Constraint Event-Driven Automated Reasoning

Hassan Aït-Kaci*
hak@acm.org

October 2012

Abstract

This is a presentation of the work plan for the research project $\mathcal{CEDAR}$ (Constraint Event-Driven Automated Reasoning)[1] to be carried out over a two-year period at the *Laboratoire d'InfoRmatique en Image et Systèmes d'information* (LIRIS) of the *Université Claude Bernard Lyon 1* (UCBL) under a grant by the *Agence Nationale pour la Recherche* (ANR) as part of its Chair of Excellence program (CHEX 2012) and the UCBL. The purpose of this document is to define the schedule of actions to take place after the planned start of the project.[2] The objective is to present the project's scientific contents, but also to define administrative tasks that need to be carried out to ensure that the project start off in the best possible conditions and reach productive working status according to the planned agenda. This document is also meant to provide background information on the $\mathcal{CEDAR}$ project to potential collaborators.

Keywords: Ontologies; Semantic Web; Knowledge Representation; Automated reasoning; Distributed Knowledge; Knowledge-Based Technology; Reasoning Software Tools

*http://www.hassan-ait-kaci.net/
[1]In French: "*Contraintes et Evènements Dirigeant l'Automatisation du Raisonnement.*"
[2]January 15, 2013

Contents

1 Introduction

This section overviews the contents of this document. Section 1.1 defines its purpose, and Section 1.2 summarizes its organization.

1.1 Utility of this document

This document describes the work the author has proposed for an experimental research project on Constraint Event-Driven Automated Reasoning ($\mathcal{CEDAR}$). It is to be carried out at the *Laboratoire d'InfoRmatique en Image et Systèmes d'information* (LIRIS) of the *Université Claude Bernard Lyon 1* (UCBL) under a grant by the *Agence Nationale pour la Recherche* (ANR) as part of its Chair of Excellence program (CHEX 2012) and the UCBL. The author is the recipent of the Chair for two years, to start on January 15, 2013, ending on on January 15, 2015. The local host and primary technical collaborator of the Chair on the subject of the $\mathcal{CEDAR}$ project is Prof. Mohand-Saïd Hacid, full professor of Computer Science at UCBL.[3]

1.2 Organization of document

The rest of this document is organized as follows. In Section 2 we describe the context and position of the project. In Section 3 we give a scientific and technical description of the objective we propose to achieve. Section 4 we delve into some details of the scientific and technical program that we intend to carry out to achieve these objectives. Section 6 recapitulates the gist of our program and its expected schedule of delivery. An appendix follows that contains additional relevant background information on the participants and collaborators of the $\mathcal{CEDAR}$ project.

2 Context and Position of the Project

This section positions the project in the current scientific context. Section 2.1 describes today's main challenges concerning distributed knowledge processing and automated reasoning. Section 2.2 makes general statements justifying how our specific approach addresses these challenges.

2.1 Context of the project

From what existing systems have to offer for ontology processing in terms of RDF querying optimization such as used by SPARQL (see *e.g.*, [30, 32]), it appears that such technology is still in its prime infancy. Knowledge bases in the format of RDF-graph databases must be queried based on relational-data processing technology. However, the nature of queries in either context is not the same, and the differences between the two data storage models have often the unfortunate effect of producing severe performance bottlenecks [32].

Although several promising formalisms and technologies have been proposed in the context of the emerging Semantic Web, the main unresolved issue today remains to **develop a set of techniques for distributed knowledge processing that would be on a par with those that have evolved and**

[3] See Appendix Section D.1.

matured as processing tools for relational databases. Although much of the latter may be relevant and adaptable to the new challenges, there are key questions in knowledge processing that necessitate that novel formal models and algorithmic techniques be conceived, tested, and deployed. The necessity of testing is paramount because all such proposals so far have failed delivering systems that can scale up to the size of existing or expected knowledge bases.

The first obstacle, like the one that originally plagued DBs until the Relational Model and Calculus were proposed by Codd, has been for all to agree on a common simple and expressive KB representation format. This issue seems to have now been overcome, at least in principle, with the spreading adoption of the W3C's Resource Description Framework (RDF), and its distributed extension: Linked Data. Other proposals aiming at endowing Semantic Web (SW) knowledge with actual reasoning capabilities—*e.g.*, $\mathcal{OWL}$ and its variations based on Description Logic (DL)—have yet to prove their worth when used on astronomical amounts of distributed evolving knowledge.

Thus, the challenge we address is to **define, prototype, and test such a system that is (1) *formal*, (2) *operational*, (3) *efficient*, (4) *scalable*, and (5) *extensible*.**

The author pioneered the use of Order-Sorted Featured (OSF) constraints in KR, field traditionally using logical or graphical formalisms. His motivation is that most, if not all, KR formalisms developed in AI have been variations on labeled-graph notations taking meaning in some logic interpretation. With the advent of the Semantic Web, there has been a growing tension between proposed standard formal notations describing knowledge, on one hand, and the proof theory for these notations on the other hand. While formal notation to express knowledge has been striving for simplicity, this is usually offset by the algorithmic complexity and implementation of reasoning using this notation (syntax-directed semantics). Even when this tension is somehow resolved through clever techniques, these do not scale up to very large KBs. Indeed, the main challenge faced by any KR system today, whatever its formal basis may be, is managing effective reasoning over KBs of enormous size distributed over all sorts of networks and repositories. While classical DB technology has been useful for many parts, there are important differences between the essential natures of the two worlds (data *vs.* knowledge) to warrant a new approach tailored to the idiosyncrasies of Knowledge Base management. This is essentially what the author's research has contributed to: reconciling the most intuitive and popular representations of knowledge (*i.e.*, labeled graphs) with formal semantics including efficient and scalable proof methods based on graph-constraint normalization. This proposal's essential objective is to put this claim to the test.

2.2 Position of the project

Véritable mémoire du temps, le cèdre de l'Atlas nous raconte l'Histoire . . .

OMAR M'HIRIT—*Le Cèdre de l'Atlas : Mémoire du Temps*

Our essential motivation in the $\mathcal{CEDAR}$ project is the systematic study of, and experimentation with, an approach to Knowledge Representation (KR) that is an alternative to what has prevailed so far. The two main challenges to overcome for the coming to pass of the Semantic Web are (1) *scalability* and (2) *distribution*. The problem of scalability is that a well-designed web-oriented KB system must be able to handle larger and larger volumes of knowledge without unbearable degradation of performance. Dealing with the second challenge—distribution—is as complex an issue since it must deal efficiently and seamlessly with knowledge spread all over the net under "real-life" conditions (cache faults, handling faulty connections and time delays, query distribution, *etc.*, ...). We believe that a key to a satisfactory handling of both challenges is offered by the $\mathcal{OSF}$ constraint approach. The reason of our belief is that, contrary to most mainstream approaches to the Semantic Web, the $\mathcal{OSF}$ constraint formalism is operationally lazy (*i.e.*, it does not do anything that is not needed), endowed with instant (*i.e.*, 0-cost) "memo-sorting" (*viz.*, proof-caching sorts) [3], and capable of handling very large concept hierarchies using modulated binary encodings [4] and techniques taking advantage of the specific structure of the $\mathcal{OSF}$-graphs making up a KB [17, 16, 15]. One of the most important objectives of this proposed work is to test our work on existing benchmarks and realistic simulations.

Using the $\mathcal{OSF}$-constraint approach, we propose a declarative programming system based on structural and temporal constraint-solving compatible with the proposed W3C standards (RDF and Linked Data). Such a system is to be exploited in a distributed KB context in real-time—hence the essential necessity of temporal constraint-solving—and thus needs to be "event-aware" according to the emerging event-processing standards.

What we propose can be characterized as a synthesis of various prior work in AI, KR, and Constraint-Logic Programming ($\mathcal{CLP}$) by the author and others in the field of constraint-based processing, with the latest technology for maintaining and accessing distributed knowledge in the new context of interlinked media. The adoption of RDF as a W3C Semantic Web standard for a universal triple-based idiom expressing graph-format knowledge happens to be another serendipitous timely justification. Indeed, the formal basis upon which the essentials of $\mathcal{OSF}$ constraint-solving relies uses exactly the same basic universal labeled-graph representation. This is explained next.

Over the several past decades, a simple and general formal data structure has proven to be pervasively adopted in almost all venues of Computer Science—namely, fielded classes and objects populating them. Such data structures have also turned out to be most adequate for KR and NLP. Thus, contributions in AI, DB, and KB have formalized such structures as labeled graphs. A standard light-weight (*i.e.*, non-XML) notation for encapsulations of attribute/value pairs, such as object records and classes, is lately being specified—the JavaScript Object Notation (JSON) and its Linked Data version (JSON-LD).

One particular approach (that we advocate) is to see such graphs as very simple and easily enforceable constraints—and this, essentially for practical reasons. It allows representing and manipulating graph-based objects (*e.g.*, record types) as order-sorted featured objects that is simple, efficient, and practical. This formalism is a basic formal and effective constraint-based rendition of the essential informal insights underlying Semantics Networks of the 80s and 90s. The most interesting aspect of it is that it offers a direct interpretation of labeled graphs representing structural knowledge as efficiently solvable constraints. Reasoning with large and complex structures is done by interpreting such graphs as conjunctive or disjunctive sets of elementary constraints. Moreover, it turns out that these elementary graph constraints map naturally into a triple-based representation such as offered by RDF—proposed by the W3C as the universal format to represent all SW knowledge on Internet—and heirs of RDF, such as RDF Schema, RDFa, LinkedData, SKOS, *etc.*, [33, 11, 13, 1, 24, 25, 28].

In simple terms, $\mathcal{OSF}$ technology provides a set of formal and practical tools appropriate for the Semantic Web. As it evolved out of algebraic term unification, the $\mathcal{OSF}$ approach lends itself also to efficient implementation as it can be compiled into machine-level instructions [5, 8]. Moreover, these basic instructions are seen themselves as elementary constraints. This renders possible a generic abstract run-time machinery by simple instantiation of independent constraint solvers over constrained logical variables. The idea is to exploit years of research into such execution models (*e.g.*, $\mathcal{LP}$, $\mathcal{CLP}$, LIFE [2], Oz [31], *etc.*) using and adapting the techniques to work with current standards where appropriate. The objective is not to rebuild the universe, only better. Rather, it is to take advantage of a few techniques that were invented, prototyped, and implemented mostly in the context of $\mathcal{CLP}$ languages and systems, and exploit them in the context of the Semantic Web.

Today, the overwhelming majority of efforts aiming at supporting reasoning for the Semantics Web have been variations on the Ontology Web Language ($\mathcal{OWL}$)—*i.e.*, using the formalism of Description Logics (DL). In addition, very few have addressed the most essential challenge posed as "real-life" knowledge materializes in larger and larger amounts and distributed over scattered sources [30, 32, 17].

Recently, use of the $\mathcal{OSF}$ formalism in areas relating to distributed knowledge management has been shown useful by some of the participants of this project [27, 26, 12, 29, 23, 14, 22].

3 Scientific and Technical Description

In this section, we position the $\mathcal{CEDAR}$ project's scientific and technical contents with respect to current knowledge. Section 3.1 reviews the state of the art. Section 3.2 describes how $\mathcal{CEDAR}$'s objectives are expected to innovate in this context.

3.1 State of the art

With the advent of the Semantic Web and the W3C's commitment to make it a reality, the predominantly cited software enabling actual KR reasoning has been the $\mathcal{OWL}$ family of languages. These languages are based on a set of techniques that are variations of bottom-up finite-model building. These methods are variations on tableau-based reasoning techniques and make the operational basis for DLs.[4] Technically, tableau-based reasoning works by building the extension of a DL-sentence—its model—following a least fixed-point semantics. It is precisely because $\mathcal{OWL}$ reasoning works by actually populating a sentences model that it is limited to finite-domain formulas.

Independently, another approach to KR evolved out of unification-based computing—the order-sorted feature graph constraint formalism. It too has a logical semantics—but that can be formalized as a constraint-based logic. In addition, its operational semantics may be realized using lazy proof methods. Unlike tableau-based techniques, these methods do not prove a descriptive sentence by building its model. Instead, they work by "simplifying" notation (its syntax) into normal form. Normalized forms can be one of three kinds: (1) solved, (3) inconsistent, or (3) undecided. This latter form arises when no part of the graph-represented constraint may be simplified hence the "laziness." Formally, these techniques follow a greatest fixed-point semantics. Contrary to DL-based reasoning, $\mathcal{OSF}$-reasoning does not prove a formula by building its model; it simply keeps formulas in normal (not necessarily solved) form. Those normal forms that are recognized as solved forms denote all the solutions in intension. In other words, there is no need systematically to enumerate the elements populating a solution. It works by keeping approximations consistent. This allows formulas denoting both finite and infinite models.

Once understood, the formal relation between DL and $\mathcal{OSF}$ proof techniques makes it clear why the former faces formidable challenges in order to scale up to sentences denoting large or infinite models, while the latter does not since it relies on constraint normalization—*i.e.*, syntax simplification rules.

[4] $\mathcal{OWL}$ and related formal languages are all DL-based.

The price to pay for such efficiency is some form of incompleteness as reduction to normal-form may yield inconclusive forms. The remaining forms—the residual unsolved constraints—are called *residuations* [2]. One of the most important properties of the $\mathcal{OSF}$-constraint normalization proof system is that it is an *incremental* system. In other words, approximation and implication commute. If any residuation remains with no further progress possible, it is possible (1) to "*fail*," as unification would; (2) to answer "*maybe?*," modulo residuations; or, (3) to submit the pending residuations to a special-purpose constraint solver if one exists for their type of constraints, or to several solvers for each subset of constraints of a given type, each solver communicating with others only through shared Prolog-like logical variables.

Like data, knowledge is not static. Conjugating the time dimension into the structural ones, it is possible to make graph-based constraints controlable by temporal event-triggered agents. Interestingly, the reasoning power needed for the realistic orchestration and choreography of such events can also be formalized using constraint-based reasoning (most notably, soft temporal constraint-solving [10] and Ken Kaneiwa's work [18, 20]). Dynamic schema evolution must always be kept faithful to the knowledge actually encoded. Although it is expected that most DB maintenance and operation technology can—and will—be reusable or easily adaptable to the KB needs, there will also be notable differences [30, 32].

Last, but not least, $\mathcal{OSF}$ graph-constraint technology has been at work with great success in two essential areas of AI: NLP and Machine Learning:

- Interestingly, though not surprisingly, the formal approach we advocate for expressive knowledge representation and efficient implementation thereof (using $\mathcal{OSF}$ constraints) based on "featured-object with type inheritance" has been a major paradigm in the field of NLP for a long time [6]—so-called "Head-driven Phrase Stucture Grammar" (HPSG) and Unification Grammar technology. This is indeed not surprising given the ease with which feature structure unification enables combining both syntactic and semantic information in a clean, declarative, and efficient way.
- Similarly, while most of the attention in the $\mathcal{OSF}$ literature has been devoted to unification, an operation that basically computes the most general $\mathcal{OSF}$ term subsumed by two given $\mathcal{OSF}$ terms, the dual operation—namely, generalization—is just as simple to use, which computes the most specific $\mathcal{OSF}$ term that subsumes two given terms (see also [9]). This operation is central in Machine Learning and with it $\mathcal{OSF}$ technology lends itself to be combined with popular Data Mining techniques such as Support Vector Machines using frequency or probabilistic information (*e.g.*, [7]).

3.2 Objectives and innovative nature of the project

The objectives of the $\mathcal{CEDAR}$ project are:

1. **to develop, prototype, and test a constraint-based approach to Knowledge Representation (KR) and automated reasoning** where all knowledge is expressed in a universal graph-based representation format such as RDF (*e.g.*, Linked Data), in the same manner as all data has been represented as tables in the Relational Model;
2. **to enable such a constraint-based system to handle time-aware reasoning using multiple knowledge sources for event-driven computing in a distributed KB context**, where the environment evolves in real-time (*e.g.*, intelligent adaptive Quality-of-Service monitoring, maintaining evolving KBs, reconciling distributed KBs, *etc.*, ...).

Attaining these objectives project will be a contribution in an essential area, with original and innovative results of important potential—**offering scalable Semantic Web processing over distributed KBs** and tested on challenging benchmarks, thanks to a formal basis that differs from that of most similar pursuits.

Our overall technical goal is to provide a tangible and testable proof that the $\mathcal{OSF}$-constraint logic approach to knowledge representation can:

1. be expressed and used computationally on the emerging standard representation format for all Semantic Web knowledge bases (RDF triple-based)—and used both for expressing and maintaining structural and temporal constraints;
2. through testing and simulation, experiment with architectural issues in managing and accessing distributed RDF-based knowledge that must be scalable;
3. use, test, and demonstrate the new $\mathcal{OSF}$-constraint engine on actual RDF-expressed knowledge by using the most efficient architecture as indicated per simulation benchmarks for ***scalable distributed knowledge processing***.[5]

The work proposed in this project must innovate in addressing two essential technological challenges regarding triple-based ontologies: knowledge reasoning and managing.

(1) The key innovation of the proposed work is that its ontological reasoning technology is to be based on $\mathcal{OSF}$-graph Constraint Logic rather than Description Logic like the $\mathcal{OWL}$-family of KB representation and reasoning that constitute the majority of extant KB systems. The originality of $\mathcal{OSF}$-graphs is that they map directly to RDF and to formal constraints that may be interpreted both as structural and temporal constraints (for the latter, see Ken Kaneiwa's work [18, 19, 20, 21]).

(2) The other essential contribution of this project is the management of very large amounts of distributed triple-based knowledge. It is to experiment with low-level organization and optimization, through testing and simulation, of the RDF-represented KBs upon which an $\mathcal{OSF}$-constraint processing engine could be used.

In both (1) and (2), techniques for scalable KB processing will concentrate on the specific nature of knowledge as expressed in ontologies—namely concept inheritance lattices, lazy and cached proofs, and other such optimization (partial-order modulated binary encodings, proof inheritance [4, 17, 16, 15]).

The project's success measure will be in demonstrating the outcome of (1) to be fully operational on actual benchmarks thanks to the results of (2).

[5] It is interesting to note that results from such experiments can be used as the basis for synthesizing structural and behavioral knowledge from data into ontology processing systems endowed with our constraint-based reasoning using appropriate knowledge-mining techniques (both symbolic and statistical), and deploy exemplars of such self-learning self-evolving systems.

4 Scientific and Technical Program

Il ne faut pas trop confondre le savoir-faire et le faire-savoir
Ou alors le bien être se transforme en mauvais-avoir

JACQUES PRÉVERT—*Spectacles*

This section gives the scientific and technical details making up the $\mathcal{CEDAR}$ project's program. Section 4.1 presents the project's structure. Section 4.2 explains the project's management. Section 4.3 describes the project's tasks.

4.1 Project structure

Figure 1 shows the overall temporal structure of the $\mathcal{CEDAR}$ Project and a graph diagram showing its breakdown into tasks and subtasks and their mutual dependencies. The managerial and technical contents of each task are then discussed in detail in the remainder of Section 4.

4.2 Project management

Task T0 will be dedicated to the coordination of the project. The Project Chair—or PC—(Hassan Aït-Kaci) will see to effective cooperation and communication among all the project participants, as well as be responsible of all final decision-making. As explained in detail below, the nature of the proposed project is a two-tracked parallel format—a language design track (Task T1) and an experimental simulation track (Task T2). For this reason, everyday task management team will consist of a dual team: the Project Chair and the Experiment Coordinator—or EC—(Mohand-Saïd Hacid), the host professor of the proposed Chair at Université Claude Bernard Lyon 1. The latter will assist the Project Chair in overseeing all experimental work of Task T2 and ensuring its adequation with the specified need of Task T1. However, the Chair will be responsible for the general operation of the project, and the final decision-maker in all technical matters. The Chair, the Experiment Coordinator, and the nominated task leaders will be in charge of monitoring the progress of the research activities, as well as making decisions regarding the global orientation of the project. The Chair already has a solid and proven experience in high-profile technical AI and KR language research project management. The Experiment Coordinator has recognized expertise in the setup, conduct, and analysis of massive distributed knowledge and data bases. All other junior members of the team are to be selected upon proven solid successful experience in managing projects and various collaborations, among them international collaborations.

The Chair will also be in charge of the general organisation and management of the project. He will be supported in these responsibilities by Lyon Ingénierie Projets (LIP), which is a subsidiary of UCBL dedicated to the management of research projects.

In addition to the Project Chair and the Experiment Coordinator, it is planned to hire two junior postdoctoral-level research personnel to put in charge of the daily technical operation for each of the two tracks of the project. Ideally they should have background and experience in complementary fields—one proficient in AI language design and implementation (Language Track Manager or LTM),[6] and the other in DB and KB architecture and management (Experiment Track Manager or ETM).[7]

[6] See Appendix Section C.1.1 for the LTM's expected profile.
[7] See Appendix Section C.1.2 for the ETM's expected profile.

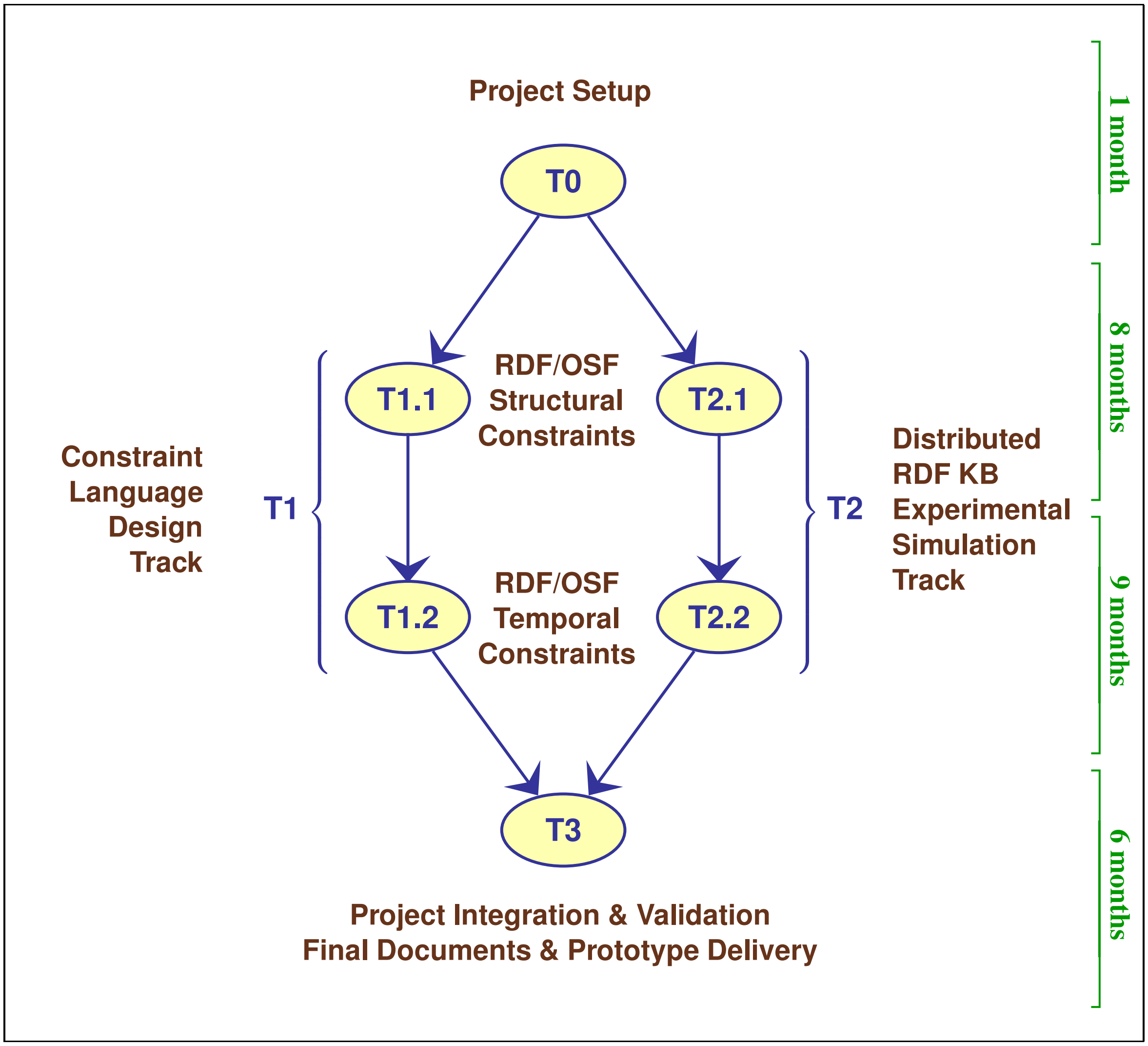

Figure 1: The $\mathcal{CEDAR}$ Project Structure

The Project Chair together with the Experiment Coordinator, assisted by both track managers (LTM and ETM), will be in charge of the daily follow-up of the project progress towards the planned objectives or the project, including the delivery, adoption and approval records upon achievement of each milestone. They will meet on a regular basis. The Chair can decide, if needed, to organise an extraordinary meeting or to consult with the participants when urgent decisions have to be made.

Management Team. The specifically dedicated project management team will consist of the Project Chair (Hassan Aït-Kaci), assisted by the project host and Experiment Coordinator (Mohand-Saïd Hacid), and of Lyon Ingénierie Projets. The team will be in charge mainly of being a permanent contact point for the Chair and all project participants; assisting the Chair on administrative tasks and notifying the team of due dates; reminding the participants of the deadlines for the deliverables; managing delivery and follow-up of administrative and financial documents; preparing and maintaining contractual documents (agreements, contract, annex, *etc.*, . . .); following up the budget; informing and reminding participants of deadlines, if any; being a help desk.

All the participants must be aware that communication is the key for the successful completion of the project, and are expected always to abide carefully by all management decisions, report faithfully and regularly all situations, and consult higher advice in case of misunderstanding, problem, or conflict.

4.3 Description of the tasks

Besides the project-long task of management work (Task T0), the technical activities of the $\mathcal{CEDAR}$ project are organized into three mains tasks : T1, T2, and T3. Task T1 and Task T2 are planned to run in parallel, with some mutual interaction. Task T3 will consist in the synthesis of the outcomes of Tasks T1 and T2.

Task T1 will be responsible for the knowledge and reasoning language design and conformity with W3C's Semantic Web RDF-related standards. Its essential focus will be on providing the formal reasoning engine in the form of a constraint-handling rule language and software for structural and temporal knowledge as expressed by the W3C standards (RDF, RDFS, RDFa, SKOS, Linked Data, *etc.*) and the emerging event-processing standards for Complex Event Processing.

Task T2 will be responsible for experimental testing/simulation of an RDF-oriented knowledge-base architecture design with objective to support efficient and scalable access to, and manipulation of, distributed knowledge expressed conformingly to W3C standards.

Task T3 will consist of operating the language design resulting from Task T1 on top of a distributed KB architecture optimized as justified by Task T2, and to report and document all the results. It will also be the task seeing to the proper delivery of all software and related documentation, as well as recapitulating the $\mathcal{CEDAR}$ project's deliverables and how to access any such information.

The task schedule, its components, and their dependencies, are illustrated in Figure 1 and further described in detail in the remainder of this section. First the project's management Task T0 is described, then for Tasks T1 and T2, the technical points are developed. Tasks T1 and T2 running in parallel will tackle the following, respectively:

1. Task T1—Language Track: RDF-representation and reasoning engine design issues,
2. Task T2—Experimental Track: RDF-based architecture and deployment issues.

Finally, we describe how Task T3 will focus on combining the results of Tasks T1 and T2.

Task T0—Project setup, coordination, and management

T0.1–1 month: PC 5 d/w + EC 3 d/w—T0.2–23 months: PC 1 d/w + EC 1 d/w + LTM 1 d/w + ETM 1 d/w

In its initial phase (one month), this task will be devoted to finding and training the technical staff needed for the tasks that will be expected of them, and to defining and explaining the project's basic administrative and organizational work plan to all involved. This will also define a protocol of how the two tracks will interact.

This task will also run during the whole duration of the project, seeing to all administrative requirements, accounting responsibility, day-to-day management, and effective coordination of all planned technical tasks—including web page management, progress reports, expense reporting, publications/patent management, document and software deliverables, and all general administrative issues.

The rest of this section focuses on describing and justifying the contents of the project's technical tasks and their temporal dependency. For space-saving reasons, each task's deliverable's details are given in Section 4.4 in Tables 1–6, which sum up all the $\mathcal{CEDAR}$ project's management calendar and technical information per task.

Task T1—Language track

17 months: PC + LTM + 2 Interns

Task T1.1—Structural constraint-graph language track

8 months: PC 3 d/w + LTM 4 d/w + Intern 5 d/w

T1.1.a

2 months: PC 3 d/w + LTM 4 d/w + Intern 5 d/w

Specify and prototype a detailed and systematic framework for expressing the essentials of $\mathcal{OSF}$ Constraint Logic in a computer-readable syntax and map this syntax into an internal RDF-based representation, as well as a reverse operation generating more legible $\mathcal{OSF}$ notation from internal RDF-graph representations (or equivalent JSON or JSON-LD variations). This is to be planned as an incremental language design, depending on increased expressivity of RDF dialects [33, 11, 13, 1, 24]. This will have the vocation to serve as the basic practical paraphernalia for reading and writing $\mathcal{OSF}$ graphs.

- *Success Indicators*: Basic specification and initial prototyping going as expected.
- *Failure Fallbacks*: Adapt and use publicly available tools.

T1.1.b

3 months: PC 3 d/w + LTM 4 d/w + Intern 5 d/w

Enable the syntax now so parsable to be interpreted using $\mathcal{OSF}$-graph unification and matching capabilities made to work on their RDF/LD internal representation. Specify and prototype a compiling scheme for RDF-represented $\mathcal{OSF}$ structures. The main difference with Prolog-like implementations being (1) universal representation of $\mathcal{OSF}$ terms as RDF and extensions; (2) conceive the basic engine as truly abstract independently of any constraint system (clean separation of "relational" level and "constraint" level); (3) develop incrementally the operational semantics of the basic $\mathcal{OSF}$ constraint system on RDF-represented test sets, as well as the basic support for aggregations as monoid comprehensions, the most straightforward and relation-optimized effective calculus for managing collections of any kind (sets, bags, sums, products, unions, *etc.*, ...).

- *Success Indicators*: Basic specification and initial prototyping going as expected.
- *Failure Fallbacks*: Document technical issue and revise the task's objective accordingly.

T1.1.c

3 months: PC 3 d/w + LTM 4 d/w + Intern 5 d/w

Software documentation and debugging phase, developing a "user-friendly" syntax and GUI Eclipse interaction environment over a set of examples meant to illustrate its functionality. Write-up of initial versions for documentation for all software realization—both user-oriented and system-oriented. Write-up of research articles meant for publication describing and justifying the designs realized. Deliver a first version of the $\mathcal{CEDAR}$ knowledge-base language as a system interacting with RDF-based $\mathcal{OSF}$-graphs and the documentation for its use and design.

- *Success Indicators*: First $\mathcal{CEDAR}$ language implementation and documentation ready.
- *Failure Fallbacks*: Document technical issue and revise the task's objective accordingly.

Task T1.2—Temporal constraint-graph language track

9 months: PC 3 d/w + LTM 4 d/w + Intern 5 d/w

T1.2.a

3 months: PC 3 d/w + LTM 4 d/w + Intern 5 d/w

Specify and prototype a language extension design an implementation support for the $\mathcal{CEDAR}$ language based on the alternative interpretation of $\mathcal{OSF}$ terms for temporal event-related reasoning proposed by Ken Kaneiwa [18, 20]. The objective of this subtask is to integrate the dual semantics of $\mathcal{OSF}$ graphs both as structural constraints (as will be done in Task T1.1) but also as temporal constraints (as proposed by Ken Kaneiwa and others [10, 9]). The expected deliverable will be a version of the $\mathcal{CEDAR}$ $\mathcal{OSF}$ graph-constraint language compatible with RDF and JSON notation and offering both structural and temporal modes of reasoning using the $\mathcal{OSF}$ constraint formalism.

- *Success Indicators*: Basic specification and initial prototyping going as expected.
- *Failure Fallbacks*: Document technical issue and revise the task's objective accordingly.

T1.2.b

3 months: PC 3 d/w + LTM 4 d/w + Intern 5 d/w

Make the new capacities of time-aware $\mathcal{OSF}$ constraint graph structures adaptable and usable for the KB time-management protocols proposed by MS Hacid—especially for working on distributed time-management using $\mathcal{OSF}$ constraints for more efficient caching and pre-computing of very large KB

RDF-structures interpreted as $\mathcal{OSF}$ constraint-graphs. Initiate developing the necessary lowerlevel architecture structures and operations for making RDF-represented $\mathcal{OSF}$-graphs capable of taking advantage of time-aware optimization techniques for scalable KB processing devised along experiments and simulations done as part of Track 2.

- *Success Indicators*: Basic specification and initial prototyping going as expected.
- *Failure Fallbacks*: Document technical issue and revise the task's objective accordingly.

T1.2.c

3 months: PC 3 d/w + LTM 5 4/w + Intern 5 d/w

Finalize the design of the complete $\mathcal{CEDAR}$ language addressing all necessary issues encountered in Tasks T1.2.a and T1.2.b. Document in details both successful results and remaining issued. Focus on delivering a usable language that can already meet open challenges in handling distributed and scalable KB processing on existing benchmarks.

- *Success Indicators*: First $\mathcal{CEDAR}$ language prototyping going as expected.
- *Failure Fallbacks*: Document technical issue and revise the task's objective accordingly.

Task T2—Experimental track

17 months: PC 1 d/w + EC 2 d/w + ETM 4 d/w + 2 Interns 5 d/w

Task 2.1—Basic experimental track (structural architecture KB simulation)

8 months: PC 1 d/w + EC 2 d/w + ETM 4 d/w + Intern 5 d/w

We will need knowledge bases of different sizes and from different domains in order to evaluate the scalability. We will use both synthetic data sets and concrete data sets:

- Synthetic data sets: we will use the Berlin SPARQL BenchMark (BSBM)[8] generator to create data sets. The generator supports triples as an output format. We already generated some large data sets (105,124 triples, 10,036,982 triples, 189,905,757 triples, and 1,993,469,898 triples).[9] The target application area is e-commerce. An interesting point here is that one can generate *as many triples as desired*.
- Concrete data sets: we will use the GeoNames datatabase.[10] It contains geographic information about the entire world (eight million geographical names).

- *Success Indicators*: Efficiency compared to existing semantic-based reasoning systems.[11]
- *Failure Fallbacks*: Revise the approach.

[8] http://www4.wiwiss.fu-berlin.de/bizer/berlinsparqlbenchmark/
[9] In the framework of the AOC ANR project (http://aoc.irit.fr/).
[10] http://download.geonames.org/export/dump/
[11] http://www.seals-project.eu/

Task 2.2—Extended experimental track (temporal architecture KB simulation

9 months: PC 1 d/w + EC 2 d/w + ETM 4 d/w + Intern 5 d/w

The goal here is to provide a framework (benchmarks and tools) for an extensive evaluation of the resulting approach. Based on the performance indicators collected in task Task 2.1, we will work on additional requirements and criteria by which the approach will be assessed. We will work on the identification of useful performance indicators and specific metrics. The tests that will be employed will focus on the performance of fundamental aspects of the approach in some controlled scenarios.

- *Success Indicators*: Selected performance indicators are satisfied
- *Failure Fallbacks*: Indicators will be analyzed. They will provide the required information to assist in revising some fundamental aspects of the approach.

Task T3—Synthesis track

6 months: PC 4 d/w + EC 2 d/w + LTM 4 d/w + ETM 4 d/w + Intern 5 d/w

Task T3.1

Operate the $\mathcal{CEDAR}$ language branch design resulting from Task 1 on top of a distributed KB architecture organized and optimized as justified by Task 2 and experiment on its query performance on larger and larger KBs. Report behavior and adjust any issue on either language or architecture side as needed,

Task T3.2

Finalize all documentation, reports, and software packaging constituting the $\mathcal{CEDAR}$ project deliverables. Prepare and deliver a Project Results document giving a synthesis of the work accomplished, the results achieved, publications, and any issue relevant in the $\mathcal{CEDAR}$ project experience.

Task T3.3

Organize and host a $\mathcal{CEDAR}$ workshop on Scalable Distributed Knowledge Management, locally in Lyon. The technical motivation for such a small-scale event if to contribute to disseminating our work by sharing results with known experts in the field, as well as inseminate such leaders in the field with our own results and perspectives.

4.4 Task schedule, deliverables and milestones

Tables 1–6 below summarize the specific time schedules for each planned task and subtask, with their mutual dependencies (*viz.*, previous prerequisite tasks in the rightmost column).[12] They also specify each task's deliverables in the form of documents, software, and publications, as well as the expected duties of its participants.

[12] See the overall graphical task diagram in Figure 1.

Task	Task Label	Start of	End of	Deliverables	Participants	Prev.
T0	Project Setup	Month 1	Month 1	Staff and task assignments & task schedule administration setup	Project Chair + Experiment Coordinator	

Table 1: Schedule for Task T0

Task	Task Label	Start of	End of	Deliverables	Participants	Prev.
T1.1	1st phase of language and compiler design for $\mathcal{OSF}$ constraints on RDF format	Month 2	Month 9	**Documents**: formal specification report & implementation documentation **Software**: initial version of KR language for RDF with basic $\mathcal{OSF}$ reasoning **Publications**: one national workshop paper; one international conference paper	Project Chair + 1 Postdoc (LTM) + 1 Intern	T0

Table 2: Schedule for Task T1.1

Task	Task Label	Start of	End of	Deliverables	Participants	Prev.
T2.1	1st phase of experimental and testing for RDF-based KBs	Month 2	Month 9	**Documents**: formal specification report & implementation documentation **Software**: initial version of benchmarking and testing framework for scalable simulation of large distributed RDF KBs **Publications**: one national workshop paper and/or one international conference paper	Project Chair (for the specs) + Experiment Coordinator + 1 Postdoc (ETM) + 1 Intern	T0

Table 3: Schedule for Task T2.1

Task	Task Label	Start of	End of	Deliverables	Participants	Prev.
T1.2	2nd phase of language and compiler design extension with temporal constraint-handling	Month 10	Month 18	**Documents**: initial report on temporal-constraint handling with RDF-based $\mathcal{OSF}$ KR **Software**: extended KR language with time-aware $\mathcal{OSF}$ constraints on RDF KBs **Publications**: one national workshop paper; one international conference paper and/or one technical journal article	Project Chair + 1 Postdoc (LTM) + 1 Intern	T1.1

Table 4: Schedule for Task T1.2

Task	Task Label	Start of	End of	Deliverables	Participants	Prev.
T2.2	2nd phase of experimental and testing for RDF-based KBs	Month 10	Month 18	**Documents**: formal specification report & implementation documentation **Software**: extended framework for benchmarking and testing scalable simulation of large distributed event-controlled RDF KBs **Publications**: one national or international workshop paper and/or one international conference paper	Project Chair (for the specs) + Experiment Coordinator + 1 Postdoc (ETM) + 1 Intern	T2.1

Table 5: Schedule for Task T2.2

Task	Task Label	Start of	End of	Deliverables	Participants	Prev.
T3	Synthesis of language and experimental tracks	Month 19	Month 24	**Documents**: formal report on integrating scalable distributed RDF KB processing with time-aware $\mathcal{OSF}$ constraints + measured and comparative benchmark results **Software**: scalable CP language tested on large distributed event-controlled RDF KBs **Publications**: one international conference paper and one international technical journal article	Project Chair + Experiment Coordinator + 2 Postdocs (LTM & ETM) + 1 Intern	T1.2 + T2.2

Table 6: Schedule for Task T3

Table 7 gives a summary of the days per week (d/w) each staff member will be expected to spend on a given task. The headers for columms 3–11 indicate the staff member: PC for Project Chair, EC for Experiment Coordinator, LTM for Language Track Manager, ETM for Experiment Track Manager, Int_i for Intern $\#i, (i = 1, \ldots, 5)$. Interns will only be concerned with 6 months, each for the duration of a Master's project concerning the task where they appear. Each entry for each staff member is in number of days per week (d/w). Concurrent tasking is as indicated in the Task diagram of Figure 1.

5 Dissemination and Exploitation of Results

This section covers our plan for disseminating and exploting the expected contributions made in the course of the $\mathcal{CEDAR}$ project. Section 5.1 explains what scientific community at large will be targeted for publication and interaction. Section 5.2 describes how we intend to reach out to maximize the project's visibility. Section 5.3 clarifies our position with respect to intellectual property of the project's achievements.

5.1 Dissemination of the results

Communication on the project will be made at local, regional, national and international level. Results generated by the project will be published in renowned international scientific journals in order to rapidly reach other experts in the field. Researchers of the team will be encouraged to participate in international conferences, such as IJCAI, ECAI, VLDB, ISWC, WWW, ICDE, and top-quality technical journals such as SWJ, ACM-TOIT, ACM-TIST, ACM-TWEB, ACM-JEA, ACM-TACO, ACM-TKDD, ACM-TODS, ACM-TOIS, and similar IEEE Computer Society's journals (AI, TKDE), and in order to present their work and share ideas and experience with the scientific community. However, these highly

Task	Months	PC	EC	LTM	ETM	Int_1	Int_2	Int_3	Int_4	Int_5
T0.1	1	5	3							
T0.2	23	1	1	1	1					
T0	**24**	**1**	**1**	**1**	**1**					
T1.1	8	3		4		5				
T1.2	9	3		4			5			
T1	**17**	**3**		**4**		**5**	**5**			
T2.1	8	1	2		4			5		
T2.2	9	1	2		4				5	
T2	**17**	**1**	**2**		**4**			**5**	**5**	
T3	**6**	**4**	**2**	**4**	**4**					**5**

Table 7: Summary of *CEDAR* Project Staff Expected d/w Timetable per Task

competitive archival publication venues only constitute our ultimate goal—especially for such a short term as 24 months. All members and teams of the project will be expected to report preliminary results in the form of citable technical reports in the host lab of the Université Lyon 1, and submit such results to be published and shared in national or international workshops on advances in relevant topics. All technical work worth a publication will be submitted at appropriate scientific events or media.

In all technical reports, publications, presentations, demos, and software, the project's ANR support will be explicitly mentioned and acknowledged. All dissemination actions such as the aforementioned technical deliverables, as well as those concerning communication media (website, interviews, invitations) will give all due credit to the ANR as sponsor.

In addition to the frequent and periodic participants meetings, a collaborative internet working platform will be developed and will represent the core instrument for networking activities and communication on the project. This platform will be supervised by the Project Chair and under the technical management of the most senior team member (preferably a Postdoc with network-system management experience). It will be accessible in a password-protected mode and secure network channels to all project participants. The following functionalities will be made available: summary of research subject, major achievements and updated results, content of presentations and meeting's minutes, program information on events with contact and useful links, as well as all other relevant useful and promotional information. This information, relevant links, and promotional announcements will be made accessible through the public web page of the project. Care will be taken to present its contents both in scientific and in non-initiate language to spread and disseminate knowledge to a wider community.

Finally, toward the end of the 24 month period of the project (after its 18th month), we plan to organize a workshop on Scalable Distributed Ontology Management with all the *CEDAR* project participants and a few invited respected researchers from France, Europe, or North Africa, working in the same area (see Task T3.3, and also Section E). The point for such a gathering is to recapitulate the project's contribution in the context of the current work, as well as suggest further work in this vein to a wider audience and demonstrate with respect to the state of the art.

5.2 Scientific mediation and teaching

A part of our work in this project is to focus on the spread of results, research outcomes and innovation in the area of distributed knowledge based systems across the wider scientific community, industry and user communities. $\mathcal{CEDAR}$ will disseminate its results to the widest possible audience through several means:

- **Sharing events**—A summer school and training activities will constitute an important means to spread the results across the scientific and other communities. $\mathcal{CEDAR}$ will organize one training workshop of an international and interdisciplinary character that will help/contribute forge a new research and scientific community on constraint event-driven automated reasoning and a summer school at the end of the project.
- **The $\mathcal{CEDAR}$ web portal**—The $\mathcal{CEDAR}$ web portal will be one of the key instruments for external dissemination and communication to spread project knowledge to wider community. It will provide an efficient and controlled mechanism to distribute electronically the information and to communicate over the Web with interested persons or institutions. The portal will give access to material for the technical issues that can be used to train graduate students and young researchers and professionals.
- **Collaboration with other projects**—Our aim is also to establish collaborations with other projects, apply each other's (intermediate) research results, and increase awareness of novel developments beyond projects.

5.3 Intellectual property

Software, and documentation thereon, will be made available (*i.e.*, downloadable and/or remotely usable from freely accessible servers), though not necessarily as (fully) open source. The reason for reserving fully free access to some parts of the code is essentially motivated by the Project Chair's wish to ensure consistency in the evolution and maintenance of the software.

Legal protection mechanisms should be investigated every time exploitable results have been achieved. A careful attention will be paid to any potential patent that could emerge from the project. The interest of all researchers involved in the project will be considered when protection and exploitation of results will be concerned. Any Intellectual Property issue will be discussed within the project meetings. Communication among participants (and external stakeholders if any) and good faith will be the key for successful and efficient handling of Intellectual Property Rights matters. In any case, French regulation and guidelines will be followed, and the expertise of the relevant structures will be sought.

6 Conclusion

In the document we have overviewed the $\mathcal{CEDAR}$ project. We have explicated its goals, justified its motivation, placed it in context, and presented a work plan, and detailed tasks toward reaching its stated objectives. These are to experiment processing and managing distributed knowledge using oridersorted graph constraints as an alternative approach to mainstream proposals relying on Tableaux-based Description Logic such as $\mathcal{OWL}$ and its varions declensions. The main challenge the $\mathcal{CEDAR}$ project addresses is coping effectively with distributed massive KBs in a scalable manner.

Following is an appendix overviewing complementary information and background relevant to the work plan we have presented.

Appendix

This appendix adds background information regarding the $\mathcal{CEDAR}$ Project Chair in Section A. Section B recalls the stature and achievements of the project's hosting institution, making it a great venue for such a project. The description of prerequisite and expected duties of the project's staffing is given in Section C.1. Section C.1.1 describes the job profile for the Language Track Manager, and Section C.1.2 the profile for the Experimental Track Manager. Section C.2 gives the broad lines for MS-level research topics offered by work on $\mathcal{CEDAR}$ for interns. Section D describes expected collaborations—Section D.1 presents the set of academic scientists in France who have expressed interest in being involved as $\mathcal{CEDAR}$ project collaborators, and Section D.2 summarizes collaborative commitment by the CERIST research center in Algeria. Following that, Section E describes an end-of-project research workshop to be organized as an effective means for sharing the project's contributed highlights with external colleagues and peers leading the field. Finally, Section F describes and justifies the logo identifying the $\mathcal{CEDAR}$ project. This document ends with the complete bibliography of all cited references.

A The $\mathcal{CEDAR}$ Project Chair

This section presents the $\mathcal{CEDAR}$ Project Chair. Section A.1 gives a succint summary of his research biography, in French and English. Section A.2 gives more details of his scientific contributions. Section A.3 lists his technical publications.

A.1 Short biography

Hassan Aït-Kaci a un doctorat en informatique de l'université de Pennsylvanie (1984), et une habilitation à diriger des recherches de l'Université de Paris 7 (1990). Jusqu'au au terme de l'année 2012, il a occupé le poste de membre sénior du personnel technique d'IBM Canada. Les aires d'intérêt de Dr Aït-Kaci sont le raisonnement automatisé, la représentation des connaissances, la programmation déclarative, et la linguistique informatique.?

Hassan Aït-Kaci holds a PhD in Computer Science from the University of Pennsylvania (1984), and a Research Director Habilitation from University of Paris 7 (1990). Until the end of year 2012, he held the position of Senior Member of Technical Staff at IBM Canada. Dr. Aït-Kaci's interests are in automated reasoning, knowledge representation, declarative programming, and language processing.

A.2 Detailed introduction

At IBM Canada Ltd., Hassan Aït-Kaci held the position of Senior Member of Technical Staff, Level "10" (*i.e.*, top of IBM's scale, under "Distinguished Engineer," and "IBM Fellow"). He became so after IBM's acquisition of ILOG in February 2009, the French INRIA spin-off that made its business success and fame in the technology of constraint-processing and Business Rules (BR). Dr. Aït-Kaci's interests and contributions have been in automated reasoning, knowledge representation, declarative programming, and language processing. In these areas, he has been a fervent advocate of constraint-based computing as the versatile key to essential locks that all these subjects have in common, and that we are facing in the pursuit of making the Semantic Web an intelligent reality. He had joined ILOG in 2000, as a Distinguished Scientist, originally on leave from Simon Fraser University (SFU), where he was senior NSERC Research Chair in Intelligent Systems, a tenured full professor in the SFU School of Computing Science since 1994. Before that, he was Project Leader at Digital Equipment's Paris Research Lab, where he led the Paradise project developing the $\mathcal{OSF}$-constraint programming language LIFE.[13] Before joining Digital in 1989, he was a member of technical staff at the Microelectronics and Computer technology Corporation (MCC), in Austin, TX (USA), in Bob Boyer's Intelligent Architecture group, part of MCC's AI Program headed by the late Woody Bledsoe. There, he headed a research team that designed and prototyped the original version of LIFE.

A.3 List of publications

This section contains a chronologically sorted list of the major scientific publications of $\mathcal{CEDAR}$ Chair, Hassan Aït-Kaci—49 in total. It is sorted by venue: journals, meetings, books, collaborative projects. Each sort's entry is listed starting with most recent first. Most of these publications are available on the Internet, or by request to the author; all others from their publishers.

The publications that had significant impact are thus highlighted. By "significant impact," we mean "ground-breaking or highly cited in the field."

- Refereed journals (15)

1. "Children's Magic Won't Deliver the Semantic Web," *Communications of the ACM*, vol. 52, no. 3, pp. 8–9, March 2009.
2. "Data models as constraint systems: A key to the semantic web," *Constraint Programming Letters*, vol. 1, pp. 33–88, November 2007.
3. "Order-Sorted Feature Theory Unification," (with Andreas Podelski and Seth Copen Goldstein), *Journal of Logic Programming*, 30(2), pp. 99–124, February 1997.
4. "Label-Selective lambda-Calculus—Syntax and Confluence," (with Jacques Garrigue), *Theoretical Computer Science* 151, pp. 353–383, 1995.
5. "Functions as Passive Constraints in LIFE," (with Andreas Podelski), *ACM Transactions on Programming Languages and Systems*, 16(4), pp. 1279–1318, July 1994.
6. "A Feature Constraint System for Logic Programming with Entailment," (with Andreas Podelski and Gert Smolka), *Theoretical Computer Science*, 122, pp. 263–283, 1994.
7. "Towards a Meaning of LIFE," (with Andreas Podelski), *Journal of Logic Programming*, 16(3-4), pp. 195–234,1993.
8. "LIFE—a Natural Language for Natural Language," (with Patrick Lincoln), *T.A. Informations*, 30(1-2), Association pour le Traitement Automatique des Langues, Paris, France, pp. 37–67, 1989.
9. "Implementing a Knowledge-Based Library Information System with Typed Horn Logic," (with Roger Nasr and Jungyun Seo), *Information Processing & Management*, 26(2), pp. 249–268, 1990.
10. "Integrating Logic and Functional Programming," (with Roger Nasr), *Journal of Lisp and Symbolic Computation*, 2, pp. 51–89, 1989.
11. "Inheritance Hierarchies: Semantics and Unification," (with Gert Smolka), *Journal of Symbolic Computation*, 7, pp. 343–370, 1989.

[13] LIFE is an acronym that stands for "**L**ogic, **I**nheritance, **F**unctions, and **E**quations."

12. "Efficient Implementation of Lattice Operations," (with Robert Boyer, Patrick Lincoln and Roger Nasr) *ACM Transactions on Programming Languages and Systems*, 11(1), pp. 115–146, January 1989.
13. "An Algebraic-Semantics Approach to the Effective Resolution of Type Equations," *Theoretical Computer Science*, 45, pp. 293–351, 1986.
14. "LOGIN: A Logic-Programming Language with Built-In Inheritance," (with Roger Nasr), *Journal of Logic Programming* 3, pp. 185–215,1986.
15. "An Algorithm for Finding a Minimal Recursive Path Ordering," *Revue d'Automatique, d'Informatique, et de Recherche Opérationnelle—Informatique théorique*, 19(4), pp. 359–382, 1985.

- Refereed conferences (24)

1. "LIFE Su Doku," *Proceedings of the 2nd Tunisia–Japan Workshop on Symbolic Computation in Software Science (SCSS 2009)*, Gammarth, Tunisia, September 2009.
2. "Constraint-based data models for the Semantic Web," *Proceedings of the 2nd International Workshop on Applications of Logic Programming to the Web, Semantic Web and Semantic Web Services*, Porto, Portugal, September 2007.
3. "Description Logic vs. Order-Sorted Feature Logic," *Proceedings of the 20th Workshop on Description Logics*, Brixen-Bressanone, Italy, June 2007.
4. "Satisfiability Modulo Structures as Constraint Satisfaction: an Introduction," (with Bruno Berstel, Ulrich Junker, Michel Leconte, and Andreas Podelski), *Journées Francophones sur les Langages Applicatifs, Aix-les-Bains*, France, January 2007, pp. 2–8.
5. "An Axiomatic Approach to Feature Term Generalization," (with Yutaka Sasaki), *Proceedings of the European Conference on Machine Learning*, Freiburg, Germany, September 2001.
6. "An Object-Oriented Constraint-Logic Programming Implementation of a Toolkit for Graphical User Interfaces," (with Bruno Dumant), *Proceedings of the Euro-Graphics Workshop on Programming Paradigms in Graphics*, Maastricht, The Netherlands, March 1995.
7. "A Database Interface for Complex Objects," (with Marcel Holsheimer and Rolf de By), *Proceedings of the 11th International Conference on Logic Programming*, (Genoa, Italy), June 13–17, 1994.
8. "The Typed Polymorphic Label-Selective Lambda-Calculus," (with Jacques Garrigue), *Proceedings of the 21st Annual ACM Symposium on Principles of Programming Languages*, Portland, Oregon, pp. 35–47. January, 1994.
9. "Label-Selective Lambda-Calculus: Syntax and Confluence (short version)," (with Jacques Garrigue) *Proceedings of the 13th International Conference on Foundations of Software Technology and Theoretical Computer Science*, Bombay, India. Lecture Notes in Computer Science 761, December 1993.
10. "An Introduction to LIFE—Programming with Logic, Inheritance, Functions, and Equations," *Proceedings of the 10th International Logic Programming Symposium*, Vancouver, BC, Canada, pp. 1–17, October 1993.
11. "Order-Sorted Feature Theory Unification (short version)," (with Andreas Podelski and Seth Copen Goldstein), *Proceedings of the 10th International Logic Programming Symposium*, Vancouver, BC, Canada, October 1993, pp. 506–524.
12. "Entailment and Disentailment of Order-Sorted Feature Constraints," (with Andreas Podelski), *Proceedings of the 4th International Conference on Logic Programming and Automated Reasoning*, Saint Petersburg, Russia, Andrei Voronkov, ed., Lecture Notes in A.I. 698, Springer-Verlag, pp. 1–18, 1993.
13. "Logic Programming with Functions over Order-Sorted Feature Terms," (with Andreas Podelski) *Proceedings of the 3rd International Workshop on Extensions of Logic Programming*, Bologna, Italy, E. Lamma and P. Mello, eds., Springer-Verlag, LNAI 660, pp. 100–119, 1992.

14. "A Feature-Based Constraint System for Logic Programming with Entailment (short version)," (with Andreas Podelski and Gert Smolka) *Proceedings of the 5th International Conference on Fifth Generation Computer Systems*, Tokyo, Japan, pp. 1012–1022, 1992.
15. "Towards a Meaning of LIFE (short version)," (with Andreas Podelski) *Proceedings of the 3rd International Symposium on Programming Language Implementation and Logic Programming*d, Passau, Germany, Jan Maluszynski and Martin Wirsing, eds., Springer-Verlag, LNCS 528, pp. 255–274, 1991.
16. "An Overview of LIFE," *Next Generation Information System Technology: Proceedings of the 1st International East/West Data Base Workshop*, Kiev, USSR, October, 1990, J.W. Schmidt and A.A. Stogny, eds.. Springer-Verlag, LNCS 504, pp. 42–58, 1991.
17. "A Glimpse of Paradise," *Next Generation Information System Technology: Proceedings of the 1st International East/West Data Base Workshop*, Kiev, USSR, October, 1990, J.W. Schmidt and A.A. Stogny, eds., Springer-Verlag, LNCS 504, pp. 17–25, 1991.
18. "BABEL: A Base for an Experimental Library," (with Roger Nasr and Jungyun Seo), *Proceedings of the ACM SIGIR International Conference on Information Retrieval*, Grenoble, France, June 1988.
19. "Integrating Data Type Inheritance into Logic Programming," (with Roger Nasr) *Data Types and Persistence*, Malcolm Atkinson, Peter Buneman, and Ron Morrison, eds., Springer-Verlag, pp. 121–136, 1988.
20. "Le Fun: Logic, equations, and Functions," (with Patrick Lincoln and Roger Nasr) *Proceedings of the Symposium on Logic Programming*, San Francisco, CA, pp. 17–23. September 1987.
21. "Logic Programming and Inheritance," (with Roger Nasr), *Proceedings of the 13th ACM Symposium on Principles of Programming Languages*, Saint-Petersburg, FL, pp. 219–228, January 1986.
22. "Type Subsumption as a Model of Computation," *Expert Database Systems*, L. Kerschberg, ed., Benjamin Cummings Publishing Co., pp. 115–139, 1986.
23. "Object-Oriented Database and Knowledge Systems," (with David Beech, Stephanie Cammarata, Larry Kerschberg, David Maier, and Carlo Zaniolo) *Expert Database Systems*, L. Kerschberg, ed., Benjamin Cummings Publishing Co., pp. 115–139. 1986.
24. "Solving Type Equations by Graph-Rewriting," Proceedings of First International Conference on Rewriting Techniques and Applications, Dijon, France, May 1985. J.P. Jouannaud, ed.. Springer-Verlag, LNCS 202, pp. 158–179, 1985.

- Books (3)

1. *Warren's Abstract Machine: A Tutorial Reconstruction*, MIT Press, Series in Logic Programming, 1991.
2. *Resolution of Equations in Algebraic Structures–Vol. 1: Algebraic Techniques*, (co-edited with Maurice Nivat), Academic Press, Boston, 1989.
3. *Resolution of Equations in Algebraic Structures–Vol. 2: Rewriting Techniques*, (co-edited with Maurice Nivat), Academic Press, Boston, 1989.

- Other scientific venues (3)

1. "Complexity and Optimization of Combinations of Rules and Ontologies," *M24 Deliverable D3.3, Technical Report, ONTORULE Project*, coordinated by Cristina Feier, with contributions from Hassan Aït-Kaci, Jürgen Angele, Jos de Bruijn, Hugues Citeau, Thomas Eiter, Adil El Ghali, Volha Kerhet, Eva Kiss, Roman Korf, Thomas Krekeler, Thomas Krennwallner, Stijn Heymans, Alessandro Mosca, Martn Rezk, Guohui Xiao (2010).
2. "Initial Combinations of Rules and Ontologies," *M12 Deliverable D3.2, Technical Report, ONTORULE Project*, coordinated by Stijn Heymans, with contributions from Jos de Bruijn, Martn Rezk, Hassan Aït-Kaci, Hugues Citeau, Roman Korf, Jörg Pührer, Cristina Feier, and Thomas Eiter (2009).

3. "Processing of initial combinations of rules and ontologies," *M12 Deliverable D3.5, Technical Report ONTORULE IST-2009-231875 Project*, coordinated by Hassan Aït-Kaci, with contributions from Hugues Citeau and Roman Korf (2009).

B Introduction of the Host Laboratory

The *CEDAR* Project's host laboratory is the LIRIS.[14] This lab is affiliated to the CNRS6 under the label UMR 5205. The laboratory involves 280 people with 94 faculty members and researchers from Université Claude Bernard Lyon 1, INSA de Lyon, Ecole Centrale de Lyon, Université Lumière Lyon 2 and CNRS. The laboratory is organized into two departments:

- **Data, Knowledge and Services**—The DCS department is organized around six research teams involving 44 faculty members (11 professors, 33 associate professors). The research activities of the department cover a wide variety of theories, methods, and applications of information technology to the management of data, knowledge and services. It covers the following areas of Knowledge management and discovery (data mining, complex systems modeling, knowledge engineering) and data and services engineering (security and confidentiality, modeling, integration and querying, service composition)
- **Image Processing**—This department counts 43 permanent faculty members (12 professors, 28 associate professors) and 3 permanent researchers. The department incorporates five research teams. The research activities of the department cover a wide variety of methods for sensors (2D, 2D+t, 3D) data analysis, for a better understanding and for multidimensional data modeling.

Evaluated in 2009, the laboratory has been rated "A" (top mark) by the AERES.[15]

C Staff Profiles

C.1 Track managers

C.1.1 Language track manager profile

The Language Track Manager (LTM) will hold a recent PhD in Computer Science, and have experience participating in research projects in the area of AI language design and implementation (including compilation of reasoning algorithms for efficient execution), knowledge representation, Semantic Web reasoning, and be proficient with software programming in Java, and have some experience with other programming language—especially LP, FP, and DB programming. A strong working experience in software development and maintenance is required. Experience with the Eclipse development platform will be a useful asset. Working familiarity with formal methods will be a plus. This person will be in charge of managing the daily technical operation of the Language Track set of tasks, software development and documentation, and will participate in the preparation of technical reports and publications (knowledge of research publication tools such as Emacs, LaTeX, *etc.*, is needed). The LTM must also be capable of setting up and maintaining a secure public web site. The LTM will work under the direct management

[14] **L**aboratoire d'**I**nfo**R**matique en **I**mage et **S**ystèmes d'information (`http://liris.cnrs.fr`)
[15] `http://www.aeres-evaluation.fr/`

of the Project Chair and be responsible for the seamless integration and maintenance of language-level implementation and its documentation. The LTM will also interact as needed with the Experiment Coordinator and the Experiment Task Manager, and all other concerned project participants.

Keywords: *Declarative Programming Software Specification, Compiler Technology, Implementation, Documentation, and Maintenance.*

C.1.2 Experimental track manager profile

The Experimental Track Manager (ETM) will hold a recent PhD in Computer Science, and have experience participating in advanced architecture design testing and simulation, and have first-hand experience working with distributed networked DB or KB systems. High proficiency is expected at least in Java programming with working knowledge of DB and networking libraries and packages. Experience with the Eclipse development platform will be a useful asset. A strong experience in applied CS is required. Some understanding of formal methods will be a plus. The ETM will be in charge of managing the daily technical operation of the Experiment Track set of tasks, software development and documentation, and will participate in the preparation of technical reports and publications (knowledge of research publication tools such as Emacs, LaTeX, *etc.*, is needed). The ETM must also be capable of setting up and maintaining a secure public web site. The ETM will work under the direct management of the Project Chair, but mostly under the technical lead of the Experiment Coordinator. The ETM will be responsible for the architecture-level design, implementation, maintenance, and documentation of the experimental part of the project. The ETM will also interact as needed with the Project Chair and the Language Task Manager, and all other concerned project participants.

Keywords: *Networking, Distributed Systems, Simulation, Distributed DB or KB Management, Software Specification, Implementation, Documentation, and Maintenance.*

C.2 Masters students projects

The $\mathcal{CEDAR}$ project's tasks provide research topics for five MS-level interns: one per each of the project's technical tasks T1.1, T1.2, T2.1, T2.2, and T3 as follows:

- **Task T1.1—Structural constraint-graph language**: Designing and prototyping in Java a basic KB representation and query language based on $\mathcal{OSF}$-graph constraints using RDF-triples (using $\mathcal{OSF}$ unification as constraint system).
- **Task T12.—Temporal constraint-graph language**: Extending Task T1.1's design and prototype using temporal interpretation of $\mathcal{OSF}$-graphs constraints in a dynamic evolving time-aware context (using $\mathcal{OSF}$-graph generalization as a constraint system).
- **Task T2.1—Basic experimental structural architecture KB simulation**: Designing and implementing experiments for testing massive amounts of KBs of RDF-triples by simulation of accessing large amounts of generated such structures in a distributed environment corresponding to RDF-represented $\mathcal{OSF}$-graph KBs (by building a translator from DL-based KBs to $\mathcal{OSF}$-based KBs and using $\mathcal{OSF}$-specific optimizations).
- **Task T2.2—Experimental temporal architecture KB simulation**: Extending T2.1's basic simulation generation setup with evolving and distributed KBs by event-controlled $\mathcal{OSF}$-graph constraint solving (using Task T2.1's set up for generation of test KBs and $\mathcal{OSF}$-specific operations).

- **Task T3—Synthesis of language and experimental tacks**: Bringing the results of Task T1.2 to be tested on the productions of simulated RDF-represented $\mathcal{OSF}$-graph KBs yielded by Task T2.2, and measure performances both in the absence and presence of event-processing and adapt language design according to test results on a real use case.

D Scientific Collaboration

D.1 French academia

- **Prof. Mohand-Saïd Hacid**, PhD—Full Professor of Computer Science at the Université Claude Bernard Lyon 1, France. He is the deputy chair of LIRIS CNRS UMR 5205 and leader of the department "Data, Knowledge and Services." He has been Director of the Lyon High Education Center since 2006. He has been involved in several national and international projects among which: S-cube,[16] Tarchna, COMPAS, SemWeb,[17] FORUM,[18] AOC.[19] He served as a general co-chair of ICSOC 2009, ICSOC 2011, and IEEE/WIC/ACM WI-IAT 2011, and as an organizing co-chair of VLDB 2009. He has been involved in over 40 scientific program committees to date, most of which major international events such as: ICSOC, BPM, EDBT, DS, CAISE, ICWS, *etc.*, … His research interests include data management, semantic web and web services, data security (partial list of publications).

- **Prof. Hamamache Kheddouci**, PhD—Full Professor, Department of Computer Sciences, Université Claude Bernard Lyon 1, France. He is leading the GAMA research group. His research interests include: Graph Algorithms, Graph Mining, Graph Matching, Data Structures, Knowledge Management, Peer-to-Peer Computing, Mobile Ad-hoc Networks, Mobile Sensor Networks, Social Networks, Semantic Web and Services, and Self-Stabilizing and Self-Organizing Complex Systems.

- **Dr. Mohammed Haddad**, PhD—Associate Professor of Computer Science, Polytech Lyon since 2010. His research interests include edit distance computation problem for labeled graphs (modeling structured documents) and graph decomposition methods.

- **Dr. Fabien Duchateau**, PhD—Associate Professor at the Université Claude Bernard Lyon 1 since 2011. In 2010, he obtained a 18-months postdoctoral fellowship sponsored by ERCIM. He spent the first part of his fellowship at CWI, The Netherlands. Then, he joined the team of Trond Aalberg at NTNU, Norway. His research interests include databases, information systems, data integration, schema and ontology matching, semantic web, digital libraries and machine learning.

- **Dr. Nicolas Lumineau**, PhD—Associate Professor of Computer Science at the Université Claude Bernard Lyon 1. His research topics include dynamic environment modeling, query optimization and data integration.

[16] European Union's project; 2008–2012.
[17] ANR, 2004–2007
[18] ANR, 2006–2009
[19] ANR

- **Dr. Yacine Sam**, PhD—Associate Professor at the Université François Rabelais, Tours, France. He received his PhD degree in Computer Science from Université Paul Cézanne Aix-Marseille 3 (France) in 2008. His research interests include Knowledge Representation and Reasoning, the Semantic Web, as well as Data and Application Integration.

D.2 CERIST

Part of the CEDAR project will benefit from techcical collaboration with the *Centre de Recherche sur l'Information Scientifique et Technique* (CERIST), based in Algiers, Algeria.[20]

> *"The main specific objectives of this collaboration are the following:*
>
> - *Build a joint bilateral research team for this project.*
> - *Design a joint comprehensive framework whereby the CERIST will provide participating doctoral and post-doctoral scientists and engineers with opportunities to contribute to the development and implementation of a constraint-based approach to Knowledge Representation and Automated Reasoning.*
>
> *The proposed project will receive the CERIST's continuous support through our research staff for its duration. We are very pleased to work swith Dr. Hassan Aït-Kaci as well as all appropriate parties at the Université Claude Bernard Lyon 1 to ensure seamless assistance to researchers participating in the proposed project."*

E The CEDAR Workshop

Toward the end of the project (from its 18th to its 24th month) it is planned to organize and host a CEDAR workshop on Constraint Event-Driven Automated Reasoning, locally in Lyon. The technical motivation for such a small-scale event if to contribute to disseminating our work by sharing results with known experts in the field, as well as inseminate such leaders in the field with our own results and perspectives. It is to be a one-day event held locally in Lyon, a gathering of 20 to 30 people, counting about three invited research scientists of international stature.

F The CEDAR Project Logo

This document's front page and the running heads on all pages display the logo of the CEDAR Project. This logo evokes both the use of partial-order reasoning techniques and the distribution of ontological knowledge sources over the Internet. It also illustrates in a graphical way the objective of the project as a flexible and responsive architecture meant to be capable to adapt to scale and provide structure in a highly inter-related set of network devices interacting in real time. Lastly, it also reminds one of the shapes of cedar trees—from roots, through trunks, to leaves.

[20]What is quoted below is paraphrased from the contents of the *official support letter for the CEDAR project written by Dr. Nadjib Badache, CERIST Director.*

References

[1] ADIDA, B., AND BIRBECK, M. RDFa Primer—Bridging the human and data webs. World Wide Web Consortium Recommendation, October 14, 2008. [Available online[21]].

[2] AÏT-KACI, H. An introduction to LIFE—Programming with Logic, Inheritance, Functions, and Equations. In *Proceedings of the 10th International Symposium on Logic Programming* (Vancouver, BC, Canada, October 26–29, 1993), D. Miller, Ed., Association for Logic Programming, MIT Press, pp. 52–68. [Available online[22]].

[3] AÏT-KACI, H. Data models as constraint systems—A key to the Semantic Web. *Constraint Processsing Letters 1*, 1 (November 2007), 33–88. [Available online[23]].

[4] AÏT-KACI, H., BOYER, R., LINCOLN, P., AND NASR, R. Efficient implementation of lattice operations. *ACM Transactions on Programming Languages and Systems 11*, 1 (January 1989), 115–146. [Available online[24]].

[5] AÏT-KACI, H., AND COSMO, R. D. Compiling order-sorted feature term unification. PRL Technical Note 7, Digital Paris Research Lab, Rueil-Malmaison, France, December 1993. [Available online[25]].

[6] AÏT-KACI, H., AND LINCOLN, P. LIFE—A natural language for natural language. *T.A. Informations 30*, 1–2 (1989), 37–67. [Available online[26]].

[7] AÏT-KACI, H., AND SASAKI, Y. An axiomatic approach to feature term generalization. In *Proceedings of the 12th European Conference on Machine Learning—ECML 2001* (Freiburg, Germany, September 6–7, 2001), Springer-Verlag, pp. 1–12. [Available online[27]].

[8] AÏT-KACI, H., AND VORBECK, M. Compiling order-sorted feature unification. Intelligent software group technical report, Intelligent Software Laboratory, School of Computing Science, Simon Fraser University, Burnaby, BC, Canada, February 1996. [Available online[28]].

[9] ALPUENTE, M., ESCOBAR, S., ESPERTA, J., AND MESEGUER, J. A modular order-sorted equational generalization algorithm. *Information and Computation* (2011). [Available online[29]].

[10] BISTARELLI, S., MONTANARI, U., AND ROSSI, F. Semiring-based constraint solving and optimization. *Journal of the ACM 44*, 2 (March 1997), 201–236. [Available online[30]].

[11] BIZER, C., HEATH, T., AND BERNERS-LEE, T. Linked data—The story so far. In *International Journal on Semantic Web and Information Systems—IJSWIS*, T. Heath and C. Bizer, Eds. Special Issue on Linked Data, 2009. [Available online[31]][32].

[12] BOUCELMA, O., HACID, M.-S., LIBOUREL, T., AND PETIT, J.-M. Modèles et langages pour les bases de données. *Revue des sciences et technologies de l'information—Ingénierie des Systèmes d'Information 13*, 5 (2008), 7–8. [Available online[33]].

[21] http://www.w3.org/TR/xhtml-rdfa-primer/
[22] http://www.hassan-ait-kaci.net/pdf/life.pdf
[23] http://www.cs.brown.edu/people/pvh/CPL/Papers/v1/hak.pdf
[24] http://www.hassan-ait-kaci.net/pdf/encoding-toplas-89.pdf
[25] http://www.hassan-ait-kaci.net/pdf/PRL-TN-7.pdf
[26] http://citeseer.ist.psu.edu/viewdoc/summary?doi=10.1.1.51.2021
[27] http://www.springerlink.com/content/xlha6x9kewvg4ccg/
[28] http://www.hassan-ait-kaci.net/pdf/osfucomp.pdf
[29] https://www.ideals.illinois.edu/bitstream/handle/2142/25871/journal.pdf
[30] ftp://ftp.di.unipi.it/pub/Papers/rossi/jacm.ps.gz
[31] http://tomheath.com/papers/bizer-heath-berners-lee-ijswis-linked-data.pdf
[32] http://www.w3.org/DesignIssues/LinkedData.html
[33] http://isi.revuesonline.com/gratuit/ISI13_5_03_edito.pdf

[13] BRINKLEY, D., AND GUHA, R. V. RDF vocabulary description language 1.0—RDF Schema. World Wide Web Consortium Recommendation, February 10, 2004. [Available online[34]].

[14] ELGHAZEL, H., AND HACID, M.-S. Aggregated search in graph databases—Preliminary results. In *Proceedings of the 8th IAPR-TC-15 International Workshop on Graph-Based Representations in Pattern Recognition—GbRPR 2011* (Münster, Germany, May 2011), Lecture Notes in Computer Science, Springer-Verlag, pp. 92–101. [Available online[35]].

[15] GUPTA, S. K. A system for interfacing LIFE with database and persistent storage. Master's thesis, Simon Fraser University, School of Computing Science, Burnaby, BC, Canada, November 1996.

[16] HOLSHEIMER, M. LIFE-WISDOM—A database interface for the LIFE system. Master's thesis, University of Twente, Computer Science Department, Enschede, The Netherlands, September 1992.

[17] HOLSHEIMER, M., DE BY, R., AND AÏT-KACI, H. A database interface for complex objects. In *Proceedings of the 11th International Conference on Logic Programming—ICLP 1994* (Genoa, Italy, June 13–17, 1994), pp. 437–455. [Available online[36]].

[18] KANEIWA, K. *An Order-Sorted Logic with Predicate-Hierarchy, Eventuality, and Implicit Negation.* PhD thesis, School of Information Science, Japan Advanced Institute of Science and Technology, Nomi, Ishikawa, Japan, March 2001. [Available online[37]].

[19] KANEIWA, K., AND NGUYEN, P. H. An order-sorted logic with predicate-hierarchy, eventuality, and implicit negation. In *Proceedings of the 8th International Semantic Web Conference—ISWC 2009* (Washington, DC, USA, October 25–29 2009), vol. 5823 of *Lecture Notes in Computer Science*, Springer-Verlag, pp. 328–343. [Available online[38]].

[20] KANEIWA, K., AND TOJO, S. Event, property and hierarchy in order-sorted logic. In *Proceedings of the International Conference on Logic Programming—ICLP 1999* (Las Cruces, NM, USA, 1999), MIT Press, pp. 94–108. [Available online[39]].

[21] KANEIWA, K., AND TOJO, S. Logical aspects of events—Quantification, sorts, composition and disjointness. In *Proceedings of Australasian Ontology Workshop—Advances in Ontologies—AOW 2005* (Sydney, Australia, December 6, 2005), T. Meyer and M. A. Orgun, Eds., ACS, pp. 33—40. [Available online[40]].

[22] KARAM, N., BENBERNOU, S., HACID, M.-S., AND SCHNEIDER, M. Semantic matching of natural language web queries. In *Proceedings of the 4th International Conference on Web Engineering—ICWE 2004* (Munich, Germany, July 26–30, 2004), N. Koch, P. Fraternali, and M. Wirsing, Eds., vol. 3140 of *Lecture Notes in Computer Science*, Springer-Verlag, pp. 416–429. [Available online[41]].

[23] LE, T.-H., ELGHAZEL, H., AND HACID, M.-S. Aggregated search in graph databases—A relational-based approach. In *Proceedings of the 17th International Conference on Database Systems for Advanced Applications—DASFAA 2012* (Busan, South Korea, April 15–18, 2012), S. Lee, Z. Peng, X. Zhou, Y.-S. Moon, R. Unland, and J. Yoo, Eds., vol. 7238 of *Lecture Notes in Computer Science*, Springer-Verlag, pp. 33–47.

[24] MILES, A., AND BECHHOFER, S. SKOS—Simple knowledge organization system reference. World Wide Web Consortium Recommendation, August 18, 2009. [Available online[42]].

[34] http://www.w3.org/TR/rdf-schema/
[35] http://liris.cnrs.fr/publis/?id=5371
[36] http://www.hassan-ait-kaci.net/pdf/iclp94.pdf
[37] http://www2.nict.go.jp/x/x163/kaneiwa/PhDthesis.pdf
[38] http://www.ai.cis.iwate-u.ac.jp/kaneiwa/kk-iswc2009.pdf
[39] http://www.ai.cis.iwate-u.ac.jp/kaneiwa/kk-iclp99.pdf
[40] http://www.ai.cis.iwate-u.ac.jp/kaneiwa/kaneiwa-ea.pdf
[41] http://www.springerlink.com/content/971n5x1kb087a1wl/
[42] http://www.w3.org/TR/skos-reference/

[25] MORGENSTERN, L., WELTY, C., AND BOLEY, H. RIF primer. World Wide Web Consortium Recommendation, August 18, 2010. [Available online[43]].

[26] NIANG, C., BOUCHOU, B., LO, M., AND SAM, Y. Appropriate global ontology construction—A domain-reference-ontology based approach. In *Proceedings of the 13th International Conference on Information Integration and Web-based Applications and Services—iiWAS 2011* (2011), pp. 166–173. [Available online[44]].

[27] NIANG, C., BOUCHOU, B., LO, M., AND SAM, Y. On automating global-schema construction in data integration systems. Rapport Interne 304, Université François Rabelais, École Polytechnique de l'Université de Tours, Département Informatique, Tours, France, Mars 2012. [Available online[45]].

[28] ROGOVA, E., CHOUNTAS, P., AND ATANASSOV, K. Imprecise data and knowledge based OLAP. In *Proceedings of the 11th East-European Conference on Advances in Databases and Information Systems—ADBIS 2007* (Varna, Bulgaria, September 29–October 3 2007). [Available online[46]].

[29] SAM, Y., BOUCELMA, O., AND COLONNA, F.-M. Feature logic for web resources customization—Design and implementation. *KES Journal 15*, 4 (2011), 189–201.

[30] SCHMIDT, M., MEIER, M., AND LAUSEN, G. Foundations of SPARQL query optimization. In *Proceedings of the 13th International Conference on Database Theory—ICDT 2010* (Lausanne, Switzerland, March 22–26, 2010). [Available online[47]].

[31] SMOLKA, G. The Oz programming model. In *Computer Science Today* (Berlin, Germany, 1995), vol. 1000 of *Lecture Notes in Computer Science*, Springer-Verlag, pp. 324–343. [Available online[48]].

[32] VERHEIJEN, W. Efficient query processing in distributed RDF databases. Master's thesis, Eindhoven University of Technology, Department of Mathematics and Computer Science, Eindhoven, The Netherlands, February 2008. [Available online[49]].

[33] W3C'S RDF WORKING GROUP. Resource Description Framework (RDF). World Wide Web Consortium Recommendation, February 10, 2004. [Available online[50]].

[43] http://www.w3.org/2005/rules/wiki/Primer
[44] http://www.info.univ-tours.fr/~bouchou/Publis/nbls-iiWAS11.pdf
[45] http://www.info.univ-tours.fr/~bouchou/Publis/rapport304.pdf
[46] http://www.adbis.org/docs/lp/5.pdf
[47] http://www.edbt.org/Proceedings/2010-Lausanne/icdt/papers/N112E6.html
[48] http://www.ps.uni-saarland.de/Papers/abstracts/volume1000.html
[49] http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.143.3141
[50] http://www.w3.org/RDF/

