Ontology Based Semiformal design of Teleteaching System

Animesh Dutta
Dept. of I.T.
NIT Durgapur
Durgapur, India
animeshrec@gmail.com

Vivek Nair
Dept. of I.T.
NIT Durgapur
Durgapur, India
vivekaxl@gmail.com

Ranjan Dasgupta
Dept. of C.S.E.
NITTTR Kolkata
Kolkata, India
ranjandasgupta@hotmail.com

Swapan Bhattacharya
Dept. of C.S.E.
NIT Durgapur
Durgapur, India
bswapan2000@yahoo.com

Abstract - Teleteaching has become a major tool to impart education, where it introduces a concept of geographical independence. In this paper we design an environment of a university where in all the activities can be done “online”. We use ontology to design the environment by describing, structuring, defining the environment.

Keywords—Ontology, Conceptualization, Teleteaching, Architecture, Design.

I. INTRODUCTION

Teleteaching has become a major tool to impart education in many countries, where the student doesn’t have to think about the geographical boundaries while going studying or learning. In comfort of their homes the students have greater efficiency. Teleteaching has become a major tool, in areas where lifelong learning is necessary i.e. Companies, universities etc. Teleteaching has in it the provisions which cater to the needs of the teachers and students alike, which enhances web learning experience. Teachers can design new courses, use various models as templates making work easy for them and reuses the existing data. The same resources can be combined with others to set up different courses. Also, more student groups can learn many courses at the same time. Teleteaching can be helpful for students in both the self-paced (on demand Downloads) and the instructor-led learning processes (lecture Sessions).

Here in this paper we try to design an environment of a virtual university, where there are activities like Lecture sessions, evaluation of students and administrative activities. Here the students have to register themselves, select courses, collect their schedule, attend classes and sit for evaluation. Teachers and admin would take care of the administrative duties. We try to formalize an environment where there are n numbers of students and m courses spread over various geographical locations. The same case holds case of Teachers. \( N_1, N_2, \ldots, N_m \) are subsets of N which The students apply to the institute after looking at the notice board. These subsets can overlap as in the subset of students can apply to more than one course. Same is for the teachers as well they can take more than one course and also participate in administration activities.

The term ontology has been widely used in recent years in the field of Artificial Intelligence, computer and information science especially in domains such as, cooperative information systems, database management systems etc[1]. An ontology is an explicit specification of a conceptualization[2]. They can be useful for systems which have to interact (interoperability) and for the development process itself (reusability, reliability and specification process). We use ontology in a threefold manner:

1. describing the semantics of the process,
2. structuring activities and communication facilities, and
3. defining the context and the environment of teleteaching.

Ontology = (Concepts; Relations; Restriction; Actions)

Relation consist of
i. Is a part of \( \rightarrow \) Task Decomposition
ii. Association \( \rightarrow \) Dependency.

II. RELATED WORK

A. Basic learning method:

Learning methods are specialized methodologies derived from pedagogical theories which guide the creation of courses by identifying activities that provide a better understanding of the topic under study adapted to various learning styles.

Behaviorist learning:

Goal: Efficiently transmit knowledge from the instructor to the learner.[1,2]. Instructors are at the center, which means that they play a major role in learning activities.
Learners can’t express ideas because it doesn’t engage the mind properly. **Constructivist Learning:**

Goal: Let the learners construct their own knowledge through active construction rather than a passive one. [1,2,4,5]. Takes learners prior learner as a base to build further knowledge. Concentrates on solving practical problems. “Learner Centered” model because learners themselves discover knowledge while solving problems.

**Collaborative Learning:**

Goal: Learners interacts with study materials and hence learns.[1,5,6]. Expects that interaction of learner with others (teachers and students) would lead to learning.

We need to use a optimized mix of these strategies to make a method suitable for the environment of learning. B.

**B. Telteaching:**

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.

One of the very serious synchronization issues, group synchronization for conferencing in virtual class room for telelecture has been taken care in [14,17,18]. Agent based goal oriented requirements has been analyzed in [15] and some interactivity between the agents in teleteaching domain has been shown [16].

**C. Ontology:**

Ontology [19][20] is collection of definitions of entities and the relationships between them [21]. The formalization of ontology was described in [22,23], but their abilities of reasoning were inefficient and undecidable particularly for complex interactive system. This approach will give a valid tool for the learning process [24] and the ontologies represent conceptual descriptions of the specific content as they help to identify appropriate items and relationships in a given knowledge.

### III. ONTOLOGY OF TELETEACHING DOMAIN

**Fig 2: Ontology Diagram**

In the ideal case the modeling entities should be exactly at the level of abstraction that entirely covers the domain of the software application. “An ontology is an explicit specification of a conceptualization” [10]. Ontological categories define the concepts that exist in the domain, as well as relationships between these concepts.[11]

**Schema Definition**

**1) Schema:**

<table>
<thead>
<tr>
<th>Entity:</th>
<th>Teacher</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class:</td>
<td>Registered_Agent</td>
</tr>
<tr>
<td>Attribute:</td>
<td>Teacher_Recruitment ↔ Organization.Teacher</td>
</tr>
<tr>
<td>Relation:</td>
<td>Teacher ∪ Student ∪ Admin = Registered_Agent</td>
</tr>
<tr>
<td>Function:</td>
<td>∀ Teacher P1(Teacher) → Organization_event ∪ Ex_Event</td>
</tr>
<tr>
<td>Restriction:</td>
<td>∃ x, ∀ y P(x,y) = true</td>
</tr>
</tbody>
</table>

**2) Schema:**

<table>
<thead>
<tr>
<th>Entity:</th>
<th>Student</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class:</td>
<td>Student ∈ Registered_Agent</td>
</tr>
<tr>
<td>Attribute:</td>
<td>Student_Selection ↔ Organization.Student</td>
</tr>
<tr>
<td>Relation:</td>
<td>Teacher ∪ Student ∪ Admin = Registered_Agent</td>
</tr>
<tr>
<td>Function:</td>
<td>∀ Student P1(Student) → Organizational_Event</td>
</tr>
<tr>
<td>Restriction:</td>
<td>∀ x, ∀ y P(x,y) = true</td>
</tr>
</tbody>
</table>

**3) Schema:**

<table>
<thead>
<tr>
<th>Entity:</th>
<th>Admin</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class:</td>
<td>Admin.Aids ∪ Teacher.Aids</td>
</tr>
<tr>
<td>Function:</td>
<td>P(x,y) : x cannot access y</td>
</tr>
<tr>
<td>Restriction:</td>
<td>∀ x, ∀ y P(x,y) = true</td>
</tr>
</tbody>
</table>

**Conclusion:**

By using a combination of constructs and collaborative learning, we can create a method that is suitable for the environment of learning. Teleteaching has engaged researches all over the world, and many solutions have been proposed for distance learning and collaboration over the Internet. The synchronization of group synchronization for conferencing in virtual class room for telelecture has been taken care of. Agent based goal oriented requirements has been analyzed in the literature. The ontologies represent conceptual descriptions of the specific content as they help to identify appropriate items and relationships in a given knowledge domain.
Entity: Admin
Class: Admin ∈ Registered_Agent
Relation: Teacher ∪ Student ∪ Admin = Registered_Agent
Student ∩ Admin = φ
Admin ∩ Teacher ≠ φ
Function: ∀ Admin P1(Admin) →
Organization_Event ∪ Ex_Event
∀ Admin P2(Admin) → True
P1(x): x Triggering an Event
P2(x): x can access all resources

3) Schema:
Entity: Teaching_Aids.
Class: Teaching_Aids ∈ Organization.Resource.
Attributes: ∀ Aids, P1(Aids) → True.
P1(x): x is accessed by Registered_Agent.
Function: P2(x,y): x is accessed by y
i) P2(x,y)= true x= Teacher_WS, y= Teacher
ii) P2(x,y)= true x= Teacher_WS, y= Teacher
iii) P2(x,y)= true x= Audio ∪ Video ∪ Class_notes
iv) P2(x,y)= true x=C_WS,
y=Teacher ∪ Student
Restriction: P3(Organization.Aids) → False.
P3(x): x is accessed by External_Agent.
∀ Teacher_WS P2(Teacher_WS) = false
∀ x P2(x) = true x: Asynchronous teaching mode.

4) Schema:
Entity: Organizational_Event.
Class: Organizational_Event ∈ Event.
Attributes: P(x,y) : x can be accessed by y
i) ∀ x ∀ y P(x,y) = true y = Admin, x = Organizational_Event
ii) ∃ x ∀ y P(x,y) = true y = Student, x = Organizational_Event
Relation: Mode_Organizational_Event →
(Synchronous ∪ Asynchronous)
Restriction: P3(Organizational_Event, Organizational_Event) → Event.
P3(x): On triggering x.

5) Schema:
Class: Event
Attributes: Exeevent → ∀ Agent, P1(Agent).
Exeevent → ∃ Exeevent, P2(new(Exeevent)).
P1(x): x can access.
P2(x): on triggering x is generated.
Relation: Exeevent ∩ Inevent= φ.
Mode.Exeevent → Asynchronous.

6) Schema:
Entity: Synchronous mode.
Class: Synchronous mode ∈ Mode of operation.
Attributes: Lecture ∪ Test ∪ Viva = Synchronous_Event.
∃ time ∀ Synchronous event,
P1(time,Synchronous event) → True.
P1(x,y): y can occur at x.
Relation: P2(x,y): x is the main coordinator of y.
i) ∀ y ∀ x P2(x,y) = true. x=Teacher,
y= Lecture_Session.
ii) ∀ y ∃ x P2(x,y) = true. x=Admin,
y= Test.
iii) ∀ y ∀ x P2(x,y) = true. x=Teacher ∪ Student, y= Viva.

7) Schema:
Entity: Asynchronous mode.
Class: Asynchronous mode ∈ Mode of operation.
Attributes: ∀ time ∀ Asynchronous event,
P1(time,Asynchronous event) → True.
P1(x,y): y can occur at x.
Relation: P2(x): Registered_Agent can works on x
∀ x P2(x) = true x: Asynchronous teaching mode.

IV. ARCHITECHTURE DESIGN
A systematic and appropriate design has to be established.
Requirements of the various learners must be catered to.
Learners have to be able to control the level and process of
their lessons. With external users included in the system,
external events and management should also be included.

V. MODEL DESIGN
A Login Model
Node:
Interface ∪ Matching ∪ Granted ∪ Deny ∪ DB.

Depends On:
Matching → Interface × DB
Approval → Matching
Mode of Operation:
Asynchronous
Operated On:
Interface ∪ Matching ∪ Granted ∪ Deny
DB: LoginId ∪ Passwd.

Fig 4: Login Model

B Registration Model
Node: Interface ∪ Checking ∪ Deny ∪ Granted ∪ Updation
Depends On:
Checking → Interface × DB
Approval → Checking
Granted → Updation
Mode of Operation:
Asynchronous
Operates on:
Interface ∪ Checking ∪ Deny ∪ Granted ∪ Updation

C Admin Model
Node:
(Account ∪ Usage_Record ∪ (Courses_Offered ∪ Faculty_Directory ∪ Exams)) ∪ (Viva ∪ Exam ∪ Lecture ∪ Schedule ∪ Course_Structure) ∪ (Conference ∪ Seminar) ∪ (Registered_Agent ∪ External_Agent)
Depends On:
Lecture_Session → Schedule × Course_Structure × Time
Viva → Schedule × Course_Structure × Time
Test → Schedule × Course_Structure × Time
External Event → Schedule × Agent × Time
Mode of Operation:
Synchronous ∪ Asynchronous
Operates On:
Synchronous → Lecture_Session ∪ Viva ∪ Test
Asynchronous → OnDemand_Download

E. Student Model
Node:
OnDemand_Download ∪ Assignment_Submission ∪ Test ∪ Viva ∪ Lecture_Session ∪ Scheduling ∪ Selecting_Course ∪ Student_WS ∪ Teacher_WS ∪ Audio ∪ Video ∪ Class_Notes
Depends On:
Assignment_Submission → Schedule × Time
OnDemand_Download → Schedule × Time
Organisational_Event → Schedule × Time
Mode of Operation:
Asynchronous ∪ Synchronous
Operates On:
Synchronous → Lecture_Session ∪ Viva_Session ∪ Exam_Session
Asynchronous → OnDemand_Download ∪ Assignment_Upload

Fig 5: Login Model

Fig 6: Admin Model

5.4 Teacher Model
Node:
(Admin_Duties ∪ OnDemand_Dialogue ∪ (Lectures ∪ Viva ∪ Test) ∪ (Student_Selection ∪ Teacher_Selection ∪ Selecting_Course ∪ Scheduling ∪ Course_Structure) ∪ (Seminar ∪ Conference) ∪ (Student_WS ∪ Teacher_WS ∪ Audio ∪ Video ∪ Class_Notes).
Depends On:
Lecture_Session → Schedule × Course_Structure × Time
Viva → Schedule × Course_Structure × Time
Test → Schedule × Course_Structure × Time
External_Event → Schedule × Agent × Time
Mode of Operation:
Synchronous ∪ Asynchronous
Operates On:
Synchronous → Lecture_Session ∪ Viva ∪ Test
Asynchronous → OnDemand_Download

Fig 7: Teacher Model
Asynchronous → OnDemand_Download ∪ Assignment_Submission

F Lecture Model
Node:
Student_WS ∪ Teacher_WS ∪ Audio ∪ Video ∪ ClassNotes ∪ Audio Video ∪ Student ∪ Teacher
Depends On:
Student_WS(READ) → Student × Teacher × Time
Student_WS(WRITE) → Student × Teacher × Time
Teacher_WS(READ) → Student × Teacher × Time
Teacher_WS(WRITE) → Student × Teacher × Time
Teaching_Aids.Audio → Student × Teacher × Time
Teaching_Aids.Video → Student × Teacher × Time
ClassNotes(READ) → Student × Teacher × Time
Audio → Student × Teacher × Time
Video → Student × Teacher × Time
Operation Mode:
Synchronous
Operates on:
Synchronous → Student_WS ∪ Teacher_WS ∪ Audio ∪ Video

G Viva Model
Node:
C_WS ∪ Student ∪ Teacher ∪ DB ∪ Audio ∪ Video
Depends on:
C_WS → Student × Teacher × Time
Audio → Student × Teacher × Time
Video → Student × Teacher × Time
Operation Mode:
Synchronous
Operates on:
Synchronous → C_WS ∪ Audio ∪ Video
DB:
Student_Directory ∪ Schedule ∪ Course_Structure

H. Test Model
Node:
Student_WS ∪ Teacher_WS ∪ Student ∪ Admin ∪ DB
Depends on:
Student_WS(Read + Write) → Admin × Student × Time
Teacher_WS(R) → Admin × Student × Time
Teacher_WS(W) → Admin × Student × Time
Operation Mode: Synchronous
Operates on:
Synchronous → Student_WS ∪ Teacher_WS
DB:
Question_Paper ∪ List_of_Students ∪ Schedule

I. External Event Model
Node:
Time ∪ Online_Submission ∪ Forms ∪ Notification ∪ Courses
Depends on:
Online_Submission → Time
Mode of Operation:
Synchronous ∪ Asynchronous
Operates on:
Synchronous → Online_Submission
Asynchronous → Forms Notification Courses

VI. CONCLUSION
The authors implemented an ontology based teleteaching system for both synchronous and asynchronous types of interaction between teacher and student like the traditional classroom. Through the teleteaching model we developed, one can create virtual class room for the student...
from youngsters to doctoral candidates, who may be located any parts of the globe.

VII. REFERENCES