



Deutsches  
Forschungszentrum  
für Künstliche  
Intelligenz GmbH

**Research  
Report**  
RR-91-15

# **Attributive Description Formalisms and the Rest of the World**

**Bernhard Nebel and Gert Smolka**

**May 1991**

**Deutsches Forschungszentrum für Künstliche Intelligenz  
GmbH**

Postfach 20 80  
D-6750 Kaiserslautern, FRG  
Tel.: (+49 631) 205-3211/13  
Fax: (+49 631) 205-3210

Stuhlsatzenhausweg 3  
D-6600 Saarbrücken 11, FRG  
Tel.: (+49 681) 302-5252  
Fax: (+49 681) 302-5341

# Deutsches Forschungszentrum für Künstliche Intelligenz

The German Research Center for Artificial Intelligence (Deutsches Forschungszentrum für Künstliche Intelligenz, DFKI) with sites in Kaiserslautern und Saarbrücken is a non-profit organization which was founded in 1988 by the shareholder companies ADV/Orga, AEG, IBM, Insiders, Fraunhofer Gesellschaft, GMD, Krupp-Atlas, Mannesmann-Kienzle, Philips, Siemens and Siemens-Nixdorf. Research projects conducted at the DFKI are funded by the German Ministry for Research and Technology, by the shareholder companies, or by other industrial contracts.

The DFKI conducts application-oriented basic research in the field of artificial intelligence and other related subfields of computer science. The overall goal is to construct *systems with technical knowledge and common sense* which - by using AI methods - implement a problem solution for a selected application area. Currently, there are the following research areas at the DFKI:

- Intelligent Engineering Systems
- Intelligent User Interfaces
- Intelligent Communication Networks
- Intelligent Cooperative Systems.

The DFKI strives at making its research results available to the scientific community. There exist many contacts to domestic and foreign research institutions, both in academy and industry. The DFKI hosts technology transfer workshops for shareholders and other interested groups in order to inform about the current state of research.

From its beginning, the DFKI has provided an attractive working environment for AI researchers from Germany and from all over the world. The goal is to have a staff of about 100 researchers at the end of the building-up phase.

Prof. Dr. Gerhard Barth  
Director

# **Attributive Description Formalisms and the Rest of the World**

**Bernhard Nebel and Gert Smolka**

DFKI-RR-91-15

© Deutsches Forschungszentrum für Künstliche Intelligenz 1991

This work may not be copied or reproduced in whole or in part for any commercial purpose. Permission to copy in whole or in part without payment of fee is granted for nonprofit educational and research purposes provided that all such whole or partial copies include the following: a notice that such copying is by permission of Deutsches Forschungszentrum für Künstliche Intelligenz, Kaiserslautern, Federal Republic of Germany; an acknowledgement of the authors and individual contributors to the work; all applicable portions of this copyright notice. Copying, reproducing, or republishing for any other purpose shall require a licence with payment of fee to Deutsches Forschungszentrum für Künstliche Intelligenz.

To appear in: O. Herzog, C. Rollinger (eds.), *Textunderstanding in LILOG: Integrating Computational Linguistics and Artificial Intelligence*, Springer-Verlag, Berlin, Germany, 1991.

# Attributive Description Formalisms ... and the Rest of the World

Bernhard Nebel and Gert Smolka

German Research Center for Artificial Intelligence  
Saarbrücken, Germany

## Abstract

Research in knowledge representation has led to the development of so-called terminological logics, which have the purpose to support the representation of the conceptual and terminological part of Artificial Intelligence applications. Independently, in Computational Linguistics so-called feature logics have been developed, which are aimed at representing the semantic and syntactic information natural language sentences convey. Since both of these logics rely mainly on attributes as the primary notational primitives for representing knowledge, they can be jointly characterized as attributive description formalisms.

Although the intended applications for terminological logics and feature logics are not identical, and the computational services of systems based on the respective formalisms are quite different for this reason, the logical foundations turn out to be very similar – as we pointed out elsewhere. In this paper, we will show how attributive description formalisms relate to “the rest of the world.” Recently, a number of formal results in the area of attributive description formalisms have been obtained by exploiting other research fields, such as formal language theory, automata theory, and modal logics. This connection between different fields of formal research will be highlighted in the sequel.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Logical Foundations</b>	<b>3</b>
<b>3</b>	<b>Regular Languages and Finite State Automata</b>	<b>6</b>
3.1	Terminological Axioms and the Lexicon . . . . .	7
3.2	Terminological Cycles . . . . .	9
<b>4</b>	<b>Thue Systems</b>	<b>10</b>
4.1	Feature Agreement and Role-Value-Maps . . . . .	10
4.2	Arbitrary Axioms . . . . .	11
4.3	Functional Uncertainty . . . . .	12
<b>5</b>	<b>Modal Logics</b>	<b>13</b>
<b>6</b>	<b>Conclusion</b>	<b>15</b>

# 1 Introduction

*Terminological logics*, which have their roots in the knowledge representation formalism KL-ONE [Brachman, 1979; Brachman and Schmolze, 1985], have been developed to support the representation of the conceptual and terminological part of Artificial Intelligence applications.

Starting with *primitive concepts* and *attributes* (in this context usually called *roles*), new concepts are defined by employing attributive descriptions. For instance, supposing the *concept* Human and the *attribute* child, the concept of a Parent can be defined by the description

a Human who has at least one child who in turn is a Human,

or, more formally,

$$\text{Parent} = \text{Human} \sqcap \exists \text{child: Human}.$$

The main computational services provided by terminological representation systems are the computation of the *concept hierarchy* according to the *subsumption*-relation between concepts and the computation of *instance relationships* between concepts and objects of the application domain.

*Feature logics* grew out of research in Computational Linguistics. They are the constraint logic underlying the family of *unification grammars* that originated with Lexical Functional Grammar (LFG) [Kaplan and Bresnan, 1982] and Functional Unification Grammar (FUG) [Kay, 1979; Kay, 1985]. In unification grammars, syntactical and semantical objects are described by employing attributive descriptions. For instance, the class of linguistic objects that are

third-person singular noun phrases

can be described formally as follows [Shieber, 1986]:

$$\left[ \begin{array}{l} \text{cat:} \quad \text{NP} \\ \text{agreement:} \quad \left[ \begin{array}{ll} \text{number:} & \text{singular} \\ \text{person:} & \text{third} \end{array} \right] \end{array} \right]$$

or, in a linear notation as:

$$\text{cat: NP} \sqcap \text{agreement: (number: singular} \sqcap \text{person: third)}.$$

While parsing a sentence, such descriptions are combined by “unification,” and, in the end, the combined descriptions provide the syntactic and semantic structure of the sentence. One main step during this process is

the test whether a newly formed description is *satisfiable*, i.e., describes any linguistic structure at all.

As we pointed out in [Nebel and Smolka, 1990], terminological logics and feature logics are closely related. Although the intended applications are not identical, and for this reason, the computational services of systems based on the respective logics are quite different, the logical foundations turn out to be the same. Both logics employ restrictions on attributes as the primary notational primitives and are best formalized using a Tarski-style model theory. The main difference between terminological logics and feature logics is that the former permit set-valued attributes (called *roles*), while the latter permit only single-valued attributes (called *features*). This seemingly minor difference has drastic consequences as it comes to computational complexity. Nevertheless, for a large range of problems, formal results apply to both kinds of logics.

In the the LILOG project, there two applications of attributive descriptions. The STUF formalism [Bouma *et al.*, 1988; Dörre and Seiffert, 1991] is based on feature logic and is employed in the linguistic components. The knowledge representation language L-LILOG [Pletat and von Luck, 1989; Pletat, 1991] is a hybrid formalism combining predicate logic and attributive descriptions.

The remainder of the paper is organized as follows. In the next section, we will briefly introduce the logical foundations of terminological and feature logics. Section 3 shows the applicability of results from *automata theory* to attributive description languages in terms of computational complexity results and algorithms. Section 4 summarizes a number of undecidability results which have been obtained by reductions of the *word problem for Thue systems*. In fact, for some proofs a slightly stronger condition is necessary, namely, that the semigroup generated by the Thue system is a group. In particular, we consider the problem of determining satisfiability for feature terms containing *functional uncertainty* in the case that the feature logic is propositionally complete. In Section 5, a correspondence between a certain terminological logic and the *propositional polymodal logic*  $K_{(m)}$  is considered, which leads to quite a number of interesting applications of results from modal and dynamic logic to attributive description formalisms. Finally, in the conclusion we will sketch some applications of results achieved in the area of attribute descriptions to other research fields. A summary of the relations discussed in the paper is shown in Figure 1.



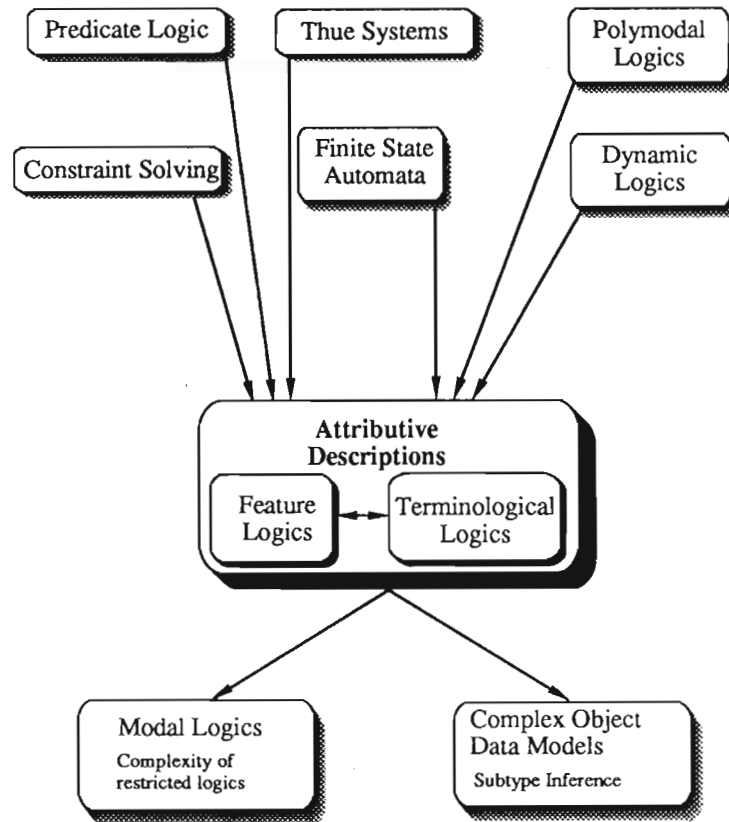


Figure 1: Attributive Description Formalisms and the Relation to the Rest of the World

## 2 Logical Foundations

While terminological logics were introduced originally with an informal semantics only, it became quickly obvious that a formal semantics is necessary to describe the intended meaning – and the obvious candidate, first-order predicate calculus and its associated model theory, was used for this purpose [Schmolze and Israel, 1983; Brachman and Levesque, 1984]. A similar process took place in the area of unification grammars [Kasper and Rounds, 1986; Johnson, 1987; Smolka, 1988].

This logical reconstruction revealed in both cases that the formalisms correspond to subsets of ordinary first-order predicate logic. Although this correspondence is very helpful for understanding the meaning of the formalism and yields a firm base for extensions, it does not help much in determining the computational properties. Nevertheless, a logical foundation is

a necessary prerequisite for an analysis of computational properties. In the following, the logical foundations of attributive description formalisms are briefly recalled.

In terminological logics, we start with an alphabet  $\mathbf{C}$  of **concept symbols** (denoted by  $C$ ) and an alphabet  $\mathbf{R}$  of **role symbols** (denoted by  $R$ ), which are disjoint. Concept symbols are intended to denote some subset of a domain, and role symbols are intended to denote unary, set-valued functions or, equivalently, two-place relations on the domain.<sup>1</sup> From concept and role symbols, complex **concept descriptions** (denoted by  $D$ ) are composed using a variety of description-forming operations. In order to give an example, the language  $\mathcal{ALC}$  will be specified, originally introduced in [Schmidt-Schauß and Smolka, 1991]:

$$D \longrightarrow C \mid \top \mid \perp \mid D \sqcap D' \mid D \sqcup D' \mid \neg D \mid \forall R: D \mid \exists R: D.$$

The formal meaning of concept descriptions built according to the above rule is given by an interpretation  $\mathcal{I} = (\mathbf{D}^{\mathcal{I}}, \cdot^{\mathcal{I}})$ , where  $\mathbf{D}^{\mathcal{I}}$  (the **domain**) is an arbitrary nonempty set and  $\cdot^{\mathcal{I}}$  (the **interpretation function**) is a function such that:

$$\begin{aligned} C^{\mathcal{I}} &\subseteq \mathbf{D}^{\mathcal{I}} \\ R^{\mathcal{I}} &\subseteq \mathbf{D}^{\mathcal{I}} \times \mathbf{D}^{\mathcal{I}}. \end{aligned}$$

The denotation of complex concept descriptions is inductively given by:

$$\begin{aligned} \top^{\mathcal{I}} &= \mathbf{D}^{\mathcal{I}} \\ \perp^{\mathcal{I}} &= \emptyset \\ (D \sqcap D')^{\mathcal{I}} &= D^{\mathcal{I}} \cap D'^{\mathcal{I}} \\ (D \sqcup D')^{\mathcal{I}} &= D^{\mathcal{I}} \cup D'^{\mathcal{I}} \\ (\neg D)^{\mathcal{I}} &= \mathbf{D}^{\mathcal{I}} - D^{\mathcal{I}} \\ (\forall R: D)^{\mathcal{I}} &= \{d \in \mathbf{D}^{\mathcal{I}} \mid R^{\mathcal{I}}(d) \subseteq D^{\mathcal{I}}\} \\ (\exists R: D)^{\mathcal{I}} &= \{d \in \mathbf{D}^{\mathcal{I}} \mid R^{\mathcal{I}}(d) \cap D^{\mathcal{I}} \neq \emptyset\}. \end{aligned}$$

Based on this semantics, the notion of **subsumption** mentioned above is defined as set-inclusion. A concept  $D$  is **subsumed** by another concept  $D'$ , written  $D \preceq D'$ , iff  $(D)^{\mathcal{I}} \subseteq (D')^{\mathcal{I}}$  for every interpretation  $\mathcal{I}$ . From this relation, a concept hierarchy can be computed. If the logic is extended to describe single objects by using role and concept symbols, then the notion of **instance relationship** can be formalized as set-membership in concepts.

Note that one can think of quite different terminological logics employing, for instance, role-forming operators, cardinality restrictions on roles, and so

---

<sup>1</sup>We will use both notations interchangeably.

on. Indeed, quite a number of different representation systems have been built using a variety of terminological logics (for a survey, see [Nebel, 1990a]).

Turning now to feature logic, we notice that the formalization of so-called *feature terms* resembles the formalization of concept descriptions. In feature logics, we start with three pairwise disjoint alphabets, namely, a set **S** of **sort symbols** (denoted by  $S$ ), a set **F** of **feature symbols** (denoted by  $f$ ), and a set **A** of **atoms** (denoted by  $a$ ). Based on that, the following rule (see, e.g., [Smolka, 1988]) specifies how to built complex **feature terms** (denoted by  $F$ ):

$$F \longrightarrow a | S | \top | \perp | F \sqcap F' | F \sqcup F' | \neg F | (f_1 \dots f_n) : F | (f_{1,1} \dots f_{1,m}) \downarrow (f_{2,1} \dots f_{2,n}).$$

The formal meaning is provided by interpretations  $\mathcal{I} = (\mathbf{D}^{\mathcal{I}}, \cdot^{\mathcal{I}})$ , also called **feature algebras** in this context, where  $\mathbf{D}^{\mathcal{I}}$  is a nonempty set and  $\cdot^{\mathcal{I}}$  is a function such that:

$$\begin{aligned} a^{\mathcal{I}} &\in \mathbf{D}^{\mathcal{I}} \\ S^{\mathcal{I}} &\subseteq \mathbf{D}^{\mathcal{I}} \\ f^{\mathcal{I}} &\subseteq \mathbf{D}^{\mathcal{I}} \times \mathbf{D}^{\mathcal{I}}. \end{aligned}$$

Additionally, the restrictions

$$\begin{aligned} (d, e), (d, e') \in f^{\mathcal{I}} &\implies e = e' \\ a \neq b &\implies a^{\mathcal{I}} \neq b^{\mathcal{I}} \\ a \in \mathbf{A}, f \in \mathbf{F}, d \in \mathbf{D}^{\mathcal{I}} &\implies (a^{\mathcal{I}}, d) \notin f^{\mathcal{I}}, \end{aligned}$$

have to be satisfied formalizing that features are functional, that different atoms denote different elements in the domain, and that atoms are never in the domain of a feature.

The meaning of chains of features  $f_1 \dots f_n$ , also called **feature paths**, is the composition of functional relations:

$$(d, e) \in f_1 \dots f_n^{\mathcal{I}} \iff \exists d_0, \dots, d_n : d_0 = d \wedge d_n = e \wedge \bigwedge_{i=1}^n (d_{i-1}, d_i) \in f_i^{\mathcal{I}}$$

Feature paths will also be denoted by the letters  $p$  and  $q$ . Using these definitions, the denotation of complex feature terms is given inductively by:

$$\begin{aligned} (a)^{\mathcal{I}} &= \{a^{\mathcal{I}}\} \\ \top^{\mathcal{I}} &= \mathbf{D}^{\mathcal{I}} \\ \perp^{\mathcal{I}} &= \emptyset \\ (F \sqcap F)^{\mathcal{I}} &= F^{\mathcal{I}} \cap F^{\mathcal{I}} \end{aligned}$$

$$\begin{aligned}
(F \sqcup F)^{\mathcal{I}} &= F^{\mathcal{I}} \cup F^{\mathcal{I}} \\
(\neg F)^{\mathcal{I}} &= \mathbf{D}^{\mathcal{I}} - F^{\mathcal{I}} \\
(p: F)^{\mathcal{I}} &= \{d \in \mathbf{D}^{\mathcal{I}} \mid \emptyset \neq p^{\mathcal{I}}(d) \subseteq F^{\mathcal{I}}\} \\
(p \downarrow q)^{\mathcal{I}} &= \{d \in \mathbf{D}^{\mathcal{I}} \mid p^{\mathcal{I}}(d) = q^{\mathcal{I}}(d) \neq \emptyset\}.
\end{aligned}$$

A feature term  $F$  is **satisfiable** iff there exists an interpretation such that  $F^{\mathcal{I}} \neq \emptyset$ .

If attributive descriptions formalisms contain **intersection**  $\sqcap$  and **complement**  $\neg$ , they are called **propositionally complete**. In such formalisms, the notions of satisfiability and subsumption are obviously closely related. More precisely, subsumption and unsatisfiability are linear time reducible to each other (see, e.g., [Nebel and Smolka, 1990]).

### 3 Regular Languages and Finite State Automata

As mentioned in the previous section, the logical semantics for attributive description formalisms proved to be quite useful in understanding the expressive power of these formalisms. Terminological logics as well as feature logics are obviously subsets of ordinary first-order logic. These subsets, however, were unexplored previously with respect to their computational properties. For instance, it was not known until 1988 whether there are undecidable terminological logics [Schild, 1988] and only in 1989, it was shown that subsumption in KL-ONE [Brachman and Schmolze, 1985; Schmidt-Schauß, 1989] and NIKL [Moser, 1983; Patel-Schneider, 1989b] is undecidable – a point, we will return to in the next section.

Since in knowledge representation and computational linguistics, efficiency is an important issue, decidability of a formalism is not the only concern. *Tractability*, i.e. solvability in polynomial time, is also relevant. As a matter of fact, Brachman and Levesque [1984] requested that knowledge representation formalisms should always permit polynomial time computations. They started an inquiry concerning the tradeoff between expressiveness and tractability of representation formalisms, which led to a number of analyses of different terminological logics [Nebel, 1988; Patel-Schneider, 1989a; Schmidt-Schauß and Smolka, 1991]. However, only recently, terminological logics that are maximally expressive and still tractable were identified [Donini *et al.*, 1991a] using the *constraint solving* technique introduced in [Schmidt-Schauß and Smolka, 1991].

Another open problem was whether the computational complexity of subsumption for tractable terminological logics is preserved under the introduc-

tion of *terminological axioms*. This problem was solved by discovering a correspondence between *nondeterministic finite state automata* and a particular, simple terminological logic. Exploiting complexity results from the theory of finite state automata, it was possible to show that the addition of terminological axioms increases the computational complexity considerably [Nebel, 1990b]. Further, the mentioned correspondence proved to be useful for characterizing the semantics of so-called *terminological cycles* [Baader, 1990; Nebel, 1991].

### 3.1 Terminological Axioms and the Lexicon

Investigations of the computational complexity of terminological logics are usually based on the semantics given in Section 2. It is analyzed, how much resources are necessary for checking subsumption between two concept descriptions. In particular, it is assumed that all concept symbols appearing in the descriptions are undefined. In existing systems, however, it is possible to assign a *name* to a concept description and to use this new name in other expressions instead of the original description. This aspect of the use of terminological logics can be straightforwardly formalized by the notion of **terminological axioms**, which have the following form:

$$C \doteq D$$

Usually, it is assumed that sets of such axioms, also called **terminologies** (denoted by  $T$ ), satisfy two restrictions, namely,

1. a concept symbol  $C$  appears at most once on a left hand side of a terminological axiom, and
2. the terminology is cycle-free, i.e., there is a partial order on the set of concepts  $C$  such that for every terminological axiom  $C \doteq D$ , every concept symbol in  $D$  is strictly less than  $C$ .

If  $C$  appears on the left hand side of a terminological axiom, it is also called a **defined concept**. If it never appears on the left hand side, it is said to be a **primitive concept**.

Supposing such a terminology  $T$ , subsumption is relativized to this terminology, written as  $D \preceq_T D'$ , by considering set-inclusion of concept denotations only in interpretations that are *models* of the terminology. An interpretation  $\mathcal{I}$  is a **model** of a set of terminological axioms iff for all axioms  $C \doteq D$  the interpretation satisfies  $C^{\mathcal{I}} = D^{\mathcal{I}}$ .

Given the above restrictions, subsumption relative to a terminology can be easily reduced to subsumption over concept descriptions relative to “the

empty terminology” by *expanding* all defined concepts by their definitions. However, in the worst case, this can lead to an exponential increase of the size of a concept description [Nebel, 1990b]. Thus, even when subsumption determination for a particular terminological logic is tractable, this does not mean that subsumption determination relative to terminologies is tractable, as well. On the other hand, all results on the complexity of subsumption seemed to have assumed that the reduction from  $\preceq_T$  to  $\preceq$  can be done in polynomial time – and indeed, in applications this reduction did not seem to be a source of computational problems, provided some caching is performed [Lipkis, 1982].

Finally, it turned out that there is indeed a “hidden computational cliff.” The minimal terminological language abstract syntax rule

$$D \longrightarrow C \mid D \sqcap D' \mid \forall R: D$$

is closely related to nondeterministic finite state automata and, by this, to regular expressions – provided terminological axioms are permitted.

Suppose two **nondeterministic finite state automata**  $\mathcal{A}_1, \mathcal{A}_2$  with  $\mathcal{A}_i = (\Sigma, Q^i, \delta^i, q_0^i, \mathcal{F}^i)$ , where  $\Sigma$  is the alphabet,  $Q^i$  are the sets of states, where we assume without loss of generality that  $Q^1 \cap Q^2 = \emptyset$ ,  $\delta^i \subseteq Q^i \times (\Sigma \cup \{\epsilon\}) \times Q^i$  are the transition functions,  $q_0^i \in Q^i$  are the initial states, and  $\mathcal{F}^i \subseteq Q^i$  are the sets of accepting states. The **language accepted by these automata** is denoted by  $\mathcal{L}(\mathcal{A}_i)$ . If such automata are *cycle-free*, a cycle-free terminology can be specified such that language inclusion corresponds to subsumption relative to the terminology [Nebel, 1990b]:

Automata	Terminology
$\mathcal{A}_1, \mathcal{A}_2$	$T$
$\Sigma$	$\mathbf{R}$
$(Q^1 \cup Q^2) \uplus \{F\}$	$\mathbf{C}$
$q \in \mathcal{F}^1 \cup \mathcal{F}^2$	$q \doteq \dots \sqcap F \sqcap \dots$
$\epsilon$ -transition from $q$ to $q'$	$q \doteq \dots \sqcap q' \sqcap \dots$
$s$ -transition from $q$ to $q'$	$q \doteq \dots \forall s: q' \sqcap \dots$
$\mathcal{L}(\mathcal{A}_1) \supseteq \mathcal{L}(\mathcal{A}_2)$	$q_0^1 \preceq_T q_0^2$

Since inclusion of languages accepted by cycle-free automata is known to be co-NP-complete [Garey and Johnson, 1979], it follows that  $\preceq_T$  is co-NP-hard.

Interestingly, this correspondence also works the other way around. Given a terminology and two concepts, we can construct two automata such that subsumption coincides with language inclusion, which gives us co-NP-completeness for  $\preceq_T$  in the language considered.

Note that for the proof of this correspondence the set-valued nature of attributes in terminological logics is unessential. The same arguments are

valid for functional attributes, which gives us an interesting corollary in the area of unification grammars. Satisfiability of feature terms relative to a lexicon – which is nothing else than a cycle-free terminology for a feature logic [Nebel and Smolka, 1990] – is also NP-hard, even if satisfiability for the underlying feature logic is polynomial. For instance, supposing  $\psi$ -terms introduced in [Aït-Kaci, 1984], for which satisfiability can be decided in quasi-linear time leads to a NP-complete satisfiability problem if a lexicon is added.

This intractability result does not seem to show up in practical applications very often, however. As a matter of fact, it is not easy to construct a terminology that exhibits exponential time behavior when an efficient algorithm is used that resembles the language inclusion algorithm for finite automata, such as the one described in [Lipkis, 1982]. Nevertheless, it shows us that provable tractability is hardly achievable in the area of attributive description formalisms.

### 3.2 Terminological Cycles

The correspondence between automata and terminologies did not only help to solve the problem concerning the complexity of subsumption relative to a terminology, but also provides a good tool to analyze so-called **terminological cycles**. Such cycles appear when the second restriction on terminologies mentioned above is dropped. In this case, the definition of a concept refers, either directly or indirectly, to the concept itself. Such constructions present problems because neither the right semantics nor the computational properties are obvious.

Based on the correspondence spelled above, Baader [1990] shows that the three possible styles of semantics, namely, *descriptive*, *least fixpoint*, and *greatest fixpoint semantics* [Nebel, 1990a; Nebel, 1991] can be characterized by finite state automata. In particular, the greatest fixed point semantics has an elegant characterization, because it corresponds to automata isomorphic to the terminology.

Besides confirming the conjecture in [Nebel, 1990b] that subsumption becomes PSPACE-complete for least and greatest fixpoint semantics, this characterization also led directly to sound and complete subsumption algorithms for these cases. In addition, this result gave rise to the idea to extend the expressive power of terminological logics by regular expressions over roles [Baader, 1991].

## 4 Thue Systems

For feature logics, the computational complexity was analyzed quite early. The feature logic described in Section 2 without union  $\sqcup$  and complement  $\neg$ , which are essentially the  $\psi$ -terms mentioned above, was shown to have a quasi-linear satisfiability problem [Ait-Kaci, 1984]. The addition of union or complement leads to NP-completeness, as shown in [Kasper, 1987; Johnson, 1987; Smolka, 1988].

The situation in terminological logics was more problematical because of the variety of possible concept- and role-forming operators. As mentioned above, for a long time it remained an open problem whether there are terminological logics such that subsumption is undecidable. The first undecidability result [Schild, 1988] considered a language containing role complements – which do not have practical relevance. Subsequently, Schmidt-Schauß [1989] proved a small subset of KL-ONE to be undecidable using a reduction to the word problem in *invertible Thue systems*. Since this result proved to be quite fruitful for solving other related problems, we will briefly describe the correspondence.

### 4.1 Feature Agreement and Role-Value-Maps

In the presentation of the logical foundations of attributive descriptions, we mentioned already that other terminological logics than  $\mathcal{ALC}$  are conceivable. The reader might have noticed already that **feature-path agreement**  $p \downarrow q$  has no counter-part in the presented terminological logic. As a matter of fact, some terminological logics support such an operator, for instance, KL-ONE and NIKL. Let us consider a subset of those formalisms as specified below:

$$D \longrightarrow C \mid D \sqcap D' \mid \forall R: D \mid (R_{1,1} \dots R_{1,m}) \downarrow (R_{2,1} \dots R_{2,n}),$$

where the denotation of role chains is identical to the denotation of feature chains, i.e., relational composition, and role chains are denoted by  $P$  and  $Q$ . The agreement of such role chains, often called **role-value-map** is defined by:

$$(P \downarrow Q)^I = \{d \in \mathbf{D}^I \mid P^I(d) = Q^I(d)\}.$$

Such a construct could be used, for instance, to define the concept of a father such that all his children have the same surname as the father:

$$\text{Father} \sqcap (\text{surname}) \downarrow (\text{child surname}).$$

Although a very useful construct, it leads unfortunately to undecidability of subsumption. This means, as long as our attributes are functional, subsumption stays decidable (NP-complete for the feature logic considered in



this paper or even quasi-linear for the more restricted  $\psi$ -terms). If we allow for set-valued attributes, subsumption becomes undecidable.

This result follows easily from a reduction to the word problems for a special class of Thue systems. A **Thue system**  $\mathcal{T}$  over an alphabet  $\Sigma$  is a finite set of pairs of words  $u_i, v_i \in \Sigma^*$ :  $\mathcal{T} = \{\{u_i, v_i\}\}$ . Such a Thue system defines a binary relation  $\overset{\mathcal{T}}{\leftrightarrow}$  on  $\Sigma^*$  by:

$$u \overset{\mathcal{T}}{\leftrightarrow} v \iff \exists w_1, w_2 \in \Sigma^* \exists \{u_i, v_i\} \in \mathcal{T}: u = w_1 u_i w_2 \wedge v = w_1 v_i w_2.$$

The symbol  $\overset{\mathcal{T}}{\sim}$  is used to denote the transitive and reflexive closure of  $\overset{\mathcal{T}}{\leftrightarrow}$ . The **word problem** is the problem to decide  $u \overset{\mathcal{T}}{\sim} v$  for given  $\mathcal{T}$  and words  $u, v \in \Sigma^*$ .

An **invertible Thue system** is a Thue system such that for each  $s \in \Sigma$  there exists  $r \in \Sigma$  such that  $sr \overset{\mathcal{T}}{\sim} \epsilon$ , where  $\epsilon$  is the empty word. In other words, the quotient  $\Sigma / \overset{\mathcal{T}}{\sim}$  is a group under concatenation. It is known that there exist invertible Thue systems such that the word problem is undecidable [Boone, 1959].

Undecidability of subsumption in the above mentioned terminological logic can now be shown by using the following correspondence:

Invertible Thue system	Terminological logic
$\Sigma \uplus \{R\}$	<b>R</b>
$\mathcal{T} = \{\{u_i, v_i\}\}$	$D = \prod_{s \in \Sigma} (R \ s) \downarrow (R) \sqcap \prod_i \forall R: (u_i \downarrow v_i)$
$u \overset{\mathcal{T}}{\sim} v$	$D \preceq \forall R: (u \downarrow v)$

## 4.2 Arbitrary Axioms

Since, on one hand, agreements of role-chains are a very useful construction, and on the other hand, they lead to undecidability in case of set-valued attributes, it seems to be a good idea to restrict agreements to chains of functional attributes. Indeed, the terminological logic employed in the CLASSIC system [Borgida *et al.*, 1989; Brachman *et al.*, 1991] is based on this insight. Beside ordinary roles also functional attributes are supported and agreements are only permitted on the latter kind of attributes.

While such a move preserves decidability for the terminological logic [Hollunder and Nutt, 1990], it leads to problems if terminologies containing cycles are allowed. Using a similar reduction as above, Smolka [1989] shows that  $\psi$ -terms plus cyclic terminological axioms result in undecidability of satisfiability of feature terms w.r.t. terminological axioms. This result can be easily reformulated for the corresponding terminological logics, and it turns out that subsumption for descriptive and greatest fixpoint semantics becomes

undecidable [Nebel, 1991]. For this reason, CLASSIC does not support terminological cycles.

Nevertheless, in the CLASSIC system, **implicational rules** are supported. These rules are interpreted procedurally, and they act on a database of objects that are described using concept and role symbols. Given such a rule of the form

$$C(x) \Rightarrow C'(x),$$

any object which the system has been classified to belong to the denotation of the concept  $C$  will be asserted to belong also to the denotation of  $C'$ . If this assertion leads to an inconsistency, i.e., to a situation, where an object is interpreted to belong to the denotation of  $\perp$ , the system signals this contradiction. Although these rules are not identical to axioms, we have the following restriction. A CLASSIC database can be consistently “completed,” i.e., mention explicitly all objects that have to exist because of terminological axioms, only if the database plus the terminology have a model. This, in turn, however, is equivalent to satisfiability of the terminological axioms plus the implicational rules, which is undecidable in the general case by the above result. This means it is undecidable whether a CLASSIC database has a consistent completion.

### 4.3 Functional Uncertainty

Another interesting application of the undecidability of the word problem in Thue systems is a reduction from satisfiability of feature terms that contain **functional uncertainty** [Kaplan and Maxwell, 1988]. This term-forming operator was invented for the concise description of so-called long-distance dependencies in LFG [Kaplan and Bresnan, 1982]. It has the form,

$$\exists(L)F$$

where  $L$  is some finitely represented regular set of words over  $\mathbf{F}$ . It denotes all individuals  $d \in \mathbf{D}^{\mathcal{I}}$  such that there is some feature path  $p \in L$  and an element  $e \in F^{\mathcal{I}}$ , where  $e \in p^{\mathcal{I}}(d)$ . One can think of  $\exists(L)F$  as an infinite union:

$$p_1: F \sqcup p_2: F \sqcup \dots \sqcup p_i: F \sqcup \dots,$$

where all  $p_i$  are elements of  $L$ . Formally, the denotation of functional uncertainty is defined as

$$(\exists(L)F)^{\mathcal{I}} = \{d \in \mathbf{D}^{\mathcal{I}} \mid \exists p \in L: \emptyset \neq p^{\mathcal{I}}(d) \subseteq F^{\mathcal{I}}\}$$

Satisfiability of feature terms containing functional uncertainty has been an open problem. A restricted version of the problem has been addressed in

[Kaplan and Maxwell, 1988], where a partial solution involving an acyclicity condition is given.

Recalling from Section 3.2 the fact that terminological cycles under the greatest fixpoint semantics are closely related to terminological logics that permit regular expressions over roles, one would expect that undecidability would show up again in this case. In fact, if the feature logic specified in Section 2 is extended by functional uncertainty, then satisfiability of feature terms is undecidable [Baader *et al.*, 1991].<sup>2</sup> An even stronger result can be shown. Satisfiability of a feature term relative to a set of arbitrary axioms can be reduced to satisfiability of a feature term without axioms [Baader *et al.*, 1991].<sup>3</sup> However, these results strongly depend on the presence of the complement operator. Thus, decidability for functional uncertainty in weaker feature logics – feature logics that are not propositionally complete – is still an open problem.

## 5 Modal Logics

The most surprising connection between attributive description formalisms and other research areas was recently discovered by Schild [1991]. He showed that a large number of possible terminological logics are notational variants of different propositional modal and dynamic logics. Exploiting this correspondence, a number of interesting properties for the latter logics, such as finite model properties, complexity results, and algorithms, can be straightforwardly applied to the corresponding terminological logics.

In order to demonstrate the connection between the different fields, we will focus on the correspondence between the terminological logic  $\mathcal{ALC}$  [Schmidt-Schauß and Smolka, 1991] introduced in Section 2 and the **propositional polymodal logic**  $K_{(m)}$  [Halpern and Moses, 1985].

Given a set of **atomic propositions**  $\Psi = \{a, b, c, \dots\}$ , the constants  $\top$  and  $\perp$  denoting the truth-values **true** and **false**, a set of  $m$  operators  $K_1, \dots, K_m$ , the set of well-formed  $K_{(m)}$ -formulas (denoted by  $\phi$ ) is defined by

$$\phi \longrightarrow a \mid \top \mid \perp \mid \phi \wedge \phi' \mid \phi \vee \phi' \mid \neg\phi \mid K_i\phi.$$

Satisfiability of such formulas is defined with respect to **Kripke structures**

$$M = (S, \pi, \kappa_1, \dots, \kappa_m),$$

where  $S$  is a set of states,  $\pi(s)$  is a truth-assignment for all atomic propositions in  $\Psi$  at the state  $s \in S$ , and  $\kappa_i \subseteq S \times S$  are the accessibility relations.

<sup>2</sup>Note that no terminological axioms are involved here!

<sup>3</sup>A similar result for terminological logics is shown in [Schild, 1991].

A formula  $\phi$  is said to be **satisfied at a world**  $(M, s)$ , written  $(M, s) \models \phi$ , under the following conditions:

$$\begin{aligned}
(M, s) \models a &\iff \pi(s)(a) = \text{true} \\
(M, s) \models \top & \\
(M, s) \not\models \perp & \\
(M, s) \models \phi \wedge \phi' &\iff (M, s) \models \phi \text{ and } (M, s) \models \phi' \\
(M, s) \models \phi \vee \phi' &\iff (M, s) \models \phi \text{ or } (M, s) \models \phi' \\
(M, s) \models \neg\phi &\iff (M, s) \not\models \phi \\
(M, s) \models K_i\phi &\iff \forall t \in \kappa_i(s): (M, t) \models \phi
\end{aligned}$$

A  $K_{(m)}$ -formula  $\phi$  is **satisfiable**, iff there exists a world that satisfies  $\phi$ .  $\phi$  is **valid**, written  $\models \phi$ , iff all worlds satisfy  $\phi$ .

This notion of satisfiability is obviously closely related to satisfiability of  $\mathcal{ACC}$ -concepts. Indeed, there is a one-to-one correspondence between  $\mathcal{ACC}$  and  $K_{(m)}$ , as can be seen from the following table:

Polymodal logic $K_{(m)}$	Terminological logic $\mathcal{ACC}$
$\Psi$	$\mathbf{C}$
$\{1, \dots, m\}$	$\mathbf{R} = \{R_1, \dots, R_m\}$
$\top$	$\top$
$\perp$	$\perp$
$\phi \wedge \phi'$	$\phi \sqcap \phi'$
$\phi \vee \phi'$	$\phi \sqcup \phi'$
$\neg\phi$	$\neg\phi$
$K_i\phi$	$\forall R_i: \phi$
$\neg K_i\neg\phi$	$\exists R_i: \neg\phi$
$\phi$ satisfiable	$\phi$ is a satisfiable $\mathcal{ACC}$ -concept
$\models \neg\phi \vee \phi'$	$\phi \preceq \phi'$

From that PSPACE-completeness of subsumption in  $\mathcal{ACC}$  follows immediately, because satisfiability in  $K_{(m)}$  is known to be PSPACE-complete [Halpern and Moses, 1985]. Hence, we have an alternative proof of the complexity of subsumption to the one presented in [Schmidt-Schauß and Smolka, 1991]. The most interesting aspect of this close correspondence is that it also works for other variants of propositional modal and dynamic logics [Schild, 1991], giving us a large number of complexity results and algorithms for free. This correspondence also applies to feature logics. In this context, deterministic dynamic logics are the right kind of logics to establish the correspondence. However, although these correspondences can be used to solve a number of open problems, there are aspects which have not been considered in modal and dynamic logics. For instance, agreements of feature paths do not have a counter-part in modal or dynamic logics.

## 6 Conclusion

We demonstrated that the study of formal properties of attributive description formalisms, which jointly characterize terminological and feature logics, is quite closely connected to other areas of formal research. In particular, we showed how the theory of finite state automata helped in solving some open problems in terminological logics, how the word problem for Thue systems was applied to a number of problems to prove undecidability, and, finally, we examined the close correspondence between attributive description formalisms on one side and modal and dynamic logics on the other side.

Interestingly, the study of attributive description formalisms is not only a sink for results in other areas, but also provides insights which can be applied elsewhere. For instance, complex object data models, such as  $O_2$  [Lécluse *et al.*, 1989], are closely related to attributive description formalisms, so that the techniques are applicable. Such an application revealed that the subtype-inference algorithm specified in [Lécluse *et al.*, 1989] is incomplete, and that the subtype-inference problem is PSPACE-complete [Bergamaschi and Nebel, 1990]. Further, the study of sublanguages of  $\mathcal{ALC}$  [Donini *et al.*, 1991b; Donini *et al.*, 1991a] can be directly applied to sublogics of  $K_{(m)}$ . For example, if only negation of propositional atoms is allowed and there is no disjunction, then satisfiability of a  $K_{(m)}$ -formula is co-NP-complete. Finally, the undecidability result for subsumption constraints in feature logics yields the undecidability of semi-unification over rational trees [Dörre and Rounds, 1990].

## Acknowledgement

We would like to thank Werner Nutt for his comments on an earlier version of this paper.

## References

- [Aït-Kaci, 1984] Hassan Aït-Kaci. *A Lattice-Theoretic Approach to Computations Based on a Calculus of Partially Ordered Type Structures*. PhD thesis, University of Pennsylvania, Philadelphia, Pa., 1984.
- [Baader *et al.*, 1991] Franz Baader, Hans-Jürgen Bürckert, Bernhard Nebel, Werner Nutt, and Gert Smolka. On the expressivity of feature logics with negation, functional uncertainty, and sort equations. DFKI Report RR-91-01, German Research Center for Artificial Intelligence (DFKI), Saarbrücken, Germany, January 1991.

- [Baader, 1990] Franz Baader. Terminological cycles in KL-ONE-based knowledge representation languages. In *Proceedings of the 8th National Conference of the American Association for Artificial Intelligence*, pages 621–626, Boston, Mass., 1990.
- [Baader, 1991] Franz Baader. Augmenting concept languages by transitive closure of roles: An alternative to terminological cycles. In *Proceedings of the 12th International Joint Conference on Artificial Intelligence*, Sydney, Australia, 1991. To appear.
- [Bergamaschi and Nebel, 1990] Sonia Bergamaschi and Bernhard Nebel. Theoretical foundations of complex object data models. Technical Report 74, CIOC-CNR, Bologna, Italy, December 1990.
- [Boone, 1959] W. W. Boone. The word problem. *Ann. of Math.*, 70(2):207–265, 1959.
- [Borgida *et al.*, 1989] Alexander Borgida, Ronald J. Brachman, Deborah L. McGuinness, and Lori Alperin Resnick. CLASSIC: a structural data model for objects. In *Proceedings of the 1989 ACM SIGMOD International Conference on Management of Data*, pages 59–67, Portland, Oreg., June 1989.
- [Bouma *et al.*, 1988] Gosse Bouma, Esther König, and Hans Uszkoreit. A flexible graph-unification formalism and its application to natural-language processing. *IBM Journal of Research and Development*, 32(2):170–184, March 1988.
- [Brachman and Levesque, 1984] Ronald J. Brachman and Hector J. Levesque. The tractability of subsumption in frame-based description languages. In *Proceedings of the 4th National Conference of the American Association for Artificial Intelligence*, pages 34–37, Austin, Tex., August 1984.
- [Brachman and Schmolze, 1985] Ronald J. Brachman and James G. Schmolze. An overview of the KL-ONE knowledge representation system. *Cognitive Science*, 9(2):171–216, April 1985.
- [Brachman *et al.*, 1991] Ronald J. Brachman, Deborah L. McGuinness, Peter F. Patel-Schneider, Lori Alperin Resnick, and Alexander Borgida. Living with CLASSIC: When and how to use a KL-ONE-like language. In J. Sowa, editor, *Principles of Semantic Networks*. Morgan Kaufmann, San Mateo, Cal., 1991.

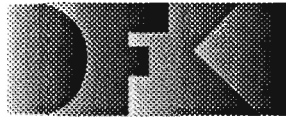
- [Brachman, 1979] Ronald J. Brachman. On the epistemological status of semantic networks. In N. V. Findler, editor, *Associative Networks: Representation and Use of Knowledge by Computers*, pages 3–50. Academic Press, New York, N.Y., 1979.
- [Donini *et al.*, 1991a] Francesco M. Donini, Maurizio Lenzerini, Daniele Nardi, and Werner Nutt. Tractable concept languages. In *Proceedings of the 12th International Joint Conference on Artificial Intelligence*, Sydney, Australia, August 1991. To appear.
- [Donini *et al.*, 1991b] Francesco M. Donini, Maurizio Lenzerini, Daniele Nardi, and Werner Nutt. The complexity of concept languages. In J. A. Allen, R. Fikes, and E. Sandewall, editors, *Principles of Knowledge Representation and Reasoning: Proceedings of the 2nd International Conference*, Cambridge, Mass., April 1991. To appear.
- [Dörre and Rounds, 1990] Jochen Dörre and William C. Rounds. On subsumption and semi-unification in feature algebras. In *Proceedings of the 5th IEEE Symposium on Logic in Computer Science*, pages 301–310, Philadelphia, Pa., 1990. Also available as IWBS Report 97, IBM Germany Scientific Center, IWBS, Stuttgart, Germany, December 1989.
- [Dörre and Seiffert, 1991] Jochen Dörre and Roland Seiffert. STUF – formalism and workbench. In O. Herzog and C. Rollinger, editors, *Textunderstanding in LILOG: Integrating Computational Linguistics and Artificial Intelligence*. Springer-Verlag, Berlin, Germany, 1991. To appear.
- [Garey and Johnson, 1979] Michael R. Garey and David S. Johnson. *Computers and Intractability—A Guide to the Theory of NP-Completeness*. Freeman, San Francisco, Cal., 1979.
- [Halpern and Moses, 1985] Josef Y. Halpern and Yoram Moses. A guide to the modal logics of knowledge and belief: Preliminary draft. In *Proceedings of the 9th International Joint Conference on Artificial Intelligence*, pages 480–490, Los Angeles, Cal., August 1985.
- [Hollunder and Nutt, 1990] Bernhard Hollunder and Werner Nutt. Subsumption algorithms for concept languages. DFKI Report RR-90-04, German Research Center for Artificial Intelligence (DFKI), Kaiserslautern, Germany, 1990.
- [Johnson, 1987] Mark Johnson. *Attribute-Value Logic and the Theory of Grammar*. CSLI Lecture Notes 16. Center for the Study of Language and Information, Stanford University, Stanford, Cal., 1987.

- [Kaplan and Bresnan, 1982] Ronald M. Kaplan and Joan Bresnan. Lexical-Functional Grammar: A formal system for grammatical representation. In J. Bresnan, editor, *The Mental Representation of Grammatical Relations*, pages 173–381. MIT Press, Cambridge, Mass., 1982.
- [Kaplan and Maxwell, 1988] Ronald M. Kaplan and John T. Maxwell, III. An algorithm for functional uncertainty. In *Proceedings of the 12th International Conference on Computational Linguistics*, pages 297–302, Budapest, Hungary, 1988.
- [Kasper and Rounds, 1986] Robert T. Kasper and William C. Rounds. A logical semantics for feature structures. In *Proceedings of the 24th Annual Meeting of the ACL*, pages 257–265, New York, N.Y., 1986.
- [Kasper, 1987] Robert T. Kasper. *Feature Structures: A Logical Theory with Applications to Language Analysis*. PhD thesis, University of Michigan, Ann Arbor, Mich., 1987.
- [Kay, 1979] Martin Kay. Functional grammar. In *Proceedings of the Fifth Annual Meeting of the Berkeley Linguistics Society*, Berkeley, Cal., 1979. Berkeley Linguistics Society.
- [Kay, 1985] Martin Kay. Parsing in functional unification grammars. In D. Dowty and L. Karttunen, editors, *Natural Language Parsing*. Cambridge University Press, Cambridge, England, 1985.
- [Lécluse *et al.*, 1989] Christophe Lécluse, Philippe Richard, and Fernando Velez. Modeling complex structures in object-oriented databases. In *Proceedings of the 8th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database-Systems*, pages 360–367, March 1989.
- [Lipkis, 1982] Thomas Lipkis. A KL-ONE classifier. In J. G. Schmolze and R. J. Brachman, editors, *Proceedings of the 1981 KL-ONE Workshop*, pages 128–145, Cambridge, Mass., 1982. The proceedings have been published as BBN Report No. 4842 and Fairchild Technical Report No. 618.
- [Moser, 1983] M. G. Moser. An overview of NIKL, the new implementation of KL-ONE. In *Research in Knowledge Representation and Natural Language Understanding*, BBN Report No. 5421, pages 7–26. Bolt, Beranek, and Newman Inc., Cambridge, Mass., 1983.
- [Nebel and Smolka, 1990] Bernhard Nebel and Gert Smolka. Representation and reasoning with attributive descriptions. In K.-H. Bläsius,



- U. Hedtstück, and C.-R. Rollinger, editors, *Sorts and Types in Artificial Intelligence*, pages 112–139. Springer-Verlag, Berlin, Germany, 1990. Also available as IWBS Report 81, IBM Germany Scientific Center, IWBS, Stuttgart, Germany, September 1989.
- [Nebel, 1988] Bernhard Nebel. Computational complexity of terminological reasoning in BACK. *Artificial Intelligence*, 34(3):371–383, April 1988.
- [Nebel, 1990a] Bernhard Nebel. *Reasoning and Revision in Hybrid Representation Systems*, volume 422 of *Lecture Notes in Computer Science*. Springer-Verlag, Berlin, Germany, 1990.
- [Nebel, 1990b] Bernhard Nebel. Terminological reasoning is inherently intractable. *Artificial Intelligence*, 43:235–249, 1990. Also available as IWBS Report 82, IBM Germany Scientific Center, IWBS, Stuttgart, Germany, October 1989.
- [Nebel, 1991] Bernhard Nebel. Terminological cycles: Semantics and computational properties. In J. Sowa, editor, *Principles of Semantic Networks*. Morgan Kaufmann, San Mateo, Cal., 1991. To appear. A preliminary version of this paper appears in the Preprints of the Workshop on *Formal Aspects of Semantic Networks*, Catalina, Cal., 1989.
- [Patel-Schneider, 1989a] Peter F. Patel-Schneider. A four-valued semantics for terminological logics. *Artificial Intelligence*, 38(3):319–351, April 1989.
- [Patel-Schneider, 1989b] Peter F. Patel-Schneider. Undecidability of subsumption in NIKL. *Artificial Intelligence*, 39(2):263–272, June 1989.
- [Pletat and von Luck, 1989] Udo Pletat and Kai von Luck. Knowledge Representation in LILOG. In Karl-Hans Bläsius, Uli Hedtstück, and Claus Rollinger, editors, *Sorts and Types in Artificial Intelligence*, pages 140–164. Springer-Verlag, Berlin, Germany, 1989.
- [Pletat, 1991] Udo Pletat. The knowledge representation language L-LILOG. In O. Herzog and C. Rollinger, editors, *Textunderstanding in LILOG: Integrating Computational Linguistics and Artificial Intelligence*. Springer-Verlag, Berlin, Germany, 1991. To appear.
- [Schild, 1988] Klaus Schild. Undecidability of  $\mathcal{U}$ . KIT Report 67, Department of Computer Science, Technische Universität Berlin, Berlin, Germany, October 1988.
- [Schild, 1991] Klaus Schild. A correspondence theory for terminological logics: Preliminary report. In *Proceedings of the 12th International Joint Conference on Artificial Intelligence*, Sydney, Australia, 1991. To appear.

- [Schmidt-Schauß and Smolka, 1991] Manfred Schmidt-Schauß and Gert Smolka. Attributive concept descriptions with complements. *Artificial Intelligence*, 48:1–26, 1991. Also available as IWBS Report 68, IBM Germany, Scientific Center, IWBS, Stuttgart, Germany, June 1989.
- [Schmidt-Schauß, 1989] Manfred Schmidt-Schauß. Subsumption in KL-ONE is undecidable. In R. J. Brachman, H. J. Levesque, and R. Reiter, editors, *Principles of Knowledge Representation and Reasoning: Proceedings of the 1st International Conference*, pages 421–431, Toronto, Ont., May 1989.
- [Schmolze and Israel, 1983] James G. Schmolze and David J. Israel. KL-ONE: Semantics and classification. In *Research in Knowledge Representation and Natural Language Understanding*, BBN Technical Report, No. 5421, pages 27–39. Bolt, Beranek, and Newman Inc., Cambridge, Mass., 1983.
- [Shieber, 1986] Stuart M. Shieber. *An Introduction to Unification-Based Approaches to Grammar*. CSLI Lecture Notes 4. Center for the Study of Language and Information, Stanford University, Stanford, Cal., 1986.
- [Smolka, 1988] Gert Smolka. A feature logic with subsorts. LILOG Report 33, IWBS, IBM Germany Scientific Center, Stuttgart, Germany, May 1988. To appear in: J. Wedekind and C. Rohrer (eds.); *Unification in Grammar*; The MIT Press, 1991.
- [Smolka, 1989] Gert Smolka. Feature constraint logics for unification grammars. IWBS Report 93, IWBS, IBM Germany Scientific Center, Stuttgart, Germany, November 1989. To appear in the *Journal of Logic Programming*.



Deutsches  
Forschungszentrum  
für Künstliche  
Intelligenz GmbH

DFKI  
-Bibliothek-  
PF 2080  
6750 Kaiserslautern  
FRG

## DFKI Publikationen

Die folgenden DFKI Veröffentlichungen oder die aktuelle Liste von erhältlichen Publikationen können bezogen werden von der oben angegebenen Adresse.

## DFKI Publications

The following DFKI publications or the list of currently available publications can be ordered from the above address.

---

### DFKI Research Reports

#### RR-90-01

*Franz Baader*: Terminological Cycles in KL-ONE-based Knowledge Representation Languages  
33 pages

#### RR-90-02

*Hans-Jürgen Bürckert*: A Resolution Principle for Clauses with Constraints  
25 pages

#### RR-90-03

*Andreas Dengel, Nelson M. Mattos*: Integration of Document Representation, Processing and Management  
18 pages

#### RR-90-04

*Bernhard Hollunder, Werner Nutt*: Subsumption Algorithms for Concept Languages  
34 pages

#### RR-90-05

*Franz Baader*: A Formal Definition for the Expressive Power of Knowledge Representation Languages  
22 pages

#### RR-90-06

*Bernhard Hollunder*: Hybrid Inferences in KL-ONE-based Knowledge Representation Systems  
21 pages

#### RR-90-07

*Elisabeth André, Thomas Rist*: Wissensbasierte Informationspräsentation:  
Zwei Beiträge zum Fachgespräch Graphik und KI:

1. Ein planbasierter Ansatz zur Synthese illustrierter Dokumente
2. Wissensbasierte Perspektivenwahl für die automatische Erzeugung von 3D-Objektdarstellungen

24 pages

#### RR-90-08

*Andreas Dengel*: A Step Towards Understanding Paper Documents  
25 pages

#### RR-90-09

*Susanne Biundo*: Plan Generation Using a Method of Deductive Program Synthesis  
17 pages

#### RR-90-10

*Franz Baader, Hans-Jürgen Bürckert, Bernhard Hollunder, Werner Nutt, Jörg H. Siekmann*: Concept Logics  
26 pages

#### RR-90-11

*Elisabeth André, Thomas Rist*: Towards a Plan-Based Synthesis of Illustrated Documents  
14 pages

#### RR-90-12

*Harold Boley*: Declarative Operations on Nets  
43 pages

#### RR-90-13

*Franz Baader*: Augmenting Concept Languages by Transitive Closure of Roles: An Alternative to Terminological Cycles  
40 pages

#### RR-90-14

*Franz Schmalhofer, Otto Kühn, Gabriele Schmidt*: Integrated Knowledge Acquisition from Text, Previously Solved Cases, and Expert Memories  
20 pages

#### RR-90-15

*Harald Trost*: The Application of Two-level Morphology to Non-concatenative German Morphology  
13 pages

**RR-90-16**

*Franz Baader, Werner Nutt: Adding Homomorphisms to Commutative/Monoidal Theories, or: How Algebra Can Help in Equational Unification*  
25 pages

**RR-91-01**

*Franz Baader, Hans-Jürgen Bürckert, Bernhard Nebel, Werner Nutt, and Gert Smolka: On the Expressivity of Feature Logics with Negation, Functional Uncertainty, and Sort Equations*  
20 pages

**RR-91-02**

*Francesco Donini, Bernhard Hollunder, Maurizio Lenzerini, Alberto Marchetti Spaccamela, Daniele Nardi, Werner Nutt: The Complexity of Existential Quantification in Concept Languages*  
22 pages

**RR-91-03**

*B.Hollunder, Franz Baader: Qualifying Number Restrictions in Concept Languages*  
34 pages

**RR-91-04**

*Harald Trost: X2MORF: A Morphological Component Based on Augmented Two-Level Morphology*  
19 pages

**RR-91-05**

*Wolfgang Wahlster, Elisabeth André, Winfried Graf, Thomas Rist: Designing Illustrated Texts: How Language Production is Influenced by Graphics Generation.*  
17 pages

**RR-91-06**

*Elisabeth André, Thomas Rist: Synthesizing Illustrated Documents  
A Plan-Based Approach*  
11 pages

**RR-91-07**

*Günter Neumann, Wolfgang Finkler: A Head-Driven Approach to Incremental and Parallel Generation of Syntactic Structures*  
13 pages

**RR-91-08**

*Wolfgang Wahlster, Elisabeth André, Som Bandyopadhyay, Winfried Graf, Thomas Rist: WIP: The Coordinated Generation of Multimodal Presentations from a Common Representation*  
23 pages

**RR-91-09**

*Hans-Jürgen Bürckert, Jürgen Müller, Achim Schupeta: RATMAN and its Relation to Other Multi-Agent Testbeds*  
31 pages

**RR-91-10**

*Franz Baader, Philipp Hanschke: A Scheme for Integrating Concrete Domains into Concept Languages*  
31 pages

**RR-91-11**

*Bernhard Nebel: Belief Revision and Default Reasoning: Syntax-Based Approaches*  
37 pages

**RR-91-13**

*Gert Smolka: Residuation and Guarded Rules for Constraint Logic Programming*  
17 pages

**RR-91-15**

*Bernhard Nebel, Gert Smolka: Attributive Description Formalisms ... and the Rest of the World*  
20 pages

**RR-91-16**

*Stephan Busemann: Using Pattern-Action Rules for the Generation of GPSG Structures from Separate Semantic Representations*  
18 pages

---

**DFKI Technical Memos**
**TM-89-01**

*Susan Holbach-Weber: Connectionist Models and Figurative Speech*  
27 pages

**TM-90-01**

*Som Bandyopadhyay: Towards an Understanding of Coherence in Multimodal Discourse*  
18 pages

**TM-90-02**

*Jay C. Weber: The Myth of Domain-Independent Persistence*  
18 pages

**TM-90-03**

*Franz Baader, Bernhard Hollunder: KRIS: Knowledge Representation and Inference System -System Description-*  
15 pages

**TM-90-04**

*Franz Baader, Hans-Jürgen Bürckert, Jochen Heinsohn, Bernhard Hollunder, Jürgen Müller, Bernhard Nebel, Werner Nutt, Hans-Jürgen Profillich: Terminological Knowledge*  
Representation: A Proposal for a Terminological Logic  
7 pages

**TM-91-01**

*Jana Köhler*  
Approaches to the Reuse of Plan Schemata in Planning Formalisms  
52 pages

**TM-91-02**

*Knut Hinkelmann*  
Bidirectional Reasoning of Horn Clause Programs: Transformation and Compilation  
20 pages

**TM-91-03**

*Otto Kühn, Marc Linster, Gabriele Schmidt*  
Clamping, COKAM, KADS, and OMOS:  
The Construction and Operationalization  
of a KADS Conceptual Model  
20 pages

**TM-91-04**

*Harold Boley*  
A sampler of Relational/Functional Definitions  
12 pages

**TM-91-05**

*Jay C. Weber, Andreas Dengel and Rainer Bleisinger*  
Theoretical Consideration of Goal Recognition  
Aspects for Understanding Information in Business  
Letters  
10 pages

**D-90-03**

*Ansgar Bernardi, Christoph Klauck, Ralf Legleitner: Abschlußbericht des Arbeitspaketes PROD*  
36 Seiten

**D-90-04**

*Ansgar Bernardi, Christoph Klauck, Ralf Legleitner: STEP: Überblick über eine zukünftige Schnittstelle zum Produktdatenaustausch*  
69 Seiten

**D-90-05**

*Ansgar Bernardi, Christoph Klauck, Ralf Legleitner: Formalismus zur Repräsentation von Geo-metrie- und Technologieinformationen als Teil eines Wissensbasierten Produktmodells*  
66 Seiten

**D-90-06**

*Andreas Becker: The Window Tool Kit*  
66 Seiten

**D-91-01**

*Werner Stein, Michael Sintek*  
Relfun/X - An Experimental Prolog  
Implementation of Relfun  
48 pages

**D-91-03**

*Harold Boley, Klaus Elsbernd, Hans-Günther Hein, Thomas Krause*  
RFM Manual: Compiling RELFUN into the  
Relational/Functional Machine  
43 pages

**D-91-04**

DFKI Wissenschaftlich-Technischer Jahresbericht  
1990  
93 Seiten

---

**DFKI Documents****D-89-01**

*Michael H. Malburg, Rainer Bleisinger:*  
HYPERBIS: ein betriebliches Hypermedia-  
Informationssystem  
43 Seiten

**D-90-01**

DFKI Wissenschaftlich-Technischer Jahresbericht  
1989  
45 pages

**D-90-02**

*Georg Seul: Logisches Programmieren mit Feature-  
-Typen*  
107 Seiten

**Attributive Description Formalisms  
and the Rest of the World**

Bernhard Nebel and Gert Smolka

**RR-91-15**  
Research Report