# $JClassic^{+}_{\delta\epsilon}$: A Description Logic Reasoning Tool: Application to Dynamic Access Control

Narhimene Boustia
Saad Dahlab University of Blida
Algeria
nboustia@gmail.com

Aicha Mokhtari
USTHB, Algeria
aissani_mokhtari@yahoo.fr

***Abstract*—This paper presents $JClassic^{+}_{\delta\epsilon}$, a description logic with default and exception that is expressive enough to be of practical use, which can reason on default knowledge and handle a "weakened kind of disjunction", allowing a tractable subsumption computation. $JClassic^{+}_{\delta\epsilon}$ is an extension of $JClassic_{\delta\epsilon}$, with the connective *lcs*, which has the same properties as the LCS external operation to compute the least common subsumer of two concepts. $JClassic^{+}_{\delta\epsilon}$ is defined with an intensional semantics. We developp this reasoner to define an access control model, where default and exception connectives are used in representation of context to allow authorization. Consideration of context in access control allows definition of dynamic permissions, for example, permissions given to a doctor in a normal context are not the same that we are in an emergency context.**

***Index Terms*—Description Logic; Defaults and Exceptions; Reasoner; disjunction; access control.**

## I. Introduction

The purpose of access control models is to assign permissions to users. The most interesting would be to have the ability to set dynamic permissions dynamic, i.e., context-dependent.

Context may be unique, as it may be a relationship between a number of situations such as emergency or epidemic risk. For this, we need connectors that allow us to represent this information. The reasoner $JClassic^{+}_{\delta\epsilon}$ has been developed for this purpose. Unlike the work of Ventos et al. [1], [2], we did not stay at the theoretical level, but rather we implemented the reasoner.

Donini [3] shows that concept disjunction makes subsumption computation co-NP-Complete. However, disjunction is very useful for knowledge representation.

In this paper, we present a decription logic-based system, named $JClassic^{+}_{\delta\epsilon}$, whose set of connectives is the union of $JClassic_{\delta\epsilon}$ connectives and the "lcs" connective. The "lcs" connective is a kind of "weakened disjunction" allowing us to preserve a tractable subsumption computation (subsumption in $JClassic_{\delta\epsilon}$ has been proved correct, complete and tractable in [4]).

The "lcs" connective has the same properties as the LCS external operation introduced by Borgida et al. [5] which computes the least common subsumer of two concepts. It was introduced by Ventos et al. in Classic to allow disjunction with a reasonable computation [1], [2].

Because of $JClassic_{\delta\epsilon}$ has been given an intensional semantics, $JClassic^{+}_{\delta\epsilon}$ is provided with an intentional semantics (called $\mathcal{CL}^{+}_{\delta\epsilon}$) based on an algebraic approach. For this, we have first to build an equational system which highlights the main properties of the connectives. The equational system allows to define axiomatically the notion of LCS.

In this paper, we first present our system $JClassic^{+}_{\delta\epsilon}$, we give then definition of "lcs". We finally illustrate the use of this tool for access control.

In access control, permission are given to user depending on the actual context. The context can be that the default one in our case represented by the default connector ($\delta$), it can be an exception to the current context represented by the connector Exception ($\epsilon$), as it can be a conjunction or a disjunction of several contexts.

To this end, the reasoner $JClassic_{\delta\epsilon}$ has been enriched by the operator of minimum disjontion in order to have a good level of expressiveness with a polynomial complexity.

## II. $JClassic^{+}_{\delta\epsilon}$

$JClassic^{+}_{\delta\epsilon}$ is an non monotonic reasoner based on description logic with default and exception [6] which allows us to deal with default and exceptional knowledge.

The set of connectives of $JClassic^{+}_{\delta\epsilon}$ is the union of the set of connectives of $\mathcal{AL}_{\delta\epsilon}$ [6] presented in [4], [7], [8] and the connective "lcs".

The connective $\delta$ intuitively represents the common notion of default. For instance, having $\delta Fly$ as a conjunct with Animal in the definition of the concept **Bird** states that birds generally fly.

The connective $\epsilon$ is used to represent a property that is not present in the description of the concept or of the instance but that should be. For instance, the definition of **Penguin** in $JClassic^{+}_{\delta\epsilon}$ is $Penguin \equiv Bird \sqcap Fly^{\epsilon}$. The $Fly^{\epsilon}$ property expresses the fact that fly should be in the definition of Penguin since it is a bird. The presence of $Fly^{\epsilon}$ in the definition of Penguin makes it possible to classify Penguin under the concept Bird.

Formally, the subsumption relation uses an algebraic semantics. The main interest of this approach is the introduction of the definitional point of view of default knowledge: from the definitional point of view, default knowledge can be part of concept definition whereas from the inheritance one it is

 ISBN: 978-1-61208-159-5 

only considered as a weak implication. A map between the definition of concept and its inherited properties is described. This combinating of definitional and inheritance levels improves the classification process. Figure 1 describes the general architecture of our system.

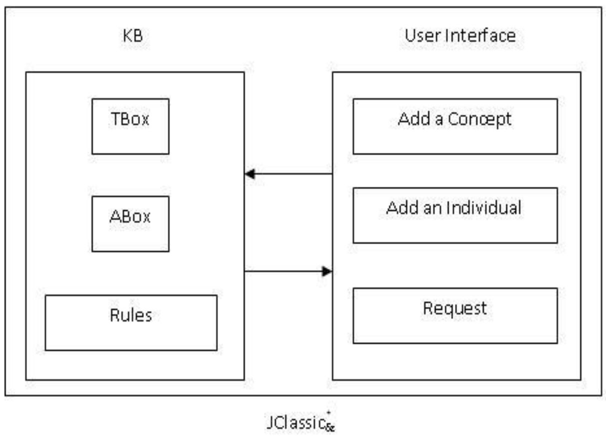


Fig. 1. Architecture of $JClassic_{\delta\epsilon}$

In this section, we first present the syntax of our system, we then give details about its algebraic semantic and we conclude this section by presenting the mechanism of inference in our tools.

### A. Syntax of $JClassic^{+}_{\delta\epsilon}$

The set of connectives of $JClassic^{+}_{\delta\epsilon}$ is the union of the set of connectives of $\mathcal{CL}_{\delta\epsilon}$ [6] and the connectives $\delta$ and $\epsilon$. $JClassic^{+}_{\delta\epsilon}$ is defined using a set **R** of primitive roles, a set **P** of primitive concepts, the constant $\perp$ (Bottom) and $\top$ (Top) and the following syntax rule (C and D are concepts, P is a primitive concept, R is a primitive role).

$C, D \rightarrow \top$ the most general concept
$\mid \perp$ the most specific concept
$\mid P$ primitive concept
$\mid C \sqcap D$ concept conjunction
$\mid \neg P$ negation of primitive concept (This restriction to primitive concept in the negation is a choice to avoid the untractability)
$\mid \forall r : C$ C is a value restriction on all roles R
$\mid \delta C$ default concept
$\mid C^{\epsilon}$ exception to the concept
$\mid ClcsD$ concept disjunction

$\delta$ and $\epsilon$ are unary connectives, $\sqcap$ is a binary conjunction connective and $\forall$ enables universal quantification on role values.

### B. Semantic of $JClassic^{+}_{\delta\epsilon}$

We endow $JClassic^{+}_{\delta\epsilon}$ with an intentional algebraic semantic denoted $\mathcal{CL}^{+}_{\delta\epsilon}$.

This framework covers the different aspects of the formal definition of concepts and subsumption in our language. The calculating of denotations of concepts in $\mathcal{CL}^{+}_{\delta\epsilon}$ is used in computing subsumption in the algorithm $Sub_{\delta\epsilon}$. $\mathcal{CL}^{+}_{\delta\epsilon}$ allows first to show that $Sub^{+}_{\delta\epsilon}$ is correct and complete and secondly to give a formal characterization of calculation of subsumption used in the implementation of $JClassic^{+}_{\delta\epsilon}$.

*1) EQ: an equational system for $JClassic^{+}_{\delta\epsilon}$:* In order to serve as the basis for the definition of an algebraic semantics, an equational system EQ is defined. From a descriptive point of view, the calculation of subsumption consists on the comparison of terms through the equational system **EQ**. This system fixes the main properties of the connectives and is used to define an equivalence relation between terms and then to formalize the subsumption relationship.

$\forall A, B, C \in JClassic^{+}_{\delta\epsilon}$:
01: $(A \sqcap B) \sqcap C = A \sqcap (B \sqcap C)$
02: $A \sqcap B = B \sqcap A$
03: $A \sqcap A = A$
04: $\top \sqcap A = A$
05: $\perp \sqcap A = \perp$
06: $(\forall R : A) \sqcap (\forall R : B) = \forall R : (A \sqcap B)$
07: $\forall R : \top = \top$
08: (A lcs B) lcs C = A lcs (B lcs C)
09: A lcs B = B lcs A
10: A lcs A = A
11: A lcs $\top$ = $\top$
12: A lcs $\perp$ = A
13: $(\delta A)^{\epsilon} = A^{\epsilon}$
14: $\delta(A \sqcap B) = (\delta A) \sqcap (\delta B)$
15: $A \sqcap \delta A = A$
16: $A^{\epsilon} \sqcap \delta A = A^{\epsilon}$
17: $\delta\delta A = \delta A$

Axioms 01 to 07 are classical; they concern description logic connectives properties [9], [10]. Axioms 08 to 12 concern the connective "lcs". The following ones correspond to $\mathcal{AL}_{\delta\epsilon}$ connectives properties[6], i.e., properties of $\delta$ and $\epsilon$ connectives.

**Descriptive Subsumption:**

We denote $\sqsubseteq_d$ for descriptive subsumption. $\sqsubseteq_d$ is a partial order relation on terms. Equality (modulo the axioms of EQ) between two terms is denoted $=_{EQ}$. $=_{EQ}$ is a congruence relation which partitions the set of terms, i.e., $=_{EQ}$ allows to form equivalence classes between terms. We define the descriptive subsumption using the congruence relation and conjunction of concepts as follow:

*Definition 1:* (Descriptive Subsumption)

Let C and D two terms of $JClassic^{+}_{\delta\epsilon}$, $C \sqsubseteq_d D$, i.e., D subsume descriptively C, iff $C \sqcap D =_{EQ} C$.

From an algorithmic point of view, terms are not easily manipulated through subsumption. We adopt a structural point of view closer to the algorithmic aspect of computing subsumption. This allows us to first formalize calculation of subsumption in the implementation of $JClassic^{+}_{\delta\epsilon}$ and secondly to endow $JClassic^{+}_{\delta\epsilon}$ with an intensional semantics.

To define the subsumption relation between two concepts using their description, we need to compare them. For this, concepts are characterized by a normal form of their properties rather than by the set of their instances.

*2) Normal Form of concept:* We present in this section the structural point of view for the subsumption in $JClassic^{+}_{\delta\epsilon}$. This point of view has two main advantages: it is very close to the algorithmic aspects and is a formal framework to validate the algorithmic approach which is not the case description graph.

We define a structural concept algebra $\mathcal{CL}^{+}_{\delta\epsilon}$ which is used to give an intensional semantic in which concepts are denoted by the normal form of their set of properties. The structural point of view of subsumption consist then to compare the normal forms derived by applying a homomorphism from set of terms of $JClassic^{+}_{\delta\epsilon}$ to elements of $\mathcal{CL}^{+}_{\delta\epsilon}$.

**$\mathcal{CL}^{+}_{\delta\epsilon}$: an intensional semantic for $JClassic^{+}_{\delta\epsilon}$**

From the class of CL-algebra, we present a structural algebra $\mathcal{CL}^{+}_{\delta\epsilon}$ which allows to endow $JClassic^{+}_{\delta\epsilon}$ with an intentional semantic.

Element of $\mathcal{CL}^{+}_{\delta\epsilon}$ are the canonical intentional representation of terms of $JClassic^{+}_{\delta\epsilon}$ (i.e., Normal form of the set of their properties). We call an element of $\mathcal{CL}^{=}_{\delta\epsilon}$ normal forms.

Definition of $\mathcal{CL}^{+}_{\delta\epsilon}$ means definition of a homomorphism h which allows to associate an element of $\mathcal{CL}^{+}_{\delta\epsilon}$ to a term of $JClassic^{+}_{\delta\epsilon}$.

Using the equational system, we calculate for each concept a structural denotation which is a single normal form of this concept. The calculation of a normal form from a description of a concept can be seen as a result of term "rewriting" based on the equational system EQ.

The normal form of a concept defined with description T (noted nf(T)) is a couple $\langle t_\theta, t_\delta \rangle$ where $t_\theta$ contains strict properties of T and $t_\delta$ the default properties of T.

$t_\theta$ and $t_\delta$ are 3-tuple of the form ($\pi$,r,$\epsilon$) with:

$\pi$: is a set of primitive concepts in description T.

r: has the form $\langle R, c \rangle$ where :

R: is the name of Role.

c: is the normal form of C, if the description contains the property $\forall R : C$.

$\epsilon$: set of 3-tuple with the form ($\pi$, r, $\epsilon$).

**Example:**

The normal form of concept $A \equiv B \sqcap C \sqcap \delta D$ is:

nf (A) = $(\langle \{B, C\}, \emptyset, \emptyset \rangle, \langle \{B, C, D\}, \emptyset, \emptyset \rangle)$.

**Structural Subsumption:**

Two terms C and D of $JClassic^{+}_{\delta\epsilon}$ are structurally equivalent iff their normal forms are equal. We denote $\sqsubseteq_s$ for structural subsumption. $\sqsubseteq_s$ is a partial order relation.

The structural equality of two terms of $JClassic^{+}_{\delta\epsilon}$ is noted $=_{CL}$. $=_{CL}$ is a congruence relation as $_{EQ}$ in descriptive subsumption.

We define the structural subsumption using the congruence relation and conjunction of concepts as follow:

*Definition 2:* **(Structural Subsumption)**

Let C and D two terms of $JClassic^{+}_{\delta\epsilon}$, $C \sqsubseteq_s D$; i.e., D subsume structurally C, iff $C \sqcap D =_{CL} C$.

*Theorem 1:* **(Equivalency between descriptive subsumption and structural subsumption)**

Let C and D two terms of $JClassic^{+}_{\delta\epsilon}$, $C \sqsubseteq_s D \Leftrightarrow C \sqsubseteq_d D$.

To infer new knowledge in this system, the susbsumption relation is used. In the next section, we outline the subsumption algorithm handling defaults and axceptions named Sub$_{\delta\epsilon}$.

## III. Inference in $JClassic^{+}_{\delta\epsilon}$

There are several reasoning services to be provided by a DL- system. We concentrate our work on the following basic ones, which are Classification of concepts (TBox) and instance checking (ABox). These two services basically use the subsumption relation.

### *A. The Subsumption Relation*

Borgida [5] defines the subsumption based on a set theoretic interpretation as follow: "The concept C subsume D, if and only if the set of instances of C include or is equal to a set of instances of D".

However, the general principle of computing subsumption between two concepts is to compare their sets of properties, not their sets of instances.

For this, we use an intensional semantics which is closer to the algorithmic aspects of computing subsumption, and this by defining a normal form of description called descriptive normal form.

**Algorithm of Computing Subsumption** $Sub_{\delta\epsilon}$

$Sub_{\delta\epsilon}$ is an algorithm of computing subsumption of the form Normalization- Comparison. It is **consists** of two steps, first, the normalization of description, and then a syntactic comparison of the obtained normal forms.

Let C and D be two terms of $JClassic^{+}_{\delta\epsilon}$. To answer the question "Is C subsumed by D?" we apply the following procedure. The normal forms of C and "C $\sqcap$ D" are calculated with the procedure of normalisation.

There are two steps in the comparison. We compare the strict parts of the two concepts. If these are equal, then we compare the default parts. If the two normal forms are equal, the algorithm returns "Yes". It returns "No" otherwise.

The completeness, correctenness and the polynomial computation of $JClassic_{\delta\epsilon}$ have been proved in [4].

We detailed in the next section the connective "lcs".

## IV. The computation of "lcs"

The least common subsumer has been introduced in description logic by Borgida et al. [5] as an external operation to compute the LCS of two concepts.

The LCS of two concepts A and B belonging to a language L is the most specific concept in L that subsumes both A and B.

*Definition 3:* Let L a terminological language, $\sqsubseteq$ the notation of subsumption relation in L

 ISBN: 978-1-61208-159-5 

LCS: L $\times$ L $\rightarrow$ L

LCS(A,B) $\rightarrow$ C $\in$ L iff:

$A \sqsubseteq B$ and $B \sqsubseteq C$ (C subsume both A and B),

$\nexists$ D $\in$ L such that $A \sqsubseteq D, B \sqsubseteq D$ and $D \subseteq C$ (i.e., there is no common subsumers to A and B which is subsumed strictly by C)

The next algorithm is to compute the LCS where input are the normal form of two concepts $A_1$ and $A_2$ and the output is the LCS of $A_1$ and $A_2$.

Let *a* and *b* two normal forms *A* and *B* with *a* and $b \neq b_0$ ($b_0$ is the normal form of $\perp$).

**Algorithm 1** LCS

**Require:** a=$\prec a_\theta$,$a_\theta \succ$C and b=$\prec b_\theta$,$b_\theta \succ$C two normal forms of A and B.

**Ensure:** c=$\prec c_\theta$,$c_\theta \succ$C the normal form of LCS(A,B)

$c_{\theta\pi} \leftarrow a_{\theta\pi} \cap b_{\theta\pi}$
$c_{\theta r} \leftarrow \emptyset$
**for all** $\prec$r, d $\succ$ $\in$ a$_{\theta r}$ **do**
  **if** $\exists$ $\prec$r, e $\succ$ $\in$ b$_{\theta r}$ **then**
    f $\leftarrow$ LCS(d,e)
    $c_{\theta r} \leftarrow c_{\theta r} \cup$ $\prec$r, f $\succ$
  **end if**
**end for**
$c_{\theta\epsilon} \leftarrow a_{\theta\epsilon} \cap b_{\theta\epsilon}$
$c_{\delta\pi} \leftarrow a_{\delta\pi} \cap b_{\delta\pi}$
$c_{\delta r} \leftarrow \emptyset$
**for all** $\prec$r, d $\succ$ $\in$ a$_{\delta r}$ **do**
  **if** $\exists$ $\prec$r, e $\succ$ $\in$ b$_{\delta r}$ **then**
    f $\leftarrow$ LCS(d,e)
    $c_{\delta r} \leftarrow c_{\delta r} \cup$ $\prec$r, f $\succ$
  **end if**
**end for**
$c_{\delta\epsilon} \leftarrow a_{\delta\epsilon} \cap b_{\delta\epsilon}$

Our system can be used in differents application; we choose to use to model an access control model.

## V. Application to Access Control

To show how we can use our description logic-based system and how we can infer a new knowledge, we define a knowledge base adapted to formalize a dynamic access control model.

In this model, authorization to subject are assigned depending on context. We consider first that the context is by default normal, and we represent it using the operator of default ($\delta$). Then, each change of context is considered as an exception to the current context, this change is represented by the operator of exception ($\epsilon$). We give, as an example, one ABox of a medical information system to show how authorization can be deduced.

- Using the instances in Table 1, the system infers that in organization **X**, each person who play the role of **Patient** is by default permitted to **consult** his **Med-rec** and add this instance to the ABox : $\delta Permission(P1)$.

where:

$\delta Permission(P1) \sqsubseteq PermisionAv.Activity(Consult) \sqcap PermissionR.Role(Patient) \sqcap PermissionV.View(Med-rec) \sqcap PermissionOr.Organization(X)$

Using the previous ABox, we show how deduction can be done in differents contexts.

- **Access control in a default context**: Suppose that user Marc wants to read Med-rec1; can he obtains that privilege?
  We know that:
  -Marc plays role of Patient in organization X: Employ(E1);
  -and, Med-rec1 is an object used in the view Med-rec: Use(U1);
  -and, Read is considered as a consultation activity: Consider(C1);
  -and finally, by default, in organization X, each person who plays the role of Patient is permitted to consult his Medical records, when Normal context is true: $\delta Permission(P1)$.
  Formally, we write:
  $Employ(E1) \sqcap Use(U1) \sqcap Consider(C1) \sqcap \delta Permission(P1)$
  Using security rules, we can deduce that the preceding proposition subsumes $\delta Is-permitted(I1)$.
  where:
  $Is - permitted(I1) \sqsubseteq Is - permittedAc.Action(Read) \sqcap Is - permittedS.Subject(Marc) \sqcap Is - pemittedO.Object(Med-rec1)$
  And because $Is - permitted(I1) \sqsubseteq \delta Is - permitted(I1)$, we can deduce that Marc is permitted to read his medical records.
- **Access control if context "Serious-disease" is true**: Suppose that Marc has a serious disease and he wants to read his medical records; did he have this right?
  In the context **Serious-disease**, the system deduce a new instance P2 and we add to the ABox the next rule:
  $Permision(P1)^\epsilon \sqsubseteq \delta Permission(P2)$
  We know that:
  -Marc plays role of Patient in organization X: Employ(E1);
  -and, Med-rec1 is an object used in the view Med-rec: Use(U1);
  -and, Read is considered as a consultation activity: Consider(C1);
  -and finally, by default, in organization X, each person who plays the role of Patient is permitted to consult his Medical records, when context Serious-disease is true: $\delta Permission(P2)$.
  We obtain:
  $Employ(E1) \sqcap Use(U1) \sqcap Consider(C1) \sqcap \delta Permission(P2)$
  $\equiv Employ(E1) \sqcap Use(U3) \sqcap Consider(C1) \sqcap \delta Permission(P1)^\epsilon$

| **ABox** |
| --- |
| *Organization(X)*; |
| *Role(Patient)*; |
| *Subject(Marc)*; |
| *View(Med-rec)*; |
| *Object(Med-rec1)*; |
| *Action(Read)*; |
| *Activity(Consult)*; |
| $Employ(E1) \sqsubseteq EmployS.Subject(Marc) \sqcap EmployR.Role(Patient)$ |
| $\sqcap EmployOr.Organization(X)$; |
| $Use(U1) \sqsubseteq UseO.Object(Med-rec1) \sqcap UseV.view(Med-rec)$ |
| $\sqcap UseOr.Organization(X)$; |
| $Consider(C1) \sqsubseteq ConsiderAc.Action(Read) \sqcap ConsiderAv.Activity(Consult)$ |
| $\sqcap ConsiderOr.Organization(X)$; |

TABLE I
ABOX

We know that $A^\epsilon \equiv \delta A^\epsilon$, we obtain:
$\equiv Employ(E1) \sqcap Use(U1) \sqcap Consider(C1) \sqcap Permission(P1)^\epsilon$
Using security rules, we can deduce that the precedent proposition subsumes $Is-permitted(I1)^\epsilon$.
And, because $Is - permitted(I1) \not\sqsubseteq Is - permitted(I1)^\epsilon$, we cannot deduce Is-permitted(I1). Therefore Marc is not permitted to read his medical records when he has a serious disease.

Our policy language allows us to have more than one exception in a context. Exception at an even level cancel the effects of exceptions and therefore infers the property by default [6].

Supose that we have a disjunction of context, for example ”context default or context serious disease”, here we can use the connective ”lcs” to deduce permission

- **lcs(default context, context of serious disease)**: Suppose that user Marc wants to read Med-rec1; can he obtain that privilege?
  We know that:
  -Marc plays role of Patient in organization X: Employ(E1);
  -and, Med-rec1 is an object used in the view Med-rec: Use(U1);
  -and, Read is considered as a consultation activity: Consider(C1);
  and we have the two previous permissions Permission(P1) and Permission(P2), defined respectively for the default context and context of serious disease.
  We obtain:
  $Employ(E1) \sqcap Use(U1) \sqcap Consider(C1) \sqcap lcs(\delta Permission(P1), \delta Permission(P2))$
  $\equiv Employ(E1) \sqcap Use(U3) \sqcap Consider(C1) \sqcap lcs(\delta Permission(P1), \delta Permission(P1)^\epsilon)$
  using lcs properties, we obtain:
  $\equiv Employ(E1) \sqcap Use(U1) \sqcap Consider(C1) \sqcap \delta Permission(P1)$
  Using security rules, we can deduce that the precedent proposition subsumes $\delta Is-permitted(I1)$.
  And, because $Is - permitted(I1) \sqsubseteq \delta Is - permitted(I1)$, we can deduce Is-permitted(I1). Therefore Marc is permitted to read his medical records when he has a serious disease or when the context is normal.

## VI. CONCLUSION

The work presented in this paper has led to the definition of a new system based on description logic expressive enough to be used as part of an application and to represent default knowledge and exceptional knowledge. The $JClassic^+_{\delta\epsilon}$ highlights the interests and the relevance of defaults in conceptual definition. For the $JClassic^+_{\delta\epsilon}$ language, we have given a set of axioms outlining the essential properties of the connectives from this definitional point of view: property links default characteristics to exceptional or strict ones. This set of axioms induces a class of $CL^+_{\delta\epsilon}$-algebra of which the terms are concept descriptions. Using the conjunction connectives $\sqcap$ and ”lcs”, the set of concept can be partially ordered w.r.t the equational system (descriptive subsumption in free algebra). $JClassic^+_{\delta\epsilon}$ is defined with a universel algebraic corresponding to a denotational semantic, where terms are denoted exactly by sets of strict and default properties.

This system consists of three modules: a module for representing knowledge, a module to use that knowledge and a module to update knowledge. The module which allows to use knowledge is endowed with a subsumption algorithm which is correct, complete and polynomial.

In our work, the description logic is endowed with an algebraic intensional semantics, in which concepts are denoted by a normal form of all their properties. These normal forms (i.e., elements of the intensional semantic) are used directly as an input to the algorithm of subsumption and algorithm of deductive inferences.

To show how we can use our system, we applied it to access control. We developed a contextual access control model in which authorization are assigned to users depending on context. We represent this kind of authorization using the two operators of default ($\delta$) and exception ($\epsilon$).

An interseting topic for future research is to extend our tool to take into account spacial-temporal context to make our system more expressive with keeping a reasonable complex-

 ISBN: 978-1-61208-159-5 

ity. We also envisage to explore other appropriate and real applications.

## References


[1] V. Ventos and P. Brésellec. Least Common Subsumption as a connective. In *Proceeding of International Workshop on Description Logic*, Paris, France, 1997.

[2] V. Ventos, P. Brésellec and H. Soldano. Explicitly Using Default Knowledge in Concept Learning: An Extended Description Logics Plus Strict and Default Rules. In *Logic Programming and Nonmonotonic Reasoning, 6th International Conference, LPNMR 2001, pp. 173-185*, Vienna, Austria, September 17-19, 2001.

[3] F.M. Donini, M. Lenzerini, D. Nardi, and W. Nutt. The complexity of concept languages. In *Principles of language representation and reasoning: second international conference, pp. 151-162*, 1991.

[4] N. Boustia and A. Mokhtari. A dynamic access control model. In *Applied Intelligence Journal, DOI 10.1007/s10489-010-0254-z*, 2010, To appear.

[5] A. Borgida and P.F. Patel-Schneider Complete algorithm for subsumption in the CLASSIC description logic. *Artificial Intelligence Research, vol 1, pp. 278-308*, 1994.

[6] F. Coupey and C. Fouqueré. Extending conceptual definitions with default knowledge. *Computational Intelligence, vol 13, no 2, pp. 258-299* , 1997.

[7] N. Boustia and A. Mokhtari. Representation and reasoning on ORBAC: Description Logic with Defaults and Exceptions Approach. In *Workshop on Privacy and Security - Artificial Intelligence (PSAI), pp. 1008-1012, ARES'08, Spain*, 2008.

[8] N. Boustia and A. Mokhtari. $DL_{\delta\epsilon} - OrBAC$: Context based Access Control. In *WOSIS'09, pp. 111-118, Italy*, 2009.

[9] R.J. Brachman, D.L. McGuinness, P.F. Patel-Schneider, L. Alperin Resnick, and A. Borgida. Living with CLASSIC: When and How to Use a KL-ONE-Like Language. In *John Sowa, ed., Principles of Semantic Networks: Explorations in the representation of knowledge, pp. 401-456*, Morgan-Kaufmann: San Mateo, California, 1991.

[10] R.J. Brachman, D.L. McGuinness, L. Alperin Resnick, and A. Borgida. CLASSIC: A Structural Data Model for Objects. In *Proceedings of the 1989 ACM SIGMOD International Conference on Management of Data, pp. 59-67*, June 1989.


 ISBN: 978-1-61208-159-5 