A Proposal of Description Logic on Concept Lattices*

N.V. Shilov[1,2] and S.-Y. Han[1]

[1] School of Computer Science and Engineering
Chung-Ang University 410, 2nd Engineering Building 221,
Heukseok-dong, Dongjak-gu Seoul 156-756, Korea
shilov@cau.ac.kr, hansy@cau.ac.kr
[2] A.P. Ershov Institute of Informatics Systems,
Lavren'ev av., 6, Novosibirsk 630090, Russia,
shilov@iis.nsk.su

Abstract. There are two major formalisms that are developed around concepts: (1) Formal Concept Analysis (FCA) by R. Wille and B. Ganter, and (2) Description Logic (DL) that goes back to the universal terminological logic by P.F. Patel-Schneider. It has been demonstrated that FCA constructs (upper and lower derivatives, formal concepts) are expressible in DL. Present paper demonstrates how to interpret (positive) DL concepts over concept lattices in a compatible manner.

1 Introduction

Two major formalisms developed around concepts are

- Formal Concept Analysis (FCA) by R. Wille and B. Ganter [4, 12],
- Description Logics (DL) that go back to P.F. Patel-Schneider [2].

Description Logic (DL) is a set of knowledge representing languages closely related to modal [3] and program logics [6]. These languages can be used for description of the terminological knowledge in a structured way. They have become a cornerstone of the W3C-endorsed Web Ontology Language (OWL) [7]. DL basic notions are concept and role terms. Concept terms correspond to formulas of modal and program logics. An interpretation assigns sets of objects (that are called concepts in DL) to concept terms. The most important DL notion is knowledge base. A knowledge base is a collection of (subsumption) statements between pairs of concept terms and pairs of role terms. In this way DL represents that a concept is a subconcept of another one, a role is a subrole of another role.

* This research is supported in parts by (a) MIC(Ministry of Information and Communication), Korea, under the Foreign Professor Invitation Program of the IITA(Institute for Information Technology Advancement), (b) by joint grant RFBR 05-01-04003-a - DFG project COMO, GZ: 436 RUS 113/829/0-1, (c) by Integration Grant n.14 Siberia Branch, Russian Academy of Science.

Formal concept analysis is an algebraic framework for data representation and analysis. It takes an input matrix that specifies a relation between objects and attributes. Then FCA finds corresponding 'closed' sets of attributes and 'closed' sets of objects. Every pair of corresponding 'closed' sets of objects and attributes forms a formal concept[3]. The set of attributes in each formal concept can be interpreted as a set of necessary and sufficient conditions for defining the set of objects in the concept. The family of formal concepts obeys the mathematical axioms defining a lattice, and is called a concept lattice.

To the best of our knowledge, there are few research on combination of Formal Concept Analysis with Description Logic [9, 1, 8]. Roughly speaking, all these attempts can be classified as follows:

- accelerating one formalism by another,
- emulating one formalism by another.

The dissertation [9] and paper [1] attempted to accelerate DL by variants of a so-called attribute exploration technique that goes back to FCA. A variant suggested by [9] is called Relational Exploration. It allows determining all 'attribute implications' that follows from knowledge base presented in terms of a simple description logic FLE. (This logic admits concept intersections, existential and universal restrictions.) A variant of attribute exploration proposed by [1] can deal with a partial context and any description logic that is closed under complement and conjunction. An algorithm developed in the cited paper constructs a complete extension of a given partial knowledge base and at the same time it guarantees a minimum communication with knowledge engineers.

In contrast, paper [8] has attempted to emulate DL universal restriction in FCA by some algebraic fix-point construction and to demonstrate its utility for analysis of relational data.

The present paper discusses two variants of a 'combination' of DL and FCA. The first variant is sketched in brief since it has been discussed in full details in a recent workshop paper [11]. This variant emulates FCA in DL by extending language of any description logic $\mathcal{L}$ by upper and lower derivatives that come from FCA. The resulting description logic is denoted by $\mathcal{L}$/FCA. Paper [11] has proved that $\mathcal{L}$/FCA can be expressed in $\mathcal{L}(\neg, -)$ – another variant of $\mathcal{L}$ that is closed with respect to role complement and inverse.

A new variant of a 'combination' of DL and FCA emulates a particular description logic – (positive fragment of) Attribute Language with Complements ($\mathcal{ALC}$) [10] – by algebraic operations on concept lattices. This emulation can be considered also as a usage of concept lattices as domains for terminological interpretations, i.e. as a conceptual semantics for $\mathcal{ALC}$. Informal soundness of this emulation is supported by a compatibility of the conceptual semantics with the standard semantics of $\mathcal{ALC}$.

The paper is organized as follows. The next section introduces basic notions of Description Logic. Then Section 3 presents basic definitions and some facts of Formal Concept Analysis. In Section 4 we present our proposal of a description

[3] For explicit distinction with DL, we use here a combined term 'formal concept'.

logic on concept lattices. Section 5 discuses the compatibility of the conceptual semantics of $\mathcal{ALC}$ with the standard one. Topics for further research are discussed in Conclusion.

2 Basics of Description Logics

There are many variants of description logics, but we define only some of them.

Definition 1. *Syntax of every description logic is constructed from disjoint alphabets of concept, role, and object symbols CS, RS, and OS, respectively. The sets of concept terms (or concepts) CT and role terms (or roles) RT are defined by induction. Usually definition admits the following clauses.*

- *(Concept terms)*
  - *the top concept $\top$ and the bottom concept $\bot$ are concept terms;*
  - *any concept symbol is a concept term;*
  - *for any concepts X and Y their union $(X \sqcup Y)$ and intersection $(X \sqcap Y)$ are concept terms;*
  - *for any concept X its complement $(\neg X)$ is a concept term;*
  - *for any role R and any concept X the universal $(\forall R.\ X)$ and the existential $(\exists R.\ X)$ restrictions are concept terms;*
- *(Role terms)*
  - *the top role ∇ and the bottom role $\triangle$ are role terms;*
  - *any role symbol is a role term;*
  - *for any roles R and S their union $(R \sqcup S)$, intersection $(R \sqcap S)$, and composition $(R \circ S)$ are terms;*
  - *for any role R its complement $(\neg R)$, inverse (R^-), and transitive closure (R^+) are role terms.*

Concept and role terms altogether form the set of terminological expressions.

Definition 2.

- *For any concepts X and Y, any roles R and S the following expressions are called terminological sentences: $X \dot\sqsubseteq Y$, $X \doteq Y$, $R \dot\sqsubseteq S$, and $R \doteq S$. A TBox is a set of terminological sentences.*
- *For any concept X, any role R, and any object symbols a and b the following expressions are called assertional sentences: a concept assertion $a : X$ and a role assertion $(a, b) : R$. An ABox is a set of assertional sentences.*
- *A knowledge base is a finite set of terminological and assertional sentences. Every knowledge base consists of an appropriate TBox and ABox.*

Definition 3. *Semantics of any description logic is defined in Kripke-like terminological interpretations. Every terminological interpretation is a pair (D, I), where D is a set (that is called domain) and I is a mapping (that is called interpretation function). This function maps object symbols to elements of D, concept symbols to subsets of D, role symbols to binary relations on D: $I =$*

$I_{OS} \cup I_{CS} \cup I_{RS}$, where $I_{OS} : OS \to D$, $I_{CS} : CS \to 2^D$, and $I_{RS} : RS \to 2^{D\times D}$. The unique name assumption holds for this function: $I(a) \neq I(b)$ for all different object symbols a and b. The interpretation function can be extended to all terminological expressions as follows.

- *(Concept semantics)*
  - $I(\top) = D$ *and* $I(\bot) = \emptyset$;
  - $I(X \sqcup Y)) = I(X) \cup I(Y)$ *and* $I(X \sqcap Y) = I(X) \cap I(Y)$;
  - $I(\neg X) = D \setminus I(X)$;
  - $I(\forall R.\ X) = \{s \in D : \forall t \in D(\textit{if } (s,t) \in I(R) \textit{ then } t \in I(X))\}$,
    $I(\exists R.\ X) = \{s \in D : \exists t \in D((s,t) \in I(R) \textit{ and } t \in I(X))\}$;
- *(Role semantics)*
  - $I(\nabla) = D^2$ *and* $I(\triangle) = \emptyset$;
  - $I(R \sqcup S) = I(R) \cup I(S)$, $I(R \sqcap S) = I(R) \cap I(S)$, $I(R \circ S) = I(R) \circ I(S)$ *(righthand side '$\circ$' is composition of binary relations);*
  - $I(\neg R) = D^2 \setminus I(R)$, $I(R^-) = (I(R))^-$, *and* $I(R^+) = (I(R))^+$ *(righthand side '$^+$' is transitive closure of binary relations);*
  - $I(R!X) = \{(s,t) \in I(R) : s \in I(X)\}$,
    $I(R?X) = \{(s,t) \in I(R) : t \in I(X)\}$.

Definition 4. *Semantics of sentences is defined in terminological interpretations in terms of satisfiability relation as follows:*

- $(D,I) \models a : X$ *iff* $I(a) \in I(X)$;
- $(D,I) \models X \dot{\sqsubseteq} Y$ *iff* $I(X) \subseteq I(Y)$;
- $(D,I) \models X \doteq Y$ *iff* $I(X) = I(Y)$;
- $(D,I) \models (a,b) : R$ *iff* $(I(a), I(b)) \in I(R)$;
- $(D,I) \models R \dot{\sqsubseteq} S$ *iff* $I(R) \subseteq I(S)$;
- $(D,I) \models R \doteq S$ *iff* $I(R) = I(S)$.

This satisfiability relation can be extended on knowledge bases in a natural way: $(D,I) \models KBase$ iff $(D,I) \models \phi$ for every sentence $\phi \in KBase$. In the case of $(D,I) \models KBase$, the terminological interpretation (D,I) is said to be a (terminological) model for the knowledge base $KBase$. Let us say that a knowledge base $KBase$ entails a sentence ψ (and write[4] $Kbase \models \psi$) iff $(D,I) \models \psi$ for every model (D,I) of $KBase$.

Definition 5. *A concept X is said to be coherent (or satisfiable) with respect to a knowledge base $KBase$ iff there exists a terminological model (D,I) for $KBase$ such that $I(X)$ is not empty. A knowledge base $KBase$ is said to be satisfiable iff the top concept $\top$ is coherent with respect to $KBase$.*

Attribute Language with Complements ($\mathcal{ALC}$) [10] is a particular example of description logic. In simple words, $\mathcal{ALC}$ adopts role symbols as the only role terms, concept symbols – as elementary concept terms, and permits 'Boolean' constructs '$\neg$', '$\sqcup$', '$\sqcap$', universal and existential restrictions '$\forall$' and '$\exists$' as the only concept constructs. The formal definition follows.

[4] When $KBase$ is empty then '$\models \psi$' can be written instead of '$\emptyset \models \psi$'.

Definition 6. *$\mathcal{ALC}$ is a fragment of DL that comprises concepts that are defined by the following context-free grammar:*

$$C_{\mathcal{ALC}} ::= \\ CS|\top|\bot|(\neg C_{\mathcal{ALC}})|(C_{\mathcal{ALC}} \sqcup C_{\mathcal{ALC}})|(C_{\mathcal{ALC}} \sqcap C_{\mathcal{ALC}})|(\forall RS.\ C_{\mathcal{ALC}})|(\exists RS.\ C_{\mathcal{ALC}})$$

where metavariables CS and RS represent any concept and role symbols, respectively. Semantics of $\mathcal{ALC}$ is defined in the standard way in accordance with Definition 3.

Many description logics can be defined as extensions of $\mathcal{ALC}$ by concept and/or role constructs. For example, the website [13] uses the following approach: for any collection of concept and/or role constructs $C\&R$, let $\mathcal{ALC}(C\&R)$ be a closure of $\mathcal{ALC}$ that admits all concept and/or role constructs in $C\&R$.

Definition 7. *A variant of DL is a description logic $\mathcal{L}$ with syntax that*

- *contains all concept and role symbols CS and RS,*
- *is closed under concept constructs '$\neg$', '$\sqcup$', '$\sqcap$', '$\forall$' and '$\exists$'.*

From the viewpoint of the above definition, $\mathcal{ALC}$ is the smallest variant[5] of DL.

Definition 8. *Let $\mathcal{L}$ be a variant of DL and $C\&R$ be a collection of concept and/or role constructs. Then let $\mathcal{L}(C\&R)$ be the smallest variant of DL that includes $\mathcal{L}$ and is closed under all constructs in $C\&R$.*

For instance, $\mathcal{ALC}(\neg, -)$ is an extension of $\mathcal{ALC}$ where any role symbol can be negated and/or inverted.

The following definition takes into account DL variants without role constructs for so-called domain restriction and range restriction.

Definition 9. *A concept term is said to be positive if it does not use the concept complement construct. For every DL variant $\mathcal{L}$ the positive fragment of $\mathcal{L}$ comprises all positive concept terms of $\mathcal{L}$ and all role terms of $\mathcal{L}$. The positive fragment of DL variant $\mathcal{L}$ is denoted by $\mathcal{L}^+$.*

For example $\mathcal{ALC}^+$, the positive fragment of $\mathcal{ALC}$, comprises concepts that are defined by the following context-free grammar:

$$C_{\mathcal{ALC}^+} ::= \\ CS|\top|\bot|(C_{\mathcal{ALC}^+} \sqcup C_{\mathcal{ALC}^+})|(C_{\mathcal{ALC}^+} \sqcap C_{\mathcal{ALC}^+})|(\forall RS.\ C_{\mathcal{ALC}^+})|(\exists RS.\ C_{\mathcal{ALC}^+}).$$

The following lattice-theoretic characterization of semantics of concept constructs '$\top$', '$\bot$', '$\sqcup$', '$\sqcap$', '$\forall$', and '$\exists$' is easy to prove.

Proposition 1. *Let (D, I) be a terminological interpretation and $P(D) = (2^D, \emptyset, \subseteq, D, \cup, \cap)$ be the complete lattice of subsets of D. Then $\mathcal{ALC}^+$ semantics enjoys the following properties in $P(D)$:*

[5] Of course, 'smaller' description logics can be defined by means of stronger syntax restrictions.

– *for any concept symbol $I(p)$ is an element of the lattice $P(D)$;*
– $I(\top) = \sup P(D)$ *and* $I(\bot) = \inf P(D)$;
– $I(X \sqcup Y) = \sup(I(X), I(Y))$, *and* $I(X \sqcap Y) = \inf(I(X), I(Y))$;
– $I(\forall R.\ X) = \sup\{S \in P(D) : \forall s \in S \forall t \in D((s,t) \in I(R) \Rightarrow t \in I(X))\}$,
$I(\exists R.\ X) = \sup\{S \in P(D) : \forall s \in S \exists t \in D((s,t) \in I(R)\ \&\ t \in I(X))\}$.

3 Basics of Formal Concept Analysis

Basic Formal Concept Analysis (FCA) definitions below follow monograph [4], but we use a little bit different notation.

Definition 10. *A formal context is a triple (O, A, B) where O and A are sets of 'objects' and 'attributes' respectively, and $B \subseteq O \times A$ is a binary relation connecting objects and attributes. Let us say that a formal context (O, A, B) is homogeneous*[6] *iff $O = A$, i.e. the set of objects coincides with the set of attributes. A symmetric context is a homogeneous context with symmetric binary relation*[7]

Every terminological interpretation (D, I) and every role term R define a formal context $(D, D, I(R))$. It implies that every terminological interpretation (D, I) defines a family of homogeneous formal contexts $(D, D, I(R))$ indexed by role symbols $R \in RS$ or by role terms $R \in RT$.

Vise versa, there is a number of ways to define a terminological interpretation for given formal contexts. For example, if we have a family of formal contexts (O_j, A_j, B_j) indexed by elements of some set J, then we can adopt the set of indexes J as the alphabet role symbols RS, a set of symbols $\{o_j : j \in J\} \cup \{a_j : j \in J\}$ as the alphabet of concept symbols CS, and define a terminological interpretation (D, I) as follows:

– $D = \cup_{j \in J}(O_j \cup A_j)$,
– $I(j) = B_j \subseteq (O_j \times A_j) \subseteq D \times D$ for every $j \in J$,
– $I(a_j) = A_j \subseteq D$ and $I(o_j) = O_j \subseteq D$ for every $j \in J$.

Two major algebraic operations for formal contexts are upper and lower derivations. These operations are used in the definition of a notion of a formal concept, its extent and intent.

Definition 11. *Let (O, A, B) be a formal context.*

– *For every set of objects $X \subseteq O$ its upper derivation $X^{\uparrow}$ is the following set of attributes $\{t \in A :$ for every $s \in O$, if $s \in X$ then $(s,t) \in B\}$, i.e. the collection of all attributes that all objects in X have simultaneously.*
– *For every set of attributes $Y \subseteq A$ its lower derivation $Y^{\downarrow}$ is the following set of objects $\{s \in O :$ for every $t \in A$, if $t \in Y$ then $(s,t) \in B\}$, i.e. the collection of all objects that all attributes in Y have simultaneously.*

[6] 'Homogeneous context' is our own term.
[7] Recall that a binary relation B is symmetric, if $(x,y) \in B \Leftrightarrow (y,x) \in B$ for all x and y. 'Symmetric context' is also our own term.

– *A formal concept is a pair* (Ex, In) *such that* $Ex \subseteq O$, $In \subseteq A$, *and* $Ex^{\uparrow} = In$, $In^{\downarrow} = Ex$; *components* Ex *and* In *of the formal concept* (Ex, In) *are called its extent and its intent respectively.*

Definition 12. *For every formal context* $K = (O, A, B)$

– *let* $FC(K)$ *be the set of all formal concepts over* K, $\top_K$ *be a formal concept* $(O, O^{\uparrow})$, *and* $\bot_K$ *be a formal concept* $(A^{\downarrow}, A)$;
– *let* $\preceq_K$ *be the following binary relation* $FC(K)$:
$$(Ex', In') \preceq_K (Ex'', In'') \text{ iff } Ex' \subseteq Ex'' \text{ and/or[8] } In'' \subseteq In';$$
– *let* $\sup_K$ *be the following operation on subsets of* $FC(K)$:
$$\sup_K\{(Ex_j, In_j) \in FC(K) : j \in J\} \ = \ ((\cup_{j\in J} Ex_j)^{\uparrow\downarrow} \ , \ \cap_{j\in J} In_j);$$
– *let* $\inf_K$ *be the following operation on subsets of* $FC(K)$:
$$\inf_K\{(Ex_j, In_j) \in FC(K) : j \in J\} \ = \ (\cap_{j\in J} Ex_j \ , \ (\cup_{j\in J} In_j)^{\downarrow\uparrow}).$$

The following fact is a part of the Basic Theorem on Concept Lattices [4] (Theorem 3).

Fact 1 *For any formal context* K *an algebraic system* $(FC(K), \preceq_K, \top_K, \bot_K, \sup_K, \inf_K)$ *is a complete lattice.*

The definition below makes sense due to the above theorem.

Definition 13. *For every formal context* K *let the concept lattice* $CL(K)$ *be a complete lattice* $(FC(K), \preceq_K, \top_K, \bot_K, \sup_K, \inf_K)$.

The next fact is a corollary of the Basic Theorem on Concept Lattices.

Fact 2 *Every complete lattice is isomorphic to some concept lattice.*

For example, for any set D the complete lattice $P(D) = (2^D, \emptyset, \subseteq, D, \cup, \cap)$ is isomorphic to the concept lattice of a homogeneous context $K_D^{\neq} = (D, D, \neq)$. A particular isomorphism is a function $\iota : 2^D \to FC(K_D^{\neq})$ that maps every subset $S \subseteq D$ to a formal concept $(S, \overline{S})$. Let us remark that for every $S \subseteq D$, $\iota(\overline{S}) = (\overline{S}, S)$, i.e. permutation of $\iota(S)$. – Let us refer this observation by complement property of the powerset concept lattice.

4 Towards Description Logics on Concept Lattices

One can observe that 'concepts' in Description Logic and in Formal Concept Analysis are of different nature. The former are just subsets of domains in terminological interpretations, the later are compatible pairs of object and attribute sets in formal contexts.

We see two opportunities to combine these two formalizations of 'concepts'. The first opportunity is 'integration' of basic constructions of Formal Concept Analysis to a framework of Description Logic. The second opportunity is to 'interpret' concept terms by formal concepts.

[8] These two conditions are equivalent each other.

The first opportunity has been examined in a workshop paper [11]. But there was criticism after publication of [11], that the paper has adopted a pure set-theoretic approach to concepts. As a consequence of this, a lattice-theoretic structure (that is very special advantage of Formal Concept Analysis) has been lost. Hence it is important to investigate the second opportunity of combination of FCA and DL and develop (in a compatible manner) a description logic directly on concept lattices. Below we present our proposal of a particular variant of description logic on concept lattices.

Syntax of description logics on concept lattices is the same as in Definition 1. Semantics of description logics on concept lattices comes from lattice-theoretic characterization of semantics of 'positive' concept constructs that is given in Proposition 1 and the complement property of the powerset concept lattice. Conceptual interpretation is a formal context provided by an interpretation function.

Definition 14. *Conceptual interpretation is a four-tuple* (O, A, B, I) *where* (O, A, B) *is a context, and* I *is an interpretation function. This function maps object symbols to elements of* O*, attribute symbols to elements of* A*, concept symbols to formal concepts in* $FC(O, A, B)$*, role symbols to binary relations on* $O \cup A$*:* $I = I_{OS} \cup I_{AS} \cup I_{CS} \cup I_{RS}$*, where* $I_{OS} : OS \to O$*,* $I_{AS} : AS \to A$*,* $I_{CS} : CS \to CL(O, A, B)$*, and* $I_{RS} : RS \to 2^{(O\times O)\cup(A\times A)}$*. The unique name assumption holds for this function:* $I(o') \neq I(o'')$ *and* $I(a') \neq I(a'')$ *for different object and attribute symbols* o'*,* o''*,* a'*, and* a''*. A conceptual interpretation* (O, A, B, I) *is said to be homogeneous (symmetric), if* (O, A, B) *is a homogeneous (symmetric respectively) context.*

Definition 15. *Semantics of any description logic on concept lattices is defined by means of conceptual interpretations. Let* (O, A, B, I) *be a conceptual interpretation,* K *be a formal context* (O, A, B)*, and* $CL(K)$ *be a concept lattice* $(FC(K), \preceq_K, \top_K, \bot_K, \sup_K, \inf_K)$*. The interpretation function* I *can be extended to all role terms in a terminological interpretation* $((O \cup A), I)$ *in the standard manner in lines with the definition 3 so that* $I(R)$ *is a binary relation on* $(O \cup A)$ *for every role term* R*. The interpretation function* I *can be extended to all concept terms as follows.*

- $I(\top) = \sup_K FC(K)$ *and* $I(\bot) = \inf_K FC(K)$*;*
- $I(X \sqcup Y) = \sup_K(I(X), I(Y))$*, and* $I(X \sqcap Y) = \inf_K(I(X), I(Y))$*;*
- *Let* K *be a symmetric context and* $I(X) = (Ex, In) \in CL(K)$*.*
  Then $I(\neg X) = (In, Ex)$*.*
- *Let* $I(X) = (Ex', In') \in CL(K)$*. Then*
  - $I(\forall R.\ X) = \sup_K\{(Ex, In) \in CL(K) :$
    $\forall o \in Ex\ \forall a \in In\ \forall o' \in O\ \exists a' \in A$
    $((o, o') \in I(R) \Rightarrow o' \in Ex',\ (a, a') \in I(R),\ \text{and}\ a' \in In')\}$,
  - $I(\exists R.\ X) = \sup_K\{(Ex, In) \in CL(K) :$
    $\forall o \in Ex\ \forall a \in In\ \exists o' \in O\ \forall a' \in A$
    $((a, a') \in I(R) \Rightarrow (o, o') \in I(R),\ o' \in Ex',\ \text{and}\ a' \in In')\}$.

Proposition 2.

1. *For any conceptual interpretation (O, A, B, I), for every positive concept term X, semantics $I(X)$ is an element of $FC(O, A, B)$.*
2. *For any symmetric conceptual interpretation (D, D, B, I), for every concept term X, semantics $I(X)$ is an element of $FC(D, D, B)$.*

Proof. By induction on structure of a (positive) concept. It is trivial in all cases but exploits (a) completeness of concept lattices (i.e. sup always exists), and (b) that permutation of extent and intent of a formal concept is also a formal context in every symmetric context. ■

It follows from the above proposition that in general the positive fragment $\mathcal{ALC}^+$ is 'the smallest' description logic on concept lattices, while $\mathcal{ALC}$ is 'the smallest' description logic on symmetric concept lattices.

5 $\mathcal{ALC}$ on a powerset concept lattice

Let (D, I) be a terminological interpretation. We remarked in section 3 (after fact 2), that the powerset lattice $P(D) = (2^D, \subseteq, \emptyset, D, \cup, \cap)$ is isomorphic to the concept lattice of a homogeneous formal context $K_D^{\neq} = (D, D, \neq)$. A particular isomorphism is a function $\iota : 2^D \to FC(K_D^{\neq})$ that maps every subset $S \subseteq D$ to a formal concept $(S, \overline{S})$. This isomorphism defines conceptual interpretation $(D, D, \neq, (\iota I))$ where (ιI) equals to I on all object symbols and on all role symbols, but on concept symbols it is 'induced' by ι: $(\iota I)(p) = (I(p), \overline{I(p)})$ for every concept symbol p. The following proposition demonstrates that semantics of $\mathcal{ALC}$ in terminological interpretation (D, I) and in conceptual interpretation $CL(D, D, \neq, (\iota I))$ are closely connected.

Proposition 3. *For every $\mathcal{ALC}$ concept term Z and every terminological interpretation (D, I), the following equality holds: $(\iota I)(Z) = \iota(I(Z))$.*

Proof. Since $\iota(I(Z)) = (I(Z), \overline{I(Z)})$, hence we have to prove that $(\iota I)(Z) = (I(Z), \overline{I(Z)})$. The proof proceeds by induction on structure of a concept term.

Basis of induction: Z is $\perp$, $\top$, or a concept symbol. For concept symbols proof follows from the definition of (ιI). For $\perp$ and $\top$ proofs are similar to each other, so let us present just one of them: $(\iota I)(\top) = sup_{K_D^{\neq}}\{(S, \overline{S}) : S \subseteq D\} = (D, \emptyset)$.

Induction hypothesis: let us assume that $(\iota I)(Z) = (I(Z), \overline{I(Z)})$ for every positive concept term that has at most $k \geq 0$ concept constructs $\neg$, $\sqcup$, $\sqcap$, $\forall$, $\exists$.

Induction step: Z is (1) $(\neg X)$, (2) $(X \sqcup Y)$, (3) $(X \sqcap Y)$, (4) $(\forall R.\ X)$, or (5) $(\exists R.\ X)$, where concept terms X and Y contain at most k constructs $\neg$, $\sqcup$, $\sqcap$, $\forall$, and $\exists$. Let us remark that proofs for cases (2) and (3) are very similar to each other, as well as proofs for cases (4) and (5). So we present proofs for cases (1), (2) and (5) only.

Case $Z \equiv (\neg X)$: Let $(\iota I)(X) = (Ex, In)$. Then $(\iota I)(Z) = (In, Ex)$. By induction hypothesis, $In = \overline{I(X)}$ and $Ex = I(X)$. But $I(Z) = \overline{I(X)}$ and $\overline{I(Z)} = \overline{\overline{I(X)}} = I(X)$. Hence $(\iota I)(Z) = (In, Ex) = (\overline{I(X)}, I(X)) = (I(Z), \overline{I(Z)}$.

Case $Z \equiv (X \sqcup Y)$: $(\iota I)(Z) = sup_{K^{\neq}_D}((\iota I)(X)\ ,\ (\iota I)(Y)) =$ (by induction hypothesis) $= sup_{K^{\neq}_D}((I(X), \overline{I(X)})\ ,\ (I(Y), \overline{I(Y)})) = ((I(X) \cup I(Y))\ ,\ (\overline{I(X)} \cap \overline{I(Y)})) = ((I(X) \cup I(Y))\ ,\ (\overline{I(X) \cup I(Y)})) = (I(Z), \overline{I(Z)})$.

Case $Z \equiv (\exists R.\ X)$: Let $(\iota I)(X) = (Ex', In')$. Then $(\iota I)(Z) =$

$= \sup_{K^{\neq}_D}\{(Ex, In) \in CL(K^{\neq}_D) : \forall o \in Ex\ \forall a \in In\ \exists o' \in D\ \forall a' \in D$
$((a, a') \in I(R) \Rightarrow (o, o') \in I(R),\ o' \in Ex'$, and $a' \in In')\} =$
(by induction hypothesis and
the complement property of the powerset concept lattice)
$= \sup_{K^{\neq}_D}\{(S, \overline{S}) : S \subseteq D$ and $\forall o \in S\ \forall a \in \overline{S}\ \exists o' \in D\ \forall a' \in D$
$((a, a') \in I(R) \Rightarrow (o, o') \in I(R),\ o' \in I(X)$, and $a' \in \overline{I(X)}\ )\}$.
Let us denote the righthandside set of formal concepts by $\mathcal{SX}$. Since $\overline{I(\exists R.X)} = I(\forall R.(\neg X))$, then $(I(\exists R.X), \overline{I(\exists R.X)}) = (I(\exists R.X), I(\forall R.(\neg X))) \in \mathcal{SX}$. Hence $(\iota I)(Z) \succeq_{K^{\neq}_D} (I(\exists R.X),\ I(\forall R.(\neg X)))$. At the same time it follows from proposition 1, that $I(\exists R.X)$ is the greatest subset S of D such that $\forall o \in S\ \exists o' \in D\ ((o, o') \in I(R)\ \&\ o' \in I(X))$. Hence $(\iota I)(Z) = (I(\exists R.X), \overline{I(\exists R.X)}) = (I(Z), \overline{I(Z)})$. ■

Informally speaking, the above proposition states that semantics of $\mathcal{ALC}$ on concept lattices that is defined in Definition 15 is compatible with the standard Kripke-like set-theoretic semantics of $\mathcal{ALC}$ that is given in Definition 3. Due to this interpretation of the proposition, we would like to refer it as the compatibility property, and consider as a strong evidence for a naturalness of Definition 15.

Definition 16. *A concept term X is said to be a tautology, if $I(X) = D$ for every terminological interpretation (D, I). A concept term X is said to be a conceptual tautology, if $I(X) = \top_{(O,A,B)}$ for every conceptual interpretation (O, A, B, I). Conceptual tautology problem for a particular description logic is an algorithmic problem to decide for an input concept term (in the given logic) whether it is a conceptual tautology.*

It follows from the compatibility property that every conceptual tautology is a tautology. But we do not know yet whether the inverse implication holds. We also do not know whether the conceptual tautology problem for $\mathcal{ALC}$ or the positive fragment of $\mathcal{ALC}$ is decidable.

6 Conclusion

The present paper presents a variant of a description logic on concept lattices. It expands a research on an emulation-based combination of Description Logic and Formal Concept Analysis that has been started by a workshop paper [11]. The cited paper has discussed how to emulate basic constructs of Formal Concept Analysis in terms of Description Logic. It has been demonstrated in [11] that FCA can be 'integrated' by Description Logic at least from a viewpoint of 'abstract' expressive power. More formally it can be explained as follows.

Assume that $\mathcal{S}$ is a finite collection of set-theoretic (in)equalities written in terms of uninterpreted symbols for individual objects and attributes, for sets of objects and attributes, for formal contexts and concepts, with aid of set-theoretic operations, upper and lower derivative, intent and extent operations. Then $\mathcal{S}$ can be translated to a description logic knowledge base $KBase(\mathcal{S})$ so that $KBase(\mathcal{S})$ is satisfiable iff there $\mathcal{S}$ is 'consistent', i.e. there exists a formal context that realizes all (in)equalities of $\mathcal{S}$ simultaneously. Since the satisfiability problem is decidable for many description logics, the realization problem for collections of (in)equalities of this kind can be solved (as a rule) automatically (i.e. by some algorithm).

The present paper defines a particular variant of description logic on concept lattices that emulates major Description Logic constructs in terms of operations on concept lattices. The paper demonstrates that one of the basic description logics $\mathcal{ALC}$ with the standard set-theoretic semantics can be naturally interpreted on powerset concept lattices. Hence the proposed approach to Description Logic on concept lattices can be considered compatible with the standard Kripke semantics.

Application of $\mathcal{ALC}$/FCA and FC-$\mathcal{ALC}$ to the knowledge representation and processing will be illustrated with examples and discussed in the full version of publication. Algorithmic and reasoning issues[9] for description logics extended by upper and lower derivatives, for $\mathcal{ALC}$ on concept lattices, interpretation of $\mathcal{ALC}$ on concept lattices in terms of standard description logics are topics for further research.

Acknowledgement. We would like to thank Prof. Karl Erich Wolff, Prof. Bernhard Ganter, and Prof. Franz Baader for discussions at early stage of this research. We are especially obliged to anonymous review of our paper who found out a mistake in Proposition 2, and suggested a correction. We follow this correction in the present version.

References

1. Baader F., Ganter B., Sattler U., and Sertkaya B. *Completing Description Logic Knowledge Bases using Formal Concept Analysis.* In Proceedings of the Twentieth International Joint Conference on Artificial Intelligence (IJCAI-07). AAAI Press, 2007.
2. Baader F., D. Calvanese, D. Nardi D.McGuinness, and P. Patel-Schneider, editors. *The Description Logic Handbook: Theory,Implementation and Applications.* Cambridge University Press, 2003.
3. Bull R.A., Segerberg K. *Basic Modal Logic.* Handbook of Philosophical Logic, v.II, Reidel Publishing Company, 1984 (1-st ed.), Kluwer Academic Publishers, 1994 (2-nd ed.), p.1-88.
4. Ganter B., Wille R. *Formal Concept Analysis.* Mathematical Foundations. Springer Verlag, 1996.
5. Hustadt U, Schmidt R.A., and Georgieva L. *A Survey of decidable first-order fragments and description logics.* J. Relational Methods in Computer Science, Vol. 1, 2004, pp. 251-276.

[9] like decidability, complexity, axiomatization, and model checking

6. Kozen D., Tiuryn J. *Logics of Programs.* Handbook of Theoretical Computer Science, v.B, Elsevier and The MIT Press, 1990, p.789-840.
7. *OWL Web Ontology Language Guide: W3C Recommendation 10 February 2004.* W3C (2004-02-10).
8. Rouane A.H., Huchard M., Napoli A., Valtchev P. *A proposal for combining formal concept analysis and description logics for mining relational data.* Lecture Notes in Artificial Intelligence, Vol. 4390, 2007, pp. 51-65.
9. Rudolph S. *Relational Exploration - Combining Description Logics and Formal Concept Analysis for Knowledge Specification.* PhD Dissertation, Universitatsverlag Karlsruhe, 2006.
10. Schmidt-Schauß M., and Smolka G. *Attributive concept descriptions with complements.* J. Artificial Intelligence, Vol. 48, 1991, pp. 1-26.
11. Shilov N.V. Garanina N.O., Anureev I.S. *Combining Two Formalism for Reasoning about Concepts.* Manuscript. Has been accepted for presentation at 20th International Workshop on Description Logics (DL-2007), Brixen-Bressanone, Italy 8-10 June 2007.
12. Wolff K.E. *A first course in Formal Concept Analysis.* F. Faulbaum StatSoft'93, Gustav Fischer Verlag, 1994, p.429438.
13. Zolin E. Complexity of reasoning in Description Logics. Web-resource available at URL `http://www.cs.man.ac.uk/~ezolin/logic/complexity.html`.