INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Project-Team Atoll

Atelier d'Outils Logiciels pour le Langage naturel

Rocquencourt

THEME 3A

Table of contents

1. Team

Head of project-team
Éric Villemonte de la Clergerie [CR]

Vice-head of project team
Pierre Boullier [DR]

Administrative assistant
Emmanuelle Grousset [Remplacement assistante de projet, starting October 2003]
Nadia Mesrar [AJT]

Staff member
Bernard Lang [DR]
Philippe Deschamp [CR]
François Thomasset [DR, starting October 2003]

Research scientist (partner)
François Barthélemy [Maître de conférences, CNAM]

Visiting scientist
Areski Nait Abdallah [joint invitation with project LOGICAL]
Gabriel Pereira Lopes [November 2003, New University of Lisbon]
Francisco Riberra [December 2003, University of La Coruña]

Ph. D. student
Benoît Sagot [Détachement du corps des Télécoms]

Technical staff
Lionel Clément
Stéphane Laurière [until October 31st]
Guillaume Rousse [starting December 1st]

Student intern
Pascal Manchon [Engineer Internship, École Polytechnique, Summer 2003]
Julien Lafaye [Engineer Internship, École Polytechnique, Summer 2003]
Angélique Pochon [DEA, University of Orléans, Spring/Summer 2003]

2. Overall Objectives

2.1. Tools for Natural Language Processing

Project-team ATOLL was formed from people with strong competences in Parsing, essentially acquired in the context of Programming Language Compilation. This competence is now applied to *Natural Language Processing* (NLP), mainly in its parsing aspects but evolving toward more semantic aspects. Besides promising industrial applications, this domain of research also offers many scientific problems that may benefit from a strong formal and algorithmic approach.

In our exploration of fundamental parsing techniques, we focus on the use of tabular techniques, almost mandatory to efficiently handle the ambiguities inherent in any human language. The genericity of our techniques is also an asset because of the large diversity of grammatical formalisms. We also explore more recent and important issues related to robustness. We validate these techniques through the development of two prototype environments (SYNTAX and DYALOG) that may be used for building and running parsers.

However, a parser is only one component of a linguistic processing chain that requires other tools and also linguistic resources like lexicons. Besides interesting software engineering issues, designing and running such a chain raises questions about the availability and reusability of linguistic resources. These observations motivate our interest about the normalization, distribution and exploitation of linguistic resources. In particular,

we explore how the production cost of some linguistic resources could be reduced by using automatic or semi-automatic acquisition methods, possibly based on parsing corpora with our parsers. Obviously, such an approach is also an opportunity to test ATOLL's tools on a larger scale. We also believe that the use of well-designed tools for linguists can speed up the hand-crafting of linguistic resources as we try to promote with MetaGrammars, a level of abstraction above grammars allowing easier linguistic descriptions.

From a wider point of view, the acquisition of linguistic resources share some common aspects with the extraction of information from corpora or documents, a rapidly growing domain of research and applications. Indeed, the huge development of the World Wide Web (WWW) and the recent emergence of the notion of Semantic WEB plead for accessing information rather than simply accessing raw documents. As a consequence, tools are needed for extracting information from documents.

The diversity of the tools and resources needed to process natural language overcomes the capacities of project-team ATOLL. Therefore, we favor partnerships for reusing existing tools and resources or for developing new ones in common. An important issue, related to these cooperations and also very present in the NLP community, concerns the standardization and reusability of these tools and resources.

While marginal within ATOLL but nevertheless related to better accessing linguistic resources and tools, a reflexion is led by Bernard Lang on the issues of free access to scientific and technical resources, issues whose scientific, economical, and political interest becomes more and more visible.

3. Scientific Foundations

3.1. Grammatical formalisms

Key words: *NLP*, *Parsing*, *computational linguistics*, *dynamic programming*, *logic programming*.

Participants: Pierre Boullier, Éric Villemonte de la Clergerie.

Glossary

CFG *Context-Free Grammars*

DCG *Definite Clause Grammars*

TAG *Tree Adjoining Grammars*

TIG *Tree Insertion Grammars*

LIG *Linear Indexed Grammars*

LFG *Lexical Functional Grammars*

HPSG *Head-driven Phrasal Structure Grammars*

RCG *Range Concatenation Grammars*

MCG *Mildly Context-sensitive Grammars*

LPDA *Logical Push-Down Automata*

2SA *2-Stack Automata*

TA *Thread Automata*

Dynamic Programming Algorithmic method based on dividing a problem into elementary sub-problems whose solutions are tabulated to be reused whenever possible

This theme explores the use of generic parsing techniques covering a large continuum of NLP grammatical formalisms, focusing especially on efficient handling of ambiguities.

3.1.1. From programming languages to linguistic grammars

The shift from programming language grammars to NLP grammars seriously increases complexity and requires ways to handle the ambiguities inherent in every human language. It is well known that these ambiguities are the sources of many badly handled combinatorial explosions.

Furthermore, while most programming languages are expressed by (subclasses) of well-understood context-free grammars (CFGs), no grammatical formalism has yet been accepted by the linguistic community for the description of human languages. On the contrary, new formalisms (or variants of older ones) appear constantly. Many of them may be classified into the two following large families:

Mildly context-sensitive formalisms : They manipulate possibly complex elementary structures with enough restrictions to ensure the possibility of parsing with polynomial time complexities. They include, for instance, Tree Adjoining Grammars (TAGs) with trees as elementary structures, Linear Indexed Grammars (LIGs), and Range Concatenation Grammars (RCGs).

Unification-based formalisms : They combine a context-free backbone with logic arguments as decoration on non-terminals. Most famous representatives are Definite Clause Grammars (DCG) where PROLOG powerful unification is used to compute and propagate these logic arguments. More recent formalisms, like Lexical Functional Grammars (LFGs) [23] and Head-Driven Phrasal Structure Grammars (HPSGs) [26] rely on more expressive Typed Feature Structures (TFS) [21] or constraints.

The above-mentioned characteristics may be combined, for instance by adding logic arguments or constraints to non-terminals in TAGs. We should also mention that we also concur to this large diversity of formalisms with the introduction of RCGs (Section 6.1).

However, despite this diversity, most formalisms take place in a so-called **Horn continuum**, i.e. a set a formalisms with increasing complexities, ranging from Propositional Horn Clauses to first-order Horn Clauses (roughly speaking equivalent to PROLOG), and even beyond.

This observation motivates our exploration of generic parsing techniques covering this continuum, through two complementary approaches. Both of them use dynamic programming ideas to reduce the combinatorial explosions resulting from ambiguities :

Multi-pass approach : Parsing is broken into a sequence (or cascade) of parsing passes, of (practical or theoretical) increasing complexities, each phase guiding the next one ;

Global Approach : It is mainly based on the use of Push-Down Automata [PDA] to describe parsing strategies for complex formalisms.

These two approaches enrich each other: studying some specificities observed for the multi-pass approach has triggered theoretical advances ; conversely, well-understood and identified theoretical concepts have suggested a widening of the scope of the multi-pass approach.

3.1.2. Multi-pass approach

Programming languages processing is usually broken into several successive phases of increasing complexity : lexical analysis, parsing, static semantics,... The decomposition is motivated by theoretical and practical reasons. The finite state automata (FSA) that model lexical analysis are very efficient but do not have enough expressive power to describe the syntax, which requires, at least, Context-Free Grammars. Similarly, CFGs are not powerful enough to describe some contextual phenomena needed in static semantics. Beside a better efficiency (each phase being handled with the best level of complexity), decomposing increases modularity.

The multi-pass approach for NLP results from similar observations. We try to identify and capture, within adequate grammatical formalisms, subparts of grammars which can guide the remaining processing. For instance, we observe that most formalisms found in the Horn continuum are structured by a non-contextual backbone. This backbone may be first parsed with a very efficient and generic non-contextual parser, namely SYNTAX (cf. 5.1). More formalism-specific treatment can then be applied to check additional constraints.

3.1.3. Global approach

The multi-pass approach is less easy to implement when there is no obvious decomposition, for instance when the CF backbone of a formalism can not be extracted (as in PROLOG) or when the possible phases would be mutually dependent (for instance, when some constraints have a strong impact of the processing of the CF backbone). A more global approach is then needed where constraints and parsing are handled simultaneously.

This very general approach relies on abstract Push-Down Automata formalisms that may be used to describe parsing strategies for various unification-based formalisms [6]. The notion of stack allows us to apply dynamic programming techniques to share elementary sub-computations between several contexts : the intuitive idea relies upon temporarily forget information found in stack bottoms. Elementary sub-computations are represented in a compact way by *items*. The introduction of 2-Stack Automata [2SA] allowed us to handle formalisms such as TAGs and LIGs [7]. More recently, *Thread Automata* (TA) [5] have been introduced to cover mildly-context sensitive formalisms such as Multi-Component TAGs (MC-TAGs).

This global approach may be related to *chart parsing* [22] or *parsing as deduction* [25] and generalizes several approaches found in Parsing but also in Logic Programming. The DyALog system (cf. 5.2) implements this approach for Logic Programming and several grammatical formalisms.

3.1.4. Shared parse and derivation forests

Both previously presented approaches share several characteristics, for instance the use of dynamic programming ideas and also the notion of *shared forest*. A shared forest groups in a compact way the whole set of possible parses or derivations for a given sentence. Formally, a shared forest may be seen as a grammar or a logic program [4]. For instance, parsing with a CFG may lead to an exponential (or unbounded) number of parse trees for a given sentence, but the parse forest remains cubic in the length of the sentence and is itself equivalent to a CFG (as an instantiation of the original CFG by intersection with the parsed sentence). Moreover, these shared forests are natural intermediary structures to be exchanged from one pass to the next one in the multi-pass approach. They are also promising candidates for further linguistic processing (semantic processing, translation, ...). One can also relatively easily extract dependency information between words from these forests.

3.2. Linguistic Infrastructure and Normalization

Participants: Éric Villemonte de la Clergerie, Pierre Boullier, Philippe Deschamp, François Barthélemy, Lionel Clément, François Thomasset.

We are interested by the many issues related to the installation of a whole linguistic processing chain, in particular for accessing and representing the needed linguistic resources (cf. 6.5).

To facilitate the installation of such linguistic chains, we develop two systems to build parsers, namely SYNTAX (cf. 5.1) and DyALog (cf. 5.2). We also develop and distribute several linguistic components (cf. 5.4).

Because we realized that diffusing or reusing tools and resources is not really possible without some standardization, ATOLL is involved in on-going national and international efforts to normalize linguistic resources, using XML-based representations (cf. 7.2). This decision follows preliminary experimentations we have conducted to normalize TAGs and shared forests.

3.3. Resource acquisition and crafting

Participants: Éric Villemonte de la Clergerie, Benoît Sagot, Lionel Clément.

Glossary

MG *MetaGrammars*

Linguistic resources are scarce and expensive, because they are difficult to build, especially when hand-crafted. This observation motivates us to investigate methods to automatically or semi-automatically acquire linguistic resources. In particular, we would like to reach some bootstrap level where parsing corpora may be used to enrich lexica that may themselves be used for better parsing.

Preliminary experiments have been conducted during the now ended ARC (Action de Recherche Concertée) RLT « Linguistic resources for TAGs » and we are currently working on processing botanical corpora (cf. 6.7).

For hand-crafted resources, we try to design adequate tools and adequate levels of representation for linguists. For instance, we are currently involved in developing grammars through a more abstract notion of *MetaGrammar* (MG) (cf. 6.3). Introduced by [20], a MetaGrammar allows the linguist to focus on a modular description of the linguistic aspects of a grammar, rather than focusing on the specific aspects of a given grammatical formalism. Translation from MGs toward grammatical formalisms such as TAG or LFG may be automatically handled (cf. 5.3). Graphical environments can be used to design MGs and their modularity provides a promising way for sharing the description of common linguistic phenomena across human languages [11].

4. Application Domains

4.1. Applications

Computational Linguistic offers a wide range of potential applications, especially with the emerging of information systems. More specifically for ATOLL, one can (non exhaustively) list the following application domains:

Grammatical checking Parsing is used to detect grammatical errors and suggests corrections. Tabulation-based parsing techniques present a great potential for grammatical checking because they allow the exploration of many alternatives (for correcting errors) without combinatorial explosions.

Knowledge acquisition Linguistic (and statistical) techniques may be used to extract knowledge from corpora, ranging from a simple terminological list of words to more complex semantic networks with concepts and relations. In this continuum, we also find lexicons, thesaurus, and ontologies. We strongly believe that this domain can benefit from more sophisticated parsing-based techniques.

Text mining and Questions/Answers Parsing and possibly semantic or pragmatic processing may be used to extract precise information from a document, for instance to feed a (knowledge) database or to answer questions formulated by users.

Among these various application domains, ATOLL focuses its efforts on knowledge acquisition and text mining, in particular through the action BIOTIM for processing botanical corpora (cf. 7.3).

5. Software

5.1. System Syntax

Participants: Pierre Boullier [maintainer], Philippe Deschamp.

The (not yet released) version 6.0 of the SYNTAX system is currently developed under Linux. Release 3.9 essentially handled deterministic CFGs of type LALR(1) while Release 6.0 extends it by including RLR (an extension of LR parsing strategy in which an unbounded number of look-ahead terminal symbols may be used, if necessary), non-deterministic CF parsers based upon push-down automata of type LR, RLR or left-corner, and a parser generator for Range Concatenation Grammars (RCGs).

The current development architecture allows us to easily port this version for various 32bit platforms such as Solaris, HP and Windows. We are currently porting the release 6.0 to 64bit Unix platforms. Another objective is to port SYNTAX on Apple's PowerPC, both G4 and G5 models.

5.2. System DyALog

Participant: Éric Villemonte de la Clergerie [maintainer].

DYALOG*: http://atoll.inria.fr Rubrique « Logiciels »*
DYALOG provides an environment to compile and execute grammars and logic programs. It is essentially based on the notion of tabulation, i.e. of sharing computation by tabulating traces of them. DYALOG is mainly used to build parsers for Natural Language Processing (NLP). It may nevertheless be used as a replacement for traditional PROLOG systems in the context of highly ambiguous applications where sub-computations can be shared.

The current release 1.10.3 of DYALOG is freely available by FTP under an open source license and runs on Linux platforms for x86 architectures.

The current release handles logic programs, DCGs (*Definite Clause Grammars*), FTAGs (*Feature Tree Adjoining Grammars*), FTIGs (*Feature Tree Insertion Grammars*) and XRCGs (*Range Concatenation Grammars* with logic arguments). Several extensions have been added to (some of) these formalisms such as intersection, Kleene star, and interleave operators. Typed Feature Structures (TFS) as well as finite domains may be used for writing more compact and declarative grammars. Cyclic terms are correctly handled by DYALOG.

C libraries can be used from within DYALOG to import APIs (`mysql`, `libxml`, ...) and modules with namespaces are now available.

DYALOG is largely used within ATOLL to build parsers but also derivative softwares, such as a compiler of MetaGrammars (cf. 5.3). It is also an essential component in the development of a robust Portuguese parser at the New University of Lisbon. It is occasionally used by several people at LORIA (Nancy) and at University of Pennsylvania.

5.3. MG compiler mgcomp

Participant: Éric Villemonte de la Clergerie [correspondant].

MGCOMP*: http://atoll.inria.fr Rubrique « Logiciels »*
DYALOG (cf. 5.2) has been used to implement MGCOMP, a compiler of MetaGrammar (cf. 6.3). Starting from an XML representation of a MG, MGCOMP produces an XML representation of its TAG expansion.

The current version **1.2.0** is freely available by FTP under an open source license. It is used within ATOLL and (occasionally) at University of Pennsylvania.

5.4. Linguistic Tools

Participants: Éric Villemonte de la Clergerie, Lionel Clément, Benoît Sagot, François Barthélemy, François Thomasset.

ATOLL develops severals tools that may be used for the first levels of linguistic processing preceding parsing, in particular morpho-syntax (cf. 6.5). They are freely available under open source licenses, keeping in mind that most of these tools are still beta versions.

LEXED (4.3.1) a C software developed by L. Clément to build efficient and compact lexica from lists of words (completed with additional information).

TOKENIZER (5.2.1) a C tokenizer developed by L. Clément for French that may be easily adapted for other languages. It can output an XML stream of tokens.

LINGPIPE (0.1.0) a small set of Perl modules developed by É. de la Clergerie to setup and configure a linguistic pipeline. The current version of lingpipe comes with a basic set of wrappers for the various linguistic tools we use for the morpho-syntactic processing of French (tokenizer, tagger, lexicon lookup, ...)

Other tools are being developed by ATOLL but are not yet distributed. We can mention a chunker and a tool to conjugate French verbs.

6. New Results

6.1. Contextual Parsing

Participant: Pierre Boullier.

Key words: *Context-sensitive grammatical formalisms*, *shared forests*, *Range Concatenation Grammars*, *Polynomial Parse Time*, *Modular Grammars*.

Glossary

MCS *Mildly Context-sensitive Grammars*

RCG *Range Concatenation Grammars*

TAG *Tree Adjoining Grammars*

Our work on efficient parsing strategies mostly concentrates on deeper investigations on so-called guiding techniques. We have already demonstrated how guiding techniques can be used in range concatenation language (RCL) parsers and how practical speed-ups may be achieved. This year, we have shown that, in practice, even the old Earley context-free (CF) parsing algorithm can be improved by a guiding technique.

6.1.1. Guided Parsing of Context-Free Languages

The investigation of approaches offering parsing speed-ups is an important topic of research, especially for real-word applications based upon large-scale grammars. However, due to the huge amount of research that has already been performed by the parsing community, it seems difficult to find a technique that would improve the efficiency of CF parsers. First, we must stress out that we are not looking for a new CF parsing strategy with a better theoretical parse-time complexity, but are rather looking for techniques which improve practical parse-times in some cases (i.e., for some grammars or for some input sentences). Even with such a restricted purpose, the success was not ensured beforehand and previously reported attempts were not very encouraging. That's why Martin Kay challenged the parsing community "*to construct a weakened version of a particular grammar that will serve as an effective guide*".

When a nondeterministic choice occurs during a nondeterministic process, one usually explores all possible ways, either in parallel or one after the other, using a backtracking mechanism. In both cases, the nondeterministic process may be assisted by another process which directs its way. This assistant may be either a guide or an oracle: an *oracle* always indicates all the right branches that will eventually lead to success, and only these branches, whereas a *guide* will indicate all the right branches but possibly some wrong ones. Obviously, an oracle is a perfect guide.

Of course, nondeterministic parsers could greatly benefit from guiding. A guided parsing is split into two phases: the first *guiding parser* builds a structure called the *guide*, while in the second phase, the *guided parser* consults the guide in order to help (some of) its nondeterministic choices.

In [8], we show how a (slightly modified) general CF parser can, in practice, be sped up using guiding techniques. We choose to guide Earley type parsers, the guide being called during its *Predictor* phase: an initial Earley item $[A \rightarrow .\alpha, i]$ is added to a given table T_i, for a given source index i, only when that initial item is in the guide. This guide is the output of a guiding parser which is in fact a finite transducer (FT), based upon a regular grammar (RG), or more precisely a finite-state automaton (FA), which defines a (regular) superset of the original CFG.

The design of regular supersets (or more generally regular approximations) of CF Languages (CFLs) is not original. However, for a realistic CFG, the associated RG or FA is often too large or not restrictive enough for efficient guiding. Thus, we have designed and implemented a new FT called *dynamic set automaton* which, given a CFG and a sentence, outputs a guide in time linear w.r.t. the length of the sentence.

Experiments of an Earley parser using the guide produced by a dynamic set automaton have been performed on a large test suite with a large, highly ambiguous, wide coverage English CFG. The grammar has been automatically extracted from the TAG defined in the XTAG project at the University of Pennsylvania. The

test suite includes around 42 000 sentences, extracted from the Wall-Street Journal, for a total length of 1M words.[1] The experiments have shown that a guided Earley recognizer can run more than three times faster that its standard counterpart.

Grammar approximation also provides a new approach to *supertagging* [9]. Introduced by Joshi and Srinivas, supertagging improves the efficiency of NLP parsers for lexicalized formalisms by selecting, for each word, its appropriate descriptions given its context in a sentence. For instance, for Lexicalized TAGs (LTAGs), each elementary tree contains at least one lexical item called an *anchor*. All the arguments of this anchor are instantiated at places (nodes) through either a substitution or an adjunction operation. Thus an elementary tree may be seen as a description of the context, the domain of locality, of its anchor, possibly including long distance dependencies. Since an elementary tree (either initial or auxiliary) defines its anchor precisely, it is called a *supertag*, because it conveys much more information than the standard part-of-speech tag. As a consequence, in the LTAG context, the lexical ambiguity of a word (i.e., the number of supertags associated with it) is generally much greater than its number of standard part-of-speech tags.

Most LTAG parsers run in two phases: for each word in a sentence, the first phase of *supertagging* selects the appropriate supertags, while the second phase combines the selected supertags through substitutions and adjunctions. *Supertag disambiguation* is the process by which local lexical ambiguity can be reduced or, eventually, resolved, assigning a single supertag per word. This selection of the most probable supertag is usually achieved using statistical distributions of supertag co-occurrences extracted from (large) annotated corpora of parses. In this context, the result of a supertagger is almost a parse in the sense that a parser only needs to link together its supertags into a single structure. If such a single structure cannot be built, we have a partial parser. However, this approach which assigns a single supertag per word, even if it is extended to produce the n-best supertags for each word, may well eliminate trees which are parts of valid parses.

Our approach to supertagging differs from usual ones on several points:

1. non-statistical: we do use statistical information to discard supertags, hence avoiding the need for training on annotated corpora, which are very costly to develop;
2. strictness: we only discard supertags than cannot be part of a complete parse;
3. parsing-based: the choice of the supertags for a word results from a true (but simplified) parsing, in contrast with Srinivas and Joshi who estimate that only "*local techniques can be used to disambiguate supertags" and that "full parsing is contrary to the spirit of supertagging*".

A potential advantage of being parsing-based is that our supertaggers may use global information to take their decisions while traditional supertaggers only rely on local (statistical) information (n-gram model). However, the parser must be "sufficiently" efficient. Obviously, it cannot be based upon the original LTAG but, rather, on some (automatically deduced) approximation. In other words, our purpose is to reduce supertag ambiguity by using only structural information which can be automatically extracted from any given LTAG.

We have studied the possibility to define a CF superset and a regular superset for any given tree-adjoining language (TAL) and have shown that general recognizers for these two kinds of supersets can be transformed into supertaggers for the original LTAG. Finally, [9] reports some experiments that have been performed on the previously mentioned Wall-Street Journal test suite with the original LTAG of the XTAG project. Both kinds of supertaggers give encouraging results with very good precision scores: 88% for the regular supertagger and above 96% for the CF supertagger, and, of course, a 100% recall score for both of them.

[1] In this test suite, the lengths of individual sentences show great variations: with an average length of almost 23 words per sentence, there are single word sentences while the longest one contains 158 words!

6.2. Automata and Tabulation for Parsing

Participants: Éric Villemonte de la Clergerie, Areski Nait Abdallah.

Key words: *Tabulation*, *Parsing*, *Dynamic Programming*, *Logic Programming*, *Push-Down Automata*, *TAG*.

Glossary

TAG *Tree Adjoining Grammars*

TIG *Tree Insertion Grammars*

TA *Thread Automata*

With Henk Harkema (from Sheffield University), we examine the possibility to use Thread Automata [TA] [5] for parsing *Minimalist Grammars*, a recent grammatical formalism issued from Noam Chomsky's ideas and formalized by Ed Stabler. We explore several approaches for describing parsing strategies for Minimalist Grammars with TAs [19]. Minimalist Grammars are weakly equivalent to Mildly Context-Sensitive formalisms, such as MC-TAGs, but still present interesting challenges to define simple, elegant, and efficient parsing strategies.

On the other hand, rather than moving to more complex grammatical formalisms, we have investigated how more expressive power could be squeezed out from simpler formalisms such as TAGs, mimicking Joshi's program.

For instance, we examine how to identify grammar subparts with lower complexities and how to adapt parsing strategies to take this information into account. In particular, we have implemented within system DyALog the possibility of building hybrid TAG-TIG parsers. TIGs (Tree Insertion Grammars) are a variant of TAGs with constraints on adjunction and auxiliary trees that ensure an strong equivalence with CFG and a cubic parsing time complexity. Building hybrid TAG-TIG parsers allows us to restrict TAG complexity only where necessary in a grammar, knowing that TAG linguistic grammars are actually almost TIG. This work has been completed by implementing within DyALog a phase of analyze to detect TIG trees in a TAG. Preliminary experiments on a small French TAG have shown the effectiveness of this approach.

Another way of getting more power without increasing complexity has been achieved by studying and introducing new generic operators to write more compact grammars. The Kleene star operator `@*` is used to express repetitions, with the possibility to set up a range condition. The second operator, inspired by [24], is an *interleave* operator `##` allowing the interleaving in any order of several sequences of constituents while preserving order within each sequence. For instance, the expression `(a,b)##(c,d)` recognizes the 6 expressions *abcd*, *acbd*, *acdb*, *cdab*, *cadb*, and *cabd*. Combining both operators, one can write compact and elegant representations such as `np <-- det, ((adj @*) ## n), (pp @*)` stating, for French, that a nominal phrase is formed from a determiner followed in any order by a noun and any number of adjectives (before and/or after the noun) and followed by any number of prepositional phrases. Such a clause may be expressed by an equivalent set of (recursive) clauses but the factorized representation can be processed more efficiently, even if the theoretical worst-case complexity does not change. These operators also allow to stay closer from the underlying linguistic structure when examining the shared derivation forests returned by the parsers: indeed, coming back to the nominal phrase example, a direct link will exist between a noun and its adjectives or attached prepositional phrases, rather than an indirect one through recursive intermediary non terminals.

These new operators have been deployed for DCGs, but also for TAGs, TIGs, and logic programs. The Kleene star mechanism also allows an immediate handling of multiple adjunctions for TIGs.

The implementation of the interleave operator has raised several questions, still under investigation, about the current design of DyALog's underlying abstract machine. The idea is to keep an open list of closures (or "threads") that may be followed at any time. A better understanding of how to handle these closure lists is important as a first step toward a full implementation of Thread Automata in general, and of Multi-Component TAGs in particular.

The addition to DYALOG of a new formalism (TIG) and of generic new operators for several formalisms (Prolog, DCG, TAG, TIG) has once again raised the issue of the evolution in the design of DYALOG compiler. New solutions have to be explored to quickly and easily extend this compiler with new formalisms or with new cross-formalism features. Among already present features to be generalized, we can, for instance, mention left-corner parsing strategies, bidirectional parsing, and autoloading (a way to load only the subset of grammatical structures that is pertinent for parsing a given sentence). A greater modularity is an option but we are also considering an object-oriented layer in DYALOG to exploit inheritance.

6.3. MetaGrammars

Participants: Lionel Clément, Éric Villemonte de la Clergerie, Benoît Sagot.

Glossary

MG *MetaGrammars*

L. Clément has pursued the development of a MetaGrammar (MG) for French and of an environment adapted to design MGs. This environment includes a graphical editor and a compiler from MGs to TAGs or LFGs. Several papers have resulted from this work [10][12][11].

Because the exact formalization of MGs is still a subject of research and also to explore some algorithmic issues, É. de la Clergerie has developed, with DYALOG, a new prototype of MG compiler, called MGCOMP. This new prototype is actually very efficient and should allow quick explorations of new features for MetaGrammars. One issue is, in particular, to understand how one MG class, describing some linguistic aspect and some partial subtree, may be reused more than once in a larger class to form a complete tree. The use of distinct namespaces for different instances of a same class should provide a possible solution.

This work around MGs takes place within an active cooperation with Project-Teams "Langues & Dialogue" and "Calligramme" (LORIA), LATTICE/TALaNa (Paris 7) and University of Pennsylvania. In particular, we have organized an informal one-day workshop on this topic (September 2003).

6.4. LFG

Participants: Lionel Clément, Pierre Boullier, Philippe Deschamp.

L. Clément has developed **XFLG**, a parser for Lexical Functional Grammars (LFG), based on LR techniques. In cooperation with P. Boullier and Ph. Deschamp, he has examined, through experimental versions, how to interface XFLG with system SYNTAX (cf. 5.1). The key idea is to delegate the handling of the CFG backbone of LFG to SYNTAX, which is very efficient on CFGs, in particular to handle non-determinism. Some LFG decoration parts could also been handled by SYNTAX as attributes of an Attributed Grammar. However, a clearer understanding of how LFG semantic parts can be computed is needed, in particular for efficient computation sharing.

6.5. NLP Infrastructure and standardization

Participants: Lionel Clément, Éric Villemonte de la Clergerie, Julien Lafaye, François Barthélemy, François Thomasset.

ATOLL tries to design and setup an XML-based linguistic pipeline, easing the integration of new components by wrapping them if necessary. A prototype have been developed and we are currently thinking to possible evolutions, for instance by using SAX events, to improve its efficiency.

The pipeline mainly covers the first layers of linguistic processing, namely morpho-syntactic processing (segmentation, tagging, lexicon lookup, ...). It integrates several tools which are developed within ATOLL by L. Clément (cf. 5.4) and which have been improved.

The main role of the pipeline is to feed entry to our parsers. It is also a platform for testing and demoing propositions for standardizing morpho-syntactic annotations in the context of action Normalangue (cf 7.2). In particular, it is planned to setup a Web service to access our linguistic pipe.

During its internship, J. Lafaye has pursued the investigation of using XML databases to store linguistic resources (such as parse forests) and designed a specialized request language to query data. Using XML databases becomes a really interesting alternative with the emergence of native XML databases, such as eXists (http://exist.sourceforge.net/), and their integration in Java servlet environments, such as Tomcat (http://jakarta.apache.org/tomcat/). However, the design of an elegant, efficient and linguistic-oriented query language is still a work in progress [15].

6.6. Lexicon acquisition and definition

Participants: Lionel Clément, Benoît Sagot, Bernard Lang.

L. Clément, B. Sagot, and B. Lang have designed and applied a new technique to (almost) automatically extract wide-coverage morphological lexica from corpora, using morphological knowledge [14]. It relies on the basic idea that a lemma can be hypothesized when several different words found in the corpus are best interpreted as morphological variants of this candidate lemma. The technique has been validated by extracting verbs and adjectives on a general 25 million word French corpus. The results are very encouraging with the extraction of many words, often derived words, that are not always present in other lexica. The resulting French lexicon, named **LeFFF**, is now freely available as an open resource (http://www.lefff.net/).

Application to the acquisition of domain-specific adjectives on a botanic corpus gave also very good results, thus demonstrating its usability to extract domain-specific lexica. Moreover, it is generalizable to any language with a substantial morphology.

B. Sagot has started his PhD work by an important bibliographical review on the design of lexica with rich lexical semantic information. A theoretical framework is being elaborated, with a beginning of implementation.

6.7. Processing Botanical Corpora

Participants: Éric Villemonte de la Clergerie, Angélique Pochon, Pascal Manchon, Benoît Sagot.

BIOTIM Action: http://atoll.inria.fr Rubrique « Projets »
In the context of French action BIOTIM (cf. 7.3), ATOLL is involved in processing botanical corpora.

Generalizing previous experiments, P. Manchon has developed, during his internship, a Perl prototype to model the logical structure of a document (for instance, a botanical corpus) and apply this model on a plain linear textual version of the document in order to get a logically structured XML version. Preliminary experiments have been tried on methods for discovering the logical structure of a document, by examining repeated segments (modulo small variations) [17].

During her internship, A. Pochon has tried several tools, namely ACABIT (http://www.sciences.univ-nantes.fr/info/perso/permanents/daille/acabit.html) and FASTR (http://www.limsi.fr/Individu/jacquemi/FASTR/), to extract specific terminology from botanical corpora. She has also applied various linguistic tools to add morpho-syntactic information to the corpora, namely tagging, lemmatization, named entity recognition (dates, measures, Latin scientific names, abbreviations, ...). With these different elements of information, she has explored methods to structure the terminology into clusters appearing in similar contexts and to discover relationships between terms, using a small set of template expressions [18].

6.8. Open Source Software

Participants: Bernard Lang, Stéphane Laurière.

Key words: *Open Source Software*, *Linux*.

The evolution of market and the availability of software and linguistic resources (such as lexicons, grammars, or corpora) raised our interest for the development of open source resources. This new model of production and diffusion for non-material goods has since emerged as a major economical, political, and technical issue in the area of the Information Society, justifying our investment during these last 5 years.

Technically, efforts has been done within the context of RNTL action e-COTS (cf. 7.1) to set up an open and cooperative internet portal devoted to software components, either open source or commercial.

7. Contracts and Grants with Industry

7.1. RNTL Action e-COTS

Participants: Bernard Lang, Stéphane Laurière.

The main goal of RNTL action **e-COTS** was the realization of an open and cooperative internet portal devoted to software components, either open source or commercial. This portal is now operational, with a freely reusable content.

The other participants to **e-COTS** are Thomson-CSF (project leader), EDF and Bull (group Pharos in project Dyade). The action was officially closed at the end of October 2003.

7.2. Action Normalangue/RNIL

Participants: Éric Villemonte de la Clergerie, Lionel Clément.

Normalangue Home Page: http://www.normalangue.org/
RNIL Home Page:http://atoll.inria.fr/RNIL/
TC37SC4 Home Page: http://www.tc37sc4.org/
ATOLL is a leader participant in the RNIL subpart of action Normalangue, funded by French program Technolangue. This action promotes the emergence of standardized representations for linguistic resources, in parallel with the definition of API for the corresponding linguistic tools. The action supports the French mirror group of ISO sub-committee TC37 SC4 for the normalization of linguistic resources.

É. de la Clergerie chairs this mirror group, which has organized 6 meetings in 2003. L. Clément and É. de la Clergerie are the promoters of a French proposition of a morpho-syntactic annotation framework (MSAF), which has been accepted as a new work item by ISO TC37SC4. An ISO technical meeting has been held on MSAF in Paris (December 8-9th) with a resulting Working Draft being finalized (in French and English).

7.3. Action BIOTIM

Participants: Éric Villemonte de la Clergerie, Benoît Sagot, Guillaume Rousse.

BIOTIM home page: http://www-rocq.inria.fr/imedia/biotim/
Funded by ACI program on “Masses de données” (Data Warehouses), action BIOTIM has recently been accepted for the next 3 years. Its thematic is the processing of botanical textual corpora and image collections in order to extract knowledge and establish bridges between texts and images for more intelligent navigations at a semantic level. ATOLL is essentially concerned with the linguistic processing of textual corpora with generic methods to extract terminologies, ontologies and knowledge bases.

The other participants to BIOTIM are INRIA project-team IMEDIA (leader), CNAM team Vertigo, INRA team URGV, IRD, and LIFO (University of Orléans).

7.4. Action EVALDA

Participants: Éric Villemonte de la Clergerie, Pierre Boullier, Lionel Clément.

ATOLL participates to the action EVALDA of French program Technolangue. The main goal of this action is to organize an evaluation campaign for parsers, to be held in 2004. This year activity has been essentially devoted on the definition of an annotation schema for parsing output and the preparation of corpora. Members of ATOLL have attended the EVALDA meetings and follow the evolution of the proposed annotation schema.

7.5. ARC Geni

Participants: Éric Villemonte de la Clergerie, Benoît Sagot.

We have participated to the ARC Geni « Generation and Inference ». Started in 2002 and closed end of 2003, Geni's main objective is to improve text generation by exploiting lexical semantic knowledge through inference processes. The other participants to GENI are "Langues et Dialogue" (LORIA, coordinator), ILPL (IRIT), Lattice (University Paris 7) and Orpailleur (LORIA).

8. Other Grants and Activities

8.1. National Actions

Ph. Deschamp is a member of the French "Commission spécialisée de terminologie de l'informatique et des composants électroniques" (terminology committee for Computer Science and Electronic), and distributes on-line the glossary http://www-rocq.inria.fr/who/Philippe.Deschamp/CMTI/ resulting of his work (more than 130 000 downloads). Ph. Deschamp is also a member of the French "Commission spécialisée de terminologie et de néologie des télécommunications" (terminology committee for telecommunication).

B. Lang is vice-president of AFUL (http://www.aful.org), "Association Francophone des Utilisateurs de Linux et des Logiciels Libres", and member of the administration board of ISoc-France, the Internet Society French branch. He is also a member of the scientific board of association SOISSON Informatique Libre.

8.1.1. Open Source Software

B. Lang has presented the notion of open source software in several workshops, talks and conferences, organized by local collectivities and administrations.

8.2. International networks and working groups

8.2.1. Open Source Software

B. Lang has been several times invited to talk on Open Source Software.

B. Lang is a member of an expert committee on Open Source Software for the European Commission General Direction for Information Society (ex DG 13) (http://eu.conecta.it/).

8.2.2. Action INRIA-ICTTI FASTLING

Funding for visits has been granted for 2002 and 2003 by French-Portuguese program INRIA-ICTTI for pursuing a long-lasting cooperation between ATOLL, team CENTRIA of Lisbon New University and team LIFO at University of Orléans.

8.2.3. PAI PICASSO CATALINA-2

Funding for visits has been granted by the French-Spanish PAI (Programme d'actions intégrées) PICASSO to renew a cooperation named CATALINA-2 between ATOLL and team COLE at University of La Coruña. We cooperate on parsing techniques (in particular for TAGs) and are interested by establishing a more ambitious project on information extraction.

8.2.4. XTAG Collaboration

We have pursued an informal cooperation with the group XTAG at University of Pennsylvania on TAGs (normalization and parsing evaluation) and MetaGrammars. We plan in some near future to establish a more formal cooperation within NSF-INRIA program.

8.2.5. ISO subcommittee TC37SC4

The participation of ATOLL to French Technolangue action Normalangue has resulted in a strong implication in ISO subcommittee TC37 SC4 on the normalization of linguistic resources (http://www.tc37sc4.org/). É. de la Clergerie and L. Clément have participated to ISO events (TC37SC4 meeting at Sapporo with É. de la Clergerie as leader of the French delegation; TEI/ISO joint meeting in Nancy on Feature Structures) and have played a role of experts (in particular on Morpho-Syntax and Feature Structures).

8.3. Visits and invitations

Joint invitation with project COQ of A. Nait Abdallah.

Two weeks visit of Gabriel Pereira Lopes in November 2003 (action INRIA-ICTII FASTLING)

One month visit of Francisco Jose Ribadas Pena in December 2003 (action PICASSO CATALINA-2)

9. Dissemination

9.1. Animation at INRIA

B. Lang is an elected member of INRIA's "Conseil Scientifique".

É. de la Clergerie has participated to the INRIA Rocquencourt "section d'audition" for the 2003 CR2 recruitment campaign.

9.2. Supervising

É. de la Clergerie has supervised the internships of Angélique Pochon [18], Julien Lafaye [15], and Pascal Manchon [17]. He also co-supervises the PhD thesis of Benoît Sagot with Laurence Danlos (TALaNa/LATTICE, University Paris 7).

9.3. Jury

- B. Lang is a member of the CNAM expert committee for computer science courses.
- É. de la Clergerie is a member of the recruitment committee of University of Orléans.
- É. de la Clergerie has been a member of the Dissertation Proposal Defense jury for Alexandra Kinyon (University of Pennsylvania, April 28th 2003).

9.4. Teaching

Starting December 2003, L. Clément has delivered courses (40h) on XML in DESS "Gestion de documents électroniques et de flux d'information" (management of electronic documents and information flows) at University Paris X.

9.5. Committees

- Participation of É. de la Clergerie to the editorial board of French journal T.A.L. http://www.atala.org/tal/tal.html and Guest Editor of a T.A.L. issue 44/3 on "Evolutions in Parsing", to appear in 2004 (because of delays).
- Participation of É. de la Clergerie to both program and organization committees for IWPT'03 (*International Workshop on Parsing Technologies*) and participation to the Program Committee for TALN'04, the French national conference on NLP. Reviews for EACL'03, TALN'03, and IWPT'03.
- Participation of P. Boullier to the Program Committee of the *8th Meeting on Mathematics of Language* (MoL8, Bloomington, Indiana, June 20-22, 2003). He has reviewed papers for the international conference on *Principle of Programming Languages*, (POPL'04) and for the *Journal of Logic, Language and Information*, (JoLLI).
- Participation of B. Lang to program committees for several professional events.

9.6. Participation to workshops, conferences, and invitations

- Participation and contribution of B. Lang to several meetings on the potential of Open Source Software and on their economical impact. He has talked in several events about the notion of intellectual property, especially in regard of software development or scientific publishing.
- Participation of É. de la Clergerie and L. Clément to ISO TC37SC4 meeting (Sapporo, Japan, July 2003) and ACL'03 workshop on Linguistic Annotation (LINGAN, Sapporo, July 2003). Participation to a joint TEI-ISO meeting on "Feature Structure Representation" (FSR, Nancy, November 2003).
- Participation of É. de la Clergerie to IWPT'03 (Internation Workshop on Parsing Technologies) and ACL'03 (Association for Computation Linguistics).
- Participation of L. Clément to ACL'03, FG'03 (Formal Grammars), and to MTT'03 (Meaning-Text Theory). Presentation at FG'03 [10].
- Participation and contribution of P. Boullier to IWPT'03 [8][9].
- Participation of B. Sagot to the 15th European Summer School in Logic Language and Information (ESSLLI'03) and to MTT'03. He has also presented his work during a "Journée TALaNa" at Lattice/TALaNa (University Paris 7).
- One week visit of É. de la Clergerie at University of La Coruña (PAI PICASSO CATALINA-2) and another visit at Lisbon New University (INRIA-ICTII FASTLING).
- A few days visit of É. de la Clergerie at University of Pennsylvania with a presentation on Thread Automata [19].
- Invitation of L. Clément at the University of Louvain la Neuve (Belgium) to deliver a talk on "annotation d'un corpus de référence pour la morphosyntaxe" (Morphosyntactic annotation of a reference corpus). L. Clément has also delivered a presentation on "Écriture factorisée d'une grammaire du français" (Factorized design of a grammar for French) at LIMSI (Orsay).

10. Bibliography

Major publications by the team in recent years

[1] P. BOULLIER. *A Cubic Time Extension of Context-Free Grammars.* in « Grammars », number 23, volume 3, 2000.

[2] P. BOULLIER. *On TAG Parsing.* in « Traitement Automatique des Langues (T.A.L.) », number 3, volume 41, 2000, pages 111-131, issued June 2001.

[3] B. LANG. *Complete Evaluation of Horn Clauses: an Automata Theoretic Approach.* Technical report, number 913, INRIA, Rocquencourt, France, November, 1988, http://www.inria.fr/rrrt/rr-0913.html.

[4] B. LANG. *Towards a Uniform Formal Framework for Parsing.* M. TOMITA, editor, in « Current issues in Parsing Technology », Kluwer Academic Publishers, 1991, chapter 11, also appear in the Proc. of Int. Workshop on Parsing Technologies - IWPT89.

[5] É. VILLEMONTE DE LA CLERGERIE. *Parsing Mildly Context-Sensitive Languages with Thread Automata.* in « Proc. of COLING'02 », August, 2002, ftp://ftp.inria.fr/INRIA/Projects/Atoll/Eric.Clergerie/COLING02.pdf.

[6] É. VILLEMONTE DE LA CLERGERIE. *Automates à Piles et Programmation Dynamique. DyALog : Une application à la programmation en Logique.* Ph. D. Thesis, Université Paris 7, 1993.

[7] É. VILLEMONTE DE LA CLERGERIE, M. A. ALONSO PARDO. *A tabular interpretation of a class of 2-Stack Automata.* in « Proc. of ACL/COLING'98 », August, 1998, ftp://ftp.inria.fr/INRIA/Projects/Atoll/Eric.Clergerie/SD2SA.ps.gz.

Publications in Conferences and Workshops

[8] P. BOULLIER. *Guided Earley Parsing.* in « Proceedings of the 8th International Workshop on Parsing Technologies (IWPT 03) », pages 43–54, Nancy, France, April, 2003, ftp://ftp.inria.fr/INRIA/Projects/Atoll/Pierre.Boullier/earley_final.pdf.

[9] P. BOULLIER. *Supertagging: A Non-Statistical Parsing-Based Approach.* in « Proceedings of the 8th International Workshop on Parsing Technologies (IWPT 03) », pages 55–65, Nancy, France, April, 2003, ftp://ftp.inria.fr/INRIA/Projects/Atoll/Pierre.Boullier/supertaggeur_final.pdf.

[10] L. CLÉMENT, A. KINYON. *Automating the generation of a wide-coverage LFG for French using a MetaGrammar.* in « Proc. of Formal Grammars (FG'03) », pages 33–46, 2003.

[11] L. CLÉMENT, A. KINYON. *Generating parallel multilingual LFG-TAG grammars from a MetaGrammar.* in « Proc. of ACL'03 », 2003.

[12] L. CLÉMENT, A. KINYON. *Generating LFGs with a MetaGrammar.* in « Proc. of Lexical Functional Grammars (LFG'03) », CSLI Publications, 2003.

[13] N. IDE, L. ROMARY, E. VILLEMONTE DE LA CLERGERIE. *International Standard for a Linguistic Annotation Framework.* in « Proceedings of HLT-NAACL'03 Workshop on The Software Engineering and Architecture of Language Technology », 2003, Journal version submitted to the special issue of JNLE on Software Architecture for Language Engineering.

Miscellaneous

[14] L. CLÉMENT, B. SAGOT, B. LANG. *Morphology based acquisition of large-coverage lexica.* November, 2003, submitted to LREC'2004.

[15] J. LAFAYE. *Bases de données XML pour des ressources linguistiques.* Stage X, DIX – École Polytechnique, July, 2003, ftp://ftp.inria.fr/INRIA/Projects/Atoll/Eric.Clergerie/Lafaye-X03.ps.gz.

[16] K. LEE, H. BUNT, S. BAUMAN, L. BURNARD, L. CLÉMENT, E. DE LA CLERGERIE, T. DECLERCK, L. ROMARY, A. ROUSSANALY, C. ROUX. *Towards an international standard on feature structures representation.* November, 2003, submitted to LREC'2004.

[17] P. MANCHON. *Structuration de Documents.* Stage X, DIX – École Polytechnique, July, 2003, ftp://ftp.inria.fr/INRIA/Projects/Atoll/Eric.Clergerie/Manchon-X03.ps.gz.

[18] A. POCHON. *Intégration d'une méthode d'acquisition de terminologie et recherche de relations.* Mémoire de

DEA, LIFO – Université d'Orléans, September, 2003, ftp://ftp.inria.fr/INRIA/Projects/Atoll/Eric.Clergerie/pochon-DEA03.ps.gz.

[19] É. VILLEMONTE DE LA CLERGERIE. *Thread Automata for Mildly Context-Sensitive Languages.* May, 2003, ftp://ftp.inria.fr/INRIA/Projects/Atoll/Eric.Clergerie/UPenn03-slide-small.ps.gz, Slides presented at IRCS, University of Pennsylvania.

Bibliography in notes

[20] M.-H. CANDITO. *Organisation modulaire et paramétrable de grammaires électroniques lexicalisées.* Ph. D. Thesis, Université Paris 7, January, 1999.

[21] B. CARPENTER. *The Logic of Typed Feature Structures with Applications to Unification Grammars, Logic Programs and Constraint Resolution.* number ISBN 0-521-41932, Cambridge University Press, 1992.

[22] S. EARLEY. *An Efficient Context-Free Parsing Algorithm.* in « Communications ACM 13(2) », ACM, 1970, pages 94-102.

[23] R. M. KAPLAN, J. BRESNAN. *Lexical-Functional Grammar: A formal system for grammatical representation.* J. BRESNAN, editor, in « The Mental Representation of Grammatical Relations », The MIT Press, Cambridge, MA, 1982, pages 173-281, Reprinted in Mary Dalrymple, Ronald M. Kaplan, John Maxwell, and Annie Zaenen, eds., *Formal Issues in Lexical-Functional Grammar*, 29-130. Stanford: Center for the Study of Language and Information. 1995..

[24] M.-J. NEDERHOF, G. SATTA, S. SHIEBER. *Partially ordered multiset context-free grammars and free-word-order parsing.* in « In 8th International Workshop on Parsing Technologies (IWPT'03) », pages 171–182, April, 2003.

[25] F. PEREIRA, D. WARREN. *Parsing as Deduction.* in « Proc. of the 21st Annual Meeting of the Association for Computationnal Linguistic », pages 137-144, Cambridge (Massachussetts), 1983.

[26] C. POLLARD, I. A. SAG. *Head-Driven Phrase Structure Grammar.* University of Chicago Press, Chicago, 1994.