La TAO comme technologie scientifique : le cas de la TA fondée sur le dialogue

Christian BOITET

GETA, IMAG
(Université Joseph Fourier & CNRS)
BP 53X, 38041 Grenoble Cedex, FRANCE
boitet@imag.fr

(pour le livre "Études et recherches en traductique")

Résumé

La TAO n'est ni une science, ni une collection de recettes, mais plutôt une technologie scientifique, c'est-à-dire un ensemble de méthodes qui progresse autant par l'intégration épisodique d'idées théoriques que par l'amélioration incrémentale des savoir-faire. Les fameux "paradigmes", trop souvent défendus de manière quasi-idéologique, ne peuvent être placés sur une seule échelle de valeurs, mais sont à comparer dans chaque type de situation traductionnelle. Enfin, chaque paradigme correspond à un assez grand nombre de choix fondamentaux sur l'architecture linguistique d'un système de TAO, et il reste de nombreuses voies intermédiaires à explorer. Ces idées peuvent être illustrées par le nouveau paradigme de la "TA fondée sur le dialogue", ou TAO personnelle pour auteur monolingue, que nous proposons pour le cas de situations traductionnelles où d'autres approches, telles que la TA fondée sur la langue, la TA fondée sur la connaissance, ou encore les aides à la traduction humaine, sont inadéquates.

Introduction

Les avis sur la traduction automatique (TA) sont souvent extrêmes. Les uns ne la conçoivent que comme l'expérimentation scientifique de leurs théories ou de leurs formalismes favorits, tandis que les autres y voient une entreprise purement technologique et utilitaire. Le plus souvent, chacun défend un "paradigme" particulier, comme on défendrait une idéologie. Par exemple, on soutient que le progrès ne pourrait venir qu'en construisant des systèmes munis d'ontologie, qui "comprennent" explicitement, alors que c'est rarement possible, et, quand ça l'est, rarement nécessaire. Ou encore, on affirme qu'il est nécessaire de passer par un interlingua pour construire des systèmes multilingues, alors que cette idée déjà ancienne est particulièrement difficile à mettre en œuvre, et que d'autres approches sont également possibles et nettement moins coûteuses, sauf dans les rares situations très fortement multilingues (au moins 8 langues, avec flux équilibrés pour toutes les paires [14, 34, 35]).

Notre thèse est que la TAO n'est ni une science, ni une collection de recettes, mais plutôt une technologie scientifique, c'est-à-dire un ensemble de méthodes qui progresse autant par l'intégration épisodique d'idées théoriques que par l'amélioration incrémentale des savoir-faire. D'autre part, les fameux "paradigmes" ne peuvent être placés sur une seule échelle de valeurs, mais sont à comparer dans chaque type de situation traductionnelle. Enfin, chaque paradigme correspond à un assez grand nombre de choix fondamentaux sur l'architecture linguistique d'un système de TAO, et il reste de nombreuses voies intermédiaires à explorer.

Cette thèse peut être illustrée par le paradigme de la "TA fondée sur le dialogue" (TAFD), ou TAO personnelle pour auteur monolingue, que nous proposons pour le cas de situations traductionnelles où d'autres approches, telles que la TA fondée sur la langue (TAFL), la TA fondée sur la connaissance (TAFC), et les aides informatisées au traducteur (THAM), sont inadéquates. Dans cette approche, bien que la base de connaissances linguistiques du système reste cruciale, et doive même être la plus couvrante possible, et que des connaissances extralinguistiques puissent éventuellement être utilisées si elles sont disponibles, l'accent est mis sur une préédition indirecte, effectuée grâce à un dialogue de normalisation et de clarification avec l'auteur, et destinée à obtenir des traductions de haute qualité sans révision.

Nous étudions diverses questions relatives à cette approche grâce à une maquette, LIDIA-1, première étape vers la construction d'un système de TAFD pour une situation particulière, celle d'auteurs monolingues rédigeant en hypertexte de la documentation technique devant être diffusée en plusieurs langues. Nous terminons en évoquant quelques problèmes intéressants et nouveaux apparus durant ce travail, et auxquels il faudrait au moins trouver des solutions partielles pour passer ultérieurement à un prototype.

I. TAO et IA : un système de TAO ne peut et ne doit souvent pas comprendre explicitement

Un système de traduction automatique doit-il et peut-il comprendre ? Si cette question est encore vivement controversée, 45 ans après les débuts de la TA, c'est parce que l'analyse se place à un niveau trop "angélique", trop anthropomorphique, et trop éloigné des réalités techniques.

On pose souvent comme acquis qu'un traducteur humain peut et doit comprendre pour bien traduire, et on prend comme étalons la qualité de ses traductions, supposée parfaite, et le degré de sa compréhension, supposé complet. Mais il faut s'interroger sur ces prémisses, car on peut aussi bien soutenir qu'on ne peut traduire sans comprendre que démontrer qu'à vouloir trop comprendre, on ne peut plus traduire. On doit alors admettre que, selon la qualité de traduction recherchée, le traducteur ou l'interprète *doit* plus ou moins comprendre, et qu'en fonction de sa formation antérieure, de l'information accessible et des délais imposés, il *peut* plus ou moins comprendre.

Poser cette question suppose aussi que les systèmes de TA soient faits pour "remplacer l'homme", c'est-à-dire pour fournir des traductions *équivalentes à celles que l'on demande aux humains.* Mais, en entrant un peu dans le détail des types d'automatisation de la traduction, on s'aperçoit que cette supposition est bien souvent erronée. Traduction humaine et TAO ne visent souvent pas la même chose, et la "compréhension" d'un système ne peut se définir ni se juger comme celle d'un humain. Pour cela, il convient d'étudier les différentes architectures envisageables pour les systèmes de TAO, et de déterminer le ou les "degrés de compréhension" qu'elles autorisent (compréhension explicite, apparente directe ou indirecte, implicite.

En examinant ce qui existe ou est réalisable actuellement, et ce qui reste du domaine des perspectives à plus long terme, on arrive alors à la conclusion suivante : dans certains cas, les systèmes de TAO peuvent comprendre au sens fort (explicitement), mais il n'est pas sûr qu'ils le doivent jamais, car des systèmes utilisant les autres degrés de compréhension peuvent souvent produire des traductions de qualité équivalente, pour un coût bien moindre.

I.1. Compréhension et traduction humaine

Très souvent, on entend des arguments du type : "un traducteur ne peut bien traduire que ce qu'il comprend" ou "les interprètes (simultanés) traduisent bien sans avoir le temps de comprendre". Qu'entend-on donc par "bien traduire" ? (Et ce n'est déjà pas le même "bien" dans les deux assertions précédentes !)

Bien traduire, c'est en premier lieu transmettre le contenu objectif d'un message (ce qui est dit d'une réalité externe, concrète ou abstraite — contenu propositionnel, et comment cela est dit — modalité, type de discours, situation de communication…). En second lieu, c'est aussi rendre ses aspects plus subjectifs (style, tonalité affective, environnement culturel, aspects esthétiques ou rhétoriques, intentions cachées…).

On emploie le terme de "traduction" aussi bien pour la poésie que pour les romans, les rapports et manuels techniques, et les nomenclatures de pièces détachées, alors qu'il conviendrait, au moins, de distinguer entre :

- la "re-création", par exemple la traduction d'Edgar Poe par Baudelaire, qui vise avant tout à transmettre l'aspect subjectif, fût-ce au prix d'une légère transformation du contenu ;
- la "localisation", largement pratiquée pour les manuels de micro-ordinateurs, qui vise à adapter un contenu à un environnement culturel particulier ;
- la "traduction-diffusion" [16], en particulier la traduction de documentations techniques dont le contenu doit être strictement rendu, sans ajout ni omission, même si le style "sent la traduction" ;
- la "traduction rapide" enfin, dans laquelle nous rangerons la "traduction-dépistage" de textes écrits et l'interprétation simultanée.

La même traduction pourra donc être jugée "bonne" en traduction rapide, et détestable en re-création. À l'évidence, le traducteur humain qui effectue la localisation d'un manuel informatique comprend plus profondément qu'un interprète qui traduit des interventions techniques sur la politique agricole commune.

1.1. Compréhension et traduction : le mieux est parfois l'ennemi du bien !

Il semble même, à l'expérience, qu'à chaque type de traduction corresponde un "degré de compréhension" (humaine) optimal, et en particulier que, comme en bien d'autres domaines, le mieux puisse ici être l'ennemi du bien.

Lors du colloque COLING-73, à Pise, on put en voir une démonstration éclatante. Un soviétique, le Pr. Andreev, devait présenter une communication en russe, et le Dr. Andreewski, d'origine russe et parfaitement à l'aise dans les deux langues, avait accepté d'effectuer une interprétation "différée" (traduction par morceaux de quelques minutes de parole). Le malheur voulut qu'il connût aussi parfaitement le sujet de l'exposé : avant de traduire le premier fragment, il interrogea l'orateur pour être sûr d'avoir bien compris un point de détail, et l'exposé se transforma rapidement en un dialogue en russe, d'où sortaient de temps à autre quelques phrases en français, hors contexte et donc sybillines… La traduction avait été tuée par la compréhension, alors qu'un interprète professionnel, totalement incompétent en linguistique quantitative et en statistique, aurait produit une traduction tout à fait acceptable après une étude convenable de la terminologie utilisée.

Un autre danger qui guette le traducteur est celui de chercher l'élégance en croyant comprendre, alors qu'il n'a pas la compétence nécessaire, et de commettre beaucoup alors plus de contresens que s'il se limitait à une traduction plus littérale. Ainsi, en 1977, nous avions rédigé une communication en français au colloque "Franchir la barrière linguistique" organisé par la CEE à Luxembourg. Comme il fallait la traduire en anglais et en allemand, deux chercheurs d'origine anglaise et allemande avaient envoyé toute la terminologie nécessaire dans les trois langues. Les deux premiers jets de traduction furent absolument incompréhensibles, les traducteurs n'ayant pu résister au désir de manifester qu'ils comprenaient ce qu'ils traduisaient, et d'améliorer les traductions proposées par nos collègues. Il fallut à peu près tout refaire.

Enfin, il faut voir que la traduction humaine de haute qualité est souvent produite par plusieurs personnes. Typiquement, le traducteur qui effectue le premier jet a une compétence technique très superficielle, mais connait très bien la terminologie et les deux langues. Le premier réviseur est un traducteur "senior", spécialiste du type de document considéré, et à même d'assurer l'homogénéité terminologique et stylistique. Enfin, on fait parfois intervenir un second réviseur, spécialiste du contenu technique du document et éventuellement ignorant de la langue source, pour détecter les contresens sémantiques linguistiquement plausibles et d'éventuelles ambiguïtés accidentellement introduites en langue cible. Chacun de ces intervenants *comprend,* certes, mais de manière différente.

1.2. Quelles connaissances doit-on utiliser pour "bien traduire" ?

Il y a essentiellement trois grands types de connaissances qui interviennent dans l'activité de traduction. Elles concernent le texte (linguistique) et le contexte, à savoir, l'univers de référence (sémantique) et l'aspect communicatif (pragmatique).

La connaissance linguistique ne se limite pas seulement à la connaissance des deux langues en présence. Elle concerne aussi les particularités du type de documents à traduire (vocabulaire, grammaire). Ainsi, il faut une assez longue expérience pour bien traduire des bulletins météo.

En général, un traducteur humain comprend facilement le contexte pragmatique et communicatif d'un texte. Par contre, sa connaissance du domaine spécialisé concerné est souvent très superficielle, voire nulle. Pourtant, s'il est expérimenté, il arrive dans une large mesure à faire illusion, c'est-à-dire à traduire comme s'il avait compris, en se fondant uniquement sur son habitude du type du texte. Il s'agit donc de *compréhension apparente* (humaine). L'absence de compréhension réelle se manifeste alors dans des fautes particulières. Prenons un exemple en théorie des langages formels et des automates, en supposant l'original en russe (où il n'y a pas d'articles).

Isxodæ iz pravoj linearnoj grammatiki G1, stroæt associirovannuº sistemu uravnenij, i vyvodæt regulærnoe vyraÂenie L(G1).	*À partir de* ***la*** *grammaire linéaire droite G1, on construit* ***le*** *système d'équations associé, et on en déduit* ***une*** *expression régulière pour L(G1).*

On peut "deviner" qu'il s'agit probablement de *la* grammaire et non d'*une* grammaire à cause de la rérérence à G1, si G1 apparaît auparavant. Par contre, pour ne pas traduire par *"**un** système d'équations"* ni *"**l**'expression régulière"*, il faut impérativement connaître la théorie en question. En fait, il n'est pas nécessaire de comprendre pourquoi il s'agit *du* système et d'*une* expression, il suffit de le savoir.

Qu'il s'agisse de compréhension explicite ou de compréhension apparente, il est donc possible de parler de "niveaux" de compréhension chez le traducteur humain, en fonction de son degré de connaissance du domaine et d'habitude des particularités des textes. Enfin, rien n'interdit à un traducteur — et c'est même fortement conseillé — de se renseigner auprès d'un collègue plus expérimenté, d'un spécialiste du domaine, ou de l'auteur. Qu'elle soit explicite ou apparente, sa compréhension est alors *indirecte*.

1.3. En pratique, des compromis sont nécessaires

Selon la qualité de traduction recherchée, le traducteur ou l'interprète *doit* plus ou moins comprendre. Pour une re-création ou une localisation, il faut une compréhension réelle, explicite, assez profonde. Par contre, pour une traduction-diffusion ou une traduction rapide, une compréhension apparente est suffisante. Notons toutefois que, dans le premier cas, la traduction doit être bien meilleure que dans le second, pour qu'un réviseur acccepte de réviser. Dans le second, une traduction purement littérale peut être utile à un utilisateur pressé d'accéder à une information peu fouillée (par exemple, y a-t-il eu une expérience ou un brevet sur tel sujet au Japon ?).

Il ne faut cependant pas oublier que la traduction est une activité soumise à bien des contraintes. Un traducteur est souvent obligé d'accepter des traductions dans des domaines qu'il connait peu, sur des textes divisés en petits morceaux et donc difficilement compréhensibles, et avec des délais toujours trop courts. En fonction de sa formation antérieure, de l'information accessible et des délais imposés, il *peut* plus ou moins comprendre.

I.2. Niveaux de compréhension en TAO

Voilà donc un point de départ. Cependant, pour parler des systèmes informatiques, il faut être plus précis sur les notions de qualité et de compréhension. La qualité ne peut à notre avis s'évaluer intrinsèquement, mais doit l'être par rapport à l'usage qu'on veut faire des traductions fournies.

Pour déterminer si la compréhension d'un système est explicite ou apparente, il convient d'analyser sa structure, et non son comportement : contient-il, oui ou non, un composant qui modélise l'univers de référence et la situation de communication, une "ontologie autonome" ?

2.1. De quelle TAO parle-t-on ?

Vers 1949, les USA, puis l'URSS, ont lancé des programmes motivés par le besoin de renseignements. C'est la *TAO du veilleur*. Il s'agit de traduction totalement automatique, dont on attend des traductions "grossières", produites rapidement, en grand volume et à bas coût. La qualité de ces traductions n'est pas essentielle, car elles servent à filtrer des documents, dont les plus intéressants peuvent, si nécessaire, être traduits ou communiqués à des spécialistes bilingues. Préédition et postédition doivent être absentes ou très limitées (ex : séparer les phrases, les formules, les figures…)[1]. Ce besoin est toujours actuel. Cependant, il s'agit maintenant plus de veille scientifique, technique, économique et financière que de renseignement militaire[2].

Une quinzaine d'années plus tard, on a commencé à travailler sur la *TAO du réviseur*, dans laquelle on produit des traductions "brutes", destinées à être révisées. Dans cette optique, la machine doit remplacer le traducteur, promu réviseur[3]. Typiquement, en moyenne industrielle, une page technique de 250 mots est traduite en 1 h et révisée en 20 mn. Avec 4 personnes, on passerait donc de 3p/h à 12p/h, et on multiplierait donc la productivité par 4 ? Il s'agit d'une limite, le chiffre le plus vraisemblable étant plutôt de 8 p/h, en comptant une révision plus lourde, de 30 mn/p.

---

[1] Les systèmes Systran sont essentiellement de ce type (par exemple, le système russe-anglais installé depuis 20 ans à la Wright-Patterson Air Force Base traduit, d'après nos informations, environ 18 millions de mots par an, avec une qualité tout à fait satisfaisante pour l'usage visé).

[2] A titre d'exemple, on peut citer l'accès en anglais à des bases de données japonaises depuis la Suède [51], ou encore depuis l'Europe (service Japinfo utilisant des traductions "grossières" d'ATLAS-II à peines "arrangées").

[3] La plupart des systèmes commerciaux récents visent ce créneau. On peut citer des systèmes japonais (AS-Transac de Toshiba, ATLAS-II de Fujitsu, PIVOT de NEC, HICAT de Hitachi, DUET de Sharp, SHALT d'IBM-Japon, PENSÉE de OKI, Majestic de l'Université de Kyoto et du JICST,…), ou américains (LOGOS, METAL), ou français (Ariane/aéro/F-E de SITE-B'VITAL, fondé sur les outils informatiques et les méthodes linguistiques du GETA). Il est aussi possible d'adapter des systèmes de conception plus ancienne à cet usage, si on les spécialise à un langage fortement contrôlé, comme cela a été fait chez Xerox avec Systran pour la traduction de notices d'anglais en 4 langues.

Que faire pour la plus grande partie des textes dont on veut obtenir de bonnes traductions ? La bureautique a commencé à apporter des solutions, sous forme d'outils de *TAO du traducteur*. Il s'agit de traduction humaine assistée par la machine, ou THAM, et pas de traduction automatique. C'est l'utilisateur qui traduit, en s'aidant de dictionnaires bilingues, de bases terminologiques, de thésaurus de "bitextes" (textes + traductions) etc., accessibles depuis un traitement de texte, le tout formant un "poste de travail pour le traducteur", réalisé sur microordinateur ou station de travail. Il s'agit d'outils comme Mercury/Termex™ sur PC, WinTool™ sur Macintosh, ou de systèmes complets (Weidner-Bravice, TSS de Alps). Le réviseur peut utiliser le même environnement, ou préférer travailler directement sous l'outil final de PAO (publication assistée par ordinateur).

Pour la majorité des besoins, et en particulier pour la traduction de manuels d'enseignement dans des pays où la langue nationale ne s'est que récemment affirmée comme support de l'enseignement secondaire et universitaire, la THAM est actuellement la seule voie réaliste. Il en est de même de toutes les traductions scientifiques et techniques de faibles volumes homogènes, voire de grands volumes homogènes trop mal rédigés (résumés avec des phrases de 15 lignes, par exemple) ou non disponibles sous forme magnétique cohérente et sans erreurs.

Enfin, on commence à voir apparaître des outils de *TAO de l'auteur,* destinés à des personnes ignorant la langue source. Cela correspond à des besoins croissants. Pour l'instant, il s'agit en fait de textes multilingues préenregistrés, personnalisables grâce à des parties variables. Par exemple, Ambassador™*, disponible en anglais-japonais, anglais-français, anglais-espagnol et français-japonais, offre environ 200 "formats" de lettres et formulaires, et 450 "moules" de texte (phrase ou paragraphe). Pour ce qui est de textes libres, il n'y a pas encore de produits commerciaux (bien que le système JETS d'IBM-Japon [19] soit prêt au moins depuis 1992). En tout état de cause, il ne pourra s'agir que de systèmes fondés sur le dialogue (TAFD), dont nous reparlerons plus loin.

2.2. Traduction humaine et TAO ne visent souvent pas la même chose

La question de savoir si un système de TAO doit ou peut comprendre ne s'applique évidemment pas aux systèmes d'aide au traducteur humain, mais seulement aux trois autres types de systèmes, qui présentent une automatisation totale ou partielle de la "fonction traduisante" (production d'un premier jet, traduction grossière ou brute).

Avec cette restriction, il est donc clair que ni la re-création ni la localisation ne sont abordées dans les systèmes de TAO actuels ou à l'étude : ici, l'homme n'est pas en concurrence avec la machine. De plus, si l'on classe les systèmes de TAO par type d'utilisateur (TAO du veilleur, du réviseur, du traducteur, du rédacteur), on s'aperçoit que, dans deux cas sur quatre (veilleur, traducteur), on ne cherche pas à produire une traduction comparable à la traduction humaine.

Ainsi, un système de TAO n'est pas toujours fait pour "traduire", au sens "humain" du terme ! Là encore, la machine ne va pas remplacer l'homme, mais couvrir des besoins non couvrables par des humains, soit qu'on ne puisse trouver assez de traducteurs pour traduire dans le délai imparti, ou en temps réel (bases de données), soit que leur coût soit de toutes façons trop élevé, soit encore qu'ils refusent un travail trop répétitif et inintéressant (bulletins météo, par exemple).

2.3. Architectures des systèmes de TAO et degrés de compréhension

Un système de TAO, ou de TALN en général, peut comporter trois composants distincts, linguistique, sémantique et humain, le premier étant seul nécessaire. Comme les logiciens, nous dirons qu'il ne peut y avoir compréhension au sens fort, ou compréhension explicite, que si le texte est interprété dans le composant sémantique ("ontologie" dans KBMT-89 [30] et KANT [6]). Le "niveau de compréhension" dépend alors du détail et de la complétude de cette ontologie par rapport à l'univers de référence réellement associé au texte, et de l'exactitude de l'analyse.

La "fonction traduisante" des systèmes de TAO du veilleur et du réviseur ne comporte qu'un composant linguistique, dans lequel on représente la connaissance linguistique de base, et les régularités spécifiques au type de textes visé. Cependant, comme les langues possèdent une sémantique "intrinsèque" [50], on peut coder certaines connaissances sur le domaine dans la base lexico-grammaticale (traits et grammaires sémantiques), et on peut lever un certain nombre d'ambiguïtés sémantiques et pragmatiques au moyen d'heuristiques linguistiques. Même dans ce dernier cas, la compréhension n'est qu'apparente, ou "implicite", c'est-à-dire que le système ne fait que permettre au lecteur (utilisateur final ou réviseur) de comprendre, il ne comprend pas lui-même.

---

* Disponible sur Macintosh et PC, produit par Language Engineering Corp., Belmont, MA 02178.

Quel type de compréhension peut-on envisager pour la TAO de l'auteur ? Elle ne permet aucune révision, et, sauf si l'on conçoit un système pour un groupe de spécialistes, elle ne permet pas non plus le recours à une base de connaissances : la compréhension ne peut être qu'apparente. Enfin, elle ne permet pas de se restreindre à un langage "contrôlé" (langage artificiel, non ambigu, à "consonance naturelle"), ni même à un "sous-langage" déterminé. C'est le grand public qu'il faut viser. Par conséquent, il semble difficile de s'appuyer sur les régularités formelles des textes pour obtenir la quasi-perfection nécessaire.

Est-ce la quadrature du cercle ? Oui, si l'on veut tout à la fois : du texte libre, pas de préédition, pas de postédition, et une grande qualité. Non, si l'on accepte un langage guidé, et une interaction avec l'auteur, permettant d'arriver :

- à une "standardisation" du texte à traduire, sous ses aspects lexicaux, grammaticaux et stylistiques ;
- à une "clarification" de ce texte, permettant de réduire les ambiguïtés lexicales, grammaticales et sémantiques (ex : *"Jean a acheté un livre à Pierre"*).

Les systèmes de TAO personnelle seront donc essentiellement fondés sur le dialogue, même si, dans certains cas, on peut imaginer de les lier à un système expert du domaine traité. Si l'on ne peut coder la connaissance nécessaire à une compréhension explicite parfaite, on peut utiliser le dialogue pour "aller la chercher dans la tête de l'auteur", et aboutir à une compréhension apparente "indirecte", c'est-à-dire à une représentation linguistique "profonde" du texte de qualité et de finesse au moins égales à celles que pourrait produire un processus de compréhension explicite (au moins égales, car l'auteur sera toujours plus compétent que tout système expert sur ce qu'il veut dire).

I.3. Réalités et perspectives

Comment ces considérations s'appliquent-elles en pratique ? On peut le voir en étudiant quelques systèmes, prototypes ou maquettes, par rapport aux situations concrètes dans lesquelles on les utilise ou les utiliserait.

Il n'est peut-être pas inutile de préciser ici ce que nous entendons actuellement par *systèmes, prototypes et maquettes* en TAO.

systèmes[4]: Il s'agit d'applications opérationnelles, ou de générateurs supportant de telles applications, avec (sauf pour le cas particulier de METEO) au moins 20 à 30.000 termes, 300 à 400 pages pour les grammaires d'analyse, et application à des flux de textes réels.

prototypes[5]: Il s'agit d'expériences de laboratoire portant sur des corpus, des grammaires ou des dictionnaires relativement limités, par exemple 5 à 10.000 termes, et n'ayant pas été testés dans des conditions opérationnelles, i.e. sur un flux de textes nouveaux.

maquettes[6]: Il s'agit de programmes développés pour l'expérimentation réduite de techniques ou d'architectures nouvelles.

3.1. La traduction automatique quasi-parfaite sans compréhension explicite est possible dans des cadres restreints (TITUS, METEO)

Depuis près de 15 ans, le système METEO [39, 71], issu des travaux du groupe TAUM de l'Université de Montréal, traduit les bulletins météo d'anglais en français. Actuellement, le volume est d'environ 40.000 mots par jour, et le taux de correction par les réviseurs humains est maintenu inférieur à 3% (en adaptant périodiquement le linguiciel à la dérive lexicale et grammaticale inévitable). On entend par là qu'il y a moins de 3 manipulations pour 100 mots, un remplacement comptant pour 2 (suppression suivie d'insertion). 6 ou 7 postes de traducteurs sont épargnés.

---

4 Comme (par ordre alphabétique) ARIANE (R-F au GETA, F-E à SITE-B'VITAL), AS-TRANSAC (E-J/J-E, Toshiba), ATLAS (E-J/J-E, Fujitsu), DUET (E-J/J-E, Sharp), ENGSPAN (E-S, PAHO), HICAT (E-J/J-E, Toshiba), JETS (J-E, IBM-Japon), KANT (E-F.S à Caterpillar), LOGOS (D-E.F...), METEO (E-F), METAL (D-E…, Siemens), MU-MAJESTIC (E-J/J-E, JICST), PENSEE (E-J/J-E, OKI), PIVOT (E-J/J-E, NEC), SHALT-J (E-J, IBM-Japon), SPANAM (S-E, PAHO), SUSY (R.F.E-D, IAI), SYSTRAN (CEE, Xerox...).

5 Comme ARIANE (pour le F-M à l'USM), ATLAS (pour diverses langues), ETAP (R-E/E-R à l'IPPI, Moscou), EUROTRA (CEE), ITS (BYU, Provo), JEMAH (USM, Penang), LMT (IBM-USA), METAL (pour le F-D à Liège, ou les autres paires en développement), SYGMART (E-S au CELTA, Nancy), ULTRA (UNM, Las Cruces), etc.

6 Comme DIALOG (Kitano, CMU), ELU (Bouillon & Estival, Genève), KBMT-89 (CMT, CMU), LIDIA-1, etc.

Autrefois, les traducteurs fuyaient ce “purgatoire”. Maintenant, ceux qui restent n’ont plus que le travail intéressant de révision, et restent volontiers plus longtemps.

À l’Institut Textile de France, on rédige à l’aide du système TITUS [41, 67] des résumés d’articles techniques dans un langage très fortement contrôlé, et on fabrique sur demande les résumés dans plusieurs langues à partir de la forme interne, seule conservée. La situation est différente de METEO, puisque l’auteur est forcé de rentrer dans un moule prédéfini, dans lequel toutes les ambiguïtés sont soigneusement éliminées, alors que les rédacteurs météo suivent plus ou moins des règles de rédaction : il s’agit d’un sous-langage observé et non d’une “pseudo-langue” construite, ce qui entraîne d’ailleurs la présence de nombreuses ambiguïtés.

Les grammaires et dictionnaires de METEO codent dans une certaine mesure la sémantique de la météo. Par exemple, on utilise des groupes du type “phénomène météo”, ou “évolution de situation” plutôt que groupe nominal, groupe verbal, etc.

Ces deux techniques sont excellentes dans le cas de domaines restreints. Cependant, elles ne sont ni extensibles ni portables. De plus, on trouve en pratique très peu de situations où l’on peut définir un langage contrôlé et imposer son usage, ou bien observer un sous-langage portant sur un domaine très particulier et écrire des grammaires et des dictionnaires “sémantiques” permettant la compréhension apparente. D’ailleurs, ces deux systèmes restent des cas isolés, malgré les efforts déployés depuis plus de quinze ans pour trouver des opportunités analogues.

3.2. L’approche par “sous-optimisation” permet d’obtenir une qualité suffisante pour la révision, en compréhension apparente

Les sytèmes actuels de “TAO du réviseur” ont une couverture trop large pour qu'on puisse, même à long terme, imaginer de les munir d'une ontologie. Ils ne peuvent donc comprendre au sens fort. Cependant, on peut obtenir d'excellents résultats en spécialisant leurs dictionnaires, leurs grammaires et leurs heuristiques de désambiguïsation à des sous-langages convenables.

L. Bourbeau et J. Lehrgerger [16, 42] parlent à ce propos de “sous-optimisation”. C’est la même idée qu’en IA : puisqu’on n’arrive pas à résoudre le problème dans toute sa généralité, on construit des sortes de “systèmes experts” de la traduction d’un certain type de textes. Bien sûr, on peut procéder de la même façon en TAO du veilleur, puisque tout gain de qualité se traduit par une plus grande satisfaction de l’utilisateur final.

Pour montrer ce qu’on peut obtenir par sous-optimisation, et illustrer les idées précédentes de façon plus précise, voici quelques exemples tirés de traductions brutes produites par le système Ariane/aéro/F-E de B’VITAL.

Il s’agit d’un système dédié à des manuels de maintenance en aéronautique, et écrit à l’aide d’Ariane-G5, le générateur de systèmes de TAO construit au GETA, et d’outils annexes, en particulier la base lexicale BDTAO, construits par B’VITAL.

Après essai, s’assurer du fonctionnement correct de l’ensemble raccord.	*After test, check that the coupling assembly works correctly.*

On remarque ici le passage d’un groupe nominal prépositionnel “du fonctionnement correct” à un groupe verbal, “that… works correctly”, avec pour corollaire le passage de l’adjectif “correct” à l’adverbe “correctly”. Ces transformations ne sont pas effectuées au transfert. C’est la première étape de la génération syntaxique qui, à partir de la “g-structure” (structure génératrice), considérée comme sous-spécifiée relativement aux fonctions syntaxiques et aux classes syntagmatiques et morphosyntaxiques, recalcule ces niveaux en fonction de l’objectif initial (ici, construire une phrase verbale), à partir des niveaux plus profonds (relations logiques à l’intérieur du cadre prédicatif strict, relations sémantiques pour les compléments circonstanciels).

Grâce à la notion d’unité lexicale (famille dérivationnelle comme *réparer, réparateur, réparation, réparable*, ou *utile, utilité, utilement,* le générateur sait, sans avoir besoin de consulter un dictionnaire, quels lemmes contient la famille dérivationnelle considérée, ce qui conditionne les paraphrases possibles. Ici, “fonctionnement” a été ramené à l’UL “fonctionner-V”, traduite par “work-V”, qui porte la potentialité de dérivation vers un nom d’action. C’est donc simplement l’ordre de préférence des règles de choix des catégories syntagmatiques qui provoque la construction d’une subordonnée plutôt que d’un groupe nominal (“the correct working of the coupling assembly”).

Porter sur celle-ci la date de la dernière réception ou révision.	*Write on this one the date of the last reception or of service.*

"Porter" est ici un verbe support, et "porter une date" est traduit par "to write a date" et non par "to carry a date", grâce à un test effectué en transfert lexical sur les traits syntaxiques et sémantiques de l'argument 1 de "porter" (l'objet logique). Mais le système ne "comprend" pas ce qu'est une date, ni ce qu'est l'écriture.

Effectuer la vidange générale et la purge du carburant (voir chapitre 12).	*Drain in a general manner and bleed fuel (see chapter 12).*

"Effectuer la vidange" est traduit par le verbe simple "to drain", grâce à la notion d'unité lexicale, et à l'organisation du transfert lexical. "Vidange" est ramené à "vidanger-V", et cette unité lexicale donne en traduction un arbre dans lequel on code la possibilité de la présence d'un verbe support du genre de "effectuer", "faire", etc., verbe dont la traduction sera effacée au cours du transfert structural.

Le bouchon a pour but d'assurer la protection d'un raccord auto-obturable lorsque celui-ci n'est pas utilisé au sol ou en vol.	*The trap is used for carrying out the self-sealing coupling protection when this one is not used at the ground or in flight.*

"Avoir pour but" est reconnu comme un prédicat composé, avoir-but-V(x0,x1), qui est traduit par use-V(x1,x0), avec conversion d'arguments, ce qui explique la génération d'un passif.

Enduire légèrement le joint neuf de liquide d'utilisation.	*Slightly coat the new joint with operating fluid.*

La traduction des prépositions est toujours délicate. Il faut savoir si elles introduisent des arguments ou des circonstants. Ici, "enduire-V" est un prédicat à 3 arguments (qn enduit qn/qc de qc), le troisième étant introduit par "de". L'analyseur préfère compléter le cadre argumentaire, et l'introducteur de cet argument pour "coat-V" est "with".

Ouvrir progressivement le robinet (3), appliquer une pression jusqu'à 1,5 bar jusqu'à l'allumage du voyant lumineux DS2 et l'extinction du voyant DS1.	*Gradually open tap (3), apply a pressure up to 1,5 bar until the light DS2 switching on ((ignition)) and the signal lamp DS1 extinction.*

La préposition "jusqu'à" introduit ici deux circonstants. Ce qu'on traduit en fait, c'est la relation sémantique (ici, RS=LOC avec SEM=TEMPS et SLOC=QUA), précisée par la préposition et par les traits sémantiques du gouverneur ("head") du groupe, ici PROCESSUS pour "allumer-V", et du prédicat (appliquer-V).

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.*

Aucune transformation explicite n'est effectuée. La proposition infinitive est rendue par une subordonnée par le simple fonctionnement du générateur, expliqué plus haut. Comme l'argument 0 (sujet logique) n'est pas exprimé, on génère un passif. Il s'agit d'une préférence de style, et on pourrait aussi bien générer "until one obtains a pressure of 9 bars", ou "until obtaining…", comme dans l'exemple suivant.

Procéder à la dépose des panneaux.	*Remove the panels.*
IMPORTANT : avant de déposer ou de reposer le panneau central intrados de voilure, il est nécessaire de procéder à certaines modifications.	*IMPORTANT : before removing or reinstalling the lower central wing panel, it is necessary to proceed with some modifications.*

Ici, la construction préférée pour la conjonction "before" est le gérondif. D'autre part, la préposition "à" introduit l'argument 1. Dans la structure produite par l'analyseur, elle peut fort bien avoir été supprimée. "With" est contenu dans le cadre de valence de "proceed-V" pour la même position argumentaire, et est fabriqué par le générateur.

3.3. L'approche "fondée sur la connaissance" donne de très bons résultats par compréhension explicite, mais reste encore peu utilisable

L'approche "fondée sur la connaissance" (TAFC) a été défendue pendant près de vingt ans par Shank et son école comme la seule permettant de résoudre les problèmes de la TAO. Mais on ne réussissait pas à construire de maquette convaincante, tandis que les systèmes de TAFL progressaient et se diffusaient. Les sceptiques avaient beau jeu de souligner les difficultés théoriques et pratiques de l'entreprise. C'est seulement quand les chercheurs en IA ont abandonné leur quête irréaliste (ou prématurée) de solutions générales pour se tourner vers des objectifs réalistes mais limités, en développant des "systèmes experts", ainsi que des outils permettant le développement de bases de connaissances non triviales, que cette idée devint réalisable.

Le premier prototype de système de TAO "fondé sur la connaissance", KBMT-89 [30], fut développé au CMT (Center for Machine Translation) de CMU (Carnegie Mellon University). Il

utilisait une “ontologie” de son domaine (PC-XT et PC 5550 japonisé), le corpus étant constitué d’une vingtaine de pages tirées des manuels de ces PC. Son lexique représentait environ 1.200 termes, et son ontologie 1.600 “concepts”. Même dans ce cadre restreint, il apparut qu’on ne pouvait pas résoudre toutes les ambiguïtés automatiquement, et qu’il fallait donc une révision, ou un dialogue avec un humain. Cette dernière solution fut retenue, et le système fut muni d’un “augmenteur” [27], posant des questions à un spécialiste ne connaissant que la langue source.

Le CMT a récemment trouvé une première application industrielle, et développé à partir de KBMT-89 le système KANT [6], utilisé pour traduire de la documentation d’équipements lourds chez Caterpillar. La différence essentielle avec KBMT-89 est que, pour supprimer toute interaction durant le processus de traduction, on exige que le texte d’entrée appartienne à un langage non-ambigu strictement contrôlé, ce qui entraîne une interaction assez lourde au moment de la création.

Il y a actuellement environ 14.000 sens pour les termes généraux, et quelques centaines de termes spécialisés (non-ambigus), ce qui est encore loin des tailles courantes dans les systèmes de TAFL. D’autre part, les traductions obtenues sont grammaticalement et stylistiquement très bonnes, mais seraient souvent rejetées par des réviseurs professionnels comme des paraphrases inexactes. Prenons deux exemples donnés par Nyberg & Mitamura.

Anglais source	Français cible	Allemand cible
`In order to prevent a fire hazard, do not overload AC outlets.`	`Afin d’éviter tout risque d’incendie, ne jamais surcharger les prises CA.`	`Vermeiden Sie Feuergefahren, indem Sie die Netzanschlüsse nicht überlasten.`
Traductions plus exactes (topic et focus ont été inversés en allemand, “tout” devrait venir de “any”, et “jamais” de “never”)	*Afin d’éviter les risques d’incendie, ne pas surcharger les prises CA.*	*Um Feuergefahren zu vermeiden, überlasten Sie die Netzanschlüsse nicht.*
`If the TV set has been dropped, a shock hazard may exist.`	`La chute du téléviseur peut provoquer un risque de choc électrique.`	`Wenn Sie das Fernsehgerät fallen lassen, kann die Gefahr eines Elektroschocks bestehen.`
Traductions plus exactes (les relations sémantiques et temporelles ont été incorrectement traduites, “Sie” devrait venir de “you”)	*Si on a laissé tomber le téléviseur, il peut y avoir un risque de choc électrique.*	*Wenn man das Fernsehgerät hat fallen lassen, kann die Gefahr eines Elektroschocks bestehen.*

Ces traductions ne sont en fait pas meilleures que des traductions obtenues par des systèmes de TAFL spécialisés à des langages contrôlés analogues [41], ou même à des sous-langages observés très restreints [39], qui sont excellentes. Même si la qualité de la TAFC arrivait à dépasser celle de la TAFL sur ce genre de textes, on y gagnerait fort peu. On ne sait pas non plus si la TAFC peut produire des traductions brutes aussi bonnes que la TAFL sur des sous-langages observés (et non contrôlés) assez larges, — voir à cet égard les exemples de TAFL donnés plus haut.

La faisabilité technique de la TAFC a maintenant été établie. Pour l’instant, elle est cependant moins applicable que la TAFL, puisqu’elle impose un langage fortement contrôlé, et que, selon les mots (peut-être trop optimistes) de Nyberg & Mitamura eux-mêmes, “il est probable que la construction d’une base de connaissances sur le monde suffisante pour permettre la TAFC dans tout domaine de discours ne sera pas réalisée avant quelques années”[7].

Cependant, avec l'émergence des systèmes experts de grande taille, on peut espérer trouver dans le futur des entreprises ou des organismes qui auront développé de tels systèmes pour la CAO ou la CFAO, et qui auront besoin de traduire des volumes importants de textes portant sur les produits correspondants. On pourra alors développer des systèmes de TAFC sans avoir à construire (et à maintenir) une ontologie complète dans le seul but de traduire, ce qui coûte fort cher. L’ontologie pourra être obtenue à partir de la base de connaissances, ou même être réduite à une simple interface avec cette base, et il suffira de la coupler à un système de TAFL robuste [59].

[7] «It is probably the case that the implementation of a world knowledge base sufficient to support KBMT in any domain of discourse is some years from realization.»

Conclusion sur la compréhension et la TAO

Il n'est absolument pas envisageable pour l'instant d'automatiser la traduction-recréation ni la traduction-localisation plus que par la mise à disposition d'outils d'aide au traducteur humain. Par contre, la "fonction traduisante" est automatisable dans les contextes de la traduction-diffusion et de la traduction-dépistage.

Dans certains cas, les systèmes de TAO "traduisants" *peuvent* comprendre (explicitement). Même quand cela est possible, un dialogue ou une révision semble rester nécessaire si l'on veut atteindre la qualité d'un spécialiste humain bilingue sans recourir à un langage strictement contrôlé. Dire que les systèmes de TAO *doivent* comprendre au sens fort serait se condamner à ne pas utiliser des systèmes de TAFL disponibles et… sans doute insurpassables dans certains cas.

L'utilisation de la "sémantique intrinsèque" des langues et des régularités des textes permet d'ores et déjà d'obtenir une qualité permettant de parler de "compréhension apparente directe", suffisante pour la TAO du réviseur, et *a fortiori* pour la TAO du veilleur. Enfin, l'introduction d'un dialogue homme-machine approprié devrait permettre de construire des systèmes à "compréhension apparente indirecte" donnant des traductions de haute qualité sans révision de textes non contrôlés.

II. Aspects situationnels, scientifiques et ergonomiques de la TAFD

II.1 Opportunité de la TAFD

1.1. Motivations

Les recherches et développements en TAFD sont motivés par la limitation des paradigmes actuels, par l'importance croissante des langues nationales dans le contexte de l'internationalisation, et par de récents progrès méthodologiques et technologiques.

a. Limitation des paradigmes actuels

La TAFL est très bien adaptée à la TA du veilleur. Des systèmes portables de qualité tout à fait convenable pour l'usage visé commencent à se répandre au Japon, à des prix abordables (environ 5 M¥ pour un système logiciel et matériel complet[8]). Par contre, elle est loin de pouvoir répondre à tous les besoins en traduction du réviseur. Outre le fait qu'elle demande évidemment autant de révisions que de langues cibles, elle reste trop chère pour des usages légers. En effet, selon les chiffres donnés par les producteurs de systèmes de TA [28], la création *ex nihilo* d'un système opérationnel de TAFL coûte entre 200 et 300 hommes-années, avec des développeurs spécialisés, et l'adaptation d'un système de TAFL existant à un nouveau domaine et à une nouvelle typologie de textes coûte de l'ordre de 5 à 10 hommes-années. Un utilisateur n'a intérêt à s'équiper d'un tel système que s'il a à traduire de gros flux de textes homogènes et informatisés, comme des manuels d'utilisation ou de maintenance[9]. Adapter un système disponible à des besoins ponctuels n'est pas non plus une solution viable[10].

D'autre part, une condition essentielle de succès de ce type de TAO est de constituer une équipe de développement et de maintenance des linguiciels (dictionnaires, grammaires) qui soit en liaison constante avec l'équipe de révision, et si possible avec les auteurs des documents à traduire[11]. C'est ce qu'a réussi la PAHO (Pan American Health Association) [44], avec ses systèmes

---

8 Station Sparc, lecteur optique, système avec 80.000 termes, dictionnaire utilisateur, et options de personnalisation.

9 En prenant l'hypothèse d'un système coûtant 1 MF (400 KF de base et 600 KF de spécialisation au vocabulaire et au type de texte) et d'un amortissement sur 2 ans, il faut un flux de 10.000 p/an (en comptant 10%/an de maintenance, 60 F/page de coût machine, et 100 F/page de révision, contre 150 F/page de traduction et 70 F/page de révision pour la méthode manuelle, soit 60 F/page de gain pour amortir 1,2 MF). À coût machine nul, il faudrait encore 5.000 p/an.

10 Ce serait comme réoutiller une usine pour produire quelques dizaines de voitures. En effet, sans compter la saisie optique ou manuelle, entraînant toujours un coût important de vérification, ni la maintenance, ni même l'achat du système de base, mais seulement sa spécialisation (600 KF) et les coûts de traduction et de révision, on arrive avec les hypothèses précédentes à 632, 680, 760 et 920 KF pour 200, 500, 1.000 et 2.000 pages, contre 44, 110, 220 et 440 KF pour la méthode classique manuelle, soit environ 14,5, 6, 3,5 et 2 fois plus, respectivement.

11 Dans le "contre-rapport ALPAC" du JEIDA [28] comme au MTS-II à Munich en août 1989, Fujitsu reconnaissait clairement avoir fait une erreur en distribuant largement ATLAS-II : seules étaient en effet rentables les traductions effectuées chez Fujitsu, soit pour sa documentation, soit dans le cadre d'un contrat avec la CEE (Japinfo) concernant la veille technologique et ne demandant donc qu'une révision minimale.

ENGSPAN et SPANAM. On peut faire un parallèle avec les systèmes experts, qui peuvent être développés par des tierces parties, mais qui doivent ensuite être totalement maîtrisés par leurs utilisateurs, seuls à même de les faire évoluer de façon adéquate.

En ce qui concerne la TAFC, elle est totalement inapplicable en TAO du veilleur, et nous avons montré qu'elle était moins applicable que la TAFL en TAO du réviseur. Tant qu'il faudra construire les ontologies spécialement pour la traduction, elle restera aussi plus onéreuse.

b. Importance croissante des langues nationales et de l'internationalisation

De plus en plus, nous désirons rédiger dans notre langue, et transmettre nos textes à l'étranger, qu'il s'agisse de messages électroniques, de lettres, d'articles, de manuels techniques, voire de livres. Contrairement à ce que d'aucuns prédisaient il y a une cinquantaine d'années, l'internationalisation croissante ne s'est pas accompagnée d'une uniformisation linguistique vers l'anglais, mais au contraire d'un renforcement considérable de l'usage scientifique et technique des langues traditionnellement importantes de ce point de vue, et d'une promotion volontariste de bien d'autres, pour les amener au même niveau (malais-indonésien ou arabe, par exemple). A notre sens, cette évolution ne fera que se renforcer, les langues étant, comme le notait le Pr. Hagège dans un article paru dans *Le Monde* début 1990, les "drapeaux des identités nationales".

Il ne s'agit pas seulement de politique, mais d'efficacité. Dans les projets coopératifs européens (Esprit, Eureka), par exemple, la communication est gênée par la nécessité de lire et d'écrire en anglais. Pour la grande majorité des participants, lire en anglais pose des problèmes de compréhension et prend beaucoup de temps. Quant à écrire, si même c'est envisageable, le résultat est souvent difficile à comprendre, voire illisible.

Les trois types de TAO "classique" ne peuvent évidemment répondre à ce nouveau besoin. En effet, la TAO du veilleur, sans préédition ni postédition, ne peut donner une qualité suffisante, et la TAO du réviseur comme la TAO du traducteur s'adressent par définition à des spécialistes au moins bilingues, et non à des rédacteurs supposés ne connaître aucune des langues cibles (ou au plus une, et ce imparfaitement).

c. Progrès méthodologiques et technologiques

L'idée de la TAFD date des années soixante [75], et a été incorporée à plusieurs maquettes ou prototypes dans les années soixante-dix et quatre-vingt [32, 46, 47, 49, 52, 61, 65, 74]. Si ces travaux n'ont pas donné lieu à des systèmes utilisables en pratique, c'est que les dialogues devaient être conduits par des spécialistes[12], que la couverture linguistique était trop limitée, et que l'on ne disposait pas encore d'environnements interactifs conviviaux.

La méthodologie s'est affinée ces dernières années. Tout d'abord, l'utilisateur envisagé n'est plus un spécialiste, mais un rédacteur, ou plutôt un *auteur* [13, 15, 18, 19, 22, 25, 31]. Nous préférons ce dernier terme. D'un côté, en effet, "auteur" est moins restrictif que "rédacteur" : un auteur est quelqu'un qui veut créer un texte, et peut le faire en l'écrivant, en le dictant, ou encore en le construisant interactivement. De l'autre, "auteur" est plus restrictif que "rédacteur", "locuteur", ou "commentateur", car "auteur" désigne quelqu'un qui désire créer un produit final "propre", alors que les autres termes peuvent renvoyer à des personnes désirant seulement produire un message écrit ou parlé de façon "spontanée", en vue d'une communication immédiate, et non disposées à conduire un dialogue éventuellement lourd pour rendre leur message "propre"[13].

D'autre part, l'informatique personnelle a fait des progrès gigantesques. On dispose maintenant d'ordinateurs personnels très puissants et bon marché, d'environnements conviviaux, de l'intégration du multimédia, et d'outils de télécommunication permettant éventuellement le recours à des serveurs. Enfin, les techniques et outils de génie logiciel modernes (essentiellement la programmation par objets), permettent de construire des systèmes complexes et interactifs bien plus rapidement et sûrement que par le passé.

---

12 ITS [74] demandait même *plusieurs* intervenants, un linguiste spécialiste du système pour l'analyse, et un bilingue pour chaque langue cible.

13 "Propre" signifie ici conforme à une certaine grammaire (permettant éventuellement des constructions incorrectes, pourvu qu'elles soient attestées) *et* ne comportant pas de parties "réflexives", ou "automodificatrices", si fréquentes dans la parole et même l'écriture spontanée (au moins manuscrite), comme des hésitations, des faux départs, des reprises, des répétitions, des corrections, des abréviations arbitraires (apocopes), etc.

1.2. Situations traductionnelles pour la TAFD

a. Citères de choix de l'approche par dialogue

Nous proposons quatre critères pour choisir la TAFD :

- la qualité visée doit être élevée, et la révision impossible ou très coûteuse ;
- le contexte doit être fortement multilingue (1—>n, comme pour la dissémination de documentation technique, ou n<—>n, comme dans des projets internationaux) ;
- l'entrée ou le domaine ne doivent pas être trop contraints ou contrôlés (sinon, mieux vaut utiliser la TAFL ou la TAFC) ;
- les utilisateurs doivent être prêts à participer à des dialogues de normalisation et de désambiguïsation.

Dans toute situation, il faut de plus pouvoir rendre les dialogues acceptables, en les laissant à l'initiative de l'utilisateur, en lui fournissant des moyens de les contrôler et de les réduire (en jouant sur des paramètres, en insérant directement des marques de désambiguïsation, etc.), et en lui laissant si possible le choix entre plusieurs média.

b. Situations traductionnelles adaptées à la TAFD

Parmi les situations favorables avec entrée écrite, on peut mentionner :

- la traduction de volumes relativement faibles de documentation technique en plusieurs langues, typiquement 5.000 à 8.000 pages à distribuer sur un DON[14], par exemple dans les 9 langues de la CEE (et peut-être dans d'autres aussi, comme le russe, l'arabe, le japonais, ou le chinois).
- la diffusion d'information dans plusieurs langues (sur la circulation, sur la météo, ou dans des congrès, des manifestations sportives, des situations d'urgence…), qui demande une sortie orales aussi bien qu'écrite ;
- l'échange télématique de notes et de documents de travail dans des projets internationaux.

Comme l'état de l'art en reconnaissance de parole ne permet pas de traiter à la fois un grand vocabulaire, de la parole continue, et un locuteur arbitraire, il semble n'y avoir que très peu de situations favorables à la TAFD avec entrée orale :

- la production de commentaires ou de résumés à partir de scènes visuelles et auditives (par exemple, pour le sous-titrage d'émissions de télévision en langues étrangères) ;
- l'interprétation de dialogues bilingues très contraints, tels que les appels téléphoniques de politesse entre parents d'enfants échangés entre familles pour apprendre les langues, ou l'assistance téléphonique à des voyageurs étrangers (consultation médicale, réservation…). Ici, le dialogue doit être le plus réduit possible, et une combinaison entre TAFD et TAFC (analogue à l'architecture finale de KBMT-89) semblerait indiquée.

Il y a enfin beaucoup de situations intéressantes où le message source n'est créé que pour vérifier le contenu du (ou des) message cible, résultant d'une négociation avec un expert. C'est par exemple le cas de lettres officielles ou formelles, qui ont des structures très différentes dans différentes cultures. Somers & Tsujii [22] ont proposé le terme de "traduction sans texte source", mais il est peut-être plus exact de parler de génération multilingue de textes que de TAFD.

II.2 Aspects scientifiques

2.1. La controverse entre praticiens et théoriciens

Comme nous le disions en commençant, les avis sur la traduction automatique (TA) sont souvent extrêmes, et la controverse est souvent vive. Certains ne la conçoivent que comme l'expérimentation scientifique de leurs théories ou de leurs formalismes favorits. À COLING-90, par exemple, on ne comptait plus les titres du genre "le formalisme XYZG et son application à la TA", alors que cette application était tout au plus mentale et hypothétique. Dans un autre registre, M. Gross a soutenu pendant des décennies qu'on ne pouvait espérer faire de la TA sans avoir "mis la langue à plat". Outre que cette ambition est sans doute illusoire, dans son principe même, la

[14] Disque optique numérique (CD-ROM), contenant près de 600 M d'octets.

langue étant essentiellement productive, la réalité a démontré par l'absurde la fausseté de ce point de vue, puisqu'en fin de compte il y a des systèmes qui traduisent réellement, et bien, mais justement parce qu'ils sont limités à des sous-langages.

D'autres voient la TA comme une entreprise essentiellement technologique et utilitaire. A. Colmerauer, qui fit de la TA à l'université de Montréal, a parlé d'un "gigantesque bricolage". M. Nagao insiste aussi souvent sur le fait que les théories n'apportent qu'un cadre incomplet, et que l'essentiel du travail linguistique consiste à décrire une énorme quantité d'exceptions. Y. Wilks, chercheur en IA et auteur d'une expertise sur Systran, soutint même, au séminaire "Sémantique formelle et linguisique computationnelle" (Lugano, 1988), que "il n'y a pas de théorie linguistique assez mauvaise pour ne pas faire de TA avec" !

D'autre part, vouloir développer une théorie élégante et l'utiliser comme fondement exclusif d'un système opérationnel mène à l'échec sur les deux fronts, comme cela a été démontré avec éclat par le projet Eurotra, et par plusieurs autres projets centrés sur les grammaires d'unification : la théorie ne progresse pas d'un iota, et on n'obtient pas de système utilisable.

Comme B. Vauquois, M. Kay, S. Nirenburg, M. Nagao, Yu. Apresyan, et bien d'autres, nous préférons dire clairement que la TA est essentiellement une entreprise technologique, bien qu'elle entretienne avec les diverses sciences qui la sous-tendent (linguistique, informatique, ergonomie) des relations tout à fait analogues à celles du génie civil avec la physique. L'incorporation d'idées théoriques (mais non de théories entières) peut mener à des progrès tangibles, mais seulement de façon incrémentale, et souvent après un recul initial des performances. En retour, la pratique amène parfois à poser des questions théoriques intéressantes.

2.2. Des progrès techniques peuvent venir de la théorie

Citons brièvement quelques exemples d'idées théoriques qui ont fait progresser la TA. Il y a d'abord eu l'incorporation de formalismes chomskyiens comme les automates d'états finis, les grammaires hors-contexte, et les grammaires transformationnelles, qui ont permis de travailler sur des représentations arborescentes des phrases au lieu de se limiter aux chaînes. Ces formalismes ont été immédiatement étendus (calculs d'attributs, primitives de contrôle, etc.) pour devenir utilisables en pratiques mais ces extensions n'ont été reprises dans la théorie que 20 ans plus tard (GPSG).

Il y a ensuite l'usage de structures de dépendances, adaptées de Tesnière, de l'école de Prague, et de l'école russe, et combinées avec des idées de la logique formelle, ce qui a mené à des structures comportant prédicats, arguments et circonstants, les circonstants (et plus rarement les arguments) portant des "relations sémantiques", ou "cas profonds".

L'introduction d'unités lexicales (familles dérivationnelles) comme éléments lexicaux de base pour le transfert et la génération se révéla très fructueuse, mais ne fut possible qu'en simplifiant notablement la théorie de Mel'tchuk, par réduction de l'ensemble des fonctions lexico-sémantiques aux principales dérivations productives.

Enfin, il faut admettre que la technique fait feu de tout bois, et que les systèmes qui tournent se rattachent à *plusieurs* théories à la fois. S. Nirenburg a trouvé un terme qui exprime très bien ce que nous venons de dire : un bon système de TA doit être capable d'intégrer progressivement de nombreuses *microthéories.*

Y a-t-il des idées théoriques inutilisables en TAFL ou TAFC, et qui pourraient bénéficier à la TAFD ? Nous le pensons. Un premier exemple serait l'utilisation effective de la notion zembienne de "statut". La "triade statutaire" est la décomposition d'une proposition en thème, rhème et phème [70][15], et est très importante pour la traduction (choix des articles, portée et place de la négation…). Mais elle est pratiquement impossible à calculer automatiquement, même s'il y a des critères formels pour certains cas précis (subordonnées négatives en allemand, "ordre communicatif" des circonstants dans les langues slaves…). Pour déterminer le statut d'un terme de la proposition, il convient de poser des questions à son auteur. Et c'est justement ce que permet la TAFD.

Par exemple, pour traduire la phrase suivante (adaptée de Chomsky), on peut demander si Jean est un laveur de voitures. Si oui, "les voitures" est rhématique (première traduction).

Jean ne lave pas les voitures	*Hans wäscht keine Wagen* *Hans wäscht die Wagen nicht.*

[15] Le rhème est ce qu'on dit du thème, et le phème la modalité (kantienne) de cette assertion. Le thème persiste dans l'existence quand le phème varie : quand "Hans wäscht die Wagen nicht", il y a toujours des voitures à laver !

D'autres exemples concernent la détermination de l'aspect et de la modalité, toujours au moyen de questions. Voici un exemple dû à Tomita [12] :

Le courrier est arrivé ce matin	*The mail arrived this morning* *The mail has arrived this morning*

2.3. La technologie suscite des questions théoriques

Dans la majorité des situations adaptées à la TAFD, il faut un système de couverture lexicale et grammaticale très large. Cela amène à poser aux théoriciens des questions importantes :

- Sachant qu'on n'obtient de bons résultats que sur des langages restreints, *comment construire une base de connaissances linguistiques utilisable comme une union de sous-langages ?* Est-il possible de séparer les aspects grammaticaux et lexicaux ?
- *Comment atteindre la large couverture nécessaire ?* Typiquement, un système de TAFL contient de 3.10^4 à 3.10^5 termes, en 2 langues. Le cas de METEO (3.10^3) est atypique, à cause de son domaine très restreint. Mais un système de TAFD visant le grand public et non restreint à un domaine particulier demandera de 3.10^5 à 3.10^6 termes, en plusieurs langues !
- Dans des situations fortement multilingues, l'approche par interlingua est séduisante. Mais. *comment surmonter les difficultés d'ingénierie rencontrées dans la construction d'un grand lexique interlingue* par les récents projets japonais ATLAS, PIVOT, EDR, et CICC ?
- Il est crucial que des non-spécialistes puissent facilement comprendre les questions du système, éventuellement lui demander les raisons de certaines questions, et comprendre ses réponses. Une question importante (et nouvelle) est donc de trouver *comment rendre la base de connaissances linguistique d'un système de TAFD accessible à un utilisateur naïf.*

II.3 Aspects ergonomiques

Les aspects ergonomiques sont bien sûrs importants en TAO classique. En TA du veilleur ou de l'auteur, il y a une notion de qualité *apparente,* liée à la présentation. La même traduction paraît bien meilleure si elle est formatée comme un texte que si elle est présentée phrase par phrase, ou pire en colonne. De même, on accepte plus volontiers des choix portant sur de petits groupes de mots que sur des phrases complètes. Les constructeurs de systèmes ont beaucoup travaillé sur les éditeurs bilingues, sur les outils de paramétrage, et sur les aides à la création de dictionnaires.

En TAFD, l'ergonomie est un aspect absolument crucial, et les choix ergonomiques influencent directement l'architecture de tout le système. Les choix principaux sont les suivants :

- Le système doit-il fonctionner en temps réel, ou l'asynchronie est-elle préférable ?
- Doit-il tourner sur des ordinateurs personnels bon marché, ou sur des stations de travail ? Une architecture avec serveur est-elle possible ? Si oui, pourrait-on simplement connecter un PC au minitel ?
- Comment les dialogues doivent-ils être organisés ? Est-il nécessaire et/ou possible de les conduire dans un environnement multimédia ? Plus précisément, l'utilisation de synthèse de parole peut-elle améliorer l'efficacité et la convivialité des dialogues de désambiguïsation ?

III. Architecture linguistique et voies intermédiaires nouvelles

Le cadre nouveau de la TAFD peut amener à rechercher de nouvelles architectures linguistiques. Il ne s'agira sans doute pas de solutions radicalement nouvelles, mais, comme souvent dans un domaine technique, de nouveaux compromis, de voies intermédiaires nouvelles, avec ça et là des innovations intéressantes.

III.1. Transfert multiniveau à acceptions, propriétés et relations interlingues

Les systèmes de TAO modernes sont fondés sur un *transfert sémantique,* le *passage par un interlingua,* ou un *transfert multiniveau.* "Transfert sémantique" est un terme introduit par les Japonais pour désigner exactement l'approche du CETA entre 1960 et 1970, que B. Vauquois appelait le "pivot hybride" : la structure interface source fournie au transfert contient des éléments lexicaux de la langue source, et des propriétés et relations interlingues (traits sémantiques, cas profonds, nombre, aspect, modalité, détermination abstraits…). Dans un "interlingua", les éléments

lexicaux sont de plus interlingues (les auteurs des systèmes actuels parlent de “concepts”, mais il n’y a pas toujours d’ontologie).

Le transfert (sémantique) multiniveau, au sens de Vauquois [45, 62] diffère du transfert sémantique en ce que, outre les attributs et relations interlingues, on garde des attributs et des relations spécifiques à la langue source (classe syntagmatique, genre, nombre, détermination, temps, mode, fonction syntaxique…)

En TAFD, nous proposons de rajouter aux représentations multiniveaux un niveau lexical, celui des *acceptions interlingues,* sans aller jusqu’à introduire des concepts, puisqu’il faudrait alors construire une ontologie. La base lexicale multilingue sous-jacente (BDLM) contiendra alors un dictionnaire monolingue pour chaque langue traitée par le système, et un dictionnaire interlingue pour les acceptions interlingues. Chaque acception interlingue a une image dans chaque dictionnaire monolingue, avec une définition appropriée dans la langue correspondante, utilisée lors de la désambiguïsation interactive du sens[16].

Remarquons aussi que “interlingue” ne signifie pas “indépendant des langues”. Par exemple, si le système travaille avec le fançais, l’anglais et le russe, il y aura une seule acception pour “mur” en tant qu’objet concret. Dès qu’on ajoutera l’allemand ou l’italien, il faudra ajouter les raffinements “mur vu de l’extérieur” (Mauer, muro) et “mur vu de l’intérieur” (Wand, parete).

III.2. Langage guidé

2.1. Séparation lexique/grammaire

La notion de "sous-langage" a été introduite et étudiée par le linguiste R. Kittredge [51, 64], à la suite de son expérience en TAO (il fut directeur du groupe TAUM de l'Université de Montréal au début des années 70). Kittredge a donné un certain nombre de critères, essentiellement lexicaux et syntaxiques, pour évaluer la difficulté d'un sous-langage pour les techniques de TAO du réviseur, et pour déterminer si l'approche dite de "deuxième génération avec sous-langage" (ou “sous-optimisation”) était prometteuse, et les a appliqués à un certain nombre de types de textes.

Son analyse, extrêmement complète, distingue les aspects lexicaux et grammaticaux, la liaison plus ou moins forte avec un domaine sémantique clos, et la possibilité d'écrire une grammaire textuelle dépassant le niveau de l'énoncé. Il introduit la notion formelle de "clôture lexicale", qui signifie en gros que le nombre de nouveaux termes rencontrés dans une nouvelle page diminue rapidement et tend vers zéro ou une valeur très faible quand le nombre de pages augmente.

Mais il ne propose aucune notion formelle analogue pour l’aspect grammatical : un sous-langage est défini comme l’ensemble des phrases (ou des énoncés) produisibles dans des conditions fixées (par exemple, bulletins météo, appels d'offres du Secrétariat d'Etat, manuels de maintenance de tel avion, etc.), sans qu’on sache comment choisir un échantillon convenable et comment généraliser autrement qu’intuitivement. De plus, le terme choisi est gênant : Kittredge fait lui-même remarquer qu’un sous-language d’une langue n’est pas un sous-ensemble de cette langue (cela est dû au fait que “le français” signifie en général le français standard des manuels, et pas l’union de tous ses jargons et parlers régionaux).

D’autre part, si l’usage de la TA se répand grâce à la TAFD, il sera impossible de produire et de maintenir une collection très variée de bases lexicales et grammaticales de grande taille correspondant à des sous-langages au sens de Kittredge, une pour chaque type d’utilisation. À terme, il faudra donc un dictionnaire total aussi exhaustif que possible, et cela ne sert à rien d’utiliser la notion de “clôture lexicale” pour le limiter. La même chose est vraie de la grammaire.

Pour réduire le problème (diviser pour régner !), il y a une voie intermédiaire, qui consiste à séparer les deux aspects, puis à définir chacun d’eux en deux temps, d’abord de façon grossière à l’aide d’un formalisme symbolique simple, puis de façon plus fine en ajoutant des paramètres numériques.

Un type de texte donné pourra alors être défini comme un ensemble de poids relatifs à des arcs et à des nœuds d’une BDLM structurée en réseau sémantique, son “profil lexical”, et comme un style d’énoncés ou un genre de texte vérifiant certaines contraintes numériques.

---

16 On peut aussi *expliquer* à l’auteur pourquoi une telle question est posée, et même montrer les mots en question dans les autres langues. L’introduction d’aspects d’auto-apprentissage dans ce genre de systèmes les rendrait plus acceptables par les utilisateurs potentiels.

2.2. Préférences lexicales

La BDLM d'un système de TAFD doit contenir des relations entre acceptions qui permettent d'en extraire des thésaurus, en particulier la synonymie, la quasi-synonymie et l'hypéronymie, et des relations analogues aux relations lexico-sémantiques entre termes (comme entité —> qualité, action —> argument 1 de l'action…).

Les poids attachés aux acceptions, aux termes et aux relations entre eux constituent ce qu'on pourrait appeler un *profil lexical.* Au fur et à mesure que le temps passe, le système de TAFD peut les faire varier en fonction de l'interaction, ce qui permet un certain réglage automatique, et la définition de nouveaux profils lexicaux reflétant les préférences lexicales courantes. Les poids sur les termes reflètent leur "degré de pertinence" par rapport à la tâche en cours, et peuvent être utilisées pour indiquer à l'auteur le terme préféré parmi un ensemble de (quasi-)synonymes (comme par exemple avion, appareil, aéronef). Associés aux poids sur les acceptions et sur les relations entre termes et acceptions, ils peuvent aussi être utilisés pour lever des ambiguïtés de sens, comme cela a été montré dans [21]. En TAFD, cela peut servir à présélectionner le sens le plus probable.

Remarquons que, dans le contexte d'un système de "TAFD pour tous", la base lexicale devra contenir une très grande variété de termes, même incorrects ou douteux, alors que les bases de données terminologiques ne contiennent d'habitude que des termes normalisés ou recommandés.

2.3. Types de textes : "styles d'énoncés" and "genres de textes"

En ce qui concerne l'aspect grammatical, on peut proposer de diviser encore le problème en deux, en définissant des *styles d'énoncés* pour les phrases et autres énoncés traduisibles individuellement (titres, éléments homogènes de longues énumérations…) et des *genres de textes* pour les textes plus longs[17]. Pour cela, nous supposons que nous avons recensé toutes les constructions d'une langue, y compris les constructions rares ou n'apparaissant que dans des types de textes très techniques (ex. : "Mettre interrupteur sécurité train avant sur OFF"), et que nous les avons représentées au moyen d'un très grand ensemble R de règles déclaratives. Tout formalisme déclaratif simple convient[18].

Un *style d'énoncé* est alors un sous-ensemble de R vérifiant certaines restrictions numériques (par exemple, sur le degré d'imbrication, d'ellipse, ou de coordination). Par exemple, M1 pourrait être le style des phrases simples sans sujet ("permet de sauver sur disque une copie de votre fichier"), fréquentes dans les documents techniques, et M2 le style des phrases explicatives simples.

En ce qui concerne les *genres de textes,* il est souhaitable qu'on puisse dans le futur les prendre en compte dans des éditeurs de documents structurés fondés sur SGML, qui peuvent traiter des textes beaucoup plus longs que les outils linguistiques actuels, le plus souvent limités à la phrase ou au paragraphe. On peut alors proposer de définir un genre de texte comme une expression algébrique sur les styles d'énoncés et les genres de textes. Par exemple, le genre de texte S1 des paragraphes commençant par une phrase de style M1 suivie par une suite (éventuellement vide) de phrases de style M2 peut être définie par une simple expression régulière :

```
<S1>          = M1 M2*
```

Pour décrire le genre de textes d'un "document" commençant par un titre (microlangage M3), et se poursuivant par une liste non vide de paragraphes (microlangages M2 et/ou M4) et/ou de sections de même structure qu'un document, on peut de même écrire[19] :

```
<Document>    = <Title> <Content>
<Title>       = M3
<Content>     = (<Paragraph> | <Document>)+
<Paragraph>   = (M2 | M4)+
```

Il faudra plus de recherche pour trouver comment guider les auteurs dans la sélection d'un style d'énoncés ou d'un genre de textes particulier, de façon à ce que la critique textuelle, la standardisation et la clarification puissent être menées avec efficacité.

---

[17] Les termes de "microlanguage" et de "sous-language" proposés dans [15], se sont révélés surchargés et anti-intuitifs.

[18] Par exemple, les grammaires hors-contexte (CFG), avec ou sans attributs, les TAG, les STCG [38, 56, 63]. Pour le moment, nous utilisons ROBRA dans LIDIA-1, malgré son caractère procédural, et testons dans chaque règle transformationnelle si le style d'énoncé attendu est l'un des styles contenant cette règle.

[19] Un symbole entre crochets désigne un genre de texte, "|" l'alternation, "*" et "$^{+}$" la répétition. On pourrait ajouter des conditions sur des attributs associés aux symboles principaux.

III.3. Accessibilité des connaissances

Dans la plupart des systèmes de TAO, les informations linguistiques sont très détaillées, et codées de façon compréhensible uniquement par des spécialistes. Dans certains, qui se présentent plutôt comme des aides au traducteur humain (Weidner, ALPS), on a au contraire cherché à n'utiliser que des informations très simplifiées, pour que des utilisateurs naïfs puissent coder eux-mêmes les dictionnaires. Le résultat a simplement été que les traductions étaient trop mauvaises pour servir de base à une révision[20].

En TAFD, il nous semble que l'information ne doit pas être simpliste, mais peut-être moins fouillée que dans le cas de la TAFL. Par exemple, il est sans doute inutile d'avoir un système trop riche de codes sémantiques. Une hiérarchie à 3 ou 4 niveaux, avec au plus une dizaine de codes par niveau, est sans doute le maximum si l'on veut que des utilisateurs puissent compléter les dictionnaires, qui, même très grands, ne seront jamais complets.

Un second aspect, très délicat et intéressant, est de trouver comment exprimer les notions linguistiques obscures pour le commun des mortels d'une façon compréhensible. Pour certaines, comme l'aspect (voir plus haut l'exemple du courrier qui "est arrivé"), il faut sans doute utiliser des exemples, ou de la reformulation, et éviter de parler de la notion elle-même.

Enfin, il faut arriver à organiser le système de façon à ce que l'utilisateur, même monolingue, puisse contrôler ce que produit le système dans les langues cibles. Nous proposons pour cela un mécanisme de "rétrotraduction" (cf. infra).

3.3. Annotations et préédition indirecte

L'idée de base de la TAFD est de remplacer la postédition par une préédition indirecte. Cela signifie que le texte est enrichi, normalement indirectement, grâce à l'interaction avec l'auteur. Mais des utilisateurs expérimentés doivent pouvoir faire une partie de cette préédition directement, pour éviter de longs dialogues. C'est pourquoi il faut représenter un texte, avec son système d'écriture, sa structure logique, ses marques de désambiguïsation, et ses résultats d'analyse, par une chaîne de caractères *portable* et *lisible,* dans l'esprit de la TEI (Text Encoding Initiative).

IV. Une situation concrète pour la TAFD et la maquette LIDIA-1

Pour l'instant, la TAFD est encore en phase de recherche préliminaire. Au GETA, nous avons lancé autour de ce concept le projet LIDIA (Large Internationalisation des Documents par Interaction avec leurs Auteurs). Pour commencer, nous nous limitons à une situation particulière, où un rédacteur monolingue produit de la documentation technique à traduire en plusieurs langues.

Dans une première étape, nous construisons une maquette de petite taille, LIDIA-1, dont l'objectif est uniquement de nous permettre d'expérimenter une architecture nouvelle, et d'attaquer un certain nombre de problèmes linguistiques, informatiques, et ergonomiques. Cependant, notre souci constant est d'effectuer des choix cohérents avec l'énormité des bases lexicales et grammaticales qu'un système grand public devrait offrir pour être viable.

IV.1. Quelques choix techniques

1.1. Situation multicible

Pour LIDIA-1, on suppose qu'un ingénieur français crée de la documentation technique sous la forme d'une pile HyperCard, sur un Macintosh de puissance moyenne, et aide le système à la traduire en anglais, allemand et russe. Nous avons choisi une architecture distribuée (station de travail de l'auteur sur Macintosh et serveur de TAO sur un mini IBM-4361). Le choix des langues est uniquement dû aux compétences disponibles, et le choix d'une situation monosource et multicible permet évidemment de réduire le coût de l'opération en n'ayant qu'une analyse à construire ! Il y a des arguments moins contingents pour les deux autres choix.

1.2. Hypertexte

Avec l'arrivée d'HyperCard™, les hypertextes sont sortis des laboratoires. Pour un prix très modique, n'importe qui peut réaliser des documentations vivantes, des animations, etc., avec image

---

[20] En 1985, le Bureau des Traductions d'Ottawa mena une étude consistant à comparer la productivité de traducteurs humains utilisant le système Weidner sans, puis avec l'option de TA, en leur faisant traduire le même livre sur l'odontologie à 3 mois de distance. Dans le second cas, la productivité baissa de 40% !

et son en prime. Certaines documentations de voitures (Renault, Peugeot) sont diffusées sur DON sous HyperCard (8.000 pages de documentation, en 10 langues, avec figures et éventuellement messages oraux, sous forme phonétique codée, 30 à 40 fois plus compacte que le signal).

Du point de vue ergonomique, l'hypertexte privilégie l'interaction. Par conséquent, on peut penser qu'un rédacteur sera plus prêt à accepter une interaction linguistique sous hypertexte que sous traitement de texte. Dans le premier cas, on reste dans la logique de l'outil. Dans le second, il faut changer ses habitudes, ce qui est souvent fort difficile.

Du point de vue linguistique, les parties textuelles sont bien isolées, et connexes, au contraire des traitements de texte ou des logiciels de PAO, où l'on trouve un mélange de codes de formatage, de texte, de figures, de formules, le tout parfois présenté de façon non connexe (tableaux avec tabulations…). De plus, les fragments de texte d'une pile HyperCard typique (champs et boutons) sont petits, et grammaticalement très homogènes : on peut parler de "types de fragments" beaucoup mieux que de "types de documents". Par exemple, un champ donné peut contenir, dans chaque carte, un titre avec verbe à l'infinitif ("Naviguer dans HyperCard"), un élément de menu fonctionnant comme un nom dans d'autres contextes ("Quitter" — "…cliquer sur Quitter"), un ordre à l'infinitif ("Prendre la disquette"), une explication simple sans sujet pour la première phrase ("vous permet de quitter l'application et…"), ou encore un titre simple sans verbe ("Rotation à gauche"), etc.

Le concept de TAFD n'impose pas l'utilisation d'hypertexte. On pourrait également le mettre en œuvre dans le cadre de "texteurs" ou de "documenteurs" plus classiques. Certains, comme WinText™, contiennent déjà des marques indiquant, outre la fonte, le corps et le relief, la langue naturelle de segments de texte. En ajoutant le type de fragment, et en développant une interface analogue, on pourrait les étendre pour y introduire les fonctions précédentes. Malgré tout, il faudrait se limiter à des texteurs ou documenteurs programmables (comme Interleaf™), ou modifier les codes sources. Le même problème se pose d'ailleurs aux concepteurs de "postes du traducteur".

1.3. Architecture distribuée et traitements asynchrones

Nous désirons que la station de rédaction soit un microordinateur convivial et largement diffusé, d'où le choix du Macintosh sous HyperCard. Nous désirons aussi réutiliser la puissance de notre générateur de systèmes de TAO (Ariane-G5), pour écrire les parties "lourdes" du traitement linguistique. Or, il ne tourne que sur mini IBM (Eurolang l'a en partie poté sur SUN en 1992), mais pas sur Macintosh. Quand bien même il le serait, son exécution, même en tâche de fond, serait trop lente sur ce type de matériel, et augmenterait inévitablement les temps de réponse.

D'autre part, l'exemple du système CRITIQUE [43] a montré qu'un traitement asynchrone convenablement organisé pouvait donner un système très convivial. Nous avons donc opté pour un traitement distribué entre la station de rédaction et un serveur de TA, et un fonctionnement asynchrone, pour ne pas pénaliser le rédacteur par des interruptions forcées ou des attentes interminables. Tout doit rester sous son contrôle.

Enfin, ce type d'organisation devienne de plus en plus réaliste, avec la disponibilité de réseaux télématiques puissants et relativement peu coûteux. Il est tout à fait possible d'imaginer un système de TAFD où le micro se connecte au serveur à des intervalles réglables par l'utilisateur, et où la connexion ne dure que quelques secondes, tout comme un utilitaire de messagerie.

IV.2. Aspects informatiques de la maquette LIDIA-1

2.1. Intégration dans HyperCard et piles "traduisibles"

a. HyperCard

Il y a cinq sortes d'objets en HyperCard, les boutons, les champs, les cartes, les fonds et les piles. Les boutons sont des zones actives de l'écran, qui provoquent des actions quand ils sont "cliqués". Les champs peuvent contenir du texte éditable, et les boutons du texte non éditable.

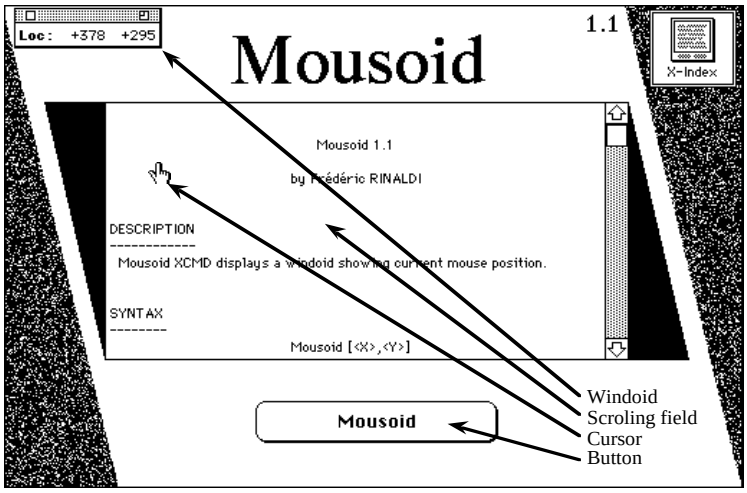
Une carte a ses propres boutons et champs, et un fond qui a à son tour des boutons et des champs. Un fond peut être partagé par plusieurs cartes. La carte recouvre son fond (les deux ont la même taille). Des dessins peuvent être dessinés sur les cartes et sur les fonds.

Les cartes sont regroupées en piles, chaque pile étant un fichier Macintosh. Une pile peut avoir plusieurs fonds. L'utilisateur communique avec HyperCard en agissant sur la carte affichée à l'écran, le mode exact d'interaction étant déterminé par un jeu de préférences. Nous avons ajouté une nouvelle préférence (une case à cocher) pour lancer ou arrrêter LIDIA-1.

b. Contraintes à respecter pour qu'une pile soit traduisible

Nous dirons qu'une pile est organisée de façon "traduisible" si l'on n'a pas à traduire les scripts[21], mais seulement les noms des boutons et les contenus textuels des champs[22].

Les scripts doivent donc être écrits indépendamment des langues.



Exemple de carte HyperCard

Cela nous conduit à introduire les restrictions suivantes :

- les références à des boutons ne doivent jamais être leurs noms, mais leurs numéros, invariants en traduction ;
- les messages ne doivent jamais être contenus dans les scripts, mais toujours pris dans des champs normaux[23], invisibles dans la version finale de la pile ;
- le texte contenu dans les dessins ne doit pas être traduit ;
- toute version personnalisée de la barre de menus doit être traduite à la main.

c. Scénario

Si l'auteur coche la case de préférence HyperCard LIDIA, le Macintsoh se connecte périodiquement au serveur de TAO. L'utilisateur travaille normalement sur sa pile. Quand il décide que certains objets (champs, boutons, cartes, ou toute la pile) sont prêts à être traduits, il active l'outil "traduction" dans la palette LIDIA, sélectionne ce ou ces objets, et continue à travailler pendant que les processus liés à la traduction s'exécutent.

L'état de traduction de tout objet peut être contrôlé grâce à un *témoin d'état* (status watcher). Quand l'intervention de l'utilisateur est requise, LIDIA envoie un signal (paramétrable), exactement comme une tâche de fond comme PrintMonitor ou Eudora. L'utilisateur est libre d'interagir tout de suite ou plus tard.

Pour interagir avec un objet, on double-clique simplement sur son témoin d'état et on active l'item approprié dans le menu déroulant qui apparaît. Le mode interaction peut être quitté à tout moment. Les objets changent d'état lors de l'interaction et sont repris en charge par LIDIA pour poursuivre le traitement si nécessaire.

2.2. Organisation des traitements linguistiques

Expliquons brièvement les traitements principaux illustrés dans la figure ci-dessous.

1. Après l'étape de standardisation,

- tous les champs et boutons doivent être porter un type de texte[24], permettant de contrôler le correcteur stylistique, puis de guider l'analyse (d'où un module de *catégorisation textuelle) ;*

---

21 Chaque objet d'HyperCard a un *script*, éventuellement vide, écrit dans le langage de programmation HyperTalk. Un script est une collection de *handlers*, chacun de la forme "on <message> do <suite d'instructions HyperTalk> end". Un handler est *invoqué* quand son <message> (un clic de souris, par exemple) est reçu par l'objet dont le script le contient.

22 On a fait ici le choix de traduire une pile complètement, en autant de piles qu'il y a de langues cibles. Une autre possibilité serait de rendre la pile multilingue en créant une copie de chaque conteneur de texte pour chaque langue cible.

23 Cependant, elles peuvent contenir des variables : "Où est le fichier &1 ?" doit certainement être traduit.

24 Dans le cas de textes "incomplets", par exemple si le sujet de la première phrase est contenu dans un autre champ (comme dans des tables contenant des noms de commandes et leurs explications), ce module demande aussi comment construire le texte complet.

- les textes doivent être orthographiés correctement, et être conformes aux paramètres stylistiques associés au type de leur conteneur (nous espérons avoir intégré les *correcteurs grammaticaux et stylistiques* de Machina Sapiens quand cet ouvrage paraîtra) ;
- les syntagmes figés se comportant de façon spéciale (comme l'item de menu Cacher les bulles d'aide dans "Cacher les bulles d'aide arrête l'aide par bulles") doivent être marqués (l'utilisateur doit aider le *module de syntagmes figés spéciaux* à établir la liste des syntagmes figés spéciaux de la pile)[25];
- les préférences terminologiques (cf. supra) doivent avoir été plus ou moins imposées par le *module de préférence lexicale,* selon le degré de normalisation terminologique souhaité.

2. Le texte standardisé est alors analysé sur le serveur. La *mmc-structure source* produite (Multisolution, Multiniveau et Concrète) est transformée en une forme portable et lisible (directement par Lisp… et par les développeurs) et envoyée au Macintosh.

3. La mmc-structure source est utilisée pour produire le dialogue de désambiguïsation sur le Macintosh. Le processus de désambiguïsationla transforme en une *umc-structure source* non-ambiguë (Unisolution, Multiniveau et Concrète) correspondant à l'analyse choisie par l'auteur[26].

4. Cette umc-structure source est alors "abstraite", ou "réduite", à une *uma-structure source* (Unisolution, Multiniveau et Abstraite).

5. À partir de la uma-structure source, le système Ariane-G5 produit les *gma-structures cibles* (Génératives, Multiniveaux et Abstraites), en utilisant les transferts adéquats. Une gma-structure est plus "générale" et plus "générative" qu'une uma-structure, car ses niveaux de surface (fonctions syntaxiques, catégories syntagmatiques…) peuvent être vides, et sinon ne sont que des préférences indiquées par le transfert.

6. Pour chaque langue cible, la génération structurale produit à partir de la gma-structure cible une *uma-structure cible* homogène avec ce que serait le résultat de l'analyse (et de la désambiguïsation) du texte cible qui sera généré. Cette étape consiste à choisir la paraphrase à générer en calculant les niveaux de surface[27] et une première approximation de l'ordre des mots à partir des niveaux plus profonds (relations logiques et sémantiques, traits sémantiques, etc.).

7. Le processus de traduction se termine par les générations syntaxiques et morphologiques. Quand tous les objets ont été traduits, on obtient la ou les piles images dans la ou les langues cibles.

8. Les uma-structures cibles peuvent être utilisées comme point de départ de *rétrotraductions* permettant à l'auteur (monolingue) de contrôler les traductions.

L'idée est que le système, traduisant par exemple de français en russe, retraduise au rédacteur ce qu'il va produire en russe, lui permettant ainsi de contrôler le résultat sans connaître un mot de russe. C'est d'ailleurs souvent ce qui se passe entre un interprète et son client.

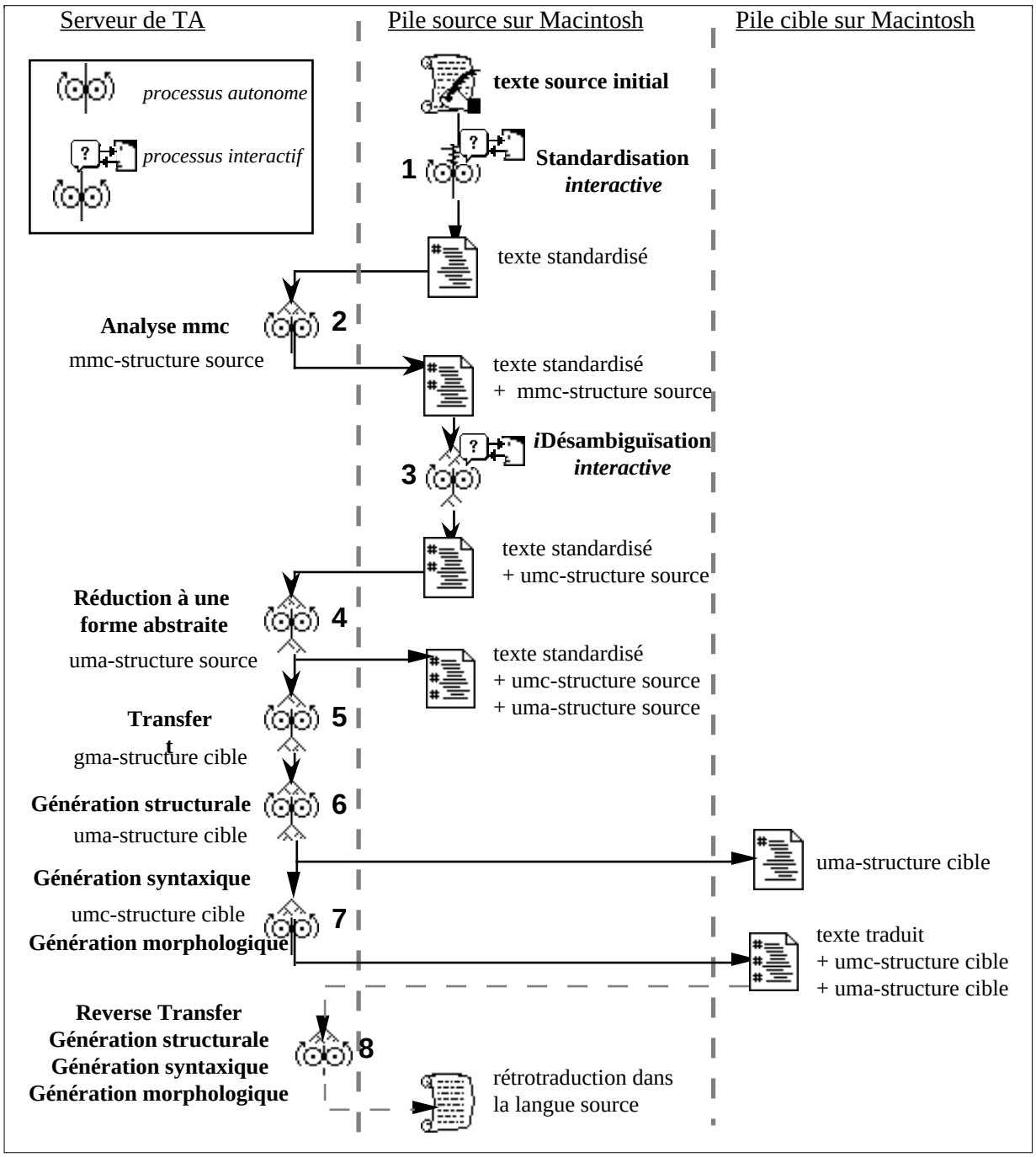
Comme on suppose que la génération ne pose pas de problèmes, et qu'on veut éviter d'avoir à écrire tous les analyseurs dans une situation monosource et multicible, on ne repart pas du texte final en russe, mais de son uma-structure produite par la génération structurale. La rétrotraduction n'a pas à être identique à l'original pour que la traduction soit bonne, tout comme en traduction et surtout en interprétation humaine : on pourrait par exemple conserver actif et passif dans un sens (F-R), et les échanger dans l'autre (R-F) : il n'y aurait même pas de point fixe.

---

[25] Les deux modules précédents peuvent travailler directement avec le texte contenu dans l'objet HyperCard. À partir d'ici, le système travaille sur une transcription contenue dans un enregistrement du "fichier miroir" associé à la pile, ainsi que sur des résultats intermédiaires de traitement. Cela nous force à verrouiller le champ textuel original (sauf si l'auteur décide de le modifier et accepte de recommencer l'interaction depuis le début).

[26] "Concrète" signifie que le texte original peut être retrouvé à partir de la structure de façon directe (par un parcours préfixe des feuilles pour des structures de constituants et par un parcours infixe de tous les nœuds pour des structures de dépendances). Les nœuds et/ou les arcs de la structure peuvent contenir aussi bien de l'information "de surface" que de l'information "profonde" (organisation en prédicats/arguments, relations sémantiques…). dans les structures "abstraites", les négations, les auxiliaires, les articles, etc., peuvent avoir été supprimées en tant que nœuds et être représentées dans les décorations, certains éléments élidés peuvent avoir été insérés, l'ordre peut avoir été normalisé, etc.

[27] En particulier, les fonctions syntaxiques, les classes syntagmatiques, et les classes morphosyntaxiques, ces dernières en fonction des schémas dérivationnels des unités lexicales.



Organisation générale du processus de traduction en LIDIA-1

2.3. Implémentation

a. Répartition des tâches entre le serveur de TA et le Macintosh

Comme indiqué sur le diagramme, on trouve sur le serveur de TAO des "morceaux" de systèmes de TAFL (dictionnaires et grammaires écrits dans des langages de règles), écrits en Ariane-G5. Nous n'entrerons pas ici dans le détail.

Sur la station de rédaction, on trouve les modules mentionnés plus haut, ainsi que les ressources correspondantes (dictionnaire des syntagmes spéciaux, BDLM par acceptions interlingues) et les programmes propres à LIDIA. De plus, chaque pile à traduire est "doublée" par un "fichier miroir", stockant les unités de traduction et les résultats des divers traitements.

Sur la station de rédaction comme sur le serveur de TAO, on trouve bien sûr des outils de gestion de la communication et de l'interaction.

b. Sur le Mac

Outils de programmation

HyperCard est le frontal de tout le système, mais, comme HyperTalk n'est pas assez puissant pour supporter certaines des tâches nécessaires, en particulier la génération des dialogues de désambiguïsation, nous avons écrit la plus grande partie de notre logiciel en CLOS (Common Lisp Object System). HyperCard et les programmes CLOS communiquent par des "AppleEvents", en utilisant le protocole standard IACP (Inter Application Communication Protocol) de Mac.OS-7 . l'échange des données entre le Macintosh et le serveur de TA est implémenté à l'aide du kit de programmation d'Avatar (Mac-MainFrame programmer's Toolkit).

Objets

LIDIA-1 a 2 catégories principales d'objets, les *conteneurs* et les *contrôleurs*, avec 3 classes de conteneurs (`LIDIA-File, Mirror-Object, Disambiguation-Scheduler`) et 4 de contrôleurs (`LIDIA/HC-Communication-Controller, Remote-Translation-Jobs-Entry-Controller, Translation-Jobs-Fetch-Controller,` et `Translation-Process-Controller`). `LIDIA-File` a 5 instances, `Mirror-File, Translation-Jobs-in-Demand, To-Be-Fetched-Translation-Jobs, Prepared-Dialogues, Suspended-Treatments`. Ces fichiers contiennent toute l'information nécessaire sur l'état global de tout le système et sont constamment mis à jour (comme les piles HyperCard elles-mêmes).

Les *objets miroirs* contiennent toute l'information nécessaire au processus de traduction et à la construction des piles cibles.

Nous distinguons entre les *informations statiques* et les *informations dynamiques.*

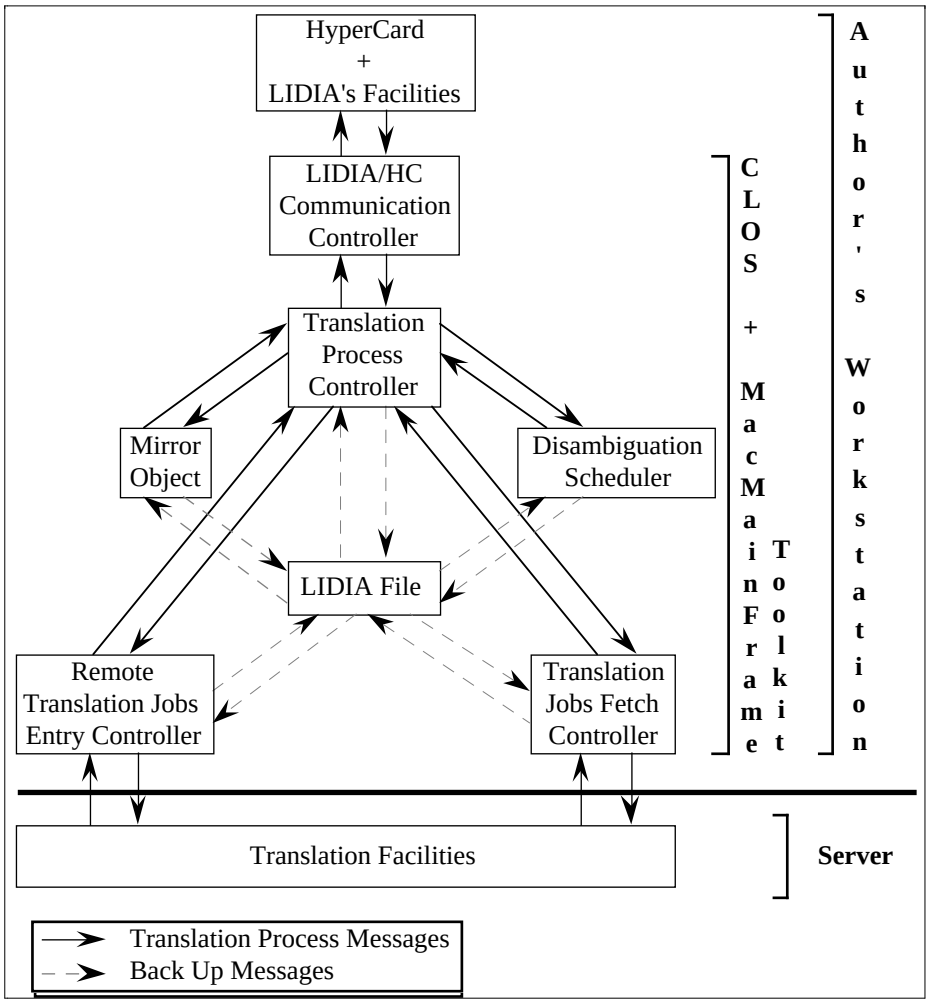
Les informations statiques sont celles attachées par HyperCard à chaque objet. Les informations dynamiques sont celles utilisées par LIDIA pour traduire le contenu d'un objet.

Informations statiques :

- numéro de carte.
- localisation (fond ou carte).
- ID d'objet[28].
- type d'objet (champ ou bouton).
- fonte, taille et style.
- visibilité de l'objet.

Informations dynamiques :

- style d'énoncé/genre de texte.
- information textuelle sous la forme directement manipulée par HyperCard.
- transcription du texte, avec les annotations éventuelles.
- traitement à effectuer.
- langue(s) cible(s).
- étape courante du traitement.
- structures concrète et abstraite source.
- résultat(s) de traduction.
- uma-structure(s) cible(s) et rétrotraduction(s) éventuelle(s).



Communication entre les objets de LIDIA-1

---

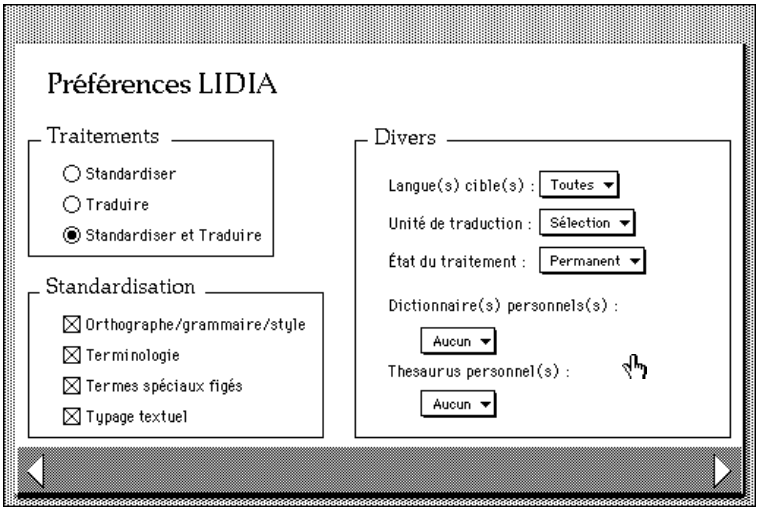
[28] HyperCard affecte un numéro dit "ID d'objet" (object ID number) à chaque objet d'une pile. Ce numéro est unique pour chaque type d'objet à l'intérieur de l'objet qui le contient, ne change jamais et, si l'objet est supprimé, n'est pas réaffecté à un objet nouvellement créé. LIDIA utilise ces IDs d'objet de façon interne.

Interface utilisateur

i. Préférences

Il y a trois sortes de préférences : Traitements (en haut à gauche) standardisation (en bas à gauche) et divers (à droite).

Pour les traitements, l'utilisateur choisit entre standardisation seule, traduction seule, ou les deux. La standardisation est personnalisable. Si le module de catégorisation textuelle n'est pas utilisé, un type par défaut est attaché à tous les conteneurs de texte.



Carte Préférences de LIDIA-1

Les préférences diverses concernent la ou les langues cibles, les unités de traitement (objet, carte ou pile sélectionné(e), ou automatique, un texte étant alors soumis dès que l'auteur quitte son conteneur), le type de retour (sur demande ou automatique), et les dictionnaires et thésaurus personnels actifs.

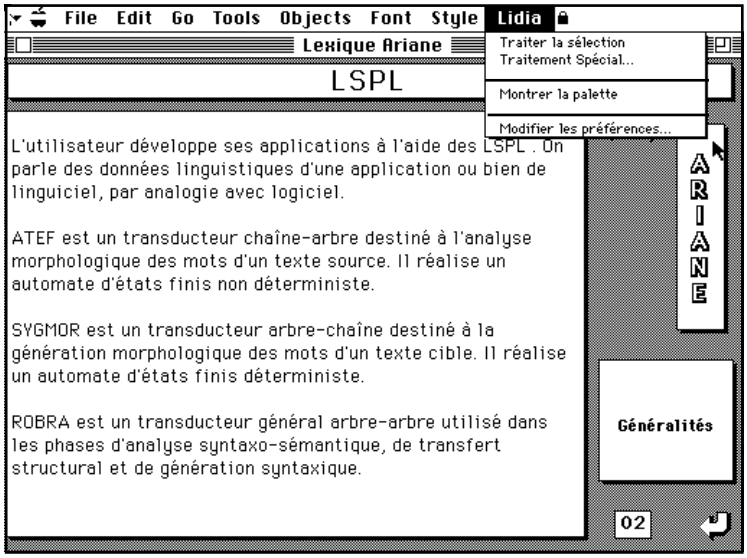
ii. Menu

LIDIA est accessible par le menu `Lidia`, qui change selon les préférences choisies.

Le menu montré ici offre quatre choix : traiter l'objet sélectionné selon les préférences en cours, ou le traiter avec d'autres préférences, montrer la palette, et modifier les préférences.

Quand l'auteur choisit l'un des deux premiers items, le curseur change de forme (✓ pour le premier cas, et ✓? pour le second) et on peut sélectionner l'objet à traiter.

La palette LIDIA est montrée ci-dessous.



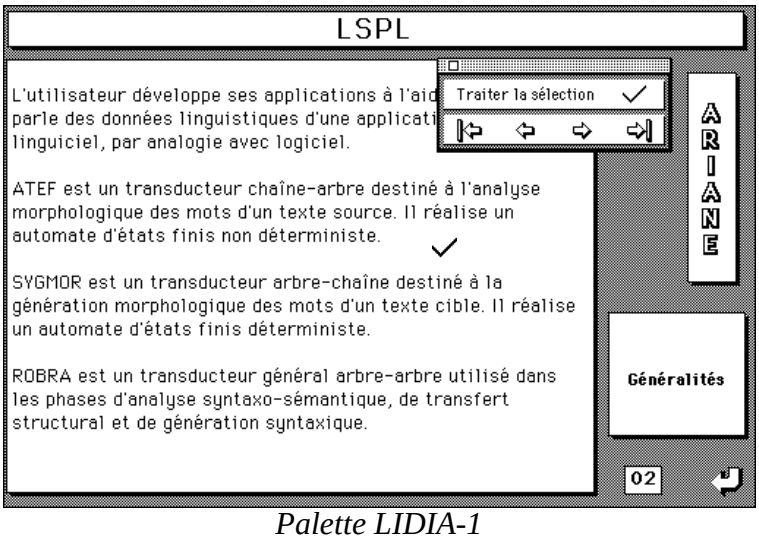
Menu LIDIA

iii. Palette

Une palette est une fenêtre flottante qui est toujours au-dessus de toutes les fenêtres ouvertes d'une application.

En cliquant sur la partie du haut de la palette LIDIA, on active l'outil de traitement par défaut (✓), et on sélectionne ensuite les objets à traiter.

En cliquant sur les icônes du bas (⇦|, ⇦, ⇨, ou |⇨), on va à la première carte de la pile, à la précédente, à la suivante, ou à la dernière.



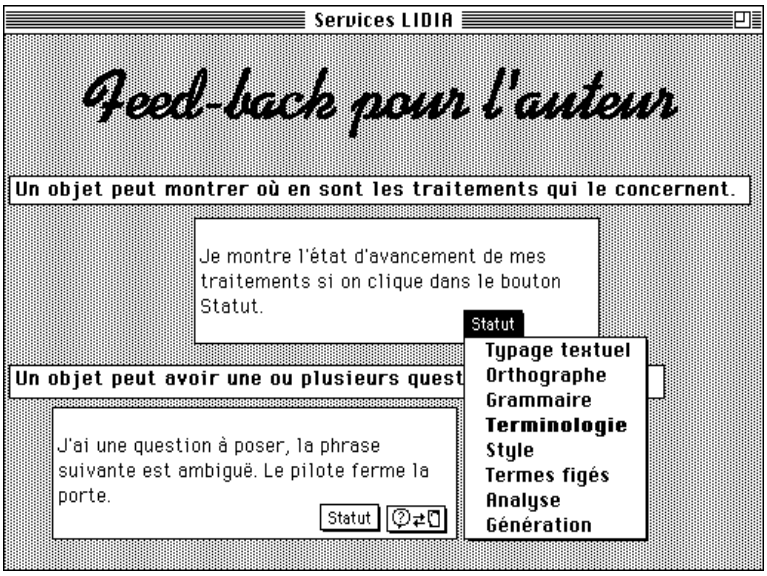
Palette LIDIA-1

iv. Contrôle

Un témoin d'état est un champ fugitif (pop-up) qui montre toutes les tâches, dans l'ordre d'exécution. La tâche en cours est affichée en gras.

Quand des questions sur un objet sont prêtes à être posées, un bouton (⍰⇄🗋) apparaît au-dessus de lui. Si l'auteur désire répondre, il clique sur ce bouton et le dialogue commence.

Quand la dernière réponse a été donnée, le traitement suivant peut commencer (si un dialogue est quitté sans être terminé, LIDIA-1 attend que l'auteur y revienne).



Un observateur d'état de LIDIA-1

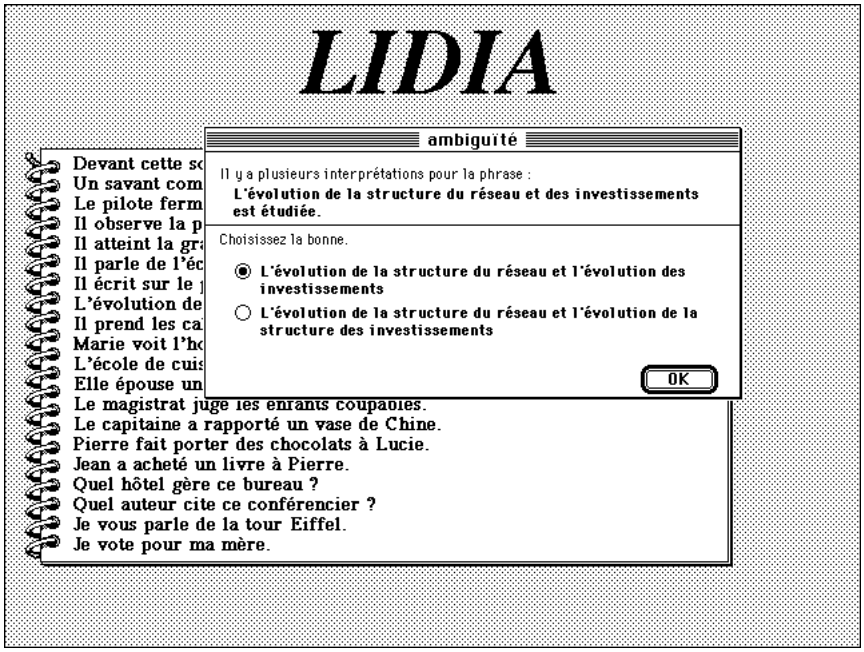
v. Dialogues

Chaque question donne lieu a un dialogue par menu, illustré par l'image d'écran ci-contre.

Ici, on demande à l'auteur de choisir, grâce à deux paraphrases, l'interprétation correcte dans un cas d'ambiguïté de "géométrie" (cf. infra).

Pour la clarification lexicale, on lui demande de choisir l'acception correcte d'un terme.

Pour la résolution d'anaphores, on lui demande de choisir le bon référent (en utilisant encore le paraphrasage).



Un dialogue de désambiguïsation

c. Encodage interne de textes multilingues dans un jeu de caractères universel

Le premier problème est de traiter des textes dans des langues écrites avec des systèmes d'écriture variés. Jusqu'aux années quatre-vingt, les systèmes informatiques n'offraient qu'un choix très réduit de jeux de caractères (comme romain sans diacritiques, mélange cyrillique/romain sans minuscules), de sorte qu'il était impératif d'utiliser des transcriptions. Dix ans après, presque tous les constructeurs d'ordinateurs commencèrent à offrir des jeux de caractères étendus et les systèmes d'exploitation "localisés".

Mac.OS-7.1, fondé sur Unicode (un sous-ensemble de la norme multiscript ISO-10646) et disponible depuis fin 1992, est le premier système d'exploitation réellement multiscript[29] : avec n'importe quel texteur utilisant le "Script Manager" standard, il est possible de composer un document contenant des parties dans presque toutes les langues d'Europe, ainsi qu'en arabe, japonais, chinois, etc.

[29] Mac.OS-7.1 n'est pas encore lui-même multilingue : bien que tous ses utilitaires soient indépendants des langues, chaque version distribuée n'a les messages et autres ressources spécifiques des langues que dans une langue.

Cependant, Mac.OS-7.1 est encore un cas unique[30], et le problème reste entier si l'on veut échanger du texte entre ordinateurs de différentes marques, ou simplement transmettre du texte via les réseaux télématiques. Par exemple, l'ASCII français n'est pas le même su un PC et sur un Macintosh. dans notre cas, le serveur de TA n'utilise même pas de l'ASCII, mais de l'EBCDIC.

Notre solution est d'utiliser des transcriptions romaines pour les représentaitons internes des textes, des grammaires et des dictionnaires. Une transcription consiste en un jeu de caractères et une méthode pour représenter le matériau textuel considéré (pas seulement les mots, mais aussi la structure logique, la mise en page, et éventuellement d'autres informations), en n'utilisant que ces caractères. Le jeu de caractères et la méthode à utiliser dépendent de l'importance relative accordée à la portabilité, à la lisibilité et à la compacité.

Pour la TA, nous avons depuis longtemps utilisé un jeu de caractères de transcription presque identique à celui de PL/I (ni minuscules ni diacritiques, mais seulement les majuscules simples, les signes de ponctuation usuels et quelques signes spéciaux). Cela donne une portabilité totale[31], aux dépens de la lisibilité.

Par exemple, `'*A!2 *NOE!4L , *MAC*ALLISTER VA AUX **USA'`
code : `'À Noël, MacAllister va aux USA'.`

Pour la partie micro de LIDIA-1, la lisibilité est plus importante, et nous utilisons un sous-ensemble plus grand de l'ISO-646, contenant les majuscules et les minuscules, mais pas les diacritiques. Les diacritiques sont représentés par des "séquences spéciales" introduites par "!" parce que, dans la documentation technique, les pièces sont souvent référencées par des identificateurs où lettres et chiffres sont mélangés (ex.: XA1). L'information relative à la structure ou à la mise en page du texte est représentée par des marques, ou "balises", dans l'esprit de SGML et de la TEI (`<parag>, <section>, <greek>,` etc.). On indique de façon analogue un changement de langue et/ou de système d'écriture (certaines langues en utilisent plus d'un).

d. Encodage d'informations linguistiques pouvant provenir d'une préédition (indirecte)

Les syntagmes figés spéciaux sont transformé en des occurrences spéciales, de façon à aider l'analyse et la traduction. Par exemple, "Cacher les bulles" devient `&FXN_Cacher_les_bulles,` qui peut être traité par une sous-grammaire morphologique appropriée.

Après la désambiguïsation lexicale, nous pourrions attacher à chaque occurrence le numéro du sens dans la BDLM, par exemple `glace.1` pour "glace à manger" et `glace.2` pour "miroir". Mais cela n'est pas très lisible et interdit la préédition directe. Nous permettons donc d'ajouter au numéro de sens un fragment de la définition qui donne la distinction, la "clé sémantique", d'habitude un autre mot ou terme, ce qui donne `glace.1=aliment` ou `glace.2=miroir`.

Dans le futur, LIDIA devrait permettre des présentations alternatives plus lisibles, utilisant par exemple les majuscules (et "*" pour indiquer les "vraies" majuscules) pour le texte lui-même, et les minuscules pour les annotations (par exemple, `GLACE.1=aliment)`. Comme en général plus d'une clé sémantique peut être associée à une même acception, par exemple `glace.1=a_manger` ou `glace.1=dessert`, il faudra aussi permettre à l'utilisateur d'entrer `glace=dessert,` et que le système consulte la BDLM, trouve l'acception de "glace" la plus proche de "dessert" est le sens 1, et transforme `glace=dessert` en `glace.1=dessert` ou même en `glace.1=aliment`.

Les annotations concernant les informations grammaticales relatives aux mots, comme la catégorie morphosyntaxique (verbe, nom…), le nombre, le genre, le temps, le mode, etc., sont attachées aux occurrences de façon analogue.

Le dernier type d'annotations concerne les structures concrètes (mmc ou umc). Pour délimiter les groupes syntagmatiques, nous utilisons des parenthèses spéciales, comme `{&rel…}` pour une proposition relative[32], ou simplement `{…}` si la catégorie syntagmatique n'est pas connue ou trop ésotérique pour les utilisateurs naïfs (par exemple, "groupe adjectival" ou "groupe cardinal"). Pour représenter un lien anaphorique, on attache au pronom une copie de son référent. En cas d'élision,

---

30 Le "documenteur" Star™ de Xerox a été le seul outil vraiment multilingue jusqu'à la sortie de WinText™ sur Macintosh en 1987, mais les systèmes d'exploitation sous-jacents étaient respectivement strictement monolingues, ou seulement localisables.

31 Dans beaucoups de pays, comme la Thaïlande, les minuscules romaines sont remplacées par les caractères locaux dans les terminaux bilingues.

32 Ou bien on ajoute "rel" à l'information attachée à sa tête, comme dans l'exemple suivant ("compose.&v,phvb").

on rajoute des occurrences "cachées" `(centrale .&eld=inertielle)`. D'autres informations grammaticales et sémantiques peuvent être attachées aux terminaux et aux non-terminaux.

Par exemple, "Devant cette somme, il ne rend pas sa glace immangeable pour autant" pourrait avoir la représentation intermédiaire suivante (avant d'avoir fini la désambiguïsation) :

```
{ {&grd,cause *devant.&vrb cette somme.2&nf , } il ne rend.2 pas=ne sa glace.1
immangeable pour autant }
```

qui lèverait l'ambiguïté sur "devant" (verbe/préposition) et "rendre" (vomir/restituer/faire devenir), d'où en anglais

"Owing this sum, he doesn't vomit his unedible ice cream for that reason", ou
"Owing this amount, he doesn't vomit his unedible ice cream for that reason",

plutôt que, entre autres,

"Owing this amount, he doesn't render his ice cream unedible for that reason",

ou

"Facing this summa [Opus Magnum], he doesn't give back his mirror for that reason".

Le point principal est que *le système d'annotations concerne plusieurs niveaux de description linguistique, mais est incomplet à chaque niveau,* parce qu'aucune notion non familière ne doit apparaître. Par exemple, "verbe" est une notion familière pour presque tout adulte instruit, mais pas "verbe modal". Au niveau des fonctions syntaxiques, "sujet", objet" et "complément" sont familiers, mais sans doute pas "attribut", "épithète", "tête" (ou "gouverneur"). Il en va de même au niveau des cas profonds.

Après désambiguisation, on obtient un nouveau texte annoté par projection systématique de la umc-structure (qui doit contenir des informations plus complètes et détaillées). On le garde à part, pour que les utilisateurs expérimentés puissent l'éditer directement. Voici à titre d'exemple un texte de 3 phrases contenu dans la pile de documentation d'Ariane.

```
Un processus de traduction en ARIANE-G5 se compose d'une suite de trois étapes
(analyse, transfert et génération). Chaque étape est constituée d'une suite de
différentes phases de traitement. Chaque phase est relative à l'emploi d'un LSPL
précis.
```

Et voici les présentations actuelle et future de la forme annotée :

```
{ { *un.&art processus.&n,suj { de.&prep traduction.&n,comp { en.&prep **ariane-
g5.&np,comp } } } se.&refl compose.&v,phvb { d'une.&art suite.&n,obj1 { de.&prep
trois.&card e!1tapes.&n,comp { (.&lp analyse.&n,app ,.&ponc transfert.&n,coord
et.&cjcoord ge!1ne!1ration.&n,coord ).&rp } } } ..&ponc } { { *chaque.&art
é!1tape.&n,suj } est.&v,aux constitue!1e.&v,phvb { d'.&prep une.&art
suite.&n,comp { de.&prep { diffe!1rentes.&adj,epit } phases.&n,comp { de.&prep
traitement.&n,comp } } } ..&ponc } { { *chaque.&art phase.&n,suj } est.&v,phvb {
relative.&adj,atsubj { a!2.&prep l'.&art emploi.&n,obj1 { d'.&prep un.&art
**lspl.&np,comp { pre!1cis.&adj,epit } } } } ..&ponc }
```

```
{ { *UN.&art PROCESSUS.&n,suj { DE.&prep TRADUCTION.&n,comp { EN.&prep **ARIANE-
G5.&np,comp } } } SE.&refl COMPOSE.&v,phvb { D'UNE.&art SUITE.&n,obj1 { DE.&prep
TROIS.&card E!1TAPES.&n,comp { (.&lp ANALYSE.&n,app ,.&ponc TRANSFERT.&n,coord
ET.&cjcoord GE!1NE!1RATION.&n,coord ).&rp } } } ..&ponc } { { *CHAQUE.&art
E!1TAPE.&n,suj } EST.&v,aux CONSTITUE!1E.&v,phvb { D'.&prep UNE.&art
SUITE.&n,comp { DE.&prep { DIFFE!1RENTES.&adj,epit } PHASES.&n,comp { DE.&prep
TRAITEMENT.&n,comp } } } ..&ponc } { { *CHAQUE.&art PHASE.&n,suj } EST.&v,phvb {
RELATIVE.&adj,atsubj { A!2.&prep L'.&art EMPLOI.&n,obj1 { D'.&prep UN.&art
**LSPL.&np,comp { PRE!1CIS.&adj,epit } } } } ..&ponc }
```

Le codage interne est bien plus qu'une question technique de second ordre, comme on le pense souvent. Sa définition n'est pas seulement très importante pour les développeurs de grammaires et de dictionnaires, mais, pour la concevoir de façon cohérente, il faut comprendre le fonctionnement interne d'un système de TA, et, dans notre cas, satisfaire la *contrainte d'accessibilité* (par des utilisateurs naïfs). C'est aussi un défi pour des linguistes habitués à faire des distinctions très subtiles que de devoir bâtir des sytèmes n'utilisant que des informations "rustiques" obtenables de non-spécialistes. Or, cela est nécessaire pour que la TAFD "pour tous" puisse réussir.

Enfin, notons que cette idée (de préédition indirecte et/ou directe) est aussi utilisée dans d'autres projets, comme le projet LMT d'IBM [10][33].

IV.3. Dialogues de désambiguïsation

Dans la maquette LIDIA-1, les dialogues sont conduits uniquement à l'écran. Nous espérons expérimenter dans le futur l'introduction d'autres média, et notamment de synthèse vocale.

Pour ne pas surcharger les utilisateurs avec des choses nouvelles à apprendre, nous avons préféré nous en tenir à des dialogues par menu. Par exemple, nous avions pensé proposer un outil de manipulation graphique des structures concrètes, mais des utilisateurs potentiels et des ergonomes ont trouvé cela plus difficile que de choisir entre des paraphrases textuelles avec mise en relief des différences.

Enfin, une suggestion de [49] était de retarder toutes les interactions jusqu'au transfert. Compte tenu de nos objectifs de grande couverture et de "rusticité", il nous a paru préférable de résoudre dès que possible les ambiguïtés impossibles ou très difficiles à résoudre automatiquement par l'un des processus ultérieurs.

3.1. Classification des ambiguïtés

Rappelons que les *ambiguïtés lexicales* concernent non seulement la classique polysémie de termes (par exemple, "diplôme" se traduit par "diploma" ou "degree"), mais aussi les *ellipses lexicales*[34]. Dans les deux cas, LIDIA construit un menu avec les choix possibles, rangés dans l'ordre de leurs poids courants.

Les autres ambiguïtés présentes dans la mmc-structure sont partitionnées en trois classes :

- Il y a *ambiguïté de classe syntaxique* si deux classes syntaxiques ou plus sont affectées à une occurrence dans l'ensemble des solutions produites par l'analyseur.
- Il y a *ambiguïté de géométrie* si les structures arborescentes de deux solutions donnant les mêmes valeurs de classes aux occurrences sont différentes.
- Il y a *ambiguïté de fonction syntaxique et/ou de relation sémantique* si l'analyseur produit deux structures de mêmes classes et de même géométrie, différant donc par l'information portée par certains nœuds non-terminaux.

Si plusieurs problèmes apparaissent dans la même phrase (énoncé), nous utilisons la stratégie suivante :

1. déterminer la segmentation correcte en groupes simples.
2. déterminer les arguments et les circonstants de chaque prédicat commun à deux solutions.
3. déterminer les relations syntaxiques et sémantiques correctes entre les groupes simples.

C'est l'ordre naturel que suivent les humains en cas de problème, et il est important que les utilisateurs comprennent facilement et clairement ce que fait le système. Au niveau du système, cela revient à résoudre dans l'ordre les ambiguïtés de classe, de géométrie, et de relations.

3.2. Production des dialogues

a. Raffinement des classes et schémas de problèmes

Nous ne pouvons pas proposer une méthode de désambiguïsation unique pour chaque classe d'ambiguïté, en particulier à cause des croisements multiples. D'autre part, nous nous sommes donné la contrainte de produire des menus proposant des choix entre des paraphrases. En examinant

---

[33] «The user can mark the input string selectively with brackets <…> (to any degree) to force parsing choice and desambiguate the input. "User" in this context can also apply tools (such as the interactive disambiguator) which may introduce such marks in their output.»

[34] Supposons qu'un texte parle d'un vaisseau spatial contenant une "centrale électrique" ("electric plant") et une "centrale inertielle" ("inertial guidance system"). La forme complète est souvent remplacée par la forme élidée ("centrale"). Bien qu'il soit crucial de désambiguïser pour traduire correctement (par les formes élidées correspondantes, "plant" ou "system"), on ne connait aucune solution automatique. Une occurrence donnée de "centrale" peut être ou ne pas être une élision. Et si elle l'est, il est encore plus difficile de rechercher un candidat pour la forme complète dans un hypertexte que dans un texte usuel.

un grand nombre de configurations d'ambiguïtés, nous sommes arrivés à distinguer huit types de problèmes, pour lesquels on sait produire des paraphrases "désambiguïsantes" :

A. ambiguïté de classe syntaxique:

1. ambiguïté de classe syntaxique sans groupes coordonnés ambigus

Le pilote ferme la porte:	*The firm pilot carries her.* *The pilot shuts the door.*

2. ambiguïté de classe syntaxique associée à un groupe coordonné

Il regarde la photo et la classe:	*He looks at the photograph and the class.* *He looks at the photograph and files it.*

B. ambiguïté de géométrie:

3. ambiguïté de structure argumentaire du verbe

Il parle de l'école de cuisine:	*He talks about the cooking school.* *He talks from the cooking school.* *He talks from the school about cooking.*

4. ambiguïté de coordination

Il prend des crayons et des cahiers noirs:	*He takes pencils and black notebooks.* *He takes black pencils and black notebooks.*

5. ambiguïté de subordination

L'école de cuisine lyonnaise est fermée:	*The lyonnaise cooking school is closed.* *The school of lyonnaise cooking is closed.*

C. ambiguïté de fonction syntaxique et/ou de relation logico-sémantique

6. ambiguïté de relation logico-sémantique

Pierre fait porter des chocolats à Lucie:	*Pierre is having chocolates sent to Lucie.* *Pierre is having chocolates sent with Lucie.*

7. ambiguïté sur les positions d'arguments d'un verbe transitif direct

Quel auteur cite ce conférencier:	*Which author is this lecturer quoting?* *Which lecturer is this author quoting?*

8. ambiguïté de fonction syntaxique

Il parle de la tour Eiffel:	*He is talking about the Eiffel Tower.* *He is talking from the Eiffel Tower.*

b. Description des opérateurs de base

Pour produire ces paraphrases, nous utilisons les six opérateurs de base suivants, qui opèrent sur des fragments de la mmc-structure :

`Generate` produit une forme fléchie à partir d'une unité lexicale.

`Distribute` distribue une occurrence ou un groupe d'occurrences sur d'autres groupes d'occurrences, et établit un lien entre les résultats des distributions locales.

```
ex: Distribute(A, B C, D, 1, 2, ou, 1, 3) -> A B C ou A D
```

`Permute` change l'ordre d'une liste d'occurrences.

```
ex: Permute( A, B, C D, E, F, G, 1, 2, 5, 6) -> A B F G
```

`Substitute` substitue à un terme un autre terme recherché dans la BDLM en fonction de conditions.

`Project` appliqué à un nœud de l'arbre, projette la classe syntaxique, et, en fonction de cette classe, certaines informations contenues dans la BDLM.

`Bracket` met entre parenthèses la sous-liste d'une liste d'occurrences comprise entre deux bornes.

c. Discussion

De nombreux projets, en particulier [17-19, 24, 31, 49, 72, 75], ont expérimenté diverses stratégies de désambiguïsation. Si aucun système utilisable en pratique n'en est sorti, c'est à notre avis surtout parce que le processus de désambiguïsation apparaît comme une boîte noire d'où sortent des questions assez imprévisibles et ésotériques. Nous avons donc consciemment préféré la

rusticité à la sophistication, et cherché à construire un système de structure simple, posant des questions compréhensibles par tout bachelier.

Pour l'instant, nous utilisons une stratégie de désambiguïsation "câblée" dans les programmes de LIDIA-1. Or, il est impossible d'affirmer que c'est "la meilleure" (et nous doutons fort qu'il en existe une). En tout état de cause, les utilisateurs devraient pouvoir choisr entre divers modes de désambiguïsation. C'est pourquoi nous travaillons sur un outil qui permettrait à des linguistes et des ergonomes de définir et d'expérimentr des stratégies de désambiguïsation variées, dans divers contextes (type d'utilisateur, autre média…).

Une première étape a été la construction d'un ensemble d'opérateurs de base pour la génération de paraphrases. La suivante devrait permettre aux linguistes de définir eux-mêmes les schémas de problèmes. Un but plus lointain serait d'offrir des outils de "programmation ambiguë", permettant de décrire les types d'ambiguïtés et les stratégies de désambiguïsation (y compris le recours à l'auteur ou à une ontologie) à l'intérieur des grammaires du système de TA lui-même.

Conclusion

Le concept de TAFD cristallise des idées venant de systèmes et de recherches antérieurs (critique textuelle, TAFL interactive, TAFL avec préédition, TAFC avec augmentation, langages contrôlés, sous-langages…). Cependant, la contrainte d'interagir avec un auteur n'ayant aucune connaissance des langues cibles ni de la linguistique permet de parler d'un nouveau paradigme.

Comme nous l'avons dit, les paradigmes ne sont pas exclusifs, et ont chacuns leurs domaines d'emploi, qui peuvent se recouvrir plus ou moins. Il s'agit en effet de paradigmes techniques, et non de théories scientifiques. Mais la TAO est une technologie scientifique, qui bénéficie de temps en temps de concepts innovateurs provenant des sciences qui la sous-tendent, en en tirant des progrès incrémentaux, et qui peut aussi proposer aux théoriciens des problèmes intéressants.

En TAFD, les progrès incrémentaux dont nous parlons pourraient venir de la mise en œuvre d'approches intermédiaires nouvelles, comme :

- le transfert multiniveau par acceptions interlingues ;
- le "langage guidé" (préférences lexicales, styles d'énoncés, genres de textes), avec la combinaison de techniques symboliques et numériques ;
- l'accessibilité et la rusticité des connaissances linguistiques, permettant à terme l'enrichissement par l'utilisateur.

Parmi les problèmes nouveaux posés par la TAFD, les plus importants (et difficiles) nous semblent être :

- le typage automatique de textes et d'énoncés, ainsi que le guidage de l'auteur pour la partie interactive ;
- la conception (et la validation) de techniques de désambiguïsation interactive multimédia ;
- l'enrichissement du système par l'utilisateur, combiné avec un "réglage" automatique.

Il ne faut bien sûr pas confondre une étape d'étude et de maquettage avec la réalisation d'un produit. D'abord, il n'est pas exclu que la recherche bute sur des obstacles non prévus ou sous-estimés. Cela est arrivé plus d'une fois dans l'histoire de la TAO. Ensuite, dans le cas de la TAFD "pour tous", le facteur d'échelle sera très grand (de l'ordre de 1000 pour les bases de données lexicales) : même si la plupart des problèmes étaient résolus au niveau de LIDIA-1, rien ne dit que les méthodes de développement de bases de données linguistiques auront assez progressé pour qu'un système opérationnel de TAFD destiné au grand public soit techniquement réalisable et économiquement viable.

Cependant, l'avancement actuel des connaissances et des techniques nous permet d'être raisonnablement optimistes. Les enjeux scientifiques sont importants, et les enjeux économiques et culturels encore plus. En effet, si elle est faisable, la TAFD permettra de résoudre, au moins en partie, le problème crucial rencontré par les promoteurs des langues nationales en général, et par les défenseurs de la francophonie en particulier, à savoir l'impossibilité absolue, pour le plus grand nombre, d'écrire dans sa langue et d'être traduit dans d'autres dans un délai raisonnable et avec une garantie suffisante de qualité.

Remerciements

Je tiens à remercier ici H. Blanchon, qui a réalisé la version initiale des figures et images d'écran de la quatrième partie, ainsi que la programmation non linguistique de LIDIA-1 sur le Macintosh. Merci également à tous les collègues du GETA qui ont participé à ce maquettage : M. Axtmeyer, E. Blanc, N. Denos, J.-Ph. Guilbaud, P. Guillaume, M. Lafourcade, D. Levenbach, N. Nédobejkine, F. Peccoud, B. Roudaud, M. Quézel-Ambrunaz, G. Sérasset, ainsi qu'à WinSoft et à Machina Sapiens, qui nous ont permis d'utiliser leurs logiciels gracieusement. Merci enfin (last, but not least), à A. Clas et P. Bouillon, pour avoir conçu cet ouvrage et m'avoir aimablement invité à y présenter le paradigme de la TA fondée sur le dialogue.

-o-o-o-o-o-o-o-o-o-o-

Références bibliographiques

[1-76]

[1] **Blanchon H. (1992)** *A Solution to the Problem of Interactive Disambiguation.* Proc. COLING-92, Nantes, 23-28 July 1992, C. Boitet, ed., vol. 4/4, 1233-1238.

[2] **Blanchon H., Guilbaud J. p. & Nédobejkine N. (1992)** *LIDIA : the disambiguation process — le processus de désambiguïsation.* Exposition COLING-92, Nantes, 23-28 juillet 1992, Pile HyperCard.

[3] **Boitet C. (1992)** *TAO personnelle et promotion des langues nationales.* Turjumān, revue de traduction et d'interprétation, École Supérieure Roi Fahd de Traduction, Université Abdelmalek Essaâdi, Tanger, **1**/1, avril 1992, 35—49.

[4] **Cowie J., Guthrie J. & Guthrie L. (1992)** *Lexical Disambiguation Using Simulated Annealing.* Proc. COLING-92, Nantes, 23-28 July 1992, C. Boitet, ed., vol. 1/4, 359-365.

[5] **Hutchins W. J. & Somers H. L. (1992)** *An Introduction to Machine Translation.* Academic Press, Harcourt Brace Jovanovich, 362.

[6] **Nyberg E. H. & Mitamura T. (1992)** *The KANT system: Fast, Accurate, High-Quality Translation in Practical Domains.* Proc. COLING-92, Nantes, 23-28 July 92, C. Boitet, ed., ACL, vol. 3/4, 1069—1073.

[7] **Phan H. K. & Boitet C. (1992)** *Multilinguization of an editor for structured documents. Application to a trilingual dictionary.* Proc. COLING-92, Nantes, juillet 1992, C. Boitet, ed., ACL, vol. 3/4, 966—971.

[8] **Wehrli E. (1992)** *The IPS System.* Proc. COLING-92, Nantes, 23-28 July 1992, C. Boitet, ed., vol. 3/4, 870-874.

[9] **Hirakawa H., Nogami H. & Amano S.-Y. (1991)** *EJ/JE Machine Translation System AS-TRANSAC - Extension toward Personalization.* Proc. MTS-III (MT Summit), Boston, 1-4 July 1991, vol. 1/1, 73-80.

[10] **Rimon M., McCord M. C., Schwall U. & Martínez p. (1991)** *Advances in Machine Translation Research in IBM.* Proc. MTS-III (MT Summit), Boston, 1-4 July 1991, 11-18.

[11] **Wehrli E. (1991)** *Pour une approche interactive au problème de la traduction automatique.* Proc. Colloque "L'environnement traductionnel. La station de travail du traducteur de l'an 2001", Mons, 25-27 avril 1991, APELF & UREF, 59-68.

[12] **Blanc E. & Boitet C., ed. (1990)** *DBMT-90, Post-COLING Seminar on Dialogue-Based MT (Machine Translation of/with dialogues). Organization, General Spirit, Final Program & Content of the 8 Sessions.* Le Sappey, 26-28 août 1990, 328 p.

[13] **Blanchon H. (1990)** *LIDIA-1 : Un prototype de TAO personnelle pour rédacteur unilingue.* Proc. X-èmes Journées sur les systèmes experts et leurs applications. Conférence spécialisée "Le traitement automatique des langues naturelles et ses applications", Avignon, 28 mai-1 juin 1990, EC2, 51-60.

[14] **Boitet C. (1990)** *Multilingual Machine Translation does not have to be saved by Interlingua.* Proc. MMT'90, Tokyo, 5-6 Nov. 1990, 2 p.

[15] **Boitet C. (1990)** *Towards Personal MT : on some aspects of the LIDIA project.* Proc. COLING-90, Helsinki, 20-25 août 1990, H. Karlgren, ed., ACL, vol. 3/3, 30-35.

[16] **Bourbeau L. (1990)** *Élaboration et mise au point d'une méthodologie d'évaluation linguistique de systèmes de Traduction Assistée par Ordinateur.* Rapport final de contrat, Secrétariat d'État du Canada, Ottawa, mars 1990, 203 p.

[17] **Brown R. D. & Nirenburg S. (1990)** *Human-Computer Interaction for Semantic Disambiguation.* Proc. COLING-90, Helsinki, 20-25 août 1990, H. Karlgren, ed., ACL, vol. 3/3, 42-47.

[18] **Huang X. M. (1990)** *A Machine Translation System for the Target Language Inexpert.* Proc. COLING-90, Helsinki, 20-25 Aug. 1990, H. Karlgren, ed., ACL, vol. 3/3, 364-367.

[19] **Maruyama H., Watanabe H. & Ogino S. (1990)** *An Interactive Japanese Parser for Machine Translation.* Proc. COLING-90, Helsinki, 20-25 Aug. 1990, H. Karlgren, ed., ACL, vol. 2/3, 257-262.

[20] **Nirenburg S. & Goodman K. (1990)** *Treatment of Meaning in MT systems.* Proc. ROCLing-III, Taipeh, 20—22 Aug. 1990, 83—101.

[21] **Rolling L. (1990)** *Trends of Multilingual Machine Translation in Europe.* Proc. MMT'90, Tokyo, 5-6 Nov. 1990, 2 p.

[22] **Somers H. L., Tsujii J.-I. & Jones D. (1990)** *Machine Translation without a source text.* Proc. COLING-90, Helsinki, 20-25 Aug. 1990, H. Karlgren, ed., ACL, vol. 3/3, 271-276.

[23] **Veronis J. & Ide N. (1990)** *Word Sense Disambiguation with Very Large Neural Networks Extracted from Machine-Readable Dictionaries.* Proc. COLING-90, Helsinki, 18—25 Aug. 1990, H. Karlgren, ed., ACL, 389—394.

[24] **Wehrli E. (1990)** *STS: An Experimental Sentence Translation System.* Proc. COLING-90, Helsinki, 20-25 Aug. 1990, H. Karlgren, ed., ACL, vol. 1/3, 76-78.

[25] **Boitet C. (1989)** *Motivation and Architecture of the LIDIA Project.* Proc. MTS-II (MT Summit), Munich, 16-18 août 1989, 12 p.

[26] **Boitet C. (1989)** *Speech Synthesis and Dialogue Based Machine Translation.* Proc. ATR Symp. on Basic Research for Telephone Interpretation, Kyoto, December 1989, 12 p.

[27] **Brown R. D. (1989)** *Augmentation.* Machine Translation, 4, 1299-1347.

[28] **JEIDA (1989)** *A Japanese view of Machine Translation in light of the considerations and recommendations reported by ALPAC, USA.* Japanese Electronic Industry Development Association, Tokyo, 197 p.

[29] **Nirenburg S. (1989)** *Knowledge-Based Machine Translation.* Machine Translation, 4, 5-24.

[30] **Nirenburg S. & al. (1989)** *KBMT-89 Project Report.* Center for Machine Translation, Carnegie Mellon University, Pittsburg, April 1989, 286 p.

[31] **Sadler V. (1989)** *Working with analogical semantics : Disambiguation technics in DLT.* T. Witkam, ed., Distributed Language Translation (BSO/Research), Floris Publications, Dordrecht, Holland, 256 p.

[32] **Wood M. M. (1989)** *Japanese for speakers of English: The UMIST/Sheffield Machine Translation Project.* In "Recent Developments and Applications of Natural Language Processing", J. Peckham, ed., Kogan Page ltd, London, 56-64.

[33] **Abbou A., ed. (1988)** *Traduction Assistée par Ordinateur. Actes du séminaire international sur la TAO et dossiers complémentaires.* Observatoire des Industries de la Langue (OFIL), Paris, mars 1988, 234 p.

[34] **Boitet C. (1988)** *Hybrid Pivots using m-structures for multilingual Transfer-Based MT Systems.* Jap. Inst. of Electr., Inf. & Comm. Eng., June 1988, NLC88-3, 17—22.

[35] **Boitet C. (1988)** *PROs and CONs of the pivot and transfer approaches in multilingual Machine Translation.* Proc. Int. Conf. on "New directions in Machine Translation", Budapest, 18–19 August 1988, BSO, 13 p.

[36] **Boitet C. (1988)** *Representation and Computation of Units of Translation for Machine Interpretation of Spoken Texts.* Comp. & AI, 6, 505—546 (and TR-I-0035, ATR, Osaka).

[37] **Boitet C. (1988)** *Software and lingware engineering in modern M(A)T systems.* In "Handbook for Machine Translation", Bátori, ed., Niemeyer.

[38] **Boitet C. & Zaharin Y. (1988)** *Representation trees and string-tree correspondences.* Proc. COLING-88, Budapest, 22–27 Aug. 1988, D. Várgha, ed., ACL, 59—64.

[39] **Chandioux J. (1988)** *10 ans de METEO (MD).* In "Traduction Assistée par Ordinateur. Actes du séminaire international sur la TAO et dossiers complémentaires", A. Abbou, ed., Observatoire des Industries de la Langue (OFIL), Paris, mars 1988, 169—173.

[40] **Coutaz J. (1988)** *Interface Homme-ordinateur : Conception et Réalisation.* Thèse d'Etat, Université Josepf Fourier, Grenoble.

[41] **Ducrot J.-M. (1988)** *Le système TITUS IV.* In "Traduction Assistée par Ordinateur. Actes du séminaire international sur la TAO et dossiers complémentaires", A. Abbou, ed., Observatoire des Industries de la Langue (OFIL), Paris, mars 1988, 55—71.

[42] **Lehrberger J. & Bourbeau L. (1988)** *Machine Translation. Linguistic characteristics of MT systems and general methodology of evaluation.* John Benjamins, 240 p.

[43] **Richardson S. D. & Braden-Harder L. C. (1988)** *The experience of developing a large-scale natural language text processing system: CRITIQUE.* Proc. 2nd conference on Applied Natural Language Processing, 1988.

[44] **Vasconcellos M. & León M. (1988)** *SPANAM and ENGSPAM : Machine Translation at the Pan American Health Organization.* In "Machine Translation systems", J. Slocum, ed., Cambridge Univ. Press, 187—236.

[45] **Vauquois B. (1988)** *BERNARD VAUQUOIS et la TAO, vingt-cinq ans de Traduction Automatique, ANALECTES. BERNARD VAUQUOIS and MT, twenty-five years of MT.* C. Boitet, ed., Ass. Champollion & GETA, Grenoble, 700 p.

[46] **Weaver A. (1988)** *Two Aspects of Interactive Machine Translation.* In "Technology as Translation Strategy", M. Vasconcellos, ed., State University of New York at Binghamton, Binghamton, 116-123.

[47] **Wood M. M. G. & Chandler B. (1988)** *Machine Translation For Monolinguals.* Proc. COLING-88, Budapest, 22-27 Aug. 1988, D. Várgha, ed., 760—763.

[48] **Zajac R. (1988)** *Interactive Translation : a new approach.* Proc. COLING-88, Budapest, D. Várgha, ed., Aug. 1988.

[49] **Chandler B., Holden N., Horsfall H., Pollard E. & McGee Wood M. (1987)** *N-tran Final Report.* Alvey Project, 87/9, CCL/UMIST, Manchester.

[50] **Desclés J.-P. (1987)** *Sémantique.* Technologos, LISH-CNRS, printemps 1987.

[51] **Sigurdson J. & Greatex R. (1987)** *MT of on-line searches in Japanese Data Bases.* RPI, Lund Univ., 124 p.

[52] **Hutchins W. J. (1986)** *Machine Translation: Past, Present, Future.* E. Horwood, ed., John Wiley & Sons, Chichester, England, 382 p.

[53] **Kittredge R. (1986)** *Analyzing Language in Restricted Domains.* In "Sublanguage Description and Processing", R. Grishman & R. Kittredge, ed., Lawrence Erlbaum, Hillsdale, New-Jersey.

[54] **Tomita M. (1986)** *Sentence Disambiguation by asking.* Computers and Translation, **1**/1, 39-51.

[55] **Whitelock p. J., Wood M. M., Chandler B. J., Holden N. & Horsfall H. J. (1986)** *Strategies for Interactive Machine translation : the experience and implications of the UMIST Japanese project.* Proc. COLING-86, Bonn, 25-29 août 1986, IKS, 25-29.

[56] **Zaharin Y. (1986)** *Strategies and heuristics in the analysis of a natural language in Machine Translation.* Ph.D. thesis, Universiti Sains Malaysia, Penang (research conducted in cooperation with GETA, Grenoble).

[57] **Boitet C. (1985)** *Traduction (assistée) par Ordinateur: ingéniérie logicielle et linguicielle.* Proc. Colloque RF&IA, Grenoble, AFCET.

[58] **Carbonnell J. G. & Tomita M. (1985)** *New Approaches to Machine Translation.* Proc. TMI-85 (Conf. on Theoretical and Methodological Issues in Machine Translation of Natural Languages), Hamilton, N.Y.,14-16 Aug. 1985, S. Nirenburg, ed., 59-74.

[59] **Gerber R. & Boitet C. (1985)** *On the design of expert systems grafted on MT systems.* Proc. TMI-85 (Conf. on Theoretical and Methodological Issues in Machine Translation of natural language), Hamilton, N.Y., S. Nirenburg, ed., Colgate Univ., 116-134.

[60] **Richardson S. D. (1985)** *Enhanced Text Critiquing Using a Natural Language Parser : the CRITIQUE system.* RC 11332, IBM, Thomas J. Watson Research Center, Yorktown Heights.

[61] **Tomita M. (1985)** *Feasibility Study of Personal/Interactive Machine Translation System.* Proc. TMI-85 (Conf. on Theoretical and Methodological Issues in Machine Translation of Natural Languages), Hamilton, N.Y., 14-16 Aug. 1985, S. Nirenburg, ed., Colgate Univ., vol. 1/1, 289-297.

[62] **Vauquois B. & Boitet C. (1985)** *Automated translation at Grenoble University.* Comp. Ling., **11**/1, January-March 85, 28—36.

[63] **Vauquois B. & Chappuy S. (1985)** *Static grammars: a formalism for the description of linguistic models.* Proc. TMI-85 (Conf. on theoretical and metholodogical issues in the Machine Translation of natural languages), Colgate Univ., Hamilton, N.Y., Aug. 1985, S. Nirenburg, ed., 298-322.

[64] **Isabelle p. & Bourbeau L. (1984)** *TAUM-AVIATION: its technical features and some experimental results.* Comp. Ling., **11**/1, 18 27.

[65] **Tomita M. (1984)** *Disambiguating Grammatically Ambiguous Sentences by Asking.* Proc. COLING-84, Stanford, 2-6 juillet 1984, ACL, 476-480.

[66] **Kittredge R. (1983)** *Sublanguage — Specific Computer Aids to Translation — a survey of the most promising application areas.* Contract n° 2-5273, Université de Montréal et Bureau des Traductions, mars 1983, 95 p.

[67] **Ducrot J.-M. (1982)** *TITUS IV.* In "Information research in Europe. Proc. of the EURIM 5 conf. (Versailles)", P. J. Taylor, ed., ASLIB, London.

[68] **Heidorn G. E., Jensen K. & Miller L. A. (1982)** *The EPISTLE Text-Critiquing System.* IBM System Journal, **21**/1, 305-326.

[69] **Melby A. K. (1982)** *Multi-Level Translation Aids in a Distributed System.* Proc. COLING-82, Prague, 5-10 juillet 1982, vol. 1/2, 215-220.

[70] **Zemb J.-M. (1982)** *Les occurrences phématiques, rhématiques et thématiques des archilexèmes "modaux". La notion sémantico-logique de modalité.* In "Recherches Linguistiques VIII", Université de Metz & Klincksieck, 75—116.

[71] **Chandioux J. & Guérard M.-F. (1981)** *METEO: un système à l'épreuve du temps.* META, 1, 17—22.

[72] **Melby A. K. (1981)** *Translators and Machines - Can they cooperate ?* META, **26**/1, 23-34.

[73] **Kay M. (1980)** *The Proper Place of Men and Machines in Language Translation.* Research Report, CSL-80-11, Xerox, Palo Alto Research Center, Oct. 1980.

[74] **Melby A. K., Smith M. R. & Peterson J. (1980)** *ITS : An Interactive Translation System.* Proc. COLING-80, Tokyo, 30 septembre-4 octobre 1980, M. Nagao, ed., 424-429.

[75] **Kay M. (1973)** *The MIND system.* In "Courant Computer Science Symposium 8: Natural Language Processing", R. Rustin, ed., Algorithmics Press, Inc., New York, 155-188.

[76] **Wilks Y. (1973)** *An Artificial Intelligence approach to Machine Translation.* In "Computer Models of Thought and Language", Shank & Colby, ed., Freeman & Co, 114—151.

-o-o-o-o-o-o-o-o-o-o-