liées à d'autres arcs. Ils sont gérés récursivement et assurent les liens entre les différents sous-réseaux du réseau global.

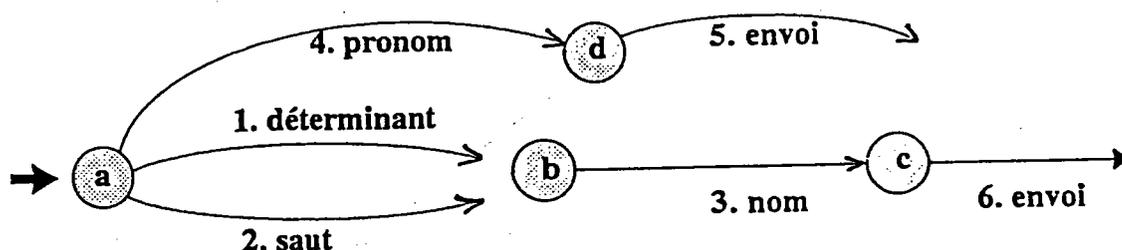
Nous ne ferons que citer le modèle de J. THORNE<sup>1</sup> qui n'utilise pas de registres, dont les conditions ne portent que sur des tests d'accord et dont les actions se limitent à l'insertion de marqueurs et à l'étiquetage de constituants.

En y ajoutant la possibilité d'une manipulation symbolique, D. BOBROW et B. FRASER<sup>2</sup> ont donné au modèle la puissance d'une machine de TURING générale. Dans ces systèmes, les structures des données auxquelles s'appliquent les conditions et les actions doivent être écrites dans le langage de programmation utilisé, ce qui limite l'efficacité du concepteur linguiste.

Les travaux de W. WOODS ont conduit à une réalisation plus complète grâce à une utilisation poussée de la notion de registre. Il introduit un nouveau type d'arc : l'arc Envoi, dont l'état initial est un état final du réseau et qui n'a pas d'état final. A chaque réseau est associé un ensemble de registres (un attribut et sa valeur) qui précisent les attributs et les rôles attachés à l'ensemble engendré.

Soit un réseau pour la représentation d'un groupe nominal simple dans lequel les formes pronom, déterminant, adjectif et nom sont autorisées.

GN :



(1) J. THORNE, P. BRATLEY, H. DEWAR : "The Syntactic Analysis of English by Machine" in *Machine Intelligence*, 3, Michie, American Elsevier, New York, 1968

(2) D. BOBROW, B. FRASER : "An Augmented State Transition Network Analysis Procedure", *Actes 1er IJCAI*, Washington DC, pp. 557-567, 1969

L'arc saut ne possède pas de contraintes et permet d'exécuter des actions sous certaines conditions, indépendamment du mot analysé.  
Le registre associé au réseau est NOMBRE. On indique alors son domaine de variation NOMBRE(singulier, pluriel).

On associe aux arcs les conditions et les actions suivantes :

-> assigne une valeur à un attribut

\* représente un registre spécial qui contient le mot lu ou, lorsqu'on fait appel à un sous-réseau, le résultat produit par ce sous-réseau.

Arc1 : Condition : 0  
Action : Nombre <--- Nombre de (\*)

Arc3 : Condition : Nombre = 0 ou Nombre = Nombre de (\*)  
Action : Nombre <--- Nombre de (\*)

Arc4 : Condition : 0  
Action : Nombre <--- Nombre de (\*)

Quand on emprunte l'arc 1, on stocke dans le registre NOMBRE du réseau le nombre correspondant au déterminant.

Pour emprunter l'arc 3, il faudra que le registre soit vide (c'est le cas lorsque l'on est passé par l'arc 2, autrement dit, lorsque le nom n'a pas de déterminant), ou, s'il a une valeur, qu'elle soit identique à celle du nom.

Pour analyser correctement les constituants discontinus (Combien cet homme possède-t-il de voitures ?), il faut pouvoir transmettre des informations d'un noeud à ses descendants. Ceci peut être réalisé par trois méthodes :

- *La communication explicite* : on recopie dans des registres de noeuds descendants les informations du noeud père qui pourraient être utilisées ou reportées sur un autre noeud descendant. Il faut alors vider le registre lorsqu'il est utilisé et ajouter à l'arc une action qui teste si des informations ont été transmises sans avoir été exploitées. Le registre correspondant du réseau appelant doit alors être vidé.

- *la recherche par contexte* : La technique, plus complexe, consiste à consulter les registres liés aux noeuds ascendants du noeud traité, en cas de nécessité seulement, pour rechercher un constituant de type voulu qui semble manquer dans la phrase analysée.

- *les registres globaux* : On ne remplit qu'un seul registre, au niveau général de la phrase. Un arc peut remplir le registre avec un constituant qu'il n'utilise pas, un autre arc pourra ensuite récupérer l'information, pour un constituant manquant.

Pour le fonctionnement des ATNs, la manipulation des registres s'accommode mieux d'une analyse descendante, de la gauche vers la droite.

#### 1.5.4.4 Le CATN ("Cascaded Augmented transition Network")

Créé à l'origine dans un but syntaxique, le modèle qu'a développé W. WOODS<sup>1</sup> introduit une partie sémantique qui n'intervient qu'après l'analyse syntaxique menée par un

---

(1) W. WOODS : "Progress in Natural Language Understanding : an Application to Lunar Geology", Actes AFIPS, 42, pp. 441-450, 1973

ATN, pour éliminer les analyses correctes mais dénuées de sens. Le système LUNAR est un système de question-réponse lié à une base de données de la NASA, rassemblant des informations sur les roches rapportées de la lune. L'analyse descendante, effectuée de gauche à droite, existe sous deux versions (la première fonctionne en parallèle, l'autre utilise les retours en arrière).

W. WOODS<sup>1</sup> a cherché par la suite à resserrer l'interaction entre la syntaxe et la sémantique tout en les distinguant nettement. Son modèle de réseau en cascade doit être assimilé à une cascade de deux réseaux.

- le premier réseau accède directement aux mots de la phrase et produit une analyse syntaxique.
- le second réseau accède indirectement aux mots, grâce à un ordre attaché à certains arcs du premier réseau, pour en produire une interprétation sémantique.

#### 1.5.4.5 Le réseau à noeuds procéduraux

Conçu par J.M. PIERREL<sup>2</sup> et utilisé dans le système MYRTILLE (Compréhension de la parole), le réseau à noeuds procéduraux doit son nom au fait que les procédures (conditions et actions) ne sont plus liées aux arcs mais aux noeuds du réseau. Elles servent à contrôler le parcours du réseau en déterminant l'ordre dans lequel il faudra emprunter les branches partant d'un noeud.

#### 1.5.4.6 Exemples

Avec LUNAR de W. WOODS, on peut citer :

PROGRAMMAR de T. WINOGRAD, responsable dans le système SHRDLU de l'analyse procédurale des phrases du dialogue étudié. Très proche d'un ATN, il s'inspire des grammaires systémiques (paragraphe 2.5.3.5) et intègre syntaxe, sémantique et raisonnement.

GUS (Genial Understander System)<sup>3</sup> doit gérer des dialogues homme-machine réalistes. L'exemple concerne un agent de voyage et un client qui désire réserver un billet d'avion.

L'analyse morphologique précède l'analyse syntaxique (RTN) et l'analyse sémantique au cours de laquelle une grammaire génère une structure de rôles convertie par le composant en structure de cas. Lors d'une étape de raisonnement, la question est interprétée et la réponse construite. La phase de génération traduit la réponse en anglais.

La particularité de GUS réside dans la modularité des différents programmes qui ne sont pas déclenchés séquentiellement mais dont le fonctionnement est contrôlé par un agenda qui contient les tâches à effectuer.

SOPHIE<sup>4</sup> enseigne les méthodes de dépannage de circuits électroniques. Le système fonctionne grâce à un ATN fondé sur des catégories lexicales sémantiques. Les aspects pragmatiques peuvent ainsi être pris en compte dès le début de l'analyse.

---

(1) W. WOODS : "Cascaded ATN", *AJCL* 6, pp. 1-12, 1980

(2) J.M. PIERREL : "Etude et mise en oeuvre de contraintes linguistiques en compréhension automatique du discours continu", *Thèse de doctorat ès sciences*, Université de Nancy I, 1981

(3) D. BOBROW, J. KAPLAN, D. NORMAN, H. THOMPSON, T. WINOGRAD : "GUS : a Frame-driven Dialog System", *Artificial Intelligence*, 8, pp. 155-173, 1977

(4) J. BROWN, R. BURTON, A.G. BELL : "SOPHIE : a Step toward a Reactive Environment", *International Journal of Man Machine Studies*, 7, pp. 675- 696, 1975

SPEECHLIS<sup>1</sup> est issu de LUNAR. Il traite les entrées vocales en utilisant les ATNs. Sa grammaire est importante (448 états, 881 arcs et 280 actions) mais le système est étroitement lié au domaine.

W. WOODS<sup>2</sup> souligne l'intérêt des ATNs, dans la perspective d'une généralisation, si l'on considère ces réseaux comme un automate abstrait d'analyse unifiant divers langages de spécification syntaxique. On peut rappeler que les grammaires lexicales fonctionnelles génèrent des langages compatibles avec les ATNs et noter que différents auteurs ont poussé très loin la comparaison avec l'analyseur de M.P. MARCUS<sup>3,4</sup> ou les DCG<sup>5</sup> (Definite Clause Grammar, paragraphe 2.7.7.2). R. BURTON et W. WOODS<sup>6</sup> ont envisagé de compiler la grammaire correspondant à un ATN en un code commun à la grammaire et au mécanisme d'analyse. La vitesse de l'analyseur ainsi conçu est multipliée par 10.

## 1.5.5 Les analyseurs déterministes

### 1.5.5.1 Présentation

M.P. MARCUS<sup>7</sup> a tenté de résoudre le problème de l'indéterminisme qui n'a pas pour origine, comme on a tendance à l'affirmer, l'ambiguïté inhérente à la langue, mais plutôt notre façon de conduire une analyse. Nous respectons en effet des habitudes sans grand fondement.

- analyse immédiate et complète des mots lus de gauche à droite
- niveaux d'analyse distincts
- choix a priori des points de reprise

Les analyseurs déterministes apportent des innovations destinées à minimiser les ambiguïtés artificielles.

- Il est possible de stopper une analyse lorsqu'une ambiguïté est rencontrée. Comme avec les ATNs, on conserve les informations obtenues afin de lever l'ambiguïté au moment opportun, de façon plus appropriée.
- Les différents aspects d'un texte peuvent interférer.
- Pour lever des ambiguïtés, ces analyseurs examinent les mots du contexte immédiat.

- 
- (1) W. WOODS : *SPEECHLIS* : "An Experimental Prototype for Speech Understanding Research", *IEEE Transactions on Acoustics, Speech and Signal Processing*, 23, 1, pp. 2-10, 1975
  - (2) W. WOODS : "Generalisations of ATN Grammars", in *Research in Natural Language Understanding*, Woods et Brachman, BBN Report n° 3963, Bolt, Beranek et Newman, Cambridge, Mass., 1978
  - (3) M.P. MARCUS : *A Theory of Syntactic Recognition for Natural Language*, MIT Press, Cambridge, Mass., 1980
  - (4) W. SWARTOUT : "A Comparison of PARSIFAL with Augmented Transition Networks", *AI memo 462*, MIT, Artificial Intelligence Laboratory, Cambridge, Mass., 1978
  - (5) F. PEREIRA, D. WARREN : "Definite Clause Grammar for Language Analysis - a Survey of the Formalism and a Comparison with Augmented Transition Networks", *Artificial Intelligence*, 13, 3, pp. 231-278, 1980
  - (6) R. BURTON, W. WOODS : "A Compiling System for Augmented Transition Networks", *Actes COLING76*, Ottawa, pp. 65-83, 1976
  - (7) M.P. MARCUS : *A Theory of Syntactic Recognition for Natural Language*, MIT Press, Cambridge, Mass., 1980

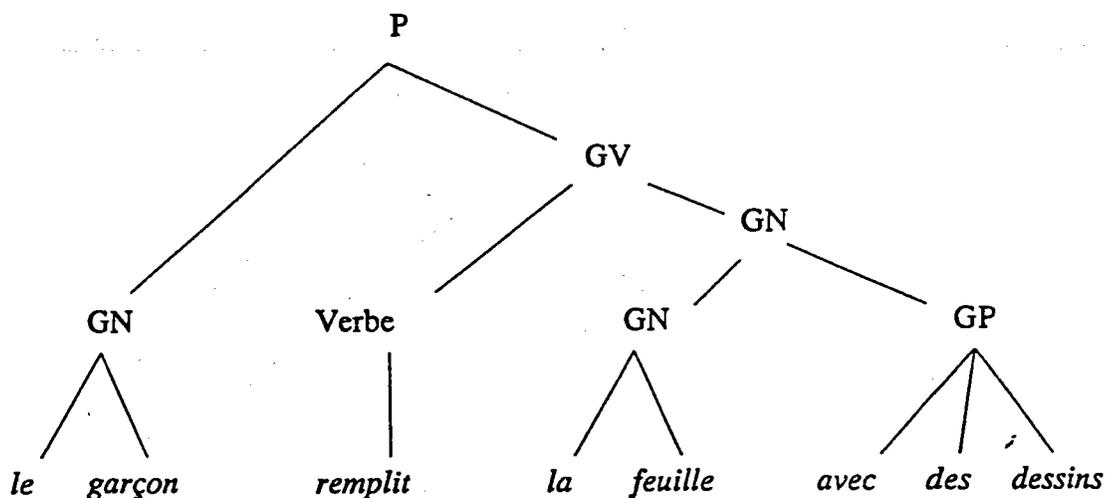
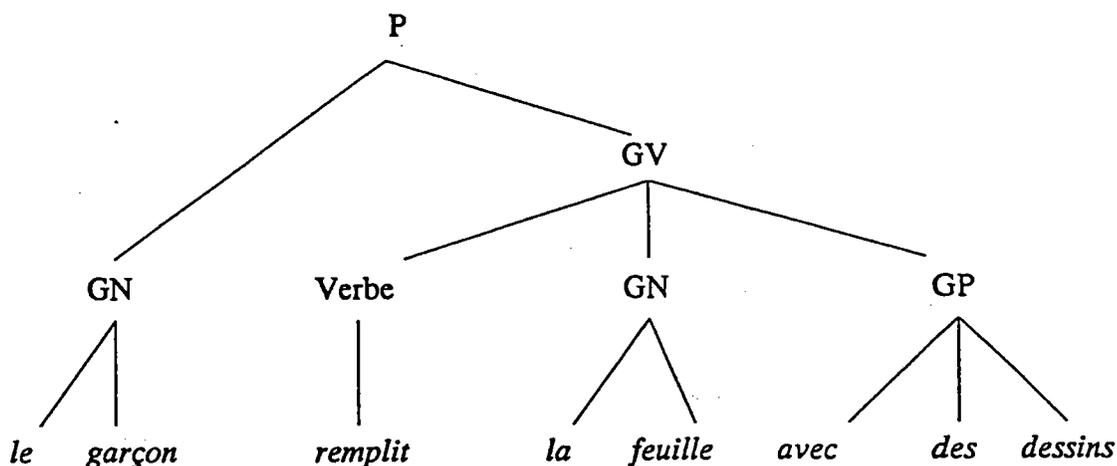
L'analyseur déterministe ignore le parallélisme et le retour en arrière. Il est capable de choisir la bonne voie lorsqu'il y a plusieurs possibilités. Notre façon de comprendre les énoncés s'appuie sur des principes qu'il est intéressant de connaître :

1. - En cas d'ambiguïté, le choix s'oriente vers la solution qui crée le moins de noeuds dans l'arbre d'analyse.

ex : *Le garçon remplit la feuille avec des dessins*

grammaire correspondante (indépendante du contexte) :

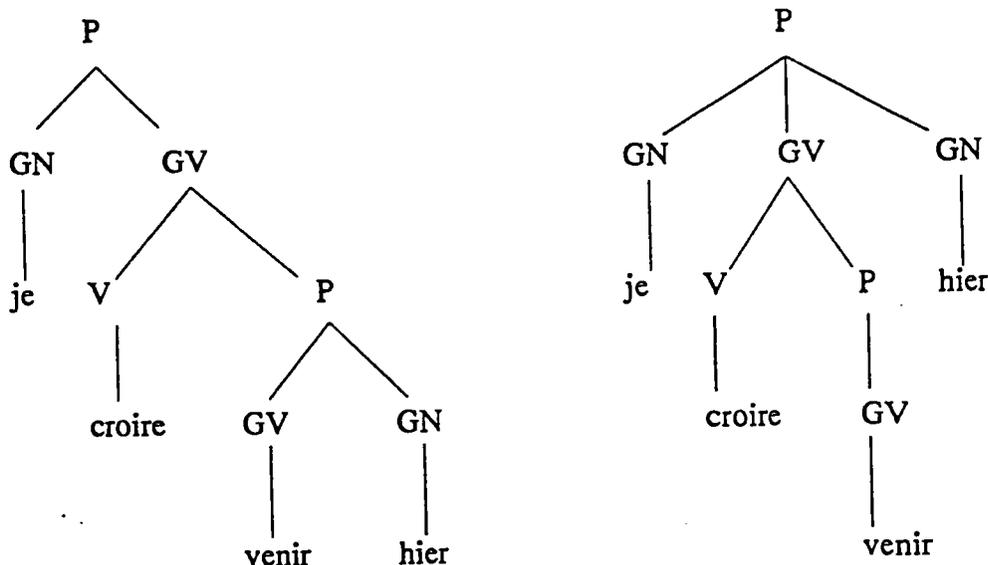
- P -> GN GV
- GN -> ART + NOM
- GN -> GN + GP
- GP -> PREP + GN
- GV -> VERBE + GN
- GV -> VERBE + GN + GP



Le choix s'orientera ici vers le premier arbre, (le groupe prépositionnel est attaché au verbe). Le second arbre (le groupe prépositionnel est attaché au nom) possède en effet un noeud de plus.

2. - Il est préférable d'interpréter un constituant en l'associant au constituant qui est en cours de développement, plutôt qu'à un constituant de niveau supérieur.

ex : *Je croyais que tu viendrais hier*



L'automate privilégie la première solution où *hier* représente le moment où la personne vient plutôt que le moment où on le croit.

3. - Les deux principes que nous venons d'évoquer peuvent être contradictoires. On introduit alors la notion de préférence lexicale.

On constate que ces questions ont été abordées, d'une certaine manière, dans le paragraphe précédent, avec les stratégies mises en oeuvre sur les ATNs. La mise en place d'un ordre de passage sur les arcs correspond aux deux premiers principes que nous venons de citer.

Quant au mécanisme qui permet d'examiner les éléments du contexte immédiat dans le but de pouvoir choisir la bonne solution, son champ d'action a été limité à 3 mots vers la droite. Cette limite a été déterminée expérimentalement, dans le sens du meilleur rapport efficacité/complexité. Il faut également préciser que ce "regard en avant" concerne autant les mots et certaines de leurs caractéristiques, que les structures.

### 1.5.5.2 Exemples

#### 1.5.5.2.1 PARSIFAL

M. MARCUS a conçu PARSIFAL comme un analyseur syntactique déterministe, n'utilisant ni le parallélisme, ni le retour en arrière, (les structures construites sont définitives et doivent être utilisées dans la représentation finale, les structures temporaires sont impossibles). La couverture linguistique de l'analyseur n'est pas très vaste mais son fonctionnement a validé les principes de l'analyse déterministe et son efficacité.

### 1.5.5.2.2 ANDI

Cette méthode a été étendue. L'analyseur ANDI (Analyseur Déterministe Intégré)<sup>1,2</sup> doit comprendre des phrases complexes, lever des ambiguïtés lexicales, sémantiques et, dans une moindre mesure, de structures.

Voisin de l'analyseur de M. MARCUS, il s'en écarte cependant sur un point, en intégrant la syntaxe et la sémantique. Il génère en effet "simultanément la structure syntaxique de la phrase et sa représentation sémantique"<sup>3</sup>.

Si les analyseurs déterministes apportent de réels avantages :

- leur architecture informatique permet d'éviter les ambiguïtés artificielles dues aux mécanismes d'analyse.
- ils sont plus efficaces que les analyseurs non déterministes, qui perdent beaucoup de temps à tester des mauvaises hypothèses.
- ils conservent une trace de leur fonctionnement, ce qui permet de retrouver les causes d'un échec de l'analyse. Il est alors envisageable d'effectuer un apprentissage automatique des règles de grammaire.

ils présentent de sérieux inconvénients :

- il faut prévoir toutes les ambiguïtés au niveau des règles pour garantir le déterminisme<sup>4</sup>.
- les règles sont spécifiques selon le type d'analyse, montante ou descendante (paragraphe 1.5.3.2.2). Utilisant les deux, l'analyseur déterministe nécessite un plus grand nombre de règles.
- les règles ne doivent pas interférer. Leur mise au point est donc plus délicate, la modularité en souffre.
- le principe du déterminisme est un obstacle au traitement des phrases "labyrinthes" lorsque l'analyse du contexte immédiat est limitée. Il est également difficile, pour un tel mécanisme, de donner plusieurs interprétations d'une phrase ambiguë.

### 1.5.6 Les analyseurs à mots clés

La particularité de cet analyseur et des suivants est d'associer à des unités des éléments de connaissance utiles pour l'analyse de la phrase. Le traitement est plus souple puisque le contrôle de l'analyse est décentralisé.

Fonctionnant dans des domaines limités et ne traitant aucun problème complexe concernant le langage, ils sont très efficaces pour gérer des dialogues dans des domaines clos.

L'analyseur à mots clés ne comprend que certains mots de la langue qui déclenchent alors des actions spécifiques. Il s'appuie sur un lexique qui rassemble les mots significatifs pour le domaine abordé. Il repère ces mots ou leurs variantes et effectue des actions appropriées.

---

(1) M. RADY : "L'ambiguïté du langage est-elle la source du non-déterminisme des procédures de traitement ?" *Thèse de Doctorat ès Sciences*, Université Pierre et Marie Curie, Paris, 1983

(2) G. SABAH, M. RADY : "A Deterministic Syntactic-semantic Parser to French", *Actes 8ème IJCAI*, Karlsruhe, pp. 707-710, 1983

(3) La structure syntaxique engendrée est un arbre syntagmatique aux noeuds duquel sont attachés des traits. La structure sémantique repose sur une grammaire de cas. On y ajoute des informations pour tenir compte des aspects énonciatifs.

(4) E. CHARNIAK : *Parsing, How To in Automatic Natural Language Parsing*, Sparck-Jones & Wilks, University of Essex, 1982

BASEBALL<sup>1</sup> est un système associé à une base de données concernant les matches du championnat de baseball aux Etats Unis. Il répond aux questions simples, à partir d'un vocabulaire d'une soixantaine de mots classés selon quatorze catégories correspondant à des concepts tels que adversaire, match, équipe...

Le mécanisme est simple, l'analyseur reconnaît des mots et construit la question sous une forme canonique, coïncidant avec la forme de stockage dans la base de données.

exemple emprunté à G. SABAH<sup>2</sup> :

A la question : *combien de matches les Yankees ont-ils joué en mai ?*  
correspondra : ((Equipe=Yankee) (Mois=mai) (Match(Nombre\_de)=?))

STUDENT<sup>3</sup> fonctionne comme BASEBALL, pour résoudre des exercices d'algèbre élémentaire. Il traite de plus certaines structures de phrases prédéfinies.

ELIZA<sup>4</sup> simule un psychiatre dans le contexte d'un entretien non directif avec des patients. Il dispose d'un ensemble de mots clés avec des ensembles de structures dans lesquelles ils interviennent. Lorsqu'il reconnaît une forme, il lui associe une forme de phrase à compléter, s'il ne trouve aucun mot clé, il choisit une réponse au hasard.

### 1.5.7 Les analyseurs conceptuels

Les unités de base sont ici des concepts primitifs, qui doivent pouvoir représenter tous les concepts de la langue, par combinaisons. Appliquant les principes de la dépendance conceptuelle, l'analyseur reconnaît d'abord l'occurrence d'un graphe conceptuel (primitives PTRANS, ATRANS, SPEAK...) et valide ensuite les hypothèses impliquées.

ELI (English Language Interpreter) est une variante améliorée de l'analyseur appartenant au programme MARGIE<sup>5</sup> et fait partie du programme SAM<sup>6</sup> qui simule une compréhension à l'aide de scénarios.

Le principe de base est la prévision qui découle des indications apportées par les mots, les concepts correspondants et les structures conceptuelles déjà construites. Les prévisions concernent des mots, des concepts ou des structures conceptuelles que l'on s'attend à rencontrer dans la phrase.

A chaque mot est lié un ensemble de requêtes : {(Conditions, Actions)}. Quand le mot est traité, les requêtes qui lui sont associées sont activées. Nous limiterons notre description de l'analyseur, qui malgré de nombreuses modifications<sup>7</sup> a finalement été abandonné.

D'autres analyseurs ont été conçus selon les mêmes principes:

(1) B. GREEN, A.K. WOLF, N. CHOMSKY, K. LAUGHERY : "BASEBALL : An Automatic Question Answer" in *Computers and Thought*, Feigenbaum & Feldman, Mc Graw-Hill, New York, pp. 207-216, 1960

(2) G. SABAH : *L'intelligence artificielle et le langage*. Vol. n 2, Processus de compréhension, Hermes, Paris, pp. 104-105, 1989

(3) D. BOBROW : "Natural Language Input for a Computer Problem Solving System" in *Semantic Information Processing*, Minsky, MIT Press, Cambridge Ma, 1968

(4) J. WEIZENBAUM : "ELIZA : A Computer Program for the Study of Natural Language Communication between Man and Machine", *CACM*, 9, pp. 26-45, 1966

(5) R. SCHANK, N. GOLDMAN, C. RIEGER, C. RIESBECK : "MARGIE : Memory, Analysis, Response Generation and Inferences in English", *Actes 3ème IJCAI*, 1972

(6) R. SCHANK : "SAM : A Story Understander", Report n°43, Yale University Press, 1977

(7) A.V. GERSHMAN : "Conceptual Analysis of Noun Groups in English", *Actes 5ème IJCAI*, Cambridge Mass., pp. 132-148, 1977

**FRUMP** (Fast Reading Understanding and Memory Program)<sup>1</sup> analyse des textes, en utilisant une méthode partielle descendante et en complétant des occurrences de scénarios qui ont été reconnus.

**IPP** (The Integrated Partial Parser)<sup>2</sup> analyse des textes selon une méthode partielle montante et descendante en complétant les MOPs pertinents qu'il a reconnus.

**McMAP**<sup>3</sup> analyse des textes, en montant et construit les dépendances conceptuelles de la phrase analysée. Il reconnaît les buts des personnages d'une histoire et les plans qu'ils appliquent pour les atteindre.

### 1.5.8 Les grammaires logiques

Nous les avons assimilées à des outils au même titre que les ATNs, dans le sens où elles laissent une marge de manœuvre importante à leurs utilisateurs. Dans le contexte de leur utilisation pour l'analyse syntaxique et sémantique, elles servent dans le dernier cas autant à l'analyse qu'à la représentation des connaissances.

Leurs possibilités sont très importantes<sup>4</sup>. Elles mènent à des représentations internes autres que les arbres syntagmatiques. Il est possible, d'autre part, lors de leur écriture, de définir les constructions de ces représentations. Il faut rappeler que leur formalisme obéit à une volonté de mise sous forme informatique en utilisant le langage PROLOG pour la description des règles.

Langage de PROGRAMMATION en LOGIQUE, PROLOG a été conçu en 1972 par A. COLME-RAUER. Les premières étapes de son développement ont été assurées principalement à Edimbourg, avant que les japonais le choisissent pour leur ambitieux projet d'ordinateurs de "5ème Génération".

Un programme PROLOG est une suite de "clauses" qui peuvent être considérées comme une généralisation des règles de réécriture et comme une formule représentant le sens :

$GN(X,Y) \rightarrow DET(X,Y) NOM(Y,Z)$  est équivalent à  $GN \rightarrow DET, NOM$

$ONCLE(X,Z) \rightarrow FRERE(X,Y) PERE(Y,Z)$  X est l'oncle de Z lorsque X est frère de Y, et Y père de Z

Utilisé pour programmer les analyseurs syntaxique et sémantique, ils servent également à représenter les connaissances nécessaires à la compréhension.

Les règles de réécriture ne combinent plus seulement de simples identificateurs, mais des prédicats et des variables, ce qui implique des processus d'unification pour que les variables aient une valeur identique dans toutes les règles.

Des contraintes sur la nature des variables vont permettre de soumettre l'application des règles à certaines conditions.

---

(1) G. DEJONG : *An Overview of the FRUMP System in Strategies for Natural Language Processing*, Lehnert et Ringle, Erlbaum, Hillsdale, N.J., pp. 149-176, 1982

(2) R. SCHANK, M. LEBOWITZ, L. BIRNBAUM : *An Integrated Understander*, *American Journal of Computational Linguistics*, 6, 1, pp. 13-30, 1980

(3) M. DYER : *In-Depth Understanding*, MIT Press, Cambridge MA., 1983

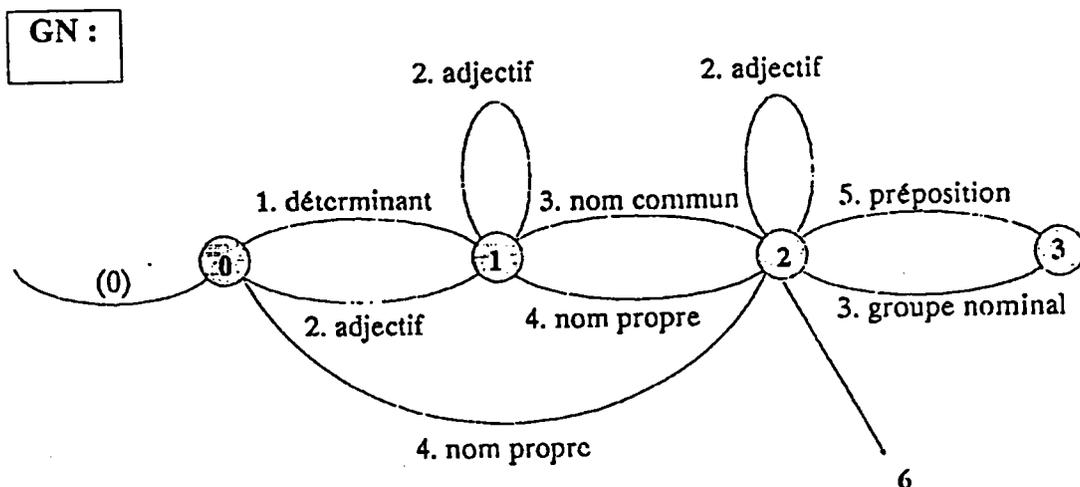
(4) Actes du Colloque "Compréhension du langage naturel et programmation en logique", Rennes, sept. 1984, North Holland, 1985

### 1.5.8.1 Les grammaires de métamorphose. Les DCG (Definite Clause Grammars)

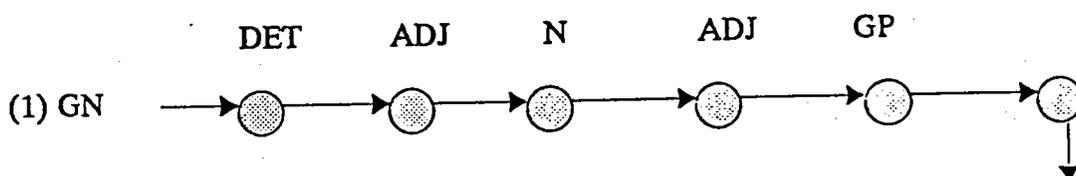
A l'origine de ces grammaires logiques, on trouve les grammaires de métamorphose<sup>1</sup>, exécutables par un interpréteur PROLOG. Lorsque la partie gauche d'une règle n'est pas un symbole logique non terminal suivi facultativement de symboles logiques terminaux mais se résume à un seul élément non terminal, nous avons les grammaires à clauses définies (DCG, "Definite Clause Grammars")<sup>2,3</sup>.

L'ajout de conditions et d'actions aux règles de ces grammaires les rendent aussi puissantes dans leur formalisme que les ATNs, tout ATN pouvant être transformé en une DCG :

Empruntons l'exemple d'un ATN pour analyser les groupes nominaux élémentaires à D. COULON et D. KAYSER<sup>4</sup> :



Une décomposition en graphes simples sans circuit donne :

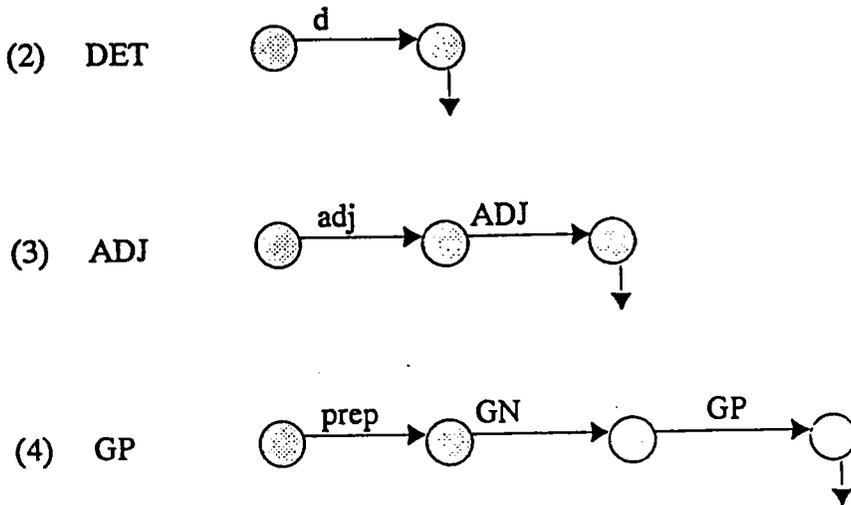


(1) A. COLMERAUER : "Les grammaires de métamorphose" in L. BOLC (ed.), *Natural Language Communication with Computers*, Lecture Notes in Computer Science, n°63, Springer Verlag, pp. 133-189, 1978

(2) F. PEREIRA, D. WARREN : "Definite Clause Grammar for Language Analysis - A Survey of the Formalism and a Comparaison with Augmented Transition Networks", *Artificial Intelligence*, 13, 3, pp. 231-278, 1980

(3) D. WARREN, F. PEREIRA : "An Efficient Easily Adaptable System for Interpreting Natural Language Queries", *American Journal of Computational Linguistics*, 8, (3-4), July-dec., 1982

(4) D. COULON, D. KAYSER : "Informatique et langage naturel" in *Technique et science informatiques*, Vol. 5, n 2, p. 117, 1986



Pour chaque chemin, on écrit une règle PROLOG :  
 En partie gauche : le prédicat avec, pour argument, la structure correspondante.  
 En partie droite : les prédicats à tester.

$PGN(GN < DET, ADJ1, N, ADJ2, GP ) \rightarrow PDET(DET), PADJ(ADJ1), PN(N),$   
 $PADJ(ADJ2), PGP(GP)$

Les prédicats à tester (par ex.  $PDET(DET)$  ) engendrent de nouvelles structures par unification avec les parties gauches des autres règles.

$PDET(DET < d >) \rightarrow [d], \{d\u00e9terminant(d)\}$

Ils servent à tester l'existence de mots terminaux, écrits entre crochets et déclenchent des vérifications, comme la nature du mot, par consultation du lexique, ce qui est indiqué par les accolades. Pour obtenir une traduction complète de l'ATN, il faut retranscrire les actions, ce qui s'obtient en employant des conditions supplémentaires ou des arguments supplémentaires qui permettent alors de stocker des résultats intermédiaires comme les registres des ATNs.

### 1.5.8.2 Les extensions

Les grammaires d'extrapositions<sup>1,2</sup> introduisent la notion de discontinuité en permettant de noter dans une règle une partie de phrase à laquelle on ne s'intéresse pas. Nous pouvons mentionner deux autres développements de ces modèles : les grammaires à structures modifiées<sup>3</sup> qui expriment plus clairement les représentations que l'on souhaite engendrer et les grammaires d'arbres et de puzzle<sup>4</sup> qui élaborent des arbres par unification entre les feuilles et les racines.

(1) V. DAHL, H. ABRAMSON : "On Gapping Grammars", *Actes de la 2ème Logic Programming Conference*, Uppsala University, 1984  
 (2) F. PEREIRA : "Extrapolation Grammars", *Computational Linguistics*, Vol. 7, 1981  
 (3) V. DAHL, Mc CORD : "Traiting Coordination in Logic grammars", *American Journal of Computational Linguistics*, Vol. 9, n2, pp. 69-91, 1983  
 (4) P. SABATIER : "Puzzle Grammars", *Actes du premier Colloque International "Natural Language and Logic Programming"*, Rennes, 1984

### 1.5.9 Les outils intégrés

Le traitement des ambiguïtés met en valeur la nécessité d'une plus grande intégration des différentes phases de l'analyse. Les informations qui permettent d'opérer les bons choix sont diverses. Les modules de traitement s'enchaînent sans qu'il soit possible d'utiliser au maximum le fait que les résultats des uns peut aider au fonctionnement des autres et inversement.

L'utilisation du langage PROLOG pour formaliser toutes ces phases (paragraphe 1.5.8) est une première tentative.

La technique du tableau noir (paragraphe 1.5.3.2.1) préfigure une intégration plus poussée. Chaque module utilise les informations qui lui sont utiles et stocke celles qu'il trouve<sup>1</sup>.

Le principe de l'interaction des différents programmes qui fonctionnent en parallèle (parallélisme réel ou simulé) devrait être développé avec l'apparition des langages objets<sup>2</sup>. L'interaction des niveaux est facilitée par un mécanisme de propagation de messages associé à une décentralisation du contrôle<sup>3</sup>.

Il est probable que la réalisation d'une intégration optimale sera liée aux futurs ordinateurs parallèles<sup>4</sup>.

### 1.5.10 Le traitement des erreurs

Le traitement automatique des langues se heurte parfois :

- à des erreurs d'orthographe (niveau lexical)
- à des structures incorrectes (niveau syntaxique)
- à des représentations contradictoires, incomplètes ou à des erreurs conceptuelles (niveau sémantique)
- à des ambiguïtés ou des imprécisions (niveau pragmatique)

Certains auteurs ont proposé une typologie des "non-attendus", en distinguant les erreurs de compétence (manque de connaissances) et les erreurs de performance (manque d'attention, problème matériel, ...)<sup>5</sup>.

Nous les passerons rapidement en revue en distinguant plutôt les niveaux (lexical, syntaxique, sémantique et pragmatique), tout en reconnaissant que les erreurs ne sont pas toujours repérées au niveau où elles interviennent, et peuvent être corrigées grâce à des informations d'un autre niveau. Nous citerons à chaque fois les méthodes de correction envisageables, sachant que certains systèmes préfèrent ignorer les erreurs et continuer le traitement, d'autres diffèrent l'interprétation, recommencent l'analyse, étudient le contexte immédiat ou demandent de l'aide en fonctionnement interactif.

---

(1) L. ERMAN, F. HAYES-ROTH, V. LESSER, R. REDDY : "The Hearsay II Speech Understanding System", *Computing Surveys*, 12(2), 1980

(2) J. FERBER : "Les langages objets : une affaire de messages", *Revue Microsystèmes*, avril 1985, pp. 152-159, 1985

(3) K. De SMEDT : "Using Object-oriented Knowledge Representation Techniques in Morphology and Syntax Programming", *Actes ECAI*, pp. 181- 184, sept. 1984

(4) D. WALTZ, J. POLLACK : "Massively Parallel Parsing : A Strongly Interactive Model of Natural Language Interpretation", *Cognitive Science*, 9(1), pp. 51-74, 1985

(5) J. VERONIS : "Le traitement de l'erreur dans le dialogue homme- machine en langage naturel, application à l'enseignement assisté de la géométrie", *Thèse de l'Université d'Aix-Marseille*, 1988

### 1.5.10.1 Plan lexical

#### 1.5.10.1.1 Types d'erreur

- Les erreurs concernent l'écriture du mot et s'appliquent à une lettre ou à un groupe de lettres. Ce sont les fautes classiques de substitution (*povr* au lieu de *pour*), d'insertion (*dues* au lieu de *dues*), de suppression (*exempe* au lieu de *exemple*), d'erreur de signe diacritique (*facon* au lieu de *façon*), d'interversions (*immédait* au lieu de *immédiat*), de redoublements (*constitutition* au lieu de *constitution*).
- A ces fautes de frappe, il faut ajouter les fautes grammaticales de l'utilisateur qui ignore l'orthographe du mot et propose une écriture phonétique (*laupain* au lieu de *lopin*) et les fautes de flexion (genre, nombre, conjugaison).
- Citons enfin les mots que le système ne peut analyser parce qu'ils ne sont pas dans le lexique dont il dispose (le domaine est ouvert et le mot n'a pas été prévu, le domaine est fermé, le mot est absent ou il s'agit d'un néologisme, le problème des noms propres étant, dans ce cadre, assez complexe).

#### 1.5.10.1.2 Techniques de correction

Deux études sur la répartition statistique des fautes ont été réalisées, la première par F. DAMERAU<sup>1</sup>, la seconde par J.J. POLLOCK et A. ZAMORA<sup>2</sup>. L'une indique que 80% des mots erronés sont des erreurs uniques dans le mot. Pour l'autre, 90% des mots erronés contiennent une seule erreur. On peut cependant reprocher à ces travaux de ne pas être très représentatifs, et aux résultats, de ne pas être précis. En effet, si l'on replace ces résultats dans leur contexte, on constate que les buts étaient différents :

- indexation et recherche de documents par mots clés, pour F. DAMERAU, avec un matériel de stockage sur ruban papier et les risques que cela implique.
- analyse de textes scientifiques (25.000.000 mots) malheureusement saisis par des professionnels.

Ces données permettront cependant d'élaborer des méthodes de correction.

- méthodes fondées sur la distribution des graphèmes : Lorsqu'un mot n'est pas reconnu, elles proposent une correction en utilisant les probabilités d'apparition de toutes les paires et triades de graphèmes de la langue traitée<sup>3,4</sup>.

Elles n'ont pas besoin de dictionnaires, concernent tous les domaines mais ne sont efficaces que pour les erreurs de substitution. On les utilise surtout pour la relecture des documents saisis par lecture optique.

- méthodes de recherche dans les dictionnaires : Elles sont mieux adaptées aux fautes d'insertion, de suppression et d'inversion.

Les dictionnaires envisagés doivent répondre à deux exigences :

- retrouver un mot correctement orthographié, le plus rapidement possible et avec une consultation minimale.

---

(1) F. DAMERAU : "A Technique for Computer Detection and Correction of Spelling Errors", *Journal des ACM*, 7, 3, pp. 171-176, 1964

(2) J.J. POLLOCK, A. ZAMORA : "Automatic Spelling Correction in Scientific and Scholarly Texts", *Communications ACM*, 27, 4, pp. 358-368, 1984

(3) E.M. RISEMAN, R.W. EHRICH : "Contextual Word Recognition Using Binary Digrams", *I.E.E.E. Transactions on Computers*, 20, 4, pp. 397-403, 1971

(4) E.M. RISEMAN, A.R. HANSON : "A Contextual Post-processing for Error Correction Using Binary n-grams", *I.E.E.E. Transactions on Computers*, 23, 5, pp. 480-493, 1974

- repérer l'absence d'un mot mal orthographié et proposer des solutions de substitutions. Le meilleur moyen de concilier ces nécessités contradictoires est de concevoir une structure appropriée du lexique<sup>1</sup>. On détermine une structure adaptée à une recherche exacte rapide (paragraphe 1.5.2.1) et on lui associe un système de génération automatique des variantes possibles pour chaque mot que l'on recherche séquentiellement.

Pour la substitution, l'insertion, la suppression et l'interversion, en considérant un mot de longueur  $l$  et un alphabet de  $n$  lettres, on engendre  $l-1$  mots voisins par inversion de deux lettres consécutives,  $(n-1)*l$  par substitution,  $n*(l+1)$  par inversion d'une lettre et  $l$  par suppression d'une lettre. Pour traiter 90% des fautes (une erreur par mot), on aboutit à  $l(2n+1)+n-1$  candidats. Avec un mot de 7 lettres et un alphabet de 32 lettres on obtiendrait 486 possibilités, ce qui représente finalement une certaine lourdeur en temps d'accès mémoire. Le programme SPELL<sup>2</sup> utilise cette technique qu'il est possible d'améliorer en testant les invraisemblances et en limitant ainsi les recherches<sup>3</sup>.

- Il existe une autre méthode : on partitionne le lexique et on ne compare ainsi le mot erroné qu'avec un sous-ensemble du lexique, en espérant trouver rapidement sa classe d'équivalence en se basant sur la longueur du mot<sup>4</sup> ou les caractères à l'initiale du mot<sup>5</sup>. Les classes d'équivalence sous-tendent la notion de "clé de similarité" qui correspond à un code calculé à partir des lettres du mot. Des chaînes voisines pourront avoir une clé identique et définir ainsi une classe d'équivalence. La clé permet alors d'appeler tous les candidats au remplacement. Notons que cette clé, pour être efficace, doit tenir compte de plusieurs informations. Construite uniquement sur la longueur du mot, elle regrouperait des chaînes trop différentes.

Le système de J.J. POLLOCK et A. ZAMORA utilise deux clés. (*la clé squelette et la clé d'omission*). F. DEBILI<sup>6</sup> propose une clé construite en reprenant dans l'ordre alphabétique toutes les lettres constituant le mot. L'avantage des clés réside dans le fait que leur longueur facilite leur gestion et l'accès aux mémoires de stockage.

- Méthodes phonologiques : Peu de systèmes traitent ces erreurs de substitution d'un graphème à un graphème de même prononciation (*meigre, mègre, maigre...*), d'inversion ou de suppression d'un graphème muet (*éthymologie - étymologie...*). PIAF<sup>7</sup> construit toutes les formes phonétiques d'un mot analysé et génère les formes orthographiques possibles à partir d'un transducteur d'états finis. VORTEX<sup>8</sup> traite les fautes typographiques et les fautes orthographiques. Après avoir transcrit le mot phonétiquement, il génère toutes les fautes possibles à partir d'un codeur orthographique (modèle des fautes les plus fréquentes), d'un canal typographique (modèle des fautes typographiques) et d'un canal de permutation (modèle des erreurs de permutation).

(1) P.A.V. HALL, G.R. DOWLING : "Approximate String Matching", *Computers Surveys*, 12, 4, 1980

(2) J.L. PETERSON : "Computers Programs for Detecting and Correcting Spelling Errors", *Communications ACM*, 23, 12, pp. 676-687, 1980

(3) J. ULLMAN : "A Binary n-gram Technique for Automatic Correction of Substitution, Deletion, Insertion and Reversal Errors in Words", *Computer Journal*, 20, 2, pp. 141-147, 1977

(4) A.J. SZANZER : "Error Correcting Methods in Natural Language Processing", *Information Processing*, 68, IFIP, pp. 1412-1416, 1969

(5) H.L. MORGAN : "Spelling Correction in System Programs", *Communications ACM*, 18, 1, pp. 54-64, 1970

(6) F. DEBILI : "Les fonctionnalités linguistiques exigibles d'un système de documentation automatique", *Congrès informatique et Droit, ADIJ*, Strasbourg, 1987

(7) J. COURTIN : "Algorithmes pour le traitement interactif des langues naturelles", *Thèse de Doctorat ès Sciences*, Grenoble, 1977

(8) G. PERENNOU, P. DAUBEZE, F. LAHENS : "La vérification et la correction automatique de textes : le système VORTEX", *Technique et Science informatiques*, 5, 4, pp. 285-305, 1986

E. LAPORTE propose un système qui part d'une transcription phonétique du mot testé et la compare avec un dictionnaire de formes phonétiques générées automatiquement à partir du DELAS, le dictionnaire du LADL (paragraphe 2.5.3.6.3).

Les méthodes évoquées ci-dessus permettent de trouver un mot voisin du mot que l'on recherche dans le lexique des mots attestés. Un autre point de vue consiste à proposer des caractéristiques potentielles à propos d'un mot qui existe mais ne figure pas dans le dictionnaire. Les techniques proposées par J. VERGNE<sup>1</sup> et P. PAGES<sup>2</sup> s'appuient sur l'étude du système dérivationnel et la décomposition du mot (racines et affixes) afin d'en déduire des informations morphosyntaxiques et sémantiques<sup>3</sup>.

## 1.5.10.2 Plan syntaxique

### 1.5.10.2.1 Types d'erreur

Les fautes rencontrées concernent les relations qu'entretiennent entre eux les mots de la phrase : on peut noter :

- l'oubli : *il n'y a pas d'accord ce point / il n'y a pas d'accord sur ce point*
- le redoublement : *la décomposition du du mot / la décomposition du mot*
- la permutation : *ainsi de que nombreux signes / ainsi que de nombreux signes*
- les erreurs de construction : *je vais au dentiste / je vais chez le dentiste, je m'en rappelle / je me le rappelle*
- les fautes d'accord : *j'irais / j'irai, les chat / les chats, vous parler / vous parlez*

### 1.5.10.2.2 Techniques de correction

Il existe plusieurs voies pour traiter les erreurs de construction :

- augmenter la grammaire : Certaines fautes sont si fréquentes qu'on peut les assimiler à des structures acceptables en les réunissant dans une "grammaire de fautes".
- assouplir les contraintes : L'application des règles de grammaire est soumise à certaines contraintes. Certains auteurs<sup>4</sup> ont montré qu'il était possible, dans certains cas, de négliger des augmentations dans un ATN pour relancer l'analyse bloquée sur un échec.
- réduire le champ de l'analyse : Analyser une phrase revient à lui faire correspondre un sous-ensemble de règles. On peut imaginer des procédures qui limiteraient cette correspondance à certaines règles<sup>5</sup> ou à certains mots.

---

(1) J. VERGNE : "Une méthode structurelle de reconnaissance des formes pour l'analyse morpho-syntaxique du français sans dictionnaire ; Etude de faisabilité sur les groupes nominaux complexes", *Congrès RF-LA, AFCET-INRIA, 1987*

(2) P. PAGES : "Analyse morphologique automatique du français, extraction des verbes et mise en valeur morpho-sémantique de la dérivation", *Thèse de 3ème cycle, INALCO, Paris III, 1984*

(3) M. PONAMALE : "Compréhension automatique de mots inconnus à partir de leur morphologie", *Mémoire de l'ITIE, CNAM, 1987*

(4) R. WEISCHEDEL, J. BLACK : "Responding to Potentially Unparseable Sentences", *American Journal of Computational Linguistics*, 6, pp. 26-45, 1980

(5) P. HAYES, G. MOURADIAN : "Flexible Parsing", *American Journal of Computational Linguistics*, 7, 4, pp. 232-241, 1983

- réduire l'analyse au plan sémantique et ignorer les erreurs syntaxiques.

Ces différentes voies ont été reprises en partie par C. FOUQUERE<sup>1</sup> qui pour analyser une phrase mal construite, propose de réaliser une analyse non linéaire capable de repérer des zones correctes (ilôts de confiance) et d'extraire des informations pour interpréter les zones qui ne peuvent être interprétées.

Basé sur les ATNs, un ensemble d'analyses partielles extrait les structures sûres. Il faut ensuite supprimer les points d'embarras en utilisant les méthodes citées plus haut pour rassembler les analyses partielles.

Pour raccourcir le temps de traitement, C. FOUQUERE a construit un formalisme "d'analyse tolérante" utilisant une grammaire dite "à configuration minimale". Les analyses partielles engendrent des arbres que des procédures tentent de relier lorsque l'analyse est bloquée.

En ce qui concerne les règles d'accord, les systèmes limités peuvent s'offrir le luxe de les ignorer.

Il est évident que les informations liées au genre, au nombre, à la personne, au mode et au temps sont d'une importance capitale. Les fautes qui sont repérées concernent essentiellement des paires de mot, le déterminant ou l'adjectif avec un nom, le sujet et le verbe.

Plusieurs critères sont alors indispensables pour déterminer la bonne correction qui, en général, répond à l'heuristique bien connue selon laquelle "la bonne correction" implique le minimum de changements.

### 1.5.10.3 Plan sémantique

#### 1.5.10.3.1 Les types d'erreur

Même à l'intérieur d'un domaine limité, les erreurs sont difficilement repérables.

- les erreurs conceptuelles reflètent une conception erronée de l'auteur dans ses suppositions ou dans son raisonnement.
- les erreurs pragmatiques concernent les messages corrects que l'on émet, dans une situation donnée, de façon inappropriée.

#### 1.5.10.3.2 Techniques de correction

Dans l'état actuel des recherches, on préfère opter pour une limitation claire des systèmes, et une transparence pour l'utilisateur qui pourra lui-même en contrôler les possibilités. Les travaux réalisés jusqu'à maintenant concernent surtout les systèmes de questions-réponses<sup>2</sup>

Nous dirons, en résumé, que si la correction automatique des erreurs lexicales est une pratique courante et agrémente d'une façon plus ou moins efficace les logiciels actuels de traitement de texte, la correction des erreurs syntaxiques et sémantiques en est au stade des balbutiements et ne pourra progresser qu'avec le développement des recherches sur la représentation des connaissances et les mécanismes de compréhension.

---

(1) C. FOUQUERE : "Systèmes d'analyse tolérante du langage naturel", *Thèse de l'Université de Paris-Nord*, 1988

(2) J.B. BERTHELIN, G. SABAH : "Degrés qualitatifs d'acceptabilité en informatique linguistique", *Actes Cognita, AFCET-ARC-CESTA*, Paris, pp. 269-275, 1985

## 1.6 Conclusion

Nous avons abordé les relations du langage avec la linguistique et l'informatique à travers les modèles de la langue et les outils d'analyse. Cette présentation simplifiée et réduite à l'analyse illustre la complexité d'un domaine dont la Traduction Automatique et la Traduction Assistée par Ordinateur ne sont que des exemples d'application.

Le traitement automatique des langues naturelles couvre en effet de nombreux autres domaines, parmi lesquels :

- L'interrogation de banques de données : Elle est au coeur des recherches depuis plus de 25 ans. Les systèmes concernés traduisent une requête exprimée en langue naturelle sous la forme d'un langage d'interrogation. L'interrogation de banques de données déductives ne se contente pas de chercher des informations explicites, elle implique une certaine déduction.
- Dans une approche fondée sur la logique, les informations contenues dans la banque sont les axiomes d'un système logique au sein duquel on applique les règles d'inférences de la logique classique. PROLOG est un outil privilégié dans le sens où il permet de représenter les informations dans la base, les règles de déduction, les règles de grammaire et la génération des réponses.
- Dans une approche fondée sur les réseaux sémantiques, on utilise par défaut les processus de déduction de la logique de premier ordre, en l'absence de moyens plus efficaces<sup>1</sup>.
- La génération de texte : Bien que le problème soit vital, il a toujours été masqué par les techniques de représentation des connaissances. A la méthode fill-in-the-blanks, qui consiste à remplir des phrases stéréotypées (ELIZA<sup>2</sup>) par des paramètres fournis en cours de traitement par différents modules, on peut ajouter les travaux de N. GOLDMAN<sup>3</sup> visant à générer du texte dans le cadre du système de paraphrasage de R. SCHANK, les recherches de J. SLOCUM<sup>4</sup> et de R. SIMMONS<sup>5</sup> à partir des réseaux sémantiques, celles de R. GRISHMAN<sup>6</sup> à partir du calcul des prédicats, de L. DANLOS<sup>7</sup> et de N. SIMONIN<sup>8</sup>. Citons également les générateurs liés aux systèmes de Traduction Automatique.
- La documentation automatique : Les quantités de texte peuvent être énormes, aussi ne travaille-t-on pas sur le texte lui-même. On préfère utiliser un système d'indexage géré par des fonctions booléennes appliquées à des mots clés. J.M. DAVID<sup>9</sup> propose des techniques plus complexes d'accès aux documents. L'Intelligence Artificielle n'a cependant rien apporté de plus, pour l'instant sur un plan opérationnel.

- 
- (1) G. HENDRIX : "Encoding Knowledge in Partitioned Networks" in N. FINDLER (ed.) *Associative Networks : Representation and Use of Knowledge by Computer*, Academic Press, New York, 1979
  - (2) J. WEIZENBAUM : "ELIZA, a Computer Program for the Study of Natural Language Communication between Man and Machine", *Communications ACM*, 9(1), pp. 36-45, 1966
  - (3) N. GOLDMAN : "Conceptual Generation" in R. SCHANK (ed.) *Conceptual Information Processing*, North Holland, pp. 289-371, 1975
  - (4) J. SLOCUM : "Generating a Verbal Response" in D. WALKER (ed.) *Understanding Spoken Language*, North Holland, pp. 375-380, 1975
  - (5) R. SIMMONS, J. SLOCUM : "Generating English Discourse from Semantic Networks", *Communications ACM*, 15(10), pp. 891-905, 1972
  - (6) R. GRISHMAN : "Response Generation in Question Answering Systems", *Proc. of 17th Am. for Computational Linguistics*, La Jolla, pp. 99-101, 1979
  - (7) L. DANLOS : "Génération automatique de textes en langue naturelle", Paris, Masson, 1985
  - (8) N. SIMONIN, J.M. LANCEL : "Essai de modélisation de l'expertise en rédaction de textes", *Actes Cognitiva*, Paris, pp. 263-268, 1985
  - (9) J.M. DAVID : "Simulation de l'activité d'un documentaliste", *Actes du congrès AFCET Informatique*, Nancy, pp. 362-371, 1980

- Le traitement de texte : Les progrès de la bureautique encouragent le développement d'outils linguistiques pouvant manipuler les textes à des niveaux morpho-lexicaux ou même syntaxiques mais de façon très rudimentaire (dictionnaires et correcteurs, aides à la rédaction).
- La robotique : L'application concerne la communication avec les robots (dialogue homme-machine).
- L'Enseignement Assisté par Ordinateur (E.A.O.) : Le contexte de l'E.A.O. se prête à la construction de dialogues de bonne qualité car la situation pédagogique restreint le champ sémantique et le système ne requiert pas de grandes masses de connaissances. M. PEUCHOT<sup>1</sup>, D. COULON et D. KAYSER<sup>2</sup> ont construit des systèmes assez rudimentaires qui donnent de bons résultats. Des systèmes plus évolués, comme SOPHIE<sup>3</sup> et GUIDON<sup>4</sup> incorporent des analyseurs du langage naturel et étendent la marge de liberté de l'étudiant. Nous pensons qu'il serait intéressant d'appliquer ces techniques non pas seulement à la forme mais aussi au contenu des didacticiels, qui auraient tout à y gagner, en ce qui concerne le domaine des langues, bien entendu. En annexe, nous donnerons un exemple de ce qu'il est possible de proposer dans ce cadre, comme sous-produit de notre analyseur.
- Résolution d'énoncés en langage naturel : La sémantique réduite permet d'obtenir de très bons résultats. De nombreuses applications ont suivi le programme STUDENT<sup>5</sup>, dont celle de M. ROUSSEAU<sup>6</sup>
- La traduction automatique : On peut considérer que ce domaine constitue l'exemple des difficultés qui ont été mal mesurées en ce qui concerne le traitement automatique des langues naturelles. Y. BAR-HILLEL<sup>7</sup> avait déjà subordonné la bonne qualité d'une traduction à la représentation de grandes quantités de connaissances. Des résultats satisfaisants ont pu être obtenus en utilisant au maximum les règles syntaxiques ou en incorporant localement des traitements sémantiques.

On constate que la majorité des thèmes évoqués concernent surtout l'Intelligence Artificielle. La rapidité et l'ampleur de son développement éclipsent quelque peu la T.A. et la T.A.O. qui n'occupent plus le devant de la scène du Traitement Automatique des Langues Naturelles. La raison nous paraît fort simple : les objectifs de la traduction automatique sont proprement inaccessibles, dans l'état de nos connaissances sur la langue et les résultats ne sont pas à la mesure des investissements consentis. La linguistique a beaucoup gagné dans l'aventure mais les progrès que l'on attend d'elle sont-ils à sa portée ?

Les chagrins diront que le formalisme a ses limites. Les méthodes sont nombreuses, nous venons d'en voir quelques unes, et peut-être inadaptées. N'a-t-on pas cherché à imposer à la langue un formalisme qui lui était complètement extérieur et qui, par conséquent, ne peut pas toujours fonctionner correctement ? Issu des mathématiques (de la logique et de la théorie des graphes), son origine n'est-elle pas malencontreuse ?

---

(1) M. PEUCHOT : "De l'enseignement assisté à la diffusion automatique de l'information", *Zéro-Un Informatique*, mai 1971

(2) D. COULON, D. KAYSER : "Analyse de réponses rédigées en Français courant pour une réalisation d'enseignement programmé", *RAIRO*, revue bleue, n°2, pp. 61-98, 1972

(3) J. BROWN, A. BELL : SOPHIE, "A Step Toward Creating a Reactive Learning Environment", *Int. Journal of Man-Machine Studies*, 7(5), pp. 675-696, 1975

(4) W. CLANCEY : "Tutoring Rules for Guiding a Case Method Dialogue", *Int. Journal of Man-Machine Studies*, 11(1), pp. 25-49, 1979

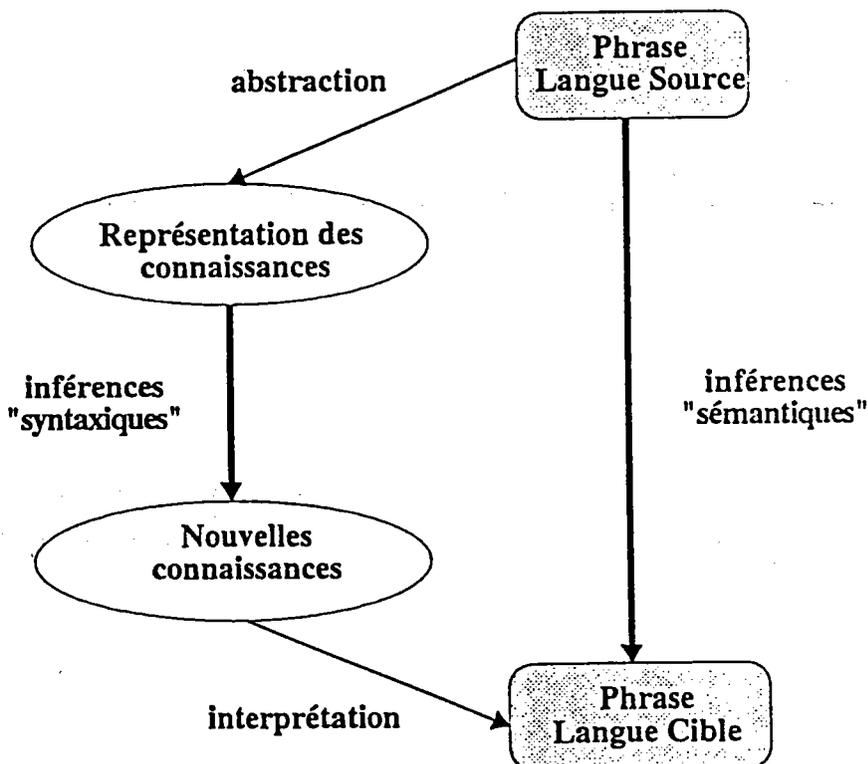
(5) D. BOBROW : "A Question-answering System for High School Algebra Word Problems", Fall Joint Computer Conference, pp. 591-614, 1964

(6) M. ROUSSEAU : "Résolution automatique d'exercices d'électricité posés en Français", Thèse de 3ème Cycle, Université de Paris VI, 1973

(7) Y. BAR-HILLEL : "A Demonstration of the Non Feasibility of Fully Automatic High Quality Translation" in Y. BAR-HILLEL : *Language and Information*, Addison-Wesley, pp. 174-179, 1964

Peut-on parler d'une confusion entre la langue agissant comme métalangue mathématique et la langue ordinaire ? Il est bien évident que ce n'est pas parce que l'on peut tout exprimer de l'univers mathématique avec sa propre langue que cette même langue présente inéluctablement une structure mathématique. Il n'y a pas un seul *et*, un seul *ou*, mais une vingtaine, leur définition étant particulièrement incertaine. Qu'il s'agisse de la syntaxe ou de la sémantique, nous avons évoqué leurs aspects formels et fonctionnels. Si les premiers ont pour but de proposer des modèles de représentation des structures sous-jacentes aux langues analysées, les secondes s'attachent surtout à décrire le rôle des différents constituants. Les théories qui mettent l'accent sur le fonctionnel ne s'appuient pas sur la même rigueur mathématique que les descriptions formelles. Il n'est d'ailleurs pas prouvé que les langues soient clairement formalisables et il est fort probable que l'homme ne les utilise pas ainsi. Dès lors, les formalismes conviennent-ils au traitement automatique des langues ?

Les activités cérébrales liées au langage semblent informelles. L'exemple que fournit la traduction illustre le fait que nous raisonnions probablement sur des significations sans passer par un formalisme quelconque. Selon ARSAC, un ensemble de règles morphologiques (formelles) permet de reconnaître les constituants d'une phrase latine. Il est alors possible de construire une phrase en français ayant la même structure syntaxique, en remplaçant les mots latins par leur équivalent français. L'opération est ici de type essentiellement syntaxique. Le sens est par contre au centre des activités de traduction simultanée où l'on exprime aussitôt dans la langue cible l'image qu'exprime la langue source. Le traitement sémantique court-circuite alors le traitement syntaxique. On distingue dans tous les cas *la forme* d'une donnée (support matériel) et *son sens* (ce que manipule le cerveau de celui qui interprète la forme). Si l'on traite une phrase d'une langue source vers une langue cible en utilisant le formalisme, il faudra passer par une représentation des connaissances (abstraction) et la construction de nouvelles connaissances (inférences "syntaxiques") avant de parvenir à la phrase cible (interprétation). L'autre méthode consisterait à passer directement par le sens (inférences "sémantiques").



L'ordinateur doit traiter les problèmes que l'homme résout de façon sémantique. Il peut obtenir les mêmes résultats sans reproduire les différents processus ou chercher au contraire à reproduire, à un certain niveau, les opérations mentales impliquées. On se heurte alors à un obstacle de taille car, pour formaliser le sens, il faut passer par un traitement syntaxique qui ne conserve que la forme, tandis que disparaît le sens. Ceci nous conduit à une question fondamentale : *notre cerveau accomplit-il quelque chose qui ne peut pas être formalisé ?*

- Si l'on répond de façon positive, on rejoint les critiques qui affirment qu'en ce qui concerne l'homme, "tout ce qui touche à la compréhension est intimement lié à son expérience sensorielle et émotionnelle, éléments fondamentalement non formalisables". On pourra se référer à H. DREYFUS<sup>1</sup> qui condamne l'Intelligence Artificielle. T. WINOGRAD<sup>2</sup> s'en rapproche quelque peu lorsqu'il ne croit plus que la sémantique soit formalisable. Considérant que "la notion d'engagement est à la base du langage", il affirme que les machines ne seront jamais intelligentes. J. WEIZENBAUM<sup>3</sup>, un pionnier dans le domaine, insiste sur les différences fondamentales entre l'homme et la machine et rappelle que certaines opérations seront toujours réservées à l'homme.

- Les optimistes, quant à eux, pensent que l'on peut appréhender le sens et le formaliser sans le vider de sa substance. Ils feront état d'avancées spectaculaires dues essentiellement aux apports de l'Intelligence Artificielle dont se réclament les systèmes récents et les projets de 5ème génération.

- Nous pensons qu'il est impossible d'émettre un jugement définitif. Les recherches actuelles s'attachent à étudier les limites théoriques des formalismes utilisés (développement des sciences cognitives) tout en appliquant les modèles au traitement des données (réalisation de programmes visant à renforcer la crédibilité de l'Intelligence Artificielle). Cette double perspective sera peut-être féconde, à long terme.

Nous avons par conséquent adopté une position intermédiaire. Sans condamner la T.A. et la T.A.O., nous croyons qu'elle se développera, mais en adaptant ses objectifs aux réalités linguistiques et économiques. S'il est vain de viser une traduction entièrement automatique, il n'est pas impossible de s'en approcher, en limitant les domaines abordés (systèmes en sémantique fermée (2.3.3.3.7.1), en contrôlant la forme du texte d'entrée ((2.3.3.3.9) ou en important des techniques de l'I.A.. Proposant une autre voie, nous limiterons nos ambitions à une forme particulière de traduction, à l'interprétation automatique, avec l'Automate de Compréhension Implicite (chapitre III et IV).

Nous allons maintenant illustrer les modèles de ce chapitre, avec une présentation des différentes approches, des réalisations et des projets de T.A. et de T.A.O.

---

(1) H. DREYFUS : *Intelligence artificielle : mythes et limites*, Flammarion, 1984

(2) T. WINOGRAD, F. FLORES : *Understanding computers and cognition*, Norwood, N.J. Ablex publishing corporation, 1986

(3) J. WEIZENBAUM : *Computer power and human reason : from judgment to calculation*, W.H. Freeman, San Francisco, 1976

## **II.**

**TRADUCTION AUTOMATIQUE (T.A.)  
TRADUCTION ASSISTÉE PAR ORDINATEUR (T.A.O.)**

## 2.1. Industrie de la langue<sup>1,2</sup>

Nées du mariage de la linguistique et de l'informatique, les machines parlent, lisent, traduisent et déchiffrent la voix humaine. Un des grands marchés du siècle.

Les industries de la langue sont en train de réaliser un vieux rêve de l'humanité : construire des machines capables de dialoguer avec l'homme. Le principal marché sera monolingue (correcteurs d'orthographe, systèmes d'accès aux bases de connaissances, systèmes d'apprentissage) mais les institutions communautaires et la mondialisation de l'économie consommeront des produits multilingues (banques terminologiques, systèmes de traduction assistée par ordinateur...).

C'est en juin 1982 (Colloque de l'Association COFORMA, Communication-Formation) qu'apparaît le concept des industries de la langue, repris dans les Actes du Colloque (décembre 1982, Palais Bourbon) sous la forme d'un titre : "Pour une filière industrielle du langage".

Cette industrie sous-tend une recherche fondamentale et une recherche appliquée qui débouchent sur des produits visant des marchés à court et moyen terme et s'appuient de plus en plus sur l'Intelligence Artificielle au point qu'elle deviendra rapidement le dénominateur commun de toutes ces activités. Leur domaine d'usage s'étend sans cesse, de la bureautique aux services, en passant par les télécommunications, les automatismes industriels, la productique, l'enseignement, la formation, la santé, l'aide aux handicapés, l'électronique grand-public, la sécurité, l'informatique, l'électronique professionnelle.

A l'intersection des activités linguistiques, des traitements informatiques correspondants, des pratiques industrielles et des marchés, parmi les nombreuses applications qui vont de la compréhension du langage naturel<sup>3,4,5,6,7</sup> aux systèmes experts<sup>8,9,10,11,12</sup>, en passant par la communication parlée, les interfaces en langage naturel<sup>13,14,15</sup>, la lecture optique<sup>16-25</sup>, le traitement de texte<sup>26-30</sup>, l'hypertexte<sup>31</sup>, la documentation<sup>32-36</sup>, nous concentrerons notre attention sur la T.A. (Traduction Automatique), la T.A.O. (Traduction Assistée par Ordinateur) et les dictionnaires électroniques multilingues.

- (1) R. GELLY : "Les vrais trésors de la langue française" in *Ça m'intéresse*, novembre 1986
- (2) W. BARANES : "Les industries de la langue" in *Qui Vive International*, septembre 1986
- (3) D. KAYSER : "Des machines qui comprennent notre langue" in *La Recherche*, octobre 1985
- (4) J.-P. DESCLES : "La linguistique informatique" in *Le Courrier du CNRS*, mai 1986
- (5) E. SEYDEN : "Langage naturel : L'ordinateur comprend ce qu'il peut" in *L'Ordinateur Individuel*, octobre 1986
- (6) P. LOMBARD : "Jouer sur le sens des mots" in *01 Informatique*, février 1987
- (7) X. DALLOZ : "Compréhension du langage naturel à la Convention Informatique" in *Minis et Micros*, octobre 1986
- (8) C. DAVID : "Les progrès de l'Intelligence Artificielle" in *Problèmes Economiques/Le Nouvel Economiste*, janvier 1987
- (9) J.-G. GANASCIA : "La conception des systèmes experts" in *La Recherche*, octobre 1987
- (10) I. GROSSE : "Quelques systèmes experts originaux" in *01 Informatique*, mai 1987
- (11) F. HAUTIN : "Intelligence Artificielle" in *Bulletin de liaison de la Recherche en Informatique et en Automatique*, n° 111, 1987
- (12) C. BRANCIER : "LIA au service de la langue" in *Décision Informatique*, mai 1987
- (13) H. MADEC : "Intelligence Artificielle : Les interfaces en langue naturelle" in *Minis et Micros*, février 1986
- (14) H. KEMPF : "Le minitel parlera couramment français" in *Science et Vie Micro*, avril 1987
- (15) E. MONTAGNE : "LIA sur le réseau Télétel" in *01 Informatique*, avril 1987
- (16) A. BALAID, J.-P. HATON : "La reconnaissance de l'écriture" in *La Recherche*, octobre 1985
- (17) D. CHARRAUT, J. DUVERNOY, L. HAY : "L'analyseur automatique de l'écriture" in *La Recherche*, janvier 1987

## 2.2 La traduction

La traduction automatique et la traduction manuelle partagent tous les problèmes qui découlent de la signification. Le linguiste peut identifier et expliquer les difficultés d'un texte source. Le traducteur et l'informaticien ne disposent pas des mêmes outils et envisagent des solutions diverses qu'il est beaucoup plus difficile de déterminer en Traduction Automatique qu'en traduction humaine.

La conception et la mise au point de l'ordinateur ont suscité un enthousiasme immodéré au cours "des années 50". On pensait que des programmes adéquats traduiraient le terme d'une langue par son équivalent dans une autre langue. Cette vision simpliste assimilait le processus de traduction au déchiffrement de cryptogrammes à l'aide d'un dictionnaire. La désillusion fut profonde lorsque les non-linguistes mesurèrent la complexité et le nombre de paramètres impliqués.

### 2.2.1 Les outils de la traduction informatisée

La réduction brutale des subventions consacrées à ce domaine par le gouvernement des Etats-Unis, en 1955, a fait prendre conscience des difficultés du sujet.

A partir de ce moment et parallèlement à la poursuite des recherches fondamentales, les efforts se sont dirigés vers des systèmes opérationnels, chargés de soulager la tâche du traducteur. L'évolution de l'informatique a une grande influence sur la mise en place de ces outils, tandis que les gros calculateurs restent les instruments privilégiés d'applications lourdes et centralisées.

On distingue ainsi trois catégories d'outils de traduction informatisée.

1. La Traduction Assistée par Ordinateur (T.A.O.) qui offre une assistance à la traduction en augmentant la productivité du traducteur. Les limites des systèmes et la qualité

- (18) J.-P. PETIT : "Acquisition de données : La reconnaissance de forme sort des brumes" in *L'Ordinateur Individuel*, décembre 1986
- (19) H. KEMPF : "L'écriture réinventée" in *Science et Vie Micro*, juin 1987
- (20) P. DESMEDT : "Les systèmes de reconnaissance de caractères disponibles sur micro" in *L'Ordinateur Individuel*, décembre 1986
- (21) NICHIREN : "Scanners et lecture optique : la saisie de documents" in *Soft et Micro*, juin 1987
- (22) F. COUTROT : "Lecture automatique de textes" in *Micro Systèmes*, août 1987
- (23) X. CHIFFELLE : "Edition électronique : les scanners en vedette" in *01 Informatique*, n° 962, juin 1987
- (24) E. MONTAGNE : "La lecture intelligente" in *L'Ordinateur Individuel*, mars 1987
- (25) T. OUTREBON : "Innovatic, un précurseur" in *Décision Informatique*, juin 1987
- (26) A. CAPPUCIO : "Turbo Lightning : L'orthographe assistée par ordinateur" in *Micro Systèmes*, octobre 1986
- (27) Y. GARRET : "Alpha : un correcteur orthographique Borland- Larousse" in *Science et Vie Micro*, juillet 1987
- (28) Y. DARGERER : "Au banc d'essai, les traitements de texte vedettes de la rentrée" in *Science et Vie Micro*, août 1987
- (29) M. GUENZET : "Traitement de texte et édition" in *Décision Informatique*, mai 1987
- (30) T. COURTOIS : "Manuscript : au-delà du texte, un traitement de document" in *Décision informatique*, juin 1987
- (31) G. BOUSQUET : "L'hyper-texte : la grande réforme de l'écriture électronique" in *01 Informatique*, mai 1987
- (32) J. ARSAC : "Informatique documentaire : agir sans plus attendre" in *01 Informatique*, mai 1987
- (33) F. GICQUEL : Dossier Documentation in *01 Informatique*, mai 1987
- (34) C. FLUHR : "Informatique documentaire : peut mieux faire" in *01 Informatique*, mai 1987
- (35) M. OLANIE : "CD-ROM : Enfin le démarrage" in *Décision Informatique*, mai 1987
- (36) M. GUENZET : "Intelligence Artificielle et CD-ROM" in *Décision Informatique*, mai 1987

des résultats nécessitent une révision du traducteur qui dispose d'un environnement adapté (poste de travail et outils intégrés). Certains auteurs subdivisent la T.A.O. en deux groupes<sup>1</sup> :

Traduction Automatique Assistée par l'Homme (HAMT Human-Assisted Machine Translation) : l'ordinateur accomplit la traduction en soi mais il y a interaction avec l'homme à différentes étapes du processus :

- Préparation du texte d'entrée (pré-édition).
- L'ordinateur demande l'aide du traducteur pour lever certaines ambiguïtés, pour expliciter des liaisons de phrase, pour choisir la traduction d'un mot ou d'une phrase parmi les possibilités qu'offrent le dictionnaire de transfert. Ces systèmes qui nécessitent une interaction homme-machine pendant le processus de traduction sont des systèmes "interactifs".
- Edition du texte de sortie (post-édition).

Traduction Humaine Assistée par Ordinateur (MAHT Machine-Assisted Human Translation) : l'homme accomplit la traduction en soi mais peut utiliser l'ordinateur dans certaines situations :

- Assistance pour la recherche dans un thésaurus.
- Accès à une banque terminologique lointaine.
- Affichage d'exemples à propos de l'utilisation d'un mot dans une phrase. Le KWIC (Key Word in Context) permet parfois de reconnaître des homographes.
- Traitement de texte avec un dictionnaire spécifique. On distingue les ADL (Automatic Dictionary Look-up) dont les entrées peuvent être introduites directement dans la traduction en utilisant une simple touche de fonction et les SDL (Selective Dictionary Look-up) qui ne livrent une traduction que pour certaines expressions.
- Mise à disposition d'informations grammaticales.
- Analyse morphologique.
- Accès à des traductions de textes voisins.
- Correction automatique des fautes de frappe.

2. La Traduction Automatique (T.A.) : Les systèmes sont prévus pour accomplir des traductions sans intervention de l'homme. On ne peut toutefois pas éviter une pré-édition et/ou une post-édition. Ces systèmes assurent le processus complet de traduction, depuis l'entrée du texte source jusqu'à la sortie du texte cible, sans intervention humaine, en enchaînant des programmes, en utilisant des dictionnaires et des règles de grammaires. Ils sont souvent limités à des domaines d'application étroitement spécifiés relatifs à un univers langagier appauvri du point de vue lexical et syntaxique.

3. Les banques de données terminologiques : L'accès est indépendant du processus de traduction. L'avantage ne réside pas dans leur automatisation (on peut trouver des mots rapidement dans un dictionnaire imprimé) mais dans le fait qu'elles soient actuelles. La terminologie évolue continuellement et les dictionnaires sont obsolètes lorsqu'ils sont édités. Un autre avantage : leur volume croît au fur et à mesure qu'interviennent les utilisateurs qui ont intérêt à les enrichir.

---

(1) J. SLOCUM : "A Survey of Machine Translation : its History, Current Status, and Future Prospects" in J. SLOCUM (ed.) : *Machine Translation Systems*, Cambridge University Press, pp. 1- 45, 1988

### 2.2.2 Les besoins

Le développement de ce nouveau domaine a dû se plier aux lois de l'efficacité et de la rentabilisation au fil d'une demande qui n'a cessé de croître de façon vertigineuse. Pour l'année 1988, 240 millions de pages ont été traduites en Europe, 200.000 périodiques spécialisés ont été publiés dans le monde.

La Communauté Européenne emploie en permanence 1600 traducteurs, sans compter les traducteurs indépendants à qui elle fait appel en permanence. Le prix d'une page traduite et révisée varie de 60 à 300 F., les budgets consacrés sont énormes, 21 milliards de Francs pour la CEE en 1985.

L'intérêt de la T.A.O. devient évident, lorsque les systèmes atteignent un certain niveau de qualité pour un prix raisonnable. Nous sommes ici au coeur du problème. Il n'est pas question, à notre sens, de condamner les recherches longues et coûteuses qui ont tant fait progresser la linguistique, en prétendant qu'elles conduisent ou conduiront à un échec, comme l'annonçait le rapport ALPAC<sup>1</sup> ou comme l'affirment aujourd'hui, pour un domaine récent, les détracteurs de l'Intelligence Artificielle<sup>2</sup>. Il serait préférable de mieux cerner le problème de la traduction automatisée, en intégrant dans le raisonnement les besoins et les moyens de tous types. La définition des buts à atteindre et des projets correspondants devrait alors respecter des critères de faisabilité et de gestion qu'elle n'observe pas souvent pour des raisons politiques ou économiques malheureuses. Il est d'autant plus regrettable de consommer une énergie considérable dans des travaux de recherche décousus ou répétés par plusieurs laboratoires que l'investissement indispensable est énorme si l'on considère les bénéfices à court terme.

### 2.2.3 Evolutions théoriques et techniques

Sans examiner dans le détail l'histoire de la T.A. et de la T.A.O., nous le ferons dans le paragraphe 3.3.1, prenons un peu de recul. Deux aspects majeurs se dégagent :

- En 1960, la T.A.O. et ses devancières, la Traduction Automatique ou semi-automatisée, sont au centre du traitement automatique des langues naturelles. Trente ans après, elles n'occupent plus cette place privilégiée. La communication parlée, l'informatique documentaire, la communication homme-système, les bases de connaissances, l'assistance à la compréhension et à la génération de textes concentrent sur elles l'essentiel des budgets de recherche/développement et génèrent des marchés considérables. Beaucoup plus que le discrédit dû à des ambitions irréalistes et des résultats décevants, le vieillissement de ses fondements théoriques a privé la T.A.O. du premier rôle. La conception de l'automatisation est continuellement remise en question par le développement extraordinairement rapide des techniques de l'informatique. Les modèles linguistiques qui sont à la source des premiers systèmes ont montré leurs limites. Il ne faut cependant pas, pour autant, négliger les intérêts en jeu comme la capacité de certaines communautés scientifiques et techniques à exploiter l'information présentée dans des langues étrangères, et à communiquer directement avec ces dernières. La zone d'influence des unes s'accroît, celle des autres se réduit, tout comme les potentiels scientifique, technique, industriel et économique qui garantissent l'autonomie ou la dépendance des nations.

---

(1) ALPAC : Languages and Machines : Computers in Translation and Linguistics. Report of the Automatic Language Processing Advisory Committee, Division of Behavioral Sciences, National Academy of Sciences, National Research Council Publication 1416, Washington D.C., 1966

(2) H. DREYFUS : *Intelligence Artificielle : mythes et limites*, Paris, Flammarion, 1985

## 2.2.2 Les besoins

Le développement de ce nouveau domaine a dû se plier aux lois de l'efficacité et de la rentabilisation au fil d'une demande qui n'a cessé de croître de façon vertigineuse. Pour l'année 1988, 240 millions de pages ont été traduites en Europe, 200.000 périodiques spécialisés ont été publiés dans le monde.

La Communauté Européenne emploie en permanence 1600 traducteurs, sans compter les traducteurs indépendants à qui elle fait appel en permanence. Le prix d'une page traduite et révisée varie de 60 à 300 F., les budgets consacrés sont énormes, 21 milliards de Francs pour la CEE en 1985.

L'intérêt de la T.A.O. devient évident, lorsque les systèmes atteignent un certain niveau de qualité pour un prix raisonnable. Nous sommes ici au coeur du problème. Il n'est pas question, à notre sens, de condamner les recherches longues et coûteuses qui ont tant fait progresser la linguistique, en prétendant qu'elles conduisent ou conduiront à un échec, comme l'annonçait le rapport ALPAC<sup>1</sup> ou comme l'affirment aujourd'hui, pour un domaine récent, les détracteurs de l'Intelligence Artificielle<sup>2</sup>. Il serait préférable de mieux cerner le problème de la traduction automatisée, en intégrant dans le raisonnement les besoins et les moyens de tous types. La définition des buts à atteindre et des projets correspondants devrait alors respecter des critères de faisabilité et de gestion qu'elle n'observe pas souvent pour des raisons politiques ou économiques malheureuses. Il est d'autant plus regrettable de consumer une énergie considérable dans des travaux de recherche décousus ou répétés par plusieurs laboratoires que l'investissement indispensable est énorme si l'on considère les bénéfices à court terme.

## 2.2.3 Evolutions théoriques et techniques

Sans examiner dans le détail l'histoire de la T.A. et de la T.A.O., nous le ferons dans le paragraphe 3.3.1, prenons un peu de recul. Deux aspects majeurs se dégagent :

- En 1960, la T.A.O. et ses devancières, la Traduction Automatique ou semi-automatisée, sont au centre du traitement automatique des langues naturelles. Trente ans après, elles n'occupent plus cette place privilégiée. La communication parlée, l'informatique documentaire, la communication homme-système, les bases de connaissances, l'assistance à la compréhension et à la génération de textes concentrent sur elles l'essentiel des budgets de recherche/développement et génèrent des marchés considérables. Beaucoup plus que le discrédit dû à des ambitions irréalistes et des résultats décevants, le vieillissement de ses fondements théoriques a privé la T.A.O. du premier rôle. La conception de l'automatisation est continuellement remise en question par le développement extraordinairement rapide des techniques de l'informatique. Les modèles linguistiques qui sont à la source des premiers systèmes ont montré leurs limites. Il ne faut cependant pas, pour autant, négliger les intérêts en jeu comme la capacité de certaines communautés scientifiques et techniques à exploiter l'information présentée dans des langues étrangères, et à communiquer directement avec ces dernières. La zone d'influence des unes s'accroît, celle des autres se réduit, tout comme les potentiels scientifique, technique, industriel et économique qui garantissent l'autonomie ou la dépendance des nations.

(1) ALPAC : Languages and Machines : Computers in Translation and Linguistics. Report of the Automatic Language Processing Advisory Committee, Division of Behavioral Sciences, National Academy of Sciences, National Research Council Publication 1416, Washington D.C., 1966

(2) H. DREYFUS : *Intelligence Artificielle : mythes et limites*, Paris, Flammarion, 1985

Les évolutions théoriques et techniques de ce secteur ont été ignorées ou diversement perçues. Une lecture attentive de la littérature consacrée aux différents systèmes révèle de nombreuses incertitudes, lorsqu'il ne s'agit pas de mensonges ou de contradictions. Les distributeurs ou responsables de marketing s'acharnent à confondre une traduction automatique avec une traduction partiellement automatique, rejettent l'idée d'une classification précise en fonction de fondements linguistiques et techniques, refusent les tests qui pourraient être défavorables et se retranchent derrière la notion de secret industriel pour entretenir un flou savant autour de dispositifs récents visant à l'amélioration de l'outil présenté. Ces préoccupations à très court terme sont condamnables.

Il est impossible de croire que des systèmes, spécialisés et construits il y a 15 ans, puissent, sans budget et sans personnel suffisant, rivaliser avec les systèmes entre-tenus à grands frais par la CEE ou les industriels japonais. Prisonniers des techniques et des modèles à partir desquels ils ont été conçus, leur modernisation est problématique, et de toute manière, ne peut être que limitée.

Il est en effet possible d'étudier et d'évaluer le coût, la vitesse et la qualité d'une traduction selon des critères à peu près objectifs. On connaît les caractéristiques que devront posséder les futurs systèmes et les investissements qu'il faudra consentir, tant publics qu'industriels, pour ne pas avoir à utiliser, dans un avenir proche, des systèmes conçus ailleurs qu'en France et pour d'autres langues sources.

L'incohérence des choix opérés par les entreprises illustre les incertitudes qui pèsent sur un secteur plein de contradictions :

- IBM France a renoncé au système ALPS, retenu cependant par IBM Canada, alors qu'IBM Danemark lui préfère LOGOS<sup>1</sup>.
- L'Aérospatiale utilise SYSTRAN à Suresnes, recourt parfois à ARIANE et a choisi WEIDNER à Marignane.

Les deux principales estimations concernant le marché mondial de la traduction ont été réalisées par DATAQUEST en 1986 (500 millions à 4,5 milliards de dollars) et COOPERS ET LYBRAND<sup>2</sup> en septembre 1987 (1 à 3 milliards de dollars). Une étude récente en France situe le marché réel à 1,1 milliard de Francs et le marché potentiel à 1,6 milliards de Francs.

En résumé, nous dirons que le secteur de la T.A. et de la T.A.O. est en pleine évolution. Il suscite de grands appétits commerciaux. Les investissements consentis pour la recherche, le rachat d'anciens systèmes, leur modernisation et leur extension à d'autres couples de langue devront être rentabilisés. Le marché, face à la diversité des offres en T.A.O. et au caractère fantaisiste des caractéristiques techniques annoncées, attend des produits et des services capables de favoriser et d'accroître la circulation des informations nécessaires au progrès scientifique et technique.

Tout en améliorant le rendement des traducteurs, ces outils devront en même temps permettre de diminuer les coûts. Quels seront ces outils, qui les construira et les exportera ? Quelles langues parleront-ils ? En 1980 et pour les 9000 revues scientifiques principales, 12 % des articles étaient rédigés en français, en 1989, ils ne représentent

---

(1) A. ABOU : "Place et perspectives de la T.A.O. dans les industries de la langue" in *Actes du séminaire Traduction Assistée par Ordinateur*, Paris, pp. 7-8, mars 1988

(2) COOPERS et LYBRAND Consulting Group : *Market and Industry Study for Computer Assisted Translation Systems*, Canada, 1987

plus que 7 %. La tentation du "tout anglais" est une réalité, et les Etats-Unis nous promettent la machine à écrire à dictée vocale pour 1991. Le prototype mis au point par IBM à Yorktown reconnaît déjà 5000 mots, la Kurzweil Voice Writer (KVW) de KURZWEIL reconnaît 20.000 mots et a été commercialisée fin 1990 pour 140000 F. La déclaration du Secrétaire Général du Conseil de l'Europe, Marcello OREJA, a valeur d'avertissement : *"Les langues qui ne s'industrialiseront pas cesseront, à plus ou moins brève échéance, d'être des langues véhiculaires, des langues de civilisation"*.

## 2.2.4 Architecture générale

Les modèles de la langue et les outils que nous avons présentés dans le chapitre I de notre exposé concernent la linguistique informatique et ses applications, comme la traduction automatique.

On retrouve ici l'architecture propre à tout traitement du langage naturel, avec des extensions spécifiques (des grammaires de transfert aux différents niveaux de traitement) et des missions particulières (résolution des problèmes de compréhension et de paraphrase des textes sources, transfert d'une représentation vers une autre, à partir d'une représentation contextuelle de la sémantique des catégories de la langue source et à l'aide de procédés capables de les exprimer dans la langue cible).

L'élaboration de dictionnaires de grande taille (tels que le dictionnaire informatique du LADL, Laboratoire d'Automatique Documentaire et Linguistique, paragraphe (1.4.3.6.3.1)), la description des valeurs sémantiques des catégories grammaticales et l'écriture des grammaires de transfert conditionnent la construction de ces représentations pour laquelle on dispose de trois méthodes.

1. On utilise des représentations formelles. Leur formalisme a été mis au point à partir d'une théorie linguistique. Cela revient à définir des représentations informatiques associées aux représentations formelles et aux données linguistiques.  
ex. : L'adaptation des grammaires formelles de N. CHOMSKY aux exigences de l'informatique.

2. En se contentant d'une théorie peu évoluée, on construit des représentations informatiques en adaptant directement des implantations informatiques. Cela revient à fournir une interprétation linguistique aux représentations informatiques et à en justifier la pertinence théorique (R. SCHANK).

3. On utilise un formalisme emprunté à la logique (logique des prédicats...) et on l'adapte à la structure informatique pour représenter les énoncés du langage naturel (langage de programmation PROLOG).

Plus de 20 ans séparent la première version de SYSTRAN (Peter TOMA) et les systèmes conçus actuellement par les industriels japonais.

- Le traitement linguistique des langues naturelles a beaucoup évolué, de l'analyse contrastive et comparative à l'analyse sociolinguistique et pragmatique, en passant par l'analyse transformationnelle.

- La technologie informatique a effectué des progrès considérables et la conception des systèmes en a bénéficié. Les architectures matérielles et les puissances de calcul qu'elles assurent ont modifié la présentation des systèmes qui deviennent interactifs.

## 2.2.5 Techniques linguistiques

On a jusqu'ici classé les systèmes de T.A. en deux catégories selon qu'ils traitent ou non la sémantique. Cette distinction est trop simpliste, lorsqu'elle n'est pas fautive.

Tous les systèmes prétendent faire de la sémantique, ce qui n'est pas exact, dans le détail, et ce qui implique d'autres critères de classement.

Si l'on cherche à classer les systèmes d'après les techniques linguistiques utilisées, on pourra opposer la traduction directe à la traduction indirecte, l'approche "interlingue" à l'approche dite de "transfert" et la vision locale à la vision globale.

**Traduction directe** : le système est créé, à l'origine, pour traduire d'une langue dans une autre langue. Il se limite au travail minimal qui permet d'effectuer la traduction. La levée des ambiguïtés n'est réalisée que lorsque cela s'avère indispensable, et pour une langue cible donnée, sans tenir compte de ce qui serait nécessaire pour une autre langue.

ex. : G.A.T. (Georgetown Automatic Translation, paragraphe 2.3.2.1)

**Traduction indirecte** : l'analyse de la langue source et la synthèse de la langue cible sont deux processus totalement indépendants. Les ambiguïtés sont levées jusqu'à ce que l'on détermine le sens de l'entrée en langue source (quelle que soit la façon dont on le représente) sans tenir compte de la ou des langues cibles.

ex. : EUROTRA (European Translation System, paragraphe 2.3.5.7)

**Approche "interlingue"** : la représentation du sens de ce qui est entré en langue source est indépendante de toute langue. Cette représentation est utilisée dans la synthèse de la langue cible, en sortie. La notion des "universaux" est sous-jacente. La représentation d'une unité de sens serait unique, quelle que soit la langue ou la structure grammaticale.

ex. : CETA (Centre d'Etudes pour la Traduction Automatique, paragraphe 2.3.2.2)

**Approche dite de "transfert"** : la représentation du sens diffère, pour une unité, selon la langue dont on l'extrait ou selon la langue dans laquelle on la génère, ce qui implique une troisième phase dans la traduction. Cette phase organise la représentation spécifique à une langue selon le système d'une autre langue. Il s'agit du transfert. Le processus complet de traduction se décompose alors en trois phases : Analyse, Transfert et Synthèse. La dichotomie "interlingue"/"transfert" ne s'applique pas à tous les systèmes, les systèmes "directs" n'utilisant, précisément, aucune de ces deux approches, puisqu'ils ne cherchent pas à représenter le sens.

**L'analyse** décrit le texte source (lexique et structure) en définissant les relations qui lient les mots. Pour chaque mot un dictionnaire électronique fournit les données morphologiques. Un dictionnaire ou un ensemble de règles proposent ensuite les structures possibles de la phrase analysée. L'analyse reclasse les unités en groupe nominaux, groupes verbaux, groupes prépositionnels.

**Le transfert** : Après qu'un analyseur<sup>1</sup> ait décomposé la phrase en éléments structurels, on recherche l'unité lexicale qui traduira le mot du texte source, conformément aux données fournies par l'analyse, pour obtenir finalement une chaîne de mots ne respectant pas encore la syntaxe de la langue cible.

**La synthèse** constitue la dernière étape au cours de laquelle les unités sont assemblées dans le respect de la morphologie et de la syntaxe, à l'aide des dictionnaires et des grammaires de la langue cible.

ex. : TAUM (Traduction Automatique de l'Université de Montréal, paragraphe 2.3.2.4)

---

(1) On rencontre souvent dans la littérature le terme équivalent en langue anglaise : parser

**Vision locale** : les mots constituent l'unité essentielle dans la conduite de l'analyse. Des procédures distinctes déterminent le niveau de langue, traitent l'expression idiomatique, évaluent le sens, pour chaque mot, en se fondant sur le contexte (à gauche et/ou à droite). Ceci constitue un handicap très sérieux dans le cas des homographes (orthographe identique mais niveaux de langue, schémas dérivationnels et sens différents), puisqu'il n'y a pas de tentative d'analyse globale de la phrase.

ex. : SYSTRAN (System Translation, paragraphe 2.3.3.2.9)

**Vision globale** : le sens du mot est déterminé par son contexte, dans le cadre d'une analyse globale de la phrase (ou, rarement, du paragraphe). Le traitement des homographes pose ici moins de problèmes, dans le sens où le contexte pris en compte est nettement plus large qu'en vision locale.

ex. : METAL (Mechanical Translation and Analysis of Languages, paragraphe 2.3.2.3)

## 2.2.6 L'Intelligence Artificielle

Les systèmes abordés jusqu'ici sont des applications de la linguistique centrées sur l'analyse syntaxique. Les marques sémantiques sont liées aux structures syntaxiques, les études sémantiques suivent l'identification de ces structures.

L'analyse et la synthèse n'ont pas d'autre champ d'observation que la phrase. Les dimensions du contexte leur échappent. Il devient alors impossible de traiter les phénomènes comme la pronominalisation et la topicalisation.

Les projets qui s'inspirent des concepts et des méthodes de l'Intelligence Artificielle accordent à la sémantique le rôle central. Les composantes principales sont désormais l'analyse des éléments sémantiques qui remplace l'analyse des catégories grammaticales, la détermination des réseaux sémantiques (Dépendances conceptuelles de SCHANK) et la résolution des ambiguïtés et des incertitudes sur la base d'informations extralinguistiques. Le but est de comprendre le texte avant de le traduire. Nous avons eu l'occasion de présenter des résultats obtenus dans le cadre des recherches en Intelligence Artificielle (Paragraphe 1.5). Les programmes cités sont parfois très intéressants pour la Traduction Automatique mais n'apportent pas toujours une solution à des problèmes qui restent spécifiques. Il nous semble pourtant évident que ce domaine aurait beaucoup à gagner dans une combinaison des méthodes d'Intelligence Artificielle et des techniques d'analyse automatique.

## 2.3 Les systèmes de T.A. et de T.A.O.

### 2.3.1 Historique

La Traduction Automatique a près de 40 ans. Au fil d'une histoire longue et colorée, elle n'a jamais totalement convaincu. Les adeptes inconditionnels caressent les espoirs les plus fous, les ennemis irréductibles la condamnent.

#### De 1946 à 1964 :

Après l'avènement des premiers ordinateurs (1946) W. WEAVER et A.D. BOOTH envisagent la possibilité d'obtenir une traduction intelligible en remplaçant chaque mot d'une phrase par son équivalent. C'est à l'université de Washington (Seattle) que E. REIFLER envisage une première traduction automatique (1949) à l'aide de la machine de KING, traduction mot à mot effectuée par un automate transducteur d'états finis, incapable, par conséquent, de lever les ambiguïtés.

Commence alors une période d'euphorie, confirmée par l'apparition de revues spécialisées, la succession de congrès internationaux de plus en plus fréquentés (Conférence du M.I.T. sur la Traduction Automatique en 1952, Conférence sur la T.A. à Moscou en 1958 avec 350 participants, Conférence sur le traitement de l'information à Paris en 1959) et l'apparition des premiers systèmes :

- système japonais YAMATO (1959)

- système de KULAGINA et MEL'CUK, en URSS (1956-1957), du français vers le russe. Le partage du processus de traduction en analyse du texte d'entrée et synthèse du texte de sortie préfigure les méthodes de deuxième génération. Beaucoup plus évolué que les réalisations de REIFLER, il fonctionne sur un ordinateur et effectue une traduction en 5 étapes à partir de 3 dictionnaires, français, russe et expressions idiomatiques.

- Le système GAT (Georgetown Automatic Translation)

D'autres travaux ont lieu pendant cette période, en Grande-Bretagne, en URSS, aux Etats-Unis et au Japon. Le structuralisme américain asémantique permet alors de concevoir la possibilité d'une traduction par substitution de morphèmes.

#### Le tournant :

Le lancement par l'Union Soviétique du premier Spoutnik en octobre 1957 déclencha une vive réaction des américains qui mobilisèrent leurs efforts afin d'être les premiers à envoyer un homme sur la lune. La nécessité d'analyser les résultats et l'orientation de la technologie soviétique a entraîné une intense demande de traduction du russe vers l'anglais.

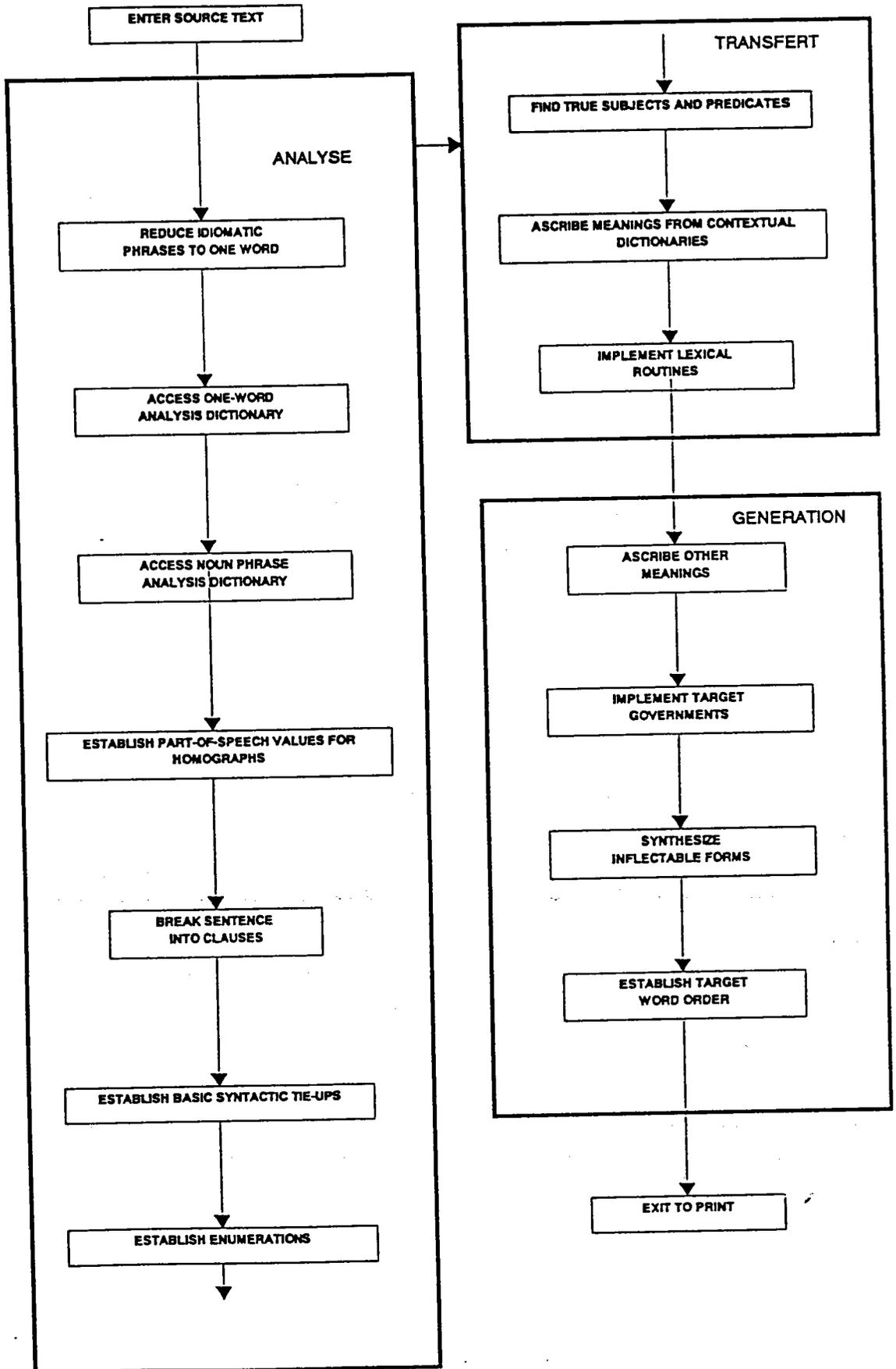
Thème de recherche prioritaire, la Traduction Automatique engloutit des sommes considérables et occupe plus de 40 organismes dans les années 1960. C'est à cette même époque qu'est créé à Grenoble, le CETA (Centre d'Etudes pour la T.A.).

Les espoirs entretenus en dépit de l'insuffisance des théories linguistiques et de la méconnaissance des capacités de l'ordinateur ont vite été déçus. En l'absence de résultats, le gouvernement américain chargea un comité d'évaluer les recherches en Traduction Automatique. Publié en 1964, le rapport ALPAC n'accorda aucun avenir technique ni économique au domaine. Les subventions supprimées, presque toutes les recherches cessèrent aux Etats-Unis.

#### 1965-1970 :

Cette période fut difficile pour les autres pays. Certains chercheurs préparaient la seconde génération, que le rapport ALPAC ne concernait pas. Après "Les structures syntaxiques" de N. CHOMSKY (1958), la théorie des grammaires de cas proposée par C. FILLMORE (1958) constitue, avec les études de D.G. HAYS sur la théorie des constituants et sur celle des dépendances pour le calcul des structures syntaxiques, un nouveau progrès du point de vue de la T.A.. Les résultats acquis à l'université de Georgetown permirent de construire un système présenté en 1970 : SYSTRAN

Le schéma de la page suivante illustre la séquence des opérations qui se divise en trois étapes : analyse, transfert et génération.



1970-1980

SYSTRAN fut presque le seul système américain, le rapport ALPAC ayant détourné les efforts de recherche vers la linguistique informatique, les théories du langage et l'intelligence artificielle.

T. WINOGRAD (1.4.3.5.4) montre expérimentalement que la compréhension du langage ne procède pas seulement de la connaissance des lois internes mais du domaine où il est utilisé, ce qui nécessite des informations et des connaissances sur ce qui lui est extérieur. Son robot est dirigé par l'intermédiaire d'un ordinateur, en langue naturelle, ici, en anglais. Les robots de type winogradien ont pour caractéristique essentielle la faculté de déduction logique et le pouvoir d'inférence (théories de Y. WILKS ou de R. SCHANK 1.4.3.3.4). Ils fonctionnent dans un univers limité et possèdent un savoir emmagasiné sous forme de dictionnaires et de relations.

L'ordinateur ne peut accéder à une telle compréhension que dans des domaines étroitement limités. (J. PITRAT et ses travaux sur le jeu des échecs).

C'est la première leçon tirée après les années 60. La seconde leçon sera la nécessité de construire des systèmes où toute amélioration et toute évolution puissent être aisément mises en oeuvre, ce qui implique une séparation : les grammaires et dictionnaires d'une part, le logiciel responsable de l'analyse et de la traduction d'autre part.

L'université de Sarrebruck développe le système SUZY à partir de 1974 et l'Institut Textile de France met au point le système TITUS appliqué à un texte source rédigé dans un langage "à syntaxe contrôlée".

Deux réalisations concrétiseront ensuite les progrès effectués à partir des conclusions citées plus haut :

- pour la première, le système canadien TAUM METEO (1977)
- pour la seconde, le système ARIANE 78 (1978).

Troisième et quatrième génération

Si l'on considère les différents niveaux de traduction, on peut estimer que les systèmes actuels correspondent au niveau de l'analyse syntaxique qui consiste à traduire une phrase en ayant identifié sa structure grammaticale.

- le niveau supérieur consiste à analyser non seulement la structure grammaticale de la phrase mais aussi son contenu sémantique. On s'intéresse au sens des mots et à leurs relations dans la phrase. Fondée sur le concept de grammaire de cas proposé par C. FILLMORE, cette approche nous mène à des systèmes de traduction automatique de troisième génération, comme celui que développe l'université de KYOTO : le "Lexicon Driven Translation System" fait appel à des règles propres à chaque mot, ce qui a conduit à construire des dictionnaires en trois niveaux :

- noyau : avec mots ordinaires 20.000 à 30.000 mots
- mots spécifiques au domaine de la traduction avec leurs règles d'utilisation
- dictionnaire personnalisé où l'utilisateur accumule ce qui lui est nécessaire.

Le dernier niveau est atteint lorsqu'il s'agit de tenir compte du contexte. Aucun système possédant une faculté de raisonnement et centrant l'analyse du langage sur le contexte n'a encore vu le jour. Un tel objectif correspond cependant au projet japonais de l'ordinateur de cinquième génération, qui n'aboutira pas avant 1998 ans si tout se déroule comme prévu.

On peut se demander s'il n'existe pas, au delà de ces niveaux de profondeur, un dernier stade qui permettrait d'obtenir une description interne indépendante de la langue. Ce langage pivot, dans une optique multilingue, permettrait de réduire le nombre des procédures de transfert d'une langue à l'autre. L'élaboration d'une grammaire du langage pivot par MONTAGUE constitue une démarche intéressante mais ses représentations ne sont pas assez précises pour décrire les phénomènes du langage avec la finesse que requiert la traduction automatique.

### Présent

Les pas de géant de la technologie ont sérieusement entamé le crédit du rapport ALPAC. De plus, la quantité de documents à traduire croît sans cesse. Ces deux points laissent entrevoir une extension du domaine dans trois directions, avec toutefois la certitude de ne pouvoir espérer, désormais, de traduction automatique parfaite à 100%.

- Traduction Automatique sur gros systèmes (révision humaine indispensable), utilisés sur site ou accessibles au grand public par télématique.
- Avant que les recherches en Traduction Automatique et en linguistique ne conduisent à la mise au point des systèmes de troisième ou quatrième génération, et nous pensons qu'il faudra encore attendre longtemps, les laboratoires développeront des systèmes mixtes de traduction assistée, sur mini- ou microsystèmes, loués par l'utilisateur et autorisant le développement de dictionnaires propres : *Traduction par ordinateur assistée par l'homme* (HAMT Human Aided Machine Translation) avec le système interactif ALPS et le système WEIDNER et *Traduction humaine assistée par ordinateur* (MAHT Machine Aided Human Translation) la plus répandue actuellement.
- Station de traduction, intégrant tous les outils d'aide à la traduction, d'édition, de mise en forme et de communication.

**Nous avons choisi une quatrième voie qui consiste à développer non pas un outil de traduction automatique ou partiellement automatique mais un automate capable de traiter un texte technique ou scientifique très rapidement et d'en livrer une interprétation en français dans le but :**

- de permettre à un lecteur qui ne connaît pas la langue du texte source, d'en apprécier le contenu
- de lui fournir à moindre coût les éléments qui permettront de juger de l'intérêt d'une véritable traduction.

Il convient de préciser qu'il n'existe pas de coupure nette entre la MAHT et notre automate qui accomplit tout de même une traduction dans le sens ou des formes de la langue source sont mises en correspondance avec des formes de la langue cible. On pourrait y voir une forme limite de la MAHT, l'ordinateur fournissant le minimum de transfert à partir duquel le lecteur compétent "se fera une idée" de ce dont il s'agit.

Avant d'aborder les techniques que nous avons retenues et qui font l'objet du chapitre III, nous présenterons les systèmes de TA et de TAO, avec les systèmes de base (2.3.2), les systèmes actuels (2.3.3), les nombreux projets (2.3.4, 2.3.5), des grands services de traduction (2.4) et pour être complet, les banques de données terminologiques (2.5).

### 2.3.2 Les systèmes de base

Nous citons ces systèmes anciens pour illustrer la continuité des recherches et rappeler que certains d'entre eux sont à l'origine des nombreux développements que nous aurons l'occasion de présenter plus loin.

#### 2.3.2.1 GAT (Georgetown Automatic Translation)

Le système GAT a été développé à l'université de Georgetown à partir de 1952 et été subventionné par le gouvernement américain. En collaboration avec IBM (1954), L. DOSTERT (anglais-russe) et BROWN (français-anglais) ont livré une version opérationnelle (1964-1979) à l'Atomic Energy Commission at Oak Ridge National Laboratory (ORNL) puis à EURATOM à Ispra (Italie) où il a été utilisé jusqu'en 1976.

Ces deux versions ont été utilisées de nombreuses années pour traduire des textes de physique du russe vers l'anglais<sup>1</sup>. La qualité de la traduction était très mauvaise, comparée aux traductions humaines.

Afin de passer très rapidement des documents en revue pour en déterminer le contenu et l'intérêt, le GAT était néanmoins supérieur à l'alternative : traduction lente et plus coûteuse/Pas de traduction.

Dans ce système "direct" et "local" le remplacement mot à mot précède des transpositions de mots en quantité limitée pour aboutir à quelque chose qui ressemble vaguement à de l'anglais, un mot étant défini comme un simple mot ou une suite de mots formant un "idiome". Le système ne reposait pas véritablement sur une base linguistique. Développé pour traiter un texte donné, il fallait l'adapter au texte suivant et ainsi de suite. Ces contraintes ont abouti à un système monolithique d'une complexité inextricable qui n'a pas subi de grosses modifications après avoir été livré à l'ORNL et à EURATOM. La longévité n'est pas ici une preuve de sa qualité. Elle souligne bien plus le besoin énorme des bureaux de traduction.

Le projet GAT disparaîtra en 1960 avec son incorporation dans la Société LATSEC (Peter TOMA, un des créateurs du GAT) qui développera alors le système SYSTRAN (basé sur la même technologie).

#### 2.3.2.2 CETA (Centre d'Etudes pour la Traduction Automatique)

En 1961 l'université de Grenoble lance le projet de traduction du russe vers le français. A la différence du GAT, les chercheurs de Grenoble construisent leur programme autour d'une théorie linguistique précise, avec l'intention de conduire une analyse en structures de dépendances pour chaque phrase (approche globale) plutôt que de s'appuyer sur une heuristique intérieure à la phrase et aboutir à un contrôle limité du transfert (approche locale).

L'approche est inter-langue au niveau grammatical (utilisation d'un langage indépendant pour représenter le sens de façon neutre). La technique de transfert s'applique au niveau lexical (dictionnaire) et implique une structure qui permet de passer d'une représentation du sens à une autre.

Les techniques informatiques ne sont pas très évoluées à l'époque. Le centre adopte le langage Assembleur d'IBM<sup>2</sup>.

(1) S.R. JORDAN, A.F.R. BROWN, F.C. HUTTON : "Computerized Russian Translation at ORNL" in *Proceedings of the ASIS Annual Meeting*, San Francisco, 1976, p. 163

(2) W. J. HUTCHINS : "Progress in Documentation : Machine Translation and Machine-Aided Translation" in *Journal of Documentation* n° 34, June 1978, pp. 119-159

Le développement du système dura 10 ans, de 1961 à 1971. Il fut utilisé de 1967 à 1971 pour traduire 400 000 mots de textes sur les mathématiques et la physique, du russe vers le français. La découverte majeure concerne l'utilisation d'un langage pivot. Le concept sera abandonné plus tard (système ARIANE 78) lorsqu'un changement d'environnement (matériel informatique) aura stoppé le développement du CETA, immédiatement remplacé par un nouveau projet, le GETA (Groupe d'Etudes pour la Traduction Automatique).

### 2.3.2.3 METAL (MEchanical Translation and Analysis of Languages)<sup>1</sup>

Les recherches en Traduction Automatique ont débuté en 1956 à l'université du Texas (Austin). Le LRC (Linguistics Research Center), créé en 1961, a lancé le projet METAL sur le couple de langues allemand-anglais. Il s'appuie sur la grammaire transformationnelle de CHOMSKY tout en admettant ses insuffisances pour aboutir à un système opérationnel. La traduction "indirecte" est effectuée en 14 étapes. Le temps de calcul est très important (entrées/sorties dans des fichiers de données trop importants). Les premiers résultats sont obtenus en 1974 à l'aide d'un programme de 80 000 lignes de FORTRAN fonctionnant sur une machine dédiée (CDC 6600).

Interrompues pendant quelques années, les recherches reprennent grâce à de nouveaux crédits du gouvernement américain. Le programme est réécrit en LISP et fonctionne sur un interpréteur DEC-10. Des améliorations considérables amènent alors le département des langues de SIEMENS AG (Münich) à sponsoriser de nouveaux travaux à partir de 1980.

### 2.3.2.4 TAUM (Traduction Automatique de l'Université de Montréal)<sup>2</sup>

L'université de Montréal et le gouvernement canadien lancent le projet TAUM en 1965. Ayant retenu l'approche de "transfert", le système repose sur un ensemble de programmes écrits en FORTRAN et fonctionnant sur un CDC 6600 puis sur un CYBER 173. Après une période de recherche et de mise au point, des buts lui sont assignés et donnent lieu à deux projets précis :

- TAUM-METEO pour le Canadian Meteorological Center, en 1975 doit traduire automatiquement de l'anglais vers le français les bulletins de prévision météorologique. Le système sera présenté en 1976 et installé en 1977.
- TAUM-AVIATION pour traduire automatiquement de l'anglais vers le français un ensemble de manuels de maintenance en aéronautique (90 millions de mots). L'introduction d'une composante sémantique en 1977 ne comble pas les lacunes d'un système que le gouvernement canadien soumet à de nombreux tests à partir de 1979. Le coût de fabrication des dictionnaires et le coût de la traduction automatique (6 cents/mot + 10 cents/mot de postédition) par rapport à la traduction humaine (8 cents/mot + 4 cents/mot de postédition)<sup>3</sup> condamne le projet en 1981.

### 2.3.2.5 ALP (Automated Language Processing)

Le projet de traduire automatiquement des textes mormons de l'anglais vers d'autres langues parmi lesquelles le français, l'allemand, le portugais et l'espagnol, a été lancé en 1971 à la Brigham Young University.

(1) W.S. BENNETT, J. SLOCUM : "The LRC Machine Translation System" in *Computational Linguistics* 11(2-3), 1985, pp. 111-121

(2) P. ISABELLE : "Machine Translation at the TAUM Group" presented at the *ISSCO Tutorial on Machine Translation*, Lugano, avril 1984

(3) A. GERVAIS : "Regular Use of Machine Translation of Russian at Oak Ridge National Laboratory", *AJCL* 13, 2, 1976, microfiche 46, pp. 53-56

Le système visait une traduction automatique. Il évolue vers un système de Traduction Assistée par Ordinateur (Interactive Translation System, ITS), à vision globale et approche de "transfert". Il ne verra jamais le jour (coût du matériel et difficultés de gérer l'interactivité avec le traducteur humain).

En 1980, pendant que les chercheurs travaillent au nouveau système ALPS (Automated Language Processing Systems) et poursuivent le développement de ITS, quelques programmeurs de la BYU rejoignent la Weidner Communications Corporation et participent à la construction du système automatique WEIDNER.

## 2.3.3 Présentation des systèmes actuels

### 2.3.3.1 Introduction

#### 2.3.3.1.1 Contexte

Les facteurs du développement de la Traduction Automatique et de la Traduction Assistée par Ordinateur sont à l'échelle mondiale.

- Les échanges internationaux ne cessent de s'intensifier et s'appliquent aux domaines les plus variés : industrie, commerce, politique, sciences, culture, formation, recherche...
- La concurrence acharnée que se livrent les entreprises pour vendre et exporter leurs produits les obligent à disposer le plus rapidement possible d'une traduction correcte des descriptifs et modes d'emploi. C'est à ce prix qu'elles peuvent conserver un léger avantage sur le marché.
- Les technologies de l'électronique envahissent notre société et lui imposent une informatisation croissante.

#### 2.3.3.1.2 Les utilisateurs

Les besoins que nous évoquons au paragraphe 2.2.1 se répartissent sur quatre niveaux :

- **Les états** : la demande croît au fil de l'intégration européenne. La Communauté ne pourrait pas faire face au volume des traductions générées par ses 9 langues de travail sans l'aide du traitement automatique. En 1986, le Parlement Européen a consacré la moitié de son budget aux traductions.
- **Les grands organismes** (ONU, OTAN, EURATOM, NASA, US AIR FORCE, Pan American Health Organization...), les entreprises internationales (Xerox, IBM, General Motors, Nixdorf, Triumph-Adler, Hewlett-Packard, ComputerVision, Control Data...).
- **Les petites et moyennes entreprises, les établissements d'enseignement et de recherche, les cabinets de traduction.**
- **Les particuliers** (exemple : traduction via le Minitel).

#### 2.3.3.1.3 Les problèmes spécifiques

Bien qu'ils soient nombreux, nous ne citerons que les ambiguïtés des relations syntaxiques et sémantiques (polysémie, homographie, synonymie, interprétation de la phrase, pronominalisation, topicalisation...) pour souligner qu'une partie seulement des problèmes ont été formalisés et résolus. L'insuffisance des modèles linguistiques et la nature de l'outil informatique limiteront sans doute très longtemps les capacités des

systèmes de Traduction qui ne seront jamais en mesure de remplacer le traducteur humain pour des prestations de qualité. Ce sont des outils très précieux, cependant, pour traduire rapidement des textes techniques et scientifiques, moins complexes que les textes littéraires. Leur utilisation est de plus en plus fréquente lorsqu'il s'agit de documents dont le seul but est d'informer. La qualité de la traduction est en effet suffisante pour les lecteurs qui sont en général des spécialistes du domaine traité. Reprenons des exemples présentés par la société B'VITAL<sup>1</sup> (2.3.3.8.2), à Paris, en mars 1988, lors du séminaire international "Traduction Assistée par Ordinateur". Ces exemples sont tirés de textes réels et sont des traductions brutes (non révisées) obtenues par le système français-anglais traitant des documents techniques dans le domaine de l'aéronautique. Ils illustrent les possibilités de paraphrasage (grâce à la notion d'unité lexicale ou de famille dérivationnelle), les diverses possibilités de traduction des prépositions, en fonction du contexte (critères syntaxiques et/ou sémantiques, suivant les cas) et les choix de traduction des mots en fonction du contexte (critère syntaxique et sémantique, en général).

*français : Après essai, s'assurer du fonctionnement correct de l'ensemble raccord.*

*anglais : After test, check that the coupling assembly works correctly.*

On observe le passage d'une phrase nominale "du fonctionnement correct" à une phrase verbale "that ... works correctly" et le passage d'un adjectif "correct" à un adverbe "correctly".

*français : le remplissage des pompes s'effectue dans un local fermé, très propre, à l'abri de toute possibilité d'introduction de poussière dans la graisse.*

*anglais : The pump filling is carried out in a very clean closed room (premises), away from any dust introduction possibility ((opportunity)) into grease.*

On note l'interprétation de la forme réfléchie "s'effectue" comme un passif et les traductions différentes de la proposition "dans", en fonction du contexte (traduite par "in" ou "into"), grâce à des critères sémantiques.

*français : Porter sur celle-ci la date de la dernière réception ou révision.*

*anglais : Write on this one the date of the last reception or of service.*

Le verbe "porter" a été traduit par "to write" en fonction du contexte (i.e. en fonction de la sémantique de l'objet "date" : critère syntaxique et sémantique).

*français : Effectuer la vidange générale et la purge de carburant (voir chapitre 12).*

*anglais : Drain in a general manner and bleed fuel (see chapter 12).*

"effectuer la vidange" a été traduit par le verbe seul "to drain" grâce à la notion d'unité lexicale.

Parmi les nombreux systèmes qui feront l'objet de ce chapitre, citons encore TITUS V (2.3.3.3.9).

*Phrase d'entrée : Les progrès récents des logiciels appropriés au traitement on-line des systèmes d'information ont favorisé le développement des bases de données dans tous les domaines des sciences et des techniques.*

*anglais : The recent progress of the software suitable for the on-line treatment of information systems has favored the development of data bases in all the fields of sciences and techniques.*

---

(1) Cette société est issue du milieu universitaire grenoblois. Elle a été créée en 1985 avec l'appui du GETA (Groupe d'Études pour la Traduction Automatique) dans le but de faire passer la Traduction Assistée par Ordinateur au niveau industriel.

*allemand : Die neuen Fortschritte der für die on-line verarbeitung der informations-systeme geeigneten software haben die entwicklung der databases in allen bereichen der wissenschaft und der techniken begünstigt.*

*espagnol : Los progresos recientes del software apropiado para el tratamiendo on-line de los sistemas de documentacion favorecieron el desarrollo de las bases de datos en todos los campos de las ciencias y de las tecnicas.*

#### 2.3.3.1.4 Les critères de choix

Les machines manquent d'intuition et rien ne permet de croire, pour l'instant, qu'elles en auront un jour suffisamment. Il convient donc de ne recourir à la Traduction Automatique ou à ses dérivés qu'en connaissance de cause. L'analyse des coûts d'utilisation sous-tend une comparaison de la traduction automatique et de la traduction manuelle en tenant compte, dans les deux cas, des coûts de révision, des méthodes et des outils utilisés<sup>1</sup>. Les points essentiels sont :

##### Coûts :

- Acquisition, installation du matériel et du logiciel.
- Entretien et maintenance
- Formation des utilisateurs
- Pour les systèmes qui ne sont pas complètement automatisés, travaux de pré-édition, post-édition, communication homme-machine en mode interactif.
- Construction et mise à jour des dictionnaires
- Amortissements

##### Performances :

- Evaluation des temps requis pour la saisie du texte, le traitement automatique et la révision.
- Qualité de la traduction
- Possibilités d'améliorer les performances du système en étendant le domaine d'application (Types de texte) ou en introduisant de nouveaux couples de langues.
- Convivialité

Nous passerons en revue les systèmes actuels en distinguant les systèmes commercialisés, les systèmes dédiés à des tâches spécifiques et les systèmes d'aide à la traduction.

#### 2.3.3.2 Les systèmes commercialisés actuellement

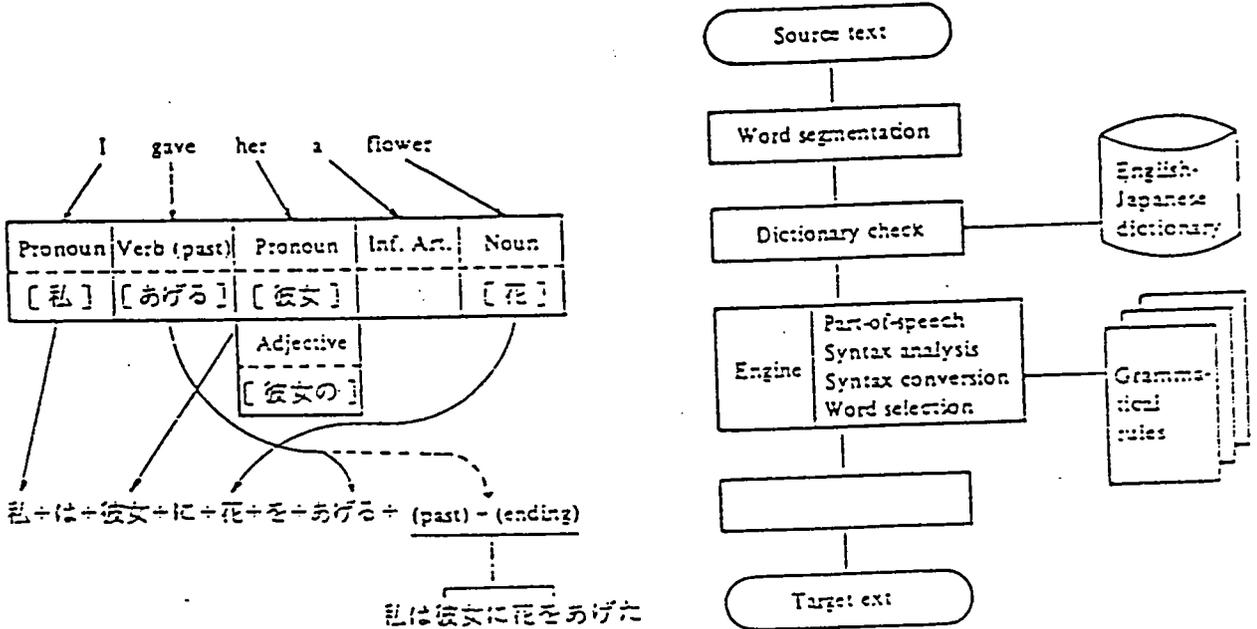
##### 2.3.3.2.1 ATLAS I et ATLAS II<sup>2</sup>

Premier système industriel de traduction (anglais-japonais) conçu par FUJITSU et disponible depuis septembre 1984, il compte à ce jour une bonne centaine d'utilisateurs. Ce système de "première génération" repose sur une approche syntaxique : la phrase est traduite mot à mot puis recomposée dans la langue cible grâce à un ensemble de règles syntaxiques. La résolution des ambiguïtés est abordée en fin de traitement au moyen d'une analyse sémantique très sommaire. Les résultats sont assez mauvais. L'intervention d'un opérateur est de plus indispensable pour la pré- et la postédition.

(1) G. VAN SLYPE : "Conception d'une méthodologie générale d'évaluation de la traduction automatique" in : *Multilingua*, Mouton, 1-4/1982

(2) H. UCHIDA, T. HAYASHI, K. KUSHIMA : "ATLAS : Automatic Translation System" in *Fujitsu Science & Technology Journal*, n°21, 3/1985, pp. 317-329

Le schéma suivant illustre son fonctionnement<sup>1</sup> :



ATLAS I tourne sur un ordinateur FACOM ou un S-3000 Fujitsu Minicomputer, dans un environnement OSIV (système d'exploitation). Le logiciel est rédigé en PL/1 et en Assembleur. La traduction est directe, s'appuie sur la syntaxe et les grammaires syntagmatiques (1.4.3.2.4.3) Pour la traduction anglais-japonais des textes techniques ou scientifiques, il dispose d'un dictionnaire de 53000 mots et de 25000 termes techniques. Sa vitesse atteint 60000 mots/seconde.

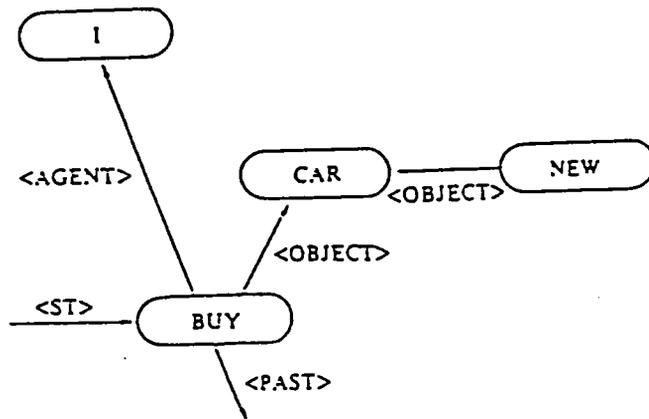
ATLAS II : Une "deuxième génération" de systèmes est apparue, basée sur une approche sémantique : les phrases sont analysées en fonction de leur sens grâce à "des modèles du langage" et des "modèles du monde". Représentées dans un "langage pivot" sous forme de réseaux sémantiques, structures conceptuelles indépendantes de la langue source, elles sont réécrites dans la langue cible.

Le schéma de la page suivante visualise la structure conceptuelle de la phrase :

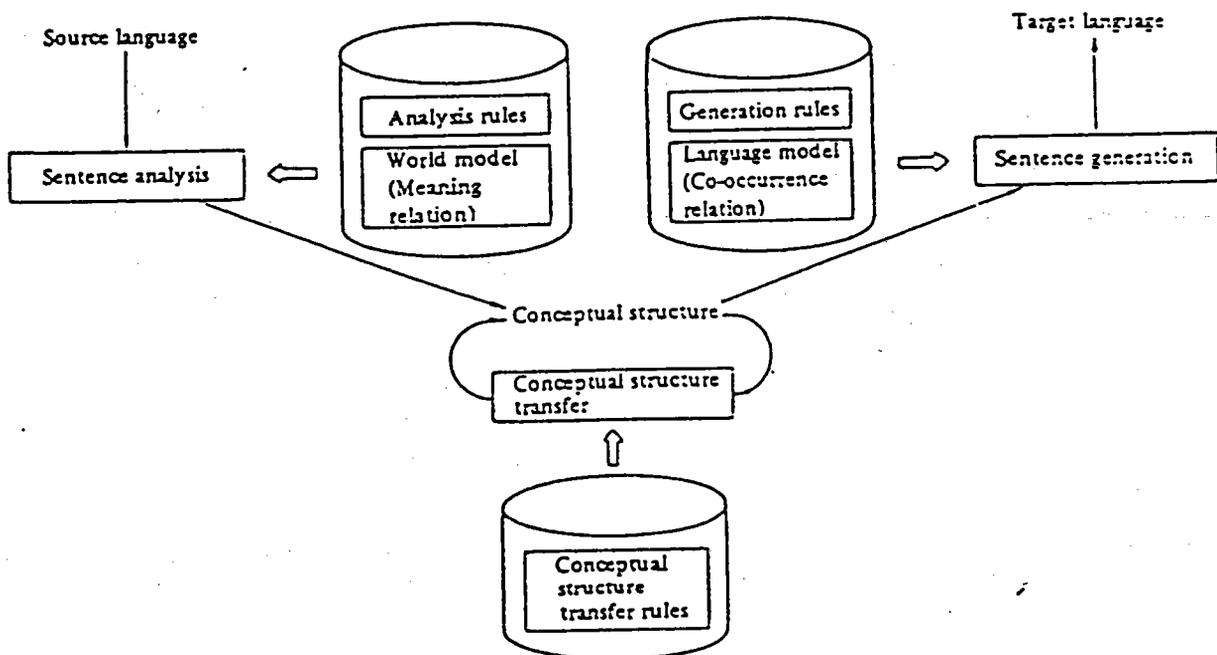
*I bought a new car.*

Le réseau est constitué de noeuds et d'arcs. Les noeuds représentent le concept qui correspond au sens des mots I, BUY, CAR et NEW, les arcs des relations telles que <AGENT>, <OBJET> ou telles que <CAUSE> et <SEQUENCE>. A ces arcs binaires s'ajoutent des arcs unaires qui complètent les informations sur le temps, l'aspect et le style.

(1) H. UCHIDA : "Fujitsu Machine Translation System : ATLAS" in : *Proceedings of the International Symposium on Machine Translation*, 14 octobre 1985, Japan Information Processing Development Center, pp. 29-37



ATLAS II appartient à ces nouveaux systèmes et traduit du japonais vers l'anglais depuis fin 1985. Diffusé à une trentaine d'exemplaires, il fonctionne sur un ordinateur FACOM, est écrit en langage C, utilise la technique de "langage pivot" et traduit 60000 mots à la seconde. Les résultats sont loin d'être parfaits en raison du niveau des modèles d'analyse sémantique. Comme pour ATLAS I, la traduction n'est pas totalement automatique, le processus nécessite l'intervention d'un opérateur pour des phases de pré- et post-édition.



Si l'on examine le processus complet dans le détail, on distingue les trois étapes classiques : l'analyse, le transfert et la génération.

**L'analyse** de la phrase source doit aboutir à la représentation de son sens en une structure conceptuelle, sous la forme d'un réseau sémantique. Elle correspond à deux modules :

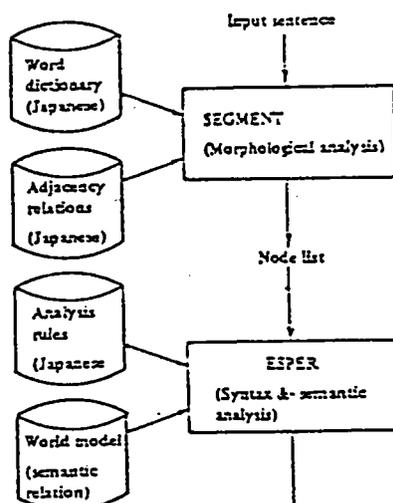
- SEGMENT (analyse morphologique) décompose les mots en "morphèmes" à partir de dictionnaires de mots et de relations. Les "morphèmes" sont rangés dans une liste de noeuds.

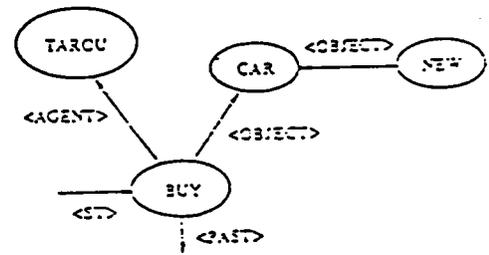
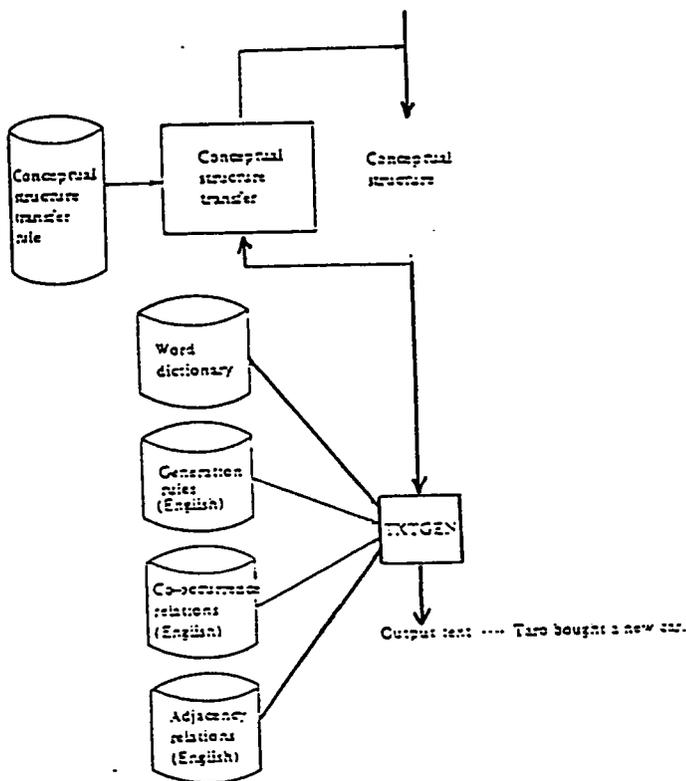
- ESPER (analyse syntaxique et analyse sémantique) parcourt la liste des noeuds en traitant chaque morphème comme un noeud terminal. Chaque noeud reçoit des informations grammaticales (ensemble d'attributs grammaticaux) et sémantiques du dictionnaire des mots.

L'analyse syntaxique et l'analyse sémantique sont parallèles et s'appuient sur des règles de grammaire indépendante du contexte. L'analyse sémantique associe des marques sémantiques au nouveau noeud et détermine la relation entre deux noeuds. On aboutit à la structure conceptuelle de la phrase source, qui est vérifiée par référence au modèle du monde.

**Le transfert** qui permet le passage de la langue source à la langue cible appréhende les différences non pas en ce qui concerne les mots ou la grammaire mais les concepts et la pensée. Il les traite au niveau de la représentation intermédiaire.

**La génération** produit le texte cible à partir de la structure conceptuelle représentée par un réseau sémantique. Cette structure subit une conversion linéaire qui la remplace par une chaîne de mots. Certaines transformations deviennent inutiles, ce qui permet non seulement au mécanisme de génération mais aux règles d'être indépendants du langage. Le système de génération réunit une "fenêtre" de génération, une liste de sortie et un interpréteur de règles. Ce dernier parcourt chaque noeud de la structure conceptuelle en déplaçant la "fenêtre" de génération qui teste les noeuds et les arcs, et retourne la liste de sortie contenant les résultats de la traduction. Les mots y sont stockés dans l'ordre de la génération qui correspond à l'ordre des mots des structures de surface. L'interpréteur de règles interprète chaque règle, parcourt chaque noeud en déplaçant la "fenêtre" de génération et sélectionne les mots en vérifiant les relations de co-occurrence et de contiguïté. Il ajoute chaque mot sélectionné à la liste de sortie.





### 2.3.3.2.2 HICATS/JE (Hitachi Computer Aided Translation System Ja-En)<sup>1</sup>

Lancé sur le marché en 1987 par la firme Hitachi, le système traduit du japonais vers l'anglais. Une version anglais-japonais est en préparation. Il applique une grammaire de dépendance pour l'analyse sémantique et une grammaire syntagmatique pour la synthèse. Fonctionnant sur un ordinateur Hitachi M, sous VOS3, il est rédigé en PL/I et en GDL. La pré- et post-édition sont interactives. La traduction japonais-anglais s'applique à des textes techniques ou scientifiques, à des catalogues et à des brevets. Sa vitesse atteint 60000 mots/seconde. L'utilisateur peut établir son propre dictionnaire.

### 2.3.3.2.3 LOGOS<sup>2,3,4</sup>

La société LOGOS a été créée en 1969 (B. E. SCOTT) aux Etats-Unis (Waltham, Massachusetts) pour effectuer des recherches en linguistique automatique et développer des systèmes de Traduction Automatique. De 1970 à 1972 elle met au point pour l'US Air Force un système fonctionnant sur le couple anglais-vietnamien qui permettra de traduire plus de 5 millions de mots. Elle travaille ensuite sur les couples anglais-russe et anglais-parsi. A partir de 1981, l'arrivée de nouveaux investisseurs oriente les recherches vers des produits commercialisables. En 1982, SIEMENS finance les couples anglais-allemand et allemand-anglais.

(1) HITACHI : "Research on Machine Translation in Hitachi Ltd", *Technical note*, Systems Development Laboratory, Hitachi Ltd, Tamaku, Kawasaki, 19981

(2) C.O. STAPLES : "The LOGOS Intelligent Translation System", Presented at the *Joint Conference on Artificial Intelligence*, Karlsruhe, août 1983

(3) P. WHEELER (LOGOS Computer Systems Deutschland GmbH, Frankfurt).

(4) Documents fournis par N. DROUIN, Directeur et Vice Président exécutif de LOGOS Canada

Depuis 1984, LOGOS est implanté aux Etats-Unis (Mount Airlington, Boston), en Europe (Francfort, Zurich) et au Canada (Montréal, Ottawa).

L'ensemble de logiciels proposés par LOGOS pour les couples anglais-français, anglais-allemand, anglais-espagnol, allemand-anglais et allemand-français se compose de trois parties :

**THE INTELLIGENT TRANSLATOR** : Ce module de traduction reprend les trois étapes classiques de l'analyse, du transfert et de la génération.

**ALEX (Automatic Lexicographer)** : Option "obligatoire", ce programme permet d'introduire des données dans le dictionnaire de l'utilisateur (mais en aucun cas dans le dictionnaire de LOGOS qui est intégré au système de base). L'utilisateur construit lui-même son dictionnaire, spécifique à son organisation et aux domaines qu'il traite.

LOGOS ÜBERSETZUNGSSYSTEM  
Wörterbuchfunktionen: ALEX

Grundform des deutschen Wortes eintragen Maschinenführung		Sachgebiets-Code 000 GENERAL USAGZ
Wortart	Eintragsart	Genus
- Substantiv	- Wort	- Maskulinum
- Adjektiv	- Abkürzung/ Akronym	- Femininum
- Adverb		- Neutrum
Wenn ein Kompositum ist, Grundwort angeben Maschinenführung		
Englischen Transfer eintragen: machine guide		Falls vorhanden, die abgeleitete Adjektivform eintragen

LOGOS ÜBERSETZUNGSSYSTEM  
Wörterbuchfunktionen: ALEX

Grundform des deutschen Wortes eintragen Maschinenführung		Sachgebiets-Code 140 Engineering
Wortart	Eintragsart	Genus
<input checked="" type="checkbox"/> Substantiv	<input checked="" type="checkbox"/> Wort	- Maskulinum
- Adjektiv	- Abkürzung/ Akronym	<input checked="" type="checkbox"/> Femininum
- Adverb		- Neutrum
Wenn ein Kompositum ist, Grundwort angeben führung		
Englischen Transfer eintragen: machine guide		Falls vorhanden, die abgeleitete Adjektivform eintragen

**SEMANTHA (Semantic Table)** : Cet utilitaire vendu en option et fonctionnant en mode interactif permet à l'utilisateur de rentrer dans le programme de nouvelles règles sémantiques. Il stocke un ensemble de règles conceptuelles destinées essentiellement à repérer les constituants des formes verbales pour restituer les nuances au niveau du langage de transfert (langue cible).

Deux points caractérisent l'approche des concepteurs de LOGOS :

- Les informations concernant la langue source et la langue cible sont rangées dans des fichiers distincts, ce qui a permis d'étendre le nombre des couples de langues sans réécrire à chaque fois l'ensemble du système.

- La sémantique est intégrée dans l'algorithme de traduction au moyen d'un langage de type sémantico-syntaxique, le SAL (Semantic Abstraction Language), de structure arborescente. Le programme traduit la chaîne source en SAL avant de l'analyser. La continuité est ainsi assurée entre la sémantique et les traitements lexical et syntaxique.

L'analyse est descendante, de gauche à droite. Elle ne s'applique pas à une chaîne en langue naturelle ni à une représentation symbolique de cette chaîne, mais à une chaîne exprimée en SAL et riche en informations de nature sémantique. Ceci permet de réduire la complexité du traitement du sens, particulièrement redoutable lorsqu'il s'agit de la langue naturelle. Il est difficile de caractériser l'approche retenue par LOGOS qui, à notre sens, tient des systèmes à transfert (sa grammaire est un ensemble de règles de transfert) et des systèmes à "langage pivot" (la traduction initiale du texte source en langage SAL peut être assimilée à une technique "inter-langues").

Le processus de traduction peut être décomposé en neuf étapes. Nous allons les résumer et les illustrer par quelques exemples empruntés à P. WHEELER (LOGOS Computer Systems Deutschland GmbH, Frankfurt).

**Etape 1 :** Le texte source est mis sous une forme telle qu'il puisse être traité automatiquement. Les marques de formatage sont conservées pour être réutilisées dans la dernière étape. Voici un exemple qui montre le texte original en allemand et sa traduction anglaise dans le même format :

TECHNISCHER BERICHT    Berichts-Nr.    00/000-000  
zu Projekt Nr. 00 111 222

Verteiler:123 456/789 321 654 100 200 1/111 2/222 X11 Y22

"Bearbeitungszeiten für 888-XXX und 888-YYYY"

1.            Alte Maschinengeneration, konventionell gesteuert.

		Maschinen-Typ
Spitzendrehmaschine		0.000/00
XXX	Außendrehautomat	1.111
YYY	Innendrehautomat	2.222

2.            Neue Maschinengeneration, 888-gesteuert

		Maschinen-Typ
XXX	Außendrehautomat	3.333
YYY	Innendrehautomat	4.444
XXYY	Außen/Innendrehautomat	5.555



Prenons la phrase :

*Alle funktionsfähigen Großrechner und brauchbaren Bildschirme, die zur Verfügung stehen, werden heute nach Frankfurt geschickt.*

A la fin de l'étape, la phrase prend la forme suivante :

*Großrechner und Bildschirme, die zur Verfügung stehen, werden nach Frankfurt geschickt.*

Les adjectifs *alle*, *funktionsfähigen* et *brauchbaren* ont disparu. La méthode consiste à condenser le texte au maximum pour réduire la complexité du traitement. Les trois unités *alle*, *funktionsfähigen* et *Großrechner* ne sont plus considérées comme un groupe nominal de trois éléments mais comme un substantif. Le code qui correspond à *Großrechner* a été complété pour décrire les deux adjectifs qui ont disparu de l'écran. Le second résultat de cette étape est le remplacement des deux groupes nominaux par leur représentation dans la langue source. S'il était possible de visualiser la phrase à ce stade de l'analyse, elle aurait la forme suivante :

*Mainframes und screens, die zur Verfügung stehen, werden heute nach Frankfurt geschickt.*

Les concepts *all*, *operational* et *usable* qui se rapportent à *mainframes* et à *screens* sont encore sous forme de code. La consultation des tables sémantiques intervient avant le démarrage de la sixième étape. Si les tables donnent une nouvelle correspondance au terme *Großrechner* lorsqu'il est modifié par *alle* ou *funktionsfähig*, le code est modifié en conséquence.

**Etape 6 :** La "condensation" des groupes nominaux complexes est poursuivie aussi loin que possible. *Alle funktionsfähigen Großrechner und brauchbaren Bildschirme* a donné *Großrechner und Bildschirme* qui est à son tour réduit à *Großrechner* avec la modification du code que cela implique.

Cette étape traite les relatives et les complétives (ou assimilées, comme les chaînes entre parenthèses) en les confrontant d'abord avec la table sémantique. Une première règle indiquera qu'en combinaison, *zur* et *Verfügung* ne doivent pas être associés aux formes courantes du dictionnaire *to the availability* mais à *available* - classe des adjectifs. Une seconde règle indiquera que le verbe *stehen* associé à *zur Verfügung* ne doit pas être traduit par *stand* mais par *be*.

Après ce traitement sémantique, les parties de phrase sont condensées, les mots qui restent sont traduits dans la langue cible et la relative disparaît.

La phrase a maintenant la forme :

*Großrechner, heute nach Frankfurt werden geschickt.*

**Etape 7 :** le système traite les combinaisons préposition-substantif. Selon la méthode décrite précédemment, *nach Frankfurt* devient *Frankfurt*. Il lui est également possible de confronter des propositions complètes aux données des tables sémantiques. Pour la phrase citée en exemple, il pourrait trouver une règle précisant que *schicken* + objet (contexte informatique) ne se traduit pas par *send* mais par *transport* ou *transfer* selon le choix du traducteur.

**Etape 8 :** Les éléments de la phrase sont assemblés. L'examen du verbe permet de distinguer ce qui est sujet de ce qui est complément, quels éléments modifient le verbe... Les blocs sont réorganisés dans l'ordre de la langue cible. (*which available are* devient *which are available*).

**Etape 9** : Les codes SAL sont traduits en langue normale. Les formats stockés au cours de la première étape sont restitués et l'on obtient la sortie :

*All operational mainframes and usable screens which are available are transported today to Frankfurt.*

Il faut souligner que les règles sémantiques fournies avec le logiciel s'appliquent aux termes de base. Si l'utilisateur souhaite créer des associations spécifiques au domaine ou au texte qu'il traduit, il les introduira comme étant une nouvelle règle sémantique. Ces règles faisant intervenir obligatoirement des unités lexicales, il serait plus exact de parler de règles contextuelles.

Exemples<sup>1</sup> de traductions obtenues avec le système LOGOS :

#### Texte source

Allgemeine Nebenzeiten sind bezahlte Betriebspausen, Anlauf- und Auslaufzeiten.

Die Anlaufzeit ist der Zeitraum von der Inbetriebnahme des ersten Aggregates eines Betriebsmittels bis zum Ausstoß des ersten brauchbaren Produktes. Die Auslaufzeit ist der Zeitraum vom Ende der vollen Beaufschlagung des ersten Aggregates eines Betriebsmittels bis zur Außerbetriebnahme des letzten Aggregates eines Betriebsmittels.

Rüstzeit ist der zum Umstellen der Betriebsmittel auf entsprechende Formate, auf die Füllgutart und -mengen usw. benötigte Zeitraum. Dies geschieht überwiegend z. B. bei Formatwechsel durch Maschineneinstellung (Rüstzeit I) oder bei Präparate- bzw. Auftragswechsel (Rüstzeit II). Zur Rüstzeit gehört auch die Schmierung der Maschinenführungen.

#### Traduction brute

General non-operating times are paid breaks, start and shutdown times.

The start time is the period of/by the start-up of the first aggregate of a resource up to ejection of the first needable product. The shutdown time is the period of/by the end of the full admission of the first aggregate of a resource up to the taking out of operation of the last aggregate of a resource.

Set-up time is the one to convert the resources to corresponding formats, on the product type and mix period required etc. This occurs predominant e. g. during/upon format change via machine setting (set-up time I) or with preparation or change in order (set-up time II). The lubrication of the machine guides belongs at set-up time also.

(1) Extrait de P. WHEELER : "LOGOS" in *Sprache und Datenverarbeitung*, 9. Jahrgang 1985 Heft 1, 11/21, pp. 11-21

## Traduction révisée

General non-operating times are paid breaks, start and shutdown times.

The start time is the period from the start-up of the first unit of a machine up to discharge of the first usable product. The shutdown time is the period from the end of the full-scale working of the first unit of a machine up to the switching off of the last unit of a machine.

Set-up time is the period required to convert the machinery to corresponding formats, to the products types and quantities etc. This occurs predominantly e. g. at format changes via machine setting (set-up time I) or with changes in orders or preparations (set-up time II). Lubrication of the machine guides belongs to set-up time also.

Ci-dessous, un autre exemple<sup>1</sup> :

### Automatische deutsch-englische Übersetzung mit OIS von Wang

Logos Computer Systems, Inc. hat ein Übersetzungsprogramm mit Datenbank entwickelt, das dem Benutzer eines Wang Büro-Informationssystems maschinelle Übersetzung vom Deutschen ins Englische ermöglicht. Ein Übersetzer kann mit dem LOGOS Automatic Translator bis zu 10.000 Wörter pro Tag textgetreu übersetzen.

**Ein neuer Algorithmus.** Die Stärke des LOGOS Systems liegt in einem neuen Algorithmus, der fortschrittliche Methoden künstlicher Intelligenz zur Lösung syntaktischer und semantischer Probleme anwendet. Frühere Versuche waren durch das Prinzip ein Wort/eine Bedeutung – oft die falsche Bedeutung – zum Scheitern verurteilt. LOGOS berücksichtigt die Bedeutung des Wortes innerhalb eines Satzes. Das Ergebnis ist ein Übersetzungssystem, das allen bisherigen weit überlegen ist.

**Einzigartiger Systementwurf.** LOGOS ist ein von einzelnen Sprachen unabhängiges Übersetzungssystem. Jede Sprache ist in ausbaufähigen Tabellen gespeichert. Logos und die Übersetzer können die Sprachtabellen laufend erweitern und so kontinuierlich genauere Übersetzungen erstellen, ohne erneut programmieren zu müssen. Deutsch-Englisch ist die erste Sprachengruppe, die Logos herausgibt.

### German-English Automated Language Translation on the Wang OIS

With a Wang Office Information System (OIS) and a translation program available from Logos Computer Systems, Inc., it is now possible to automate the translation of German into English. The LOGOS Automatic Translator™ allows one person to translate up to 10,000 words per day with a high rate of accuracy.

**A Powerful Algorithm.** The power of the LOGOS software is in its new translation algorithm, which applies advanced techniques of artificial intelligence to the problems of syntax and semantics. The result is machine translation superior to any produced in the past, when systems were limited by the principle of one-word/one-meaning – often the wrong meaning. LOGOS is sensitive to context – how the meaning of a word is influenced by the words around it.

**Unique System Design.** The LOGOS software features a translating program that is independent of the languages involved. Language definitions reside in memory in separate expandable tables. Users can extend the language tables, increasing the accuracy of the translation without having to reprogram. German-to-English is the first set of tables released by Logos.

**Der Übersetzungsvorgang.** Bisher wurden zur elektronischen Sprachübersetzung immer große Computersysteme benötigt. Mit der speziellen Übersetzungseinrichtung von Wang (TRD) und der Übersetzungssoftware von Logos können elektronische Übersetzungen jetzt auf kostensparenden und für viele Zwecke geeigneten Text- und Datenverarbeitungssystemen ausgeführt werden.

Schließen Sie die Übersetzungseinrichtung einfach an Ihr Wang Büro-Informationssystem an, weisen Sie der Logos Übersetzungssoftware, die ein Lexikon mit 100.000 Wörtern enthält, 50 MB Plattenraum zu und überlassen Sie den Rest dem System.

**Eingabe.** Sie rufen an Ihrem OIS Arbeitsplatz das LOGOS Übersetzungsprogramm auf, wählen die zu übersetzenden Texte aus und drücken auf AUSFÜHREN. Die Dokumente werden in die Warteschlange für LOGOS eingereiht. Während LOGOS in dem im Hintergrund ablaufenden Stapelbetrieb übersetzt, können Sie auf Ihrem OIS System andere Arbeiten ausführen.

**Anwendungen.** Der Logos Automatic Translator läßt sich für eine Vielzahl von Bereichen, einschließlich wissenschaftlicher, technischer und kommerzieller Dokumente, einsetzen.

**Ausgabe.** Innerhalb von 24 Stunden können bis zu 10.000 Wörter übersetzt werden. Der Statusbericht zeigt an, welche Übersetzungen fertiggestellt sind. Der Übersetzer kann die einzelnen Dokumente nun ganz in englischer Sprache auf den Bildschirm rufen und Korrekturen in der Sprache oder Änderungen im Stil vornehmen. Er kann sie auch in einem Format auf dem Bildschirm anzeigen, in dem auf jeden deutschen Satz seine englische Übersetzung folgt. So können die beiden leicht verglichen werden.

**Vorteile.** Der LOGOS Automatic Translator vermindert nicht nur die Übersetzungszeit für den Benutzer, sondern ermöglicht auch ein neues Maß an Genauigkeit und Übereinstimmung innerhalb eines Textes und von einer Übersetzung zur nächsten. Da es an das Textverarbeitungssystem von Wang angeschlossen ist, ermöglicht LOGOS Übersetzungen in großem Umfang, und die Kosten bleiben im Rahmen. Die mechanische Arbeit wird vom System erledigt, und Beurteilung und schöpferische Leistung bleiben dem Übersetzer überlassen.

Das Wang OIS System. Die Familie der Büro-Informationssysteme (OIS) bietet integrierte Informationssysteme mit der bewährtesten, fortschrittlichsten und am einfachsten anzuwendenden Textverarbeitung, die Sie kaufen können: Wang Textverarbeitung. Über 100.000 Benutzer haben Wang zu einem der größten Lieferanten von Bildschirm-Textverarbeitungssystemen der Welt gemacht. Merkmale wie einfache Bedienung, Kompatibilität und Erweiterungsfähigkeit gewähren maximale Produktivität und Rentabilität.

Die vielseitigen OIS-Systeme verbinden Wang Textverarbeitung mit anderen fortschrittlichen Eigenschaften, um allen Informationsbedürfnissen Ihrer Firma gerecht zu werden. Ergänzungen wie Mailway\* (das elektronische Post- und Nachrichtensystem), WangNet (Breitband lokales Netzwerk) und WISE (Wang Inter-System Exchange), Wang's OIS Verbundsystem, ermöglichen erhöhte Leistungsfähigkeit und Produktivität im Büro.

Mit Wang ist Ihre Investition geschützt. Wang Systeme werden von einem weltweiten Netz von über 4000 Fachleuten unterstützt, die Wartung und Beratung für Wang Systeme bieten.

**How The Translation Facility Works.** Until now, electronic language translation has required the resources of large computer systems. Now, with the optional Wang Translation Device (TRD) and LOGOS software, electronic translation can be performed on cost-effective, multi-purpose Wang OIS systems, fully integrated with Wang's proven capabilities for text editing and data processing.

Simply attach the TRD to your OIS, allocate 50MB of disk space to the LOGOS programs, which include a 100,000 word dictionary, and let the system do the rest.

**Input.** At your OIS workstation, through a simple menu prompt, select "LOGOS" and press EXECUTE. Now select a document in German from your word processing system library. Press EXECUTE and it will enter the queue for translation. While LOGOS translates the document in a background batch mode, you can perform other tasks on your OIS.

**Applications.** This system can accommodate a wide variety of applications, including scientific, technical, and commercial documents.

**Output.** The Translation Facility can translate over 10,000 words in a 24-hour period. A status report will tell the user which documents have been translated.

The user can display the complete English translation on the screen to make any desired changes in style or usage, or the document can be brought to the screen in a format that compares each English sentence with its German counterpart.

**Benefits.** The LOGOS Automatic Translator not only decreases translation time for the user, but provides a new degree of accuracy and consistency. Its interface with Wang word processing now makes possible high volume translation on the Wang OIS while keeping costs manageable. Mechanical labor is done by the system, leaving matters of judgment and creation to the human translator.

Wang OIS. The Office Information Systems family are integrated information systems that feature the most proven, and easiest-to-use word processing you can buy: Wang word processing. Over 100,000 users have made Wang the world's number one supplier of video display word processing systems. Features like ease-of-use, compatibility, and expandability ensure maximum productivity and return on your investment.

The versatile Wang OIS systems combine Wang word processing with other advanced features to meet all the information processing needs of your company. Options such as Mailway\* electronic mail, WangNet broadband local area networking, and the Wang Inter-System Exchange (WISE), help to further increase office efficiency and productivity.

With Wang your investment is protected. Every system is backed by a worldwide network of over 4,000 people who service and support Wang systems.

En ce qui concerne le contexte d'utilisation, LOGOS tourne sur ordinateur IBM ou IBM compatible (système d'exploitation MVS et VM/CMS), sur WANG (système VS), sur WANG OIS140 Wordprocessor et sur UNISYS (système UNIX). Les documents techniques fournis par la firme recommandent une mémoire principale de 1 MB et des capacités de 130 MB par couple de langues. Les logiciels sont écrits en VS Fortran et en Assembleur. Il traduit les textes techniques à une vitesse de 20000 mots/24 heures.

Le système n'est pas vendu mais livré avec une licence d'utilisation. Le prix d'achat d'une configuration moyenne peut être évalué à 350 000 F auxquels il faut ajouter 70000 F de droit d'utilisation (y compris la formation, la libre utilisation pendant les trois premiers mois et l'assistance sur site). Passé ce délai, il faut compter environ 5000 F/mois pour la licence d'utilisation (3 000 lignes /mois) et 1,5 F par ligne supplémentaire (10 mots). A titre d'indication, nous précisons qu'une traduction technique effectuée de façon traditionnelle pour 175 F (320 F si elle est confiée à un bureau de traduction) revient à 60 F avec LOGOS. Ces chiffres sont approximatifs et ne tiennent pas compte du couple de langue ni du type de texte.

Plus précisément, le prix du mot traduit est de 37 centimes après 18 mois d'utilisation, en incluant tous les coûts, le logiciel, l'équipement, le personnel et l'organisation. Une étude des coûts beaucoup plus poussée révèle que le gain de productivité, même minime, est d'autant plus rentable que le volume des textes à traduire est important. En 1985, LOGOS était le système le plus vendu sur le marché européen (26 %), suivi par WEIDNER (23%), SYSTRAN (13%) et ALPS (12%)<sup>1</sup>.

#### 2.3.3.2.4 LINGUISTIC PRODUCTS<sup>2</sup>

La société Houston Texas commercialise une série de modules CAT (Computer Aided Translation) qui fonctionnent de façon interactive, pour les couples anglais-français, anglais-espagnol, anglais-suédois et anglais-danois.

Le système de traduction directe livre une traduction mot à mot de mauvaise qualité. C'est un produit bon marché qui a le mérite de fonctionner sur les microordinateurs compatibles IBM PC/XT. Les services de douane du Texas utilisent le système espagnol-anglais disponible depuis peu. De nouvelles versions sont prévues dans l'année avec 16 couples de langues, dans les deux sens.

#### 2.3.3.2.5 METAL<sup>3</sup>

Le système METAL est commercialisé depuis peu sous le nom de LITRAS par la firme SIEMENS et ne fonctionne que sur le couple anglais-allemand. C'est un système indépendant de la langue (module d'analyse) auquel sont associées des composantes propres aux langues traitées (les règles de grammaire et les dictionnaires).

- Le même module d'analyse peut servir à traduire vers différentes langues. Il est alors plus simple d'ajouter de nouvelles langues cibles à un couple existant que de développer un nouveau couple à chaque fois.

- Les règles de grammaire servent à analyser les phrases. Les concepteurs de METAL sont partis du principe qu'il existait une infinité de structures. Certaines règles sont donc récursives, ce qui présente deux avantages : la possibilité de reconnaître des constructions peu usitées et la réduction du nombre de règles (seulement 500 pour METAL).

(1) Extrait de presse : COMPUTERWOCHE AKTUELL, numéro du 19. 12. 1986, "Le marché européen de la traduction automatique jusqu'en 1990", (source : Dataquest).

(2) S. MELI : "Informationsmarkt der maschinellen Übersetzung" in *Terminologie et Traduction*, Commission des Communautés Européennes, 1989, p. 84

(3) Documents recueillis auprès de J. SLOCUM à l'occasion d'une visite de son laboratoire à l'université d'Austin (Texas).

- Le dictionnaire contient des informations grammaticales, morphologiques et syntaxiques, ce qui permet à METAL d'appréhender les fonctions de chaque mot dans la phrase. Son organisation est hiérarchisée. Au sommet, les mots outils (conjonctions, prépositions et déterminants), vient ensuite le vocabulaire général puis le vocabulaire technique. L'utilisateur dispose d'un système-expert qui facilite le codage des mots qu'il souhaite ajouter dans le dictionnaire. On considère que 5000 entrées couvrent 90% du vocabulaire employé dans un texte général, pour une langue indo-européenne. Un dictionnaire de 100 000 termes aurait surtout pour inconvénient de multiplier les ambiguïtés et d'accroître considérablement les besoins en capacités de stockage.

Applicant le principe de "vision globale", il traduit phrase par phrase et dispose pour chaque couple de langues, de dictionnaires monolingues et d'un dictionnaire de transfert.

### Saisie du texte

S'il n'est pas livré sur un support informatique, le texte source peut être tapé au clavier ou saisi par un logiciel de reconnaissance optique (OCR). Tout ce qui concerne la mise en forme et le formatage est extrait du texte source. Ce détail est important. On constate en effet que 60% seulement des pages de documentation technique sont effectivement à traduire. Le texte est transmis à l'interpréteur Lisp qui replacera la traduction dans le masque de la page.

### Analyse

L'analyse de l'allemand est fondée sur une grammaire syntagmatique indépendante du contexte, complétée par un ensemble de procédures capables de sélectionner les transformations. L'analyse de l'anglais, par contre, s'inspire plutôt d'une grammaire syntagmatique généralisée sans utiliser les transformations. L'analyse est distincte du transfert et le système peut être considéré comme multilingual dans le sens où les résultats de l'analyse peuvent être exploités pour le transfert et la synthèse dans plusieurs langues cibles. (Expérimentation sur le chinois et l'espagnol, tout comme sur le couple anglais-allemand). On peut citer également le projet lancé par SIEMENS et le gouvernement belge en 1985, (université Catholique de Louvain, universités de Mons et de Liège) sur les couples néerlandais-français et français-néerlandais.

Le système examine la phrase de la gauche vers la droite et analyse chaque mot d'un point de vue lexical. Il applique ensuite les règles de grammaire. En se basant sur un indice de probabilité affecté selon différents critères, il sélectionne la règle qui produira l'interprétation de la phrase la plus vraisemblable. Après avoir interprété tous les éléments de la phrase et défini leurs relations, il reconstitue les syntagmes. Une fois que toutes les règles de grammaire ont été appliquées et que les caractéristiques morphologiques, syntaxiques et sémantiques de chaque mot ont été définies, le système construit une structure arborescente qui illustre les relations syntaxiques. Les informations apportées par le dictionnaire et les règles appliquées sont associées à chaque noeud de l'arbre. Elles sont indépendantes de la langue cible. Cette étape consomme énormément de capacité mémoire (120 MB requis par le système).

### Transfert

Deux composantes, les règles de la grammaire de transfert et les entrées du dictionnaire de transfert interviennent simultanément dans un processus descendant qui étudie les scores associés aux différents arbres construits lors de la phase d'analyse.

Ce processus est contrôlé par les règles de transfert dont chacune est le plus souvent associée aux règles de grammaire impliquées dans l'analyse de départ. Si une grammaire générale de transfert est alors inutile pour déterminer la règle à appliquer, elle est cependant consultée pour la traduction des parties de phrases. La concomitance des transferts lexical et structural implique leur interaction. Les entrées du dictionnaire de transfert peuvent préciser le contexte syntaxique et/ou sémantique qui les validera. L'arbre qui a été obtenu lors de l'étape d'analyse est transformé en une structure correspondante dans la langue cible.

### Génération

Le module de génération part de l'arbre ainsi obtenu pour générer les phrases cibles, conformément aux règles grammaticales, lexicales et syntaxiques. METAL peut, dans certains cas, résoudre des problèmes d'anaphore en partant d'un pronom et en effectuant une recherche dans la phrase précédente. Il peut également lever des ambiguïtés en ajoutant des attributs sémantiques aux entrées du lexique utilisé par le système lors des phases d'analyse et de traduction.

### Résultats

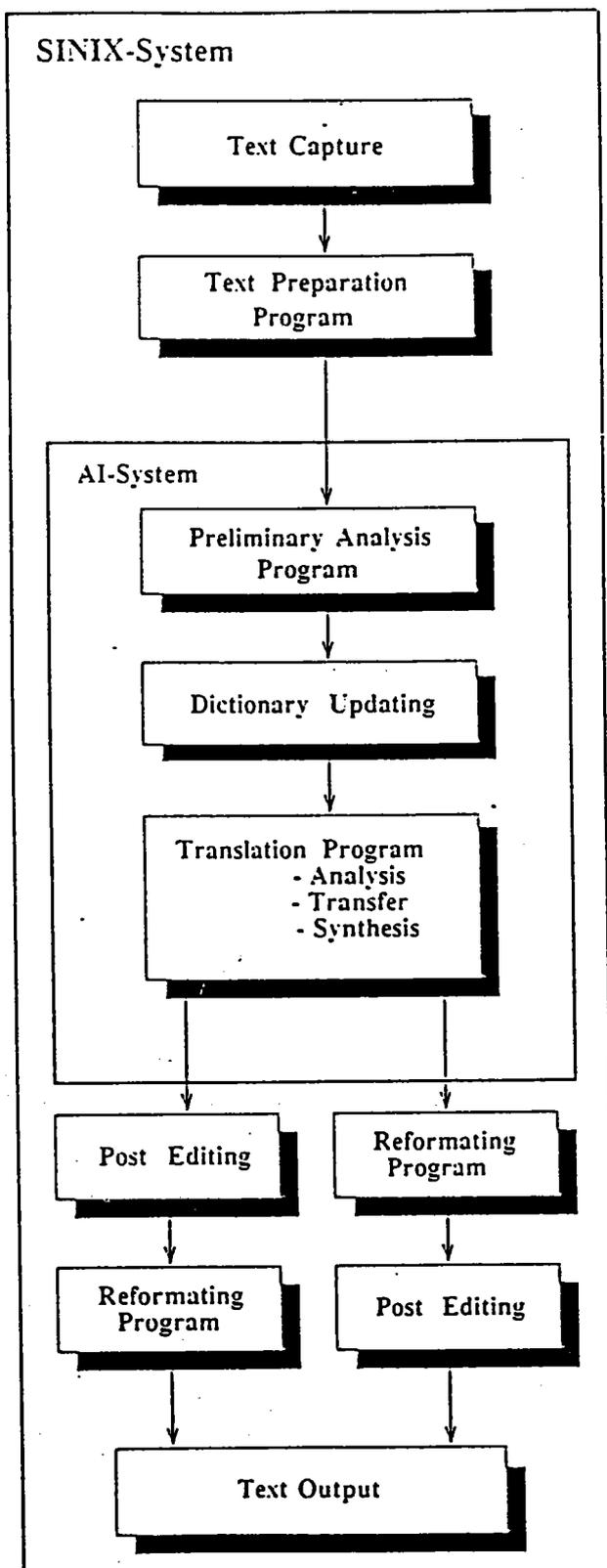
La post-édition est effectuée sur un réseau de PC connectés à l'interpréteur Lisp. La traduction brute peut être présentée dans le format du texte source ou dans un format spécifique proposant le texte source en regard du texte cible, pour comparaison.

Pour améliorer le système, la laboratoire d'Austin a mis au point plusieurs utilitaires :

- un traitement de texte qui permet de conserver la mise en page du document d'entrée pour le formatage de la traduction.
- un système de gestion de base de données pour les entrées du dictionnaire et des règles de grammaire
- un système de validation pour éliminer les erreurs que pourraient renfermer les dictionnaires et les règles de grammaire.
- Pour le vocabulaire technique, et lié au domaine que privilégie METAL, un module nommé "Data Processing" qui introduit des subdivisions telles que le hardware, le software... Le traducteur peut alors demander au système, avant le processus de traduction, d'accorder une priorité à la subdivision concernée. Le mot *Fehler* sera alors traduit par fault (hardware), error (software), defect (technique courant), sinon mistake. Le traducteur pourrait créer d'autres types de subdivisions qui tiendraient compte par exemple des différences de terminologie pour l'espagnol d'Amérique du sud et le castillan. Le système traduirait alors correctement *computer* par computadora ou ordenador.

Système complet de Traduction Automatique, il est plus particulièrement destiné aux textes techniques (informatique et télécommunication) et fonctionne sur un interpréteur Lisp (Symbolics série 36) depuis qu'il a été traduit du Fortran en Lisp. La post-édition peut être effectuée sur des PC en réseau. Sa vitesse de traduction est d'un mot par seconde. La traduction est correcte pour 50% des phrases. Le dictionnaire contient 15000 entrées.

# TRANSLATING USING METAL



## Text Input

- Text transfer via networks or data carriers (diskette, tape, scanner)

## Text Formats

- Separation of text and formatting code
- Treatment of special formats (figures, tables)

## Dictionary

- Check against dictionary
- Interactive expert system

## Translation

- Machine translation

## Post Editing

- Combining of text and formatting codes
- Correction of translated text and format
- Text system
- Printer
- Type setting

Exemple de traduction extrait de W. BENNETT, J. SLOCUM : "The LRC Machine Translation System" in *Machine Translation Systems, Studies in Natural Language Processing*, Cambridge University Press, 1988, pp. 135-140

Einteilung des Plattenspeichers

Blockstruktur

Die kleinste adressierbare Informationseinheit ist ein Block = 1 Sektor. Zu jedem Block gehoert ein Header. Der Header enthaelt die gesamte Adresse, sowie Angaben ueber den Zustand des Blockes (Benutzbarkeit!). Zur Sicherung der Header-Information und der Daten befindet sich am Ende des Headers und des Datenfeldes ein Pruefzeichen von 16 Bit.

Vor dem Headerfeld befindet sich eine Praeambel von 42 Byte Laenge fuer den Ausgleich aller Toleranzen.

Vor dem Datenfeld befindet sich eine Praeambel von 5 Byte Laenge zur Aufsynchonisierung der Leseverstaerker. Vor und hinter dem Datenfeld befindet sich eine Luecke. Die Luecken sind aus folgenden Gruenden notwendig:

Luecke 1: 56 Bit wegen Schreib-Loesch-Kopfabstand. Zu Beginn der Daten-Schreiboperation muss gewaehrleistet sein, dass der Loeschkopf den Header nicht zerstoen kann.

Luecke 2: 316 Bit im Normalmodus wegen der Toleranzen in der Umdrehungsgeschwindigkeit. Es muss die Moeglichkeit beruecksichtigt werden, dass das Schreiben des Blockes (Header + Datenfeld) an der unteren und oberen Grenze der Umdrehungsgeschwindigkeit erfolgen kann. Im Spezialmodus wird diese Luecke wegen der kleineren Blocke 1340 Bit lang.

Division of disk storage

Block structure

The smallest addressable information unit is a block = 1 sector. A header is part of every block. The header includes the entire address, sowie specifications about the state of the block (usability!). A check character of 16 bits is found for the saving of the header information and the data at the end of the header and the data field.

A preamble of 42 byte length for the adjustment of all tolerances is found in front of the header field.

A preamble of 5 byte length is found in front of the data field for the synchronization of the read amplifier. A gap is found in front of and behind the data field. The gaps are necessary from the following reasons:

Gap 1: 56 Bit because of distance between write and erase heads. At the beginning of the data write operation it must be guaranteed, that the erase head can not destroy the header.

Gap 2: 316 Bit in the normal mode because of the tolerances in the rotational speed. The possibility must be considered that writing the block (header + data field) at the lower and upper limit/boundary of the rotational speed can occur. This gap becomes 1340 bits long in special mode because of the smaller blocks.

Am Ende des Header- und Datenfeldes befindet sich 1 Postamble von 8 Bit Laenge.

### Spurstruktur

Eine Spur wird eingeteilt in 4 bzw. 8 Sektoren. Die Unterteilung der Spur in Sektoren erfolgt durch Index- und Sektormarken.

Die Indexmarke wird magnetisch durch einen Schlitz auf der untersten Platte des Plattenstapels erkannt und dient als allgemeiner Bezugspunkt fuer den Aufbau der Spurstruktur. Vom Indexpunkt ausgehend wird die Spur mit einem eigens dafuer vorgesehenen Dienstprogramm (oder Simulator!) mit Headern beschrieben. Die Bitzahl fuer das Datenfeld wird so bemessen, dass auch bei unguenstiger Drehzahl (= 2448 U/min) immer noch 4 bzw. 8 vollstaendige Bloecke Platz finden. (Siehe Abschnitt 4.1 Luecke 2). Je nachdem bei welcher Geschwindigkeit die Spur beschrieben wird, entsteht zwischen Ende des Datenfeldes und Indexmarke bzw. Sektormarke eine mehr oder weniger grosse Luecke.

### Sektormarkierung

Die Sektormarke wird ebenso wie die Indexmarke von der Schlitzplatte, die sich als Bodenplatte an jedem Plattenstapel befindet, magnetisch abgenommen. Im Handel werden Plattenstapel mit 32 und mit 20 Schlitzten angeboten. Im vorliegenden Fall soll der Plattenstapel mit 20 Schlitzten beim WSP 411 und mit 32 Schlitzten beim WSP 414 verwendet werden. Eine Maske, dargestellt durch einen Zaehler blendet aus den 20 bzw. 32 Sektormarken 4 bzw. 8 aus, so dass 4 bzw. 8 gleich grosse Sektoren entstehen. Die Maske bzw. der Zaehler wird von der Herstellerfirma (CDS) in jeden Wechselplattenspeicher fest eingebaut.

The 1 postamble of 8 bit length is found at the end of the header and data field.

### Track structure

A track is divided into 4 and/or 8 sectors. The subdivision of the track into sectors occurs through index label and sector marks.

The index label is recognized magnetically by a slot on the lowest disk of the disk pack and is used for the track structure as the general reference point for establishing. By the index point, the track with a utility program designated especially for that (or simulator!) is described with headers. The number of bits for the data field is calculated then that always still 4 and/or 8 complete blocks do also find space with unfavorable rotational speed/number of revolutions (= 2448 r.p.m.s). (See section 4.1 gaps 2.) Depending on with which speed the track is described, a more or less large gap occurs between the end of the data field and index label and/or sector mark.

### Sector marker

Likewise the index label is read in the sector mark as by the slot disk which is found as a bottom disk at every disk pack magnetically. The disk packs with 32 and with 20 slotting are offered on the market. The disk pack should be used in this case with the 20 slots with the WSP 411 and with the 32 slots with the WSP 414.

A mask, represented through a counter masks out 4 and/or 8 from the 20 and/or 32 sector marks so that large sectors result similar to 4 and/or 8. The mask and/or the counter is incorporated by the Herstellerfirma (CDS) into every removable disk storage.

Exemples présentés par Monsieur le Professeur BOURQUIN dans le cadre de ses séminaires de D.E.A. :

METAL T54

11 MARS 1982 16H 41MN 14S

LANGUES DE TRAITEMENT: ANG-FRA

-- TEXTE ORIGINE --

for all pairs of materials a coating of the contact materials with Celloseel® lowered the corrosion current density.

-- TEXTE TRADUIT --

----- ( TRADUCTION DU 11 MARS 1982 16H 34MN 24S ) -----  
VERSIONS : ( A : 2/03/82 ; T : 8/03/82 ; G : 5/03/82 )

pour toutes paires de matériaux un revêtement des matériaux de contact avec Celloseel a abaissé la densité du courant de corrosion.

METAL T55

11 MARS 1982 16H 49MN 30

LANGUES DE TRAITEMENT: ANG-FRA

-- TEXTE ORIGINE --

previously abstracted from original as item 7704-64-0108Y.

-- TEXTE TRADUIT --

----- ( TRADUCTION DU 11 MARS 1982 16H 47MN 52S ) -----  
VERSIONS : ( A : 2/03/82 ; T : 8/03/82 ; G : 5/03/82 )

precedemment resumées a partir original comme l'article <7704-64-0108Y>.

METAL T56

11 MARS 1982 16H 56MN 40

LANGUES DE TRAITEMENT: ANG-FRA

-- TEXTE ORIGINE --

the effect of the thickness of the Al coating on the corrosion and H absorption of steel sheet (1-3 mm) and steel wire (0.5 mm diam) was studied .

-- TEXTE TRADUIT --

----- ( TRADUCTION DU 11 MARS 1982 16H 54MN 23S ) -----  
 VERSIONS : ( A : 2/03/82 ; T : 8/03/82 ; G : 5/03/82 )

l'effet de l'épaisseur du revêtement de Al sur la corrosion et absorption de H de la tôle d'acier (<1-3> mm) et on a étudié le câble d'acier (<0.5> mm de diam).

METAL T58

11 MARS 1982 17H 26MN 38S

-LANGUES DE TRAITEMENT: ANG-FRA

-- TEXTE ORIGINE --

for thicker coatings the samples behaved like pure Al.

-- TEXTE TRADUIT --

----- ( TRADUCTION DU 11 MARS 1982 17H 25MN 43S ) -----  
 VERSIONS : ( A : 2/03/82 ; T : 8/03/82 ; G : 5/03/82 )

pour des revêtements plus épais les échantillons se sont comportés comme le Al pur.

METAL T59

11 MARS 1982 17H 53MN 45

LANGUES DE TRAITEMENT: ANG-FRA

-- TEXTE ORIGINE --

electron diffraction studies showed that Fe spinel  $FeAl_2O_4$  is formed on the steel surface.

-- TEXTE TRADUIT --

----- ( TRADUCTION DU 11 MARS 1982 17H 49MN 01S ) -----  
 VERSIONS : ( A : 2/03/82 ; T : 8/03/82 ; G : 5/03/82 )

diffraction d'electron etudie que spinelle de Fe  $FeAl_2O_4$  est formee sur la surface d'acier.

METAL T60

11 MARS 1982 17H 59MN 12

LANGUES DE TRAITEMENT: ANG-FRA

-- TEXTE ORIGINE --

the adhesion of the Al coating is high and is retained almost up to the complete destruction of the sample.

-- TEXTE TRADUIT --

----- ( TRADUCTION DU 11 MARS 1982 17H 58MN 09S ) -----  
 VERSIONS : ( A : 2/03/82 ; T : 8/03/82 ; G : 5/03/82 )

l'adherence du revetement de Al est elevee et est ete conservee presque jusqu'a la destruction complete de l'echancillon.

METAL T62

11 MARS 1982 18H 14MN

LANGUES DE TRAITEMENT: ANG-FRA

-- TEXTE ORIGINE --

the corrosion and the diffusion of H in H2SO4 decreases greatly as the coating thickness is increased.

-- TEXTE TRADUIT --

----- ( TRADUCTION DU 11 MARS 1982 18H 13MN 50S ) -----  
 VERSIONS : ( A : 2/03/82 ; T : 8/03/82 ; G : 5/03/82 )

la corrosion et la diffusion de H dans H2SO4 diminue considérablement comme on augmente l'épaisseur de revêtement.

METAL T63

11 MARS 1982 18H 20MN

LANGUES DE TRAITEMENT: ANG-FRA

-- TEXTE ORIGINE --

the purpose of this continuing investigation was to measure limiting C diffusion current densities as a function of seawater velocity, to study cathodic kinetics on 5456-H117 Al alloy as a function of velocity and to conduct constant potential tests at more active potentials than -0.9 VSCE.

-- TEXTE TRADUIT --

----- ( TRADUCTION DU 11 MARS 1982 18H 19MN 58S ) -----  
 VERSIONS : ( A : 2/03/82 ; T : 8/03/82 ; G : 5/03/82 )

le but de cette recherche en cours était pour mesurer les densités du courant limite de diffusion de C comme une fonction de la vitesse de l'eau de mer, pour étudier la cinétique cathodique sur l'alliage d'Al 5456-H117 comme une fonction de la vitesse et pour mener les tests constants de potentiel à des potentiels plus actifs que' <-0.9> VSCE.

METAL T64

11 MARS 1982 18H 24MN 30S

LANGUES DE TRAITEMENT: ANG-FRA

-- TEXTE ORIGINE --

finally, a model was proposed to explain the observations from this and previous studies.

-- TEXTE TRADUIT --

----- ( TRADUCTION DU 11 MARS 1982 18H 24MN 13S ) -----  
 VERSIONS : ( A : 2/03/82 ; T : 8/03/82 ; G : 5/03/82 )

finalement, on a propose un modele pour expliquer les observations a partir ceci et etudes precedentes.

METAL T65

11 MARS 1982 18H 29MN 24S

LANGUES DE TRAITEMENT: ANG-FRA

-- TEXTE ORIGINE --

the corrosion of some steels and Al alloys in ammoniacal solutions (50-200 g/l NH<sub>3</sub>) was investigated.

-- TEXTE TRADUIT --

----- ( TRADUCTION DU 11 MARS 1982 18H 29MN 06S ) -----  
 VERSIONS : ( A : 2/03/82 ; T : 8/03/82 ; G : 5/03/82 )

on a etudie la corrosion de quelques aciers et des alliages de Al dans des solutions ammoniacales (<50-200> g-l de NH<sub>3</sub>).

METAL T67

11 MARS 1982 18H 37MN 15

LANGUES DE TRAITEMENT: ANG-FRA

-- TEXTE ORIGINE --

even low concentrations (0.1 percent) of NaOH increase the corrosion rate.

-- TEXTE TRADUIT --

----- ( TRADUCTION DU 11 MARS 1982 18H 35MN 19S ) -----  
VERSIONS : ( A : 2/03/82 ; T : 8/03/82 ; G : 5/03/82 )

même des basses concentrations (<0.1> pour cent) de NaOH augmentent le taux de corrosion.

METAL T66

11 MARS 1982 18H 33MN 3.

LANGUES DE TRAITEMENT: ANG-FRA

-- TEXTE ORIGINE --

initially, the Al alloys are strongly corroded by ammoniacal solutions, but the corrosion is stopped in time.

-- TEXTE TRADUIT --

----- ( TRADUCTION DU 11 MARS 1982 18H 33MN 12S ) -----  
VERSIONS : ( A : 2/03/82 ; T : 8/03/82 ; G : 5/03/82 )

au départ, on corrode les alliages de Al fortement par ces solutions ammoniacales, mais on arrête la corrosion dans le temps.

### 2.3.3.2.6 PENSEE<sup>1</sup>

Le système a été créé et commercialisé par la firme japonaise OKI, en 1986. Il fonctionne sur le couple japonais-anglais. Une version anglais-japonais est en développement. Le logiciel est écrit en langage C et tourne sur un ordinateur MC-68010 ou MC-68020C (système d'exploitation UNI PLUS+). Le système est interactif et réalise une traduction automatique à la vitesse de 4 000 mots/heure. Une étape de post-édition est nécessaire.

### 2.3.3.2.7 PIVOT<sup>1</sup>

Destiné à devenir un système multilingual, PIVOT a été développé par la firme japonaise NEC et commercialisé en 1986 pour les couples anglais-japonais et japonais-anglais. Sur le principe de l'utilisation d'un langage pivot, le système utilise des techniques de l'Intelligence Artificielle et fonctionne en mode interactif pour réaliser une traduction totalement automatique. Le dictionnaire contient 40 000 entrées pour le japonais, 53 000 pour l'anglais et 400 000 termes techniques. Il est écrit en langage C et fonctionne sur un microordinateur ACOS 4 ou sur un gros ordinateur (système d'exploitation ACOS 4). Les textes traités sont techniques ou scientifiques.

### 2.3.3.2.8 SANYO<sup>1</sup>

Développé par la firme japonaise SANYO et commercialisé sous le nom de SWP-7800, il fonctionne sur le couple japonais-anglais. Construit sur le modèle des systèmes de transfert, il est écrit en langage C, tourne sur un Intel 80186 (système d'exploitation IRMX), nécessite une phase de pré- et de post-édition et traduit à la vitesse de 3 500 mots/heure à l'aide d'un dictionnaire de 55 000 entrées.

### 2.3.3.2.9 SYSTRAN<sup>2,3,4,5</sup>

SYSTRAN est vraisemblablement le plus ancien des systèmes de traduction automatique en fonctionnement. Il faut cependant souligner les efforts déployés pour l'améliorer et le développer. Ses origines remontent au projet GAT de l'université de Georgetown (2.3.2.1).

Les améliorations concernent la partie logicielle avec une modularité accrue et l'abandon d'une méthode de traduction directe pour une approche de transfert.

SYSTRAN est le système le plus répandu (sur gros ordinateur)<sup>6</sup>. L'U.S. Air Force l'utilise depuis 1970 à Dayton, Ohio, (100 000 pages traduites par an depuis 1987 sur le couple russe-anglais). Depuis 1989, il traduit des brevets (russe-anglais). General Motors Canada l'utilise depuis longtemps pour l'anglais-français et depuis peu pour l'anglais-allemand et l'anglais-espagnol, Xerox pour la traduction de manuels techniques (50 000 pages/an anglais-français, anglais-espagnol, anglais-portugais, anglais-italien,

(1) S. MELI : "Informationsmarkt der maschinellen Übersetzung" in *Terminologie et Traduction*, Commission des Communautés Européennes, n°3, 1989, pp. 83-84

(2) S. TRABULSI : "Le système SYSTRAN" in *TAO*, Observatoire des Industries de la langue, DAI-CADIF, 1989, pp. 15-34

(3) Documents de la société GACHOT

(4) Proceedings of the World SYSTRAN Conference, Luxembourg, février 1986, published by the Commission of the European Communities

(5) G. VAN SLYPE : "Description du système de Traduction Automatique SYSTRAN de la Commission des Communautés Européennes" in *Documentaliste*, vol. 16, n°4, 1979, pp. 150-159

(6) W. HUTCHINS : "Recent Developments in Machine Translation. A Review of the Last Five Years" in : *New Directions in Machine Translation*, Conference Proceedings, Budapest, 18-19/08/1988

anglais-allemand, l'OTAN à Bruxelles, le Centre de Recherche nucléaire à Karlsruhe (français-anglais), la Deutsche Bundesbahn et d'autres grandes sociétés comme Dornier, Festo, l'Agence Internationale de l'Energie Atomique (Vienne), l'Aérospatiale pour des manuels de documentation aéronautique (anglais-français, français-anglais). Nous ne retracerons pas l'histoire de SYSTRAN mais rappellerons qu'en février 1976 la Communauté Européenne, grande consommatrice de traduction, a décidé d'acquérir une version de SYSTRAN (fonctionnant sur gros calculateur IBM) pour l'évaluer et s'en équiper. Les résultats ont été décevants mais l'expérience a été poursuivie en espérant que l'ajout de dictionnaires améliorerait les performances du système au point de justifier son installation. Les versions français-anglais (1978) et anglais-italien (1979) furent à leur tour installées en 1981.

Deux couples de langues supplémentaires anglais-français et français-allemand ont été créés en 1982. La Communauté l'utilise à Luxembourg, mais les traductions réalisées n'ont représenté en 1987 que 2% de l'ensemble des traductions, l'avantage essentiel résidant dans l'élargissement considérable des lexiques. Le fait que les textes soumis au système soient au préalable sélectionnés explique le faible pourcentage, il est probable qu'il augmentera si les traducteurs acceptent de rédiger la version brute que leur propose la machine, tant il est prévisible que le futur système (EUROTRA) n'est pas près de lui succéder.

Depuis 1986, les sociétés américaines (LATSEC et WTC aux Etats-Unis) et européennes (Systran Institute en Allemagne et sa filiale au Luxembourg) qui participaient à l'exploitation de SYSTRAN ont été rachetées par le groupe Gachot qui est chargé d'assurer la cohérence de ses développements (Au Japon, c'est la société IONA), et a conclu un accord de coopération avec la Commission des Communautés Européennes. Cette dernière consacre d'ailleurs un budget très important au développement des dictionnaires pour les langues européennes (32 personnes de la CCE et de la société Informalux). Des accords de développement ont également été conclus pour l'extension des dictionnaires avec Rank Xerox et General Motors.

SYSTRAN a été développé sur des ordinateurs IBM, selon le principe d'une absence de limitation en matière de temps calcul et de capacité de stockage. Les clients devaient posséder de gros ordinateurs et construire les interfaces en amont et en aval du système. La société Gachot a comblé le manque de convivialité inhérent aux applications lourdes en conjuguant les techniques de l'informatique et de la télématique. Le groupe a créé un serveur SYSTRAN qui permet à l'utilisateur de communiquer avec le système à partir de son microordinateur ou via le Minitel par le réseau commuté (par TRANSPAC).

SYSTRAN est le seul système capable en principe de traduire des textes de toute sorte, en raison de ses dictionnaires nombreux et constamment élargis. Le système dispose de 500 catégories sémantiques dans une structure hiérarchisée, pour la résolution des ambiguïtés.

SYSTRAN est un système de première génération en mutation. Le classement par "génération" est peu significatif<sup>1</sup>. Cela signifie qu'il prenait un certain nombre de décisions définitives à chaque pas de l'analyse, contrairement "au principe de base heuristique récuratif" des systèmes de seconde génération. D'un point de vue linguistique, le caractère linéaire de son principe facilitera les transformations tant au niveau des programmes qu'au niveau des dictionnaires. On peut ajouter facilement des modules ici et là, qu'il s'agisse d'une liste de règles lexicales (dictionnaires) pour améliorer la qualité de la traduction, ou d'un module de traitement complémentaire pour le système de base.

---

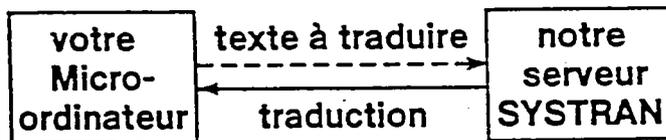
(1) P. TOMA : "SYSTRAN : Ein maschinelles Übersetzungssystem der 3. Generation" in *Sprache und Datenverarbeitung*, 1/1977, pp. 38-45, Niemeyer, Tübingen, 1977

Documentation SYSTRAN International :

# SYSTRAN

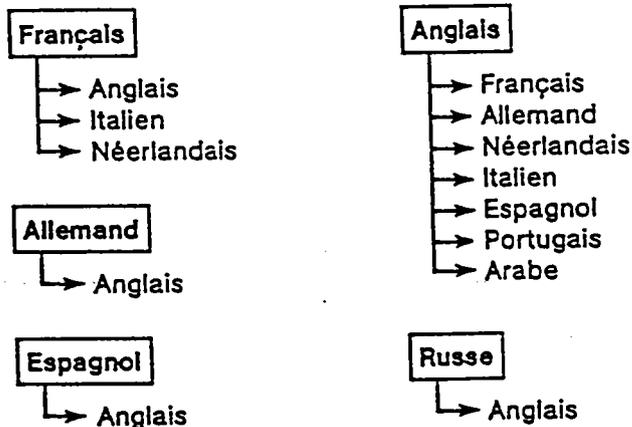
## INTERNATIONAL

### Traduction automatique sur ordinateur



10 pages - 10 minutes

- Traduisez sans contrainte vos textes techniques, documentations spécialisées, notices explicatives, etc., dans les langues suivantes :



- Nos références :

U.S. Air Force, C.E.E., A.T.T., Rank Xerox...

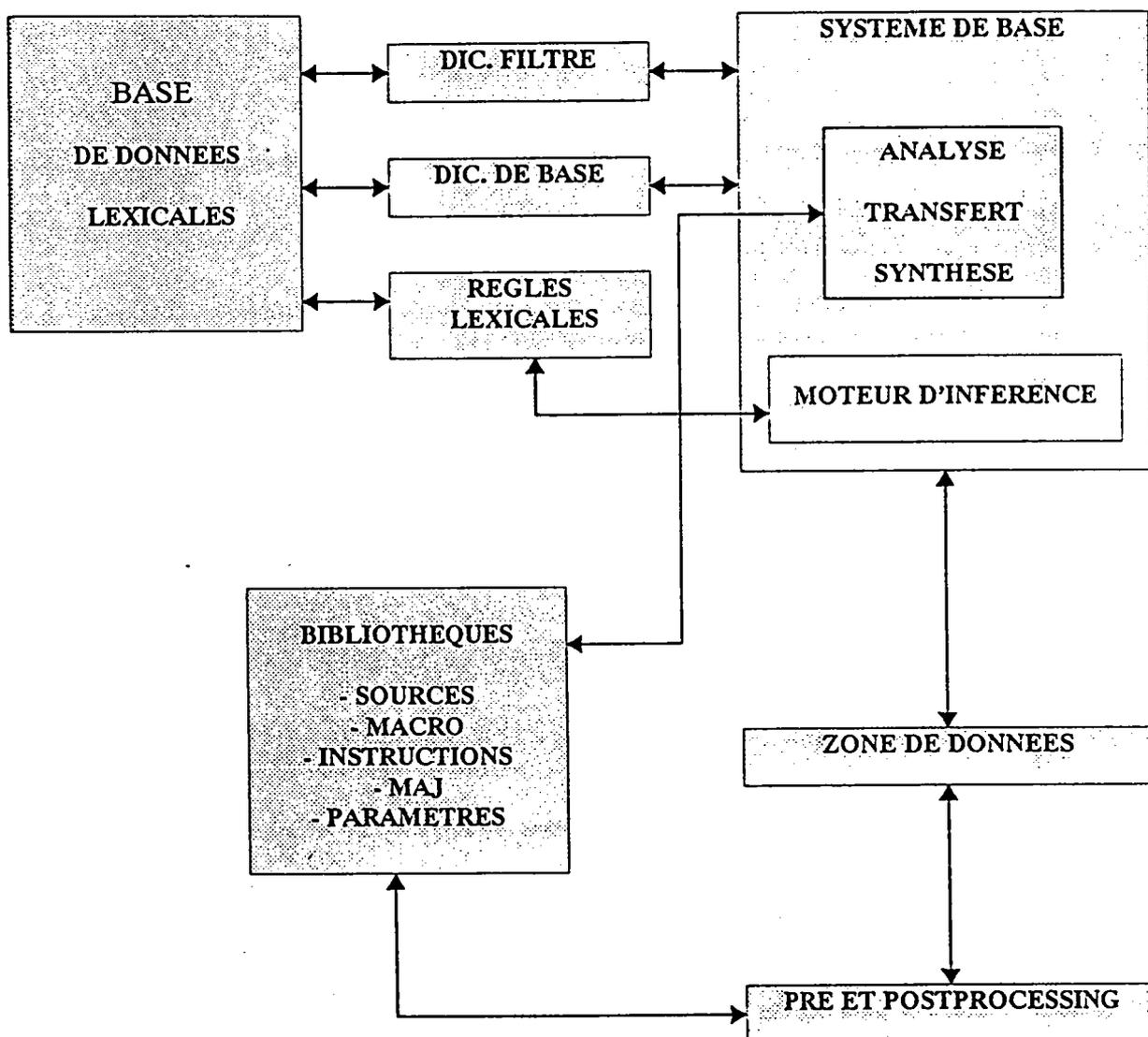
- Votre contact : 33 1 39 89 90 11

**SYSTRAN INTERNATIONAL**  
Groupe GACHOT

26 bis, avenue de Paris  
95230 SOISY-sous-MONTMORENCY - FRANCE  
Télex : 606754 F - Téléfax : 33 1 39 89 49 34

**NOUS TRADUISONS L'AVENIR**

Le schéma suivant réunit les composants du système :



Le noyau du système rassemble les programmes responsables des étapes classiques : analyse, transfert et synthèse. Ces programmes regroupent des sous-programmes ayant chacun un rôle précis. Ils sont écrits dans une variante du Fortran inspirée de l'Assembleur et adaptée aux exigences de la linguistique. Chaque couple de langues correspond à un ensemble de 130 000 lignes de programmation (dont les 5/6 pour l'analyse) gérées par un langage de contrôle qui enchaîne les traitements et assure la continuité des tâches puisqu'il est capable d'interrompre l'exécution lorsqu'il rencontre une anomalie.

Les dictionnaires (30 Mo dans un format machine pour un couple de langues) constituent un ensemble qui fonctionne à la manière d'un système expert. Les règles lexicales sont classées sur douze niveaux, dans une base de connaissances, sous la forme de groupes d'opérateurs. Chaque niveau est subdivisé en seize niveaux de priorité. Le tout est piloté par un moteur d'inférence indépendant des langues sources et cibles. Ce moteur décide d'appliquer ou non chaque règle lexicale. (Pour un texte de 100 000



Un troisième ensemble regroupe les bibliothèques de programmes sources, d'utilitaires, de dictionnaires sources, de bases de données, de programmes d'interface avec les différents types de périphériques (530 bibliothèques au total, 5000 MB !).

Les logiciels comprennent en fait :

- le logiciel de traduction
- le logiciel de création et de mise à jour des dictionnaires
- les utilitaires

Les programmes qui assurent la création et la mise à jour des dictionnaires, les modules qui contrôlent le déroulement du processus de traduction et les utilitaires sont écrits en Assembleur. Ils sont indépendants des langues traitées.

Les programmes qui analysent la langue source et assurent la synthèse dans la langue cible sont rédigés dans le macro-langage SYSTRAN qui permet de coder plus simplement sous une forme symbolique les règles syntaxiques et sémantiques. Ils sont spécifiques à chaque couple de langues.

En moyenne, selon le couple de langues, un mot traduit implique 25 à 30 000 opérations. On peut résumer le processus de traitement de la façon suivante :

#### **Etape I : Lecture du texte source et consultation des dictionnaires**

le programme MTST lit le texte à traduire, le convertit à l'aide des caractères hexadécimaux du standard SYSTRAN, stocke chaque phrase dans un enregistrement de longueur variable, enregistre les données concernant la mise en forme et le formatage pour les restituer lors de l'impression de la traduction.

Le programme LOADTXT charge les dictionnaires d'unités fréquentes et les listes d'idiotismes en mémoire centrale, puis le texte, phrase par phrase. Il compare très rapidement chaque mot de la phrase avec les entrées des dictionnaires (méthode binaire, binary search). S'il y a concordance avec un mot ou une racine de haute fréquence, il ajoute un code au mot du texte (qui introduit de ce fait les informations grammaticales correspondantes en mémoire centrale). Si c'est une expression idiomatique, il ajoute le code au premier mot de l'idiotisme. Il en profite pour numéroter les mots du texte, traiter les mots à trait d'union et trier alphabétiquement tous les mots qui n'ont pas été reconnus. (Cet ordre correspond au type de classement du dictionnaire général). Ce travail est effectué à une vitesse de 20 000 mots/minute.

Le programme MDL (Main Dictionary Lookup) charge en mémoire centrale, bloc par bloc, le dictionnaire et le texte trié des mots non fréquents et n'appartenant pas à une expression idiomatique pour les comparer (vitesse de 4 000 mots/minute). Les mots y sont classés par ordre alphabétique et par racines. Figurent également dans ce dictionnaire, outre les informations grammaticales, les expressions techniques multitermes et les traductions dans la langue cible. S'il y a correspondance, il copie à côté du mot les informations grammaticales du dictionnaire et les traductions possibles. Après consultation du dictionnaire, il replace les mots dans l'ordre initial du texte (en reprenant les numéros d'ordre affectés par LOADTXT) à une vitesse de 10 000/minute.

La routine NFWRN (Not Found Word Routine) examine les mots qui n'ont pas été trouvés dans le dictionnaire pour leur associer éventuellement un code, s'il s'agit d'un nombre, si la terminaison figure dans le fichier des désinences, si c'est une abréviation ou enfin un nom propre.

Pour un texte source de 140 000 mots :

- 40 000 mots sont reconnus lors du processus de comparaison avec le dictionnaire des mots courants et reçoivent un code (2 minutes 30 secondes de temps calcul).
- Les 100 000 mots restants sont triés alphabétiquement en 4 minutes.
- Recherche de ces mots et analyse morphologique (4 minutes).
- Rangement des mots dans l'ordre initial avec les informations afférentes (5 minutes).

## Etape 2 : Analyse

le programme INITCALL charge et exécute les programmes de traduction responsables de l'analyse du texte source, du transfert, de la synthèse et de l'édition de la traduction.

le programme GETSENTN réserve une zone d'analyse de 16,8 Ko en mémoire centrale (105 mots de 160 caractères) et une zone complémentaire pour les homographes. Le texte est chargé, phrase par phrase dans la zone principale. Il reclasse dans l'ordre d'origine les mots à haute fréquence (auxquels LOADTXT a associé l'adresse des informations grammaticales dans la table des règles grammaticales) et les mots à basse fréquence (auxquels MDL a associé les informations grammaticales et les traductions possibles).

Le programme de résolution des homographes lit la phrase de la gauche vers la droite pour repérer les homographes. Lorsqu'il en localise un, il appelle la routine correspondant au type d'homographe, et l'étudie en examinant le contexte gauche et le contexte droit. En dernière extrémité, le programme retient la solution considérée comme la plus fréquente.

Le programme LSLOOKUP identifie les groupes de mots contigus qui forment une expression composée (expression LS, Limited Semantics). Chaque mot du texte, à ce stade, a été complété par un numéro d'identification propre à chaque mot et à ses flexions lors de la consultation du dictionnaire des unitermes. LSLOOKUP compare alors ces numéros d'identification à un fichier contenant les séquences de numéros constitutives d'expressions LS. S'il y a concordance, la traduction des mots de l'expression LS remplace la traduction qui avait été proposée par le dictionnaire des unitermes. Si les informations syntaxiques et sémantiques liées à l'expression LS ne correspondent pas aux informations fournies initialement par le dictionnaire des unitermes, elles les remplacent.

Le programme de délimitation des propositions (SP0, Structural Pass 0) lit la phrase pour localiser les mots susceptibles de délimiter une proposition (pronom relatif, conjonction de subordination, ponctuation). Il peut identifier 11 types de propositions subordonnées, incluses ou pas.

Nous empruntons à G. v. SLYPE l'exemple de la phrase anglaise suivante :

*the parameters THAT affect the launch time will be given.*

La routine cherche le premier verbe conjugué sur la droite de *THAT* --> *affect*. Elle cherche ensuite une virgule qui ne soit pas une virgule d'énumération mais n'en trouve pas. Elle cherche alors un verbe conjugué sur la gauche de *THAT* et n'en trouve pas (la proposition relative est donc incluse). Elle cherche un autre verbe sur la droite de *THAT* --> *will*. La routine cherche ensuite sur la gauche de *THAT* un sujet potentiel de la proposition principale, c'est à dire un nom ou un pronom qui ne soit pas introduit par une préposition --> *parameters*. Le mot qui précède le second verbe est un délimiteur --> *launch*. La proposition relative est *THAT affect the launch*. Les propositions sont numérotées et chacun de leurs constituants identifiés selon le type de la proposition.

Le programme d'identification des relations syntaxiques primaires (SP1, Structural Pass 1) lit la phrase de la droite vers la gauche en ignorant les propositions incluses pour les reprendre ultérieurement, et identifie 10 types de groupes syntaxiques. (le nom et ses modificateurs, le verbe et ses objets, le pronom relatif et ses antécédents...). Il positionne des pointeurs pour relier les modificateurs aux mots modifiés, les mots gouvernants aux mots gouvernés, en indiquant le type du groupe :

groupe verbe - objet direct	<i>Have a cigarette</i>
nom/ajectif+infinitif - infinitif	<i>the attempt to think clearly</i>
adverbe - adverbe/adjectif	<i>the extremely disagreeable Woman</i>

le programme identifie les mots qui fonctionnent comme des noms, associe aux verbes les informations relatives au temps, à la personne, aux auxiliaires, traite les degrés de l'adjectif.

Le programme d'identification des relations syntaxiques étendues (SP2, Structural Pass 2), pour l'anglais, parcourt la phrase de droite à gauche pour localiser les "AND", les "OR" et les virgules qui ne sont pas des délimiteurs de proposition. Il tente alors d'identifier les éléments susceptibles de constituer une énumération et positionne des pointeurs pour les relier.

Le programme d'identification du sujet et du prédicat (SP3, Structural Pass 3) lit les propositions en ignorant les incluses, cherche un verbe fini ou une énumération de verbes finis dans la proposition et positionne des pointeurs qui relient le premier mot de la proposition au premier verbe. Il recherche ensuite le sujet. S'il n'en trouve pas, le prédicat est marqué comme un verbe à l'impératif. Il relie enfin par pointeur, le sujet et le prédicat, le premier mot de la phrase et le sujet et recherche si le sujet ne fait pas partie d'une éventuelle énumération.

Le programme d'appartenance des prépositions (PREPPGM1) lit la phrase et recherche les prépositions puis les mots dont elles dépendent ou qui en dépendent. (possibilité de trouver un mot qui gouverne plusieurs prépositions).

### Etape 3 : Le transfert

Le programme CLSLOOKUP reconnaît les expressions conditionnelles. Il s'agit alors d'utiliser les critères qui permettent de déterminer comment et sous quelles conditions les mots de l'expression seront traduits. (on compare les relations syntaxiques qui ont été déterminées entre les mots de l'expression avec les règles grammaticales qui sont intégrées dans le dictionnaire des expressions LS).

Le programme de traduction des prépositions (PREPPGM2) traduit les prépositions soit directement à partir du dictionnaire soit à partir des indications fournies par le dictionnaire pour les mots qui en dépendent ou dont elles dépendent.

Le programme de résolution des ambiguïtés sémantiques (LEXICAL) doit résoudre les ambiguïtés sémantiques restantes à l'aide de routines lexicales.

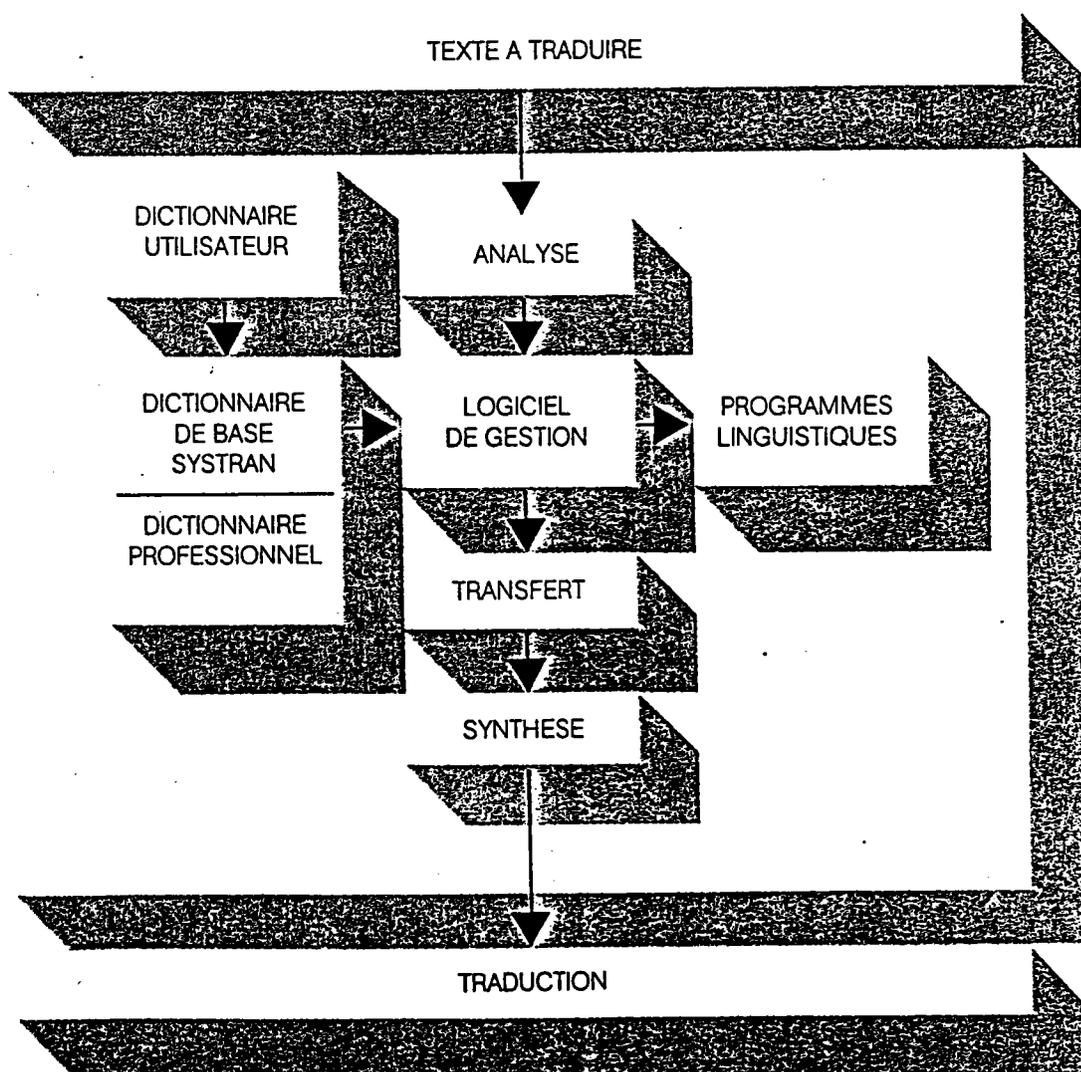
#### Etape 4 : La synthèse

Le programme de synthèse lit la phrase de gauche à droite et traduit chaque mot en utilisant les différents dictionnaires et des routines spécifiques à certaines parties du discours (pour le verbe, par exemple avec le français comme langue cible, , détermination de la personne, du nombre, du temps et du mode, choix de l'auxiliaire, respect des accords du participe....).

Le programme de réarrangement replace les adverbes, les adjectifs et les noms dans l'ordre correct pour la langue cible.

#### Etape 5: L'édition

Le programme d'impression de la traduction réalise la mise en page du texte traduit en reprenant les indications stockées lors de la première étape.



## Exemples de traductions obtenues par SYSTRAN :

SOURCE (V 1)	CIBLE (V 2) <span style="float: right;">①</span>
<p>During its first ten years, the common agricultural policy has been mainly based on the common organization of the agricultural markets.</p> <p>The common organization of the markets is thus based on three fundamental principles the validity of which has not been affected in any way by the accession of the three new Member States.</p> <p>The possibility that the US Environmental Protection Agency may ban the use of chemical sterilants for sterilizing packaging materials is mentioned, and consideration is given to the development of systems whereby packaging materials (e.g. plastics pouches) are sterilized by exposure to UV light.</p> <p>Proposal for a Council Regulation on a common measure for forestry in certain dry Mediterranean zones of the Community (submitted to the Council by the Commission).</p> <p>The methods previously used for denitrification studies, involving isotopic studies (<math>^{15}\text{N}</math> and <math>^{13}\text{N}</math>) and gas chromatography, were insensitive, laborious or affected by contamination by atmospheric <math>\text{N}_2</math> (Yoshinari et al., 1977).</p>	<p>Pendant ses les dix premiers ans, la politique agricole commune principalement a été fondée sur l'organisation commune des marchés agricoles.</p> <p>L'organisation commune des marchés est fondée ainsi sur trois principes fondamentaux la validité desquels n'a pas été affectée de quelque façon que ce soit par l'adhésion des trois nouveaux Etats membres.</p> <p>On mentionne la possibilité que l'agence Etats-Unis de protection de l'environnement peut interdire l'utilisation de stérilisants chimiques pour stériliser des matériaux de conditionnement, et on donne la considération au développement des systèmes par lequel des matériaux de conditionnement (par exemple poches plastiques) sont stérilisés par l'exposition à la lumière UV.</p> <p>Proposition pour un règlement du Conseil sur une mesure commune pour la sylviculture dans certaines zones méditerranéennes sèches de la Communauté (soumise au Conseil par la Commission).</p> <p>Les méthodes employées précédemment pour des études de dénitrification, comportant les études isotopiques (<math>^{15}\text{N}</math> et <math>^{13}\text{N}</math>) et la chromatographie en phase gazeuse, étaient insensibles, laborieuses ou affecté par la contamination par le <math>\text{N}_2</math> atmosphérique (Yoshinari et autres, 1977).</p>

SOURCE (V 1)	CIBLE (V 2) <span style="float: right;">②</span>
<p>(c) the problem of the lost unity of the Community market, caused by monetary compensatory amounts and hindrances such as persistent obstacles to the free movement of certain products or to the specialization of agricultural areas in accordance with their relative advantages.</p> <p>If, however, during a period of frequent changes in the state of international markets in agricultural produce, the Community were to be faced with surpluses, the maximum priority would be given to disposing of these surpluses on the internal market.</p> <p>Plantations in Cameroon comprise numerous aged coffee trees which must be regenerated either to increase the profitability of estates or diminish planted areas, while maintaining the overall production of the country at the same level.</p> <p>It has been necessary to analyse various market data, both on the supply and demand sides, in order to enable a decision to be taken in keeping with market conditions.</p>	<p>(c) le problème de l'unité perdue du marché communautaire, provoquée par des montants compensatoires et des obstacles monétaires tels que les obstacles persistants à la libre circulation de certains produits ou à la spécialisation des zones agricoles en conformité avec leurs avantages relatifs.</p> <p>Si, cependant, pendant une période des changements fréquents dans l'état de marchés d'international dans des produits agricoles, la Communauté devait être confrontée par des excédents, la priorité maximum serait donnée à éliminer ces excédents sur le marché intérieur.</p> <p>Les plantations au Cameroun comportent les nombreux arbres âgés de café qui doivent être régénérés pour augmenter la rentabilité des domaines ou pour diminuer des régions plantées, tout en maintenant la production générale du pays au même niveau.</p> <p>C'a été nécessaire d'analyser de différentes informations sur le marché, tous deux sur l'approvisionnement et des côtés de demande, pour permettre à une décision d'être pris en accord avec des conditions de marché.</p>

SOURCE (V 1)	CIBLE (V 2) <span style="float: right;">③</span>
<p>Given the complexity of actual situations and the wide variety of instruments employed, the across-the-board instruments will first be examined, followed by the general guidelines concerning a number of products and, finally, the specific improvements proposed for the common organization of the principal markets.</p> <p>Constraints for moving sprinklers and pipes, and labour cost involved prove high when spray irrigation concerns large plots.</p> <p>Undertakings which were still concerned with agricultural production on the formation of the EEC have gradually shifted to the production-processing area.</p> <p>On the supply side this has, for example, concerned the product, product quality and product developments by the agricultural undertaking concerned and by competitors' market force and strategy, the potential in respect of basic materials and auxiliary materials and other equipment, suppliers' capacities, financing aspects and marketing methods.</p> <p>The relationship between the length of sarcomeres and the tenderness of meat has been well established since the initial observations of Locker (1960).</p>	<p>Donné la complexité des situations réelles et la grande variété d'instruments utilisés, les instruments de à travers--panneau d'abord seront examinés, suivi ces lignes directrices générales au sujet d'un certain nombre de produits et, finalement, les améliorations spécifiques proposées pour l'organisation commune du principal vendent.</p> <p>Les contraintes pour des arroseuses et des conduits mobiles, et le coût de la main-d'oeuvre en question s'avèrent que haut arrosera quand parcelles de terrain de soucis d'irrigation de grandes.</p> <p>Les engagements qui encore ont été concernés par une production finale de l'agriculture sur la formation de la CEE ont décalé graduellement à la région de production-traitement.</p> <p>Sur le côté d'approvisionnement ceci, par exemple, a concerné le produit, la qualité du produit et les développements de produit par l'engagement agricole concerné et par la force de marché des concurrents et strategy, le potentiel en ce qui concerne des matières de base et les matières auxiliaires et d'autres équipement, capacités des fournisseurs, aspects de financement et méthodes de vente.</p> <p>Le rapport entre la longueur des sarcomeres et la tendreté de la viande a été bien établi depuis les observations initiales de l'armoire (1960).</p>



COMMISSION  
DES COMMUNAUTÉS  
EUROPÉENNES

Direction Générale  
Télécommunications, Industries de l'Information et Innovation

EXPOLANGUES '87, Paris

Exemple de traduction automatique Systran

TEXTE ORIGINAL

It will be particularly important to ensure the consistency of policy in the field of database services with that in other related areas, for example to:

- ensure that the potential economic, social and regional impact are fully taken into account in information market initiatives;
- reinforce the research and development and use of automatic translation facilities;

TRADUCTION BRUTE SYSTRAN

Sarà in particolare importante assicurare la consistenza di politica nel campo di servizi della base di dati con quello in altri settori riferiti, per esempio a:

- assicurarsi che l'effetto economico, sociale e regionale potenziale interamente sia considerato in iniziative del mercato dell'informazione;
- rinforzare la ricerca e lo sviluppo e l'impiego di agevolazioni della traduzione automatica;

- encourage the implementation of international standards in the field of information services.

#### 5.2.2. Policy development process

##### 5.2.2.1. Senior Officials Advisory Group (SOAG)

Some urgent policy objectives have been identified in this document. However, the policy development process is an ongoing one requiring a great deal of consultation, analysis and consensus-forming among the Member States. It is proposed that the focal point for this policy development and review process is the Senior Officials Advisory Group (SOAG) for information market policy. SOAG's responsibilities include the following:

- to advise the Commission on information market policy goals and priorities;

- to help the Commission examine problem areas and issues within the information market policy framework; dell'informazione;

- to aid the formulation of convergent information market policies at national and Community level, avoiding unnecessary duplication of effort.

As legal aspects are an area requiring particularly urgent attention at the present time, it is proposed that the findings of the Legal Observatory expert group currently advising the Commission are submitted also to SOAG as part of the input for policy formulation.

- incoraggiare l'esecuzione di norme internazionali nel campo di servizi d'informazione.

#### 5.2.2. Processo di sviluppo politico

##### 5.2.2.1. Gruppo consultivo dei funzionari (SOAG)

Alcuni obiettivi urgenti della politica sono identificati in questo documento. Tuttavia, il processo di sviluppo politico è in corso che richiede molta consultazione, analisi e che si forma consenso fra gli stati membri. Si propone che il punto focale per questo processo di sviluppo politico e di esame sia il gruppo consultivo dei funzionari (SOAG) per la politica del mercato dell'informazione. Le responsabilità del SOAG includono ciò che segue:

- consigliare la Commissione su scopi e su priorità della politica del mercato dell'informazione;

- aiutare la Commissione ad esaminare settori problematici e questioni nel quadro della politica del mercato

- aiutare la formulazione di politiche convergenti del mercato dell'informazione a livello comunitario e nazionale, prevenzione di doppione superfluo di sforzo.

Poichè gli aspetti legali sono un settore che richiede in particolare l'attenzione urgente attualmente, si propone che i risultati del gruppo di esperti dell'osservatorio legale attualmente che consiglia la Commissione siano presentati inoltre a SOAG come componente dell'input per la formulazione della politica.



COMMISSION  
DES COMMUNAUTÉS  
EUROPÉENNES

Direction Générale  
Télécommunications, Industries de l'Information et Innovation

## EXPOLANGUES '87, Paris

Exemple de traduction automatique Systran

### TEXTE ORIGINAL

#### 1. INTRODUCTION

##### 1.1. Généralités

FSSRS (Farm Structure Survey Retrieval System) est un logiciel qui permet: la consultation, l'extraction et la manipulation des données des enquêtes sur la structure des exploitations agricoles.

Ce logiciel, suite à une demande de la division E4 "Comptes et Structures Agricoles" de l'OSCE, a été conçu par Monsieur Guido Vervaet de l'unité A1 de l'OSCE et réalisé par Messieurs Neal Garlick et Steve Ditchburn de la société "ARONWORTH".

FSSRS est opérationnel depuis le début de 1984 et tourne actuellement sur un ICL 39/80.

La gestion de la base et l'aide aux utilisateurs sont assurées par M. LENTSCHAT Jean Marie (OSCE-E4, JMO B3/010, BAT. JEAN MONNET, LUXEMBOURG, Tél. 4301-2064).

L'utilisation de FSSRS est basée sur un système de menus hiérarchisés d'un usage assez aisé. L'utilisateur a la possibilité de travailler en Français, Anglais ou Allemand. Ce manuel de consultation et le manuel "Contenu de la base" ne sont disponibles qu'en Français et en Anglais; cependant si la demande existe ils pourraient être traduits en Allemand.

### TRADUCTION BRUTE SYSTRAN

#### 1. INTRODUCTION

##### 1.1. General information

FSSRS (Farm Structure Survey Retrieval System) is a software which allows: the consultation, the retrieval and the manipulation of the data investigations about the structure of the farms.

This software, further to a request of the division E4 "agricultural accounts and structures" of the SOEC, was conceived by Mr Guido Vervaet of the unit A1 of the SOEC and carried out by Messrs Neal Garlick and Steve Ditchburn of "ARONWORTH" company.

FSSRS is operational since the beginning of 1984 and turns currently on ICL 39/80.

The management of the base and the aid for users is ensured by Mr LENTSCHAT Jean Marie (OSCE-E4, JMO B3/010, BAT. JEAN MONNET, LUXEMBOURG, tel. 4301-2064).

The use of FSSRS is based on a system of hierarchised fragments rather easy use. The user has the possibility of working as Frenchmen, Englishmen or German. This consultation handbook and the handbook "contents of the base" is available only as Frenchmen and as Englishmen; meanwhile if the request exists they could be translated into a German.

## Documentation

Le présent manuel contient uniquement les explications sur les facilités d'utilisation de FSSRS. Pour savoir quelles sont les données contenues dans FSSRS l'utilisateur doit se reporter au manuel "FSSRS contenu de la Base", et pour une bonne interprétation de ces données il est indispensable de se référer aux publications suivantes:

Pour l'enquête structure agricole 1975:

- Enquête Communautaire sur la structure des exploitations agricoles 1975 Volume I "Introduction et bases méthodologiques" (Luxembourg 1978)

Pour l'enquête structure agricole 1977:

- Introduction: Doc. D/SB/317 (disponible sur demande auprès de l'EUROSTAT, division E4)

Pour l'enquête structure agricole 1979/80:

- Enquête communautaire sur la structure des exploitations agricoles 1979/80 Volume I "Introduction et bases méthodologiques"

Pour l'enquête structure agricole 1983:

- Enquêtes communautaires sur la structure des exploitations agricoles Volume I "Introduction et bases méthodologiques"

N.B.: Ce volume sera également valable pour les enquêtes de 1985 et 1987.

## 1.3. Autorisation et sécurité d'accès

Les données contenues dans FSSRS sont publiques, mais le logiciel ne permettant actuellement que l'accès à la base via des terminaux "mode page" connectés au site central, seuls les utilisateurs internes aux institutions européennes ont la possibilité de consulter la base.

## Documentation

This handbook only contains the explanations on the uses of FSSRS. to know what are the data contained in FSSRS the user has to refer to the handbook "contained FSSRS of the base", and for a good an interpretation of these data it is essential to refer to the following publications:

For the farm structure survey 1975:

- Community survey on the structure of agricultural holdings 1975 volume I "introduction and methodological bases" (Luxembourg 1978)

For the farm structure survey 1977:

- Introduction: Doc. D/SB/317 (available on request to EUROSTAT, division E4)

For the farm structure survey 1979/80:

- Community survey on the structure of agricultural holdings 1979/80 volume I "introduction and methodological bases"

For the farm structure survey 1983:

- Community surveys on the structure of agricultural holdings Volume I "methodologiques introduction and bases"

N.B.: this volume will be also valid for the surveys of 1985 and 1987.

## 1.3. Access authorization and security

The data contained in FSSRS are public, but software the not allowing currently that access to the base via terminals "page mode" connected to the central site, only the users within the European institutions has the possibility of consulting the base.

b) Programme de la Présidence luxembourgeoise

Le 10 juillet 1985, M. Jacques POOS, ministre des affaires étrangères luxembourgeois et Président en exercice du Conseil, a présenté devant l'Assemblée plénière le programme de la Présidence luxembourgeoise. En évoquant les principaux axes sur lesquels devrait se développer l'action du Conseil au cours du semestre de la Présidence luxembourgeoise, il a indiqué que la véritable priorité restait la lutte contre le chômage, notamment par un redressement de l'économie communautaire garantissant une croissance plus dynamique et créatrice d'emplois. A ce titre, il a particulièrement mis l'accent sur la nécessité d'aller de l'avant dans la réalisation d'un marché unique, la convergence des politiques économiques et la coopération technologique.

Quant à la consolidation de l'acquis communautaire, il a énuméré les dossiers que la Présidence se proposait de faire progresser, à savoir ceux afférents aux politiques sociales, sidérurgique, énergétiques, des transports, de protection de l'environnement et des consommateurs, ainsi que la poursuite de la réflexion sur la politique agricole commune.

b) Programme of the Luxembourg presidency

10 July 1985, Mr Jacques POOS, Luxembourg Foreign Minister and President-in-Office of the Council, presented before the plenary assembly the programme of the Luxembourg presidency. By evoking the principal axes on which should develop the action of the Council during the six-month period of the Luxembourg presidency, it showed that the genuine priority remained the fight against unemployment, in particular by a recovery of the Community economy guaranteeing more dynamic and creative growth of jobs. In this connection, it particularly laid the stress on the need to go of before in the realization of a single market, the convergence of the economic policies and technological cooperation.

As for the consolidation of the 'acquis communautaire', it listed the files that the presidency proposed making progress, namely that related to policies the social, iron and steel, energy, of transport, of protection of the environment and of the consumers, as well as the continuation of the reflexion on the Common Agricultural Policy.



COMMISSION  
DES COMMUNAUTÉS  
EUROPÉENNES

Direction Générale  
Télécommunications, Industries de l'Information et Innovation

EXPOLANGUES '87, Paris

Exemple de traduction automatique Systran

TEXTE ORIGINAL

In the discussions the proposal has been watered down, removing the principle of free passage and largely excluding Greece, Ireland and the United Kingdom from the effects of the directive. This is achieved by distinguishing between land borders - where spot checks are duly "introduced" (for the greater part in confirmation of existing bilateral or multilateral agreements) - and ports and airports where Member States are merely exhorted to take all possible steps to ease controls.

TRADUCTION BRUTE SYSTRAN

In den Verhandlungen ist der Vorschlag, das Prinzip freier Durchfahrt und mit Ausnahme Griechenlands, Irlands und des Vereinigten Königreichs von den Auswirkungen der Richtlinie grosstenteils entfernend verwässert worden. Wird dieses durch Unterscheidung zwischen Grenzen des Landes - wo Stichproben ordnungsgemäss "eingeführt werden erreicht" (für den grösseren Teil in Bestätigung existierender bilateraler oder multilateraler Uebereinstimmungen) - und Häfen und Flughäfen, wo die Mitgliedstaaten bloss ermuntert werden um alle möglichen Schritte zu unternehmen um Kontrollen zu erleichtern.

b) The first obstacle to adoption of the directive is the Danish insistence on retention of the benefits of the passport agreement under which it is possible to move between the Nordic states without a passport. The Nordic states guarantee to each other effective controls at their external borders (in the case of Denmark the Danish-German border) so that the passport agreement is incompatible with the draft directive which would require easing of checks at the Danish-German border. The Internal Market Council of 23rd June 1986 invited the Commission to hold exploratory talks to see whether the Nordic states would be interested in negotiating an agreement with the Community on the easing of border checks. Lord Cockfield reported to the Internal Market Council of 1st December 1986 on the talks and suggested that the Commission continue to seek the conditions under which reciprocal benefits could be extended.

b) Das erste Hindernis für Annahme der Richtlinie ist das dänische Beharren auf Beibehaltung der Vorteile der Passübereinstimmung, unter der es möglich ist zwischen die nordischen Staaten ohne einen Pass sich zu bewegen. Die nordischen Staaten garantieren miteinander wirksame Kontrollen an ihren äusseren Grenzen (im Falle Dänemarks die dänisch-deutsche Grenze), so dass die Passübereinstimmung inkompatibel mit dem Richtlinienentwurf ist der Erleichterung von Kontrollen an der dänisch-deutschen Grenze erfordern würde. Der Rat für Fragen des Binnenmarktes vom 23. Juni 1986 forderte die Kommission auf, informatorische Besprechungen zu halten, zu sehen, ob die nordischen Staaten an Verhandlung eines Abkommens mit der Gemeinschaft auf der Erleichterung von Grenzkontrollen interessiert würden. Lord Cockfield berichtete dem Rat für Fragen des Binnenmarktes vom 1. Dezember 1986 auf den Besprechungen und schlug vor, dass die Kommission fortfährt die Bedingungen zu suchen unter denen gegenseitige Vorteile ausgedehnt.

He called on the Council to adopt the proposal with a temporary derogation for Denmark covering the period of talks.

Nach der Diskussion identifizierte der Vorsitzende wie folgt in das Protokoll aufgenommene Erklärungen des Kompromisses:

After discussion the President identified compromise minutes declarations as follows: Delegations did not comment on these declarations, which will be submitted to the Council, except that the Commission and Netherlands representatives had technical reserves on the 18 month definition of old butter, and the Netherlands representative resisted the

Die Delegationen nahmen nicht zu diesen Erklärungen Stellung, die dem Rat unterbreitet werden werden, ausser dass hatten die Kommission und die Vertreter der Niederlande technische Vorbehalte betreffend der 18-monatigen Definition alter Butter, und der Vertreter der Niederlande widersetzte sich der Beschränkung der Operationen zu jenen die im erläuternden Memorandum erklärt wurden. restriction of the operations to those stated in the explanatory memorandum.

e) The Irish representative complained that the provisions on interest rates were unfair to countries with high domestic rates, but said that this should not prevent agreement on the regulation. Mr O'Leary considered there was an element of double funding as the sales of concentrated butter were already funded under the co-responsibility levy. He noted the statement of the Commission representative that the estimated disposal costs included any export refunds which would be payable.

e) Der irische Vertreter beklagte, dass die Bestimmungen zu Zinssätzen unfair zu den Ländern mit hohen einheimischen Quoten waren, aber sagte dass dieses keine Uebereinstimmung zur Verordnung verhindern sollte. Herr O'Leary erwog, es gäbe ein Element doppelter Finanzierung da die Verkäufe konzentrierter Butter schon unter der Mitverantwortungsabgabe finanziert wurden. Er nahm die Erklärung des Kommissionsvertreters zur Kenntnis, dem die geschätzten Beseitigungskosten alle Ausfuhrerstattungen einbezogen die zahlbar sein würden.

- 23 -

396 - 23 -

l'analyse de l'essai CABRI B1 était difficile en raison des effets bidimensionnels du mouvement et de la perturbation apportée par la cage de centrage. Le code a retrouvé l'instant et l'endroit où s'amorce le mouvement et le fait que celui-ci est d'abord ascendant. La propagation du front de fusion et la hauteur de la zone fondue ont été calculées à 7 % près. De plus comme dans l'expérience on constate que la relocalisation se fait à l'intérieur de la zone fondue, c'est-à-dire dans la partie fissile (faible pénétration de l'acier liquide sur la gaine intacte) et que les centrages constituent une zone de resolidification préférentielle. Par contre le calcul n'a pas retrouvé la fusion des centrages qui a pourtant été observée. Par ailleurs on peut vérifier sur la figure 32 que si l'emplacement de certains bouchons est correct, d'autres n'ont pas été retrouvés; mais ceci est lié à des particularités de l'expérience qui ne se retrouveront pas en réacteur. On peut donc considérer que ces essais ont permis de qualifier le code ALFA; des essais complémentaires permettront de préciser l'effet des centrages (essais MOHOBRI) et de l'irradiation (essais CABRI). Par ailleurs ils ont permis d'établir plusieurs points dont nous verrons l'importance pour l'application au réacteur:

397 The analysis of the CABRI test B1 was difficult owing to the two-dimensional effects of the movement and of the perturbation brought by the cage of centering. The code  
398 found the moment and the place when starts the movement and the fact that this one is initially ascending. The  
399 propagation of the front of fusion and the height of the melted zone were calculated with a margin of 7 %. Moreover  
400 as in the experience it is noted that the relocalisation is done inside the melted zone, i.e. in the fissile part (small penetration of molten steel on the intact clad) and that the centering devices constitute a preferential resolidification zone. On the other hand calculation did not find the fusion  
401 of the centering devices who however was observed. In  
402 addition one can check on figure 32 only if the site of some plugs is correct, others were not found; but this is  
403 connected with characteristics of the experience who will not be found in reactor. One thus can consider that these  
404 tests allowed to qualify the ALFA code; complementary tests will make it possible to specify the effect of the centering  
405 devices (MOHOBRI tests) and of the irradiation (CABRI tests). In addition they allowed to draw up several points  
406 of which we will see the importance for the application to the reactor:

La qualité des traductions peut être évaluée à partir des indications suivantes :

### SYSTRAN 3.7 : QUALITE DE TRADUCTION

couple de langues	date	total des mots	pourcent. d'erreurs	qualité générale de la traduction	TAILLE DES DICTIONNAIRES DE SYSTRAN	
					total des entrées	total des significations
anglais espagnol	25/2/88	6.545	12.3	87.7	154.135	44.912
anglais français	25/2/88	6.545	11.1	88.9	154.135	154.538
anglais italien	25/2/88	6.545	29.0	71.0	154.135	142.397
anglais russe	01/06/87	6.545	32.8	67.2	19.329	34.773
anglais portug.	01/06/87	6.545	29.6	70.4	154.135	31.918
anglais allemand	25/2/88	6.545	29.8	70.2	154.135	124.245
anglais japonais	25/2/88	6.517	30.3	69.7	154.135	66.481
allemand espagnol	01/06/87	5.405	40.4	59.6	126.617	47.902
allemand français	01/06/87	5.405	67.6	32.4	126.727	28.610
allemand italien	01/06/87	5.405	80.2	19.0	125.426	17.736
français allemand	01/06/87			expérim.	62.994	47.398
russe anglais	01/06/87	8.654	4.6	95.4	346.018	487.585
français anglais	01/06/87	8.344	13.7	86.3	117.401	136.367
allemand anglais	01/06/87	5.405	20.9	79.1	129.768	159.258
japonais anglais	01/06/87	9.140	33.0	67.0	45.394	56.523
espagnol anglais	01/06/87	9.017	23.0	77.0	32.669	36.439
italien anglais	01/06/87			expérim.		
portugais anglais	01/06/87			expérim.		

## SYSTRAN 3.7 : QUALITE DE TRADUCTION (suite)

couple de langues	CATEGORIE D'ERREUR et POURCENTAGE D'ERREURS PAR CATEGORIE						
	problèmes de sens	mots non trouvés	POS incorr. (homograph)	forme incorrecte	relations syntaxiques	ordre des mots	autres
anglais espagnol	28.0	26.9	11.8	21.1	1.5	9.9	0.8
anglais français	28.6	20.2	11.7	7.0	13.9	6.0	12.6
anglais italien	41.0	18.5	12.9	7.0	14.5	2.8	3.4
anglais russe	12.2	23.2	12.7	41.3	5.5	1.0	4.1
anglais portug.	40.3	21.6	6.5	22.3	8.7	0.1	0.5
anglais allemand	31.2	21.5	11.9	10.3	3.5	8.1	13.5
anglais japonais	32.9	11.2	20.1	3.7	18.4		13.7
allemand espagnol	37.9	55.7	0.3	4.8	4.3	1.3	1.5
allemand français	9.9	75.3	1.2	4.1	0.4	3.7	0.3
allemand italien	1.0	95.1	0.3	2.2	14.0	0.7	12.5
français allemand							
russe anglais	26.0	0.5	2.5	3.3	27.3	17.3	23.1
français anglais	47.6	6.2	15.0	8.0		6.5	
allemand anglais	31.3	25.7	6.5	4.3	4.7	5.7	12.0
japonais anglais	36.4	4.6	1.4	6.4	18.4	14.0	17.3
espagnol anglais	41.0	18.0	17.0	14.0	6.0	2.5	1.5
italien anglais							
portugais anglais							

## TESTS DE QUALITE DE SYSTRAN

N°		% de qualité
1	russe-anglais	95.4
2	anglais-français	88.9
3	anglais-espagnol	87.7
4	français-anglais	86.3
5	anglais-arabe	83.0
6	allemand-anglais	79.1
7	espagnol-anglais	77.0
8	anglais-italien	71.0
9	anglais-portugais	70.4
10	anglais-allemand	70.2
11	anglais-russe	67.2
12	japonais-anglais	67.0
13	allemand-français	*
14	allemand-italien	*
15	anglais-néerlandais	*
16	français-néerlandais	*
17	français-allemand	*
18	français-italien	*

Ces critères ont été établis par la Sté LATSEC pour le service réception de ses clients.

SYSTRAN fonctionne sur un IBM 360/50. Un système opérationnel doit être disponible sur l'ordinateur AMDHAL de la Commission, (sous MVS). Une adaptation du logiciel à l'environnement UNIX est à l'étude. Ce système à transfert, basé sur une approche directe est complètement automatisé.

Les performances annoncées sont de 2 millions de mots/ heure de temps calcul. On prévoyait une production de 300 000 pages/an. Or SYSTRAN n'a traduit que 2800 pages en 1987 pour la CEE (Si l'on se livre à un rapide calcul, en comptant 250 mots pour une page type (standard européen), 2800 pages représentent 700 000 mots. Conclusion : SYSTRAN a fonctionné moins de 20 minutes - temps calcul - en 1987, pour traduire les textes de la CEE ! ).

Sa conception modulaire facilite son extension à d'autres couples de langues. Aux Communautés Européennes, il fonctionne sur 12 couples.

Extraits de la documentation SYSTRAN 1990 :

LA QUALITE SYSTRAN

LANGUE SOURCE	LANGUE CIBLE	QUALITE en%
ANGLAIS	Français	89
	Arabe	86
	Espagnol	75
	Italien	72
	Néerlandais	71
	Portugais	70
	Allemand	70
	Russe	65
FRANÇAIS	Anglais	86
	Italien	80
	Néerlandais	75
	Allemand	70
RUSSE	Anglais	96
ALLEMAND	Anglais	80
	Français	70

## EXEMPLE CHIFFRE

CONDITIONS	NOMBRE DE PAGE/HEURE (moyenne)	EXEMPLE: 1 000 PAGES EN HEURES	COUT	
TRADUCTION HUMAINE	1	1 000 H (18 semaines)		250 000
TRADUCTION PAR SYSTRAN	2 000	0H 30	75 000	
REVISION	3	330 heures (6 semaines)	82 500	
TOTAL			157 500	250 000

BILAN PAR SYSTRAN POUR 1 000 PAGES :

- ECONOMIE : 92 500 F
- TEMPS GAGNE : 12 SEMAINES

### 3.3.3.2.10 SYSTRAN Japon

Les systèmes anglais-japonais et japonais-anglais ont été développés par la société IO-NA (responsable du développement et de l'exploitation de SYSTRAN au Japon). Ils sont utilisés par de grandes organisations. La société dispose d'un bureau de traduction à Tokyo qui produit plus de 10 000 pages /an. Les deux logiciels fonctionnent sur un FACOM M-380, à une vitesse de 2 millions de mots/heure de temps calcul.

Exemples de traduction :

シストラン世界大会に出席するために、世界各地から  
シストランに興味を持つ人々がルクセンブルグに集まった。

IN ORDER TO ATTEND THE WORLD SYSTRAN CONFERENCE,  
PEOPLE WHO ARE INTERESTED IN SYSTRAN GATHERED IN  
LUXEMBOURG FROM DIFFERENT PARTS OF THE WORLD.

本  
(HON)



A BOOK, BOOKS, THE BOOK, THE BOOKS

沢山の本  
(TAKUSAN NO HON)



MANY BOOKS

PEOPLE INTERESTED IN SYSTRAN GATHERED HERE TODAY FROM ALL OVER THE WORLD

1 2 3 4 5 6 7 8 9 10 11 12



世界中から今日ここにシストラに興味を持つ人々が集まった。

SEKAI JU KARA KYO KOKONI SISUTORAN NI KYOMI WO MOTU HITOBITO GA ATUMATTA

12 9 8 7 6 4 3 2 1 5

I WILL TAKE MY SISTER WITH ME.



連れていく。

(TSURETEIKU)

I WILL TAKE MY CAMERA WITH ME.



持っていく。

(MOTTEIKU)

### 3.3.3.2.11 TAURAS<sup>1</sup>

Lancé sur le marché en 1987 par la société TOSHIBA (Japon), TAURAS (Toshiba Automatic Translation System Reenforced by Semantics) est un système multilingual qui fonctionne pour l'instant sur le couple anglais-japonais et très bientôt sûr le couple japonais-anglais.

---

(1) S. MELI : "Informationsmarkt der maschinellen Übersetzung" in *Terminologie et Traduction*, Commission des Communautés Européennes, n°3, 1989, pp. 83-84

L'analyse syntaxique est basée sur une grammaire d'ATNs, (réseaux de transition augmentés, paragraphe 2.7.4.3). L'analyse sémantique repose sur des règles sémantiques associées à des règles lexicales dans le dictionnaire, au niveau duquel syntaxe et sémantique fonctionnent de façon complémentaire : si le traitement sémantique échoue, on soumet à nouveau la phrase à l'analyse syntaxique.

Le logiciel fonctionne sur un MC 68020 sous le système d'exploitation UNIX. Il est écrit en langage C. C'est un système de transfert qui traduit les textes techniques ou scientifiques, de l'anglais vers le japonais et dispose d'un dictionnaire de 50 000 entrées, 50 000 termes techniques auxquels il faut ajouter les 30 000 termes du dictionnaire de l'utilisateur. Il traduit à une vitesse de 7 000 mots/heure.

### 3.3.3.2.12 WEIDNER<sup>1,2</sup>

La Weidner Communications Corporation créée en 1977 par Bruce WEIDNER a développé avec des chercheurs qui avaient travaillé sur le projet BYU (paragraphe 3.3.2.5) le système anglais-français pour le MITEL (Canada) et le système anglais-espagnol pour SIEMENS (Etats-Unis).

En 1981 le MITEL utilise le couple anglais-espagnol et anglais-allemand, BRAVICE<sup>4</sup> (bureau de traduction à Tokyo) acquiert le système anglais-espagnol, espagnol-anglais. En 1989, 25 systèmes WEIDNER sont installés dans le monde. Le processus de traduction (méthode directe) est totalement automatisé, peu évolué (analyse locale), mais le système est vendu comme un système d'aide à la traduction qui fonctionne en mode interactif (pré-analyse lexicale, construction de dictionnaires...) et intègre un traitement de texte avec des interfaces variées. Une des caractéristiques du logiciel est d'accepter un texte dans n'importe quel format et de le conserver pour l'impression de la sortie. Contrairement aux autres systèmes de T.A.O., il ne nécessite pas de pré-édition.

On distingue quatre étapes dans la traduction, indépendantes les unes des autres.

- Pré-analyse : Repérage des phrases, insertion des marqueurs de mise en forme, identification des commandes de composition, analyse morphologique des mots du texte source (suppression des inflexions, repérage des racines), désambiguïsation des homographes.
- Isolement des groupes de mots qui fonctionnent comme une même entité linguistique (compactage), recherche des syntagmes verbaux... Ces groupes sont considérés comme des éléments uniques et on leur associe des informations syntaxiques et sémantiques pour une analyse ultérieure.
- Analyse syntaxique, effectuée de gauche à droite. Les informations relatives à la structure de la phrase sont conservées pour une réorganisation ultérieure dans la structure de la langue cible.
- Transfert : la structure de la langue source est transformée en structure de la langue cible. Un programme d'insertion insère le texte selon la structure de la langue cible, les articles, les prépositions. Les mots et expressions sont remis à leur place et le texte mis en forme. Le programme d'extension transforme les expressions compactées en les plaçant dans la structure de la langue cible.
- La synthèse restitue les marques de déclinaison et ajuste l'orthographe.
- Le post-traitement rétablit la mise en page et les formats et traite les mots qui n'ont pas été traduits. La traduction est stockée dans un fichier de sortie.

---

(1) Secretary of State of Canada : Essai de traduction assistée par ordinateur : système WEIDNER Approvisionnements et Services Canada, Project 5-5462, 1985

(2) M.G. HUNDT : "Working with the Weidner Machine-Aided Translation System", in V. LAWSON (Ed.), 1982, *Practical Experience of Machine Translation*, Conference in London, novembre 1981, pp. 45-51, Amsterdam, North-Holland

(3) TAKEHIKO YAMAMOTO : "Le pari du dictionnaire universel" in *Dynasteurs*, novembre 1987

- L'édition conduit à la traduction définitive. Le système met à la disposition du traducteur un ensemble de fonctions de révision : l'AMENDER (fonctions de traitement de texte, affichage des textes source et cible sur écran partagé, ajout ou suppression de mots ou expressions, dictionnaire des synonymes).

On constate que l'ordre dans lequel interviennent les étapes n'est pas compatible avec une traduction de qualité, seule la possibilité de traduire 800 mots à l'heure justifiant l'utilisation du système. Un affichage des menus assure la convivialité du système. Une option "recherche du vocabulaire" améliore la traduction. Le programme recherche les mots du texte qu'il ne comprend pas pour les proposer à l'utilisateur sous plusieurs formes (liste alphabétique, liste par fréquence d'apparition, liste avec contexte). Un module de mise à jour du dictionnaire permet à l'utilisateur de personnaliser le système.

Après avoir développé les couples anglais-allemand et allemand-anglais pour ITT (Grande-Bretagne) en 1982, WEIDNER a travaillé avec la compagnie BRAVICE (Japon) pour la mise au point des couples japonais-anglais, anglais-japonais.

Les programmes écrits en FORTRAN ont été implantés sur microordinateur (IBM PC...) pour s'affranchir des gros calculateurs. Les produits ont alors été commercialisés par la société TAO International sous le nom de MICROCAT (IBM XT et compatibles) et MACROCAT (PDP 11 ou VAX, Microvax II de DEC). Ces produits semblent avoir disparu du marché nord-américain et européen. Seule la société BRAVICE continue de l'exploiter au Japon.

# MicroCat

At last... foreign language translation-on a personal computer!

Now computer-aided translation is practical and affordable for organizations or individuals with moderate translation demand.

The MicroCat from TAO International is a personal computer-based foreign language translation system. It is designed for use by one translator working in one language direction, such as English to Spanish. In addition, when the MicroCat is not in use for its translation application, it can be used for word processing, financial spread sheets, and other personal computer programs.

#### Protect your information.

For those who must translate proprietary text, the Microcat system guarantees complete internal security. You control sensitive information at all times.

#### MicroCat is IBM compatible.

The MicroCat system runs on an enhanced IBM XT Personal Computer and will accept all compatible software.

#### Speed, Productivity, and Savings.

The MicroCat can translate at speeds of up to 3000 or more words per hour. Rapid rough translations are highly accurate and relieve the translator of a major, tedious task. Since translators spend their valuable time editing and polishing the text stylistically, their productivity goes up and your costs go down.

#### Expandable Word Bank Improves Technical Translations.

For each language direction, a core dictionary of about 9000 words is built into the MicroCat system. In addition, a Dictionary Update function allows you to add or delete words or phrases in seconds. With this powerful MicroCat feature, you can add to the system dictionary words and idioms unique to your international communications. Technical documents such as training manuals, scientific reports, and parts lists are handled more easily than ever before.

#### On-screen editing with Multilingual Word Processor

The translator's job is made even easier because the MicroCat system displays on a split screen both the source and target languages. And its built-in multilingual word processing capability, with an international character set, provides powerful text manipulation capability, including Word Swap, Phrase Movement, and all other standard word processing functions.

#### MicroCat specifications.

##### Language Directions: (Available or being developed).

English to French,  
French to English,  
English to Spanish,  
Spanish to English,  
English to German,  
German to English,  
English to Portuguese.

##### Hardware:

Enhanced IBM XT Personal Computer,  
640 KB Memory,  
10 MB Disk Drive,  
NEC 3550 Letter Quality Printer  
International Character Set  
Key Template.

Foreign language translation: in the international marketplace, how well you do it can determine how well you do. TAO International Can Help You Do It Better.

# MacroCat

The best computer-aided translation system on the market today.

For organizations with a heavy demand for foreign language translations. The MacroCat system from TAO International provides speed, accuracy, and a major reduction in expense.

The MacroCat system is a multi-user, multi-language translation system. It is designed for use by multiple translators working in one or more language directions simultaneously. This MacroCat system handles all major European language. Other language directions are currently in development.

Eight thousand words per hour.

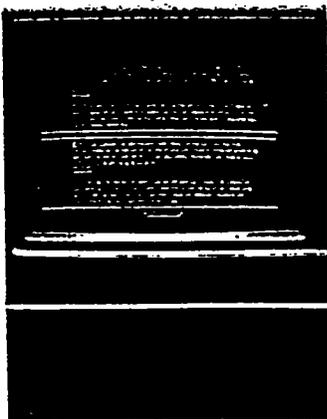
There is no faster translation system. The MacroCat produces highly accurate rough translations at speeds of up to 8000 words per hour. For high volume users, the MacroCat is the key to translations that are fast, accurate, consistent, and economical. Because the MacroCat system produces high speed rough translations, your translators are spared that tedious and time consuming task. They spend their valuable time editing and polishing stylistically, producing precise final text. Productivity goes up. Costs go down.

No text is too difficult.

The MacroCat, with its Dedicated Word Bank, can translate even the most technical literature and it can handle great volume with relative ease. The more comprehensive your customized dictionary, the faster and more accurate are the rough translations. Technical documents such as training manuals, scientific reports, and parts lists are handled more easily than ever before.

Protect your information.

For those who must translate proprietary texts, the Macrocat system guarantees complete internal security. You control sensitive information at all times.



### On-screen editing with Multilingual Word Processor

The translator's job is made even easier because the MacroCat system displays on a split screen both the source and target languages. And its built-in multilingual word processing capability, with an international character set, provides powerful text manipulation capability, including Word Swap, Phrase Movement, and all other standard word processing functions.

Foreign language translation: in the international marketplace, how well you do it can determine how well you do. TAO International Can Help You Do It Better.

### Your Dedicated Word Bank.

For each language direction, the MacroCat system includes a core dictionary of about 9000 most commonly used words and phrases. In addition, will help you build, or build for you, a totally customized dictionary of words and phrases common to your organization or industry. Since adding a new word or idiom takes only seconds, your word base is easily and continuously expandable. This customized dictionary—your Dedicated Word Bank—is an exclusive feature of the CAT system. It is the key to consistent translations of even the most difficult text.

### MacroCat specifications.

#### Language Directions:

(Available or being developed).

- English to French,
- French to English,
- English to Spanish,
- Spanish to English,
- English to German,
- German to English,
- English to Portuguese.

#### Hardware:

PDP or VAX series from DEC.

To contact: TAO International  
Call: (61) 22.87.77 — Telex: 520.379

### 2.3.3.3 Les systèmes dédiés

Ces systèmes ne sont pas commercialisés. Ils ont été conçus pour des applications précises et sont exploités en général par leurs concepteurs.

#### 2.3.3.3.1 AMPAR<sup>1</sup>

Ce système de traduction "directe" est bilingue (anglais-russe). Développé à Moscou (All-Union Centre for Translation of Scientific and Technical Literature and Documentation), il fonctionne depuis 1979. Conçu à l'origine pour la traduction de texte politique, le centre l'utilise dorénavant pour traiter les domaines de l'informatique et de la programmation. Il peut être adapté à d'autres types de littérature scientifique et technique. Tout comme NERPA (paragraphe 2.3.3.3.5), il est présenté comme "multifonctionnel", c'est à dire qu'il peut être adapté à des documents de thèmes divers, de forme variable en entrée et propose une édition interactive, ce qui facilite la recherche dans les dictionnaires et autorise une correction et une actualisation rapides.

#### 2.3.3.3.2 CHIMKENT<sup>1</sup>

Développé au CHIMKENT Pedagogical Institute for the Kazakh Academy of Sciences, le système traduit des textes sur la chimie et les polymères de l'anglais vers le russe. Il ne s'agit en fait que d'une traduction mot à mot.

#### 2.3.3.3.3 CULT (Chinese University Language Translator)<sup>2</sup>

Utilisé pour assurer la traduction chinois-anglais des Acta Mathematica Sinica (1975) et des Acta Physica Sinica (1976), CULT est un des systèmes de Traduction Assistée par Ordinateur les plus intéressants. Il a été développé par l'université chinoise de Hong Kong à partir de 1968. Après avoir fonctionné sur un texte introduit sous forme de cartes perforées, le système a été réactualisé en 1978, avec l'ajout d'un module de traitement de texte.

La préédition est une étape très importante pour ce système à approche directe qui ne fonctionne que sur les textes cités plus haut et pour le couple chinois-anglais. La post-édition se limite à la réintroduction de formules et de figures dans le texte de sortie. L'intervention du traducteur est requise en cours de traitement, pour la conjugaison des verbes et la déclinaison des noms.

#### 2.3.3.3.4 FRAP<sup>1</sup>

Développé dans le même centre que le système AMPAR, il est en exploitation depuis 1981. Système à transfert, il traite des textes d'électronique, d'informatique, de navigation aérienne et de construction aéronautique, du français vers le russe.

#### 2.3.3.3.5 NERPA<sup>1</sup>

Développé par le même centre, ce système repose sur les mêmes principes linguistiques qu'AMPAR et fonctionne sur le même environnement informatique. Il traduit de l'allemand vers le russe des textes d'informatique et de programmation.

(1) S. MELI : "Informationsmarkt der maschinellen Übersetzung" in *Terminologie et Traduction*, Commission des Communautés Européennes, n°3, 1989, pp. 83-84

(2) S.-C. LOH, L. KONG : "Automatische Übersetzung chinesischer wissenschaftlicher Zeitschriften", *Dritter Europäischer Kongress über Dokumentationssysteme und Dokumentationsnetze*, luxembourg, 3/6 mai 1977, vol. 1, pp. 563-578

### 2.3.3.3.6 SPANAM et ENGSPAN<sup>1</sup>

En 1975 la Pan American Health Organization (Washington D.C.) a entrepris le développement d'un système de traduction automatique pour les quatre langues officielles anglaises, espagnol, français et portugais de l'organisation. Le système SPANAM traduit de l'espagnol vers l'anglais depuis 1980, ENGSPAN de l'anglais vers l'espagnol depuis 1984.

Les programmes (écrits en PL/1) fonctionnent sur ordinateur IBM 4381, dans l'environnement DOS/VSE/SP. Une autre version est compatible avec l'IBM 3081 (OS/MVS). Les textes sont soumis et traités en sortie par une station dédiée au traitement de texte (WANG OIS/140), le traitement est automatique et ne fonctionne qu'en mode "batch". Il n'y a aucune restriction quant au domaine abordé par les textes à traduire et à la syntaxe utilisée, (la plupart du temps, médecine et santé). Des traducteurs entraînés au maniement des outils dont ils disposent, révisent les traductions à l'aide du traitement de texte. La production est de deux à trois fois plus importante que lors d'une révision manuelle (4000-10000 mots/jours pour la post-édition, 1500-3000 mots/jour pour une traduction manuelle).

La vitesse de SPANAM sur gros ordinateur atteint 1500 mots/minute (495 000 mots/heure de temps CPU) et ses dictionnaires contiennent 61 282 entrées (langue source), 58 485 entrées (langue cible). Pour ENGSPAN, elle atteint 836 mots/minute (102 000 mots/heure de temps CPU), les dictionnaires contiennent respectivement 45 614 et 47 545 entrées. Les dictionnaires occupent chacun de 7 à 10 Mo.

Construits selon la même architecture modulaire, ils se différencient sur le plan de la linguistique, ENGSPAN, plus avancé, utilise les ATNs (paragraphe 1.5.4.3) et sépare les modules de transfert lexical et syntaxique. Le système SPANAM est amélioré au fil des découvertes et des progrès du système ENGSPAN.

#### Environnement

Dans les deux cas, il n'y a pas de *pré-édition* au sens linguistique du terme. Avec SPANAM, le document peut être saisi de façon automatique (Optical Character Recognition), ce qui nécessite alors une révision.

La *post-édition* est par contre indispensable. Confiée à des traducteurs-réviseurs entraînés (variété des applications oblige), elle peut être modulée selon plusieurs critères (destination de la traduction, délais impartis, structure linguistique du texte...) et permet de modifier rapidement les constructions fréquentes qui ne sont pas conformes, grâce à des outils appropriés comme les QFP ("Quick Fix" post-editing expedient) et des utilitaires de traitement de texte pour la recherche et la modification des chaînes de caractères. La philosophie retenue est d'introduire le minimum de correction pour obtenir une traduction acceptable. Réalisée à l'écran, elle est associée au traitement de texte qui automatise les opérations répétitives (fonctions SEARCH and REPLACE, SEARCH and DELETE...). Dans le but d'accélérer le fonctionnement du système, les tâches apparemment simples sont également optimisées. Le positionnement du curseur peut induire une perte de temps considérable de sorte qu'on lui préfère la fonction clavier SEARCH et l'utilisation d'une souris. Les techniques du traitement de texte et de la pré-édition sont mises à la disposition du traducteur pour l'actualisation des dictionnaires.

---

(1) M. VASCONCELLOS : "Management of the Machine Translation Environment : Interaction of functions at the Pan American Health Organization" in V. LAWSON, (ed.), *Practical Experience of Machine Translation*, North-Holland, Amsterdam, 1982, pp. 115-129

Comme pour SYSTRAN, les spécialistes du département des Langues ont la possibilité de trier les textes destinés à la traduction automatique ou manuelle. Une des particularités des systèmes ENGSPAN et SPANAM est leur intégration dans un service complet de traduction, dans le sens où ils peuvent être associés à des postes de travail compatibles et complémentaires. Ils aident alors la traduction humaine en apportant la puissance de leurs microglossaires spécialisés, inversement, des programmes de comptage de mots et des correcteurs orthographiques soulagent l'ordinateur central.

### Fondements des systèmes

Le système SPANAM est au départ un système de traduction directe, avec analyse de la langue source (espagnol), transformation des structures de surface pour obtenir le cadre syntaxique propre à la langue cible (anglais), introduction des termes anglais selon les résultats de l'analyse, insertion et/ou suppression de certains morphèmes grammaticaux, synthèse des terminaisons pour l'anglais. Les principales étapes de la traduction sont : l'analyse morphologique, recherche des unitermes puis des multitermes dans le dictionnaire, résolution des homographes, identification du sujet, traitement des prépositions, des pronoms, des syntagmes verbaux, insertion du sujet, réorganisation des groupes nominaux, recherche du vocabulaire de la langue cible et synthèse.

L'analyse morphologique recherche les mots espagnols (formes radicales), reconnaît les morphèmes du pluriel et du féminin pour les noms, pronoms, déterminants, quantificateurs et adjectifs, détermine la personne et le temps des verbes, traite les composés et la capitale à l'initiale de la phrase. L'utilisateur a la possibilité d'entrer des mots dans le dictionnaire, sous leur forme complète (obligatoire pour les formes irrégulières et les homographes), fléchies ou radicales.

SPANAM traite les ambiguïtés à différents stades du traitement (recherche dans le dictionnaire, examen du contexte). Les différentes classes auxquelles peut appartenir le mot testé sont codées à la suite de l'entrée du dictionnaire, sous la forme d'un ensemble de bits.

Le système ENGSPAN est un système à transfert lexical et syntaxique, avec l'analyse séparée de l'anglais langue source, l'application de routines de transfert basées sur l'analyse contrastive de l'anglais et de l'espagnol, et la syntaxe de la langue cible l'espagnol. Les principales étapes sont : l'analyse morphologique, la recherche des unitermes, l'analyse au niveau de la phrase, le transfert lexical, la recherche des traductions dans le dictionnaire, la transfert syntaxique et la synthèse. Si l'analyse de la phrase échoue, des routines sont activées pour la résolution des homographes, l'analyse des groupes verbaux et des groupes nominaux. L'analyse morphologique (LEMMA) ne démarre que si la forme complète n'a pas été reconnue dans le dictionnaire. La procédure teste la présence de certaines terminaisons, les supprime pas à pas et compare avec le dictionnaire, en s'appuyant sur des règles morphologiques et orthographiques (ajout ou suppression du *e* final selon ce qui précède...), accompagnées d'une longue liste d'exceptions. Les mots non trouvés sont considérés comme des noms communs (noms propres s'ils commencent par une majuscule), verbes ou adjectifs potentiels. Les informations provenant de LEMMA et l'étude de la suffixation confirment ou infirment la nature du mot, suppriment ou ajoutent des possibilités d'ambiguïtés.

Dans ENGSPAN, la résolution des homographes a lieu au niveau de l'ATN. La fonction de chaque mot dépend du chemin emprunté dans le réseau. Pour aboutir à une analyse correcte, l'information lexicale fournie par le dictionnaire est utilisée de trois façons :

- les idiotismes ou les expressions multitermes sont stockées dans un enregistrement unique et n'appartiennent ainsi qu'à une classe.
- les unités analysées peuvent indiquer qu'un groupe de mots est associé à tel type de fonction.
- Un uniterme est associé à un code qui indique la probabilité pour qu'il appartienne à une classe donnée.

## Traitement de la polysémie

SPANAM et ENGSPAN disposent de deux outils pour traiter la polysémie : les microglossaires et les unités de transfert.

Les microglossaires sont des sous-ensembles des dictionnaires (langue source-langue cible) que l'utilisateur peut appeler pour des applications particulières, (loi, agriculture, finance, informatique, ingénierie médicale, recherches biomédicales), au moment de la traduction.

L'unité de transfert est une règle de transfert lexical stockée dans le dictionnaire source, qui spécifie une condition à tester (ce verbe a  $n$  objets) et une action à réaliser (choix d'une traduction, insertion d'une préposition, suppression d'un mot...).

## Les caractéristiques syntaxiques et sémantiques

Elles sont codées sous forme binaire dans chaque enregistrement d'une entrée de dictionnaire et sont utilisées à toutes les étapes de la traduction. Dans ENGSPAN l'analyse produit un graphe de noeuds correspondant à chaque proposition, à chaque phrase. A chaque noeud correspond un tableau contenant les constituants, leur rôle et leur localisation (numéro du mot s'il s'agit d'une unité lexicale simple, pointeur du noeud approprié si c'est une proposition ou une phrase). Chaque noeud contient des règles applicables (type, mode, personne, nombre, temps, aspect, voix...).

Le formalisme des ATNs et la représentation structurale s'inspirent profondément de la théorie des réseaux de transition augmentés et de la grammaire systémique de WINOGRAD (1.4.3.5), reprenant des travaux de W.A. WOODS<sup>1,2</sup> et de R. M. KAPLAN<sup>3</sup>. La grammaire du système ENGSPAN est basée sur 11 réseaux (phrase, proposition, groupe nominal, groupe verbal, nominalisation, composés avec tiret, groupes prépositionnels, comparaison, groupe adverbial, proposition relative, proposition indépendante). Chaque réseau est un ensemble d'états réunis par des arcs (4 types : CATEGORY, JUMP, SEEK et SEND), ce qui représente globalement 150 conditions et 60 actions.

## Aspects informatiques

Les dictionnaires sont stockés sur des disques en ligne. Distincts pour la langue source et la langue cible, les enregistrements d'une longueur fixe de 160 octets sont liés à un nombre de 12 chiffres qui permet d'y accéder. Ils contiennent des mots simples et des groupes de mots.

- *Les unités de substitution* : (de 2 à 5 mots, ex : By leaps and bounds). Si une telle unité est repérée, les enregistrements du dictionnaire correspondant à chaque mot sont remplacés par un seul enregistrement pour toute la séquence. La traduction figure dans un enregistrement unique du dictionnaire cible. Ce type d'unité permet de traduire correctement des noms d'institution, des titres de publication, des idiomes, des expressions techniques...).

- *Les unités d'analyse* : (de 2 à 5 mots) qui n'ont pas de contrepartie dans le dictionnaire cible et indiquent à l'analyseur les limites d'un syntagme, d'une unité d'analyse ou si les mots appartiennent ou non à une unité. (ex : *drinking water* permet d'analyser *the children have been drinking water with a high fluoride content*).

- *Les unités de substitution différée* pour l'analyse des formes verbales disjointes (ex : *look up, carry out...*).

(1) W.A. WOODS : Augmented Transition Networks for Natural Language Analysis, report N C51 to the National Science Foundation, BBN, Cambridge Ma, 1969

(2) W. A. WOODS : An experimental Parsing System for Transition Network Grammars in R. RUSTIN (ed.) : Natural Language Processing, Algorithmics Press, New York, 1971, pp. 111-154

(3) R. M. KAPLAN : A General syntactic Parser in R. RUSTIN (ed.) : Natural Language Processing, Algorithmics Press, New York, 1971, pp. 193-241

- *Les unités de transfert* (dictionnaire source) ne sont utilisées qu'après l'analyse pour indiquer la possibilité de donner plusieurs traductions et effectuer le choix<sup>1</sup>.

### Règles de grammaire

Pour SPANAM, elles sont de deux types :

- *Pattern Matching* : reconnaissance et réorganisation des phrases nominales. Les schémas grammaticaux peuvent être stockés et réactualisés sans que cela n'implique une recompilation du programme.

- *Transformations* : Identification et synthèse des groupes verbaux. Chaque groupe de règles est testé pour chaque phrase. La structure impliquée par chaque règle est comparée avec la chaîne testée. En cas de correspondance, elle est appliquée et entraîne une permutation, une concaténation, une destruction, ou une substitution des mots ou groupes de mots concernés.

Pour ENGSPAN, la configuration du réseau indique les séquences de constituants possibles. La construction des noeuds dans la structure et les fonctions sont déterminées selon les actions représentées par les arcs. Les conditions et les actions sont contenues dans des modules qui font partie du programme compilé.

### Algorithme d'analyse

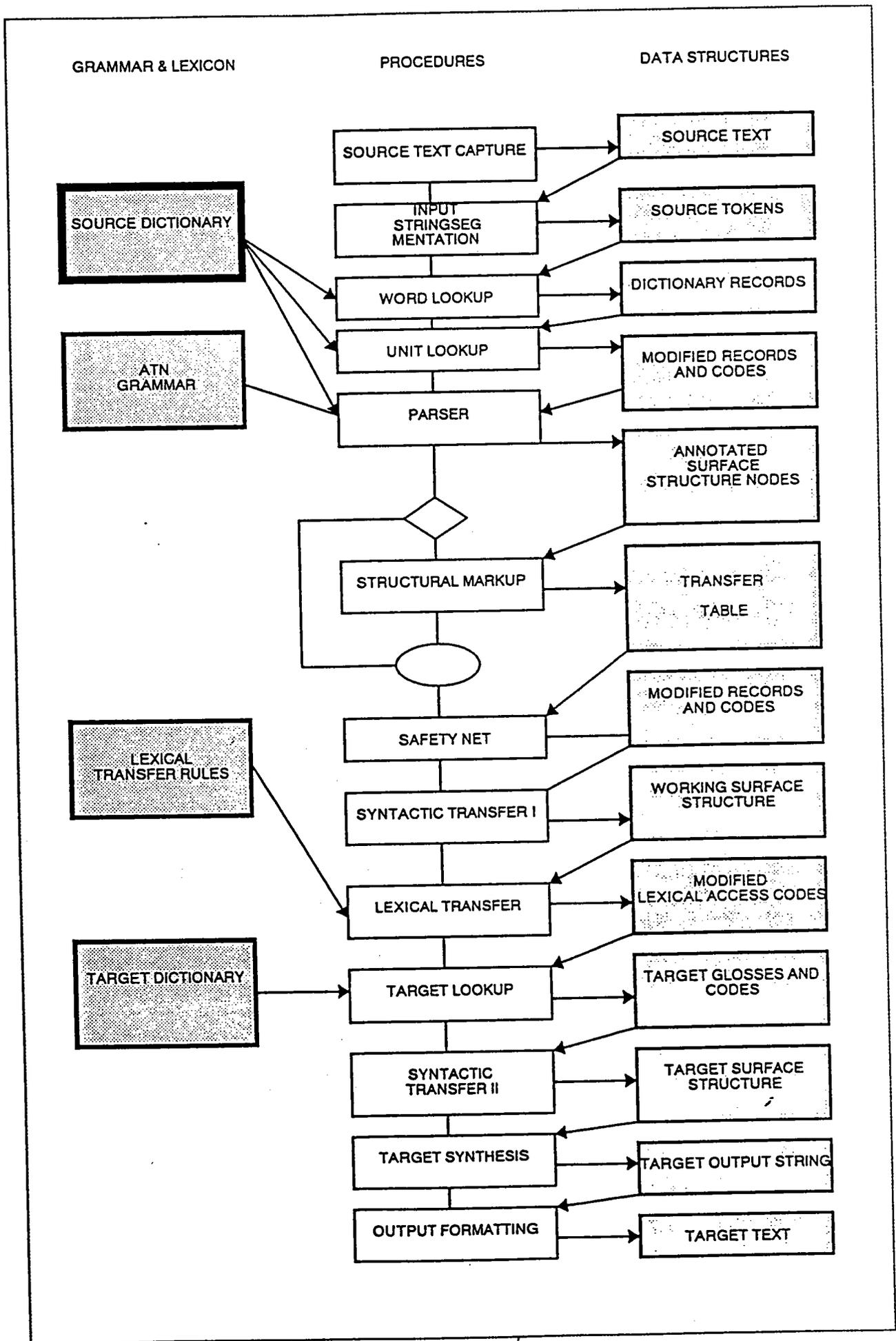
Dans ENGSPAN, l'analyseur fonctionne séquentiellement de haut en bas et de gauche à droite, avec des retours en arrière. Il s'arrête dès qu'il aboutit à une solution. Le parcours effectué dans le réseau dépend des arcs affectés pour chaque état, des informations disponibles sur la structure de la phrase et des codes associés aux entrées de dictionnaire consultées. Dans une première étape, des procédures ont permis de stocker des données sur la ponctuation ainsi que sur la présence de majuscules et de signes diacritiques. L'analyseur traite les mots un par un et teste tous les arcs pour chaque état. Les arcs possibles sont placés sur une pile. L'arc qui est au sommet de la pile est emprunté. L'analyse se poursuit alors, aussi longtemps que se présentent d'autres arcs. S'il n'y a plus d'arc, le parser revient en arrière. L'analyse aboutit lorsque l'on atteint l'extrémité de la chaîne et un état final du réseau. Si l'analyseur ne parvient pas à parcourir le réseau, il y a échec. Pour le traitement des relatives et des interrogatives indirectes, le programme dispose d'une liste qui fonctionne comme une mémoire et dans laquelle il copie la phrase rencontrée. Il utilise cette copie lorsqu'il rencontre la rupture de construction. Les retours en arrière utilisent également ce procédé.

Développé sous la forme d'un module indépendant écrit en PL/1, l'analyseur ATN est compatible avec le système SPANAM. La grammaire de réseau fonctionne avec un run-time, ce qui permet d'expérimenter différents types de réseau sans recompiler le programme. L'analyseur vient d'être incorporé dans le système SPANAM.

La figure suivante illustre les relations entre les composantes principales de l'algorithme de traduction et les structures de données utilisées.

---

(1) M. VASCONCELLOS, M. LEON : SPANAM and ENGSPAN : Machine Translation at the Pan American Health Organization in J. SLOCUM (ed.) : Machine Translation Systems, Cambridge University Press, 1988, p. 211



Afin de livrer une traduction dans tous les cas, y compris lorsqu'une analyse échoue, l'algorithme applique une stratégie de secours ("safety net") qui doit permettre au module de transfert de fonctionner même si les informations que lui fournit le module d'analyse sont incomplètes.

### Module de transfert

Lorsque l'analyse est terminée, l'étape suivante consiste à examiner la structure de surface qu'elle révèle. L'information qui en découle sera utilisée pour la traduction. Les marques de cette structure de surface sont rangées dans une table qui contient des indications sur les fonctions syntaxiques et sémantiques des principaux constituants de la phrase.

Le transfert lexical : Les unités d'analyse (AUS) et les unités de transfert (TUs) déterminent le choix des termes dans la langue cible. Les *unités d'analyse* sont des entrées composées de plusieurs mots, qui figurent dans le dictionnaire source, sans qu'il n'y ait de correspondance avec le dictionnaire cible. Les *unités de transfert* indiquent les traductions possibles d'un mot ou d'un groupe de mots selon le contexte. Elles sont stockées dans le dictionnaire source et n'interviennent qu'une fois l'analyse terminée. Si le contexte l'implique, la clé de correspondance avec le terme de la langue cible est modifiée et le système retiendra la bonne traduction. Le transfert s'accomplit en remplaçant l'entrée de l'anglais source (LEX) par l'équivalent dans le dictionnaire espagnol cible, en tenant compte des codes qui pourraient indiquer des pluriels irréguliers, des particularités d'ordre morphologique ou encore l'existence de plusieurs équivalences et entraîner le recours aux microglossaires.

Le transfert syntaxique : La stratégie retenue est d'utiliser la structure de surface de la phrase source pour obtenir après traitement une phrase grammaticalement et stylistiquement correcte. Le système respecte l'ordre de la phrase dans la mesure du possible. Les règles de transfert syntaxiques sont de trois types, elles peuvent en effet être liées au lexique (cible ou source) ou en être indépendantes, dans ce cas, elles permettent de convertir une structure de surface d'une langue vers l'autre, par transformation.

### Module de synthèse

Lorsque le transfert est terminé, des indications associées à chaque entrées lexicales permettent d'ajouter les morphèmes grammaticaux (terminaisons, inflexions), des suffixes et/ou des préfixes dans certains cas et des mots outils (articles définis, indéfinis, prépositions...). A la synthèse du verbe succèdent ensuite la génération des formes pronominales de surface (réfléchies, objet direct, objet indirect), la correction du groupe nominal (genre et nombre), l'insertion de mots-outils, l'introduction de variantes phonologiques et la mise en place des majuscules et des signes diacritiques.

### Exemples de traduction

Les exemples qui suivent sont extraits de J. SLOCUM : Machine Translation systems, pp. 233-235, Cambridge University Press, 1987.

Les phrases qui ont été complètement analysées sont marquées par "OK", celles qui n'ont été que partiellement analysées par "PP" et celles qui n'ont pu être analysées par "NO".

A l'application d'une règle de transfert correspond "TU" et à l'absence d'un mot dans le dictionnaire "SD".

The mothers who lived within a 5-mile radius were asked to bring their children to the vaccination center.	OK TU TU	A las madres que vivían dentro de un radio de 5 millas se les pidió que trajeran a sus hijos al centro de vacunación.
Often the cold chain is thought to refer only to the refrigeration of vaccine.	OK	A menudo la cadena de frío se piensa que se refiere solamente a la refrigeración de vacuna.
The relationship between dietary fat and mamary carcinogenesis in experimental models will be presented and, finally, the possible relationship between dietary fat and hormones will be discussed.	OK	Se presentará la relación entre grasa en la alimentación y carcinogénesis mamaria en modelos experimentales y, finalmente, se tratará la relación posible entre grasa en la alimentación y hormonas.
A privately organized by publicly funded foundation is responsible for primary health care in the interior of the country where 10% of the population lives.	OK	Una fundación privadamente organizada pero públicamente financiada es responsable de atención primaria de salud en el interior del país donde vive un 10% de la población.
The National Program for Drinking Water Supply has the goal to expand to full coverage the already existing distribution system.	PP	El Programa Nacional para Abastecimiento de Agua Potable tiene la meta para ampliar a cobertura total el sistema de distribución ya existente.
Other analyses will test the epidemiologic association between 1) ethylene oxide exposure and leukemia and 2) *PAH exposure in the cola hydrogenation process and cancer of the respiratory system, urogenital system, and the skin.	SD	NO Otros análisis examinarán la asociación epidemiológica entre 1) la exposición de óxidos de etileno y leucemia y 2) la exposición de PAH en el proceso de hidrogenación de carbón y cáncer del aparato respiratorio, aparato urogenital, y la piel.

Sample Input		ENGSPAN Output
We request that each potential participant send biographical information and a 100-word abstract of the demonstration or paper, so that we can select those who will make the greatest contribution to a useful exchange of information at the symposium.	OK TU	Solicitamos que cada participante potencial envíe información biográfica y un resumen analítico de 100 palabras de la demostración o documento, para que podamos seleccionar los que harán la contribución mayor a un intercambio útil de información en el simposio.
The task of hiring and assigning staff to perform the work is one which must be completed prior to training.	OK	La tarea de contratación y asignación de personal para realizar el trabajo es una que se debe completar anterior a adiestramiento.
Laboratory studies have shown marijuana to impair perceptual and perceptual-motor functions important to driving.	OK TU	Estudios de laboratorio han revelado que marihuana deteriora funciones de percepción y perceptomotrices importantes a conducción.

### 2.3.3.3.7 TAUM (Traduction Automatique de l'Université de Montreal)

Soutenu par le gouvernement canadien, le projet TAUM voit le jour en 1965 à l'université de Montréal. Les programmes de traitement automatique sont rédigés en FORTRAN et tournent sur un CDC 6600 puis sur un CYBER 173.

Après une période de recherche décousue, on lui assigne un objectif en 1965 : traduire automatiquement les bulletins météorologiques de l'anglais vers le français. Le prototype METEO est présenté en 1976. Il sera installé en 1977 et remportera le succès que l'on sait.

Un deuxième objectif est fixé en 1977 : traduire automatiquement des manuels de maintenance aéronautique (90 millions de mots) de l'anglais vers le français. Le système AVIATION s'enrichit d'un module d'analyse sémantique mais les délais impartis et les coûts démesurés conduisent les responsables à abandonner le projet.

#### 2.3.3.3.7.1 TAUM-METEO<sup>1</sup>

Mis au point par le groupe TAUM en 1975-1976, le prototype a été développé dans un langage spécialisé (Systèmes-Q) et commercialisé par la société *Consultants en linguistique computationnelle Ltée* sous le nom de METEO 1, après une année d'importants travaux qui lui ont permis de passer de 40% à 80% des phrases effectivement traduites.

---

(1) M. CHEVALIER, J. DANSEREAU, G. POULIN : *TAUM-MTEO : Description du système*, Groupe TAUM, Université de Montréal, Montréal, 1978

La société *J. Chandioux experts-conseils inc.* développe un nouveau langage de programmation (GramR) et après avoir démontré la faisabilité d'une traduction automatique sur microordinateur, construit un nouveau système, METEO 2, qui remplacera METEO 1 en 1983, parce que plus performant, moins coûteux, plus fiable et plus convivial. Il est actuellement loué par le *Bureau des Traductions* sous une formule clé-en-main (ordinateurs, logiciels, service 24 heures par jour, mise à jour des dictionnaires).

En 1950, le problème de la traduction automatique était assimilé à un simple problème de décryptage. Les systèmes de première génération se composaient d'un énorme dictionnaire et s'appuyaient sur une analyse de contexte (deux mots avant et après le mot courant). Les limites de SYSTRAN sont alors notoires et le rapport ALPAC donne un coup d'arrêt à la recherche sans tenir compte d'une deuxième génération de systèmes qui présentent des caractéristiques fondamentales nouvelles. Ils sont programmés dans un langage évolué accessible aux linguistes, l'analyse de la phrase est complète et ils travaillent en trois étapes (l'analyse, le transfert et la génération) qui correspondent à la compréhension du texte, à la transposition de l'idée et à la formulation du résultat en langue cible.

### Langage de programmation GramR

Fortement influencé par les travaux du GETA (Groupe d'Etudes pour la Traduction Automatique) à Grenoble, ce langage est un transducteur de chaînes d'arborescences paramétrées qui permet au linguiste initié au formalisme de la grammaire transformationnelle développée par CHOMSKY de définir un modèle linguistique en écrivant des règles. Le modèle est compilé puis évalué sur des exemples.

Premier langage de deuxième génération à fonctionner sur microordinateur en 1982, les applications de GramR dépassent le cadre de la traduction automatique, avec l'E.A.O. (Enseignement Assisté par Ordinateur et mise au point de didacticiels), les correcteurs orthographiques (SpellR), les interfaces en langue naturelle (Easy-Dos). Le langage est, de plus, disponible en versions MS-DOS, UNIX 68 000 et VAX VMS.

### Analyse de la phrase

Le texte présente des caractéristiques qui ne facilitent pas la tâche, contrairement à ce que l'on pourrait imaginer : Entièrement en majuscules, pratiquement sans ponctuation. Les mots qui servent de pivot dans une analyse classique sont souvent absents (conjonctions, prépositions, articles...). L'analyseur est de ce fait plus orienté vers la sémantique, ce qui n'est pas une gageure, tant il est vrai qu'une sémantique fermée (météorologie) peut être mieux cernée qu'une sémantique ouverte.

On distingue cinq phases :

- *la pré-édition* : Elimination des redondances ou des mots non significatifs, normalisation de certains énoncés, identification des expressions idiomatiques, développements des abréviations.
- *le dictionnaire* : il associe à chaque mot sa catégorie syntaxique et des informations sémantiques et morphologiques nécessaires au fonctionnement des modules d'analyse suivants. Il contient environ 2000 mots et expressions.
- *l'analyseur* : les groupes nominaux sont repérés et classés en conditions météorologiques, en circonstanciels de temps et circonstanciels de lieu selon des compatibilités ou incompatibilités de classes sémantiques. L'analyse du groupe verbal vient ensuite, puis son intégration avec les groupes nominaux dans un énoncé.
- *la génération syntaxique* : les grammaires utilisées opèrent une série de transformations sur les résultats de l'analyse et génèrent les mots cibles dans un ordre correct.

- la génération morphologique : pour les accords en genre et en nombre, les problèmes d'élision et de contraction et les ajustements stylistiques.

Le transfert n'apparaît pas explicitement dans cette description. Il s'effectue en réalité au niveau des dictionnaires et essentiellement de l'analyseur, puisque l'analyse est plus sémantique que syntaxique.

### L'environnement

Un ensemble de modules complète les grammaires écrites en langage GramR.

- GERTELEX : capte le bulletin météo sur la ligne TELEX, note son identification (heure d'arrivée, nombre de mots...) et le soumet à la traduction.
- EXEMETEO : applique les grammaires, relève le nombre de marques de révision et soumet le bulletin à la révision.
- SUPRVIZR : affiche en permanence l'état des files d'attente en ce qui concerne la traduction, la révision et la transmission.
- REVISEUR : permet au traducteur de réviser la traduction très rapidement (éditeur spécialisé), note le nombre de corrections, met le texte en format TELEX, relève l'heure et met le bulletin dans la file d'attente pour être transmis.
- GERTELEX : programme de communication, est chargé d'expédier le bulletin.

Le système tourne sur un micro équipé d'un processeur 68010 et d'un disque dur de 5 Mo, avec une mémoire centrale de 512 Ko. La liaison avec la ligne TELEX 300 bauds et le terminal intelligent affecté à la révision est assurée par une interface de communication série multi-ports. Le système d'exploitation est une variante d'UNIX appelée CROMIX. La programmation est réalisée en FORTRAN IV. Le temps de panne est inférieur à un jour par an !

Le système traduit actuellement les bulletins des stations de Vancouver, Edmonton, Calgary, Regina, Toronto et Halifax, ce qui correspond à une moyenne de 30000 mots par jour, soit 8,5 millions de mots par an. Il faut souligner que ce système, un des plus anciens en service, donne entière satisfaction à ses utilisateurs (90%-95% de phrases correctement traduites). Les erreurs d'analyse peuvent être imputées à des défauts de communication, des fautes d'orthographe ou des lacunes dans le dictionnaire. Il démontre que dans un certain contexte, la traduction automatique peut devenir réalité. Une version français-anglais fonctionne depuis le début de l'année 1990 et traduit les bulletins de tout le Québec.

#### 3.3.3.3.7.2 TAUM-AVIATION<sup>1</sup>

Après que le groupe TAUM ait mis au point différents prototypes qui aboutiront à une première application, le système TAUM-METEO, un second projet mobilisera ses efforts. L'objectif est beaucoup plus audacieux car l'ampleur de la tâche implique un renforcement de l'équipe en personnel (7 chercheurs en 1976, 20 en 1979) et en moyens techniques (nouveaux langages de programmation, LEXTRA et SISIF). Les travaux démarrent en 1976 et aboutissent à la présentation d'un prototype en 1979, limité à la traduction des manuels de maintenance des systèmes hydrauliques. Une évaluation du système en 1980 dénonce la non-rentabilité du programme et entraîne la dissolution du groupe.

---

(1) P. ISABELLE, L. BOURBEAU, M. CHEVALIER, S. LEPAGE : *TAUM-AVIATION : description d'un système de traduction automatisée de manuels d'entretien en aéronautique*, COLING-78, Bergen

Le noyau du système est indépendant des couples de langues traités, les données linguistiques sont traitées séparément des programmes. Dans le cas présent, il ne s'agit que du couple anglais-français. Les descriptions linguistiques externes sont cependant distinctes de la langue générale et ne concernent que la langue spécifique aux manuels de maintenance (sublangage)<sup>1</sup>.

Installé sur un CYBER 173 avec le système d'exploitation NOS/BE 1.4, TAUM-AVIATION a été conçu pour être indépendant du matériel. Une grande partie des données sont compilées en un code objet interprétable par un run-time. Le tableau suivant indique les tailles de fichiers compilés.

Component	Compiled code (6-bit chars)		Runtime req's (60-bit words)
	(a)	(b)	
Pre-processing	60K	15K	34K
Morphological analysis	77K	48K	40K
Source language dictionary	1958K	72K	40K
Syntactic/semantic analysis	205K	56K	120K
Bilingual dictionary	1919K	107K	40K
Syntactic transfer/synthesis	113K	63K	56K
Morphological synthesis	99K	64K	24K
Post-processing	43K	15K	34K

- (a) : données linguistiques compilées  
(b) : interpréteur compilé pour (a)

Le tableau suivant donne la taille des programmes utilisés pour compiler les données linguistiques. La taille est exprimée en mots de 6 bits.

Metalanguage	Used for	Size (x 6 bits)
SISIF	pre- and post-processing	43K
REZO	syntactic/semantic analysis	155K
LEXTRA	lexical transfer	130K
SYSTEMES-Q	structural transfer; syntactic synthesis	28K

Les dictionnaires ne contiennent que les formes non fléchies (entrées classiques dans un dictionnaire conventionnel). Lorsque le projet a été stoppé, le dictionnaire de la langue source (anglais) contenait 4054 entrées représentant le noyau du vocabulaire de maintenance et un sous-ensemble de vocabulaire propre à l'hydraulique. Plus des 2/3 de ces entrées avaient une correspondance dans le dictionnaire bilingue anglais-français.

### Fonctionnement

La sortie devait être exploitable directement pour la photocomposition, ce qui a nécessité un traitement initial du format et un enregistrement des codes correspondants. Le processus de traduction est entièrement automatique mais peut être interrompu avant la

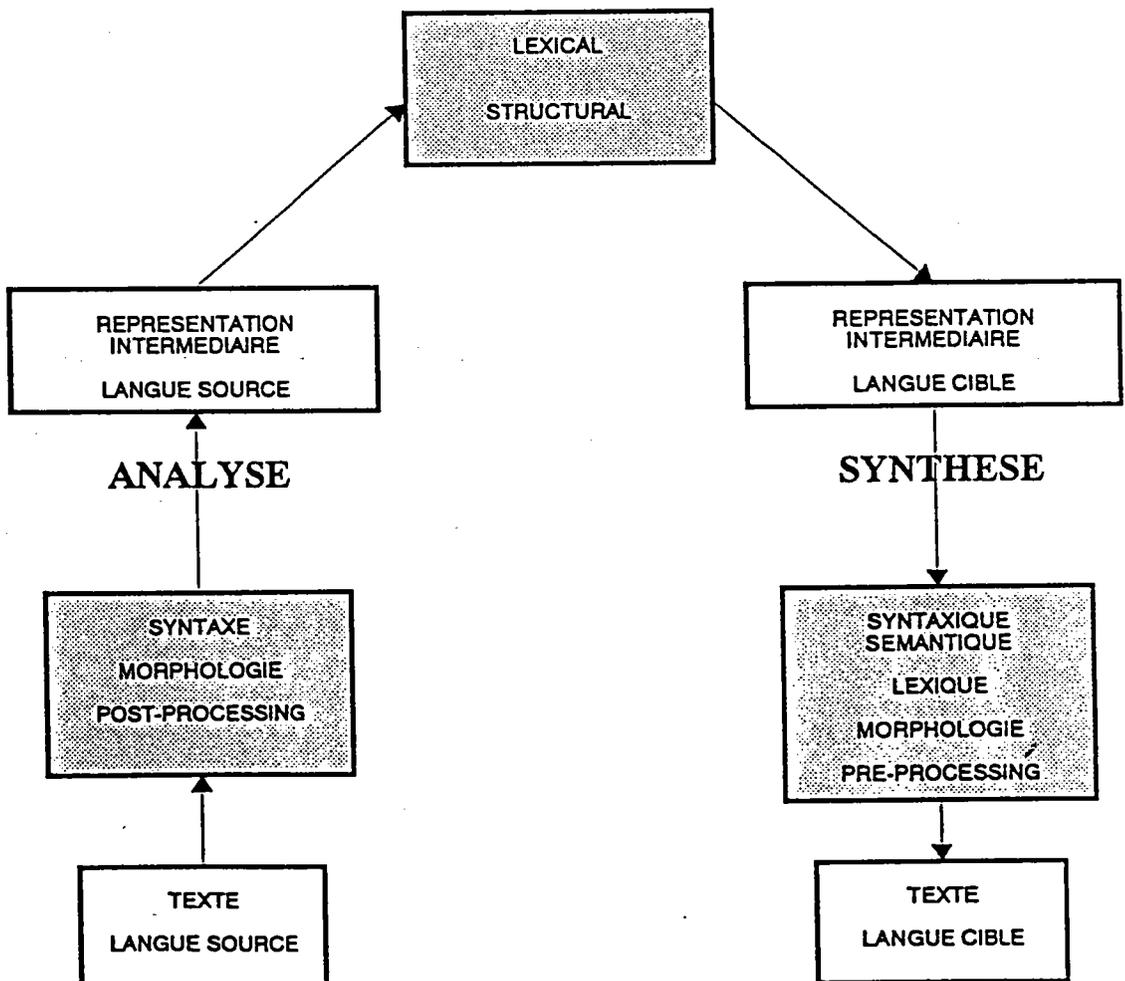
(1) J. LEHRBERGER : "Automatic Translation and the Concept of Sublanguage", in : R. KIT-TREDGE, J. LEHRBERGER (eds.), *Sublanguage : Studies of Language in Restricted Semantic Domains*, De Gruyter, Berlin, 1982, pp. 81-106

interrompu avant la consultation des dictionnaires pour lister les premiers résultats du découpage. La traduction obtenue en fin de traitement nécessite la révision d'un traducteur. La conception du système ne prévoit pas la prise en compte des analyses partielles, de sorte que 20% à 40% des phrases ne sont pas traitées ! Système de deuxième génération, il obéit au schéma classique : analyse, transfert et synthèse. L'organisation de la plupart de ses composantes s'appuie sur des critères linguistiques. L'algorithme est du reste distinct des données concernant les langues.

La méthode du transfert a été retenue. Les règles de traduction ne peuvent pas être appliquées à la chaîne d'entrée mais plutôt à un objet formalisé représentant une description de la structure de son contenu. Le texte source est représenté dans un langage intermédiaire (la structure normalisée) avant l'application des règles liées à la langue cible. Dans TAUM-AVIATION, ce langage intermédiaire dépend de la langue et renvoie aux structures profondes de la phrase par des marques sémantiques. L'application d'une grammaire indépendante du contexte fournit une liste des structures profondes admises (les composants de base pour la langue source et la langue cible) et conduit à un certain degré d'indépendance. Les items lexicaux ne sont pas modifiés et le module de transfert assure le passage à la structure en langue cible.

Les éléments qui composent les trois modules correspondent à des divisions classiques de la linguistique (lexique, morphologie, syntaxe et sémantique). Le schéma suivant représente la structure du système :

### TRANSFERT



## Analyse

Le contexte de l'analyse est limité à la phrase.

- *Pré-processing* : Les mots sont repérés et les unités de traitement isolées. Les règles du module correspondant, qui fonctionne comme un automate déterministe d'états finis, sont écrites en langage SISIF et compilées en structures de listes.

- *Morphologie* : Deux programmes écrits en PASCAL traitent les règles et les exceptions de la morphologie anglaise, n'abordent pas les phénomènes de dérivation mais traitent certains phénomènes de composition.

- *Lexique* : SYDICAN, le dictionnaire de langue source, rassemble les règles lexicales associées à la chaîne de base. Elles sont compilées. Pour le dictionnaire bilingue, le langage LEXTRA permet aux règles lexicales de transfert de réaliser des transformations complexes de structures arborescentes en associant à chaque item lexical un ensemble de transformations d'arbre. Il utilise les éléments de la représentation intermédiaire (règles de grammaire indépendante du contexte) comme des données, ce qui garantit la validité des manipulations d'arborescence. Les règles de LEXTRA sont compilées dans des structures de liste. L'interpréteur recherche la structure des items lexicaux de langue source, relève les règles lexicales associées et les applique à l'arbre.

- *Syntaxe et sémantique* : la grammaire est écrite en REZO<sup>1</sup>. Ce métalangage s'appuie sur la théorie des réseaux de transitions augmentés mais s'écarte légèrement des ATNs de WOODS. REZO n'assure pas l'analyse morphologique, les noeuds sont des symboles complexes qui incluent des éléments sur lesquels il est possible d'effectuer des opérations booléennes, il regroupe un certain nombre de primitives qui pourront être comparées aux structures arborescentes. Alors qu'à chaque état d'un réseau toutes les transformations sont testées, REZO introduit la notion d'états déterministes dans lesquels on n'emprunte que la première transition testée. Il accorde d'autre part un statut spécial aux états qui peuvent être appelés de façon récursive. La grammaire est compilée pour tourner sur une machine virtuelle simulée par un interpréteur. L'analyse est descendante, de gauche à droite.

Le traitement sémantique filtre les structures syntaxiques et élimine ainsi le maximum d'ambiguïtés. Il associe à chaque noeud de l'arbre des marques sémantiques pour les règles de transfert. Le mécanisme essentiel repose sur l'élaboration d'un ensemble de restrictions destinées à éliminer les ambiguïtés lexicales et les ambiguïtés structurales induites par les règles syntaxiques. Les marques assignées à un noeud peuvent être transmises par héritage.

### Transfert

En principe, les structures manipulées ne sont pas ambiguës. Le module (Systèmes-Q) met en relation les items de la langue source et de la langue cible. Des mécanismes transformationnels interviennent au niveau lexical.

### Synthèse

- *syntaxique* : une grammaire transformationnelle du français produit une chaîne de termes auxquels sont associées des informations concernant les flexions correctes.

(1) G. STEWART : *Le langage de programmation REZO*, M.Sc. thesis, Université de Montréal, Montréal, 1975

(2) G. STEWART : *Spécialisation et compilation des ATNs : REZO*, COLING-78, Bergen, Norvège, 1978

- *morphologique* : le module détermine les finales de mot (déclinaison du français et liste des exceptions).
- *post-processing* : reformatage du texte conformément au format du texte source

### Informatique

La construction du système est beaucoup plus complexe que pour TAUM-METEO, la philosophie retenue consistant à créer des outils adaptés à chaque tâche, dans un souci d'efficacité. L'hétérogénéité de la programmation explique sans doute, pour une part, la difficulté de l'entreprise. Les Systèmes-Q ne peuvent pas manipuler les arbres dont les noeuds sont des symboles complexes. Les compilateurs et les interpréteurs sont écrits en FORTRAN alors que d'autres modules sont écrits en PASCAL...

### Evaluation du système

En 1980, une expertise réalisée par A. GERVAIS<sup>1</sup> fixe le seuil de rentabilité à un volume de 6 millions de mots par an. Pour atteindre cet objectif, il est indispensable d'adapter le système à d'autres domaines, ce qui entraînerait de nouveaux coûts. Dans ces conditions, TAUM-AVIATION ne peut concurrencer la traduction manuelle. Le projet est arrêté.

Exemples de traduction :

### Texte d'entrée :

HYDRAULIC PRESSURE IN-LINE RELIEF VALVE

(See figures 2-4 and 2-5.)

30 Identical interchangeable hydraulic pressure in-line relief valves (in-line relief valve) are provided for each ac hydraulic pump and for the dc hydraulic pump. The in-line relief Valves are located in the hydraulic service center. Those for the No. 1, No. 1A, and No. 1B ac and dc hydraulic pumps are on the left side next to the No. 1 service centre assembly. The in-line relief valve for the No. 2 ac hydraulic pump is incorporated in the No. 2 service centre assembly.

31 The in-line relief valves are poppet-type, spring-loaded to the closed position. A pressure of 3450 psi impinging on the poppet is sufficient to overcome the opposing spring force, and the poppet will move from its knife-edge seat.

---

(1) A. GERVAIS : Evaluation of the TAUM-AVIATION Machine Translation Pilot System, Translation Bureau, Secretary of State, Ottawa, Canada, 1980

Traduction automatique brute :

## CLAPET DE DECHARGE INCORPORE DE PRESSION HYDRAULIQUE

(Voir les figures 2-4 et 2-5.)

30 Les clapets de décharge incorporés interchangeable identiques de pression hydraulique (clapets de décharge incorporés) sont prévus pour chaque pompe hydraulique ca et pour la pompe hydraulique cc. Les clapets de décharge incorporés sont situés dans le compartiment hydraulique. Ceux pour les pompes hydrauliques ca et cc no 1, no 1A et no 1B sont du côté gauche à côté du bloc collecteur no 1. Le clapet de décharge incorporé pour la pompe hydraulique ca no 2 est intégré au bloc collecteur no 2.

31 Les clapets de décharge incorporés sont champignon, sont rappelés par ressort à la position fermée. Une pression de 3450 psi s'exerçant sur le clapet-champignon est suffisante pour vaincre la force de rappel du ressort et le clapet-champignon se déplacera de son siège en couteau.

Traduction révisée :

## CLAPET DE DECHARGE INCORPORE DE PRESSION HYDRAULIQUE

(Voir les figures 2-4 et 2-5.)

30 *Des* clapets de décharge incorporés interchangeable identiques de pression hydraulique (clapets de décharge incorporés) sont prévus pour chaque pompe hydraulique ca et pour la pompe hydraulique cc. Les clapets de décharge incorporés sont situés dans le compartiment hydraulique. Ceux *des* pompes hydrauliques ca et cc no 1, no 1A et no 1B sont du côté gauche à côté du bloc collecteur no 1. Le clapet de décharge incorporé pour la pompe hydraulique ca no 2 est intégré au bloc collecteur no 2.

31 Les clapets de décharge incorporés, *du type* champignon, sont rappelés par ressort à la position fermée. Une pression de 3450 psi s'exerçant sur le clapet-champignon est suffisante pour vaincre la force de rappel du ressort et le clapet-champignon *s'écartera* de son siège en couteau.

**2.3.3.3.9 TITRAN<sup>1</sup>**

TITRAN a été développé à l'université de Kyoto, au Japon, en collaboration avec le groupe de Sarrebruck. Il ne traduit que les titres d'articles scientifiques et techniques, dans les couples anglais-japonais, japonais-anglais et japonais-français. Une version japonais-allemand fait l'objet d'un nouveau projet de collaboration avec l'université de Sarrebruck.

---

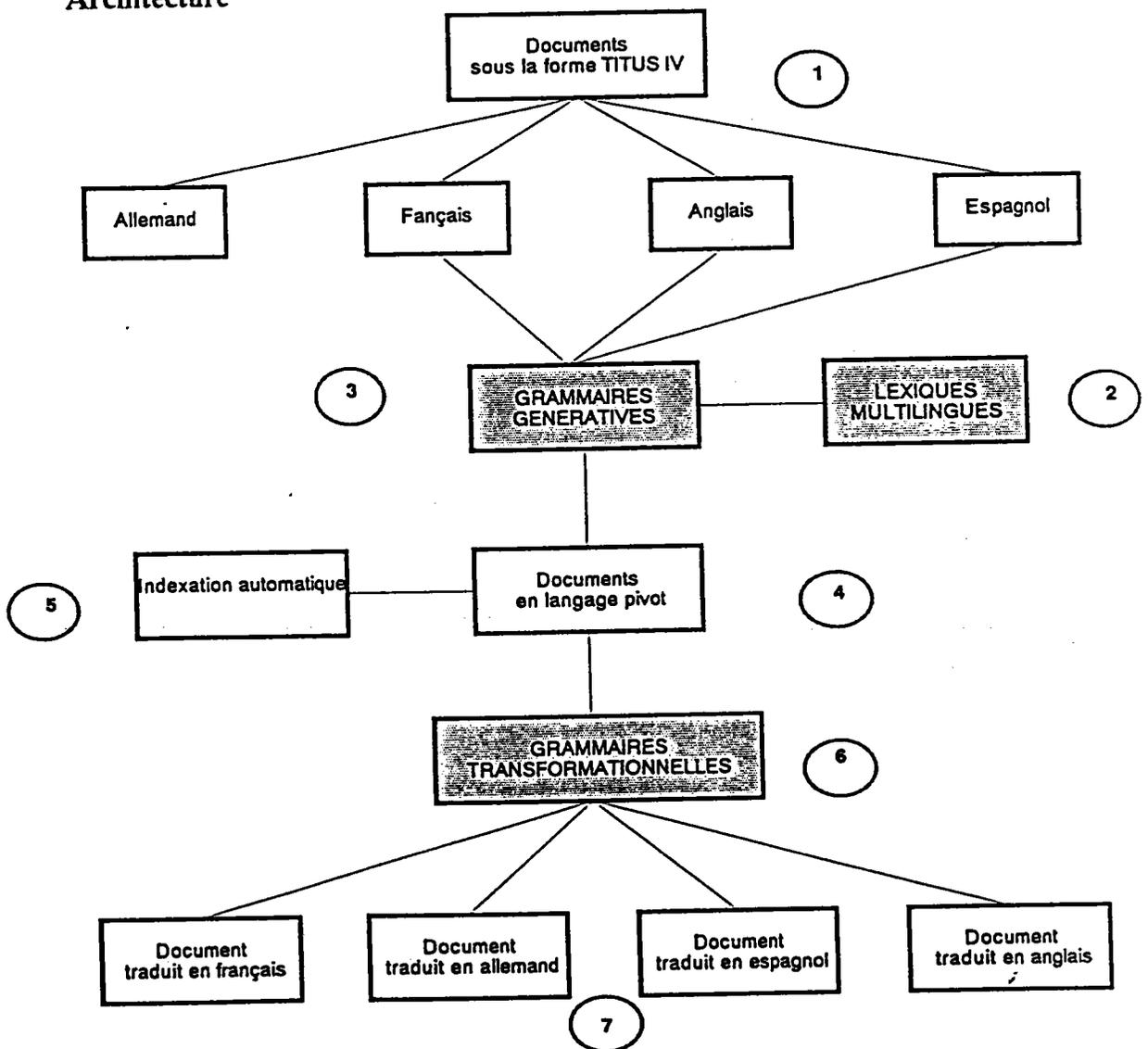
(1) S. MELI : "Informationsmarkt der maschinellen Übersetzung" in *Terminologie et Traduction*, Commission des Communautés Européennes, n°3, 1989, pp. 83-84

2.3.3.3.10 TITUS IV<sup>1</sup>

Système de traduction automatique et simultanée en quatre langues (français, allemand, anglais, espagnol), TITUS IV a été développé par des ingénieurs de l'Institut du Textile de France pour le traitement multilingue de bases de données scientifiques et techniques. Les données sont entrées dans l'une des quatre langues. Elles peuvent être récupérées et traduites automatiquement dans l'une des quatre langues. Afin de surmonter les difficultés de la T.A., certains chercheurs ont limité leur automate à une sémantique fermée (TAUM-METEO). Une deuxième voie consiste à contrôler la syntaxe du texte d'entrée en n'autorisant que certaines structures. TITUS IV manipule un langage à syntaxe contrôlée constitué de deux éléments de base :

- Pour la langue considérée, tout le vocabulaire d'un domaine (variable selon le domaine) et une partie du vocabulaire de base (commun à tous les domaines).
- un sous-ensemble de toutes les règles syntaxiques

## Architecture



(1) J.-M. DUCROT : "Le système TITUS IV : Système de traduction automatique et simultanée en quatre langues in TAO, Observatoire des industries de la langue, DAICADIF, Actes du séminaire international (Paris, mars 1988), pp. 55-75

Pour qu'un texte soit accepté par le système, il doit être rédigé selon les règles de syntaxe préétablies, en français, allemand, anglais ou espagnol (1), et contenir des termes faisant obligatoirement partie du vocabulaire prévu (2). Lorsque le texte est entré sur un terminal conversationnel, des lexiques multilingues et des grammaires génératives (3) en testent la validité lexicale et syntaxique. Toute erreur provoque l'affichage d'un message et l'intervention de l'opérateur. En détectant des mots-clés, ces grammaires effectuent en parallèle une indexation automatique du document (5). Le texte est ensuite réécrit en un langage pivot condensé (4). Dans ce langage binaire, chaque terme n'occupe pas plus de 10 octets. La phrase ainsi stockée est recodée dans une des langues en un temps qui varie de 0,03 à 0,8 s. selon l'ordinateur utilisé. Le texte en langage pivot est traité par des grammaires transformationnelles de sortie (6) qui assurent la traduction automatique, l'affichage, l'édition ou l'enregistrement sur un support magnétique.

Le vocabulaire réunit les mots outils de chaque langue (déterminants, prépositions, conjonctions, adverbes, verbes auxiliaires...) qui font partie du système, et le vocabulaire proprement dit, stocké dans le lexique multilingue. Les correspondances entre langues ne sont pas toujours réalisées au niveau des items de sorte que l'élément du lexique n'est pas le terme mais l'unité lexicale (UL) qui peut contenir un terme ou une combinaison de termes. Les *unités lexicales substantives* figurent sous toutes leurs formes (genre, nombre, cas). Les *unités lexicales adjectivales* regroupent les adjectifs qualificatifs sans complément, les adjectifs qualificatifs avec complément possible et les participes passés adjectivés sous toutes leurs formes, y compris les formes irrégulières de comparatif et de superlatif. Pour les *unités lexicales verbales*, on a stocké l'ensemble des formes conjuguées. Les *unités lexicales adverbiales* regroupent les adverbes courants et les locutions adverbiales.

Le lexique multilingue peut être géré de façon interactive. Il est couplé à un lexique source qui réunit tous les paramètres nécessaires à une unité lexicale, sous forme de pages ou d'écrans : informations indépendantes de la langue en première page, paramètres relatifs au français en deuxième page, à l'anglais en troisième page, à l'espagnol en quatrième page et à l'allemand en cinquième et sixième page.

La syntaxe contrôlée permet au système de traduire rapidement des textes en plusieurs langues. TITUS IV examine surtout la structure de surface. Les schémas de phrase acceptés constituent un modèle linguistique dans lequel doivent s'inscrire toutes les phrases. La version actuelle n'accepte qu'une proposition par phrase. La prochaine version traitera les phrases contenant jusqu'à trois propositions (principale, subordonnée). Le modèle de base du système (*le modèle basique de proposition*) correspond à la structure maximale possible d'une proposition. Il n'y a que le groupe nominal sujet (GNS) qui soit obligatoire.

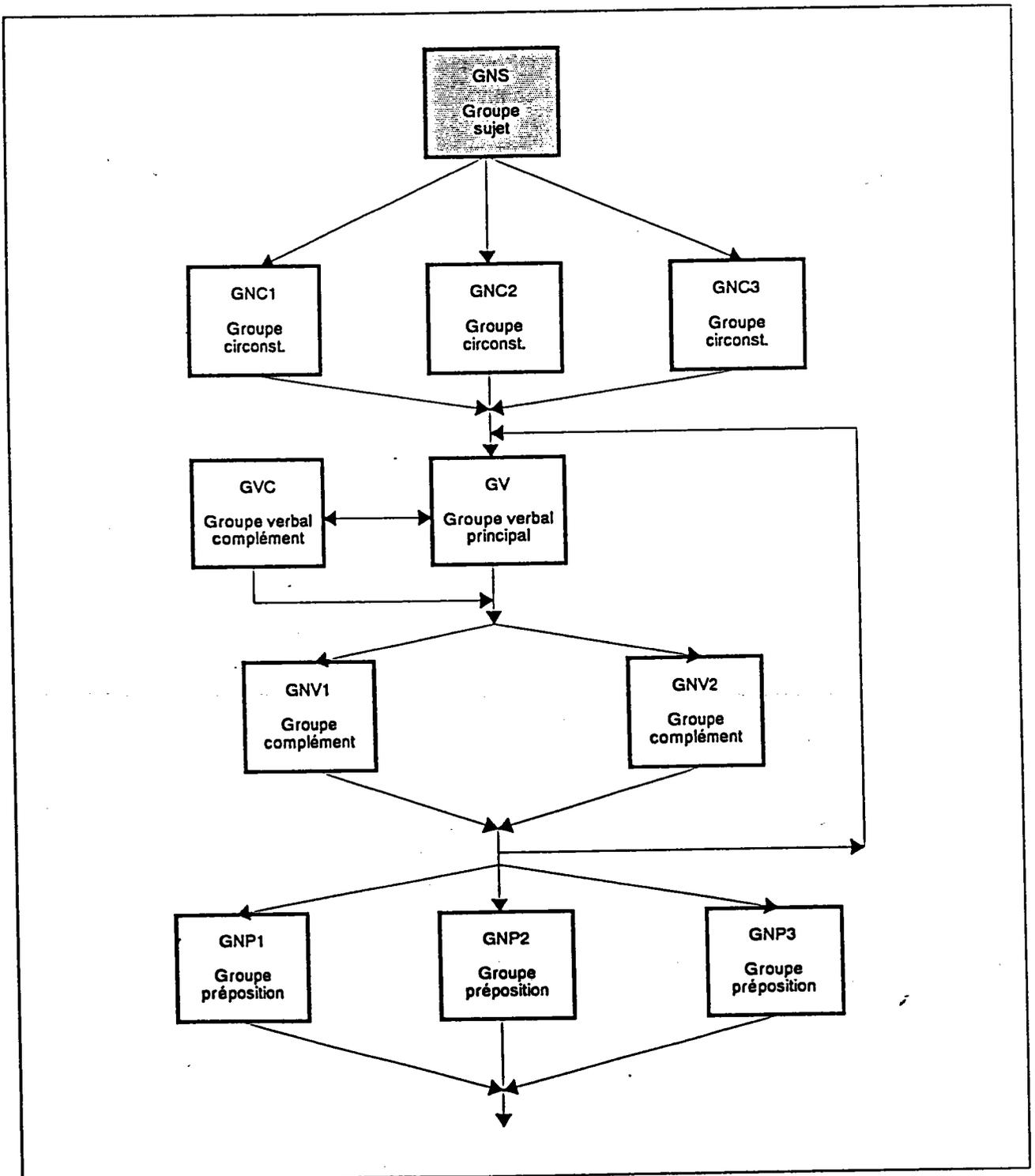
Les groupes sont de deux types, selon leur élément principal :

- le groupe nominal (GN)
- le groupe verbal (GV)

On distingue :

- *le groupe nominal sujet* (GNS), le seul qui soit obligatoire, doit être formé d'au moins un substantif figurant dans le lexique, ou d'un pronom.
- *les groupes nominaux circonstanciels* (GNC1, GNC2, GNC3) sont optionnels. En nombre variable, de 0 à 3, ils commencent par une préposition ou une locution prépositionnelle

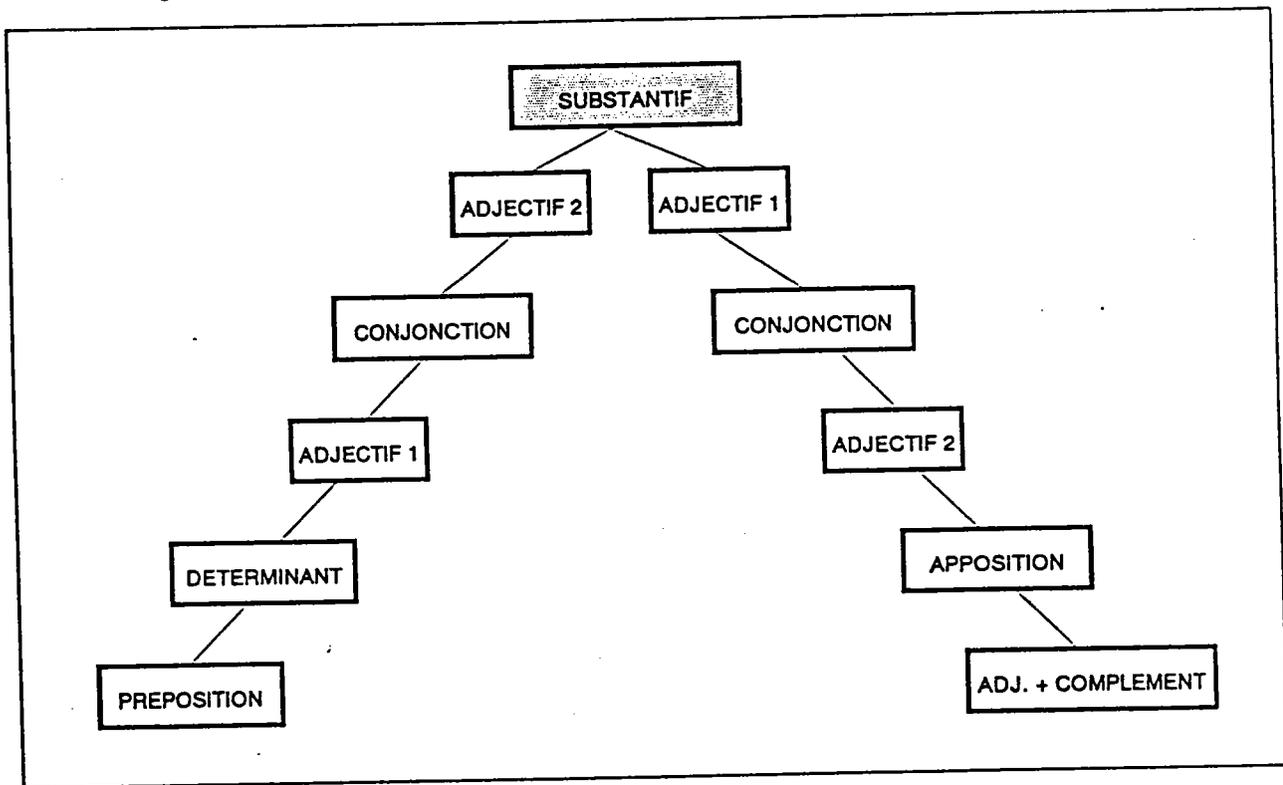
- le *groupe verbal* (GV) est facultatif. C'est une forme verbale simple ou composée, à la voix active ou passive.
- le *groupe verbal complément* (GVC) est un infinitif complément d'un verbe appartenant au GV précédent.
- le *groupe nominal complément* (GNV1) est au minimum un substantif ou un adjectif attribut, complément d'objet du verbe.
- le *groupe nominal complément* (GNV2), avec le(s) complément(s) d'attribution ou d'agent du verbe.
- les *groupes nominaux prépositionnels* (GNP1, GNP2, GNP3)



On constate que la phrase traitée dans TITUS IV se décompose en groupes nominaux et en groupes verbaux.

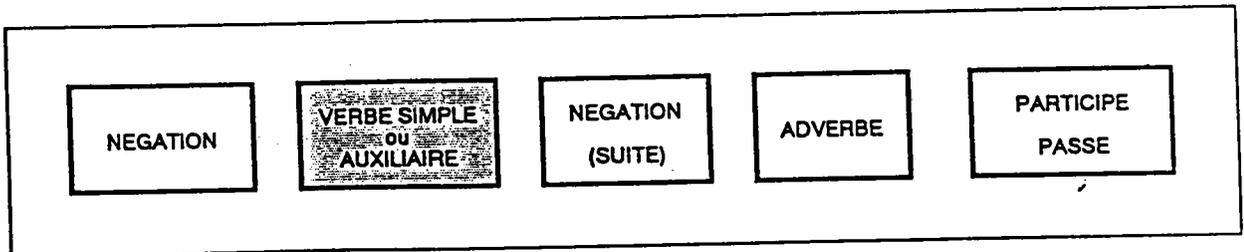
Le groupe nominal (GN) est formé d'un ou plusieurs sous-groupes nominaux (SN). (jusqu'à 15). Le sous-groupe nominal doit contenir un substantif au minimum auquel on ajoute des composants facultatifs (une préposition, un déterminant, un ou deux adjectifs simples avant ou après le substantif, une conjonction de coordination entre deux adjectifs, une ou deux appositions, un adjectif avec un complément (autre sous-groupe nominal...)). Les relations autorisées entre sous-groupes sont les suivantes :

- sujet
- complément du nom
- complément de l'adjectif
- complément de comparaison



Le groupe verbal (GV) contient :

- un verbe auxiliaire ou modal qui fait partie des éléments outils, partie intégrante du système.
- un verbe du lexique



Le système est interactif. L'opérateur rédige les phrases, les introduit sur un terminal et obtient leur traduction instantanément. Il peut corriger ses fautes au fur et à mesure. De

même, il peut aider la machine dans son travail de traduction, en levant les ambiguïtés qui posent problème et font l'objet d'un dialogue à l'écran comme ci-dessous<sup>1</sup> :

Titus IV system

input sentence = the sewing  
machine is controlled  
by a microprocessor

*"controlled"*  
*polysemic term*

1) *ver. (to command, to drive)*  
2) *ver. (to test, to verify)*

*which ?*

Il faut mentionner l'utilisation de signes diacritiques qui aident le système à différencier des formes homographes lorsqu'elles sont en majuscule (A => a ou à), et l'emploi de virgules pour reconnaître facilement l'antécédent d'un pronom relatif, d'un adjectif...

En résumé, le système a ses contraintes avec l'obligation pour l'opérateur de se plier à une certaine discipline et l'impossibilité de faire appel à un néophyte ignorant le domaine et donc incapable de donner suite aux requêtes du système. En contrepartie, la rigueur de la syntaxe assure la clarté du texte.

### Evaluation

Le respect de la syntaxe contrôlée rallonge le temps d'entrée de 10% pour un résumé de 10 phrases.

Un opérateur qui maîtrise bien le système rentre 25 documents complets en 6 heures.

Le contrôle de la syntaxe et du lexique d'un document, l'affichage des messages d'erreur, la transformation des phrases en langage pivot, l'enregistrement des données binaires et les corrections effectuées a posteriori nécessitent 2,5 secondes de temps calcul sur un IBM 4331-2, en temps partagé et sous le système VM. Pour la recherche d'un document, sa traduction dans une des quatre langues et son édition, il ne faut qu'une seconde.

### Exemples de traductions :

les exemples de traduction sont extraits du document cité en note.

---

(1) extrait de J. M. DUCROT : "Le système TITUS IV. Système de traduction automatique et simultanée en quatre langues" in : *Traduction Assistée par Ordinateur. Perspectives technologiques, industrielles et économiques envisageables à l'horizon 1990* (Actes du séminaire international - Paris - mars 1988), DAICADIF-Observatoire des industries de la langue, Paris, 1989, p. 62

DATE 12/05/82

TITUS 4 SYSTEM

FILE NAME : DEMO

SENTENCE NR : 02

INPUT LANGUAGE (DE,EN, ES, FR) =  
FR

#### INPUT SENTENCE

*Les progrès récents des logiciels appropriés au traitement on-line des systèmes d'information ont favorisé le développement des bases de données dans tous les domaines des sciences et des techniques.*

#### FRANCAIS

*Les progrès récents des logiciels appropriés au traitement on-line des systèmes d'information ont favorisé le développement des bases de données dans tous les domaines des sciences et des techniques.*

#### ENGLISH

*The recent progress of the software suitable for the on-line treatment of information systems has favored the development of data bases in all the fields of sciences and techniques.*

#### DEUTSCH

*Die neuen Fortschritte der fuer die on-line verarbeitung der informationssysteme geeigneten software haben die entwicklung der databasen in allen bereichen der wissenschaft und der techniken beguenstigt.*

#### ESPAÑOL

*Los progresos recientes del software apropiado para el tratamiento on-line de los sistemas de documentacion favorecieron el desarrollo de las bases de datos en todos los campos de las ciencias y de las tecnicas.*

next step command =

#### phrase d'entrée en français :

*L'analyse du fluage des fibres de polyéthylène après irradiation sous vide montre ; qu'un processus de pontage survient dans la région amorphe ; tandis qu'une coupure de la chaîne moléculaire de la région cristalline est observée.*

#### traduction en français :

*L'analyse du fluage des fibres de polyéthylène après irradiation sous vide montre que un processus de pontage survient dans la région amorphe tandis que une coupure de la chaîne moléculaire de la région cristalline est observée.*

#### traduction en anglais :

*The polyethylene fiber creep analysis after irradiation under vacuum shows that a cross-linking process occurs in the amorphous region whereas a molecular chain scission of crystalline region is observed.*

#### traduction en allemand :

*Die Analyse des Kriechens der Polyethylenfasern nach der Bestrahlung unter Vakuum zeigt, dass ein Quervernetzungsprozess im amorphen Bereich auftritt, wogegen eine Molekulkettenspaltung des kristallinen Bereichs beobachtet wird.*

#### traduction en espagnol

*El analisis de la fluencia de las fibras de polietileno despues de la irradiacion debajo vacio muestra que un proceso de enlace transversal ocurre en la region amorfa mientras que una escision de la cadena molecular de la region cristalina es observada.*

## phrase d'entrée en français :

*L'utilisation des systèmes de CAO dans l'industrie peut conduire à une réduction importante des coûts de production ; si ces systèmes sont intégrés dans les lignes de fabrication.*

## traduction en français :

*L'utilisation des systèmes de CAO dans l'industrie peut conduire à une réduction importante des coûts de production si ces systèmes sont intégrés dans les lignes de fabrication.*

## traduction en anglais :

*The use of CAD systems in industry can lead to a great reduction in production costs if these systems are integrated in production lines.*

## traduction en allemand :

*Die Verwendung der CAD-Systeme in der Industrie kann zu einer grossen Produktionskostenreduzierung führen, wenn diese Systeme in den Fertigungsstrassen integriert werden.*

## traduction en espagnol :

*La utilización de los sistemas de CAO en la industria puede conducir a una reducción importante de los costos de producción si estos son integrados en las líneas de fabricación.*

## phrase d'entrée en allemand :

*Die Analyse des Kriechens der Polyethylenfasern nach Bestrahlung unter Vakuum zeigt; dass ein Quervernetzungsprozess \* im amorphen Bereich auftritt; wogegen eine Molekulkettenspaltung \* im kristallinen Bereich beobachtet wird.*

## traduction en français :

*L'analyse du fluage des fibres de polyéthylène après l'irradiation sous vide montre que un processus de pontage survient dans la région amorphe tandis que une coupure de la chaîne moléculaire est observée dans la région cristalline.*

## traduction en anglais :

*Polyethylene fiber creep analysis after irradiation under vacuum shows that a cross-linking process occurs in the amorphous region whereas a molecular chain scission is observed in the crystalline region*

## traduction en allemand :

*Die Analyse des Kriechens der Polyethylenfasern nach der Bestrahlung unter Vakuum zeigt ; dass ein Quervernetzungsprozess im amorphen Bereich auftritt, wogegen eine Molekulkettenspaltung im Kristallinen Bereich beobachtet wird.*

## traduction en espagnol :

*El analisis de la fluencia de las fibras de polietileno despues de la irradiacion debajo vacío muestra que un proceso de enlace transversal ocurre en la region amorfa mientras que una escision de la cadena molecular es observada en la region cristalina.*

## phrase d'entrée en anglais :

*An increase in flexural strength, and shear resistance by 40 pc, and 25 pc is observed ; when aramid fibers are used in composite materials for automotive industry.*

## traduction en français :

*Une augmentation de la résistance à la flexion et de la résistance au cisaillement de 40 % et de 25 % est observée quand des fibres aramides sont utilisées dans les matériaux composites pour l'industrie automobile.*

## traduction en anglais :

*An increase in flexural strength and shear resistance by 40 % and 25 % is observed when aramid fibers are used in composite materials for automotive industry.*

## traduction en allemand :

*Eine Erhöhung der Biegefestigkeit und des Scherwiderstandes um 40 % und 25 % wird beobachtet, wenn Aramidfasern in den Verbundwerkstoffen für die Automobilindustrie verwendet werden.*

## traduction en espagnol :

*Un incremento de la resistencia a la flexion y de la resistencia al cizallamiento de 40 % y de 25 % es observado cuando fibras aramida son utilizadas en los materiales compuestos para la industria del automovil.*

## phrase d'entrée en espagnol :

*La naturaleza compleja de los productos formados por esta resina no permite de determinar ; si una correlacion estrecha existe : entre la liberacion de formaldehido, y la estructura quimica de los residuos y de los reactivos.*

## traduction en français :

*La nature complexe des produits formés par cette résine ne permet pas de déterminer si une corrélation étroite existe entre la libération de formaldéhyde et la structure chimique des résidus et des réactifs.*

## traduction en anglais :

*The complex nature of products formed by this resin does not permit to determine if a close correlation exists between the formaldehyde release and chemical structure of residues and reactants.*

## traduction en allemand :

*Die Komplexe Natur der durch dieses Harz gebildeten Produkte erlaub nicht zu bestimmen, wenn eine enge Korrelation zwischen dem Friesetzen von Formaldehyd und der chemischen Struktur der Rückstände und der Reaktanten existiert.*

## traduction en espagnol :

*La naturaleza compleja de los productos formados por esta resina no permite determinar si una correlacion estrecha existe entre la liberacion de formaldehido y la estructura quimica de los residuos y de los reactivos*

### 2.3.3.3.11 TITUS V

Les performances de TITUS IV ont suscité l'intérêt de nombreux organismes. Il a été décidé de construire une version plus complète pour en élargir le cadre d'utilisation. Des études préliminaires ont permis de jeter les bases d'une conception qui s'inspire d'une nouvelle méthode d'analyse, qui tient compte des progrès techniques de l'informatique et de l'apparition de nouveaux outils issus de l'Intelligence Artificielle.

Parmi les caractéristiques du nouveau système :

- Construction d'un ensemble modulaire qui facilite l'ajout ultérieur de termes, de règles et de langues.
- Possibilité de stocker en langage pivot, à côté des éléments lexicaux de base et de leurs paramètres syntaxiques, des idées inférées ou même sous-jacentes dans certains cas. Il s'agit là d'une tentative d'analyse en profondeur... Le nouveau langage pivot devrait être indépendant des langues traitées. Les modules de génération du langage pivot seront écrits en PROLOG puis en langage C pour en garantir la portabilité, tout comme pour le logiciel de base.
- Assouplissement des règles de syntaxe contrôlée et accroissement du nombre des structures autorisées.
- Les langues traitées ne seront plus limitées aux quatre langues de TITUS IV. Les travaux préliminaires ont montré qu'il serait possible d'introduire sans grandes difficultés le portugais, l'italien et le roumain. Des recherches plus longues en ce qui concerne les grammaires sont envisagées, dans l'optique de traiter l'arabe littéraire, le chinois, le japonais et le russe. Les lexiques seront adaptés aux nouvelles techniques d'analyse et de transformation ainsi qu'aux nouveaux paramètres lexicaux.
- Les phrases ne seront plus limitées à une proposition. Seront acceptées : les indépendantes ou nominales (sans verbe), les principales, les subordonnées (de condition, temps, cause, conséquence, etc...), les relatives (même incluses), les complétives, les interrogatives (directes ou négatives indirectes, pseudo-négatives), les impératives et déclaratives, les formules idiomatiques.
- Si TITUS IV n'accepte que deux personnes, TITUS V acceptera toutes les formes verbales.
- L'entrée du texte sera facilitée par un traitement de texte souple et adapté aux langues étrangères traitées.
- La mise en forme et la sortie s'effectueront dans un environnement monolingue ou multilingue, avec un logiciel de P.A.O. (Publication Assistée par Ordinateur) et sur imprimante Laser.

### 2.3.3.4 Aides à la traduction

Il est difficile d'opérer des distinctions, parmi les systèmes qui traduisent automatiquement : certains nécessitent une intervention manuelle importante, d'autres laissent à l'homme le soin d'effectuer le travail essentiel de traduction, la machine n'intervenant que pour l'aider.

---

(1) J.-M. DUCROT : "Le projet TITUS V de traduction automatique" in *TAO*, Observatoire des industries de la langue, Actes du séminaire international (Paris, mars 1988), pp. 69-76

### 2.3.3.4.1 ALPS (Automated Language Processing Systems Ltd.)<sup>1,2,3</sup>

Comme WEIDNER (2.3.3.2.11), ALPS a bénéficié des travaux en informatique et en linguistique automatique effectués depuis les années 70 à l'Institut des Sciences de la Traduction (Brigham Young University à Provo dans l'Utah). En 1980, un groupe de chercheurs quitte l'institut et créent un centre de recherche pour développer des logiciels de traduction. Le centre teste l'impact de ses produits sur le marché et concentre ses efforts sur leur convivialité. En 1983, le produit est commercialisé. En 1984, ALPS s'implante en Europe (Neuchâtel). En 1990, IBM, Sperry, Texas Instruments, l'École polytechnique de Coventry, Unisys, NCR France, Norsk Data, l'Union des Banques Suisses, Rank Xerox, l'OTAN et Lexitel comptent parmi ses clients.

Indépendamment de son propre service de traduction, ALPS propose un large éventail d'outils informatisés, plus ou moins automatisés pour assurer une certaine souplesse d'emploi dans le sens où l'intervention de l'ordinateur devient modulable, où on l'adapte au type de texte et aux situations de traduction.

Le système fonctionne à plusieurs niveaux, chaque niveau englobant les niveaux inférieurs. Le niveau le plus élevé correspond au module de traduction automatique et fonctionne de façon interactive (phrase après phrase, il propose une traduction à l'opérateur après lui avoir posé des questions pour lever les ambiguïtés). Les modules principaux, SELECTERM, AUTOTERM et TRANSACTIVE, traitent les couples anglais-français, anglais-allemand, anglais-espagnol, anglais-italien, français-anglais et allemand-anglais. ALPS propose un ensemble de logiciels, le TRANSLATION SUPPORT SYSTEM, qui doit aider le traducteur dans toutes les phases du processus de traduction : entrée, édition et analyse du texte source, création et édition de dictionnaires, traduction du texte source, édition, impression et transmission des documents traduits. L'ensemble des fonctions est accessible par menus. Des fenêtres d'aide en facilitent l'utilisation.

L'éditeur (TRANSLATION EDITOR) affiche les textes sources et cibles, page par page, avec une fenêtre au bas de l'écran pour visualiser les références lexicales. Il est possible de faire défiler les deux textes en parallèle. L'éditeur segmente automatiquement le texte source en unités de traduction, assure la consultation de dictionnaires on-line et affiche dans la fenêtre de référence lexicale l'expression équivalente dans la langue demandée. Il met à disposition des fonctions avancées de traitement de texte et réinsère dans le texte cible les codes de formatage (mise en page et typographie) du texte source. Il est également capable de stocker des passages à retranscrire sans modification dans le document de sortie.

- AUTOTERM recherche automatiquement les équivalences d'un terme source dans les dictionnaires on-line et les affiche. Il effectue une analyse du texte au niveau lexical et reconstitue la forme canonique des termes fléchis. Il indexe les équivalents affichés dans la fenêtre de référence et les incorpore directement dans le texte cible.

- REPETITION PROCESSING facilite la traduction de chaînes apparaissant plusieurs fois dans un même texte. Le remplacement est automatique mais peut être interrompu à tout moment.

Le traitement de texte multilingue (MULTILINGUAL WORD PROCESSING) comprend 400 caractères (lettres, chiffres, accents, signes de ponctuation, symboles, caractères spéciaux) qui peuvent être affichés à l'écran ou imprimés. Les claviers sont disponibles dans les différentes langues et respectent les dispositions normalisées des touches. Il répond aux besoins du traitement de textes classiques et aux besoins de la traduction (consultation de dictionnaires on-line).

(1) Notice fournie par le siège européen de ALPS S.A. (3, avenue Beauregard, 2035 Corcelles, Suisse)

(2) Présentation du produit dans *Language Monthly*, the international journal for language and translation, n°20, may 1985

(3) A. DANIK : "The ALPS Computer Assisted Translation System", BCS Natural Language Translation Specialist Group Newsletter 14, 1984, pp. 5-14

Pour analyser le texte source et identifier les termes à entrer dans les dictionnaires de traduction, plusieurs outils sont proposés :

- TERMLOCATOR analyse les textes et identifie les mots et les phrases qui apparaissent plusieurs fois. Il crée un fichier des termes et un fichier de référence pour l'établissement des listes de fréquence (FREQUENCY LISTER) et de contexte pour une ligne ou pour une phrase (CONTEXT LISTER).
- La liste des mots non trouvés (WORDS-NOT FOUND LISTER) rassemble les mots qui n'ont pas été trouvés dans les dictionnaires. Ils pourront y être intégrés ultérieurement avec leurs traductions.

Les dictionnaires ont un format simple avec, pour chaque entrée, sa catégorie et ses traductions. L'éditeur (DICTIONARY EDITOR) permet d'en éditer le contenu. DIXTRAC-TION examine un texte avec plusieurs dictionnaires et produit un lexique spécifique après une analyse morphologique et syntaxique qui permet d'isoler les formes canoniques. Il recherche les formes de bases, classe les équivalents selon les priorités définies par l'utilisateur et supprime automatiquement les données dupliquées dans plusieurs dictionnaires. DICTIONARY/LOADING/UNLOADING convertit des fichiers de texte en dictionnaires. Des glossaires et des listes de mots émanant d'autres systèmes peuvent être convertis rapidement en données compatibles avec le TRANSLATION SUPPORT SYSTEM. La fonction inverse est également possible. DICTIONARY MERGING fusionne plusieurs dictionnaires on-line en un seul, l'intérêt consistant à construire un dictionnaire principal à partir de nombreux documents traitant un domaine identique.

A côté de ces outils, un ensemble de modules gère le fonctionnement du système :

- AERA MANAGEMENT organise et supervise des zones dans lesquelles sont stockés les textes, les dictionnaires et les traductions. Un menu permet à l'utilisateur de passer d'une zone à une autre, d'enregistrer dans une zone, de renommer une autre zone ou encore de lui assigner une langue de traitement.
- PRINTER QUEUE MANAGEMENT contrôle la file d'attente des textes à imprimer et prévoit tout type d'intervention.
- FORMATTING DISKETTES remplace avantageusement l'utilitaire du système UNIX.
- ARCHIVING/RETRIEVING se substitue aux fonctions backup et restore.
- IMPORTING/EXPORTING est un traducteur de format qui assure les transferts vers un autre système et l'importation de fichiers incompatibles.
- SENDING/RECEIVING est un support de communication qui utilise une sortie RS 232 en asynchrone.

Le système tourne sous XENIX (version Microsoft d'UNIX) sur un IBM AT, avec 2 Mo de mémoire centrale, un disque dur de 220 Mo (40 ms de temps d'accès), un moniteur monochrome EGA et une imprimante HP Laser.

Nous ne disposons pas d'exemples de traductions et n'avons pas eu l'occasion de tester le système. Une analyse du produits<sup>1</sup> révèle que lors d'un test chez Hewlett Packard à Guadalajara au Mexique, et sur un texte de 20 000 mots, les traducteurs sont passés de 200 à 567 mots traduits dans l'heure, (1321 mots pour le meilleur résultat, 465 pour le plus mauvais). Un autre test, effectué au Canada<sup>2</sup>, évalue à 30 000 mots la taille minimale du texte pour un gain d'efficacité de 15%.

(1) "Positive reactions from ALPS customers" in : *Language Monthly*, The international journal for language and translation, n° 20, mai 1985, Praetorius Limited, Nottingham, réimpression

(2) S. O'BRIEN : Customer Support Manager pour TSS (Translation Support System) chez Multiscript-La Langagerie à Montréal.

### 2.3.3.4.2 MERCURY

Commercialisé par LinguaTech aux Etats-Unis et par InfoARBED en Europe sous le nom de TERMEX, le produit doit être considéré comme l'élément d'une station de traduction. Il met à disposition un dictionnaire multilingue (anglais, français, allemand et néerlandais) et assure l'accès à des banques de données terminologiques par télécommunication<sup>1</sup>.

### 2.3.3.4.3 MULTI LINGUA<sup>2</sup>

Le concept recouvre trois produits :

- TERMEX Dictionary Manager est intégré au programme de traitement de texte, il compile et édite la terminologie multilingue et permet d'accéder à une banque de données terminologiques multilingue.
- HARRAP'S CD-ROM propose une base de données et un dictionnaire multilingue (accès à une bibliothèque des plus grands dictionnaires bilingues).
- LINGUAWRITE compile des lettres d'affaires en cinq langues à partir d'une base de 10 000 phrases standard.

### 2.3.3.4.4 SITE (Sonovision ITEP Technologies)<sup>2</sup>

Le centre a développé deux outils d'aide à la traduction :

- PHENIX : un système de gestion de la terminologie.
- AQUILA : un ensemble de logiciels pour traducteurs professionnels et indépendants.

### 2.3.3.4.5 SMART<sup>3</sup>

SMART Communications Inc., implantée à New York, commercialise un système de traduction depuis 1972 en Amérique du Nord. Le produit n'est pas vendu en Europe. Le système est capable, semble-t-il, de livrer une traduction brute de 200 000 mots en une heure, sur un gros calculateur.

Il est utilisé par le Ministère du travail et de l'immigration à Ottawa pour traduire des descriptions d'emploi, de l'anglais vers le français. Il peut les afficher dans les minutes qui suivent sur 3500 terminaux répartis sur tout le Canada. La traduction est générée sur l'écran et révisée par l'opérateur. Si la qualité de la prestation est moyenne, il n'en reste pas moins vrai que le système est très rapide.

Il est composé de deux modules, MAX (SMART Expert Editor) et ST (SMART Translator). MAX analyse le texte et le réécrit sous une forme particulière, propre à faciliter le processus de traduction (insertions de marques grammaticales et lexicales).

Le produit est utilisé par plus de 30 grandes sociétés, comme General Electric USA. Les versions sont disponibles pour les couples anglais-français, anglais-espagnol, anglais-portugais, anglais-italien, ceci dans les deux sens. Des versions anglais-allemand, anglais-japonais, anglais-grec sont en préparation.

(1) W. J. HUTCHINS : "Recent Developments in Machine Translation. A review of the Last Five Years", in : *New Directions in Machine Translation*, Conference Proceedings, Budapest, 18-19 août 1988, pp. 21-22

(2) S. MELI : "Informationsmarkt der maschinellen Übersetzung" in *Terminologie et Traduction*, Commission des Communautés Européennes, n° 3, 1989, pp. 90-91

(3) G. KINGSCOTT : *Applications of Machine Translation*, Study for the Commission of the European Communities, pp. 28-30

#### 2.3.3.4.6 TII (Telecommunications Industries Inc.)<sup>1</sup>

La société commercialise le système TWP/70 (Translating Word Processor) pour des micro-ordinateurs compatibles IBM PC. Cet outil d'aide à la traduction est interactif et fonctionne sur les couples anglais-espagnol, anglais-russe, anglais-français, et ceci dans les deux sens. En prévision, les couples anglais-italien, anglais-allemand et anglais-portugais, dans les deux sens.

### 2.3.4 Prototypes en voie d'achèvement

#### 2.3.4.1 ATAMIRA<sup>1</sup>

Développé en Bolivie, le système est multilingual et utilise une langue naturelle, l'aymara, comme langage pivot. Cette langue naturelle est particulièrement régulière dans sa morphologie et sa syntaxe. Il devrait fonctionner de façon interactive sur l'anglais, l'espagnol et l'allemand. Un bureau de traduction de Panama utilise déjà la version anglais-espagnol.

#### 2.3.4.2 IBM Japon<sup>1</sup>

La société IBM Japon a construit un système anglais-japonais pour traduire ses propres manuels d'informatique. Il s'agit d'un système à transfert qui réalise d'excellentes traductions, pour ce qui concerne le domaine évoqué plus haut, tout au moins.

#### 2.3.4.3 MELTRAN<sup>1</sup>

Mis au point par Mitsubishi, MELTRAN est un système interactif pour le couple japonais-anglais. Système à transfert, il nécessite une pré-édition et une post-édition, travaille en automatique ou par étapes. Le dictionnaire contient 50 000 entrées et 30 000 mots techniques. Écrit en ESP, il traduit 5 000 mots à l'heure sur un MELCOM PSI, dans un environnement SIMPOS. Il est pratiquement commercialisable.

#### 2.3.3.4 RMT (Ricoh Machine Translation)<sup>1</sup>

C'est un système de transfert de structures pour le couple anglais-japonais. Il est bâti sur les *Augmented CFG Dependency Trees*. Le dictionnaire de base contient 30 000 entrées, le dictionnaire utilisateur 30 000 et la terminologie 50 000 entrées. Il est écrit en langage C et tourne sur un 3B2 (AT&T-CPU) dans un environnement UNIX V. Il est capable de traduire 4 500 mots à l'heure. Une version japonais-anglais est en développement. Il est probablement commercialisé au moment où ces lignes sont écrites.

#### 2.3.3.5 SHARP<sup>1</sup>

OA-110WB est un système interactif, à transfert, et concerne le couple anglais-japonais. Il nécessite une étape de pré-édition et de post-édition. Le dictionnaire de base contient 60 000 entrées et 40 000 mots techniques. Il traduit des textes d'économie, de traitement de l'information, d'électronique et de constructions mécaniques, à la vitesse de 5 000 mots à l'heure. Il est écrit en langage C et tourne sur des ordinateurs Sharp (OA-110WS, OA-210, OA-310 et IX-7) dans un environnement UNIX V.

---

(2) S. MELI : "Informationsmarkt der maschinellen Übersetzung" in *Terminologie et Traduction*, Commission des Communautés Européennes, n°3, 1989, pp. 90-91

### 2.3.3.6 TOVNA

Présenté lors de la Conférence "*Translating and the Computer*" à Londres, en 1987, le système a été développé par D. COHEN, en Israël, puis à Londres. Les travaux ont démarré en 1970 et n'ont débouché sur une application commerciale qu'en 1985. La première version concerne le couple anglais-français. Les couples français-anglais et anglais-russe doivent être disponibles actuellement. le couple anglais-espagnol est prévu. Un système pilote est installé à la *World Bank* à Washington. Sa capacité d'apprentissage le distingue des autres systèmes. Il serait capable de mémoriser les corrections apportées par le traducteur pour en tenir compte ultérieurement. Son concepteur a annoncé<sup>1</sup> en août 1989 que la société Lexi-Tech à Moncton au Canada avait décidé de le substituer au système LOGOS. A l'origine de cette société, un contrat de 21 millions de dollars canadiens, et un volume de 100 000 pages de manuels techniques à traduire et à publier pour le ministère de la Défense. Avec un dictionnaire de 75 000 termes et expressions et un logiciel capable de faire l'interface entre le système de traduction et le système de publication assistée par ordinateur, Lexi-Tech compte faire de Moncton un centre international de traduction et d'édition.

### 2.3.3.7 TRANSTAR-1

Le projet a été conduit par la *Military Academy of Sciences*. Développé sur le couple anglais-chinois, il s'agit d'un système à transfert, automatique, nécessitant une étape de pré-edition (réduite) et de post-édition. TRANSTAR-1 inclut des programmes de compilation de lexiques anglais-chinois et des utilitaires d'analyse statistique de textes en anglais. Les domaines traités sont les recherches militaires, l'électronique, la chimie et les sciences économiques.

## 2.3.5 Projets

L'énumération de ces projets, bien que partielle, nous donne un aperçu de la situation actuelle et des tendances, dégage des nouvelles orientations de la recherche et du développement.

### 2.3.5.1 ATTP (Automatic Translation Typing Phone)

Ce prototype imaginé par Toshiba (Japon) introduit une nouvelle composante : la communication par satellite. La liaison s'effectue entre deux partenaires en anglais ou en japonais. Le texte anglais est transmis à une station de travail Toshiba-3000 qui le traduit en japonais grâce à un logiciel résident. La traduction est envoyée au correspondant par satellite. La réponse est rédigée en japonais sur une station de même type, traduite en anglais et expédiée par satellite. Les textes s'affichent de part et d'autre, sur des écrans partagés.

### 2.3.5.2 BRITISH TELECOM

Les recherches lancées en 1984 par British Telecom ont pour objectif de réaliser un système de traduction automatique des communications téléphoniques d'affaire, opérationnel et commercialisable en 1995.

L'interlocuteur parle lentement et distinctement dans un micro relié à un micro-ordinateur Merlin 2000. La communication est validée, traduite et générée par un programme

---

(1) au cours du *Machine Translation Summit II* à Munich, en août 1989.

de synthèse de la parole. Les langues à l'étude sont l'anglais, le français, l'allemand, l'italien, l'espagnol et le suédois. Le corpus contient 400 expressions courantes. Aux difficultés habituelles de la traduction automatique s'ajoutent ici les problèmes de reconnaissance de la parole. KEY (Key Technology Centre) travaille sur un projet similaire au Japon.

### 2.3.5.3 BYU-TAS<sup>1</sup>

L'équipe de la Brigham Young University (USA) développe une station de travail pour traducteur. Elle devra rassembler un traitement de texte, une gestion de fichiers terminologiques personnels, communiquer avec d'autres stations, pouvoir interroger des bases de données extérieures et se connecter à un système de traduction automatique. MELBY<sup>2</sup> propose un nouveau type de base de données composée de textes sources et de textes cibles manipulés avec des routines de recherche de chaînes de caractères. L'introduction des termes dans la base et leur traduction en sont considérablement simplifiées.

### 2.3.5.4 CAP SOGETI<sup>1</sup>

Comme le GETA de Grenoble, la société Cap SOGETI Innovations a orienté ses travaux vers la réalisation d'une station de traduction ("Translator's workstation language engineering workshop") qui doit offrir des outils "intelligents", un traitement de texte multilingue, une base de connaissances de la langue naturelle et un analyseur.

### 2.3.5.5 DLT (Distributed Language Translation)<sup>3</sup>

Ce projet de recherche et de développement à long terme est une initiative de BSO/Research à Utrecht (Pays Bas). Il a démarré en 1982 par une étude de faisabilité financée par la Communauté Européenne. Financé depuis 1984 à 50% par le ministère des Affaires Economiques et à 50% par BSO, il bénéficie d'un budget annuel d'1 million de dollars, 5 millions de dollars pour les années 1988-1991. La commercialisation n'est pas envisagée avant 1992.

Les caractéristiques essentielles du système sont les suivantes :

- Architecture à langue-pivot<sup>4</sup> : DLT a retenu l'esperanto comme langage pivot. Il s'agit d'une langue artificielle très formalisée qui se prête sans problèmes à l'application des grammaires de CHOMSKY et permet de désambiguïser les textes à un niveau sémantique.
- Ouverture plurilingue : A cet égard, le choix de l'esperanto ouvre des perspectives excellentes à long terme. Les racines de son vocabulaire sont d'origine européenne (langues romanes et langues germaniques), la structure de la phrase est slave, les mécanismes de formation des mots et l'invariance des morphèmes l'assimilent à une langue agglutinante. Les efforts de recherches se sont portés sur le couple anglais-français. Des travaux exploratoires ont été menés sur le finnois, le hongrois, le chinois et le japonais.

(1) S. MELI : "Informationsmarkt der maschinellen Übersetzung" in *Terminologie et Traduction*, Commission des Communautés Européennes, n°3, 1989, pp. 90-91

(2) A. MELBY : "Creating an Environment for the Translator, in : M. KING (Ed.), *Machine Translation Today*, Edinburgh University Press, Edinburgh, 1987

(3) K. SCHUBERT : "Interlingual terminologies and compounds in the DLT project", in *Proceedings of the International Conference on Machine and Machine-Aided Translation*, Birmingham, April 1986

(4) T. WITKAM : "Interlingual Machine Translation - An Industrial Initiative" présenté au Machine Translation Summit, Hakone, Japon, septembre 1987

- Intelligence Artificielle : DLT inclut des techniques de raisonnement issues de l'Intelligence Artificielle (IA) pour réaliser une interprétation basée sur la compréhension. Ces techniques<sup>1</sup> sont conçues pour exploiter au mieux l'architecture à langage pivot et assurer le potentiel d'extension du système.
- Saisie de texte en interactivité : Les concepteurs de DLT pensent, sans doute à juste titre, qu'en dépit des apports de l'Intelligence Artificielle, la traduction automatique de haute qualité n'est pas pour demain. La possibilité d'initier un dialogue entre la machine et l'auteur du texte est un moyen beaucoup plus simple et plus fiable d'utiliser l'intelligence. Au correcteur d'orthographe et au correcteur de style sont associés des outils performants capables d'interroger l'auteur du texte et de lui demander des précisions quant à l'interprétation d'un mot ou d'une phrase. Cette opération de désambiguïsation, lors de l'étape d'analyse ou de transfert, est en langue source. Un système d'apprentissage stocke les données recueillies dans la base de données SWESIL (Semantic Word Expert System for the Intermediate Language).
- Mise en réseau : DLT s'écarte de la configuration traditionnelle des grands systèmes de traduction automatique concentrés sur un gros ordinateur. La traduction se déroule à l'émission et à la réception, au niveau des terminaux mis en réseau. Les données transitent entre les terminaux sous une forme intermédiaire (esperanto). Les terminaux disposent d'un exemplaire du système (sur CD-ROM ou équivalent), d'une banque de connaissances et des dictionnaires afférents à une langue spécifique. On ne partage pas le logiciel ou les capacités de stockage, on partage le langage pivot.

Le niveau de qualité attendu est élevé. Il n'y aura pas de post-editing.

### 3.3.5.6 EDR (Electronic Dictionary Research Institute)<sup>2</sup>

Alors qu'en 1981, le Japon avait lancé le projet ambitieux de l'ordinateur de 5ème génération<sup>3</sup>, 1986 est l'année d'un gigantesque effort national en faveur de la traduction automatique qui avait déjà mobilisé, sans grands résultats et depuis 1950, les équipes universitaires de Kyoto et de Kyushu. Il faut cependant citer ATLAS (2.3.3.2.1) conçu par Fujitsu et disponible depuis 1984, PIVOT (2.3.3.2.7) de NEC, TAURUS de Toshiba et TRANSWORD PRO (2.3.3.2.8) de SANYO. Les insuffisances de ces systèmes de deuxième génération liées à la nature des modèles actuels d'analyse sémantique ont conduit les japonais à créer, en avril 1986, l'Electronic Dictionary Research Institute qui, un peu à la manière de l'ICOT (Institute for New Generation Computer Technology), va gérer un programme<sup>4</sup> de 9 ans et un crédit de 1,5 milliard de francs.

Le programme a été mis sur pieds par huit industriels (Fujitsu, Hitachi, Nec, Matsushita Electronic, Mitsubishi Electric, Oki Electric, Sharp et Toshiba) avec l'aide du JKTC (Japan Key Technology Center), agence gouvernementale pour la promotion de la recherche dans le secteur privé. Le dictionnaire électronique regroupera un dictionnaire maître de 200 000 mots japonais, de 200 000 mots anglais et de 100 000 termes techniques, ainsi qu'un dictionnaire composé de la description des concepts (description du sens de chaque mot du dictionnaire maître sous forme de "frames<sup>5</sup>") et de leur

(1) B. C. PAPEGAALJ : "Word Expert Semantics : an Interlingual Knowledge Based Approach", in : V. SADLER, T. WITKAM (Eds.), Dordrecht/Riverton, Foris, 1986

(2) M. NAGAO, J. TSUJII, J. NAKAMURA : "The Japanese Government Project for Machine Translation" in Computational Linguistics, 11-(2-3), 1986, pp. 91-110

(3) Sciences et Techniques, n° 17, janvier 1985, p. 38

(4) E. LAUNET : "Traduction automatique : Effervescence japonaise" in Sciences et Techniques, n° 35, mars 1987, pp. 34-36

(5) Frame : ce terme désigne en Intelligence Artificielle, un type particulier de représentation des connaissances

classement sous forme de réseaux sémantiques. La gestion de cette énorme base de données fera également l'objet de recherches approfondies. Un réseau à commutation de paquets relie les postes de travail de l'institut et les huit laboratoires industriels. Le réseau est étendu progressivement aux organismes qui souhaitent participer à l'alimentation du dictionnaire (maisons d'édition, centres de recherche...). Un prototype est prévu pour fin 1991. Il tirera sans doute bénéfice de nouvelles architectures symboliques développées à l'ICOT et améliorera la qualité des systèmes de traduction actuels.

### 2.3.5.7 EUROTRA (EUROPEAN TRANSLATION SYSTEM)

Lancé par la Communauté Economique Européenne en novembre 1982, il s'agit du projet de Traduction Automatique le plus ambitieux et le plus important quant au personnel et au potentiel d'extension. L'objectif de ce programme de recherche et de développement est de construire le prototype préindustriel d'un système de T.A. de conception avancée, qui couvre les 9 langues (7 initialement) de la Communauté : allemand, anglais, danois, espagnol, français, grec, italien, néerlandais et portugais. Des équipes universitaires des pays membres (242 personnes, dont 80 % de linguistes, traducteurs, informaticiens et mathématiciens, réparties sur 21 sites) travaillent pour leur langue respective, sur les modules d'analyse et de synthèse. Ils ont à charge également la réalisation du module de transfert vers leur langue. Le projet se déroule en trois phases. Après la détermination du modèle théorique et la conception du formalisme (1982-1984), les études linguistiques<sup>1</sup> (1985-1988) ont préparé l'élaboration d'une maquette de prototype qui devrait clore la phase 3 (1988-1990) et fonctionner avec un vocabulaire de 20 000 mots (dont 15 000 termes techniques du domaine des télécommunications).

Il s'agit d'un système multilingue<sup>2</sup> au sens global du terme, qui traite les 12 langues de la Communauté dans leur ensemble, avec une même approche linguistique. On utilise un modèle unique pour passer d'une langue à une des huit autres et on s'appuie sur une même description pour toutes les étapes de la traduction, que la langue soit source ou cible. Le système est organisé schématiquement en trois étapes : analyse, transfert et génération. L'analyse et la synthèse sont conçues dans un environnement monolingue. Il existe donc 9 modules d'analyse, 9 modules de génération et 72 modules de transfert. Pour que ce dernier soit le plus simple possible, il est indispensable d'éviter des modifications de structure pendant le transfert en analysant la syntaxe et la sémantique des unités de traduction (paragraphes, phrases,...) et en présentant les résultats sous une forme indépendante de la langue (ordre canonique des mots). Il est programmé en C-Prologue dans un style déclaratif basé sur les grammaires d'unification mais devrait être réécrit en langage C à un stade industriel.

L'hypothèse de base est que le processus de traduction est une succession de plusieurs transformations au cours desquelles un texte doit être retranscrit, de sa forme originale (texte source) jusqu'à la sortie correspondante (texte cible) sous des représentations grammaticales successives correspondant à des impératifs linguistiques (niveaux de la morphologie, de la syntaxe de surface, de la syntaxe profonde, de la sémantique). Chaque niveau de représentation est défini par une grammaire ("générateur") composée d'un ensemble de règles ("constructeurs"). L'application de ces règles produit des arbres décorés. Les objets d'un niveau de représentation sont transcrits dans le niveau suivant par un autre ensemble de règles ("translateurs").

(1) EUROTRA : *Reference Manual*. Final Version 5.0, non publié, 1988

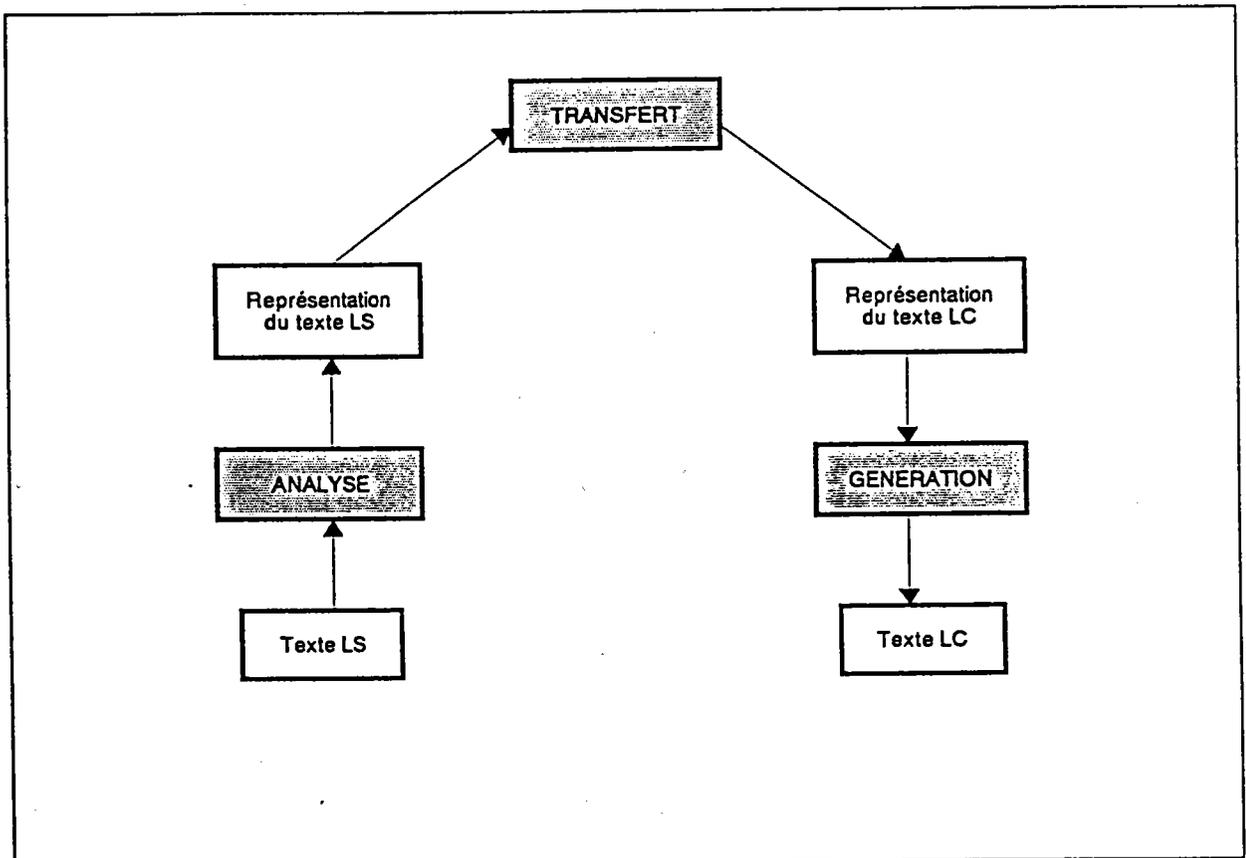
(2) L. DANLOS : "Rapport préliminaire sur Eurotra-France", rapport à usage interne, février 1987

(3) M. KING : "EUROTRA : General System Description", ISSCO, Genève, novembre 1985, ETL-6 : Final report

(4) H.-D. MAAS : "The dictionary in the Eurotra Engineering Framework" in *Sprache und Datenverarbeitung*, 1/1987, Niemeyer, Tübingen, 1987, pp. 15-21

Examinons le système d'un peu plus près. Pour un couple de langues (langue source LS et langue cible LC), trois modules sont responsables du processus de traduction :

- le module d'analyse produit une représentation du texte LS
- le module de transfert transforme la représentation du texte LS en une représentation du texte LC
- le module de génération traduit la représentation du texte LC en un texte LC.



Pour 9 modules d'analyse et 9 modules de génération (ils sont construits pour chaque langue, dans une optique monolingue), nous avons 72 modules de transfert qui permettent de passer de la représentation du texte LS à la représentation du texte LC. Parce qu'ils sont nombreux, ces modules doivent être aussi simples que possible et par conséquent les représentations IS (structure d'interface) aussi proches que possible pour la langue source et la langue cible. La représentation IS décrit la phrase en termes de *prédicats*, d'*arguments* et de *modifieurs*. Elle prend la forme d'un arbre décoré dont les feuilles correspondent aux entrées lexicales des langues source ou cible. C'est au prix de cette abstraction que le système atteindra un niveau supérieur à ce que donnaient jusqu'ici des représentations essentiellement syntaxiques et que l'intégration ultérieure d'une ou plusieurs langues ne devrait pas poser de problèmes insurmontables.

## Analyse

L'analyse s'effectue en trois étapes.

**1. ECS (Eurotra Constituant Structure)** : au cours de cette analyse syntaxique, les mots de la phrases sont étiquetés selon leur catégorie (V, N, Det, Adj. ...) et regroupés en GN (Groupe Nominal), GV (Groupe Verbal) ou GP (Groupe Prépositionnel).

Prenons la phrase *un employé a volé un gâteau au président.*

Le dictionnaire ECS<sup>1</sup> donne la catégorie et les traits flexionnels de chaque mot :

*volé* = verbe, participe passé, masculin singulier

*président* = nom, masculin singulier

*président* = verbe, 3ème personne du pluriel, présent, indicatif

La grammaire ECS dans une version simplifiée donnerait pour la catégorie "phrase" notée (P) :

P -> Advp\* GN [nbre = n] GV [nbre = n]

GN [nbre = n] -> Dét [nbre = n] (Adj) [nbre = n] N [nbre = n]

GV [nbre = n] -> SGV [nbre = n] (GN) (GP) Advp\*

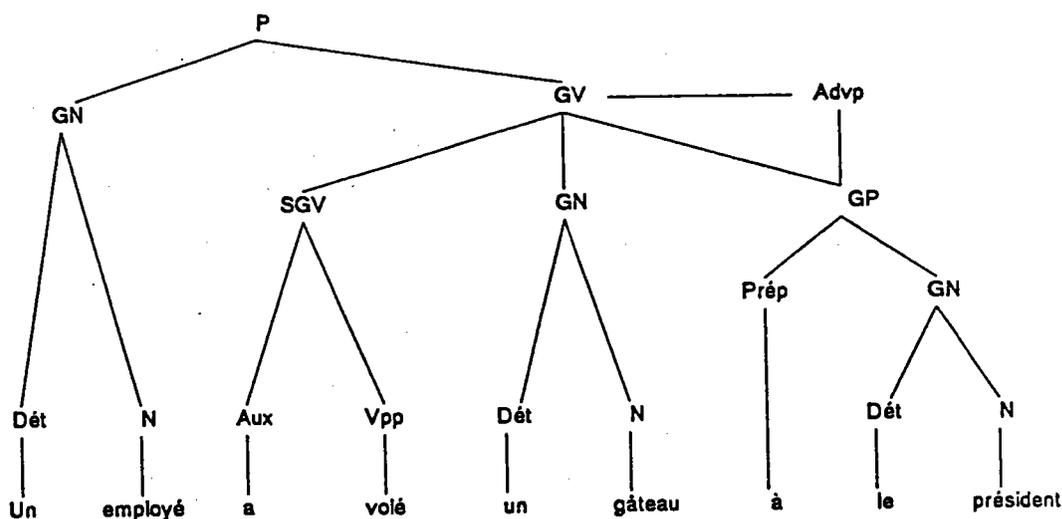
GP ->Prép GN

Advp -> Adv

Advp -> GP

Un constituant entre parenthèses apparaît 0 ou 1 fois. Le signe \* signifie qu'un constituant apparaît 0,1 ou plusieurs fois. Un élément entre [ ] est un couple attribut-valeur. La valeur est une constante ou une variable. Dans l'élément [nbre = n], n est une variable qui indique l'accord en nombre sujet-verbe et l'accord entre les constituants d'un groupe nominal.

L'analyse de la phrase donne les arbres non décorés suivants :

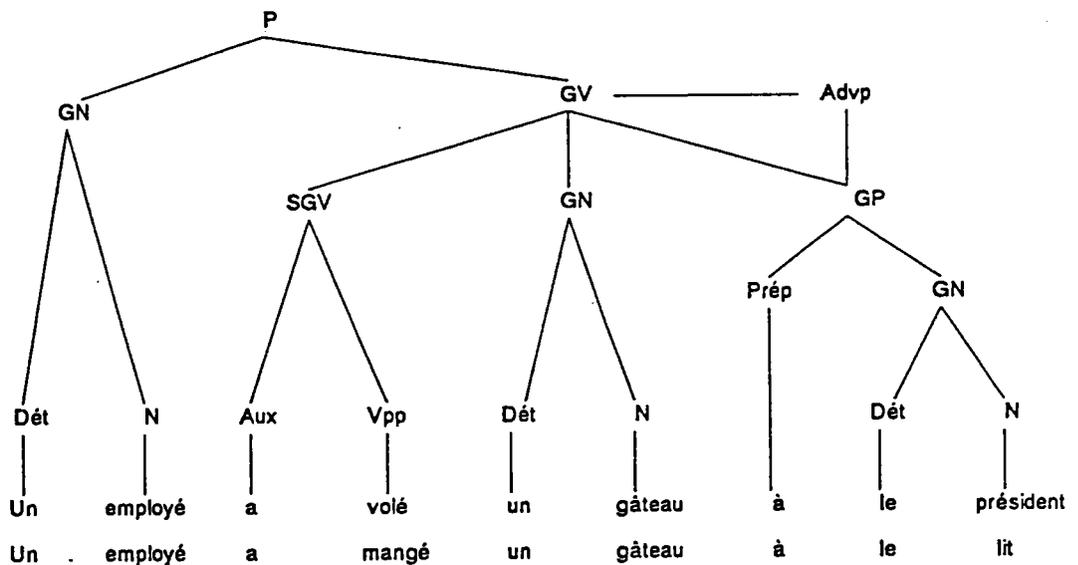


La grammaire évoquée donne deux résultats possibles :

- *au président* est complément de verbe, le noeud GP est un fils du noeud GV
- *au président* est un adverbial, le noeud GP est un fils du noeud Advp.

(1) Exemples extraits de L. DANLOS : "Le projet EUROTRA. Annexe technique" in : A. ABBOU : *Traduction Assistée par Ordinateur*, actes du séminaire international, Paris, mars 1988, pp. 79-86

Au niveau ECS, l'analyse de deux phrases formellement identiques donne le(s) même(s) résultat(s), sans que l'on tienne compte des feuilles de l'arbre. Soit la phrase *un employé a mangé un gâteau au lit..* On obtiendra deux analyses identiques à celles de la phrase précédente *un employé a volé un gâteau au président.*



**2. ERS (Eurotra Relational Structure) :** l'analyse relationnelle réorganise les constituants ECS selon leur fonction syntaxique : verbe, sujet, objet-direct, modifieur...

Le dictionnaire ERS des verbes donne leur complémentation, à l'actif, pour chaque emploi :

- voler1 (*il a volé un jouet à son frère*) : se construit avec un objet-direct et un objet prépositionnel introduit par à. On représente ces indications en écrivant : [ers-frame = dir-obj\_à-objet].
- voler2 (*le commerçant vole sa clientèle*) : se construit avec un objet direct. On écrira : [ers-frame = dir-objet].
- voler3 (*l'aigle vole vers son aire*) : se construit avec un objet locatif. On écrira : [ers-frame = loc-objet].
- manger se construit avec un objet direct : [ers-frame = dir-objet].

La grammaire ERS indique pour une phrase les compléments du verbe qui peuvent apparaître selon sa complémentation :

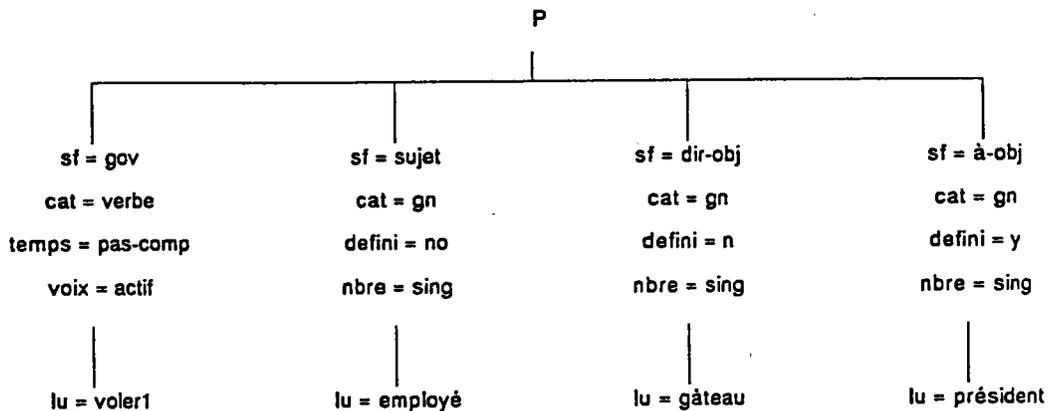
P -> verbe[ers-frame = dir-objet\_à-objet] sujet dir-objet à-objet à-objet Modif\*

P -> verbe[ers-frame = dir-objet] sujet dir-objet Modif\*

Au cours du transfert ECS -> ERS, les informations lexicales du niveau ERS vont permettre d'éliminer l'analyse de *un employé a mangé un gâteau au lit* qui donne *au lit* comme groupe prépositionnel complément de verbe. En effet, *manger* n'accepte pas de à-objet.

Pour la phrase *un employé a volé un gâteau au président*, la grammaire ERS donne trois solutions, deux avec voler1 (*au président* à-objet ou modifieur car le à-objet de voler1 est codé comme facultatif), une avec voler2 (*un gâteau* est objet-direct et *au président* modifieur).

On peut représenter d'une façon simplifiée le résultat de l'analyse pour voler1 avec *au président* comme à-objet.



**3. IS (Interface Structure) :** l'analyse sémantique réorganise les fonctions ERS selon leur rôle sémantique : prédicat (Gov), argument (Argi) et modifieur (Modif).

Le dictionnaire IS indique la complémentation du verbe en termes d'argument ("is-frame") dont il spécifie les catégories distributionnelles.

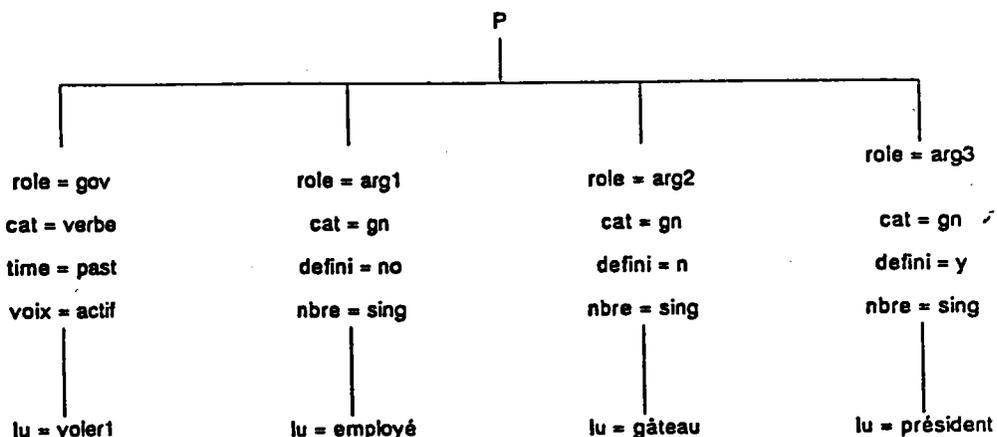
- voler1 a 3 arguments, le premier (sujet) et le troisième (à-objet) sont humains. Cela s'écrit : is-frame = arg123, hum-arg1 = y, hum-arg2 = n, hum-arg3 = y.

- voler2 a 2 arguments humains : is-frame = arg12, hum-arg1 = y, hum-arg2 = y.

Pour les noms, il indique leur catégories distributionnelles.

- employé est un humain : i.c. hum = y  
 - gâteau est non-humain : i.c. hum = no

La grammaire IS garantit pour une phrase qu'il y a compatibilité entre les catégories distributionnelles des compléments de verbe et ces compléments. Elle élimine l'analyse de *un employé a volé un gâteau au président* avec voler2 (2 arguments humains). La grammaire ne valide pas un groupe adverbial de la forme à N où N est humain. L'analyse avec voler1 où *au président* est adverbial est éliminée.

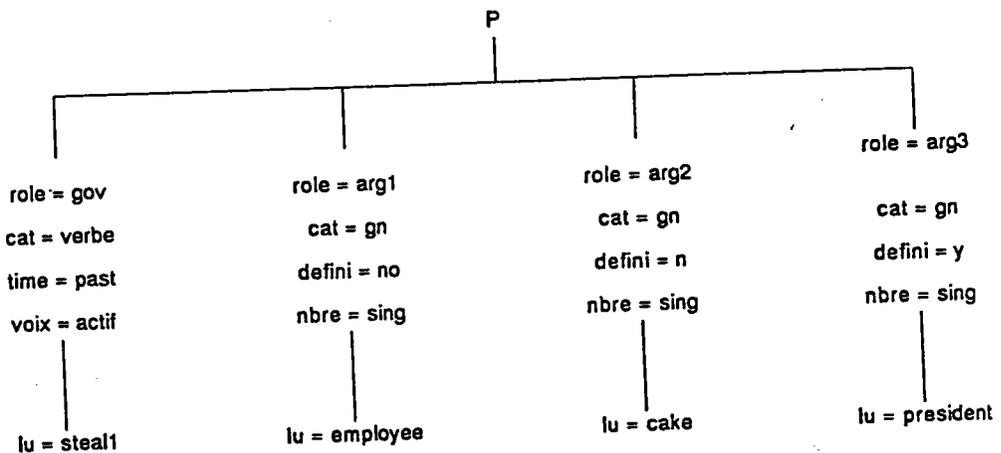


En résumé, à chaque niveau d'analyse correspond un dictionnaire et une grammaire. On obtient à chaque fois un arbre décoré, ou plusieurs s'il y a ambiguïté. Lorsque l'on passe d'un niveau à un autre, ECS -> ERS ou ERS -> IS par exemple, on effectue "un transfert" et on modifie l'arbre décoré.

## Transfert IS -> IS<sup>1</sup>

Lorsque le transfert est simple, l'arbre IS de la langue source et l'arbre IS de la langue cible sont isomorphes. Les feuilles de l'arbre LC représentent la traduction des feuilles de l'arbre LS. Pour la phrase *un employé a volé un gâteau au président* nous obtenons un arbre anglais isomorphe, de sorte que le transfert IS -> se résumera à un transfert lexical.

(lu = voler1) -> (lu = steal1)  
(lu = employé) -> (lu = employee)



Comme les transferts précédents (ECS -> ERS et ERS -> IS) le transfert IS -> IS s'effectue de la même façon. La grammaire et le dictionnaire IS permettent d'éliminer les transferts lexicaux ou syntaxiques qui génèrent des ambiguïtés. Le transfert peut être complexe lorsque les structures IS de la langue source et de la langue cible ne sont pas isomorphes (absence de correspondance lexicale, correspondance lexicale mais nombre différent d'arguments...).

## Génération

On effectue les étapes de l'analyse en ordre inverse :  
IS -> ERS -> ECS -> Texte source

Les dictionnaires et les grammaires de chaque niveau (ECS, ERS et IS), pour une langue donnée, sont utilisés pour l'analyse et la génération. Il n'y a qu'une différence entre l'analyse et la génération, au niveau des transferts :  
Pour ECS -> ERS, en analyse, les compléments du verbe sont rangés dans l'ordre canonique de la phrase à l'actif.  
Pour ERS -> ECS, en génération, l'ordre canonique des compléments est modifié.

(1) J. HALLER : "Anwendung linguistischer Forschungsergebnisse in der Maschinellen Übersetzung : Die Diskussion der Interface-Struktur (IS) in EUROTRA" in *Sprache und Datenverarbeitung*, SDV 1/1987, pp. 8-14

En résumé, nous dirons que quel que soit le niveau de représentation, on utilise un même formalisme pour les grammaires, les dictionnaires et les "transferts", un formalisme déclaratif basé sur les grammaires d'unification. On soulignera que les étapes d'analyse et de génération sont poussées au maximum pour réduire le transfert au minimum, c'est-à-dire à un transfert lexical. L'analyse doit déterminer les structures morpho-syntaxiques (syntaxe de surface) et les structures sémantiques<sup>1</sup> (relations de type grammatical, syntaxique, argumentaire, modal, aspectuel et temporel). L'interface obtenue à la fin de l'étape d'analyse conserve la structure sémantique et les indications lexicales, elle laisse les informations morphosyntaxiques devenues inutiles. La phase de génération s'appuie sur cette interface pour chercher les équivalents dans la langue cible, sans recours donc à la morphosyntaxe de la langue source. Il est malheureusement des situations dans lesquelles on ne peut pas être suffisamment abstrait au niveau de l'interface et où il faut alors recourir à des "stratégies de compensation".

### 2.3.3.8 GETA

En 1961, le CNRS crée le CETA (Centre d'Etudes pour la Traduction Automatique). De 1961 à 1971, le centre élabore de nombreux outils informatiques et les expérimente sur le couple russe-français. De nombreuses expériences sont conduites sur l'analyse de l'allemand et du japonais. En 1971, après de longs développements sur différents types de grammaire et de dictionnaire, un corpus de 400 000 mots est traduit du russe vers le français par un système de seconde génération<sup>2</sup> (analyse morphologique d'états finis, analyse syntaxique par grammaire augmentée indépendante du contexte, analyse sémantique procédurale vers un langage pivot, transfert lexical, génération syntaxique puis morphologique).

En 1972, apparaît le GETA<sup>3</sup> (Groupe d'Etudes pour la Traduction Automatique), qui est un laboratoire du Département d'Informatique de l'université de Grenoble. L'équipe s'oriente vers la mise au point de langages spécialisés pour la programmation linguistique et produit ARIANE, un environnement complet de programmation qui permet de construire des modèles linguistiques. Pour exploiter ses résultats, le GETA participe au PNTAO (Projet National de TAO) et à la construction du système CALLIOPE.

#### 2.3.3.8.1 ARIANE<sup>4,5</sup>

La société B'VITAL, issue du GETA, propose deux types de produits :

- les linguiciels (définition et réalisation de dictionnaires automatiques, conception et développement de grammaires syntaxo-sémantiques, adaptation ou extension de systèmes de TAO)
- les logiciels (outils automatisés pour le codage des dictionnaires, réalisation d'environnements de traduction, conception et réalisation d'ateliers spécialisés pour la programmation linguistique, traitement de texte et de programmes dans un contexte multilingue, constructions de bases de données lexicales...).

(1) G. BOURQUIN : "Le programme EUROTRA : les aspects linguistiques" in *Traduction Assistée par Ordinateur*, sous la direction de A. ABBOU, Editions Daicadif, séminaire international de Paris, mars 1988, pp. 87-88

(2) C. BOIFET, N. NEDOBEJKINE : "Recent Developments in Russian-French Machine Translation at Grenoble". in : *Linguistics* 19, 1981, pp. 199-271

(3) B. VAUQUOIS : *La Traduction Automatique à Grenoble*. Documents de Linguistique Quantitative, n° 29, Dunod, Paris, 1975

(4) M. QUEZEL-AMBRUNAZ : "ARIANE-78. Système interactif pour la traduction automatique multilingue", *Document GETA*, 1978

(5) J.-P. GUILBAUD : "Le modèle allemand-français comme exemple d'application du système ARIANE-78", *11th International ALLC Conference*, Nice, juin 1985

Parmi tous ces produits, elle commercialise ARIANE-78<sup>1</sup>, logiciel de base pour la TAO, sous plusieurs versions :

- pour matériel IBM 43XX, 30XX, 93XX et compatibles
- avec maquette de traduction sur PC/AT 370

Le logiciel de base ARIANE-78 a été développé par le GETA. Il offre un environnement interactif et des métalangages spécialisés pour la programmation linguistique, dans le but de développer des systèmes de traduction. Les langages spécialisés permettent de rentrer des données, d'écrire des grammaires et des dictionnaires qui seront compilés et interprétés par des programmes adaptés. Le système créé est assimilable à un système expert avec la base de connaissances et le moteur d'inférence. C'est un système de deuxième génération, entièrement automatique, qui applique le processus de traduction en trois étapes : analyse, transfert, génération. Bien que né en 1978, il est de conception moderne et traite des structures de données sous forme d'arbres décorés. Chaque noeud de l'arbre comporte un registre (décoration) dans lequel sont stockées toutes les variables d'information fournies au système.

## 1. Analyse

*L'analyse morphologique* est écrite dans le langage ATEF et transforme le texte en un arbre décoré. Chaque mot est éventuellement découpé en radicaux (allemand) et reçoit d'un dictionnaire les informations qui lui correspondent.

*L'analyse structurale* est écrite dans le langage ROBRA<sup>2</sup> et transforme l'arbre plat en une structure intermédiaire source qui donne une interprétation complète de la phrase, sur quatre niveaux :

- classes morphosyntaxiques et syntagmatiques
- fonctions syntaxiques
- relations logiques
- relations sémantiques

Les règles de grammaire expriment les transformations qui seront appliquées localement dans l'arbre.

## 2. Transfert

*Le transfert lexical* transforme chaque noeud de l'arborescence source en une sous-arborescence cible après consultation d'un dictionnaire bilingue écrit dans le langage TRANSF. Il se poursuit au début du transfert structural tant que des tests sont nécessaires sur l'arborescence.

*Le transfert structural* transforme l'arbre intermédiaire source en arbre intermédiaire image en effectuant les opérations suivantes :

- désambiguïsation partielle de l'arborescence en cas de polysémies.
- traitement de l'article
- calcul des modes et des temps
- calcul de pondération pour préparer la mise en ordre syntaxique
- nettoyage et mise en forme de la structure

(1) C. BOITET, P. GUILLAUME, M. QUEZEL-AMBRUNAZ : "ARIANE-78 ; an Integrated Environment for Automated Translation and Human Revision". in : *Proceedings of COLING-82*, Prague, 1982

(2) C. BOITET, P. GUILLAUME, M. QUEZEL-AMBRUNAZ : "Manipulations d'arborescences et parallélisme : le système ROBRA". in : *Proceedings of COLING-78*, Bergen, 1978

### 3. Génération

*La génération syntaxique* génère les feuilles de l'énoncé cible et les ordonne en parcourant l'arborescence de haut en bas de façon récursive.

*La génération morphologique* est écrite dans le langage SYGMOR et utilise les résultats précédents.

L'analyse et la génération sont strictement monolingues. Seul le transfert est bilingue. Le fait que les étapes soient distinctes et qu'ARIANE puisse gérer l'aspect multilingue permet de concevoir des systèmes qui n'utilisent qu'une même analyse pour traduire vers plusieurs langues ou une même génération pour traduire à partir de plusieurs langues.

On constate que le GETA a abandonné le concept de langage pivot qu'il avait retenu pour le projet initial. Son système repose sur des langages spécialisés, (ATEF, ROBRA, TRANSF, SYGMOR). Langage de haut niveau, ROBRA est un transducteur d'arbres utilisé pour l'analyse structurale, le transfert structural et la génération syntaxique. Il utilise des données (variables et arbres décorés avec des opérateurs associés - comparaison, affectation, transformations par règles -) et des structures de contrôle (affectations conditionnelles de variables, application des règles en parallèle, non-déterminisme dans le graphe de contrôle). La caractéristique essentielle de ROBRA est sa récursivité qui permet au système, lorsqu'il emprunte une mauvaise voie dans le graphe de contrôle des sous-grammaires, de revenir en arrière et de tester une autre possibilité. Dans le cas d'une impasse, ARIANE peut interrompre la procédure à différents niveaux et produire des traductions "sous-optimales". La complexité se paye cher puisque la traduction d'un mot nécessite 1,5 million d'opérations.

ARIANE a été retenu pour le projet National de TAO qui a démarré en 1983 et fait l'objet du paragraphe suivant.

#### 2.3.3.8.2 CALLIOPE<sup>1,2</sup>

L'objectif du projet était de réaliser et de commercialiser un ensemble de produits diffusés auprès de l'industrie, à partir des résultats obtenus par le GETA. L'ensemble de ces produits a été baptisé CALLIOPE.

Les caractéristiques originales en sont :

*Abandon du langage pivot* : L'idée d'un langage universel et non ambigu qui permettrait de passer lors de l'analyse, d'un texte source à une représentation en langage pivot, puis, lors de la génération, du langage pivot à la langue cible, a toujours été séduisante. L'intérêt de la stratégie est l'économie que l'on peut alors réaliser dans l'élaboration de systèmes multilingues. Le Projet National contourne les difficultés insurmontables de définir un tel langage en décomposant le processus de traduction en trois phases distinctes : analyse, transfert et génération, le transfert assurant la transition entre l'analyse aussi poussée que possible et la génération.

*Les SLLPs (Specialized Languages for Linguistic Programming)*<sup>3</sup> : Le projet prévoit la mise au point de langages spécialisés pour la programmation linguistique.

(1) O. VAISSADE : "Le projet national de traduction aidée par ordinateur et le système Calliope" in *Encrages*, n°17, Journées européennes de la traduction professionnelle, 1987

(2) C. BOITET : "The French National MT-Project : Technical Organization and Translation Results of CALLIOPE-AERO", *IBM Conference on Translation Mechanization*, Copenhague, août 1986

(3) J. CHAUCHE : *Transducteurs et arborescences. Etude et réalisation de systèmes appliqués aux grammaires transformationnelles*. Thèse d'état, Grenoble, 1974

*Une linguistique développée* permet d'atteindre un meilleur niveau de traduction. Les ambiguïtés sont étudiées le plus tard possible, l'analyse dépasse la syntaxe de surface, l'étude grammaticale très poussée permet d'éviter les dictionnaires gigantesques. Les informations syntaxiques et sémantiques acquises au cours de cette phase et fournies pour le transfert constituent une aide précieuse pour le choix de la traduction juste. En génération, le lexique est organisé en familles dérivationnelles. La compréhension est implicite : elle n'utilise pas de connaissances extralinguistiques entrées sous forme de représentation explicite du domaine traité et ne dispose pas de connaissances pragmatiques.

*Les dictionnaires* sont de deux types :

- le dictionnaire de base contient le noyau de la langue.
- les dictionnaires terminologiques sont spécialisés par domaine.

Ils contiennent les données morphologiques (pluriel, classe de conjugaison, type de déclinaison), les données syntaxiques (liste des constructions possibles pour le terme) et les données sémantiques (traits sémantiques, valences sémantiques).

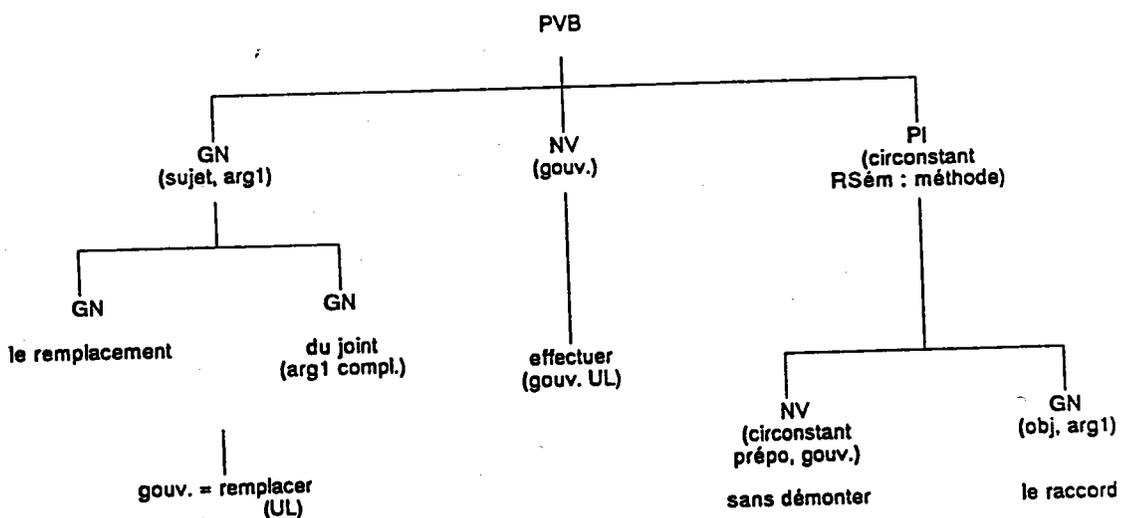
Les termes sont rangés par famille dérivationnelle ou référence lexicale.

### Processus de traduction

*L'analyse morphologique* analyse les termes en recherchant les structures base + désinence (*install-ation*).

*L'analyse structurale* détermine la construction grammaticale de la phrase. L'application des règles de grammaires modifient progressivement l'arborescence. Empruntons l'exemple<sup>1</sup> : *le remplacement du joint s'effectue sans démonter le raccord*

La phrase réflexive passive aura la représentation suivante :



PVB : proposition verbale  
 NV : noyau verbal  
 UL : unité lexicale

GN : groupe nominal  
 PI : proposition infinitive

(1) extrait d'un document réalisé par la société SG2, maître d'oeuvre du Projet National de TAO, février 1987

*L'analyse structurale* cherche à mettre en évidence la syntaxe profonde du texte sous forme de relations logiques et sémantiques.

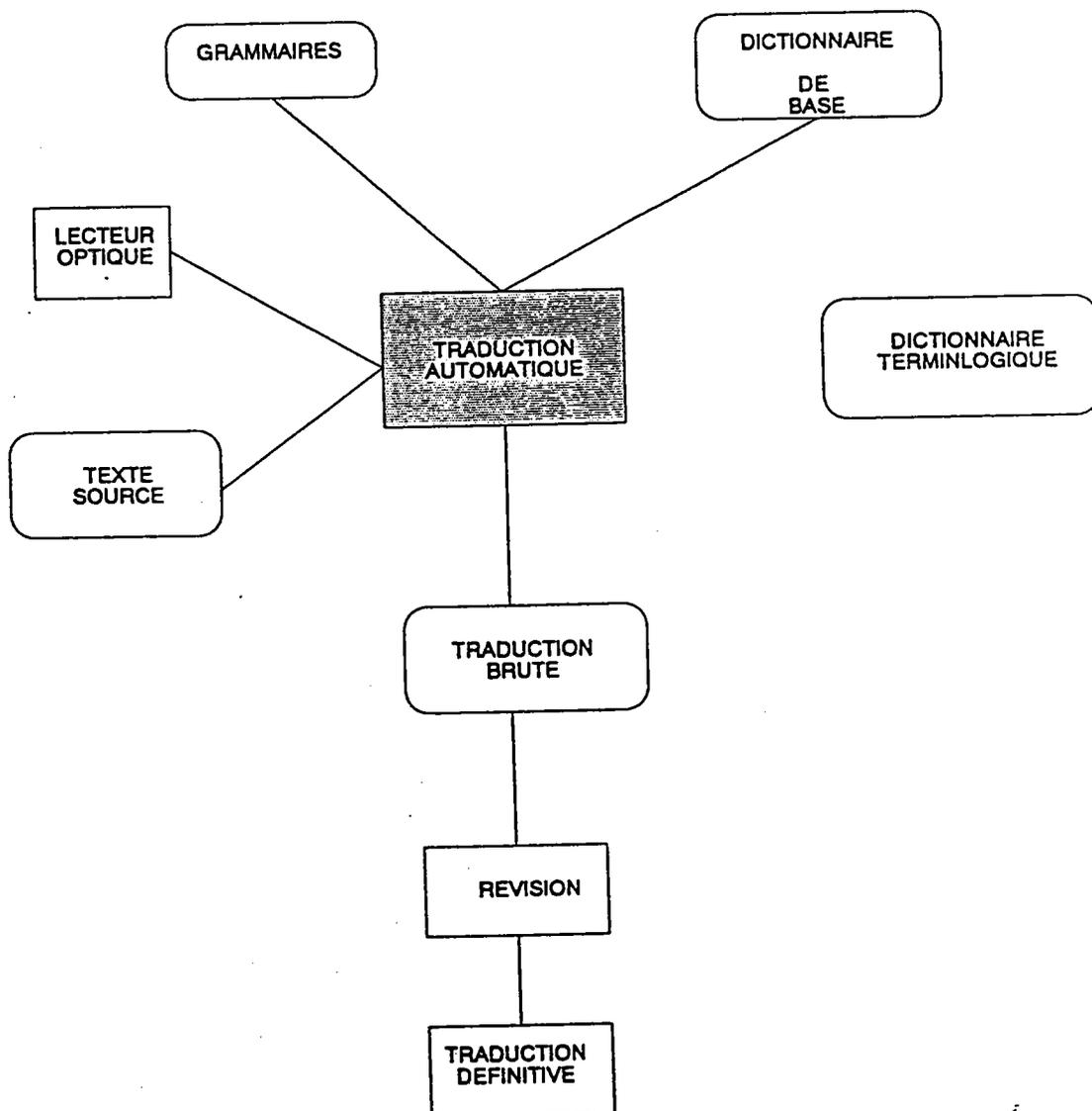
*Le transfert lexical* remplace chaque terme par un ou plusieurs équivalents de la langue cible. Les dictionnaires de transfert déterminent les choix.

*Le transfert structural* parfait le transfert lexical lorsque le choix d'un terme ne peut s'opérer qu'en fonction du contexte et effectue les transformations contrastives entre les deux langues. Si des ambiguïtés subsistent, le système retient la traduction la plus fréquente et proposera les autres au traducteur.

*La génération syntaxique* fixe la syntaxe qui sera retenue dans la langue cible.

*La génération morphologique* donne aux mots leur forme définitive.

Le système CALLIOPE peut être ainsi représenté :



Multilingue, et fondé sur la méthode analyse-transfert-génération, il utilise le concept des systèmes experts :

- un moteur d'inférence ou de déduction qui est indépendant des langues traitées.
- une base de faits qui est la représentation de l'arborescence étiquetée.

- une base de règles ou base de connaissances linguistiques constituée par les grammaires et spécialisée par domaine.

### Fonctionnement

- saisie du texte : OCR (document fourni sur support papier) ou traitements de textes divers selon environnement bureautique de l'entreprise.
- préparation éventuelle du texte s'il a été saisi par OCR ou si l'on désire ne pas en traduire une partie.
- recherche des mots inconnus au cours de l'analyse morphologique du texte. Obtention d'une liste des mots inconnus triés par fréquence, ordre alphabétique et contexte.
- indexation interactive de la terminologie
- traduction automatique
- révision de la traduction brute sur écran partagé
- traduction manuelle si cela est souhaité

L'ensemble CALLIOPE, est constitué des produits suivants :

- **CALLIOPE-SYSTEME** : Moteur de traduction basé sur un système-expert indépendant des langues traduites. Les bases de connaissances linguistiques sont limitées à l'aspect linguistique et ne tiennent pas compte du domaine. Le système fonctionne sur matériel IBM sous VM/CMS. La puissance minimale du matériel est de 1 MIPS, (1 million d'opérations par seconde).
- **CALLIOPE-REVISION** est un environnement de travail pour traducteur. Une pré-édition est nécessaire pour isoler une portion de texte qui ne doit pas être traduite ou corriger un document saisi par lecteur optique. Le traducteur peut dialoguer avec le système pour corriger la traduction brute. L'affichage se fait sur écran partagé avec un défilement synchronisé des textes source et cible. Les dictionnaires sont accessibles pendant la révision ou pendant une phase de traduction manuelle. Les accès peuvent être soumis à un système de priorités pour contrôler les mises à jour. Le produit s'insère sans problème dans le système bureautique d'une entreprise. Des systèmes de conversion assurent la compatibilité des traitements de textes et la saisie par OCR permet d'utiliser des documents fournis sur support papier. Des procédures de partage d'écran et de manipulation de chaînes de caractères donnent la possibilité au traducteur de travailler en manuel, sans recours à la traduction automatique. Le système est opérationnel sur micro-ordinateur BULL QUESTAR 400. Une installation comprend une à quatre stations de travail qui peuvent partager des ressources communes (dictionnaires, imprimantes, lecteur optique - DEST 2XX de la société WALTON -, télécommunications).

Le système de traduction se limite aux traductions techniques. Il est spécialisé par domaines.

- **CALLIOPE-AERO** : les modules linguistiques de CALLIOPE sont spécialisés par domaines. Le module spécialisé en aéronautique traduit du français vers l'anglais.
- **CALLIOPE-INFORMATIQUE** : Spécialisé en informatique, il traduit de l'anglais vers le français. Ces deux systèmes sont portables, c'est à dire qu'ils peuvent être implantés sur n'importe quelle machine. Les logiciels sont écrits en LISP.

La phase d'expérimentation a montré que le système impliquait un gain évident au niveau de l'homogénéité des traductions et que le niveau de traduction était inégal selon les textes. A la suite de problèmes d'organisation, ces travaux ont été arrêtés en 1987.

## Exemple de traduction par CALLIOPE-AERO :

ILLUSTRATION DE LA PUISSANCE LINGUISTIQUE ET DE LA  
TYPOLOGIE DE CALLIOPE

Effectuer un examen complet de l'échangeur de température et s'assurer de l'étanchéité dans le circuit d'air à refroidir.

*Examine in a full [AMBIGU : complete] manner the temperature intersection and check the leaktightness in the air flow to be cooled.*

Effectuer la vidange générale et la purge de carburant (voir chapitre 12).

*Perform general defuelling and the fuel bleeding (see chapter 12).*

Ouvrir progressivement le robinet (3) jusqu'à obtenir une pression de 9 bars.

*Gradually open tap (3) until a pressure of 9 bars is obtained.*

Brancher une source extérieure de courant mais sans alimenter les réseaux de bord.

*Connect [AMBIGU : plug in] external power supply but without applying power to the aircraft systems.*

Avant de déposer ou de reposer le panneau central intrados de voilure, il est nécessaire de procéder à certaines modifications.

*Before removing or reinstalling the lower central wing [AMBIGU : \*] panel, it is necessary to proceed with some modifications.*

La pose du bouchon de protection s'effectue en comprimant le ressort par déplacement de la bague.

*The protection trap installation is carried out compressing the spring by ring movement.*

08 Vérifier, à chaque poste, l'allumage du voyant CAUTION ambre.

*08 Check, at each station, that the amber CAUTION light is switched on [AMBIGU : ignited].*

SSS 08 Vérifier, à chaque poste, l'allumage du voyant CAUTION ambre.  
SSS

*[Inconnu : 08 Vérifier à chaque poste, l'allumage du voyant CAUTION ambre].*

Vérifier le bon fonctionnement de la jonction bloc frein et raccord de tuyauterie.

*Check the validity of the junction block brake and piping coupling.*

Soumettre le raccord à une pression de 3 + 0.5 bars.

*Subject the coupling to a pressure of 3 + 0.5 bars.*

Matériel nécessaire.  
Un bouchon 15-88774-101  
Liquide d'utilisation AIR3520B

*Necessary equipment.  
A trap 15-88774-101  
Use fluid AIR3520B.*

Ouvrir la porte.  
L'opérateur la porte sur le support.

*Open the door.  
The operator carries [AMBIGU : wears] it on the support.*

2.3.3.9 LUTE (Language Understander Translator Editor)<sup>1</sup>

Ce projet de NTT (Nippon Telegraph and Telephone) a démarré en 1981 et constitue une des expériences les plus avancées actuellement. LUTE est un système à transfert, pour les couples japonais-anglais et anglais-japonais, basé sur les techniques de l'Intelligence Artificielle. Il s'agit avant tout de recherche fondamentale destinée à définir le rôle des connaissances linguistiques et non-linguistiques dans chaque phase de la traduction.

(1) H. NOMURA, S. NAITO, Y. KATAGIRI, A. SHIMAZU : "Translation By Understanding : A Machine Translation System LUTE", *Proceedings of the 11th ICCL, COLING 86, Bonn, West Germany, 25-29 August 1986*, pp. 621-626

### 2.3.3.10 MARIS (Multilinguale Anwendung von Referenz-Informationssystemen)<sup>1</sup>

Ce projet a démarré en 1985 à l'université de Sarrebruck, avec l'ambition de développer un système multilingue de traitement de l'information à l'usage de germanophones qui souhaiteraient consulter des documents rédigés en anglais. Les premiers travaux concernent un système de traduction assistée par ordinateur appelé STS (Saarbrückener Translation Service), pour le couple anglais-allemand seulement. Son champ d'action est limité aux résumés et aux titres d'articles de revues.

Le développement est prévu en trois phases :

- système de traduction manuelle avec la possibilité pour le traducteur d'accéder à des banques terminologiques automatisées.
- recherche automatique de la terminologie
- système de traduction complètement automatique, appliquant la méthode du transfert et basé sur le système expérimental SUSY, développé à Sarrebruck. Domaines privilégiés : construction, environnement, normes, sciences sociales. Langues prévues : français-allemand, allemand-français.

### 2.3.3.11 Autres projets<sup>1,2</sup>

Il est impossible d'être exhaustif en la matière. Les projets cités en 3.3.5 sont les plus intéressants ou les plus importants. Certaines langues orientales commencent à faire recette et concernent des projets que nous ne mentionnerons pas tous. Nous pouvons cependant ajouter, dans l'ordre alphabétique :

- AIDTRANS : système de TAO, japonais-anglais, université de Sheffield
- ASCOF<sup>3</sup> : système français-allemand (recherche pure), université de Sarrebruck
- ATT (Automatic Translation Telephone) : projet de KTC (Key Technology Centre), NTT (Nippon Telephon Technologies) et KDD (communications internationales au Japon).
- CADA (Computer Assisted Dialect Adaptation) : système de traduction assistée par ordinateur pour des dialectes voisins, Amérique du sud.
- **Université de Carnegie-Mellon :**  
un projet de traduction automatique avec base de connaissances,  
un projet conjoint avec IBM et Hewlett-Packard,  
un projet avec des fabricants japonais d'ordinateurs dans le domaine de l'Intelligence Artificielle et de la compréhension de textes.
- **Chine :** projets conduits par des universités et des organismes d'état. A Peking, un système anglais-chinois, à Shangaï, un système interactif chinois-anglais, à Taïwan, un système anglais-chinois.
- CONTRAST (Context Translation System) : système expérimental, Tokyo.

(1) S. MELI : "Informationsmarkt der maschinellen Übersetzung" in *Terminologie et Traduction*, Commission des Communautés Européennes, n°3, 1989, pp. 97-105

(2) "Ten Years of Translating and the Computer, Report on the tenth annual Translating and the Computer conference, London, November 1988, in *Language International*, vol. 1, Issue 1, 1989

(3) A. BIEWER, C. FENEYROL, J. RITZKE, E. STEGENTRITT : "ASCOF - A modular Multilevel System for French-German Translation" in *Computational Linguistics*, vol.11, April-September 1985

- **CSK (CSK Research Unit)** : système anglais-japonais pour les besoins propres de la société.
- **ENTRA** : projet de système anglais-norvégien en collaboration avec WEIDNER, à l'université de Bergen, en Norvège.
- **IBM (Espagne-Israel)** : système anglais-espagnol et anglais-hébreux pour les besoins propres d'IBM.
- **ICL (International Computers Ltd.)** : projet anglais de traduction automatique des Telex, anglais-français, anglais-allemand
- **NAAAT (The National Agency for the Assessment and Application of Technology)** : projet indonésien d'un système de TAO anglais-indonésien.
- **JETR** : projet de système de traduction avec base de connaissances, université de Californie (Irvine).
- **KIT (Künstliche Intelligenz und Textverstehen)** : Technische Universität Berlin.
- **Corée** : développement de nombreux projets, coréen-anglais et anglais-coréen surtout, dont un entre IBM et l'université de Seoul, anglais-coréen et basé sur la méthode de transfert.
- **LAMB** : projet de CANON (Japon), japonais-anglais, basé sur la méthode de transfert et l'utilisation d'une base de connaissances.
- **Malaisie** : collaboration entre l'université de Malaisie et le GETA de Grenoble pour un système anglais-malais adapté à des textes d'informatique et la mise au point d'une station de traduction.
- **NASEV (Neue Analyse- und Syntheseverfahren zur maschinellen Übersetzung)** : projet de recherche théorique (dans le cadre d'EUROTRA) du BMFT ((Bundesministerium für Forschung und Technologie) réunissant les universités de Berlin, Bielefeld et Stuttgart.
- **NARA** : projet de l'université de Tokyo, sur le coréen-japonais et le japonais-coréen.
- **NIPPON** : système à usage interne de NDGC (Nippon Data General Corporation, anglais-japonais).
- **NTRAN** : système interactif anglais-japonais de l'UMIST (University of Manchester Institute of Science and Technology).
- **PAROLE** : système à transfert, japonais-anglais, de Matsushita (Japon)
- **ROSETTA** : système multilingue expérimental de Philips à Eindhoven aux Pays-Bas. Un des rares exemples de projets, hors Japon, impliquant l'industrie (tout comme SIEMENS et IBM). Les travaux ont commencé en 1980 pour aboutir à une première tentative de traduction néerlandais-anglais et néerlandais-espagnol (ROSETTA 3). Le développement du système opérationnel est l'objectif de ROSETTA 4.
- **SEMSYN (Semantische Synthese)** : à Stuttgart, système de traduction automatique des titres de revues sur les technologies de l'information, japonais-allemand.

- **SEPPLI** : projet de l'ISSCO (Institute for Semantic and Cognitive Studies) à Genève d'un système de traduction automatique d'offres d'emploi, allemand-français, allemand-italien.
- **SERI** (System Engineering Research Institute) : projet commun du GETA avec la Corée d'un système anglais-coréen, français-coréen.
- **SUSY, SUSY II<sup>1</sup>** : projet de système multilingue à transfert (université de Sarrebruck).
- **SWETRA** : projet suédois de système multilingue.
- **TEXTUS** : système anglais-français, français-anglais de l'université de Georgetown.
- **TRANPRO** : projet anglais d'un ensemble de traitement de texte multilingue.
- **Thaïlande** : projet commun avec la Malaisie d'un système anglais-thaï.
- **Université de Prague** : Lexiques et traduction assistée par ordinateur.
- **TUMTS** : université de Tamil, (Inde du sud), système de traduction directe, pour le couple russe-tamoul, adapté à des textes sur l'astronomie.
- **VALANTINE** : système à usage interne, KDD (Japon).
- **XTRA** : université du Nouveau-Mexique, système anglais-chinois.

## 2.4 Les services de traduction

En dehors d'institutions nationales, de bureaux privés de traduction plus ou moins importants et de traducteurs "libres" utilisant des systèmes de TA ou de TAO, nous citerons 7 cas qui semblent confirmer les prévisions de F. ZIRKLE, président de ALPS, et selon lequel les bureaux certes compétents mais de petite taille et dispersés seront progressivement remplacés par des sociétés mondiales, à intégration verticale, avec des équipes importantes de traducteurs "maison" et une gestion globale.

1. **ALPNET** : Cet ensemble de 9 sociétés de traduction, avec 22 bureaux répartis dans le monde propose à ses clients des traductions réalisées en partie avec le système ALPS (2.3.3.4.1).
2. **BRAVICE** : Créée par T. Yamamoto<sup>2</sup> après le rachat du système WEIDNER en 1979, la société est implantée à Tokyo et propose des traductions sur le système WEIDNER (2.3.3.2.11) qu'elle commercialise et qu'elle a vendu à plus de 3000 exemplaires dans le monde. Des bureaux ont été ouverts à Londres, Toronto, Miami et Shanghai en joint-venture avec une compagnie chinoise, en attendant Paris, Seoul et Francfort.
3. **CSATA** : Ce centre italien travaille avec SYSTRAN (2.3.3.2.9)
4. **ECAT** (European Centre for Automatic Translation) : Le centre est au Luxembourg.
5. **Mendez Service Bureau** : implanté à Bruxelles et propose des traductions sur SYSTRAN.
6. **SOCATRA** (Société Canadienne de Traduction Assistée) : La société implantée au Canada à Montréal dispose de son propre système (SOCATRA). Mis au point par son fondateur, C. RICHARD, il fonctionne sur le couple anglais-français à une vitesse de

---

(1) H.-D. MAAS : "SUZY I und SUZY II, Verschiedene Analysestrategien in der maschinellen Übersetzung" in *Sprache und Datenverarbeitung*, SDv 1-2/1981, pp. 9-15

(2) *Dynasteurs*, novembre 1987, pp 54-55

60 000 mots/heure. Le client fournit le texte à traduire et récupère une sortie brute qu'il peut réviser lui-même ou qui sera révisée sur place par une équipe de traducteurs.

7. SYSTRAN S.A. : à l'initiative de J. Gachot, le système SYSTRAN est disponible par Minitel ou accessible par modem.

## 2.5 Les banques de données terminologiques

Utilisées par le traducteur, elles n'ont bien sûr rien à voir avec les dictionnaires des systèmes de traduction automatique. Leur contenu se limite aux formes de base étendues à des définitions et des mises en contexte.

Le *Directory of Online Databases*<sup>1</sup> recense 3800 banques de données dont 3 seulement ne sont pas monolingues et s'adressent aux traducteurs. Il faut cependant souligner que le classement du Directory repose sur des critères contraignants. Il existe en fait un plus grand nombre de banques multilingues. Pour ne citer que les plus importantes :

- AEROSPACE TERMINOLOGY : anglais, français et allemand
- EUROCAUTOM : toutes les langues de la Communauté Européenne, pour tous les domaines.
- LEXIS : Bundessprachenamt à Bonn.
- SMART TRANSLATORS : expressions techniques de l'anglais vers l'allemand, l'espagnol, l'italien et le russe.
- TEAM chez SIEMENS
- TERMIUM à Montréal

## 2.6 Conclusion

L'idée de traduction automatique remonte à l'origine des ordinateurs. Elle ne jouit cependant d'une reconnaissance scientifique que depuis quelques années, grâce à une poignée d'applications réussies, une augmentation de la demande (les besoins augmentent de 15% par an) et une évolution du type de traductions exigées.

Les systèmes existants ont pour la plupart vu le jour aux Etats-Unis, entre les années 60 et 70. Ils entrent dans deux catégories de base : les systèmes complexes implantés sur des gros calculateurs, centralisés et interrogés à distance (LOGOS, SPANAM et SYSTRAN), et les systèmes plus simples (SMART, WEIDNER et LINGUISTIC PRODUCTS) qui fonctionnent sur des micros, chez l'utilisateur.

La qualité des résultats dépend des couples de langues traitées, du type de document et de la base terminologique disponible. La couverture des langues est assez complète, la majorité des systèmes opérationnels couvrant le français et l'anglais dans les deux sens, l'allemand et l'espagnol en langue source ou en langue cible. L'anglais est la langue source la plus développée.

Les utilisateurs appartiennent à deux catégories : ceux qui analysent les textes en langue étrangère pour des besoins d'information (l'US Air Force avec SYSTRAN depuis 1970, à partir du russe, du français et de l'allemand et le Centre de Recherches nucléaires de Karlsruhe, du français vers l'anglais) et ceux qui traduisent des documents dans l'optique d'une diffusion multilingue (traduction des manuels de maintenance chez XEROX, FORD, IBM, Dornier et Siemens). Dans le premier cas, la traduction brute est confiée à des experts des différents domaines qui se contentent de traductions de moyenne qualité sans révision humaine, dans le second cas, un niveau de qualité élevé est exigé, qui sous-tend une préparation soignée des documents sources, une termi-

---

(1) *Directory of Online Databases*, Vol. 9, n° 3, juillet 1988

nologie technique fiable et une post-édition humaine. Les avantages essentiels sont la rapidité, la cohérence et le coût. En ce qui concerne le secteur public, les grandes institutions ont recours à la T.A. (l'OTAN, certaines agences de l'ONU et la Commission Européenne) pour traduire des rapports techniques, des documents administratifs et des compte-rendus de réunion. La qualité des traductions brutes est rarement satisfaisante, mais une post-édition rapide (4 pages/heure) fournit des résultats acceptables. En France, la traduction automatique a fait son apparition sur le réseau Minitel où GACHOT S.A. propose des services en direct sur SYSTRAN. La majorité des textes soumis ressemblent à ceux du Minitel Rose<sup>1</sup>. L'originalité du vocabulaire explique sans doute la qualité médiocre des traductions.

Les tendances sont inégales. En général, les fabricants de systèmes ont été dépassés par les événements. Leurs investissements sont allés au delà du potentiel qu'offre le marché actuel. LOGOS a perdu de gros clients au Canada, ce qui l'a obligé à réduire ses efforts de développement linguistique. WEIDNER a cessé ses activités en Amérique du Nord et en Europe, la société mère (BRAVICE) continuant au Japon. ALPS plutôt spécialisé en T.A.O. a également dû se limiter à des services généralisés en traduction et ne vend pratiquement plus de logiciels. SMART a beaucoup de succès en Amérique du Nord, pour la traduction de manuels de maintenance technique et d'avis de vacances d'emploi. Il offre depuis peu le couple anglais-grec. Le logiciel développé par la firme texane Linguistic Products, qui tourne sous MS-DOS pénètre rapidement le marché grâce à un prix très compétitif (100 logiciels vendus en 1988). SYSTRAN progresse à l'OTAN, chez XEROX et dans l'US Air Force. Les grandes entreprises japonaises, enfin, ont toutes développé des systèmes qui sont maintenant opérationnels pour les couples anglais-japonais et japonais-anglais.

La recherche évolue. Les avancées les plus notables sont à porter au crédit des principaux fabricants japonais d'ordinateurs. Les systèmes anglais-japonais et japonais-anglais sont nombreux. Fujitsu est au dessus du lot avec son système ATLAS. En Europe, le projet principal est EUROTRA, cofinancé par la Communauté Européenne et ses états membres. Des systèmes pilotes sont attendus pour 1995. Il ne faut cependant pas oublier d'autres projets avec DLT aux Pays-Bas et METAL créé au Texas mais maintenant développé par Siemens à partir de l'allemand. Aux Etats-Unis, IBM s'intéresse de nouveau à la T.A. et travaille à partir de l'anglais vers l'espagnol, l'hébreu et le finlandais. Dans l'ensemble, les résultats sont décevants, (le projet CALLIOPE a été suspendu, les systèmes japonais butent sur de nombreux obstacles et EUROTRA connaît des difficultés de coordination en ce qui concerne les travaux de ses différentes équipes). Tout ceci explique sans doute pourquoi les approches classiques ont toujours rencontré plus de succès que les stratégies d'innovation reposant sur des théories linguistiques difficilement programmables.

Si l'on souhaite dresser une échelle de réussite de tous ces systèmes, l'examen des résultats conduit à fixer un ensemble de critères d'évaluation<sup>2</sup>. Il est difficile cependant d'effectuer une analyse complète et fiable des documents, dont on regrettera qu'ils ne soient pas toujours accessibles et significatifs. La matière est au contraire abondante en ce qui concerne les systèmes "reconnus". On constate alors que les systèmes à sémantique fermée (TAUM) ou à syntaxe contrôlée (TITUS) s'imposent devant SYSTRAN qui, si on voulait l'affranchir de ces contraintes, deviendrait un système gigantesque. C'est ce qui fait l'intérêt de notre Automate de Compréhension Implicite, qui accepte une sémantique ouverte et fournit une sortie moins ambitieuse.

---

(1) Entretien avec Ian M. PIGOTT, responsable des développements SYSTRAN, Commission des Communautés européennes

(2) G. Van SLYPE : "Conception d'une méthodologie générale d'évaluation de la traduction automatique" in *Multilingua*, Mouton, 1-4/1982

### **III.**

**UN SYSTÈME D'INTERPRÉTATION AUTOMATIQUE :  
L'AUTOMATE DE COMPRÉHENSION IMPLICITE (ACI)**

Les deux premiers chapitres donnent une idée des techniques utilisées en T.A.L.N., de leurs applications et des résultats obtenus. Dans ce chapitre III, nous proposerons une autre approche, avec l'Automate de compréhension Implicite (ACI). Ce système a été mis au point dans le cadre du Centre de Recherche Jean Favard. Nous le situerons d'abord par rapport à la T.A. et à la T.A.O., avant d'en détailler les principes, les caractéristiques puis la structure d'ensemble. La réalisation complète n'a pas été menée à terme, aussi les sorties que nous présentons au paragraphe 3.2.2.2 et en conclusion (p. 668) ne sont-elles qu'une simulation. Le chapitre IV sera consacré à la description exhaustive des modules responsables de l'analyse, dans le contexte germanique-roman, pour respecter les limites de notre sujet.

### 3.1 Le contexte

#### 3.1.1 Analyse des systèmes et évolution

##### 3.1.1.1 De 1950 à 1970

Au cours de ces 20 premières années, le domaine de la TA a connu des fortunes diverses (2.3.1), allant de l'euphorie générale à une paralysie quasi-totale (Rapport AL-PAC). L'idée était d'employer la rapidité de calcul et la mémoire des ordinateurs pour associer un mot d'une langue à un mot d'une autre langue puis d'obtenir une traduction automatique à peu près correcte, moyennant un léger réarrangement (ordre des mots en particulier). *Il est clair que la traduction ne concerne ici qu'un couple de langues et ne s'effectue que dans un sens. L'analyse de la langue source dépend de la langue cible.*

E. REIFLER tente une traduction automatique à l'aide de la machine de KING<sup>1</sup> à l'Université de Wahington (Seattle). Il s'agit d'une traduction mot à mot réalisée par un automate transducteur d'états finis qui lit en entrée un symbole et renvoie une chaîne de sortie. C'est en fait une consultation de table (un lexique) qui possède a priori le meilleur correspondant pour un mot polysémique. *Un tel système ne peut lever les ambiguïtés.*

Le système de KULAGINA et MEL 'CUK est infiniment plus évolué. Il fonctionne sur un ordinateur classique des années 1955-1960 et traite un corpus de textes mathématiques (250 expressions idiomatiques et 1300 mots différents). Le processus en deux étapes (analyse du texte d'entrée et synthèse du texte de sortie) préfigure les méthodes de 2<sup>ème</sup> génération. La partie lexicologique est organisée en 3 dictionnaires : français, russe et dictionnaire d'expressions idiomatiques. Le premier dictionnaire contient des bases associées à des indications concernant leur identification, le caractère idiomatique, l'existence d'homonymes, de traductions multiples le cas échéant, les traits morphologiques et syntaxiques. Le programme réalise une traduction en 5 étapes (recherche des mots dans le dictionnaire français, traitement des expressions idiomatiques, résolution des homographies, analyse de la phrase française, synthèse de la phrase russe).

Le système de la Georgetown University, opérationnel depuis 1963-1964 est d'un type voisin. Il repose sur un dictionnaire de bases qui peut posséder plusieurs mots équivalents et un processus qui, après consultation du dictionnaire, réalise une analyse morphologique, une analyse syntaxique (traitement des constituants contigus) suivie d'un transfert vers l'anglais, d'un réarrangement des mots et insertion de particules. *Ce système a la particularité de pouvoir traiter les mots inconnus. L'ensemble de travaux qui concernent cette période se caractérise par une focalisation sur les problèmes lexicaux et une absence complète de théorie linguistique.*

---

(1) La machine photocopique de G. KING (IBM) est à notre connaissance la seule tentative "hardware" de traduction automatique.

Un commentaire de L. DOSTERT, cité par B. VAUQUOIS<sup>1</sup> traduit les faits : "La traduction est une affaire de mémoire (storage) et d'intelligence ; la mémoire, c'est le dictionnaire ; l'intelligence, c'est le programme". Remarquons au passage que le rapport ALPAC paru en 1966, c'est à dire au milieu de la période de 2<sup>ème</sup> génération, ne concerne pas du tout celle-ci, mais uniquement les travaux de 1<sup>ère</sup> génération pour lesquels *toute insertion dans les programmes provoque des réactions insoupçonnées et dont la perfectibilité est effectivement très douteuse.*

Les études de D.G. HAYS sur la théorie des constituants et sur celle des dépendances pour le calcul des structures syntaxiques vont contribuer au lancement de la seconde génération, caractérisée par la distinction des trois phases : ANALYSE, TRANSFERT et GENERATION. *Les tâches sont désormais séparées et indépendantes. La phase de transfert est seule à concerner un couple de langues, le choix d'un pivot judicieux permettant cependant une traduction multilinguale, quelles que soient les langues d'entrée et de sortie.* Nous pouvons schématiser la structure d'un système de 2<sup>ème</sup> génération de la façon suivante :

### Analyse

- Analyse morphologique : Le texte d'entrée est considéré comme une suite de graphèmes (y compris le caractère blanc). On passe du niveau graphémique au niveau des morphes pour lesquels on peut constituer des dictionnaires ("verbal", "verbale", "verbaux", "verbales" correspondent aux morphes "verb", "al", "ale", "ales" et "aux")<sup>2</sup>. L'analyse a pour effet de substituer au mot une expression formée de différents éléments : une référence lexicale qui correspond dans certains systèmes à une base de dérivation, un indice de dérivation qui précise le type de dérivation, une classe syntaxique, une liste de variables grammaticales, syntaxiques (reactions) et sémantiques (traits sémantiques). Le dictionnaire morphologique comprend une zone d'entrée (avec le morphe) et une zone d'information (code morphologique, code syntaxique, code d'étiquetage). Il se présente sous deux formes : bases triées par ordre alphabétique et par numéro d'unité lexicale. L'algorithme d'analyse repose sur une consultation du dictionnaire et une élimination des ambiguïtés à l'aide de l'indice de dérivation.

- Analyse syntaxique : En dehors des grammaires "en chaîne" (1.4.3.2.3), deux théories linguistiques ont la faveur des équipes pour l'étude de la syntaxe de surface : l'analyse en constituants immédiats largement utilisée aux USA et l'analyse en structures de dépendances privilégiée en URSS.

*L'analyse en constituants immédiats* repose sur les grammaires "hors contexte" (1.4.3.1.2) qui permettent des descriptions sous forme parenthésée. La principale difficulté réside dans l'élimination des structures parasites.

*L'analyse en structures de dépendances* permet de marquer qu'un mot dépend d'un autre. Elle repère sa situation linéaire par rapport à son gouverneur. Une règle initiale donne l'élément sommet du graphe (prédicat de forme verbale dans la majorité des cas). L'intérêt de débiter les recherches avec l'élément sommet du graphe est qu'il possède des relations bien étudiées avec les éléments immédiatement dépendants (arguments - compléments du verbe - appartenant au cadre verbal et soumis à des réactions connues). On dispose d'une représentation parenthésée.

(1) B. VAUQUOIS : *La traduction automatique à Grenoble*, Documents de Linguistique Quantitative n°24, Paris, 1977, p. 29

(2) cf. supra, p. 37

Outre les analyseurs par relations de dépendances (KULAGINA à Moscou, TSEITIN et FITIALOV à Leningrad), on trouve deux autres types d'analyseurs :

*L'analyseur prédictif* (1.6.3.3) explore une phrase de la gauche vers la droite en avançant d'un symbole terminal qui doit réaliser la prédiction en tête d'une liste de prédictions et enregistrer en tête de liste les prédictions qu'il impose à son tour. L'hypothèse de base est qu'en tout point de l'analyse, il doit être possible de déterminer la structure syntaxique du mot qu'on est en train d'étudier, sur la base des prédictions faites pendant l'analyse des mots qui le précèdent, et d'autre part, de prévoir les structures syntaxiques que l'on rencontrera à la droite de ce mot. L'analyseur fonctionne comme un automate à pile (push-down store) qui accepte une grammaire hors-contexte sous forme de GREIBACH<sup>1</sup>. L'analyse prédictive a été particulièrement étudiée par OETTINGER, GREIBACH et KUNO (Harvard).

*Les analyseurs syntaxiques ascendants* s'appuient sur l'algorithme de COCKE. La grammaire peut être mise sous la forme normale de CHOMSKY. Cette analyse permet de conserver toutes les parties communes des diverses analyses, ce qui n'est pas possible avec les méthodes descendantes. Si la phrase analysée est composée de  $n$  mots, l'arbre correspondant comportera  $n-1$  sommets non terminaux. L'analyse montante consiste à construire l'arborescence à partir des sommets terminaux. Cet analyseur a été utilisé par SAKAI et NAGAO à Kyoto, TAMACHI à Fukuoka, en Grande-Bretagne et à Grenoble (Au CETA, l'analyse syntaxique est réalisée au moyen d'un analyseur "Extended Context-free" qui est une variante de l'algorithme de COCKE et fournit des arborescences en constituants. L'analyse se poursuit par une phase d'interprétation qui aboutit à des structures de dépendances).

### Le transfert

Pour passer de l'analyse à la génération, de la langue source à la langue cible, il est nécessaire de passer par l'étape de transfert, caractérisée par l'emploi d'un *langage pivot*. Afin d'assurer une traduction multilingue, il convient de créer un pivot capable de conserver le sens au niveau le plus profond et de représenter des phrases ayant un sens identique mais éventuellement des structures morphologiques et syntaxiques différentes. Ce langage pivot est réalisé sous la forme de structures de dépendances accompagnées de variables linguistiques recherchant la valeur approximative de "cas profonds". Il fait apparaître une partition des unités lexicales en 2 classes.

- les éléments à valeur prédicative qui se comportent comme des prédicats à une, deux, trois places d'argument. L'unité lexicale de ce type, munie de ses arguments, forme un schéma d'énoncé. "Le schéma d'énoncé est la base de la structure syntaxique du langage pivot".
- les éléments à valeur non prédicative servent d'argument aux précédents. Ce sont souvent des mots descripteurs (noms propres, noms communs qui ne dérivent pas de verbes ou d'adjectifs).

### La génération

A partir du langage pivot, dans l'ordre inverse, on génère une structure syntaxique (de surface) puis une structure morphologique.

---

(1) L'analyseur prédictif de Harvard (KUNO et OETTINGER) est le plus important parmi les premiers analyseurs descendants, construits sur une grammaire non contextuelle dont les règles suivent la forme normale de GREIBACH : le premier symbole de chaque membre droit doit être un symbole terminal. La grammaire utilisée ici contenait 133 classes de mots et 2100 règles. Elle ne traitait pourtant pas l'accord du sujet et du verbe et acceptait des phrases non grammaticales !

Il est intéressant de rappeler que le système du CETA, développé de 1961 à 1967 a effectué les premières traductions russe-français avec des grammaires et des dictionnaires qui ont été perfectionnés jusqu'en 1971. L'expérience a porté sur plus de 400 000 mots traduits sur IBM 7044 à la vitesse de 4500 mots/heure. Une classification des phrases selon 2 types (phrases compréhensibles et non compréhensibles) donne, sur un échantillon important de ces textes, 61% de phrases compréhensibles, 39% d'incompréhensibles.

On notera que de façon paradoxale, ce sont les systèmes de 1<sup>ère</sup> génération, améliorés mais inférieurs aux systèmes de 2<sup>ème</sup> génération, qui ont été exploités.

### 3.1.1.2 De 1970 à nos jours

Il semble que le coup d'arrêt apporté par le rapport ALPAC ait largement contribué au développement gigantesque de l'Intelligence Artificielle. C'est entre 1970 et 1973 qu'apparaissent les premiers travaux, avec la thèse de T. WINOGRAD et sa publication<sup>1</sup> en 1972. L'enthousiasme renaît et suscite un grand nombre d'études, ( J. PITRAT<sup>2</sup> en France).

Le robot de WINOGRAD est dirigé par l'intermédiaire d'un ordinateur, en langue naturelle (anglais). Il évolue dans un monde constitué d'une table, d'une boîte et de cubes, boules, pyramides de diverses couleurs. Il manipule les objets décrits dont il connaît les coordonnées, selon les ordres exprimés en anglais. La grande originalité de ces travaux, outre que le robot répond en anglais (ce qui suppose une génération, même légère, de l'anglais), est de pouvoir, à partir des éléments introduits en mémoire, raisonner à leur sujet. Il fait preuve d'un pouvoir de déduction logique et d'inférence. Les robots de ce type (winogradiens) fonctionnent dans un univers limité, restreint. Ils possèdent leurs connaissances sous forme de dictionnaires et de relations. Le volume des informations que l'on fournit à la machine ne peut cependant pas s'accroître indéfiniment. On parle de *sémantique fermée*.

D'autres travaux sont effectués dans la même période, avec W. WOODS<sup>3</sup>, R. SIMMONS<sup>4</sup> et R. SCHANK qui cherchent à traduire un texte dans une représentation différente du langage naturel et ceci à l'aide d'une analyse conceptuelle. La sortie de tels analyseurs est une représentation des concepts et de leurs relations explicites ou implicites (1.5.3.1.2 et 1.6.7). Un autre mérite de R. SCHANK est de créer un "script", structure en trois parties, pour formaliser la connaissance des étapes d'un évènement : début d'action, acteurs et objets nécessaires à l'évènement, déroulement des opérations. Comme dans nos travaux de compréhension, il est à remarquer que le point délicat consiste à repérer la fin d'un phénomène ou d'une action.

Les projets de TA menés à terme ou en voie de réalisation (2.3.3, 2.3.4) ont essayé de tirer des leçons du passé en concentrant leurs efforts sur les problèmes linguistiques plus ou moins bien maîtrisés par les théories que nous venons d'évoquer. Après une absence complète de théorie linguistique, la syntaxe s'est imposée, puis la sémantique.

(1) T. WINOGRAD : *Understanding Natural Language*, Academic Press, Edinburgh, 1972

(2) J. PITRAT : "La programmation informatique du langage" in *La Recherche*, n°93, 1978

(3) W. WOODS : *Transition Networks grammars for Natural Language Analysis*, CACM (Communications of the Association for Computing Machinery), Vol. 13, n°10, 1970, pp. 437-445

(4) R. SIMMONS : "Semantic Networks : their Computation and Use for Understanding English Sentences", in *Computer Models of Thought and Language*, Schank & Colby, Freeman, San Francisco, 1973, pp. 63-113

Les améliorations ne sont pas probantes, en attendant une entrée en force de l'Intelligence Artificielle, dont se réclament les systèmes récents et que nous promet le projet japonais de 5<sup>ème</sup> génération.

### 3.1.2 Evolution des conceptions

Nous avons vu qu'un des grands progrès accomplis avec les systèmes de 2<sup>ème</sup> génération a été de *séparer les grammaires et les dictionnaires* (les données sur la langue) du logiciel d'analyse. Ce principe est le seul qui autorise une évolution ultérieure satisfaisante. Le système ARIANE 78 (2.3.3.8.1) en est une parfaite illustration.

Une autre conclusion s'est imposée. L'ordinateur ne peut accéder à une compréhension satisfaisante que pour *un domaine bien limité*. Les systèmes performants sont le plus souvent spécialisés. TAUM-METEO (2.3.3.3.7.1) est une réussite exemplaire. La dernière version, METEO2, tout en étant plus performante que METEO1, fonctionne sur un micro-ordinateur de taille moyenne et traduit correctement 90-95% des phrases.

Les ingénieurs de l'Institut du Textile de France ont limité non seulement la sémantique mais aussi la syntaxe. TITUS IV (2.3.3.3.10) accepte exclusivement les textes rédigés selon des règles de syntaxe préétablies et ne contenant que des termes appartenant à un vocabulaire prévu. L'opérateur doit se plier à une certaine discipline et connaître le domaine. La rigueur de *la syntaxe contrôlée* garantit la clarté des textes.

D'autres voies offrent une alternative avantageuse à la Traduction Automatique. Afin d'aboutir plus sûrement à un outil opérationnel, les recherches se sont souvent orientées vers des systèmes d'aide à la traduction, recueillant les fruits des énormes progrès de l'informatique en miniaturisation et en puissance. Les ambitions sont plus réalistes. Les systèmes sont limités, *la qualité très moyenne des résultats nécessite la révision d'un traducteur qui dispose d'un environnement adapté. et gagne ainsi un temps précieux*. On distingue la traduction par ordinateur assistée par l'homme (système interactif ALPS 2.3.3.4.1, système WEIDNER 2.3.3.2.11) et la traduction humaine assistée par ordinateur (outils d'aide tels que MERCURY 2.3.3.4.2, TERMEX 2.3.3.4.3, TII 2.3.3.4.6...).

Nous pensons que ces systèmes présentent des défauts, parfois décourageants lorsqu'ils ne sont pas réhivitoires. La convivialité n'est pas toujours le point fort de ces outils qui exigent de l'utilisateur une longue période d'adaptation. Les fabricants livrent des dictionnaires spécialisés en option et n'autorisent pas le client à compléter les lexiques. Ils proposent des réactualisations périodiques nécessairement coûteuses. Il est tout de même parfois possible de créer son propre dictionnaire. La structure des programmes qui l'utiliseront limite les rajouts aux noms communs. Les niveaux de prix sont bien entendu fonction des performances et du matériel recommandé. Certains de ces produits tournent sur un micro-ordinateur haut de gamme (AT 386, 8 Mo de mémoire centrale, disque dur de 80 Mo, 30 000 à 40 000 F), d'autres fonctionnent sur des minis ou des calculateurs spécialisés (machines dédiées). Le logiciel est vendu ou implanté sur site avec une licence d'exploitation.

De son côté, la TA s'adapte au marché. Elle doit vaincre pour cela le handicap de sa démesure : les programmes et les données exigent une maintenance et un développement constant. Le matériel informatique nécessaire n'est pas à la portée de tous les porte-monnaie. Aussi a-t-elle recours aux techniques de partage des ressources. La télématique lui permet de communiquer, d'un site central (centre serveur) vers des périphériques (minitel, micro-ordinateurs). Elle devient prestataire de services (SYSTRAN/Gachot 2.3.3.2.9).

### 3.2 Une autre approche de la compréhension avec l'ACI

Les modèles linguistiques sont encore insuffisants pour que la Traduction Automatique parvienne à des résultats vraiment satisfaisants. Les systèmes qui proposent une traduction entièrement automatique exigent en fait une prédiction ou une sélection préalable du document à traduire. On ne peut de toute façon jamais faire l'économie d'une révision qui s'impose quel que soit le type de système utilisé.

Si l'on considère l'aspect économique du domaine et nous pensons qu'il est essentiel, les systèmes plus modestes ont un intérêt certain. Les investissements ne sont pas démesurés, qu'il s'agisse de leur développement ou de leur exploitation. Ils ne sont efficaces que pour des domaines délimités mais l'acquisition de modules proposés en option étend quelque peu leur "couverture".

Ces limites posées, l'état des recherches en linguistique automatique et par voie de conséquence, l'insuffisance des modèles de la langue sont à l'origine d'une démarche beaucoup plus pragmatique, centrée sur la compréhension. Lorsque nous sommes arrivé au Centre de Recherche Jean Favard en 1976, D. HERAULT et P. POGNAN avaient jeté les bases du système que nous allons décrire, et développé un analyseur prédictif du tchèque, testé sur d'autres langues slaves. Les travaux de J.P. HUBAC sur le français, de M. FANTON sur le groupe prépositionnel allemand, de H. FERVERS sur le dictionnaire de bases allemand-français et de R. SAUSSE sur le japonais ont permis de vérifier certaines hypothèses sur les principes que nous avons alors retenus. Après avoir résolu le problème de la localisation automatique des mots composés (1979) et préparé le module de désimbrication des mots d'origine germanique et d'origine romane, nous avons développé l'analyseur de l'Automate de Compréhension Implicite.

#### 3.2.1 Définition

Nous pensons que la compréhension est un élément fondamental des systèmes de Traitement Automatique des Langues Naturelles. Il suffit de se rappeler le conseil mille fois répété de notre professeur de latin : "*Lisez plusieurs fois votre texte, avant de traduire !*" pour découvrir l'évidence ; lire veut dire ici *comprendre en gros* le texte. Pourquoi en irait-il autrement en T.A. ?

L'examen des livres et articles étrangers montre qu'ils ne sont pas toujours accompagnés d'un résumé dans une langue comme l'anglais, et si résumé il y a, que l'idée n'est peut-être pas toujours fidèle. Le cycle traduction-impression-édition est de plus très long, de sorte que la traduction n'a plus d'intérêt parce qu'elle date, sans compter la traduction qui ne correspondait pas aux besoins.

Partant de ces quelques constatations, l'idée a été de construire un système de "lecture rapide", de saisie du contenu, *un système capable d'extraire automatiquement les éléments nécessaires à la compréhension d'un texte technique ou scientifique écrit dans une langue étrangère, pour un lecteur spécialiste du domaine mais ne maîtrisant pas cette langue.*

Un certain nombre de principes se sont alors imposés, au fil des recherches. Nous allons les rappeler.

#### 3.2.2 Principes

##### 3.2.2.1 Compréhension explicite, compréhension implicite

Nous distinguons deux types de compréhension, la compréhension explicite et la compréhension implicite.

- la **compréhension explicite** extrait du texte les informations nécessaires pour aboutir à des réalisations concrètes, tangibles (robots, systèmes experts...). Il s'agit là d'Intelligence Artificielle où l'on relève un noyau de compréhension et où l'exécution de tâches pratiques et visibles en apporte la preuve.

- la **compréhension implicite** extrait d'un texte suffisamment d'informations pour qu'un familier du domaine puisse en déduire le contenu et son organisation. Dans ce cas de compréhension implicite, c'est l'intelligence seule de l'utilisateur qui permet de savoir si le système a, ou non, fonctionné correctement.

Cette distinction implicite/explicite, qui ne devrait pas provenir de l'analyse initiale, prend un sens par rapport à la nature de l'univers sémantique dans lequel on s'est placé. **Dans un univers sémantique ouvert**, il y a toujours un résidu d'implicite (pré-supposition, sous-entendu...) mais, pour prendre une comparaison acoustique, le bruit de fond n'empêche pas de traiter le signal. La compréhension est explicite, sur fond d'implicite, sans que l'on puisse fixer de frontières entre les deux couches. On peut en effet plus ou moins expliciter l'implicite. **Un univers sémantique fermé** est, quant à lui, la condition d'une compréhension explicite sans "trace d'implicite".

La compréhension peut cependant devenir implicite si l'on "réduit" l'explicite comme le fait notre ACI. Le caractère implicite de sa compréhension ne tient pas seulement à l'univers sémantique dans lequel se situe le message. Il tient aussi à la destructuration relative de la langue, liée à l'exploitation de l'hypothèse du parallélisme dérivationnel des langues que nous présentons de 3.2.2.3 à 3.2.2.4.

Les systèmes de TA opèrent généralement en sémantique largement ouverte (il peut y avoir certaines limitations de domaine), mais avec des méthodes relevant de la sémantique fermée, puisque les dictionnaires, tels qu'ils sont conçus et implantés, "ferment" obligatoirement la sémantique dans laquelle le système est placé. Il en va de même pour les systèmes de Documentation Automatique qui, avec des réseaux de mots- ou d'expressions-clés, s'inscrivent dans la sémantique la plus fermée que l'on puisse imaginer. Il y a là une profonde contradiction. Partir de la compréhension permet de limiter considérablement les ambiguïtés et surtout, permet de considérer un texte dans sa totalité, chaque phrase apportant éventuellement une information. Dans la plupart des systèmes, aucune liaison sémantique n'est établie de phrase en phrase. Ces systèmes ne créent pas une mémoire de référence dans le déroulement du texte. Chaque phrase est analysée isolément. Ces arguments soulignent la nécessité d'une première étape privilégiant la compréhension pour aboutir à l'extraction d'information, quel que soit le but recherché.

### 3.2.2.2 Interprétation automatique

Le but de la compréhension implicite est de passer en *interprétation*, de la langue source LS à la langue cible LC.

Il convient de définir avant tout ce que nous entendons par "**interprétation automatique**" et en quoi ce concept diffère de "traduction automatique". Un système d'interprétation ne cherche pas à traduire, il essaie d'extraire de tout texte (non-littéraire) suffisamment d'informations pour qu'un utilisateur du système, connaissant bien le domaine dont parle le texte, en *compre* convenablement le contenu. La différence avec la notion de traduction repose sur la nécessité de compréhension et les compétences de l'utilisateur. Cette hypothèse ne nous semble pas trop restrictive dans le sens où un spécialiste ne s'intéressera pas souvent à un domaine très éloigné du sien. Ceci a une conséquence immédiate : les problèmes d'ambiguïté peuvent être considérablement réduits, cela permet de proposer à l'utilisateur un texte interprété compréhensible, écrit

dans sa langue, mais d'une mauvaise qualité linguistique (des problèmes délicats peuvent en effet être traités de façon plus superficielle : traduction des prépositions...). De nombreux tests ont montré qu'à partir des informations données, l'utilisateur reconstruit le texte initial. Nous sommes loin de la traduction qui n'exige pas la participation du lecteur et dans cette optique, nous pouvons même dire que la T.A.O. est en quelque sorte à mi-chemin entre la T.A. et l'interprétation.

Voici quelques exemples d'interprétations obtenues par simulation sur notre texte d'essai :

*Man kann diese Spannung messen, wenn man die Kohlestäbe außerhalb des Gefäßes durch ein Voltmeter verbindet.*

00023000100000	*05	MAN	On
0V002000200000	4300	KANN	peut
*V002000500000	2130	MESS	mesurer
0002300030N003	V00	DIESE	cette
000230004*N003	\$00	SPANN.UNG	tension
00023000600000	,01	,	,
0002300070C007	N00	WENN	quand/si
00023000800000	*05	MAN	on
.V0170017*C007	2300	VERBIND	relie
0002300090N009	IVK	DIE	les
000230010*N009	\$00	STA*B.E KOHLE	tige carbone
0002300110P011	F00	AUS*ERHALB	en dehors de/du
00023001200000	I00	DES	le
00023001200000	I00	GE.FA*S*.ES	réceptient
0002300140P014	F00	DURCH	par
00023001500000	J00	EIN	un
000230016*F014	\$00	VOLT METER	voltmètre
00023001800000	.	.	.

*Tatsächlich erhält man infolge von Energieverlusten in der Zelle jedoch nur zwischen 0,6 und 0,85 Volt.*

00019000100000	AZ0	TATSA*CHLICH	En fait
00019000300000	*05	MAN	on
.V002000200000	2300	ERHALT	obtient
0001900040P004	F00	INFOLGE VON	à la suite de
000190005*P004	\$00	VER.LUST.EN/EN.ERG.IE	perte d'énergie
0001900060P006	F00	IN	dans
00019000700000	I00	DER	la
000190008*P006	\$00	\$ZELLE	cellule
00019000900000	A00	JEDOCH	pourtant
00019001000000	A00	NUR	seulement
0001900110P011	F00	ZWISCHEN	entre
00019001200000	C00	0,6	0.6
00019001300000	MO+	UND	et
00019001400000	C00	0,85	0,85
000190015*P011	\$00	\$VOLT	volt
00019001600000	.	.	.

*An der Anode bewirkt ein Katalysator die Aufspaltung der Wasserstoff-Moleküle in Wasserstoff-Ionen (H+) und Elektronen (e).*

0001100010F001	F00 AN	A
00012000200000	I00 DER	l'
000130003*P001	\$00 AN.OD.E	anode
0001100050N005	JS0 EIN	un
000110006*N005	\$00 KATA.LYS.ATOR	catalysateur
.V004000400000	1380 BEWIRK	effectue
0001100070N007	IVK DIE	la
00011000800000	\$00 AUF.SPALT.UNG	séparation
00011000900000	IVK DER	des
000110010*N007	\$00 MOLEK.U*L.E WASSERSTOFF	molécule hydrogène
0001100110F011	F00 IN	dans/en
00011001200000	\$00 ION.EN WASSERSTOFF	ion hydrogène
000110013+0000	(	(
000110014#0000	\$00 H+	H+
000110015&0000	)	)
00011001600000	M0+ UND	et
00011001700000	\$00 ELEKTRON.EN	électron
000110018+0000	(	(
000110019#0000	E	e
000110020*P011	)	)
00011002100000	.	.

*Die Größe eines aus ihnen konstruierten Kraftwerks kann sich nach dem Elektrizitätsbedarf des Versorgungsgebietes richten und mit diesem wachsen.*

0005600010N001	IVK DIE	La
00056000200000	\$00 GRD*S*.E	taille
00056000300000	JS0 EINES	d'une
000560007*N001	\$00 KRAFT.WERK.S	centrale
00056000600000	5Z00 KONSTRUIER.	construit
0005600040P004	F00 AUS	avec
000560005*P004	Y00 IHNEN	eux/elles
0V008000800000	4300 KD*NN	peut
0V008000900000	W00 SICH	se
0V008001500000	1130 RICHT	régler
0005600100P010	F00 NACH	vers/selon
00056001100000	I00 DEM	le
00056001200000	\$00 BE.DARF ELEKTR.IZ.ITA*T.S	besoin d'électricité
00056001300000	I00 DES	du
000560014*P010	\$00 GE.BIET.ES VER.SORG.UNG.S	domaine d'alimentation
00056001600000	M01 UND	et
.V008001900000	2130 WACHS	croître
0005600170P017	F00 MIT	avec
000560018*P017	V00 DIESEM	celui-ci
00056002000000	.	.

*Eine dünne Schicht aus geschmolzenem Carbonat und inertem Füllmaterial liegt hier zwischen zwei Elektroden aus porösem Nickel.*

0007200010N001	JS0	EINE	Une
000720003*N001	\$00	SCHICHT	couche
00072000200000	Z00	DU*NN.E	épais
0007200040P004	F00	AUS	en/de
00072000600000	\$00	CARBON.AT	carbonate
00072000500000	2Z00	GE.SCHMOLZ.EN.EM	fondu
00072000700000	M0+	UND	et
000720009*P004	\$00	MATERIAL FU*LL	matériau de remolissage
00072000800000	Z00	INERT.EM	inerte
.V010001000000	2300	LIEG	se trouve
00072001100000	A00	HIER	ici
0007200120P012	F00	ZWISCHEN	entre
00072001300000	C01	ZWEI	deux
000720014*P012	\$00	ELEKTRO.DE.N	électrode
0007200150P015	F00	AUS	en/de
000720017*P015	\$00	NICKEL	nickel
00072001600000	Z00	PORO*S.EM	poroux
00072001800000	.	.	.

*Hier liegt momentan das entscheidende Problem, denn die ersten Zellen werden verhältnismäßig teuer sein, und der Preis wird nur in dem Maß sinken, in dem die Produktion wächst.*

00085000100000	A00	HIER	Ici
00085000300000	AZ0	MOMENT.AN	momentanément
0008500040N004	IVK	DAS	le
000850006*N004	\$00	PRO.BLEM	problème
00085000500000	2Z00	ENT.SCHEID	décisif
.V002000200000	2300	LIEG	se trouve
00085000700000	,01	,	,
00085000800000	M01	DENN	car
0008500090N009	I0K	DIE	les
00085001000000	C02	ERST.EN	premier
000850011*N009	\$00	ZELL.EN	cellule
0V012001200000	3130	WERD	
*V012001500000	3100	SEIN	seront
00085001300000	AZ0	VER.HA*LT.NIS.MA*S*.IG	en proportion
00085001400000	AZ0	TEUER	cher
00085001600000	,01	,	,
00085001700000	M01	UND	et
0008500180N018	I0K	DER	le
000850019*N018	\$00	FREIS	prix
0V020002000000	3300	WIRD	
*V020002500000	2310	SINK	baissera
00085002100000	A00	NUR	seulement
0008500220P022	F00	IN	dans/en
00085002300000	I00	DEM	la
000850024*P022	\$00	MAS*	mesure
00085002600000	,01	,	,
0008500270R027	K01	IN	dans/en
0008500280R028	K00	DEM	lequel/laquelle
0008500290N029	IVK	DIE	la
000850030*N029	\$00	PRO.DUKT.I0N	production
.V0310031*R028	2300	WACHS	croit

Le système se compose de deux parties principales :

- L'analyse de la langue source
- La reconstruction de la langue cible

Ces deux parties ont un même fondement linguistique : le système dérivationnel d'une langue, d'une sous-famille ou d'une famille de langue. Nous allons examiner rapidement cet aspect, qui sera repris par la suite au cours de la présentation des différents principes.

### 3.2.2.3 Le système dérivationnel

Pour les nombreuses langues indo-européennes qui ont été examinées, la structure des mots est pratiquement identique. Tout est organisé autour d'une racine à laquelle on associe des préverbes, affixes, infixes et morphèmes grammaticaux. D. HERAULT cite LOMONOSOV, qui, à la demande de Catherine II de Russie, a utilisé cette notion pour créer à partir du français les mots qui manquaient à la langue russe de l'époque. Beaucoup de mots russes actuels sont par conséquent l'image exacte de mots français, sans qu'il s'agisse d'emprunts.

ex. :            la racine PIS                            -> *écrire*  
                   le préverbe PERE                   -> *une nouvelle fois*  
                   le suffixe AT' est un des suffixes verbaux usuels  
                   PERE.PIS.AT'                        -> *réécrire.*

Il n'y a ici aucune relation étymologique entre les éléments français et les éléments russes.

Les travaux effectués dans cette optique sur les langues européennes se sont raréfiés depuis 1950. Pourtant, si l'on se place au niveau des systèmes dérivationnels qui correspondent aux langues germaniques, romanes, slaves... on constate des propriétés importantes qui peuvent être exploitées par des systèmes informatiques :

- Les systèmes dérivationnels des langues évoquées plus haut ont à peu près les mêmes caractéristiques quantitatives.
- Au niveau d'un texte, tous les éléments du système apparaissent avec une régularité prévisible.
- Le système dérivationnel impose une classification du lexique, très différente de celle que l'on trouve dans les dictionnaires ordinaires.

Nous allons expliciter ceci dans les deux paragraphes ci-dessous.

#### 3.2.2.3.1 Les propriétés quantitatives des systèmes dérivationnels

Si l'on se limite aux racines, préverbes, préfixes et suffixes, les résultats obtenus par D. HERAULT sont les suivants :

- Quelle que soit la langue, le nombre des racines importantes, celles qui donnent lieu à une dérivation non négligeable, se situe autour de 800, non compris les variantes (2 par racine en moyenne). Ce ne sont jamais des racines d'emprunt. On est donc amené à manipuler 1600 racines environ, qui correspondent à 75% des entrées d'un dictionnaire usuel (50 000 entrées). Le quart restant correspond à des mots isolés ou des dérivés de racines tombées en désuétude. Ces mots ne sont jamais des verbes et appartiennent le plus souvent au patrimoine artisanal, rural ou technique.

- Sur le plan sémantique, l'ensemble des racines principales couvrent à peu près le même univers pour chaque langue, avec des correspondances étymologiques fortes à l'intérieur des diverses familles, et faibles entre familles. Pour des raisons que nous rappelons dans 3.2.2.4., on ne peut pas parler d'un mais de deux systèmes dérivationnels pour l'anglais : le système germanique figé qui n'a plus d'activité dérivationnelle et le système roman, très voisin du système français.

- D. HERAULT distingue, en ce qui concerne la partie gauche de la racine, les préverbes et les préfixes. Les préverbes appartiennent à un ensemble fermé et s'associent fondamentalement aux racines de la langue. Leur action sémantique sur la racine est difficile à définir. Les préfixes relèvent d'une sémantique ouverte et modifient chaque racine de la même façon. Leur origine est souvent extérieure à la langue (grecque, latine). Ainsi, pour toutes les langues citées, les préfixes sont à peu près identiques. Les préverbes se distinguent par le phénomène d'assimilation, très faible dans les langues slaves et germaniques, forte dans les langues romanes. Il n'y a presque jamais plus de 3 préverbes devant la racine, l'assimilation est pratiquement nulle à leur niveau. En cas de non-assimilation, il y a entre 80 et 150 groupes préverbaux, dans le cas contraire, il y en a environ 250<sup>1</sup>.

Il est extrêmement difficile d'attribuer un sens précis à un préverbe ou un groupe de préverbes, qui fonctionnent de façon variable sur des racines distinctes. Par contre, ils ont en commun de permettre pour chaque langue, de construire des bases de sens identique (la base est une association d'un ou de plusieurs préverbes à une racine, et le sens de cette association apparaît clairement). Le fondement du système des bases est composé d'environ 2500 éléments. A partir de ce niveau, tous les systèmes de bases sont sémantiquement équivalents, tout en présentant parfois des structures largement différentes.

- Les suffixes apparaissent dans une grande variété d'éléments (250), pour chaque langue. Ils jouent un rôle sémantique déterminé. Les suffixes peuvent être classés par rapport aux catégories syntaxiques (verbal, nominal, adjectival, adverbial...). On peut alors établir une assez bonne correspondance de langue à langue. On constate que certains suffixes fonctionnent de façon "automatique". On peut les associer à une base quelconque pour obtenir un mot attesté, un mot qui n'existe pas encore ou un mot inconnu mais compréhensible<sup>2</sup>. La maîtrise du système des racines et des bases (bases purement verbales, purement nominales, irrégulières) conduit à l'exemple :

Racine : TRAC

Bases : TRAC, SOUSTRAC, ABSTRAC et EXTRAC

TRAC est une base régulière et verbale, on crée automatiquement TRACER, TRACEUR, TRACABLE, TRACTION, TRACTEUR.

SOUSTRAC, ABSTRAC et EXTRAC sont des bases nominales et représentent les bases verbales (SOUSTRAI, ABSTRAI et EXTRAI). On peut créer :

- SOUSTRACTION, SOUSTRACTEUR, SOUSTRACTABLE
- ABSTRACTION, ABSTRACTEUR, ABSTRACTABLE

---

(1) Pour le russe et le bulgare, le nombre des préverbes de base est faible, moins d'une vingtaine. Le nombre des groupes préverbaux est donc d'au plus 80. Le français et l'italien sont à l'opposé et ont un nombre important de groupes lié à une forte assimilation (AB.SIMIL -> AS.SIMIL).

(2) Nous pensons, en français à *-able/-ible* au niveau adjectival, *-er* au niveau verbal, *-eur/-teur* et *-ion/-tion/-ation/-ition* au niveau nominal.

### - EXTRACTION, EXTRACTEUR, EXTRACTABLE

On aboutit à des mots inexistant mais compréhensibles.

#### 3.2.2.3.2 La classification du lexique

Les mots sont ordinairement classés par ordre alphabétique, avec une marque syntaxique (verbe, nom, adjectif...). Cette classification n'est pas satisfaisante si l'on part du principe que tout tourne autour des propriétés dérivationnelles de la racine ou de la base. Le système repose sur une autre classification et distingue:

- Pour chaque langue, une classe importante de "*mots-outils*" (prépositions, conjonctions, certains adverbes, déterminants, ...). Cette classe contient environ 500 éléments, dont certains sont ambigus.

- Les "*mots-objets*", en dehors ou presque du système dérivationnel. Ils représentent réellement des objets (*arbre, feuille...*), peuvent provenir d'êtres vivants (*aile, écaille...*) ou sont construits par l'homme (*chaise, table...*). Pour passer d'une langue à l'autre, il faudra, dans ce cas, adopter la méthode lexicale classique. Cela correspond à environ 3000 entrées. Beaucoup de ces mots ont un caractère international (médecine, biologie, physique...).

- Des "*mots-exceptionnels*", normalement dérivés, mais qui sont devenus des "*mots-objets*" pour des raisons extra-linguistiques (réaction, éprouvette, ensemble, fonction...). Ces mots ne seront exceptionnels que par rapport à un domaine donné.

- Les "*mots-foncteurs*" (genre, classe, espèce, famille, type, modèle...) indiquent souvent un pluriel indéfini, dans une situation déterminée. Ils ne sont pas plus de 30 mais il est intéressant de les connaître dans le sens où leur rôle sémantique est pratiquement nul.

- Les "*mots-dérivationnels*". Pour les analyser, il faut les décomposer, isoler la racine, les préverbes, préfixes, suffixes, morphèmes grammaticaux et reconstruire la base. Il faudra ensuite connaître les caractéristiques de la base.

Avant de voir comment cette classification sera utilisée, nous évoquerons le problème essentiel qu'elle pose et que nous devons résoudre : de nombreuses langues contiennent énormément de mots d'emprunt. Il faut donc construire un système de sélection, qui devra associer chaque mot dérivationnel à son système, que ce soit celui de la langue ou un système d'emprunt.

#### 3.2.2.3.3 La sélection entre plusieurs systèmes dérivationnels

Pour quelles langues la sélection s'impose-t-elle ? Dans les langues germaniques, la sélection germanique/roman est essentielle. Dans quelques langues slaves, le russe et le bulgare par exemple, il en va de même. Pour le tchèque, une sélection slave/roman/germanique s'impose. Dans les langues romanes, aucune sélection n'est indispensable (ce que nous appelons système dérivationnel roman contenant une partie grecque importante). En guise d'exemple et pour en souligner l'inutilité, citons "*barycentre*" (grec *bary* : lourd, latin *centre* d'origine grecque) ou "*anormal*" (*a-* grec privatif et un dérivé de *norm* purement latin : angle, équerre). Notons qu'il faudra distinguer ce *a-* privatif du *a-* de *abaisser*, distinction que permettront certaines caractéristiques de la racine ABAISS-.

On peut se demander s'il est possible, pour l'allemand, de "désenchevêtrer" les deux systèmes, d'une façon automatique, c'est à dire, sans intervention humaine. Nous verrons que c'est possible à 100% dans 4.3.2.5.3 et 4.3.2.5.4. On peut dès maintenant ajouter quelques commentaires. On constate que deux systèmes dérivationnels distincts présentent très peu de racines homographes (russe PRAV : *droit* et français PRAV : *mauvais* dans *dépravé*, allemand STAB : *bâton* et français STAB : *stable*).

La sélection de ces racines et des autres est fondée sur le fait que les systèmes préverbaux sont presque toujours complètement disjoints (IN et AB sont communs pour le français et l'allemand), et qu'il en est de même pour le système suffixal.

Ces constatations ne conduisent pas à une conclusion, il faudrait un travail colossal pour y parvenir, mais à l'hypothèse selon laquelle les systèmes dérivationnels d'une même famille de langues (ici indo-européennes) sont "conçus" de telle sorte qu'ils peuvent cohabiter les uns avec les autres, sans que soient créées de nombreuses ambiguïtés.

#### 3.2.2.4 La compréhension et sa localisation

Avant de parler du transfert, nous allons préciser à quels endroits se trouvent les éléments indispensables à la compréhension du texte.

De façon très globale, un texte est en fait la superposition de deux structures : les "mots-objets" et "mots-exceptionnels" d'une part, les "mots dérivationnels" (plutôt associés à des racines verbales) d'autre part. La première partie permet de savoir de quoi on parle, la seconde partie comment on en parle.

La sélection joue un rôle important dans la localisation des "mots-objets". En allemand par exemple, la mise en évidence des mots romans simplifie le travail de compréhension. Ces mots sont souvent internationaux et n'ont besoin d'aucune interprétation. Ces mots restent nominaux et ont par conséquent peu ou pas d'influence verbale. Cette sélection est insuffisante pour savoir ce dont on parle. Il est indispensable de disposer d'un analyseur syntaxique pour mettre en évidence les syntagmes nominaux, prépositionnels ou non. Il sera décrit pour l'allemand en 4.3.3.

Ces syntagmes nominaux sont reliés les uns aux autres, par des syntagmes verbaux de deux types :

- les syntagmes verbaux "hypersyntaxiques" comme *nous allons voir, nous allons examiner...*
- les syntagmes verbaux d'action qui agissent sur les objets du texte. Ces syntagmes verbaux peuvent prendre un aspect nominal, éventuellement, comme dans un titre Dans l'exemple "*Méthodes techniques de construction des barrages hydro-électriques*", le syntagme verbal est représenté par *CONSTRUCTION* et agit sur "*...barrages...*", tandis que "*Méthodes (techniques)*" est un simple mot foncteur.

#### 3.2.2.5 Le transfert "global"

Passer de la LS à la LC, c'est transférer les informations recueillies dans LS vers son image simplifiée dans LC. Habituellement, le transfert s'effectue par petites unités et s'oppose à une analyse globale, complète et profonde du texte en LS.

Le principe de globalité du transfert signifie que pour passer de LS à LC, il s'effectue en une seule fois, dès que l'analyse du texte en LS est terminée (l'analyse est conduite aussi loin qu'il est possible et raisonnable d'aller).

On retrouve dans le schéma du système, les trois étapes suivantes :

- Analyse du texte LS privilégiant la compréhension
- Transfert global de LS vers LC
- Génération du texte LC

Ces trois étapes appellent trois questions :

1. Comment privilégier la compréhension ? : L'analyse se résume couramment à l'analyse morphologique et syntaxique de chaque mot avec une segmentation de la phrase. Quel que soit son degré de perfectionnement, cette analyse ne donne pas une idée sur le contenu du texte et n'a qu'une relation lointaine avec la compréhension. Pour comprendre, il semble préférable d'agir en sens inverse, en n'accordant à la morphologie et à la syntaxe qu'un rôle secondaire. Ce ne sont pas les mots qui sont des unités de raisonnement, mais l'ensemble des éléments qui constituent le système dérivationnel de LS (racines, préverbes, affixes). Cela correspond au système de formation des mots de LS. Le noyau de ce système est constitué par une partie qui regroupe les "racines verbales", celles qui donnent l'idée d'une action, d'un processus.

TABL, CHAIS, POUTR, POIT : Ces mots sont inertes

CÉDer, sucCÉDer, déCÉDer, proCÉDer... :  
CESSion, sucCESSion, proCESSion, sucCÈS, déCÈS, proCÈS

Pour les mots suivants, le sens apparaît dès lors que l'on connaît la racine et le 1er préverbe. Ce groupe est une véritable unité sémantique. Nous l'appellerons base.

CÉD/CESS, SUCCÉD/SUCCESS,/SUCCÈS, CONCÉD/CONCESS sont trois bases avec les variantes.

insUCCÈS ou rétroCESSion ont SUCCÈS et CESS pour base.

Cette partie, remarquablement organisée dans toute langue, jouera le rôle central dans notre analyse et privilégiera de ce fait la compréhension. Ainsi le système ne fera pas de distinction entre *concéder* conjugué et *concession*, quelle que soit pratiquement la situation syntaxique.

*Il faut que soit concédé...*

*Il faut que l'on obtienne la concession...*

*Il est nécessaire que l'on obtienne la concession...*

*Il faut que soit obtenue la concession...*

2. Que va-t-on transférer ? : Ce sont toutes les bases qui seront rencontrées, avec ce qui les entoure (préfixes, suffixes). On transmet également la liste des "mots inertes" sur laquelle sera effectué un travail préliminaire.

3. La génération du texte LC ? : Il faut distinguer deux étapes distinctes :

- Il faudra d'abord assurer une correspondance convenable entre les systèmes dérivationnels de sorte qu'à une base de la langue source supportant un certain type de dérivation soit associée une base de la langue cible, ayant à peu près le même niveau d'ambiguïté que la base de langue source et dont le type de dérivation sera connu, en vertu des règles de correspondance des deux systèmes.

Allemand	Français
LEIT	CONDUIR/CONDUCT
LEITen	CONDUIRE
LEITer (der)	CONDUCTeur

LEITer (die)	l'origine est différente (lehnen). Mais la racine commune a ici la forme <i>klei</i> apparentée à κλινειν (Klinik, Klimat...) et au latin <i>clinare</i> (deklinieren, Klient...), (incliner...).
LEITung	*CONDUCTION une contrainte devrait donner CONDUITE REDUIR/REDUCT sans contrainte donnera REDUCTION

Ce simple exemple montre que la correspondance des systèmes dérivationnels est une partie complexe du transfert.

- Les phrases en langue cible doivent être construites selon le schéma de l'analyse du texte source. Cette construction touche à la notion de correspondance de surface que nous évoquerons en 3.2.2.7. Avant de parler de la correspondance de surface et de l'analyse superficielle, il convient de préciser ici notre approche des dictionnaires.

### 3.2.2.6 L'absence de dictionnaire

Qu'il s'agisse de l'analyse du texte source, de la génération du texte cible ou du transfert global, le système opère sans dictionnaire, au sens traditionnel du terme. Nous allons prendre l'exemple de l'automate allemand-français pour indiquer les informations lexicales dont il a besoin.

- Maîtrise complète des systèmes dérivationnels de l'allemand, du français, ainsi que du roman germanisé (système roman très proche du français intégrant les modifications orthographiques consécutives à son implantation en allemand). Les trois systèmes sont voisins d'un point de vue quantitatif (1500 racines avec 2500 variantes, 150 préverbes et 250 suffixes approximativement). Il est difficile de dire à combien d'entrées du dictionnaire cela correspond (à 15 000-20 000 entrées), en langue source, sans compter les entrées qui existent potentiellement, mais ne sont pas usuelles (ex. : *PARable* opposé à *inPARable*) et seront engendrées par le système en langue cible.

Les systèmes dérivationnels ne permettent pas de dominer la totalité des deux langues.

- Il faut ajouter une liste des mots outils (articles, prépositions, conjonction...). On compte environ 500 éléments.

- Les mots restants sont les mots "inertes". Leur quantité varie selon le type de texte. Ils appartiennent à un ensemble très ouvert. Dans le cas précis de notre exemple, un sélecteur roman/germanique doit indiquer si un mot inerte donné est d'origine romane ou non. En effet, s'il est d'origine romane, le transfert dans la langue cible se fait sans modification. Ces mots sont très nombreux dans le discours scientifique. Il reste un noyau que l'on ne peut pas atteindre, de petite taille pour chaque texte. C'est à ce noyau que correspondra le seul vrai dictionnaire du système. Contenant également les mots inertes romans qui n'auront pas été reconnus, il constitue la limite du principe de l'absence de dictionnaire. C'est la qualité du sélecteur qui détermine l'importance réelle et théorique de ce noyau, l'existence du sélecteur dépendant des groupes de langues traités.

### 3.2.2.7 Analyse de surface

L'analyse complète du texte source est limitée à l'analyse de surface. Elle reproduit informatiquement l'analyse traditionnelle enseignée dans les établissements de l'enseignement secondaire. Pour les langues que nous traitons, il s'agit de :

- repérer le verbe avec précision

- reconstituer le groupe verbal
- trouver les compléments prépositionnels obligatoires (*se méfier de, se souvenir de...*)
- localiser les autres compléments prépositionnels
- déduire les syntagmes nominaux sujet et complément direct
- segmenter la phrase

Il nous faut introduire deux nouvelles notions, le **module hypersémantique** qui facilitera grandement la détermination des syntagmes nominaux sujet, complément direct et complément prépositionnels et le **module hypersyntaxique** qui sera d'une aide précieuse dans la segmentation de la phrase.

Nous obtenons ainsi une description simple de la phrase, facilement transférable vers une autre langue, sans aucun formalisme. C'est là un point fondamental qui tranche avec ce qui a été l'objet du chapitre I et rappelle les termes de sa conclusion.

### 3.2.2.8 Le principe des groupes

Il nous semble impossible d'imaginer que l'on pourra construire des systèmes universels où les langues n'apparaîtront que comme données. Il est peu raisonnable, a contrario, de construire des systèmes capables de ne travailler que sur un couple de langues déterminé.

Nous empruntons une voie moyenne et raisonnons sur des groupes de langues, en distinguant :

- le groupe germanique (allemand, néerlandais et sous-groupe scandinave)
- le groupe anglais (anglais)
- le groupe slave (sous-groupe bulgare, autres langues)
- le japonais
- l'indonésien
- l'arabe

Il est toujours possible de raisonner en terme de groupes, encore faut-il en monter l'intérêt pratique.

Pourquoi le groupe de langues, dans son organisation traditionnelle, constitue-t-il la meilleure unité de raisonnement en Traitement Automatique des Langues Naturelles ?

Les nombreuses langues qui ont été examinées, disposent d'un système dérivationnel autonome et d'un système d'adoption des mots étrangers. Pour des raisons historiques, D. HERAULT rappelle qu'il n'en est rien pour l'anglais. Le conflit qui a opposé le système germanique (saxon) au système roman (normand) a entraîné leur défaite. Le système germanique s'est figé dans la situation du XV<sup>e</sup> siècle environ et n'a procédé depuis lors à aucune création de mots. Le système roman, n'a pas évolué énormément et s'est contenté de recueillir des mots français et italiens, à partir de la Renaissance. La partie romane de la langue anglaise, en d'autres termes, suit strictement l'évolution des systèmes dérivationnels des langues romanes, sans procéder à des créations. La langue anglaise n'a donc pas le même comportement que les autres langues indo-européennes occidentales. Il est très difficile de raisonner sémantiquement avec elle. L'anglais, apparemment simple, est en fait très complexe.

### 3.2.2.9 Hyperanalyse, hypersémantique et hypersyntaxe

Dans le paragraphe 3.2.2.7, nous avons évoqué le recours aux module hypersémantique et hypersyntaxique pour la détermination des syntagmes nominaux essentiels à la compréhension et le repérage des schémas d'enchaînement de phrases du type raisonnement, définition...

**Pourquoi hyperanalyse ?** Parce que ce ne sont ni le mot, ni la phrase qui font office d'unité principale. Ce sont de grandes portions de texte, et à la limite, le texte entier. L'hyperanalyse consiste donc à tenir compte de tout ce qui se passe dans le texte et s'efforce de ne jamais traiter une phrase indépendamment de celles qui l'ont précédée.

**Comment est-ce possible ?** On ne peut pas utiliser ici le système dérivationnel, qui, par sa nature prédicative, verbale, apparaît normalement dans toute phrase ou groupe de phrases et apporte les indications principales sur son ossature (syntaxe). On ne possède en fait pratiquement aucune information sur l'ensemble des syntagmes nominaux qui représentent ce dont on parle, ainsi que sur leur cohérence. Le système dérivationnel permet juste de déterminer la façon dont on en parle. C'est là qu'intervient le module hypersémantique.

#### 3.2.2.9.1 Le module hypersémantique<sup>1</sup>

Il sélectionne les "objets spécifiques" du texte (objets véritables du texte) et les "objets standard", c'est à dire les objets que l'auteur utilise sans les définir, car il pense que ses lecteurs les connaissent parfaitement. Ces objets extraits (syntagmes nominaux parfois complexes), on les suit dans le texte, tels qu'ils ont été repérés, ou plus ou moins modifiés, *une fonction continue* donnant par exemple *une fonction uniformément continue*.

A priori, rien ne les distingue d'un point de vue linguistique. Pour accéder à ces unités sans faire appel au système des mots-clés ou à des systèmes analogues qui sont extérieurs au texte, on a recours à une méthode extra-linguistique. On aborde le traitement sans d'autre information que l'équivalence informatique d'un texte et de sa richesse typographique. Nous allons donc analyser sa typographie, partant du principe qu'une partie de l'information réside dans les moyens utilisés par l'auteur pour attirer l'attention du lecteur : hiérarchie des titres et sous-titres, paragraphes marqués par une lettre puce, modification des fontes, des polices, des corps, des chasses et des styles.

De même qu'il sera d'un accès difficile pour le lecteur humain, un texte sans marques typographiques se prêtera peu à ce genre d'analyse. Il faut noter à propos des textes informatisés que les marques de typographie sont absentes ou cachées. Elles correspondent en effet à des codes insérés dans le texte ou stockés en fin de fichier. Ils indiquent aux périphériques de sortie (écran ou imprimante) qu'une chaîne de caractères doit subir telle modification. C'est la raison pour laquelle nous avons codé les textes d'essai (4.3.1) en clair, pour restituer la présentation du texte source et utiliser ces particularités.

Le programme localise automatiquement tous les fragments mis en relief, en distinguant le texte, les titres, les sous-titres, les annotations, les légendes, les notes de bas de page... L'analyse de ces fragments permet de saisir de nombreux syntagmes nominaux qui deviennent alors des objets spécifiques potentiels (OSP).

---

(1) Dans "Analyse automatique du Tchèque. Définition d'un module prédictif général", Thèse d'Etat, Paris, 1979, P. POGNAN parle de module notionnel (p. 298).

Ils sont ensuite confrontés à la totalité du texte par un procédé d'hybridation et un procédé d'épuration<sup>1</sup>. Après avoir débarrassé le fichier des expressions en double, on procède à une hybridation, sans aucune référence au texte. Si l'on obtient par exemple les expressions *A de B* et *B de X*, on crée *AB de X* et *BA de X*. Cela permet d'associer les adjectifs aux substantifs et d'engendrer d'autres objets spécifiques potentiels. On obtient ainsi trop d'objets spécifiques potentiels dont certains sont sémantiquement inacceptables. Lors d'une phase d'épuration, on vérifiera alors sur le texte ou une partie du texte la présence ou l'absence des expressions obtenues, ce qui nous permet de travailler sur un plan discursif avec d'autres critères que les marques typographiques. Ces syntagmes nominaux-objets spécifiques ont une très forte probabilité de se retrouver dans le texte. Ils sont alors remis dans l'ossature prédicative, ce qui permet de vérifier la compatibilité entre cette ossature et les objets spécifiques ("*la façon dont on parle*", l'ossature, doit correspondre à "*ce dont on parle*").

### 3.2.2.9.2 Le module hypersyntaxique

Il est composé de deux parties qui correspondent à l'hypersyntaxe locale et à l'hypersyntaxe globale.

#### 3.2.2.9.2.1 Hypersyntaxe locale

Il s'agit de localiser :

- tout ce qui, à propos d'un objet, permet de l'introduire, de le définir, de le nommer
- tout ce qui correspond à une déduction, une remarque, une conclusion, un renvoi...

Les membres de phrase qui remplissent ces fonctions sont peu nombreux et sont étroitement liés, par leur structure, au système dérivationnel, dont on sait qu'il est essentiellement prédicatif puisque fondé sur des racines qui représentent des actions. On ne retrouvera ce type de phrases spécifiques qu'au niveau hypersyntaxique local. La maîtrise des opérations hypersyntaxiques locales permet de mieux situer la localisation des syntagmes nominaux-objets, et même, dans certains cas, de les extraire.

Soit la phrase type : "*On appelle suite numérique convergente toute suite qui...*". Le schéma est : On appelle XY avec Y commençant par un délimiteur indéfini.

Dans ce schéma, X (*suite numérique convergente*) est automatiquement extrait, et l'on sait simultanément que le membre Y contient la définition de l'objet X. Les situations ne sont pas toujours aussi simples mais la combinaison avec les techniques de repérage typographique donne d'excellents résultats.

#### 3.2.2.9.2.2 Hypersyntaxe globale

Dans certains textes à volonté didactique affirmée, des schémas de paragraphe se répètent, à quelques variantes près. Ces paragraphes sont en général consacrés à des raisonnements, des déductions, des descriptions d'expérience, des définitions...

---

(1) Ce module a été réalisé pour le français par J.-P. HUBAC. Il est décrit dans D.HERAULT : *Compréhension automatique et spectre sémantique*, Editions J. Favard, Paris, 1981, pp. 112-116

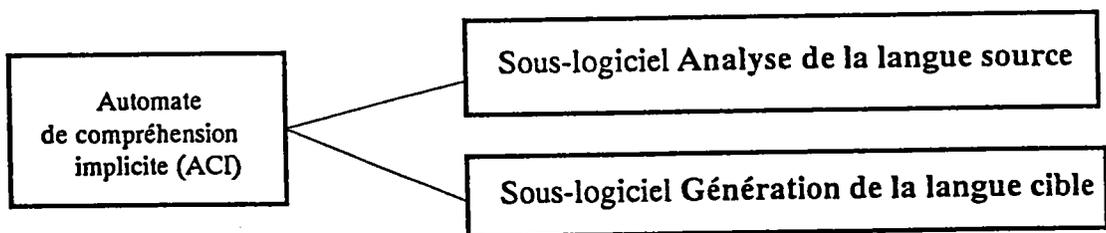
L'examen de ces schémas, qui varie peu d'une langue à l'autre, conduit à penser que l'on peut, par ce biais, accéder à la connaissance de pans entiers et en général, importants. Ces schémas hypersyntaxiques globaux, largement indépendants du sujet abordé, permettront peut-être d'envisager un transfert d'architecture, de langue à langue.

Nous allons maintenant montrer à quoi ressemble un logiciel qui effectue toutes ces opérations, avec la description du système d'interprétation automatique des langues germaniques. Nous en donnerons les caractéristiques (3.3.1), justifierons le choix des langues (3.3.2), résumerons (3.3.3) les modules de la première partie (ANALYSE) détaillés dans le chapitre IV, puis ébaucherons ce que devrait être la seconde partie (TRANSFERT) en définissant ses propriétés (3.3.4).

### 3.3 Le Système d'interprétation automatique des langues germaniques

#### 3.3.1 Caractéristiques générales

Il s'agit d'un logiciel de grande taille (25 000 lignes dans sa version définitive), écrit en PL/I Optimizer<sup>1</sup>. Le choix de ce langage doit beaucoup à sa maniabilité et à ses performances dans la manipulation des chaînes de caractères, tout comme au cadre des recherches (Centre de calcul du CNRS à Orsay) et aux ordinateurs utilisés au début des travaux. Une actualisation des travaux effectués, pour tirer un avantage des méthodes employées et justifier l'approche retenue, exigerait un passage à un autre langage de programmation. Nous pensons au langage C, pour sa souplesse et sa transportabilité et au système d'exploitation UNIX qui permettrait de s'affranchir des gros calculateurs. Le logiciel a la structure d'ensemble suivante:



Chaque sous-logiciel est divisé en *modules*, chaque module est constitué de *programmes*. Chaque programme ne dépasse pas 200 lignes. La description que nous allons proposer concernera le logiciel allemand-français. Néanmoins, le chapitre IV présentera des résultats sur l'anglais, le luxembourgeois et le néerlandais, pour le module verbal (4.3.2.2) et le module mots-composés (4.3.2.5).

#### 3.3.2 Langues sources germaniques

Comme nous l'avons indiqué en 3.2.2.6 le système n'utilise pas de dictionnaires bilingues traditionnels. Les langues germaniques ont été retenues pour étayer les raisons de ce choix. En effet, ces langues ont la propriété de permettre à leurs utilisateurs de composer les mots entre eux, de créer des mots composés.

---

(1) Un avantage du pl/1 : la possibilité de compiler les programmes et de les manipuler sous forme de modules chargeables, très rapides.

Ces mots sont par définition en nombre infini. Aucun dictionnaire ne peut les répertorier. Les systèmes de traitement automatique ont par conséquent beaucoup de mal à traiter ces mots composés, si ce n'est en sémantique très fermée, ce qui est d'autant plus regrettable que ce sont en général des "mots-objets" de la plus grande importance sémantique, puisque l'auteur du texte a pris un soin particulier à les créer.

C'est au prix d'un module très complexe et de grande taille que nous pouvons localiser tous les mots-composés d'un texte allemand, sans dictionnaire, et les décomposer en segments, chaque segment étant lui-même décomposé morphématiquement. Les données utilisées se résument à des tables de racines et d'affixes, ce qui permet, en raison d'un encombrement réduit, de travailler en mémoire centrale<sup>1</sup>.

### 3.3.3 Le sous-logiciel "ANALYSE DE LA LANGUE SOURCE"

- Un module initial qui met le texte en forme pour le travail à effectuer (900 lignes).
- Le premier module concerne les verbes. Quelle que soit la langue, la première partie de l'analyse concerne le verbe. Il est d'une importance capitale pour le reste des opérations (900 lignes).
- Le module "mots-composés" lui succède. Il est formé de quatre sous-modules (analyse générale, décomposition morphématique du lexique des mots/segments obtenus, segmentation par cohérence interne (dans le contexte du texte complet), et recherche des mots-composés verbaux (8000 lignes).
- L'analyseur syntaxique regroupe une quarantaine de programmes (6500 lignes).

Pour notre texte d'essai (3500 mot, 149 phrases), l'analyse complète a été effectuée en moins d'une minute sur le NAS 8090 du CNRS. Ce chiffre n'a cependant aucun sens, dès l'instant où il dépend du langage de programmation et de la puissance de calcul de l'ordinateur utilisé. Il fournit cependant un ordre d'idée sur l'efficacité de structure modulaire<sup>2</sup>.

#### 3.3.3.1 Le module verbal

Composé de quatre programmes principaux, il traite

- les verbes d'emprunts (*-ieren*)
- les verbes irréguliers
- les verbes réguliers
- la désambiguïsation de ZU

Un programme initial filtre et élimine les mots qui ne peuvent pas être des verbes ("mots-outils" et mots commençant par une majuscule). Son efficacité sera réduite pour les autres langues (absence de la majuscule du substantif).

En sortie, on associe au verbe sa base et un code qui précise sa situation (conjugaison et ambiguïtés). Une présentation particulière des données simplifie la tâche des programmes (1800 racines verbales et quelques bases pour traiter correctement le "GE").

---

(1) Les temps d'accès s'expriment en nanoseconde alors que sur une mémoire de masse (disquette, bande, disque dur), ils s'expriment en milliseconde.

(2) Il est en effet possible de faire tourner deux modules en même temps si leurs objectifs le permettent. L'architecture des appareils qui autorisent un fonctionnement "multitâche" est alors d'un grand avantage.

L'analyse est donc menée aussi loin qu'il est possible de la mener. C'est l'analyseur syntaxique qui lèvera la plupart des ambiguïtés mises en évidence.

### 3.3.3.2 Le module "mots-composés"

Les quatre sous-modules sont :

#### 3.3.3.2.1 Le sous-module "Analyse générale"

Indépendamment de tout texte, il segmente les mots à tiret, localise des foncteurs-avant et des foncteurs-arrière (la liste est subjective, ils correspondent à des éléments autonomes fonctionnant souvent en suffixation). Il traite ensuite les positions du "B", certaines chaînes (-HEITS, -KEITS, -SCHAFTS, -UNGS...) et surtout les paires et triades de segmentation (nous avons montré<sup>1</sup> que la jonction entre deux mots ou entre une racine et un groupe préfixal ou suffixal n'était possible que dans la mesure où la paire/triade de jonction était "anormale", c'est à dire, ne se trouvait pas à l'intérieur d'un mot non-composé. Toutes les jonctions ne présentent pas de telles caractéristiques (nous utilisons 400 chaînes). L'algorithme est le suivant : Si dans un mot ou segment, une paire ou une triade est repérée, on doit vérifier s'il ne s'agit pas de la jonction de la racine avec sa partie avant (jusqu'à 3 préverbes) ou sa partie arrière (nombre quelconque de suffixes). Après vérification, on peut segmenter. Le système est récursif, il s'applique aux segments obtenus et ainsi de suite. La méthode est très rapide mais ne localise que 75% des composés.

#### 3.3.3.2.2 Le sous-module "Décomposition Morphématique"

Une dizaine de programmes recherchent successivement les mots germaniques, puis les mots dérivant du système dérivationnel roman et orthographiés germaniquement. Par mot, on entend les mots simples, les mots composés non encore localisés et les segments des mots composés partiellement ou totalement décomposés. Ces programmes sont capables d'analyser correctement les racines romanes et germaniques homographes. Les résultats obtenus sont intéressants : les mots ou segments qui ne sont ni germaniques ni romans sont, soit des noms propres, soit des mots composés. On peut donc aboutir, dorénavant à une localisation intégrale des composés. A ce stade, il ne s'agit que de localisation sans décomposition.

#### 3.3.3.2.3 Le sous-module "Segmentation par Cohérence Textuelle"

On peut imaginer qu'un texte non-littéraire est cohérent dans son développement et que cette cohérence se retrouve au niveau des mots composés. On va donc tester cette hypothèse en recherchant la présence des segments déjà obtenus dans les mots que le sous-module précédent a relevés comme potentiellement composés. La présence est testée par la droite, sans la moindre lemmatisation, puis par la gauche et finalement, si cela est possible, centralement. Cette technique améliore de 20% les résultats obtenus par le premier sous-module. Il faut indiquer que sans la décomposition morphématique sélective, on n'obtiendrait peu de résultats.

A ce stade des résultats partiels sur le texte d'essai ("Des centrales électriques à piles à combustible"), les mots composés localisés mais non segmentés sont : *Baustein, Kohlestäbe, Notstromaggregate, Überlandleitung, Zeiteinheit, höhersiedenden, Lastabhängig, raumsparend, umweltschonenden, wagemutige et Wasserstoff*. Ce dernier mot est

---

(1) P. DIMON : "Aspects de la composition dans les langues germaniques et réalisation d'un algorithme de localisation des composés pour l'allemand", *Thèse de 3ème cycle*, Paris III, 1978

un segment intérieur sur lequel le foncteur *-stoff* n'a pu être mis en évidence. Les mots soulignés sont des formes verbales qui seront traitées par le sous-module suivant.

### 3.3.3.2.4 Le sous-module "Mots composés verbaux"

Les mots composés verbaux se retrouvent sous forme de participes présents ou passés, décliné ou non. Il n'y a jamais plus de deux segments, le second étant précisément le verbe, qui, dans la majorité des cas, n'est pas un verbe d'emprunt. A partir de ces hypothèses, on construit le sous-module en utilisant les programmes de décomposition-sélection et les programmes verbaux.

### 3.3.3.3 Le module "Analyseur syntaxique"

L'analyse est linéaire, sans invoquer le moindre arbre. Elle découpe chaque phrase en propositions (coordonnées, subordonnées) et les propositions en syntagmes verbaux et nominaux. C'est une analyse "logique" classique. Pour aboutir à ce résultat, chaque mot d'une phrase donnée est codé de sorte que l'on puisse circuler très facilement dans la phrase, dans les deux sens, afin que certains ensembles soient constitués (propositions, syntagmes) en respectant leur hiérarchie.

- programme "LIAISON" : repérage de certains groupes, indispensables par la suite (als daß, so daß, bis zu, ohne daß...)
- programme "VIRGUL1" : repérage des virgules délimiteurs, c'est à dire des virgules qui limitent les syntagmes nominaux et a fortiori, les propositions. Les virgules de liaison à l'intérieur d'un syntagme nominal sont strictement exclues.
- programme "COORDIN" : repérage des conjonctions-délimiteurs (aber, denn, oder, sondern, und...).
- programme "VIRGUL2" : certaines virgules, compte tenu de ce qui précède, deviennent délimiteurs.
- programme "PARSEP1" : repérage des particules séparables (sauf *ein*..)
- programme "ZUINF" : repérage de ZU particule qui, presque toujours, précède une forme infinitive. Ce repérage permet d'éliminer beaucoup d'ambiguïtés verbales.
- programme "RELAT1" : traitement des pronoms relatifs non ambigus (*denen, deren, dessen, was, wo, wobei...*). Pour les propositions concernées, il met en évidence le syntagme verbal associé.
- programme "RELAT2" : premier filtre des pronoms relatifs identiques à l'article défini.
- programme "RELAT3" : second filtre qui, lui, s'intègre au système verbal éventuellement associé, lequel doit avoir certaines propriétés pour qu'il s'agisse réellement d'un pronom relatif.
- programme "VIRGUL3" : adjonction de virgules délimiteurs, à la clôture des propositions relatives
- programme "RELAT4" : rôle de nettoyage en tenant compte de la position des virgules délimiteurs

- programme "ZUCOMP" : mise en évidence des liaisons entre UM, OHNE, STATT, AN-STATT et d'autre part ZU-infinitif.
- programme "CONJC1" : premier filtre pour les conjonctions de subordination, (*da, bevor, ehe, indem...*).
- programme "CONJ2" : second filtre traitant certaines conjonctions ambiguës en adverbe ou en proposition (*als, da, damit, seit, bis, während*)
- programme "PROSUB" : mise en évidence de la limite finale droite des propositions subordonnées non-relatives.
- programmes "G/ALS1" et "G/ALS2" : Ils délimitent à droite les "groupes-als" (*als=comme, en tant que*).
- programmes "ZU" : détermine des "ZU-quantité" (*zu=trop*), parmi les ZU qui ne sont pas des "ZU-infinitifs".
- programmes "GP1" à "GP5" : ils déterminent les groupes prépositionnels, avec une méthode voisine de celle des "groupes-ALS", mais plus complexe.
- programme "EIN" : examen de tous les EIN (article indéfini, particule séparable) et détermine les véritables particules séparables.
- programme "PARSEP2" : localisations des particules séparables "internes" (en d'autres termes, celles qui ne sont pas rejetées en fin de proposition).
- programmes "SNDIS1" à "SNDIS5" : déterminent les syntagmes nominaux disjoints, c'est à dire les syntagmes nominaux qui contiennent un ou plusieurs groupes prépositionnels.
- programmes "SNCON1" à "SNCON3" : déterminent les syntagmes nominaux connexes.

A ce niveau de l'analyse, tous les éléments des propositions ont été déterminés, sauf certains adverbes ou des adjectifs attributs. Il reste donc à reconstituer les syntagmes verbaux des propositions principales ou indépendantes.

- programmes "SV1" à "SV3" : reconstitution des syntagmes verbaux avec élimination des dernières ambiguïtés.

Tout ce qui pouvait être réalisé au niveau de l'analyse du texte source a été fait. L'ensemble des renseignements indispensables pour la réalisation du transfert sont désormais disponibles.

### 3.3.4 Le sous-logiciel "GENERATION DE LA LANGUE CIBLE"

Le second sous-logiciel a pour mission générale d'écrire, par transfert, le texte cible. Pour cela, on utilise de façon constante et soutenue, la correspondance entre les systèmes dérivationnels, afin de créer les mots ou segments de mots compréhensibles.

#### 3.3.4.1 Le module d'interprétation des mots composés

C'est le module qui s'impose en premier. Dans sa première version, le système interprète de la droite vers la gauche, en mettant des "de" de liaison, sauf dans certains cas

de foncteur (comme *METER*... en allemand, qui devient *mètre* en français sans renversement de l'ordre des segments).

ex. : *Wechselspannung*

La décomposition donne WECHSEL/SPANNung et fait apparaître un mot composé avec les racines WECHSEL (CHANG) et SPANN (TEND).

Par dérivation automatique et propriété particulière WECHSEL donne CHANGement et SPANNung devient TENSION. L'inversion systématique donne le résultat :

*Wechselspannung -> Tension (de) changement*

Le mot n'est pas compréhensible hors-contexte, mais dans les textes où il apparaît (électricité, électronique) le lecteur spécialiste reconstruira plus ou moins facilement *tension alternative*.

ex. : *Spannbeton -> béton tendu*

Le résultat est ici obtenu grâce au sélecteur roman/germanique et sera rétabli par un spécialiste du génie civil par *béton précontraint*.

Il convient de rappeler que que dans un texte non-littéraire allemand, les mots composés occupent une place considérable, plus de 25% des occurrences le plus souvent. Par ailleurs, comme on l'a déjà indiqué, ils représentent la majeure partie des "mots-objets" du texte et permettent de savoir immédiatement ce dont parle le texte. On peut donc donner du texte, en dehors de toute analyse syntaxique, une image qu'un spécialiste du domaine pourra exploiter, ceci pour un coût très faible. Ceci est à l'origine d'un système de documentation automatique "dynamique" que nous souhaiterions mettre au point pour les avantages qu'il apporterait (non lié à un système de mots-clés).

### 3.3.4.2 Le module d'interprétation des syntagmes verbaux

Il devra respecter les conjugaisons, pour le mieux. Dans son état actuel, le système se borne à un transfert rudimentaire. Au présent correspond le présent, au prétérit correspond le passé composé, le futur engendre le futur. Certaines formes font apparaître un subjonctif ou un conditionnel. Pour les temps composés, la présence d'un auxiliaire entraîne la présence de l'auxiliaire normal du verbe français.

### 3.3.4.3 Le module d'interprétation des mots non composés

Cette interprétation dépend de la nature germanique ou romane. Un mot roman est pris tel quel, sans correction orthographique dans la version actuelle. Un mot germanique, suivant son analyse, implique une reconstruction en français, ou bien, étant un "mot-objet", implique la consultation du lexique bilingue de ces mots.

### 3.3.4.4 Le module du transfert de syntaxe

Le module doit effectuer un transfert de syntaxe, proposition à proposition, phrase à phrase, en donnant à chaque syntagme, verbal ou nominal, la place qui lui convient. Ce transfert est pour l'instant très rudimentaire puisqu'il limite au maximum toute modification de l'ordre initial.

### 3.3.4.5 Le module de création des syntagmes nominaux français

En tenant compte de la hiérarchie qui a pu apparaître lors de l'analyse, le module construira les syntagmes nominaux français. Les adjectifs seront systématiquement placés après les substantifs qu'ils déterminent. Il en est de même pour les compléments des participes, passés et présents, jouant le rôle adjectival.

### **3.3.4.6 Le module final**

Il se borne à rassembler tous ces éléments et à donner comme résultat, une phrase approximativement française mais qui se veut, malgré ses défauts, parfaitement compréhensible.

On remarque que la morphologie n'a jamais été évoquée jusqu'ici. Nous espérons ne pas en avoir besoin. Son absence implique cependant quelques défaillances au niveau de l'interprétation (les genres ne sont pas ou mal reconnus). Pour les combler, il faudrait sans doute avoir recours à un dictionnaire monolingue de grande taille, ce que nous refusons. Nous avons pris le parti de ne pas utiliser la morphologie afin de rester dans les limites de nos objectifs.

### **3.3.4.7 L'aspect multilingual**

Une partie des résultats présentés dans le chapitre IV concernent le néerlandais et le luxembourgeois.

Pour savoir quelle était la qualité multilinguale de notre système, nous avons établi tous les fichiers de données indispensables, en néerlandais et en luxembourgeois (grâce aux conseils et à l'aide de J.-Cl. LEJOSNE). Nous les avons appliqués sans changer une ligne des modules "verbal" et "mots composés". Nous précisons d'ailleurs à ce propos, que nos programmes ne contiennent aucune donnée, qu'une indépendance totale existe entre les programmes et les données. Les résultats concernant le luxembourgeois sont parfaits, ceux concernant le néerlandais le sont presque (il suffirait d'intégrer le traitement de quelques anomalies qui ne perturberait pas l'analyse de l'allemand). Ces essais nous ont convaincu du caractère multilingual profond de notre système.

Dans le chapitre suivant, nous rappellerons quelques caractéristiques du système d'interprétation (4.1) avec ses contextes linguistique et informatique (4.2), puis nous verrons dans le détail la structure de l'automate (4.3) avec la mise en forme du texte (4.3.1), l'analyse lexicale et morphologique (4.3.2), l'analyse syntaxique (4.3.3) et les résultats complets (4.4).

**IV.**

**L'AUTOMATE GERMANIQUE-ROMAN**  
**Structure et fonctionnement**

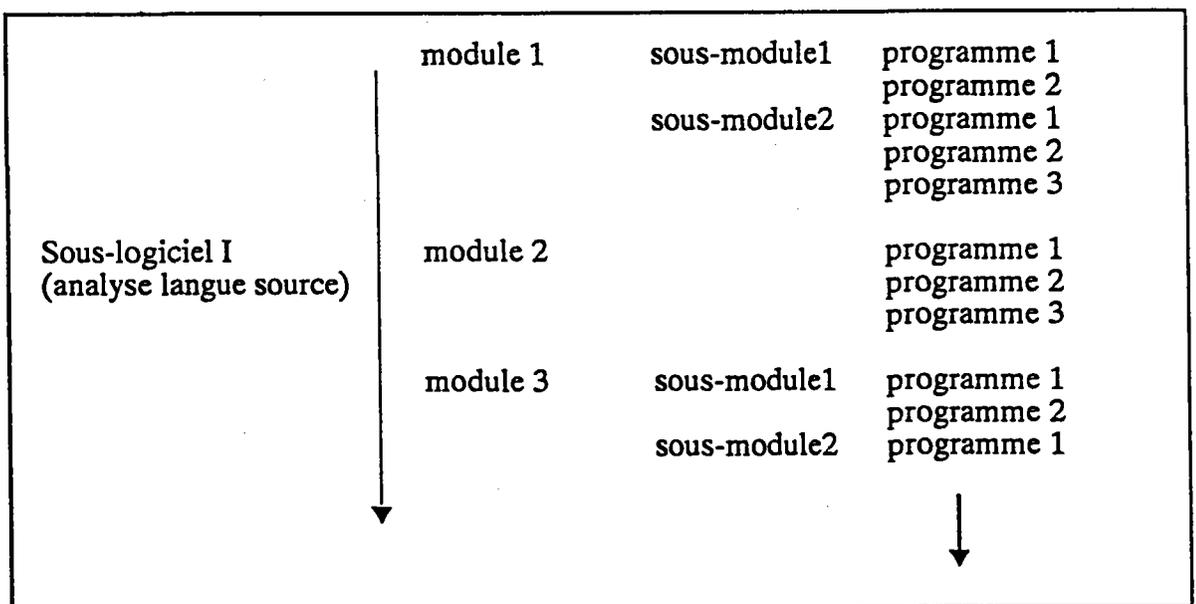
## 4.1 Caractéristiques générales

Rappelons l'intitulé du titre : *"Un système multilingual d'interprétation automatique, étape du sous-logiciel ANALYSE des langues germaniques"* et ce que sous-tendent les mots qui le composent

### 4.1.1 Système

Parce qu'il regroupe un ensemble d'idées, logiquement solidaires, en un appareil théorique et parce que cet appareil possède une structure de modules qui fonctionnent tous en interaction, nous l'appellons "système" .

Dans sa totalité, il devrait correspondre à un logiciel de grande taille, 25000 lignes de programme, écrites en PL/1 Optimizer. Ce logiciel est composé de deux sous-logiciels, le premier correspondant à l'analyse de la langue source et qui est l'objet de notre thèse, le second, à la génération de la langue cible. Chaque sous-logiciel est divisé en plusieurs modules, chaque module étant lui-même constitué de sous-modules et de programmes.



Ces programmes ont rarement plus de 200 lignes, situation qui permet de progresser par petits pas dans l'élaboration de l'automate et de vérifier minutieusement chaque étape.

L'implantation sur micro-ordinateur est un objectif prioritaire qui implique cette modularité. La possibilité d'envisager l'utilisation d'un bi- et même triprocesseur nous amène à déployer des niveaux de traitement tels que certains modules pourraient fonctionner simultanément pour un gain de temps calcul appréciable. La dénomination de tous les fichiers utilisés obéit à un ensemble de règles qui permet au système de fonctionner en multilingualité.

- Les données concernant les langues sont rangées dans des fichiers partitionnés (fichiers composés de plusieurs fichiers).
- Les langues traitées sont caractérisées par un suffixe: AL(L) pour l'allemand, OL(L) pour le néerlandais, L(U)X pour le luxembourgeois, A(N)G pour l'anglais, RO(M) pour les formes romanes. Dès lors, le fichier des désinences verbales s'intitulera, pour les différentes langues et dans l'ordre où nous venons de les citer:

.MORPHAL(DES)  
.MORPHOL(DES)  
.MORPHLX(DES)  
.MORPHAG(DES)  
.MORPHRM(DES)

Le fichier .MORPHAL, quant à lui, regroupera dans ses membres, toutes les données nécessaires à l'analyse de l'allemand, de même que .TEXALL tous les textes étudiés.

- Les programmes et les procédures qui les enchaînent sont indépendants des langues. Leur dénomination est donc générale et nous avons recours comme précédemment à des fichiers partitionnés. Le programme qui découpe un texte en phrases sera :

.CRJF(DECPHR)

A chaque membre de ce fichier de programmes est associée sa forme compilée :

.CHARGE(DECPHR)

Le fichier .PDPROC rassemble des procédures qui gèrent un groupe de programmes ou plus exactement, une succession de modules chargeables (programmes sous forme compilée) et les fichiers de données nécessaires.

.PDPROC(MOTCOMP)

- Le fonctionnement de chaque programme donne lieu à deux sorties distinctes. La première correspond au fichier d'entrée dans lequel sont insérés les résultats, la seconde ne contient que les résultats bruts et disparaîtra en phase opérationnelle. Les fichiers intermédiaires séquentiels (.TRANSIT1, .TRANSIT2) sont détruits au fur et à mesure, de sorte que la mémoire occupée soit minimale.

- Adapté à tout texte sans la moindre intervention humaine, il fonctionne en temps réel et ne requiert jamais plus de 300 K de mémoire vive. Il pourrait fonctionner actuellement sur un IBM AT (386 ou 486) à la vitesse de l'affichage naturel, beaucoup plus vite, même, en réalité. Il est géré par une procédure écrite dans le langage du système de gestion MVS/TSO, qui peut s'écrire dans d'autres systèmes. Les ordinateurs sur lesquels nous avons travaillé au CIRCE (Centre Inter Régional de Calcul Electronique du CNRS), IBM 370/168 et AMDAHL 470/V7 sont gérés par un système d'exploitation composé de deux logiciels, MVS et JES3.

MVS (Multiple Virtual Storage) gère l'ordinateur et ses composants matériels. T.S.O. (Time Sharing Option) est un sous-système de MVS et offre un mode de traitement autre que le traitement par lots. C'est le type "conversationnel" qui permet d'accéder aux ressources de l'ordinateur sous forme d'un dialogue maintenu entre l'utilisateur et le système.

• Tous les programmes ont été rédigés en PL1. Nous en expliquerons le choix en 4.2.2

Niveaux	Modules	Sous-ensembles	Programmes
0	1. Module PREPAR a) mise en forme du texte b) codage c) lexique tassé d) préparation pour le traitement des mots composés		a) lectext, decphr b) prolog c) classmot, tassmot d) mixer1, mixer2, classmot2, tassmot2, mixer3
	2. Module VERBAL a) traitement du verbe b) Adjectif / Adverbe c) Désambiguïsation Majuscule en tête de phrase		a) filtral, verball, verbal2 b) adv c) desamb
1	3. Module MOTS COMPOSES	MOTCOMP	tirets, fonct1, fonct2, jonct11, jonct12, jonct21, jonct22, gram11, gram12
		MOTCOMPA	fich11, fich12, fich13, aselect, sufall, netall1, netall2, rselect, sufrom, netrom, segmd, segmd1, segmint, recapit, aselect1, sufall1, netall11, reselect1, sufrom1, netrom1
		MOTCOMPB	
2	4. Module SYNTAXE		syntax1, syntax2, syntax3, syntax4, syntax5, syntax6, syntax7, syntax8, syntax9, syntax10, syntax11, syntax12

#### 4.1.2 Multilinguisme

Conformément au principe du choix des groupes de langues comme unité de raisonnement (3.2.2.8), et des langues germaniques (3.3.2) dans la cas présent, les programmes sont conçus pour fonctionner dans un environnement multilingual.

Deux exemples illustrent la généralité du logiciel :

- le traitement de la majuscule en début de phrase (voir 4.3.2.3)

- l'utilisation du 'B' et de son contexte dans le repérage des composés (voir 4.3.2.5.2.4)

Dans les deux cas, la fonction, la boucle ou la procédure est ou n'est pas activée selon la langue traitée.

#### 4.1.3 Interprétation

Un système d'interprétation ne traduit pas, dans le sens classique du terme. Il met en correspondance des signifiants de deux langues différentes et ne traduit que des morphèmes. C'est le lecteur qui, précisément, interprète grâce à sa compétence. Il cherche

à extraire de tout texte (non littéraire) suffisamment d'information pour que l'utilisateur qui connaît le domaine dont parle le texte en comprenne le contenu. Cette remarque a deux conséquences essentielles :

- les problèmes d'ambiguïtés, si complexes en ce qui concerne la traduction, sont ici considérablement réduits.

- Il est possible de proposer à l'utilisateur un texte écrit dans sa langue, compréhensible mais d'une mauvaise qualité linguistique (exemples pp. 226-228, pp. 668-674).

Les expériences conduites jusqu'à présent montrent que le spécialiste parvient à reconstruire le texte initial à partir des informations recueillies. Cette reconstruction de la langue-cible est l'objet du second sous-logiciel, et, comme le premier, repose sur un même fondement linguistique : celui du "système dérivationnel" d'une langue ou d'un groupe de langues.

L'**interprétation** sous-entend la compréhension. Le terme interprétation doit être entendu ici comme le synonyme vieilli de traduction. Nous ne pensons pas qu'un système qui doit extraire d'un texte écrit dans une langue naturelle un certain ensemble d'informations à transférer ou non vers une autre langue, puisse fonctionner convenablement à une quelconque étape de son fonctionnement si la **compréhension** n'est pas son principal fondement.

Dans le cas du système présenté, il s'agit de compréhension implicite. Les informations sont extraites d'un texte pour qu'un familier du domaine puisse en déduire le contenu et l'organisation. C'est l'intelligence seule de l'utilisateur qui permet de savoir si le système a, oui ou non, fonctionné correctement, contrairement à ce qui se passe pour la compréhension explicite, où l'on extrait du texte ce qui est nécessaire pour aboutir à des réalisations concrètes (réservation d'une chambre d'hôtel, gestion du trafic routier...). Le terme d'**interprétation** est d'autant plus justifié que les techniques de compréhension utilisées sont trop imprécises actuellement pour mériter le substantif "traduction" :

- absence de dictionnaire au sens usuel du terme.
- absence d'analyse morphologique (sauf pour traitement du verbe)
- génération, dans la phase de transfert, d'unités lexicales inexistantes.

La qualité du style de la sortie ne présente cependant qu'une très faible proportions d'erreurs, elle est indépendante du domaine traité et ne pourra que s'améliorer au fil des versions.

#### 4.1.4 Automatique

Comme une partie de l'analyse, le transfert s'appuie sur la notion de **système dérivationnel**. Pour les langues sur lesquelles nous travaillons, la structure du mot est pratiquement identique : Concaténation à une racine de préverbes, préfixes, infixes, suffixes, morphèmes de déclinaison ou de conjugaison. C'est l'utilisation de la correspondance entre systèmes dérivationnels qui nous permettra de créer automatiquement les mots ou segments de mot compréhensibles.

*Exemples d'interprétation des mots composés du texte source :*

Les segments romans sont reconnus et retranscrits. Les segments germaniques sont redécoupés, non pas par rapport à la racine, mais par rapport à la base, qui est, en général un préverbe + racine. On réalisera un transfert standard (indépendant du domaine abordé) vers le français, à partir de cette base.

Prenons l'exemple du mot Stromerzeuger découpé de la façon suivante:

1\$STROM + 1ER&ZEUG@ER# : à STROM en position initiale correspond COURANT. Pour l'autre segment, la base est ERZEUG à laquelle correspond PRODUCT. Il ne s'agit pas d'une forme verbale composée, et dans ces conditions, au suffixe ER correspondra le suffixe EUR.

**PRODUCTEUR (DE) COURANT**

Pour Dauerprüfung nous avons:

1\$DAUER + 1PRU\*F@UNG# : à DAUER correspond DURÉE et à PRU\*F correspond VÉRIFI/VÉRIFIC ce qui avec UNG donne :

*VÉRIFICATION (DE) DURÉE.*

Le transfert tient compte de la position du segment dans le mot.

*Interprétation des mots non composés:*

Les mots romans sont réécrits sans correction orthographique dans la version qui existe. Les mots germaniques, après analyse, sont soit reconstruits en français selon les principes évoqués plus haut, soit traduits en tant que mot-objet, après consultation d'un lexique bilingue de ces mots.

L'interprétation des syntagmes verbaux, le transfert de syntaxe et la création des syntagmes nominaux français tels que la hiérarchie mise à jour par l'analyse soit respectée complètent le second sous-logiciel et donnent du texte une image parfaitement compréhensible malgré ses défauts.

#### 4.1.5 Analyse

Dans l'immense majorité des cas, l'analyse d'un texte n'est pas autre chose que l'analyse morphologique et syntaxique de chaque mot, avec assez rarement une bonne segmentation de chaque phrase. Une telle analyse, quelles que soient sa finesse et sa précision, ne donne pas toujours une idée précise sur le contenu du texte. L'information qu'elle apporte est plus ou moins pertinente selon le degré d'information du décodeur. Si l'on veut privilégier la compréhension, il faut agir en sens contraire, c'est-à-dire ne faire jouer à la morphologie et à la syntaxe au niveau du mot, qu'un rôle secondaire.

En réalité, dans cette optique, ce ne sont plus les mots qui sont les unités de raisonnement, mais l'ensemble des éléments (racines, préverbes, préfixes, infixes, suffixes) qui constituent le système de formation des mots dont le sous-ensemble des racines verbales (racines prédicatives) constitue le noyau.

L'analyse va donc traiter les vecteurs de la compréhension en tenant compte du fait qu'un texte présente une double structure:

- les mots-objets, hors du système-dérivationnel, et des unités plus rares, normalement dérivées mais devenues mots-objets pour des raisons extra-linguistiques (ensemble, fonction...).
- les mots "dérivationnels" (surtout à partir de racines verbales).

- Beaucoup de langues empruntent volontiers, de sorte qu'une des missions de l'analyse consiste à associer chaque mot à son système dérivationnel, qu'il soit celui de la langue ou un système d'emprunt. Ce travail est effectué lors du *traitement des mots composés* et de la *sélection germanique/non germanique*.

- La sélection et la détermination des mots objets ne suffisent pas pour savoir ce dont on parle. Il est indispensable de mettre en évidence les syntagmes nominaux, prépositionnels ou non, c'est ce que réalise *l'analyseur syntaxique*.

- Tous ces syntagmes s'articulent autour des syntagmes verbaux sans l'étude desquels l'analyseur syntaxique ne pourrait pas fonctionner.  
( -> module VERBAL))

#### 4.1.6 Langues germaniques

Le contexte multilingual que nous avons choisi exige que le système fonctionne sur un groupe de langues. Une de leurs caractéristiques, surtout pour l'allemand, est leur polysynthétisme. Cette possibilité de "composer" les mots entre eux, de créer des mots composés de deux, trois, quatre segments et même plus, a posé d'énormes problèmes aux systèmes usuels de Traitement Automatique des Langues Naturelles (TALN). Les composés d'une langue germanique sont en nombre infini et, par conséquent, aucun dictionnaire ne peut les répertorier de façon exhaustive.

Or, la particularité de notre système est de ne pas utiliser des dictionnaires bilingues traditionnels. Ce choix original devait être étayé par des résultats irréfutables. C'est au prix d'un module complexe et de grande taille, présenté dans le chapitre IV, que nous avons traité les mots composés et démontré qu'il était possible de faire fonctionner un système-TALN sans dictionnaire monolingue de grande taille.

Le sous-logiciel ANALYSE a été élaboré, programme par programme, sur l'allemand et testé ensuite, en partie, sur le néerlandais, le luxembourgeois et l'anglais.

#### 4.2 Les contextes

La problématique du Traitement Automatique des Langues Naturelles concerne autant la linguistique que l'informatique.

##### 4.2.1 Le contexte linguistique

###### 4.2.1.1 Les difficultés

Il est nécessaire, semble-t-il, d'adapter les théories linguistiques en fonction des objectifs poursuivis (Traduction Automatique, dialogue homme-machine...), du degré de précision attendu et des contraintes liées à l'automatisation même du traitement. En effet, les théories linguistiques n'ont pas été élaborées dans l'optique d'une application immédiate dans des systèmes de traitement automatique.

Une ressemblance superficielle des langues naturelles avec les langages artificiels (langages de programmation...) ont conduit un grand nombre de chercheurs à appliquer au TALN les méthodes d'analyse utilisées pour traiter les expressions bien formées des langages artificiels. Les particularités des langues naturelles entraînent dans la plupart des cas des modifications et des extensions de modèles. Elles justifient parfois des formalismes propres pour lesquels il est indispensable de construire des fondements théoriques tels qu'on en mesure bien les limites et le pouvoir d'expression.

Les langages artificiels sont créés par l'homme qui les contrôle entièrement. Leur but est de décrire des phénomènes ou des tâches précises à l'aide d'un vocabulaire très restreint et de structures simples. Ils permettent de modéliser et de représenter à un niveau abstrait. Leurs qualités essentielles sont l'absence d'ambiguïtés et de données non-explicites. A chaque mot du langage n'est associé qu'un seul sens. Des facteurs historiques, sociologiques, cognitifs, stylistiques et techniques alimentent les nombreuses différences que nous allons résumer brièvement. Nous voulons ainsi montrer que s'il est possible de traiter de la même manière que pour le langage artificiel certains aspects de l'analyse syntaxique et sémantique du langage naturel, le principe ne peut être généralisé. Dans certains cas, il faudra construire des systèmes d'une grande complexité formelle et technique, quand bien même il ne s'agira que de sous-ensembles bien délimités de la langue. Les difficultés qui apparaissent au fil des différences fixent clairement les bornes actuelles du traitement automatique des langues naturelles.

- Pour un langage naturel, le nombre de phrases correctement construites est infini. On peut multiplier cet infini, bien que cela n'ait aucun sens, en considérant trois des niveaux classiques d'analyse. Sur *le plan lexical*, un dictionnaire n'est jamais exhaustif. Les vocabulaires spécifiques sont en évolution perpétuelle. Sur *le plan syntaxique*, l'ensemble des structures n'est pas déterminé, du fait même de la récursivité. Sur *le plan sémantique*, il se peut que l'ensemble des sens attribués à un mot ne soit pas déterminé. Et l'on ne tient pas compte du fait que le mot ne prend son sens que dans un contexte d'énonciation ! Cette première difficulté explique que le traitement automatique ne puisse s'appliquer aujourd'hui qu'à un sous-ensemble restreint de la langue, sémantiquement bien défini. Tout n'est pas si simple cependant, car il est impossible de figer une langue qui évolue sans cesse (qui crée des mots nouveaux et oublie ceux qui tombent en désuétude). Ces phénomènes sont particulièrement fréquents dans les domaines techniques et scientifiques.

- En langage artificiel, toute erreur de syntaxe entraîne immédiatement une interruption du compilateur, malgré des systèmes complexes de récupération d'erreurs. La langue naturelle autorise quant à elle une plus grande souplesse puisque, dans certains cas, une phrase agrammaticale peut être compréhensible. "*...il existe, en fait, un seuil maximal de distorsion syntaxique, au delà duquel une phrase n'est plus compréhensible. Ce seuil n'est pas fixe et dépend fortement de la signification de la phrase, de sa complexité et de sa forme ainsi que du contexte dans lequel elle a été énoncée*".<sup>1</sup>

- Une troisième source de difficultés sont les ambiguïtés que l'on trouve à des niveaux variés (cf. 1.2) et qui ne sont pas toujours évidentes dans le sens où le lecteur opère souvent naturellement le bon choix. Il est extrêmement difficile de construire une machine qui intègre immédiatement une phrase dans son contexte d'énonciation, contexte dont elle pourrait extraire, comme le fait presque inconsciemment l'homme, les données propres à lever des ambiguïtés.

- Rappelons enfin les éléments sous-entendus (ellipses...) ou contractés (anaphores...) qui constituent autant d'éléments implicites et justifie notre approche de la compréhension implicite.

Comme nous le voyons, les problèmes sont nombreux et complexes. Aucun n'est parfaitement résolu, à l'heure actuelle. Aussi, les recherches se sont orientées vers des systèmes appliqués à des sous-ensembles de la langue, afin de limiter le champ des difficultés. A quelques exceptions près, les systèmes de Traduction Automatique ont évolué vers des systèmes d'aide à la traduction.

#### 4.2.1.2 Les solutions de l'ACI

##### 4.2.1.2.1 Les niveaux morphologique et lexical

Il s'agit d'étudier la forme que prennent les mots dans la phrase. La conjugaison des verbes ainsi que les marques de genre et de nombre n'ont pas grande importance dans la stratégie que nous avons définie (cf. 3.2.2 Les principes de l'ACI). Les formes sont reconnues et les catégories identifiées - il faut souligner ici l'avantage de la majuscule en allemand - grâce à des tris rapides et des listes spécifiques ("mots-outils", "mots-objets", "mots-exceptionnels", "foncteurs")<sup>2</sup>. Les "mots-dérivationnels" sont décomposés à partir de listes de racines, préverbes, préfixes, suffixes et morphèmes grammaticaux.

(1) A. GAL, G. LAPALME, P. SAINT-DIZIER : *PROLOG pour l'analyse du langage naturel*, Ed. Eyrolles, Paris, 1989, p. 4

(2) cf. 3.2.2.3.2, la classification du lexique

Nous abordons là un des fondements de notre analyse. Partant de la structure du vocabulaire d'une langue, nous avons voulu montrer qu'il était possible de le réduire à une combinatoire. Dès lors, il semble naturel d'exploiter l'hypothèse d'une parenté étymologique pour convertir les mots d'une langue source dans une langue cible.

**Dans toute langue, le vocabulaire est illimité pour deux raisons.**

- à cause des emprunts aux autres langues (cf. en français : *un ersatz, le hard, le soft, un solo, le couscous...*).
- à cause de la dérivation et de la composition. Des mots simples sont tout à coup affectés d'un ou de plusieurs affixes, *câbler, conforter* sont des dérivés récents. En allemand, tous les composés commençant par *Bund-* (=fédération) ont été formés après la naissance de la République Fédérale (*Bundesrepublik*) il y a moins de 40 ans (*Bundesbahn, Bundeswehr* etc.).

En plus de ces possibilités d'accroître son vocabulaire, la langue a la possibilité de multiplier les significations d'un même stock de vocables par le procédé de la métaphore (littéralement "déménagement" d'un mot d'une signification à une autre). Par exemple, le mot *table* est utilisé par plusieurs techniques pour désigner des objets qui n'ont en commun que de présenter une surface plane : *table de logarithmes, table traçante, table à dessin* etc.

Les trois procédés peuvent se conjuguer pour s'appliquer à une même racine : *le coco* (fruit du cocotier) est un mot d'origine portugaise. Ce mot a été utilisé par métaphore pour la tête (*se casser le coco*). La dérivation a donné *cocotier*. Ce mot a fait à son tour l'objet d'une métaphore : *secouer le cocotier*.

Il est donc au total impossible de mettre dans la mémoire d'un ordinateur l'ensemble des mots d'une langue et celui des significations d'un nombre même limité de ces mots.

### La réduction du vocabulaire à une combinatoire

Puisque le nombre de suites de lettres entre deux blancs est illimité dans une langue et que le nombre de significations associées à chacune de ces suites de lettres est lui-même indéterminé, il faut procéder à une double réduction :

- celle du nombre illimité de suites de caractères à un nombre fini de suite de caractères
- celle des significations associées à ce nombre fini de suites de caractères à un nombre également fini.

Pour chacun de ces niveaux de réduction, on utilise actuellement les procédés suivants :

- Pour évacuer le plus possible de phénomènes de métaphore, on ne décrit pas le vocabulaire d'une langue "en général", mais le vocabulaire d'un "domaine sémantique clos", par exemple le domaine des mots et locutions décrivant une technique bien délimitée. Dans ce domaine sémantique, les mots sont utilisés avec un de leur sens seulement. En allemand, *Birne* correspond à *poire* pour le jardinier, *convertisseur* pour le sidérurgiste et *ampoule* pour l'électricien. Si l'on décrit la marche d'une aciérie, on associera *Birne* à *convertisseur*.

- On peut considérer que dans une large mesure, les emprunts faits à une langue A sont compris par les gens qui parlent la langue B et la langue C qu'on essaie de mettre en

correspondance par la TAO. De fait, les mots anglais de l'informatique sont compris en France et en Allemagne. Il est donc inutile de chercher à les traduire s'ils figurent dans un texte allemand que l'on interprète en français. L'élimination des mots d'emprunt se fait automatiquement par reconnaissance des formes graphiques (analyse des suites de lettres). Par exemple *hard* ne peut pas être un mot allemand (la seule forme possible serait *hart*), *solo* ne peut être ni un mot allemand ni un mot français etc.

- L'ACI décompose parfaitement les mots simples, dérivés et composés, en racines, affixes et morphèmes grammaticaux. Cette décomposition procède par repérage de constantes (les affixes) par rapport aux variables qui sont les racines supports des dérivations. La pertinence de l'opposition variable/constante tient à ce que les affixes sont beaucoup moins nombreux que les racines auxquelles ils s'associent.

trag-		port-	
denk-	bar	pens-	able
mach-		fais-	

La question qui se pose au terme de cette analyse est de savoir ce que l'on peut faire avec un stock de racines, d'où les emprunts sont éliminés et un stock d'affixes qui, dans l'usage réel de la langue, ne sont jamais utilisés seuls.

Le repérage des racines et des affixes associés dans un mot quelconque fournit la structure étymologique de ce mot :

Préfixe(s)	radical	suffixe(s)
DE	FORM(E)	ER
RE	ECRI-	RE
IN/IM	POSS-	IBLE

En quoi cette analyse facilite-t-elle la conversion des mots d'une langue dans une autre ?

### Les parentés étymologiques dans les langues en contact

Les langues européennes ont une longue histoire commune. Ensemble, elles se sont nourries de mots latins et grecs et au gré des variations de poids relatifs des différents pays, elles ont emprunté beaucoup les unes aux autres. Ces emprunts se sont effectués de deux façons : par emprunt des formes phoniques ou graphiques et par reconstruction.

On peut éliminer ici le cas des emprunts des formes graphiques puisqu'on fait l'hypothèse qu'un mot qui n'a pas une forme graphique "normale" dans la langue source doit se trouver aussi dans la langue cible (cas de *hard*, de *cash*, de *philosophie*...).

Dans le cas d'emprunt par reconstruction d'un mot d'une langue A par une langue B, le processus est le suivant :

L'emprunteur procède à l'analyse étymologique du mot à emprunter. Il établit une correspondance terme à terme entre les racines et affixes reconnus et des racines et affixes de sa langue. Il construit un mot qui, sur la base des correspondances adoptées, est la traduction terme à terme de la structure étymologique reconnue. C'est ainsi que beaucoup de mots qui ont une forme parfaitement allemande sont des traductions des structures étymologiques du latin.

IN -	PRESSIO	BENE -	FACTUM
EIN -	DRUCK	WOHL -	TAT

Notre hypothèse est :

- qu'entre les langues européennes, ces emprunts par reconstruction sont très nombreux.
- que dans l'histoire de chacune des langues, les règles de correspondance des combinatoires sont restées assez stables pour qu'avec un système de correspondances souvent bi-univoques entre les stocks finis de racines et affixes de deux langues A et B, on puisse générer des mots compréhensibles dans la langue A à partir des structures étymologiques. Soit par exemple les mots :

français	SOUS -	ESTIMER
anglais	UNDER-	RATE
allemand	UNTER-	SCHÄTZEN

Il est évident que l'on peut établir une correspondance "en dehors de ces mots" :

SOUS	UNTER	UNDER
ESTIMER	SCHÄTZEN	(TO) RATE

Ces correspondances étant données, on peut traduire mécaniquement l'un des mots dans les deux autres langues.

Dans la réalité des langues, ces reconstructions étymologiques sont plus souvent compréhensibles que correctes, car le seul fait que les éléments de la combinatoire entrent en contact entraîne souvent une modification de la forme de l'un des deux.

Soit les mots

anglais	UNDERWRITE
allemand	UNTERSCHREIBEN

on considèrera comme intuitif qu'ils ont la même structure étymologique dont le correspondant français est

français	SOUS - ECRIRE
----------	---------------

Mais dans ce mot, comme dans une série d'autres (inscrire, prescrire) ECRIRE prend la forme SCRIRE (cf. latin SCRIBERE).

La mise en forme réelle des mots imposerait de compliquer considérablement les règles de correspondance entre les éléments des combinatoires et ceci pose le problème du coût de la traduction assistée par ordinateur.

Il semble cependant qu'un système de correspondances simple entre deux stocks finis d'éléments fournisse une assistance appréciable à l'interprétation (SOUS-ECRIRE CHEQUE a un sens) et il est établi que ce niveau d'analyse est celui qui convient pour construire une traduction "intraordinateur" (conversion d'instructions d'une langue dans une autre), où la correction formelle de la traduction n'est qu'une complication.

Partant de la constatation que le vocabulaire de n'importe quelle langue est illimité, nous nous sommes posé le problème de savoir s'il était possible de remonter automa-

tiquement des vocabulaires réels aux stocks finis d'éléments que la combinatoire des différentes langues met en jeu pour le produire.

Les travaux ont établi la possibilité de dissocier les mots composés et les mots dérivés en racines et affixes. En d'autres termes, il nous semble acquis que pour des domaines sémantiques clos (sans métaphores), on peut effectivement ramener le vocabulaire à un ensemble fini d'éléments.

L'hypothèse directrice de la recherche est que les parentés étymologiques des langues européennes permettent de construire un système de correspondance entre les ensembles finis d'éléments des différentes langues et de générer les mots de la langue cible à partir des éléments constitutifs des mots de la langue source.

Cette mise en correspondance d'éléments de deux langues ne génère pas que des formes correctes. Elles sont cependant compréhensibles dans tous les cas.

Lorsque nous avons démarré la construction de l'automate, les techniques informatiques disponibles ou à notre disposition ont lourdement pesé sur le choix que nous commentons plus bas (cf. 4.2.2). Nous avons développé une analyse descendante avec retour en arrière et déterminé l'ordre d'intervention des modules pour réduire les accès fichiers au minimum. Après la mise en forme du texte source et la transformation des phrases en structures basées manipulées par des pointeurs, la première analyse (cf. 4.3.2.1) repère les mots-outils simples (*daß*, ...) les mots-outils disjoints (*so daß*, *selbst wenn*, ...) ainsi que les expressions numérales.

Avant d'aborder la phase de décomposition lexicale - repérage des mots simples, dérivés, composés et découpage en racine(s) et affixe(s) - il est nécessaire de traiter le verbe pour inclure dans la liste des composés potentiels, les formes verbales composées (*maschineschreiben*, ...). Le module verbal (cf. 4.3.2.2) ne fonctionne que sur les mots du texte placés en début de phrase (la majuscule est alors ambigüe) et sur les mots qui commencent par une minuscule et ne sont pas des mots-outils. La recherche est rapide car elle s'applique à des formes dont 60% sont réellement verbales. Elle s'effectue de la droite vers la gauche pour le test des terminaisons et l'application d'un filtre éliminant les formes qui ne peuvent être verbales, puis de la gauche vers la droite pour la reconnaissance et le découpage de la préverbalisation, la reconnaissance de la racine, le contrôle du résidu et le codage.

Les deux étapes suivantes peuvent se dérouler en parallèle.

1. - La désambiguïsation de la majuscule en tête de phrase (cf. 4.3.2.3) grâce aux résultats du module verbal et aux fichiers intermédiaires qui contiennent : le texte complet avec les mots-outils, la partie numérale et le verbe codés, la liste de tous les mots commençant par une majuscule et rencontrés à l'intérieur d'une phrase, la liste des mots non codés commençant par une minuscule qui ne sont de ce fait ni mot-outil ni verbe. Lui succède le codage des adjectifs et des adverbes (cf. 4.3.2.4).

2. - La préparation des données pour l'analyse des mots composés et l'analyse proprement dite (cf. 4.3.2.5). On ne retient que les mots de la phrase qui ne sont pas des mots-outils et dont la longueur est supérieure à 6 caractères, formes verbales y compris.

#### 4.2.1.2.2 Le niveau syntaxique

Nous n'avons pas utilisé les formalismes de la logique. Il est apparu que le repérage de certains délimiteurs aboutissait à un bon découpage des syntagmes. Dans un premier temps, il nous a fallu localiser les ponctuations "fortes" (cf. 4.3.3.2.1) pour traiter ensuite les relatives (cf. 4.3.3.2.2-4.3.3.2.3) et les conjonctives (cf. 4.3.3.2.4). L'analyse

se fait de la gauche vers la droite, séquentiellement, avec des retours en arrière (cf. 4.3.3.2.5). le traitement du "ZU" (cf. 4.3.3.2.6) précède l'analyse des syntagmes prépositionnels (cf. 4.3.3.2.7-4.3.3.2.8), des syntagmes nominaux (cf. 4.3.3.2.9-4.3.3.2.10) et des syntagmes verbaux (cf. 4.3.3.2.11).

## 4.2.2 Le contexte informatique

Lorsque nous avons commencé à travailler sur l'allemand, le centre de recherche Jean Favard utilisait les ressources informatiques du CNRS (Centre Inter Régional de Calcul Electronique à Orsay). Nous avons entré les textes d'essai sur cartes perforées pour les traiter ensuite sur les gros calculateurs de l'époque, la microinformatique n'en étant qu'à ses débuts. Orientés essentiellement vers le calcul numérique, les langages de programmation disponibles présentaient de grosses lacunes en matière de manipulation de chaînes de caractères. Ils ne permettaient pas d'étudier correctement les chaînes riches (majuscules, minuscules et signes diacritiques) et d'une certaine longueur. Le PLI, lié à IBM et disponible au CIRCE sous la forme d'un excellent compilateur, était alors le seul langage de programmation autorisant un traitement confortable des chaînes de caractères (longueur <32760 car.). C'est pourquoi nous l'avons alors retenu pour l'application informatique de nos hypothèses de travail. Le volume important des recherches, le respect des cahiers des charges et la dynamique propre à un petit laboratoire n'ont pas permis la remise en question d'un choix qui, s'il s'imposait en 1980, est maintenant complètement dépassé.

Sa nature procédurale n'a pas toujours permis d'aborder les problèmes comme nous pourrions le faire aujourd'hui. Dans 1.4, nous avons présenté les modèles de la langue et les différents aspects de la théorie transformationnelle qui sert de base au développement de nombreuses grammaires, dont les grammaires logiques (cf. 1.5.8). Le but de ces grammaires était de décrire la structure des phrases et certains phénomènes linguistiques sans qu'il soit question de développer un outil informatique capable de mettre en adéquation la phrase et sa structure. Or, pour un traitement automatique de la langue naturelle, il faut disposer d'algorithmes permettant de déterminer si une phrase correspond à une structure déterminée et, si c'est le cas, la manière dont sont reliés ses différents constituants. Les premiers travaux de ce genre ont concerné l'application des grammaires formelles à la définition des langages artificiels. Nous venons de voir à propos du contexte linguistique (cf. 4.2.1) que pour le langage naturel, il fallait tenir compte de phénomènes beaucoup plus complexes, ce qui complique très sérieusement les structures des grammaires et des analyseurs. Un formalisme s'est petit à petit imposé, celui des grammaires logiques, basées principalement sur le langage PROLOG. Un de ses mérites est la *déclarativité* : un programme PROLOG est en grande partie une suite d'énoncés logiques indiquant des relations entre ceux-ci. Des techniques d'unification visent à mettre en correspondance les éléments d'une phrase avec les composants d'un ensemble de règles décrivant la grammaire. On peut considérer les règles comme des règles de réécriture (génération de phrase). En inversant l'opération, on examine si une suite de mots forme une phrase acceptable (analyse de phrase). Il est ainsi possible de faire correspondre à la phrase analysée une description structurelle. Exécuter un programme, c'est prouver, à partir des énoncés disponibles, qu'un nouvel énoncé est vrai. Le modèle de l'ACI est certes loin des grammaires logiques. On pourrait cependant l'implémenter en PROLOG. L'adéquation du formalisme de la programmation en logique au formalisme de l'ACI serait peut-être moins forte. On conserverait de toutes façons les avantages de la modularité, du mode déclaratif et de la concision.

Avant d'aborder la présentation des différents modules de l'ACI (cf. 4.2.2.4) et afin de mieux cerner ce que peut être un langage déclaratif, nous introduirons rapidement le langage PROLOG en rappelant ses domaines d'application (cf. 4.2.2.1). Nous le placerons dans le cadre linguistique (cf. 4.2.2.2) et proposerons enfin quelques exemples pour illustrer son fonctionnement (cf. 4.2.2.3).

#### 4.2.2.1 Introduction à PROLOG

Utilisée comme mode de représentation et outil de modélisation, la logique apparaît comme un moyen particulièrement adapté au Traitement Automatique des Langues Naturelles. Il s'agit de la logique du premier ordre (logique des clauses de Horn) et pour une représentation sémantique satisfaisante d'énoncés en langue naturelle, des logiques temporelle, modale et non-monotone.

Les champs d'application du langage PROLOG sont variés, qui s'étendent de l'analyse automatique (grammaires logiques...) aux représentations sémantiques en passant par la réalisation d'interfaces intelligentes ou la génération automatique de texte (génération aléatoire, génération à partir d'une représentation interne, explications fournies par un système expert, réponses générées par l'interrogation d'une base de données, traduction automatique ou génération d'un dialogue).

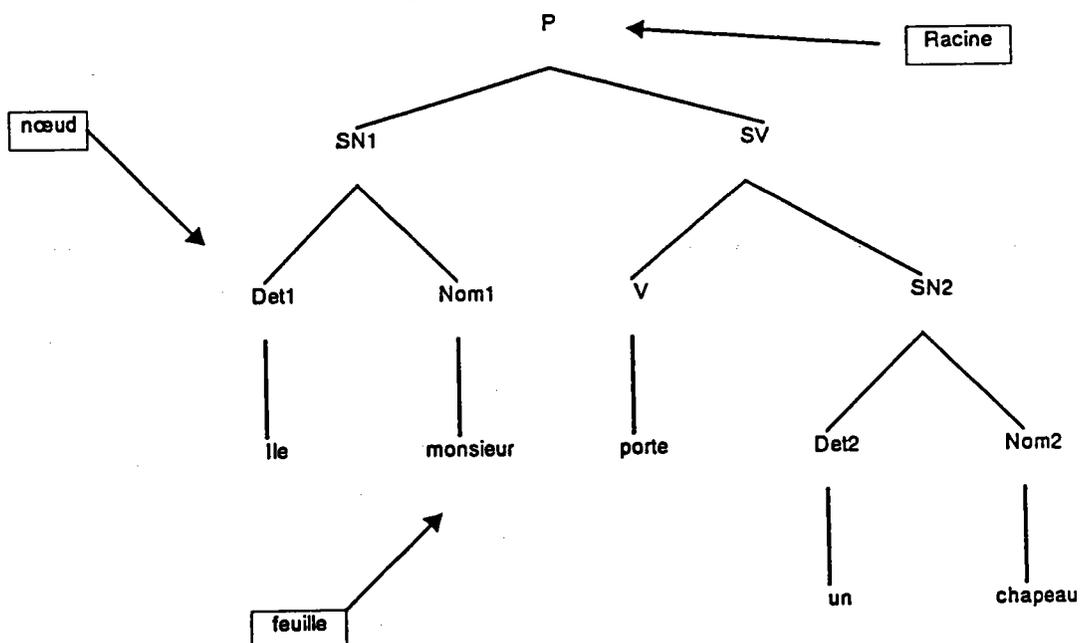
En PROLOG, tout se résume à trois types d'activité :

- On apporte des connaissances au système en *déclarant* des faits sur des objets et leurs interactions.
- On définit des *règles* de raisonnement sur ces faits.
- On pose des *questions*.

La structure fondamentale des données est ici l'arbre, véritable fondement des traitements effectués par et en PROLOG.

##### 4.2.2.1.1 L'arbre

L'arbre est composé d'une *racine* unique, en haut du schéma. Les *branches* partent de la racine en nombre quelconque mais limité et établissent des relations de dominance ou de fraternité entre les *nœuds* et les *feuilles*. Elles se subdivisent au niveau des nœuds. Dans le bas du schéma, on trouve les feuilles.



Racine : P

Nœuds : SN1, SV, Det1, Nom1, V, SN2, Det2, Nom2

Feuilles : le, monsieur, porte, un, chapeau

Le nœud SN1 est le père des nœuds Det1 et Nom1. Les nœuds Det2 et Nom2 sont les fils du nœud SN2. Det1 et Nom1 sont des frères, ils ont le même père immédiat, SN1.

Pour plus de commodités, les arbres sont représentés de façon aplatie :

P(SN1(Det1(le),Nom1(monsieur)),SV(V(porte),SN2(Det2(un),Nom2(chapeau))))

Les éléments frères sont séparés par des virgules, les relations de dominance sont exprimées par les parenthèses.

Lorsqu'on lui pose des questions, PROLOG utilise des fonctions prédéfinies et des mécanismes de démonstration automatique pour effectuer des raisonnements sur les connaissances qu'on lui apporte. Ces connaissances lui sont fournies sous forme de *faits* et de *règles*.

#### 4.2.2.1.2 Les faits

Soit la phrase : *souris est du genre féminin*.

On la décompose en deux objets identifiés par *souris* et *féminin* et une relation *genre*. En logique, *souris* et *féminin* sont deux constantes qui permettent de définir le prédicat :

`genre(argument1, argument2).`

On dira par convention que `argument1` représente le mot dont on définit le genre, et `argument2`, le genre.

En PROLOG, on obtient le fait :

`genre(souris, feminin).`

Dans le contexte du langage de programmation, on parle de *fait* au niveau sémantique et de *terme* au niveau syntaxique. On peut définir un fait avec un nombre quelconque d'arguments. Une relation comme *genre* aura toujours le même nombre d'arguments et chaque argument aura toujours la même signification. Le fait énonce une propriété ou une relation générale. Les arguments sont des valeurs concrètes qui identifient les entités qui satisfont la propriété ou la relation. On peut définir un fait sans argument(s) :

`syntagme_nominal.`

Le fait a alors un caractère absolu. Si le fait a un seul argument, il énonce en général une propriété sur un ensemble d'objets qu'il admet comme argument.

`determinant(le).`

`determinant(la).`

`nom(souris).`

`nom(chat).`

*le* et *la* ont la propriété d'être un déterminant. *Souris* et *chat* ont la propriété d'être un nom.

Les faits qui appartiennent à un programme sont vrais. Leur ensemble constitue une base de faits. Cette base de faits peut être créée à l'aide d'un éditeur puis interprétée par PROLOG ou introduite en mode interactif. Dans la cas d'applications répétées, on range la base de faits dans un fichier. On peut interroger la base de faits.

## 4.2.2.1.3 Les questions

La question reprend alors la forme du fait précédée d'un symbole spécial. Elle peut être élémentaire ou utiliser des variables. Pour répondre à une question posée, PROLOG parcourt la base de faits pour y rechercher les faits qui correspondent au fait de la question. Deux faits correspondent si leurs prédicats sont identiques et si leurs arguments de même rang sont les mêmes. Si PROLOG trouve un fait qui correspond à la question, il répond *yes*. Sinon, il répond *no*.

Lorsqu'on fait appel à des variables, PROLOG parcourt la base de faits avec les variables libres au départ et recherche un fait qui corresponde à la question. Si un argument est représenté par une variable libre, PROLOG va permettre à cet argument de correspondre à n'importe quel argument de même rang dans le fait, quel qu'il soit.

```
deteste(pierre, chien).
deteste(pierre, oiseau).
deteste(paul, oiseau).
```

```
? - deteste(pierre, X).
```

A la question *Y a-t-il quelque chose que Pierre déteste*, PROLOG répond :

```
X=chien
```

Au départ, X est libre. PROLOG va trouver un fait dont le prédicat est *deteste* et le premier argument *pierre*. X représentera alors le deuxième argument du fait, quel qu'il soit. Le premier fait trouvé (la recherche s'effectue de haut en bas) est :

```
deteste(pierre, chien).
```

X représente *chien*. X est *instancié* à *chien*. PROLOG marque l'endroit de la base où la correspondance est trouvée pour repartir éventuellement du marqueur, si la recherche se poursuit. On dit que PROLOG essaye de *re-satisfaire* la question.

Le prochain fait est : 

```
deteste(pierre, oiseau).
```

X est alors instancié à *oiseau*.

Il est possible de compliquer la question et de demander *Pierre et Paul se détestent-ils ?* PROLOG devra satisfaire deux buts pour résoudre le problème, en demandant si *Pierre déteste Paul* et si *Paul déteste Pierre*.

Soit la base :

```
deteste(pierre, chien).
deteste(pierre, oiseau).
deteste(paul, oiseau).
deteste(paul, pierre).
```

Pour exprimer la conjonction de deux buts, nous écrivons :

```
? - deteste(pierre, paul), deteste(paul, pierre).
```

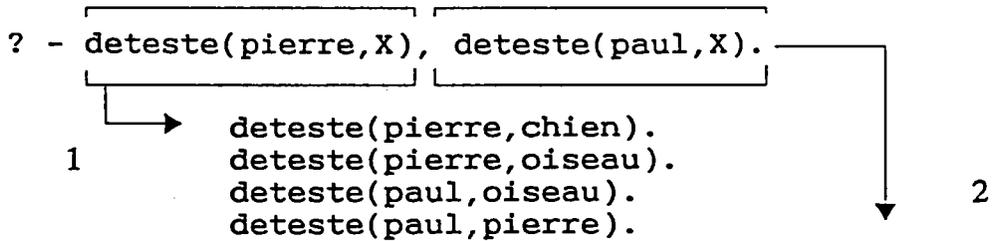
PROLOG va chercher à satisfaire un but après l'autre en cherchant pour chacun un but qui lui corresponde dans la base de faits. Le deuxième but est vérifié, pas le premier. La réponse à la totalité de la question sera NO.

On peut combiner les conjonctions et l'utilisation de variables. Pour savoir s'il y a quelque chose que Pierre et Paul détestent tous les deux. Il faudra vérifier les buts :

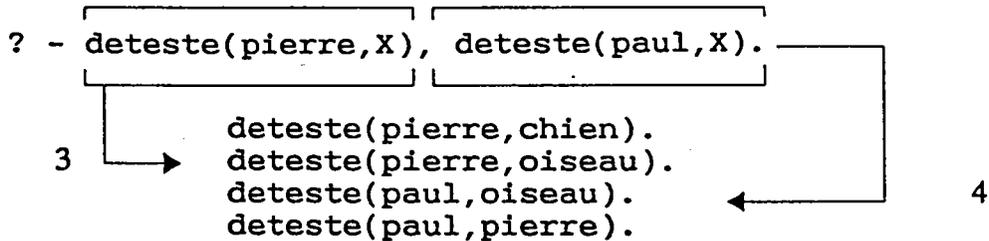
- trouver un objet X que Pierre déteste
- voir si Paul déteste cet X, quel qu'il soit

? - deteste(pierre,X), deteste(paul,X).

PROLOG essaie de satisfaire le premier but. Si le fait existe dans la base, il marque l'endroit et essaie de satisfaire le deuxième but.



1. le premier but est vérifié. X est instancié à chien.
2. Echec du deuxième but. On effectue un retour en arrière<sup>1</sup>.



3. La valeur de x est effacée. On essaie de re-satisfaire le premier but qui est à nouveau réussi. On instancie X à oiseau.
4. Le deuxième but est également réussi.

#### 4.2.2.1.4 Les règles

Il s'agit d'un moyen d'expression très puissant qui permet de généraliser la façon d'exprimer des connaissances et confère à PROLOG un pouvoir déductif.

Soit la relation : pere\_de(X,Y). qui énonce de façon générale que X est père de Y.  
Soit la base de faits :

```
pere_de(jean, paul).
pere_de(robert, henri).
pere_de(pierre, marcel).
pere_de(dominique, pierre).
```

(1) La notion de retour en arrière est très importante en PROLOG. Nous donnons plus de détails p. 266

A partir de ces faits, on veut introduire la relation : `grand_pere(X,Y)`. qui énonce que X est le grand-père de Y.

On pourrait créer une liste de nouveaux faits. On préfère définir une règle selon laquelle, si nous avons X, Y et Z tels qu'il existe :

```
pere_de(X, Y).
pere_de(Y, Z).
```

on peut déduire que X est le grand-père de Z.

Nous avons deux faits à vérifier en même temps, avec une variable commune : Y. La règle en PROLOG comporte deux parties : le résultat à déduire et les conditions. Elle s'écrira ici :

```
grand_pere(X, Z) := pere_de(Y, Z), pere_de(X, Y).
```



La règle, en PROLOG, permet d'énoncer une loi générale, qu'il y ait ou non, à un instant donné, des solutions à ces règles. Nous reprenons l'exemple ci-dessus et posons la question :

```
? - grand_pere(X, marcel).
```

On ne peut pas effectuer de recherche directement dans la base de faits puisqu'il n'existe pas de fait `grand_pere`.

PROLOG fonctionne de la façon suivante : il unifie tout d'abord la question avec la tête de la règle `grand_pere( , )`. L'unification réussit et Z est lié à `marcel`. Toute occurrence de Z dans la règle est liée à `marcel`. La règle devient :

```
grand_pere(X, marcel) :- pere_de(Y, marcel), pere_de(X, Y).
```

PROLOG va chercher à démontrer la partie droite de la règle, pas à pas. Il doit trouver un fait ou une autre règle qui s'apparie avec `pere_de(Y, marcel)`. Il trouve `Y=pierre`. Les occurrences de Y sont réécrites en `pierre`. Il traite la deuxième partie de la règle `pere_de(X, pierre)`. et trouve `x=dominique`. La valeur de X est repercutee au niveau de la règle où la variable X était restée libre. La réponse livrée, le système peut poursuivre la recherche d'autres solutions.

Une particularité très intéressante de PROLOG est de pouvoir définir plusieurs clauses avec la même tête. On parle alors d'un paquet de clauses.

Voyons maintenant ce que signifie *unification*, une opération de base en PROLOG.

#### 4.2.2.1.5 Le mécanisme d'unification

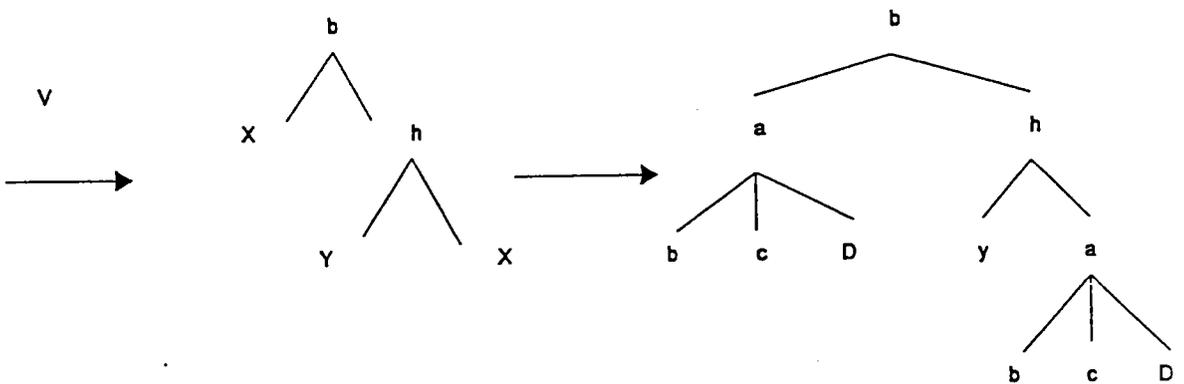
L'unification permet de tester, à l'aide de substitutions si deux arbres ou deux sous-arbres sont identiques. L'unification permet aussi d'accéder aux différents sous-constituants d'un terme par l'emploi de variables qui vont s'unifier avec une partie d'un

terme. Dans un terme, substituer une variable par un terme, c'est remplacer toutes les occurrences de cette variable par ce terme.

$S(X, T)$ . signifie qu'on va remplacer toutes les occurrences de  $X$  par le terme  $T$ , dans un autre terme.

$S(X, a(b, c, D))$  appliquée à  $V = b(X, h(Y, X))$  donnera :

$S(V) = b(a(b, c, D), h(Y, a(b, c, D)))$ .



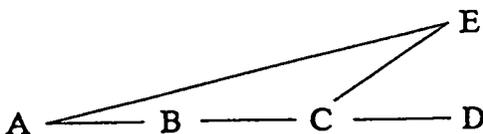
Comme certains autres langages de programmation, PROLOG autorise la récursivité.

#### 4.2.2.1.6 La récursivité

Inspirée des schémas de raisonnement par récurrence, elle constitue un processus fini et limité et repose sur :

- La condition initiale qui est donnée par l'appel récursif au paquet de clauses.
- Le schéma de développement
- La condition d'arrêt qui est exprimée par une autre clause du paquet. Elle doit être examinée avant chaque nouvel appel de la récursivité, ce qui implique qu'elle soit la première clause du paquet.

On peut donner un exemple de récursivité avec la recherche d'un ou plusieurs chemins entre deux nœuds d'un graphe.



Un arc exprime une relation entre deux sommets. On peut représenter un graphe en PROLOG par des faits qui auraient la forme  $arc(X, Y)$ . pour signifier qu'un arc va de  $X$  à  $Y$ .

```
arc(a, b).
arc(b, c).
arc(c, e).
arc(c, d).
arc(a, e).
```

Existe-t-il un chemin du nœud a au nœud d ?

? - chemin(a,d).

Le prédicat chemin(X,Y) sera vrai s'il existe un chemin entre X et Y. Le problème se décompose en deux cas, c'est à dire deux clauses pour le paquet chemin.

Clause 1 : le chemin est élémentaire, il existe un chemin entre deux points si un arc les relie.

Clause 2 : le chemin n'est pas élémentaire. Il y a un chemin de X à Y s'il existe Z, avec un chemin de X à Z et un chemin de Z à Y.

En décomposant le problème en deux parties, nous définissons un schéma récursif. Pour le cas élémentaire, il existe un arc reliant X à Y. Pour le cas général, le problème est résolu au rang inférieur (il y a un chemin de Z à Y).

Les clauses du paquet chemin sont numérotées :

1. chemin(X,Y) :- arc(X,Y).

2. chemin(X,Y) :-  
    arc(Z,Y),  
    chemin(Z,Y).

L'appel se fait par ? - chemin(a,d).

*Première étape :*

Clause 1 : chemin(X,Y) :- arc(X,Y).

Unification : chemin(a,d) :- arc(a,d). ----> échec

Clause 2 : chemin(X,Y) :- arc(X,Z), chemin(Z,Y).

Unification : chemin(a,d) :- arc(a,Z), chemin(Z,d).

La fait arc(a,b) permet d'unifier Z=b.

Il faut démontrer chemin(b,d) pour que chemin(a,d) soit vérifié.

*Deuxième étape :*

Clause 1 : chemin(X,Y) :- arc(X,Y).

Unification : chemin(b,d) :- arc(b,d). ----> échec

Clause 2 : chemin(X,Y) :- arc(X,Z), chemin(Z,Y).

Unification : chemin(b,d) :- arc(b,Z), chemin(Z,d).

Le fait arc(b,c) permet d'unifier Z=c. L'appel récursif est chemin(c,d).

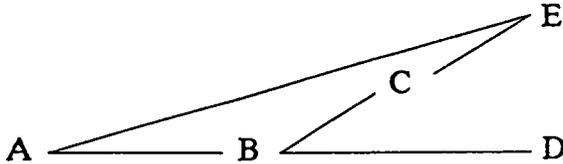
*Troisième étape :*

Clause 1 : chemin(X,Y) :- arc(X,Y).

Unification : chemin(c,d) :- arc(c,d). ----> succès

L'appel à l'étape 3 est démontré. L'appel récursif à l'étape 2 l'est aussi. Il n'y a plus d'autres conditions à démontrer. Il en va de même pour l'étape 1.

L'appel `?- chemin(a, d)` est donc démontré, après l'épuisement de la suite d'appels récurifs. Lorsqu'il y a d'autres solutions, PROLOG les recherche toutes. Si l'on ajoute un fait, par exemple : `arc(b, d)`, on peut aller de `a` à `d` selon deux chemins



(`a-b-c-d` et `a-b-d`).

On peut introduire des variantes dans la façon d'utiliser `chemin`, selon que les arguments sont libres ou instanciés.

Argument 1 libre : `?- chemin(X, d)`. PROLOG nous donnera comme résultat les sommets `X` du graphe à partir desquels il existe un chemin jusqu'à `d`.

Argument 2 libre : `?- chemin(a, X)`. PROLOG nous donnera tous les sommets `X` accessibles à partir de `a`.

Si les deux arguments sont libres, PROLOG donnera tous les chemins du graphe.

En ajoutant un troisième argument `L` à la clause `chemin` pour désigner la longueur du chemin de `X` à `Y` (nombre d'arcs pour aller de `X` à `Y`), le calcul pourra s'effectuer en considérant que pour un chemin élémentaire nous aurons :

```
chemin(X, Y, 1) :- arc(X, Y).
```

et pour un chemin non élémentaire, en posant `L1` pour l'arc `Z-Y` :

```
chemin(X, Y, L) :- arc(X, Z),
                  chemin(Z, Y, L1),
                  L is L1+1.
```

PROLOG gère la récursivité par un mécanisme de pile relativement classique.

#### 4.2.2.1.7 Mécanismes de contrôle

Lorsqu'il exécute une tâche, PROLOG gère tous les mécanismes :

- appels des règles
- unification
- retour arrière

Lorsque l'exécution est stoppée à un instant donné, on dispose de :

- points qui ont été démontrés
- points en cours de démonstration
- points qui restent à démontrer

Dans les deux premiers cas, une partie des règles n'a pas été utilisée. Elles constituent la liste des choix restants. L'état de l'exécution en cours peut évoluer selon deux processus :

- progression classique dans les points à démontrer. Les appels sont exécutés séquentiellement dans l'ordre des clauses du paquet.

- retour en arrière qui intervient après un succès pour rechercher d'autres solutions à un problème ou après un échec sur un point à démontrer.

PROLOG propose deux *prédicats prédéfinis* qui permettent de contrôler la résolution, le prédicat qui provoque l'échec d'une clause et le prédicat qui supprime des choix. Ce dernier permet d'éviter d'effectuer toutes les recherches lorsque l'on veut savoir s'il existe au moins une solution à un problème. Il facilite également l'écriture d'expressions telles que *si... alors... sinon...*

#### 4.2.2.1.8 Les prédicats définis

Ces prédicats sont définis par PROLOG et non par les règles qu'introduit le programmeur. Ils offrent des possibilités que l'on ne pourrait pas obtenir en PROLOG pur et mettent à disposition des outils pratiques (prédicats d'entrée-sortie, introduction de nouvelles clauses, classification des termes, traitement des clauses comme des termes, modification du retour arrière, manipulation des fichiers, observation du déroulement d'un programme...).

#### 4.2.2.1.9 Le retour en arrière et la coupure

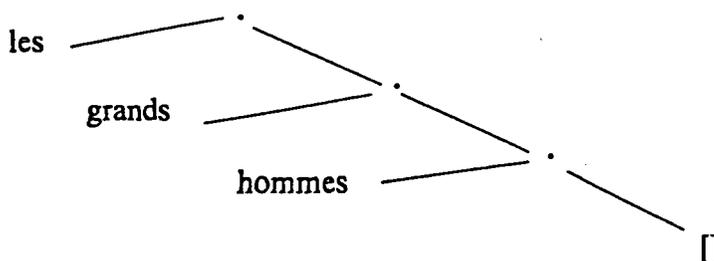
La coupure permet de signaler à PROLOG les choix antérieurs qu'il n'est pas nécessaire de reconsidérer en cas de retour arrière dans l'ensemble des buts déjà satisfaits. Le programme s'exécute plus vite car il ne perd pas de temps à essayer de satisfaire des buts dont on sait par avance qu'ils n'apportent pas de solution. On économise de la mémoire car il devient inutile de stocker des points de retour arrière pour une recherche ultérieure.

Avant de décrire ce qui sert de cadre linguistique pour l'application de PROLOG au traitement du langage naturel, nous devons dire encore quelques mots sur *les listes*.

#### 4.2.2.1.10 Les listes

En PROLOG, on utilise très fréquemment une structure de données particulière : la liste. Elle réunit un ensemble ordonné d'objets (et peut être vide).

Soit la liste [les, grands, hommes]. Les crochets ouvrant et fermant indiquent le début et la fin de la liste. Les éléments sont séparés par une virgule. Une liste est soit une liste vide (sans éléments), soit une structure à deux composants : la tête et la queue. On représente la fin de la liste, selon l'usage, comme une queue qui est la liste vide et s'écrit []. La tête et la queue sont les composants du symbole fonctionnel point qui s'écrit ".". On peut la représenter sous la forme d'un arbre :



La liste représente un ensemble d'objets, sans relations. Au niveau de l'unification, l'opération essentielle est la séparation de la tête et de la queue. Il existe un certain nombre d'opérations de base sur les listes (recherche d'un élément dans une liste, opérations ensemblistes, concaténation).

#### 4.2.2.2 PROLOG et le cadre linguistique

Nous mettrons surtout l'accent sur l'aspect syntaxique de l'analyse, en partant de la théorie transformationnelle. Les grammaires logiques que nous verrons dans le paragraphe 4.2.2.3 s'en sont inspirées. Dans un premier temps, nous rappelons ce que sont les *grammaires formelles* (4.2.2.2.1), nous présentons les notions de *catégories syntaxiques* (4.2.2.2.2) et d'*arbres syntaxiques* (4.2.2.2.3). Ensuite, nous introduirons brièvement la notion de *structure du lexique* (4.2.2.2.4) pour terminer sur *les transformations* (4.2.2.2.5). En conclusion (4.2.2.2.6), nous citons des grammaires récentes et ce qu'elles apportent de nouveau.

Une phrase en langage naturel est composée de structures appelées *syntagmes*, construits autour du nom ou du verbe. Ils sont composés de *mots* que l'on considère ici comme l'élément de base. Les mots et leurs caractéristiques sont rangés dans un *lexique* qui devient essentiel, au fil des recherches en Traitement Automatique des Langues Naturelles. Les grammaires sont plutôt considérées comme des outils destinés à contrôler la cohérence des phrases et à construire une compréhension de l'énoncé. On utilise des systèmes abstraits de règles pour représenter ces syntagmes qui reflètent la structure de la phrase. Les règles de type 2 sont augmentées par l'emploi d'arguments qui leur confère une certaine dépendance vis à vis du contexte (règles de type 1).

##### 4.2.2.2.1 Les grammaires formelles

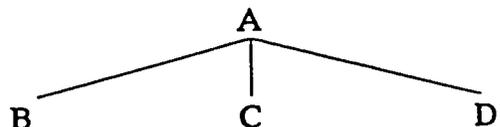
Regroupées en trois classes, les grammaires formelles permettent de décrire les structures qui explicitent les liens et les relations entre les mots qui composent une phrase. Nous avons vu qu'une grammaire de type 2 que l'on appelle également grammaire hors contexte (cf. 1.4.3.1.2) est un quadruplet (V,T,P,S) :

- V est l'ensemble des symboles non-terminaux qui représentent les catégories syntaxiques.
- T est l'ensemble fini d'éléments terminaux qui sont ici les mots de la langue.
- P est un ensemble de règles. Chaque règle a la forme  $\alpha \rightarrow \beta$ ,  $\alpha$  étant un élément de V et  $\beta$  une suite ordonnée d'éléments de  $V \cup T$ .
- S est le symbole initial ou axiome. Dans notre cas, c'est le symbole Phrase (P).

La grammaire permet d'engendrer une phrase (génération de texte). On dit que ses règles sont des règles de réécriture ( $\alpha$  se réécrit en la séquence de symboles  $\beta$ , chaque symbole se réécrivant en d'autres symboles jusqu'à ce qu'ils soient tous des symboles terminaux). Dans une opération inverse, la grammaire permet de reconnaître si une suite de mots forme une phrase acceptable (analyse de phrase).

L'analyse d'une phrase produit une description structurelle qu'il est possible de schématiser par un arbre. La règle suivante signifie qu'une structure A est composée de trois sous-structures B, C et D :

$A \rightarrow B, C, D$



Un arbre représentera la même chose :

Soit une suite de règles dans lesquelles nous allons représenter les symboles terminaux en minuscules et entre crochets, et les symboles non-terminaux en majuscules.

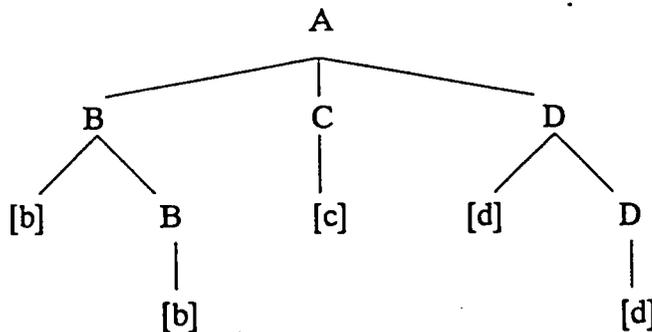
1. A ---> B, C, D
2. B ---> [b]
3. B ---> [b], B
4. C ---> [c]
5. D ---> [d]
6. D ---> [d], D

Ces règles permettent de reconnaître les phrases suivantes :

1. b, b, c, d, d.
2. b, c, d, d, d.

b, c, c, d ne peut être reconnue puisqu'il n'est possible de reconnaître qu'un seul c.

Nous pouvons représenter la structure de la phrase 1 par l'arbre syntaxique suivant :



A est la racine. B, B, C, D et D sont les nœuds. [b], [b], [c], [d] et [d] sont les feuilles.

Les règles B ---> [b], B et D ---> [d], D font appel à elles-mêmes, elles sont récursives.

Afin de préciser la forme et la nature des règles qui sont utilisées pour le traitement de la langue naturelle, nous allons passer en revue les notions qui vont nous permettre de décrire la structure d'une phrase et d'en modéliser la description.

#### 4.2.2.2 Les catégories syntaxiques

Les catégories lexicales de base sont :

- le nom (N) : *chat, souris, Paul...*
- le déterminant (DET) : *le, la, un...*
- l'adjectif (A) : *gros, petit, vert...*
- l'adverbe (ADV) : *hier, très...*
- le verbe (V) : *mange, porte...*
- l'auxiliaire (AUX) : *été, auront...*
- la préposition (PREP) : *à, sur, pour...*
- le pronom (PRO) : *qui, lesquels*

- la conjonction (CONJ) : *mais, ou, car...*

Dans la théorie standard, on définit les catégories à partir des catégories de base évoquées plus haut :

- la phrase (P) : le chat mange la souris...
- Le syntagme nominal (SN) : la plus grande ville, l'homme que je vois...
- le syntagme verbal (SV) : portera, mange la petite souris...
- le syntagme prépositionnel (SP) : sur la table, sous le parapluie...
- le syntagme adjectival (SA) : le plus amusant, très faible...
- le syntagme adverbial (SADV) : de moins en moins souvent...

On va décrire les constituants de la phrase à l'aide de règles (syntagmatiques). Pour dire qu'une phrase P est composée d'un syntagme nominal (SN) et d'un syntagme verbal (SV), on écrit la règle :

P --> SN, SV

On peut trouver de nombreuses possibilités :

SN --> DET, N  
 SN --> DET, N, SP  
 SN --> N, SA

SV --> AUX, V  
 SV --> AUX, V, SN  
 SV --> V, N  
 SV --> V, SP

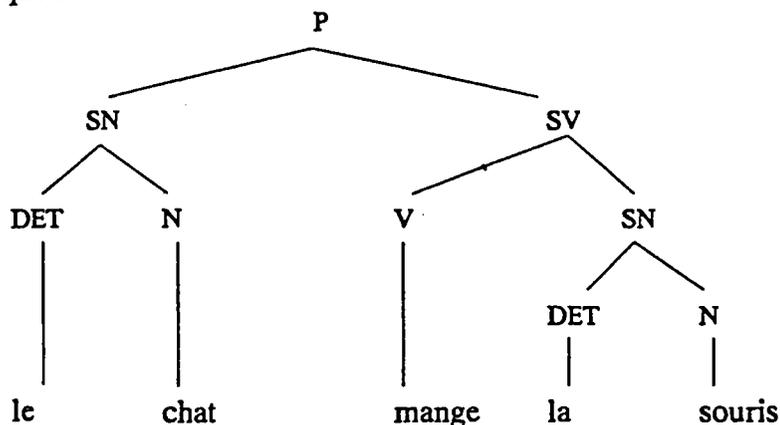
SP --> PREP, SN

La tendance est à la recherche de règles générales pour aboutir plus facilement à une description exhaustive de la langue.

Ce sont les *règles d'insertion lexicale* qui permettent l'insertion de mots appartenant au vocabulaire terminal sous les catégories lexicales comme N, DET, V, A, ADV...

#### 4.2.2.2.3 Les arbres syntaxiques

On dit que la construction d'une phrase est correcte si un ensemble de règles permet de décrire complètement sa structure, que l'on représente souvent sous la forme d'un arbre syntaxique :



Nous avons vu dans le chapitre I qu'une phrase peut avoir plusieurs arbres syntaxiques lorsque sa construction présente des ambiguïtés.

#### 4.2.2.2.4 Les structures du lexique

Le vocabulaire est composé de mots auxquels sont associées des caractéristiques phonologiques, morphologiques, syntaxiques et sémantiques. En ce qui concerne la morphologie et selon les langues, le mot peut prendre différentes formes liées au genre, au nombre, à la personne, au temps, au mode, à la voix... Les rapports entre mots pourront imposer une forme, ce qui conduit à la notion d'accord. Les lexiques contiennent habituellement un ensemble de *règles morphologiques* qui déterminent les formes que prendront les mots. Des *règles de formation* permettront de dériver des mots simples. Il faut y ajouter un traitement important des exceptions.

Nous avons vu que chaque mot appartient à une catégorie syntaxique. Les indications qui en découlent sont insuffisantes pour mener une analyse classique. Aussi doit-on leur adjoindre des sous-catégories en rapport avec l'environnement qu'ils acceptent. On est alors amené à considérer un ensemble de modifications, indispensables pour préciser le sens du mot. Ainsi pour le verbe *monter*, aurons-nous la forme transitive (*monter la garde, monter la tente...*), la forme intransitive (*monter à cheval, monter à l'assaut, la route monte, les blés montent...*) et la forme pronominale (*les frais se montent à cent francs, il se monte en linge...*). Il est impossible de décrire tous les contextes pour chaque entrée du dictionnaire. On cherche par conséquent à regrouper en classes sémantiques représentées par des traits sémantiques "tout ce qui peut modifier". Ces traits varient selon les auteurs car ce domaine de recherche est en pleine évolution. Ils traduisent des relations ou des rôles thématiques tels que *agent, instrument, but, moyen...* Il est de plus fréquent que l'on ajoute à ces rôles des sous-types sémantiques primitifs pour préciser la nature du complément : *humain, objet, abstrait...* Nous verrons dans le paragraphe 4.2.3.3 comment PROLOG les met en œuvre.

Lorsque l'environnement donné d'un mot présente plusieurs variantes, on peut par exemple assurer la permutation de deux compléments et faire l'économie d'une duplication des données de sous-catégorisation grâce à l'emploi de *règles de redondance*. Des *règles de restructuration* redéfinissent l'environnement à la suite d'une transformation (voix active --> voix passive...).

#### 4.2.2.2.5 Les transformations

Les règles de description des structures syntagmatiques donnent la structure de phrases simples et correctes. En simplifiant, des phrases simples peuvent être combinées ou des structures de ces phrases peuvent être déplacées selon des règles précises pour former des phrases plus complexes. Ces mécanismes de construction s'appellent les transformations. Elles complètent le composant génératif dans la théorie transformationnelle standard. Parmi les transformations importantes, le *Qu-mouvement* qui permet de passer de la forme affirmative à la forme interrogative et le *SN-mouvement* qui assure le déplacement du syntagme nominal comme dans le passage de la forme active à la forme passive.

#### 4.2.2.2.6 Conclusion

De nombreux modèles informatiques s'efforcent de manipuler des règles de transformation (composant transformationnel) et des règles qui décrivent les structures de base (composant génératif). Ils apparaissent souvent comme des variantes de notation de la théorie standard, mieux adaptés aux modes d'expression utilisés en informatique et orientés vers une certaine simplification.

Dans les grammaires à structure syntagmatique généralisée (cf. 1.4.2.4.3) avec l'emploi de catégories dérivées et dans les grammaires lexicales fonctionnelles (cf. 1.4.3.6.2) avec l'emploi de méta-variables, une notation supplémentaire permet d'incorporer le composant transformationnel dans le composant de base. Parmi les grammaires qui accordent encore plus d'importance au lexique, on peut citer les grammaires syntagmatiques dirigées par la tête (HPSG)<sup>1</sup>, les grammaires d'adjonction d'arbre (TAG)<sup>2</sup> et les grammaires d'unification (cf. 1.4.3.6.1).

Les grammaires logiques (cf. 1.5.8) que nous avons plutôt considérées comme des outils que comme des modèles parce qu'elles sont issues des techniques de programmation en logique, font appel à des formalismes qui s'appliquent à des descriptions linguistiques très proches de la théorie transformationnelle : les grammaires de métamorphose (cf. 1.5.8.1), les grammaires à clauses définies d'implantation plus simple, les grammaires d'extrapolation<sup>3</sup> traitant le mouvement des constituants vers la gauche et les grammaires modifiant la structure<sup>4</sup>.

#### 4.2.2.3 PROLOG et l'analyse automatique

Comme nous l'avons dit dans l'introduction du paragraphe 4.2.2, PROLOG est une succession d'énoncés logiques traduisant des relations entre eux. Exécuter un programme revient à prouver la véracité d'un nouvel énoncé à partir de ceux dont on dispose.

Prenons l'exemple classique de la structure ( $p$ ) qui représente une phrase composée d'un syntagme nominal ( $sn$ ) et d'un syntagme verbal ( $sv$ ). Le syntagme nominal ( $sn$ ) pourrait être composé d'un déterminant ( $det$ ) et d'un nom ( $n$ ), le syntagme verbal d'un verbe ( $v$ ) ou d'un verbe ( $v$ ) et d'un syntagme nominal ( $sn$ ). Ceci donnerait la grammaire en forme logique :

$$\begin{aligned} p &= sn \wedge sv \\ sn &= det \wedge n \\ sv &= v \vee (v \wedge sn) \end{aligned}$$

Nous cherchons à savoir si la phrase *le chat mange la souris* est correcte. Il faut donc montrer que  $p$  est vrai. Pour cela, il faut identifier un  $sn$  et un  $sv$ . Un  $sn$  est composé d'un déterminant  $det$  et d'un nom  $n$  : cela correspond bien à *le chat*. Le reste de la phrase doit correspondre à un  $sv$  composé d'un verbe  $v$  (*mange*) et d'un syntagme nominal  $sn$  (*la souris*). On prouve ainsi que la structure de la phrase correspond bien à la grammaire définie plus haut.

La phrase est considérée comme une suite de mots que PROLOG range dans une liste :

[*le, chat, mange, la, souris*].

Codons maintenant les règles de grammaire :

(1) C. POLLARD : *Generalized Phrase structure grammars, head grammars and natural languages*, Cambridge university press, New York, 1987

(2) A. JOSHI : "Tree adjoining grammars and their relevance to generation" in *Natural language generation*, Kempen, Nijhoff Publishers, Dordrecht (Actes NATO advances research workshop on "Natural language generation"), Nimègue, 1986, pp. 233-252

(3) F. PÉREIRA : "Extrapolation grammars", *Computational linguistics*, vol. 7, 1981

(4) V. DAHL, McCORD : "Treating coordination in logic grammars", *American journal of computational linguistics*, 7,1, 1983, pp. 32-40

- *Un symbole non-terminal* sera représenté par un prédicat à deux arguments (le premier indique la chaîne d'entrée, le second indique ce qu'il reste de la chaîne après son traitement par le prédicat). Ce type de représentation s'appelle *une liste de différence*. Sa conception remonte aux premiers travaux effectués sur les grammaires de métamorphose par COLMERAUER<sup>1</sup>.

Nous définissons alors le syntagme verbal de la façon suivante :

sv(LO, L) :- v(LO, L).  
sv(LO, L) :- v(LO, L1), sn(L1, L).

Les paramètres utilisés en argument permettent de connecter les différentes listes. Pour la première règle, le syntagme n'est composé que d'un verbe, les listes du syntagme sont donc les mêmes que celles du verbe. Pour la deuxième règle, la liste d'entrée du verbe est la liste d'entrée du syntagme, mais sa liste de sortie est transmise en liste d'entrée au syntagme nominal pour analyse. Sa liste de sortie sera la liste de sortie du syntagme.

- Pour vérifier la présence d'un *symbole terminal*, on écrit une règle qui contrôle la présence du terminal au début de la chaîne et donne la suite de la chaîne en sortie. En représentant une liste composée d'une tête (le mot en début de chaîne) et d'une queue (la suite de la chaîne), on a [ M o t | L ]. La règle sera :

terminal(mot, [mot|L], L).

Pour vérifier qu'un nom peut être *souris*, on écrit :

n(LO, L) :- terminal(souris, LO, L).

Nous reprenons l'exemple de A. GAL, G. LAPALME et P. SAINT-DIZIER<sup>2</sup> pour écrire en PROLOG la grammaire correspondant à l'encadré de la page précédente (grammaire sous forme logique).

p(LO, L) :- sn(LO, L1), sv(L1, L).

sn(LO, L) :- det(LO, L1), n(L1, L).

sv(LO, L) :- v(LO, L).

sv(LO, L) :- v(LO, L1), sn(L1, L).

det(LO, L) :- terminal(le, LO, L).

det(LO, L) :- terminal(la, LO, L).

n(LO, L) :- terminal(chat, LO, L).

n(LO, L) :- terminal(souris, LO, L).

v(LO, L) :- terminal(mange, LO, L).

v(LO, L) :- terminal(court, LO, L).

terminal(mot, [mot|L], L).

(1) A. COLMERAUER a créé PROLOG en 1970.

(2) A. GAL, G. LAPALME et P. SAINT-DIZIER : *PROLOG pour l'analyse automatique du langage naturel*, Ed. Eyrolles, Paris, 1989, pp. 48-78

Si nous posons le but 1 :

? - p([le, chat, mange, la, souris], []).

PROLOG répond :

yes

Si nous posons le but 2 :

? - p([le, chat, mange, le, souris], []).

PROLOG répond :

yes

Si nous posons le but 3 :

? - p([le, chat, court, la, souris], []).

PROLOG répond :

yes

Nous voyons à propos des buts 2 et 3 que notre grammaire accepte une phrase qui a une structure correcte mais ne prévoit pas d'autres contraintes comme l'accord du genre entre l'article et le nom (but 2), et la présence ou l'absence d'un complément d'objet selon que le verbe est transitif ou intransitif (but 3). Pour contrôler ces contraintes, il va falloir coder l'information sur le genre des articles et des noms, ainsi que sur la transitivité des verbes. Nous allons la coder au niveau de la grammaire.

```
p(LO,L) :- sn(LO,L1), sv(L1,L).
```

```
sn(LO,L) :- det(genre,LO,L1), n(genre,L1,L).
```

```
sv(LO,L) :- v(_,LO,L).
```

```
sv(LO,L) :- v(transitif,LO,L1), sn(L1,L).
```

```
det(masculin,LO,L) :- terminal(le,LO,L).
```

```
det(feminin,LO,L) :- terminal(la,LO,L).
```

```
n(masculin,LO,L) :- terminal(chat,LO,L).
```

```
n(feminin,LO,L) :- terminal(souris,LO,L).
```

```
v(transitif,LO,L) :- terminal(mange,LO,L).
```

```
v(intransitif,LO,L) :- terminal(court,LO,L).
```

```
terminal(mot,[mot|L],L).
```

L'information *masculin, féminin, transitif, intransitif* est introduite en premier paramètre de *det, n* et *v*. La mise en correspondance est lancée par *sn* qui force le déterminant et le nom à avoir le même genre et par *sv* qui vérifie la transitivité. La première règle concernant le syntagme verbal accepte un verbe transitif ou intransitif sans complément

avec l'emploi d'une variable muette notée ( $\_$ ). La deuxième règle pour *sv* exige un verbe transitif. Nous obtenons maintenant :

? - p([la, chat, mange, la, souris], []).

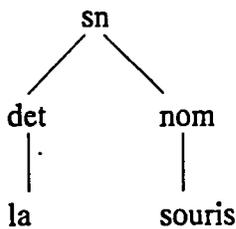
no

? - p([le, chat, court, la, souris], []).

no

Pour l'instant, nous ne faisons que vérifier si une phrase est conforme à une structure. Il serait intéressant de voir comment on peut obtenir la structure de dépendance entre les constituants de notre phrase d'entrée. PROLOG permet de construire cette structure de façon incrémentielle, grâce à une procédure d'unification.

En PROLOG on peut représenter facilement une structure arborescente avec un *foncteur* qui indique la valeur du nœud et dont les paramètres correspondent aux branches issues du nœud.

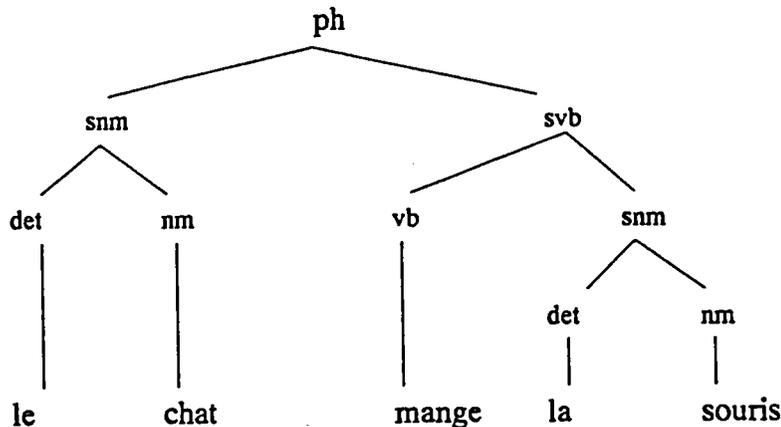


Les termes composés sont notés par un *foncteur* et des *paramètres* qui sont eux-mêmes des termes :

ex. : sn(det(la), nom(souris))

Ils représentent une notation pour les arbres. Le *foncteur* étiquette la racine. Les *paramètres* étiquettent les branches.

Soit l'arbre :



Sa représentation sera :

```

ph( snm(det(le),
        nm(chat)),
    svb(vb(mange),
        snm( det(la),
              nm(souris))))
  
```

C'est un paramètre qui porte l'information structurale et permet de construire l'arbre. Il suffit de donner la forme du foncteur pour qu'un processus d'unification le complète

ou contrôle son adéquation. On obtient la structure d'une phrase avec le foncteur *ph* et en paramètre, les structures du syntagme nominal et du syntagme verbal :

Nous ferons la même chose pour les syntagmes nominal et verbal, ainsi que pour les terminaux. Soit la nouvelle grammaire, constituée uniquement de prédicats d'analyse :

```
p(ph(SN_Struct,SV_Struct,LO,L) :-
sn(SN_Struct,LO,L1),sv(SV_Struct,L1,L)).
```

```
sn(snm(Det_Struct,N_Struct),LO,L) :-
    det(Det_Struct,Genre,LO,L1), n(N_Struct,Genre,L1,L).
```

```
sv(svb(vb(Mot)),LO,L) :- v(vb(Mot,_),_,LO,L).
sv(svb(vb(Mot),SN_Struct),LO,L) :-
v(vb(Mot),transitif,LO,L1),sn(SN_Struct,L1,L).
```

```
det(dt(le),masculin,LO,L) :- terminal(le,LO,L).
det(dt(la),feminin,LO,L) :- terminal(la,LO,L).
```

```
n(nm(chat),masculin,LO,L) :- terminal(chat,LO,L).
n(nm(souris),feminin,LO,L) :- terminal(souris,LO,L).
```

```
v(vb(mange),transitif,LO,L) :- terminal(mange,LO,L).
v(vb(court),intransitif,LO,L) :- terminal(court,LO,L).
```

```
terminal(Mot,[Mot|L],L).
```

Il est possible d'ajouter de nombreuses informations en utilisant d'autres prédicats PROLOG et en introduisant de nouveaux paramètres qui véhiculeront de l'information entre les nœuds de l'arbre. On peut alors vérifier d'autres conditions et introduire par exemple un traitement sémantique.

Il reste à contrôler le transfert des listes d'entrée et de sortie entre les prédicats d'une même règle. Ceci est réalisé automatiquement pour les grammaires de métamorphose et les grammaires à clauses définies (DCG). Dans la nouvelle grammaire écrite avec le formalisme des DCG, les parties gauches sont liées aux parties droites par (-->). Le système gère lui-même le passage de la chaîne entre les prédicats. Dans PROLOG, le système transforme la règle, à la lecture, en ajoutant deux paramètres aux prédicats d'analyse. Les prédicats qui ne doivent pas être modifiés sont placés entre accolades. Les terminaux sont entre crochets et la règle `terminal` n'est plus indispensable car prédéfinie par le système. La grammaire DCG s'écrit avec des règles exprimées par des prédicats. Les paramètres assurent le passage de l'information d'un prédicat à l'autre. On pourra ajouter des prédicats PROLOG en les plaçant entre accolades.

La version DCG de notre grammaire sera :

```

p(ph(SN_struct,SV_Struct) --->
sn(SN_Struct),sv(SV_Struct).

sn(snm(Det_Struct,N_Struct)) --->
    det(Det_Struct,Genre), n(N_Struct,Genre).

sv(svb(vb(Mot))) ---> v(vb(Mot,_),_).
sv(svb(vb(Mot),SN_Struct)) --->
v(vb(Mot),transitif),sn(SN_Struct).

det(dt(le),masculin) ---> [le].
det(dt(la),feminin) ---> [la].

n(nm(chat),masculin) ---> [chat].
n(nm(souris),feminin) --> [souris].

v(vb(mange),transitif) ---> [mange].
v(vb(court),intransitif) ---> [court].

```

Jusqu'ici, les exemples concernent des phrases dont les constituants sont atomiques. Pour l'analyse automatique d'un domaine plus vaste et la construction d'un système évolutif, il convient de dissocier le traitement de la morphologie et celui de la structure de la phrase. On pourra alors enrichir le dictionnaire séparément et considérer le volume des entrées. Une bonne structuration du dictionnaire évitera de conserver toutes les variantes d'une forme (*petit, petite, petits, petites* ou *ai, as, a, avons...*) en lui associant des informations invariantes et des règles de conjugaison, déclinaison ou lemmatisation.

L'analyse de phrases plus complexes telles que les relatives ou les interrogatives fait appel à la notion de *trace*. On part du principe, en effet, qu'une relative est comme une principale dont quelques informations auraient disparu, remplacées par des *traces* qui conservent toutes les propriétés des objets qu'elles remplacent. Pour une analyse correcte, on gardera cette information pour la propager entre les syntagmes, en ajoutant le paramètre *trace* en deuxième argument des prédicats d'analyse. On peut utiliser cette information pour vérifier par exemple l'accord des participes passés.

Nous avons vu que le processus d'unification était très puissant en PROLOG. Dans certains cas, cependant, les structures à unifier n'ont pas la même forme, c'est à dire le même nombre de paramètres et pas dans le même ordre. Si nous reprenons le cas du dictionnaire dans le lequel nous rentrons de nombreux traits, il nous faudra prévoir des valeurs pour des traits non pertinents. L'idéal serait de ne conserver que les traits pertinents et dans un ordre quelconque. Une modification de l'algorithme d'unification permet justement d'"unir" des traits : si un trait se trouve dans deux structures, les valeurs associées devront s'unifier. S'il n'apparaît que dans une, il fera partie de la structure résultante.

En résumé, nous dirons qu'avec PROLOG l'utilisateur dispose d'un formalisme évolué pour :

- décrire et structurer les objets et les relations d'un domaine spécifique,
- en exprimer les connaissances sous forme de faits et de règles,
- poser des questions et exécuter des actions.

PROLOG peut être porté sur tous les systèmes d'exploitation et sur toutes les machines. Si on le compare avec le LISP, un autre langage utilisé en Intelligence Artificielle, il a de gros atouts. Le LISP a été développé à l'époque où l'algorithme était roi.

Il "consomme" beaucoup de logique procédurale. La solution doit avoir été trouvée au préalable. LISP est considéré comme "l'assembleur de l'I.A." parce que justement il suit pas à pas tout raisonnement dans une structure de listes. Pour concevoir un logiciel en LISP, il faut spécifier une série de détails souvent inutiles au regard de l'application. Rien n'est laissé au hasard ou de manière implicite, ce qui peut être un handicap lors de l'élaboration de logiciels complexes. Du reste, les adeptes de LISP utilisent PROLOG en complément car il manquait à LISP un mécanisme de déduction intégré au sein du langage. On reconnaît à PROLOG les vertus de son système d'unification et son non-déterminisme. Parmi ses avantages par rapport à d'autres langages évolués : l'identité de forme syntaxique entre les données et les programmes de PROLOG en permet une manipulation simple, les paramètres d'entrée-sortie peuvent être incomplètement déterminés pour être affectés ailleurs dans d'autres appels, les appels de règles peuvent générer, à partir de retour arrière des itérations de haut niveau, aucune structure d'enregistrement n'est imposée aussi bien au niveau type que longueur de champ d'enregistrement, PROLOG enfin supprime les boucles, étiquettes, instructions et pointeurs au profit d'une définition du problème qui contient en soi sa propre recherche de la solution.

#### 4.2.2.4 La réalisation de l'ACI

##### 4.2.2.4.1 Matériel utilisé

La mise au point informatique a été réalisée à partir d'un télétype ITT model 43 associé à un modem Anderson Jacobson, en 300 Bauds, sur les ordinateurs du CIRCE, directement ou via TRANSPAC. Les programmes écrits en PL1 Optimizer ont tourné sur IBM 370/168, AMDAHL 470/V7 (5,4 millions d'instructions par seconde) et, depuis 1983, sur NAS 9080 (20 millions d'instructions par seconde), la taille mémoire maximum disponible étant d'environ 5500 K.

##### 4.2.2.4.2 Données

En l'absence de grands dictionnaires, les données relatives à une langue (source ou cible) n'occupent jamais plus de 128 K, ce qui limite le temps calcul et la présence de mémoires de masse. Elles sont rangées dans des fichiers séquentiels ou partitionnés qui n'excèdent jamais 1000 entrées. Le fonctionnement automatique du système impose une gestion automatique de tous les fichiers d'entrée et de sortie. C'est la raison pour laquelle les premières lignes indiquent toujours le nombre d'enregistrements, le LRECL (longueur de l'enregistrement logique) et le BLKSIZE (taille du bloc). Les enregistrements ont un RECFM (format et caractéristiques des enregistrements) VB, c'est à dire qu'ils sont de longueur variable et bloqués.

```
listds .morphall
ASU4416.DIRON.MORPHAL1
--RECFM=LRECL=BLKSIZE=DSORG
  VB   30   6334   PO
--VOLUMES--
  RES305
READY

e .morphall(FAIBLE) data
E
1 10 80
00010 *.MORPHAL1(FAIBLE)
00020 811
00030 0030
00040 6334
00050 100ACHT
00060 100A+CHT
00070 100A+CHZ
00080 100ACKER
end s
READY
```

Le PL1 permet de les utiliser avec une extrême rapidité grâce aux variables et aux structures basées accompagnées de l'attribut REFER. Cette technique vaut pour les fichiers de traitement que les programmes taillent sur mesure en calculant les paramètres cités plus haut et en les insérant en début de fichier. Cette démarche permettra d'utiliser un fichier sortie (résultats) en entrée pour l'exécution d'un autre programme, de façon automatique. C'est ainsi qu'une modification des données n'entraîne qu'une réécriture des premiers enregistrements et aucune intervention sur les programmes, ces derniers n'étant retouchés qu'à l'apparition de problèmes imprévus.

#### 4.2.2.4.3 Programmes

Nous utilisons le langage PL1 (4ème version). Plusieurs éléments ont guidé notre choix :

- C'est un des rares langages qui permettent de donner des caractéristiques de fichier et même de les calculer, contrairement à d'autres où l'on ne peut agir qu'au niveau du langage de gestion.
- A la maîtrise de la création des fichiers de sortie s'ajoute la maîtrise des chaînes de caractères. S'il n'a pas été conçu pour la manipulation de textes en langue naturelle, PL1 offre la possibilité de manipuler des chaînes de 32767 caractères et les opérations sur ce type de chaîne sont suffisamment développées pour que ce domaine puisse être un de ses domaines d'application. Ceci veut dire que nous n'en utilisons qu'une partie réduite :
  - variables statiques, automatiques, contrôlées;
  - variable basée (avec la version 4 de PL1), pointeurs, variables pointées, tableaux et structures : structures basées avec option REFER, variables PIC
  - fonctions de manipulation de chaîne
  - fichiers STREAM RECORD avec option ENV, pour imposer les caractéristiques calculées aux DATASETS correspondants
  - bloc, procédure et fonction récursive ou non, fonction incorporée, procédure paramétrée avec des pointeurs
  - macro-instructions

Les programmes sont courts (400 lignes en moyenne). On peut en contrôler facilement le fonctionnement et ils occupent moins de place. Les lectures de fichier se font par variables basées, ce qui nous ramène presque au niveau du langage machine.

#### 4.2.2.4.4 Procédures

Afin de disposer d'une grande souplesse dans l'enchaînement des programmes nous n'utilisons pas le JCL (langage de contrôle des travaux) dont la syntaxe nous semble assez contraignante, mais la procédure.

La procédure de commandes est un fichier dont les enregistrements sont composés de commandes, sous-commandes ou d'instructions de contrôle. Lors de son exécution, chaque enregistrement correspond à l'image d'une ligne de terminal. La ligne est alors traitée comme si elle avait été entrée directement. Les commandes sont liées au système d'exploitation (TSO sous-système de MVS) :

- allocation des fichiers d'entrée et de sortie (mise en place des fichiers physiques par DATASET, des fichiers logiques par FILE)
- appel des modules chargeables
- destruction des fichiers
- mesure du temps calcul
- désallocation

- vérification de l'existence des fichiers et tassement lorsque nécessaire.

Cette très grande souplesse, alliée à la simplicité d'écriture, permet de contrôler les temps de calcul pour chaque programme et d'en enchaîner une infinité, en détruisant les fichiers temporaires qui ne sont plus utiles.

### 4.3 La structure de l'ACI

#### 4.3.1 La mise en forme

##### 4.3.1.1 Textes

Nous disposons de textes d'essai en allemand, néerlandais, luxembourgeois et anglais.

##### Allemand

1 - *Umweltplanung in der Industriegesellschaft: Lösungen und ihre Probleme.* de Konrad STAHL et Gerhard CURDES, ROWOHLT Taschenbuch (1970), 28 000 mots.

2 - *Leistungsreaktortypen:* document AG.TELEFUNKEN (50 000 mots)

3 - *Differenzverstärker:* document DRET (9 000 mots)

4 - *Hirnfunktion und Hirndurchblutung:* de Niels LASSEN, David INGVAR et Eiril SKINHØT: (4 800 mots)

5 - *Brennstoffzellenkraftwerke* de Arnold P. FICKETT (3 500 mots)

##### Néerlandais

- *Kernenergie in de lage landen* de J.A. GOEDKOOP: (45 000 mots)

##### Luxembourgeois

- *D'Reform vun der Lëtzebuenger Ekonomie* de J.P. SCHNEIDER: (50 000 mots)

##### Anglais

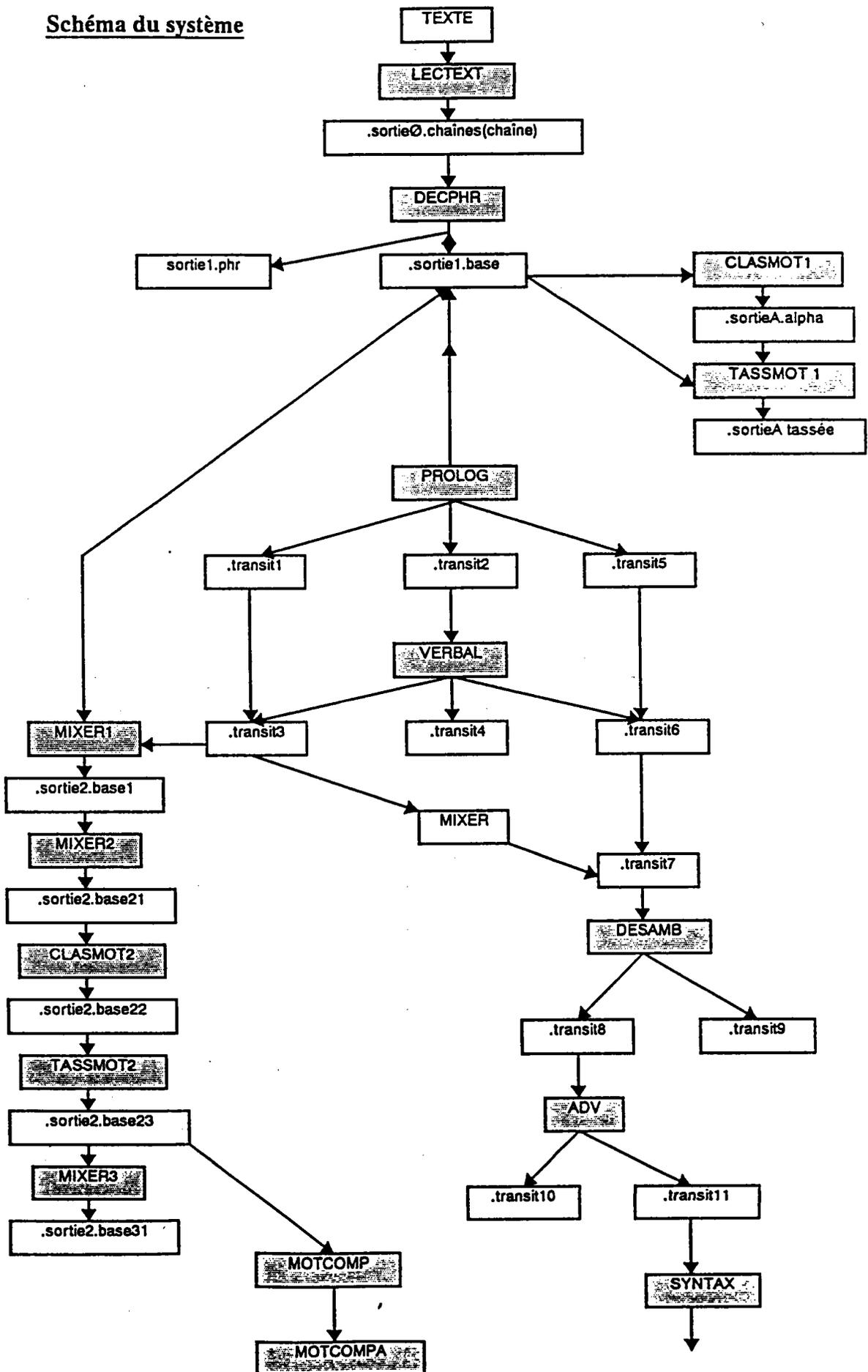
- *Fuel-Cell Power Plants* de Arnold FICKETT : (3 800 mots)

#### 4.3.1.2 Schéma d'ensemble

Les exemples que nous proposerons pour l'allemand seront tous extraits du même texte, dans un souci de cohérence, sauf en de rares occasions où nous serons obligé de faire appel à d'autres sources. Le choix du texte n°5 nous semble s'imposer, tant il est riche en difficultés de toutes sortes.

Avant d'examiner le niveau Ø dans le détail, nous schématiserons dans la page qui suit, l'enchaînement des divers modules (en majuscule) avec les fichiers d'entrée et de sortie (en minuscule). Les ensembles importants, pour plus de clarté, figurent dans des cadres à trame grisée. Ils seront "décortiqués" par la suite, VERBAL dans le chapitre 4.3.6, MOTCOMP et MOTCOMPA dans les chapitre 4.4.2, 4.4.3 et 4.4.4, SYNTAX enfin dans le chapitre 4.5.

Schéma du système



### 4.3.1.3 Procédure générale

Comme l'indique le tableau de la page 178, le niveau 0 correspond à la préparation du texte (LECTEXT et DECPHR), au codage des mots-outils, des nombres en lettres ou en chiffres, des symboles et des noms autres qu'en début de phrase (PROLOG), à l'élaboration du lexique tassé du texte proposé en option (CLASMOT1 et TASSMOT1), à l'analyse du verbe (FILTRAL, VERBAL1, VERBAL2), à l'étude des adjectifs/adverbes (ADV) et à la désambiguïsation de la majuscule en début de phrase (DESAMB).

La procédure qui assure le fonctionnement de ce niveau 0 s'appelle PDROC(CRJF). Elle contient plus de 600 lignes et enchaîne les programmes mentionnés p. 178, présents sous forme de module chargeable. Le fichier de départ est un des membres du fichier partitionné .TEXALL, .TEXOLL ou .TEXANG. Le fichier d'arrivée est .TEXALL1.SORTIE2.BASE23, .TEXOLL1.SORTIE2.BASE23 ... (selon la langue choisie), pour le traitement ultérieur des mots composés, .TRANSIT11 pour l'analyse syntaxique.

#### .PDPROC(CRJF) :

De manière forfaitaire, le système alloue quinze fichiers, quelle que soit la longueur du texte :

```

00100 PROC 0
00200 WRITE
00300 D'INTERPRETATION
00400 WRITE
00500 *****
00600 CONTROL END(STOP)
00700 WRITE FONCTIONNEMENT DU SYSTEME CRJF EN DATE DU &SYSDATE, &SYSTEME
00800 WRITE CONTROLE DU TEMPS DE SESSION ET DU TEMPS CPU :
00900 TIME
01000 WRITE CHOISIR LA LANGUE-SOURCE : ECRIRE ALL POUR ALLEMAND, ...

05400 WRITE AUCUNE ANALYSE N'A ETE FAITE SUR CE TEXTE. LE SYSTEME SE PLACE +
05500 DONC AU NIVEAU 0
05600 GOTO ETIQ15
05700 STOP
05800 ETIQ9: DO
05900 WRITE LA LECTURE INITIALE DU TEXTE A DEJA ETE FAITE. LE SYSTEME SE +
06000 PLACE DONC AU NIVEAU 1
06100 GOTO ETIQ16
06200 STOP
06300 ETIQ10: DO
06400 WRITE LA MISE EN FICHIER DU TEXTE ET SON ANALYSE 'VERBE' SONT +
06500 TERMINEES. LE SYSTEME SE PLACE DONC AU NIVEAU 2
06600 GOTO ETIQ16A
06700 STOP
06800 ETIQ11: DO
06900 WRITE LES OPERATIONS DU NIVEAU 2 SONT TERMINEES. LE SYSTEME SE PLACE +
07000 AU NIVEAU 3
07100 GOTO ETIQ18
07200 STOP
07300 ETIQ15: DO
07400 FREEALL
07500 WRITE ALLOCATION INITIALE DES FICHIERS
07600 LISTA STATUS
07700 WRITE **FONCTIONNEMENT DE LA LECTURE INITIALE
07800 WRITE ALLOCATION DU FICHIER-ENTREE ET DES 15 FICHIERS-SORTIE -
07900 FORFAITAIRES

```

Il appelle les programmes responsables de la mise en forme du texte :

```

08000 ALLOC DATASET(.TEX&A.( &B.)) FILE (ENTREE)
08100 WRITE ALLOCATION DE DATASET(.TEX&A.( &B.)) SUR FILE (ENTREE)
08200 ALLOC DATASET(. &B..SORTIE0.CHAINES) NEW DIR(5) SPACE (4,1) TRACKS
08300 SET N=1

08400 DO WHILE &N<16
08500 ALLOC DATASET(. &B..SORTIE0.CHAINES(CHAINEN.)) FILE (SORTIE&N.)
08600 WRITE ALLOCATION DE DATASET(. &B..SORTIE0.CHAINES(CHAINEN.)) SUR FILE
08700 (SORTIE&N.)
08800 SET N=&N+1
08900 STOP
09000 CALL .CHARGE(LECTEXT)
09100 FREEALL
09200 WRITE CHAINES DU TEXTE ETUDIEES :
09300 LISTH . &B..SORTIE0.CHAINES
09400 ZPRESSION . &B..SORTIE0.CHAINES
09500 SPACE . &B..SORTIE0.CHAINES
09600 GOTO ETIQ16
09700 STOP
09800 ETIQ16: DO
09900 WRITE **LECTURE INITIALE TERMINEE
10000 ALLOC DATASET(. &B..SORTIE0.CHAINES(CHAINEN.)) FILE (ENTREE)
10100 ALLOC DATASET(. &B..SORTIE1.BASE) NEW SPACE(4,1) TRACKS +
10200 FILE (SORTIE1)
10300 ALLOC DATASET(. &B..SORTIE1.PHR) NEW SPACE(4,1) TRACKS FILE (SORTIE2)
10400 CALL .CHARGE(DECPHR)
10500 RLSE . &B..SORTIE1.BASE
10600 SPACE . &B..SORTIE1.BASE
10700 RLSE . &B..SORTIE1.PHR
10800 SPACE . &B..SORTIE1.PHR
10900 FREEALL

```

Intervient ensuite le début du traitement du verbe :

```

11000 ALLOC DATASET(.VERBE(ENDUNG)) FILE (PAENDUN)
11100 ALLOC DATASET(.VERBE(VERB)) FILE (PAVER)
11200 ALLOC DATASET(.VERBE(NOVERB)) FILE (PANOVER)
11300 ALLOC DATASET(.VERBE(PREFI)) FILE (PAPREFI)
11400 ALLOC DATASET(.VERBE(KOLL)) FILE (PAKOLL)
11500 ALLOC DATASET(.VERBE(FAIBLE)) FILE (PAFAIBL)
11600 ALLOC DATASET(.VERBE(FORT)) FILE (PAFORTS)
11700 ALLOC DATASET(.TEX&A.( &B.)) FILE (RALLE)
11800 ALLOC DATASET(. &B..SORTIE1.VERBE) NEW SPACE (3,1) TRACKS -
11900 FILE (SORTIE)
12000 CALL .CHARGE(VERBAL)
12100 RLSE . &B..SORTIE1.VERBE
12200 SPACE . &B..SORTIE1.VERBE
12300 FREEALL
12400 GOTO ETIQ16A
12500 STOP
12600 ETIQ16A: DO
12700 WRITE ** VOULEZ-VOUS EDITER UN LEXIQUE ALPHABETIQUE ET UN LEXIQUE +
12800 TASSE (DE FACON SPECIALE) DES MOTS DE CE TEXTE ?
12900 WRITE ** ATTENTION...CES LEXIQUES NE SERONT PAS UTILISES PAR LE +
13000 SYSTEME ET, DE PLUS, COUTENT ASSEZ CHER.
13100 WRITE ** REpondre PAR 'OUI' OU PAR 'NON' (SANS METTRE DE QUOTES).

```

Programmes de tri et de tassement pour l'utilisateur qui demande un lexique :

```

14000 TIME
14100 CALL .CHARGE(CLASMOT1)
14200 FREEALL
14300 ALLOC DATASET(.&B..SORTIE1.BASE) FILE (ENTREE1)
14400 ALLOC DATASET(.&B..SORTIEA.ALPHA) FILE (ENTREE2)
14500 ALLOC DATASET(.&B..SORTIEA.TASSEE) NEW SPACE (4,1) TRACKS
14600     FILE (SORTIE)
14700 CALL .CHARGE(TASSMOT1)
14800 TIME

```

Programmes qui préparent le fichier d'entrée du module MOTS COMPOSES :

```

16500 CALL .CHARGE(MIXER1)
16600 RLSE .&B..SORTIE2.BASE1
16700 SPACE .&B..SORTIE2.BASE1
16800 FREE DATASET(.&B..SORTIE2.BASE1)
16900 ALLOC DATASET(.&B..SORTIE2.BASE21) NEW SPACE (1,1) TRACKS +
17000     FILE (SORTIE2)
17100 CALL .CHARGE(MIXER2)
17200 RLSE .&B..SORTIE2.BASE21
17300 SPACE .&B..SORTIE2.BASE21
17400 FREE DATASET(.&B..SORTIE2.BASE21)
17500 ALLOC DATASET(.&B..SORTIE2.BASE21) FILE (ENTREE6)
17600 ALLOC DATASET(.&B..SORTIE2.BASE22) NEW SPACE (2,1) TRACKS +
17700     FILE (SORTIE4)
17800 CALL .CHARGE(CLASMOT2)
17900 FREE DATASET(.&B..SORTIE2.BASE22)
18000 ALLOC DATASET(.&B..SORTIE2.BASE22) FILE (ENTREE7)
18100 ALLOC DATASET(.&B..SORTIE2.BASE23) NEW SPACE (2,1) TRACKS +
18200     FILE (SORTIE5)
18300 CALL .CHARGE(TASSMOT2)
18400 RLSE .&B..SORTIE2.BASE22
18500 RLSE .&B..SORTIE2.BASE23
18600 SPACE .&B..SORTIE2.BASE22
18700 SPACE .&B..SORTIE2.BASE23
18800 ALLOC DATASET(.&B..SORTIE2.BASE31) NEW SPACE (2,1) TRACKS +
18900     FILE (SORTIE3)
19000 CALL .CHARGE(MIXER3)

```

#### 4.3.1.4 LECTEXT, DECPHR

Sans poursuivre l'examen de la procédure et en laissant de côté pour l'instant ce qui concerne PROLOG, VERBAL, ADV et DESAMB nous allons observer les transformations successives que subit la forme du texte à l'aide de quelques exemples suivis d'un rapide commentaire.

Pour tenir compte de toutes les informations que nous apporte le texte, nous transcrivons les éléments que l'oeil transmet au cerveau comme étant différents de l'écriture, et uniquement ceux-là. Il n'y a pas de "prédiction". Le texte entré en machine est la réplique de l'original, aux éléments de simulation près.

Texte allemand étudié, sous sa forme originale :

## BRENNSTOFFZELLEN- KRAFTWERKE

Von Arnold P. Fickett

Im nächsten Jahrzehnt wird man von einem Elektrizitätswerk fordern, daß es einen hohen Wirkungsgrad hat, möglichst wenig Schmutz emittiert, keinen Lärm verursacht und rasch zu installieren ist. Ein vielversprechender Kandidat ist die Brennstoffzelle.

Eine Brennstoffzelle wandelt die chemische Energie eines Brennstoffs direkt, das heißt ohne den Umweg über die Wärme, in Elektrizität um. Schon 1839 erfand Sir William Grove, ein englischer Jurist, das Gerät, aber es dauerte lange, ehe es zu seinem Recht kam. Brennstoffzellen versorgten die Gemini- und Apollo-Raumschiffe mit Energie und fanden damit eine ebenso exotische wie teure Anwendung. Heute gibt es Brennstoffzellen in verbesserten und tausendmal größeren Versionen, und es scheint, als sei das Stadium erreicht, in dem man von ihnen einen nennenswerten Beitrag zur öffentlichen Elektrizitätsversorgung erwarten kann.

Eine Brennstoffzelle (Bild 2) besteht aus zwei Elektroden - einer positiven (der Kathode) und einer negativen (der Anode) -, die durch einen Elektrolyten getrennt sind. Im Elektrolyten können Ionen (positiv oder negativ geladene Atome, Moleküle oder Molekülteile) transportiert werden, nicht aber die negativ geladenen Elektronen, die für den Elektrizitätstransport in metallischen Leitern verantwortlich sind. Der Brenn-

*Brennstoffzellen wandeln die chemische Energie eines Brennstoffs direkt in Energie um. Sie emittieren weder Schmutz noch Schadstoffe und haben, wenn man ihre Abfallwärme nutzt, einen hohen Wirkungsgrad. Gegenwärtig wird untersucht, ob sie auch im großen Maßstab wirtschaftlicher Arbeiten.*

stoff (beispielsweise Wasserstoff) wird der Anode zugeführt, während man die Kathode mit dem Oxidationsmittel (beispielsweise Sauerstoff aus der Luft) versorgt. An der Anode bewirkt ein Katalysator die Aufspaltung der Wasserstoff-Moleküle in Wasserstoff-Ionen ( $H^+$ ) und Elektronen ( $e$ ). Da Anode und Kathode porös sind und der Elektrolyt aus Phosphorsäure besteht, können die Wasserstoff-Ionen von der Anode zur Kathode wandern, wo sie sich mit dem Sauerstoff und den über den Stromkreis von der Anode zur Kathode fließenden Elektronen zu Wasser ( $H_2O$ ) vereinigen, das die Brennstoffzelle verläßt. Im Effekt "verbrennt" die Zelle also Wasserstoff mit Sauerstoff zu Wasser, setzt die dabei freiwerdende Energie aber nicht wie bei einer normalen Verbrennung vollständig in Wärme um, sondern verwendet sie zum Teil, um Elektronen durch den Stromkreis von der Anode zur Kathode zu treiben, das heißt einen elektrischen Strom zu erzeugen, der eine Lampe zum Leuchten bringen oder einen Elektromotor antreiben kann.

Die Art des Elektrolyten hängt von der chemischen Reaktion ab, mit der die Brennstoffzelle arbeitet, denn die Natur dieser Reaktion bestimmt, welches Ion von der Anode zur Kathode gelangen muß. Saure Elektrolyte transportieren Wasserstoff-Ionen ( $H^+$ ), in alkalischen Elektrolyten wandern Hydroxid-Ionen

(OH-). Carbonat-Ionen (CO<sub>3</sub><sup>2-</sup>) brauchen ein Salz der Kohlensäure als Elektrolyt, und um Sauerstoff-Ionen (O<sup>2-</sup>) zu transportieren, verwendet man ein festes Oxid.

Die Gleichspannung, die eine Brennstoffzelle erzeugen kann, hängt gleichfalls von der Art der chemischen Reaktion ab, aus der sie ihre Energie gewinnt. Eine mit Wasserstoff und Sauerstoff betriebene Zelle ist theoretisch in der Lage, bei normalem Druck und normaler Temperatur eine Spannung von 1,23 Volt zu liefern. Tatsächlich erhält man infolge von Energieverlusten in der Zelle jedoch nur zwischen 0,6 und 0,85 Volt. Die Stromstärke, also die Zahl der in der Zeiteinheit zur Verfügung gestellten Elektronen, ergibt sich aus der Geschwindigkeit der chemischen Reaktion und aus der Größe der Oberflächen von Anode und Kathode.

Eine einfache Brennstoffzelle erhält man schon, wenn man zwei Kohlestäbe, die etwas Platin als Katalysator enthalten, in ein Gefäß mit Schwefelsäure taucht und den einen Stab mit Wasserstoff, den anderen mit Sauerstoff umspült. Die Schwefelsäure wirkt als Elektrolyt, und die Reaktion des Wasserstoffs mit dem Sauerstoff zu Wasser erzeugt in der in Bild 2 gezeigten Weise eine Gleichspannung von etwas weniger als einem Volt. Man kann diese Spannung messen, wenn man die Kohlestäbe außerhalb des Gefäßes durch ein Voltmeter verbindet. Natürlich eignet sich diese einfache Anordnung nicht für technische Zwecke: die Oberfläche der Kohlestäbe ist zu klein, Schwefelsäure ist ein unpraktischer Elektrolyt, und offene Gefäße lassen sich nicht raumsparend zu Blöcken von Brennstoffzellen zusammenfassen. Man baut Brennstoffzellen daher so, daß man den Elektrolyten als dünne, flache Schicht zwischen zwei gleichfalls flache, porös gestaltete und mit dem Katalysator imprägnierte Elektroden packt. In dieser Form kann man sie zu hundertern übereinander stapeln (Bild 1). Die Spannung, die ein solcher Stapel liefert, ergibt sich, wenn man die Spannung der Einzelzelle mit der Anzahl der Zellen multipliziert.

Zu einem Brennstoffzellen-Kraftwerk gehören neben zahlreichen Brennstoffzellen-Stapeln noch zwei weitere Anlagen (Bild 3): ein Brennstoff-Aufbereiter (auch als Reformer bezeichnet) hat die Aufgabe, aus einem billigen und leicht speicherbaren Brennstoff, beispielsweise Kohle, den Brennstoff zu erzeugen (etwa ein wasserstoffreiches Gas), mit dem die Zellen arbeiten können, und der Wechselrichter wandelt die von den Zellen gelieferte Gleichspannung in Wechselspannung um, die dann dem öffentlichen Elektrizitätsnetz zugeführt werden kann.

Der Wirkungsgrad eines Brennstoffzellen-Kraftwerks, das heißt der Prozentsatz der im angelieferten Brennstoff enthaltenen Energie, der in Form von elektrischer Energie schließlich dem öffentlichen Elektrizitätsnetz zu Verfügung gestellt wird, ergibt sich auf etwa fünf Prozent genau aus der empirischen Formel  $n=59 \times V$ , wobei V die von der einzelnen Zelle gelieferte Spannung ist. Das für den Bau eines Brennstoffzellen-Kraftwerks erforderliche Kapital besteht nur zu zwölf bis dreißig Prozent aus Kosten für Brennstoffzellen-Stapel und Wechselrichter. Den Rest braucht man für den Bau der Anlage, die den Brennstoff aufbereitet, wobei die Kosten mit Kohle als primärem Brennstoff höher sind als bei Verwendung von Erdöl. Die meisten Entwicklungsarbeiten verfolgen gegenwärtig das Ziel, Leistung, Zuverlässigkeit und Stabilität der einzelnen Brennstoffzelle zu steigern und deren Kosten zu senken.

Brennstoffzellen unterscheiden sich in ihrer Arbeitstemperatur sowie im Elektrolyten, im Brennstoff und im Oxidationsmittel. Nicht alle denkbaren Kombinationen dieser vier Merkmale sind praktisch brauchbar, denn mit der Wahl des Elektrolyten trifft man eine Vorentscheidung hinsichtlich der anderen Faktoren und umgekehrt. So kann man mit Phosphorsäure als Elektrolyten nur zwischen 150 und 200 Grad Celsius arbeiten. Unterhalb dieses Temperaturbereiches hat die Phosphorsäure eine zu geringe Leitfähigkeit, und oberhalb greift sie das Elektrodenmaterial an.

Eine Brennstoffzelle mit einer Carbonat-Schmelze als Elektrolyten braucht eine Arbeitstemperatur zwischen sechshundert und siebenhundert Grad Celsius und einen aus Kohlenstoffoxiden und Wasserstoff bestehenden Brennstoff. Mit Kaliumhydroxid als Elektrolyten, legt man sich auf eine Temperatur zwischen 50 und 150 Grad Celsius fest und muß Brennstoff und Oxidationsmittel von Kohlenstoffoxiden freihalten, da sich aus dem Kaliumhydroxid sonst Kaliumcarbonat bildet, das die Leistung der Zelle drastisch verringert. Brennstoffzellen mit Kaliumhydroxid als Elektrolyten, reinem Wasserstoff als Brennstoff und reinem Sauerstoff als Oxidationsmittel haben sich in Raumschiffen bewährt, aber ihre irdische Verwendung scheitert am Preis des reinen, das heißt von Kohlenstoffoxiden freien Wasserstoffs und Sauerstoffs.

Brennstoffzellen mit Schwefelsäure, Sulfonsäuren oder festen Polymeren als Elektrolyten hängen davon ab, daß der Elektrolyt den richtigen Wassergehalt hat. Man muß sie unterhalb hundert Grad Celsius betreiben, wenn man Luft als Oxidationsmittel verwendet, denn anderenfalls nimmt der in der Luft enthaltene Stickstoff, der die Brennstoffzelle unverändert verläßt, zuviel Wasserdampf mit sich. bei dieser Temperatur erreicht die Brennstoffzelle aber nicht den Wirkungsgrad, den sie in einem an das öffentliche Elektrizitätsnetz angeschlossenen Kraftwerk haben muß.

In erster Linie interessiert man sich heute für Brennstoffzellen, die mit Phosphorsäure als Elektrolyten arbeiten, beschäftigt sich in kleinerem Umfang auch mit Carbonat-Schmelzen als Elektrolyten und untersucht feste Elektrolyte, die Sauerstoff-Ionen transportieren und Arbeitstemperaturen um tausend Grad Celsius verlangen, doch sind Brennstoffzellen auf dieser Basis vorerst noch nicht praktikabel.

Die Vorteile der Brennstoffzelle liegen in ihrem Wirkungsgrad, ihren umweltschonenden Eigenschaften und in der Tatsache, daß sie sich wie ein Baustein

verwenden läßt, um beliebig große Systeme zusammensetzen. Ihr Wirkungsgrad, bleibt zwischen 25 und 100 Prozent ihrer Nennleistung nahezu unverändert (Bild 4), was beträchtliche Einsparungen an Brennstoff ermöglicht, wenn man sie einsetzt, um die täglichen Schwankungen des Elektrizitätsbedarfs auszugleichen. Man erkennt das aus folgendem Beispiel (Bild 4) : Ein herkömmliches mit Öl beheiztes Kraftwerk, in dem die bei der Verbrennung gewonnene Wärme eine Gasturbine treibt, deren Abwärme zum Betrieb einer Dampfturbine dient, braucht ungefähr 9000 Kilojoule Wärmeenergie, um eine Kilowattstunde elektrische Energie zu erzeugen, wenn es bei seiner Nennleistung arbeitet. Wird es - bei Lastabhängigem Betrieb - nur zur 40 Prozent seiner Nennleistung genutzt, so verbraucht es fast 12000 Kilojoule, um eine Kilowattstunde zu liefern. Ein Brennstoffzellen-Kraftwerk ver braucht dagegen etwa 9500 Kilojoule pro Kilowattstunde bei voller Nennleistung und 9800 Kilojoule pro Kilowattstunde bei vierzig Prozent seiner Nennleistung. Es ist also besonders wirtschaftlich, ein normales Wärmekraftwerk so auszulegen, daß es ungefähr den durchschnittlichen Elektrizitätsbedarf befriedigt, wenn es bei voller Nennleistung läuft, und es mit einem Brennstoffzellen-Kraftwerk zu koppeln, das lastabhängig betrieben wird und für den Spitzenbedarf aufkommt. In einem Brennstoffzellen-Kraftwerk erzeugt nur die Anlage, die der Brennstoff-Aufbereitung dient, umweltbelastende Abgase. Mit Öl oder Kohle als primären Brennstoffen sind das vor allem Schwefeldioxid und Stickstoffoxid, doch liegt die Schwefeldioxid-Emission besonders niedrig, weil die Brennstoffzelle gegen Schwefelverbindungen empfindlich ist, so daß diese in der Brennstoff-Aufbereitungsanlage entfernt werden müssen. Ein Brennstoffzellen-Kraftwerk sollte kaum Lärm verursachen und bei Außentemperaturen bis zu 35 Grad Celsius auch kein Kühlwasser brauchen. Es müßte also möglich sein, solche Kraftwerke in unmittelbarer Nähe des Verbrauchers zu installieren, was sowohl den Bau neuer Überlandleitungen als

auch die mit dem Elektrizitätstransport über große Entfernungen verbundenen Energieverluste vermeidet.

Da sich Brennstoffzellen bausteinartig zu Kraftwerken zusammensetzen lassen, kann man sie in einer Fabrik in großen Mengen produzieren. Die Größe eines aus ihnen konstruierten Kraftwerks kann sich nach dem Elektrizitätsbedarf des Versorgungsgebietes richten und mit diesem wachsen. Überdies vermindert die Dezentralisierung der Elektrizitätsversorgung durch den Bau lokaler Brennstoffzellen-Kraftwerke die Anfälligkeit des Systems gegen Störungen.

Mit Brennstoffzellen sollte es also möglich sein, ein abgestuftes System der Elektrizitätsversorgung aufzubauen: Verhältnismäßig keine Anlagen (mit Leistungen zwischen 25 und 200 Kilowatt) könnten in großen Wohn- oder Bürogebäuden stehen, mit Erdgas arbeiten und sowohl elektrische Energie erzeugen als auch Wärmeenergie (in Form von Abfallwärme). Sie würden für beide Energieformen zusammen nicht mehr Brennstoff verbrauchen als gegenwärtig für Heizzwecke allein benötigt wird und ihr Wirkungsgrad sollte nahezu hundert Prozent betragen.

Größere Anlagen mit Leistungen zwischen 5 und 25 Megawatt wären an verschiedenen Stellen eines Elektrizitätsnetzes zu installieren, um Schwankungen des Elektrizitätsbedarfes aufzufangen. Wenn man in der Lage, auch die Abfallwärme dieser Anlagen nutzbringend zu verwenden, ließen sich Wirkungsgrade über achtzig Prozent (bezogen auf verbrauchten Brennstoff) erreichen.

Erst in fernerer Zukunft wird man Brennstoffzellen mit Kohlevergasungsanlagen kombinieren, um zu Kraftwerken von 150 bis 600 Megawatt Leistung zu kommen, für die Kohle der primäre Brennstoff ist. Bezogen auf den Heizwert der verbrauchten Kohle sind hier Wirkungsgrade über 45 Prozent zu erwarten. Die mit Phosphorsäure als Elektrolyten arbeitende Brennstoffzelle ist

am weitesten fortgeschritten. Sie besteht aus einer dünnen, porösen und mit konzentrierter Phosphorsäure getränkten Platte, die zwischen zwei porösen Kohle-Elektroden liegt. Letztere enthalten als Katalysator zwischen 2,7 und 8 Gramm Platin pro Quadratmeter. Die Phosphorsäure-Zelle arbeitet bei 150 bis 190 Grad Celsius, erzeugt eine Gleichspannung von 0,64 Volt und hat eine Leistungsdichte von 1,1 bis 2,2 Kilowatt pro Quadratmeter. Man schätzt, daß Kraftwerke, die aus solchen Zellen bestehen, 1980 in den USA für 350 Dollar pro Kilowatt produziert werden können, sofern sich die Gesamtproduktion von Brennstoffzellen auf 500 Megawatt pro Jahr beläuft. Dieser Preis wäre mit den Kosten anderer Stromerzeuger für lastabhängigen Betrieb vergleichbar. Kleine Stapel von Phosphorsäure-Zellen haben heute schon viele tausend Betriebsstunden erreicht, aber die über vierzigtausend Stunden (viereinhalb Jahre) gehende Dauerprüfung eines großen Stapels steht noch aus.

Die Entwicklung von Brennstoffzellen, die mit einer Carbonat-Schmelze als Elektrolyten arbeiten, ist um mindestens fünf Jahre weiter zurück als die Entwicklung der Phosphorsäure-Zellen. Eine dünne Schicht aus geschmolzenem Carbonat und inertem Füllmaterial liegt hier zwischen zwei Elektroden aus porösem Nickel. Die Arbeitstemperatur beträgt 650 Grad Celsius, und unter diesen Bedingungen verlaufen die Reaktionen an den Elektroden so schnell, daß man keine Katalysatoren braucht. An der Kathode bildet sich aus Nickel und Sauerstoff Nickeloxid, das als das eigentliche Kathodenmaterial wirkt, während die Anode unverändert bleibt. Carbonat-Zellen erzeugen eine Spannung von 0,785 Volt und haben Leistungsdichten zwischen 1,1 und 1,3 Kilowatt pro Quadratmeter. Die höhere Spannung und die Unempfindlichkeit gegenüber Kohlenstoffoxiden sind Vorteile der Carbonat-Zellen. Eine kleine Zelle hat schon mehr als 14000 Betriebsstunden hinter sich, und ein aus 19 Zellen von je ungefähr 0,1 Quadratmeter Fläche bestehender Stapel arbeitet seit über 1400 Stunden.

Trotz dieser ermutigenden Resultate gibt es noch immer Probleme. Sie bestehen vor allem in strukturellen Änderungen und im allmählichen Verlust des Elektrolyten und begrenzen Betriebsdauer und Zuverlässigkeit der Zellen. Günstige Schätzungen kommen für die Carbonat-Zelle auf ähnliche Produktionskosten wie für die Phosphorsäure-Zelle.

Die anderen Teilsysteme eines Brennstoffzellen-Kraftwerks, die Anlage für die Brennstoff-Aufbereitung und der Wechselrichter, sind so weit entwickelt, daß sie zusammen mit der ersten Generation der Brennstoffzellen eingesetzt werden können. Die Brennstoff-Aufbereitungsanlagen können Erdgas und eine als Naphtha bezeichnete, zwischen 150 und 180 Grad Celsius siedende Fraktion des Erdöls zu einem wasserstoffreichen Gasmisch verarbeiten. Gegenwärtig bemüht man sich, sie so weit zu entwickeln, daß sie auch mit höhersiedenden Erdölfraktionen und mit Produkten der Kohleverflüssigung betrieben werden können.

Die von den Brennstoffzellen gebotenen Vorteile werden sich erst nutzen lassen, wenn solche Zellen großtechnisch produziert werden. Hier liegt momentan das entscheidende Problem, denn die ersten Zellen werden verhältnismäßig teuer sein, und der Preis wird nur in dem Maß sinken, in dem die Produktion wächst (Bild 5). Nur wagemutige Unternehmer, finanzstarke Kunden und eine Energiepolitik, die einer Regierung die Möglichkeit gibt, beide zu unterstützen, werden zusammen in der Lage sein, eine Entwicklung einzuleiten und fortzusetzen, die im Interesse aller liegt, da sie es gestattet, Energie zu sparen und die Belastung der Umwelt zu vermindern.

In den USA gibt es zur Zeit drei Arbeitsprogramme, deren Absicht darin besteht, die Brennstoffzelle möglichst rasch so weit zu entwickeln, daß sie für die öffentliche Elektrizitätsversorgung nützlich wird. Das erste Programm läuft unter dem Akronym TARGET (Team to Advance Research for Gas Energy Transformation). Es besteht seit 1967

und hat sich der mit Erdgas oder Synthesegas betriebenen Phosphorsäure-Zelle angenommen. Ziel der Arbeiten ist die Entwicklung kleiner Kraftwerke, die in großen Büro- oder Wohngebäuden installiert werden können. Da praktisch die gesamte Abfallwärme für Heiz- und Kühlzwecke von je 12,5 kilowatt Leistung ausprobiert. Aufgrund der so gewonnenen Erfahrungen baute man 1975 eine 40-Kilowatt-Anlage, die zwischen 1979 und 1981 in fünfzig Prototypen erprobt werden und 1982 in den Handel kommen soll. Für TARGET hat die Industrie bisher etwa hundert Millionen Dollar ausgegeben.

Das zweite Programm wurde 1972 von einigen Elektrizitätsversorgungsunternehmen unter der Bezeichnung FCG-1 (Fuel Cell Generator) ins Leben gerufen. Es beschäftigt sich mit Megawatt-Kraftwerken, die aus Phosphorsäure-Zellen bestehen. 1976 und 1977 zeigte eine Demonstrationsanlage mit einer Leistung von einem Megawatt, daß ein mit Naphtha betriebenes Brennstoffzellenkraftwerk geeignet ist, als lastabhängiger Elektrizitätserzeuger zu arbeiten, und den Ansprüchen der Industrie hinsichtlich Wirkungsgrad und Umweltfreundlichkeit genügt. Bis 1980 will man mitten in New-York, an einer Stelle also, an der niemals ein herkömmliches Kraftwerk stehen könnte, ein 4,5-Megawatt-Brennstoffzellen-Kraftwerk errichten, dessen Kosten man auf sechzig Millionen Dollar schätzt. Ein drittes Programm, das zunächst allein von der Industrie getragen wurde, an dem sich jetzt aber auch das Energie-Ministerium beteiligt, widmet seine Aufmerksamkeit den Brennstoffzellen, die mit einer Carbonat-Schmelze als Elektrolyten arbeiten. Ein aus solchen Zellen bestehendes Kraftwerk, soll Kohle als primären Brennstoff verwenden können, seine Leistung wird einige hundert Megawatt betragen, und man rechnet mit einer Entwicklungszeit von etwa zehn Jahren.

Auch in der Bundesrepublik Deutschland wird an der Entwicklung von Brennstoffzellen gearbeitet, allerdings weniger mit dem Ziel, zu Kraftwerken

großer Leitung zu kommen. Notstromaggregate und Geräte für die elektrische Versorgung kleiner Einheiten (Gabelstapler, automatisierte Meßstationen) bis hinab zu Spannungsquellen, die in den menschlichen Körper implantiert werden können (beispielsweise für den Antrieb eines Herzschrittmachers) und mit Traubenzucker aus dem Blut als Brennstoff arbeiten, stehen hier im Vordergrund des Interesses.

Bild 1 : Dieser Stapel von nahezu fünfhundert Brennstoffzellen soll zusammen mit neunzehn anderen in einem 4,5-Megawatt-Brennstoffzellen-Kraftwerk in New York installiert werden. Jede Schicht des Stapels ist eine Brennstoffzelle, die eine Spannung von 0,64 Volt erzeugt und fünfhundert Watt leistet. Die Brennstoffzellen arbeiten mit Phosphorsäure als Elektrolyten und verwenden als Brennstoff ein wasserstoffreiches Gas, das aus einer als Naphtha bezeichneten, zwischen 150 und 180 Grad Celcius siedenden Erdölfraktion erzeugt wird. Als Oxidationsmittel dient der Sauerstoff der Luft. Das Schema einer solchen Zelle zeigt Bild 2.

Bild 2 : Schema einer Brennstoffzelle. Als Brennstoff dient ein aus Kohle oder Erdöl erzeugtes wasserstoffreiches Gas. An der Anode werden die Wasserstoffmoleküle ( $H_2$ ) in positive geladene Elektronen Wasserstoff-Ionen ( $H^+$ ) und negativ geladene Elektronen ( $e$ ) aufgespalten. Die Wasserstoff-Ionen werden im Elektrolyten zur Kathode transportiert. Die Elektronen fließen durch den äußeren Stromkreis von der Anode zur Kathode. Dort vereinigen sich Wasserstoff-Ionen und Elektronen mit dem als Oxidationsmittel dienenden Sauerstoff ( $O_2$ ) der Luft zu Wasser, das die Brennstoffzelle verläßt.

Bild 3 : Schema eines Brennstoffzellen-Kraftwerks. Steht zum Betrieb der Brennstoffzellen kein wasserstoffreiches Gasgemisch zur Verfügung, so kann das Kraftwerk auch mit Erdöl oder Kohle arbeiten, muß daraus aber in einem Brennstoff-Aufbereiter (oder Reformer) ein wasserstoff-reiches Gasgemisch er-

zeugen. Die Brennstoffzellen liefern eine Gleichspannung, die vom Wechselrichter in Wechselspannung umgewandelt werden muß, bevor sie an das öffentliche Elektrizitätsnetz abgegeben wird.

Bild 4 : Die meisten Maschinen, die Wärmeenergie in elektrische Energie umwandeln, nutzen die Wärmeenergie umso besser aus, je mehr sie in der Nähe ihrer vollen Nennleistung (der vom Hersteller angegebenen Maximalleistung) arbeiten. Dagegen unterliegt der Wirkungsgrad einer Brennstoffzelle zwischen 25 und 100 Prozent ihrer Nennleistung keiner sehr großen Änderung. Da von einem Kraftwerk nicht ständig die volle Leistung verlangt wird, braucht man Anlagen, die elektrische Energie in dem Maß erzeugen, in dem der Bedarf einen durchschnittlichen Wert über steigt. Hier erweisen sich Brennstoffzellen als günstig. Ihre Verwendung ermöglicht es anderen Maschinen, ständig bei voller Nennleistung zu arbeiten und so bei maximalen Wirkungsgrad den mittleren Bedarf zu decken.

Bild 5 : Die Kosten für den Bau eines Brennstoffzellen-Kraftwerks (farbige Kurve und farbige Zahlen) sind hier der Entwicklung der Kraftwerksgröße (schwarze Kurve und schwarze Zahlen) gegenübergestellt. Die ausgezogenen Kurven geben Erfahrungswerte, die gestrichelten Teile geschätzte Werte wieder. Die erste 1,5-Kilowatt-Anlage wurde in den späten sechziger Jahren für die US-Armee gebaut und kostete über hunderttausend Dollar. 1980 sollten die Kosten nur noch bei 350 Dollar pro Kilowatt liegen, sofern Brennstoffzellen bis dahin großtechnisch, das heißt im Umfang von fünfhundert Megawatt pro Jahr produziert werden können. Das wiederum setzt voraus, daß der Markt für Brennstoffzellen bis dahin so groß geworden ist, daß er die gesamte Produktion aufnehmen kann.

Le texte a été saisi directement sur une console du C.I.R.C.E. et subi un codage destiné à conserver les richesses typologiques que ne permettaient pas de traiter les installations de l'époque.

La majuscule est codée par le '\$', le Umlaut par une '\*\*', le passage à la ligne par '\$\_', les titres entre '||', les sous- titres entre '|', le 'ß' est écrit 'S\*'.  
.

Sur la page suivante, nous montrons les premières phrases du texte allemand dans les fichiers créés par le module de mise en forme.

**.TEXALL(TEXALL1)** : C'est le fichier contenant le texte allemand d'origine.

**.TEXALL1.SORTIEØ.CHAINES(CHAINÉ1)** : Les phrases sont concaténées en chaînes de caractères, le langage de programmation PL/1 acceptant de traiter des chaînes de 32767 caractères.

**.TEXALL1.SORTIE1.BASE** : La phrase est découpée.

**.TEXALL1.SORTIE2.BASE1** : On a inclus dans le fichier précédent les résultats de l'analyse du verbe.

**.TEXALL1.SORTIE2.BASE21** : On ne conserve dans cette version que les mots composés potentiels.

**.TEXALL1.SORTIE2.BASE23** : Les données précédentes sont tassées et disposées de façon à accélérer le traitement ultérieur.

TEXALL(TEXALL1) :

00010 :: Brennstoffzellenkraftwerke :: 1 Im nächststen Jahrzehnt wird man von einem Elektrizitätswerk fordern, 2  
 00020 das es einen hohen Wirkungsgrad hat, möglichst wenig Schutz emittiert, keinen Larv verursacht und rasch  
 00030 ch zu installieren ist. Ein viel versprechender Kandidat ist die Brennstoffzelle. 3 keine Brennstoffzelle 4  
 00040 wandelt die chemische Energie eines Brennstoffs direkt, das heißt ohne den Umweg über die Wärme, in 5  
 00050 Elektrizität um. Schon 1839 erfand Sir William Grove, ein englischer Jurist, das Gerät, aber es dau  
 00060 erte lange, ehe es zu seinem Recht kam. Brennstoffzellen versorgten die Gemini- und Apollo-Raumfahrzeuge  
 00070 mit Energie und fanden damit eine ebenso exotische wie teure Anwendung, heute gibt es Brennstoffzellen in 6  
 00080 verbesserten und tausendmal größeren Versionen, und es scheint, als sei das Stadium erreicht, in dem man  
 00090 n von ihnen einen nennenswerten Beitrag zur öffentlichen Elektrizitätsversorgung erwarten kann. 7 seine 8%  
 00100 Rennstoffzelle (Bild 2) besteht aus zwei Elektroden - einer positiven (der Kathode) und einer negativen  
 00110 (der Anode) -, die durch einen Elektrolyten getrennt sind. Im Elektrolyten können Ionen (positiv odz  
 00120 er negativ geladene Ionen, Moleküle oder Ionen) transportiert werden, nicht aber die negativ gel  
 00130 ladene Elektronen, die für den Elektrizitätstransport in metallischen Leitern verantwortlich sind. Der  
 00140 Brennstoff (beispielsweise Wasserstoff) wird der Anode zugeführt, während man die Kathode mit dem 10  
 00150 xidationsmittel (beispielsweise Sauerstoff aus der Luft) versorgt. Man der Anode bewirkt ein Katalysator  
 00160 die Aufspaltung des Wasserstoff-Moleküle

1 :: Brennstoffzellenkraftwerke :: 1 Im nächststen Jahrzehnt wird man von einem Elektrizitätswerk fordern, das es einen hohen  
 Wirkungsgrad hat, möglichst wenig Schutz emittiert, keinen Larv verursacht und rasch zu installieren ist. Ein viel verspre  
 chender Kandidat ist die Brennstoffzelle. 3 keine Brennstoffzelle 4 wandelt die chemische Energie eines Brennstoffs direkt, das  
 heißt ohne den Umweg über die Wärme, in 5 Elektrizität um. Schon 1839 erfand Sir William Grove, ein englischer Jurist  
 . Das Gerät, aber es dauerte lange, ehe es zu seinem Recht kam. Brennstoffzellen versorgten die Gemini- und Apollo-Raumf  
 1ffe mit Energie und fanden damit eine ebenso exotische wie teure Anwendung. Heute gibt es Brennstoffzellen in verbesserten und  
 tausendmal größeren Versionen, und es scheint, als sei das Stadium erreicht, in dem man von ihnen einen nennenswerten Beitr  
 ag zur öffentlichen Elektrizitätsversorgung erwarten kann. 7 keine Brennstoffzelle (Bild 2) besteht aus zwei Elektroden - e  
 iner positiven (der Kathode) und einer negativen (der Anode) -, die durch einen Elektrolyten getrennt sind. Im Elektrolyt  
 in können Ionen (positiv oder negativ geladene Ionen, Moleküle oder Ionen) transportiert werden, nicht aber die ne  
 gativ geladenen Elektronen, die für den Elektrizitätstransport in metallischen Leitern verantwortlich sind. Der Brennstoff  
 (beispielsweise Wasserstoff) wird der Anode zugeführt, während man die Kathode mit dem 10xidationsmittel (beispielsweise S  
 auerstoff aus der Luft) versorgt. Man der Anode bewirkt ein Katalysator die Aufspaltung des Wasserstoff-Moleküle in Wasser  
 stoff-Ionen (H+) und Elektronen (e-). Da 11

.TEXALLI.SORTIEI.BASE :

```

*0025
0206
0030
0000
0000
0003 006 013 020
0000
0000
0000
00025007020001 $MAN
00025007030002 BAUT
00025007040003 $BRENNSTOFFZELLEN
00025007050004 DAHER
00025007060005 SO
00025007070006 ,
00025007080007 DAS*
00025007090008 MAN
00025007100009 DEN
00025007110010 $ELEKTROLYTEN
00025007120011 ALS
00025007130012 DU* NNE
00025007140013 ,
00025007150014 FLACHE
00025007160015 $SCHICHT
00025007170016 ZWISCHEN
00025007180017 ZWEI
00025007190018 GLEICHFALLS
00025007200019 FLACHE
00025007210020 ,
00025007220021 PORO* S
00025007230022 GESTALTETE
00025007240023 UND
00025007250024 MIT
00025007260025 DEM
00025007270026 $KATALYSATOR
00025007280027 IMPRA*GNIERTE
00025007290028 $ELEKTRODEN
00025007300029 PAKT
00025007310030 .

```

Chaque phrase se présente ici sous une forme facilement exploitable. L'extrait ci-dessus concerne la phrase n°25. Son numéro est précédé d'une '\*' à la première ligne.

Elle a 206 caractères (deuxième ligne) et 30 mots (troisième ligne). Les six lignes qui suivent indiquent le nombre et la position, dans l'ordre, des deux-points, points-virgules, virgules, parenthèses ouvrantes, parenthèses fermantes, tirets. Les 6ème, 13ème et 20ème mots sont bien des virgules. Les lignes suivantes commencent par une numérotation à 14 chiffres :

00025 numéro de phrase, 00702 numéro du mot dans le texte, 0001 numéro du mot dans la phrase.

.TEXALL1.SORTIE2.BASE1 :

```

*0025
0206
0030
0000
0000
0003 006 013 020
0000
0000
0000
00025007020001 $MAN
00025007030002      BAUT      BAU
00025007040003 $BRENNSTOFFZELLEN
00025007050004 DAHER
00025007060005 SO
00025007070006 ,
00025007080007 DAS*
00025007090008 MAN
00025007100009 DEN
00025007110010 $ELEKTROLYTEN
00025007120011 ALS
00025007130012 DU*NNE
00025007140013 ,
00025007150014 FLACHE
00025007160015 $SCHICHT
00025007170016 ZWISCHEN
00025007180017 ZWEI
00025007190018 GLEICHFALLS
00025007200019 FLACHE
00025007210020 ,
00025007220021 PORO*S
00025007230022      GESTALTETE      GESTALT
00025007240023 UND
00025007250024 MIT
00025007260025 DEN
00025007270026 $KATALYSATOR
00025007280027      IMPRA*GNIERTE      IMPRA*GNIERTE
00025007290028 $ELEKTRODEN
00025007300029      PACKT      PACK
00025007310030 .

```

Tous les fichiers ...SORTIE2... sont des combinaisons du fichier ...SORTIE1.BASE et des résultats verbaux. La préparation du fichier de départ pour le module MOTS COMPOSES passe par .TEXALL1.SORTIE2.BASE1 dans lequel on inclut les verbes, .TEXALL1.SORTIE2.BASE21 dans lequel on supprime les verbes, les mots trop courts pour être composés (longueur inférieure à 6 caractères), les mots-outils de plus de 6 caractères. Il ne reste que des unités candidates à être des composés. Elles sont classées alphabétiquement dans .TEXALL1.SORTIE2.BASE22 et tassées de façon particulière dans .TEXALL1.SORTIE2.BASE23 :

N'y figurent, lorsqu'il y a plusieurs occurrences, que la première avec le mot et la dernière avec la numérotation seulement. Cela permet au module MOTS COMPOSES de fonctionner plus rapidement.

TEXALL1.SORTIE2.BASE21

00024006860024 UNPRAKTISCHER  
 00024006870025 \$ELEKTROLYT  
 00024006910029 \$GEFA+S\*E  
 00024006950033 RAUMSPAREND  
 00024006970035 \$BLO\*CKEN  
 00024006990037 \$BRENNSTOFFZELLEN  
 00025007040003 \$BRENNSTOFFZELLEN  
 00025007110010 \$ELEKTROLYTEN  
 00025007160015 \$SCHICHT  
 00025007190018 GLEICHFALLS  
 00025007270026 \$KATALYSATOR  
 00025007290028 \$ELEKTRODEN  
 00026007400009 U\*BEREINANDERSTAPELN  
 00027007470002 \$SPANNUNG

TEXALL1.SORTIE2.BASE23

0000694 00028007920022 \$WECHSELRICHTER  
 0000698 0012403220009  
 0000699 00029008460053 \$WECHSELSPANNUNG  
 0000700 00124032220011  
 0000701 00005000730005 \$WILLIAN  
 0000702 00002000200016 \$WIRKUNGSGRAD  
 0000712 00136033740018  
 0000713 00061018460019 \$WIRKUNGSGRADE  
 0000714 00063018990010  
 0000715 00058017530032 \$WOHN-  
 0000716 00090025450015 \$WOHNGEBA\*UDEN  
 0000717 00145034230017 \$ZAHLEN  
 0000718 00145034360030  
 0000719 00020005480010 \$ZEITEINHEIT  
 0000720 00027007680023 \$ZELLEN  
 0000729 00100027740004  
 0000730 00062018620005 \$ZUKUNFT  
 0000731 00033009860011 \$ZUVERLA\*SSIGKEIT  
 0000732 00079022460018  
 0000733 00024006730011 \$ZWECKE  
 0000734 00080022570008 A\*HNLICHE  
 0000737 00015004340010 ALKALISCHEN  
 0000738 00079022380010 ALLNA\*HLICHEN  
 0000739 00042012280018 ANDERENFALLS  
 0000775 00055016670005 BAUSTEINARTIG  
 0000778 00010002200004 BEISPIELSMASSE  
 0000781 00102028610030  
 0000783 00045013550029 BELIEBIG  
 0000803 00046013780019 BETRA\*CHTLICHE  
 0000818 00029008000015 BILLIGEN  
 0000819 00035010260010 BRAUCHBAR

0001146 00147034550003 1,5-\$KILOWATT-\$ANLAGE  
 0001147 00098027210025 4,5-\$HEGAWATT-\$BRENNSTOFFZELLEN-\$K  
 0001148 00103029020018  
 0001149 00093025930010 40-\$KILOWATT-\$ANLAGE  
 \* LEXIQUE DE 00620 MOTS  
 \* 00794 LIGNES AU TOTAL  
 END OF DATA

Pour le texte de néerlandais, nous aurons de la même façon :

TEXOLL(TEXOLLI) :

00010 ; KERNENERGIE IN DE LAGE LANDEN . \$J. \$A. \$GOEDKOO . ; ; INHOUD : . \$ # VOORWOORD # . \$ 1. KERNREACTIES : 2  
 00020 10 . \$ 2. DODEVAARD : 25 . \$ 3. RADIOACTIVITEIT : 37 . \$ 4. LANGS \$HAAS EN \$SCHELDE : 55 . \$ 5. DE SPLIJTSTOFCE  
 00030 YCLUS : VAN URAANMIJN TOT ZOUTMIJN : 77 . \$ 6. ALMELD EN \$HOL : 94 . \$ 7. ANDERE KERNREACTORTYPEN : 107 . \$ 8. 2  
 00040 KUEKEN AAN DE \$RIJN : 126 . 9. KERNVERSMELTING : 139 . \$ 10. KOEPELS EN EEN KASTEEL : 155 . \$ # SLOTUWOORD : #2  
 00050 164 . \$ # LITERATUUR : # 167 . \$ # REGISTER : # 171 . 2+ KERNENERGIE IN DE LAGE LANDEN . \$ SYMB. 1 KERNCENTRA  
 00060 LE IN BEDRIJF . \$ SYMB. 2 KERNCENTRALE IN AANBOU . \$ SYMB. 3 MINSTENS EKE<N ANDERE KERNREACTOR . \$ SYMB. 4 EN2  
 00070 KELE FABRIEKEN EN LABORATORIA 2+ . ; VOORWOORD ! . \$ VERHANDELINGEN OVER KERNENERGIE LIJKEN ER SOMS OP UIT DE 2  
 00080 LEZER TE IMPONEREN . TOT VOOR ENKELE JAREN GEBEURDE DAT VOORAL DOOR HEM TE DOORDRINGEN VAN DE NIETIGHEID VAN 02  
 00090 E ATOOMKERNEN EN DE GROOTTE VAN DE DAARIN SLUITERENDE KRACHTEN . DE BEHANDELING WAS DAN HEESTAL HISTORISCH , W2  
 00100 AARBIJ VAAK UITVOERIG AANDACHT WERD BESTEED AAN DE PRESTATIES VAN DE ONDERZOEKERS DIE HET BOUWERK VAN DE N00EX  
 00110 RNE NATUURWETENSCHAP HEBBEN OPGERICHT EN DIE DAARVOOR VEELAL HET DE \$NOBELPRIJS ZIJN BELOOND . \$ SEDERTDIEN IS2  
 00120 DE WIND GEDRAID . VOORAL IN DE PERS WORDEN NU IN EINDELOZE HERHALING UITSPRAKEN VAN DEZELFDE EN ANDERE \$NOBEL  
 00130 LPRIJSWINNARS VEERGEGEVEN , WAARIN GEVAARSCHEID WORDT TEGEN DE GEVAREN VAN DE TOEPASSING VAN DE KER:

; KERNENERGIE IN DE LAGE LANDEN . \$J. \$A. \$GOEDKOO . ; ; INHOUD : . \$ # VOORWOORD # . \$ 1. KERNREACTIES : 10 . \$ 2. DODEVAARD :  
 25 . \$ 3. RADIOACTIVITEIT : 37 . \$ 4. LANGS \$HAAS EN \$SCHELDE : 55 . \$ 5. DE SPLIJTSTOFCYCLUS : VAN URAANMIJN TOT ZOUTMIJN : 77 . \$  
 6. ALMELD EN \$HOL : 94 . \$ 7. ANDERE KERNREACTORTYPEN : 107 . \$ 8. KUEKEN AAN DE \$RIJN : 126 . 9. KERNVERSMELTING : 139 . \$ 10. KOEP  
 ELS EN EEN KASTEEL : 155 . \$ # SLOTUWOORD : # 164 . \$ # LITERATUUR : # 167 . \$ # REGISTER : # 171 . 2+ KERNENERGIE IN DE LAGE LANDEN  
 . \$ SYMB. 1 KERNCENTRALE IN BEDRIJF . \$ SYMB. 2 KERNCENTRALE IN AANBOU . \$ SYMB. 3 MINSTENS EKE<N ANDERE KERNREACTOR . \$ SYMB. 4 EN  
 KELE FABRIEKEN EN LABORATORIA 2+ . ; VOORWOORD ! . \$ VERHANDELINGEN OVER KERNENERGIE LIJKEN ER SOMS OP UIT DE LEZER TE IMPONEREN . 7  
 OT VOOR ENKELE JAREN GEBEURDE DAT VOORAL DOOR HEM TE DOORDRINGEN VAN DE NIETIGHEID VAN DE ATOOMKERNEN EN DE GROOTTE VAN DE WAARIN 52  
 UJHERENDE KRACHTEN . DE BEHANDELING WAS DAN HEESTAL HISTORISCH , WAARBIJ VAAK UITVOERIG AANDACHT WERD BESTEED AAN DE PRESTATIES VAN  
 DE ONDERZOEKERS DIE HET BOUWERK VAN DE MODERNE NATUURWETENSCHAP HEBBEN OPGERICHT EN DIE DAARVOOR VEELAL HET DE \$NOBELPRIJS ZIJN BEL  
 OOND . \$ SEDERTDIEN IS DE WIND GEDRAID . VOORAL IN DE PERS WORDEN NU IN EINDELOZE HERHALING UITSPRAKEN VAN DEZELFDE EN ANDERE \$NOBEL  
 LPRIJSWINNARS VEERGEGEVEN , WAARIN GEVAARSCHEID WORDT TEGEN DE GEVAREN VAN DE TOEPASSING VAN DE KERNENE!

TEXOLLJ.SORTIEØ.CHAINES(CHAINEL) :

Le module de mise en forme découpera le texte en phrases :

.TEXOLL1.SORTIE1.BASE

y intégrera les résultats de l'analyse verbale

.TEXOLL1.SORTIE2.BASE21

et préparera le traitement des mots composés

.TEXOLL1.SORTIE2.BASE21 et .TEXOLL1.SORTIE2.BASE23

.TEXOLL1.SORTIE1.BASE :

:#0029

0200

0030

0000

0000

0001 017

0000

0000

0000

00029002490001 VOORAL

00029002500002 IN

00029002510003 DE

00029002520004 PERS

00029002530005 WORDEN

00029002540006 NU

00029002550007 IN

00029002560008 EINDELOZE

00029002570009 HERHALING

00029002580010 UITSPRAKEN

00029002590011 VAN

00029002600012 DEZELFDE

00029002610013 EN

00029002620014 ANDERE

00029002630015 \$NOBELPRIJSWINNAARS

00029002640016 WEERGEGEVEN

00029002650017 ,

00029002660018 WAARIN

00029002670019 GEWAARSCHUWD

00029002680020 WORDT

00029002690021 TEGEN

00029002700022 DE

00029002710023 GEVAREN

00029002720024 VAN

00029002730025 DE

00029002740026 TOEPASSING

00029002750027 VAN

00029002760028 DE

00029002770029 KERNENERGIE

00029002780030 .

.TEXOLL1.SORTIE2.BASE1 :

#0046  
 0199  
 0032  
 0000  
 0000  
 0001 016  
 0000  
 0000  
 0000  
 00046007160001 DE  
 00046007170002 TITEL  
 00046007180003 VAN  
 00046007190004 DIT  
 00046007200005 BOEKJE  
 00046007210006 ZOU  
 00046007220007 DE  
 00046007230008 VERWACHTING  
 00046007240009 KUNN EN KUNN  
 00046007250010 WEKK EN WEKK  
 00046007260011 DAT  
 00046007270012 OOK  
 00046007280013 BELEIDSASPECTEN  
 00046007290014 WORD EN WORD  
 00046007300015 BE SPROK EN BESPROK  
 00046007310016 ,  
 00046007320017 ZOALS  
 00046007330018 DE  
 00046007340019 NATE  
 00046007350020 WAARIN  
 00046007360021 KERNENERGIE  
 00046007370022 ZAL  
 00046007380023 WORD EN WORD  
 00046007390024 TOE GE PAS T TOEPAS  
 00046007400025 EN  
 00046007410026 DE  
 00046007420027 MANIER  
 00046007430028 WAAROP  
 00046007440029 DAT  
 00046007450030 WORD T WORD  
 00046007460031 GEORGANISEER D ORGANISEER  
 00046007470032 .

.TEXOLL1.SORTIE2.BASE21 :

00045007020006 ONTWIKKELING  
 00045007050009 INTERESSANT  
 00045007130017 NOODZAKELIJKE  
 00046007230008 VERWACHTING  
 00046007280013 BELEIDSASPECTEN  
 00046007360021 KERNENERGIE  
 00047007500003 ASPECTEN  
 00047007510004 INCIDENTEEL  
 00047007650018 TECHNISCHE  
 00047007660019 INFORMATIE

.TEXOLL1.SORTIE2.BASE23 :

```

*...SORTIE2.BASE23
*00051
*04186
0000001 00057009620016 $ARDENNEN
0000002 00033004230036 $BELGIE*
0000005 00057009500004
0000006 00032003620027 $BELGISCHE
0000007 00049008240007 $CENTRUM
0000008 00035004800031 $DODEWAARD
0000010 00219049000009
0000011 00033004290042 $DUITSE
0000012 00067012120013 $GASUNIE
0000013 00002000110003 $GOEDKOOP
0000014 00063011030033 $GRONINGSE
0000015 00040005980031 $KALKAR
0000016 00033004210034 $NEDERLAND
0000022 00063010900020
0000023 00031003210014 $NEDERLANDSE
0000026 00219049060015
0000027 00027002380035 $NOBELPRIJS
0000028 00029002630015 $NOBELPRIJSWINNAARS
0000029 00049008230006 $REACTOR
0000030 00009000470006 $SCHELDE

```

```

0000952 00071013360022 ZUURSTOFATOMEN
0000953 00120025240019
0000954 00112022980024 ZUURSTOFATOON
0000955 00115023800010
0000956 00213047730017 ZWAARDER
0000957 00175039040009 ZWAARSTE
0000958 00126027050022 0,7/100
0000959 00184041240017 10*-14
0000960 00126026990016 99,3/100
* LEXIQUE DE 00487 MOTS
* 00648 LIGNES AU TOTAL
END OF DATA

```

Pour les textes en luxembourgeois, le système est identique. Le fichier texte s'intitulera .TEXLUX(TEXTLUX1), la chaîne .TEXLUX1.SORTIE0.CHAINES(CHAINÉ1), avec les fichiers intermédiaires .TEXLUX1.SORTIE1.BASE, .TEXLUX.SORTIE2.BASE1....

00010 : \$JEAN-\$PAUL \$SCHNEIDER : . \$ !! D' \$REFORM VUN DER \$LE+TZEUEGER \$EKONOMIE ( 1. \$DEEL ) !! . \$ ! 1. \$ D' \$STRUKZ  
00020 TUR VUN DER \$LX  
00030 E+TZEUEGER \$EKONOMIE : . \$ BIS AN D' \$NE+TT VUN 19TEN \$JOERHONDERT WAR \$LE+TZEUEGER EN AARNT \$LAND : E+T GOUF Z  
00040 DEEMOLS KENG GRE>ISSER \$INDUSTRIE , DEN \$DE>NGSCHLEESCHTUNGSSECTEUR WAR GANZ KLENG . ONGEFE>IER 60/100 VUN DE \$LEIT Z  
00050 HUN AN DER \$LANDWIRTSCHAFT GESCHAFFT , WOUVUN DE>I MEESCHT E KLENGT \$AKOMMES HATEN , DAT KNAPPS DUERGONG FIR ZE LIX  
00060 EUEN , 22/100 HUN AN DER \$INDUSTRIE AN AM \$HANDWIERK GESCHAFFT , 18/100 AN \$HANDEL , DE FRA+IE \$BERUFFER AN DE \$SERVZ  
00070 ICERZ  
00080 . \$ AM \$JOER 1877 HUET DEN ENGLISCHEN \$INGENIEUR \$GILCHRIST \$THOMAS ENG \$METHOD ERFONNT , FIR AUS PHOSPHORRA+IZ  
00090 CHEN \$EISENE>IERZ \$QUALITE>ITSSTOL ZE KRE>IEN . DE+S \$ERFINDUNG HUET E+T ME>IGLECH GEMAACH , LE+TZEUEGERESCHT  
00100 A LOTHRE>NGESCHT \$EISENE>IERZ ZE VERSCHAFFEN , DAT VIRDRUN WE>INT DEN HE>IGE \$PHOSPHORGEHALT NE+T GEBRAUCHT  
00110 KONNT GIN . AN DE \$JORZE>NGTEN DONO HUET SECH ENG BEDA+ITEND \$STOLINDUSTRIE BEI ONS ENTUE>CKELT , DE>I OCZ  
00120 H D' \$GRE+NNUNG VUN ANEREN \$INDUSTRIEN E+NNERSTE+TIZT HUET . AN ENGEMS HUET DEN \$AUSBAU VUN \$EISENBUNN-Z  
00130 \$NETZ E SE>CHEREN \$TRANSPORTWEE GESCHAF AN ESOU ENG VUN DE WICHTEGSTE \$VIRAUSETZONGEN ERFE+LLT , FIR D' \$BLE>Z  
00140 I VUN EISER \$EKONOMIE . EN \$NIEWEPRODUKT VUN DER \$STOLPRODUKTION , D' \$THOMASKIEL , HUET EISER \$LANDZ  
00150 WIRTSCHAFT GEHOLLEF , FIR ME>I ZE PRODUZE>IEREN . \$ D' \$LANDWIRTSCHAFT HUET , NODDEEM SI RATIONALISE>IERT A MODZ  
00160 ERNISE>IERT WAR , HIR \$PRODUKTIVITE>IT GANZ SCHE>IN AN D' \$LUT GEDRIWEN : VUN 1870 BIS 1907 AS D' \$ZUEL VUN DZ  
00170 E \$LEIT , DE>I AN DER \$LANDWIRTSCHAFT ( D.H. AN \$PRIMA+RE-\$SECTEUR ) GESCHAFFT HUN , VUN !

TEXTLUX(TEXTLUX1) :

: \$JEAN-\$PAUL \$SCHNEIDER : . \$ !! D' \$REFORM VUN DER \$LE+TZEUEGER \$EKONOMIE ( 1. \$DEEL ) !! . \$ ! 1. \$ D' \$STRUKTUR VUN DER \$LE+T  
ZUEUEGER \$EKONOMIE : . \$ BIS AN D' \$NE+TT VUN 19TEN \$JOERHONDERT WAR \$LE+TZEUEGER EN AARNT \$LAND : E+T GOUF DEEMOLS KENG GRE>ISSER  
\$INDUSTRIE , DEN \$DE>NGSCHLEESCHTUNGSSECTEUR WAR GANZ KLENG . ONGEFE>IER 60/100 VUN DE \$LEIT HUN AN DER \$LANDWIRTSCHAFT GESCHAFFT ,  
WOUVUN DE>I MEESCHT E KLENGT \$AKOMMES HATEN , DAT KNAPPS DUERGONG FIR ZE LIEWEN , 22/100 HUN AN DER \$INDUSTRIE AN AM \$HANDWIERK GESC  
HAFFT , 18/100 AN \$HANDEL , DE FRA+IE \$BERUFFER AN DE \$SERVICER . \$ AM \$JOER 1877 HUET DEN ENGLISCHEN \$INGENIEUR \$GILCHRIST \$THOMAS  
ENG \$METHOD ERFONNT , FIR AUS PHOSPHORRA+ICHEN \$EISENE>IERZ \$QUALITE>ITSSTOL ZE KRE>IEN . DE+S \$ERFINDUNG HUET E+T ME>IGLECH GEMAACH  
, LE+TZEUEGERESCHT A LOTHRE>NGESCHT \$EISENE>IERZ ZE VERSCHAFFEN , DAT VIRDRUN WE>INT DEN HE>IGE \$PHOSPHORGEHALT NE+T GEDRAUCHT KONN  
T GIN . AN DE \$JORZE>NGTEN DONO HUET SECH ENG BEDA+ITEND \$STOLINDUSTRIE BEI ONS ENTUE>CKELT , DE>I OCH D' \$GRE+NNUNG VUN ANEREN \$IND  
USTRIEN E+NNERSTE+TIZT HUET . AN ENGEMS HUET DEN \$AUSBAU VUN \$EISENBUNN-\$NETZ E SE>CHEREN \$TRANSPORTWEE GESCHAF AN ESOU ENG VUN DE W  
ICHTEGSTE \$VIRAUSETZONGEN ERFE+LLT , FIR D' \$BLE>I VUN EISER \$EKONOMIE . EN \$NIEWEPRODUKT VUN DER \$STOLPRODUKTION , D' \$THOMASKIEL  
, HUET EISER \$LANDWIRTSCHAFT GEHOLLEF , FIR ME>I ZE PRODUZE>IEREN . \$ D' \$LANDWIRTSCHAFT HUET , NODDEEM SI RATIONALISE>IERT A MODERNI  
SE>IERT WAR , HIR \$PRODUKTIVITE>IT GANZ SCHE>IN AN D' \$LUT GEDRIWEN : VUN 1870 BIS 1907 AS D' \$ZUEL VUN DE \$LEIT , DE>I AN DER \$LAN  
DWIRTSCHAFT ( D.H. AN \$PRIMA+RE-\$SECTEUR ) GESCHAFFT HUN , VUN 60/100 OP 43,2/100 GEFALL , GE>INT 38,4/100 AN DER \$INDUSTRIE ( \$SEKU  
NDA+RE-\$SECTEUR ) AN 18,4/100 AN DEN SERVICER ( \$TERTIA+RE-\$SECTEUR ) AN \$JOER 1907 . \$ DE+S ENTUE>CKLUNG AS AM 20TEN \$JOERHONNERT U  
EIDERGANG AN AS CHARAKTERISTESCH FIR A

TEXTLUX1.SORTIEØ.CHAINES(CHAINÉ1) :

\*0006

0155

0022

0000

0000

0001 014

0000

0000

0000

00006001070001 \$

00006001080002 AM

00006001090003 \$JOER

00006001100004 1877

00006001110005 HUET

00006001120006 DEN

00006001130007 ENGESCHEN

00006001140008 \$INGENIEUR

00006001150009 \$GILCHRIST

00006001160010 \$THOMAS

00006001170011 ENG

00006001180012 \$METHOD

00006001190013 ERFONNT

00006001200014 ,

00006001210015 FIR

00006001220016 AUS

00006001230017 PHOSPHORRA\*ICHEN

00006001240018 \$EISENE>IERZ

00006001250019 \$QUALITE>ITSSTOL

00006001260020 ZE

00006001270021 KRE>IEN

00006001280022 .

TEXLUX1.SORTIE1.BASE :

\*0006

0155

0022

0000

0000

0001 014

0000

0000

0000

00006001070001 \$

00006001080002 AM

00006001090003 \$JOER

00006001100004 1877

00006001110005 HUE T

00006001120006 DEN

00006001130007 ENGESCHEN

00006001140008 \$INGENIEUR

00006001150009 \$GILCHRIST

00006001160010 \$THOMAS

00006001170011 ENG

00006001180012 \$METHOD

00006001190013 ER FONN T

00006001200014 ,

00006001210015 FIR

00006001220016 AUS

00006001230017 PHOSPHORRA\*ICHEN

00006001240018 \$EISENE>IERZ

00006001250019 \$QUALITE>ITSSTOL

00006001260020 ZE

00006001270021 KRE>IE N

00006001280022 .

TEXLUX1.SORTIE2.BASE1 :

HUE

ERFONN

KRE>IE

Pour un texte anglais :

TEXANG(TEXANG1) :

00010 !! FUEL-CELL POWER PLANTS !! . \$ COVER THE NEXT DECADE ELECTRIC UTILITIES IN THE U.S. WILL REQUIRE POWER X  
 00020 GENERATORS THAT FULFILL CERTAIN UNUSUAL REQUIREMENTS : HIGH EFFICIENCY , LOW EMISSION OF POLLUTANTS , QUIET OPERA  
 00030 TION AND QUICK INSTALLATION . \$THE GENERATORS MUST BE ABLE TO SUPPLY ELECTRICITY IN URBAN AREAS WHERE CONVENTIO  
 00040 NAL GENERATORS WOULD BE UNACCEPTABLE FOR ENVIRONMENTAL REASONS . \$A LIKELY CANDIDATE IS THE FUEL CELL . \$ \$A FUEL  
 00050 L CELL CONVERTS THE CHEMICAL ENERGY OF A FUEL INTO ELECTRICITY DIRECTLY , WITH NO INTERMEDIATE COMBUSTION CYCLE X  
 00060 . \$FOR A DEVICE THAT WAS INVENTED IN 1839 ( BY \$SIR \$WILLIAM \$GROVE , A \$BRITISH JURIST WHO MADE NOTEWORTHY CONTRI  
 00070 BUTIONS TO SCIENCE ) THE FUEL CELL HAS TAKEN A LONG TIME TO COME INTO ITS OWN . \$IT DID FURNISH POWER FOR THE X  
 00080 \$GEMINI AND \$APOLLO SPACECRAFT , BUT THAT WAS AN EXOTIC AND EXPENSIVE APPLICATION . \$NOW THE FUEL CELL , VASTLY X  
 00090 IMPROVED OVER THE SPACECRAFT VERSIONS AND IN ASSEMBLIES 1,000 TIMES LARGER THAN THE ONES CARRIED BY THE SPACECRA  
 00100 FT , APPEARS TO HAVE REACHED A STAGE WHERE IT CAN MAKE A SIGNIFICANT CONTRIBUTION TO A NATION'S SUPPLY OF ELECTRIC  
 00110 ICITY . \$INDEED , IT IS VIEWED BY THE ELECTRIC-UTILITY INDUSTRY IN THE U.S. AS AN IMPORTANT ALTERNATIVE FOR MEET  
 00120 ING THE LOAD GROWTH EXPECTED OVER THE NEXT TWO DECADES AND FOR DOING SO IN A WAY THAT IS ENVIRONMENTALLY ACCEPTA  
 00130 BLY AND CONSERVES FUEL . \$ \$IN THE SHORT RUN THE HOPE IS THAT FUEL-CELL SYSTEMS CAN BE DEPLOYED IN THE 1980'S X  
 00140 FOR HANDLING PEAK LOADS AND FOR WHAT THE UTILITY INDUSTRY CALLS LOAD FOLLOWING . \$PEAKING GENERATORS ARE STARTED X  
 00150 ONLY WHEN THE NEED FOR ADDITIONAL POWER IS TEMPORARILY HIGH . \$LOAD-FOLLOWING GENERATORS ARE STARTED DAILY AND RUN  
 00160 UN MOST OF THE TIME TO COPE WITH!

!! FUEL-CELL POWER PLANTS !! . \$ COVER THE NEXT DECADE ELECTRIC UTILITIES IN THE U.S. WILL REQUIRE POWER GENERATORS THAT FULF  
 ILL CERTAIN UNUSUAL REQUIREMENTS : HIGH EFFICIENCY , LOW EMISSION OF POLLUTANTS , QUIET OPERATION AND QUICK INSTALLATION . \$THE GENER  
 RATORS MUST BE ABLE TO SUPPLY ELECTRICITY IN URBAN AREAS WHERE CONVENTIONAL GENERATORS WOULD BE UNACCEPTABLE FOR ENVIRONMENTAL REASO  
 NS . \$A LIKELY CANDIDATE IS THE FUEL CELL . \$ \$A FUEL CELL CONVERTS THE CHEMICAL ENERGY OF A FUEL INTO ELECTRICITY DIRECTLY , WITH N  
 O INTERMEDIATE COMBUSTION CYCLE . \$FOR A DEVICE THAT WAS INVENTED IN 1839 ( BY \$SIR \$WILLIAM \$GROVE , A \$BRITISH JURIST WHO MADE NOT  
 EORTHY CONTRIBUTIONS TO SCIENCE ) THE FUEL CELL HAS TAKEN A LONG TIME TO COME INTO ITS OWN . \$IT DID FURNISH POWER FOR THE \$GEMINI  
 AND \$APOLLO SPACECRAFT , BUT THAT WAS AN EXOTIC AND EXPENSIVE APPLICATION . \$NOW THE FUEL CELL , VASTLY IMPROVED OVER THE SPACECRAFT  
 VERSIONS AND IN ASSEMBLIES 1,000 TIMES LARGER THAN THE ONES CARRIED BY THE SPACECRAFT , APPEARS TO HAVE REACHED A STAGE WHERE IT CA  
 N MAKE A SIGNIFICANT CONTRIBUTION TO A NATION'S SUPPLY OF ELECTRICITY . \$INDEED , IT IS VIEWED BY THE ELECTRIC-UTILITY INDUSTRY IN T  
 HE U.S. AS AN IMPORTANT ALTERNATIVE FOR MEETING THE LOAD GROWTH EXPECTED OVER THE NEXT TWO DECADES AND FOR DOING SO IN A WAY THAT  
 IS ENVIRONMENTALLY ACCEPTABLE AND CONSERVES FUEL . \$ \$IN THE SHORT RUN THE HOPE IS THAT FUEL-CELL SYSTEMS CAN BE DEPLOYED IN THE 198  
 0'S FOR HANDLING PEAK LOADS AND FOR WHAT THE UTILITY INDUSTRY CALLS LOAD FOLLOWING . \$PEAKING GENERATORS ARE STARTED ONLY WHEN THE N  
 EED FOR ADDITIONAL POWER IS TEMPORARILY HIGH . \$LOAD-FOLLOWING GENERATORS ARE STARTED DAILY AND RUN MOST OF THE TIME TO COPE WITH WITH  
 ILY SWINGS IN THE LOAD ; THEY MAY BE SHUT !

TEXANG1.SORTIE6.CHAINES(CHAINED) :

.TEXANG1.SORTIE1.BASE :

\*0002  
0210  
0034  
0000  
0001 020  
0002 023 028  
0000  
0000  
0000  
00002000000001 \$  
00002000090002 \$OVER  
00002000100003 THE  
00002000110004 NEXT  
00002000120005 DECADE  
00002000130006 ELECTRIC  
00002000140007 UTILITIES  
00002000150008 IN  
00002000160009 THE  
00002000170010 \$U.\$S.  
00002000180011 WILL  
00002000190012 REQUIRE  
00002000200013 POWER  
00002000210014 GENERATORS  
00002000220015 THAT  
00002000230016 FULFILL  
00002000240017 CERTAIN  
00002000250018 UNUSUAL  
00002000260019 REQUIREMENTS  
00002000270020 :  
00002000280021 HIGH  
00002000290022 EFFICIENCY  
00002000300023 ,  
00002000310024 LOW  
00002000320025 EMISSION  
00002000330026 OF  
00002000340027 POLLUTANTS  
00002000350028 ,  
00002000360029 QUIET  
00002000370030 OPERATION  
00002000380031 AND  
00002000390032 QUICK  
00002000400033 INSTALLATION  
00002000410034 .

Pour un texte en français :

.TEXTFRA(TEXTFRA1) :

00020 ARD . EN TANT QUE DISCIPLINE MATHÉMATIQUE , ELLE NE PEUT SE DÉVELOPPER D'UNE MANIÈRE RIGoureuse QUE SI ELLE  
 00030 SE FONDE SUR UN SYSTÈME DE DÉFINITIONS ET D'AXIOMES BIEN EXPLICITES . HISTORIQUEMENT LA FORMULATION D'UNE  
 00040 TELLE BASE AXIOMATIQUE ET L'ÉLABORATION MATHÉMATIQUE DE LA THÉORIE REMONTENT AUX ANNÉES 1930 . CE N'EST  
 00050 EN EFFET QU'À CETTE ÉPOQUE QUE LA THÉORIE DE LA MESURE ET DE L'INTEGRATION S'EST TROUVÉE SUFFISAMMENT DÉ  
 00060 VÉLOPPÉE SUR DES ESPACES MÉTRIQUES POUR FOURNIR À LA THÉORIE DES PROBABILITÉS SES DÉFINITIONS FONDAMENT  
 00070 NTALES EN MÊME TEMPS QUE SON PLUS PUISSANT OUTIL DE DÉVELOPPEMENT . \$ DEPUIS LORS , LES NOMBREUSES RECHERCHES  
 00080 S'ENTREPREISES DANS LE DOMAINE THÉORIQUE COMME DANS LE DOMAINE CONCRET , EN PARTICULIER CELLES UTILISANT LES ES  
 00090 PACES FONCTIONNELS , N'ONT FAIT QUE CONFIRMER LES LIENS ÉTROITS ÉTABLIS ENTRE LA THÉORIE DES PROBABILITÉS  
 00100 ET LA THÉORIE DE LA MESURE . CES LIENS SONT D'AILLEURS TELLEMENT ÉTROITS QUE CERTAINS AUTEURS N'ONT VO  
 00110 ULU VOIR DANS LA THÉORIE DES PROBABILITÉS QU'UN PROLONGEMENT ( AUSSI IMPORTANT FUT-IL ! ) DE LA THÉORIE  
 00120 DE LA MESURE . \$ EN TOUT CAS , IL EST ACTUELLEMENT IMPOSSIBLE DE POURSUIVRE DES ÉTUDES APPROFONDIES DE THÉ  
 00130 ORIE DES PROBABILITÉS ET DE STATISTIQUE MATHÉMATIQUE SANS UTILISER CONSTamment LA THÉORIE DE LA MESURE  
 00140 . \$ MOINS DE SE LIMITER À L'ÉTUDE DE MODELES PROBABILISTES TRÈS ÉLÉMENTAIRES ET DE S'INTERDIRE EN PARTI  
 00150 CULIER DE CONSIDÉRER DES FONCTIONS ALÉATOIRES . ON ESSAIE PARFOIS , IL EST VRAI , DE TRAITER LES PROBLÈMES  
 00160 DE CONVERGENCE DU CALCUL DES PROBABILITÉS DANS LE CADRE LIMITÉ DE L'ÉTUDE DES FONCTIONS DE RÉPARTITION :  
 00170 CETTE MANIÈRE DE PROCÉDER NE FOURNIT QU'UNE FAUSSE SIMPLIFICATION DE LA QUESTION ET DISSIMULE DE PLUS LES  
 00180 ASPECTS INTUITIFS DE CES PROBLÈMES . \$ CE LIVRE REPRODUIT L'É

Cette série de modifications, qui part du texte et aboutit à .SORTIE2.BASE23 pour l'étude des mots composés, inclut en cours de route les résultats de .PROLOG et de .VERBAL.

#### 4.3.2 L'analyse lexicale et morphologique

##### 4.3.2.1 Les mots outils et la partie numérale (.PROLOG)

Après .LECTEXT et .DECPHR, .PROLOG code l'ensemble des mots-outils, simples ou disjoints, rangés dans deux listes ouvertes. (.FIXALL(OUTALL) et .FIXALL(EXPRALL) ainsi que la partie numérale.

##### 4.3.2.1.1 Les mots outils atomiques (.FIXALL(OUTALL) )

Ce fichier partitionné contient 842 enregistrements, composés d'un mot-outil simple (atomique) associé à un code de trois caractères.

La classification, propre au système, n'est que provisoire. Elle devrait évoluer au fil des versions et lors de l'élaboration du logiciel de transfert, dans le sens d'une simplification. Les ambiguïtés sont assez nombreuses mais dépassent rarement le nombre de 3 (d'où la taille du code utilisé).

Les unités qui sortent de ce cadre sont traitées à part (\*ZU...).

Code (pour les deux listes) suivi d'un exemple:

A	adverbe ebenfalls
B	adverbe pronominal daneben
C	partie numérale dritte (cf. 4.3.5.3)
D	adverbe interrogatif wo
E	adverbe relatif wobei
F	locution adverbiale ab und zu
G	préposition + déterminatif im
H	adjectif invariable anderthalb
I	article défini der
J	article indéfini ein
K	pronom relatif der
L	outil de la comparaison wie
M	conjonction de coordination und
N	conjonction de subordination daß
O	particule séparable auf
P	préposition auf
Q	posposition zuliebe
R	pré/postposition gegenüber
S	indéfini wenig
T	interrogatif was
U	possessif sein
V	démonstratif dieser
W	réfléchi sich
X	pronom invariable seinesgleichen
Y	pronom personnel er
Z	adjectif lustig

La distinction entre adverbe (A), adverbe pronominal (B) et locution adverbiale (F) n'est pas exploitée au cours de l'analyse syntaxique, telle que nous l'avons construite. De même, pourrait-on réduire les formes contractées préposition-déterminatif ou articles définis-articles indéfinis à une classe sans que le système en souffre.

C'est le fonctionnement du mot-outil qui importe. Cela signifie, par exemple, que la classe des postpositions ne pourra jamais se substituer à la classe des prépositions.

Nous avons maintenu ces classes artificielles, à l'issue d'un cheminement sinueux, au long duquel la recherche pragmatique a souvent bousculé la théorie linguistique.

Dans cet ordre d'idées, nous avons constaté à maintes reprises que le traitement automatique de longs textes (80 000 mots minimum) ne s'accommodait pas des modèles élaborés par telle ou telle école. Les méthodes de l'analyse classique que nous ont enseignées nos professeurs de latin ou de grec ne sont pas très différentes, parfois, d'algorithmes présentés.

.FIXALL(OUTALL) :

00010 .FIXALL(OUTALL)  
00020 00042  
00030 00052  
00040 00340  
00050 APO AB  
00060 A00 ABENDS  
00070 MA0 ABER  
00080 A00 ABERNALS  
00090 APO ABSEITS  
00100 A00 ABWA\*RTS



00110	F00	ABZU*GLICH	00770	A00	AUS*EN	01430	B00	BARUNTER
00120	DS0	ALL	00780	F00	AUS*ER	01440	IVK	DAS
00130	DS0	ALLE	00790	A00	AUS*ERBEM	01450	A00	DASELBST
00140	A00	ALLEDEM	00800	AP0	AUS*ERHALB	01460	V00	DASJENIGE
00150	DS0	ALLEM	00810	A00	AUS*ERSTANDE	01470	V00	DASSELBE
00160	A00	ALLEMAL	00820	AF0	AUS	01480	N00	DAS*
00170	DS0	ALLEN	00830	A00	AUSEINANDER	01490	B00	DAVON
00180	A00	ALLENFALLS	00840	A00	AUSNAHMSWEISE	01500	B00	DAVOR
00190	A00	ALLENHALBEM	00850	A00	BALD	01510	B00	DAWIDER
00200	DS0	ALLER	00860	A00	BALDIGST	01520	B00	DAZU
00210	A00	ALLERART	00870	A00	BALDNO*GLICHST	01530	B00	DAZUISCHEN
00220	A00	ALLERDINGS	00880	A00	BAS*	01540	U00	DEIN
00230	H00	ALLERHAND	00890	A00	BEDINGUNGSWEISE	01550	U00	DEINE
00240	H00	ALLERLEI	00900	PA0	BEI	01560	U00	DEINER
00250	A00	ALLERSEITS	00910	DS0	BEIDE	01570	U00	DEINEN
00260	A00	ALLERWA*RTS	00920	A00	BEIDEMAL	01580	U00	DEINER
00270	DS0	ALLES	00930	H00	BEIDERLEI	01590	U00	DEINES
00280	A00	ALLESAMT	00940	A00	BEIDERSEITS	01600	X00	DEINESGLEICH*EN
00290	A00	ALLEWEGE	00950	DS0	BEIDES	01610	IVK	DEN
00300	A00	ALLEWEIL	00960	A00	BEIEINANDER	01620	A00	DEKENTGENEN
00310	A00	ALLEZEIT	00970	G00	BEIM	01630	A00	DERGEGENUEBER
00320	NAM	ALLEIN	00980	A00	BEINAHE	01640	A00	DERGEMAS*E*
00330	DS0	ALLE	00990	A00	BEISAMMEN	01650	A00	DEHMACH
00340	A00	ALLHIER	01000	A00	BEISEITE	01660	A00	DEHNA*CHST
00350	A00	ALLSEITS	01010	A00	BEISPIELSWEISE	01670	V00	DENSELBEH
00360	A00	ALLU*BERALL	01020	A00	BEREITS	01680	A00	DEHZUFOLGE
00370	A00	ALLZU	01030	A00	BESONDERS	01690	IVK	DEN
00380	A00	ALLZUMAL	01040	A00	BESTENFALLS	01700	VK0	DENEN
00390	A00	ALLZUSEHR	01050	A00	BESTENS	01710	NAM	DENN
00400	A00	ALLZUVIEL	01060	P00	BETREFFS	01720	A00	DENNOCH
00410	*01	ALS	01070	N00	BEVOR	01730	V00	DENSELBEH
00420	A00	ALSBALD	01080	CP0	BEZU*GLICH	01740	IVK	DER
00430	A00	ALSDANN	01090	F00	BINNEN	01750	A00	DERART
00440	AM0	ALSO	01100	PH0	BIS	01760	VK0	DEREN
00450	GL0	AM	01110	A00	BISHER	01770	VK0	DERER
00460	PA0	AN	01120	AN0	DA	01780	B00	DERGLEICHEN
00470	A00	ANBEI	01130	B00	DABEI	01790	A00	DEREINST
00480	S00	ANDEREM	01140	B00	DADURCH	01800	V00	DERJENIGE
00490	S00	ANDEREN	01150	B00	DAFU*R	01810	A00	DERMAS*EN
00500	S00	ANDERER	01160	B00	DAGEGEN	01820	V00	DERSELBE
00510	S00	ANDERES	01170	A00	DAHEIM	01830	V00	DERSELBEH
00520	S00	ANDERE	01180	A00	DAHER	01840	A00	DERWEIL
00530	A00	ANDERNFALLS	01190	A00	DAHIN	01850	A00	DERZEIT
00540	A00	ANDERNORTS	01200	A00	DAHINAB	01860	I00	DES
00550	A00	ANDERNTAGS	01210	A00	DAHINAUF	01870	A00	DESFALLS
00560	A00	ANDERS	01220	A00	DAHINEIN	01880	AD0	DESGLEICHEN
00570	A00	ANDERERSEITS	01230	A00	DAHINTEN	01890	A00	DESHALB
00580	A00	ANDRERSEITS	01240	B00	DAHINTER	01900	V00	DESJENIGEN
00590	A00	ANDERSRUM	01250	A00	DAHINU*BER	01910	V00	DESSELBEH
00600	A00	ANDERSWO	01260	A00	DAHINUNTER	01920	VK0	DESSEN
00610	A00	ANDERSWOHER	01270	A00	DAFALS	01930	A00	DESSENHALBEM
00620	A00	ANDERSWOHIN	01280	BN0	DAMIT	01940	A00	DESSENTUESEN
00630	H00	ANDERTHALB	01290	B00	DANACH	01950	A00	DESTO
00640	A00	ANEINANDER	01300	B00	DANEBEN	01960	A00	DESWEGEN
00650	AP0	ANFANGS	01310	A00	DANIEDER	01970	Y00	DICH
00660	F00	ANGESICHTS	01320	P00	DANK	01980	IVK	DIE
00670	P00	ANHAND	01330	P00	DANN	01990	V00	DIES
00680	P00	ANLA*S*LICH	01340	U00	DAR	02000	V20	DIESE
00690	G00	ANS	01350	B00	DARAN	02010	V00	DIESELBEH
00700	F00	ANSTATT	01360	B00	DARAUF	02020	V00	DIESELBE
00710	P00	ANSTELLE	01370	A00	DARAUFHIN	02030	V00	DIESEM
00720	A00	AUCH	01380	B00	DARAUS	02040	V00	DIESEN
00730	PA0	AUF	01390	A00	DAREIN	02050	V00	DIESEN
00740	AD0	AUFEINANDER	01400	B00	DARIN	02060	V00	DIESES
00750	P00	AUFGRUND	01410	B00	DARU*BER	02070	A00	DIESFALLS
00760	G00	AUFS	01420	B00	DARUM	02080	A00	DIESMAL

02090	AF0	DIESSEITS	02750	JS0	EINES	03410	A00	HERUNTER
02100	Y00	DIR	02760	C00	EINHER	03420	A00	HERVOR
02110	AN0	DOCH	02770	V00	EINIGE	03430	A00	HERZU
02120	A00	DORI	02780	V00	EINIGEN	03440	A00	HEUTE
02130	A00	DORTHER	02790	V00	EINIGEN	03450	A00	HEUTZUTAGE
02140	A00	DORTHERAUS	02800	A00	EINIGERMASS*EN	03460	A00	HIER
02150	A00	DORTHIN	02810	V00	EINIGES	03470	A00	HIERBEI
02160	A00	DORTHINAB	02820	AF0	EINSCHLIES*FLICH	03480	A00	HIERFU*ER
02170	A00	DORTHINAUF	02830	A00	EINST	03490	A00	HIERHER
02180	A00	DORTHINAUS	02840	A00	EMPOR	03500	A00	HIERHIN
02190	A00	DORTHINUNTER	02850	R00	ENTGEGEN	03510	A00	HIERMIT
02200	A00	DORTZULANDE	02860	AR0	ENTLANG	03520	A00	HIERSELEST
02210	B00	DRAN	02870	CR0	ENTSPRECHEND	03530	A00	HIERZU
02220	B00	DRAUF	02880	M00	ENTWEDER	03540	A00	HIN
02230	B00	DRAUS	02890	Y00	ER	03550	A00	HINAB
02240	A00	DRAUS*EN	02900	A00	ERSTENS	03560	A00	HINAUF
02250	B00	DREIN	02910	AF0	ERST	03570	A00	HINAN
02260	B00	DRIN	02920	Y00	ES	03580	A00	HINAUS
02270	A00	DRINNEN	02930	S00	ETLICHE	03590	A00	HINDURCH
02280	A00	DRITTENS	02940	S00	ETLICHEN	03600	A00	HINEIN
02290	A00	DROBEN	02950	S00	ETLICHER	03610	A00	HINSICHTLICH
02300	A00	DRU*EBEN	02960	A00	ETWA	03620	A00	HINTAN
02310	B00	DRUM	02970	XA0	ETWAS	03630	A00	HINTEN
02320	B00	DRUNTEN	02980	Y00	EUCH	03640	A00	HINTERAN
02330	B00	DRUNTER	02990	YU0	EUER	03650	A00	HINTENDREIN
02340	Y00	DU	03000	U00	EUERN	03660	A00	HINTENHERAUS
02350	PA0	DURCH	03010	U00	EURE	03670	A00	HINTENHIN
02360	A00	DURCHAUS	03020	U00	EUREN	03680	A00	HINTENHIN
02370	AO0	DURCHEINANDER	03030	U00	EUREN	03690	A00	HINTENHIN
02380	G00	DURCHS	03040	U00	EURER	03700	P00	HINTER
02390	A00	DURCHWEG	03050	A00	EURERSEITS	03710	P00	HINTERAN
02400	A00	DURCHWEGS	03060	U00	EURES	03720	P00	HINTERS
02410	A00	EBENDA	03070	X00	EURESGLEICHEN	03730	A00	HINU*BER
02420	A00	EBENDAHER	03080	N00	FALLS	03740	A00	HINUNTER
02430	A00	EBENDAHIN	03090	A00	FAST	03750	A00	HINWEG
02440	A00	EBENDARUM	03100	A00	FERNAB	03760	A00	HINZU
02450	V00	EBENDER	03110	AZ0	FEST	03770	A00	HO*CHSTENS
02460	V00	EBENDIE	03120	A00	FEKNERHIN	03780	Y00	ICH
02470	V00	EBENDAS	03130	A00	FOLGENDERWEISE	03790	Y00	IHM
02480	V00	EBENDERSSELBE	03140	A00	FORT	03800	Y00	IHN
02490	V00	EBENDIESELBE	03150	A00	FORTAB	03810	Y00	IHNEN
02500	V00	EBENDASSELBE	03160	A00	FORTAN	03820	YU0	IHR
02510	A00	EBENDESHALB	03170	A00	FURTHIN	03830	U00	IHRE
02520	A00	EBENDESWEGEN	03180	P00	FUR*ER	03840	U00	IHREN
02530	A00	EBENFALLS	03190	G00	FU*RS	03850	U00	IHREN
02540	A00	EBENS0	03200	F00	GEGEN	03860	YU0	IHREN
02550	V00	EBENSOLCHER	03210	A00	GEGENEINANDER	03870	A00	IHRERSEITS
02560	V00	EBENSOLCHE	03220	ADR	GEGENU*BER	03880	U00	IHRES
02570	V00	EBENSOLCHES	03230	AZ0	GEGENU*RTIG	03890	X00	IHRESGLEICHEN
02580	AN0	EHE	03240	R00	GENA*S*	03900	G00	IN
02590	A00	EHEDEM	03250	A00	GENUG	03910	A00	INNER
02600	A00	EHEHALS	03260	A00	GERN	03920	A00	INNERBAR
02610	A00	EHER	03270	A00	GERNE	03930	A00	INNERFORT
02620	A00	EIGENS	03280	A00	GESCHWEIGE	03940	A00	INNERHIN
02630	S00	EIGENEM	03290	C00	HALBER	03950	A00	INNERZU
02640	S00	EIGENEN	03300	A00	HER	03960	A00	INEINANDER
02650	S00	EIGENER	03310	A00	HERAB	03970	P00	IN
02660	S00	EIGENES	03320	A00	HERAN	03980	AN0	INDEN
02670	S00	EIGENE	03330	A00	HERAUF	03990	P00	INFOLGE
02680	A00	EILENDS	03340	A00	HERAUS	04000	A00	INFOLGEBEISEN
02690	JS0	EIN	03350	A00	HERBEI	04010	P00	INNITTEN
02700	JS0	EINE	03360	A00	HEREIN	04020	A00	INNE
02710	JS0	EINEN	03370	A00	HERFU*ER	04030	A00	INNEN
02720	JS0	EINEN	03380	A00	HERNACH	04040	AF0	INNERHALB
02730	JS0	EINER	03390	A00	HERU*BER	04050	G00	INS
02740	A00	EINERSEITS	03400	A00	HERUM	04060	A00	INSBESONDERE

14070	A00	INSGEMEIN	04732	A00	LA*NGELANG	05390	A00	NAHEBEI
14080	A00	INSGEMEIN	04740	PA0	LA*NGS	05400	A00	NAHEZU
14090	A00	INSGESANT	04750	A00	LA*NGST	05410	A00	NAHENS
14100	AN0	INSOFERN	04760	A00	LA*NGSTENS	05420	AZ0	NATURLICH
14110	AN0	INSOWEIT	04770	CAF	LAUT	05430	P00	NEBEN
14120	A00	INZWISCHEN	04780	A00	LETZTERS	05440	A00	NEBENAN
14130	A00	IRGEND	04790	A00	LINKS	05450	A00	NEBENEINANDER
14140	S00	IRGENDEIN	04800	CA0	LOS	05460	A00	NEBENHER
14150	S00	IRGENDEINE	04810	S00	MAN	05470	P00	NEBST
14160	S00	IRGENDEINEM	04820	S00	MANCH	05480	A00	NEIN
14170	S00	IRGENDEINEN	04830	S00	HANCHE	05490	A00	NEUERDINGS
14180	S00	IRGENDEINER	04840	S00	HANCHEM	05500	A00	NEUESTENS
14190	S00	IRGENDEINES	04850	S00	HANCHEM	05510	N00	NICHT MEHR
14200	S00	IRGENDWELCH	04860	S00	HANCHEM	05520	A00	NICHT
14210	S00	IRGENDWELCHE	04870	S00	HANCHES	05530	S00	NICHTS
14220	S00	IRGENDWELCHEM	04880	A00	HANCMAL	05540	A00	NIEMALS
14230	S00	IRGENDWELCHEN	04890	P00	HANGELS	05550	S00	NIEHANO
14240	S00	IRGENDWELCHER	04900	A00	HEHR	05560	S00	NIEHANGEN
14250	S00	IRGENDWELCHES	04910	S00	HEHRERE	05570	S00	NIEHANGEN
14260	S00	IRGENDWAS	04920	S00	HEHREREN	05580	S00	NIEHANGS
14270	A00	IRGENDWIE	04930	S00	HEHRERER	05590	AN0	NOCH
14280	A00	IRGENDWO	04940	S00	HEHRERES	05600	A00	NOCHMALS
14290	A00	IRGENDWANN	04950	A00	HEHRMALS	05610	A00	NUN
14300	A00	JA	04960	U00	HEIN	05620	A00	NUR
14310	A00	JAUOHL	04970	U00	HEINE	05630	AFN	03
14320	A00	JAHRAUS	04980	U00	HEINEM	05640	A00	OBEIN
14330	A00	JAHREIN	04990	U00	HEINEN	05650	A00	OBEINAN
14340	A00	JE	05000	UY0	HEINER	05660	A00	OBEINAUF
14350	A00	JEHER	05010	U00	HEINES	05670	A00	OBEINDRAUF
14360	A00	JEWELLS	05020	X00	HEINESGLEICHEN	05680	A00	OBEINDREIN
14370	S00	JEDE	05030	A00	HEINETWEGEN	05690	A00	OBEINHIN
14380	S00	JEDER	05040	A00	HEINETHALBEN	05700	P00	OBERHALB
14390	S00	JEDERMANN	05050	A00	HEISTENS	05710	N00	OEGLEICH
14400	S00	JEDEN	05060	Y00	HICH	05720	N00	OESCHON
14410	S00	JEDEN	05070	A00	HINDESTENS	05730	N00	OBWOHL
14420	S00	JEDES	05080	A00	HINUS	05740	N00	ODER
14430	A00	JEDERZEIT	05090	Y00	HIR	05750	PN0	OHNE
14440	A00	JEDESMAL	05100	O00	MITAN	05760	A00	OHNEDEM
14450	A00	JEDUCH	05110	O00	HITU*BER	05770	A00	OHNEHIN
14460	S00	JEGLICH	05120	O00	HITEIN	05780	A00	OHNEHIES
14470	S00	JEGLICHE	05130	O00	HITUNTER	05790	A00	OHNGEFA*HR
14480	S00	JEGLICHEN	05140	P00	HIT	05800	P00	PER
14490	S00	JEGLICHEN	05150	A00	HITTAGS	05810	A00	PLUS
14500	S00	JEGLICHER	05160	P00	HITTELS	05820	P00	PRO
14510	S00	JEGLICHES	05170	A00	HITTEN	05830	A00	RECHTS
14520	A00	JEHALS	05180	A00	HITTENDREIN	05840	AP0	SANT
14530	S00	JEMAND	05190	A00	HITTENDRIN	05850	A00	SCHLIES*LICHT
14540	S00	JEMANDEM	05200	A00	HITTENDRUNTER	05860	A00	SCHON
14550	S00	JEMANDEN	05210	A00	HITTENDURCH	05870	A00	SEHR
14560	S00	JEMANDS	05220	A00	HITTERNACHTS	05880	*22	SEIN
14570	V00	JENE	05230	A00	HITTLERWEILE	05890	U00	SEINE
14580	V00	JENEM	05240	P00	HITSAHT	05900	U00	SEINER
14590	V00	JENER	05250	A00	NO*GLICHST	05910	U00	SEINER
14600	V00	JENES	05260	A00	NO*GLICHENFALLS	05920	UY0	SEINER
14610	P00	JENSEITS	05270	A00	NORGEN	05930	A00	SEINERSEITS
14620	A00	JETZT	05280	A00	NORGENS	05940	X00	SEINESGLEICHEN
14630	A00	KAUH	05290	RA0	NACH	05950	U00	SEINES
14640	S00	KEIN	05300	N00	NACHDEM	05960	A00	SEINETWEGEN
14650	S00	KEINE	05310	A00	NACHHER	05970	A00	SEINETHALBEN
14660	S00	KEINEM	05320	A00	NACHMALS	05980	A00	SEINETWELLEN
14670	S00	KEINEN	05330	A00	NACHMITTAGS	05990	PN0	SEIT
14680	S00	KEINER	05340	ACP	NA*CHST	06000	A00	SEITAE
14690	S00	KEINES	05350	A00	NA*CHSTEN	06010	AN0	SEITDER
14700	A00	KEINMAL	05360	A00	NA*CHSTENS	06020	A00	SEITHER
14710	P00	KRAFT	05370	A00	NACHTS	06030	A00	SEITLINGS
14720	AZ0	LANGE	05380	A00	NACHTSU*BER	06040	P00	SEITENS

06050	A00	SELBST	06710	U00	UNSEREN	07370	R00	WEGEN
06060	W00	SICH	06720	U00	UNSEREN	07380	A00	WEITAB
06070	Y00	SIE	06730	U00	UNSERER	07390	A00	WEITAB
06080	*03	SO	06740	U00	UNSERES	07400	A00	WEITHER
06090	AN0	SODALD	06750	A00	UNSERERSEITS	07410	A00	WEITHIN
06100	A00	SODANN	06760	X00	UNSERESGLEICHEN	07420	TK0	WELCH
06110	A00	SOEBEN	06770	A00	UNSERSEITS	07430	TK0	WELCHE
06120	N00	SOFERN	06780	X00	UNSRRESGLEICHEN	07440	TK0	WELCHER
06130	A00	SOFORT	06790	U00	UNSRE	07450	TK0	WELCHEN
06140	A00	SOGAR	06800	U00	UNSREM	07460	TK0	WELCHER
06150	A00	SOGLEICH	06810	U00	UNSKEN	07470	TK0	WELCHES
06160	A00	SOHIN	06820	U00	UNSRER	07480	A00	WELCHERLEI
06170	N00	SOLANGE	06830	U00	UNSRIS	07490	TK0	WER
06180	S00	SOLCHEM	06840	A00	UNTEN	07500	A00	WENIG
06190	S00	SOLCHEN	06850	A00	UNTENAN	07510	TK0	WEN
06200	S00	SOLCHER	06860	A00	UNTENHER	07520	S00	WENIGEN
06210	S00	SOLCHES	06870	A00	UNTENHIN	07530	S00	WENIGEN
06220	S00	SOLCHE	06880	CP0	UNTER	07540	S00	WENIGER
06230	A00	SOHIT	06890	P00	UNTERHALB	07550	S00	WENIGES
06240	A00	SONACH	06900	P00	UNTERN	07560	A00	WENIGST
06250	A00	SONDERN	06910	P00	UNTERN	07570	A00	WENIGSTENS
06260	CAP	SONDER	06920	P00	UNTERS	07580	N00	WENN
06270	A00	SONST	06930	AP0	UNWEIT	07590	N00	WENNGLEICH
06280	A00	SONSTUIE	06940	P00	VERNITTELS	07600	N00	WENN SCHON
06290	A00	SONSTUD	06950	P00	VERNOEIGE	07610	TK0	WER
06300	A00	SOOFT	06960	A00	VIELMEHR	07620	D00	WESHALE
06310	AN0	SOSEHR	06970	A00	VIELLEICHT	07630	L00	WESSEN
06320	N00	SOVIEL	06980	S00	VIELER	07640	P00	WIDER
06330	AN0	SOWEIT	06990	ZS0	VIELEN	07650	DL0	VIE
06340	AN0	SOWENIG	07000	S00	VIELE	07660	A00	WIEDER
06350	N00	SOWIE	07010	AZ0	VIEL	07670	A00	WIEDERUN
06360	A00	SOWIESO	07020	A00	VOLLAUF	07680	A00	WIEFERN
06370	N00	SOWOHL	07030	A00	VOLLENDS	07690	A00	WIESO
06380	A00	SOZUSAGEN	07040	G00	VON	07700	D00	WIEVIEL
06390	FN0	STATT	07050	P00	VON	07710	D00	WIEVIELMAL
06400	A00	STETS	07060	A00	VONEINANDER	07720	D00	WIEWEIT
06410	A00	TAGAU	07070	AP0	VOR	07730	N00	WIEWOHL
06420	A00	TAGEIN	07080	A00	VORAB	07740	Y00	WIK
06430	AZ0	TATSAEHLICH	07090	A00	VORAN	07750	DE0	WO
06440	A00	TEILS	07100	A00	VORAU	07760	A00	WOANDERS
06450	P00	TROTZ	07110	A00	VORAU	07770	NE0	WOBEI
06460	AN0	TROTZDEN	07120	P00	VORBEHALTLICH	07780	DE0	WOBUCH
06470	A00	U*BERALL	07130	A00	VORBEI	07790	N00	WOERN
06480	AP0	U*BER	07140	A00	VORDEM	07800	AE0	WOFUER
06490	A00	U*BERAUS	07150	A00	VOREINANDER	07810	AE0	WOGEGEN
06500	A00	U*BERDAS	07160	A00	VORERST	07820	DE0	WOHER
06510	A00	U*BERDEN	07170	A00	VORHER	07830	DE0	WOHIN
06520	A00	U*BERDIES	07180	A00	VORHIN	07840	D00	WOHINAUS
06530	A00	U*BEREIN	07190	A00	VORHINEIN	07850	N00	WOHINGEGEN
06540	A00	U*BEREINANDER	07200	G00	VORN	07860	DE0	WOHIT
06550	G00	U*BERN	07210	A00	VORN	07870	A00	WOHIGLICH
06560	G00	U*BERN	07220	A00	VORNAN	07880	DE0	WONACH
06570	G00	U*BERS	07230	A00	VORNE	07890	DE0	WORAN
06580	AP0	UH	07240	A00	VORNHINEIN	07900	DE0	WORAUF
06590	A00	UHKER	07250	A00	VORNUEER	07910	DE0	WORAUS
06600	A00	UHKIN	07260	G00	VORS	07920	DE0	WOREIN
06610	G00	UHS	07270	A00	VORU*BER	07930	DE0	WOKIN
06620	P00	UNBESCHADET	07280	A00	VORUA*RTS	07940	LE0	WORMEER
06630	M00	UND	07290	FN0	UA*HREND	07950	DE0	WORM
06640	AP0	UNFERN	07300	A00	UA*HRENDEN	07960	DE0	WORUNTER
06650	CPN	UNGEACHTET	07310	A00	UA*HRENDDES	07970	DE0	WOVON
06660	A00	UNGEFA*HR	07320	A00	UA*HRENDDESSEN	07980	DE0	WOVOR
06670	A00	UNLA*NGST	07330	KT0	WAS	07990	DE0	WOVUER
06680	Y00	UNS	07340	M00	WEDER	08000	DE0	WOZU
06690	UY0	UNSER	07350	N00	WEIL	08010	A00	WOHL
06700	U00	UNSERE	07360	A00	WEG	08020	A00	WOHLAUF

08030 \*04 ZU  
 08040 A00 ZUALLERERST  
 08050 A00 ZUA+US+ERST  
 08060 A00 ZUEDEM  
 08070 A00 ZUEINANDER  
 08080 A00 ZUERST  
 08090 R00 ZUFOLGE  
 08100 A00 ZUGRUNDE  
 08110 P00 ZUGUNSTEN  
 08120 A00 ZUGUTE  
 08130 A00 ZUHINTERST  
 08140 A00 ZULANDE  
 08150 A00 ZULETZT  
 08160 A00 ZULIEBE  
 08170 G00 ZUM  
 08180 AN0 ZUMAL  
 08190 A00 ZUMEIST  
 08200 A00 ZUHINDEST  
 08210 A00 ZUHUTE  
 08220 AF0 ZUNA+CHST  
 08230 A00 ZUNUTZE  
 08240 A00 ZUOBERST  
 08250 G00 ZUR  
 08260 A00 ZURECHT  
 08270 A00 ZURU+CK  
 08280 A00 ZURZEIT  
 08290 A00 ZUSAMMEN  
 08300 P00 ZUSAMT  
 08310 A00 ZUTEIL  
 08320 A00 ZUTIEFST  
 08330 A00 ZUUNTERST  
 08340 A00 ZUUNGUNSTEN  
 08350 A00 ZUVIEL  
 08360 A00 ZUVOR  
 08370 A00 ZUVO+RDERST  
 08380 A00 ZUWEILEN  
 08390 A00 ZUWIDER  
 08400 A00 ZUZEITEN  
 08410 A00 ZUZU+GLICH  
 08420 A00 ZWAR  
 08430 P00 ZWECKS  
 08440 P00 ZWISCHEN  
 08450 A00 ZWISCHENDURCH  
 08460 A00 ZWISCHENHER

#### 4.3.2.1.2 Les mots outils expressions (.FIXALL(EXPRALL))

Afin de limiter les balayages de la phrase et pour faciliter l'analyse automatique ultérieure, nous avons cherché à repérer les mots-outils disjoints et à les associer dans un même enregistrement.

Le fichier ci-dessous est lu sans tenir compte du signe "@" pour les prépositions, conjonctions de subordination, outils de la comparaison et interrogatifs multitermes, du "#" pour les expressions adverbiales, et du "'" pour les expressions à balance du type "NICHT NUR ... SONDERN AUCH".

.FIXALL(EXPRALL) :

00010	.	FIXALL(EXPRALL)	00520	N00	QUENN#AUCH
00020	00117		00530	N00	QUENN#NICHT
00030	00097		00540	N00	QUIEWENN
00040	00052		00550	N00	QUOONICHT
00050	06348		00560	F00	#AB#UND#AN
00060	N00	QALSO00	00570	F00	#AN#MEISTEN
00070	F00	QANQ#HAND	00580	F00	#AB#UND#ZU
00080	N00	QANSTATT#ZU	00590	F00	#AN#SICH
00090	F00	QANQ#STELLE	00600	F00	#AUF#UND#AB
00100	N00	QAUCH#WENN	00610	F00	#AUS#UND#EIN
00110	N00	QAUF#DAS#	00620	F00	#AUS#ER#ACHT
00120	N00	QAUF#Q#GRUND	00630	F00	#BEI#ALLE#DER
00130	N00	QAUS#ER#DAS#	00640	F00	#BEI#WEITEN
00140	N00	QAUS#ER#WENN	00650	F00	#DAN#JA
00150	F00	QBIS#AN	00660	F00	#DAN#JEDOCH
00160	F00	QBIS#AUF	00670	F00	#DAN#ABER
00170	N00	QBIS#DAS#	00680	F00	#DARU#BER#HINAUS
00180	F00	QBIS#IN	00690	F00	#DAS#HEIS#T
00190	F00	QBIS#NACH	00700	F00	#DESTO#BESSER
00200	F00	QBIS#ZU	00710	F00	#DESTO#MEHR
00210	F00	QBIS#ZUM	00720	F00	#DESTO#WENIGER
00220	F00	QBIS#ZUR	00730	F00	#EIN#UND#AUS
00230	L00	QEBENS#OWIE	00740	F00	#IMMER#MEHR
00240	N00	QENER#DAS#	00750	F00	#IMMER#NOCH
00250	N00	QES#SEI#DENN	00760	F00	#IN#ACHT
00260	N00	QGESCHWEIGE#DAS#	00770	F00	#JE#UND#JE
00270	N00	QGESCHWEIGE#DENN	00780	F00	#JE#NACH#DEN
00280	N00	QINS#OFER#ALS	00790	F00	#NACH#UND#NACH
00290	N00	QINS#OWEIT#ALS	00800	F00	#NACH#WIE#VOR
00300	N00	QJEDESMAL#WENN	00810	F00	#NUR#NOCH
00310	N00	QJE#NACH	00820	F00	#OHNE#WEITERES
00320	N00	QKAUN#DAS#	00830	F00	#SCHON#DES#WEGEN
00330	L00	QMEHR#ALS	00840	F00	#SCHON#LANGE
00340	N00	QNR#DAS#	00850	F00	#SO#SEHR
00350	N00	QOB#AUCH	00860	F00	#TROTZ#ALLE#DER
00360	N00	QOHNE#DAS#	00870	F00	#UND#SO#WEITER
00370	N00	QSEIGES	00880	F00	#VON#ALLE#DER
00380	N00	QSELBST#WENN	00890	F00	#VON#ALLEIN
00390	N00	QSO#DAS#	00900	F00	#VON#DA#AB
00400	N00	QSTAT#ZU	00910	F00	#VON#DA#AN
00410	N00	QSTAT#DAS#	00920	F00	#VON#JE
00420	L00	QU#SO	00930	F00	#VOR#ALLEN
00430	N00	QUND#WENN	00940	N00	"ALS" AUCH
00440	F00	QUNWEIT#VON	00950	N00	"ALS" DAS#
00450	N00	QWAS#AUCH	00960	L00	"DESTO" MEHR
00460	T00	QWAS#FU#R#EIN	00970	L00	"JE" BESSER
00470	T00	QWAS#FU#R#EINE	00980	L00	"JE" MEHR
00480	T00	QWAS#FU#R#EINEN	00990	L00	"NICHT" MEHR
00490	T00	QWAS#FU#R#EINEN	01000	N00	"NICHT" NUR
00500	T00	QWAS#FU#R#EINER	01010	N00	"SONDERN" AUCH
00510	T00	QWAS#FU#R#EINES	01020	N00	"WIE" AUCH

Après repérage dans .SORTIE1.BASE, les segments d'une chaîne-outil<sup>1</sup> sont concaténés sur une même ligne, associés au code prévu.

```

*0052
0319
0030
0000
0000
0000 014 021 029
0000
0000
0000
00052015660001 P00 MIT
00052015670002 $00 $0*L
00052015680003 N00 ODER
00052015690004 $00 $KOHLE
00052015700005 *01 ALS
00052015710006 PRIMA*REN
00052015720007 $00 $BRENNSTOFFEN
00052015730008 SIND
00052015740009 IVK DAS
00052015750010 F00 #VOR#ALLEN ←
00052015760011 $00 $SCHWEFELDIOXID
00052015770012 N00 UND
00052015780013 $00 $STICKSTOFFOXID
00052015790014 ,
00052015800015 AM0 DOCH
00052015810016 LIEGT
00052015820017 IVK DIE
00052015830018 $00 $SCHWEFELDIOXID-$EMISSION
00052015840019 A00 BESONDERS
00052015850020 NIEDRIG
00052015860021 ,
00052015870022 N00 WEIL
00052015880023 IVK DIE
00052015890024 $00 $BRENNSTOFFZELLE
00052015900025 P00 GEGEN
00052015910026 $00 $SCHWEFELVERBINDUNGEN
00052015920027 EMPFINDLICH
00052015930028 IST
00052015940029 ,
00052015950030 N00 #SO#DAS* ←
00052015960031 V00 DIESE
00052015970032 P00 IN
00052015980033 IVK DER
00052015990034 $00 $BRENNSTOFF-$AUFBEREITUNGSANLAGE
00052016000035 ENTFERNT
00052016010036 WERDEN
00052016020037 MU*SSEN
00052016030038 .

```

(1) Lorsque les mots-outils sont associés, *so daß...*, nous utilisons le terme de chaîne-outil.

## 4.3.2.1.3 La partie numérale

Suivie d'une combinaison de deux chiffres, la lettre C permet de générer les codes suivants :

C00	nombre, quel qu'il soit, écrit en chiffres
C01	cardinal EINS, ZWEI ...
C02	ordinal ERSTE, ZWEITE ...
C03	multiplicateur VIER/FACH, VIER/FALTIG ...
C04	fractionnaire DRITTEL, VIERTEL ...
C05	adverbial ERSTENS, ZWEITENS ...
C06	répétitif VIERMAL
C07	collectif EINERLEI

```

*0040
0020
0020
0000
0000
0002 015 022
0000
0000
0002 003 007
00040014530001      WIRD
00040014540002    Y00 ES
00040014550003      -
00040014560004    PAD BEI
00040014570005    Z00 LASTABHA:NGIGREM
00040014580006    $00 $BETRIEB
00040014590007      -
00040014600008    A00 NUR
00040014610009    *04 ZU
00040014620010    C01 VIERZIG      ←
00040014630011    $00 $PROZENT
00040014640012    UY0 SEINER
00040014650013    $00 $NENNLEISTUNG
00040014660014      GENUTZT
00040014670015      ,
00040014680016    *03 SO
00040014690017      VERBRAUCHT
00040014700018    Y00 ES
00040014710019    A00 FAST
00040014720020    C00 12000      ←
00040014730021    $00 $KILOJoule
00040014740022      ,
00040014750023    #AFO UN
00040014760024    JS0 EINE
00040014770025    $00 $KILOWATTSTUNDE
00040014780026    *04 ZU
00040014790027      LIEFERN
00040014800028      .

```

Le module PROLOG nous mène du fichier .SORTIE1.BASE à trois fichiers de sortie distincts, .TRANSIT1, .TRANSIT2 et .TRANSIT5. Il est en effet indispensable de préparer convenablement la désambiguïsation de la majuscule en tête de la phrase.

.TRANSIT1 représente le texte après intégration des résultats de PROLOG, c'est-à-dire, le codage des mots outils unitermes et multitermes, le codage de la partie numérale et le codage des noms (précédés d'une majuscule) à condition qu'ils ne soient pas en tête de phrase, (la majuscule est ici ambiguë). C'est dans ce fichier que seront inclus les résultats du module VERBAL pour le traitement ultérieur de la majuscule.

.TRANSIT1 : phrase n°16

```

*0016
0027
0027
0000
0000
0002 012 021
0002 002 016
0002 004 018
0000
00016004410001   _ $CARBONAT-$IONEN
00016004420002   (
00016004430003   $00 $C$03(-- )
00016004440004   )
00016004450005   BRAUCHEN
00016004460006   JS0 EIN
00016004470007   $00 $SALZ
00016004480008   IVK DER
00016004490009   $00 $KOHLENSA+URE
00016004500010   *01 ALS
00016004510011   $00 $ELEKTROLYT
00016004520012   ,
00016004530013   M0+ UND
00016004540014   #APO UM
00016004550015   $00 $SAUERSTOFF-$IONEN
00016004560016   (
00016004570017   $00 $0--
00016004580018   )
00016004590019   *04 ZU
00016004600020   TRANSPORTIEREN
00016004610021   ,
00016004620022   . VERWENDET
00016004630023   *05 NAN
00016004640024   JS0 EIN
00016004650025   FEST0ES
00016004660026   $00 $OXID
00016004670027   .

```

## .TRANSIT1 : phrase n°44

\*0044  
 0428  
 0054  
 0000  
 0000  
 0004 010 017 032 043  
 0000  
 0000  
 0000  
 00044012690000 \$  
 00044012700001 P00 IN  
 00044012710002 ERSTER  
 00044012720003 \$00 \$LINIE  
 00044012730004 INTERESSIERT  
 00044012740005 S00 MAN  
 00044012750006 U00 SICH  
 00044012760007 A00 HEUTE  
 00044012770008 P00 FU+R  
 00044012780009 \$00 \$BRENNSTOFFZELLEN  
 00044012790010 ,  
 00044012800011 IVK DIE  
 00044012810012 P00 MIT  
 00044012820013 \$00 \$PHOSPHORSA+URE  
 00044012830014 \*01 ALS  
 00044012840015 \$00 \$ELEKTROLYTEN  
 00044012850016 ARBEITEN  
 00044012860017 ,  
 00044012870018 BESCHA+FTIGT  
 00044012880019 U00 SICH  
 00044012890020 P00 IN  
 00044012900021 KLEINEREM  
 00044012910022 \$00 \$UMFANG  
 00044012920023 A00 AUCH  
 00044012930024 P00 MIT  
 00044012940025 \$00 \$CARBONAT-\$SCHMELZEN  
 00044012950026 \*01 ALS  
 00044012960027 \$00 \$ELEKTROLYTEN  
 00044012970028 M00 UND  
 00044012980029 UNTERSUCHT  
 00044012990030 FESTE  
 00044013000031 \$00 \$ELEKTROLYTE  
 00044013010032 ,  
 00044013020033 IVK DIE  
 00044013030034 \$00 \$SAUERSTOFF-\$IONEN  
 00044013040035 TRANSPORTIEREN  
 00044013050036 M00 UND  
 00044013060037 \$00 \$ARBEITSTEMPERATUREN  
 00044013070038 AFO UM  
 00044013080039 C01 TAUSEND  
 00044013090040 \$00 \$GRAD  
 00044013100041 \$00 \$CELSIUS  
 00044013110042 VERLANGEN  
 00044013120043 ,  
 00044013130044 AM0 DOCH  
 00044013140045 SIND  
 00044013150046 \$00 \$BRENNSTOFFZELLEN  
 00044013160047 PAD AUF  
 00044013170048 V00 DIESER  
 00044013180049 \$00 \$BASIS  
 00044013190050 A00 VORERST  
 00044013200051 AM0 NOCH  
 00044013210052 A00 NICHT  
 00044013220053 PRAKTIKABEL  
 00044013230054 .

.TRANSIT2 ne contient que les mots commençant par une minuscule et non codés ainsi que les premiers mots de phrase non codés :

.TRANSIT2 :

00014004210032 GELANGEN  
 00014004220033 NUS\*  
 00015004240001 \$SAURE  
 00015004260003 TRANSPORTIEREN  
 00015004330010 ALKALISCHEN  
 00015004350012 WANDERN  
 00016004410001 \$CARBONAT-\$IONEN  
 00016004450005 BRAUCHEN  
 00016004600020 TRANSPORTIEREN  
 00016004620022 VERWENDET  
 00016004650025 FESTES  
 00017004750007 ERZEUGEN

00044012710002 ERSTER  
 00044012730004 INTERESSIERT  
 00044012850016 ARBEITEN  
 00044012870018 BESCHAFTIGT  
 00044012900021 KLEINEREM  
 00044012980029 UNTERSUCHT  
 00044012990030 FESTE  
 00044013040035 TRANSPORTIEREN  
 00044013080039 TAUSEND  
 00044013110042 VERLANGEN  
 00044013140045 SIND  
 00044013220053 PRAKTIKABEL  
 00045013290005 LIEGEN  
 00045013350011 UMWELTSCHONENDEN  
 00045013480024 VERWENDEN  
 00045013490025 LA\*ST  
 00045013520028 BELIEBIG  
 00045013530029 GROS\*E  
 00045013550031 ZUSAMMENZUSETZEN  
 00046013590003 BLEIBT  
 00046013680012 UNVERANDERT  
 00046013750019 BETRACHTLICHE  
 00046013790023 ERMOGLICHT

.TRANSIT5 rassemble les mots à majuscule à condition qu'ils ne soient pas en tête de phrase. Il sera utilisé pour la désambiguïsation de la majuscule en tête de phrase.

**.TRANSITS**

00015004270004 \$WASSERSTOFF-\$IONEN  
 00015004290006 \$H+  
 00015004340011 \$ELEKTROLYTEN  
 00015004360013 \$HYDROXID-\$IONEN  
 00015004380015 \$OH(---)  
 00016004430003 \$CO3(---)  
 00016004470007 \$SALZ  
 00016004490009 \$KOHLENSA\*URE  
 00016004510011 \$ELEKTROLYT  
 00016004550015 \$SAUERSTOFF-\$IONEN  
 00016004570017 \$O--  
 00016004660026 \$OXID  
 00017004700002 \$GLEICHSPANNUNG  
 00017004740006 \$BRENNSTOFFZELLE  
 00017004820014 \$ART

---

00043012500006 \$BRENNSTOFFZELLE  
 00043012540010 \$WIRKUNGSGRAD  
 00043012630019 \$ELEKTRIZITA\*TSNETZ  
 00043012650021 \$KRAFTWERK  
 00044012720003 \$LINIE  
 00044012780009 \$BRENNSTOFFZELLEN  
 00044012820013 \$PHOSPHORSA\*URE  
 00044012840015 \$ELEKTROLYTEN  
 00044012910022 \$UMFANG  
 00044012940025 \$CARBONAT-\$SCHMELZEN  
 00044012960027 \$ELEKTROLYTEN  
 00044013000031 \$ELEKTROLYTE  
 00044013030034 \$SAUERSTOFF-\$IONEN  
 00044013060037 \$ARBEITSTEMPERATUREN  
 00044013090040 \$GRAD  
 00044013100041 \$CELSIUS  
 00044013150046 \$BRENNSTOFFZELLEN  
 00044013180049 \$BASIS  
 00045013260002 \$VORTEILE  
 00045013280004 \$BRENNSTOFFZELLEN

**4.3.2.2 Le traitement du verbe (VERBAL)**

Le module central du logiciel d'analyse fonctionne sur .TRANSIT2 en entrée. 60 % de ses éléments sont des verbes, le reste des adjectifs ou des adverbes.

Conçu pour le traitement de toute langue source germanique, il se compose de trois programmes importants. Les nombreuses procédures dont une partie n'intervient que pour une langue donnée, ont pour fonction de localiser l'ensemble des formes verbales, de les découper pour reconstituer la base correspondante (nécessaire pour la phase de transfert) et de leur associer un code à quatre positions.

La préparation linguistique de l'étape allemande et sa réalisation informatique achevées, nous avons testé la pertinence de nos méthodes sur le néerlandais puis le luxembourgeois.

Des difficultés spécifiques au néerlandais, (règles orthographiques et nombreuses ambiguïtés dues à l'absence de majuscule) puis le choix de nouvelles techniques de lecture de fichier par variables basées nous ont amené à modifier la présentation des données pour chaque langue et l'ordre de succession des procédures dans le sens d'une rapidité et d'une finesse accrues. Nous donnerons au cours de ce chapitre une illustration de ces tâtonnements.

#### 4.3.2.2.1 Les filtres (.FILTRAL)

La partie terminale des enregistrements de .TRANSIT2 est comparée aux éléments de .MORPHAL(ENDUNG) pour l'allemand, .MORPHOL (ENDUNG) pour le néerlandais et .MORPHLX(ENDUNG) pour le luxembourgeois. Les formes dont la terminaison n'est pas verbale sont éliminées. 75 % à 80 % de ce que le programme retient est effectivement verbal.

##### .MORPHAL(ENDUNG) :

00010 0166  
 00020 0030  
 00030 6634  
 00040 -A  
 00050 +AB  
 00060 +IEB  
 00070 +OB  
 00080 +ARB  
 00090 +UB  
 00100 -C  
 00110 +IED  
 00120 +AND  
 00130 +IND  
 00140 +UD  
 00150 +IRD  
 00160 +END  
 00170 +ELND  
 00180 +ERND  
 00190 +E  
 00200 +ALE  
 00210 +SCHALE  
 00220 +MALE  
 00230 +ATE  
 00240 +RATE  
 00250 +IMATE  
 00260 +ICHE  
 00270 +BLICHE  
 00280 +GLICHE  
 00290 +GLEICHE  
 00300 +SCHLICHE  
 00310 +STRICHE  
 00320 +WICHE  
 00330 +KLICHE  
 00340 +NLICHE  
 00350 +ERLICHE  
 00360 +S+LICHE  
 00370 +ATLICHE  
 00380 +UTLICHE  
 ↓

##### .MORPHOL(ENDUNG) :

00010 0020  
 00020 0030  
 00030 6634  
 00040 +GA  
 00050 +STA  
 00060 +B  
 00070 -C  
 00080 +D  
 00090 +E  
 00100 +F  
 00110 +G  
 00120 -H  
 00130 -I  
 00140 +IJ  
 00150 +K  
 00160 +L  
 00170 +M  
 00180 +EN  
 00190 +N  
 00200 -O  
 00210 +P  
 00220 -Q  
 00230 +R  
 00240 +S  
 00250 +T  
 00270 -U  
 00280 +V  
 00290 +W  
 00300 -X  
 00310 -Y  
 00320 -Z

##### .MORPHLX(ENDUNG) :

00010 00124  
 00020 0030  
 00030 6634  
 00040 A  
 00050 B  
 00060 C  
 00070 +ID  
 00080 +ED  
 00090 +OD  
 00100 +ND  
 00110 +OD  
 00120 +RD  
 00130 +E  
 00140 -DE  
 00150 -DE+SE  
 00160 +EF  
 00170 +E+F  
 00180 +RF  
 00190 +IF  
 00200 +UF  
 00210 +AF  
 00220 +OF  
 00230 +FF  
 00240 +NG  
 00250 -ENG  
 00260 -KENG  
 00270 +EG  
 00280 +IG  
 00290 +SCH  
 00300 -BESONNESCH  
 00310 +CH  
 00320 -DUERCH  
 00330 -NACH  
 00340 -OCH  
 00350 -SECH  
 00360 +A+I  
 00370 +E>I  
 00380 -DE>I  
 ↓

00390 +ISCHE  
 00400 +HEISCHE  
 00410 +FISCHE  
 00420 +HISCHE  
 00430 +FRISCHE  
 00440 +WISCHE  
 00450 -EINZELNE  
 00460 -VIELE  
 00470 +RAF  
 00480 +IEF  
 00490 +IFF  
 00500 +OFF  
 00510 +ALF  
 00520 +ARF  
 00530 +HUF  
 00540 +AG  
 00550 +IEG  
 00560 +ANG  
 00570 +ING  
 00580 +OG  
 00590 +UG  
 00600 +ARG  
 00610 +OH  
 00620 +IEH  
 00630 +AH  
 00640 +CH  
 00650 +ICH  
 00660 +BLICH  
 00670 +GLICH  
 00680 +SCHLICH  
 00690 +STRICH  
 00700 +WICH  
 00710 -ISCH  
 00720 -HOCH  
 00730 -I  
 00740 -J  
 00750 +UK  
 00760 +AK  
 00770 +OLK  
 00780 +ACK  
 00790 +ANK  
 00800 +AHL  
 00810 +OLL  
 00820 +IEL  
 00830 -VIEL  
 00840 +ILL  
 00850 +OHM  
 00860 +AM  
 00870 +AHM  
 00880 +AHN  
 00890 +TEN  
 00900 +ENEM  
 00910 +NDEN  
 00920 +EN  
 00930 +ICHEN  
 00940 +EICHEN  
 00950 +BLICHEN

00390 -EUE>I  
 00400 -HE>I  
 00410 -UE>I  
 00420 +EI  
 00430 -HEI  
 00440 +I  
 00450 J  
 00460 +AK  
 00470 +OK  
 00480 +HK  
 00490 +EK  
 00500 +IK  
 00510 +AL  
 00520 +IL  
 00530 +LL  
 00540 -ALL  
 00550 -WELL  
 00560 +EL  
 00570 +OL  
 00580 +UL  
 00590 ++L  
 00600 +In  
 00610 +HM  
 00620 +AN  
 00630 +EN  
 00640 -DEN  
 00650 -DEENEN  
 00660 -DEEN  
 00670 -ALLEN  
 00680 -EEN  
 00690 -EISEN  
 00700 -DE+SEN  
 00710 +IN  
 00720 +UN  
 00730 -VUN  
 00740 +ON  
 00750 +NN  
 00760 -DANN  
 00770 -WANN  
 00780 -ENN  
 00790 ++N  
 00800 O  
 00810 P  
 00820 Q  
 00830 +AR  
 00840 +ER  
 00850 -DER  
 00860 -DE+SER  
 00870 -ABER  
 00880 -DEER  
 00890 -E+HNER  
 00900 -E+NNER  
 00910 -EISER  
 00920 -ENGER  
 00930 -HER  
 00940 -ODER  
 00950 -ESOUQUER

00960 +GLICHEN	01550 +EST	00960 -NER
00970 +SCHLICHEN	01560 +GENEST	00970 +*R
00980 +STRICHEN	01570 +RST	00980 +AS
00990 +WICHEN	01580 +BIRST	00990 +SS
01000 +KLICHEN	01590 +BARST	01000 +*S
01010 +NLICHEN	01600 +BORST	01010 -DE*5
01020 +ERLICHEN	01610 +*CHST	01020 +IS
01030 +S*LICHEN	01620 +WA*CHST	01030 -BIS
01040 +ATLICHEN	01630 -SCHST	01040 -EIS
01050 +UTLICHEN	01640 -U	01050 +ES
01060 +ISCHEN	01650 -V	01060 -ALLES
01070 +HEISCHEN	01660 -W	01070 +T
01080 +FISCHEN	01670 -X	01080 +AT
01090 +MISCHEN	01680 -Y	01090 -DAT
01100 +FRISCHEN	01690 +DLZ	01100 -SAT
01110 +WISCHEN		01110 +IT
01120 +ALEN		01120 +OT
01130 +SCHALEN		01130 +UT
01140 +MALEN		01140 +TT
01150 +ATEN		01150 -DATT
01160 +RATEN		01160 +RT
01170 +MATEN		01170 +LT
01180 +ELN		01180 +*T
01190 +ERN		01190 -E*T
01200 -EINZELN		01200 -NE*T
01210 -FERN		01210 +U
01220 +EIN		01220 -ESOU
01230 +SEIN		01230 V
01240 +NN		01240 W
01250 -EINZELNEN		01250 X
01260 -VIELEN		01260 Y
01270 +AN		01270 Z
01280 +GETAN		
01290 -O		
01300 -P		
01310 -Q		
01320 +UHR		
01330 +OR		
01340 +AR		
01350 +UR		
01360 +ENER		
01370 +TER		
01380 -WEITER		
01390 +ENDER		
01400 +IES		
01410 +NAS		
01420 +LAS		
01430 +CHS		
01440 +NDES		
01450 +ENES		
01460 +TES		
01470 +T		
01480 -ST		
01490 +A*ST		
01500 +EIST		
01510 -NEIST		
01520 +IEST		
01530 +KIEST		
01540 +LIEST		

Lorsque la terminaison du mot testé figure dans ces tableaux précédée du signe "+", elle est potentiellement verbale, précédée du signe "-", elle est rejetée.

## 4.3.2.2.2 Les verbes d'emprunt (.VERBAL1)

Les langues germaniques que nous analysons ont en commun d'absorber une quantité non négligeable de verbes "romans", les empruntant tantôt au français, tantôt au latin. -IEREN pour l'allemand, -ERA pour le suédois, -EREN pour le néerlandais, -ERE pour le danois et le norvégien, -EIRE pour le luxembourgeois sont les dénominateurs communs qui autorisent une première étude sans recours aux dictionnaires de racines. L'analyse s'appuie essentiellement sur une reconnaissance fine de la partie terminale, la levée des interférences avec les formes germaniques (*halbieren, gieren, verlieren, fieren, amtieren, stieren, zieren....* pour l'allemand) et le découpage des préverbes, peu fréquents il est vrai.

Les tableaux de désinences qui suivent correspondent à trois niveaux d'analyse différents :

.MORPHLX(DES) réunit les désinences luxembourgeoises sans autres indications.

.MORPHOL(DES) permet de différencier les désinences classées selon le signe diacritique qui les précède: "@" pour les participes déclinés, "\$" pour le participe présent non décliné, "%" pour l'indicatif, "\*" pour le prétérit.

.MORPHAL(DES) représente la version la plus récente : les désinences sont associées à des codes de quatre chiffres sans lesquels il serait impossible de décrire correctement la forme "EN" (et "IER/EN") rencontrée à l'infinitif (1), à l'indicatif présent (3), à l'indicatif passé (4) et au participe passé (8) .

L'incompatibilité de certaines données apparaît au découpage, qui permet de simplifier le code : L'absence ou la présence du "GE", morphème du participe passé, infirme ou confirme, pour la terminaison "EN", et selon la structure du verbe, la configuration du code. Il en va de même pour les désinences "T" et "ET" dans le cas des verbes réguliers.

.MORPHAL(DES) :.MORPHOL(DES) :.MORPHLX(DES) :

00010 0046  
00020 0032  
00030 0030  
00040 6334  
00050 3000E  
00060 3000T  
00070 1300N  
00080 1348EN  
00090 3000ET  
00100 4700TE  
00110 6000ND  
00120 6000END  
00130 5000NDE  
00140 7000ENE  
00150 4700TEN  
00160 7000TER  
00170 7000TEH  
00180 7000TES  
00190 4700ETE  
00200 5000ENDE  
00210 5000NDER  
00220 5000NDEM  
00230 5000NDEN  
00240 5000NDES  
00250 7000ENEX

00010 0020  
00020 0030  
00030 6334  
00040 ZE  
00050 ZN  
00060 ZT  
00070 ZD  
00080 ZEN  
00090 \*DE  
00100 \*TE  
00110 \*DEN  
00120 \*TEN  
00130 \$ND  
00140 ZE\*N  
00150 \$END  
00160 QENDE  
00170 QNDE  
00180 QTES  
00190 QDES  
00200 QNES  
00210 QENES  
00220 QENEN  
00230 QENDEN

00010 0010  
00020 0030  
00030 6334  
00040 +ENEM  
00050 +ENER  
00060 +ENEN  
00070 +ETEN  
00080 +ETER  
00090 +ETEM  
00100 +ENE  
00110 +ETE  
00120 +TEN  
00130 +TER  
00140 +TEH  
00150 +E\*N  
00160 +EN  
00170 +ET  
00180 +TE  
00190 +E  
00200 +N  
00210 +T

00260 7000ENEN  
 00270 7000ENER  
 00280 7000ENES  
 00290 4700ETEN  
 00300 7000ETER  
 00310 7000ETEM  
 00320 7000ETES  
 00330 5000ENDER  
 00340 5000ENDEN  
 00350 5000ENDEN  
 00360 5000ENDES  
 00370 7000IERTER  
 00380 7000IERTEM  
 00390 4700IERTEN  
 00400 7000IERTES  
 00410 4700IERTE  
 00420 3800IERT  
 00430 5000IERENDEN  
 00440 5000IERENDEN  
 00450 5000IERENDER  
 00460 5000IERENDES  
 00470 5000IERENDE  
 00480 6000IEREND  
 00490 1340IEREN  
 00500 3000IERE

#### 4.3.2.2.3 Les verbes allemands (.VERBAL2)

L'analyse de la forme potentiellement verbale s'effectue, cette fois ci, par la gauche, et en trois temps :

- reconnaissance et découpage de la préverbatation
- reconnaissance de la racine verbale
- vérification du résidu et codage

##### 4.3.2.2.3.1 Etude de la préverbatation

Dans un but de clarté, nous conviendrons d'appeler préverbes dans ce chapitre, ce que l'on nomme ailleurs particule séparable, inséparable ou mixte, et qui existe, bien que ne fonctionnant pas toujours de la même façon, dans les autres langues germaniques.

- *séparable/inséparable* : il n'y a pas lieu de rechercher dans cette étape les préverbes séparables disjoints, ces derniers n'étant pas systématiquement rejetés en fin de proposition. L'analyse syntaxique seule permettra, en fin d'analyse, de repérer ces préverbes et de les associer au verbe correspondant. PROLOG leur a adjoint le code 'O'.

ex : dans .FIXALL(OUTALL)

Ø514Ø - POØ MIT

Si une telle occurrence n'introduit pas un groupe prépositionnel, son code sera désambiguïsé : POØ --> OØØ

Les fichiers de préverbes utilisés remplissent une double fonction :

- les préverbes sont précédés du signe "+" et suivis du signe "%" (pour tenir compte du caractère blanc). +GE% représente également le "GE" du participe
- les enregistrements précédés du signe "-" éliminent les risques de confusion dans le cas de certaines racines.

ex: -ANTW% évite d'éliminer le verbe antworten  
 -GESTALT% évite le découpage d'un "GE" préverbe inséparable  
 -GEBRAUCH% évite le découpage d'un "GE" ambigu, traité ultérieurement

.MORPHAL(PREFI) :

.MORPHOL(PREFI) :

.MORPHLX(PREFI) :

00010 0333  
 00020 0176  
 00030 0078  
 00040 0040  
 00050 0034  
 00060 0030  
 00070 6334  
 00080 +ABZ  
 00090 +ANZ  
 00100 -ANTWZ  
 00110 -ANGELNZ  
 00120 -ANGELTZ  
 00130 -ANGELSZ  
 00140 -ANGLE Z  
 00150 -ANKERNZ  
 00160 -ANKERTZ  
 00170 -ANKEREX  
 00180 -ANKERSZ  
 00190 +BEZ  
 00200 -BECZ  
 00210 -BERGX  
 00220 -BERSZ  
 00230 -BESSZ  
 00240 -BEHRZ  
 00250 -BEHEZ  
 00260 -BEBTZ  
 00270 -BEBSZ  
 00280 -BETTZ  
 00290 -BEUGZ  
 00300 -BEULZ  
 00310 -BEUTZ  
 00320 -BELLZ  
 00330 -BELFZ  
 00340 -BEIZZ  
 00350 -BETERZ  
 00360 -BETETZ  
 00370 -BETESZ  
 00380 -BEVEGX  
 00390 -BEFREIZ  
 00400 -BEGU+NSZ

00010 0125  
 00020 0060  
 00030 0041  
 00040 0013  
 00050 0007  
 00060 0030  
 00070 6334  
 00080 +AFX  
 00090 -AFRONDZ  
 00100 +BEZ  
 00110 -BEDAARZ  
 00120 -BEDRIZ  
 00130 -BEDROZ  
 00140 -BENDODZ  
 00150 -BEUKZ  
 00160 -BEURZ  
 00170 -BEGAARZ  
 00180 -BEGIZ  
 00190 -BEGOZ  
 00200 -BEGRENZ  
 00210 -BELLZ  
 00220 -BEEFZ  
 00230 -BEVATZ  
 00240 -BEELDZ  
 00250 -BEVEELZ  
 00260 -BEVELZ  
 00270 -BEVINDZ  
 00280 -BEVKIJDZ  
 00290 -BENGZ  
 00300 -BETEERZ  
 00310 -BETERZ  
 00320 +ERZ  
 00330 -EREN Z  
 00340 -ERFZ  
 00350 -ERVZ  
 00360 -ERVARENZ  
 00370 -ERGERZ  
 00380 +GEZ  
 00390 -GEBRUIKEN  
 00400 -GEBEUX

00010 0127  
 00020 0007  
 00030 0060  
 00040 0030  
 00050 0027  
 00060 0003  
 00070 0007  
 00080 0000  
 00090 6334  
 00100 +AZ  
 00110 -APPZ  
 00120 -A+AFZ  
 00130 -AKERZ  
 00140 -A+IFZ  
 00150 -A+NGZ  
 00160 -A+NTWZ  
 00170 +UZ  
 00180 -U+BTZ  
 00190 +ANZ  
 00200 +BEZ  
 00210 -BSAFLDZ  
 00220 -BEETZ  
 00230 -BEERZ  
 00240 -BEETZ  
 00250 -BEODZ  
 00260 -BEFRZ  
 00270 -BEGAAZ  
 00280 -BEGANNZ  
 00290 -BEEBZ  
 00300 -BEECZ  
 00310 -BETODZ  
 00320 -BEM+LTZ  
 00330 -BEMEZ  
 00340 -BEM+LLZ  
 00350 -BEMITZ  
 00360 -BEEHERZ  
 00370 -BEMZ  
 00380 -BEMZ  
 00390 -BEM+LIZ  
 00400 -BEMZ

00410	-BEFEMZ	00410	-GELDZ	00410	-BE+SEZ
00420	-BEFIEZ	00420	-GELDÜ	00420	-BESSEREN
00430	-BEFAHZ	00430	-GENEEX	00430	-BE+HÖZ
00440	-BEFORLZ	00440	-GERENZ	00440	-BENT Z
00450	-BEFA*HLZ	00450	-GERIEZ	00450	-BENNEN Z
00460	-BEGINZ	00460	-GEERZ	00460	-BESTEN
00470	-BEGANNZ	00470	-GENEZ	00470	-BESTIN
00480	-BEGONNZ	00480	-GENERZ	00480	-BEZUD
00490	-BEFLEIX	00490	-GENAZ	00490	-BEZILZ
00500	-BEFLIX	00500	-GEURZ	00500	+BERZ
00510	-BEWOGZ	00510	-GEVENZ	00510	+GELZ
00520	-BEREITZ	00520	-GEZELZ	00520	-GE*ITZ
00530	-BEIS*Z	00530	-GEZETENZ	00530	-GEZIN
00540	+GEZ	00540	-GENIEZ	00540	-GEBOCHEN Z
00550	-GEBRECHZ	00550	-GENOEGZ	00550	-GEFALLZ
00560	-GEBRICHZ	00560	-GEEFZ	00560	-GESIN
00570	-GEBROCHZ	00570	-GEEUWZ	00570	-GECKEN
00580	-GEDENKZ	00580	-GEFZ	00580	+GÖZ
00590	-GEDACHZ	00590	-GELIJKZ	00590	-DONAT
00600	-GERINNZ	00600	+INZ	00600	-DONKZ
00610	-GERANNZ	00610	-INNERZ	00610	-DONGZ
00620	-GERONNZ	00620	+NAZ	00620	-DOU Z
00630	-GEBIETZ	00630	-NADERZ	00630	+MIZ
00640	-GEBOTZ	00640	+ONZ	00640	-HIRAZ
00650	-GEFRIERZ	00650	+ORZ	00650	-HICKZ
00660	-GEFRORZ	00660	+OPZ	00660	-HIDDZ
00670	-GESTENZ	00670	-OPENZ	00670	-HIREN Z
00680	-GESTANDZ	00680	+TEZ	00680	+EMZ
00690	-GEFALLZ	00690	+AANZ	00690	+ROZ
00700	-GEFIELZ	00700	-AANDACHTZ	00700	-NOTIZ
00710	-GERATZ	00710	-AANTALZ	00710	+OFZ
00720	-GERIETZ	00720	+ANTZ	00720	-OFFREDEZ
00730	-GEBRAUCHZ	00730	+BIJZ	00730	+OPZ
00740	-GEDULDZ	00740	-BIJTZ	00740	+UNZ
00750	-GENORCHZ	00750	+EENZ	00750	+ZEX
00760	-GEHO*HZ	00760	+HERZ	00760	-ZE>IZ
00770	-GELEITZ	00770	+LOSZ	00770	-ZEI Z
00780	-GELOBZ	00780	-LOOSZ	00780	-ZE>CKZ
00790	-GEREICHE Z	00790	+NEEZ	00790	-ZECKZ
00800	-GEREICHEN Z	00800	+NISZ	00800	-ZE*HPERZ
00810	-GEREICHT Z	00810	-MISSZ	00810	-ZE*NKZ
00820	-GEREICHTE Z	00820	+OORZ	00820	-ZENNZ
00830	-GEREICHTEN Z	00830	+ONTZ	00830	-ZE*SEZ
00840	-GERUHZ	00840	+RUGZ	00840	+ZUZ
00850	-GESEGNZ	00850	+TOEZ	00850	-ZUKUNFTZ
00860	-GETRAUZ	00860	+UITZ	00860	-ZUELZ
00870	-GEWAHRZ	00870	-UITNODZ	00870	+AUSZ
00880	-GEWA*HRZ	00880	-UITTEZ	00880	+BEIZ
00890	-GEZIENZ	00890	+VERZ	00890	-BEICHZ
00900	-GEREUX	00900	-VERBEELZ	00900	+BE*Z
00910	-GELTZ	00910	-VERBETERZ	00910	+BRATZ
00920	-GESSZ	00920	-VERDICHTZ	00920	+BÖRZ
00930	-GEUDZ	00930	-VERDRZ	00930	+BRÜZ
00940	-GEHRZ	00940	-VERDOZ	00940	-BRÜDDZ
00950	-GEGNZ	00950	-VERDUZ	00950	-BRÜNNZ
00960	-GELLZ	00960	-VERGLETZ	00960	+E*HZ
00970	-GERBZ	00970	-VERGETZ	00970	-BRUSCHZ
00980	-GEISZ	00980	-VERGATZ	00980	+ENTZ
00990	-GEIZZ	00990	-VERGROTZ	00990	+GE*Z
01000	-GEIGZ	01000	-VERGROOTZ	01000	+HIRZ
01010	-GEIFZ	01010	-VERKLEIZ	01010	+HATZ
01020	-GEMENZ	01020	-VERLIEZ	01020	+NGEZ
01030	-GEBENZ	01030	-VERLORZ	01030	-CHSZ
01040	-GEHSTZ	01040	-VERHINDZ	01040	+RAFZ
01050	-GELINGZ	01050	-VERSLZ	01050	+ROFZ

01060 -GELANGEX	01060 +VOLZ	01060 +SOUZ
01070 -GELUNGZ	01070 -VOLGEX	01070 +VERZ
01080 -GESTATTZ	01080 +WANZ	01080 -VERGLAEX
01090 -GERINGZ	01090 +WEGZ	01090 -VERGLACHZ
01100 -GEMINNZ	01100 +WODRZ	01100 -VERGLEZ
01110 -GEMANNZ	01110 +DAARZ	01110 -VERLEEX
01120 -GEMONNZ	01120 +HEENZ	01120 -VERLUEX
01130 -GEBAREX	01130 +HIERZ	01130 -VERSPEEX
01140 -GEBARTZ	01140 +HEDEX	01140 -VERSPRAC
01150 -GEBU+HRZ	01150 +HEERZ	01150 -VERUERZ
01160 -GEDEINZ	01160 +OVERZ	01160 +VIRZ
01170 -GEDIEHZ	01170 +STILZ	01170 +ZERZ
01180 -GEFA+HRDZ	01180 +VOORZ	01180 +ZOUZ
01190 -GELANGTZ	01190 +VRIJZ	01190 +BA+IZ
01200 -GELANGZ	01200 +WEERZ	01200 +BRUD
01210 -GELU+STZ	01210 +VAARZ	01210 +FESTZ
01220 -GENOSSZ	01220 -WAARDEX	01220 +OUERZ
01230 -GENO+SSZ	01230 +BOVENZ	01230 +FORTZ
01240 -GENIES+Z	01240 +KWIJTZ	01240 +IUEEZ
01250 -GENEEX	01250 +NEVENZ	01250 +LASSZ
01260 -GENU+GZ	01260 +ONDERZ	01260 +UECHTZ
01270 -GESCHIEHZ	01270 +SAHENZ	01270 +FRA+IZ
01280 -GESCHEHZ	01280 +TEGENZ	01280 +IULERZ
01290 -GESCHANZ	01290 +VOORTZ	01290 +OUERCHZ
01300 -GESELLZ	01300 +SINNENZ	01300 +S+NNERZ
01310 -GESTALTZ	01310 +PLAATSZ	01310 +UECHTZ
01320 -GEWA+LTZ	01320 +WAARTSZ	01320 +SUNNERZ
01330 -GEWA+RTZ		01330 +RAI+CHZ
01340 -GEWO+HNZ		01340 +HANNERZ
01350 -GENESZ		01350 +ZRE+CKZ
01360 -GENASZ		01360 +ZERE+CKZ
01370 -GENA+SZ		
01380 -GEBAR Z		
01390 -GEBARENZ		
01400 -GEBAR+RZ		
01410 -GEBIERZ		
01420 -GEBORZ		
01430 -GENA+Z		
01440 -GENASENZ		
01450 -GENAS Z		
01460 -GENEST Z		
01470 -GENESEZ		
01480 +UNZ		
01490 +ZUZ		
01500 -ZUCZ		
01510 -ZUNUTZE Z		
01520 -ZUPFZ		
01530 +DAZ		
01540 -DACZ		
01550 -DAUZ		
01560 -DANKZ		
01570 -DARBZ		
01580 -DAMPZ		
01590 -DANRZ		
01600 -DARFZ		
01610 +ERZ		
01620 -ERNTZ		
01630 -ERBTZ		
01640 -ERBSZ		
01650 -ERBENZ		
01660 -ERDIGZ		
01670 -ERDESZ		
01680 -ERDETZ		
01690 -ERDEN Z		
01700 -ERDENBZ		
01710 -ERMUTZ		
01720 -ERNO+Z		
01730 -ERPLEIZ		
01740 -ERSLICHZ		

01750 -ERKIEZ  
 01760 -ERKIZ  
 01770 -ERLO+SOZ  
 01780 -ERLISOZ  
 01790 -ERLOSEZ  
 01800 +INZ  
 01810 -INFOLGZ  
 01820 +OBZ  
 01830 +UMZ  
 01840 +AUSZ  
 01850 +ENTZ  
 01860 -ENTERTZ  
 01870 -ENTERNZ  
 01880 -ENTERSEZ  
 01890 +VERZ  
 01900 -VEREINZ  
 01910 -VERLANGZ  
 01920 -VERMINDZ  
 01930 -VERGRÖ+Z  
 01940 -VERBILLZ  
 01950 -VERDRIEZ  
 01960 -VERDROSEZ  
 01970 -VERGESSZ  
 01980 -VERGIS+Z  
 01990 -VERGAS+Z  
 02000 -VERLIERZ  
 02010 -VERLORZ  
 02020 -VERDERBZ  
 02030 -VERDIRBZ  
 02040 -VERDARBZ  
 02050 -VERDORBZ  
 02060 +VORZ  
 02070 +AUFZ  
 02080 +BEIZ  
 02090 -BEICZ  
 02100 -BEISSZ  
 02110 -BEITZ  
 02120 -BEIRZ  
 02130 -BEIZENZ  
 02140 +DARZ  
 02150 -DARBTZ  
 02160 -DARBSTZ  
 02170 -DARFSTZ  
 02180 -DARBENZ  
 02190 +EINZ  
 02200 -EINTENZ  
 02210 -EINTETZ  
 02220 -EINTESZ  
 02230 -EINTE Z  
 02240 -EINIOTZ  
 02250 -EINIOTZ  
 02260 -EINENDZ  
 02270 -EIHIGENZ  
 02280 -EINIGE Z  
 02290 -EINSAREZ  
 02300 -EINSANSZ  
 02310 -EINSANTZ  
 02320 -EINHEITLZ  
 02330 +HERZ  
 02340 -HERRSZ  
 02350 -HERRLZ  
 02360 -HERZIGZ  
 02370 -HERZENZ  
 02380 -HERZESZ  
 02390 +HINZ  
 02400 -HINDRZ  
 02410 -HINKRZ  
 02420 -HINKTZ

02430 -HINDERZ  
 02440 -HINKENZ  
 02450 -HINKE Z  
 02460 +LOSZ  
 02470 -LOSE Z  
 02480 -LOSTEX  
 02490 -LOSTENZ  
 02500 +MITZ  
 02510 -MITTLZ  
 02520 -MITTELZ  
 02530 +WEGZ  
 02540 -WEGTENZ  
 02550 -WEGTESZ  
 02560 -WEGTETZ  
 02570 -WEGENDZ  
 02580 +ZERZ  
 02590 -ZERRTZ  
 02600 -ZERRENDZ  
 02610 -ZERREN Z  
 02620 +NACHZ  
 02630 -NACHTE Z  
 02640 -NACHTENZ  
 02650 -NACHTETZ  
 02660 -NACHTESZ  
 02670 -NACHLA+SZ  
 02680 +FORTZ  
 02690 +FU+RZ  
 02700 +QUERZ  
 02710 +FREIZ  
 02720 -FREIE Z  
 02730 -FREIET Z  
 02740 -FREIEN Z  
 02750 -FREIT Z  
 02760 -FREITE Z  
 02770 -FREITEN Z  
 02780 +NIS+Z  
 02790 -NIS+T Z  
 02800 +WEITZ  
 02810 -WEITE Z  
 02820 -WEITET Z  
 02830 -WEITEN Z  
 02840 -WEITETE Z  
 02850 -WEITETEN Z  
 02860 -WEITERN Z  
 02870 -WEITERT Z  
 02880 -WEITERTE Z  
 02890 -WEITERTEN Z  
 02900 -WEITERE Z  
 02910 +FESTZ  
 02920 -FESTE Z  
 02930 -FESTER Z  
 02940 -FESTES Z  
 02950 -FESTEN Z  
 02960 -FESTEM Z  
 02970 -FESTIGZ  
 02980 +KENNZ  
 02990 -KENNEN Z  
 03000 -KENNE Z  
 03010 -KENNT Z  
 03020 +U+BERZ  
 03030 +ANDERZ  
 03040 +DURCHZ  
 03050 +GEGENZ  
 03060 +NEBENZ  
 03070 +RU+CKZ  
 03080 -RU+CKE Z  
 03090 -RU+CKT Z  
 03100 -RU+CKENZ  
 03110 -RU+CKTE Z

03120 -RU\*CKTESZ  
03130 -RU\*CKTENSZ  
03140 +STATZ  
03150 +UNTERZ  
03160 +WIDERZ  
03170 -WIDERN Z  
03180 -WIDERE Z  
03190 -WIDERT Z  
03200 -WIDERTE Z  
03210 -WIDERTEN Z  
03220 +RECHTZ  
03230 -RECHTE Z  
03240 -RECHTER Z  
03250 -RECHTES Z  
03260 -RECHTEN Z  
03270 -RECHTEN Z  
03280 -RECHTIG Z  
03290 +WA\*HR  
03300 -WA\*HRE Z  
03310 -WA\*HRET Z  
03320 -WA\*HREN Z  
03330 -WA\*HRT Z  
03340 -WA\*HRTE Z  
03350 -WA\*HRTEN Z  
03360 +SAMMENZ  
03370 +NIEDERZ  
03380 +WA\*RTSZ  
03390 +WEITERZ  
03400 +WIEDERZ

#### 4.3.2.2.3.2 Repérage de la racine verbale

Afin de ne pas multiplier les petits tableaux, nous avons classé les verbes en deux catégories (morphologiques), pour l'allemand, le néerlandais et le luxembourgeois.

Par opposition aux verbes réguliers, les verbes irréguliers allemands et néerlandais dépendent au moins à une des caractéristiques suivantes :

- Ils modifient la voyelle de leur radical au prétérit ou au participe passé
- Ils présentent un participe passé en **-en** ou **-n**.

Ceci inclut les auxiliaires :

- SEIN, WERDEN
- ZIJN, HEBBEN

les "préterito-présents" :

- Können, dürfen, müssen, mögen
- Kunnen, moeten, mogen, willen, zullen

les "mixtes" :

- brennen, kennen, denken, bringen, nennen, rennen, senden, wenden
- backen, scheiden. (imparfait régulier et participe en **-en**).

Aux quelque vingt verbes qui appartiennent à cette famille, il faut ajouter une série de verbes qui présentent un prétérit avec modification de la voyelle du radical et un participe régulier : brengen, denken, dunken, jagen, kopen, pflügen, vragen...

La distinction régulier-irrégulier vaut également pour le suédois :

régulier :           - le radical ne subit aucune alternance vocalique  
                           - le prétérit se distingue par la présence d'un suffixe à dentale :  
                           -**de**, **-te**, **-dde**

irrégulier : - le radical modifie sa voyelle et le prétérit n'a pas de désinence

Pour le danois et le norvégien :

régulier :           - prétérit   **-ede**, participe   **-et**  
                           - prétérit   **-te**,   participe   **-t** avec ou sans métaphonie

                          - prétérit   **-et**,   participe   **-et**  
   **-te**,                   **-t**  
   **-de**,                   **-d**  
   **-dde**,                 **-dd**

irrégulier : désinence  $\emptyset$  au prétérit et changement vocalique au prétérit et au participe passé

Ce partage, s'il a le défaut d'effacer les particularités de chaque langue, a l'avantage de se prêter à une exploitation simple, sûre et rapide. A ce stade de l'analyse, l'essentiel n'est pas de distinguer :

GEBROCHEN participe passé de GEBRECHEN

GEBROCHEN participe passé de BRECHEN

Il s'agit avant tout de générer des formes sans erreur et de les utiliser comme références.

L'étiquette AUXILIAIRE, REGULIER, IRREGULIER garantit la correction des découpages et plus tard, de la reconstruction du syntagme verbal.

L'étude du verbe n'étant pas l'objet de notre thèse, nous ne nous attarderons pas sur les différences et les similitudes des systèmes verbaux allemand, néerlandais...

Il nous a semblé que le classement proposé plus haut reflétait un dénominateur commun. Elaboré sur l'allemand, l'algorithme a fonctionné pour le néerlandais puis le luxembourgeois, sans subir de transformation, confirmant le bien-fondé de la technique utilisée (séparation des programmes et des données). Les spécificités de chaque langue sont en effet intégrées au niveau des fichiers.

#### 4.3.2.2.3.3 Fichiers de racines

Les différents degrés d'achèvement des fichiers ci-après illustreront le rôle des données.

.VERBAL2, après la reconnaissance et le découpage éventuel de la partie préverbale, appréhende la racine par comparaison, en parcourant l'ensemble des racines régulières puis irrégulières. Le repérage seul ne nécessite qu'une liste simple des racines.

*Exemple pour le luxembourgeois : MORPHLX(VERRG)*

00010 0077	00350 BEWA*LTEG	00690 BUDEL
00020 0030	00360 BEWEG	00700 BUER
00030 6334	00370 BEWE*LLEG	00710 DAACH
00040 APPER	00380 BEWIINT	00720 DABBER
00050 A*A*RD	00390 BECHER	00730 DARM
00060 AKER	00400 BE>CHS	00740 DANG
00070 A*IFER	00410 BE>CK	00750 DANZ
00080 A*NNER	00420 BEICHT	00760 DATZ
00090 A*NGSCHTES	00430 BE>I	00770 DAUSCH
00100 A*NTUER	00440 BE>ISS	00780 DA*IT
00110 BAALG	00450 BE*LZ	00790 DA*IWEL
00120 BAU	00460 BENGEL	00800 DEEL
00130 BAUPS	00470 BE*SEL	00810 DA*NNER
00140 BAASS	00480 BESSER	00820 DECK
00150 BAATSCH	00490 BIEDEL	00830 BE>ID
00160 BABEL	00500 BIISCHT	00840 BEIKSEL
00170 BALLER	00510 BLA*R	00850 DIICHT
00180 BASTEL	00520 BLECH	00860 DIEBEL
00190 BEAFLOSS	00530 BLINNEL	00870 DILL
00200 BEDEN>G	00540 BLE*TZ	00880 'DIRNER
00210 BENEID	00550 BLIEDER	00890 BRA*H
00220 BEETSCH	00560 BLUDD	00900 BRA*NK
00230 BEEN	00570 BOHNEL	00910 BRE>CHEN
00240 BEEZ	00580 BOOTSCH	00920 DRECHSEL
00250 BEDUPPS	00590 BRADDEL	00930 DRE>CK
00260 BEDUSEL	00600 BRAL	00940 DRE>IM
00270 BEDUX	00610 BRATSCH	00950 DRIIPS
00280 BEFRIDDEG	00620 BRAUCH	00960 DRUDEL
00290 BEGAACHEL	00630 BRE>I	00970 DRUMH
00300 BEGANN	00640 BRE*LL	00980 DUEBEL
00310 BEGE>IN	00650 BRE*NS	00990 DUCK
00320 BESICHT	00660 BRUCK	01000 DUSCH
00330 BETOUN	00670 BRUED	01010 DUZ
00340 BESCHA*FTEG	00680 BUDBER	01020 EELZ

01230 EICH	01720 FRASCHT	02410 HA*FEL
01240 E>IDERZ	01730 FRA*A*SCH	02420 HA*HREL
01250 E>IDERZ	01740 FRA*CKS	02430 HA*HSEL
01260 EIZ	01750 FRA*	02440 HA*TECKEL
01270 ERKLA*ER	01760 FRA*ES	02450 HEDIER
01280 ERNINN	01770 FRE*CKEL	02460 HEDISS
01290 FAASCHT	01780 FRECK	02470 HED*HREL
01100 FAASS	01790 FRIESS	02480 HETZ
01110 FABEL	01800 FRIPP	02490 HICK
01120 FACKEL	01810 FRISEL	02500 HIDD
01130 FAKEL	01820 FUCHTEL	02510 HINNEL
01140 FALZ	01830 FUDDEL	02520 HOPF
01150 FASCHT	01840 FUEBEL	02530 HOOBSEL
01160 FASEL	01850 FUERDER	02540 HOUTSCH
01170 FATZ	01860 FUERSCH	02550 HUCKEL
01180 FAULZ	01870 FURNÉL	02560 HUNNER
01190 FAUTEL	01880 FUSCH	02570 HUFF
01200 FAX	01890 FUTTER	02580 HUREL
01210 FA*LSCH	01900 FUURZ	02590 HUSCH
01220 FA*NBEL	01910 FUUSS	02600 HUTSCHEL
01230 FECHT	01920 GAACKS	02610 HUNNEL
01240 FE*CK	01930 GAAPS	02620 IBBERT
01250 FEIER	01940 GABSER	02630 IENGER
01260 FEIL	01950 GAFEL	02640 IERS
01270 FE*IMER	01960 GARNZ	02650 JABEL
01280 FEL	01970 GARREL	02660 JACHTEL
01290 FE*LL	01980 GARNER	02670 JADDER
01300 FE*NN	01990 GAUTSCH	02680 JAMER
01310 FE*NKEL	02000 GA*A*SCHTER	02690 JAPP
01320 FE*SCH	02010 GA*IP	02700 JAPE
01330 FETT	02020 GA*ISSEL	02710 JATSCHEL
01340 FE	02030 GA*IZ	02720 JE*NER
01350 FIDDEL	02040 GECKS	02730 JEXEL
01360 FIEDER	02050 GEI	02740 JHECK
01370 FIEDER	02060 GIERKS	02750 JHETZ
01380 FIERKEL	02070 GLADDER	02760 JHUNN
01390 FILL	02080 GLANN	02770 JMUFF
01400 FINNEL	02090 GLEW	02780 JUCK
01410 FIIRN	02100 GLE*NNER	02790 JUPPEL
01420 FISEL	02110 GLE*TZ	02800 JUA
01430 FISEH	02120 GLE*TZER	02810 KAALL
01440 FLAATSCH	02130 GUTZEL	02820 KAASCHT
01450 FLACKER	02140 GRAATSCH	02830 KACH
01460 FLAH	02150 GRANGEL	02840 KALKNER
01470 FLANTER	02160 GRANZ	02850 KALLEK
01480 FLAPP	02170 GRAUL	02860 KAPAUH
01490 FLATSCH	02180 GRA*TSCH	02870 KARBAATSCH
01500 FLATTER	02190 GRA*Z	02880 KASSE*IER
01510 FLAUT	02200 GRE*ISS	02890 KATZ
01520 FLA*T	02210 GRENZ	02900 KAUL
01530 FLA*TSCH	02220 GRINNEL	02910 KA*IL
01540 FLECHT	02230 GRIPS	02920 KA*IP
01550 FLECK	02240 GRUNNEL	02930 KA*NN
01560 FLE*CK	02250 GUERD	02940 KA*PP
01570 FLEG	02260 HAASEPEL	02950 KE*BBEL
01580 FLE*IU	02270 HAASS	02960 KEDIER
01590 FLE*HZ	02280 HANSTER	02970 KE*IP
01600 FLITSCH	02290 HANDLAANGER	02980 KEIR
01610 FLUCH	02300 HANDEL	02990 KELT
01620 FLUDDER	02310 HANTE*IER	03000 KE*HES
01630 FLUNN	02320 HASEL	03010 KE*PP
01640 FLUPP	02330 HAUCH	03020 KE*SS
01650 FLUTSCH	02340 HAUL	03030 KICKEL
01660 FOCHS	02350 HAUS	03040 KIERP
01670 FOCK	02360 HAUSE*IER	03050 KIERH
01680 FOLTER	02370 HAUW	03060 KIERZ
01690 FÖNKEL	02380 HA*A*SCH	03070 KILL
01700 FOSCH	02390 HA*ERD	03080 KIPP
01710 FORH	02400 HA*ERZ	03090 KLATSCH

03109	KLASSER	03799	KRE>IN	04488	NINN
03119	KLAK	03809	KREMPPEL	04498	NIPS
03129	KLAM	03819	KREPE>IER	04508	MITT
03139	KLAPP	03829	KRE*SPEL	04518	NIX
03149	KLAU	03839	KRIBBEL	04528	NOL
03159	KLAUSCHTER	03849	KROP	04538	NOLTER
03169	KLAMP	03859	KRIEWEL	04548	NONP
03179	KLAFER	03869	KUCK	04558	NONNEL
03189	KLAMTSCH	03879	KUGEL	04568	NOOSE
03199	KLE>CK	03889	KURE>IER	04578	NOSCHTER
03209	KLEMM	03899	KUSCHEL	04588	NOTZ
03219	KLENTSCH	03909	KUSCH	04598	NONZ
03229	KLE*PFEL	03919	KUTSCHE>IER	04608	HUCKS
03239	KLIBBER	03929	KUUTSCH	04618	HUERKS
03249	KLIEN	03939	LAACH	04628	MULTIPLIZIEREA
03259	KLIEM	03949	LABBER	04638	HUPP
03269	KLIMP	03959	LABORE>IER	04648	HABBEL
03279	KLOMP	03969	LA*SCH	04658	NAPP
03289	KLONK	03979	LACK	04668	NASCHEL
03299	KLOTER	03989	LAGER	04678	NA*IF
03309	KLUCKS	03999	LAPP	04688	NE>CK
03319	KLUDDER	04009	LA*A*SCHT	04698	NE>I
03329	KLUNTSCH	04019	LA*IT	04708	NE>ITSCH
03339	KNABBEL	04029	LA*NS	04718	NEL
03349	KNABBER	04039	LA*STER	04728	NE*SCHEL
03359	KNACK	04049	LE*FT	04738	NETZ
03369	KNADDER	04059	LE>IER	04748	NIEWEL
03379	KNAL	04069	LEIER	04758	HOP
03389	KNASCHT	04079	LE>IN	04768	NOTZ
03399	KNA*TSCH	04089	LE>IS	04778	NUJHEL
03409	KNAUSER	04099	LE*NHEL	04788	NUMERE>IER
03419	KNAUTER	04109	LEESCHT	04798	HUREL
03429	KNAUTSCH	04119	LICH	04808	OFFRE>IER
03439	KNA*IP	04129	LIES	04818	ORAKEL
03449	KNA*PP	04139	LIEU	04828	COTR
03459	KNA*TSCH	04149	LIICHT	04838	POCK
03469	KNA*TZEL	04159	LIUISER	04848	PACHT
03479	KNE>CHEL	04169	LONP	04858	PACK
03489	KNE>CKS	04179	LUEW	04868	PAK
03499	KNE>CK	04189	LUUGER	04878	PADDEL
03509	KNE>I	04199	LUPP	04888	PAI
03519	KNIED	04209	LUTSCH	04898	PAKTE>IER
03529	KNIEWEL	04219	LUSS	04908	PANZER
03539	KNIEPS	04229	HASSE>IER	04918	PARK
03549	KNIWUEL	04239	HAUER	04928	FASS
03559	KNONTER	04249	RAUFEL	04938	PATSER
03569	KNUBBEL	04259	HAUL	04948	PAUER
03579	KNUER	04269	HAUSCHEL	04958	PAUS
03589	KNUJHEL	04279	HAUTSCH	04968	PAU
03599	KNUPP	04289	HA*A*SCHTER	04978	PA*CKEL
03609	KNUUTSCH	04299	HA*A*SSEL	04988	PA*IF
03619	KOL	04309	HA*ERDER	04998	PECH
03629	KOLL	04319	HE*FF	05008	PEFFER
03639	KOMPONE>IER	04329	HE>IN	05018	PE*FFER
03649	KONTER	04339	HELD	05028	PE>IL
03659	KOPPEL	04349	HELL	05038	PELL
03669	KOSCHTER	04359	HENG	05048	PENDEL
03679	KONP	04369	HE*NNER	05058	PE*HHEL
03689	KRAACH	04379	HE*NZ	05068	PE*HK
03699	KRABBEL	04389	HE*SCH	05078	PE*HN
03709	KRAID	04399	HE*SS	05088	PE*HSEL
03719	KRAUD	04409	HETER	05098	PE*SPER
03729	KRAUSEL	04419	HETZEL	05108	PICKENIER
03739	KRAZ	04429	RIBDEL	05118	PICK
03749	KRA*CH	04439	RIERSEL	05128	PIDDEL
03759	KRA*LL	04449	RIERK	05138	PIFF
03769	KRA*MP	04459	RIERHEL	05148	PIIPS
03779	KRE>CKEL	04469	MILLIQUH	05158	PILGER
03789	KREGE>IL	04479	NINN	05168	PINSCH

05170 PIP	05860 RANOSCHTER	06550 SCHIEFF
05180 PIPS	05870 RAPP	06560 SCHIFF
05190 PISACK	05880 RASCHT	06570 SCHILT
05200 PISS	05890 RAS	06580 SCHIFF
05210 PLAATSCH	05900 RAUCH	06590 SCHIFFS
05220 PLACKE>IER	05910 RAU	06600 SCHLABBER
05230 PLAIN	05920 RA*ACH	06610 SCHLADDER
05240 PLAK	05930 RA*ACHER	06620 SCHLADDER
05250 PLANG	05940 RA*CH	06630 SCHLAK
05260 PLANZ	05950 RA*DEL	06640 SCHLAUTER
05270 FLAPPER	05960 RA*PS	06650 SCHLAPP
05280 LATZ	05970 RA*Z	06660 SCHLA*in
05290 PLA*TSCH	05980 RECHN	06670 SCHLEPCK
05300 PLA*TTTEL	05990 REECH	06680 SCHLEIDER
05310 PLE>CK	06000 RE>CKEL	06690 SCHLENGER
05320 PLE>ISCHTER	06010 RE*FFEL	06700 SCHLENTER
05330 PLE*NN	06020 REBE>IER	06710 SCHLISBER
05340 FLE*NNER	06030 REBEL	06720 SCHLUSCHT
05350 FLO	06040 RE>IER	06730 SCHLUPF
05360 FLOHP	06050 REIN	06740 SCHRAACH
05370 FLOOSCHTER	06060 REN	06750 SCHRA*ACH
05380 FLOU	06070 RENBEL	06760 SCHNUDEL
05390 FLOUF	06080 RENNG	06770 SCHNADDER
05400 POLITIK	06090 RENNK	06780 SCHNABEL
05410 POLSTER	06100 RE*TSCH	06790 SCHNAU
05420 POMPEL	06110 RETT	06800 SCHNECK
05430 POSTELE>IER	06120 RICH	06810 SCHNEI
05440 POTER	06130 RIICHT	06820 SCHNE*TZEL
05450 FOUF	06140 RIPP	06830 SCHNOFFEL
05460 FONKER	06150 RIMPFL	06840 SCHNUDEL
05470 FONZ	06160 ROTZ	06850 SCHOCKE
05480 PRAFF	06170 RUBBEL	06860 SCHOSSEL
05490 PRAKTIZE>IER	06180 RUDDER	06870 SCHGUN
05500 PRAL	06190 RULL	06880 SCHRAU
05510 PRA*NE>IER	06200 RUHN	06890 SCHRE>CK
05520 RE>IV	06210 RUPP	06900 SCRE>IPS
05530 PRESS	06220 SABBEL	06910 SCHRUPF
05540 PRIEDEG	06230 SAIERZ	06920 SCHRUPP
05550 PROBE>IER	06240 SAK	06930 SCHUDDER
05560 PRODUZE>IER	06250 SALZ	06940 SCHUED
05570 PROFEZEI	06260 SAUR	06950 SCHUHN
05580 PROJHEZE>IER	06270 SAUS	06960 SCHUPP
05590 PROST	06280 SA*ACH	06970 SCHUA*ACH
05600 PROTZ	06290 SA*CH	06980 SCHNA*ERRZ
05610 PROUV	06300 SA*CKEL	06990 SCHWEHR
05620 PUDBEL	06310 SA*F	07000 SCHWENK
05630 PUDDER	06320 SA*H	07010 SE>CHER
05640 PUFF	06330 SA*WEL	07020 SEGEL
05650 PULL	06340 SCHADDER	07030 SEI
05660 PUNKTSCHWA*ACH	06350 SCHAFF	07040 SE>IN
05670 PUP	06360 SCHAF	07050 SE*ANEG
05680 PUSEL	06370 SCHAIM	07060 SEN
05690 PUTSCH	06380 SCHALT	07070 SENK
05700 QUACKS	06390 SCHAL	07080 SE*RIHER
05710 QUAK	06400 SCHANZ	07090 SICH
05720 QUATSCH	06410 SCHAFF	07100 SIFF
05730 QUELL	06420 SCHARJEREL	07110 SONN
05740 QUE*LL	06430 SCHARWENZEL	07120 SOUER
05750 QUE*TSCH	06440 SCHAUER	07130 SPAASS
05760 QUIICKS	06450 SCHAUTER	07140 SPADSE>IER
05770 QUIITSCH	06460 SCHA*ERF	07150 SPAN
05780 QUIK	06470 SCHA*ER	07160 SPE*DENIR
05790 QUOKEL	06480 SCHE>CK	07170 SPE>IN
05800 RAACH	06490 SCHEI	07180 SPELL
05810 RABBEL	06500 SCHEKER	07190 SPIEEL
05820 RA*CHEL	06510 SCHELL	07200 SPILL
05830 RACKER	06520 SCHE*LL	07210 SPICHE>IER
05840 RAMH	06530 SCHE>NG	07220 SPLA*ITER
05850 RAMONER	06540 SCHE*FF	07230 SPLE>CK

07240	SPRA*0	07930	TRINN
07250	SPRENG	07940	TRINNEL
07260	SPUER	07950	TROHN
07270	SPULL	07960	TROT
07280	STACK	07970	TROTZ
07290	STAN	07980	TROUN
07300	STA*	07990	TRUDEL
07310	STA*ERK	08000	TRUTSCH
07320	STA*IP	08010	TUCK
07330	STE*BS	08020	TUDEL
07340	STE*TZ	08030	TUFF
07350	STE>CKEL	08040	TUSCHEL
07360	STE>CKEL	08050	TUT
07370	STE>CK	08060	UECHT
07380	STE>IER	08070	UETEL
07390	STEIER	08080	U2
07400	STE*LL	08090	U*2
07410	STE*LP	08100	UAACH
07420	STE*NR	08110	UAARD
07430	STEMM	08120	UABEL
07440	STEMPEL	08130	UACKEL
07450	STE*PPEL	08140	UALZ
07460	STEPP	08150	UANDEL
07470	STIERN	08160	UANN
07480	STODE>IR	08170	UATSCHEL
07490	STOLPER	08180	UATA*CH
07500	STOLSE>IR	08190	UA*0
07510	STOPP	08200	UA*ANZEL
07520	STRAPP	08210	UE>CKEL
07530	STRA*	08220	UE>ITSCHE
07540	STRA*IF	08230	UE*LZ
07550	STRE>CK	08240	UE*NK
07560	STRECK	08250	UENN
07570	STRENN		
07580	STRENZ		
07590	STRE*PP		
07600	STRE*TZ		
07610	STRIEM		
07620	STROF		
07630	STRONPEL		
07640	STROTZ		
07650	STRUEMEL		
07660	STRUHR		
07670	STRUCKEL		
07680	STUCK		
07690	STUTZ		
07700	SUCKEL		
07710	SUDEL		
07720	SUERB		
07730	TAASCHT		
07740	TA*SSEL		
07750	TE>CK		
07760	TE>ITSCHE		
07770	TE*ALTER		
07780	TIERKEL		
07790	TIERN		
07800	TONNEL		
07810	TONP		
07820	TOUF		
07830	TOZ		
07840	TRAATSCH		
07850	TRAPP		
07860	TRAUER		
07870	TRAUFEL		
07880	TRE*MMEL		
07890	TRE*NDEL		
07900	TRE*PPEL		
07910	TRE*TZ		
07920	TRICHTER		

.MORPHLX(VERBIR) contient les racines irrégulières :

01110 GANK	01660 HULLEF	02210 LE
01120 GEE	01670 HUEL	02220 LOUE
01130 GE+	01680 HUE	02230 LE>IG
01140 GI	01690 HU	02240 LE+I
01150 GE>NG	01700 HE+LL	02250 LIT
01160 GONG	01710 HU	02260 LUAN
01170 GOUNG	01720 HE+L	02270 LUSE
01180 GUNG	01730 HOLL	02280 LUJ
01190 GAANG	01740 HIDD	02290 LIEJ
01200 GE>IF	01750 HIT	02300 LIEF
01210 GE>I	01760 HUTT	02310 LOOES
01220 GE>ING	01770 IERU	02320 LUSE
01230 GIA+	01780 IERF	02330 LE>ISS
01240 GOUF	01790 IESS	02340 HAACH
01250 GQUW	01800 IE+SCH	02350 HA
01260 GO	01810 IE+S	02360 HE>ICH
01270 GE>IF	01820 E+SCH	02370 HE>CH
01280 GEROD	01830 E+S	02380 HUSS
01290 GERIK	01840 GIESS	02390 HISS
01300 GERUCH	01850 JA+IZ	02400 NEHN
01310 GESA+I	01860 JAUT	02410 NAHN
01320 GESI	01870 KAUF	02420 NETZ
01330 GESOUCH	01880 KAF	02430 NAT
01340 GESE>ICH	01890 KEEF	02440 PA+IF
01350 GESE>NG	01900 KE>IF	02450 PAFF
01360 GESCHE>ICH	01910 KE+NN	02460 RE>IER
01370 GESCHE>I	01920 KANN	02470 ROLER
01380 GESCHIT	01930 KENN	02480 RUFF
01390 GESCHI	01940 KASCH	02490 RIFF
01400 GESCHUCH	01950 KA	02500 RICH
01410 GRAIF	01960 KOHN	02510 ROCH
01420 GRAFF	01970 KOHN	02520 SANG
01430 GRUEF	01980 KOUH	02530 SAUF
01440 GRUEW	01990 KE>IH	02540 SA+IF
01450 GRUF	02000 KLANN	02550 SOFF
01460 HAE	02010 KLE+NN	02560 SCHA+ISS
01470 HUCH	02020 KLOHN	02570 SCHAES
01480 HAL	02030 KLED	02580 SCHA+IZ
01490 HA+L	02040 KLEET	02590 SCHAT
01500 HA+T	02050 KRA+ISCH	02600 SCHE+D
01510 HOUL	02060 KRASCH	02610 SCHIED
01520 HE>IL	02070 KRE>IE	02620 SCHOTT
01530 HAAL	02080 KRIT	02630 SCHUT I
01540 HE>IER	02090 KRIS	02640 SCHE>ISS
01550 HOUER	02100 KRUG	02650 SCHOSS
01560 HA+NK	02110 KRUT	02660 SCHE>ING
01570 HA	02120 KRE>IG	02670 SCHE>NG
01580 HIEF	02130 KRICH	02680 SCHE+NN
01590 HOUNG	02140 KRICK	02690 SCHA+NN
01600 HORG	02150 LAF	02700 SCHARF
01610 HE>ING	02160 LOF	02710 SCHAFF
01620 HE>NG	02170 LUF	02720 SCHAF
01630 HE+LLEF	02180 LEEF	02730 SCHLA+ICH
01640 HELLEF	02190 LEI	02740 SCHLACH
01650 HULLEF	02200 LA+IT	02750 SCHLA+IF

02760 SCHLAF  
 02770 SCHLEEF  
 02780 SCHLEF  
 02790 SCHLOF  
 02800 SCHLE>IF  
 02810 SCHLO  
 02820 SCHNE\*LI  
 02830 SCHNALT  
 02840 SCHNOLT  
 02850 SCHNOLZ  
 02860 SCHNOLT  
 02870 SCHNA\*IZ  
 02880 SCHNAUT  
 02890 SCHNEID  
 02900 SCHNUT  
 02910 SCHNID0  
 02920 SCHNE>I  
 02930 SCHNEI  
 02940 SCHNECH  
 02950 SCHNEK  
 02960 SCHREIW  
 02970 SCHRINU  
 02980 SCHRUF  
 02990 SCHREIF  
 03000 SCHRIF  
 03010 SCHRIF  
 03020 SCHWIER  
 03030 SCHWUER  
 03040 SCHWA\*IZ  
 03050 SCHWAT  
 03060 SE\*IZ  
 03070 SE>IZ  
 03080 SOUZ  
 03090 SIESS  
 03100 SIEW  
 03110 SETZ  
 03120 SAT  
 03130 SOLL  
 03140 SOT  
 03150 SO  
 03160 SEE  
 03170 SIE  
 03180 SPA\*IZ  
 03190 SPAU  
 03200 SPANN  
 03210 SPONN  
 03220 SPA\*R  
 03230 SPAAR  
 03240 SPIER  
 03250 SPUER  
 03260 SPRANG  
 03270 SPRE>NG  
 03280 STIECH  
 03290 STOUCH  
 03300 STACH

03310 STEDICH  
 03320 STIEL  
 03330 STUEL  
 03340 STIER0  
 03350 STIER0  
 03360 STIERZ  
 03370 STUERT  
 03380 ST0  
 03390 STI  
 03400 STEE  
 03410 STELL  
 03420 STALL  
 03430 STOUNG  
 03440 STONG  
 03450 STE>ING  
 03460 STE>NG  
 03470 STE>I  
 03480 STAND  
 03490 STAN  
 03500 STOUSS  
 03510 STE>ISS  
 03520 STRA\*ICH  
 03530 STRACH  
 03540 TAUSCH  
 03550 TOSCH  
 03560 TREFF  
 03570 TRAFF  
 03580 TRIED  
 03590 TRE\*IT  
 03600 TRUED  
 03610 TRATT  
 03620 VERGIES  
 03630 VERGLA\*ICH  
 03640 VERGLACH  
 03650 VERLE>IER  
 03660 VERLUER  
 03670 VERSPRIECH  
 03680 VERSPRACH  
 03690 VERUERTE\*L  
 03700 VERUERTEL  
 03710 WAR  
 03720 WA\*R  
 03730 WA\*ER0  
 03740 WIER  
 03750 WIESCH  
 03760 WA\*ER  
 03770 W0R  
 03780 WOLL  
 03790 WE>IL  
 03800 WE\*LL  
 03810 WE\*SS  
 03820 WE\*SS  
 03830 WOUSS  
 03840 W0SS  
 03850 WE>ISS

03860 W0R  
 03870 W0SCH  
 03880 WUESS  
 03890 WUESS  
 03900 ZE>I  
 03910 Z0CH  
 03920 ZUNN  
 03930 ZEL  
 03940 ZE

Pour le néerlandais, on résout simplement le problème que posent quelques règles orthographiques :

- Lorsqu'un mot contenant une voyelle impure (dans une syllabe fermée) prend une terminaison (-e, -en...), on redouble dans l'écriture la consonne qui suit la voyelle impure.

- Les voyelles pures a, e, o et u sont simples dans une syllabe ouverte et géminées dans une syllabe fermée.

- les sourdes se transforment en sonores (assimilation) à l'intérieur ou à la fin d'une syllabe et s'écrivent :

v --> f  
z --> s

S'il est possible de calculer les formes à partir de quelques règles et d'une dizaine d'instructions, il est beaucoup plus pratique et rapide d'en tenir compte dans la présentation des données. C'est ainsi que figurent dans .MORPHOL(VERBRG) :

ØØ31Ø	BEVATT	et	ØØ32Ø	BEVAT	de BEVATTEN contenir
ØØ24Ø	BEGROT	et	ØØ23Ø	BEGRODT	de BEGROTEN évaluer
Ø114Ø	DRAV	et	Ø113Ø	DRAAF	de DRAVEN trotter
Ø181Ø	GRAZ	et	Ø182Ø	GRAAS	de GRAZEN brouter

La lecture de ces fichiers par variables basées est extrêmement rapide. On peut donc rentrer toutes les variantes, ce qui donne d'autres possibilités, dans le cas des verbes irréguliers notamment, pour lesquels la modification du radical n'est pas toujours uniforme. Ceci permet de différencier quelques formes, grâce à l'adjonction d'un code dans le fichier .MORPHOL(VERBIR) :

Ø144Ø ØHANG Ø --> pas de différenciation, le radical HANG se retrouvant au participe passé de HANGEN pendre.

Ø141Ø \*GRIP \* --> radical à l'infinitif et à l'indicatif présent (GREET à l'imparfait, GEGREPEN au participe passé) de GRIPSEN saisir

Ø414Ø STORV radical au participe passé (STERV infinitif, indicatif présent et STIERF imparfait) de STERVEN, mourir

#### .MORPHOL(VERBRG) :

Ø166Ø GIECHEL	Ø185Ø GRIPP	Ø203Ø HARTIS
Ø167Ø GILL	Ø186Ø GRIP	Ø204Ø HANTEER
Ø168Ø GIL	Ø187Ø GROEI	Ø205Ø HANTER
Ø169Ø GISS	Ø188Ø GROET	Ø206Ø HAND
Ø170Ø GIS	Ø189Ø GROHM	Ø207Ø HAAT
Ø171Ø GLANZ	Ø190Ø GRON	Ø208Ø HAT
Ø172Ø GLIPP	Ø191Ø GROND	Ø209Ø HECHT
Ø173Ø GLIP	Ø192Ø GUNN	Ø210Ø HELDER
Ø174Ø GLOEI	Ø193Ø GUN	Ø211Ø HELS
Ø175Ø GLUUR	Ø194Ø HAAST	Ø212Ø HEERS
Ø176Ø GLUR	Ø195Ø HAKK	Ø213Ø HEUG
Ø177Ø GOOI	Ø196Ø HAK	Ø214Ø HJEG
Ø178Ø GLOF	Ø197Ø HAAL	Ø215Ø HINDER
Ø179Ø GOLV	Ø198Ø HAL	Ø216Ø HINK
Ø180Ø GRAAI	Ø199Ø HANDEL	Ø217Ø HITS
Ø181Ø GRAZ	Ø200Ø HANDHAAF	Ø218Ø HOED
Ø182Ø GRAAS	Ø201Ø HANDHAV	Ø219Ø HOEV
Ø183Ø GRIJNZ	Ø202Ø HANDIG	Ø220Ø HOOG
Ø184Ø GRINNIK		

02210	HOG	02760	KLEV	03310	LEEN
02220	HOLL	02770	KLOPP	03320	LEN
02230	HOL	02780	KLOP	03330	LEER
02240	HOON	02790	KNAPP	03340	LER
02250	HON	02800	KNAP	03350	LETT
02260	HOOI	02810	KNALL	03360	LET
02270	HOOP	02820	KNAL	03370	LEUR
02280	HOP	02830	KNIKK	03380	LEID
02290	HOOR	02840	KNIK	03390	LEHNER
02300	HOR	02850	KNIEL	03400	LENG
02310	HUIL	02860	KNIPPER	03410	LEUN
02320	HUIVER	02870	KNIPP	03420	LEVER
02330	HUIZ	02880	KNIP	03430	LEEF
02340	HUIS	02890	KNOEI	03440	LEV
02350	HUNKER	02900	KNOOP	03450	LIJN
02360	HURK	02910	KNOP	03460	LIJN
02370	HUUR	02920	KOESTER	03470	LIJK
02380	HUR	02930	KOOK	03480	LIK
02390	HUTS	02940	KOK	03490	LIEV
02400	INNER	02950	KOMMER	03500	LOEI
02410	IJSBEER	02960	KONDIG	03510	LOER
02420	IJSBER	02970	KOPPEL	03520	LOKK
02430	JANK	02980	KOSTIG	03530	LOK
02440	JENK	02990	KOST	03540	LOH
02450	JOKK	03000	KRABBEL	03550	LOON
02460	JOK	03010	KRABB	03560	LOSS
02470	JUICH	03020	KRAAK	03570	LOS
02480	KAARST	03030	KRAK	03580	LOOS
02490	KAATS	03040	KREUN	03590	LOZ
02500	KAKK	03050	KRESS	03600	LOV
02510	KAK	03060	KRAS	03610	LOOF
02520	KAMP	03070	KRIJS	03620	LUID
02530	KAPP	03080	KRONKEL	03630	LUISTER
02540	KAP	03090	KROON	03640	LUKK
02550	KLEDER	03100	KRON	03650	LUK
02560	KLENTON	03110	KUCH	03660	MAAI
02570	KLENTON	03120	KVAAK	03670	NACHTIG
02580	KENN	03130	KVAKK	03680	MAAK
02590	KEN	03140	KVAK	03690	MAK
02600	KEER	03150	KVEEK	03700	MANN
02610	KER	03160	KVEKK	03710	MAN
02620	KLAAG	03170	KUEL	03720	MARTEL
02630	KLAG	03180	KWETS	03730	NATIG
02640	KLAPP	03190	LAAG	03740	NEERDER
02650	KLAP	03200	LAG	03750	NELD
02660	KEUR	03210	LAKK	03760	NELK
02670	KLAAR	03220	LAK	03770	NEEH
02680	KLAR	03230	LANN	03780	NENG
02690	KLEED	03240	LAN	03790	NEN
02700	KLED	03250	LAST	03800	NERK
02710	KLEEM	03260	LAZER	03810	NIKK
02720	KLEH	03270	LEDIG	03820	NIK
02730	KLEER	03280	LEEG	03830	NINK
02740	KLER	03290	LEGG	03840	NINN
02750	KLEEF	03300	LEG	03850	NIN

03860	MISS	04410	PLAN	04960	REKK
03870	MIS	04420	PLANT	04970	REK
03880	MOEDIG	04430	PLEIT	04980	RENN
03890	MOED	04440	FLICHT	04990	REN
03900	MOEI	04450	PLOFF	05000	RENN
03910	MONPEL	04460	PLOF	05010	REN
03920	MOORD	04470	PLOOI	05020	REPP
03930	MORREL	04480	POETS	05030	REP
03940	MURMEL	04490	FLUKK	05040	RICHT
03950	NADER	04500	PLUK	05050	RILL
03960	NAAK	04510	POOG	05060	RIL
03970	NAK	04520	POG	05070	RING
03980	NAUW	04530	PULIJST	05080	RITSEL
03990	NEDER	04540	POLS	05090	ROEI
04000	NEGER	04550	POMP	05100	ROER
04010	NEIG	04560	PRAAT	05110	RUST
04020	NESTEL	04570	PRAT	05120	ROER
04030	NIETIG	04580	PREK	05130	ROOK
04040	NIEUW	04590	PREEK	05140	ROK
04050	NIEZ	04600	PRENT	05150	ROLL
04060	NIES	04610	PRESTER	05160	ROL
04070	NOEM	04620	PRIJZ	05170	ROMMEL
04080	NUTT	04630	PRIJS	05180	ROMPEL
04090	NUT	04640	PRIKKEL	05190	ROTT
04100	DEFEN	04650	PRIKK	05200	ROTT
04110	OFFER	04660	PRIK	05210	ROUV
04120	OOGST	04670	PROBEER	05220	ROY
04130	OOG	04680	PROBER	05230	ROOF
04140	OG	04690	PROEF	05240	RUCHT
04150	OPFER	04700	PROEV	05250	RUIL
04160	OPEN	04710	PUTT	05260	RUIH
04170	ORDEN	04720	PUT	05270	RUIS
04180	PAKK	04730	RAD	05280	RUKK
04190	PAK	04740	RASP	05290	RUK
04200	PAL	04750	RASS	05300	RUST
04210	PASSER	04760	RAS	05310	SCHAATS
04220	PAR	04770	RAAS	05320	SCHADIG
04230	PASS	04780	RAZ	05330	SCHAAD
04240	PAS	04790	RAK	05340	SCHAD
04250	PEINZ	04800	RAAK	05350	SCHAFF
04260	PIRK	04810	RAKHEL	05360	SCHAF
04270	PLAATS	04820	RECHT	05370	SCHAKEL
04280	PRAAT	04830	REDD	05380	SCHAKER
04290	PROEF	04840	REDENER	05390	SCHAKEER
04300	PRAT	04850	REDENER	05400	SCHAAK
04310	PROEV	04860	RED	05410	SCHAAN
04320	FEUTER	04870	REGEL	05420	SCHAK
04330	PIEKER	04880	REGEN	05430	SCHAK
04340	PIKK	04890	REGER	05440	SCHARREL
04350	PIK	04900	REID	05450	SCHATT
04360	PLAAG	04910	REIK	05460	SCHAT
04370	PLAG	04920	REINIG	05470	SCHAAF
04380	PLAKK	04930	REIZ	05480	SCHAV
04390	PLAK	04940	REIS	05490	SCHUID
04400	PLANN	04950	REKEN	05500	SCHREEL

05510	SCHEL	06260	SCHUT	06610	SPAN
05520	SCHERN	06070	SCHUU	06620	SPAR
05530	SCHERP	06080	SEFF	06630	SPARR
05540	SCHETS	06090	SEF	06640	SPATT
05550	SCHEUR	06100	SEIN	06650	SPAT
05560	SCHIED	06110	SEIZ	06660	SPELL
05570	SCHIKK	06120	SEIS	06670	SPEEL
05580	SCHIK	06130	SIER	06680	SPEL
05590	SCHILDER	06140	SISS	06690	SPEUR
05600	SCHILL	06150	SIS	06700	SPIED
05610	SCHIL	06160	SJOUW	06710	SPIEGEL
05620	SCHINP	06170	SKI	06720	SPIER
05630	SCHITTER	06180	SLAG	06730	SPLITE
05640	SCHOB	06190	SLAG	06740	SPOEL
05650	SCHOB	06200	SLAAF	06750	SPOTT
05660	SCHOEI	06210	SLAV	06760	SPOT
05670	SCHOKK	06220	SLEENTER	06770	SPREID
05680	SCHOK	06230	SLEEP	06780	SPIJ
05690	SCHONMEL	06240	SLEP	06790	STAK
05700	SCHOPP	06250	SLIKK	06800	STAAK
05710	SCHOP	06260	SLIK	06810	STARRE
05720	SCHORS	06270	SLINGER	06820	STAM
05730	SCHORREL	06280	SLISS	06830	STAPEL
05740	SCHOTEL	06290	SLIS	06840	STAPP
05750	SCHRAG	06300	SLOMNER	06850	STAP
05760	SCHRAAG	06310	SLOP	06860	STARR
05770	SCHRAAL	06320	SLOOP	06870	START
05780	SCHRAL	06330	SLOOF	06880	STAR
05790	SCHRAMM	06340	SLOV	06890	STED
05800	SCHRAM	06350	SLUIJER	06900	STEED
05810	SCHRANK	06360	SMAKK	06910	STELL
05820	SCHRAPP	06370	SMAK	06920	STEL
05830	SCHRAP	06380	SMAAL	06930	STARRE
05840	SCHREEUW	06390	SMAAL	06940	STAM
05850	SCHREI	06400	SMEED	06950	STAND
05860	SCHROBB	06410	SNEED	06960	STILL
05870	SCHROB	06420	SNEK	06970	STIL
05880	SCHROL	06430	SNEEK	06980	STENPEL
05890	SCHROOL	06440	SNETT	06990	STERK
05900	SCHROMPEL	06450	SNET	07000	STICHT
05910	SCHOUW	06460	SNOR	07010	STIKK
05920	SCHROOI	06470	SNOCR	07020	STIK
05930	SCHUDD	06480	SNAPP	07030	STILER
05940	SCHUD	06490	SNAP	07040	STILEER
05950	SCHUIER	06500	SNAUW	07050	STIPPEL
05960	SCHUIFEL	06510	SHELL	07060	STOK
05970	SCHUIN	06520	SNEL	07070	STOOK
05980	SCHUIN	06530	SNEEUW	07080	STOT
05990	SCHULDIG	06540	SNIKK	07090	STOOT
06000	SCHULD	06550	SNIK	07100	STOFF
06010	SCHULP	06560	SNOER	07110	STOP
06020	SCHUR	06570	SNUFFEL	07120	STORRE
06030	SCHUUR	06580	SORRE	07130	STOOR
06040	SCHUTTER	06590	SOR	07140	STORRE
06050	SCHUTT	06600	SPANN	07150	STOR

07160 STRAFF	07710 TRILL	08260 WAG
07170 STRAF	07720 TRIL	08270 WAAG
07180 STRAL	07730 TREUR	08280 WAK
07190 STRAAL	07740 TROOST	08290 WANDEL
07200 STRAND	07750 TROUW	08300 WANKEL
07210 STREKK	07760 TUIG	08310 WAPEN
07220 STREK	07770 TWIJFEL	08320 WAPPER
07230 STREEL	07780 TWIST	08330 WARM
07240 STREEP	07790 UITNODIG	08340 WAST
07250 STREL	07800 VAARDIG	08350 WEDD
07260 STREP	07810 VATT	08360 WED
07270 STREEF	07820 VAT	08370 WEG
07280 STREV	07830 VEILIG	08380 WEEG
07290 STROOI	07840 VELL	08390 WEIGER
07300 STROOM	07850 VEL	08400 WEKK
07310 STROM	07860 VERANDER	08410 WEK
07320 STRUIKEL	07870 VERDICT	08420 WELDIG
07330 STUIT	07880 VERBEELD	08430 WEND
07340 STUIV	07890 VERBETER	08440 WENK
07350 STUIF	07900 VERDOMM	08450 WENN
07360 STUUR	07910 VERDOM	08460 WENS
07370 STUR	07920 VERGROOT	08470 WEN
07380 SUKKEL	07930 VERGROT	08480 WERK
07390 TAL	07940 VERKLEINER	08490 WEZENLIJK
07400 TAAL	07950 VERHINDER	08500 WIJD
07410 TAPP	07960 VERV	08510 WIJL
07420 TAP	07970 VEERF	08520 WIJZIG
07430 TAST	07980 VELD	08530 WIKKEL
07440 TEKEN	07990 VESTIG	08540 WINKEL
07450 TEISTER	08000 VIER	08550 WINTER
07460 TELL	08010 VLOED	08560 WIPP
07470 TEL	08020 VLOEI	08570 WIP
07480 TER	08030 VLOEK	08580 WISSEL
07490 TEER	08040 VLUCHT	08590 WISS
07500 TIER	08050 VOEG	08600 WIS
07510 TIKK	08060 VOEL	08610 WITTIG
07520 TIK	08070 VOER	08620 VOED
07530 TILL	08080 VOETBALL	08630 WOEK
07540 TIL	08090 VOLG	08640 WOEST
07550 TINNER	08100 VORDER	08650 WONDER
07560 TINTEL	08110 VORM	08660 WOND
07570 TOG	08120 VOOU	08670 WON
07580 TOOG	08130 VREDIG	08680 WOON
07590 TOETS	08140 VREEMD	08690 WOORDIG
07600 TOX	08150 VREZ	08700 WORSTEL
07610 TOOM	08160 VREES	08710 WORTEL
07620 TOOI	08170 VRUCHT	08720 WREK
07630 TOON	08180 VULL	08730 WUIV
07640 TON	08190 VUL	08740 WUIF
07650 TRACTEER	08200 VAST	08750 ZAAI
07660 TRACTER	08210 WAAI	08760 ZAAG
07670 TRAIN	08220 WAARDER	08770 ZAG
07680 TRAPP	08230 WAARDEER	08780 ZAKK
07690 TRACHT	08240 WAARDIG	08790 ZAK
07700 TRAP	08250 WACHT	08800 ZEGEN

08810 ZEIL  
 08820 ZETT  
 08830 ZET  
 08840 ZEUR  
 08850 ZIEL  
 08860 ZINN  
 08870 ZIN  
 08880 ZOEN  
 08890 ZORG  
 08900 ZUCHT  
 08910 ZUIK  
 08920 ZUIVER  
 08930 ZWAAI  
 08940 ZWEEF  
 08950 ZUEV  
 08960 ZWEEL  
 08970 ZWEL  
 08980 ZWEEP  
 08990 ZWEP  
 09000 ZWETS  
 09010 ZWEET  
 09020 ZWET  
 09030 ZUEV

**MORPHOL(VERBIR) :**

01660 0HIEUU	02020 *KRIJT	02390 0LIEP
01670 0JAAG	02030 0KREET	02400 0HELK
01680 0JAG	02040 0KRET	02410 0MOLK
01690 0JOEG	02050 *KRIMP	02420 0HEET
01700 0KERV	02060 0KROMP	02430 0RET
01710 0KERF	02070 *KKUIP	02440 0MAT
01720 0KORF	02080 0KROOP	02450 *NIJD
01730 0KORV	02090 0KROP	02460 0NEED
01740 *KIEZ	02100 0KUNN	02470 0RED
01750 0KIES	02110 0KUN	02480 0HOET
01760 0KOOS	02120 0KAN	02490 0HOES
01770 0KOZ	02130 0KON	02500 0HOOG
01780 *KIJK	02140 0LACH	02510 0NOG
01790 0KEEK	02150 0LAAD	02520 0RAG
01800 0KEK	02160 0LAD	02530 0HOCH
01810 0KIJV	02170 0LAAT	02540 0NEEN
01820 0KIJF	02180 0LAT	02550 0NEE
01830 0KEEF	02190 0LIET	02560 0NAN
01840 0KEV	02200 0LEES	02570 0NOH
01850 *KLINN	02210 0LEZ	02580 0NIJG
01860 0KLIN	02220 0LAZ	02590 0NEEG
01870 0KLOMH	02230 0LAS	02600 0NEG
01880 0KLOH	02240 *LIEG	02610 *NIJP
01890 *KLINK	02250 0LOOG	02620 0NEEP
01900 0KLONK	02260 0LOG	02630 0NEP
01910 *KNIJP	02270 *LIGG	02640 0ONTGINH
01920 0KNEEP	02280 0LIG	02650 0ONTGIN
01930 0KNEP	02290 0LAG	02660 0ONTGONH
01940 0KOM	02300 0LEG	02670 0ONTGON
01950 0KUM	02310 *LIJD	02680 0ONTLUIK
01960 0KOOOP	02320 0LEED	02690 0ONTLOOK
01970 *KOP	02330 0LED	02700 0ONTLOK
01980 0KOCH	02340 *LIJK	02710 0PLEEG
01990 *KRIJG	02350 0LEEK	02720 *PLEG
02000 0KREEG	02360 0LEK	02730 0PLACH
02010 0KREG	02370 0LOOP	02740 0PLUIZ
	02380 0LOP	02750 0PLUIS

02760	0PLOOS	03310	0SCHOOOL	03860	0SPRAK
02770	0PLOZ	03320	0SCHOL	03870	0SPROK
02780	*PRIJZ	03330	0SCHUIF	03880	*SPRING
02790	0PRIJS	03340	*SCHUIV	03890	0SPRONG
02800	0PREES	03350	0SCHOOOF	03900	0SPRUIT
02810	0PREZ	03360	0SCHOV	03910	0SPROOT
02820	0RAAD	03370	0SLAA	03920	0SPROT
02830	0RAD	03380	0SLA	03930	0SPUG
02840	0RIED	03390	0SLOEG	03940	0SPOOG
02850	0RIEK	03400	0SLAAP	03950	0SPOG
02860	*RUIK	03410	0SLAP	03960	*SPUIT
02870	0ROOK	03420	0SLIEP	03970	0SPOOT
02880	0ROK	03430	0SLIJP	03980	0SPOT
02890	*RIJD	03440	0SLEEP	03990	0STAA
02900	0REED	03450	0SLEP	04000	0STA
02910	0RED	03460	*SLIJT	04010	0STOND
02920	0RIJZ	03470	0SLEET	04020	*STECK
02930	0RIJS	03480	0SLET	04030	*STEK
02940	0REES	03490	0SLINK	04040	0STAK
02950	0REZ	03500	0SLONK	04050	0STOK
02960	0ROEP	03510	*SLUIP	04060	*STEEL
02970	0RIEP	03520	0SLOOP	04070	*STEL
02980	0SCHEID	03530	0SLOP	04080	0STAL
02990	*SCHELD	03540	*SLUIT	04090	0STOL
03000	0SCHOLD	03550	0SLOOT	04100	*STEERF
03010	0SCHEND	03560	0SLOT	04110	*STERV
03020	0SCHOND	03570	*SNELT	04120	0STIERF
03030	*SCHENK	03580	0SNOLT	04130	0STIERV
03040	0SCHONK	03590	*SMIJT	04140	0STORV
03050	*SCHEPP	03600	0SNEET	04150	*STIJF
03060	*SCHEP	03610	0SMET	04160	*STIJV
03070	0SCHIEP	03620	*SNIJD	04170	0STEEF
03080	0SCHAP	03630	0SNEED	04180	0STEV
03090	0SCHEER	03640	0SNED	04190	0STIJG
03100	*SCHER	03650	0SNUIT	04200	0STEEG
03110	0SCHOOR	03660	0SNOOT	04210	0STEG
03120	0SCHOR	03670	0SNOT	04220	0STINK
03130	*SCHJET	03680	0SNUIF	04230	0STONK
03140	0SCHOOT	03690	*SNUIV	04240	0STOOT
03150	0SCHOT	03700	0SNOOF	04250	0STOT
03160	*SCHIJN	03710	0SNOV	04260	0STIET
03170	0SCHEEN	03720	0SPANN	04270	*STRIJD
03180	0SCHEN	03730	0SPAN	04280	0STREED
03190	*SCHRIJD	03740	*SPIJT	04290	0STRED
03200	0SCHREED	03750	0SPEET	04300	*STRIJK
03210	0SCHRED	03760	0SPET	04310	0STREEK
03220	*SCHRIJF	03770	0SPINN	04320	0STREX
03230	*SCHRIJV	03780	0SPIN	04330	*STUIF
03240	0SCHREEF	03790	0SPONN	04340	*STUIV
03250	0SCHREY	03800	0SPON	04350	0STOOF
03260	*SCHRIKK	03810	*SPLIJT	04360	0STOV
03270	*SCHRIK	03820	0SPLEET	04370	0TIJG
03280	0SCHROKK	03830	0SFLET	04380	0TOOG
03290	0SCHROK	03840	*SPREEK	04390	0TOG
03300	0SCHUIL	03850	*SPREK	04400	0TRED

04410	0TRAD	04960	0VROR	05510	0ZOND
04420	0TRED	04970	0VAAI	05520	0ZIE
04430	*TREFF	04980	0UOEI	05530	0ZAG
04440	*TREF	04990	0UASS	05540	0ZIJN
04450	0TROFF	05000	0UAS	05550	0UAR
04460	0TROF	05010	0UIES	05560	*ZING
04470	*TREKK	05020	0UEEG	05570	0ZONG
04480	*TREK	05030	*WEG	05580	*ZINK
04490	0TROKK	05040	0UOOG	05590	0ZONK
04500	0TROK	05050	0UOG	05600	*ZINN
04510	0VALL	05060	*WERP	05610	*ZIN
04520	0VAL	05070	0UIERP	05620	0ZONN
04530	0VIEL	05080	0URP	05630	0ZON
04540	0VANG	05090	0VERF	05640	0ZIED
04550	0VING	05100	0VERV	05650	0ZOD
04560	0VAAR	05110	0UIERF	05660	*ZITT
04570	0VAR	05120	0UIERV	05670	*ZIT
04580	0VOER	05130	0UORV	05680	0ZAT
04590	*VECHT	05140	0WEET	05690	0ZET
04600	0VOCHT	05150	0WET	05700	*ZOEK
04610	*VERDRIET	05160	0WIST	05710	0ZUCH
04620	0VERDROU	05170	0WEEF	05720	0ZOUT
04630	0VERDROT	05180	0WEV	05730	*ZUIG
04640	*VERDIJN	05190	*WIJK	05740	0ZOOG
04650	0VERDUEEN	05200	0WEEK	05750	0ZOG
04660	0VERDUEN	05210	0WEK	05760	0ZUIP
04670	0VERGEET	05220	*WIJT	05770	0ZOOF
04680	0VERGET	05230	0WET	05780	0ZOP
04690	0VERGAT	05240	*WIJS	05790	*ZULL
04700	*VERLIEZ	05250	0VEES	05800	0ZUL
04710	*VERLIES	05260	0WEZ	05810	0ZAL
04720	0VERLOOR	05270	*WILL	05820	0ZOU
04730	0VERLOR	05280	0WIL	05830	0ZWELG
04740	0VERSLIND	05290	0VOU	05840	0ZWOLG
04750	0VERSLOND	05300	*WIND	05850	*ZWELL
04760	*VIND	05310	0WOND	05860	0ZWEL
04770	0VOND	05320	*WINN	05870	0ZWOLL
04780	*VLECHT	05330	*WIN	05880	0ZWOL
04790	0VLOCHT	05340	0WONN	05890	*ZWEEN
04800	*Vlieg	05350	0WON	05900	0ZWEN
04810	0VLOOG	05360	0WORD	05910	0ZWONK
04820	0VLOG	05370	0WERD	05920	0ZWON
04830	0VLIET	05380	0WREEK	05930	*ZWEER
04840	0VLOOT	05390	0WREK	05940	*ZUER
04850	0VLOT	05400	0WROK	05950	0ZWUER
04860	0VOUW	05410	0WRIJF	05960	0ZWUOR
04870	0VRAAG	05420	*WRIJV	05970	0ZWOR
04880	0VRAG	05430	0WREEF	05980	*ZWERF
04890	0VROEG	05440	0WREV	05990	*ZWERV
04900	0VREET	05450	*WRING	06000	0ZWIERF
04910	0VRET	05460	0WRONG	06010	0ZWIERV
04920	0VRAT	05470	0ZEGG	06020	0ZWORV
04930	*VRIEZ	05480	0ZEG	06030	*ZWIJG
04940	0VRIES	05490	0ZEI	06040	0ZWEEG
04950	0VROOR	05500	*ZEND	06050	0ZWEG

**Pour l'allemand :**

Une analyse plus précise requiert un code plus complet. Le code actuel utilisé dans .MORPHAL(VERBRG) a trois caractères :

"1" en première position signifie qu'il s'agit d'un verbe régulier.

"\*" en seconde position : verbe à préverbe inséparable. L'absence de ge- et la terminaison -et ou -t n'impliquera pas l'indicatif présent seul. Il y aura ambiguïté avec le participe passé.

"#" en seconde position : le ge- est ambigu pour certaines formes. C'est la série *Ge- brauchen, gedulden, gehorchen, gehören...* que nous étudierons au paragraphe 4.3.6.3.2.2

"&" en troisième position : verbes en -eln ou -ern

"2" en troisième position : le radical se retrouve dans le fichier des verbes irréguliers :

GELANG --> GELINGEN (a-u) réussir  
--> GELANGEN arriver

**.MORPHAL(VERBRG) :**

00020 811	00210 100BAU	00400 100BA+KPF
00030 0030	00220 1+00BEFREI	00410 100DANK
00040 6334	00230 1+00BEGU+NSTIG	00420 100DARB
00050 100ACHT	00240 1+00BENO+TIG	00430 100DARR
00060 100A+CHT	00250 1+00BEREIT	00440 100DAUER
00070 100A+CHZ	00260 100BESSER	00450 100DAU
00080 100ACKER	00270 1+00BETEILIG	00460 100DECK
00090 100A+DER	00280 1+00BEWA+HR	00470 100DEHN
00100 100ANN	00290 1+00BUEG	00480 100DENGEL
00110 100A+KREL	00300 100BILD	00490 100DEUTLICH
00120 100AHN	00310 100BLICK	00500 100DEUT
00130 100ALTER	00320 100BLICK	00510 100DICHT
00140 100A+NDER	00330 100BRAUCH	00520 100DIEN
00150 100ARBEIT	00340 100BREIT	00530 100DOPPEL
00160 100A+RGER	00350 100BRENG	00540 100DORR
00170 100ATH	00360 100BRU+T	00550 100DRAHT
00180 100A+US+ER	00370 100DA+KNER	00560 100DRA+NGEL
00190 100BAD	00380 100DA+NN	00570 100DRESCHEL
00200 100BAHN	00390 100DANKF	00580 100DREH

00590	100DRILL	01140	100FU*ROER	01690	1*00GEUA*LTIG
00600	100DRITTEL	01150	100FORN	01700	1*00GEUA*RTIG
00610	100DROCH	01160	100FORECH	01710	1*00GEUA*HH
00620	100DRO*HH	01170	100FRAG	01720	100GIER
00630	100DROSSEL	01180	100FRA*E	01730	100GIFT
00640	100DRUCK	01190	100FREU	01740	100GIPFEL
00650	100DRU*CK	01200	100FRIEDIG	01750	100GIPS
00660	100DRUSCH	01210	100FRISCH	01760	100GLA*HZ
00670	100DUCK	01220	100FRIST	01770	100GLAS
00680	100DUFT	01230	100FUG	01780	100GLATT
00690	100DULD	01240	100FU*G	01790	100GLAUB
00700	100DUNKEL	01250	100FU*HL	01800	100GLEIS
00710	100DUNST	01260	100FU*HR	01810	100GLIEDER
00720	100DU*HST	01270	100FU*LL	01820	100GLITSCH
00730	100DUSCH	01280	100FUNKEL	01830	100GLITZER
00740	100EHR	01290	100FUNK	01840	100GLU*CK
00750	100EIFER	01300	100FUTTER	01850	100GLU*H
00760	100EIGN	01310	100FU*TTER	01860	100GNUM*G
00770	100EIL	01320	100FU*RCHT	01870	100GOLD
00780	100EIS	01330	100GABEL	01880	100GOM*HN
00790	100ENDIG	01340	100GA*HH	01890	100GRAU
00800	100END	01350	100GA*HZ	01900	100GRENZ
00810	100ENG	01360	100GAS	01910	100GRILL
00820	100ERB	01370	100GEBRAUCH	01920	100GRU*ND
00830	1*00ERMO*GLICH	01380	100GEDULD	01930	100GRU*SB*
00840	1*00ERNUTIG	01390	100GEGORCH	01940	100GUCK
00850	100FA*CHEL	01400	100GEGO*H	01950	100GU*RT
00860	100FA*LSCH	01410	100GELEIT	01960	100GU*TT
00870	100FALT	01420	100GELOB	01970	100HA*CK
00880	100FA*RB	01430	100GEREICH	01980	100HAFT
00890	100FASER	01440	100GEREU	01990	100HAGEL
00900	100FASS	01450	100GERUH	02000	100HAK
00910	100FAS*	01460	100GESEHN	02010	100HALL
00920	100FEDER	01470	100GETRAU	02020	100HA*HNER
00930	100FEG	01480	100GEWAHR	02030	100HANDEL
00940	100FEIL	01490	100GEUA*HR	02040	100HARK
00950	100FERN	01500	100GEZIEH	02050	100HARN
00960	100FERTIG	01510	1*00GEBAR	02060	100HARR
00970	100FESSEL	01520	1*00GEB*RD	02070	100HA*RT
00980	100FESTIG	01530	1*00GEJU*HR	02080	100HASPEL
00990	100FEUCHT	01540	1*00GEFA*HRD	02090	100HAFT
01000	100FEUER	01550	1*00GELANG	02100	100HAUCH
01010	100FIEBER	01560	1*00GELU*ST	02110	100HA*UFEL
01020	100FILM	01570	1*00GENU*G	02120	100HECHEL
01030	100FLANSCH	01580	100GEGN	02130	100HEFT
01040	100FLATTER	01590	100GEIFER	02140	100HEG
01050	100FLEH	01600	100GEIS*EL	02150	100HENL
01060	100FLICK	01610	100GEIZ	02160	100HEIL
01070	100FLO*H	01620	100GELL	02170	100HEISCH
01080	100FLU*CHT	01630	1*00GENEM*IG	02180	100HEIZ
01090	100FISCH	01640	100GERS	02190	100HELL
01100	100FLUSS	01650	1*00GESELL	02200	100HELLIG
01110	100FLUT	01660	1*00GESTALT	02210	100HERN
01120	100FOLG	01670	1*00GESTATT	02220	100HERK
01130	100FORDER	01680	1*00GEUA*HRLEIST	02230	100HERBERSG

02240	100HERRECH	02790	100KLATSCH	03340	100LACK
02250	100HERZ	02800	100KLEB	03350	100LAGER
02260	100HETZ	02810	100KLEID	03360	100LA*HM
02270	100HEUCHEL	02820	100KLEISTER	03370	100LAND
02280	100HINDER	02830	100KLENN	03380	100LA*RM
02290	100HINK	02840	100KLETTER	03390	100LAST
02300	100HITZ	02850	100KLIMPER	03400	100LA*STER
02310	100HOBEL	02860	100KLINGEL	03410	100LAUB
02320	100HOCK	02870	100KLIRR	03420	100LATSCH
02330	100HOFF	02880	100KLOPF	03430	100LAUER
02340	100HO*H	02890	100KNACK	03440	100LAUSCH
02350	100HO*HL	02900	100KNALL	03450	100LAUT
02360	100HOL	02910	100KNET	03460	100LA*UTER
02370	100HOLZ	02920	100KNICK	03470	100LA*UT
02380	100HORCH	02930	100KNI	03480	100LEB
02390	100HO*H	02940	100KNIPS	03490	100LECK
02400	100HORST	02950	100KNO*PF	03500	100LEDER
02410	100HORT	02960	100KNOT	03510	100LEG
02420	100HULDIG	02970	100KNU*PF	03520	100LEHN
02430	100HU*LL	02980	100KOCH	03530	100LEHR
02440	100HUMPEL	02990	100KOHL	03540	100LEIB
02450	100HUNGER	03000	100KOPPEL	03550	100LEICHTER
02460	100HU*PF	03010	100KORK	03560	100LEIER
02470	100HUSCH	03020	100KO*RN	03570	100LEIM
02480	100HUST	03030	100KOST	03580	100LEIST
02490	100HU*T	03040	100KRACH	03590	100LEIT
02500	100INNER	03050	100KRAKEL	03600	100LENK
02510	100IRR	03060	100KRALL	03610	100LERN
02520	100JAG	03070	100KRAM	03620	100LEUCHT
02530	100JANNER	03080	100KRAMPF	03630	100LIEB
02540	100JAUL	03090	100KRANK	03640	100LIEFER
02550	100KABEL	03100	100KRATZ	03650	100LIST
02560	100KALK	03110	100KREMPPEL	03660	100LOB
02570	100KA*LT	03120	100KREPP	03670	100LOCH
02580	100KA*MPF	03130	100KREUZ	03680	100LOCK
02590	100KAPSEL	03140	100KREUZIG	03690	100LOHN
02600	100KAU	03150	100KRIBBEL	03700	100LO*SCH
02610	100KAUF	03160	100KRIES	03710	100LOS
02620	100KEHL	03170	100KRO*H	03720	100LO*G
02630	100KEHR	03180	100KRO*PF	03730	100NACH
02640	100KEIL	03190	100KRU*HEL	03740	100NA*H
02650	100KEIM	03200	100KRUM	03750	100NA*CHTIG
02660	100KERB	03210	100KRU*HM	03760	100NAHL
02670	100KETT	03220	100KUGEL	03770	100NAHN
02680	100KEUCH	03230	100KU*HL	03780	100NA*HR
02690	100KIPPEL	03240	100KU*HNER	03790	100NANGEL
02700	100KIPP	03250	100KUND	03800	100NARK
02710	100KITT	03260	100KU*NDIG	03810	100NASS
02720	100KLAFF	03270	100KU*ND	03820	100NAG*
02730	100KLAG	03280	100KUPFER	03830	100NA*G*IG
02740	100KLAMMER	03290	100KUPPEL	03840	100NATT
02750	100KLAPP	03300	100KUPPEL	03850	100NAUER
02760	100KLAPPER	03310	100KURBEL	03860	100NEHR
02770	100KLAPS	03320	100KU*RZ	03870	100NEIND
02780	100KLA*R	03330	100KLACH	03880	100NEISTER

03890	100HELD	04440	100HUE	04790	100SAFTIG
03900	100HENG	04450	100QUETSCH	05000	100SARREL
03910	100HERK	04460	100QUICK	05010	100SANG
03920	100MESSER	04470	100RANDE	05020	100SATTEL
03930	100HIET	04480	100RAN	05030	100SATUG
03940	100HILDER	04490	100RAN	05040	100SAUS
03950	100HISCH	04500	100RAN	05050	100SA*UN
03960	100MITTEL	04510	100RANDE	05060	100SCHAD
03970	100NIX	04520	100RAS	05070	100SCHADIG
03980	100RU*H	04530	100RASSEL	05080	100SCHALT
03990	100MUSTER	04540	100RAST	05090	100SCHALFTIG
04000	100NACHLA*SSIG	04550	100RAUS	05100	100SCHAL
04010	100NACHT	04560	100RAUCH	05110	100SCHAL*RF
04020	100NADEL	04570	100RA*UCHER	05120	100SCHARR
04030	100NAGEL	04580	100RA*UN	05130	100SCHASS
04040	100NA*H	04590	100RAUSCH	05140	100SCHATT
04050	100NA*HR	04600	100RECHN	05150	100SCHATZ
04060	100NA*SS	04610	100RECK	05160	100SCHAUKELE
04070	100NEBEL	04620	100RED	05170	100SCHAUDE
04080	100NEID	04630	100RESEL	05180	100SCHAU*LICHT
04090	100NEIG	04640	100REG	05190	100SCHAU
04100	100NERV	04650	100REIN	05200	100SCHAU*UN
04110	100NETZ	04660	100REICHER	05210	100SCHELL
04120	100NEUER	04670	100REICH	05220	100SCHENK
04130	100NICK	04680	100REINIG	05230	100SCHEITER
04140	100NORM	04690	100REIS	05240	100SCHEU
04150	100NO*DIG	04700	100REIZ	05250	100SCHICHT
04160	100NOT	04710	100RETT	05260	100SCHICK
04170	100NUTZ	04720	100REU	05270	100SCHIFF
04180	100NU*TZ	04730	100RICHT	05280	100SCHILDER
04190	100O*D	04740	100RIEGEL	05290	100SCHIMMER
04200	100O*FFN	04750	100RIESEL	05300	100SCHIMPF
04210	100OPFER	04760	100RIND	05310	100SCHIRM
04220	100O*RTER	04770	100RINGER	05320	100SCHIRR
04230	100ORDER	04780	100RICH	05330	100SCHLACHT
04240	100ORDN	04790	100RICH*HR	05340	100SCHLEIN
04250	100PACK	04800	100RILL	05350	100SCHLECHTER
04260	100PARK	04810	100ROLLER	05360	100SCHLEIER
04270	100PRELL	04820	100ROST	05370	100SCHLENDER
04280	100PRESS	04830	100RO*ST	05380	100SCHLEUNIG
04290	100PASS	04840	100RO*ST	05390	100SCHLEUS
04300	100PACH	04850	100RU*CK	05400	100SCHLICHT
04310	100PLAN	04860	100RUDER	05410	100SCHLU*PF
04320	100PROB	04870	100RUH	05420	100SCHNECK
04330	100PRU*F	04880	100RU*HR	05430	100SCHNEICHEL
04340	100PO*HN	04890	100RU*HR	05440	100SCHNA*LER
04350	100PRU*GEL	04900	100RUMPEL	05450	100SCHREIZ
04360	100PUDER	04910	100RUND	05460	100SCHRIEG
04370	100PULS	04920	100RUFF	05470	100SCHRIE
04380	100PULVER	04930	100RU*ST	05480	100SCHRIEGEL
04390	100PUNK	04940	100RU*STIEL	05490	100SCHNITZ
04400	100PUNKT	04950	100RACH	05500	100SCHNALL
04410	100PUTZ	04960	100RACK	05510	100SCHNAPP
04420	100QUA*L	04970	100RAG	05520	100SCHNARR
04430	100QUATSCH	04980	100RAG	05530	100SCHNAUF

05540	100SCHNAUZ	06090	100SPANN	06640	100STRECK
05550	100SCHNEI	06100	100SPAR	06650	100STREICHEN
05560	100SCHNITZEL	06110	100SPEICH	06660	100STREIK
05570	100SCHNITZ	06120	100SPEICHER	06670	100STREU
05580	100SCHNUPFEL	06130	100SPEIS	06680	100STRICK
05590	100SCHNUPFR	06140	100SPEND	06690	100STRICHEL
05600	100SCHNUR	06150	100SPERR	06700	100STROM
05610	100SCHON	06160	100SPIEGEL	06710	100STU*CK
05620	100SCHOPF	06170	100SPIEL	06720	100STU*CKEL
05630	100SCHRA*NK	06180	100SPIES*	06730	100STUF
05640	100SCHRAUB	06190	100SPITZ	06740	100STU*RN
05650	100SCHROTT	06200	100SPLITTER	06750	100SCHU*RN
05660	100SCHULD	06210	100SPOTT	06760	100STUTZ
05670	100SCHUL	06220	100SPREITZ	06770	100STU*TZ
05680	100SCHU*TTTEL	06230	100SPRENG	06780	100SUCH
05690	100SCHU*TT	06240	100SPRITZ	06790	100SUNPF
05700	100SCHU*TTER	06250	100SPROSS	06800	100SU*NDIG
05710	100SCHU*TZ	06260	100SPRUCH	06810	100TADEL
05720	100SCHUA*CH	06270	100SPRUDEL	06820	100TAFEL
05730	100SCHWANCK	06280	100SPUCK	06830	100TAG
05740	100SCHUA*RN	06290	100SPUL	06840	100TANK
05750	100SCHUA*RNZ	06300	100SPUL	06850	100TANZ
05760	100SCHWATZ	06310	100SPUR	06860	100TAPP
05770	100SCHUA*TZ	06320	100SPU*R	06870	100TARN
05780	100SCHWEB	06330	100STAB	06880	100TAST
05790	100SCHWEFEL	06340	100STACHEL	06890	100TAUCH
05800	100SCHWEIS*	06350	100STAMPF	06900	100TAUF
05810	100SCHWEMM	06360	100STAMM	06910	100TAUG
05820	100SCHWENK	06370	100STAPEL	06920	100TAUNEL
05830	100SCHWINDEL	06380	100STAPP	06930	100TAUSCH
05840	100SCHWITZ	06390	100STA*RNK	06940	100TAU
05850	100SEBEL	06400	100STARR	06950	100TAX
05860	100SEGN	06410	100START	06960	100TEILIG
05870	100SEHN	06420	100STA*TIG	06970	100TEIL
05880	100SEIF	06430	100STAUB	06980	100TEST
05890	100SEIL	06440	100STA*UB	06990	100TEUER
05900	100SA*TIG	06450	100STAUN	07000	100TEUFEL
05910	100SENG	06460	100STAU	07010	100TIPP
05920	100SENK	06470	100STEIGER	07020	100TEILIG
05930	100SETZ	06480	100STELL	07030	100TO*RN
05940	100SEUCH	06490	100STENPEL	07040	100TOPF
05950	100SEUFZ	06500	100STEUER	07050	100TO*RT
05960	100SICHTIG	06510	100STIECK	07060	100TRACHT
05970	100SICHT	06520	100STIEFEL	07070	100TRANK
05980	100SIECH	06530	100STIMM	07080	100TRAU
05990	100SIEDEL	06540	100STOLPER	07090	100TRAUER
06000	100SIEGEL	06550	100STOPF	07100	100TRAUM
06010	100SINTER	06560	100STOPF	07110	100TRENN
06020	100SIEG	06570	100STO*RN	07120	100TRICK
06030	100SOHL	06580	100STOTTER	07130	100TRAG
06040	100SOLL	06590	100STRAFF	07140	100TRAG*PFEL
06050	100SONN	06600	100STREIF	07150	100TROPF
06060	100SORG	06610	100STRAHL	07160	100TRON*ST
06070	100SPAH	06620	100STRA*NG	07170	100TROTZ
06080	100SPALT	06630	100STREB	07180	100TU*RN

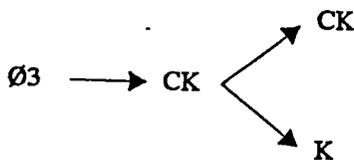
07190	100TURH	07740	100WIRTSCHAFT
07200	100U*8	07750	100WISCH
07210	1*0UNTERSUCH	07760	100WITTER
07220	100URSACH	07770	100WOHN
07230	1*0VERBILIG	07780	100WOLB
07240	1*0VERGRO*SER	07790	100WOLK
07250	1*0VEREINIG	07800	100WU*HL
07260	1*0VERLANG	07810	100WUNDER
07270	1*0VERMINDER	07820	100WU*NSCH
07280	1*0VERRINGER	07830	100WU*RFEL
07290	100VO*LKER	07840	100WU*RG
07300	100WACKEL	07850	100WURZEL
07310	100WAFFN	07860	100WU*RZ
07320	100WAG	07870	100WU*ST
07330	100WA*HL	07880	100WU*T
07340	100WAHR	07890	100ZAHL
07350	100WA*HN	07900	100ZA*HL
07360	100WA*HR	07910	100ZA*F
07370	100WALT	07920	100ZAUSER
07380	100WALZ	07930	100ZA*UN
07390	100WA*LZ	07940	100ZEICHN
07400	100WANDEL	07950	100ZEIG
07410	100WANDER	07960	100ZICHT
07420	100WA*RH	07970	100ZERR
07430	100WARN	07980	100ZEUG
07440	100WART	07990	100ZIEL
07450	100WASSER	08000	100ZIEN
07460	100WA*SSER	08010	100ZIFFER
07470	100WECHSEL	08020	100ZINS
07480	100WECK	08030	100ZITTER
07490	100WEH	08040	100ZO*GER
07500	100WEIGER	08050	100ZU*CHT
07510	100WEIH	08060	100ZU*CHTIG
07520	100WEIL	08070	100ZUCK
07530	100WEIN	08080	100ZU*CK
07540	100WEIS*	08090	100ZUG
07550	100WELK	08100	100ZU*ND
07560	100WELL	08110	100ZUA*NG
07570	100WEND	08120	100ZWECK
07580	100WERK	08130	100ZWEIFEL
07590	100WERT	08140	100ZWEIG
07600	100WETT	08150	100ZWICK
07610	100WETTER		
07620	100WICKEL		
07630	100WIDH		
07640	100WIEBEL		
07650	100WILLIG		
07660	100WIMMEL		
07670	100WINNER		
07680	100WINKEL		
07690	100WINK		
07700	100WINTER		
07710	100WIRBEL		
07720	100WIRR		
07730	100WIRK		

La morphologie des verbes irréguliers complique la reconnaissance des temps et le calcul de la base nécessaire à la consultation du dictionnaire de transfert. Le premier tableau de 186 lignes permet de générer les formes. Si les données semblent très concentrées, le mécanisme de génération engendre des formes inexistantes et ne permet pas de distinguer les temps. Il a été abandonné depuis longtemps ; son seul mérite est de souligner les vertus de MORPHAL(VERBIR), plus long puisque de 648 lignes, mais fonctionnant beaucoup plus rapidement, sans erreur et avec précision. Chaque élément est précédé d'un nombre de deux chiffres indiquant sa longueur (pour la lecture du fichier).

La ligne 100 se lit :

04 B<--ND 01I 01A 02A\* 01U 00

La forme B<--ND (composée de 4 caractères) est considérée comme un archi-segment, à partir duquel seront générées les formes BIND, BAND, BA\*ND et BUND. Les modifications consonantiques sont accolées à l'archi-segment sous la forme d'un nombre de deux chiffres précédé du signe "@".



10@03SCHR<--CK 01E 01A 02A\* 01O 01I 00

On obtient : SCHRECK, SCHRACK, SCHRA\*CK, SCHROCK, SCHRICK, SCHREK, SCHRAK, SCHRA\*K, SCHROK, SCHRIK

00010 070930+CK01A02A*01U02U*00	00270 07ER0L+CH02E101I00
00020 100100EFL+S*02E101I00	00280 06000+0001E01I01A02A*020E00
00030 060EG+HNF:101A020*01000	00290 04F+HL01E02IE01A020*0:000
00040 050EU+S01E01O020*00	00300 04F+HR01A02A*01U020*00
00050 070100+S*02E101I00	00310 07004F+LL01A02A*02IE00
00060 040+R001E01I01A020*01000	00320 04F+NG01A02A*0:100
00070 050R+RST01E01E01A02A*01000	00330 05F+CHT01E01I01O020*00
00080 030+S02IE01O020*00	00340 04F+ND01I01A02A*0:100
00090 030+T02IE01O020*00	00350 06FL+CHT01E01I0:0020*00
00100 040+ND01I01A02A*01U00	00360 04FL+S02IE01O020*00
00110 070110+T01I01A02A*0:100	00370 04FL+H02IE01O020*00
00120 040L+S01A02A*02IE00	00380 00010FL+S*02IE01O020*00
00130 040L+S02IE102IE00	00390 00027FR+0002:00101A02A*00
00140 050L+CH02E101I00	00400 04FR+R02IE01O020*00
00150 040R+T01A02A*02IE00	00410 030+R02A*01O020*00
00160 050R+CH01E01I01A02A*0:100	00420 050EB-R02A*02IE01A01000
00170 050R+HNF1E01A00	00430 030+S01E01I0:002A*00
00180 00001BR+NG01I01A02A*00	00440 050E0+H02IE02IE00
00190 07001D+HK01E01A02A*00	00450 060000+H01E01I01A00
00200 040+NG01I01A020*01U00	00460 060EL+NG01I01A02A*01U00
00210 040+R001E01I01A020*01000	00470 040+LT01E01I01A02A*01000
00220 060R+SCH01E01I01A020*01000	00480 050EN+S01E01A02A*00
00230 050R+NG01I01A02A*01U00	00490 0901:0EN+S*02IE01O020*00
00240 00010DR+S*02IE01O020*00	00500 070E0CH+NF:002IE01A02A*00
00250 07001D+NK020*02EU00	00510 060E0+HNF:101A02A*0:000
00260 040+RF020*0:001U00	00520 070000+0001E01I01A02A*00

00530 0721JG+S+02IE910020+00  
 00535 059L+CH02E101100  
 00550 080100L+S+02E101100  
 00560 009110L+T102E101100  
 00570 050L+RNF11010020+00  
 00580 040R+001A02A+010020+00  
 00590 070020R+F02E101100  
 00600 02H+02A002A1003A+T00  
 00610 04H+L01A02A+02IE00  
 00620 04H+0002A+01101A00  
 00630 02H+02A003IE000  
 00640 03H+001E010020+010020+00  
 00650 04H+0+02E102IE00  
 00660 04H+L01E01101A020+01000  
 00670 03K+S02IE010020+00  
 00680 04K+NN020+01A01000  
 00690 05KL+RNF11010020+00  
 00700 05KL+NG01101A02A+01U00  
 00710 070020K+F02E101100  
 00720 070020K+RNF1001A02A+00  
 00730 04K+NN01E01A00  
 00740 05KR+CH02IE010020+00  
 00750 03L+001A010020+02A+00  
 00760 07009L+S001A02A+02IE00  
 00770 03L+F02A003A+U02IE00  
 00780 06012L+002IE01100  
 00790 03L+R02E102IE00  
 00800 03L+R02IE010020+00  
 00810 03L+S01E02IE01A02A+00  
 00820 03L+G02IE01A02A+01E00  
 00830 03L+G020+010020+00  
 00840 05L+SCH020+01101000  
 00850 03H+002E102IE00  
 00860 04H+LK01E020+01000  
 00870 07009H+S001E01101A02A+00  
 00880 06001H+G020+01A01000  
 00890 07009H+S0020+01U00  
 00900 07003N+RNF1E01101A02A+01000  
 00910 04H+RNF1E01A00  
 00920 070020F+F02E101100  
 00930 05FFL+S01E010020+00  
 00940 04FR+S02E102IE00  
 00950 050U+LL01E011010020+00  
 00960 03R+T01A02A+02IE00  
 00970 03R+002E102IE00  
 00980 07010R+S+02E101100  
 00990 07011R+T02E101100  
 01000 04R+RNF1E01A00  
 01010 04R+CH02IE010020+00  
 01020 04R+RNF1101A02A+01U00  
 01030 04R+RNF1101A02A+01000  
 01040 03R+F001U020E00  
 01050 06002S+F02A003A+U010020+00  
 01060 03S+G02A0010020+00  
 01070 09010SCH+FF01A01U020+00  
 01080 06SCH+LL01A010020+00  
 01090 05SCH+002E102IE00  
 01100 05SCH+R02E102IE00  
 01110 06SCH+L01E01101A020+01000  
 01120 05SCH+R01E02IE010020+00  
 01130 05SCH+002E10020+00  
 01140 09010SCH+S+02IE010020+00  
 01150 06SCH+R001E01U020+00  
 01160 06SCHL+F01A02A+02IE00  
 01170 09002SCHL+F02E101100  
 01180 06SCHL+001A02A+01U020+00  
 01190 07SCHL+CH02E101100

01200 09002SCHL+F02E101100  
 01210 10010SCHL+S+02E101100  
 01220 10010SCHL+S+02E1010020+00  
 01230 07SCHL+RNF1101A02A+01U00  
 01240 10010SCHL+S+02E101100  
 01250 07SCHL+L01E011010020+00  
 01260 06SCHL+002A0010020+00  
 01270 09012SCHL+002E101100  
 01280 10003SCHL+CK01E01A02A+01001100  
 01290 06SCHL+002E102IE00  
 01300 10011SCHL+TT02E102IT00  
 01310 05SCHL+02E102IE00  
 01320 06SCHL+002E102IE00  
 01330 07SCHL+LL01E011010020+00  
 01340 07SCHL+RNF1101A02A+010020+00  
 01350 07SCHL+RNF1101A02A+01U00  
 01360 07SCHL+RNF1101A02A+01000  
 01370 06SCHL+R020+010020+01U00  
 01380 03S+R01E02IE01A02A+00  
 01390 04S+R01E01A00  
 01400 06012S+002IE010020+00  
 01410 04S+R01101A02A+01U00  
 01420 04S+R01101A02A+01U00  
 01430 04S+R01101A02A+0020+01000  
 01440 07009S+T01101A02A+01E00  
 01450 03SF+02E102IE00  
 01460 05SF+RNF1101A02A+020+01000  
 01470 06SF+RNF1101A02A+01000  
 01480 09010SFR+S+02E10020+00  
 01490 06SFR+RNF1101A02A+01U00  
 01500 05ST+CH01E01101A02A+01000  
 01510 09003ST+CK01E01A02A+00  
 01520 07007ST+001E01A02A+020+00  
 01530 05ST+HL01E02IE01A020+01000  
 01540 04ST+S02E102IE00  
 01550 05ST+RNF1101A020+01000  
 01560 04ST+002IE010020+00  
 01570 05ST+RNF1101A02A+01U00  
 01580 05ST+S+010020+02IE00  
 01590 05ST+RNF1101A02A+01000  
 01600 09011STR+TT02E101100  
 01610 04TR+001A02A+01U020+00  
 01620 09013TR+FF01E01101A02A+01000  
 01630 04TR+002E102IE00  
 01640 08011TR+TT01E01101A02A+00  
 01650 07002TR+F02IE010020+00  
 01660 05TR+RNF1101A02A+01U00  
 01670 04TR+0020+010020+00  
 01680 02T+01U01A02A+00  
 01690 05U+CH001A02A+01U020+00  
 01700 03U+002A+010020+00  
 01710 05U+SCH01A02A+01U020+00  
 01720 03U+R01A02A+00  
 01730 03U+001E010020+00  
 01740 04U+CH02E101100  
 01750 03U+S02E102IE00  
 01760 04U+RNF1E01A00  
 01770 04U+R001E01101A020+01000  
 01780 04U+R001E01101U020+01000  
 01790 04U+RF01E01101A020+01000  
 01800 03U+002IE010020+00  
 01810 04U+R001E01101A02A+01U00  
 01820 07009U+S001102E101U020+00  
 01830 04U+LL01001100  
 01840 05Z+H02E102IE00  
 01850 05014Z+H02IE010020+00  
 01860 05ZU+RNF1101A02A+01U00  
 END OF DATA

Le code utilisé dans .MORPHAL(VERBIR) a quatre chiffres.

Position 1 : type de verbe	2	verbe irrégulier
	3	auxiliaire
	4	prétérito-présent
	7	mixte
Position 2,3 et 4 : Conjugaison	1	infinitif/présent indicatif
	2	prétérit
	3	participe passé
	4	prétérit/participe passé
	6	présent indicatif
	7	subjonctif
	8	participe passé ambigu (GE)FROREN

Après le tiret, le radical de l'infinitif correspondant.

.MORPHAL(VERBIR) :

00540	26000BIRG-BERG	00870	22000BRACH-BRECH
00550	22000BARG-BERG	00880	74000BRACH-BRING
00560	27000BU*RG-BERG	00890	27000BRA*CH-BRECH
00570	27000BA*RG-BERG	00900	77000BRA*CH-BRING
00580	23000BORG-BERG	00910	21000BRECH-BRECH
00590	21000BERST-BERST	00920	71000BRING-BRING
00600	26000BIRST-BERST	00930	23000BROCH-BRECH
00610	22000BARST-BERST	00940	26000BRICH-BRECH
00620	23000BORST-BERST	00950	21000BRENN-BRENN
00630	27000BA*RST-BERST	00960	74000BRANN-BRENN
00640	27000BO*RST-BERST	00970	71000DENK-DENK
00650	21000BIEG-BIEG	00980	74000DACH-DENK
00660	24000B0G-BIEG	00990	77000DA*CH-DENK
00670	27000B0*G-BIEG	01000	21000DRESCH-DRESCH
00680	21000BIET-BIET	01010	26000DRISCH-DRESCH
00690	24000BOT-BIET	01020	24000DROSCH-DRESCH
00700	27000B0*T-BIET	01030	27000DRO*SCH-DRESCH
00710	21000BIND-BIND	01040	21000DRING-DRING
00720	22000BAND-BIND	01050	22000DRANG-DRING
00730	27000BA*ND-BIND	01060	27000DRA*NG-DRING
00740	23000BUND-BIND	01070	23000DRUNG-DRING
00750	21000BITT-BITT	01080	21000EMPFEHL-EMPFEHL
00760	22000BAT-BITT	01090	26000ENFFIEHL-EMPFEHL
00770	27000BA*T-BITT	01100	22000EMPFAHL-EMPFEHL
00780	23000BET-BITT	01110	27000EMPFO*HL-EMPFEHL
00790	21300BLAS-BLAS	01120	23000ENPFOHL-EMPFEHL
00800	26000BLA*S-BLAS	01130	21000ERBLEICH-ERBLEICH
00810	22000BLIES-BLAS	01140	24000ERBLICH-ERBLEICH
00820	21000BLEIB-BLEIB	01150	21000ERKIES-ERKIES
00830	24000BLIEB-BLEIB	01160	24000ERKOR-ERKIES
00840	21300BRAT-BRAT	01170	27000ERKOR*ERKIES
00850	26000BRA*T-BRAT	01180	21000ERLO*SCH-ERLO*SCH
00860	22000BRIET-BRAT	01190	26000ERLISCH-ERLO*SCH

01200	2400ERLOSCH-ERL0SCH	01860	2400GESTAND-(GE)STEH
01210	2100ESS-ESS	01870	2130GEFALL-(GE)FALL
01220	2600IS*-ESS	01880	2200GEFIEL-GEFALL
01230	2200IAS*-ESS	01890	2120GERAT-(GE)RAT
01240	2700IAS*-ESS	01900	2200GERIET-GERAT
01250	2300GESS-ESS	01910	2130GEB-GEB
01260	2130FAHR-FAHR	01920	2600GIB-GEIB
01270	2600FA*HR-FAHR	01930	2200GAR-GEAR
01280	2200FUHR-FAHR	01940	2700GA*B-GEAB
01290	2700FU*HR-FAHR	01950	2100GEDEIH-GEDEIH
01300	2130FALL-FALL	01960	2400GEDIH-GEDEIH
01310	2600FALL-LL-FALL	01970	2100GEH-GEH
01320	2200FIEL-FALL	01980	2200GIEH-GEH
01330	2130FANG-FANG	01990	2300GANG-GEH
01340	2700FANG-FANG	02000	2100GELING-GELING
01350	2200FING-FANG	02010	2200GELANG-GELING
01360	2100FECHT-FECHT	02020	2700GELANG-GELING
01370	2600FICHT-FECHT	02030	2300GELUNG-GELING
01380	2400FOCHT-FECHT	02040	2100GELT-GELT
01390	2700FO*CHT-FECHT	02050	2600GILT-GELT
01400	2100FIND-FIND	02060	2200GALT-GELT
01410	2200FAND-FIND	02070	2700GALT-GELT
01420	2300FUND-FIND	02080	2700GALT-GELT
01430	2700FAND-FIND	02090	2300GOLT-GELT
01440	2100FLECHT-FLECHT	02100	2130GENES-GENES
01450	2600FLICHT-FLECHT	02110	2200GENAS-GENES
01460	2400FLOCHT-FLECHT	02120	2700GENAS-GENES
01470	2700FLO*CHT-FLECHT	02130	2100GENIES*-GENIES*
01480	2100FLIEG-FLIEG	02140	2200GENOS*-GENIES*
01490	2400FLOG-FLIEG	02150	2300GENOSS-GENIES*
01500	2700FLOG-FLIEG	02160	2700GEND*SS-GENIES*
01510	2100FLIEH-FLIEH	02170	2130GESCHEH-GESCHEH
01520	2400FLOH-FLIEH	02180	2600GESCHIEH-GESCHEH
01530	2700FLOH-FLIEH	02190	2200GESCHAH-GESCHEH
01540	2100FLIES*-FLIES*	02200	2700GESCHA*H-GESCHEH
01550	2200FLOS*-FLIES*	02210	2100GEWINN-GEWINN
01560	2300FLOSS-FLIES*	02220	2200GEWANN-GEWINN
01570	2700FLOSS-FLIES*	02230	2700GEWANN-GEWINN
01580	2130FRESS-FRESS	02240	2700GEWA*NN-GEWINN
01590	2600FRIS*-FRESS	02250	2300GEWONN-GEWINN
01600	2200FRAS*-FRESS	02260	2100GIES*-GIES*
01610	2700FRAS*-FRESS	02270	2200GOS*-GIES*
01620	2100FRIER-FRIER	02280	2300GOSS-GIES*
01630	2400FROR-FRIER	02290	2700GOS*-GIES*
01640	2700FROR-FRIER	02300	2100GLEICH-GLEICH
01650	2100GAR-GAR	02310	2400GLICH-GLEICH
01660	2400GAR-GAR	02320	2100GLEIT-GLEIT
01670	2700GAR-GAR	02330	2400GLITT-GLEIT
01680	2100GEBAR-GEBAR	02340	2100GLIHN-GLIHN
01690	2600GEBIER-GEBAR	02350	2400GLIHN-GLIHN
01700	2200GEBAR-GEBAR	02360	2700GLIHN-GLIHN
01710	2300GEBOR-GEBAR	02370	2130GRAB-GRAB
01720	2700GEBOR-GEBAR	02380	2700GRAB-GRAB
01730	2100GEBRECH-GEBRECH	02390	2200GRUB-GRAB
01740	2600GEBRICH-GEBRECH	02400	2700GRUB-GRAB
01750	2400GEBROCH-(GE)BRECH	02410	2100GREIF-GREIF
01760	2100GEDENK-(GE)DENK	02420	2400GRIFF-GREIF
01770	2400GEDACH-(GE)DENK	02430	2130HALT-HALT
01780	2100GERINN-GERINN	02440	2600HALT-HALT
01790	2200GERAHN-GERINN	02450	2200HIELT-HALT
01800	2300GEROHN-(GE)RINN	02460	2100HA*NG-HA*NG
01810	2100GEBIET-GEBIET	02470	2300HANG-HA*NG
01820	2400GEBOT-(GE)BIET	02480	2200HING-HA*NG
01830	2100GEFRIER-GEFRIER	02490	2130HAU-HAU
01840	2400GEFROR-(GE)FRIER	02500	2200HIEB-HAU
01850	2100GESTEH-GESTEH	02510	2100HEB-HEB

02520	2400HOB-HEB	03100	2600NIEH-NEHH
02530	2700HO*H-HEB	03190	2200NAHH-NEHH
02540	2130HEIS*-HEIS*	03200	2700NA*HN-NEHH
02550	2200HTES*-HEIS*	03210	2300NONH-NEHH
02560	2100HELF-HELF	03220	7100NENN-NENN
02570	2600HILF-HELF	03230	7400NANN-NENN
02580	2200HALF-HELF	03240	2100PFEIF-PFEIF
02590	2700HU*LF-HELF	03250	2400PFIFF-PFEIF
02600	2300HOLF-HELF	03260	2100PFLEG-PFLEG
02610	2100KENN-KENN	03270	2400PFLOG-PFLEG
02620	7400KANN-KENN	03280	2700PFLO*G-PFLEG
02630	2100KLINN-KLINN	03290	2100PREIS-PREIS
02640	2400KLOMM-KLINN	03300	2400PRIES-PREIS
02650	2700KLO*HN-KLINN	03310	2100QUELL-QUELL
02660	2100KLING-KLING	03320	2600QUILL-QUELL
02670	2200KLANG-KLING	03330	2400QUOLL-QUELL
02680	2700KLA*NG-KLING	03340	2700QUO*LL-QUELL
02690	2300KLUNG-KLING	03350	2100RAT-RAT
02700	2100KNEIF-KNEIF	03360	2600RAT-RAT
02710	2400KNIFF-KNEIF	03370	2200RIET-RAT
02720	2130KOMM-KOMM	03380	2100REIB-REIB
02730	2200KAM-KOMM	03390	2400RIEB-REIB
02740	2700KA*H-KOMM	03400	2100REIS*-REIS*
02750	2100KRIECH-KRIECH	03410	2200RIS*-REIS*
02760	2400KROCH-KRIECH	03420	2300RISS-REIS*
02770	2700KRO*CH-KRIECH	03430	2100REIT-REIT
02780	2130LAD-LAD	03440	2400RITT-REIT
02790	2600LAD-LAD	03450	7100RENN-RENN
02800	2200LUD-LAD	03460	7400RANN-RENN
02810	2700LU*AD-LAD	03470	2100RIECH-RIECH
02820	2100LASS-LASS	03480	2400ROCH-RIECH
02830	2600LASS*-LASS	03490	2700RO*CH-RIECH
02840	2200LIES*-LASS	03500	2100RING-RING
02850	2200LAS*-LASS	03510	2200RANG-RING
02860	2130LAUF-LAUF	03520	2700RANG-RING
02870	2600LA*UF-LAUF	03530	2300RUNG-RING
02880	2200LIEF-LAUF	03540	2100RINN-RINN
02890	2100LEID-LEID	03550	2200RANN-RINN
02900	2400LITT-LEID	03560	2700RANN-RINN
02910	2100LEIH-LEIH	03570	2700RO*NN-RINN
02920	2400LIEN-LEIH	03580	2300RONN-RINN
02930	2130LES-LES	03590	2130RUF-RUF
02940	2600LIES-LES	03600	2200RIEF-RUF
02950	2200LAS-LES	03610	2100SAUF-SAUF
02960	2700LASS-LES	03620	2600SA*UF-SAUF
02970	2100LIEG-LIEG	03630	2400SOFF-SAUF
02980	2200LAG-LIEG	03640	2700SO*FF-SAUF
02990	2700LAG-LIEG	03650	2100SAUG-SAUG
03000	2300LEG-LIEG	03660	2400SOG-SAUG
03010	2100LUG-LUG	03670	2700SOG-SAUG
03020	2400LOG-LUG	03680	2130SCHAFF-SCHAFF
03030	2700LUG-LUG	03690	2200SCHUF-SCHAFF
03040	2100NEID-NEID	03700	2700SCHUF-SCHAFF
03050	2400NIED-NEID	03710	2100SCHALL-SCHALL
03060	2100MELK-MELK	03720	2200SCHOLL-SCHALL
03070	2600MILK-MELK	03730	2700SCHO*LL-SCHALL
03080	2400MOLK-MELK	03740	2100SCHEID-SCHEID
03090	2130NESS-NESS	03750	2400SCHIED-SCHEID
03100	2100NIS*LING-NIS*LING	03760	2100SCHEIN-SCHEIN
03110	2200NIS*LANG-NIS*LING	03770	2400SCHIEH-SCHEIN
03120	2700NIS*LANG-NIS*LING	03780	2100SCHEIS*-SCHEIS*
03130	2300NIS*LUNG-NIS*LING	03790	2200SCHIS*-SCHEIS*
03140	2600NIS*-NESS	03800	2300SCHISS-SCHEIS*
03150	2200NAS*-NESS	03810	2100SCHELT-SCHELT
03160	2700NAS*-NESS	03820	2600SCHILT-SCHELT
03170	2100NEHM-NEHM	03830	2200SCHALT-SCHELT

03840	2700SCHU*LT-SCHELT	04500	2700SCHW*LL-SCHWELL
03850	2300SCHOLT-SCHELT	04510	2100SCHWinn-SCHWinn
03860	2100SCHER-SCHER	04520	2200SCHWANN-SCHWinn
03870	2300SCHIER-SCHER	04530	2700SCHW*nn-SCHWinn
03880	2400SCHOR-SCHER	04540	2300SCHWonn-SCHWinn
03890	2700SCHO*R-SCHER	04550	2700SCHW*nn-SCHWinn
03900	2100SCHIEB-SCHIEB	04560	2100SCHWIND-SCHWIND
03910	2400SCHOS-SCHIEB	04570	2200SCHWIND-SCHWIND
03920	2700SCHO*B-SCHIEB	04580	2700SCHW*ND-SCHWIND
03930	2100SCHIES*-SCHIES*	04590	2300SCHWUND-SCHWIND
03940	2200SCHOS*-SCHIES*	04600	2100SCHWING-SCHWING
03950	2300SCHOSS-SCHIES*	04610	2200SCHWANG-SCHWING
03960	2700SCHO*SS-SCHIES*	04620	2700SCHW*NG-SCHWING
03970	2100SCHIND-SCHIND	04630	2300SCHWUNG-SCHWING
03980	2400SCHUND-SCHIND	04640	2100SCHW*R-SCHW*R
03990	2700SCHU*ND-SCHIND	04650	2400SCHUR-SCHW*R
04000	2130SCHLAF-SCHLAF	04660	2700SCHU*R-SCHW*R
04010	2600SCHLA*F-SCHLAF	04670	2130SEH-SEH
04020	2200SCHLIEF-SCHLAF	04680	2600SIEH-SEH
04030	2130SCHLAG-SCHLAG	04690	2200SAH-SEH
04040	2600SCHLA*G-SCHLAG	04700	2700SA*H-SEH
04050	2200SCHLUG-SCHLAG	04710	7100SEND-SEND
04060	2700SCHLU*G-SCHLAG	04720	7400SAND-SEND
04070	2130SCHLEICH-SCHLEICH	04730	2100SIED-SIED
04080	2200SCHLICH-SCHLEICH	04740	2400SOTI-SIED
04090	2100SCHLEIF-SCHLEIF	04750	2700SO*TT-SIED
04100	2400SCHLIFF-SCHLEIF	04760	2100SING-SING
04110	2100SCHLEIS*-SCHLEIS*	04770	2200SANG-SING
04120	2200SCHLIS*-SCHLEIS*	04780	2700SA*NG-SING
04130	2300SCHLISS-SCHLEIS*	04790	2300SUNG-SING
04140	2100SCHLIES*-SCHLIES*	04800	2100SINK-SINK
04150	2200SCHLOS*-SCHLIES*	04810	2200SANK-SINK
04160	2300SCHLOSS-SCHLIES*	04820	2700SA*NK-SINK
04170	2700SCHLO*SS-SCHLIES*	04830	2300SUNK-SINK
04180	2100SCHLING-SCHLING	04840	2100SINN-SINN
04190	2200SCHLANG-SCHLING	04850	2200SANN-SINN
04200	2700SCHLA*NG-SCHLING	04860	2700SA*NN-SINN
04210	2300SCHLUNG-SCHLING	04870	2700SO*NN-SINN
04220	2100SCHMEIS*-SCHMEIS*	04880	2300SONN-SINN
04230	2200SCHNIS*-SCHMEIS*	04890	2100SITZ-SITZ
04240	2300SCHNISS-SCHMEIS*	04900	2200SAS*-SITZ
04250	2100SCHNELZ-SCHNELZ	04910	2700SA*S*-SITZ
04260	2600SCHNILZ-SCHNELZ	04920	2300SESS-SITZ
04270	2400SCHNOLZ-SCHNELZ	04930	2100SPEI-SPEI
04280	2700SCHNO* LZ-SCHNELZ	04940	2400SPIE-SPEI
04290	2100SCHNAUB-SCHNAUB	04950	2100SPINN-SPINN
04300	2400SCHNOB-SCHNAUB	04960	2200SPANN-SPINN
04310	2700SCHNO*B-SCHNAUB	04970	2700SP*NN-SPINN
04320	2100SCHNEID-SCHNEID	04980	2700SPA*NN-SPINN
04330	2400SCHNITT-SCHNEID	04990	2300SPONN-SPINN
04340	2100SCHRECK-SCHRECK	05000	2100SPLEIS*-SPLEIS*
04350	2600SCHRICK-SCHRECK	05010	2200SP LIS*-SPLEIS*
04360	2200SCHRAK-SCHRECK	05020	2300SP LISS-SPLEIS*
04370	2700SCHRA*K-SCHRECK	05030	2100SPRECH-SPRECH
04380	2300SCHROCK-SCHRECK	05040	2600SPRICH-SPRECH
04390	2100SCHREIB-SCHREIB	05050	2200SPRACH-SPRECH
04400	2400SCHRIEB-SCHREIB	05060	2300SPROCH-SPRECH
04410	2100SCHREI-SCHREI	05070	2700SPRA*CH-SPRECH
04420	2400SCHRIE-SCHREI	05080	2100SPRIES-SPRIES*
04430	2100SCHREIT-SCHREIT	05090	2200SPROSS-SPRIES*
04440	2400SCHRIIT-SCHREIT	05100	2300SPROSS-SPRIES*
04450	2100SCHWEIG-SCHWEIG	05110	2700SPRO*SS-SPRIES*
04460	2400SCHWIEG-SCHWEIG	05120	2100SPRING-SPRING
04470	2100SCHWELL-SCHWELL	05130	2200SPRANG-SPRING
04480	2600SCHWILL-SCHWELL	05140	2700SPRA*NG-SPRING
04490	2400SCHWOLL-SCHWELL	05150	2300SPRUNG-SPRING

05160	2100STECH-STECH	05820	2700TA*-TUN
05170	2600STICH-STECH	05830	2100VERDERB-VERDERB
05180	2200STACH-STECH	05840	2600VERDIRB-VERDERB
05190	2700STA*CH-STECH	05850	2200VERBARB-VERDERB
05200	2300STOCH-STECH	05860	2700VERDU*RB-VERDERB
05210	2100STECK-STECK	05870	2300VERDORB-VERDERB
05220	2200STAK-STECK	05880	2100VERDRIES*-VERDRIES*
05230	2700STA*K-STECK	05890	2200VERDROS*-VERDRIES*
05240	2100STEN-STEH	05900	2300VERDROSS-VERDRIES*
05250	2400STAND-STEH	05910	2700VERDRO*SE-VERDRIES*
05260	2700STA*ND-STEH	05920	2130VERGESS-VERGESS
05270	2700STU*ND-STEH	05930	2600VERGIS*-VERGESS
05280	2100STEHL-STEHL	05940	2200VERGAS*-VERGESS
05290	2600STIEHL-STEHL	05950	2700VERGA*S*-VERGESS
05300	2200STAHL-STEHL	05960	2100VERLIER-VERLIER
05310	2300STOHL-STEHL	05970	2400VERLOR-VERLIER
05320	2700STO*HL-STEHL	05980	2700VERLO*R-VERLIER
05330	2700STA*HL-STEHL	05990	2100VERZEIH-VERZEIH
05340	2100STEIG-STEIG	06000	2400VERZIEH-VERZEIH
05350	2400STIEG-STEIG	06010	2130WACHS-WACHS
05360	2100STERB-STERB	06020	2600WA*CHS-WACHS
05370	2600STIRB-STERB	06030	2200WUCHS-WACHS
05380	2200STARB-STERB	06040	2700WU*CHS-WACHS
05390	2700STU*RB-STERB	06050	2100WA*G-WA*G
05400	2300STORB-STERB	06060	2400WOG-WA*G
05410	2100STIEB-STIEB	06070	2700WO*G-WA*G
05420	2400STOB-STIEB	06080	2130WASCH-WASCH
05430	2700STO*B-STIEB	06090	2600WA*SCH-WASCH
05440	2100STINK-STINK	06100	2200WUSCH-WASCH
05450	2200STANK-STINK	06110	2700WU*SCH-WASCH
05460	2700STA*NK-STINK	06120	2100WEB-WEB
05470	2300STUNK-STINK	06130	2400WOB-WEB
05480	2600STO*S*-STOS*	06140	2700WO*B-WEB
05490	2130STOS*-STOS*	06150	2100WEICH-WEICH
05500	2200STIES*-STOS*	06160	2400WICH-WEICH
05510	2100STREICH-STREICH	06170	2100WEIS-WEIS
05520	2400STRICH-STREICH	06180	2400WIES-WEIS
05530	2100STREIT-STREIT	06190	7100WEND-WEND
05540	2400STRITT-STREIT	06200	7400WAND-WEND
05550	2130TRAG-TRAG	06210	2100WERB-WERB
05560	2600TRA*G-TRAG	06220	2600WIRB-WERB
05570	2200TRUG-TRAG	06230	2200WARB-WERB
05580	2700TRU*G-TRAG	06240	2700WU*RB-WERB
05590	2100TREFF-TREFF	06250	2300WORB-WERB
05600	2600TRIFF-TREFF	06260	2100WERF-WERF
05610	2200TRAF-TREFF	06270	2600WIRF-WERF
05620	2700TRA*F-TREFF	06280	2200WARF-WERF
05630	2300TROFF-TREFF	06290	2700WU*RF-WERF
05640	2100TREIB-TREIB	06300	2300WORF-WERF
05650	2400TRIEB-TREIB	06310	2100WIEG-WIEG
05660	2130TRET-TRET	06320	2400WOG-WIEG
05670	2600TRITT-TRET	06330	2700WO*G-WIEG
05680	2200TRAT-TRET	06340	2100WIND-WIND
05690	2700TRA*T-TRET	06350	2200WAND-WIND
05700	2100TRIEF-TRIEF	06360	2700WA*ND-WIND
05710	2400TROFF-TRIEF	06370	2300WUND-WIND
05720	2700TRO*FF-TRIEF	06380	2100WISS-WISS
05730	2100TRINK-TRINK	06390	7600WEIS*-WISS
05740	2200TRANK-TRINK	06400	7400WUS*-WISS
05750	2700TRA*NK-TRINK	06410	2700WU*S*T-WISS
05760	2300TRUNK-TRINK	06420	2100ZIEH-ZIEH
05770	2100TRU*G-TRU*G	06430	2400ZOG-ZIEH
05780	2400TROG-TRU*G	06440	2700ZO*G-ZIEH
05790	2700TRO*G-TRU*G	06450	2100ZUING-ZUING
05800	2100TU-TUN	06460	2200ZUANG-ZUING
05810	2400TA-TUN	06470	2700ZUA*NG-ZUING
		06480	2300ZUUNG-ZUING
			END OF DATA

#### 4.3.2.2.3.4 Vérification du résidu et du codage

Après avoir identifié une racine verbale, .VERBAL2 isole le résidu et le compare aux éléments du fichier des désinences .MORPHAL(DEST). Cette démarche évite les écueils du type AB-SEH-BARE ou AB et SEH peuvent faire illusion jusqu'à ce que le test de BARE suspende l'analyse.

Deux obstacles se dressent encore avant le codage de la forme verbale reconnue, la nature du "ge-" et du "zu".

##### 1) "GE-" préverbe ou morphème du participe passé ?

Il existe trois cas de figure pour les verbes réguliers comme pour les verbes irréguliers.

a - Pour les verbes réguliers suivants : GEBÄRDEN, GEBAREN, GEBÜHREN, GENEHMIGEN, GENÜGEN, GESELLEN, GESTALTEN, GESTATTEN, GESUNDEN, GEWÄLTIGEN, GEWÄRTIGEN, GEWÖHNEN, GEFÄHRDEN, GELANGEN, GELÜSTEN, le "ge-" est toujours préverbe inséparable, d'où l'ambiguïté type GESTALTET : Présent indicatif ou Participe passé de GESTALTEN.

b - Pour les verbes réguliers suivants : GEBRAUCHEN, GEDULDEN, GEHORCHEN, GEHÖREN, GELEITEN, GELOBEN, GEREICHEN, GEREUEN, GERUHEN, GESEGNEN, GETRAUEN, GEWAHREN, GEWÄHREN, GEZIEMEN, GEMAHNEN, le "ge-" associé à la désinence "-t" ou "-et" peut être préverbe ou morphème du participe passé, d'où l'ambiguïté type GEBRAUCHT : Participe passé de BRAUCHEN et de GEBRAUCHEN, Indicatif présent de GEBRAUCHEN.

c - Pour les autres verbes réguliers, 'GE' est toujours morphème du participe passé.

a1 - Pour les verbes irréguliers suivants : GEBÄREN, GEDEIHEN, GELINGEN, GENESEN, GENIEßEN, GESCHEHEN, GEWINNEN, "ge-" est préverbe quelle que soit la désinence.

ambiguïté Prétérit/Participe passé (GEDEIHEN, GENIEßEN)  
ambiguïté Participe passé/Infinitif (GENESEN, GESCHEHEN)  
pas d'ambiguïté de temps pour GEBÄREN (a-o), GELINGEN (a,u) et GEWINNEN (a,o).

b1 - Pour les verbes irréguliers suivants : GEBRECHEN, GEDENKEN, GERINNEN, GEBIETEN, GEFRIEREN, GESTEHEN, GEFALLEN, GERATEN, le "ge-" associé à l'alternance vocalique du participe passé et à la désinence "-en" ("t" pour GEDENKEN) est ambigu :

- GE-préverbe/GE-participe passé pour GEBRECHEN (a-o), GEDENKEN (a-a), GEDENKEN (a,a), GERINNEN (a,o)

- Participe passé verbe préverbé/Prétérit pluriel verbe préverbé/Participe passé verbe *simple* pour GEBIETEN, GEFRIEREN, GESTEHEN

- Infinitif verbe préverbé/Indicatif présent pluriel verbe préverbé/ Participe passé verbe préverbé/Participe passé verbe préverbé/ participe passé verbe *simple* pour GEFALLEN et GERATEN

c1 - Pour tous les autres verbes irréguliers, le "ge-" est toujours associé au participe passé.



.TRANSIT3 pour le luxembourgeois :

00054 000180007	HUE T	HUE
00055 000180015	MAT BRUECH T	MATBRUECH
00056 000190008	GE NANN T	NANN
00057 000190010	HUE T	HUE
00058 000190013	ENT WE>CKEL T	ENTWE>CKEL
00059 000190015	BESCHA*FTEG T	BESCHA*FTEG
00060 000200009	AS	
00061 000200049	GE LUEG T	LUEG
00062 000200059	HUE T	HUE
00063 000210073	AS	
00064 000210086	ER OP GAANG EN	EROPGAANG
00065 000220012	GE* T	GE*
00066 000220027	FE>IER T	FE>IER
00067 000230005	HUE T	HUE
00068 000230016	ENT WE>CKEL T	ENTWE>CKEL
00069 000230020	HU N	HU
00070 000230027	FEST GE SAT	FESTSAT
00071 000240004	AS	
00072 000250015	BRUECH T	BRUECH
00073 000250016	HUE T	HUE
00074 000250018	HUE T	HUE
00075 000250027	GEFE>IER T	GEFE>IER
00076 000250035	SIN	
00077 000250041	OF HA*NK EN	OFHA*NK
00078 000250044	IMPORTE>IER EN	IMPORTE>IER
00079 000250046	EXPORTE>IER EN	EXPORTE>IER
00080 000260030	GE MAACH	MAACH
00081 000260031	GOUF	GOUF
00082 000270004	HA*NK T	HA*NK
00083 000270024	EXPORTE>IER T	EXPORTE>IER
00084 000270022	GE* T	GE*
00085 000280010	AS	
00086 000280011	GRAFF	GRAFF
00087 000290006	INTEGRE>IER TEN	INTEGRE>IER
00088 000290009	MUSS	MUSS
00089 000290015	IWWER DENK EN	IWWERDENK
00090 000290017	U PASS EN	UPASS
00091 000290037	SE>CHER EN	SE>CHER
00092 000300018	MUSS	MUSS
00093 000310015	GE LE ET	LE
00094 000310016	GI N	GI
00095 000310025	HU N	HU
00096 000320011	HU	
00097 000320015	MISS EN	MISS
00098 000320016	U PASS EN	UPASS
00099 000330004	AS	
00100 000330006	AS	
00101 000340005	SIN	
00102 000340033	GEE T	GEE
00103 000340035	FE>IER T	FE>IER
00104 000340040	SE>IER T	SE>IER
00105 000350020	HE*LL T	HE*LL
00106 000350025	GE* T	GE*
00107 000360013	HE*L T	HE*L
00108 000370009	ENT STI	ENTSTI

00109 000390006	GE* T	GE*
00110 000390055	ZE SUMMEN GEFALL	ZESUMMENGEFALL
00111 000380056	AS	
00112 000390017	WA*ER TEN	WA*ER
00113 000390019	HU N	HU
00114 000390026	FOND	FOND
00115 000400003	GESI	GESI
00116 000410019	HUE T	HUE
00117 000410022	KASCH T	KASCH
00118 000420004	HA T	HA
00119 000420011	GE SCHAFF	SCHAFF
00120 000430002	GE SOT	SOT
00121 000430007	AS	
00122 000430018	KRICH	KRICH
00123 000430020	VIR U GAANG EN	VIRUGAANG
00124 000430041	U GE FAANG	UFAANG
00125 000430042	HU N	HU
00126 000430044	SPILL EN	SPILL
00127 000430050	GE* T	GE*
00128 000430059	VER RE>CKEL T	VERRE>CKEL
00129 000430060	HUE T	HUE
00130 000440000	SIN	
00131 000440013	DOMINE>IER T	DOMINE>IER
00132 000450003	SIN	
00133 000450010	ZE SUMMEN (ZE) DIN	ZESUMMENDIN
00134 000460001	GE BAU T	BAU
00135 000460026	GI N	GI
00136 000470000	KRIS	KRIS
00137 000470013	HUE T	HUE
00138 000470021	PRODUZE>IER T	PRODUZE>IER
00139 000480004	HUE T	HUE
00140 000480011	PRODUZE>IER T	PRODUZE>IER
00141 000480014	SIN	
00142 000490003	HUE T	HUE
00143 000490010	GE MAACH	MAACH
00144 000490016	BESCHA*FTEG T	BESCHA*FTEG
00145 000500003	AS	
00146 000500007	KOMM	KOMM
00147 000500012	VIR AUS GONG	VIRAUSSGONG
00148 000510003	AS	
00149 000510014	ZERE>CK GAANG	ZERE>CKGAANG
00150 000560005	HUE T	HUE
00151 000560018	GE SCHAFF T	SCHAFF
00152 000560027	VER STOPP T	VERSTOFF
00153 000560028	GOUF EN	GOUF
00154 000570002	HUE T	HUE
00155 000570012	GE BRAUCH T	BRAUCH
00156 000580004	HUE T	HUE
00157 000580053	KANN T	KANN
00158 000580054	HA T	HA
00159 000590017	ZE SUMMEN KE>IP EN	ZESUMMENKE>IP
00160 000600006	HA TEN	HA
00161 000600012	WAR EN	WAR
00162 000610005	HA TE	HA
00163 000610007	GE BAU T	BAU
00164 000610018	ER AUS GONG EN	ERAUSSGONG

00165 000620005	WAR EN	WAR
00166 000620016	ENT STAN EN	ENTSTAN
00167 000630002	WAR EN	WAR
00168 000640004	HUE T	HUE
00169 000640012	BRUECH T	BRUECH
00170 000640019	HA T	HA
00171 000640027	PRODUZE>IER EN	PRODUZE>IER
00172 000640036	VER KAF EN	VERKAF
00173 000650020	SIN	
00174 000650029	LEI EN	LEI
00175 000650033	HI REN	HIREN
00176 000650036	AS	
00177 000660002	HUE T	HUE
00178 000660020	REVOLUTIONNE>IER T	REVOLUTIONNE>IER
00179 000660036	AS	
00180 000660049	AS	
00181 000670003	HU	
00182 000670011	PROFITE>IER T	PROFITE>IER
00183 000680003	LA*IT	LA*IT
00184 000690000	AS	
00185 000690012	BEAFLOSS T	BEAFLOSS
00186 000690013	GI N	GI
00187 000710023	BESTI N	BESTI
00188 000720003	HU N	HU
00189 000720006	GE WUESS ENE	WUESS
00190 000720000	GE LIEF T	LIEF
00191 000720012	HU	
00192 000720016	ERNE>IER T	ERNE>IER
00193 000720019	OF GE NOTZ T	OFNOTZ
00194 000730002	VER SCHIDD ENEN	VERSCHIDD
00195 000740007	GE WUESS ENE	WUESS
00196 000740009	AS	
00197 000740022	STI N	STI
00198 000740020	HI ER KE*NN T	HIERKE*NN
00199 000740036	PASS T	PASS
00200 000750003	AS	
00201 000750019	ER AN (ZE) PLANZ EN	ERANPLANZ
00202 000750020	GE WIEL T	WIEL
00203 000750030	WAR	WAR
00204 000760004	GEFE>IER T	GEFE>IER
00205 000760010	HA*T T	HA*T
00206 000760013	GESI N	GESI
00207 000760015	HA*T T	HA*T
00208 000760017	E*NNER LOOSS	E*NNERLOOSS
00209 000760019	AS	
00210 000760020	GE BRAUCH T	BRAUCH
00211 000760029	GI N	GI
00212 000760035	SICH EN	SICH
00213 000770001	OF GESI	OFGESI
00214 000770005	STEE T	STEE
00215 000770019	KRICH	KRICH
00216 000770023	HU N	HU
00217 000780010	KE*NN T	KE*NN
00218 000780015	FEST HAL EN	FESTHAL
00219 000780025	SIN	
00220 000780039	ENT STAN E	ENTSTAN

00221	000790002	GOUF	GOUF
00222	000790008	INVESTE>IER T	INVESTE>IER
00223	000790020	GI N	GI
00224	000790021	AS	
00225	000800012	KOMM	KOMM
00226	000800013	AS	
00227	000800031	REAGE>IER T	REAGE>IER
00228	000800032	HU	
00229	000800036	AS	
00230	000800044	KOMM	KOMM
00231	000800061	KENN EN	KENN
00232	000810004	AS	
00233	000810008	A GE RIICHT	ARIICHT
00234	000810018	AUS ZE NOTZ EN	AUSNOTZ
00235	000820005	VER STEE T	VERSTEE
00236	000830005	PRODUZE>IER EN	PRODUZE>IER
00237	000840004	AS	
00238	000850002	HU N	HU
00239	000850006	INVESTE>IER T	INVESTE>IER
00240	000850016	PRODUZE>IER EN	PRODUZE>IER
00241	000850017	KE*NN T	KE*NN
00242	000850023	AS	
00243	000860004	SIN	
00244	000860012	ER AUS KE*NN T	ERAUSKE*NN
00245	000860024	DRA SCHE>ISS T	DRASCHE>ISS
00246	000860030	STOPP EN	STOPP
00247	000870012	HUE T	HUE
00248	000870014	AS	
00249	000880004	LA*IT	LA*IT
00250	000880018	GOUF	GOUF
00251	000880030	OP GE BAU T	OPBAU
00252	000880031	GI N	GI
00253	000880032	AS	
00254	000890002	HA T	HA
00255	000890022	GE ZUNN	ZUNN
00256	000890030	MOBILISE>IER T	MOBILISE>IER
00257	000890031	HUE T	HUE
00258	000900003	AS	
00259	000910004	HU N	HU
00260	000910016	GESI N	GESI
00261	000910032	ER VIR GE RUFF	ERVIRRUFF
00262	000910043	WAR	WAR
00263	000920004	SIN	
00264	000920018	GE HOLL	HOLL
00265	000920019	GI N	GI
00266	000920026	GI N	GI
00267	000930009	VIR GE NANN TE	VIRNANN
00268	000930016	E*NNER HUEL EN	E*NNERHUEL
00269	000930024	GO EN	GO
00270	000930028	ANZEFE>IER EN	ANZEFE>IER
00271	000930035	ER OF (ZE) STE*TZ EN	EROFSTE*TZ
00272	000930051	KRE>IE N	KRE>IE
00273	000940004	HUE T	HUE
00274	000940015	GE HOLL	HOLL
00275	000940016	GI N	GI
00276	000940023	GI N	GI

00277	000950011	E*NNER HUEL EN	E*NNERHUEL
00278	000950019	GO EN	GO
00279	000950023	ANZEFE>IER EN	ANZEFE>IER
00280	000950030	ER OF (ZE) SETZ EN	EROFSETZ
00281	000950046	KRE>IE N	KRE>IE
00282	000960004	HUE T	HUE
00283	000960013	OF GE GRAFF	OPGRAFF
00284	000960018	HUE T	HUE
00285	000960023	E*N GE SAT	E*NSAT
00286	000960037	GOUF EN	GOUF
00287	000970008	AN ZE SETZ EN	ANSETZ
00288	000980009	GI N	GI
00289	000980018	AS	
00290	000980014	DUER SCHLO EN	DUERSCHLO
00291	000990018	A GAANG EN	AGAANG
00292	000990011	AS	
00293	001000013	HUE T	HUE
00294	001000036	AGEFE>IER T	AGEFE>IER
00295	001000040	AS	
00296	001000046	OF GE SE>CHER T	OFSE>CHER
00297	001000047	GI N	GI
00298	001010007	AS	
00299	001010008	ZE SO E	ZESO
00300	001010010	OF GE FAANG E	OPFAANG
00301	001010011	GI N	GI
00302	001020010	HUE T	HUE
00303	001020018	OF GE BRACH	OFBRACH
00304	001020037	AUS GE SPILL T	AUSSPILL
00305	001040003	HU	
00306	001040010	GE LEESCHT	LEESCHT
00307	001040018	WAR EN	WAR
00308	001050004	HUE T	HUE
00309	001050010	AUS GE BAU T	AUSBAU
00310	001050018	GE SAT	SAT
00311	001050030	HA T	HA
00312	001060008	HA T	HA
00313	001060018	HUE T	HUE
00314	001060022	GE MAACH	MAACH
00315	001060030	LEI E	LEI
00316	001060041	ER REECH E	ERREECH
00317	001060042	KE*NN T	KE*NN
00318	001070003	KONN T	KONN
00319	001070006	RAISONNE>IER EN	RAISONNE>IER
00320	001070012	EMORGANISE>IER EN	EMORGANISE>IER
00321	001070013	SOLL T	SOLL
00322	001070019	MISS EN	MISS
00323	001070021	MAACH EN	MAACH
00324	001070026	FRA*I GE STALL TE	FRA*ISTALL
00325	001070029	GESCHE>I EN	GESCHE>I
00326	001080015	A GE SAT	ASAT
00327	001080016	GI N	GI
00328	001080021	NO GE DUECH T	NODUECH
00329	001080022	HU N	HU
00330	001080027	FRA*I GE STALL TE	FRA*ISTALL
00331	001080038	A SETZ E	ASETZ
00332	001080039	KE*NN T	KE*NN

00333	001090008	WAR EN	WAR
00334	001090015	WAR EN	WAR
00335	001090027	GE GOLL EN	GOLL
00336	001090028	HUE T	HUE
00337	001090030	GOUF	GOUF
00338	001090044	GE HOLL	HOLL
00339	001090062	GOUF E	GOUF
00340	001090072	A GE SAT	ASAT
00341	001100014	AUS ZE BAU EN	AUSBAU
00342	001100025	ER AN (ZE) BRE>NG EN	ERANBRE>NG
00343	001110007	BE STE*MM TEN	BESTE*MM
00344	001110011	AUS BEZUEL T	AUSBEZUEL
00345	001110012	KRUT EN	KRUT
00346	001110014	GOUF	GOUF
00347	001110020	LIICHT	LIICHT
00348	001110021	GE MAACH	MAACH
00349	001110035	IUWER ZE WIESSEL EN	IUWERWIESSEL
00350	001110045	SIN	
00351	001120010	HUE T	HUE
00352	001120019	AUS GE HOLLEF	AUSHOLLEF
00353	001120023	OF ZE DECK EN	OFDECK
00354	001120036	ER AN HUEL EN	ERANHUEL
00355	001120038	KE*NN EN	KE*NN
00356	001130002	HA T	HA
00357	001130015	DRO EN	DRO
00358	001130023	HUE T	HUE
00359	001140005	HUE T	HUE
00360	001140012	AUS GE BIL T	AUSBIL
00361	001140026	BESCHA*FTEG EN	BESCHA*FTEG
00362	001150008	AS	
00363	001160003	GI N	GI
00364	001160014	NENN EN	NENN
00365	001160028	OP TRE*T T	OPTRE*T
00366	001170010	GI N	GI
00367	001170014	WUESS EN	WUESS
00368	001180005	ME>CH T	ME>CH
00369	001190009	ME>CH T	ME>CH
00370	001190014	FE*LL T	FE*LL
00371	001200008	SCHAFF EN	SCHAFF
00372	001210002	KOMM EN	KOMM
00373	001220004	BE MIERK T	BEHIERK
00374	001220006	HA T	HA
00375	001220023	BE TRAFF ENE	BETRAFF
00376	001220031	GEFALL	GEFALL
00377	001220032	SIN	
00378	001220038	FORT ZE KOMM EN	FORTKOMM
00379	001220045	MISS EN	MISS
00380	001220047	SCHAFF EN	SCHAFF
00381	001230007	AS	
00382	001230013	E*H GE SCHLO EN	E*H SCHLO
00383	001230018	GESI N	GESI
00384	001230019	HU N	HU
00385	001230033	BE STAN EN	BESTAN
00386	001230034	HUE T	HUE
00387	001230039	SCHAAF EN	SCHAAF
00388	001230049	HAL EN	HAL
00389	001230053	OF ZE SE>CHER EN	OFSE>CHER
00390	001230063	ER HAL EN	ERHAL

00391	001240014	GE KE*NNEG T	KE*NNEG
00392	001240015	KRUT EN	KRUT
00393	001240025	GE SCHLO EN	SCHLO
00394	001240026	HU N	HU
00395	001240028	KONN TEN	KONN
00396	001240043	AN UECHT HUEL EN	ANUECHTHUEL
00397	001250002	VER SCHIDD ENE	VERSCHIDD
00398	001250010	SOU GE NANN TEN	SOUNANN
00399	001250013	KANN	KANN
00400	001250019	SCHWA*TZ EN	SCHWA*TZ
00401	001260005	GE KUCK T	KUCK
00402	001260011	VERGLACH	VERGLACH
00403	001260013	HUE T	HUE
00404	001260022	GE PACK T	PACK
00405	001270006	GI N	GI
00406	001270015	AS	
00407	001270017	KANN	KANN
00408	001270022	GE PLANG TEN	PLANG
00409	001270032	GESI N	GESI
00410	001270044	SIN	
END OF DATA			

.TRANSIT3 pour le néerlandais :

\*00132

\*06340

00001	000250005	LIJK EN	LIJK
00002	000250013	IMPONER EN	IMPONER
00003	000260005	GEBEUR DE	GEBEUR
00004	000260011	DOOR DRING EN	DOORDRING
00005	000260024	SLUIKER ENDE	SLUIKER
00006	000270003	WAS	
00007	000270012	WERD	WERD
00008	000270013	BE STEED	BESTEED
00009	000270027	HEBB EN	HEBB
00010	000270028	OP GE RICHT	OPRICHT
00011	000270036	ZIJN	ZIJN
00012	000270037	BE LOON D	BELOON
00013	000280003	IS	
00014	000280006	GE DRAAI D	DRAAI
00015	000290005	WORD EN	WORD
00016	000290010	UIT SPRAK EN	UITSPRAK
00017	000290016	WEER GE GEV EN	WEERGEV
00018	000290019	GE WAAR SCHUW D	GEWAARSCHUW
00019	000290020	WORD T	WORD
00020	000300003	WORD T	WORD
00021	000300005	DOOR DRONG EN	DOORDRONG
00022	000300007	ZIJN	ZIJN
00023	000300016	GE WAPEN D	WAPEN
00024	000300020	DOOR GROND EN	DOORGROND
00025	000300025	STREV EN	STREV
00026	000300028	OVER HEERS EN	OVERHEERS
00027	000310004	GAA T	GAA
00028	000310019	STAA N	STAA
00029	000310023	LEVER EN	LEVER
00030	000320003	MEEN T	MEEN
00031	000320018	HOUD EN	HOUD
00032	000320022	IS	
00033	000320024	DOOR GROND EN	DOORGROND
00034	000320034	ZAK EN	ZAK
00035	000320044	HEBB EN	HEBB
00036	000320046	GE ACHT	ACHT
00037	000320047	WORD EN	WORD
00038	000320051	HEBB EN	HEBB
00039	000330005	UIT GAA NDE	UITGAA
00040	000330006	IS	
00041	000330009	GE KOZ EN	KOZ
00042	000330021	BIED EN	BIED
00043	000330032	BE HANDEL EN	BEHANDEL
00044	000340003	ZIJN	ZIJN
00045	000340005	VIND EN	VIND
00046	000340009	AF GE DRUK TE	AFDRUK
00047	000340015	ZIJN	ZIJN
00048	000350002	WORD EN	WORD
00049	000350013	BE HANDEL D	BEHANDEL
00050	000350033	WORD T	WORD
00051	000350034	BE SPROK EN	BESPROK
00052	000360002	DOEK EN	DOEK
00053	000360015	WORD EN	WORD
00054	000360016	BE HANDEL D	BEHANDEL

00055	000370004	KUNN EN	KUNN
00056	000370019	PASSER EN	PASSER
00057	000370025	WORD EN	WORD
00058	000370026	BE SPROK EN	BESPROK
00059	000380004	WORD EN	WORD
00060	000380014	BE SPROK EN	BESPROK
00061	000380017	ZIJN	ZIJN
00062	000380025	VOOR ZIE N	VOORZIE
00063	000390003	ZULL EN	ZULL
00064	000390011	WILL EN	WILL
00065	000390012	VOL STAA N	VOLSTAA
00066	000400002	WILL EN	WILL
00067	000400006	LEZ EN	LEZ
00068	000400019	WORD EN	WORD
00069	000400020	TOE GE PAS T	TOEPAS
00070	000400035	WORD T	WORD
00071	000400036	BE SPROK EN	BESPROK
00072	000410006	KIJK T	KIJK
00073	000420021	WORD T	WORD
00074	000420028	AF GE SLOT EN	AFSLOT
00075	000430005	IS	
00076	000430006	BE OOG D	BEOOG
00077	000430021	GEV EN	GEV
00078	000440007	IS	
00079	000440011	ONDER GE SCHIK T	ONDERSCHIK
00080	000440012	GE HAAK T	HAAK
00081	000440015	ZIJN	ZIJN
00082	000440019	GE BRUIK T	BRUIK
00083	000440024	ZULL EN	ZULL
00084	000440025	VIND EN	VIND
00085	000450002	IS	
00086	000450012	IS	
00087	000450018	VER MELD	VERMELD
00088	000460006	ZOU	
00089	000460009	KUNN EN	KUNN
00090	000460010	WEKK EN	WEKK
00091	000460014	WORD EN	WORD
00092	000460015	BE SPROK EN	BESPROK
00093	000460022	ZAL	
00094	000460023	WORD EN	WORD
00095	000460024	TOE GE PAS T	TOEPAS
00096	000460030	WORD T	WORD
00097	000460031	GEORGANISEER D	ORGANISEER
00098	000470009	KOM EN	KOM
00099	000470010	WORD T	WORD
00100	000470014	NA GE STREEF D	NASTREEF
00101	000470021	GEV EN	GEV
00102	000470026	ZAL	
00103	000470027	HELP EN	HELP
00104	000470035	VORM EN	VORM
00105	000480004	WIL	
00106	000480006	ZIJN	ZIJN
00107	000480008	UIT SPREK EN	UITSPREK
00108	000480017	ILLUSTRER EN	ILLUSTRER
00109	000480032	HEBB EN	HEBB
00110	000480033	BIJ GE STAA N	BIJSTAA
00111	000490009	WAR EN	WAR
00112	000520007	GEBRUIK EN	GEBRUIK
00113	000520008	WORD T	WORD

00114	000520009	OP GE WEK T	OPWEK
00115	000520019	GE NOEM D	NOEM
00116	000520020	WORD EN	WORD
00117	000530004	WORD T	WORD
00118	000530007	OM GE ZET	OMZET
00119	000540005	WORD T	WORD
00120	000540009	GE LEVER D	LEVER
00121	000550004	IS	
00122	000560003	WORD EN	WORD
00123	000560009	TOE GE PAS T	TOEPAS
00124	000560025	LEVER EN	LEVER
00125	000560026	WORD T	WORD
00126	000560032	AAN GE DREV EN	AANDREV
00127	000570005	IS	
00128	000570012	VOORT BE WEEG T	VOORTBEWEEG
00129	000570017	GEBEUR T	GEBEUR
00130	000570025	IS	
00131	000570034	OP GE POMP T	OPPOMP
00132	000580010	IS	
00133	000580017	AAN DRIJF T	AANDRIJF
00134	000590007	ZIJN	ZIJN
00135	000590018	BLAZ EN	BLAZ
00136	000600005	WORD T	WORD
00137	000600006	OP GE WEK T	OPWEK
00138	000600009	AAN GE DREV EN	AANDREV
00139	000620006	BENODIG DE	BENODIG
00140	000620008	WORD T	WORD
00141	000620011	OP GE WEK T	OPWEK
00142	000620014	VER WARM D	VERWARM
00143	000630013	KOM EN	KOM
00144	000630015	BE SCHOUW EN	BESCHOUW
00145	000640011	IS	
00146	000650002	KOM T	KOM
00147	000650017	BEVAT	BEVAT
00148	000650023	REAGEER T	REAGEER
00149	000660002	BE HOOR T	BEHOOR
00150	000660007	IS	
00151	000660014	BE STAA T	BESTAA
00152	000660022	OM RING D	OMRING
00153	000660029	GE SCHREV EN	SCHREV
00154	000660036	ZIE	
00155	000670002	KAN	
00156	000670005	GE STIL EER D	GESTILEER
00157	000670006	TE RUG VIND EN	TERUGVIND
00158	000670017	MOET	MOET
00159	000670025	VOOR STELL EN	VOORSTELL
00160	000680003	IS	
00161	000680010	WIL	
00162	000680011	ZEGG EN	ZEGG
00163	000680024	TOON T	TOON
00164	000680028	ZIJN	ZIJN
00165	000680033	VORM EN	VORM
00166	000690006	KAN	
00167	000690009	WORD EN	WORD
00168	000690010	GE SCHREV EN	SCHREV
00169	000700020	STAA N	STAA
00170	000710004	AF GE BEELD E	AFBEELD
00171	000710008	HAL EN	HAL
00172	000710009	KRIJG EN	KRIJG

00173	000720001	ZOU DEN	ZOU
00174	000720006	DOE N	DOE
00175	000720009	ZOU DEN	ZOU
00176	000720011	MERK EN	MERK
00177	000720017	HAL EN	HAL
00178	000720032	IS	
00179	000730002	ZOU	
00180	000730004	BLIJK EN	BLIJK
00181	000730017	IS	
00182	000730022	IS	
00183	000730035	VRIJ KOM T	VRIJKOM
00184	000730039	WORD T	WORD
00185	000730040	GE BRUIK T	BRUIK
00186	000730045	WEKK EN	WEKK
00187	000740006	IS	
00188	000740018	GEBRUIK EN	GEBRUIK
00189	000740020	ZIE	
00190	000770004	WEEG T	WEEG
00191	000770012	BRENG EN	BRENG
00192	000790006	VER MOG EN	VERMOG
00193	000790010	IS	
00194	000790019	DRUKK EN	DRUKK
00195	000790020	ZIJN	ZIJN
00196	000790021	VER SCHILL ENDE	VERSCHILL
00197	000790038	WORD T	WORD
00198	000790039	UIT GE GAA N	UITGAA
00199	000790049	KEN T	KEN
00200	000800010	GE BRUIK T	BRUIK
00201	000800018	IS	
00202	000810015	OM ZET	OMZET
00203	000810018	DUID T	DUID
00204	000810032	ZIJN	ZIJN
00205	000810034	VIND T	VIND
00206	000820014	ZAL	
00207	000820020	WORD EN	WORD
00208	000820021	GE BRUIK T	BRUIK
00209	000830005	PAS T	PAS
00210	000830016	ZAL	
00211	000830017	ZIJN	ZIJN
00212	000830020	OVER EEN KOM T	OVEREENKOM
00213	000850016	DUID EN	DUID
00214	000850017	ZAL	
00215	000850033	WORD EN	WORD
00216	000850034	GE BRUIK T	BRUIK
00217	000850037	ZAL	
00218	000850047	WORD EN	WORD
00219	000850048	GE NOEM D	NOEM
00220	000860002	ZULL EN	ZULL
00221	000860005	WORD EN	WORD
00222	000860006	GE BRUIK T	BRUIK
00223	000870005	BE HOOR T	BEHOOR
00224	000880011	KOM EN	KOM
00225	000880012	GE BRUIK T	BRUIK
00226	000890003	KOM EN	KOM
00227	000900003	ZULL EN	ZULL
00228	000900006	WORD EN	WORD
00229	000900007	GE BRUIK T	BRUIK
00230	000900025	BE SPAAR D	BESPAAR
00231	000900026	BLIJV EN	BLIJV

00232	000910003	ZIJN	ZIJN
00233	000910012	WIL	
00234	000910013	DOE N	DOE
00235	000910014	GELOV EN	GELOV
00236	000920003	ZIT	
00237	000920011	GECONCENTREER D	CONCENTREER
00238	000920025	IN NEEM T	INNEEM
00239	000930004	IS	
00240	000930017	GE LAD EN	LAD
00241	000930026	ZIJN	ZIJN
00242	000930041	TE KEN EN	TEKEN
00243	000950018	VOOR GE STEL D	VOORSTEL
00244	000970010	GEEF T	GEEF
00245	000980007	HEBB EN	HEBB
00246	000990003	DRAAG T	DRAAG
00247	000990014	ON GE LAD EN	ONLAD
00248	000990015	IS	
00249	001000004	TREKK EN	TREKK
00250	001000012	BLIJV EN	BLIJV
00251	001000016	VALL EN	VALL
00252	001010002	STUIT EN	STUIT
00253	001020005	HOET EN	HOET
00254	001020006	ZIJN	ZIJN
00255	001020007	IS	
00256	001020012	HOG EN	HOG
00257	001020014	VER WACHT EN	VERWACHT
00258	001020023	OM RING ENDE	OKRING
00259	001020025	TRACHT EN	TRACHT
00260	001020028	VOER EN	VOER
00261	001020042	OM RING ENDE	OKRING
00262	001020043	VOOR WERP EN	VOORWERP
00263	001020044	ZULL EN	ZULL
00264	001020045	GE DRAG EN	DRAG
00265	001030004	LEER T	LEER
00266	001030010	GE PLAATS T	PLAATS
00267	001030011	HOET	HOET
00268	001030012	WORD EN	WORD
00269	001030020	WORD EN	WORD
00270	001030021	OP GE VAT	OPVAT
00271	001040004	ZIE N	ZIE
00272	001040015	DRAAI EN	DRAAI
00273	001040018	UITGESHEER D	UITGESHEER
00274	001050002	ZIJN	ZIJN
00275	001050012	GE LAD EN	LAD
00276	001050019	SAMEN BIND EN	SAMENBIND
00277	001060010	BE SPROK EN	BESPROK
00278	001060015	BLIJV EN	BLIJV
00279	001060018	ON AAN GE TAST	ONAANTAST
00280	001070002	ZULL EN	ZULL
00281	001070007	BE KIJK EN	BEKIJK
00282	001090006	ZIJN	ZIJN
00283	001090010	IS	
00284	001100004	BLIJK T	BLIJK
00285	001100008	ZIJN	ZIJN
00286	001100014	DANK T	DANK
00287	001110006	GE ZEG D	ZEG
00288	001110007	WERD	WERD
00289	001110008	BEVAT	BEVAT
00290	001120001	WAS	

00291	001120009	ZOU	
00292	001120017	MOET EN	MOET
00293	001120018	WEG EN	WEG
00294	001130001	ZOU	
00295	001130011	NA WEG EN	NAWEG
00296	001130014	ZOU	
00297	001130015	BLIJK EN	BLIJK
00298	001130019	KLOP T	KLOP
00299	001130031	VER HOUD EN	VERHOUD
00300	001140005	IS	
00301	001140011	BE STAA T	BESTAA
00302	001140017	IS	
00303	001140027	DRAAG T	DRAAG
00304	001140035	DRAAG T	DRAAG
00305	001150003	MOET	MOET
00306	001150007	BEVATT EN	BEVATT
00307	001160002	KAN	
00308	001160005	WORD EN	WORD
00309	001160006	GEKARAKTERISEER D	KARAKTERISEER
00310	001160022	AAN GEEF T	AANGEEF
00311	001160047	GE NOEM D	NOEM
00312	001170007	WORD T	WORD
00313	001170012	AAN GE DUID	AANDUID
00314	001180003	MOET	MOET
00315	001180008	WORD EN	WORD
00316	001180009	AAN GE BRACH T	AANBRACH
00317	001190003	HEBB EN	HEBB
00318	001190014	HEBB EN	HEBB
00319	001190024	BLIJK T	BLIJK
00320	001190029	ZIJN	ZIJN
00321	001190031	HEEF T	HEEF
00322	001200002	BLIJK T	BLIJK
00323	001200012	HEBB EN	HEBB
00324	001200014	HEEF T	HEEF
00325	001210002	VER SCHILL ENDE	VERSCHILL
00326	001210014	VER SCHILL EN	VERSCHILL
00327	001210016	NOEM T	NOEM
00328	001220003	VOEG T	VOEG
00329	001230006	VOEG T	VOEG
00330	001230028	GE NOEM DE	NOEM
00331	001240007	HEEF T	HEEF
00332	001240013	VER WORV EN	VERWORV
00333	001240030	GE SCHREV EN	SCHREV
00334	001240031	WORD T	WORD
00335	001250005	ZIJN	ZIJN
00336	001250017	KOM EN	KOM
00337	001250025	OP GE SOM D	OP SOM
00338	001250038	AAN TREFF EN	AANTREFF
00339	001260008	VIND EN	VIND
00340	001260009	IS	
00341	001260017	BE STAA T	BESTAA
00342	001270010	WORD T	WORD
00343	001270016	GE BRUIK T	BRUIK
00344	001270020	WORD T	WORD
00345	001270022	GEKARAKTERISEER D	KARAKTERISEER
00346	001280004	ZAL	
00347	001280009	GE BRUIK T	BRUIK
00348	001280010	WORD EN	WORD
00349	001280014	GEEF T	GEEF

00350	001280021	BE KEN DE	BEKEN
00351	001290010	BE HANDEL DE	BEHANDEL
00352	001290011	BE GRIPP EN	BEGRIFF
00353	001290012	WORD T	WORD
00354	001290013	ONDER STAA ND	ONDERSTAA
00355	001290021	SAMEN GE VAT	SAMENVAT
00356	001300007	ZIJN	ZIJN
00357	001300008	IN GE VOER D	INVOER
00358	001300030	VER WAAR LOOS D	VERWAARLOOS
00359	001300031	IS	
00360	001320003	HEEF T	HEEF
00361	001320009	KUNN EN	KUNN
00362	001320010	BE PAL EN	BEPAL
00363	001320013	BLIJK T	BLIJK
00364	001320016	IN HOUD	INHOUD
00365	001320018	TOE NEEH T	TOENEEM
00366	001330002	ZITT EN	ZITT
00367	001330011	GE PAK T	PAK
00368	001340002	HOUD T	HOUD
00369	001350019	HEBB EN	HEBB
00370	001350025	BE TREF T	BETREF
00371	001350028	WERK EN	WERK
00372	001350033	AF STOT END	AFSTOT
00373	001350046	KOM EN	KOM
00374	001360002	MOET	MOET
00375	001360008	WERK EN	WERK
00376	001370002	GE BLEK EN	BLEK
00377	001370003	IS	
00378	001370010	GAA T	GAA
00379	001370011	WERK EN	WERK
00380	001370017	ZIJN	ZIJN
00381	001370022	ZULL EN	ZULL
00382	001370028	NOEM EN	NOEM
00383	001380003	WERK T	WERK
00384	001380028	ZIJN	ZIJN
00385	001390007	KON DEN	KON
00386	001390008	SPREK EN	SPREK
00387	001390014	IS	
00388	001390019	ZOU	
00389	001390020	ZIJN	ZIJN
00390	001390024	SPLITS EN	SPLITS
00391	001390031	KUNN EN	KUNN
00392	001390045	VER RICHT	VERRICHT
00393	001390046	ZOU	
00394	001390047	MOET EN	MOET
00395	001390048	WORD EN	WORD
00396	001390053	BREK EN	BREK
00397	001400002	IS	
00398	001400011	ZAL	
00399	001400012	ZIJN	ZIJN
00400	001410012	ZITT EN	ZITT
00401	001410023	GE PAK T	PAK
00402	001410032	ZIJN	ZIJN
00403	001420001	ZOU DEN	ZOU
00404	001420005	VLIEG EN	VLIEG
00405	001420007	ZOU	
00406	001420013	VRIJ KOM EN	VRIJKOM
00407	001420021	MOET	MOET
00408	001420026	KOST EN	KOST

00409	001420033	VER HINDER EN	VERHINDER
00410	001420036	DOOR BREK EN	DOORBREK
00411	001430008	IS	
00412	001430012	WERK EN	WERK
00413	001430022	GE GEV EN	GEV
00414	001430026	WIL	
00415	001430028	ZEGG EN	ZEGG
00416	001440001	GE GEV EN	GEV
00417	001440012	KAN	
00418	001440017	HER LEID EN	HERLEID
00419	001440026	IS	
00420	001440032	ZULL EN	ZULL
00421	001440033	GEBRUIK EN	GEBRUIK
00422	001450005	IS	
00423	001460003	STEL T	STEL
00424	001460008	VOOR KOM ENDE	VOORKOM
00425	001460012	BEGINN END	BEGINN
00426	001460017	EINDIG END	EINDIG
00427	001470008	VRAAG T	VRAAG
00428	001470009	IS	
00429	001470014	UIT GE DRUK T	UITDRUK
00430	001470015	IS	
00431	001480003	IS	
00432	001490010	VRIJ KOM T	VRIJKOM
00433	001490023	VOLG ENDE	VOLG
00434	001490025	IS	
00435	001490031	AF HANG T	AFHANG
00436	001500002	IS	
00437	001510001	GAA N	GAA
00438	001510009	PLAKK EN	PLAKK
00439	001510012	ZAL	
00440	001510017	TOE NEH EN	TOENEN
00441	001510026	PLAATS EN	PLAATS
00442	001510031	VER BIND EN	VERBIND
00443	001520006	ZIE N	ZIE
00444	001520010	BLIJK T	BLIJK
00445	001520017	AF NEEM T	AFNEEM
00446	001520028	KRIJG EN	KRIJG
00447	001540012	UIT GE ZET	UITZET
00448	001550011	ILLUSTRER EN	ILLUSTRER
00449	001550012	ZIJN	ZIJN
00450	001550014	GE BOG EN	BOG
00451	001550016	IN GE TEKEN D	INTEKEN
00452	001550023	AF GE BEELD E	AFBEELD
00453	001550025	WEER GEV EN	WEERGEV
00454	001550032	ONT STAA N	ONTSTAA
00455	001560009	ZIJN	ZIJN
00456	001560010	AAN GE GEV EN	AANGEV
00457	001570008	WORD T	WORD
00458	001570011	BE REIK T	BEREIK
00459	001570020	IS	
00460	001580004	IS	
00461	001580012	WORD T	WORD
00462	001580020	TOE NEEM T	TOENEEM
00463	001590002	WORD T	WORD
00464	001590021	ZAL	
00465	001590022	KOM EN	KOM
00466	001590033	BE HOUD EN	BEHOUD
00467	001600003	HEET	HEET

00468	001600021	VIND EN	VIND
00469	001610016	KWAM EN	KWAM
00470	001610018	KAN	
00471	001610021	ZIJN	ZIJN
00472	001610022	GE WEK T	WEK
00473	001610035	MOET EN	MOET
00474	001610036	ZIJN	ZIJN
00475	001620002	IS	
00476	001620003	GE BLEK EN	BLEK
00477	001620009	IS	
00478	001620013	IS	
00479	001620026	ZAL	
00480	001620027	WORD EN	WORD
00481	001620028	IN GE GAA N	INGAA
00482	001630004	LAAT	LAAT
00483	001630005	ZIE N	ZIE
00484	001630006	KOM T	KOM
00485	001630011	ZIJN	ZIJN
00486	001630012	RECHT	RECHT
00487	001640009	GAA T	GAA
00488	001640010	SPEL EN	SPEL
00489	001640011	ZIE T	ZIE
00490	001640014	OP TRED EN	OPTRED
00491	001650008	ZIJN	ZIJN
00492	001660008	NOEM DEN	NOEM
00493	001680008	ZOU DEN	ZOU
00494	001680011	KUNN EN	KUNN
00495	001680012	SCHIJN EN	SCHIJN
00496	001680017	ZOU DEN	ZOU
00497	001680018	MOET EN	MOET
00498	001680019	ONT LEN EN	ONTLEN
00499	001690004	GEBEUR T	GEBEUR
00500	001690005	IS	
00501	001690009	VER SCHILL ENDE	VERSCHILL
00502	001690011	WORD EN	WORD
00503	001690012	OM GE ZET	OMZET
00504	001700005	BLIJK T	BLIJK
00505	001700006	HEBB EN	HEBB
00506	001700013	KOM T	KOM
00507	001710008	GAA T	GAA
00508	001710024	VRIJ KOM T	VRIJKOM
00509	001720005	ZULL EN	ZULL
00510	001720008	WORD EN	WORD
00511	001720009	BE SPROK EN	BESPROK
00512	001720017	MAK EN	MAK
00513	001730010	GE BRUIK T	BRUIK
00514	001730021	ZULL EN	ZULL
00515	001730022	KRIJG EN	KRIJG
00516	001730024	HANG EN	HANG
00517	001730039	AF NEEM T	AFNEEM
00518	001740007	WINN EN	WINN
00519	001740008	MOET	MOET
00520	001740013	DOE N	DOE
00521	001740014	UIT EEN VALL EN	UITEENVALL
00522	001750002	WERD	WERD
00523	001750015	ZIJN	ZIJN
00524	001750019	HEBB EN	HEBB
00525	001760003	GAA T	GAA
00526	001760013	WORD	WORD

00527 001760014	GE GEV EN	GEV
00528 001760020	STELL EN	STELL
00529 001770007	BINNEN DRING T	BINNENDRING
00530 001770009	ONT STAA T	ONTSTAA
00531 001770022	ZIE	
00532 001780009	IS	
00533 001780010	VRIJ GE KOM EN	VRIJKOM
00534 001780012	BEVIND T	BEVIND
00535 001780019	STAA T	STAA
00536 001790005	KAN	
00537 001790008	ZIJN	ZIJN
00538 001790012	KWIJT RAK EN	KWIJTRAK
00539 001800003	IS	
00540 001810007	GE VALL EN	VALL
00541 001810009	GEBEUR T	GEBEUR
00542 001810013	REK T	REK
00543 001810023	GE GEV EN	GEV
00544 001810045	IN SNOER T	INSNOER
00545 001810058	UIT EEN Vlieg T	UITEENVLIEG
00546 001810065	VRIJ KOM EN	VRIJKOM
00547 001830009	ON STAA T	ONSTAA
00548 001830015	AAN GE SLAG EN	AANSLAG
00549 001840001	LET	
00550 001840012	IS	
00551 001850003	STELL EN	STELL
00552 001860003	ZIJN	ZIJN
00553 001860011	VER DEEL D	VERDEEL
00554 001860014	KRIJG T	KRIJG
00555 001860029	HEEF T	HEEF
00556 001860030	GE KREG EN	KREG
00557 001860038	IS	
00558 001860043	VRIJ KOM EN	VRIJKOM
00559 001880002	AAN GE SLAG EN	AANSLAG
00560 001880007	VAL T	VAL
00561 001880015	UIT ZEND EN	UITZEND
00562 001890007	IS	
00563 001890014	RESULTER END	RESULTER
00564 001900002	BE SCHOUW T	BESCHOUW
00565 001900009	PLAATS VIND T	PLAATSVIND
00566 001900012	BLIJK T	BLIJK
00567 001900021	KOM EN	KOM
00568 001910001	KWAH EN	KWAH
00569 001910008	WAS	
00570 001910020	ZOU	
00571 001910021	VIND EN	VIND
00572 001920006	OPEN T	OPEN
00573 001920024	SPLIJT EN	SPLIJT
00574 001930001	ZIJN	ZIJN
00575 001930011	VER LOOP T	VERLOOP
00576 001930017	IS	
00577 001940003	HANDEL T	HANDEL
00578 001940017	HEEF T	HEEF
00579 001950012	VER LOP EN	VERLOP
00580 001950018	NOEM T	NOEM
00581 001950029	LAAT	LAAT
00582 001950030	VER LOP EN	VERLOP
00583 001960005	WORD T	WORD
00584 001960017	BE DOEL D	BEDOEL
00585 001970005	WORD EN	WORD

00586 001970006	GE BOUW D	BOUW
00587 001970014	VRIJ KOM T	VRIJKOM
00588 001970018	WORD T	WORD
00589 001970019	GE BRUIK T	BRUIK
00590 001970029	WEKK EN	WEKK
00591 001980004	ONT STAA N	ONTSTAA
00592 001990008	VRIJ KOM T	VRIJKOM
00593 001990009	MOET	MOET
00594 001990016	BE DRAG EN	BEDRAG
00595 001990019	OVER EEN KOM T	OVEREENKOM
00596 002000003	KOM T	KOM
00597 002000022	WEG SCHIET EN	WEGSCHIET
00598 002010003	HEBB EN	HEBB
00599 002010009	VOER EN	VOER
00600 002020007	WORD T	WORD
00601 002020008	OM GE ZET	OMZET
00602 002020009	ZAL	
00603 002020012	VOLG ENDE	VOLG
00604 002020014	WORD EN	WORD
00605 002020015	BE SPROK EN	BESPROK
00606 002040008	IS	
00607 002040010	GE ZORG D	ZORG
00608 002040025	ZIJN	ZIJN
00609 002050001	ZOU	
00610 002050017	BIJ EEN VOEG EN	BIJEENVOEG
00611 002050019	WEG EN	WEG
00612 002050022	ZOU	
00613 002050027	BLIJK EN	BLIJK
00614 002050029	ZIJN	ZIJN
00615 002060004	IS	
00616 002060011	KUNN EN	KUNN
00617 002060012	OVER GAA N	OVERGAA
00618 002070004	KOM T	KOM
00619 002070011	VRIJ GE KOM EN	VRIJKOM
00620 002080008	IS	
00621 002080023	LEV EN	LEV
00622 002080026	VER SCHILL END	VERSCHILL
00623 002080027	WORD EN	WORD
00624 002080028	ERVAR EN	ERVAR
00625 002090002	BE DENK E	BEDENK
00626 002090013	LEV EN	LEV
00627 002090014	OM GAA N	OMGAA
00628 002090015	CORRESPONDER EN	CORRESPONDER
00629 002100003	KAN	
00630 002100014	UIT REKEN EN	UITREKEN
00631 002100017	ZAL	
00632 002100018	OP TRED EN	OPTRED
00633 002110005	ZULL EN	ZULL
00634 002110010	BE MERK EN	BEHERK
00635 002110016	IS	
00636 002120002	VIND EN	VIND
00637 002120021	ZIJN	ZIJN
00638 002120026	IS	
00639 002120027	VOOR GE SPIEGEL D	VOORSPIEGEL
00640 002130003	BEGINN EN	BEGINN
00641 002130004	WEEG T	WEEG
00642 002130015	IS	
00643 002140003	WEEG T	WEEG
00644 002150003	IS	

00645	002150012	ZIJN	ZIJN
00646	002150020	AF GE LEID	AFLEID
00647	002160003	VOLG ENDE	VOLG
00648	002160005	ZULL EN	ZULL
00649	002160010	NEER SCHRIJV EN	NEERSCHRIJV
00650	002160022	ZULL EN	ZULL
00651	002160023	KLOFF EN	KLOPP
00652	002170006	ZULL EN	ZULL
00653	002170014	ZIJN	ZIJN
00654	002170018	IS	
00655	002170025	VER LOP ENDE	VERLOF
00656	002170027	VRIJ KOM T	VRIJKOM
00657	002170032	MOET	MOET
00658	002170033	TOE VOEG EN	TOEVOEG
00659	002170038	DOE N	DOE
00660	002170039	VER LOP EN	VERLOF
00661	002190010	IS	
00662	002200007	LAAT	LAAT
00663	002200015	ZIE N	ZIE
00664	002200022	GEPRODUCEER D	PRODUCEER
00665	002200023	VER MOG EN	VERMOG
00666	002200030	WORD T	WORD
00667	002200031	AF GE VOER D	AFVOER
00668	002200046	TRED ENDE	TRED
00669	002200048	GECONDENSEER D	CONDENSEER
00670	002200049	WORD T	WORD
00671	002200060	UIT KOM T	UITKOM
00672	002200069	WORD T	WORD
00673	002200070	AAN GE DUID	AANDUID
00674	002210007	TOON T	TOON
00675	002210032	WORD T	WORD
00676	002210033	OP GE WEK T	OPWEK
00677	002220004	BEVIND T	BEVIND
00678	002220010	WORD T	WORD
00679	002220014	AAN GE DUID	AANDUID
00680	002230002	ZULL EN	ZULL
00681	002230004	TRACHT EN	TRACHT
00682	002230010	AF SPEEL T	AFSPEEL
00683	002230012	BE SCHRIJV EN	BESCHRIJV
00684	002240002	ZAL	
00685	002240004	WORD EN	WORD
00686	002240005	GE MAAK T	MAAK
00687	002240017	TOE NEM END	TOENEM
00688	002240024	TON EN	TON
READY			

.TRANSIT3 pour l'allemand :

L'analyse est complète et les résultats se lisent de la façon suivante :

[ \_ \_ \_ \_ \_ | \_ \_ \_ \_ ] [ VERBE DECOUPE ] [ \_ | \_ \_ \_ ] [ BASE ]  
           1          2                  3                  4   5          6

1 : Nombre de 5 chiffres : numéro de phrase

2 : Nombre de 4 chiffres : numéro d'ordre du mot dans la phrase

3 : Verbe découpé en préverbe(s) / racine / désinence. Le "zu-" de l'infinitif est mis entre parenthèses comme le "ge-" ambigu (préverbe-morphème du participe passé).

4 : Nombre à 1 chiffre : type de verbe

- 1 verbe régulier
- 2 verbe irrégulier
- 3 auxiliaire (haben, sein, werden)
- 4 prétérito-présent
- 5 verbe d'emprunt

5 : Nombre de 3 chiffres :

- 1 infinitif (pur)
- 2 infinitif (avec morphème "zu-")
- 3 indicatif présent
- 4 prétérit
- 5 participe présent décliné
- 6 participe présent non décliné
- 7 participe passé décliné
- 8 participe passé non décliné
- A subjonctif

6 : La base est construite en associant la partie préverbale au radical de l'infinitif.

Le code peut être complété facilement. Les marques retenues sont cependant suffisantes pour le traitement des documents techniques et scientifiques que nous étudions.

000020004	WIRD	3300	WERD
000020009	FORDER N	1130	FORDER
000020016	HAT	3300	HAB
000020021	EMITTIER T	5300	EHITTIER
000020025	VER URSACH T	1300	VERURSACH
000020029	INSTALLIER EN	5130	INSTALLIER
000020030	IST	3300	SEIN
000030003	VER SPRECH ENDER	2500	VERSPRECH
000030005	IST	3300	SEIN
000040003	WANDEL T	1300	WANDEL
000050003	ER FAND	2400	ERFIND
000050017	DAUER TE	1400	DAUER
000050025	KAM	2400	KOMM
000060002	VER SORG TEN	1470	VERSORG
000060010	FAND EN	2400	FIND

000070002	GIB T	2300	GER
000070006	VER BESSER TEN	1470	VERBESSER
000070014	SCHEIN T	2300	SCHEIN
000070017	SEI	3A90	SEIN
000070020	ER REICH T	1300	ERREICH
000070033	ER WART EN	1130	ERWART
000070034	KANN	4300	KO*NN
000080007	BE STEH T	2300	BESTEH
000080031	GE TRENN T	1000	TRENN
000080032	SIND	3300	SEIN
000090003	KO*NN EN	4130	KO*NN
000090009	GE LAD ENE	2700	LAD
000090016	TRANSPORTIER T	5300	TRANSPORTIER
000090017	WERD EN	3130	WERD
000090023	GE LAD ENEN	2700	LAD
000090034	SIND	3300	SEIN
000100007	WIRD	3300	WERD
000100010	ZU GE FU*HR T	1000	ZUFU*HR
000100026	VER SORG T	1300	VERSORG
000110004	BE WIRK T	1300	BEWIRK
000120006	SIND	3300	SEIN
000120012	BE STEH T	2300	BESTEH
000120014	KO*NN EN	4130	KO*NN
000120022	WANDER N	1130	WANDER
000120040	FLIES* ENDEN	2500	FLIES*
000120047	VEREINIG EN	1130	VEREINIG
000120052	VER LA*S* T	2300	VERLASS
000130003	VER BRENN T	2300	VERBRENN
000130013	SETZ T	1300	SETZ
000130016	FREI WERD ENDE	3500	FREIWERD
000130033	VER WEND ET	1300	VERWEND
000130049	TREIB EN	2130	TREIB
000130056	ER ZEUG EN	1130	ERZEUG
000130063	BRING EN	7130	BRING
000130067	AN TREIB EN	2130	ANTREIB
000130068	KANN	4300	KO*NN
000140005	HA*NG T	2300	HA*NG
000140016	ARBEIT	1300	ARBEIT
000140023	BE STIMM T	1300	BESTIMM
000140032	GELANG EN	1130	GELANG
000140033	MUS*	4300	MU*SS
000150003	TRANSPORTIER EN	5130	TRANSPORTIER
000150012	WANDER N	1130	WANDER
000160005	BRAUCH EN	1130	BRAUCH
000160020	TRANSPORTIER EN	5130	TRANSPORTIER
000160022	VER WEND ET	1300	VERWEND
000170007	ER ZEUG EN	1130	ERZEUG
000170008	KANN	4300	KO*NN
000170010	HA*NG T	2300	HA*NG
000170025	GEWINN T	2300	GEWINN
000180006	BE TRIEB ENE	2700	BETREIB
000180008	IST	3300	SEIN
000180025	LIEFER N	1130	LIEFER
000190002	ER HA*LT	2300	ERHALT
000200013	GE STELL TEN	1700	STELL
000200016	ER GIB T	2300	ERGEB
000210004	ER HA*LT	2300	ERHALT
000210018	ENT HALT EN	2130	ENTHALT
000210025	TAUCH T	1300	TAUCH
000210037	UH SPU*L T	1300	UNSPU*L

000220003	WIRK T	1300	WIRK
000220017	ER ZEUG T	1380	ERZEUG
000220023	GE ZEIG TEN	1700	ZEIG
000230002	KANN	4300	KO*NN
000230005	NESS EN	2130	NESS
000230017	VER BIND ET	2300	VERBIND
000240002	EIGN ET	1300	EIGN
000240016	IST	3300	SEIN
000240021	IST	3300	SEIN
000240029	LASS EN	2130	LASS
000240037	ZU SAMMEN FASS EN	1130	ZUSAMMENFASS
000250002	BAU T	1300	BAU
000250022	GESTALT ETE	1470	GESTALT
000250027	IMPRA*GNIER TE	5470	IMPRA*GNIER
000250029	PACK T	1300	PACK
000260004	KANN	4300	KO*NN
000260009	U*BER EIN ANDER STAPEL N	1130	U*BEREINANDERSTAPEL
000270000	LIEFER T	1300	LIEFER
000270010	ER GIB T	2300	ERGEB
000270024	MULTIPLIZIER T	5380	MULTIPLIZIER
000280004	GEHO*R EN	1130	GEHO*R
000290007	BE ZEICHN ET	1300	BEZEICHN
000290009	HAT	3300	HAB
000290027	ER ZEUG EN	1130	ERZEUG
000290039	ARBEIT EN	1130	ARBEIT
000290040	KO*NN EN	4138	KO*NN
000290045	WANDEL T	1300	WANDEL
000290050	GE LIEFER TE	1700	LIEFER
000290061	ZU GE FU*HR T	1800	ZUFU*HR
000290062	WERD EN	3130	WERD
000290063	KANN	4300	KO*NN
000300012	AN GE LIEFER TEN	1700	ANLIEFER
000300014	ENT HALT ENE	2700	ENTHALT
000300029	GE STELL T	1800	STELL
000300030	WIRD	3300	WERD
000300032	ER GIB T	2300	ERGEB
000300056	GE LIEFER TE	1700	LIEFER
000300058	IST	3300	SEIN
000310009	BE STEH T	2300	BESTEH
000320003	BRAUCH T	1300	BRAUCH
000320014	AUF BEREIT ET	1380	AUFBEREIT
000320025	SIND	3300	SEIN
000330004	VER FOLG EN	1130	VERFOLG
000330018	STEIGER N	1130	STEIGER
000330023	SENK EN	1130	SENK
000340002	UNTER SCHEID EN	2130	UNTERSCHIED
000350008	SIND	3300	SEIN
000350018	TRIFF T	2300	TREFF
000350027	UM GE KEHR T	1800	UMKEHR
000360002	KANN	4300	KO*NN
000360015	ARBEIT EN	1130	ARBEIT
000370004	HAT	3300	HAB
000370014	GREIF T	2300	GREIF
000380008	BRAUCH T	1300	BRAUCH
000380023	BE STEH ENDEN	2500	BESTEH
000390005	LEG T	1300	LEG
000390019	MUS*	4300	MU*SS
000390025	FREI HALT EN	2130	FREIHALT
000390034	BILD ET	1300	BILD
000390042	VER RINGER T	1380	VERRINGER

000400016	HAB EN	3130	HAB
000400020	BE WA*HR T	1300	BEWA*HR
000400026	SCHEITER T	1300	SCHEITER
000410011	HA*NG EN	2130	HA*NG
000410021	HAT	3300	HAB
000420002	MUS*	4300	MU*SS
000420008	BE TREIB EN	2130	BETREIB
000420015	VER WEND ET	1300	VERWEND
000420019	NIMM T	2300	NEHM
000420024	ENT HALT ENE	2700	ENTHALT
000420030	UN' VER A*NDER T	1000	VERA*NDER
000420031	VER LA*S* T	2300	VERLASS
000430004	ER REICH T	1300	ERREICH
000430020	AN GE SCHLOSS ENEN	2700	ANSCHLIES*
000430022	HAB EN	3130	HAB
000430023	MUS*	4300	MU*SS
000440004	INTERESSIER T	5300	INTERESSIER
000440016	ARBEIT EN	1130	ARBEIT
000440018	BE SCHA*FTIG T	1300	BESCHA*FTIG
000440029	UNTER SUCH T	1300	UNTERSUCH
000440035	TRANSPORTIER EN	5130	TRANSPORTIER
000440042	VERLANG EN	1130	VERLANG
000440045	SIND	3300	SEIN
000450005	LIEG EN	2130	LIEG
000450024	VER WEND EN	1130	VERWEND
000450025	LA*S* T	2300	LASS
000450031	ZU SAMMEN (ZU) SETZ EN	1200	ZUSAMMENSETZ
000460003	BLEIB T	2300	BLEIB
000460012	UN VER A*NDER T	1000	VERA*NDER
000460023	ERMO*GLICH T	1300	ERMO*GLICH
000460028	EIN SETZ T	1300	EINSETZ
000460036	AUS (ZU) GLEICH EN	2200	AUSGLEICH
000470002	ER KENN T	2300	ERKENN
000470005	FOLG ENDEM	1500	FOLG
000470016	BE HEIZ TES	1700	BEHEIZ
000470025	GEWONN ENE	2700	GEWINN
000470029	TREIB T	2300	TREIB
000470037	DIEN T	1300	DIEN
000470039	BRAUCH T	1300	BRAUCH
000470051	ER ZEUG EN	1130	ERZEUG
000470058	ARBEIT ET	1300	ARBEIT
000480001	WIRD	3300	WERD
000480014	GE NUTZ T	1000	NUTZ
000480017	VER BRAUCH T	1300	VERBRAUCH
000480027	LIEFER N	1130	LIEFER
000490003	VER BRAUCH T	1300	VERBRAUCH
000500002	IST	3300	SEIN
000500011	AUS (ZU) LEG EN	1200	AUSLEG
000500019	BE FRIEDIG T	1300	BEFRIEDIG
000500026	LA*UF T	2300	LAUF
000500034	KOPPEL N	1130	KOPPEL
000500038	BE TRIEB EN	2400	BETREIB
000500039	WIRD	3300	WERD
000500044	AUF KOMM T	2300	AUFKOMM
000510004	ER ZEUG T	1300	ERZEUG
000510012	DIEN T	1300	DIEN

000520008	SIND	3300	SEIN
000520016	LIEG T	2300	LIEG
000520028	IST	3300	SEIN
000520035	ENT FERN T	1300	ENTFERN
000520036	WERD EN	3100	WERD
000520037	HU*SS EN	4100	HU*SS
000530003	SOLL TE	4400	SOLL
000530006	VER URSACH EN	1100	VERURSACH
000530017	BRAUCH EN	1100	BRAUCH
000540002	HU*S* TE	4A00	HU*SS
000540005	SEIN	3100	SEIN
000540015	INSTALLIER EN	5100	INSTALLIER
000540031	VER BUND ENEN	2700	VERBIND
000540033	VER MEID ET	2300	VERMEID
000550007	ZU SAMMEN SETZ EN	1100	ZUSAMMENSETZ
000550008	LASS EN	2100	LASS
000550010	KANN	4300	KO*NN
000550019	PRODUZIER EN	5100	PRODUZIER
000560006	KONSTRUIER TEN	5400	KONSTRUIER
000560008	KANN	4300	KO*NN
000560015	RICHT EN	1100	RICHT
000560019	WACHS EN	2100	WACHS
000570002	VERMINDER T	1300	VERMINDER
000580003	SOLL TE	4400	SOLL
000580007	SEIN	3100	SEIN
000580010	AB GE STUF TES	1700	ABSTUF
000580014	AUF (ZU) BAU EN	1200	AUFBAU
000580028	KO*NN TEN	4A00	KO*NN
000580034	STEH EN	2100	STEH
000580038	ARBEIT EN	1100	ARBEIT
000580043	ER ZEUG EN	1100	ERZEUG
000590002	WU*RD EN	3A00	WERD
000590009	VER BRAUCH EN	1100	VERBRAUCH
000590015	BE NO*TIG T	1300	BENO*TIG
000590016	WIRD	3300	WERD
000590021	SOLL TE	4400	SOLL
000590025	BE TRAG EN	2100	BETRAG
000600010	WA*R EN	3A00	SEIN
000600017	INSTALLIER EN	5100	INSTALLIER
000600023	AUF (ZU) FANG EN	2200	AUFFANG
000610006	WA*R E	3A00	SEIN
000610015	VER WEND EN	1100	VERWEND
000610017	LIES* EN	2400	LASS
000610024	BE ZOG EN	2400	BEZIEH
000610026	VER BRAUCH TEN	1400	VERBRAUCH
000610029	ER REICH EN	1100	ERREICH
000620005	WIRD	3300	WERD
000620010	KOMBINIER EN	5100	KOMBINIER
000620022	KOMM*EN	2100	KOMM
000620030	IST	3300	SEIN
000630001	BE ZOG EN	2400	BEZIEH
000630006	VER BRAUCH TEN	1400	VERBRAUCH
000630008	SIND	3300	SEIN
000630015	ER WART EN	1100	ERWART
000640006	ARBEIT ENDE	1500	ARBEIT
000640008	IST	3300	SEIN
000640011	FORT GE SCHRITT EN	2800	FORTSCHREIT

000650002	BE STEH T	2300	BESTEH
000650010	KONZENTRIER TER	5700	KONZENTRIER
000650012	GE TRA*NK TEN	1700	TRA*NK
000650020	LIEG T	2300	LIEG
000660002	ENT HALT EN	2130	ENTHALT
000670003	ARBEIT ET	1300	ARBEIT
000670011	ER ZEUG T	1300	ERZEUG
000670018	HAT	3300	HAB
000680002	SCHA*TZ T	1300	SCHA*TZ
000680011	BE STEH EN	2130	BESTEH
000680022	PRODUZIER T	5300	PRODUZIER
000680023	WERD EN	3130	WERD
000680024	KO*NN EN	4130	KO*NN
000680037	BE LA*UF T	2300	BELAUf
000690003	WA*R E	3A00	SEIN
000700005	HAB EN	3130	HAB
000700011	ER REICH T	1300	ERREICH
000700022	GEH ENDE	2500	GEH
000700027	STEH T	2300	STEH
000710012	ARBEIT EN	1130	ARBEIT
000710014	IST	3300	SEIN
000720005	GE SCHMOLZ ENEN	2700	SCHMELZ
000720010	LIEG T	2300	LIEG
000730003	BE TRA*G T	2300	BETRAG
000730012	VER LAUF EN	2130	VERLAUF
000730025	BRAUCH T	1300	BRAUCH
000740004	BILD ET	1300	BILD
000740017	WIRK T	1300	WIRK
000740022	UN VER A*NDER T	1800	VERA*NDER
000740023	BLEIB T	2300	BLEIB
000750002	ER ZEUG EN	1130	ERZEUG
000750009	HAB EN	3130	HAB
000760009	SIND	3300	SEIN
000770004	HAT	3300	HAB
000770022	BE STEH ENDER	2500	BESTEH
000770024	ARBEIT ET	1300	ARBEIT
000780003	ERMUTIG ENDEN	1500	ERMUTIG
000780005	GIB T	2300	GEB
000790002	BE STEH EN	2130	BESTEH
000790014	BE GRENZ EN	1130	BEGRENZ
000800003	KOMM EN	2130	KOMM
000810016	SIND	3300	SEIN
000810019	ENT WICKEL T	1300	ENTWICKEL
000810030	EIN GE SETZ T	1800	EINSETZ
000810031	WERD EN	3130	WERD
000810032	KO*NN EN	4130	KO*NN
000820003	KO*NN EN	4130	KO*NN
000820009	BE ZEICHN ETE	1470	BEZEICHN
000820017	SIED ENDE	2500	SIED
000820025	VER ARBEIT EN	1130	VERARBEIT
000830002	BE MU*H T	1300	REMU*H
000830010	ENT WICKEL N	1130	ENTWICKEL
000830023	BE TRIEB EN	2480	BETREIB
000830024	WERD EN	3130	WERD
000830025	KO*NN EN	4130	KO*NN
000840005	(GE)BOT ENEN	2700	(GE)BIET
000840007	WERD EN	3130	WERD
000840010	NUTZ EN	1130	NUTZ
000840011	LASS EN	2130	LASS
000840017	PRODUZIER T	5300	PRODUZIER
000840018	WERD EN	3130	WERD

000850002	LIEG T	2300	LIEG
000850005	ENT SCHEID ENDE	2500	ENTSCHEID
000850012	WERD EN	3130	WERD
000850015	SEIN	3100	
000850020	WIRD	3300	WERD
000850025	SINK EN	2130	SINK
000850031	WA*CHS T	2300	WACHS
000860016	GIB T	2300	GEB
000860020	UNTER STU*TZ EN	1130	UNTERSTU*TZ
000860022	WERD EN	3130	WERD
000860027	SEIN	3100	SEIN
000860031	EIN (ZU) LEIT EN	1200	EINLEIT
000860033	FORT (ZU) SETZ EN	1200	FORTSETZ
000860039	LIEG T	2300	LIEG
000860044	GESTATT ET	1300	GESTATT
000860048	SPAR EN	1130	SPAR
000860055	VERHINDER N	1130	VERHINDER
000870004	GIB T	2300	GEB
000870014	BE STEH T	2300	BESTEH
000870023	ENT WICKEL N	1130	ENTWICKEL
000870032	WIRD	3300	WERD
000880004	LA*UF T	2300	LAUF
000890002	BE STEH T	2300	BESTEH
000890006	HAT	3300	HAB
000890013	BE TRIEB ENEN	2700	BETREIB
000890015	AN GE KOMM EN	2000	ANNEHM
000900004	IST	3300	SEIN
000900016	INSTALLIER T	5300	INSTALLIER
000900017	WERD EN	3130	WERD
000900018	KO*NN EN	4130	KO*NN
000910010	VER WEND ET	1300	VERWEND
000910011	WIRD	3300	WERD
000910013	IST	3300	SEIN
000920004	WURD EN	3400	WERD
000920012	AUSPROBIER T	5300	AUSPROBIER
000930004	GEWONN ENEN	2700	GEWINN
000930006	BAU TE	1400	BAU
000930020	ER PROB T	1300	ERPROB
000930021	WERD EN	3130	WERD
000930027	KOMM EN	2130	KOMM
000930028	SOLL	4300	SOLL
000940003	HAT	3300	HAB
000940011	AUS GE GEB EN	2000	AUSGEB
000950004	WURD E	3400	WERD
000950020	GE RUF EN	2000	RUF
000960002	BE SCHA*FTIG T	1300	BESCHA*FTIG
000960010	BE STEH EN	2130	BESTEH
000970004	ZEIG TE	1400	ZEIG
000970018	BE TRIEB ENES	2700	BETREIB
000970020	GE EIGN ET	1000	EIGN
000970021	IST	3300	SEIN
000970027	ARBEIT EN	1130	ARBEIT
000970038	GENU*G T	1300	GENU*G
000980003	WILL	4300	WOLL
000980021	STEH EN	2130	STEH
000980022	KO*NN TE	4000	KO*NN
000980026	ER RICHT EN	1130	ERRICHT
000980035	SCHA*TZ T	1300	SCHA*TZ

000990011	GE TRAG EN	2800	TRAG
000990012	WURD E	3400	WERD
000990022	BE TEILIG T	1300	BETEILIG
000990024	WIDM ET	1300	WIDM
000990036	ARBEIT EN	1130	ARBEIT
001000005	BE STEH ENDES	2500	BESTEH
001000008	SOLL	4300	SOLL
001000013	VER WEND EN	1130	VERWEND
001000014	KO*NN EN	4130	KO*NN
001000018	WIRD	3300	WERD
001000022	BE TRAG EN	2130	BETRAG
001000026	RECHN ET	1300	RECHN
001010006	WIRD	3300	WERD
001010012	GE ARBEIT ET	1800	ARBEIT
001010025	KOMM EN	2130	KOMM
001020013	AUTOMATISIER TE	5470	AUTOMATISIER
001020026	IMPLANTIER T	5300	IMPLANTIER
001020027	WERD EN	3130	WERD
001020028	KO*NN EN	4130	KO*NN
001020045	ARBEIT EN	1130	ARBEIT
001020047	STEH EN	2130	STEH
001030011	SOLL	4300	SOLL
001030022	INSTALLIER T	5300	INSTALLIER
001030023	WERD EN	3130	WERD
001040005	IST	3300	SEIN
001040015	ER ZEUG T	1300	ERZEUG
001040019	LEIST ET	1300	LEIST
001050003	ARBEIT EN	1130	ARBEIT
001050009	VER WEND EN	1130	VERWEND
001050021	BE ZEICHN ETEN	1470	BEZEICHN
001050029	SIED ENDEN	2500	SIED
001050031	ER ZEUG T	1300	ERZEUG
001050032	WIRD	3300	WERD
001060003	DIEN T	1300	DIEN
001070006	ZEIG T	1300	ZEIG
001080003	WANDEL N	1130	WANDEL
001090002	EMITTIER EN	5130	EMITTIER
001090008	HAB EN	3130	HAB
001090014	NUTZ T	1300	NUTZ
001100002	WIRD	3300	WERD
001100003	UNTER SUCH T	1300	UNTERSUCH
001100012	ARBEIT EN	1130	ARBEIT
001120003	DIEN T	1300	DIEN
001120009	ER ZEUG TES	1700	ERZEUG
001130004	WERD EN	3130	WERD
001130012	GE LAD ENE	2700	LAD
001130019	GE LAD ENE	2700	LAD
001130024	AUF GE SPALT EN	1800	AUFSPALT
001140003	WERD EN	3130	WERD
001140008	TRANSPORTIER T	5300	TRANSPORTIER
001150003	FLIES* EN	2130	FLIES*
001160002	VEREINIG EN	1130	VEREINIG
001160011	DIEN ENDEN	1500	DIEN
001160024	VER LA*S* T	2300	VERLASS
001170017	BE STEH T	2300	BESTEH
001230001	STEH T	2300	STEH
001230013	KANN	4300	KO*NN
001230021	ARBEIT EN	1130	ARBEIT
001230023	MUS*	4300	HU*SS
001230036	ER ZEUG EN	1130	ERZEUG

001240003	LIEFER N	1130	LIEFER
001240012	UM GE WANDEL T	1800	UMWANDEL
001240013	WERD EN	3130	WERD
001240014	MUS*	4300	MU*SS
001240022	AB GE GEB EN	2800	ABGER
001240023	WIRD	3300	WERD
001270001	WASSER	1300	WASSER
001320014	UM WANDEL N	1130	UMWANDEL
001320016	NUTZ EN	1130	NUTZ
001320020	BESSER	1300	BESSER
001320035	AN GE GEB ENEN	2700	ANGEB
001320038	ARBEIT EN	1130	ARBEIT
001330002	UNTER LIEG T	2300	UNTERLIEG
001340010	VERLANG T	1380	VERLANG
001340011	WIRD	3300	WERD
001340013	BRAUCH T	1300	BRAUCH
001340023	ER ZEUG EN	1130	ERZEUG
001340032	U*BER STEIG T	2300	U*BERSTEIG
001350002	ER WEIS EN	2130	ERWEIS
001360003	ERMÖ*GLICH T	1380	ERMÖ*GLICH
001360013	ARBEIT EN	1130	ARBEIT
001360023	DECK EN	1130	DECK
001430001	VER BESSER TE	1470	VERBESSER
001450019	SIND	3300	SEIN
001450032	GEGEN U*BER GE STELL T	1800	GEGENU*BERSTELL
001460002	AUS GE ZOG ENEN	2700	AUSZIEH
001460004	GEB EN	2130	GEB
001460008	GE STRICHEL TEN	1700	STRICHEL
001460010	GE SCHA*TZ TE	1700	SCHA*TZ
001470004	WURD E	3400	WERD
001470013	GE BAU T	1800	BAU
001470015	KOST ETE	1400	KOST
001480002	SOLL TEN	4400	SOLL
001480011	LIEG EN	2130	LIEG
001480027	PRODUZIER T	5300	PRODUZIER
001480028	WERD EN	3130	WERD
001480029	KO*NN EN	4130	KO*NN
001490003	SETZ T	1300	SETZ
001490015	GE WORD EN	3800	WERD
001490016	IST	3300	SEIN
001490023	AUF NEHM EN	2130	AUFNEHM
001490024	KANN	4300	KO*NN

Le programme .VERBAL2, dont nous venons de survoler les grands axes, donne lieu à deux sorties distinctes, .TRANSIT3 avec les résultats intermédiaires et .TRANSIT6 qui correspond au fichier d'entrée de l'analyse verbale .TRANSIT2 (mots commençant par une minuscule et non codés par PROLOG (mots outils simples ou disjoints) débarrassé des formes verbales.

En résumé :

<b>PROLOG</b> agit sur	<b>.TEXALL1.SORTIEØ.BASE</b>	Texte à analyser
en utilisant	<b>.FIXALL(EXPRALL) .FIXALL(OUTALL)</b>	Mots outils disjoints Mots outils simples
résultat :	<b>.TRANSIT1</b>	Texte complet Codage des mots outils Codage de la partie numérale
	<b>.TRANSIT2</b>	Mots débutant par minuscule et non codés et premiers mots de phrase
	<b>.TRANSIT5</b>	Mots débutant par majuscule, sans les premiers mots de phrase
<b>VERBAL</b> agit sur	<b>.TRANSIT2</b>	
en utilisant	<b>.MORPHAL(ENDUNG) .MORPHAL(PREFI) .MORPHAL(VERBRG) .MORPHAL(VERBIR) .MORPHAL(DES)</b>	Terminaisons verbales Préverbes + Collisions Racines "régulières" Racines "fortes" Désinences
résultat :	<b>.TRANSIT3</b>	Résultats intermédiaires Codage
	<b>.TRANSIT6</b>	Mots, en minuscule, non codés
<b>MELANGE</b> utilise	<b>.TRANSIT1 .TRANSIT3</b>	
résultat :	<b>.TRANSIT7</b>	Texte complet avec codage - Mots outils - Partie numérale - Verbe

### 4.3.2.3 La majuscule en début de phrase (DESAMB)

- Ne sont pas encore codés, à ce stade du traitement, les mots en tête de phrase et les adjectifs/adverbes.

Le but du présent module sera de désambiguïser la majuscule de tête en distinguant les noms et les adjectifs/adverbes, puis de nettoyer ces derniers afin de dégager un fichier de travail pour le module ADV.

- Nous disposons à cet effet du texte complet avec les mots-outils, la partie numérale et le verbe codés :

#### .TRANSIT7 :

```
*0015
0017
0017
0000
0000
0001 000
0002 005 014
0002 007 016
0000
00015004240001    $SAURE
00015004250002    $00 $ELEKTROLYTE
00015004260003    5130 TRANSPORTIEREN
00015004270004    $00 $WASSERSTOFF-$IONEN
00015004280005    (
00015004290006    $00 $H+
00015004300007    )
00015004310008    ,
00015004320009    P00 IN
00015004330010    ALKALISCHEN
00015004340011    $00 $ELEKTROLYTEN
00015004350012    1130 WANDERN
00015004360013    $00 $HYDROXID-$IONEN
00015004370014    (
00015004380015    $00 $O$:(-- )
00015004390016    )
00015004400017    .
```

- de .TRANSITS issu de PROLOG, avec tous les mots à majuscule sauf les mots en tête de phrase, autrement dit, tous les mots simples pour lesquels il n'y a aucun doute sur la nature de la majuscule.

.TRANSITS :

...TRANSITS

00002000090003 \$JAHRZEHNT  
 00002000140000 \$ELEKTRIZITA\*TSWERK  
 00002000210015 \$WIRKUNGSGRAD  
 00002000260020 \$SCHNUTZ  
 00002000300024 \$LA\*RM  
 00003000410004 \$KANDIDAT  
 00003000440007 \$BRENNSTOFFZELLE  
 00004000480002 \$BRENNSTOFFZELLE  
 00004000520006 \$ENERGIE  
 00004000540008 \$BRENNSTOFFS  
 00004000600014 \$UNWEG  
 00004000630017 \$WA\*RME  
 00004000660020 \$ELEKTRIZITA\*T  
 00005000720004 \$SIR  
 00005000730005 \$WILLIAM  
 00005000740006 \$GROVE  
 00005000780010 \$JURIST  
 00005000810013 \$GERA\*T  
 00005000920024 \$RECHT  
 00006000980004 \$GENINI-  
 00006001000006 \$APOLLO-\$RAUMSCHIFFE  
 00006001020008 \$ENERGIE  
 00006001110017 \$ANWENDUNG  
 00007001160004 \$BRENNSTOFFZELLEN  
 00007001220010 \$VERSIONEN  
 00007001310019 \$STADIUM  
 00007001410029 \$BEITRAG  
 00007001440032 \$ELEKTRIZITA\*TSVERSORGUNG  
 00008001500002 \$BRENNSTOFFZELLE  
 00008001520004 \$BILD  
 00008001500010 \$ELEKTRODEN

- de .TRANSIT6 issu de VERBAL, avec tous les mots à minuscule qui ne sont pas codés et qui ne sont donc ni mot-outil ni verbe.

.TRANSIT6 :

00002000000002 NA\*CHSTEN  
 00002000200014 HOHEN  
 00002000330027 RASCH  
 00004000510005 CHEMISCHE  
 00004000550009 DIREKT  
 00005000770009 ENGLISCHER  
 00006001000014 EXOTISCHE  
 00006001100016 TEURE  
 00007001210009 GRO\*S\*EREN  
 00007001400028 NENNENSWERTEN  
 00007001430031 O\*FFENTLICHEN  
 00008001610013 POSITIVEN  
 00008001680020 NEGATIVEN  
 00009001870006 POSITIV  
 00009001890008 NEGATIV  
 00009002030022 NEGATIV  
 00009002120031 METALLISCHEN  
 00009002140033 VERANTWORTLICH  
 00011002620019 E  
 00012002690005 FORD\*S  
 00013003430024 NORMALEN  
 00013003460027 VOLLSTA\*NDIG  
 00013003720053 ELEKTRISCHEN  
 00014003970008 CHEMISCHEN  
 00015004330010 ALKALISCHEN  
 00016004650025 FESTES  
 00017004790011 GLEICHFALLS  
 00017004840016 CHEMISCHEN  
 00018005030009 THEORETISCH  
 00018005080014 NORMALEM  
 00018005110017 NORMALER  
 00020005590022 CHEMISCHEN  
 00021005740002 EINFACHE  
 00024006670005 EINFACHE  
 00024006710009 TECHNISCHE  
 00024006800018 KLEIN

Un ensemble de 7 tests permet de résoudre tous les cas que nous avons rencontrés jusqu'ici. Ils sont de 2 types.

Les uns utilisent la mémoire du texte et sont d'autant plus performants que le texte est long, les autres sont liés à l'improbabilité de certaines distributions.

.TRANSIT8 reprend le texte complet et y incorpore les résultats.

.TRANSIT9 ne contient que les résultats.

Test 1 : Comparaison du premier mot de la phrase avec les éléments de .TRANSIT5  
 Maintien de la majuscule en cas d'identité. (La comparaison s'effectue à la dernière lettre près).

.TRANSIT8 :

```
*0006
0018
0018
0000
0000
0000
0000
0000
0000
0000
000060001.N001  $00 $BRENNSTOFFZELLEN
.V002000200000 1470 VERSORGTE
0000600030N003  IVK DIE
00006000400000  $00 $GEMINI-
00006000500000  M0+ UND
000060006*N003  $00 $APOLLO-$RAUMSCHIFFE
0000600070P007  P00.MIT
000060008*P007  $00.$ENERGIE
00006000900000  M01 UND
.V010001000000 2400 FANDEN
00006001100000  B00 DAMIT
0000600120N012  JS0 EINE
0000600130000*  L00 EBENS0
0000600140000*  Z00 EXOTISCH0E
0000600150000*  L00 WIE
0000600160000*  Z00 TEUER0E
000060017*N012  $00 $ANWENDUNG
00006001800000  .
```

## Test 2 : Comparaison à la dernière lettre près du premier mot de la phrase avec les éléments de .TRANSIT6

Retrait de la majuscule en cas d'identité.

```

*0015
0017
0017
0000
0000
0001 000
0002 005 014
0002 007 016
0000
0001500010N001 Z00 SAUERSE
000150002+N001 000 $ELEKTROLYTE
.V0003000300000 5130 TRANSPORTIEREN
000150004.N004 000 $WASSERSTOFF-$IONEN
000150005+0000 (
000150006.N006 000 $H+
00015000700000 )
00015000800000 ,01,
00015000900000 F00.IN
00015001000000 Z00 .ALKALISCHEN
000150011+P000 000 .$ELEKTROLYTEN
.V0012001200000 1130 WANDERN
000150013.N013 000 $HYDROXID-$IONEN
000150014+0000 (
000150015.N015 000 $OH(--
00015001600000 )
00015001700000 .

*0070
0030
0030
0000
0000
0001 012
0001 012
0001 021
0000
0007000010N001 Z00 KLEINSE
000700002+N001 000 $STAPEL
0007000030P003 P00.VON
000700004+P003 000 .$PHOSPHORSA+URE-$ZELLEN
00050000500000 3130 HABEN
00070000600000 A00 HEUTE
00070000700000 A00 SCHON
00070000800000 Z00 VIELSE
00070000900000 C01 TAUSEND
000700010+N000 000 $BETRIEBSSTUNDEN
.V0005001100000 1330 ERREICHT
00070001200000 ,01+,
00070001300000 M01 ABER
0007000140N014 I0K DIE
0007000150P015 P00.U+BER
00070001600000 C01 .VIERZIGTAUSEND
00070001700000 000 .0STUNDEN
000700018+0000 .(
00070001900000 C01 .VIEREINHALB
00070002000000 000 .0JAHRE
000700021+P015 .)
00070002200000 Z100 GEHENDSE
00070002300000 000 $DAUERPRU+FUNG
00070002400000 JS0 EINES
00070002500000 Z00 GROS+SEN
000700026+N014 000 $STAPELS
00070002700000 2300 STEHT
00070002800000 AM+ NOCH
.V0027002900000 000 AUS
00070003000000 .

```

Test 3 : Lorsqu'une phrase se réduit à un mot, il s'agit d'un mot majuscule

```
00001000000000
00001000000000  ::
000010001.N001  000 00RENNSTOFFZELLENKRAFTWERKE
00001000000000  ::
00001000200000  .
```

Test 4 : Si le mot est un mot composé par tiret, il prend le profil du dernier segment

```
*0075
0010
0010
0000
0000
0000
0000
0000
0000
000750001.N001  000 00CARBONAT-0ZELLEN
.V000000200000  1130 ERZEUGEN
00075000300000  JS0 EINE
000750004.N000  000 00SPANNUNG
00075000500000  P00.VON
00075000600000  C00.00,700
000750007.P000  000.0VOLT
00075000800000  M01 UND
.V000000000000  3130 HABEN
000750010.N010  000 00LEISTUNGSDICHTEN
0007500110P011  P00.ZWISCHEN
00075001200000  C00.01.1
00075001300000  M0+.UND
00075001400000  C00.01.3
000750015.P011  000.0KILOWATT
0007500160P016  P00.PRO
000750017.P016  000.0QUADRATHETER
00075001800000  .
```

Test 5 : Si le mot est suivi d'une conjonction de coordination et d'un mot à majuscule, il conserve la majuscule.

\*0102  
 0053  
 0053  
 0050  
 0050  
 0053 012 020 046  
 0052 010 029  
 0052 015 036  
 0050  
 001020001.N001 000 \$NOTSTROMAGGREGATE  
 001020002000000 N0+ UND  
 001020003.N003 000 \$GERATE  
 001020004000000 P00.FU+R  
 001020005000000 I00.DIE  
 001020006000000 Z00.ELEKTRISCHE  
 001020007000000 \$00.\$VERSORGUNG  
 001020008000000 AZI.KLEINERN  
 001020009000000 \$00.\$EINHEITEN  
 001020010+00000 .(  
 001020011000000 \$00.\$GABELSTAPLER  
 001020012000000 .,  
 001020013000000 SZ00.AUTOMATISIERTE  
 001020014000000 \$00.\$RES+STATIONEN  
 001020015+P004 .)  
 00102001600016 P00.BIS  
 001020017+P016 A00.HINAB  
 00102001800218 P00.ZU  
 001020019+P018 \$00.\$SPANNUNGSQUELLEN  
 001020020000000 ,01,  
 00102002100021 K00.DIE  
 00102002200022 P00.IN  
 001020023000000 I00.DEN  
 001020024000000 Z00.MENSCHLICHEN  
 001020025+P022 \$00.\$KO+RPER  
 0026002600021 5300 IMPLANTIERT  
 0026002700021 3130 WERDEN  
 \*0026002800021 4130 KO+NEN  
 001020029+00000 (  
 00102003000000 A00 BEISPIELSWEISE  
 00102003100031 P00.FU+R  
 001020032000000 I00.DEN  
 001020033000000 \$00.\$ANTRIEB  
 001020034000000 J00.EINES  
 001020035+P031 \$00.\$HERZSCHRITTMACHERS  
 001020036000000 )  
 001020037000000 N01 UND  
 00102003800038 P00.MIT  
 001020039+P038 \$00.\$TRAUBENZUCKER  
 00102004000040 P00.AUS  
 001020041000000 I00.DEN  
 001020042000000 \$00.\$BLUT  
 00102004300043 \*01.ALS  
 001020044+P040 \$00.\$BRENNSTOFF  
 .00450045+R021 1130 ARBEITEN  
 001020046000000 ,01,  
 .00470047+00000 2130 STEHEN  
 001020048000000 A00 HIER  
 00102004900049 P00.IM  
 001020050000000 \$00.\$VORDERGRUND  
 001020051000000 I00.DES  
 001020052+P049 \$00.\$INTERESSES  
 001020053000000 .

Test 6 : Si le mot précède un mot à majuscule suivi d'un verbe, il perd la majuscule (Ce n'est pas sûr, mais assez probable !).

\*0000  
0014  
0014  
0000  
0000  
0000  
0000  
0000  
0000  
0000  
0000  
0000000100001 Z00 GUNSTIGEE  
00000002\*0001 000 0SCHN+TZUNGEN  
.0000000300000 2100 KOMMEN  
0000000400004 P00.FUR  
0000000500000 I00.DIE  
00000006\*0004 000.0CARBONAT-0ZELLE  
0000000700007 P00.AUF  
0000000800000 Z00.A+ENLICHEE  
00000009\*0007 000.0PRODUKTIONSKOSTEN  
0000001000000 DL0 WIE  
0000001100011 P00.FUR  
0000001200000 I00.DIE  
00000013\*0011 000.0PHOSPHORSA+URE-0ZELLE  
0000001400000

Test 7 : Dans tous les autres cas que ceux que nous venons de citer, le mot conservera sa majuscule.

#### 4.3.2.4 Les adjectifs et les adverbes

Dans le prolongement du module **VERBAL** et après la désambiguïisation de la majuscule en début de phrase, le codage des adjectifs et des adverbes marque la fin du niveau Ø.

Le module **ADV** doit afficher un code qui facilite le repérage des groupes nominaux tout en respectant certaines contraintes. Il est en effet hors de question d'utiliser un dictionnaire, encore moins de parcourir la phrase pour sonder le contexte.

Si l'analyse effectuée dans ces conditions ne peut être que superficielle, elle doit cependant fournir tous les éléments qui permettront de l'affiner en cas de besoin.

Nous partirons du principe que le champ des investigations recouvre l'adjectif qualificatif épithète, l'adjectif qualificatif attribut et l'adverbe, les deux derniers ayant un statut flexionnel identique.

L'examen des formes testées, par la droite et à l'aide de petits tableaux, grâce à un programme de 370 lignes, s'applique à **.TRANSIT8**, texte complet avec majuscules de début de phrase désambiguïisées et codage de tous les enregistrements autres que les adjectifs qualificatifs et les adverbes absents de **.FIXALL(OUTALL)** et **.FIXALL(EXPRALL)**.

#### .TRANSIT8

```

52015 660001 P00 MIT
52015 670002 000 $0=L
52015 680003 N0+ ODER
52015 690004 000 $KOHLE
52015 700005 *01 ALS
52015 710006 PRIMA*REN
52015 720007 000 $BRENNSTOFFEN
52015 730008 3300 SIND
52015 740009 IVK DAS
52015 750010 F00 #VOR#ALLEN
52015 760011 000 $SCHWEFELDIOXID
52015 770012 N0+ UND
52015 780013 000 $STICKSTOFFOXID
52015 790014 ,
52015 800015 AN0 DOCH
52015 810016 2300 LIEGT
52015 820017 IVK DIE
52015 830018 000 $SCHWEFELDIOXID-GERISSION
52015 840019 A00 BESONDERS
52015 850020 NIEDRIG
52015 860021 ,
52015 870022 N00 WEIL
52015 880023 IVK DIE
52015 890024 000 $BRENNSTOFFZELLE
52015 900025 P00 GEGEN
52015 910026 000 $SCHWEFELVERBINDUNGEN
52015 920027 EMPFINDLICH
52015 930028 3300 IST
52015 940029 ,
52015 950030 N00 #SO#DAS+
52015 960031 V00 DIESE
52015 970032 F00 IN
52015 980033 IVK DER
52015 990034 000 $BRENNSTOFF-$AUFBEREITUNGSANLAGE
52016 000035 1300 ENTFERNT
52016 010036 3130 WERDEN
52016 020037 4130 MU*SSEN
52016 030038 .

```

Dans le tableau qui suit, nous résumerons les problèmes à résoudre. Ces derniers sont d'ordre morphologique, étant donné le niveau d'analyse (niveau  $\emptyset$ ).

L'adjectif qualificatif attribut et l'adverbe qualificatif portant une désinence identique et de ce fait un code ambigu (AZ $\emptyset$ ), nous attribuerons le code (A $\emptyset\emptyset$ ) aux adverbes repérés grâce à leur suffixation (*falls, lings...*). L'analyse s'applique aux participes (présent et passé) déclinés.

POSITIF	Adj. épithète Z $\emptyset\emptyset$	Désinences : -e, -em, -en, -es, -er  Particularités : syncope du e pour certains radicaux <i>hoh</i> -e de <i>lange</i> (adv)
	Adj. attribut Adverbe qual. AZ $\emptyset$	Désinence : $\emptyset$
COMPARATIF SUPERIORITE	Adj. épithète Z $\emptyset$ 1	Désinences : (")ere, (")erem, (")eren, (")eres, (")erer  Particularités : GUT, WOHL, HOCH VIEL, GERN, NAH, WENIG
	Adj. attribut Adverbe qual. AZ1	Désinence : (")er
SUPERLATIF	Adj. épithète Z $\emptyset$ 2	Désinences : -(e)ste, -(e)stem, -(e)sten, -(e)ster, -(e)stes  Particularités : GROß, GUT, WOHL HOCH, VIEL, NAH, GERN, WENIG
	Adj. attribut Adverbe qual. A $\emptyset$ 2	Désinences : am ... (")sten am ... (e)sten  Particularités : - irréguliers : hoch - -ßt, -st : möglichst - formes figées : <i>mindestens</i> , <i>meistens</i>

Le système des désinences n'enregistre qu'une collision :

adj. épithète (déclinaison forte)

-ER code AZ1

adj. attribut et adv. qualificatif au comparatif de sup.

Le code est à 3 caractères. Dans le cas d'un participe passé décliné (X7ØØ) ou d'un participe présent décliné (X5ØØ), il se superpose aux trois derniers caractères du code verbal.

AØØ	adverbe
AØ2	adverbe qual. et adj. attribut au superlatif
ZØØ	adj. qualificatif épithète
ZØ1	adj. qualificatif épithète au comparatif sup.
ZØ2	adj. qualificatif épithète au superlatif
AZØ	adj. qualificatif attribut/adverbe qualificatif
AZ1	adj. qualificatif épithète/adverbe qualificatif et adj. qualificatif attribut au comparatif de sup.

La comparaison de la partie finale du mot testé avec les données du fichier .FIXALL(DEGRE) ne peut pas se dérouler sans quelques précautions. Ces dernières répondent aux particularités énoncées dans le tableau de la page précédente et s'appuient sur trois tableaux:

.FIXALL(ADJER) pour les adjectifs en -ER,  
 .FIXALL(ADJST) pour les adjectifs en -(E)ST  
 .FIXALL(COMPSUP) pour les irrégularités

.FIXALL(DEGRE) :

```

00010 .FIXALL(DEGRE)
00020 30003
00030 0037
00040 0021
00050 0006
00060 A02ESTENS
00070 Z02ESTEN
00080 Z02ESTEN
00090 Z02ESTER
00100 Z02ESTES
00110 Z02ESTE
00120 A02EST

00140 Z02STEN
00150 Z02STEN
00160 Z02STER
00170 Z02STES
00180 Z02STE
00190 A02ST
00200 A02S*TENS
00210 Z02S*TEN
00220 Z02S*TEN
00230 Z02S*TER
00240 Z02S*TES
00250 Z02S*TE
00260 A02S*T
00270 Z01EREN
00280 Z01EREN
00290 Z01ERER
00300 Z01ERES
00310 Z01ERE
00320 AZ1ER

```

.FIXALL(ADJER) :

00010 .FIXALL(ADJER)  
00020 SEBST  
00025 0015  
00030 AND  
00040 NIED  
00050 OR  
00060 UNT  
00070 LOCK  
00080 LE  
00090 TAPP  
00100 TEU  
00110 SAU  
00120 MAG  
00130 MAG  
00140 STI  
00150 INTEG  
00160 SICH  
00170 WACK

.FIXALL(ADJST) :

00010 .FIXALL(ADJST)  
00020 SELBST  
00030 MODEST  
00040 FEST  
00050 MEIST  
00060 LIST  
00070 MIST  
00080 NIST  
00090 OIST  
00100 PIST  
00110 RIST  
00120 TIST  
00130 VIST

.FIXALL(COMPSUP) :

00010 .FIXALL(COMPSUP)  
00020 BESS  
00030 GRO\*S\*  
00040 HO\*H

La présence des suffixes adverbiaux permet d'éviter l'insertion dans le dictionnaire des mots outils d'un grand nombre d'unités qu'il est facile de repérer.

.FIXALL(ADVER) :

00010 .FIXALL(ADVER)  
00020 30003  
00030 0011  
00040 FALLS  
00050 LINGS  
00060 RAS\*EN  
00070 SEITS  
00080 TEILS  
00090 WA\*RTS  
00100 WEGS  
00110 WEISE  
00120 NALS  
00130 HAL  
00140 ENS

Obtenus en moins d'une seconde sur les 149 phrases du texte, les résultats sortent sur deux fichiers, .TRANSI10 qui ne comporte que les résultats et .TRANSI11 qui les a incorporés dans le texte. C'est ce fichier qui servira de départ à l'analyse syntaxique (niveau 2).

000100010000002 Z02 NAHSTREIEN  
 00012001200014 Z00 HOHEN  
 00022001200027 A00 RASCH  
 00010001000002 A20 VIEL  
 00010001000003 2A21 VERSPRECHENDEBERG  
 00010001000005 Z00 CHEMISCHE  
 00010001000009 AZ0 DIREKT  
 00010001000009 AZ1 ENGLISCHBERG  
 00010001000016 A00 LANGE  
 00010001000014 Z00 EXOTISCHE

00042012310024 2Z00 ENTHALTENDE  
 00042012370030 1AZ0 UNVERANDERT  
 00043012620018 Z00 OFFENTLICHE  
 00043012640020 2Z00 ANGESCHLOSSENEN  
 00044012710002 Z00 ERSTER  
 00044012900021 Z01 KLEINEREM  
 00044012990030 Z00 FEST

00050015000005 AZ0 WIRTSCHAFTLICH  
 00050015110000 Z00 NORMALER  
 00050015200017 Z00 DURCHSCHNITTLICHEN  
 00050015270024 AZ1 VOLLEREM  
 00050015400037 AZ0 LASTABHANGIG  
 00051015430014 Z00 UMWELTBELASTENDE  
 00052015710000 Z00 PRIMARER  
 00052015850020 AZ0 NIEDRIG  
 00052015920027 AZ0 EMPFINDLICH  
 00054016250004 AZ0 MOGLICH  
 00054016280007 Z00 SOLOER  
 00054016310010 AZ1 UNMITTELBARER  
 00054016420021 AZ1 NEUERER

.TRANSIT11 :

00002000060000 \$  
 00002000070001 000 IN  
 00002000080002 Z02 NAHESTEN  
 00002000090003 \$00 \$JAHRZEHT  
 00002000100004 3300 WIRD  
 00002000110005 \*05 MAN  
 00002000120006 P00 VON  
 00002000130007 JS0 EINEN  
 00002000140008 \$00 \$ELEKTRIZITA\*TSWERK  
 00002000150009 1130 FORDERN  
 00002000160010 ,  
 00002000170011 N00 DAS\*  
 00002000180012 Y00 ES  
 00002000190013 JS0 EINEN  
 00002000200014 Z00 HOHEN  
 00002000210015 \$00 \$WIRKUNGSGRAD  
 00002000220016 3300 HAT  
 00002000230017 ,  
 00002000240018 A02 NO\*GLICHST  
 00002000250019 A00 WENIG  
 00002000260020 \$00 \$SCHUTZ  
 00002000270021 5300 EMITTIERT  
 00002000280022 ,  
 00002000290023 000 KEINEN  
 00002000300024 \$00 \$LA\*RM  
 00002000310025 1300 VERURSACHT  
 00002000320026 M0+ UND  
 00002000330027 AZ0 RASCH  
 00002000340028 \*04 ZU  
 00002000350029 5130 INSTALLIEREN  
 00002000360030 3300 IST  
 00002000370031 .

00015004240001 Z00 SAUERDE  
 00015004250002 \$00 \$ELEKTROLYTE  
 00015004260003 5130 TRANSPORTIEREN  
 00015004270004 \$00 \$WASSERSTOFF-\$IONEN  
 00015004280005 (  
 00015004290006 \$00 \$H+  
 00015004300007 )  
 00015004310008 ,  
 00015004320009 P00 IN  
 00015004330010 Z00 ALKALISCHEN  
 00015004340011 \$00 \$ELEKTROLYTEN  
 00015004350012 1130 WANDERN  
 00015004360013 \$00 \$HYDROXID-\$IONEN  
 00015004370014 (  
 00015004380015 \$00 \$OH(--)  
 00015004390016 )  
 00015004400017 .

00100027580001 JS0 EIN  
 00100027590002 AP0 AUS  
 00100027600003 Z00 SOLCH0EN  
 00100027610004 \$00 \$ZELLEN  
 00100027620005 2Z00 BESTEHEND0ES  
 00100027630006 \$00 \$KRAFTWERK  
 00100027640007 ,  
 00100027650008 4300 SOLL  
 00100027660009 \$00 \$KOHLE  
 00100027670010 \*01 ALS  
 00100027680011 Z00 FRIMA\*ROEN  
 00100027690012 \$00 \$BRENNSTOFF  
 00100027700013 1130 VERWENDEN  
 00100027710014 4138 KO\*NNEN  
 00100027720015 ,  
 00100027730016 U00 SEINE  
 00100027740017 \$00 \$LEISTUNG  
 00100027750018 3300 WIRD  
 00100027760019 V00 EINIGE  
 00100027770020 C01 HUNDERT  
 00100027780021 \$00 \$MEGAWATT  
 00100027790022 2138 BETRAGEN  
 00100027800023 ,  
 00100027810024 N0+ UND  
 00100027820025 \*05 MAN  
 00100027830026 1300 RECHNET  
 00100027840027 P00 MIT  
 00100027850028 JS0 EINER  
 00100027860029 \$00 \$ENTWICKLUNGSZEIT  
 00100027870030 P00 VON  
 00100027880031 A00 ETWA  
 00100027890032 C01 ZEHN  
 00100027900033 \$00 \$JAHREN  
 00100027910034 .  
  
 00018004950001 JS0 EINE  
 00018004960002 P00 MIT  
 00018004970003 \$00 \$WASSERSTOFF  
 00018004980004 N0+ UND  
 00018004990005 \$00 \$SAUERSTOFF  
 00018005000006 2Z00 BETRIEBEN0E  
 00018005010007 \$00 \$ZELLE  
 00018005020008 3300 IST  
 00018005030009 AZ0 THEORETISCH  
 00018005040010 P00 IN  
 00018005050011 IVK DER  
 00018005060012 \$00 \$LAGE  
 00018005070013 PA0 BEI  
 00018005080014 Z00 NORMAL0EN  
 00018005090015 \$00 \$DRUCK  
 00018005100016 N0+ UND  
 00018005110017 AZ1 NORMAL0ER#  
 00018005120018 \$00 \$TEMPERATUR  
 00018005130019 JS0 EINE  
 00018005140020 \$00 \$SPANNUNG  
 00018005150021 P00 VON  
 00018005160022 AZ0 1,23  
 00018005170023 \$00 \$VOLT  
 00018005180024 \*04 ZU  
 00018005190025 1130 LIEFERN  
 00018005200026 .

00007001130001 A00 HEUTE  
 00007001140002 2300 GIBT  
 00007001150003 Y00 ES  
 00007001160004 \$00 \$BRENNSTOFFZELLEN  
 00007001170005 F00 IN  
 00007001180006 1470 VERBESSERTEN  
 00007001190007 M0+ UND  
 00007001200008 Z00 TAUSEND-MAL  
 00007001210009 Z01 GRO+S\*QER#EN  
 00007001220010 \$00 \$VERSIONEN  
 00007001230011 ,  
 00007001240012 M0+ UND  
 00007001250013 Y00 ES  
 00007001260014 2300 SCHEINT  
 00007001270015 ,  
 00007001280016 \*01 ALS  
 00007001290017 DA00 SEI  
 00007001300018 IVK DAS  
 00007001310019 \$00 \$STADIUM  
 00007001320020 1380 ERREICHT  
 00007001330021 ,  
 00007001340022 F00 IN  
 00007001350023 IVK DEM  
 00007001360024 \*05 MAN  
 00007001370025 F00 VON  
 00007001380026 Y00 IHNEN  
 00007001390027 JS0 EINEN  
 00007001400028 Z00 NENNENSWERT0EN  
 00007001410029 \$00 \$BEITRAG  
 00007001420030 600 ZUR  
 00007001430031 Z00 O\*FFENTLICH0EN  
 00007001440032 \$00 \$ELEKTRIZITA\*TSVERSORGUNG  
 00007001450033 1130 ERWARTEN  
 00007001460034 4300 KANN  
 00007001470035 .