

# Information Organizing based on Task Model Represented by DTD

6 V - 4

Calvin Woodworth

Hiroyuki Kojima

Hiroshima Institute of Technology

## 1. Introduction

The progressive information networking technology requires a lifecycle acceleration of organizational knowledge and the effective functioning of knowledge conversion from information that dominates economic structures, and therefore knowledge has become an important asset in determining the competitiveness of an organization. This is particularly true in the production sector where the lifecycle of product are becoming shorter. Therefore production workers need to innovate new products in order to remain competitive. This indicates that a platform is needed to support the knowledge workers to retrieve relevant and structured information, share and communicate ideas effectively in response to business cycles and events, like an information organizing and sharing systems. In the design field, works with a lot of images and drawings that are represented by a task model, which are task procedures that have rules and procedures. Thus a designer can formalize their knowledge through the task model and uses the image as the key for information retrieval.

The information organizing and sharing system discussed in this paper, using image as the key in information structuring and retrieval, focused on knowledge creation process based on task model that is represented by document type definition (DTD). The task model is based on ontology. Images are used because users can mentally identify an object faster than the process of, for example, reading. It is aimed at making a support facility of intuitive understanding and creation of information or knowledge.

---

Information Organizing based on Task Model Represented by DTD

Calvin WOODWORTH, Hiroyuki KOJIMA

Graduate School of Environmental Studies,

Hiroshima Institute of Technology

2-1-1 Miyake, Saekiku, Hiroshima, 731-5193

## 2. Task Model and XML's DTD

In the growing world of information technology, common standard are needed especially human-machine-human communication environment. XML (eXtended Markup Language) is one of the standard proposed because it has the ability to represent data. The difference between Hypertext Markup Language (HTML) and XML is that HTML has a set of predefined tags while XML allows the user to define its own tags. Thus XML can be seen as a means of data definition and representation. XML's DTD is a document that defines the structure of XML documents.

Definition of knowledge is an "understanding" about a subject which an individual or organization has, accepting data, information, or other knowledge<sup>1)</sup>. It means an action rule, which is established in an individual or organization. Task Ontology is a specifically indicative description of task, including terminology and procedure, same as task model. This "understanding" can be supported by organizing of data or information in which organizing operation is processed by task model operator. The task model can be facilitated using DTD as follows.

As XML is not a data model, ontology is used to map XML-encoded information into data model. DTD is not ontology and it acts as a tool that represents task model. Task model provides a set of relation that can integrate the different XML document. Inheritance mechanism can be applied using the ontology concepts as they are organized in taxonomies and axioms<sup>2)</sup>. Therefore, DTD, which define relations via simple nesting references, is suitable as a tool for task ontology. Furthermore XML gives the user the power to specify the vocabulary of the classes but does not attempt to impose or standardized the individual classes<sup>3)</sup>.

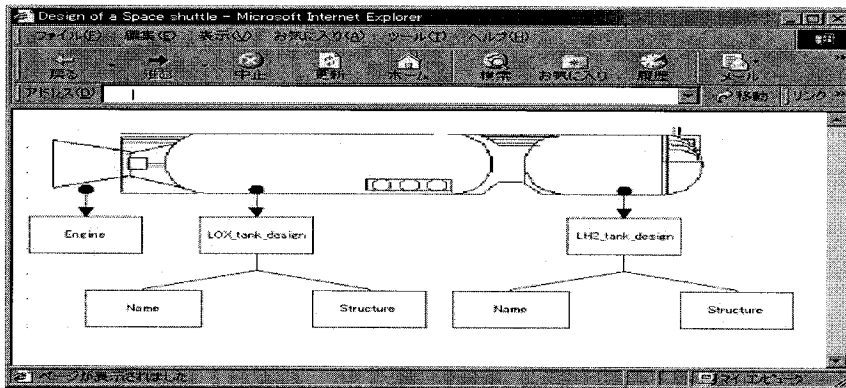


Figure 1 : Using image as a key for information retrieval

### 3. Creating the Task Model through Ontology

In order to build the task model based on task ontology, designers need to define the task, subtasks, constraints and attributes. To further explain our system we will use the engineering design of a space shuttle as an example which is shown as retrieval interface in Figure 1.

The first step of creating the task ontology is to define the attributes and also its data type. This step is to create the structure of the DTD documents and not the XML document itself. For example, the attribute description has the data type string. In DTD it is represented as follows:

```
<!Element description (#PCDATA)>
```

Once the attributes have been defined, the designer can annotate the task model. For example in a space rocket, a propellant tank is a type of tank, and in this propellant tank LOX (liquid oxygen) tank is used. The LOX tank would have the attributes of length, structure and strength and they must occur only once. The DTD is expressed as,

```
<!Element LOX_tank(length, structure, strength)
```

Figure 2 shows an example of the XML and DTD document.

### 4. Conclusion

In this paper we discuss the use of DTD in representing task model that is based on task ontology. Image is used as the key for information retrieval. Task ontology is used as it provides a hierarchical representation of the task model. As DTD is also in a hierarchical format, it is used as a tool for task ontology.

```
<!Element tank (name, size?, mixture?, ratio? )*>
<!Element engine(name, parts*, speed?)*>
<!Element LOX_tank (name, mixture?, ratio)*>
<!Element LH2_tank (name|ratio)*>
```

```
<propellant_tank>
  <LOX_tank_design>
    <name> Liquid Oxygen Tank Front
  </name>
  <structure>
    <length>7m</length>
    <diameter>4m</diameter>
    <position>1st stage </position>
    <description>desc.doc</description>
  </structure>
```

Figure 2 : Example of XML and DTD

### References

- 1) Tabata, K., Mitsumori, S., "An Email-Based Information Sharing System for Online Communities", *Information, Knowledge, and Systems Management*, 1(1999).
- 2) Benjamins, V.R., et.al, "Knowledge Management through Ontologies", *Proceedings of the 2<sup>nd</sup> International Conference on Practical Aspects of Knowledge Management* (1998).
- 3) Lu, K., "XML: The next Generation EDI" *Java Developers Journal* 4,9 pg 68-70 (1999).