Framework for Domain Analysis of Teleteaching System: A Semiformal Approach

Animesh Dutta¹, Imon Banerjee², Shrutilipi Bhattacharjee³, Ranjan Dasgupta⁴, Swapan Bhattacharya⁵

¹Dept. of IT, NIT, Durgapur, India, ²Dept of CSE, NITTR, Kolkata, India, ³Dept of CSE, NIT Durgapur, India
¹animeshrcc@gmail.com , ²imonban@gmail.com, ³shrutilipi.2007@gmail.com
⁴ranjandasgupta@hotmail.com, ⁵bwapan2000@yahoo.co.in

Abstract- Understanding user requirement is an integral part of information system design and is critical to the success of interactive systems. It is now widely understood that successful systems and products begin with a clear understanding of the domain of the system, and needs and requirements of the users. There are several purposes for modeling and analyzing the problem domain before starting the software requirement analysis. First it focuses on the problem domain so that the domain users could be involved easily. Secondly a comprehensive description on the problem domain will advantage getting a comprehensive software requirement model. This paper model an ontology based framework to satisfy the criteria mainly organizational structure, agent’s interaction, goal achievement. In this paper we show a very popular interactive system “Teleteaching” as the case study for the proposed framework.

Keywords- Ontology, Conceptualization, Teleteaching, Domain Analysis, Requirement Engineering.

1 Introduction

Several surveys indicate that a significant percentage of software fail to meet business objectives or are outright failure. One of the reasons for this is that domain and requirement analysis is typically overlooked in real life software projects. Good requirement practices can accelerate software development. The process of defining business requirements aligns the stakeholders with shared vision, goals and expectations. Substantial user involvement in establishing and managing changes to agree upon requirements increases the accuracy of requirements, ensuring that the functionality built will enable users to perform essential business tasks. Software requirement engineering encompasses the two major sub domains of requirement definition and requirement management:

Requirement definition is the collaborative process of collecting, documenting and validating a set of requirements that constitute an agreement among key project stakeholders. Requirements definition is further subdivided into the critical process areas of elicitations, analysis, specification and validation process. From a pragmatic perspective, requirement definition strives for requirements that are user validated and clear enough to allow the team to proceed with design, construction and testing at an acceptable level of risks.

Requirement management involves working with a defined set of product requirements throughout the product’s development process and its operational life. It also includes managing changes to that set of requirements throughout the project lifecycle. An effective requirement definition and management solution creates accurate and complete system requirements while helping organization communications is an effort to better align IT with business needs and objectives.

The paradigm shifts from mainframes in the 1950s-1970s to the personal computer technology of the 1980s and the networked computing devices of the 2000s can be conceptually modeled by paradigm shifts from algorithms to sequential interaction and then to distributed (multiagent, collaborative) interaction. Interactive models provide a domain independent unifying view of the design space for software intensive systems. The key dimension of change in software-intensive systems is interactivity. Whereas at one time computing was mostly batch the execution of algorithms, with all input data supplied a priori and all output generated afterword-today ongoing interaction is ubiquitous. This interaction differs semantically from iterated batch computing in that each computing entity maintains a persistent state that evolves and enables computations to be driven by their entire input/output histories.

One popular application “Teleteaching” is fundamentally interactive and can not be properly understood until more research is done on domain-independent models and on the principles of interactions.

The requirements of a software system are often ambiguous, incomplete, redundant and variant at the requirement analysis phase of a system. Requirement generated by different stakeholders may be inconsistent since different stakeholders and users may have different perspective on the system. This may bring about the same term is applied to different concepts and different terms are used to denote the same entity. A suitable solution was the use of ontology which can make sure that different designers have a common understanding of the term to be used.

Ontology was originally concept from philosophy. It describes the essence and composition of the world, as said by the philosophers. In computer science, ontology is mainly used for knowledge representation and defining the concept of the system It provides a means for knowledge sharing which is very much necessary for large scale, complex real software projects. For the last few years more and more software engineers become interested in the research system designing using ontology concept [1]. The five points
presented in the paper [2] for using ontology in requirement analysis are:

- Make relations as independent knowledge units.
- Organize the objects in ontologies.
- Take objects as basic components of ontologies.
- Let ontologies form their own inheritance hierarchies.
- Allow ontologies to be nested.

For the same reason we are also interested to design a semi formal conceptual framework to specify the requirements of different stakeholders for a large complicated interactive system. Our paper is structured as follows. In section2, it describes the related works and surveys of this area. Section3 gives the formal definition of ontology. In section4, an innovative semi formal framework is proposed for requirement specification of interactive software system. In section5 as a case study of our proposed framework, a very popular example of teleteaching systems is shown. Section6 concludes our work.

2 Related Works

The subject of Teleteaching in distributed IP network has engaged researches all over the world. Many solutions have been proposed for distance learning and collaboration over the Internet. Indicative commercial tools for collaboration over the Internet are Microsoft NetMeeting and WhitePine Enhanced CU-SeeMe (WhitePine, http://www.wpine.com). Various methods have been proposed for synchronous learning [3-5], asynchronous learning [6,7] or asynchronous learning with an on-line facilitator (Wang and Karmouch) [8,9]. Various models for collaborative Systems that come to cover the communicational needs of collaborative work in a learning System either synchronous [10,11], or asynchronous [12] have also been proposed. One of the very serious synchronization issues, group synchronization for conferencing in virtual room has been taken care in [13,16,17]. Agent based goal oriented requirements has been analyzed in [14] and some interaction between the agents in teleteaching domain has been shown [15].

In particular, the main problem with these above tools is their lack in term of QoS (Quality of service) guarantees. Among all the QoS parameters the most meaningful seem to be the audio quality that can currently range in a single work session from correct to incomprehensible (while it has to be constant), the end to end delay that has to be as short as possible to increase the interactivity level between users, and all temporal synchronization constraints. In fact, users are really disturbed when the temporal features of the audio and video streams are not enforced, as voice can be hard or impossible to understand, and the lack of lips synchronization makes the audio / video correlation disappear. Besides, multimedia synchronization is currently the key point to address and the most difficult to solve for designing multimedia conferencing systems.

In this paper we propose an ontology based conceptual framework for requirement analysis of a large and complex Teleteaching system.

3 Ontology

In computer science and information science, ontology is a formal representation of a set of concepts within a domain and the relationships between those concepts. It is used to reason about the properties of that domain, and may be used to define the domain.

3.1 Conceptualization

Conceptualization can be defined as structure consisting of domain D and their interrelationship R and can be written as <D,R>. It actually refers to the ordinary mathematical relation on D that is extensional relation. But the main difference of ordinary and conceptual relation lies in the working space of those. While ordinary relations are defined on a certain domain and conceptual relations are defined on a domain space. We shall define a domain space as a structure <D, W>, where D is a domain and W is a set of maximal states of affairs of such domain. Now a conceptual relation \( \rho^a \) of arity \( n \) on <D, W> as a total function \( \rho^a : W \rightarrow 2^n \) from W into the set of all n-ary (ordinary) relations on D. A conceptualization for D can be now defined as an ordered triple \( C = <D, W, R> \), is a set of conceptual relations on the domain space \( \Gamma \) where \( \forall <D, W> \). We can say therefore that a conceptualization is a set of conceptual relations defined on a domain space. Let \( C = <D, W, R> \) be a conceptualization. For each possible world \( w \in W \), the intended structure of \( w \) according to \( C \) is the structure \( S_C = <D, R_C> \), where \( R_C = \{ \rho(w) | \rho \in R \} \) is the set of extensions (relative to \( w \)) of the elements of \( R \). We shall denote with \( S_C \) the set \( \{ S_C | w \in W \} \) all the intended world structures of \( C \).

3.2 Formal Ontology

An ontology defines the basic terms and relation comprising the vocabulary of a topic area as well as the rules for combining terms and relation to define extension to the vocabulary[18].Ontology basically define as a formal specification of shared conceptualization. Shared reflects the notation that an Ontology captures consensual knowledge that is, it is not private of some individual but accepted by a group. Ontology is a logical theory which gives an explicit and partial account of conceptualization. For a conceptualization \( C \), Let O is a ontology with language L where Ontological commitment \( K = <C, \Gamma> \).

![Figure 1: Formal Ontology](image)

Conceptualization = \{ C_1, C_2, C_3 \}
4 Teleteaching System

Advances in telecom technologies and the wide reach of broadband internet enables us to offer our expertise and acknowledged education resources and skills to any person, anywhere on the planet. Teleteaching system need a high degree of interaction between each and every entity within the system. So we choose requirement specification of teleteaching as case study.

Teleteaching can be regarded as the process of distance learning where we are trying to provide a virtual class environment. Here administrator serves as Distance Learning Organization (DLO). DLO first selecting a course after checking their infrastructure. According to the course need, they recruit Teacher in a geographical independent manner. One Teacher is allow to take synchronous lecture at a time. A Student register himself in a Distance Learning Organization (DLO), for a particular course. Then Organization will provide a routine to all the registered student. A teleteaching lecture should be as close as possible to a conventional lecture, to guarantee the same success in leaning. Teleteaching has a target of development and adapt special methods and techniques for the increase of quality and effectiveness of the teaching system.

Synchronous Event: Synchronous event is timekeeping which requires the coordination of events to operate a system in unison. Session operating with all their parts in synchrony are said to be synchronous session.

Teleteaching Synchronous Session can be divided into three types

Lecture Session: A lecture is an oral presentation intended to present information or teach people about a particular subject by the teacher and a set of enlisted student will allow to join that session in geographical independent manner. This is a interactive session, where student can interact with the teacher synchronously. In this scenario Teacher’s workspace is accessible by all students at a time.

Test Session: In this session Teacher provide series of questions to determine knowledge, intelligence, or ability of the Student. Here Exam Space used as a Student’s workspace but with a predefine time limit.

Viva Session: It is an another aspect of assessment of students’ knowledge & intelligent like Test Session. In this session Student and Teacher directly interact for specific time period in one to one mode.

Asynchronous Event: Asynchronous events are those occurring independently of time schedule.

Assignment Upload:- Registered Student can upload their assignment online provided by the teacher in the organizational database. Teacher gives assignment after specific lecture session. It is a asynchronous event.

Table 1: Teleteaching domain Symbols

<table>
<thead>
<tr>
<th>Symbols</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Student</td>
<td>Class</td>
</tr>
<tr>
<td>Teacher</td>
<td>Object</td>
</tr>
</tbody>
</table>

Figure 2: Ontological element of Teleteaching

4.1 Teleteaching Entity Schema

Schema:

Entity: Teacher
Class: Teacher ∈ Registered_Agent
Attribute: Teacher_Recruitment → Organization.Teacher
Relation: Teacher ∩ Student ∩ Admin = Registered_Agent
Teacher ∩ Student = φ
Teacher ∩ Admin ≠ φ

Function: ∀ Teacher P₁(Teacher) → Organization_event
∪ Ex_Event
P₁(x): x is triggering event
∀ Teacher P₂(Teacher) → true
P₂(y): y is accessing some organizational event

Restriction: ∃ x, ∀ y P(x,y) = true
x = Teacher ∩ Admin, y = Admin.Aids
P(x,y) : x cannot access y

Schema:

Entity: Student
Class: Student ∈ Registered_Agent
Attribute: Student_Selection → Organization.Student
Relation: Teacher ∩ Student ∩ Admin = Registered_Agent
Student ∩ (Teacher ∩ Admin) = φ

Function: ∀ Student P₁(Student) → Organization_event
P₁(x): x is accessing an event.

Restriction: ∀ x, ∀ y P(x,y) = true
x = Student, y = Admin.Aids ∩ Teacher.Aids
P(x,y) : x cannot access y
Relation: \( \forall y \ \forall x \ P_2(x,y) = true \). \( x=\text{Teacher}, y=\text{Lecture Session}. \)

Relation: \( \exists y \ \exists x \ P_2(x,y) = true \). \( x=\text{Admin}, y=\text{Test}. \)

Relation: \( \forall y \ \forall x \ P_2(x,y) = true \). \( x=\text{Teacher} \cup \text{Student}, y=\text{Viva}. \)

P_3(x,y): x is the main coordinator of y.

Restriction: \( P_3(\neg x) = false \).

P_3(x): Synchronous event can occur only at x. \( x=\text{Predefine time schedule}. \)

Schema:

Entity: Asynchronous mode.

Class: Asynchronous mode \( \in \) Mode of operation.

Attributes: : Assignment \( \cup \) Ondemand_Download = Asynchronous_Event.

\( \forall \text{time} \ \forall \text{Asynchronous event}, \)

\( P_1(\text{time}, \text{Asynchronous_event}) \rightarrow \text{True}. \)

\( P_2(x,y): \ \text{x can occur at x}. \)

Relation: \( P_2(x): \text{Registered_Agent can works on x} \)

\( \forall x \ P_2(x) = true, x: \text{Asynchronous teaching mode}. \)

4.2 Requirements of Teletaching

Overall Usability:

1. Facility for showing power point presentation and other documents during the virtual lecture session from any node to all other nodes.

2. Reliable Video Communication – This is paramount. If the teacher or student cannot reliably interact, the product will not be worthwhile. Video should stream seamlessly with as few interruptions as possible.

3. Reliable Audio Communication – Also paramount for our users. Audio should also be streamed as smoothly as possible. All students should be able to hear each other as well as the teacher.

4. Simple User Interface – Students should be able to access the teacher’s classroom easily and quickly. Teachers should be able to set up a virtual classroom with the push of a button.

5. Ease of setup – Student should be able to plug in their computer to an external television or monitor, have a webcam/microphone, and install the software and be set to interact. Teachers should only have to install the software and setup their classroom environment.

6. Authentication- Authentication is for each member of a lecture session to prove to the others that he is indeed who he says he is.

7. Interactive Group Synchronization- It is the synchronization of audio signal generated from different user. This synchronization is necessary to minimize the audio signal overlap by choosing one user (student/Teacher) at a time as speaker and other as listener.
8. Multipoint Synchronization: Apart from interactive group synchronization, there is a requirement of another type of synchronization which is multipoint or inter-destination synchronization involving the synchronization of the play-out processes of same streams in different receivers, at the same time, to achieve fairness among the receivers.

Teacher Specific Usability:
1. Ability to see all students – As per wish, the teacher must be able to see each of his students who are connected at the lecture session. All their video feeds will be live along with live audio.
2. Ability to access student’s workspace – As per wish, the teacher must be able to see and access (write) any student’s workspace at any point of time during the lecture session, and if he (teacher) feels he can show any student’s workspace to all other students.
3. Attention grabber – Teachers should be able to get the attention of particular students. Teachers should only need a minimal number of mouse clicks to get the desired students’ attention.
4. Ability to Mute – If a teacher feels the need to mute a student or stop his/her video/audio feed, the teacher should be able to do so with ease.
5. Automatic Attendance/Grading – The program should automatically keep track of who is present within a classroom session. Teachers can also see the attendance at a glance. There should also be a grading protocol so that the teacher can find out if a student answers accurately and modify her teaching toward specific students.
6. Study material upload facility – The teacher should have the facility to upload text/audio study materials.

Student Specific Usability:
1. Classroom experience – Students will have an environment as close to a real classroom as possible. Students should be able to hear as well as see all other students. Students will also be able to see the teacher’s workspace.
2. Ability to see teacher’s workspace – The teacher’s workspace must be visible to all students during the lecture session.
3. Grab the teacher’s attention – Students will need to be able to get the teachers’ attention if they need assistance. This must be easy to do, and visual notification must be given to the student that they have requested the teachers help.
4. Interaction with the teacher – Students have the ability to use jelly bean response buttons to interact with the teacher. This will have to be addressed with the addition of external hardware that must interface with the program.
5. Study material download and assignment submission – Students should have the facility to download the study material and also can submit their assignment.

4.3 Models of Teleteaching

<table>
<thead>
<tr>
<th>Symbols</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Node" /></td>
<td>Node</td>
</tr>
<tr>
<td><img src="image" alt="Dependency" /></td>
<td>Dependency Relation</td>
</tr>
<tr>
<td><img src="image" alt="Decomposition" /></td>
<td>Decomposition Relation</td>
</tr>
</tbody>
</table>

Table 2: Symbols of Teleteaching Models

Interaction model:

![Figure 3: Interaction Model](image)

Depends on: Lecture_session → Teacher × Student × Time_schedule.
Test → Admin × Student × Time.
Viva → Teacher × Student × Time.
Ondemand_download → Registered_Agent × Resource.
Assignment_upload → Student × Resource.

Mode of operation: Asynchronous ☑ Synchronous.
Operates on: Synchronous → Lecture_Session ☑ Test ☑ Viva.
Asynchronous → Ondemand_download ☑ Assignment_Uploa d

Resource Management Model:

![Figure 4: Resource Management Model](image)

Node: (Student_WS ☑ Teacher_WS ☑ Audio ☑ Video ☑ Class_notes) ☑ Information ☑ (Student ☑ Teacher ☑ Admin ☑ Ex_Agent).

Supports: Lecture_Session ∪ Test ∪ Viva ∪ Ondemand_download ∪ Assignment_Upload ∪ Ex_Event.

Belongs to: Resource → Organization.

Teleteaching Goal Model:

Goal: Organization_Goal ∪ Agent_Goal

Organization_Goal: Improvement_of_teaching_method ∪ Satisfy_Students' need ∪ User_friendly

Teachers' Goal: Audible_to_all_students ∪ Visible_to_all_students ∪ location_Independence_of_teaching ∪ Visibility_of_Teacher's_WS

Students' Goal: Teacher's_visibility/audibility ∪ location_Independence_of_learning ∪ Visibility_of_Student_WS

Node: Goal ∪ Organization ∪ Teacher ∪ Student

Achieved by: Goal → (Organization ∪ Registered_Agent) × Organizational_Event × Resource

Event Activity flow Model:

Node: (Student_Selection ∪ Recruiting_teacher ∪ Selecting_course ∪ External_Event ∪ Teacher ∪ Student ∪ Admin ∪ External_Agent ∪ Information ∪ Synchronous ∪ Asynchronous)

Trigger Link: Teacher × Admin × Asynchronous → Student_Selection

Admin × Asynchronous → Teacher_Selection

4.2 Concept Space of TeleTeaching

Figure 5: Goal Model

Figure 7: Concept Space of Teleteaching

Formalization of Concept Space:
Concept space C_S is consisting of 7 tuples, i.e.


C = Concept:

A=Association:

R= Relation:

E_CR = Set of edges from concept to relation. Edge in which concept is related by a specific relation. It is an [1:1] mapping.

E_CA = Set of edges from concept to association. Edge in which concept is achieved by hierarchical relationship. It is an [n:1] mapping.

E_AC = Set of edges from association to concept. It is a [1:1] relationship from association to concept.

E_RC = Set of edges from relation to concept. Edge in which any relation is related to a specific concept. It is an [1:1] mapping.

E_RC ∩ E_CR ∩ R ∩ C → Relationship between concepts.
\[ E_{AC} \land E_{CA} \land A \land C \rightarrow \text{Hierarchical relationship.} \]
\[ E_{AC} \cap E_{CA} = \emptyset \]
\[ E_{RC} \cap E_{CR} = \emptyset \]

5 Conclusion

We analyzed several systems and present requirements for a teleteaching system. We consider an internet based teleteaching system to be an improvement over traditional distance education since the degree of interaction is much higher and the student get a feeling of being present in the classroom. In this paper we present an ontology based semiformal framework to analyze the working domain of the system particularly the behavior and the interaction of the stakeholders. Our future work will include the designing of software Therefore we believe that this teleteaching system is valuable extension to our traditional classroom.

6 References


