Structuring and merging Distributed Content
Luca Stefanutti, Dietrich Albert, Cord Hockemeyer

To cite this version:

HAL Id: hal-00190477
https://telearn.archives-ouvertes.fr/hal-00190477
Submitted on 23 Nov 2007

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
A flexible approach for structuring and merging distributed learning objects is presented. At the basis of this approach there is a formal representation of a learning object, called attribute structure. Attribute structures are labeled directed graphs representing structured information on the learning objects. When two or more learning objects are merged, the corresponding attribute structures are unified, and the unified structure is attached to the resulting learning object.

Keywords: distributed learning objects, knowledge structures, structuring content

1. INTRODUCTION

In order to decide which object comes next in presenting a collection of learning objects to a learner, one might establish some order. Given a set $O$ of learning objects, a surmise relation on $O$ is any partial order ‘$\preceq$’ on the learning objects in $O$. The interpretation of ‘$\preceq$’ is that, given any two learning objects $o$ and $o'$ in $O$, $o \preceq o'$ holds if a learner who masters $o'$ also masters $o$. The concept of a surmise relation was introduced by [5] as one of the fundamental concepts of a theoretical framework called Knowledge Space Theory. According to this theory the knowledge state of a learner is the subset $K$ of all learning objects in $O$ that (s)he masters. A subset $K \subseteq Q$ is said to be a knowledge state of the surmise relation ‘$\preceq$’ if $o \in K$ and $o' \preceq o$ implies $o' \in K$ for all learning objects $o, o' \in O$. Then the collection $Q$ of all knowledge states of ‘$\preceq$’ is called the knowledge space derived from ‘$\preceq$’. Knowledge spaces are used for representing and assessing the knowledge state of a learner (see, e.g.,[5] and [6] in this connection).

The construction of a surmise relation may follow different approaches. After a brief presentation of an existing approach based on vectors of components of a learning object, we extend this approach to a more flexible representation called attribute structure [2]. The mathematical properties of attribute structures make it possible to compare distributed learning objects in terms of how much informative and how much demanding they are.

2. THE COMPONENT APPROACH

According to the component approach [1, 7], every content object $o$ in $O$ is equipped with an ordered $n$-tuple $A = \langle a_1, a_2, \ldots, a_n \rangle$ of attributes where the length $n$ of the attribute $n$-tuple $A$ is fixed for all objects. Each attribute $a_i$ in $A$ comes from a corresponding attribute set $C_i$ called the $i$-th component of the content object. In this sense, given a collection $\mathcal{C} = \{ C_1, C_2, \ldots, C_n \}$ of disjoint attribute sets (or components), each object $o \in O$ is equipped with an element of the Cartesian product $P = C_1 \times C_2 \times \cdots \times C_n$. Usually each component $C_i$ is equipped with a partial order ‘$\leq_i$’ so that $\langle C_1, \leq_i \rangle$ is in fact a partially ordered set of attributes. The partial order ‘$\leq_i$’ is interpreted in the following way: for $a, b \in C_i$, if $a \leq_i b$ then a learning object characterized by attribute $a$ is less demanding than a learning object characterized by attribute $b$. To give a simple example, it might be postulated that ‘computations involving integer numbers’ (attribute $a$) are less demanding than ‘computations involving rational numbers’ (attribute $b$). A natural order $\leq$ on the elements in $P$, the so-called coordinatewise order [4], is derived from the $n$ partial orders ‘$\leq_i$’ by

$$\langle x_1, x_2, \ldots, x_n \rangle \preceq \langle y_1, y_2, \ldots, y_n \rangle \iff \forall i : x_i \leq_i y_i$$
If \( f : O \rightarrow \mathcal{P} \) is a mapping assigning an attribute \( n \)-tuple to each learning object, then a surmise relation \( \preceq \) on the learning objects is established by

\[
o \preceq o' \iff f(o) \preceq f(o')
\]

for all \( o, o' \in O \). The mapping \( f \) can easily be established even when the learning objects are distributed (see, e.g., [8]).

3. ATTRIBUTE STRUCTURES

An attribute structure is used to represent structured information on a learning object or an asset and in this sense it represents an extension of the attribute \( n \)-tupel discussed in Section 2. From a mathematical standpoint attribute structures correspond to the feature structures introduced by [3] in computational linguistics. Let \( C \) be a set of components and \( A \) a collection of attributes, with \( A \cap C = \emptyset \). An attribute structure is a labeled directed graph \( A = (Q, q_0, \alpha, \eta) \) where:

- \( Q \) is a set of nodes of the graph;
- \( q_0 \in Q \) is the root node of the graph;
- \( \alpha : Q \rightarrow A \) is a partial function assigning attributes to some of the nodes;
- \( \eta : Q \times C \rightarrow Q \) is a partial function specifying the edges of the graph.

As an example, let

\[
C' = \{ \text{picture, topic, subtopic, text, language} \}
\]

be a set of components, and

\[
A' = \{ \text{PICTURE1, TEXT1, ENGLISH, MATH, MATRIX_INVERSION} \}
\]

be a collection of attributes. Suppose moreover that a simple learning object is described by the asset structure \( A_i = (Q_i, q_i, \alpha_i, \gamma_i) \), where \( Q_i \) is the set of nodes, \( q_i \) is the root node, and \( \alpha_i \) and \( \gamma_i \) are defined as follows: \( \alpha_i(0) \) is not defined, \( \alpha_i(1) = \text{PICTURE1}, \alpha_i(2) = \text{TEXT1}, \alpha_i(3) = \text{MATH}, \alpha_i(4) = \text{ENGLISH}, \) and \( \alpha_i(5) = \text{MATRIX_INVERSION}; \eta_i(0, \text{picture}) = 1, \eta_i(0, \text{topic}) = 3, \eta_i(1, \text{topic}) = 3, \eta_i(2, \text{topic}) = 3, \eta_i(2, \text{language}) = 4, \) and \( \eta_i(3, \text{subtopic}) = 5 \). The digraph representing this attribute structure is displayed in Figure 1.

The structure \( A_i \) describes a very simple learning object containing a picture along with some text explanation. Both text and picture have MATH as topic and MATRIX_INVERSION as subtopic. The root node of the structure is node 0 and it can be easily checked from the figure that each node can be reached from this node following some path in the graph. The root node is the entry node in the asset structure of the learning object, and the edges departing from this node specify the main components of the learning object itself. Thus, in our example, the learning object represented by \( A_i \) is defined by three different components: picture, text and topic. The values of these three components are the attributes given by \( \alpha_i(\eta_i(0, \text{picture})) = \text{PICTURE1}, \alpha_i(\eta_i(0, \text{text})) = \text{DESCRIPTION}, \alpha_i(\eta_i(0, \text{topic})) = \text{MATH} \).

Observe, for instance that node 5 can be reached from node 0 following the path \( \text{picture, topic, subtopic} \). The fact that, in this example, each node is reachable from the root node through some path is not a coincidence. It is explicitly required that every node in an attribute structure be reachable from the root node.


4. COMPARING AND COMBINING ATTRIBUTE STRUCTURES

Attribute structures can be compared one another. Informally, an attribute structure $A$ subsumes another attribute structure $A'$ (denoted by $A \sqsubseteq A'$) if $A'$ contains at least the same information as $A$. In this sense an attribute structure can be thought as a class of learning objects (the class of all learning objects represented by that structure), and '$\sqsubseteq$' can be regarded as a partial order on such classes. Formally, an attribute structure $A = \langle Q, \bar{q}, \alpha, \eta \rangle$ subsumes another attribute structure $A' = \langle Q', \bar{q}', \alpha', \eta' \rangle$ if there exists a mapping $h : Q \rightarrow Q'$ fulfilling the three conditions

(1) $h(q) = \bar{q}'$;
(2) for all $q \in Q$ and all $c \in C$, if $\eta(q, c)$ is defined then $h(\eta(q, c)) = \eta'(h(q), c)$;
(3) for all $q \in Q$, if $\alpha(q)$ is defined then $\alpha(q) = \alpha'(h(q))$.

In Section 2 the attribute sets were assumed to be partially ordered according to pedagogical criteria and/or cognitive demands. Similarly we assume now that a partial order '$\leq$' is defined on the set $A$ of attributes so that, given two attributes $a, b \in A$, if $a \leq b$ then a learning object defined by attribute $a$ is less demanding than a learning object defined by attribute $b$. Then the subsumption relation '$\sqsubseteq$' is made consistent with '$\leq$' if condition (3) is replaced by

(4) for all $q \in Q$, if $\alpha(q)$ is defined then $\alpha(q) \leq \alpha'(h(q))$.

According to this new definition, if $A \sqsubseteq B$ then $A$ is either less informative than $B$ or less demanding than $B$ or both.

As an example consider the three attribute structures depicted in Figure 2. Assuming that $\text{MATRIX PRODUCT} \leq \text{MATRIX INVERSION}$, both mappings $g$ and $h$ fulfill conditions (1), (2) and (4), thus both attribute structures labeled by LO2 and LO3 subsume the attribute structure labeled by LO1. However there is neither mapping from LO2 to LO3 fulfilling the subsumption conditions, nor the opposite, thus these last two structures are incomparable to each other. The
derivation of a surmise relation for the learning objects parallels that established in section 2. If $s : o \mapsto s(o)$ is a mapping assigning an attribute structure to each learning object, then a surmise relation '$\lessdot$' on the learning objects is derived by

$$o \lessdot o' \iff s(o) \sqsubseteq s(o')$$

for all $o, o' \in O$.

Two binary operations are defined on attribute structures: unification and generalization. Mathematically, the unification of two attribute structures $A$ and $B$ (denoted by $A \sqcup B$), when exists,
is the least upper bound of \( \{ A, B \} \) with respect to the subsumption relation. Dually, generalization (denoted by \( A \sqcap B \)) is the greatest lower bound. In particular, for any two attribute structures \( A \) and \( B \) it holds that

\[
A \subseteq A \sqcup B, \quad B \subseteq A \sqcup B
\]

\[
A \sqcap B \subseteq A, \quad A \sqcap B \subseteq B
\]

When two different learning objects are merged together, or when different assets are assembled into a single learning object, the corresponding attribute structures are unified, and the resulting attribute structure is assigned to the resulting learning object. On the other hand, the generalization operation is used to find the common structure of two or more learning objects or, stated another way, to classify learning objects. An example of the generalization operation applied to two attribute structures is shown in Figure 3. Here, the resulting structure shows that two learning objects have in common topic and subtopic. Generalized attribute structures can also be used e.g. for searching a distributed environment for all learning objects whose structure is consistent with a certain ‘template’ (for instance to find out all learning objects that are ‘problems’ involving, as cognitive operation, ‘recognition’ rather than ‘recall’).

REFERENCES


