A Conceptual Modeling Approach to Virtual Organizations in the Grid*

William Song[1] and Xiaoming Li[2]

[1] Computer Science, University of Durham, UK
[2] School of Information Technology, Peking University, China

Abstract. One of the key problems with virtual organizations in the context of the (Semantic) Grid is we don't know how to describe the components used in virtual organizations because they appear to be dynamic, dispersed, transient, type-vague (or we don't know their types), heterogeneous, semantically informal, and disorderly. The existing semantic modeling approaches lack effective modeling facilities for modeling virtual organization components. Our observations were suggesting focusing on the following aspects: First, we should model both static resources and dynamic states of a virtual organization. Second, we should build up the virtual organizations with abundant static resource. Third, we should combine the semantic modeling with the users' requirements description for virtual organizations due to its importance. In this paper, we analyze the current situation of the virtual organization research and propose our experimental semantic description model for virtual organizations. We also present the design architecture for the *Realcourse* application.

1 Introduction

Current researches on the World Wide Web and the Grid, as well as Web Services, have come to a consensus issue. That is, how to extract and represent semantics for the web information and the Grid recourses. As known to us, the W3C (the World Wide Web Consortium) provides standards/recommendations for information exchange and interoperation, such as RDF(S) [5, 8], and the Grid community provides an infrastructure for resource sharing and cooperation, i.e. OGSA [3]. The interoperability and collaboration requires common understanding of information/data and resources and the understanding requires canonical and well-formed semantic description. The semantic description for the Web and Grid resources is used both for human and machine to understand each other.

The Grid is an emerging platform to support on-demand "virtual organizations" for coordinated resource sharing and problem solving on a global scale [4]. The virtual organization is seen as "a temporary alliance of contracted individuals or companies linked together by information and communication technologies, which assembles for the purpose of a specific business task" [6, 7]. In the Grid computing environment, there are many resources, e.g., courses, lectures, articles, presentation slides, demonstration videos, training systems, and testing papers in the area of online teaching/learning, which we call nodes. These nodes are connected to each other,

* The work is partially supported by the NSF project (90412010).

H. Zhuge and G.C. Fox (Eds.): GCC 2005, LNCS 3795, pp. 382 – 393, 2005.

exchanging information, sharing their resources, and collaborating on certain tasks. Usually, for a given task, a collection of nodes can be temporarily and dynamically organized together to form a virtual organization, while each node in the organization plays different role and performs different activity and these nodes should coordinate and cooperate so that the different roles and activities are integrated for achieving the assigned goal.

1.1 Related Issues

The "virtual organizations" has been considered to play an important role in a variety of applications of the Web and Grid communities. It is necessary to discuss and clarify some basic concerns of virtual organizations.

Time dimension. A virtual organization has a life cycle, noted as vo[t_{begin}, t_{end}]. It is created upon a requirement at time t_{begin}. When all the tasks of the virtual organization finish, the virtual organization itself will dissolve at time t_{end}. The lifecycle of a virtual organization provides a mechanism that the resources can be dynamically added to the organization when required and released when not used anymore. Similarly, any component or task in a VO has also a starting time and ending time. Of course, the begin time of any component or task is greater than that of the VO and the end time of any component or task is less than that of the VO.

Workflow. A virtual organization is an organization of task-driven, resource-based, and workflow-managed. Traditional workflow model concerns the execution dependence between tasks [12]. It also concerns the dependence between resources available for the VO and time sequence dependence. A workflow model for the VO represents a series of states and their transitions. In other words, at time t_k, VO_{tk} is a set of states, noted as S_{tk}, with pre-conditions and post-conditions, noted as $PreC_{tk}$ and $PostC_{tk}$ respectively, to form a triple set $<S_{tk}, PreC_{tk}, PostC_{tk}>$. Here the pre-conditions, $PreC_{tk}$, and the post-conditions, $PostC_{tk}$, indicate the required inputs to and the expected outputs from the states S_{tk}. In the life cycle of a VO, a series of states and their transition flows form a state-transition graph.

Resources. Resources are considered to be the most fundamental components [10]. All the activities or tasks carried on in a VO are the process of requesting, comparing, selecting, consuming, integrating and releasing resources. We define two types of resources: sharable and non-sharable. A resource is sharable if it can be used or consumed in more than one task or component at the same time. Such resources are as electronic games, electronic documents, and digital movies. A resource is non-sharable if it can only be used in one task at a time. For example, CPU, memory, and printer are non-sharable resources. In this paper, we only consider the sharable resources because to provide better semantic description to them is more important in the context of the VO. A resource is apparently associated to some actions, such as resource requesting and resource releasing. Therefore, the life cycle of a resource is associated with an action, denoted as Act(Res, t_k), meaning that an action Act is on the resource Res at the time t_k. There are two actions, called resource requesting and resource releasing, stipulating the begin time t_{begin} and the end time t_{end} of the resource. All other actions on this resource at the time t_k, Act(Res, t_k), must follow $t_{begin} < t_k < t_{end}$.

Unlike the interdependence between the workflow states where one state's output is the input to another state (logic dependence), the interdependence between the resources is of semantics, for example, in the relation between Lecture One and Lecture Two. Most of the time, this kind of semantic dependence may not be of time order. Therefore, it is difficult to completely and exactly capture this kind of interdependence relationship between the resources in advance.

Roles and access control. A VO involves various actors (both human beings and machine agents) and different actors play different roles and therefore possess different access rights. Since accesses are given to either resources or processes, the resources and processes will be described with access rights (metadata for access rights). In this paper, we will not consider in detail how to semantically depict the access rights of resources and processes. However, the semantic description of various resources implicitly implies a hierarchy of access rights. For example, a student, not allowed to view a test paper, will not be able to access the test paper resource and the model answers to the test paper as well.

Task and task hierarchy. We define a high level task to be a goal for building up a virtual organization. That is, the description of this high level task is to describe what demand or goal that an end-user expects to meet, such as "to study the Semantic Web". Therefore it is natural that this task will be decomposed into a number of subtasks. Using the decomposition operation, we create a hierarchical structure for tasks. The tasks that cannot be further decomposed are called leave tasks. Each leaf task corresponds to and consumes a set of resources. The process of task decomposition supports forming the workflow of the virtual organization.

1.2 Overview

The paper is organized as follows. In the next section we propose the system architecture for the virtual organization system design, where we briefly discuss various components and their functions in the system. In section 3 we discuss and clarify some basic concepts for virtual organization through the example "virtual course", where we focus on the description of resource, by defining the metadata models for logic resource description and task description and the ontology model for the conceptual description of resources in virtual organizations. In section 4, we discuss the resource storage structure in servers and the task decomposition process in the context of the *Realcourse* application. Finally in section 5 we conclude the paper by discussing our next research work.

2 System Architecture

The system architecture proposed to the virtual organization design consists of two major components: the Task Analysis (TA) component and the Resource Management (RM) component, see Fig. 1.

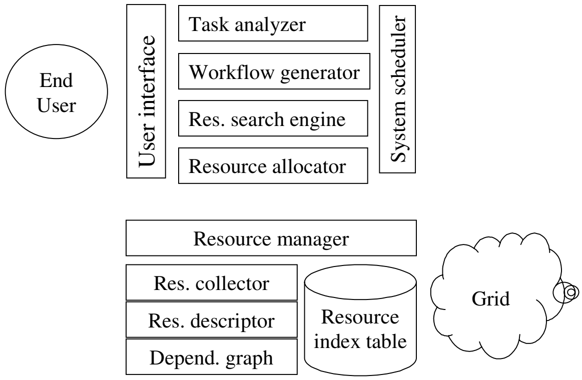

Fig. 1. The system architecture for building a virtual organization

In the Resource Management (RM) component, there are three major function blocks, i.e., the resource collection block, the resource description block, and the resource interdependence graph, and two supporting function blocks: the resource management block and the resource indexing repository. The recourse management block coordinates the functions of the other function blocks, provides an interface to the servers and the resources in the servers, and connects to the Task Analysis component. The resource index table (RIT) saves the results from the resource collection and description blocks, provides a structured organization for the resources, and manages various queries to the resources. The three main function blocks are described below.

Resource collector (RC) block finds and collects all the resources, which are related to a given task, from a distributed environment, extract the contextual information for each resource, and generate for the first round a resource index table.

Resource descriptor (RD) block uses the proposed semantic model for virtual organizations and the extracted resource description information, form a structured, resource associated graph, and generate for the second round a semantic richer resource index repository.

In general, any pair of resources possesses a certain kind of relationships, for example, sequence in time, difference in position, semantic similarity, and so on. The resource interdependence graph (RIG) provides a very general semantic relationship between two resources if they have a certain semantic association. For instance, two resources, “XML course” and “advanced XML course”, may have a semantic similarity relation between them. In this case, one resource, when it is not available, may be replaced by the other because they are semantically similar.

In the Task Analysis (TA) component, there are four main function blocks and two assistant blocks. The assistant blocks, User interface (UI) and System scheduling (SS), deal with the interactions with the end users, such as user queries, obtaining user preferences, and presenting the VO results, and respectively the internal event management and task (workflow) scheduling. The rest four function blocks are explained below.

Task analyzer (TA) block is to take in a task (in the form of user initial requirements), analyze the task and decompose it into a set of subtasks in the form of object-action pairs. The actions serve to the generation of workflow and the objects serve to the Resource Search Engine to form search formulation.

Workflow generator (WG) block uses the actions to produce a state-transition graph, which represents a flow of events/actions with pre-conditions and post-conditions. The state-transition graph will conversely be used to check the user requirements and will require the users to improve their queries accordingly.

Resource search engine (RSE) block receives the objects stated in the tasks, formulate them according to the semantic description model for resources, and finally send the queries to the RM component. The RM component then returns a set of relevant resources and the RSE performs an exact match to find the resource set which better meet the demand.

Resource allocator (RA) block will map the resources from the resource set, the result of the RSE, to the actions and objects in the workflow graph, and generate a resource mapping table (RMT).

3 Semantic Modeling for Virtual Organizations

Undoubtedly, semantics is extremely important in successfully building up a virtual organization and using it. Therefore we discuss where we should make semantic modeling in the context of virtual organizations. Firstly, user demands are very often expressed in a very informal way. So the semantic description model should be able to describe the user demands in a formal manner with less semantic loss. Secondly, resources and components are already stored and described in the Grid. However, their semantics is usually buried in their representation structures and connotations. The semantic description model should be able to capture richer semantics out from the structures and the domain experts' expressions. Thirdly, the model should be able to provide a uniform expression for them. The uniform expression requires the semantic model to be sufficiently general to accommodate various modeling approaches with different semantic foci and precise enough to be able to describe most of the details. The Entity Relationship Approach (ER) [1] and Resource Description Framework (RDF) [8] are currently two main modeling approaches. In this paper we propose a variant schema of ER and RDF, where we adopt the advantage of both modeling approaches.

3.1 Basic Concepts

The main concept to be discussed is "resource". Here we distinguish two kinds of resources, i.e., physical resources and logical resources. A physical resource is an object physically existing in a server, for example, CPU time. A typical property of such physical resources is its status, e.g., whether it is on or off (connected or disconnected to the internet). Usually a physical resource has, as its attributes, a name, an identifier, an access address, a size, etc.

A logical resource is a logical representation of an object. It usually represents a set of distributed physical resources. A logical resource has, as its attributes, a name, an address, a type, and other metadata such as subject. In a virtual organization, a resource refers to a logical one if not explicitly specified. A logical resource usually has a relationship with another logical resource, for example, having the same subject or being similar.

A virtual organization is a set of interrelated resources, which are from different servers. A virtual organization is usually involved in a number of logic structures. On the server side, each server provides a set of physical resources assumedly forming a logical structure, which contributes to the virtual organization. For example, the virtual course "Protégé" includes, as parts of the course, a demo system from Stanford and a series of tutorials from Manchester.

The logic structure for the set of resources on the server side is based on a semantic modeling method, which has two sub-models: metadata model and ontology model.

3.2 Semantic Description Models

There are many resources available on the Grid but most of them are less described and structured. The resources, such as electronic documents, images, movies, sounds, and their interrelations and links, all are overwhelmingly complicated. As the number of resources available on the Grid is explosively growing, this complexity must be managed. Semantic modeling is an effective means to manage complexity in resource information construction. Semantic models can help us to understand the web resources by simplifying some of details [2, 9] and to represent main features and main structures of information and its management.

Physical resources. As we previously described, each server provides a set of physical resources and a virtual organization consists of resources from many servers. Therefore, a collection of servers supporting the virtual organization to operate is denoted to be $S = \{s_1, s_2, \ldots, s_m\}$. Each server contains a collection of resources, denoted to be $R = \{R_1, R_2, \ldots, R_m\}$, where $R_i = \{r_1, r_2, \ldots, r_n\}$ is the resources in the server s_i and r_j is a single resource.

(Physical resource) A physical resource description is defined to be a quintuple: r_j = {**id**, **name**, **type**, **p**, **s**}, where **id** is the internal representation of the resource; **name** is the name of the resource; **type** is the type of the resource, **p** is the physical location, i.e., the path of the resource from the server root; and **s** is the name of the server holding the resource.

The **type** element is an important element for defining a resource. The types contain e.g., multimedia types, textual types, and executable file type. A multimedia type can be mpg, rm, etc. A textual type can be doc, ppt, etc.

For the path (physical location) **p**, we assume a hierarchical structure to be used to organize all the resources in a server. For each resource, there is a path, **p**, from the server root down to the folder directly containing the resource. In this tree structure, a non-leave node is a folder (represented by its name). The root node is the server, **s**, and the leave nodes are the resources. For example, the physical resource description for the resource Lect1.ppt, see the Fig 2, is as follows: {**id**: #sw-lecture-1; **name**: lect1; **type**: ppt; **p**: hudson/public/course/CS/Web Tech/Semantic Web; **s**: hudson;}.

Maintaining this hierarchy aims at acquiring the genuine and natural information of how the users organize and manage their resources. Usually a name the user gives to a resource contains a certain meaning.

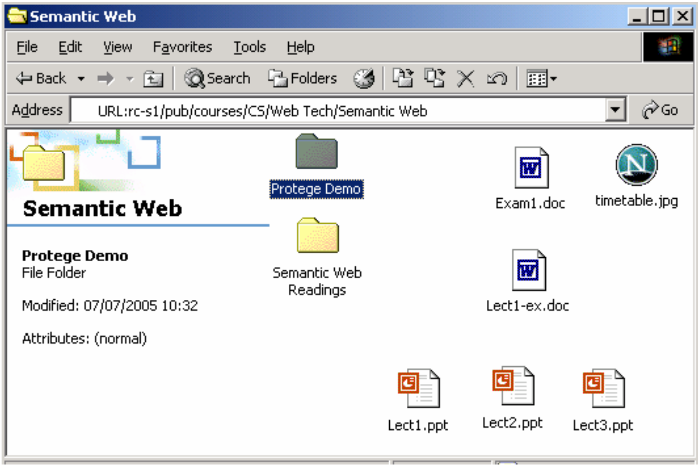

Fig. 2. The folder contains the resources for the Semantic Web course

Metadata model. As we know, the major purpose to create a virtual organization is to find a suitable set of interrelated resources to fulfill the requirements of a given task. Therefore the semantic model should target at resource description, resource integration, and requirement description. The semantic description for resources includes a metadata model and an ontology model.

The metadata model for the logic resource description is a set of normalized metadata, which can identify and describe the resources in VO. In the metadata model, each metadata represents a resource attribute. In our application domain (i.e. the *Realcourse*), the metadata model contains these attributes: subjects (associating with its ontology model), title, description, author, logic relation, length, etc. All the attributes for physical resource description are naturally included in the metadata model.

(Logic resource) A metadata model for a logic resource, **r**, is defined to be a binary <r-id, r-attr>, where r-id is the internal representation of the resource (the same as in the physical resource definition) and r-attr is a list of attributes, which are described as follows:

- **Subject:** topic of the content of the resource. This topic is the reasonably smallest concept, i.e., the leaf node in the ontology
- **Title**: the name of the resource
- **Description:** an account of the content of the resource, usually in natural language
- **Author:** the responsible one for the content of the resource. We also use creator or owner instead of author.
- **Logic relation:** the relations of the resource with other resources. The relations are collected from the Resource Interdependence Graph (RIG), see Chapter 2, which was previously defined by the domain experts.
- **Length:** the play time for the resource.

In the following we use an example (self-explanatory) to illustrate the metadata model for the resource <#sw-lecture-1, r-attr>, where r-attr = {**Subject**: Semantic

Web; **Title**: lecture 1; **Description**: This is the first lecture for the semantic Web course, mainly introducing some basic concepts and discussing background knowledge; **Author**: William; **Logic relation**s: *pre-req*(XML, Internet Tech), *post-req*(Metadata model, Semi-structured data); **Length**: 60 min}.

In a virtual organization, tasks are the major components, which realize the end users' requirements, construct workflows for the requirements, and allocate and assembly the resources. In other words, the tasks represent and formulate the users' demands and requirements at one end and deliver the basic services and consume the resources at the other end. It is obvious that the decomposition operation is a most fundamental one on the tasks. By applying the decomposition operation, the tasks form a tree structure, which supports to construct and schedule workflows for the virtual organization.

(Task) A metadata model for a task, **t**, is defined to be a binary <t-id, t-attr>, where t-id is the internal representation of the task **t** and t-attr is a list of attributes, which are described as follows:

- **<t_{begin}, t_{end}>:** this is a pair of time units to indicate the beginning and the end of the task. So it delimits the life cycle of the task.
- **Description:** a natural language description of the task.
- **Resource set:** a set of resources from various servers, to be consumed by the task.
- **Roles:** a number of actors that play various roles in performing the task, e.g., task trigger, task subject, etc.
- **Decomposition:** this is an operation on the task. The result is a set of subtasks.
- **Rules**: a set of rules, stipulating inputs to and outputs from the task.

In the following we illustrate the task metadata model with an example, see Fig. 3. In the example, the task #012 starts at 16:30 and finishes at 17:30, lasting for one hour which matches the length of the consumed resource lect1.ppt. The task also consumes other two resources lect1-ex (exercises for lecture one) and demo-pres#2 (the second demo presentation, which is located in the folder Protégé Demo, see Fig. 2). This task is further decomposed into three subtasks: #0121, #0122, and #0123. The task is triggered by the outputs of the tasks #011 and #01 and followed by the task #013.

Ontology model. The ontology model is a special structure for a set of resources, where only a special relationship between resources is maintained, for example, the generalization (is-a) relationship. From the point of view of concepts, an ontology model is a tree, where all the nodes are concepts. There is one special node, called the root, which is the most general concept in the tree.

----

t-id:	#012
<t_1, t_2>:	16:30-17:30
description:	this task is to take the lecture 1 and read some materials.
resource set:	lect1.ppt, lect1-ex, demo-pres#02
roles:	triggers <#011, #01>, followers <#013, #121>
decomposition:	<#0121, #0122, #0123>
rules:	<inputs, outputs>

Fig. 3. The example of the tasks

The purpose of defining an ontology model is to provide a referencing conceptual framework for a virtual organization, which we use to reason about, e.g., whether two resources (teacher A and lecturer B) belong to the same concept. The ontology model also supports resource search, formulating search queries for the RSE function block, and resource-task match, refer to Chapter 2.

Following is a widely accepted definition for the ontology. The ontology is defined to be a quintuple: concept (the concept itself), properties (all the relationships of the concept with other concepts and the attributes of the concept), axioms (rules and constraints on the concept), value domains and nominal (names or terms for the concept). However, based on our investigation on the application domain, we redefine the ontology to better accommodate the features of the *Realcourse* (the virtual course).

(Ontology model) An ontology model for the resources is defined as follows:

Concept: a concept name is used to represent the concept of a resource.

Properties: a set of properties of the concept, e.g., location of the resource, difficulty level.

Constraints: the constraints on the concept of the resource, e.g., *disjoint* (A, B), *overlapping* (A, B) where A and B are two concepts.

Relationships: the semantic relations between two concepts, e.g., *similar* (A, B).

Internal structure: *subClassOf*, *partOf*. This forms an ontology tree for the concept of the resource.

In the following we illustrate the ontology model through describing the course Web Technology Course, see Fig. 4.

Concept: Web Technology Course (in short, WT)
Properties: location(WT) = “dcs/courses/CS/WT/WT1”, difficulty(WT) = “beginner”
Constraints: disjoint(WT, OS), overlapping(WT, DB), where OS and DB are the course resources Operating Systems and Database Systems.
Relationships: relevant(WT, programming), similar(WT, Internet Tech)
Internal structure: WT(XML(X1), Web Engineering(WE1), Semantic Web(SW1, SW2), Web Services(WS1))

Fig. 4. An ontology model for description of the resource Web Technology Course

4 A *Realcourse* Case

University Course Online (in short, the *Realcourse*), composed of 20 servers, is a Course service supported by a collection of physical servers distributed on CERNET all over China [11]. Besides the servers that share the single URL, there are more than 800 hours of quality course services available. These services are from different universities, uploaded to different servers with a permanent backup copy created in a server other than the original one, and flow from one server to another. If one server is down, its duty is automatically taken up by some other server. In this circumstance, each server provides a number of resources for a course (usually all the resources required for a course are in one server). However, with the fast growing of the capacity of resources and variation of courses, it will be difficult and unreasonable for one server to contain all the required resources for a “single” course.

We use a metadata model to provide semantic description for resources and courses. Other than the elements which are similar to the Dublin Core elements, the metadata model also provides description for the tasks which form various workflows to realize the virtual organization for the *Realcourse*. The description for a course is more complicated. For example, a course may consist of a lot of resources. These resources include some training materials such as video clips and sound clips, a dozen of presentations (in PPT or PDF form), many research papers, articles, standards specifications documents, and other textual materials, some tools with user manuals and reference books.

4.1 Logic Structure for the "Virtual Course"

In the following, we briefly illustrate the content of the structure of virtual courses in a server and the process of task decomposition. See Fig. 5.

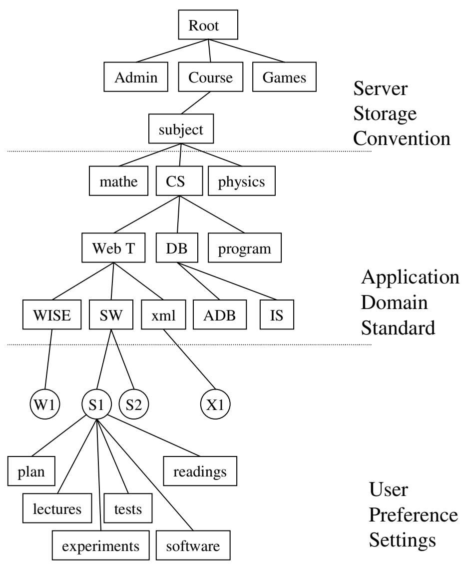

Fig. 5. The example shows the resources in a server for various courses

A server, providing various resources about learning materials, can be described as a tree structure, which has three layers in building up it, i.e., server storage convention, application domain standard, and user preference settings. We assume that the server storage convention maintains a similar structure for all the files and folders, like in a Unix system. For each resource, as an item of descriptive information, we need the path from the root (e.g., the server name) to the physical object name.

However, semantic richer information comes from the lower layers, i.e., the application domain standard (e.g., library subject category) and the user settings, where the latter contains more informal semantics. In the reality, we can obtain this kind of ontology from any standard body. For example, an ordinary library subject category (LSC) is this kind of ontology model for learning purpose. The user settings provide more semantic connotations for the concepts of resources but they are quite informal. Currently we manage this part manually. For example, the users need to prepare a resource interdependence graph (RIG) to provide semantic relationships between the resources.

4.2 Task Decomposition Process

Assume that an objective for a virtual organization is "to build up a course on Semantic Web from the existing courses" with a set of requirements "for beginners", "in English/Chinese", "12 hours long", and "self-testing". We take the objective as the first, very general task, T0. Then T0 is decomposed into two subtasks: subject search T10 ("Semantic Web") and attribute match T11 (attributes). Using the ontology in the system, we find *subClassOf*("Semantic Web", "Web Technology") and *similar*("Semantic Web", "Advanced Information Systems"). Then T11 is decomposed into the following subtasks: resource search T20, task-resource match T21, task scheduling T22, and attribute tuning T23.

The task T20, resource search, attempts to find all the resources available in the ontology internal structure. The task T21, task-resource match, collects the found resources and matches them to the subtasks (further decomposition may be needed for better matches.) and generate a resource allocation table. The task T22 is to arrange all the tasks and form an executive sequence (workflow) in time order. Finally the task T23, attribute tuning, checks whether two or more similar resources are serving the same task. If so, the requirements for the resources need further analysis and users' interference is required.

5 Conclusion

The 'Virtual organization' has become an important issue in many application areas, such as semantic grid computing, semantic web, e-Learning, e-Government, and e-Business. To make best use of the resources distributed over the Grid environment, we need to find the most suitable resources, to best meet the users' requirements, to most effectively use the resources, to efficiently integrate the resources to form a virtual organization on demand. The first step toward the above mentioned objective is to use a semantic model for resource description. In this paper we proposed a metadata model for resource description and for task description. We also proposed an ontology model for resource conceptual description.

We use the *Realcourse* as a testing application to build up a virtual course, which aims to realize some important aspects of a virtual organization, including semantic extraction and modeling from resources, ontology design and use.

When developing the semantic models, we find that a further investigation on the behavior of the tasks for resource management is crucial. Our next step is to formalize the semantic model for virtual organization resources so that a deeper analysis of various tasks and their internal features and structures is very important. In addition, we will develop an automatic task development and analysis mechanism based on the *Realcourse* to support the virtual course design.

References

1. Chen P.: The Entity-Relationship Model - Toward a Unified View of Data. ACM Transaction on Database Systems 1, 1 (1976) 9-36
2. Conallen J.: Modeling Web Application Architectures with UML, Communications of the ACM, Vol. 42, No. 10 (1999)
3. Foster I, Kesselman C, Nick J, and Tuecke S.: The physiology of the grid: open grid services architecture for distributed systems integration, GGF4 (2002)
4. Gobel C and de Rouce D.: The Grid: an application of the semantic web, SIGMOD Record Vol. 31 No. 4, ACM (2002)
5. Horrocks I,: DAML+OIL: a reason-able web ontology language, in the Proceedings of EDBT (2002)
6. “ICENI Virtual Organisation Management”, London e-Science Centre, UK (2001)
7. Introna L, Moore H, Cushman M.: The Virtual Organisation – Technical or Social Innovation? Lessons from the Film Industry, Working Paper Series 72, Department of Information Systems, London School of Economics and Political Science, UK (2002)
8. Klyne G and Carroll J.: Resource Description Framework (RDF): Concepts and Abstract Syntax, W3C, http://www.w3.org/TR/rdf-concepts/ (2004)
9. Song W.: Semantic Issues in the Grid computing and Web Services, in the Proceedings of International Conference on Management of e-Commerce and e-Government, Nanchang, China (2003)
10. Song W and Li X.: Semantic Modeling for Virtual Organization: A Case for Virtual Course, to appear in Advances in Information Systems Development - Bridging the Gap between Academia and Industry, (eds.) Nilsson A. and Gustas R., Springer (2005)
11. Zhang J-Y and Li X-M.: The Model, Architecture and Mechanism Behind *Realcourse*, the Proceedings of International Symposium on Parallel and Distributed Applications (2004).
12. Zhuge, H.: Component-based workflow systems development, Decision Support Systems, Vol. 35, Elsevier (2003) 517-536