

Un analyseur syntaxique du français

Henri Labesse

Volume 14, numéro 2, 1985

Linguistique et informatique

URI : <https://id.erudit.org/iderudit/602540ar>

DOI : <https://doi.org/10.7202/602540ar>

[Aller au sommaire du numéro](#)

Éditeur(s)

Université du Québec à Montréal

ISSN

0710-0167 (imprimé)

1705-4591 (numérique)

[Découvrir la revue](#)

Citer cet article

Labesse, H. (1985). Un analyseur syntaxique du français. *Revue québécoise de linguistique*, 14(2), 103–117. <https://doi.org/10.7202/602540ar>

UN ANALYSEUR SYNTAXIQUE DU FRANÇAIS

Henri Labesse

Cet article a pour but de présenter un analyseur syntaxique du français et d'en discuter les principales options, tant linguistiques qu'informatiques. Toutes les informations sur la langue sont enregistrées dans deux fichiers, le lexique et la grammaire. L'analyseur lui-même est un programme qui utilise ces fichiers pour analyser le texte selon le principe des grammaires en chaînes (Salkoff 1973).

1. Structure de l'analyseur

Le rôle du lexique est d'associer chaque forme de mot de la langue à une ou plusieurs chaînes, qui en précisent la fonction grammaticale et ne retiennent que celle-ci. Ainsi, dans un lexique rudimentaire, les formes *chat* et *balles* pourront être associées à la chaîne Nom, alors que *le* le sera aux chaînes Article et Pronom.

Le rôle de la grammaire est de recenser toutes les combinaisons possibles entre les chaînes issues du lexique ou de combinaisons antérieures. Voici, pour fixer les idées, un exemple très simple de lexique-grammaire capable de reconnaître certains groupes nominaux :

Lexique :	
le : Art, Pronom	la : Art, Pronom
petit : Adj	beau : Adj
chat : Nom	balle : Nom
Grammaire :	
Adjx = Adj	Adjx = Adj Adjx
GN = Nom	GN = Adjx Nom
GN = Art Nom	GN = Art Adjx Nom

Tableau 1

Le rôle de l'analyseur est de reconnaître les segments du texte analysables en chaînes. Ainsi dans la phrase *le beau petit chat joue avec la balle* il doit trouver les segments :

le : Art, Pronom
beau : Adj, Adjx
petit : Adj, Adjx
beau petit : Adjx
chat : Nom, GN
petit chat : GN
beau petit chat : GN
le beau petit chat : GN
la : Art, Pronom
balle : Nom, GN
la balle : GN

Cet exemple montre que la notion de chaîne est purement technique. C'est une étiquette que l'analyseur attribue à tout segment qu'il reconnaît, en particulier aux formes, et qu'il utilise ensuite pour déterminer, en consultant la grammaire, à quel autre segment il peut être combiné. Cette notion technique paraît bien adaptée pour décrire les structures syntaxiques dans l'optique décrite par Boons, Guillet, Leclerc (1976).

Le fait que le fichier de grammaire ne contienne pas de formes mais uniquement des chaînes ne diminue pas les possibilités de l'analyseur, mais le simplifie et améliore son efficacité. L'indépendance de l'analyseur vis-à-vis de la langue permet d'utiliser le même programme avec des lexiques-grammaires différents, pour changer de langue, ou pour coder des grammaires partielles, comme c'est le cas plus loin pour les déterminants.

L'analyseur lui-même comporte deux programmes écrits en langage PL/1. Le premier, GRALEX, transforme le lexique et la grammaire en trois fichiers : le lexique codé, la grammaire codée, et le fichier des noms de chaînes. Le deuxième, ANAX, analyse les textes; il répond au schéma de la figure 1.

Le module (procédure) ANAX procède aux initialisations avant d'appeler le module CHAGRAM, qui reconnaît les segments. Il appelle à son tour le module TEXT à chaque fois qu'il veut lire un mot du texte, et ensuite LEXI pour trouver ce mot dans le lexique. Il appelle aussi GRAM pour lire la grammaire. Une fois reconnus, les segments sont passés au

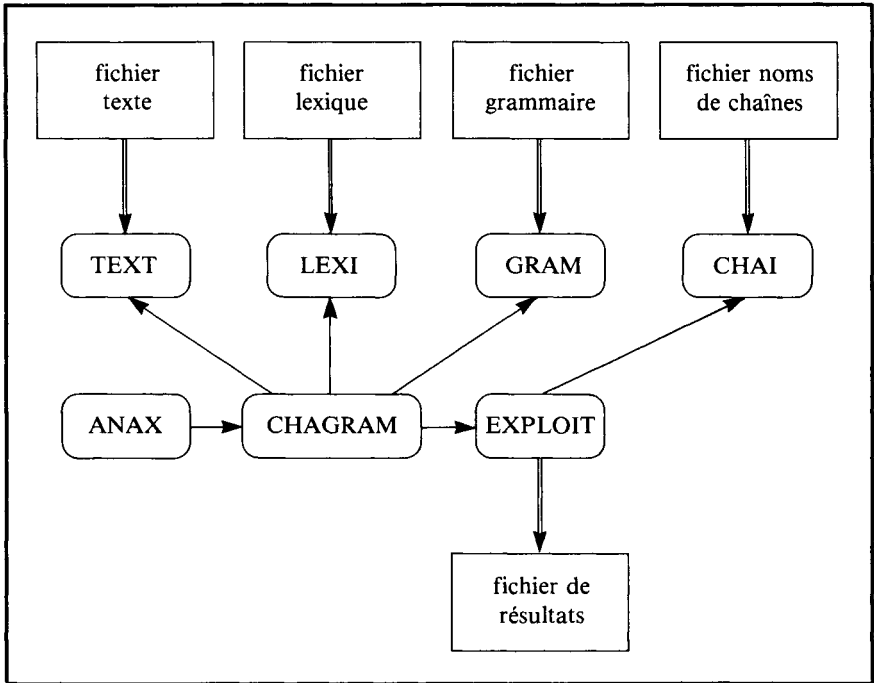


Figure 1

module EXPLOIT. Actuellement, celui-ci imprime tous les segments d'un type déterminé, par exemple GN, mais il peut être changé en fonction des buts de l'analyse.

La plupart des grammaires sont capables de reconnaître des segments qui ne sont pas a priori de longueur bornée. Cela a pour conséquence qu'il est très difficile, voire impossible, pour l'analyseur, de déterminer si un segment peut être prolongé ou non avant d'avoir traité la totalité du texte. Cela pose un problème informatique dans la mesure où on ne peut pas, à chaque segment, analyser tout le texte pour essayer de le prolonger. Or, à l'heure actuelle, on ne dispose pas de critère linguistique simple et sûr pour caractériser les fins de phrase. De plus, cela ne semble guère possible sans un minimum d'analyse. C'est pourquoi ce sont les chaînes elles-mêmes qui définiront le découpage en phrases, en tenant compte éventuellement de la ponctuation. La seule restriction imposée par l'analyseur est qu'il ne reconnaît pas les segments dépassant une longueur donnée, par exemple 1000 mots.

2. Principes de constitution de la grammaire

2.1 Accord en genre et en nombre

Les règles d'accord sont incorporées à la grammaire en codant autant de chaînes que de cas. Le petit exemple précédent devient alors :

Lexique :	
le : Art.m.s	chat : Nom.m.s
petit : Adj.m.s	beau : Adj.m.s
la : Art.f.s	balle : Nom.f.s
Grammaire :	
Adjx.m.s = Adj.m.s	Adjx.f.s = Adj.f.s
Adjx.m.s =	Adjx.f.s =
Adj.m.s Adjx.m.s	Adj.f.s Adjx.f.s
GN = Nom.m.s	GN = Nom.f.s
GN = Adjx.m.s Nom.m.s	GN = Adjx.f.s Nom.f.s
GN = Art.m.s Nom.m.s	GN = Art.f.s Nom.f.s
GN = Art.m.s Adjx.m.s	GN = Art.f.s Adjx.f.s
Nom.m.s	Nom.f.s

Tableau 2

Avec ces chaînes, l'analyseur reconnaîtra le segment *la balle*, mais pas *le balle*.

Cette méthode multiplie le nombre de chaînes dans la grammaire, mais elle permet de coder ces règles avec une grande souplesse, ce qui est nécessaire étant donné la complexité qu'elles atteignent lorsque les coordinations sont prises en compte.

2.2 Élision

Lorsqu'un mot a une forme élidée, il y a une sorte d'accord entre ce mot et son suivant, dans certaines fonctions grammaticales, mais pas dans toutes. Cette règle, qui interfère avec l'accord en genre pour les possessifs, peut être codée de façon analogue, en ajoutant les suffixes *.c* ou *.v* selon la première lettre de la chaîne, et *.e* ou *.l* selon la forme du dernier constituant.

Lexique :	
mon : Art.m.s	ma : Art.f.s.l
mon : Art.f.s.e	grande : Adj.f.s.c
armée : Nom.f.s.v	
Grammaire :	
GN = Nom.f.s.c	GN = Nom.f.s.v
GN = Adj.f.s.c Nom.f.s.c	GN = Adj.f.s.c Nom.f.s.v
GN = Adj.f.s.v Nom.f.s.c	GN = Adj.f.s.v Nom.f.s.v
GN = Art.f.s.l Nom.f.s.c	GN = Nom.f.s.e Adj.f.s.v
GN = Art.f.s.l Adj.f.s.c Nom.f.s.c	
GN = Art.f.s.l Adj.f.s.c Nom.f.s.v	
GN = Art.f.s.e Adj.f.s.v Nom.f.s.c	
GN = Art.f.s.e Adj.f.s.v Nom.f.s.v	

Tableau 3

Segments reconnus : *mon armée, ma grande armée*

Segments inconnus : *ma armée*

2.3 Concordance des temps

Il faut appliquer le même principe : définir autant de chaînes de propositions que de temps à prendre en compte, et exprimer les règles de concordance dans la définition des chaînes de phrases.

2.4 Ordre des compléments

Pour beaucoup de verbes, l'ordre des compléments n'est pas fixe. Cela oblige à dupliquer les chaînes dans lesquelles ils interviennent. Ainsi pour un verbe qui prend pour compléments deux groupes nominaux précédés de *à* et *de*, reconnus par les chaînes *GNà* et *GNde*, il faudra coder les deux chaînes :

P = GN V GNà GNde

P = GN V GNde GNà

3. Résolution de problèmes pratiques par le fichier de grammaire

3.1 *La ponctuation*

L'interprétation de la ponctuation n'est pas immédiate, elle exige de tenir compte de l'environnement. C'est pourquoi elle est traitée presque de la même manière que les mots. Plus précisément, le module TEXT passe au module CHAGRAM des items qui peuvent être des mots, c'est-à-dire des suites de lettres, ou des signes de ponctuation. Ainsi *aujourd'hui* est découpé en trois items, *a-t-il* en cinq, ... en trois, et *O.N.U.* en six. Dans la grammaire, les items de ponctuation sont traités de la même manière que les mots, ce qui permet de construire des chaînes qui exigent une virgule, par exemple.

3.2 *Les majuscules*

Dans le lexique, toutes les formes sont écrites en minuscules, car c'est la forme la plus précise. Il n'en est pas de même dans les textes pour des raisons variées. Certaines majuscules ont un rôle grammatical net : initiales de phrases, de noms propres, sigles. Pour d'autres, c'est moins net : intertitres, emphase, niveau de langue. D'autres enfin résultent uniquement de conventions d'enregistrement du texte.

C'est au module TEXT d'en tenir compte pour ramener les formes du texte à celles du lexique. Les majuscules dues aux conventions d'enregistrement seront transformées en minuscules; un mot avec une majuscule initiale sera transmis en deux items : d'abord un pseudo-signe de ponctuation, puis le mot lui-même en minuscules; quant aux passages entièrement en majuscules, TEXT les fera précéder et suivre d'une autre pseudo-ponctuation analogue aux guillemets.

Quant à la grammaire, elle peut ne pas en tenir compte en ignorant ces pseudo-ponctuations; sinon, dans une grammaire où toutes les phrases sont décrites par la chaîne *S*, où *MI* est la pseudo-ponctuation pour les majuscules initiales, toutes les phrases débutant par une majuscule seront décrites par la chaîne *MI S*.

3.3 *Les accents et autres altérations*

Les altérations (accents, cédilles,...) sont codées dans le lexique d'une manière standard. Lorsque le texte n'utilise pas les mêmes conventions, le module TEXT effectue les transpositions nécessaires. En outre, s'il ne peut

pas situer exactement tous les accents, parce qu'ils n'ont pas été codés du tout, ou parce qu'une majuscule les a masqués, TEXT reconstitue toutes les formes accentuées possibles et les passe ensemble à CHAGRAM.

3.4 Les mots composés

Le module TEXT les découpe en plusieurs items, le lexique identifie chacun d'eux et la grammaire les regroupe de la manière suivante :

Lexique :	
aujourd : Auj1	hui : Auj2
' : Papo	- : Ptir
timbre : Timl.m.s	timbres : Timl.m.p
poste : Tim2	
Grammaire :	
Adv = Auj1 Papo Auj2	
Nom.m.s = Timl.m.s Ptir Tim2	
Nom.m.p = Timl.m.p Ptir Tim2	

Tableau 4

3.5 Traitement des articles contractés

Au premier abord, il suffirait de les dédoubler dans TEXT; *des* serait passé en deux items *de les*. Malheureusement, dans la mesure où TEXT ne passe que des formes à CHAGRAM, sans aucune information linguistique, celui-ci ne ferait plus la distinction entre *du* et *de le*, ce qui introduirait des ambiguïtés supplémentaires, par exemple :

Max a mangé le rôti au lieu du goûter
 Max a mangé le rôti au lieu de le goûter

La solution retenue consiste à définir dans la grammaire une chaîne qui représente *du*, qui figure dans toutes les chaînes où *de* est suivi de l'article. Ainsi, à chaque fois que la grammaire possède une chaîne commençant par l'article et la même chaîne précédée de *de* au féminin, elle contient aussi la chaîne sans article et celle-ci précédée de *du* pour le masculin.

4. Une grammaire partielle : les déterminants

Cette étude est limitée aux déterminants reliés au nom sans préposition. Ce sont les articles, les adjectifs démonstratifs et possessifs, et ceux qui ont été classés dans les groupes *Dnum* et *Dadj* par Maurice Gross dans *Grammaire transformationnelle du français : syntaxe du nom*. La classe *Dnum* contient essentiellement les adjectifs numéraux et leurs composés, et *Dadj* des adjectifs indéfinis comme *plusieurs*, *chaque*, etc. Le but fixé à cette grammaire est de les reconnaître, sans étudier leur rôle dans la phrase. Aussi ils ont tous été regroupés dans l'unique chaîne *Dét*.

De la table *Dadj* (Gross 1977), seules les colonnes concernant la formation des déterminants ont été utilisées. Toutes les combinaisons des déterminants entre eux ont été examinées, ce qui a permis d'en rajouter quelques-uns et de les grouper en quatre classes à l'intérieur desquelles les déterminants ne se combinent pas. La première est formée des articles et des adjectifs démonstratifs et possessifs; la seconde de *Dnum* et de ses composés avec la première; la troisième de *Dadj* sauf *autre*; la quatrième de *autre*. La troisième se combine avec la première ou la deuxième, et la quatrième avec chacune des autres.

On trouvera le lexique et la grammaire en annexe. Le premier groupe de chaînes reconstitue les déterminants composés de plusieurs items, les quatre autres forment les quatre classes précédentes.

5. Autres utilisations de l'analyseur

Du fait de la répartition des rôles entre l'analyseur, la grammaire et le lexique, il est possible de l'utiliser pour d'autres buts en changeant essentiellement les deux derniers.

5.1 Détection de fautes de grammaire

L'analyseur peut les détecter et les classer par type. Pour cela, il faut rajouter à la grammaire les chaînes contenant les erreurs recherchées en leur donnant un nom qui les distingue des autres. Ainsi, en reprenant l'exemple de l'accord en genre et en nombre du paragraphe 2.1, il suffit de compléter la grammaire par le tableau 5 pour obtenir les groupes nominaux où l'adjectif ne respecte pas la règle d'accord en genre avec le nom : ils sont décrits par la chaîne GN.x.s. Et on peut définir d'autres chaînes pour d'autres types de fautes. Le principal défaut de cette méthode est qu'elle est capable d'inventer des fautes de grammaire en considérant comme un segment d'un certain type ce qui n'en est pas un, par exemple *le porte* pris

comme groupe nominal dans *Max le porte à Marie*. Ce problème devrait être résolu en poussant l'analyse au niveau de la phrase, et en modifiant le module EXPLOIT pour qu'il ne s'intéresse qu'aux chaînes des phrases dont aucune analyse correcte n'a été trouvée.

Grammaire :

Adjx.x.s = Adj.m.s Adjx.f.s

Adjx.x.s = Adj.f.s Adjx.m.s

Adjx.x.s = Adj.m.s Adjx.x.s

Adjx.x.s = Adj.f.s Adjx.x.s

GN.x.s = Adjx.x.s Nom.m.s

GN.x.s = Adjx.x.s Nom.f.s

GN.x.s = Art.m.s Adjx.x.s Nom.m.s

GN.x.s = Art.f.s Adjx.x.s Nom.f.s

GN.x.s = Art.m.s Adjx.f.s Nom.m.s

GN.x.s = Art.f.s Adjx.m.s Nom.f.s

Tableau 5

5.2 Recherche de syntagmes particuliers

L'analyseur est capable de retrouver les mots composés, les locutions ou les expressions figées dans toutes leurs variantes, à condition que le lexique en reconnaisse chacune des parties et qu'il y ait une chaîne dans la grammaire pour les combiner. Il est donc possible d'en faire une recherche automatique dans les textes.

6. Conclusion

Les choix qui ont été faits pour cet analyseur en font un outil très souple, en multipliant les chaînes. Pour en limiter le nombre, il est envisagé de définir un autre type de chaînes, où un seul suffixe (par exemple *.G*), en remplacerait plusieurs (*.m* et *.f*). Cette convention allègerait la codification, sans apporter aucune possibilité nouvelle à l'analyseur. De ce fait, les chaînes simples actuelles garderont leur intérêt théorique.

Henri Labesse

L.A.D.L. et

Université Paris-Sorbonne

Références

- BOONS, J.P., Alain Guillet, Christian Leclère (1976) *La structure des phrases simples en français*, Genève, Droz.
- GROSS, Maurice (1977) *Grammaire transformationnelle du français : syntaxe du nom*, Paris, Larousse.
- SALKOFF, M. (1973) *Une grammaire en chaîne du français*, Paris, Bordas.

ANNEXE A : Lexique des déterminants

' : Papo

un : Un.m.s

de : Un.m.p.l, Un.f.p.l, De

le : Ddef.m.s.l

l : Dl

du : Ddef.m.s.l

ce : Ddef.m.s.l

cette : Ddef.f.s

mon : Ddef.m.s, Ddef.f.s.e

mes : Ddef.m.p, Ddef.f.p

ta : Ddef.f.s.l

son : Ddef.m.s, Ddef.f.s.e

ses : Ddef.m.p, Ddef.f.p

nos : Ddef.m.p, Ddef.f.p

vos : Ddef.m.p, Ddef.f.p

leurs : Ddef.m.p, Ddef.f.p.

trois : Num

cent : Num

plusieurs : Plusieurs.m.p,

Plusieurs.f.p

diverses : Divers.f.p

différentes : Divers.f.p

maint : Quelque.m.s

maints : Quelque.m.p.

quelque : Quelque.m.s, Quelque.f.s

quelques : Quelque.m.p, Quelque.f.p

quelconque : Qcq.s

tel : Tel.m.s

tels : Tel.m.p

pareil : Tel.m.s

pareils : Tel.m.p

semblable : Tel.m.s, Tel.f.s.

certain : Certain.m.s

certain : Certain.m.p

seul : Seul.m.s

seuls : Seul.m.p

une : Un.f.s

d : Dd

la : Ddef.f.s.l

les : Ddef.m.p, Ddef.f.p

des : Ddef.m.p, Ddef.f.p

cet : Ddef.m.s.e

ces : Ddef.m.p, Ddef.f.p

ma : Ddef.f.s.l

ton : Ddef.m.s, Ddef.f.s.e

tes : Ddef.m.p, Ddef.f.p

sa : Ddef.f.s.l

notre : Ddef.m.s, Ddef.f.s

votre : Ddef.m.s, Ddef.f.s

leur : Ddef.m.s, Ddef.f.s

deux : Num

dix : Num

mille : Num

divers : Divers.m.p

différents : Divers.m.p

force : Force

mainte : Quelque.f.s

maintes : Quelque.f.p

quelconques : Qcq.p

telle : Tel.f.s

telles : Tel.f.p

pareille : Tel.f.s

pareilles : Tel.f.p

semblables : Tel.m.p, Tel.f.p

certaine : Certain.f.s

certaines : Certain.f.p

seule : Seul.f.s

seules : Seul.f.p

même : Même.m.s, Même.f.s
 mêmes : Même.m.p, Même.f.p
 tout : Dadjl.m.s
 tous : Dadjl.m.p
 chaque : Dadjl.m.s, Dadjl.f.s
 aucune : Dadjl.f.s
 aucunes : Dadjl.f.p
 nulle : Dadjl.f.s
 nulles : Dadjl.f.p
 quelle : Dadjl.f.s, Quel.f.s
 quelles : Dadjl.f.p, Quel.f.p
 laquelle : Dadjl.f.s, Laquelle
 lesquelles : Dadjl.f.p
 desquels : Dadjl.m.p
 auquel : Dadjl.m.s
 auxquelles : Dadjl.f.p
 n : N
 on : On
 sait : Sait
 sais : Sais
 qui : Qui
 autres : Autre.m.p, Autre f.p

toute : Dadjl.f.s
 toutes : Dadjl.f.p
 aucun : Dadjl.m.s
 aucuns : Dadjl.m.p
 nul : Dadjl.m.s
 nuls : Dadjl.m.p
 quel : Dadjl.m.s, Quel.m.s
 quels : Dadjl.m.p, Quel.m.p
 lequel : Dadjl.m.s
 lesquels : Dadjl.m.p
 duquel : Dadjl.m.s
 desquelles : Dadjl.f.p
 auxquels : Dadjl.m.p
 à : Prepà
 importe : Importe
 ne : Ne
 je : Je
 Dieu : Dieu
 autre : Autre.m.s, Autre.f.s

ANNEXE B : Grammaire des déterminants

Un.m.p.e = Dd Papo	Ddef.m.s.e = Dl Papo
Un.f.p.e = Dd Papo	Dadjl.f.s = Prepà Laquelle
Ddef.f.s.e = Dl Papo	Dadjl.i = N Papo Importe
Dadjl.f.s = De Laquelle	Dadjl.i = Je Ne Sais
Dadjl.i = On Ne Sait	Dadjl.i = Qui Sait
Dadjl.i = Dieu Sait	Dadjl.f.s = Dadjl.i Quel.f.s
Dadjl.m.s = Dadjl.i Quel.m.s	Dadjl.f.p = Dadjl.i Quel.f.p
Dadjl.m.p = Dadjl.i Quel.m.p	Detl.f.s = Un.f.s
Detl.m.s = Un.m.s	Detl.f.p.l = Un.f.p.l
Detl.m.p.l = Un.m.p.l	Detl.f.p.e = Un.f.p.e
Detl.m.p.e = Un.m.p.e	Detl.f.s = Ddef.f.s
Detl.m.s = Ddef.m.s	Detl.f.p = Ddef.f.p
Detl.m.p = Ddef.m.p	Detl.f.s.l = Ddef.f.s.l
Detl.m.s.l = Ddef.m.s.l	Detl.f.s.e = Ddef.m.f.e
Detl.m.s.e = Ddef.m.s.e	Det2.f = Dnum
Det2.m = Dnum	Det2.f = Ddef.f.p Dnum
Det2.m = Ddef.m.p Dnum	Det3.f.p = Plusieurs.f.p
Det3.m.p = Plusieurs.m.p	Det3.f.p = Divers.f.p
Det3.m.p = Divers.m.p	
Det3.m.p = Ddef.m.p Divers.m.p	Det3.f.p = Det2.f Divers.f.p
Det3.f.p = Ddef.f.p Divers.f.p	Det3.f.p = Force
Det3.m.p = Det2.m Divers.m.p	Det3.f.s = Quelque.f.s
Det3.m.p = Force	Det3.f.p = Quelque.f.p
Det3.m.s = Quelque.m.s	
Det3.m.p = Quelque.m.p	Det3.f.s = Un.f.s Qcq.s
Det3.m.p = Ddef.m.p Quelque.m.p	Det3.f.p = Un.f.p.l Qcq.p
Det3.f.p = Ddef.f.p Quelque.f.p	Det3.f.p = Dnum Qcq.p
Det3.m.s = Un.m.s Qcq.s	Det3.f.s = Tel.f.s
Det3.m.p = Un.m.p.l Qcq.p	Det3.f.p = Tel.f.p
Det3.m.p = Dnum Qcq.p	Det3.f.s = Un.f.s Tel.f.s
Det3.m.s = Tel.m.s	Det3.f.p = Un.f.p.l Tel.f.p
Det3.m.p = Tel.m.p	Det3.f.p = Dnum Tel.f.p
Det3.m.s = Un.m.s Tel.m.s	Det3.f.s = Certain.f.s
Det3.m.p = Un.m.p.l Tel.m.p	
Det3.m.p = Dnum Tel.m.p	
Det3.m.s = Certain.m.s	

Det3.m.p = Certain.m.p	Det3.m.p = Certain.f.p
Det3.m.s = Detl.m.s Certain.m.s	
Det3.f.s = Detl.f.s Certain.f.s	
Det3.m.p = Detl.m.p Certain.m.p	
Det3.f.p = Detl.f.p Certain.f.p	
Det3.m.s = Detl.m.s.l Certain.m.s	
Det3.f.s = Detl.f.s.l Certain.f.s	
Det3.m.p = Detl.m.p.l Certain.m.p	
Det3.f.p = Detl.f.p.l Certain.f.p	
Det3.m.p = Det2.m Certain.m.p	Det3.f.p = Det2.f Certain.f.p
Det3.m.s = Seul.m.s	Det3.f.s = Seul.f.s
Det3.m.p = Seul.m.p	Det3.f.p = Seul.f.p
Det3.m.s = Detl.m.s Seul.m.s	Det3.f.s = Detl.f.s Seul.f.s
Det3.m.s = Detl.m.s.l Seul.m.s	
Det3.f.s = Detl.f.s.l Seul.f.s	
Det3.m.p = Ddef.m.p Seul.m.p	
Det3.f.p = Ddef.f.p Seul.f.p	
Det3.m.p = Ddef.m.p Dnum Seul.m.p	
Det3.f.p = Ddef.f.p Dnum Seul.f.p	
Det3.m.s = Même.m.s	Det3.f.s = Même.f.s
Det3.m.p = Même.m.p	Det3.f.p = Même.f.p
Det3.m.s = Detl.m.s Même.m.s	
Det3.m.s = Detl.f.s Même.f.s	
Det3.m.p = Detl.m.p Même.m.p	
Det3.f.p = Detl.f.p Même.f.p	
Det3.m.s = Detl.m.s.l Même.m.s	
Det3.m.s = Detl.f.s.l Même.f.s	
Det3.m.p = Detl.m.p.l Même.m.p	
Det3.f.p = Detl.f.p.l Même.f.p	
Det3.m.p = Det2.m Même.m.p	Det3.f.p = Det2.f Même.f.p
Det3.m.s = Dadjl.m.s	Det3.f.s = Dadjl.f.s
Det3.m.p = Dadjl.m.p	Det3.f.p = Dadjl.f.p
Det = Autre.m.s	Det = Autre.f.s
Det = Autre.m.p	Det = Autre.f.p
Det = Detl.m.s Autre.m.s	Det = Detl.f.s Autre.f.s
Det = Detl.m.p Autre.m.p	Det = Detl.f.p Autre.f.p
Det = Detl.m.s.e Autre.m.s	Det = Detl.f.s.e Autre.f.s
Det = Detl.m.p.e Autre.m.p	Det = Detl.f.p.e Autre.f.p

Det = Det2.m Autre.m.p
Det = Det3.m.s Autre.m.s
Det = Det3.m.p Autre.m.p
Det = Det1.m.s
Det = Det1.m.p
Det = Det1.m.s.l
Det = Det1.m.p.l
Det = Det1.m.s.e
Det = Det1.m.p.e
Det = Det2.m
Det = Det3.m.s
Det = Det3.m.p

Det = Det2.f Autre.f.p
Det = Det3.f.s Autre.f.s
Det = Det3.f.p Autre.f.p
Det = Det1.f.s
Det = Det1.f.p
Det = Det1.f.s.l
Det = Det1.f.p.l
Det = Det1.f.s.e
Det = Det1.f.p.e
Det = Det2.f
Det = Det3.f.s
Det = Det3.f.p