Content and Interface Models for Multi Point of View Scientific Hyperbooks

FALQUET, Gilles, HURNI, Jean-Pierre & CUI - Centre Universitaire d'Informatique, Université de Genève

Abstract

We present a model for creating, managing, and viewing the contents of scientific hyperbooks. The model we propose is based on reusable information fragments and on terminological fragments that contain concept definitions. The definition of concepts, which plays an essential role in scientific writings, are represented in a formal language. These formal definitions can then be processed to automatically infer semantic links between fragments. In addition, concept definitions, which can be attached to “points of view”, can serve as a meta-information level to qualify the other fragments. Instead of having a fixed hyperdocument structure, as would be the case with HTML pages, the model includes an external view mechanism to generate a wide variety of hypertext views from the fragment repository. This enables the user to read (and write) the hyperbook in many different ways, according to different axis, needs, or tasks. A purely declarative language allows the definition of the views that make up the interface of the hyperbook. We also present the architecture of a hyperbook management system which is based on a [...]
Content and Interface Models
for Multi Point of View Scientific Hyperbooks

Gilles Falquet*, Jean-Pierre Hurni**

* Centre Universitaire d'Informatique, Université de Genève
24, rue Général-Dufour – CH 1211 Genève 4 – Suisse
tel.: +41 22 705 7772, fax +41 22 705 7780
falquet@cui.unige.ch

** Independent Scientific Research Institute (ISRI),
5, rue du vélodrome – CH 1205 Genève – Suisse
hurni@cui.unige.ch

Abstract
We present a model for creating, managing, and viewing the contents of scientific hyperbooks. The model we propose is based on reusable information fragments and on terminological fragments that contain concept definitions. The definition of concepts, which plays an essential role in scientific writings, are represented in a formal language. These formal definitions can then be processed to automatically infer semantic links between fragments. In addition, concept definitions, which can be attached to “points of view”, can serve as a meta-information level to qualify the other fragments.

Instead of having a fixed hyperdocument structure, as would be the case with HTML pages, the model includes an external view mechanism to generate a wide variety of hypertext views from the fragment repository. This enables the user to read (and write) the hyperbook in many different ways, according to different axis, needs, or tasks. A purely declarative language allows the definition of the views that make up the interface of the hyperbook. We also present the architecture of a hyperbook management system which is based on a database management system and a hypertext view generation system for databases.

Keywords: scientific hyperbooks, electronic books, reusable documents, virtual documents, terminology, views, point of view.

1. Introduction
Multi point of view scientific hyperbooks (MSHB) are a particular category of electronic book. The contents of electronic books are usually made of files in a specific format like PDF, HTML, or other proprietary formats) that can be rendered on a screen through a “viewer” (e.g. Acrobat Reader, Web browsers, etc.). This approach is clearly insufficient for scientific books because the complexity of the contents implies that the reader must be equipped with sophisticated tools enabling him or her to “actively read” the book. There are several ways to support active reading, among which one can cite: hypertext links, annotation, keyword marking, etc. One of the most well known and efficient cognitive help in this respect consists in offering different perspectives on the same topic. This is why we propose to study more deeply the multi point of view in the electronic book models. In addition, scientific books must be adaptable to different types of readers or different reading purposes. Here again, it is necessary to provide different views of the content depending on the user types and reading purposes.

This lead us to define a hyperbook model that is based on the following components:
– a repository of information fragments
– a terminological knowledge base (terminological fragments)
– several external view specification

The fragment repository is the information base of the book, every fragment is a piece of information that can be assembled with other fragments to build readable documents or hyperdocuments. The terminological knowledge base has two purposes: firstly, it contains definitions for all the specialized terms appearing in the book; secondly, the semantic links between the described concepts can be used to infer hypertext links between fragments, thus avoiding “hard wired” links. User view specifications define how to assemble fragments and how to present the resulting content on the user screen. A view specification language allows the book designer to create several dif-
ferent views to suit different needs or reading objectives. Indeed, the fragments constitute a virtual book because they can give rise to many different readable documents (views), as shown in Figure 1.

The rest of this paper is organized as follows: in the next two sections we present a conceptual model for representing the base fragments and the terminological fragments that constitute a scientific hyperbook; then we present the interface model and the associated view definition language; in the conclusion we briefly discuss some implementation techniques used in the developed prototype.

2. A Hyperbook Model

It is generally accepted that virtual documents [15] are made of fragments (or “pieces of information”) which can be assembled to constitute directly readable real documents or hypertext documents read by navigation [15][4]. The model we developed is inspired by preceding works in the domain of hypertexts and knowledge bases. For example, we keep the distinction between documents (information) and definitions of concepts (knowledge) of the MacWeb model [11], while adapting it to the specific case of the scientific hyperbook.

A fragment has a content which is a hierarchically structured XML document. According to the “XML Query Data Model”[8] from the W3 Consortium, a document is a set of nodes of various types (element, attribute, and text). A “parent-child” relation over these nodes defines the tree structure of the document. Although in a real document the role of a fragment is often related to its position, it is not the case in a virtual document. This is why each fragment must be explicitly categorized. For example, during the collective creation of a course hyperbook we identified the following categories: concept, algorithm, exercise, popularization, conceptual map, index and frequently asked question. In addition to its identity as an object, each fragment has a name that makes it possible to identify it on the level of the user interface.

A relation establishes a typed link between two fragments. At this level, relations are intended to represent either structural links among document parts, or rhetoric/argumentative links like: issue, position, argument, contradiction, etc. To specify on which part of the fragment to apply a relation, such a relation possesses a starting anchor and an ending anchor which are elements of the content of the linked fragments. Moreover, a relation can be qualified by a point of view to indicate the relevant point of view.

A point of view corresponds either to a particular knowledge domain (mathematics, physics, history, history of sciences) or to a category of user (researcher, advanced student, undergraduate, general public). Points of view are hierarchically organized, for instance the academic point of view may include the professor, student, and researcher points of view.

Figure 2 shows a simplified UML model of the fragments and relations; it serves as a basis for the implementation of a MSHB system.

3. Terminological Fragments

It is common for the virtual document storage architectures to distinguish between the document fragments and the semantic structure. The latter, for example an ontology or a conceptual graph, describes the domain and is used for indexing or qualifying the fragments [7]. Since a significant part of each scientific work is generally dedicated to the definition of concepts, there is no clear distinction between basic documents and a conceptual structure. Thus we propose to build the conceptual structure as a set of specialized fragments called terminological
fragments. A terminological fragment describes a concept, it is composed of a term and a definition. The defini-
tion itself can be either in the form of a text in natural language (stored text elements), or expressed in a formal
language. Formal definitions of concepts are expressed in a description logic language [3]. A concept is defined
by a conjunction, a disjunction, or a negation of other concepts. It can also be defined by the existence of a rela-
tion (called role) to another concept. For instance, the expression 

\[
\text{part: "vector"}
\]

defines the concept of all things that have a vector part (they may have other parts), while the expression 

\[
\text{all part: "number"}
\]

defines the concept of all things that are composed solely of numbers (they cannot have part of another kind). The
following example shows a formalized definition of the concept of electron in the high-energy physics domain
(“the lightest massive particle of negative charge that is insensitive to the strong interaction”).

\[
\text{concept "electron"}
\]
\[
\text{generic "massive particle"}
\]
\[
\text{mass: "minimum"}
\]
\[
\text{and charge: (quantity: "unit\& sign: "negative")}
\]
\[
\text{and not sensitivity: "strong interaction"}
\]

Each concept definition is associated to only one point of view, but a term may have several definitions corre-
sponding to different point of views. For instance, in the example above, the electron concept was defined accord-
ing to the high-energy physicist’s point of view. A definition provided by a chemist would probably be different,
for example: “electric corpuscle that can be dragged away, caught or shared between atoms and molecules”.

In order to represent these definitions in a hyperbook, we extend the fragment model with the data structure
shown on Figure 3, where a definition is either a conjunction, a disjunction, or a negation; and a role is in fact a
relation between two concept fragments that has a name, a quantifier, minimal and maximal number of occur-
cences.

Relations between terminological fragments and other fragments
Terminological fragments carry both information and meta-information, since one can read them to understand
the meaning of a term (information) or they can describe the content of another fragment (meta information). A
terminological fragment is implicitly linked to all the fragments that contain this term in their text. It can also be
explicitly linked through a typed relation in order to qualify the specific nature of the meta-information. Typical
relation types are: illustration, property (the fragment describes a property of the concept), assertion/statement/
hypothesis, method/algorithm, observation.
In addition, terminological fragments support the inference of semantic links between fragments. The basic inference principle consists in “going up” to a terminological fragment connected to the starting base fragment, possibly following roles between terminological fragments, and then “going down” to another base fragment.

Another type of inference consists in inferring the main point of view of a fragment by looking at the points of view of the related terminological fragments.

4. The interface model

It is well known that designing and developing a convenient user interface is hard and expensive. In order to minimize this effort we propose an approach based on the specification of hypertext views on databases. The fragment model can be easily represented in a database schema (object based or relational), it is thus possible to use the database technology to store the contents of the MSHB and a hypertext view generation system to generate the interface. The interface of a MSHB thus consists of a set of derived hyperdocuments called hypertext views.

The Hypertext View Specification Language

The view specification language is based on Lazy [5], which, like Araneus [2], belongs to the family of purely declarative language for generating hypertext views of databases. A specification is a set of node schema, each node schema determines a set of objects (tuples) to be selected according to parameters and a way to build contents and links to other nodes. The instantiation of a node consists in interpreting a node schema for a given set of arguments and the current state of the database. A node schema is declared with the following syntax:

```
node NodeName [ Parameters ]
  Content
  from Collection(s)
  selected by Condition
  order by OExpression
```

where Content is a list of XML or HTML element constructors, Collection is one of the classes of the fragment repository (Fragment, Relation, TerminologicalFragment, etc.), Condition is a selection predicate, and OExpression is an ordering expression. For example, an instance fragmentInCategory["experiment"] of the following

Figure 3. UML schema of terminological structure

In addition, terminological fragments support the inference of semantic links between fragments. The basic inference principle consists in “going up” to a terminological fragment connected to the starting base fragment, possibly following roles between terminological fragments, and then “going down” to another base fragment.

Figure 4. Inferring semantic links between base fragments

Another type of inference consists in inferring the main point of view of a fragment by looking at the points of view of the related terminological fragments.
node schema will contain the list of all the fragments in the category "experiment" together with a link ("[details]") to another node (fragment(fid)):

```html
node fragmentInCategory[c]
  <h2>"Fragments in category ", c
  { <p>( name, " ",
      href fragment[fid]("[details]"))
  }
from fragment selected by category = c order by fid
```

The specification language provides three types of links: reference links; inclusion links; and “expand in place” links (inclusion links triggered by the user). In order to simplify the design of external views, there exist a library of predefined node schemas for presenting fragment content, displaying lists of related fragments, etc. Thanks to the inclusion mechanism these nodes can be composed to from more sophisticated nodes. For example, the following node schema is intended to display the content of a fragment (elementsInFragment) in the middle columns, a list of the terminological fragments linked to this fragment (relatedTerminology) on the left, and the relations starting from this fragment (relationsFrom) on the right. This schema makes use of other nodes through inclusion links.

```html
node fragment[id]
  |
  <h3>(<center>fname) ,
  p align="right">(category) ,
  <table cellspacing="3" bgcolor="lightblue">(
  <tr>(
    <td width="20%" valign="top">( 
      <h4>(Related terminology) ,
      include relatedTerminology[id]
    ) ,
    <td width="60%">( 
      include elementsInFragment[id, root_elem]
    ) ,
    <td width="20%" valign="top">( 
      <h4>(Relations) ,
      include relationsFrom[id]
      <p>(href newRelationFrom[id] ("Add relation")
    )
  )
  }
from fragment selected by fid = id order by fid
```

Figure 5 shows an instance of this node schema on a fragment repository for a MSHB on the use of quaternions in physics.

Node schemas can also specify how to update the fragment repository in response to user inputs. This means that it is possible to define active nodes capable to modify the hyperbook. For instance, the node newRelationFrom appearing in the previous schema is intended to enable the user to create a new relation starting from the displayed fragment. Similarly one can define nodes to create different types of annotations or relations, in order to support active reading.

**Designing external views**

- External views are strongly related to point of views. In fact, reading a hyperbook according to a point of view consists in selecting those fragments and relations that are relevant for this point of view. This is accomplished by defining node schemas that have selection conditions based on the point of views associated with the fragment, elements and relations.
- From the terminological fragments and their relations one can create views representing trees of concepts or semantic networks. With a suitable style processing, these views can then be translated to a specific knowledge representation format such as RDF.
– Access structures such as indices, domain trees, conceptual map, etc., play an essential role as entry points in an electronic book [13]. In this case, one will use the relations between fragments and concepts to define this type of views. This kind of access is particularly interesting for multimedia fragments [3].

5. Conclusions and Future Work

We have presented a model for representing multi point of view scientific hyperbooks in the form of virtual hyperdocuments. In this model, the terminological fragments play a significant role to support the multi point of view aspect and to generate views for reading and accessing the hyperbook. The interface of a MSHB consists of views specified using a declarative language instead of traditional procedural languages or scripting languages.

We have developed a fully operational prototype that implements this model. It is based on very simple database technologies, such as Hypersonic SQL and simple Web technologies (http server and Java servlets). It is thus easily portable to any platform, including hand held devices.

We are currently using this system to produce the core a MSHB dedicated to collaborative learning in theoretical physics.

References