

CEPIS UPGRADE is the European Journal for the Informatics Professional, published bi-monthly at <<http://cepis.org/upgrade>>

Publisher

CEPIS UPGRADE is published by CEPIS (Council of European Professional Informatics Societies, <<http://www.cepis.org/>>), in cooperation with the Spanish CEPIS society ATI (*Asociación de Técnicos de Informática*, <<http://www.ati.es/>>) and its journal *Novática*

CEPIS UPGRADE monographs are published jointly with *Novática*, that publishes them in Spanish (full version printed; summary, abstracts and some articles online)

CEPIS UPGRADE was created in October 2000 by CEPIS and was first published by *Novática* and INFORMATIK/INFORMATIQUE, bimonthly journal of SVI/FSI (Swiss Federation of Professional Informatics Societies)

CEPIS UPGRADE is the anchor point for UPENET (UPGRADE European NETwork), the network of CEPIS member societies' publications, that currently includes the following ones:

- *inforeview*, magazine from the Serbian CEPIS society JISA
- *Informatica*, journal from the Slovenian CEPIS society SDI
- *Informatik-Spektrum*, journal published by Springer Verlag on behalf of the CEPIS societies GI, Germany, and SI, Switzerland
- *ITNOW*, magazine published by Oxford University Press on behalf of the British CEPIS society BCS
- *Mondo Digitale*, digital journal from the Italian CEPIS society AICA
- *Novática*, journal from the Spanish CEPIS society ATI
- *OCG Journal*, journal from the Austrian CEPIS society OCG
- *Pliroforiki*, journal from the Cyprus CEPIS society CCS
- *Tölvumál*, journal from the Icelandic CEPIS society ISIP

Editorial Team

Chief Editor: Llorenç Pagés-Casas
Deputy Chief Editor: Rafael Fernández Calvo
Associate Editor: Fiona Fanning

Editorial Board

Prof. Nello Scarabottolo, CEPIS President
Prof. Wolfried Stucky, CEPIS Former President
Prof. Vasile Baltac, CEPIS Former President
Prof. Luis Fernández-Sanz, ATI (Spain)
Llorenç Pagés-Casas, ATI (Spain)
François Louis Nicolet, SI (Switzerland)
Roberto Carniel, ALSI - Tecnoteca (Italy)

UPENET Advisory Board

Dubravka Dukic (*inforeview*, Serbia)
Matjaz Gams (*Informatica*, Slovenia)
Hermann Engesser (*Informatik-Spektrum*, Germany and Switzerland)
Brian Runciman (*ITNOW*, United Kingdom)
Franco Filippazzi (*Mondo Digitale*, Italy)
Llorenç Pagés-Casas (*Novática*, Spain)
Veith Risak (*OCG Journal*, Austria)
Panicos Masouras (*Pliroforiki*, Cyprus)
Thorvardur Kári Ólafsson (*Tölvumál*, Iceland)
Rafael Fernández Calvo (Coordination)

English Language Editors: Mike Andersson, David Cash, Arthur Cook, Tracey Darch, Laura Davies, Nick Dunn, Rodney Fennemore, Hilary Green, Roger Harris, Jim Holder, Pat Moody.

Cover page designed by Concha Arias-Pérez

"Liberty with Risk" / © ATI 2011

Layout Design: François Louis Nicolet

Composition: Jorge Liácer-Gil de Ramales

Editorial correspondence: Llorenç Pagés-Casas <pages@ati.es>

Advertising correspondence: <info@cepis.org>

Subscriptions

If you wish to subscribe to CEPIS UPGRADE please send an email to info@cepis.org with 'Subscribe to UPGRADE' as the subject of the email or follow the link 'Subscribe to UPGRADE' at <<http://www.cepis.org/upgrade>>

Copyright

© *Novática* 2011 (for the monograph)

© CEPIS 2011 (for the sections Editorial, UPENET and CEPIS News)

All rights reserved under otherwise stated. Abstracting is permitted with credit to the source. For copying, reprint, or republication permission, contact the Editorial Team

The opinions expressed by the authors are their exclusive responsibility

ISSN 1684-5285



The European Journal for the Informatics Professional
<http://cepis.org/upgrade>

Vol. XII, issue No. 5, December 2011

Farewell Edition

- 3 Editorial. CEPIS UPGRADE: A Proud Farewell
— *Nello Scarabottolo, President of CEPIS*

ATI, *Novática* and CEPIS UPGRADE
— *Dídac López-Viñas, President of ATI*

Monograph

Risk Management

(published jointly with *Novática**)

Guest Editor: *Darren Dalcher*

- 4 Presentation. Trends and Advances in Risk Management
— *Darren Dalcher*
- 10 The Use of Bayes and Causal Modelling in Decision Making, Uncertainty and Risk — *Norman Fenton and Martin Neil*
- 22 Event Chain Methodology in Project Management — *Michael Trumper and Lev Virine*
- 34 Revisiting Managing and Modelling of Project Risk Dynamics - A System Dynamics-based Framework — *Alexandre Rodrigues*
- 41 Towards a New Perspective: Balancing Risk, Safety and Danger
— *Darren Dalcher*
- 45 Managing Risk in Projects: What's New? — *David Hillson*
- 48 Our Uncertain Future — *David Cleden*
- 55 The application of the 'New Sciences' to Risk and Project Management — *David Hancock*
- 59 Communicative Project Risk Management in IT Projects
— *Karel de Bakker*
- 67 Decision-Making: A Dialogue between Project and Programme Environments — *Manon Deguire*
- 75 Decisions in an Uncertain World: Strategic Project Risk Appraisal — *Elaine Harris*
- 82 Selection of Project Alternatives while Considering Risks
— *Marta Fernández-Diego and Nolberto Munier*
- 87 Project Governance — *Ralf Müller*
- 91 Five Steps to Enterprise Risk Management — *Val Jonas* **..**

* This monograph will be also published in Spanish (full version printed; summary, abstracts, and some articles online) by *Novática*, journal of the Spanish CEPIS society ATI (*Asociación de Técnicos de Informática*) at <<http://www.ati.es/novatica/>>.



CEPIS

UPGRADE

The European Journal for the Informatics Professional
<http://cepis.org/upgrade>

Vol. XII, issue No. 5, December 2011

Farewell Edition

Cont.

UPENET (UPGRADE European NETWORK)

- 99 From **inforeview** (JISA, Serbia)
Information Society
Steve Jobs — *Dragana Stojkovic*
- 101 From **Informatica** (SDI, Slovenia)
Surveillance Systems
An Intelligent Indoor Surveillance System — *Rok Piltaver, Erik Dovgan, and Matjaz Gams*
- 111 From **Informatik Spektrum** (GI, Germany, and SI, Switzerland)
Knowledge Representation
What's New in Description Logics — *Franz Baader*
- 121 From **ITNOW** (BCS, United Kingdom)
Computer Science
The Future of Computer Science in Schools — *Brian Runciman*
- 124 From **Mondo Digitale** (AICA, Italy)
IT for Health
Neuroscience and ICT: Current and Future Scenarios
— *Gianluca Zaffiro and Fabio Babiloni*
- 135 From **Novática** (ATI, Spain)
IT for Music
Katmus: Specific Application to support Assisted Music
Transcription — *Orlando García-Feal, Silvana Gómez-Meire, and David Olivieri*
- 145 From **Pliroforiki** (CCS, Cyprus)
IT Security
Practical IT Security Education with Tele-Lab — *Christian Willems, Orestis Tringides, and Christoph Meinel*

CEPIS NEWS

- 153 Selected CEPIS News — *Fiona Fanning*

What's New in Description Logics

Franz Baader

© 2011 Informatik Spektrum

This paper was first published, in English, by *Informatik-Spektrum* (Volume 34, issue no. 5, October 2011, pp. 434-442). *Informatik-Spektrum* (<<http://www.springerlink.com/content/1432-122X/>>), a UPENET partner, is a journal published, in German or English, by Springer Verlag on behalf of the German CEPIS society **GI** (*Gesellschaft für Informatik*, <<http://www.gi-ev.de/>>) and the Swiss CEPIS society **SI** (*Schweizer Informatiker Gesellschaft - Société Suisse des Informaticiens*, <<http://www.s-i.ch/>>)

Main stream research in Description Logics (DLs) until recently concentrated on increasing the expressive power of the employed description language while keeping standard inference problems like subsumption and instance manageable in the sense that highly-optimized reasoning procedure for them behave well in practice. One of the main successes of this line of research was the adoption of OWL DL, which is based on an expressive DL, as the standard ontology language for the Semantic Web. More recently, there has been a growing interest in more light-weight DLs, and in other kinds of inference problems, mainly triggered by need in applications with large-scale ontologies. In this paper, we first review the DL research leading to the very expressive DLs with practical inference procedures underlying OWL, and then sketch the recent development of light-weight DLs and novel inference procedures.

Keywords: Description Logics, Logic-based Knowledge Representation Formalism, Ontology Languages, OWL, Practical Reasoning Tools.

1 Mainstream DL research of the last 25 years: towards very expressive DLs with practical inference procedures

Description Logics [BCNMP03] are a well-investigated family of logic-based knowledge representation formalisms, which can be used to represent the conceptual knowledge of an application domain in a structured and formally well-understood way. They are employed in various application domains, such as natural language processing, configuration, and databases, but their most notable success so far is the adoption of the DL-based language OWL¹ as standard ontology language for the Semantic Web [HoPH03].

The name *Description Logics* is motivated by the fact that, on the one hand, the important notions of the domain are described by *concept descriptions*, i.e., expressions that are built from atomic concepts (unary predicates) and atomic roles (binary predicates) using concept constructors. The expressivity of a particular DL is determined by which concept constructors are available in it. From a semantic point of view, concept names and concept descriptions represent sets of individuals, whereas roles represent binary relations between indi-

Author

Franz Baader is Full Professor for Theoretical Computer Science at TU Dresden, Germany. He has obtained his PhD in Computer Science at the University of Erlangen, Germany. He was Senior Researcher at the German Research Institute for Artificial Intelligence (DFKI) for four years, and Associate Professor at RWTH Aachen, Germany, for eight years. His main research area is Logic in Computer Science, in particular knowledge representation (description logics, modal logics, nonmonotonic logics) and automated deduction (term rewriting, unification theory, combination of decision procedures). <baader@tcs.inf.tu-dresden.de>

viduals. For example, using the concept names *Man*, *Doctor*, and *Happy* and the role names *married* and *child*, the concept of "a man that is married to a doctor, and has only happy children" can be expressed using the concept description

$$Man \sqcap \exists married. Doctor \sqcap \forall child. Happy.$$

On the other hand, DLs differ from their predecessors in that they are equipped with a formal, *logic*-based

“ In this paper we review the Description Logics research and recent developments ”

¹ <<http://www.w3.org/TR/owl-features/>>.

semantics, which can, e.g., be given by a translation into first-order predicate logic. For example, the above concept description can be translated into the following first-order formula (with one free variable x):

$$Man(x) \wedge \exists y.(married(x, y) \wedge Doctor(y)) \wedge \forall y.(child(x, y) \rightarrow Happy(y)).$$

The motivation for introducing the early predecessors of DLs, such as semantic networks and frames [Quil67, Mins81], actually was to develop means of representation that are closer to the way humans represent knowledge than a representation in formal logics, like first-order predicate logic. Minsky [Mins81] even combined his introduction of the frame idea with a general rejection of logic as an appropriate formalism for representing knowledge. However, once people tried to equip these "formalisms" with a formal semantics, it turned out that they can be seen as syntactic variants of (subclasses of) first-order predicate logic [Haye79, ScGC79]. Description Logics were developed with the intention of keeping the advantages of the logic-based approach to knowledge representation (like a formal model-theoretic semantics and well-defined inference problems), while avoiding the disadvantages of using full first-order predicate logic (e.g., by using a variable-free syntax that is easier to read, and by ensuring decidability of the important inference problems).

Concept descriptions can be used to define the terminology of the application domain, and to make statements about a specific application situation in the assertional part of the knowledge base. In its simplest form, a DL *terminology* (usually called *TBox*) can be used to introduce abbreviations for complex concept descriptions. For example, the concept *definitions*

$$Man \equiv Human \sqcap \neg Female, Woman \equiv Human \sqcap Female, Father \equiv Man \sqcap \exists child. \top$$

define the concept of a man (woman) as a human that is not female (is female), and the concept of a father as a man that has a child, where \top stands for the top concept (which is interpreted as the universe of all individuals in the application domain). The above is a (very simple) example of an *acyclic TBox*, which is a finite set of concept definitions that is unambiguous (i.e., every concept name appears at most once on the left-hand side of a definition) and acyclic (i.e., there are no cyclic dependencies between definitions). In *general TBoxes*, so-called *general concept inclusions (GCIs)* can be used to state additional constraints on the interpretation of concepts and roles. In our example, it makes sense to state domain and range restrictions for the role *child*. The GCIs

$$\exists child. Human \sqsubseteq Human \quad \text{and} \quad Human \sqsubseteq \forall child. Human$$

say that only human beings can have human children, and that the child of a human being must be human.

In the *assertional part (ABox)* of a DL knowledge base, facts about a specific application situation can be stated by introducing named individuals and relating them to concepts and roles. For example, the assertions

$$Man(JOHN), \quad child(JOHN, MACKENZIE), \quad Female(MACKENZIE),$$

state that John is a man, who has the female child Mackenzie.

Knowledge representation systems based on DLs provide their users with various inference services that allow them to deduce implicit knowledge from the explicitly represented knowledge. For instance, the *subsumption* algorithm allows one to determine subconcept-superconcept relationships. For example, w.r.t. the concept definitions from above, the concept *Human* subsumes the concept *Father* since all instances of the second concept are necessarily instances of the first concept, i.e., whenever the above concept definitions are satisfied, then *Father* is interpreted as a subset of *Human*. With the help of the subsumption algorithm, one can compute the hierarchy of all concepts defined in a TBox. This inference service is usually called *classification*. The instance algorithm can be used to check whether an individual occurring in an ABox is necessarily an instance of a given concept. For example, w.r.t. the above assertions, concept definitions, and GCIs, the individual *MACKENZIE* is an instance of the concept *Human*. With the help of the instance algorithm, one can compute answers to *instance queries*, i.e., all individuals occurring in the ABox that are instances of the query concept C .

In order to ensure a reasonable and predictable behavior of a DL system, the underlying inference problems (like the subsumption and the instance problem) should at least be decidable for the DL employed by the system, and preferably of low complexity. Consequently, the expressive power of the DL in question must be restricted in an appropriate way. If the imposed restrictions are too

severe, however, then the important notions of the application domain can no longer be specified using concept descriptions. Investigating this trade-off between the expressivity of DLs and the complexity of their inference problems has been one of the most important issues in DL research.

The general opinion on the (worst-case) complexity that is acceptable for a DL has changed dramatically over time. Historically, in the early times of DL research people have concentrated on identifying formalisms for which reasoning is tractable, i.e. can be performed in polynomial time [Pate84]. The precursor of all DL systems, KL-ONE [BrSc85], as well as its early successor systems, like KANDOR [Pate84], K-REP [MaDW91], and BACK [Pelt91], indeed employed polynomial-time subsumption algorithms. Later on, however, it turned out that subsumption in rather inexpressive DLs may be intractable [LeBr87], that subsumption in KL-ONE is even undecidable [Schm89], and that even for systems like KANDOR and BACK, for which the expressiveness of the underlying DL had been carefully restricted with the goal of retaining tractability, the subsumption problem is in fact intractable [Nebe88]. The reason for the discrepancy between the complexity of the subsumption algorithms employed in the above mentioned early DL systems and the worst-case complexity of the subsumption problems these algorithms were supposed to solve was due to the fact that these systems employed sound, but incomplete subsumption algorithms, i.e., algorithms whose positive answers to subsumption queries are correct, but whose negative answers may be incorrect. The use of incomplete algorithms has since then largely been abandoned in the DL community, mainly because of the problem that the behavior of the systems is no longer determined by the semantics of the description language: an incomplete algorithm may claim that a subsumption relationship does not hold, although it should hold according to the semantics. All the intractability results mentioned above already hold for subsumption between concept descriptions without a TBox. An even worse blow to the quest for a practically useful DL with a sound, complete, and polynomial-time subsumption algorithm was Nebel's result [Nebe90] that subsumption w.r.t. an acyclic TBox (i.e., an unambiguous set of concept definitions without cyclic dependencies) in a DL with conjunction (\sqcap) and value restriction ($\forall r.C$) is already intractable.²

At about the time when these (negative) complexity results were obtained, a new approach for solving inference problems in DLs, such as the subsumption and the instance problem, was introduced. This so-called *tableau-*

based approach was first introduced in the context of DLs by Schmidt-Schau [Schm89] and Smolka [ScSm91], though it had already been used for modal logics long before that [Fitt72]. It has turned out that this approach can be used to handle a great variety of different DLs (see [BaSa01] for an overview and, e.g., [HoSa05, HoKS06, LuMi07] for more recent results), and it yields sound and complete inference algorithms also for very expressive DLs. Although the worst-case complexity of these algorithms is quite high, the tableau-based approach nevertheless often yields practical procedures: optimized implementations of such procedures have turned out to behave quite well in applications [BFHN*94, Horr03, HaMo08], even for expressive DLs with a high worst-case complexity (ExpTime and beyond). The advent of tableau-based algorithms was the main reason why the DL community basically abandoned the search for DLs with tractable inference problems, and concentrated on the design of practical tableau-based algorithms for expressive DLs. The most prominent modern DL systems, FaCT++ [TSHo06], Racer [HaMo01b], and Pellet [SiPa04] support very expressive DLs and employ highly-optimized tableau-based algorithms. In addition to the fact that DLs are equipped with a well-defined formal semantics, the availability of mature systems that support sound and complete reasoning in very expressive description formalisms was an important argument in favor of using DLs as the foundation of OWL, the standard ontology language for the Semantic Web. In fact, OWL DL is based on the expressive DL $\mathcal{SROIQ}(\mathcal{D})$, for which reasoning is in the worst-case NExpTime-complete [HoPa04].

The research on how to extend the expressive power DLs has actually not stopped with the adoption of $\mathcal{SROIQ}(\mathcal{D})$ as the DL underlying OWL. In fact, the new version of the OWL standard, OWL 2,³ is based on the even more expressive DL $\mathcal{SROIQ}(\mathcal{D})$, which is 2NExpTime-complete [Kaza08]. The main new features of $\mathcal{SROIQ}(\mathcal{D})$ are the use of qualified number restrictions (\mathcal{Q}) rather than simple number restrictions (\mathcal{N}), and the availability of (a restricted form of) role inclusion axioms (\mathcal{R}). For example, with a simple number restriction we can describe the concept of a man that has three children

$$Man \sqcap (\geq 3 \text{ child});$$

but we cannot specify properties of these children, as in the qualified number restriction

$$Man \sqcap (\geq 3 \text{ child.Happy}).$$

2 More recent developments: Light-weight DLs and the need for novel inference tools

In this section, we first discuss the \mathcal{EL} and the DL-

² All the systems mentioned above supported these two concept constructors, which were at that time viewed as being indispensable for a DL. The DL with exactly these two concept constructors is called \mathcal{FL}_0 [Baad90c]

³ <<http://www.w3.org/TR/2009/REC-owl2-overview-20091027/>>.

““Description Logics are a well-investigated family of logic-based knowledge representation formalisms””

Litee families of light-weight DLs, and then consider inference problems different from the subsumption and the instance problem.

Light-weight DLs: the \mathcal{EL} family

The ever increasing expressive power and worst-case complexity of expressive DLs, combined with the increased use of DL-based ontology languages in practical applications due to the OWL standard, has also resulted in an increasing number of ontologies that cannot be handled by tableau-based reasoning systems without manual tuning by the system developers, despite highly optimized implementations. Perhaps the most prominent example is the well-known medical ontology SNOMED CT⁴, which comprises 380,000 concepts and is used as a standardized health care terminology in a variety of countries such as the US, Canada, and Australia. In tests performed in 2005 with FaCT++ and Racer, neither of the two systems could classify SNOMED CT [BaLS05]⁵, and Pellet still could not classify SNOMED CT in tests performed in 2008 [Meng09].

From the DL point of view, SNOMED CT is an acyclic TBox that contains only the concept constructors conjunction (\sqcap), existential restriction ($\exists r.C$), and the top concept (\top). The DL with exactly these three concept constructors is called \mathcal{EL} [BaKM99]. In contrast to its counterpart with value restrictions, \mathcal{FL}_0 , the light-weight DL \mathcal{EL} has much better algorithmic properties. Whereas subsumption without a TBox is polynomial in both \mathcal{EL} [BaKM99] and \mathcal{FL}_0 [LeBr87], subsumption in \mathcal{FL}_0

w.r.t. an acyclic TBox is coNP-complete [Nebe90] and w.r.t. GCIs it is even ExpTime-complete [BaBL05]. In contrast, subsumption in \mathcal{EL} stays tractable even w.r.t. GCIs [Bran04], and this result is stable under the addition of several interesting means of expressivity [BaBL05, BaBL08].

The *polynomial-time subsumption algorithm* for \mathcal{EL} [Bran04, BaBL05] actually classifies the given TBox \mathcal{T} , i.e., it simultaneously computes all subsumption relationships between the concept names occurring in \mathcal{T} . This algorithm proceeds in four steps:

1. Normalize the TBox.
2. Translate the normalized TBox into a graph.
3. Complete the graph using completion rules.
4. Read off the subsumption relationships from the normalized graph.

An \mathcal{EL} -TBox is *normalized* if it only contains GCIs of the following form:

$$A_1 \sqcap A_2 \sqsubseteq B, A \sqsubseteq \exists r.B, \exists r.A \sqsubseteq B,$$

where A, A_1, A_2, B are concept names or the top-concept \top . Any \mathcal{EL} -TBox can be transformed in polynomial time into a normalized one by applying equivalence-preserving normalization rules [Bran04]. In the next step, a *classification graph* $G_{\mathcal{T}} = (V, V \times V, S, R)$ is built, where

- V is the set of concept names (including \top) occurring in the normalized TBox \mathcal{T} ;
- S labels nodes with sets of concept names (again including \top);
- R labels edges with sets of role names.

The label sets are supposed to satisfy the following invariants:

- $S(A)$ contains only subsumers of A w.r.t. \mathcal{T} .

⁴ <<http://www.ihtsdo.org/snomed-ct/>>.

⁵ Note, however, that more recent versions of FaCT++ and Racer perform quite well on SNOMED CT [Meng09], due to optimizations specifically tailored towards the classification of SNOMED CT.

(R1)	$A_1 \sqcap A_2 \sqsubseteq B \in \mathcal{T}$	and	$A_1, A_2 \in S(A)$	then	add B to $S(A)$
(R2)	$A_1 \sqsubseteq \exists r.B \in \mathcal{T}$	and	$A_1 \in S(A)$	then	add r to $R(A, B)$
(R3)	$\exists r.B_1 \sqsubseteq A_1 \in \mathcal{T}$	and	$B_1 \in S(B), r \in R(A, B)$	then	add A_1 to $S(A)$

Figure 1: The completion rules for subsumption in \mathcal{EL} w.r.t. general TBoxes.

““Description Logics differ from their predecessors in that they are equipped with a formal, *logic*-based semantics””

■ $R(A,B)$ contains only roles r such that $\exists r.B$ subsumes A w.r.t. \mathcal{T} .

Initially, we set $S(A) := \{A, \top\}$ for all nodes $A \in V$, and $R(A,B) := \emptyset$ for all edges $(A,B) \in V \times V$. Obviously, the above *invariants* are satisfied by these initial label sets.

The labels of nodes and edges are then extended by applying the rules of Figure 1. Note that a rule is only applied if it really extends a label set. It is easy to see that these rules preserve the above invariants. The fact that subsumption in \mathcal{EL} w.r.t. TBoxes can be decided in polynomial time is an immediate consequence of the facts that (i) rule application terminates after a polynomial number of steps, and (ii) if no more rules are applicable then $S(A)$ contains exactly those concept names B occurring in \mathcal{T} that are subsumers of A w.r.t. \mathcal{T} (see [Bran04, BaBL05] for more details and full proofs).

Light-weight DLs: the DL-Lite family

Another problematic issue with expressive DLs is that query answering in such DLs does not scale too well to knowledge bases with a very large ABox. In this context, *queries* are conjunctions of assertions that may also contain variables, of which some can be existentially quantified. For example, the query

$$\exists y. Man(x) \wedge child(x, y) \wedge Woman(y)$$

asks for all men that have a child that is a woman⁶, but in general the use of variables allows the formulation of more complex queries than simple instance queries. In the database world, these kinds of queries are called *conjunctive queries* [AbHV95]; the difference to the pure database case is that, in addition to the instance data, we also have a TBox. As an example, consider the ABox assertions stating facts about John and Mackenzie from the previous section. Without any additional information about the meaning of the predicates *Man*, *child*, and *Woman*, the individual *JOHN* is not an answer to the above query. However, if we take the concept definitions

⁶ This simple query could also be expressed as an instance query using the \mathcal{EL} -concept description $Man \sqcap \exists child. Woman$, but in general the use of variables allows the formulation of more complex queries than simple instance queries.

“ DLs is used as the foundation of OWL, the standard ontology language for the Semantic Web ”

“ Knowledge representation systems based on DLs provide their users with various inference services that allow them to deduce implicit knowledge ”

and GCIs introduced in the previous section into account, then *JOHN* turns out to be an answer to this query.

Query answering in expressive DLs such as the already mentioned *SHOIN* (i.e., *SHOIN(D)* without concrete domains) is 2ExpTime-complete regarding combined complexity [Lutz08], i.e., the complexity w.r.t. the size of the TBox and the ABox. Thus, query answering in this logic is even harder than subsumption while at the same time being much more time critical. Moreover, query answering in *SHOIN* is coNP-complete [OrCE08] regarding data complexity (i.e., in the size of the ABox), which is viewed as "unfeasible" in the database community. These complexity hardness results for answering conjunctive queries in expressive DLs are dramatic since many DL applications, such as those that use ABoxes as web repositories, involve ABoxes with hundreds of thousands of individuals. It is a commonly held opinion that, in order to achieve truly scalable query answering in the short term, it is essential to make use of conventional relational database systems for query answering in DLs. Given this proviso, the question is what expressivity can a DL offer such that queries can be answered using relational database technology while at the same time meaningful concepts can be specified in the TBox. As an answer to this, the DL-Lite family has been introduced in [CGL+05, CDL+-KR06, CGL+07], designed to allow the implementation of conjunctive query answering "on top of" a relational database system.

DL-Lite_{core} is the basic member of the DL-Lite family [CGL+07]. Concept descriptions of this DL are of the form $A, \exists r. \top, \exists r^{-}. \top$ where A is a concept name, r is a role name, and r^{-} denotes the inverse of the role name r . A DL-Lite_{core} knowledge base (KB) consists of a TBox and an ABox. The TBox formalism allows for GCIs and disjointness axioms between DL-Lite_{core} concept descriptions $C; D: C \sqsubseteq D$ and $\text{disj}(C, D)$,

where $\text{disj}(C, D)$ states that C, D must always be interpreted as disjoint sets. A DL-Lite_{core}-ABox is a finite set of *concept and role assertions*: $A(a)$ and $r(a; b)$, where A is a concept name, r is a role name, and $a; b$ are individual names.

In contrast to \mathcal{EL} , DL-Lite cannot express *qualified* existential restrictions such as $\exists child. Woman$ in the TBox. Conversely, \mathcal{EL} does not have inverse roles, which

“ The medical ontology SNOMED CT comprises 380,000 concepts and is used as a standardized health care terminology in the US, Canada, and Australia ”

are available (albeit in a limited way) in \mathcal{DL} .

In principle, query answering in DL-Lite can be realized as follows:

1. use the TBox \mathcal{T} to reformulate the given conjunctive queries q into an first-order query $q_{\mathcal{T}}$ and then discard the TBox;
2. view the ABox \mathcal{A} as a relational database $\mathcal{I}_{\mathcal{A}}$;
3. evaluate $q_{\mathcal{T}}$ in the database $\mathcal{I}_{\mathcal{A}}$ using a relational query engine.

In practice, more work needs to be done to turn this into a scalable approach for query answering. For example, the queries $q_{\mathcal{T}}$ generated by the reformulation step are very different from the SQL queries usually formulated by humans, and thus relational database engines are not optimized for such queries.

Interestingly, also in \mathcal{EL} it is possible to implement query answering using a relational database system [LuToWo-IJCAI-09]. In contrast to the approach for DL-Lite, the TBox is incorporated into the ABox and not into the query. In addition, some limited query reformulation (independent of both the TBox and the ABox) is also required.

The relevance of the light-weight DLs discussed above is underlined by the fact that both of them are captured in the official W3C profiles⁷ document for OWL 2. Each of the OWL 2 profiles are designed for specific application requirements. For applications that rely on reasoning services for ontologies with a large number of concepts, the profile OWL 2 EL has been introduced, which is based on \mathcal{EL}^{++} , a tractable extension of \mathcal{EL} . For applications that deal with large sets of data and that mainly use the reasoning service of query answering, the profile OWL 2 QL has been defined. The DL underlying this profile is a member of the DL-Lite.

Novel inference problems

The developers of the early DL systems concentrated

on the subsumption and the instance problem, and the same was true until recently for the developers of highly optimized systems for expressive DLs. The development, maintenance, and usage of large ontologies can, however, also be profit from the use of other inference procedures. Certain non-standard inference problems, like *unification* [BaNa00, BaMo09], *matching* [BKBM99, BaKu00], and the problem of computing *least common subsumers* [BaKu98, BaKM99, BaST07, DCNS09] have been investigated for quite a while [BaKu06]. Unification and matching can, for example, help the ontology engineer to find redundancies in large ontologies, and least common subsumers and most specific concepts can be used to generate concepts from examples.

Others non-standard inference problems have, however, come into the focus of mainstream DL research only recently. One example is *conjunctive query answering*, which is not only investigated for light-weight DLs (see above), but also for expressive DLs [GHLS07, Lutz08].

Another is identification and extraction of *modules* inside an ontology. Intuitively, given an ontology \mathcal{O} and a signature Σ (i.e., a subset of the concept and role names occurring in \mathcal{O}), a module \mathcal{M} is a subset of \mathcal{O} such that the following holds for all concept descriptions C, D that can be built from symbols in Σ : C is subsumed by D w.r.t. \mathcal{O} if C is subsumed by D w.r.t. \mathcal{M} . Consequently, if one is only interested in subsumption between concepts built from symbols in Σ , it is sufficient to use \mathcal{M} instead of the (possibly much larger) whole ontology \mathcal{O} . Similarly, one can also introduce the notion of a module for other inference problems (such as query answering). An overview over different approaches for defining modules and a guideline for when to use which notion of a module can be found in [SaSZ09]. Module identification and extraction is computationally costly for expressive DLs, and even undecidable for very expressive ones such as OWL DL [LuWW07]. Both for the \mathcal{EL} family [LuWo07, Sunt08] and the DL-Lite family [KWZ-KR-08], the reasoning problems that are relevant in this area are decidable and usually of much lower complexity than for expressive DLs.

For a developer or user of a DL-based ontology, it is often quite hard to understand why a certain consequence computed by the reasoner actually follows from the knowledge base. For example, in the DL version of the medical ontology SNOMED CT, the concept *Amputation-of-Finger* is classified as a subconcept of *Amputation-of-Arm*. Finding the six axioms that are responsible for this error [BaSu08] among the more than 350,000 con-

“ DL-Lite_{core} is the basic member of the DL-Lite family ”

⁷ <<http://www.w3.org/TR/owl2-profiles/>>.

““The DL research of the last 30 years has lead to highly expressive ontology languages””

cept definitions of SNOMED CT without support by an automated reasoning tool is not easy. *Axiom pinpointing* [ScCo03] has been introduced to help developers or users of DL-based ontologies understand the reasons why a certain consequence holds by computing minimal subsets of the knowledge base that have the consequence in question (called MinAs or Explanations). There are two general approaches for computing MinAs: the *black-box* approach and the *glass-box* approach. The most naïve variant of the black-box approach considers all subsets of the ontology, and computes for each of them whether it still has the consequence or not. More sophisticated versions [KPHS07] use a variant of Reiter's [Reit87] hitting set tree algorithm to compute all MinAs. Instead of applying such a black-box approach to a large ontology, one can also first try to find a small and easy to compute subset of the ontology that contains all MinAs, and then apply the black-box approach to this subset [BaSu08]. The main advantage of the black-box approach is that it can use existing highly-optimized DL reasoners unchanged. However, it may be necessary to call the reasoner an exponential number of times. In contrast, the glass-box approach tries to find all MinAs by a single run of a modified reasoner.

Most of the glass-box pinpointing algorithms described in the DL literature (e.g., [ScCo03, PaSK05, LeMP06]) are obtained as extensions of tableau-based reasoning algorithms [BaSa01] for computing consequences from DL knowledge bases. To overcome the problem of having to design a new pinpointing extension for every tableau-based algorithm, the papers [BaPe07, BaPe09] introduce a general approach for extending tableau-based algorithms to pinpointing algorithms. This approach is based on a general notion of "tableau algorithm," which captures many of the known tableau-based algorithms for DLs and Modal Logics, but also other kinds of decision procedures, like the polynomial-time subsumption algorithm for the DL \mathcal{EL} sketched above. Any such tableau algorithm can be extended to a pinpointing algorithm, which is correct in the sense that a terminating run of the algorithm computes all MinAs. Unfortunately, however, termination need not transfer from a given tableau to its pinpointing extension, and the approach only applies to tableau-based algorithms that terminate without requiring any cycle-checking mechanism (usually called "blocking" in the DL community). Though these problems can, in principle, be solved by restricting the general framework to so-called forest tableaux [BaPe09], this solution makes the definitions and proofs more complicated and less intuitive.

In [BaPe08], a different general approach for obtaining glass-box pinpointing algorithms, which also applies to DLs for which the termination of tableau-based algorithms requires the use of blocking. It is well-known that automata working on infinite trees can often be used to construct worst-case optimal decision procedures for such DLs [BaTo01, CaGL02]. In this automata-based approach, the input inference problem \bar{A} is translated into a tree automaton \mathcal{A}_T , which is then tested for emptiness. Basically, pinpointing is then realized by transforming the tree automaton \mathcal{A}_T into a weighted tree automaton working on infinite trees, and computing the so-called behavior of this weighted automaton.

3 Conclusion

The DL research of the last 30 years has lead, on the one hand, to highly expressive ontology languages, which can nevertheless be supported by practical reasoning tools. On the other hand, the recent development of lightweight DLs and specialized reasoning tools for them ensures that DL reasoning scales to large ontologies with hundreds of thousands of terminological axioms (like SNOMED CT) and, by using database technology, to much larger sets of instance data. In addition, novel inference methods such as modularization and pinpointing support building and maintaining high-quality ontologies.

References

- [AbHV95] Serge Abiteboul, Richard Hull, and Victor Vianu. Foundations of Databases. Addison Wesley Publ. Co., Reading, Massachusetts, 1995.
- [Baad90c] Franz Baader. Terminological cycles in KL-ONE-based knowledge representation languages. In Proc. of the 8th Nat. Conf. on Artificial Intelligence (AAAI'90), pages 621-626, Boston (Ma, USA), 1990.
- [BaBL05] Franz Baader, Sebastian Brandt, and Carsten Lutz. Pushing the *EL* envelope. In Leslie Pack Kaelbling and Alessandro Saffiotti, editors, Proc. of the 19th Int. Joint Conf. on Artificial Intelligence (IJCAI 2005), pages 364-369, Edinburgh (UK), 2005. Morgan Kaufmann, Los Altos.
- [BaBL08] Franz Baader, Sebastian Brandt, and Carsten Lutz. Pushing the *EL* envelope further. In Kendall Clark and Peter F. Patel-Schneider, editors, In Proceedings of the Fifth International Workshop on OWL: Experiences and Directions (OWLED'08), Karlsruhe, Germany, 2008.
- [BCNMP03] Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors. The Description Logic Handbook: Theory, Implementation, and Applications. Cambridge University Press, 2003.
- [BFHN*94] Franz Baader, Enrico Franconi, Bernhard Hollunder, Bernhard Nebel, and Hans-Jürgen Protlich. An empirical analysis of optimization techniques for terminological representation systems or: Making KRIS get a move on. Applied Artificial Intelligence. Special Issue on Knowledge Based Management, 4:109-132, 1994.
- [BaKu98] Franz Baader and Ralf Küsters. Computing the least common subsumer and the most specific concept in the presence of cyclic ALN-concept descriptions. In Proc. of the 22nd German Annual Conf. on Artificial Intelligence (KI'98), volume 1504 of Lecture Notes in Computer Science, pages 129-140. Springer-Verlag, 1998.
- [BaKu00] Franz Baader and Ralf Küsters. Matching in description logics with existential restrictions. In Proc. of the 7th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR 2000), pages 261-272, 2000.
- [BaKu06] Franz Baader and Ralf Küsters. Nonstandard inferences in description logics: The story so far. In D.M. Gabbay, S.S. Goncharov, and M. Zakharyashev, editors, Mathematical Problems from Applied Logic I, volume 4 of International Mathematical Series, pages 1-75. Springer-Verlag, 2006.
- [BKBM99] Franz Baader, Ralf Küsters, Alex Borgida, and Deborah L. McGuinness. Matching in description logics. J. of Logic and Computation, 9(3):411-447, 1999.
- [BaKM99] Franz Baader, Ralf Küsters, and Ralf Molitor. Computing least common subsumers in description logics with existential restrictions. In Proc. of the 16th Int. Joint Conf. on Artificial Intelligence (IJCAI'99), pages 96-101, 1999.
- [BaLS05] Franz Baader, Carsten Lutz, and Boontawe Sontisrivaraporn. Is tractable reasoning in extensions of the description logic *EL* useful in practice? In Proceedings of the 2005 International Workshop on Methods for Modalities (M4M-05), 2005.
- [BaMo09] Franz Baader and Barbara Morawska. Unification in the description logic *EL*. In Ralf Treinen, editor, Proc. of the 20th Int. Conf. on Rewriting Techniques and Applications (RTA 2009), volume 5595 of Lecture Notes in Computer Science, pages 350-364. Springer-Verlag, 2009.
- [BaNa00] Franz Baader and Paliath Narendran. Unification of concepts terms in description logics. J. of Symbolic Computation, 31(3):277-305, 2001.
- [BaPe07] Franz Baader and Rafael Peñaloza. Axiom pinpointing in general tableaux. In Proc. of the Int. Conf. on Analytic Tableaux and Related Methods (TABLEAUX 2007), volume 4548 of Lecture Notes in Artificial Intelligence, pages 11-27. Springer-Verlag, 2007.
- [BaPe08] Franz Baader and Rafael Peñaloza. Automata-based axiom pinpointing. In Alessandro Armando, Peter Baumgartner, and Gilles Dowek, editors, Proc. of the Int. Joint Conf. on Automated Reasoning (IJCAR 2008), volume 5195 of Lecture Notes in Artificial Intelligence, pages 226-241. Springer-Verlag, 2008.
- [BaPe09] Franz Baader and Rafael Peñaloza. Axiom pinpointing in general tableaux. Journal of Logic and Computation, 2009. To appear.
- [BaSa01] Franz Baader and Ulrike Sattler. An overview of tableau algorithms for description logics. Studia Logica, 69:5-40, 2001.
- [BaST07] Franz Baader, Baris Sertkaya, and Anni-Yasmin Turhan. Computing the least common subsumer w.r.t. a background terminology. J. of Applied Logic, 5(3):392-420, 2007.
- [BaSu08] Franz Baader and Boontawe Sontisrivaraporn. Debugging SNOMED CT using axiom pinpointing in the description logic *EL+*. In Proceedings of the International Conference on Representing and Sharing Knowledge Using SNOMED (KR-MED'08), Phoenix, Arizona, 2008.
- [BaTo01] Franz Baader and Stephan Tobies. The inverse method implements the automata approach for modal satisfiability. In Proc. of the Int. Joint Conf. on Automated Reasoning (IJCAR 2001), volume 2083 of Lecture Notes in Artificial Intelligence, pages 92-106. Springer-Verlag, 2001.
- [BrLe85] Ronald J. Brachman and Hector J. Levesque, editors. Readings in Knowledge Representation. Morgan Kaufmann, Los Altos, 1985.
- [BrSc85] Ronald J. Brachman and James G. Schmolze. An overview of the KL-ONE knowledge representation system. Cognitive Science, 9(2):171-216, 1985.
- [Bran04] Sebastian Brandt. Polynomial time reasoning in a description logic with existential restrictions, GCI axioms, and what else? In Ramon López de Mántaras and Lorenza Saitta, editors, Proc. of the 16th Eur. Conf. on Artificial Intelligence (ECAI 2004), pages 298-302, 2004.
- [CGL+05] Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. DL-Lite: Tractable description logics for ontologies. In Manuela M. Veloso and Subbarao Kambhampati, editors, Proc. of the 20th Nat. Conf. on Artificial Intelligence (AAAI 2005), pages 602-607. AAAI Press/The MIT Press, 2005.
- [CDL+-KR06] Diego Calvanese, Giuseppe de Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. Data complexity of query answering in description logics. In

- Patrick Doherty, John Mylopoulos, and Christopher A. Welty, editors, Proc. of the 10th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR 2006), pages 260-270. AAAI Press/The MIT Press, 2006.
- [CGL+07] Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. Tractable reasoning and efficient query answering in description logics: The DL-Lite family. *J. of Automated Reasoning*, 39(3):385-429, 2007.
- [CaGL02] Diego Calvanese, Giuseppe DeGiacomo, and Maurizio Lenzerini. 2ATAs make DLs easy. In Proc. of the 2002 Description Logic Workshop (DL 2002), pages 107-118. CEUR Electronic Workshop Proceedings, <http://ceur-ws.org/Vol-53/>, 2002.
- [DCNS09] Francesco M. Donini, Simona Colucci, Tommaso Di Noia, and Eugenio Di Sciascio. A tableaux-based method for computing least common subsumers for expressive description logics. In Craig Boutilier, editor, Proc. of the 21st Int. Joint Conf. on Artificial Intelligence (IJCAI 2009), pages 739-745, 2009.
- [Fitt72] Melvin Fitting. Tableau methods of proof for modal logics. *Notre Dame J. of Formal Logic*, 13(2):237-247, 1972.
- [GHLS07] Birte Glimm, Ian Horrocks, Carsten Lutz, and Ulrike Sattler. Conjunctive query answering for the description logic *SHIQ*. In Manuela M. Veloso, editor, Proc. Of the 20th Int. Joint Conf. on Artificial Intelligence (IJCAI 2007), pages 399-404, Hyderabad, India, 2007.
- [HaMo01b] Volker Haarslev and Ralf Möller. RACER system description. In Proc. of the Int. Joint Conf. on Automated Reasoning (IJCAR 2001), volume 2083 of Lecture Notes in Artificial Intelligence, pages 701-706. Springer-Verlag, 2001.
- [HaMo08] Volker Haarslev and Ralf Möller. On the scalability of description logic instance retrieval. *J. of Automated Reasoning*, 41(2):99-142, 2008.
- [Haye79] Patrick J. Hayes. The logic of frames. In D. Metzger, editor, *Frame Conceptions and Text Understanding*, pages 46-61. Walter de Gruyter and Co., 1979. Republished in [BrLe85].
- [Horr03] Ian Horrocks. Implementation and optimization techniques. In [BCNMP03], pages 306-346. 2003.
- [HoKS06] Ian Horrocks, Oliver Kutz, and Ulrike Sattler. The even more irresistible *SROIQ*. In Patrick Doherty, John Mylopoulos, and Christopher A. Welty, editors, Proc. of the 10th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR 2006), pages 57-67, Lake District, UK, 2006. AAAI Press/The MIT Press.
- [HoPa04] Ian Horrocks and Peter F. Patel-Schneider. Reducing OWL entailment to description logic satisfiability. *J. Web Sem.*, 1(4):345-357, 2004.
- [HoPH03] Ian Horrocks, Peter F. Patel-Schneider, and Frank van Harmelen. From SHIQ and RDF to OWL: The making of a web ontology language. *Journal of Web Semantics*, 1(1):7-26, 2003.
- [HoSa05] Ian Horrocks and Ulrike Sattler. A tableaux decision procedure for *SHOIQ*. In Proc. of the 19th Int. Joint Conf. on Artificial Intelligence (IJCAI 2005), Edinburgh (UK), 2005. Morgan Kaufmann, Los Altos.
- [KPHS07] Aditya Kalyanpur, Bijan Parsia, Matthew Horridge, and Evren Sirin. Finding all justifications of OWL DL entailments. In Proceedings of the 6th International Semantic Web Conference and 2nd Asian Semantic Web Conference, ISWC 2007 + ASWC 2007, volume 4825 of Lecture Notes in Computer Science, pages 267-280, Busan, Korea, 2007. Springer-Verlag.
- [Kaza08] Yevgeny Kazakov. *RIQ* and *SROIQ* are harder than *SHOIQ*. In Gerhard Brewka and Jérôme Lang, editors, Proc. of the 11th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR 2008), pages 274-284. AAAI Press, 2008.
- [KWZ-KR-08] Roman Kontchakov, Frank Wolter, and Michael Zakharyashev. Can you tell the difference between DL-Lite ontologies? In Gerhard Brewka and Jérôme Lang, editors, Proc. of the 11th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR 2008), pages 285-295. Morgan Kaufmann, Los Altos, 2008.
- [LeMP06] Kevin Lee, Thomas Meyer, and Jeff Z. Pan. Computing maximally satisfiable terminologies for the description logic ALC with GCIs. In Proc. of the 2006 Description Logic Workshop (DL 2006), volume 189 of CEUR Electronic Workshop Proceedings, 2006.
- [LeBr87] Hector J. Levesque and Ron J. Brachman. Expressiveness and tractability in knowledge representation and reasoning. *Computational Intelligence*, 3:78-93, 1987.
- [Lutz08] Carsten Lutz. The complexity of conjunctive query answering in expressive description logics. In Alessandro Armando, Peter Baumgartner, and Gilles Dowek, editors, Proc. of the Int. Joint Conf. on Automated Reasoning (IJCAR 2008), Lecture Notes in Artificial Intelligence, pages 179-193. Springer-Verlag, 2008.
- [LuMi07] Carsten Lutz and Maja Milicic. A tableau algorithm for description logics with concrete domains and general tboxes. *J. of Automated Reasoning*, 38(1-3):227-259, 2007.
- [LuToWo-IJCAI-09] Carsten Lutz, David Toman, and Frank Wolter. Conjunctive query answering in the description logic *EL* using a relational database system. In Proceedings of the 21st International Joint Conference on Artificial Intelligence IJCAI09. AAAI Press, 2009. To appear.
- [LuWW07] Carsten Lutz, Dirk Walther, and Frank Wolter. Conservative extensions in expressive description logics. In Manuela M. Veloso, editor, Proc. of the 20th Int. Joint Conf. on Artificial Intelligence (IJCAI 2007), pages 453-458, Hyderabad, India, 2007.
- [LuWo07] Carsten Lutz and Frank Wolter. Conservative extensions in the lightweight description logic *EL*. In Frank Pfenning, editor, Proc. of the 21st Int. Conf. on Automated Deduction (CADE 2007), volume 4603 of Lecture Notes in Computer Science, pages 84-99, Bremen, Germany, 2007. Springer-Verlag.
- [MaDW91] E. Mays, R. Dionne, and R. Weida. K-REP system overview. *SIGART Bull.*, 2(3), 1991.
- [Mins81] Marvin Minsky. A framework for representing knowledge. In John Haugeland, editor, *Mind Design*. The MIT Press, 1981. A longer version appeared in *The Psychology of Computer Vision* (1975). Re-

- published in [BrLe85].
- [Nebe88] Bernhard Nebel. Computational complexity of terminological reasoning in BACK. *Artificial Intelligence*, 34(3):371-383, 1988.
- [Nebe90] Bernhard Nebel. Terminological reasoning is inherently intractable. *Artificial Intelligence*, 43:235-249, 1990.
- [OrCE08] Magdalena Ortiz, Diego Calvanese, and Thomas Eiter. Data complexity of query answering in expressive description logics via tableaux. *J. of Automated Reasoning*, 41(1):61-98, 2008.
- [PaSK05] Bijan Parsia, Evren Sirin, and Aditya Kalyanpur. Debugging OWL ontologies. In Allan Ellis and Tatsuya Hagino, editors, *Proc. of the 14th International Conference on World Wide Web (WWW'05)*, pages 633-640. ACM, 2005.
- [Pate84] Peter F. Patel-Schneider. Small can be beautiful in knowledge representation. In *Proc. of the IEEE Workshop on Knowledge-Based Systems*, 1984. An extended version appeared as Fairchild Tech. Rep. 660 and FLAIR Tech. Rep. 37, October 1984.
- [Pelt91] Christof Peltason. The BACK system: an overview. *SIGART Bull.*, 2(3):114-119, 1991.
- [Quil67] M. Ross Quillian. Word concepts: A theory and simulation of some basic capabilities. *Behavioral Science*, 12:410-430, 1967. Republished in [BrLe85].
- [Reit87] R. Reiter. A theory of diagnosis from first principles. *Artificial Intelligence*, 32(1):57-95, 1987.
- [SaSZ09] Ulrike Sattler, Thomas Schneider, and Michael Zakharyashev. Which kind of module should I extract? In *Proc. of the 2008 Description Logic Workshop (DL 2009)*, volume 477 of *CEUR Workshop Proceedings*, 2009.
- [ScCo03] Stefan Schlobach and Ronald Cornet. Non-standard reasoning services for the debugging of description logic terminologies. In Georg Gottlob and Toby Walsh, editors, *Proc. of the 18th Int. Joint Conf. on Artificial Intelligence (IJCAI 2003)*, pages 355-362, Acapulco, Mexico, 2003. Morgan Kaufmann, Los Altos.
- [Schm89] Manfred Schmidt-Schauß. Subsumption in KL-ONE is undecidable. In Ron J. Brachman, Hector J. Levesque, and Ray Reiter, editors, *Proc. of the 1st Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR'89)*, pages 421-431. Morgan Kaufmann, Los Altos, 1989.
- [ScSm91] Manfred Schmidt-Schauß and Gert Smolka. Attributive concept descriptions with complements. *Artificial Intelligence*, 48(1):1-26, 1991.
- [ScGC79] Len K. Schubert, Randy G. Goebel, and Nicola J. Cercone. The structure and organization of a semantic net for comprehension and inference. In N. V. Findler, editor, *Associative Networks: Representation and Use of Knowledge by Computers*, pages 121-175. Academic Press, 1979.
- [SiPa04] Evren Sirin and Bijan Parsia. Pellet: An OWL DL reasoner. In *Proc. of the 2004 Description Logic Workshop (DL 2004)*, pages 212-213, 2004.
- [Sunt08] Boontawee Suntisrivaraporn. Module extraction and incremental classification: A pragmatic approach for *EL+* ontologies. In Sean Bechhofer, Manfred Hauswirth, Joerg Hoffmann, and Manolis Koubarakis, editors, *Proceedings of the 5th European Semantic Web Conference (ESWC'08)*, volume 5021 of *Lecture Notes in Computer Science*, pages 230-244. Springer-Verlag, 2008.
- [Meng09] Boontawee Suntisrivaraporn. Polynomial-Time Reasoning Support for Design and Maintenance of Large-Scale Biomedical Ontologies. PhD thesis, Fakultät Informatik, TU Dresden, 2009. <<http://lat.inf.tu-dresden.de/research/phd/#Sun-PhD-2008>>.
- [TSHo06] Dmitry Tsarkov and Ian Horrocks. Fact++ description logic reasoner: System description. In Ulrich Furbach and Natarajan Shankar, editors, *Proc. of the Int. Joint Conf. on Automated Reasoning (IJCAR 2006)*, volume 4130 of *Lecture Notes in Artificial Intelligence*, pages 292-297. Springer-Verlag, 2006.