Abstract

Authoring of concept-based Web courseware is more complicated than the ‘standard’ courseware authoring and thus needs more adaptive and intelligent support. In order to provide such support the authoring system itself needs to have understanding about the process of courseware authoring, to be able to decompose it in steps and procedures and reason over it; to provide hints and recommendations to the author, and perform various (semi-) automatic actions. Our goal is to develop a knowledge-based system for concept-based courseware authoring, where the knowledge representation is realized with the help of Courseware Authoring Tasks Ontology (CATO).

1. Introduction

Concept-based courseware employs conceptual domain presentation to link educational materials to a course structure. Therefore concepts are the core building blocks and the main operations in the along going courseware authoring process are grounded around concept manipulation, maintenance and management. The information base of a concept-based course support environment often includes three main components: domain ontology, course structure, and library metadata, which set specific requirements to the supporting authoring tools [3]. Since the authoring of concept-based Web courseware is three-fold [4], including domain-, course-, and library authoring, the process is becoming more complicated and labor intensive than the process of a ‘standard’ courseware authoring. Apparently this calls for better authoring support for concept-based Web courseware. We envisage such support to include automatic or semi-automatic performance of some authoring activities, intelligent assistance to the authors in the form of hints, recommendations, etc., as well as supporting the activities of different instructors for collaborative building and/or reuse of domain and course ontologies with templates (eg. based on recognizing different information patterns within domain ontologies or instructional patterns within course structures). A central idea in our approach for providing adequate authoring support for concept-based courseware is to propose a knowledge-based framework for concept-based courseware authoring, where the knowledge representation is realized with the help of Courseware Authoring Tasks Ontology (CATO).

2. Courseware Authoring Tasks Ontology

CATO is functional concept ontology in the sense of [1], where the functional concepts are courseware-authoring tasks. CATO design involves an initial decomposition of the courseware authoring process into a set of basic authoring operations that are used to build an upper-level ontology describing the basic authoring functions and mapping the underlying domain-specific models (such as in AIMS [3] and other instructional systems).

CATO consists of two main layers: base layer, which includes hierarchy of atomic authoring tasks, and meta-layer, which includes hierarchy of meta-functions. The atomic authoring tasks are primitive functional concepts, which are basic for the concept-based courseware authoring process, for it’s understanding and performing. Formal definitions of the atomic tasks are presumed to support their interpretation. This will allow building courseware authoring ontology terminology (vocabulary).

The primitive functions are defined on objects (concepts, documents, course topics and course tasks) within a specific concept-based structure (domain model, course structure, document base). They express a simple functional formalism, where the object changes the structure, or the structure is manipulated. Examples of atomic authoring functions include:

- Create (Structure)
- Create (Object)
- Add (Object, Structure)
- Delete (Object, Structure)
- Edit (Object, Structure)
- Link (Object₁, Object₂, Structure)
Delete (Structure)
Compare (Object₁, Object₂)
Exist (Object, Structure)
List (Objects, Structure)

where Object ∈ {Domain_Concepts} ∪ {Course_Topics} ∪ {Course_Tasks} ∪ {Library_Docs} and Structure ∈ {Domain_Model, Course_Model, Library_Base}. Note that such a definition is independent of the structure - the only prerequisite for it is to be concept-based.

In the meta-layer, we define hierarchy of meta-functions to represent conceptual categories of relationship (interdependence) between primitive functions. These present certain aggregation criteria (including causal and other relations among components) that are used for grouping primitive functional concepts into higher-level authoring functions (classes). This way we can construct/identify functional groups of authoring tasks. The meta-functions represent a role of one base-function for another base-function [1]. They are concerned not with the actual change in the objects, but with their actual function in the concept-based courseware authoring process. We define the meta-functions with conditions for their primitive parameters in order to achieve specific authoring goals. This will be based on extracting the functional structure for courseware authoring from existing authoring models (as domain) and their connection to educational information. Examples of meta-functions include:

- 'is-a-prerequisite-for'
- 'is-assigned-to'
- 'is-achieved-by'
- 'requires'
- 'follows-from'
- 'is-followed-by'
- 'is-preceded-by'
- 'if-<goal>-then-<action>'

CATO aims at defining formal specifications of conceptualisation to provide a common understanding of the concept-based courseware authoring process (as domain) that can be communicated to support tools for adaptive courseware authoring systems. In order to realise this, we envisage defining of a rule-based model (CATO-Rules) over the schematic representation of the ontology to support the understanding of CATO ontological scheme and allow for extracting additional semantics that can be applied in the reasoning strategies of the support tools. CATO-Rules assign interpretations directly to the CATO graph (based on RDF syntax [2]). The vocabulary of the CATO graph is determined by the set of primitive functional concepts (PFC) within the base layer. An interpretation function I is defined over a range (vocabulary V of PFC) and a domain (class of CATO meta-functions).

3. Conclusion

Knowledge representation based on ontologies could be very beneficial in environments, where flexibility is required, thus it can help in meeting the dynamic needs of courseware authoring systems, their maintenance and reuse. We believe that CATO can support common reasoning over the processes accruing in the authoring of the domain model, course structure and information base of concept-based courseware. As additional advantages, CATO can provide for: easy update and altering of knowledge structures; easy change of operation sets without changing the structures; a bird-eye view over the whole authoring process, which can be helpful for process analysis; better understanding of the semantics of the authoring process; good options for visualization, e.g., authoring workflow visualization, visual system feedback, constructing visual authoring wizard, etc.

References