Description and/or Hybrid Logics

Carlos Areces Maarten de Rijke

ILLC, University of Amsterdam, Pl. Muidergracht 24,
1018 TV Amsterdam, The Netherlands. E-mail: {carlos, mdr}@wins.uva.nl

Abstract: Improving on work by Schild, De Giacomo and Lenzerini, we establish a tight connection between description logics and hybrid logics, and use this to transfer results on complexity and expressive power from one to the other.

Keywords: modal and description logic, hybrid logic, T-Box and A-Box reasoning, computational complexity, expressive power.

1 Introduction

Nearly a decade ago, Schild [23] observed and exploited the close correspondence between description logics and modal languages. He used it as a bridge to transfer complexity results and axiomatizations from modal logics to description logics but noticed that the correspondence can only be established at the level of *concept satisfiability*. Basic modal logic is not expressive enough to account for either A-Box reasoning or inference in the presence of definitions (non-empty T-Boxes). Also, some very expressive description languages include constructions for building complex roles such as intersection, converse, and even transitive closure. By lifting the correspondence to Converse Propositional Dynamic Logic (CPDL) [9], Schild accounted for these constructions and for inference from non-empty T-Boxes. De Giacomo and Lenzerini [8] extended these results by encoding A-Box reasoning in CPDL. While embeddings of description logics into CPDL have proved useful, they have two important disadvantages. Complexity-wise, the local satisfiability problem of CPDL (i.e., the problem of finding, given a CPDL formula ϕ, a model $\mathcal{M}$ and a state m such that $\mathcal{M}, m \Vdash \phi$) is already EXPTIME-complete, and this prohibits sharp complexity results. Moreover, with respect to expressive power, the model theory of CPDL is complex, because the Kleene star (and hence a weak notion of induction) needs to be taken into consideration.

In this paper, we replace CPDL by hybrid languages and in this way improve on the issues above. As we will see below, the connection between description and hybrid logics is very tight. The main aim of the paper is to establish this connection, using the description logic $\mathcal{ALC}$ as our starting point, and to give an impression of its benefits in terms of results on complexity, expressive power, and meta-logical properties like interpolation and Beth definability.

We start by providing some background. We then recall relevant work by Schild, and De Giacomo and Lenzerini. After that we set up the link between hybrid and description logics, and exploit it.

§**1.1 Description Logic.** Description logics (DLs) are a family of formal languages with a clearly specified semantics, usually in terms of first-order models, together with inference mechanisms to account for knowledge classification. One of the main aims is to identify fragments of first-order logic that are able to capture the features needed for representing a particular problem domain, and which still admit efficient reasoning algorithms.

Constructor	Syntax	Semantics
concept name	C	$C^{\mathcal{I}}$
top	$\top$	$\Delta^{\mathcal{I}}$
negation ($\mathcal{C}$)	$\neg C$	$\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$
conjunction	$C_1 \sqcap C_2$	$C_1^{\mathcal{I}} \cap C_2^{\mathcal{I}}$
disjunction ($\mathcal{U}$)	$C_1 \sqcup C_2$	$C_1^{\mathcal{I}} \cup C_2^{\mathcal{I}}$
universal quant.	$\forall R.C$	$\{d_1 \mid \forall d_2 \in \Delta^{\mathcal{I}}.(R^{\mathcal{I}}(d_1, d_2) \rightarrow d_2 \in C^{\mathcal{I}})\}$
existential quant. ($\mathcal{E}$)	$\exists R.C$	$\{d_1 \mid \exists d_2 \in \Delta^{\mathcal{I}}.(R^{\mathcal{I}}(d_1, d_2) \wedge d_2 \in C^{\mathcal{I}})\}$
one-of ($\mathcal{O}$)	$\{a_1, \ldots, a_n\}$	$\{d \mid d = a_i^{\mathcal{I}}$ for some $a_i\}$
role filler ($\mathcal{B}$)	$\exists R.\{a\}$	$\{d \mid R^{\mathcal{I}}(d, a^{\mathcal{I}})\}$
role name	R	$R^{\mathcal{I}}$
role conjunction ($\mathcal{R}$)	$R_1 \sqcap R_2$	$R_1^{\mathcal{I}} \cap R_2^{\mathcal{I}}$
inverse roles ($\mathcal{I}$)	R^{-1}	$\{(d_1, d_2) \mid R^{\mathcal{I}}(d_2, d_1)\}$

Table 1: Common operators of description logics.

Let $\mathsf{CON} = \{C_1, C_2, \ldots\}$ be a countable set of *atomic concepts*, $\mathsf{ROL} = \{R_1, R_2, \ldots\}$ a countable set of *atomic roles*, and $\mathsf{IND} = \{a_1, a_2, \ldots\}$ a countable set of *individuals*. For CON, ROL, IND, all pairwise disjoint, $\mathcal{S} = \langle \mathsf{CON}, \mathsf{ROL}, \mathsf{IND} \rangle$ is a *signature*. An *interpretation* $\mathcal{I}$ for $\mathcal{S}$ is a tuple $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$, where $\Delta^{\mathcal{I}}$ is a non-empty set, and $\cdot^{\mathcal{I}}$ assigns elements $a_i^{\mathcal{I}} \in \Delta^{\mathcal{I}}$ to constants a_i, subsets $C_i^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ to atomic concepts C_i, and relations $R_i^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ to atomic roles R_i. The atomic symbols in a DL signature can be combined by means of *concept* and *role constructors*, to form complex expressions. Table 1 defines the constructors for the DLs we will discuss, together with their semantics. It is customary to define systems by postfixing the names of some basic description languages like $\mathcal{AL}$ or $\mathcal{FL}$ with the names of the added operators from Table 1. In this paper, we will be interested in languages having full Boolean expressivity and hence focus on $\mathcal{ALC}$ and its extensions.

In DLs we want to perform inferences given certain background knowledge. Let $\mathcal{L}$ be any description logic, a *knowledge base* Σ in $\mathcal{L}$ is a pair $\Sigma = \langle T, A \rangle$ such that T is the T(erminological)-Box: a finite, possibly empty, set of expressions of the form $C_1 \sqsubseteq C_2$, where C_1, $C_2 \in \mathsf{CON}(\mathcal{L})$ ($C_1 \doteq C_2$ is short for $C_1 \sqsubseteq C_2$ and $C_2 \sqsubseteq C_1$). Formulas in T are called *terminological axioms*. In addition, A is the A(ssertional)-Box: a finite, possibly empty, set of expressions of the forms $a : C$ or $(a, b) : R$ where C is in $\mathsf{CON}(\mathcal{L})$, R is in $\mathsf{ROL}(\mathcal{L})$ and a, b are individuals. Formulas in A are called *assertions*. Our definitions of terminological axioms and assertions are amongst the most general in the literature (and we will generalize them even further below).

Let $\mathcal{I}$ be an interpretation and ϕ a terminological axiom or assertion. Then $\mathcal{I}$ *models* ϕ (notation, $\mathcal{I} \models \phi$) if $\phi = C_1 \sqsubseteq C_2$ and $C_1^{\mathcal{I}} \subseteq C_2{}^{\mathcal{I}}$, or $\phi = a : C$ and $a^{\mathcal{I}} \in C^{\mathcal{I}}$, or $\phi = (a, b) : R$ and $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$. If $\Sigma = \langle T, A \rangle$ is a knowledge base and $\mathcal{I}$ an interpretation, then $\mathcal{I}$ models Σ (notation: $\mathcal{I} \models \Sigma$) if for all $\phi \in T \cup A, \mathcal{I} \models \phi$. Given a knowledge base Σ and a terminological axiom or assertion ϕ, we write $\Sigma \models \phi$ if for all models $\mathcal{I}$ of Σ we have $\mathcal{I} \models \phi$. All standard description logic reasoning tasks (like subsumption or instance checking) can be defined in terms of this relation.

§**1.2 Hybrid Logic.** Modal formulas are evaluated at a given *state* in a model, and their truth values depend on the value of formulas at some related *states*. Yet, nothing in modal syntax gets to grips with the *states* themselves. Hybrid languages are modal languages which solve this "reference problem" by introducing special symbols, called *nominals*, to explicitly name the states in a model.

The basic hybrid language is $\mathcal{H}_{\mathsf{N}}$, basic modal logic extended with nominals. Further extensions are named by listing the added operators. The most expressive system we will discuss is $\mathcal{H}_{\mathsf{N}}(\langle R^{-1} \rangle, \mathsf{E}, @)$, the basic hybrid language extended with the converse (past) and existential modalities, and the @ operator. More precisely, let $\mathsf{REL} = \{R_1, R_2, \ldots\}$ be a countable set of *relation* *symbols*, $\mathsf{PROP} = \{p_1, p_2, \ldots\}$ a countable set of *proposition letters*, and $\mathsf{NOM} = \{i_1, i_2, \ldots\}$ a countable set of *nominals*. $\mathsf{ATOM} = \mathsf{PROP} \cup \mathsf{NOM}$ is the set of *atoms*. The formulas of the hybrid language $\mathcal{H}_{\mathsf{N}}(\langle R^{-1}\rangle, \mathsf{E}, @)$ in the signature $\langle \mathsf{REL}, \mathsf{PROP}, \mathsf{NOM}\rangle$ are

$$\mathsf{FORMS} := \top \mid a \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid \langle R\rangle\phi \mid \langle R^{-1}\rangle\phi \mid \mathsf{E}\phi \mid @_i\phi,$$

where $a \in \mathsf{ATOM}$, $R \in \mathsf{REL}$, $i \in \mathsf{NOM}$, and $\phi, \phi_1, \phi_2 \in \mathsf{FORMS}$.

A *hybrid model* $\mathcal{M}$ is a triple $\mathcal{M} = \langle M, \{R_i\}, V\rangle$ where M is a non-empty set, $\{R_i\}$ is a set of binary relations on M, and $V : \mathsf{PROP} \cup \mathsf{NOM} \to Pow(M)$ is such that for all nominals $i \in \mathsf{NOM}$, $V(i)$ is a singleton subset of M. Let $\mathcal{M} = \langle M, \{R_i\}, V\rangle$ be a model and $m \in M$. The interesting cases of the *satisfiability relation* are as follows: $\mathcal{M}, m \Vdash a$ iff $m \in V(a)$, $a \in \mathsf{ATOM}$; $\mathcal{M}, m \Vdash \langle R\rangle\phi$ iff $\exists m' (R(m, m')$ & $\mathcal{M}, m' \Vdash \phi)$; $\mathcal{M}, m \Vdash \langle R^{-1}\rangle\phi$ iff $\exists m' (R(m', m)$ & $\mathcal{M}, m' \Vdash \phi)$; $\mathcal{M}, m \Vdash \mathsf{E}\phi$ iff $\exists m' (\mathcal{M}, m' \Vdash \phi)$; and $\mathcal{M}, m \Vdash @_i\phi$ iff $\mathcal{M}, m' \Vdash \phi$, where $V(i) = \{m'\}$, $i \in \mathsf{NOM}$.

We write $\mathcal{M} \Vdash \phi$ iff for all $m \in M$, $\mathcal{M}, m \Vdash \phi$. This notion extends to sets of formulas in the standard way. A formula ϕ is *satisfiable* if there is a model $\mathcal{M}$ and a world $m \in M$ with $\mathcal{M}, m \Vdash \phi$. A formula ϕ is *valid* if for all models $\mathcal{M}$, $\mathcal{M} \Vdash \phi$. ϕ is a *local consequence* of a set of formulas T (notation, $T \models^{loc} \phi$), if for all models $\mathcal{M}$ and points $m \in M$, $\mathcal{M}, m \Vdash T$ implies $\mathcal{M}, m \Vdash \phi$; ϕ is a *global consequence* of a set of formulas T (notation, $T \models^{glo} \phi$), if for all models $\mathcal{M}$, $\mathcal{M} \Vdash T$ implies $\mathcal{M} \Vdash \phi$. When T is the empty set, we have $\{\} \models^{glo} \phi$ iff $\{\} \models^{loc} \phi$, and simply write $\models \phi$.

2 Schild's Terminologies

It is straightforward to map concepts in $\mathcal{ALC}$ into formulas of CPDL, while preserving satisfiability — actually, basic poly-modal logic is enough. Just define the translation $\cdot^t$ by putting $(C_i)^t = p_i$, for C_i an atomic concept; $(\neg C)^t = \neg(C^t)$; $(C \sqcap D)^t = C^t \wedge D^t$; and $(\exists R.C)^t = \langle R\rangle C^t$. It is clear that $\cdot^t$ preserves satisfiability. But we need further expressive power to account for T-Box and A-Box reasoning. The standard notion of bisimulation [6] helps us prove this claim. Consider the signature $\mathcal{S} = \langle\{C_1, C_2\}, \{R\}, \{a\}\rangle$ and the interpretations $\mathcal{I}_1 = \langle\{m_1, m_2\}, \cdot^{\mathcal{I}_1}\rangle$ and $\mathcal{I}_2 = \langle\{m_3, m_4, m_5\}, \cdot^{\mathcal{I}_2}\rangle$ where $C_1^{\mathcal{I}_1} = \{m_1\}$, $C_2^{\mathcal{I}_1} = \{m_1, m_2\}$, $R^{\mathcal{I}_1} = \{\}$, $a^{\mathcal{I}_1} = m_1$; and $C_1^{\mathcal{I}_2}\{m_4\}$, $C_2^{\mathcal{I}_2} = \{m_3\}$, $R^{\mathcal{I}_2} = \{\}$, $a^{\mathcal{I}_2} = m_5$.

Clearly, $\mathcal{I}_1$ models both $C_1 \sqsubseteq C_2$ and $a{:}C_1$ while $\mathcal{I}_2$ models neither. On the other hand, when we consider $\mathcal{I}_1$ and $\mathcal{I}_2$ as modal models, the relation $\{(m_2, m_3)\}$ is a bisimulation. But we should take care: $C_1 \sqsubseteq C_2$ and $a{:}C_1$ are *global* notions, they are true of an element of a model if and only if they are true of all elements. On the other hand, basic modal formulas are *local*, the point of evaluation is relevant for their truth. Let's go through our argument taking special care of this issue. If a modal formula ϕ is equivalent to $C_1 \sqsubseteq C_2$ then it would also behave globally, and $C_1 \sqsubseteq C_2$ being true of $\mathcal{I}_1$ would imply ϕ being true of m_2. By bisimulation ϕ would also be true of m_3 and by "global behavior" of $\mathcal{I}_2$. But it isn't. We can give a similar argument for $a{:}C_1$. This switch between a local and a global perspective is one of the main differences between modal and description languages. Because of this, we have incorporated the existential modality E in our hybrid languages. Let A be the dual of E, i.e., $\mathsf{A}\psi \leftrightarrow \neg\mathsf{E}\neg\psi$, then $\mathcal{M} \Vdash \phi$ iff $\mathcal{M}, m \Vdash \neg\mathsf{E}\neg\phi$ for some $m \in M$. In other words, E lets us talk about globality from a local perspective.

Instead of using E, Schild [23] accounts for terminological axioms by using the collapsed model property of CPDL (any satisfiable CPDL formula is satisfiable in a connected model) and the availability of the Kleene star. Due to the former, we can ignore states which are not reachable by a finite sequence of backwards and forwards transitions. Thanks to the Kleene star we can "step over" all these transitions in one step. Formally, extend $\cdot^t$ by putting $(C \sqsubseteq D)^t = (C^t \to D^t)$. And for a finite set of terminological axioms T, let T^t be $\bigwedge \phi_i^t$, where $\phi_i \in T$. Now, let $T \cup \{\phi\}$ be a finite set of terminological axioms, and let $R_1, \ldots, R_n$ be all the roles in $T \cup \{\phi\}$. Then $\langle T, \{\}\rangle \models \phi$ iff $\models [(R_1 \cup R_1^{-1} \cup \cdots \cup R_n \cup R_n^{-1})^*]T^t \rightarrow \phi^t$.

As Schild remarks, this translation would not work for an infinite T. On the one hand, T might contain an infinite number of roles, but even in the case of a finite signature, PDL is not compact (see [12, Theorem 2.15]), hence inference from infinite sets does not coincide with inference in terms of finite sets. But more importantly, lack of compactness has a striking effect on the complexity of the consequence problem, which becomes highly undecidable, an indication that PDL is not computationally well behaved. The computational problems caused by the Kleene star have been well investigated both in the modal and description logic community [17, 11, 22, 13]; and authors like Sattler, and Horrocks and Gough have argued that in many cases the ability to define a role as transitive is all what is needed in applications, instead of the full power of transitive closure.

It pays off to look carefully at the local vs. global issue. To fully appreciate the subtleties here, we will do so in the following section.

3 Global and Local Consequence

In Section 1 we introduced two notions of consequence for hybrid languages, a *local* one and a *global* one. The two notions of consequence are different because of the relativization to worlds. Perhaps it is simpler to discuss consequence in first-order terms, especially if we think of the first-order translation of hybrid or description formulas. The two notions of consequence are always available when we deal with formulas instead of sentences. Given a set of formulas which might contain free variables, the way we define the quantification over models and first-order assignments becomes meaningful.

The global consequence relation is the one familiar from first-order logic, but it is always defined for *sentences*. When we consider formulas instead, the local definition becomes interesting. Because modal and hybrid formulas may contain free variables when translated into a first-order language, it is important to understand the connection between these two notions of consequence.

Proposition 3.1 ([5], Lemma 2.33) *For T a set of modal formulas (in a basic mono-modal language), let $\mathsf{BOXED}(T) = \{\Box^i\psi \mid \psi \in T \text{ and } i \geq 0\}$, where $\Box^i\psi$ is the formula obtained from ψ by prefixing a string of length i, of $\Box$ operators. Then, for any set $T \cup \{\phi\}$ of modal formulas, $T \models^{glo} \phi$ iff $\mathsf{BOXED}(T) \models^{loc} \phi$.*

The proof uses the fact that the collapsed model property holds for the basic modal language. The extension to multi-modal languages is trivial, just redefine BOXED to include all possible boxed prefixes in the multi-modal signature. For languages without the E operator, the proof boils down to finding suitable notions of the collapsed model property. If the language does contain E, the relation between $\models^{glo}$ and $\models^{loc}$ is straightforward:

$$T \models^{glo} \phi \text{ iff } \{\mathsf{A}\psi \mid \psi \in T\} \models^{loc} \phi. \tag{1}$$

Goranko and Passy [10] study properties of languages containing the existential modality, and prove that global properties of a logic $\mathcal{L}$ correspond to the local properties of the logic $\mathcal{L}^{\mathsf{E}}$ which arises from $\mathcal{L}$ by adding E. In particular, for basic modal logics, global decidability, global finite model property, and global completeness of a logic $\mathcal{L}$ are equivalent to their local versions for $\mathcal{L}^{\mathsf{E}}$. This result can be extended to many hybrid languages; before stating it, we establish a normal form for hybrid formulas.

Proposition 3.2 *Let ϕ be a hybrid formula, then ϕ is equivalent to a formula ϕ' where subformulas of the form $\mathsf{E}\psi$ and $@_i\psi$ (if any) occur only at modal depth 0. In particular, ϕ' can be taken to be*

$$\bigwedge_{l\in L} \left(\bigvee_{m\in M} \mathsf{A}\rho_{(l,m)} \vee \mathsf{E}\sigma_l \vee \bigvee_{i\in\mathsf{NOM}(\phi)} @_i\nu_{(l,i)} \vee \tau_l\right),$$

for some (possibly empty) index sets L, M, where $\rho_{(l,m)}$, σ_l, $\nu_{l,i}$ and τ_l contain neither E *nor* $@$. *Furthermore,* $|\phi'|$ *is polynomial in* $|\phi|$.

Proof. We start by translating ϕ into negation normal form. Then we use the following equivalences to "push out" the E and A operators from inside the other modalities

$$\begin{array}{rcl|rcl}
[R_i]\mathsf{A}\psi & \leftrightarrow & [R_i]\bot \vee \mathsf{A}\psi & @_s\mathsf{A}\psi & \leftrightarrow & \mathsf{A}\psi \\
[R_i]\mathsf{E}\psi & \leftrightarrow & [R_i]\bot \vee \mathsf{E}\psi & @_s\mathsf{E}\psi & \leftrightarrow & \mathsf{E}\psi \\
[R_i](\theta \vee \mathsf{A}\psi) & \leftrightarrow & [R_i]\theta \vee \mathsf{A}\psi & @_s(\theta \vee \mathsf{A}\psi) & \leftrightarrow & @_s\theta \vee \mathsf{A}\psi \\
[R_i](\theta \vee \mathsf{E}\psi) & \leftrightarrow & [R_i]\theta \vee \mathsf{E}\psi & @_s(\theta \vee \mathsf{E}\psi) & \leftrightarrow & @_s\theta \vee \mathsf{E}\psi \\
[R_i](\theta \wedge \mathsf{A}\psi) & \leftrightarrow & [R_i]\theta \wedge [R_i]\mathsf{E}\psi & @_s(\theta \wedge \mathsf{A}\psi) & \leftrightarrow & @_s\theta \wedge \mathsf{E}\psi \\
[R_i](\theta \wedge \mathsf{E}\psi) & \leftrightarrow & [R_i]\bot \vee ([R_i]\theta \wedge [R_i]\mathsf{E}\psi) & @_s(\theta \wedge \mathsf{E}\psi) & \leftrightarrow & @_s\theta \wedge \mathsf{E}\psi.
\end{array}$$

Similar equivalences hold for the dual modalities $\langle R_i\rangle$ ($@$ is self dual). For pushing out $@$ we have

$$\begin{array}{rcl|rcl}
[R_i]@_i\psi & \leftrightarrow & [R_i]\bot \vee @_i\psi & @_s@_i\psi & \leftrightarrow & @_i\psi \\
[R_i](\theta \vee @_i\psi) & \leftrightarrow & [R_i]\theta \vee @_i\psi & @_s(\theta \vee @_i\psi) & \leftrightarrow & @_s\theta \vee @_i\psi \\
[R_i](\theta \wedge @_i\psi) & \leftrightarrow & [R_i]\theta \wedge [R_i]@_i\psi & @_s(\theta \wedge @_i\psi) & \leftrightarrow & @_s\theta \wedge @_i\psi.
\end{array}$$

And similarly for the $@$ operators appearing under $\langle R_i\rangle$. Now, it only remains to use propositional equivalences to obtain the normal form for ϕ. $\dashv$

Theorem 3.3 *Let the property P be either decidability, finite model property, or axiomatic completeness, and let $\mathcal{L}$ be any sublanguage of $\mathcal{H}_{\mathsf{N}}(\langle R^{-1}\rangle, @)$. Then $\mathcal{L}$ has P globally iff $\mathcal{L}^{\mathsf{E}}$ has P locally.*

Going back to description languages, notice that if we use $\models^{glo}$ instead of $\models^{loc}$, then basic modal logic *is* enough to encode terminological axioms, as the following equivalence holds: $\langle T, \{\}\rangle \models \phi$ iff $T^t \models^{glo} \phi^t$. By using (1), in the presence of E we can move to $\langle T, \{\}\rangle \models \phi$ iff $\{\mathsf{A}(T^t)\} \models^{loc} \phi^t$. And given that the local consequence relation satisfies the deduction theorem, we end up with

$$\langle T, \{\}\rangle \models \phi \text{ iff } \models \mathsf{A}(T^t) \rightarrow \phi^t.$$

By Theorem 3.3, then, we can study logical properties of inference from non-empty knowledge bases through local properties of languages containing E. If the logic is compact, we can perform this reduction even for infinite T-Boxes, but most importantly by replacing the Kleene star with the existential modality we obtain a language with a much better behaved and understood model theory.

4 De Giacomo's Individuals

Accounting for assertional information in CPDL is more complicated than encoding terminological axioms. Below we present a simplified version of a translation proposed by De Giacomo and Lenzerini [8]; the latter enforce the unique name assumption and also deal with complex structure on roles (union, composition, transitive closure, etc.) which makes for the additional complexity. Here, we only discuss the handling of individuals.

Extend the translation $\cdot^t$ defined in Section 2 to assertions by defining $(a : C)^t = p_a \rightarrow C^t$, and $((a, b) : R)^t = p_a \rightarrow \langle R\rangle p_b$, where p_a and p_b are propositional symbols. Let A be a finite set of assertions, define A^t as $\bigwedge \phi_i^t$ for $\phi_i \in A$. The problem now is that in translating *individuals* as *propositions* in CPDL we have lost the information that individuals denote a single element in the domain. Hence, we have to explicitly force these symbols to behave as individuals. Let $\Sigma = \langle T, A\rangle$ be a knowledge base, $R_1, \ldots, R_n$ the roles appearing in Σ, $a_1, \ldots, a_m$ the individuals mentioned in Σ, and let $\mathsf{SF}(\phi)$ be the set of all subformulas of ϕ. Let $[U]$ stand for $[(R_1 \cup R_1^{-1} \cup \cdots \cup R_n \cup R_n^{-1})^*]$, and let S be a role not in Σ. Let Σ^t be

$$[S][U](A^t \wedge T^t) \wedge \bigwedge_{1\leq i\leq m}\left(\langle S\rangle p_{a_i} \wedge \left(\bigwedge_{\psi\in\mathsf{SF}(T^t\wedge A^t)}[S](\langle U\rangle(p_{a_i}\wedge\psi) \rightarrow [U](p_{a_i}\rightarrow\psi))\right)\right).$$

We will prove that Σ is consistent if and only if Σ^t is satisfiable. This is enough because in sufficiently expressive languages like the ones we consider in this paper, all standard reasoning tasks can be reduced to knowledge base consistency.

Proposition 4.1 *A knowledge base Σ is consistent if and only if Σ^t is satisfiable.*

As remarked by Horrocks et al. [14], De Giacomo's translation is probably too involved and costly to provide effective decision methods. It is also difficult to extract theoretical results from it, except for the general complexity results presented in [8]. As we already remarked, the model theory of CPDL is intricate because of the inductive nature of the Kleene star, and the cryptic translation provides little help on simplifying things out.

The main difficulty with the translation above is in forcing propositional symbols in CPDL to behave as individuals. If we use hybrid logics instead, we can simply use *nominals*. In addition, given our discussion in Section 3, the E modality gives us access to globality and we don't need to rely on the Kleene star. So, hybrid logic, and not CPDL, seems to be the language of choice for a modal counterpart of description languages able to deal with full terminological and assertional reasoning.

5 Into Hybrid Logics

Consider the following translation $\cdot^h$ taking concepts, terminological axioms and assertions to hybrid formulas:

$$\begin{array}{rclrcl}
(C_i)^h & = & p_i,\ (C_i \text{ atomic}) & (\{a_1,\ldots,a_n\})^h & = & a_1 \vee \cdots \vee a_n\\
(\neg C)^h & = & \neg(C^h) & (\exists R.\{a\})^h & = & \langle R\rangle a\\
(C \sqcap D)^h & = & C^h \wedge D^h & (C \sqsubseteq D)^h & = & \mathsf{A}(C^h \rightarrow D^h)\\
(\exists R.C)^h & = & \langle R\rangle C^h & (a : C)^h & = & @_a C^h\\
(\exists R^{-1}.C)^h & = & \langle R^{-1}\rangle C^h & ((a,b) : R)^h & = & @_a\langle R\rangle b.
\end{array}$$

Theorem 5.1 *Let $\Sigma = \langle T, A\rangle$ be a knowledge base in $\mathcal{ALCOI}$, and ϕ a terminological axiom or an assertion, then $\langle T, A\rangle \models \phi$ iff $\models (\bigwedge_{\psi\in T}\psi^h \wedge \bigwedge_{\psi\in A}\psi^h) \rightarrow \phi^h$.*

The proof is obvious (and the connection between the two languages stronger than with CPDL), as any model of $\langle T, A\rangle$ and ϕ can be viewed as a model of $(\bigwedge_{\psi\in T}\psi^h \wedge \bigwedge_{\psi\in A}\psi^h) \rightarrow \phi^h$ and vice versa. By using additional nominals we can also account for role conjunction: $(\exists(R_1 \sqcap R_2).C)^h = \langle R_1\rangle i \wedge \langle R_2\rangle i \wedge @_i C^h$, for i a new nominal, while $((a,b) : R_1 \sqcap R_2)^h = @_a\langle R_1\rangle b \wedge @_a\langle R_2\rangle b$. Equivalently, we could have put $(\exists(R_1 \sqcap R_2).C)^h = \langle R_1\rangle(i \wedge C^h) \wedge \langle R_2\rangle(i \wedge C^h)$, and do without @. But this is not a linear translation and, as we will soon see, using @ and restricting the use of nominals is more "natural" from a description logic point of view. Notice that in any case, we need to move to an extended language to account for role conjunction (as we need new nominals) in this way. To remain in the spirit (and strength) of the previous translation we would do better by introducing role conjunction into hybrid logics as investigated in [21].

Like us, Blackburn and Tzakova [7] propose using hybrid languages to embed description logics, highlighting the connection between assertional information and nominals, and the use of the existential modality to encode terminological axioms. But they introduce undecidable hybrid languages for this account, arguing that the increase in expressive power of these languages is an advantage. Instead, our translation tries to remain as faithful as possible to the original description language, and pays special attention to decidability issues.

What kind of expressive power is needed to encode the different languages and reasoning tasks? For example, the existential modality is required only for translating terminological axioms, while @ is only used for assertions. The following list a number of precise correspondences:

- $\mathcal{H}_{\mathsf{N}}(\langle R^{-1}\rangle, @, \mathsf{E})$, in which the full translation of $\mathcal{ALCOI}$ with non-empty T-Boxes and A-Boxes can be made.
- $\mathcal{H}_{\mathsf{N}}(\langle R^{-1}\rangle, @)$, in which only inferences in terms of empty T-Boxes can be performed.
- $\mathcal{H}(\langle R^{-1}\rangle, @, @\Diamond, \mathsf{E})$, in which we only allow nominals to appear as subindices of @ and in the construction $@_a\langle R\rangle b$ or $@_a\langle R^{-1}\rangle b$, and hence we can translate neither the one-of operator $\mathcal{O}$ nor role fillers $\mathcal{B}$.
- $\mathcal{H}(\langle R^{-1}\rangle, @, @\Diamond)$, the "local" version of the language above, where we work with empty T-Boxes.

We have defined each of the logics above to be expressive enough to permit the encoding of certain specific DLs. But it is also important to determine if and how we have *extended* the expressive power of the source language with the move into these hybrid languages. The general answer is: we have incorporated Boolean structure into the knowledge base, and allowed explicit interaction among T-Box definitions, A-Box assertions and concepts. Take, for example, the most expressive language $\mathcal{H}_{\mathsf{N}}(\langle R^{-1}\rangle, @, \mathsf{E})$. Given Proposition 3.2, we can take $\phi \in \mathcal{H}_{\mathsf{N}}(\langle R^{-1}\rangle, @, \mathsf{E})$ to be

$$\bigwedge_{l\in L}\left(\bigvee_{m\in M}\mathsf{A}\rho_{(l,m)} \vee \mathsf{E}\sigma_l \vee \bigvee_{i\in\mathsf{NOM}} @_i\nu_{(l,i)} \vee \tau_l\right),$$

where $\rho_{(l,m)}$, σ_l, $\nu_{(l,i)}$ and τ_l contain neither E nor @. By allowing negations in the T-Box we can encode validity of formulas in $\mathcal{H}_{\mathsf{N}}(\langle R^{-1}\rangle, @, \mathsf{E})$ as instance checking as follows. Define *Boolean knowledge bases* as pairs $\Sigma = \langle T, A\rangle$ where T is a set of Boolean combinations of terminological axioms, and A a set of Boolean combinations of assertions. Authors with a modal logic background like Wolter and Zakharyachev have already considered this kind of knowledge bases [26].

For $l \in L$, define the knowledge base $\Sigma^l_\phi = \langle T^l_\phi, A^l_\phi\rangle$ to be $T^l_\phi = \{\neg(\top \sqsubseteq \rho^{h^{-1}}_{(m,l)}) \mid m \in M\} \cup \{\top \sqsubseteq \neg\sigma^{h^{-1}}_l\}$ and $A^l_\phi = \{i\!:\!\neg\nu^{h^{-1}}_{(l,i)} \mid i \in \mathsf{NOM}(\phi)\}$, where the mapping $\cdot^{h^{-1}}$ is the backwards translation from the hybrid language into $\mathcal{ALCOI}$ that sends Boolean and modal operators to the corresponding description logic ones and using singleton one-of sets $\{i\}$ for translating nominals.

Theorem 5.2 *For any formula ϕ in $\mathcal{H}_{\mathsf{N}}(\langle R^{-1}\rangle, @, \mathsf{E})$, let $a \notin \mathsf{NOM}(\phi)$, then ϕ is valid iff for all $l \in L$, $\Sigma^l_\phi \models a\!:\!\tau^{h^{-1}}_l$.*

Interestingly, even if we allow Boolean knowledge bases, we cannot recast validity of hybrid formulas as inference in terms of a *unique* knowledge base. This is because the separation between terminological axioms, assertions and simple concepts imposes syntactic restrictions which don't exist in hybrid logic. Trivially, if the index set L above is a singleton, then a unique knowledge base is sufficient.

I.e., we can precisely characterize the fragment of $\mathcal{H}_{\mathsf{N}}(\langle R^{-1}\rangle, @, \mathsf{E})$ that captures the expressivity of $\mathcal{ALCOI}$ with Boolean knowledge bases.

As we will see in Section 6, allowing the extra flexibility that Boolean knowledge bases offer does not modify the complexity class in which the reasoning tasks fall (for the languages we are considering), but it does increase expressive power.

6 Pay Day

The links between hybrid logics and DLs discussed in the previous sections are so strong, that we can immediately start harvesting by interpreting results from one field in the other. This is what we will do now, from many different perspectives: complexity, expressive power, meta-logical properties, new operators, etc.

§**6.1 Complexity.** We start by exporting complexity results for hybrid logics to DLs. We need to pay attention to the difference between local and global notions. For a modal language, we can distinguish between the local-Sat problem (given a formula ϕ, does there exist a model $\mathcal{M}$ and $m \in M$ with $\mathcal{M}, m \Vdash \phi$?), and the global-$Sat$ problem (is there a model $\mathcal{M}$ with $\mathcal{M} \Vdash \phi$?). If the logic contains the E modality, the problems coincide, as we argued in Section 3.

First, we consider the "pure future" fragments of the hybrid languages defined in Section 5, i.e., we only consider formulas without the $\langle R^{-1}\rangle$ operator. The local-Sat problem for $\mathcal{H}_{\mathsf{N}}(@)$ is PSPACE-complete [1]. This result also settles the complexity of $\mathcal{H}(@, @\Diamond)$, because this language contains the basic modal language. As a corollary of the EXPTIME-completeness of CPDL, we obtain an EXPTIME upper bound for the local-Sat problem for $\mathcal{H}_{\mathsf{N}}(@, \mathsf{E})$. It follows from Spaan's results on the EXPTIME-completeness of modal logics with the existential modality [24], that both $\mathcal{H}(@, @\Diamond, \mathsf{E})$ and $\mathcal{H}_{\mathsf{N}}(@, \mathsf{E})$ are EXPTIME-complete.

Switching to the DL perspective, the results above imply that it is the move from empty T-boxes to full T-boxes that modifies complexity. And this does not depend on our extension to "Boolean" knowledge bases, as the same complexity results obtain when we restrict ourselves to standard knowledge bases. The one-of operator $\mathcal{O}$ and role fillers $\mathcal{B}$ offer more expressivity at no cost (up to a polynomial). At this very point the encoding of DLs into hybrid languages, instead of CPDL, works to our advantage, since we can identify cases with a PSPACE upper bound.

Theorem 6.1

1. *Instance checking for Boolean knowledge bases with empty T-boxes is PSPACE-complete for the language $\mathcal{ALCROB}$.*
2. *Instance checking for Boolean knowledge bases is solvable in EXPTIME (hence EXPTIME-complete) for the language $\mathcal{ALCROB}$.*

Notice that we don't need to restrict to empty A-boxes in item 1, and recall that the complexity results for instance checking extend to all standard reasoning tasks like knowledge base consistency, or subsumption checking.

What about the $\langle R^{-1}\rangle$ operator? Adding just *one* nominal to basic temporal logic moves the complexity of the local-Sat problem from PSPACE-hard to EXPTIME-hard. The known EXPTIME upper bound for CPDL plus nominals and E [1] also covers $\mathcal{H}_{\mathsf{N}}(\langle R^{-1}\rangle, @, \mathsf{E})$; hence, the local-$Sat$ problems of $\mathcal{H}_{\mathsf{N}}(\langle R^{-1}\rangle, @)$, $\mathcal{H}(\langle R^{-1}\rangle, @, @\Diamond, \mathsf{E})$ and $\mathcal{H}_{\mathsf{N}}(\langle R^{-1}\rangle, @, \mathsf{E})$ are EXPTIME-complete.

A PSPACE upper bound for $\mathcal{H}(\langle R^{-1}\rangle, @, @\Diamond)$ is easy to establish by using the fact that @ operators need only appear at modal depth 0. We give a sketch of the proof. To avoid confusion we will write $@_i\langle R_r\rangle j$ as $R_r(i,j)$. Let

$$\phi = \bigwedge_{l\in L}\left(\bigvee_{i\in\mathsf{NOM}} @_i\nu_{(l,i)} \vee \bigvee T_l \vee \sigma_l\right),$$

where each T_l is a collection of formulas of the form $R_r(i,j)$ or $\neg R_r(i,j)$, and $\nu_{(l,i)}, \sigma_l$ contain neither $@$ nor nominals. As PSPACE = NPSPACE, non-deterministically choose from each conjunct of ϕ the disjunct satisfied by a model of ϕ. Call such a set CHOICE. Now, for each i, let $S_i = \{\phi \mid @_i\phi \in \mathsf{CHOICE}\}$, and create a polynomial model satisfying S_i at the point m_i (notice that all formulas in S_i are basic temporal formulas and hence a PSPACE model can be constructed). Similarly, create a polynomial model for all formulas in CHOICE which are not @-formulas. Let $\mathcal{M}$ be the disjoint union of all these models. Finally, if $R_r(i,j) \in \mathsf{CHOICE}$, add the pair (m_i, m_j) to R_r. The model of ϕ obtained in this way has size polynomial in $|\phi|$.

With the translation into CPDL it would be impossible to evaluate the difference made by the presence or absence of the $\langle R^{-1}\rangle$ operator in terms of complexity.

Theorem 6.2

1. *Instance checking for Boolean knowledge bases with empty T-Boxes is solvable in PSPACE (hence PSPACE-complete) for the language $\mathcal{ALCRBI}$.*
2. *Instance checking for knowledge bases with empty T- and A-boxes is EXPTIME-hard for the language $\mathcal{ALCIO}$.*
3. *Instance checking for Boolean knowledge bases is solvable in EXPTIME (hence EXPTIME-complete) for the language $\mathcal{ALCROBI}$.*

The complexity results listed so far were based on importing hybrid logic results into DL. The EXPTIME-hardness result for $\mathcal{H}_{\mathsf{N}}(\langle R^{-1}\rangle)$ (basic temporal logic with at least one nominal) contrast sharply with the good complexity behavior of $\mathcal{H}_{\mathsf{N}}(@)$. For example, if we move to the class of transitive models, even $\mathcal{H}_{\mathsf{N}}(@, \mathsf{E})$ is PSPACE-complete (meaning that there are PSPACE algorithms even for inference from non-empty T-Boxes), while $\mathcal{H}_{\mathsf{N}}(\langle R^{-1}\rangle)$ remains obstinately in EXPTIME. Results concerning the complexity of hybrid logics in different classes of models are investigated in [2]. One of the main results in this paper implies that instance checking for Boolean knowledge bases in $\mathcal{ALCROBI}$ can be solved in PSPACE if we consider only transitive trees as models.

Going in the opposite direction, known complexity results from DLs can be translated in hybrid terms. For example, as we will discuss below, little is know about the extension of hybrid languages with counting. Further, there is a "folklore" result which states that instance checking for $\mathcal{ALC}$ with T-Boxes restricted to simple and acyclic terminological axioms is PSPACE-complete; this implies that when syntactic restrictions are imposed on the use of E, we can avoid EXPTIME-hardness for the local-Sat problem of $\mathcal{H}(@, @\Diamond, \mathsf{E})$. Lutz [18, 19] provides the first detailed complexity analysis of inference from simple, acyclic T-Boxes. Interestingly, the restriction to simple, acyclic T-Boxes does not always preserve complexity: instance checking in $\mathcal{ALCF}$ ($\mathcal{ALC}$ extended with features, feature agreement and feature disagreement) is PSPACE-complete for empty T-Boxes, but it turns NEXPTIME-complete even when only simple, acyclic T-Boxes are allowed.

§**6.2 Expressive Power.** We now consider expressive power, and we do so by taking advantage of hybrid bisimulations. Bisimulations are binary relations on the domain of hybrid models. Kurtonina and de Rijke [16] provide a detailed analysis of the expressive power of concept languages by means of (bi-)simulations. But Kurtonina and de Rijke's results only address the expressive power of *concepts*. In this section, we will instead study the expressive power offered by full knowledge bases.

Let $\mathcal{M} = \langle M, \{R_r^{\mathcal{M}}\}, V^{\mathcal{M}}\rangle$ and $\mathcal{N} = \langle N, \{R_r^{\mathcal{N}}\}, V^{\mathcal{N}}\rangle$ be two hybrid models. For $i \in \mathsf{NOM}$, let $i^{\mathcal{M}}$ be the denotation of i in $\mathcal{M}$ and similarly for $i^{\mathcal{N}}$. Let $\sim$ be a non-empty binary relation on $M \times N$, and consider the following properties on $\sim$ in addition to the conditions for bisimulation for a basic temporal language [6]:

(@) For all nominals i in NOM, $i^{\mathcal{M}} \sim i^{\mathcal{N}}$.

(@◇) Let i, j be nominals in NOM, then $R_r(i^{\mathcal{M}}, j^{\mathcal{M}})$ iff $R_r(i^{\mathcal{N}}, j^{\mathcal{N}})$.

(**E**) $\sim$ is total and surjective.

A bisimulation $\sim$ for a basic temporal language is a $\mathcal{H}(\langle R^{-1}\rangle, @, @\Diamond)$-bisimulation if it satisfies the conditions (@) and (@◇). Further, $\sim$ is a $\mathcal{H}_{\mathsf{N}}(\langle R^{-1}\rangle, @)$-bisimulation if bisimilar states agree on all nominals (and in this case (@◇) can be derived from the others). $\mathcal{H}(\langle R^{-1}\rangle, @, @\Diamond, \mathsf{E})$- and $\mathcal{H}_{\mathsf{N}}(\langle R^{-1}\rangle, @, \mathsf{E})$-bisimulations are obtained from $\mathcal{H}(\langle R^{-1}\rangle, @, @\Diamond)$- and $\mathcal{H}_{\mathsf{N}}(\langle R^{-1}\rangle, @)$-bisimulations, respectively, by requiring the additional condition (**E**).

Proposition 6.3 *Let $\mathcal{H}$ be one of $\mathcal{H}(\langle R^{-1}\rangle, @, @\Diamond)$, $\mathcal{H}_{\mathsf{N}}(\langle R^{-1}\rangle, @)$, $\mathcal{H}(\langle R^{-1}\rangle, @, @\Diamond, \mathsf{E})$ and $\mathcal{H}_{\mathsf{N}}(\langle R^{-1}\rangle, @, \mathsf{E})$. Let $\mathcal{M} = \langle M, \{R_r\}, V\rangle$ and $\mathcal{N} = \langle N, \{R_r^{\mathcal{N}}\}, V^{\mathcal{N}}\rangle$, and let $\sim$ be an $\mathcal{H}$-bisimulation between $\mathcal{M}$ and $\mathcal{N}$. Then for $m \in M, n \in n$, and ϕ in $\mathcal{H}$, $m \sim n$ implies $\mathcal{M}, m \Vdash \phi$ iff $\mathcal{N}, n \Vdash \phi$.*

For two logics we write $\mathcal{H} \preccurlyeq \mathcal{H}'$ to denote that for each formula ϕ in $\mathcal{H}$ there exists a formula ϕ' in $\mathcal{H}'$ such that ϕ is satisfiable if and only if ϕ' is. We write $\mathcal{H} \prec \mathcal{H}'$ if $\mathcal{H} \preccurlyeq \mathcal{H}'$ and not $\mathcal{H}' \preccurlyeq \mathcal{H}$. It is immediate that $\mathcal{H}(\langle R^{-1}\rangle, @, @\Diamond) \preccurlyeq \mathcal{H}_{\mathsf{N}}(\langle R^{-1}\rangle, @)$ and $\mathcal{H}(\langle R^{-1}\rangle, @, @\Diamond, \mathsf{E}) \preccurlyeq \mathcal{H}_{\mathsf{N}}(\langle R^{-1}\rangle, @, \mathsf{E})$. More interestingly, each of the relations is actually strict, which can be shown by means of bisimulations. In DL terms this means, for instance, that the one-of operator $\mathcal{O}$ does increase the expressive power of the language, both with full and empty T-boxes.

The relation between $\mathcal{H}_{\mathsf{N}}(\langle R^{-1}\rangle, @)$ and $\mathcal{H}(\langle R^{-1}\rangle, @, @\Diamond, \mathsf{E})$ is more complex. By using bisimulations, we can prove both that $\mathcal{H}_{\mathsf{N}}(\langle R^{-1}\rangle, @) \not\preccurlyeq \mathcal{H}(\langle R^{-1}\rangle, @, @\Diamond, \mathsf{E})$ and $\mathcal{H}(\langle R^{-1}\rangle, @, @\Diamond, \mathsf{E}) \not\preccurlyeq \mathcal{H}_{\mathsf{N}}(\langle R^{-1}\rangle, @)$. Nevertheless, $\mathcal{H}(\langle R^{-1}\rangle, @, @\Diamond, \mathsf{E})$ is at least as expressive as $\mathcal{H}_{\mathsf{N}}(\langle R^{-1}\rangle, @)$ if we are only interested in satisfiability (and not in the existence of an equivalent formula).

Proposition 6.4 *Let ϕ be a formula in $\mathcal{H}_{\mathsf{N}}(\langle R^{-1}\rangle, @)$, then there exists a formula $\phi' \in \mathcal{H}(\langle R^{-1}\rangle, @, @\Diamond, \mathsf{E})$ such that ϕ is satisfiable iff ϕ' is satisfiable.*

These expressive separation results easily translate to description languages. For two description languages $\mathcal{L}_1$ and $\mathcal{L}_2$, define $\mathcal{L}_1 \preccurlyeq \mathcal{L}_2$ if for any knowledge bases Σ in $\mathcal{L}_1$ there is a knowledge base Σ' in $\mathcal{L}_2$ such that for all interpretations $\mathcal{I}$, $\mathcal{I} \models \Sigma$ iff $\mathcal{I} \models \Sigma'$. Now that the formulas used to separate the languages can easily be recast as assertions or terminological definitions, and similarly for the translation used in the proof of Proposition 6.4.

The notions of bisimulation we have defined not only separate the fragments of first-order logic which corresponds to the hybrid logics we have been discussing, they also *characterize* them. For $\mathcal{H}$ any of our hybrid languages, we say that a first-order formula $\alpha(x)$ in the first-order language over $\langle \mathsf{REL} \cup \{P_j \mid p_j \in \mathsf{PROP}\}, \mathsf{NOM}, \{x, y\}\rangle$ is *invariant for $\mathcal{H}$-bisimulations* if for all models $\mathcal{M}$ and $\mathcal{N}$, and all states m in $\mathcal{M}$, n in $\mathcal{N}$, and all $\mathcal{H}$-bisimulations $\sim$ between $\mathcal{M}$ and $\mathcal{N}$ such that $m \sim n$, we have $\mathcal{M} \models \alpha(x)[m]$ iff $\mathcal{N} \models \alpha(x)[n]$.

Theorem 6.5 *For $\mathcal{H}$ any of $\mathcal{H}(\langle R^{-1}\rangle, @, @\Diamond)$, $\mathcal{H}_{\mathsf{N}}(\langle R^{-1}\rangle, @)$, $\mathcal{H}(\langle R^{-1}\rangle, @, @\Diamond, \mathsf{E})$ or $\mathcal{H}_{\mathsf{N}}(\langle R^{-1}\rangle, @, \mathsf{E})$, a first-order formula $\alpha(x)$ over the signature $\langle \mathsf{REL} \cup \{P_j \mid p_j \in \mathsf{PROP}\}, \mathsf{NOM}, \{x, y\}\rangle$ is invariant for $\mathcal{H}$-bisimulations iff it is equivalent to the hybrid translation of a hybrid formula in $\mathcal{H}$.*

Results describing the classes of models that can be globally defined, can be proved for many hybrid languages, but have to be omitted because of space limitations.

§**6.3 Interpolation and Beth Definability.** In [3] results concerning the interpolation and Beth definability properties for a variety of hybrid languages are provided. What is the role of these two properties in the setting of description logics? Let's first introduce some notation. For $\Sigma = \langle T, A \rangle, \Sigma' = \langle T', A' \rangle$ two knowledge bases, let $\Sigma \cup \Sigma'$ be $\langle T \cup T', A \cup A' \rangle$, and $\Sigma[C/D]$ be the knowledge base obtained from Σ by replacing each occurrence of the concept C by D. Now, suppose that for a given knowledge base Σ the following holds,

$$\Sigma[C/D_1] \cup \Sigma[C/D_2] \models D_1 \doteq D_2 \text{ for some } D_1, D_2 \notin \mathsf{CON}(\Sigma). \quad (2)$$

Notice that this equation need not be true for all knowledge bases Σ and concepts C. Actually, (2) implies that Σ encodes enough information concerning C to provide a complete—though not necessarily explicit—definition. Now, if the (global) Beth definability property holds for the language of Σ, then there actually exists an explicit definition of C. I.e., there is a concept D not involving C such that $\Sigma \models C \doteq D$. Given that description languages take definitions very seriously, the Beth definability property (i.e., the capacity of the language to turn implicit definitions into explicit ones) seems highly relevant.

There doesn't seem to be a uniform direct way of proving or disproving Beth definability. The standard approach to establish the property is via a detour through interpolation. In first-order and modal languages, the (arrow) interpolation property implies the Beth definability property and the same relation holds for hybrid languages.

Hence, positive interpolation results for hybrid languages would translate into nice definability properties of the corresponding description language. Unfortunately, for languages where nominals appear free in formulas, and which do not provide a binding mechanism, failure of arrow interpolation seems to be the norm. In particular, [3] provides counter-examples to the arrow interpolation property for the basic modal language extended with nominals, $\mathcal{H}_{\mathsf{N}}(@)$ and $\mathcal{H}_{\mathsf{S}}(@)$. The extensions of these languages with the $\langle R^{-1} \rangle$ operator fare no better, and adding the E operator doesn't help either. Hence, in all these cases, the most traded path to establish Beth definability is closed for us.

The case is different for $\mathcal{H}(@, @\Diamond)$ and $\mathcal{H}(\langle R^{-1} \rangle, @, @\Diamond)$. As we will now show, we can extend the constructive method for establishing arrow interpolation presented in [15, Section 3.8], to handle $@$ and $@\Diamond$. Again we will make use of the normal form introduced in Proposition 3.2.

Theorem 6.6 *$\mathcal{H}(@, @\Diamond)$ and $\mathcal{H}(\langle R^{-1} \rangle, @, @\Diamond)$ have arrow interpolation.*

As we said, arrow interpolation implies global Beth definability: implicit definitions in $\mathcal{H}(@, @\Diamond)$ can be turned into explicit definitions. And we can attempt to transfer this property to the description logic counterpart of $\mathcal{H}(@, @\Diamond)$. We would proceed as follows, suppose a knowledge base $\Sigma = \langle T, A \rangle$ in $\mathcal{ALC}$ satisfies the conditions in (2). Then we can translate Σ into a theory T of $\mathcal{H}(@, @\Diamond)$ (as we are using global consequence this time we don't need E), and $T[p_C/p_{D_1}] \cup T[p_C/p_{D_2}] \models^{glo} p_{D_1} \leftrightarrow p_{D_2}$. Applying Beth definability for $\mathcal{H}(@, @\Diamond)$ we obtain a formula θ such that $T \models^{glo} \theta \leftrightarrow p_C$. Now, θ is an explicit definition of C, but it is in the *full* language $\mathcal{H}(@, @\Diamond)$, i.e., it might contain subformulas of the form $@_i\psi$ and $@_i\Diamond j$. Because of the syntactic restrictions imposed by the division into T- and A-Box information it will not always be possible to translate θ into a concept in $\mathcal{ALC}$. To see an example, suppose θ is of the form $@_i\nu \vee \psi$. Hence we will have that $\Sigma \models (@_i\nu \to (p_C \leftrightarrow \top)) \wedge (@_i\neg\nu \to (p_C \leftrightarrow \psi))$. That is, we obtain a definition of C that is conditional on assertional information.

More generally, we first write θ in normal form to obtain

$$T \models^{glo} \left(\bigwedge_{l \in L} (\bigvee_{i \in \mathsf{NOM}} @_i\nu_{(i,l)}) \vee \tau_l \right) \leftrightarrow p_C.$$

Notice that for a hybrid formula ψ and $@_i\nu \in \mathsf{SF}(\psi)$ such that $@$ does not appear in ν, ψ is equivalent to $(@_i\nu \rightarrow \psi[@_i\nu/\top]) \wedge (@_i\neg\nu \rightarrow \psi[@_i\nu/\bot])$. By iterating this rewriting on the formula $(\bigwedge_{l\in L}(\bigvee_{i\in\mathsf{NOM}} @_i\nu_{(i,l)}) \vee \tau_l) \leftrightarrow p_C$, we finally obtain a series of definitions of C in terms of concepts of $\mathcal{ALC}$, but conditional on assertional information to be inferred from Σ.

There is an interesting connection between the Beth definability property and acyclic definitions in T-Boxes. This restriction was aimed at avoiding the introduction of circular concepts, i.e., concepts defined in terms of themselves. This kind of concepts, it was argued, called for some kind of fixed point semantics and this kind of semantics was computationally expensive [20, 4]. But if the language has the Beth definability property, any concept implicitly defined in a knowledge base also has an explicit definition *without self reference*. Hence, considering only acyclic definitions does not carry any expressivity loss.

§**6.4 Further Outcomes.** It is striking how description and hybrid logics are similar and different at the same time, like twin brothers raised separately. Because the connection between hybrid and description logics is so tight, complexity and expressive power results can easily be moved between the two, as we have seen. Having *two* different perspectives also brings extra flexibility. E.g., we can investigate meta-logical properties on the "hybrid logic side" which is presented in a format more amenable to standard model-theoretic techniques, and these results throw light on the behavior of description languages. We give some examples.

One possibility concerns *binders and variables*. We have focused on "weak" hybrid languages which remain close to the basic DLs. But a natural step from the hybrid point of view is to regard nominals not as names but as *variables* over individual states, and to add quantifiers. Undecidability quickly shows up in this setting, but syntactic restriction can restore decidability, while providing interesting new concepts when introduced in a description language.

A different direction concerns *counting*. Graded or counting modalities $\langle n\rangle\phi$ restrict the number of possible successors satisfying ϕ that a state has in a model. While their theory is not so well developed, qualifying number restrictions are actively used in description formalisms, as they provide important modeling power. Recent work by Tobies [25] provides promising new complexity results that are worth linking to findings of the present paper.

7 Conclusion

Nearly a decade after Schild started exploring the connection between modal and description logic, we have made an important step forwards in finding a formal logical counterpart of DLs. One of the key points of DLs is their use of non-empty T-Boxes and A-Boxes. The former essentially concerns global information, which lifts the complexity of the satisfiability problem to EXPTIME. With our analysis of DLs in terms of hybrid logics, we can reason with non-empty A-Boxes (and empty T-Boxes) in PSPACE. Our fine-grained analysis in terms of hybrid logics allows us to capture the expressive power offered by T-Boxes, A-Boxes, and both. Because hybrid logics offer a mathematical logical counterpart of DLs, we can use formal logical results and techniques such as interpolation and Beth definability to analyze DLs. Different enough to make the comparisons interesting, but similar enough to allow for extensive traffic of results, extensions and variations, description logics and hybrid logics are different sides of the same coin.

Acknowledgments. We would like to thank the anonymous referees for their valuable comments. Maarten de Rijke was supported by the Spinoza project 'Logic in Action' and by a grant from the Netherlands Organization for Scientific Research (NWO), under project number 365-20-005.

References

[1] C. Areces, P. Blackburn, and M. Marx. A road-map on complexity for hybrid logics. In J. Flum and M. Rodríguez-Artalejo, editors, *Computer Science Logic*, LNCS 1683, pages 307–321. Springer, 1999.

[2] C. Areces, P. Blackburn, and M. Marx. The computational complexity of hybrid temporal logics. *Logic J. of the IGPL*, 8(5):653–679, 2000.

[3] C. Areces, P. Blackburn, and M. Marx. Hybrid logics: characterization, interpolation and complexity. *J. of Symbolic Logic*, 2000. To appear.

[4] F. Baader. Terminological cycles in KL-ONE-based knowledge representation languages. In *Proc. of the 8th National Conference on Artificial Intelligence (AAAI-90)*, pages 621–626, 1990.

[5] J. van Benthem. *Modal Logic and Classical Logic*. Bibliopolis, Naples, 1983.

[6] P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic*. Cambridge University Press, 2000. To appear.

[7] P. Blackburn and M. Tzakova. Hybridizing concept languages. *Annals of Math. and Art. Int.*, 24:23–49, 1998.

[8] G. De Giacomo and M. Lenzerini. Boosting the correspondence between description logics and propositional dynamic logics. In *Proc. AAAI'94*, pages 205–212, 1994.

[9] M. Fischer and R. Ladner. Propositional dynamic logic of regular programs. *J. of Computer and System Sciences*, 18(2):194–211, 1979.

[10] V. Goranko and S. Passy. Using the universal modality: gains and questions. *J. of Logic and Computation*, 2:5–30, 1992.

[11] J. Halpern and Y. Moses. A guide to completeness and complexity for modal logics of knowledge and belief. *Artificial Intelligence*, 54:319–379, 1992.

[12] D. Harel. Dynamic logic. In D. Gabbay and F. Guenthner, editors, *Handbook of Philosophical Logic. Vol. II*, volume 165 of *Synthese Library*, pages 497–604. D. Reidel Publishing Co., Dordrecht, 1984. Extensions of classical logic.

[13] I. Horrocks and G. Gough. Description logics with transitive roles. In M. Rousset, R. Brachmann, F. Donini, E. Franconi, I. Horrocks, and A. Levy, editors, *Proc. of the International Workshop on Description Logics*, pages 25–28, Gif sur Yvette, France, 1997.

[14] I. Horrocks, U. Sattler, and S. Tobies. Practical reasoning for very expressive description logics. In C. Areces, E. Franconi, R. Goré, M. de Rijke, and H. Schlingloff, editors, *Methods for Modalities, 1*, volume 8(3), pages 239–264. Logic J. of the IGPL, 2000.

[15] M. Kracht. *Tools and Techniques in Modal Logic*. North-Holland Publishing Co., Amsterdam, 1999.

[16] N. Kurtonina and M. de Rijke. Expressiveness of concept expressions in first-order description logics. *Artificial Intelligence*, 107(2):303–333, 1999.

[17] R. Ladner. The computational complexity of provability in systems of modal propositional logic. *SIAM J. of Computing*, 6(3):467–480, 1977.

[18] C. Lutz. Complexity of terminological reasoning revisited. In *Proc. LPAR'99*, LNAI, pages 181–200. Springer-Verlag, 1999.

[19] C. Lutz. On the complexity of terminological reasoning. LTCS-Report 99-04, LuFg Theoretical Computer Science, RWTH Aachen, Germany, 1999.

[20] B. Nebel. Terminological cycles: semantics and computational properties. In J. Sowa, editor, *Principles of Semantic Networks*, pages 331–361. Morgan Kaufmann, Los Altos, 1990.

[21] S. Passy and T. Tinchev. PDL with data constants. *Information Processing Letters*, 20(1):35–41, 1985.

[22] U. Sattler. A concept language extended with different kinds of transitive roles. In G. Görz and S. Hölldobler, editors, *20. Deutsche Jahrestagung für KI*, LNAI 1137. Springer-Verlag, 1996.

[23] K. Schild. A correspondence theory for terminological logics. In *Proc. of the 12th IJCAI*, pages 466–471, 1991.

[24] E. Spaan. *Complexity of Modal Logics*. PhD thesis, Institute for Logic, Language and Computation, University of Amsterdam, 1993.

[25] S. Tobies. The complexity of reasoning with cardinality restrictions and nominals in expressive description logics. *J. of Artificial Intelligence Research*, 2000. To appear.

[26] F. Wolter and M. Zakharyaschev. Modal description logics: modalizing roles. *Fund. Informaticae*, 2000. To appear.