Four-valued Knowledge Augmentation for Representing Structured Documents

Mounia Lalmas and Thomas Roelleke

Department of Computer Science, Queen Mary, University of London
London, E1 4NS, England
{mounia,thor}@dcs.qmul.ac.uk
http://qmir.dcs.qmul.ac.uk

Abstract. Structured documents are composed of objects with a content and a logical structure. The effective retrieval of structured documents requires models that provide for a content-based retrieval of objects that takes into account their logical structure, so that the relevance of an object is not solely based on its content, but also on the logical structure among objects. This paper proposes a formal model for representing structured documents where the content of an object is viewed as the knowledge contained in that object, and the logical structure among objects is captured by a process of knowledge augmentation: the knowledge contained in an object is augmented with that of its structurally related objects. The knowledge augmentation process takes into account the fact that knowledge can be incomplete and become inconsistent.

1 Introduction

We view a structured document as being composed of *objects* with a content and a logical structure [1, 9]. These objects correspond to the document components. The content refers to the content of objects. The logical structure refers to the way structured documents are organised. The *root* object, which is unique, embodies the whole document. *Atomic* objects are document components that are not composed of other components. All other objects are referred to as *inner* objects.

With the widespread development of structured document repositories (e.g. CD-ROM, the Internet and digital libraries), there is a need to develop retrieval models that dynamically return objects of varying granularity (e.g. [4, 6–8]). A retrieval result may then consist of entry points to a same document: an entry to the root object when the entire document is relevant to a query, an entry to an atomic object when only that object is relevant to the query, or an entry to an inner object when that object and its structurally related objects are relevant to the query. Returning objects of varying granularity requires sophisticated models that go beyond the purely term-based models of classical information retrieval, so that the relevance of an object is not solely based on its content, but also takes into account the logical structure among objects.

This paper proposes a formal model for representing structured documents that allows the retrieval of objects of varying granularity[1]. The model exploits the content and the logical structure among objects to arrive at a representation of the structured document. The content of an object is viewed as the *knowledge* contained in that object, and the structure among objects is captured by a process of *knowledge augmentation*: the knowledge contained in an object is *augmented* with the knowledge contained in its structurally related objects. The knowledge augmentation process is a combination of knowledge specifically defined to provide for a representation of the object that is based on its own content and that of its structurally related objects. It is this representation that can then be used to estimate the relevance of the object to a query. This paper is concerned with the definition of the knowledge augmentation process, which takes into account the fact that knowledge can be incomplete and become inconsistent. Incompleteness is due to the fact that knowledge that is not explicit in an object should not be considered false. Inconsistent arises when two objects upon which the augmentation process is based contain knowledge that is contradictory.

The outline of this paper is as follows. Section 2 describes the model for representing content and structural knowledge of objects. Section 3 and 4 present two approaches for describing the knowledge augmentation process. Section 5 shows that the two approaches are equivalent. Section 6 concludes and discusses future work.

2 Representing content and structural knowledge

program	::=	clause | clause program
clause	::=	fact | context
fact	::=	truth-list proposition
proposition	:=	NAME
truth-list	::=	'1' | '0/1' | '0/0/1' | '0/0/0/1'
context	::=	NAME '[' program ']'
NAME	::=	[a-z][a-zA-Z0-9_]*

Fig. 1. Syntax of programs

The syntax used to characterise the content and structural knowledge of a structured document is given in Figure 1. A structured document is described by a *program*, which is a set of *clauses*, which are either *facts* or *contexts*. A *context* consists of an object identified by an object name, called the *context name*, in which a program is nested. We use "this" to refer to the context name associated with the global program (i.e. representing the structured document). The set of object names is called $\mathcal{C}$. A *fact* is a *proposition* preceded by *truth-list*. We call Φ the set of propositions. *truth-list* represents the four truth values: "1" means *true*, "0/1" means *false*, "0/0/1" means *inconsistent*, and "0/0/0/1" means *unknown*. In this paper, we use "proposition" for "1 proposition".

[1] A full version of this paper, which includes examples and proofs, can be found at http://www.dcs.qmul.ac.uk/~mounia/CV/pubconf.html/KA.pdf.

Let d and s be context names and "sailing" be a proposition. A context clause is called an *atomic context* clause when it contains only facts, but no context clauses; e.g., "d[sailing]" is an atomic context clause expressing that the content of the object modelled by the context name d is described by the proposition "sailing". Otherwise, a context clause is called a *structured context* clause; e.g., "d[s[sailing]]" is a structured context clause expressing that the object represented by d is composed of the object represented by s, which content is described by the proposition "sailing". We refer to s as *a subcontext* of d, and d as *the supercontext* of s. A supercontext can have several subcontexts, whereas a subcontext has exactly one supercontext. Atomic context clauses characterise atomic objects, whereas structured context clauses characterise non-atomic objects (i.e. inner and root objects). A function $sub : \mathcal{C} \rightarrow \wp(\mathcal{C})$ yields the set of subcontexts of a supercontext for a given program.

The semantics of programs is defined upon an *interpretation structure* M and an *interpretation function* $\models$. We consider contexts as "agents" that possess *knowledge* concerning their content and their structure and use a Kripke structure (see [3, 5]) to define the semantics of programs. Different from [5], we use the terminology "context" instead of "agent" and we consider four truth values [2]. We first define an interpretation structure with respect to a set of propositions Φ and a set of context names $\mathcal{C}$.

Definition 1. (Interpretation structure M): *An* interpretation structure *M for a set Φ of propositions and a set $\mathcal{C}$ of context names is a tuple $M = (W, \pi, \mathcal{R})$ where*

- *$W := \{w_d : d \in \mathcal{C}\} \cup \{w_{slot}\}$ is a finite set of possible worlds. For each context name, a possible world is defined. W includes the possible worlds w_{this} for the global context "this" and w_{slot} to model the access to "this".*
- *$\pi : W \rightarrow (\Phi \rightarrow \{\text{true}, \text{false}, \text{inconsistent}, \text{unknown}\})$ is a function that yields a truth value assignment for all world w in W, which is a function that assigns a truth value to each proposition in Φ, where the four truth values are defined as* true$:=\{t\}$*,* false$:=\{f\}$*,* inconsistent$:=\{t, f\}$*, and* unknown$:=\{\}$*.*
- *$\mathcal{R} := \{R_d : \text{d} \in \mathcal{C}\}$ is a finite set of binary relations on W, called accessibility relations, one for each context name. For any supercontext* d *and subcontext* s*, i.e.* $\text{s} \in sub(\text{d})$*, $\{(w_d, w_s)\} \in R_s$ where R_s is the accessibility relation associated to* s*, and w_d and w_s are the possible worlds associated with* d *and* s*, respectively. We say that context* s accesses *or* reaches *world w_s from world w_d. The accessibility relation for the global context "this" is defined as $R_{this} := \{(w_{slot}, w_{this})\}$.*
- *The function $R_d(w) := \{w' | (w, w') \in R_d\}$ for $w \in W$ and $R_d \in \mathcal{R}$ yields the set of worlds that can be reached from world w by the context* d*.*

The semantics of the content and structural knowledge of objects is based upon considering context names as modal operators, referred to as *knowledge modal operators*. The atomic context clause "d[sailing]" becomes interpreted as "d knows sailing" and captures the content knowledge of the corresponding object. The structured context clause "d[s[sailing]]" becomes interpreted as "d knows that s knows sailing" and captures the structural knowledge of the object represented by d. With this interpretation in mind, we define the interpretation function $\models$ that assigns truth values to propositions, facts, atomic and structured context clauses, and programs with respect to the interpretation structure defined above. The following and all definitions in this paper are based on an interpretation structure $M = (W, \pi, \mathcal{R})$ as defined above.

Definition 2. (Interpretation function $\models$): *Let $\varphi \in \Phi$ and $w \in W$. Let* d, s $\in \mathcal{C}$. *Let $R_d \in \mathcal{R}$ be the accessibility relation corresponding to* d. *The interpretation of facts is defined as follows:*

$$(M, w) \models 1\ \varphi :\iff \pi(w)(\varphi) = \text{true}$$
$$(M, w) \models 0/1\ \varphi :\iff \pi(w)(\varphi) = \text{false}$$
$$(M, w) \models 0/0/1\ \varphi :\iff \pi(w)(\varphi) = \text{inconsistent}$$
$$(M, w) \models 0/0/0/1\ \varphi :\iff \pi(w)(\varphi) = \text{unknown}$$

The interpretation of atomic context clauses is defined as follows:

$$(M, w) \models \text{d}[1\ \varphi] :\iff \forall w' \in R_d(w) : (M, w') \models 1\ \varphi$$
$$(M, w) \models \text{d}[0/1\ \varphi] :\iff \forall w' \in R_d(w) : (M, w') \models 0/1\ \varphi$$
$$(M, w) \models \text{d}[0/0/1\ \varphi] :\iff \forall w' \in R_d(w) : (M, w') \models 0/0/1\ \varphi$$
$$(M, w) \models \text{d}[0/0/0/1\ \varphi] :\iff \forall w' \in R_d(w) : (M, w') \models 0/0/0/1\ \varphi$$

The interpretation of structured context clauses is defined as follows:

$$(M, w) \models \text{d}[\text{s}[\text{fact}]] :\iff \forall w' \in R_d(w) : (M, w') \models \text{s}[\text{fact}]$$

The interpretation of a program within a context d *is defined as follows, where* program* *is the set of clauses given in the program:*

$$(M, w) \models \text{d}[\text{program}] \iff \forall \text{clause} \in \text{program}^* : (M, w) \models \text{d}[\text{clause}]$$

An interpretation structure M is called a model *of a program P iff* this[P] *is true in all worlds with respect to M. We use "valid" for "true in all worlds". The notation $M \models$ this[P] reads "*this[P] *is valid".*

The formalism is next extended so that the knowledge of an object is augmented with that of its structurally related objects, necessary to allow for the retrieval of objects at varying granularity.

3 Knowledge augmentation using modal operators

Consider the program "d[s1[sailing] s2[boats]]". A model for the program is based on an interpretation structure M defined upon a set of possible worlds $W = \{w_{slot}, w_{this}, w_d, w_{s1}, w_{s2}\}$, the accessibility relations $R_{this} = \{(w_{slot}, w_{this})\}$, $R_d = \{(w_{this}, w_d)\}$, $R_{s1} = \{(w_d, w_{s1})\}$ and $R_{s2} = \{(w_d, w_{s2})\}$, and the truth value assignments $\pi(w_{s1})(sailing) = \pi(w_{s2})(boats) = true$, and *unknown* in all other cases. Definition 2 yields $(M, w_{w_d}) \models$ s1[sailing] and $(M, w_d) \models$ s2[boats] characterising the knowledge content of s1 and s2 (s1 knows sailing and s2 knows boats), and $(M, w_{this}) \models$ d[s1[sailing]] and $(M, w_{this}) \models$ d[s2[boats]] characterising the structural knowledge of d (d knows that s1 knows sailing and s2 knows boats). Contexts s1 and s2 would be relevant to queries about "sailing" and "boats", respectively, but none of the contexts s1 and s2 would be relevant to a query about "sailing and boats". The supercontext d should be relevant to such a query because it knows that one of its subcontext contains sailing and the other contains boats. This could be inferred if the knowledge content of d is augmented with that of its subcontexts. However, a knowledge augmentation process can lead to inconsistent knowledge. Consider the program "d[s1[sailing] s2[0/1 sailing boats]]". Subcontext s1 knows sailing; subcontext s2 knows the opposite. The example makes evident that augmenting the content knowledge of d by that of s1 and s2 leads to inconsistent knowledge regarding the proposition "sailing", since we have evidence for true from s1 and false from s2. In the augmented context d(s1,s2), "sailing" is therefore inconsistent.

To distinguish between the content knowledge of a context and its *augmented content knowledge*, we introduce the terminology of *augmented context* as opposed to *basic context*. Basic contexts are context names (e.g. d, s1 and s2), whereas *augmented contexts* consist of a supercontext name and a list (group) composed of augmented contexts or basic contexts (e.g. d(s1(s11),s2)). A context clause with only basic contexts is called a *basic context clause* (e.g. "d[sailing]" is a basic atomic context clause and "d[s[sailing]]" is a basic structured context clause). A context clause with augmented contexts is called an *augmented context clause* (e.g. "d(s1,s2)[sailing]").

The knowledge augmentation process *combines* knowledge of a supercontext with that of its subcontexts. Modal operators have been defined to formalise specific combination of knowledge (e.g. common knowledge and distributed knowledge [5]). However, as discussed in [9], these operators are not appropriate for a combination of knowledge that arises from a knowledge augmentation process. We therefore define other modal operators.

The first operator is the *united knowledge modal operator* denoted U_G, which is used to represent the combined knowledge of a group of context $G =$ (s1,...,sn) referred to as a *united context*. Here, we are aiming at capturing the combined knowledge of a group of context, not the knowledge of an augmented context.

Definition 3. (United knowledge operator U_G): *Let w be a world in W, φ a proposition in Φ, and G a united context from $\mathcal{C}$.*

$$
\begin{aligned}
(M,w) \models U_G[1\ \varphi] :\iff\ & \exists \mathrm{s} \in G : (M,w) \models \mathrm{s}[1\ \varphi] \ \ \textit{and} \\
& \forall \mathrm{s} \in G : ((M,w) \models \mathrm{s}\ [1\ \varphi] \ \textit{or}\ (M,w) \models \mathrm{s}\ [0/0/0/1\varphi]) \\
(M,w) \models U_G[0/1\ \varphi] :\iff\ & \exists \mathrm{s} \in G : (M,w) \models \mathrm{s}[0/1\ \varphi] \ \ \textit{and} \\
& \forall \mathrm{s} \in G : ((M,w) \models \mathrm{s}\ [0/1\ \varphi] \ \textit{or}\ (M,w) \models \mathrm{s}[0/0/0/1\varphi]) \\
(M,w) \models U_G[0/0/1\ \varphi] :\iff\ & (\exists \mathrm{s} \in G : (M,w) \models \mathrm{s}[1\ \varphi] \ \textit{and} \\
& \ \exists \mathrm{s} \in G : (M,w) \models \mathrm{s}[\ 0/1\ \varphi]) \ \ \textit{or} \\
& \exists \mathrm{s} \in G : (M,w) \models \mathrm{s}[0/0/1\ \varphi] \\
(M,w) \models U_G[0/0/0/1\ \varphi] :\iff\ & \forall \mathrm{s} \in G : (M,w) \models \mathrm{s}[0/0/0/1\ \varphi]
\end{aligned}
$$

We define now the knowledge of augmented contexts through the introduction of an *augmented knowledge modal operator* $A_{\mathrm{d}G}$ where dG is an augmented context (d is the supercontext and G is the united context formed with the subcontexts of d, i.e. $G =(\mathrm{si} : \mathrm{si} \in sub(\mathrm{d}))$). The knowledge of the united context is combined with the knowledge of the supercontext. The augmented knowledge modal operator is therefore defined upon the united knowledge modal operator.

Definition 4. (Augmented knowledge operator $A_{\mathbf{d}G}$): *Let w be a world in W, φ a proposition in Φ,* d, s1, ..., sn *contexts in $\mathcal{C}$ such that $sub(\mathrm{d}) = \{\mathrm{s1},\ldots,\mathrm{sn}\}$, and R_d the accessibility relation in $\mathcal{R}$ associated with the supercontext* d.

$$
\begin{aligned}
&(M,w) \models A_{\mathrm{d(s1,\ldots,sn)}}[1\ \varphi] :\Longleftrightarrow\\
&\quad ((M,w) \models \mathrm{d}[1\ \varphi]\ \textit{and}\\
&\quad\quad \forall w' \in R_d(w) : ((M,w') \models U_{\mathrm{(s1,\ldots,sn)}}[1\ \varphi]\ \textit{or}\ (M,w') \models U_{\mathrm{(s1,\ldots,sn)}}[0/0/0/1\ \varphi]))\ \textit{or}\\
&\quad ((M,w) \models \mathrm{d}[0/0/0/1\ \varphi])\ \textit{and}\ \forall w' \in R_d(w) : (M,w') \models U_{\mathrm{(s1,\ldots,sn)}}[1\ \varphi])
\end{aligned}
$$

$$
\begin{aligned}
&(M,w) \models A_{\mathrm{d(s1,\ldots,sn)}}[0/1\ \varphi] :\Longleftrightarrow\\
&\quad ((M,w) \models \mathrm{d}[0/1\ \varphi]\ \textit{and}\\
&\quad\quad \forall w' \in R_d(w) : ((M,w') \models U_{\mathrm{(s1,\ldots,sn)}}[0/1\ \varphi]\ \textit{or}\ (M,w') \models U_{\mathrm{(s1,\ldots,sn)}}[0/0/0/1\ \varphi]))\ \textit{or}\\
&\quad ((M,w) \models \mathrm{d}[0/0/0/1\ \varphi])\ \textit{and}\ \forall w' \in R_d(w) : (M,w') \models U_{\mathrm{(s1,\ldots,sn)}}[0/1\ \varphi])
\end{aligned}
$$

$$
\begin{aligned}
&(M,w) \models A_{\mathrm{d(s1,\ldots,sn)}}[0/0/1\ \varphi] :\Longleftrightarrow\\
&\quad ((M,w) \models \mathrm{d}[0/0/1\ \varphi]\ \textit{or}\ \forall w' \in R_d(w) : (M,w') \models U_{\mathrm{(s1,\ldots,sn)}}[0/0/1\ \varphi])\ \textit{or}\\
&\quad ((M,w) \models \mathrm{d}[1\ \varphi]\ \textit{and}\ \forall w' \in R_d(w) : (M,w') \models U_{\mathrm{(s1,\ldots,sn)}}[0/1\ \varphi])\ \textit{or}\\
&\quad ((M,w) \models \mathrm{d}[0/1\ \varphi]\ \textit{and}\ \forall w' \in R_d(w) : (M,w') \models U_{\mathrm{(s1,\ldots,sn)}}[1\ \varphi])
\end{aligned}
$$

$$
\begin{aligned}
&(M,w) \models A_{\mathrm{d(s1,\ldots,sn)}}[0/0/0/1\ \varphi] :\Longleftrightarrow\\
&\quad (M,w) \models \mathrm{d}[0/0/0/1\ \varphi]\ \textit{and}\ \forall w' \in R_d(w) : (M,w') \models U_{\mathrm{(s1,\ldots,sn)}}[0/0/0/1\ \varphi]
\end{aligned}
$$

4 Knowledge augmentation using truth value assignment functions

In this section, we define the knowledge augmentation process based on a graphical representation of worlds and accessibility relations formalised through the notions of G-world-trees. This follows the approach adopted in [5], where combination of knowledge was defined upon the notion of “G-reachability”. Two truth value assignment functions are defined upon G-world trees to characterise united knowledge and augmented knowledge. The logical structure of a non-atomic object is described as trees, referred to as *G-world-trees*. These tree structures are also reflected in the way possible worlds and accessibility relations are defined with respect to contexts. An empty united context is denoted "()", and for a context name s in $\mathcal{C}$, s is treated as augmented context with an empty united context (e.g. s is the same as s()).

Definition 5. (G-world-trees): *Let* d *be a context in* $\mathcal{C}$ *with accessibility relation* R_d *in* $\mathcal{R}$*. Let* w *and* w_0 *be worlds in* W*. Let* G_n *be the united context* (s1,...,sn) *and let* G_{n+1} *be the united context* (s1,...,sn+1)*, for* s1, ..., sn+1 *in* $\mathcal{C}$*. The set of G-world-trees associated with a united context is defined inductively as follows:*

$$\begin{aligned}trees(w, G_{n+1}) := \{(w, S) | \exists S_n, t : (w, S_n) \in trees(w, G_n) \wedge \\ t \in trees(w, s_{n+1}) \wedge S = S_n \cup \{t\}\}\end{aligned}$$

A tuple (w, S) *is a* G-world-tree *of a world* w *and a united context* G_{n+1} *if there exists a set* S_n *such that* (w, S_n) *is a G-world-tree of the world* w *and the united context* G_n*, there exists a G-world-tree* t *of the world* w *and the context* sn+1*, and* $S = S_n \cup \{t\}$*.*

The G world-tree of a world w *and an empty united context* () *is* $(w, \{\})$*.*

The set of trees associated with an augmented context is defined inductively as:

$$trees(w_0, \mathrm{d}G) := \{(w, S) | w \in R_d(w_0) \wedge (w, S) \in trees(w, G)\}$$

A tuple (w, S) *is a* G-world-tree *of a world* w_0 *and an augmented context* dG *if* $w \in R_d(w_0)$ *and* (w, S) *is a G-world-tree of the world* w *and the united context* G*.*

The G-world tree of an augmented context d(s1,...,sn) formalises the accessibility of possible worlds associated with the subcontexts si from the possible world associated with the context d, and for each si, the accessibility of the possible worlds associated with the subcontexts of the si from the possible world associated with the context si, etc. This reflects the logical structure among d, its subcontexts si, the subcontexts of si, etc. We call $\mathcal{T}$ the set of G-world trees associated with an interpretation structure M.

The next step is to define truth value assignment functions with respect to G-world trees to capture the logical structure among the contexts. First we define the truth value value assignment function, referred to as *united truth value assignment function* associated with a united context to model the united knowledge of the united context. Here, the definition of the four truth values as sets (see Definition 1) is exploited to arrive at the *united truth value.*

Definition 6. (United truth value assignment function π_U): *Let* (w, S) *be a G-world-tree in* $\mathcal{T}$ *of world* $w \in W$ *and a united context* G *from* $\mathcal{C}$*. The united truth value function* $\pi_U : \mathcal{T} \rightarrow (\Phi \rightarrow \{\text{true}, \text{false}, \text{inconsistent}, \text{unknown}\})$ *of a G-world-tree* (w, S) *is defined as the union of the truth value functions* $\pi(w')$ *where the worlds* w' *are the roots of the subtrees* $(w', \{\})$ *in the set* S*. Formally, for all proposition* φ *in* Φ*:*

$$\pi_U((w, S))(\varphi) := \bigcup_{(w', \{\}) \in S} \pi(w')(\varphi)$$

For an empty set S, the united truth value $\pi_U((w, \{\}))(\varphi)$:= unknown.

We define now the truth value assignment function, referred to as *augmented truth value assignment function*, associated with an augmented context to model its augmented knowledge.

Definition 7. (Augmented truth value assignment function π_A): *Let d be a context in $\mathcal{C}$ and G a united context in $\mathcal{C}$. Let (w, S) be a G-world-tree in $\mathcal{T}$ of world w_0 in W and the augmented context dG. The augmented truth value function $\pi_A : \mathcal{T} \rightarrow (\Phi \rightarrow \{\text{true}, \text{false}, \text{inconsistent}, \text{unknown}\})$ of a G-world-tree (w, S) is defined as the union of the truth value function $\pi(w)$ of world w and the united truth value function $\pi_U((w, S))$ of the G-world-tree (w, S). Formally, for all φ in Φ:*

$$\pi_A((w, S))(\varphi) := \pi(w)(\varphi) \cup \pi_U((w, S))(\varphi)$$

5 Equivalence of the two approaches

Sections 3 and 4 present two approaches that formalise the knowledge augmentation process. The two approaches are based on different formalisations of united knowledge and augmented knowledge. This section presents two theorems that show that the two approaches are equivalent, that is, they lead to the same knowledge augmentation process (the detailed proofs can be found at http://www.dcs.qmul.ac.uk/~mounia/CV/pubconf.html/KA.pdf). Theorem 1 states that the definitions of united truth value assignment function (Definition 6) and united knowledge modal operator (Definition 3) lead to the same truth value assignment to a proposition in a united context. The same applies for Theorem 2, with respect to the definitions of the augmented truth value assignment (Definition 7) and the augmented knowledge modal operator (Definition 4).

Theorem 1. (United knowledge): *In a world $w \in W$, a proposition $\varphi \in \Phi$ is true in a united context G iff for each tree (w, S) of world w and united context G, the united truth value equals* true. *Formally $\forall$ s in $G : R_s(w) \neq \{\}$:*

$$(M, w) \models U_G[\varphi] \iff \forall (w, S) \in \mathit{trees}(w, G) : \pi_U((w, S))(\varphi) = \text{true}$$

Theorem 2. (Augmented knowledge): *In a world w_0 in W, a proposition φ in Φ is true in an augmented context* dG *iff for each tree (w, S) of world w_0 and an augmented context* dG*, the augmented truth value equals* true.

$$(M, w_0) \models \text{d}G[\varphi] \iff \forall (w, S) \in \mathit{trees}(w_0, \text{d}G) : \pi_A((w, S))(\varphi) = \text{true}$$

Our first formalism (Section 3) is, therefore, compatible with the well-known framework defined in [5] for modelling the combination of knowledge (Section 4).

6 Conclusion and future work

This paper describes a formal model for representing the content and logical structure of structured documents for the purpose of their retrieval. To obtain a representation that allows for the retrieval of objects of varying granularity, the content of an object is viewed as the knowledge contained in that object, and the structure among objects is captured by a process of knowledge augmentation, which leads to a representation of the object that is based on its own content and that of its structurally related objects. The knowledge augmentation process takes into account the fact that knowledge can be incomplete and become inconsistent. The model is based on the definitions of four truth values, modal operators, possible worlds, accessibility relations and truth value assignments used to characterise content knowledge, structural knowledge, augmented knowledge, incompleteness and inconsistency.

The formalism described in this paper is the basis of the development of a model for structured document retrievals that allows for the retrieval of objects of varying granularity. Future work will present the representation of the uncertainty inherent to the information retrieval process, which will be used to estimate the degree to which an object is relevant to a query.

References

1. Abiteboul, S., Cluet, S., Christophides, V., Milo, T., Moerkotte, G., and Simeon, J. Querying documents in object databases. *International Journal on Digital Libraries 1* (1997), 1–9.
2. Belnap, N. A useful four-valued logic. In *Modern Uses of Multiple-valued Logic*, J. Dunn and G. Epstein, Eds. Reidel, Dordrecht, 1977.
3. Chellas, B. *Modal Logic*. Cambridge University Press, 1980.
4. Chiaramella, Y., Mulhem, P., and Fourel, F. A model for multimedia information retrieval. Tech. Rep. Fermi ESPRIT BRA 8134, University of Glasgow, 1996.
5. Falgin, R., Harpen, J., Moses, J., and Vardi, M. *Reasoning about Knowledge*. MIT Press, Cambridge, Massachusetts, 1995.
6. Frisse, M. Searching for information in a hypertext medical handbook. *Communications of the ACM 31*, 7 (1988), 880–886.
7. Fuhr, N., and Grossjohann, K. XIRQL: A query language for information retrieval in XML documents. In *Proceedings of ACM-SIGIR Conference on Research and Development in Information Retrieval* (New Orleans, USA, 2001), pp. 172–180.
8. Myaeng, S., Jang, D. H., Kim, M. S., and Zhoo, Z. C. A flexible model for retrieval of SGML documents. In *Proceedings of ACM-SIGIR Conference on Research and Development in Information Retrieval* (Melbourne, Australia, 1998), pp. 138–145.
9. Roelleke, T. *POOL: Probabilistic Object-Oriented Logical Representation and Retrieval of Complex Objects - A Model for Hypermedia Retrieva.* PhD thesis, University of Dortmund, Germany, 1999.