<table>
<thead>
<tr>
<th>Title</th>
<th>学習管理システムの分散化に関する研究</th>
</tr>
</thead>
<tbody>
<tr>
<td>Author(s)</td>
<td>トンチャイ ケウキリア</td>
</tr>
<tr>
<td>Editor(s)</td>
<td></td>
</tr>
<tr>
<td>Citation</td>
<td></td>
</tr>
<tr>
<td>Issue Date</td>
<td>2014-02</td>
</tr>
<tr>
<td>URL</td>
<td><a href="http://hdl.handle.net/10466/14144">http://hdl.handle.net/10466/14144</a></td>
</tr>
<tr>
<td>Rights</td>
<td></td>
</tr>
</tbody>
</table>
Study on Distributed e-Learning Management Systems

Thongchai Kaewkiriya

February 2014

Doctoral Thesis at Osaka Prefecture University
Contents

1. Introduction ................................................................. 1

2. Sharing Learning Content in e-Learning Systems .................. 8
   2.1 Introduction .............................................................. 8
   2.2 e-Learning Management Systems .................................. 9
   2.3 Options for Sharing Content among e-Learning Systems .... 11
   2.4 Discussion of Options and Evaluation of DLMS ............... 15
   2.5 Conclusion .............................................................. 28

3. Transparent Digital Content Sharing by Science Teachers ........ 30
   3.1 Introduction .............................................................. 30
   3.2 Digital Content Sharing by Science Teachers ................. 32
   3.3 Evolutionary Process for Content Sharing Based on CMMI .... 34
   3.4 Scenario for Digital Content Sharing by Science Teachers .. 40
   3.5 Conclusion .............................................................. 42

4. Implementation for Distributed e-Learning Management Systems 43
   4.1 Introduction .............................................................. 43
   4.2 Communication between e-Learning Systems .................... 44
   4.3 Conceptual Design for DLMS ......................................... 50
   4.4 Design and Evaluation of DLMS ...................................... 56
   4.5 Conclusion .............................................................. 59

5. Possibility to DLMS for Learning Plan Support ..................... 60
   5.1 Introduction .............................................................. 60
   5.2 Capability Trajectory Mining for Learning Log ................. 62
   5.3 Basic idea for Proposal ................................................ 68
Chapter 1  
Introduction

Internet technologies are currently being increasingly employed in the field of education such as e-Learning and distance learning systems [S02], [N04]. e-Learning systems generally have three components of [T05] 1) a user interface that is customized for target users in an organization (students, teachers, and administrators), 2) a general software component, called a common learning engine for learning and teaching, which is independent of the user interface and learning content, and 3) a database for domain oriented learning content and learning logs. Of these, the common learning engine has reduced the load in developing individualized e-Learning systems for specific schools or organizations.

Although the number of organizations in Thailand that implemented e-Learning systems was around 40,000 in 2009, it exceeded 60,000 in 2012, and it is still increasing [TM13]. Figure 1.1 is a graph of the number of companies in Thailand.

![Figure 1.1 Total number of companies in Thailand](image)
Moreover, the volume of learning content has been increased every year. Thailand has an organization, called TCU (Thailand Cyber University) [TU12]. It is responsible for managing distance learning via e-Learning. The objective of TCU is: 1) To assist all higher educational institutes to deliver distance learning via the Internet. 2) To ensure that all online courses are of a high quality and meet government standards. 3) To promote the sharing of teaching resources and human resources.

![Image of Traditional learning management system][LM10]

**Figure 1.2 Traditional learning management system [LM10]**

Students and the teacher can only apply learning content from their own system in Figure 1.2, which is unable to apply learning content from other systems. However, the e-Learning systems are able to share learning content when necessary. Organizations or schools generally use e-Learning systems to enable employees or students to learn, and teach and train employees or students. The composition of e-Learning has problems in Learning Management Systems (LMSs) because of centralized systems [OPBNN08], [HT07], [IBM11]. The main limitation is
Chapter 1. Introduction

sharing learning content with organizations or other systems. LMSs take various forms such as having their own development programs [KM11], [PO10], [CU10]. LMSs use free software (open source) to assist in developments such as Moodle [MO10], [A10], [LO11]. However, LMSs still have problems imposed by the limits of learning content. Student can only use learning content in the local LMS of their own organization.

Therefore, each e-Learning system has its own learning content. Teachers should prepare learning content by themselves for their students with their limited resources even if excellent content may exist in other e-Learning systems. Such tasks in preparing learning content is time-consuming and costly [TN13], [SU12], [TU12]. It is also well known that some teachers cannot develop their own learning content because they have deficient IT knowledge. There is no chance, on the other hand, for students in e-Learning systems to obtain learning content from other e-learning systems even if it is openly disclosed, which is wasteful. The more learning content students use, the smarter they become.

Sharing of learning content by schools and organizations has become more significant and necessary under such circumstances. This should be transparent among e-Learning systems where transparency means that users do not need to be conscious of the real location of learning content. Users do not generally need to share learning content in the first stage. Once they cannot help using e-Learning systems, the requirement to share learning content becomes clear. Learning logs that can be shared among schools and organizations should suggest future learning directions in the final stage. There should be an account on stepwise growth for sharing learning content.

The main purpose of this research is to devise a new mechanism by combining existing and prospective LMSs on the Internet and to share their learning content among users in a richer learning environment. This research proposes a new concept, called a Distributed Learning Management System (DLMS). This thesis clarifies the reasons DLMS is required. Then, a
practical example for the proposed DLMS is presented. The thesis also explains how DLMS was designed and how the prototype was evaluated by comparing REST base and SOAP base implementation. In addition, it proposes the future possibility of DLMS based growth of capabilities in trajectory mining.

A two-stage system was developed in this research:

1. A DLMS for sharing learning content and
2. A planning support function to make recommendations to students.

We focused on the sharing of learning content in an e-Learning system for DLMS. The main challenge in this task was how users could search and apply the learning content from other organizations or schools. The process for development in this research was divided into four steps. A concept for DLMS [KU11] and [KUT11] is proposed in the first step (why DLMS is required). The applications for DLMS are presented in the second step (how to use DLMS). The transparent sharing of digital content for science teachers [KST13b] is also proposed. The data flow design for DLMS [KT12a] is presented in the third step (how to implement DLMS). The prototype for DLMS work that was developed for the experiment to evaluate the presentation of [KT12b] and [KST13a] is presented in the fourth step (how to implement DLMS). Some case studies are presented to test the possibility of gathering advice for students in the DLMS system to establish the planning support function that will provide recommendations to students. The growth of capabilities for students in trajectory mining [KSIANT13] is also presented. Further, support for planning in the growth of capabilities for trajectory mining is presented for students’ suggestions. [KST13c]. The structure of this thesis is divided into six chapters. The structure of thesis is shown in Figure 1.3.
How this research paper is organized is explained in what follows.

Chapter 2 provides four alternatives for sharing learning content in e-Learning systems (why DLMS is required): (1) a method of replicating shared content [AT89], (2) a method of centralizing shared content [BHG87], [OPBN08], [HT07], [IBM11], (3) a method of remotely logging in to shared content [RE13], [KM11], [PO10], [CU10] (multi-login), and (4) a method of sharing distributed content [KT12b], [KST13a], [JO10], [KU11], [KUT11]. This chapter discusses the potential problems with these alternatives with simple illustrations that present these system structures and data flows. This chapter also clarifies the features embedded in the alternatives by comparing the viewpoints of students, teachers, and administrators. Comparison of alternatives is provided as both strengths and weaknesses. This chapter evaluates the performance of DLMS and LMS for ten students and seven LMS systems on the 250 Mbps Internet. The proposed distributed approach was found to be preferable in the final stage as a result of the comparison.
Chapter 3 focuses on a DLMS application (how to use DLMS). The main purpose of the application was to support science teachers to prepare for their classes. The teachers in this application should be regarded as students within the context of the e-Learning system. Then, the chapter introduces the concepts and principles underlying shared digital content based on a working DLMS. The proposed framework is divided into three parts: The first part describes the background for digital content sharing by science teachers. The second part explains the designs of the evolutionary process for contents sharing based on Capability Maturity Model Integration (CMMI), which includes five stages. The final part illustrates a scenario on how the proposed framework works. Content constructors such as science teachers are expected to save a great deal of time in preparing digital content that will be presented to their students with this framework.

Chapter 4 explains the designs for the data flows for the Distributed e-Learning Management System based on a standard protocol (how to implement DLMS). First, it reviews the features of learning content that are different from those of simple HTML pages. Their sizes are generally too large to use the Simple Object Access Protocol (SOAP) for communication between e-Learning systems. The SOAP and Representational State Transfer (REST) ful Web services were compared, and why the latter is superior to the former for DLMS is clarified. A data flow diagram and class diagram are specified that follow the conceptual design. This chapter explains an evaluation of the performance of RESTful base implementation and SOAP base implementation.

Chapter 5 suggests future possibilities by introducing capabilities for trajectory mining [KSIANT13]. It discusses ways to support improved individualized learning processes based on trajectory mining. Trajectory mining is a type of mining for extracting dependency knowledge from time-series data. The system is expected to provide students with better learning support through the use of this technology. This chapter consists of four parts. The first part introduces
Chapter 1. Introduction

data mining and related work. The second part reviews what capabilities trajectory mining has for learning logs. The third part introduces the basic idea behind the proposal. This part reviews the Spiral Enhancement capability support system (SPICE) [NT10] and CMMI used for analysis. The third part explains the design of a questionnaire for a sample group (TNI student capabilities). The final part proposes the possibility of learning log under DLMS and an algorithm for message generation. In addition, this part proposes support for planning in the growth of capabilities for trajectory mining for students’ recommendations [KST13c].

Chapter 6 finally concludes this thesis by summarizing the previous chapters.
Chapter 2
Sharing Learning Content in e-Learning Systems

2.1 Introduction

People can not only currently learn and teach in classrooms but also through any methods and forms through electronic media such as CD-ROMs, the Internet, intranets, extranets, television, or satellites. These media-based learning styles have been imported into the Thai market, such as computer-assisted instruction with CD-ROMs, instruction on the Web (Web-based Learning), and online learning. Of these, e-Learning is becoming crucial, which is similar to distance learning [G05] [CLL10], because e-Learning provides education for people in distant areas using electronic media [J09].

Individual organizations have generally developed learning content for their own purposes. Therefore, the rapid growth of organizations has increasingly required more learning content. However, the volume of learning content has been limited in specific organizations and has occasionally not been sufficient for students as individual organizations have different learning content. Moreover, it is expensive to develop learning content. Therefore, learning content needs to be shared among organizations to enable voluminous learning content to be quickly and economically prepared.

Past research on sharing learning content has been as follows; [CLC05] presented a method of sharing Web-based learning objects by using the Network News Transfer Protocol (NNTP) architecture. However, the working process for the system was complicated and incurred high development costs. Moreover, the research on [KFK06] sharing learning resources for an online
Chapter 2. Sharing Learning Content in e-Learning Systems

engineering presentations course emphasized the effectiveness of Web pages. However, they did not discuss any methods of sharing learning content across other systems.

Although another past research on sharing learning content proposed duplicating data between systems [G79], it also had three limitations: 1) a large volume was required, 2) it was difficult to maintain consistency, and 3) human support was essential. Further, [BHG87] presented a method of sharing learning content based on a centralized database but it also suffered from two limitations: 1) problems at databases (DBs) stopped systems and 2) high-end power was required for center systems.

This chapter offers four alternatives for content sharing in e-Learning management systems (e-LMSs). First, it explains the reasons for sharing learning content. Second, it introduces an e-LMS. Third, it compares methods of sharing learning content. Finally, it proposes a new concept of DLMS and presents an example process of how DLMS works.

2.2 e-Learning Management Systems

First, let us review the structure and components of e-LMSs. The LMS [LM10] in Figure 2.1 is a software application that has been developed in many forms. Some systems only stored and managed learning content such as Web portals that collected Web links for students or video libraries to help them do searches. Other systems provided tests to assess student levels. More sophisticated systems contained learning records (learning logs).
Chapter 2. Sharing Learning Content in e-Learning Systems

Figure 2.1 Traditional learning management system [LM10]

The process of traditional LMSs [E09] is divided into two main parts as seen in Figure 2.1: 1) a user interface that is customized for organizations and 2) an engine, which is common (general) software that supports learning and teaching.

There are three roles for users of LMS, i.e., as administrators, teachers, and students. Administrators are responsible for system management including updates and backups. Teachers are responsible for creating learning content for students. Students are responsible for learning by completing assignments and taking tests.

Learning content is a kind of medium for the learning and teaching material. There is an example in Figure 2.2. The characteristics of learning content in organizations are as follows. Even if there is standard learning content, individual organizations often customize it based on their experience. Individual organizations occasionally develop original learning content according to their own needs. Software companies may develop their own learning content on information technology (IT). Logistics companies may develop their own learning content on
supply chain management. Then, both customized content and original content may be available at other organizations.

![Image](image.png)

Figure 2.2 Example of learning content in organization

### 2.3 Options for Sharing Content among e-Learning Systems

#### 2.3.1 Replication

One simple method of information sharing has been replication [AT89]. This is used to copy data from one system to others to share information, as shown in Figure 2.3. There should be a responsible administrator who orders data to be copied from their origin to the destination when data are replicated. The main strengths of this method are that it is very reliable and easy to manage. Its three main limitations are that:

1) A large volume is required because the same data are stored redundantly,

2) It is difficult to maintain consistency because there is a time delay in copying, and

3) A human cost is incurred in monitoring replication.
2.3.2 Centralization

Figure 2.3 Digital content sharing based on replication

Figure 2.4 Digital content sharing based on centralization
Chapter 2. Sharing Learning Content in e-Learning Systems

The second alternative for sharing learning content is centralization, as seen in Figure 2.4 [BHG87]. This method requires all systems to be connected with a centralized DB. The main advantage of this method of sharing data is the ease of maintaining consistency because users can access the same data from the centralized DB. In fact, there is no time delay in updating data because all data are at the center. However, the two main limitations of the centralized method are:

1) When the centralized DB is down or unable to work, it causes problems for all users and

2) The centralized system becomes expensive because it has to work effectively with other systems.

2.3.3 Remote Login

Figure 2.5 Digital content sharing based on remote login
Chapter 2. Sharing Learning Content in e-Learning Systems

The third alternative is the remote login method shown in Figure 2.5 [RE13]. The process of remote login is mostly authorized by an administrator. The main advantage of this method is to save costs in development and infrastructure. The three main limitations of remote logins are:

1) Operation costs because e-Learning systems may have different user interfaces,

2) Authorization costs because systems may have their own user management, and

3) Combining learning content is difficult in one LMS with that in another. Operation time is also involved.

2.3.4 New Concept of DLMS

This thesis proposes a new method of sharing learning content called a Distributed e-Learning Management System (DLMS). The proposed DLMS is responsible for learning management via the Internet with the ability of gathering data on learning content to deal with learning content in various subjects for teachers, students, and administrators. The target image of DLMS is outlined in Figure 2.6.

Figure 2.6 Digital content sharing based on DLMS
LMSs in this method are comprised of administrators, teachers, and students. Users can transparently collect learning content not only from their LMS but also from other LMSs. Students thus have the chance of selecting the best learning content from many LMSs. Administrators and teachers are responsible for their own LMSs, but they also have the means to connect with other LMSs. Researchers have presented some parts [KU11], [KUT11] by explaining the principles of the preliminary process in DLMS.

2.4 Discussion of Options and Evaluation of DLMS

2.4.1 Qualitative Evaluation

Tables 2.1, 2.2 and 2.3 clarify the position of the proposed DLMS.

The four options for sharing learning content in the three tables are discussed in what follows.

(1) Storage Costs

The replication option requires $N$ times storage cost because all LMSs should retain all learning content if there are $N$ LMSs. The centralization option may incur twice the storage cost if each LMS stores its learning content in its server and the centralized server. However, the other two options do not need extra storage. As $N$ has recently become large, a great deal of attention should be focused on this cost.

(2) Administration Costs

The replication option incurs administration costs for copying data from LMSs while the centralized option incurs costs for copying data from local to central servers. The former involves $N\times C_2$ combinations and the latter involves just $N$ combinations. The more LMSs there are, the higher the administration costs. The other two options, on the other hand, do not incur extra administration costs. As was previously pointed out, as $N$ increases, attention should also be focused on these costs.
(3) User Operation Costs

Users of the remote login option should know which LMS they intend to access. If there are \( N \) LMSs, they may know the names of the \( N \) LMSs, their accounts with passwords, and operation methods. In this sense, the costs of user operation with this option are very high. However, the other three options incur almost the same user operation costs because users do not need to be conscious of LMSs. As the number of users increases, attention should also be focused on these costs.

<table>
<thead>
<tr>
<th>No</th>
<th>Methods/Details</th>
<th>Students</th>
<th>Teachers</th>
<th>Administrators</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Replication</td>
<td>- Slow</td>
<td>- Slow</td>
<td>- Huge work load for administrators to copy learning content</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- If there is much content, it could take a lot of time</td>
<td>- Difficult to update content</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Centralized</td>
<td>- If the system has much content, it may be slow to find specific content</td>
<td>- Easy to manage courses and updates</td>
<td>- Huge work load for administrators to copy learning content</td>
</tr>
<tr>
<td>3</td>
<td>Remote login</td>
<td>- Easy to find content</td>
<td>- Easy to find content</td>
<td>- Huge work load for administrators to manage remote logins and copy learning content</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Involves operation time</td>
<td>- Involves operation time</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Delayed content from administrator.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Distributed</td>
<td>- Easy to find content</td>
<td>- Easy to manage content</td>
<td>- Easy to manage system</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- Easy to find content</td>
<td></td>
</tr>
</tbody>
</table>

Table 2.1 Comparison from viewpoints of users
### Table 2.2 Comparison of four options (Qualitative Evaluation)

<table>
<thead>
<tr>
<th>No</th>
<th>Methods/Details</th>
<th>Strengths</th>
<th>Weaknesses</th>
</tr>
</thead>
</table>
| 1  | Replication     | - Reliable data  
|     |                 | - Simple method   | - Large volumes are required  
|     |                 |                      | - Difficult to update (inconsistency)  
|     |                 |                      | - Human involvement for operation  
|     |                 |                      | - Much content that slows the system |
| 2  | Centralized     | - Efficient control  
|     |                 | - Easy to maintain consistency | - High risk (Only one DB)  
|     |                 |                      | - Difficult to customize  
|     |                 |                      | - Extra cost of preparing central system  
|     |                 |                      | - System is excessively burdened  
|     |                 |                      | - Much content slows the system |
| 3  | Remote login    | - Saves costs  
|     |                 | - Easy to start (no special integration tasks) | - Incurs operational costs (different user interface)  
|     |                 |                      | - Complex authorization  
|     |                 |                      | - Involves operation time |
| 4  | Distributed     | - Saves costs  
|     |                 | - Easy to use (transparent) | - Base on speed of network  
|     |                 |                      | - Difficult to implement |
### Table 2.3 Comparison of four options (Quantitative Evaluation)

<table>
<thead>
<tr>
<th>No</th>
<th>Details</th>
<th>Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Reliable data</td>
<td>1. Replication -&gt; High Reliable data</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2. Centralized -&gt; Medium Reliable data</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3. Remote Login -&gt; Medium Reliable data</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4. Distributed -&gt; Medium Reliable data</td>
</tr>
<tr>
<td>2</td>
<td>Operation Costs</td>
<td>1. Replication -&gt; High Operation costs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2. Centralized -&gt; High Operation costs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3. Remote Login -&gt; High Operation costs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4. Distributed -&gt; Low Operation costs</td>
</tr>
<tr>
<td>3</td>
<td>Data Consistency</td>
<td>1. Replication -&gt; Low Data Consistency</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2. Centralized -&gt; High Data Consistency</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3. Remote Login -&gt; Medium Data Consistency</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4. Distributed -&gt; High Data Consistency</td>
</tr>
<tr>
<td>4</td>
<td>User friendly</td>
<td>1. Replication -&gt; Very easy to use</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2. Centralized -&gt; Easy to use</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3. Remote Login -&gt; Difficult to use</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4. Distributed -&gt; Very Easy to use</td>
</tr>
</tbody>
</table>
(4) **Response Time**

Suppose that users like to search learning content over LMSs. The response time for the centralized option depends on the power of the server. That of DLMS greatly depends on the power of the network while that of the replication option depends on scale of learning contents and efficiency of network. However, the response time of the remote login option requires extra time for users and it is not practical if $N$ increases.

(5) **Implementation Costs**

There are no extra implementation costs for the remote login option while the other options incur implementation costs. Although the extra implementation cost of the replication option is small, the two other options incur greater costs. A method that uses standard software and a standard protocol should be developed to reduce the implementation costs of these options.

(6) **Consistency**

If there are no other versions of learning content in LMSs, the original can be regarded as consistent. If there is one version in one LMS and another version in another LMS, they can be regarded as inconsistent. There is inconsistency in the replication option because new content in a destination LMS is not always copied as soon as the learning content is updated in the original LMS. Although there is a room for inconsistency in the centralized option, consistency is maintained in the remote login and DLMS options.

DLMS is the most preferable, based on the above discussion, if there are means of saving implementation costs. We should evaluate available standard protocols to reduce implementation costs. It is known that one of the distributed technologies is SOAP Web services [GHM07] based on the service-oriented architecture (SOA) [HS05]. Really Simple Syndication (RSS) [BC03] is another means of sharing learning content. Details on their implementation will be discussed in Chapter 3.
2.4.2 Quantitative Evaluation

(1) Analysis of average response, total, and operation times

This subsection considers user operation time, computing time as response time, and total turn-around time as metrics to compare the performance of DLMS with that of conventional LMS. Related terms are given in Figure 2.7 for the latter and Figure 2.8 for the former.

Figure 2.7 Repetitious process to evaluate conventional LMS

The meanings of the abbreviations in the figure are listed below.

\[ T_t = \text{Total turn-around time} \]

\[ T_o = \text{Time for operation (starting by typing username and password to log in, typing keyword to search, and clicking submit button).} \]

\[ T_r = \text{Response time for computation (receiving searched result displayed on monitor.)} \]
Chapter 2. Sharing Learning Content in e-Learning Systems

\[ T_t = T_o + T_r \quad (1) \]

\[ T_t = T_1 + T_2 + T_3 + \ldots + T_n \quad (2) \]

\[ T_t = (T_{o1} + T_{r1}) + (T_{o2} + T_{r2}) + (T_{o3} + T_{r3}) + \ldots + (T_{on} + T_{rn}) \quad (3) \]

Figure 2.8 One process to evaluate DLMS

There were ten students of TNI in the evaluation who were asked to use both systems under conditions where there were seven servers as LMSs where the Internet speed was 250 Mbps. Let us consider their average response times, operation times, and total time separately.

The results for the average response times are summarized in Table 2.4 and Figure 2.9. There is a linear relation between the number of servers and response times. If there are fewer than five servers under these conditions, the repetitive use of LMS outperforms single DLMS. Therefore, DLMS does not always perform better for the average turn-around time.
Table 2.4 Comparison of response times (Tr)

<table>
<thead>
<tr>
<th>Systems</th>
<th>LMS₁/Sec</th>
<th>LMS₂/Sec</th>
<th>LMS₃/Sec</th>
<th>LMS₄/Sec</th>
<th>LMS₅/Sec</th>
<th>LMS₆/Sec</th>
<th>LMS₇/Sec</th>
</tr>
</thead>
<tbody>
<tr>
<td>LMS</td>
<td>0.461</td>
<td>1.042</td>
<td>1.598</td>
<td>2.057</td>
<td>2.657</td>
<td>3.082</td>
<td>3.568</td>
</tr>
<tr>
<td>DLMS</td>
<td>0.508</td>
<td>1.295</td>
<td>1.767</td>
<td>2.315</td>
<td>2.641</td>
<td>2.901</td>
<td>3.056</td>
</tr>
</tbody>
</table>

Figure 2.9 Comparison of response times

Next, let us check the difference in the average operation times between repetitive LMS and single DLMS. The results from the experiment are given in Table 2.5 and Figure 2.10.
Table 2.5 Comparison of operation times ($T_o$)

<table>
<thead>
<tr>
<th>Systems</th>
<th>LMS$_1$/sec</th>
<th>LMS$_2$/sec</th>
<th>LMS$_3$/sec</th>
<th>LMS$_4$/sec</th>
<th>LMS$_5$/sec</th>
<th>LMS$_6$/sec</th>
<th>LMS$_7$/sec</th>
</tr>
</thead>
<tbody>
<tr>
<td>LMS</td>
<td>28.735</td>
<td>58.29</td>
<td>88.096</td>
<td>118.091</td>
<td>147.015</td>
<td>177.161</td>
<td>206.309</td>
</tr>
</tbody>
</table>

![Time/seconds vs. Operation times](image)

**Figure 2.10 Comparison of operation times**

The average operation time for DLMS is constant while that for repetitive LMS increases as the number of servers increases, as was expected. The graph indicates that users take 29–30 sec for one LMS.
Table 2.6 and Figure 2.11 provide the average turn-around times \( (T_t = T_r + T_o) \), which suggest that DLMS is helpful to reduce the average turn-around time.

Table 2.6 Comparison of averages for total turn-around times \( (T_t) \)

<table>
<thead>
<tr>
<th>Systems</th>
<th>LMS_1/ sec</th>
<th>LMS_2/ sec</th>
<th>LMS_3/ sec</th>
<th>LMS_4/ sec</th>
<th>LMS_5/ sec</th>
<th>LMS_6/ sec</th>
<th>LMS_7/ sec</th>
</tr>
</thead>
<tbody>
<tr>
<td>LMS</td>
<td>29.196</td>
<td>59.332</td>
<td>89.694</td>
<td>120.148</td>
<td>149.672</td>
<td>180.243</td>
<td>209.877</td>
</tr>
<tr>
<td>DLMS</td>
<td>29.993</td>
<td>30.023</td>
<td>31.154</td>
<td>31.709</td>
<td>32.046</td>
<td>32.922</td>
<td>32.494</td>
</tr>
</tbody>
</table>

Figure 2.11 Comparison of total turn-around times

(2) Analysis of variance

The analysis of variance for validating DLMS is discussed in this section. The parameter settings for statistical analysis are given below.
Table 2.7 Analysis of variance for response times

<table>
<thead>
<tr>
<th>Systems</th>
<th>LMS$_1$/sec</th>
<th>LMS$_2$/sec</th>
<th>LMS$_3$/sec</th>
<th>LMS$_4$/sec</th>
<th>LMS$_5$/sec</th>
<th>LMS$_6$/sec</th>
<th>LMS$_7$/sec</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>LMS</td>
<td>0.461</td>
<td>1.042</td>
<td>1.598</td>
<td>2.057</td>
<td>2.657</td>
<td>3.082</td>
<td>3.568</td>
<td>2.066</td>
</tr>
<tr>
<td>DLMS</td>
<td>0.508</td>
<td>1.295</td>
<td>1.767</td>
<td>2.315</td>
<td>2.641</td>
<td>2.901</td>
<td>3.056</td>
<td>2.069</td>
</tr>
</tbody>
</table>

Table 2.7 summarizes the analysis of variance for response times ($Tr$). The significance level is defined as 0.05. Whether the average response times differed for LMS and DLMS needed to be tested.

Assumption.

$H_0 : \mu_1 = \mu_2$ (If $\mu_1$ is the average response time of LMS, $\mu_2$ is the average response time of DLMS)

$H_1 : \mu_1 \neq \mu_2$ (If $\mu_1$ and $\mu_2$ are different)

Here, a t-test was used for analysis. The results obtained from testing are:

$t$ (Calculate) = -0.00468 and

$t$ (Crisis) = -2.172.

As $t_{\text{Calculate}} > t_{\text{Crisis}}$, $H_1$ is rejected and $H_0$ is accepted. This means that the average response times for LMS and DLMS do not differ.

The results from the analysis of variance for the response times for LMS and DLMS did not differ. However, the possibility of the response time for DLMS was decreasing. Then, the possibility of the response time for LMS was increasing (See Table 2.7 and Figure 2.9). It was concluded that DLMS had a similar response time to LMS.
Table 2.8 Analysis of variance for operation times

<table>
<thead>
<tr>
<th>System</th>
<th>LMS₁/sec</th>
<th>LMS₂/sec</th>
<th>LMS₃/sec</th>
<th>LMS₄/sec</th>
<th>LMS₅/sec</th>
<th>LMS₆/sec</th>
<th>LMS₇/sec</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>LMS</td>
<td>28.735</td>
<td>58.290</td>
<td>88.096</td>
<td>118.091</td>
<td>147.015</td>
<td>177.161</td>
<td>206.309</td>
<td>117.671</td>
</tr>
</tbody>
</table>

Table 2.8 summarizes the analysis of variance for operation times (To). The significance level was defined as 0.05. Whether the average operation times differed for LMS and DLMS needed to be tested.

Assumption.

\[ H_0 : \mu_1 = \mu_2 \text{ (If } \mu_1 \text{ is the average operation time for LMS, } \mu_2 \text{ is the average operation time for DLMS)} \]

\[ H_1 : \mu_1 \neq \mu_2 \text{ (If } \mu_1 \text{ and } \mu_2 \text{ are different)} \]

A t-test was used for analysis. The results from testing are:

\[ t \text{ (Calculate)} = 3.648 \text{ and } t \text{ (Crisis)} = 2.179. \]

As \( t_{\text{Calculate}} > t_{\text{Crisis}} \), \( H_0 \) is rejected and \( H_1 \) is accepted. This means that the average operation times for LMS and DLMS differ.

The results from the analysis of variance for the operation times of LMS and DLMS differed. However, the possibility of the operation time for LMS was increasing as a linear function. Then, the possibility of the operation time for DLMS was stable (See Table 2.8 and
Figure 2.10). It was concluded that the operation time for DLMS was less than the operation time for LMS. DLMS used less operation time than LMS.

Table 2.9 Analysis of variance for total turn-around time

<table>
<thead>
<tr>
<th>Systems</th>
<th>LMS₁/sec</th>
<th>LMS₂/sec</th>
<th>LMS₃/sec</th>
<th>LMS₄/sec</th>
<th>LMS₅/sec</th>
<th>LMS₆/sec</th>
<th>LMS₇/sec</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>LMS</td>
<td>29.196</td>
<td>59.332</td>
<td>89.694</td>
<td>120.148</td>
<td>149.672</td>
<td>180.243</td>
<td>209.877</td>
<td>119.737</td>
</tr>
<tr>
<td>DLMS</td>
<td>29.993</td>
<td>30.023</td>
<td>31.154</td>
<td>31.709</td>
<td>32.046</td>
<td>32.922</td>
<td>32.494</td>
<td>31.477</td>
</tr>
</tbody>
</table>

Table 2.9 summarizes the analysis of variance for total turn-around times (Tt). The significance level was defined as 0.05. Whether the average total turn-around time differed for LMS and DLMS needed to be tested.

Assumption.

\[ H₀ : μ₁ = μ₂ \] (If \( μ₁ \) is the average total turn-around time of LMS, \( μ₂ \) is the average total turn-around time of DLMS)

\[ H₁ : μ₁ ≠ μ₂ \] (If \( μ₁ \) and \( μ₂ \) are different)

A t-test was used for analysis. The results from testing are:

\[ t \text{ (Calculate)} = 3.584 \] and

\[ t \text{ (Crisis)} = 2.179. \]

As \( t \text{Calculate} > t \text{Crisis} \), \( H₀ \) is rejected and \( H₁ \) is accepted. This means that the average total turn-around time for LMS and DLMS differs.
The results from the analysis of variance for the total time for LMS and DLMS differed. However, the possibility of a total turn-around time for LMS was increasing as a linear function. Then, the possibility of a total turn-around time for DLMS was stable (See Table 2.9 and Figure 2.11). It could be concluded that the total turn-around time for DLMS was less than the total turn-around time for LMS from the results for the total time. DLMS used less total turn-around time than LMS.

2.5 Conclusion

As the number of organizations that use LMSs increases, content sharing among LMSs will becomes more important. There are four alternative options for sharing learning content. Based on a discussion from the six viewpoints of storage costs, administration costs, user operation costs, response time, implementation costs, and consistency, “the distributed” approach is the most preferable.

This chapter emphasized the importance of DLMS. First, it reviewed LMS and its framework. Second, four alternative options for sharing learning content were presented of 1) replicating shared learning content, 2) sharing centralized learning content, 3) sharing learning content from remote logins, and 4) sharing distributed learning content. Then, methods of sharing learning content were compared qualitatively by using three tables (table 2.1, 2.2 and 2.3). The reasons the distributed option was preferred were clarified as a result of the comparison.

In addition chapter 2 compared the performance of DLMS and LMS quantitatively. The performance of the proposed DLMS was evaluated for ten students and seven LMS systems with this prototype system. The results from the analysis of variance for total turn-around times for LMS and DLMS differed as a result of the evaluation. It was concluded that the total turn-around time (Total turn-around time = Response time + Operation time) of DLMS was less than
that of LMS by considering their total turn-around times. DLMS used less total turn-around time than LMS. Finally, it was concluded that DLMS became more efficient as the number of LMSs that had sharable learning content increased. The next chapter will discuss matters of how to use DLMS which is based on a case study of the transparent digital content sharing by science teacher.
Chapter 3
Transparent Digital Content Sharing by Science Teachers

3.1 Introduction

Information technology in the field of education has become more important in recent years, especially for distance learning through the Internet [G05]. The volume of digital content on the Internet has increased daily for teachers as well as for students. Varieties of digital content are scattered over schools. Each school has normally provided an LMS for teachers, students, and an administrator. One of the main problems for teachers is how to spend time creating and sharing digital content even if there is already excellent content at other schools. Thus, teachers create and share digital content only within their own school’s LMS. Therefore, how can teachers share digital content with teachers from other schools to prepare for classes more efficiently.

The answer is to provide transparent digital content sharing mechanisms. Let us review the current status on information sharing methods. There are still teachers who hesitate to use information technology. They can only exchange tacit ideas on teaching materials based on discussions. Some teachers have created digital content as teaching materials that are externalized from their mindsets using standalone personal computers (PCs). They should copy these into media such as USB memories to share such externalized knowledge as explicit knowledge. Then, there would be less chance of effectively collecting large amounts of digital content. If there is an LMS at a school that is connected to a LAN, teachers have the opportunity of reusing digital content that is externalized and disclose their ideas to others. The discussion in
Digital content sharing in previous research has been done in a social network service with maturity levels for science teachers [SWTS09]. In addition, the design of a knowledge network for Japanese science teachers has been presented [WSTAS09] to support knowledge sharing among science teachers through 1) relationships 2), reputation, and 3) personalization. Research has focused on these three factors [SWTS09], [WSTAS09].

However, how to implement proposals has been strongly dependent on social network services. It is difficult to describe scenarios in which teachers mature regarding digital content sharing. This chapter introduces a maturity model for them by referring to the concepts of the Socialization Externalization Combination and Integration (SECI) [NT95] and Capability Maturity Model Integration (CMMI) models [PWCC95], [BKS03] by taking pioneer teachers and follower teachers into account. Next, it also describes what DLMS is and how it will work for future digital content sharing. A RESTful Web service [FT02], [F00] was used to develop the prototype. This chapter also presents scenarios in which teachers mature and what DLMS should do to support this.

This chapter presents concepts and principles on digital content sharing by science teachers (how to use DLMS) to support them in preparing for classes. First, it reviews the background for digital content sharing by science teachers. Next, it describes the evolutionary process for content sharing on DLMS based on CMMI, which allows science teachers to efficiently share digital content. Then, it illustrates a scenario on how the proposed framework works. The possibility of a prototype system based on RESTful Web services is also discussed. Science teachers are expected to save time in preparing digital content by following these concepts and principles.
Chapter 3. Transparent Digital Content Sharing by Science Teachers

3.2 Digital Content Sharing by Science Teachers

The Japanese government has prepared digital content to help science teachers. Today, there are more than 45,000 users and more than 40,000 content items. There are examples of digital content in Figure 3.1 [JA13], which is stored at a central database (DB) that has a keyword query function. However, there are two main problems.

(1) While there are pioneers, there are also followers. In fact, some teachers are so proactive that they prepare digital content for classes by themselves. They may modify the prepared digital content for classes because it may sometimes be too long or redundant. This modified content cannot therefore be shared. Other teachers, on the other hand, may be novices to IT and hesitate to take the lead in using digital content.

(2) The IT environments of schools differ. Some schools may be equipped with up-to-date technology and others may not. However, it is also true that such environments will be annually upgraded once the digital content has worked well for classes. When environments are upgraded, there are extra tasks such as movement of databases and changes in usability.

Figure 3.1 Examples of digital content sharing
Term definition

*Transparent Digital Content* means the ability to access digital content which can search and access across the digital content of other e-Learning system services. Transparent in common meaning is a transparency or penetrate. This research used the Transparent Digital Content.

*Pioneer teachers* mean teachers who have responsibilities in initiatives of sharing knowledge. In addition, Pioneer teachers who have IT skills and can also be developed with digital content, as well.

*Follower teachers* mean teachers who have responsibilities in exchanging of knowledge in various types. Follower teachers will learn about digital content from existing e-Learning system which can be developed from pioneer teacher.

Next, let us consider the initial and ideal states in Figure 3.2. There is no IT at a school in the initial state. Neither pioneer teachers nor follower teachers have the chance of using digital content but they can initially exchange ideas on how to start using them. Once IT is introduced at the school, pioneer teachers start using them. Pioneers make plans for using digital content in the ideal state and give lessons in class. They review the class materials and may modify both their lessons and digital content. They actively share their ideas with others. The follower teachers start using digital content by learning the best practice from pioneer teachers. Even if teachers hesitate to express their ideas, they like to search digital content that fits their classes.

We should consider scenarios where we give science teachers the chance to transit from the initial to the ideal state.
Chapter 3. Transparent Digital Content Sharing by Science Teachers

3.3 Evolutionary Process for Content Sharing Based on CMMI

3.3.1 Maturity Upgrade Process

Let us consider the bias to establish a model for digital content sharing to further develop teachers’ skills. This subsection proposes the introduction of two biases: the CMMI [PWCC95], [BKS03] and SECI models.

CMMI is well-known in the software development process. CMMI assumes that the upgrading of maturity levels is measured by using existing processes that imply the levels of
capabilities. The original CMM includes five maturity levels of ad hoc (initial), repeatable, defined, managed, and optimized (ideal), as seen in Figure 3.3.

![Maturity levels in CMMI](image)

**Figure. 3.3 Maturity levels in CMMI**

The SECI model is well-known for knowledge creative companies [NT95]. It includes four spiral processes of knowledge socialization (S), knowledge externalization (E), knowledge combination (C), and knowledge internalization (I). The S by teachers within our context can occur even if there are no special IT tools and I by teachers can occur if there is a search mechanism. For teachers to implement E, there should at least be personal storage and preferably shared storage. There should also be a computer network for sharing knowledge for C.

Let us consider the relations between the CMMI and SECI models. All teachers in the initial stage conduct traditional classes where there are no processes at school for using digital content. Therefore, without IT, communication at schools occurs through socialization. Communication occasionally contributes to encouraging teachers to improve their classes.
Teachers in the final and ideal states are very active in sharing best practice on digital content. They not only use prepared digital content but they also modify it and share their modifications. They search appropriate content not only from their own schools but also from other schools. This means that sophisticated digital content as best practice is transparently accessed by all teachers.

There is an overview of the maturity upgrade process used in this research domain summarized in Table 3.1. Let us examine each level in detail.

**Table 3.1 Maturity upgrade process for science teachers**

<table>
<thead>
<tr>
<th>Level</th>
<th>Detail</th>
<th>Teacher (User)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Tool</td>
</tr>
<tr>
<td>1</td>
<td>No IT</td>
<td>-</td>
</tr>
<tr>
<td>2</td>
<td>Stand alone</td>
<td>PC</td>
</tr>
<tr>
<td>3</td>
<td>School Network</td>
<td>LMS</td>
</tr>
<tr>
<td>4</td>
<td>Global Network</td>
<td>DLMS+SNS</td>
</tr>
</tbody>
</table>

The meaning of symbols in the Table 3.1; cross mark means No chance, triangle means possible, circle means stationary use.

Maturity level 1 is the customary state where IT is not available in schools even if teachers have IT skills. The teachers at this level share knowledge or experience with colleagues. However, this is time consuming. Even if one finds best practice, it is not easy to share it with others at other schools as there is no space to store it for others.
Maturity level 2 is a case where Rika-net appeared, which is a well-known central DB of content for Japanese science teachers that was prepared by the government. If pioneer teachers have standalone PCs, they have a chance of searching digital content that is stored in the central DB. They learn best practice, use it in their classes, and store their experience as tacit knowledge by viewing digital content. Teachers may also modify the digital content for their classes. However, even if the modified content is available to other teachers, there is no chance of them sharing it electronically. What they can do is to communicate with follower teachers on their practice at meetings. Such socialization allows follower teachers to start using digital content.

Maturity level 3 is a state where a local network is available at schools. Teachers at this level use the conventional LMS in their schools, which allows them to share information electronically. Once one teacher finds best practice in Rika-net, he lets his colleagues know via LMS. Pioneer teachers not only share knowledge or experience with teachers but create their own digital content, and save and store it in their own school’s DB. The followers then start using digital content for their classes while each school continues to store best practice in its LMS. There is no chance of sharing content with schools unless they use special channels. The process for maturity level 3 is outlined in Figure 3.4.
Maturity level 4 is a higher state for science teachers to share digital content. There is DLMS that allows them to transparently access the DBs of other schools. Once pioneer teachers have stored their best practice on their schools’ LMSs, everybody (not only teachers at their own schools but also teachers at other schools) has a chance of accessing it without difficulty. They feel as if all content were in one database. This transparency is due to the functions of DLMS.

### 3.3.2 System Components and Data Flow for DLMS

Let us describe the system components of the distributed LMS that allows us to access maturity level 4. As can be seen from Figure 3.5, there are more than two LMSs. Each school prepares an LMS for science teachers to share knowledge based on digital content. The LMS is composed of a student’s module for students, an administration module for administrators, and a teacher’s module for teachers. This subsection focuses on the teacher module.

![Figure 3.5 System component for DLMS](image)

A teacher who is in LMS₁ not only uses digital content in the LMS of his/her school but also that in other schools. All teachers have a chance of externalizing their tacit knowledge for other teachers in this environment. Once the digital content is prepared, it can be accessed from any
Chapter 3. Transparent Digital Content Sharing by Science Teachers

LMS. Thus, prepared digital content is combined as explicit knowledge. Finally, all teachers can search digital content for their classes from other schools. If they learn something from the new digital content, this is regarded as internalization.

Figure 3.6 outlines the data flow for DLMS. A teacher who logs in to LMS will be able to study new digital content that other teachers in the system have created. Note that the ability to create digital content is based on the maturity level of individual teachers.

The data flow indicates that a teacher is able to search digital content from the database of LMS. In addition, the teacher can create digital content and modify it into a private DB of LMS. Also, he/she can disclose private digital content and make it published digital content if his/her maturity level is four (review Table 3.1).

![Figure 3.6 Data flow for DLMS](image-url)
3.4 Scenario for Digital Content Sharing by Science Teachers

Let us discuss scenarios in which information technology progressively contributes to digital content sharing. Each scenario illustrates a case.

First, suppose that teacher A would like to prepare teaching content for the subject of geometry but the DB of school A does not have geometry content. Teacher A should therefore ask the administrator of school A to copy geometry content from another school, e.g., school B, to prepare for his/her own class. Such a scenario is illustrated in Figure 3.7.

Second, suppose that teacher A would like to prepare teaching content for the subject of geometry but the DB of school A does not have it. At the same time, suppose that teacher B would like to prepare teaching content for the subject of calculus but the DB of school B does not have it. If there is a centralized DB that includes all content, both teachers A and B have a
chance of searching content from the DB as well. Scenario 2 is outlined in Figure 3.8. Rika-net can also be used for this scenario. Although Rika-net allows teachers to share global content, there is still a problem with the sharing of real-time content because the central DB should be carefully maintained.

![Figure 3.8 Scenario for Rika-net](image)

There is a third scenario. Suppose that teacher A would like to prepare teaching content for the subject of chemistry but the DB of school A does not have it. If there is a function for remote login, Teacher A has a chance of downloading learning content from school B to prepare for his/her class. Although remote login allows teachers to share learning content globally and on time, there is still a problem with usability because individual teachers should have multi-user IDs and passwords.
Let us consider a final scenario where DLMS is used. Suppose that each school has LMS and school A does not have learning content on calculus but school B has. Then, if there is a function for transparent sharing based on DLMS, teacher A also has a chance of using the content at school B as if it were at school A. In other words, teacher A uses DLMS to search course content from any schools, such as B, C, and D, to prepare for his/her own class by searching for calculus content at school C. This scenario is illustrated in Figure 3.9.

![Figure 3.9 Scenario for DLMS](image)

**3.5 Conclusion**

This chapter proposed transparent sharing of digital content and knowledge sharing by science teachers as an application of DLMS (how to use DLMS). It also described how transparent digital content sharing works in Japanese schools. Further, the maturity levels of science teachers including pioneers and followers were discussed based on CMMI. Scenarios on how IT contributes to science teachers were also illustrated. As a result, science teachers are expected to use the prototype to transparently share their digital content and knowledge.
Chapter 4
Implementation of Distributed e-Learning Management Systems

4.1 Introduction

Information and internet technologies in the field of education have recently become important [G05]. Such applications are called e-Learning systems [TS09]. The volume of learning content on the Internet has progressively expanded without limit. The varieties of knowledge about education are available on Websites [SU12], [TU12].

Numerous resources are required to develop effective learning content such as expert teachers in specific fields, capabilities to convey content or teaching techniques, and resource people who produce teaching materials that support knowledge transfer [L09], [W11]. We should be able to widely share effective practices for e-Learning systems if they are available. However, organizations are still limited in sharing learning content via the Internet [WO08], [GA02].

These problems with e-Learning systems have led to two categories of solutions. The first is to prepare processes to accumulate learning content that is spread over the Internet. The second is to implement content-aggregated Learning Management Systems (LMSs). Web service technology and links with databases [LM08] gave us hints to solve these problems. Previous research has revealed the design of a Data Flow Diagram (DFD) and a syntactic data flow in Web service composition [YH09]. This research has also been applied to many types of studies [CV92], [YSBR05] [SKH97] that are related to data analysis and research & development. The previous chapter offered four alternatives to content sharing in e-Learning management systems.
Chapter 4. Implementation for Distributed e-Learning Management Systems

The first explained the reasons for sharing learning content. The second described LMS. The third compared methods of sharing learning content. The fourth proposed a new concept of Distributed e-Learning Management Systems (DLMSs). This chapter extends this and presents the implementation of DLMSs.

It presents the concept and principles underlying data flow design and explains how a DLMS is implemented based on a standard protocol. First, it compares the standard protocol that consists of a simple object access protocol (SOAP) Web service and a REpresentational State Transfer (RESTful) Web service. Second, it proposes a data flow for DLMS as well as a class diagram. Finally, this chapter explains the evaluation of the prototype system based on the proposed concept.

4.2 Communication between e-Learning Systems

Figure 4.1 Communication between DLMSs
This subsection presents the concept behind and communication between e-Learning systems to clarify what DLMSs are. A standard protocol should be applied to communication between e-learning systems to effectively implement software, as shown in Figure 4.1. The next subsection will review SOAP and RESTful, which are both based on HTTP.

4.2.1 Principles behind SOAP based Web Services

Web Services are software that is designed to support data exchanges between computers via network systems [SW10]. Most Web services use XML language and SOAP is a representative protocol. The seven-step process for the Web service structure is outlined in Figure 4.2.

1). The client on the user side queries the registry to locate a service as a service requester.

2). The service registry refers the client to the Web services description language
(WSDL) as universal description, discovery, and integration (UDDI, including availability)

3). The client accesses the WSDL document.

4). WSDL provides data to interact with the Web service.

5). The client sends a SOAP message request to the service provider.

6). The service provider processes the request at the server.

7). The Web service returns the SOAP message as a response to the client.

The Service Requestor is a client that requests a service by the service provider that can be searched from the UDDI registry. The Service Registry is a provider that acts as an intermediary to register the WSDL file by using a detailed description of the offered services. The Service Provider is responsible for opening the service. The service requestor in Web services is the same as the user (a person) who navigates information through a Web browser. The data exchange format requires a standard format for both sides to ensure that they can understand each other easily. XML is currently the standard for the assigned detail protocol format.

Let me now explain Registration and Searching (Web Service Registry). When Web users want more information but do not know the exact location (URL), they specify the query terms to find them with a search engine. The concept of a Web service registry has been applied to meet these needs. A computer server that is responsible for this kind of service is called a UDDI registry in the SOAP architecture.
4.2.2 Principles behind RESTful Web Services

There is another architecture for Web services (distributed systems) called REST. The basic principles behind the RESTful structure are outlined in Figure 4.3 [F00], [FT02]. The main idea is to use standard HTTP and XML because data exchange becomes simple. This method uses the GET command for data retrieval, the POST command for appending data to the server, the PUT command for inserting data, and the DELETE command for data removal.

REST Web services are a form of software that regards various data as resources. HTTP allows data to be sent back to users in any XML format. Once users of RESTful Web services know the URL of a REST Web service destination, they are able to read XML data based on their logical requests. Evaluation of the performance of RESTful Web services has been proposed in the past [HSA10]. Moreover, it has been ascertained [LS11] that the working process for RESTful Web services was more efficient than that SOAP Web services.

![Figure 4.3 Principles behind RESTful based Web Services [CP11]](image-url)
Figure 4.4 outlines the basic four-step process for RESTful based Web services, which are listed below.

*Step 1:* LMS$_1$ obtains the URL of LMS$_2$ from the registry by using the HTTP GET method, and requests a service for LMS$_2$.

*Step 2:* When LMS$_2$ receives a request from LMS$_1$, it calculates a command.

*Step 3:* Once LMS$_2$ completes the required processing, it sends the result back to LMS$_1$ by encoding.

*Step 4:* When LMS$_1$ receives the data, it decodes them and displays them to the user.
### 4.2.3 Comparison of Standard Protocols

**Table 4.1 Comparison of standard protocols**

<table>
<thead>
<tr>
<th>Methods</th>
<th>Strengths</th>
<th>Weaknesses</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td>SOAP based HTTP</td>
<td>-Excellent for small domains</td>
<td>-Complicated process</td>
<td>[Winer, 1998]</td>
</tr>
<tr>
<td></td>
<td>-Distributed objects</td>
<td>-Consumes much bandwidth</td>
<td></td>
</tr>
<tr>
<td></td>
<td>-Across platforms</td>
<td>-Scalability is complex</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>-Slower performance</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>-Only supports XML for transfer data</td>
<td></td>
</tr>
<tr>
<td>REST based HTTP</td>
<td>-Good for large domains</td>
<td>-Data security</td>
<td>[Fielding, 2000]</td>
</tr>
<tr>
<td></td>
<td>-Good for distributed media</td>
<td>-Lack of standards</td>
<td></td>
</tr>
<tr>
<td></td>
<td>-Across platforms</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-REST consumes less bandwidth</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 4.1 summarizes the strengths and weaknesses of SOAP and REST based HTTPs. Both are useful to handle distributed objects and media. They have capabilities to connect between systems with different platforms such as Windows and LINUX. RESTful is better than SOAP in two ways: 1) SOAP is effective for small domains but not suitable for larger ones and 2) RESTful consumes less bandwidth than SOAP for the same amount of data. However, RESTful has two weaknesses of 1) poor security during data transfer and 2) a lack of standards because of newcomers.
4.3 Conceptual Design for DLMS

4.3.1 Definitions of Terms

First, let us define the related terms:

The *user interface* is customized software for interaction with a specific organization between the user and LMS.

The *engine* has general software functions for learning and teaching. It refers to learning content.

*Learning content* is data that are stored in the e-Learning system. Examples of learning content are topics, lecturers’ names, and learning documents.

The *learning log* has historical data on user actions. The recorded data include user names and status, time-stamps, and actions.

*Students* are users who study lessons. *Teachers* are users who prepare learning content. *Administrators* are users who are responsible for managing the entire system.
### 4.3.2 Data Flow Design

![Figure 4.5 Design concept for DLMS (a)](image-a)

![Figure 4.5 Design concept for DLMS (b)](image-b)

---

51
Let us direct our attention to Figures 4.5 (a) and (b) to design data flows [LM08], [YH09], [JNKC08] in DLMS. Suppose that a student at organization A accesses an LMS to search learning content through user interface A. First, the system searches the database of Content A (Organization A). Then, it also searches in the other DLMS (i.e., Content B at Organization B) that are nominated in the registry (See Figure 4.5 (b)).

There is a diagram of the data flow in Figure 4.6 based on this analysis of the scenario. The six components in this figure are listed below.

- The *Web interface* is responsible for the main monitor as the provider of DLMSs.
- The *search course module* is responsible for finding the course after receiving keywords from the Web interface and then connecting to the database (Course).
- The *get course content module* receives the content ID through keywords.
- The *call service module* is the center function for connecting DLMSs with other systems.
- The first *another DLMS: Web service module* is a function that is used to connect to the destination.
- The second *another DLMS: Searching course module* is used to search the course by receiving keywords.
Figure 4.6 Data flow diagram for proposed DLMS
4.3.4 Class Diagram

The Class Diagram is another diagram that represents the static relations between objects. There are 14 main classes in DLMS in Figure 4.6 that are listed below.

1. The page class is responsible for handling the screen for the system.
2. The CourseView class displays learning content on the screen.
3. The CourseForm class is the form for the user to add course content to LMS.
4. The menu class displays the menu on the main screen of the system.
5. The ConfigForm class is for the manager to set values such as the destination IP address (LMS).
6. The AddUserContent class is responsible for user management.
7. The user class is responsible for managing all users’ names in the system.
8. The security class is responsible for checking on the security of the systems such as checking the user names and passwords of users who log in to the systems.
9. The WebService class is responsible for sending keywords from searches by users.
10. The CallService class is responsible for using Web services to send and receive data between other LMSs.
11. The image class is responsible for managing images of learning content.
12. The course class is responsible for managing details on learning content.
13. The database class is responsible for managing the database.
14. The controller class is responsible for managing the back office to add, delete, and edit user data.
Chapter 4. Implementation for Distributed e-Learning Management Systems

Figure 4.7 Class diagram for DLMS system
4.4 Design and Evaluation of DLMS

4.4.1 Evaluation of RESTful and SOAP based Implementation for DLMS

Table 4.2 Evaluation of implementation for DLMS

<table>
<thead>
<tr>
<th>No</th>
<th>Detail</th>
<th>RESTful Web service</th>
<th>SOAP Web service</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Response time</td>
<td>1.297 sec</td>
<td>1.568 sec</td>
</tr>
<tr>
<td>2</td>
<td>Turn-around process</td>
<td>4 processes</td>
<td>7 processes</td>
</tr>
<tr>
<td>3</td>
<td>Across platform</td>
<td>Support</td>
<td>Support</td>
</tr>
<tr>
<td>4</td>
<td>Design software</td>
<td>Simple</td>
<td>Complicate</td>
</tr>
</tbody>
</table>

The first considers the Response time. Response time is a response time for computation (receiving searched result displayed on a monitor). We have compared RESTful and SOAP based implementation for DLMS. There are ten students of TNI in the evaluation who were asked to use both systems under conditions where there are two servers as DLMSs with the Internet speed of 250 Mbps. The average of response time for RESTful base implementation for DLMS is less than the average of response time for SOAP based implementation for DLMS. The result for the average response times are summarized in Table 4.2.

The second considers the turn around-process, turn-around process means the working process of DLMS based on RESTful implementation and SOAP implementation. DLMS based on RESTful implementation consists of 4 processes 1) User requests service for searching learning content 2) Destination DLMS computes the request 3) Return the result to users 4)
Display to users. DLMS based on SOAP implementation which consists of 7 processes; 1) The client on the user side queries the registry to locate a service as a service requester. 2) The service registry refers the client to the Web services description language (WSDL) as universal description, discovery, and integration (UDDI, including availability) 3) The client accesses the WSDL document. 4) WSDL provides data to interact with the Web service. 5) The client sends a SOAP message request to the service provider. 6) The service provider processes the request at the server. 7) The Web service returns the SOAP message as a response to the client.

The third considers across platform evaluation. We have tested by setting up RESTful base DLMS on a difference operating system for server which consists of 2 platforms. 1) Setup RESTful based DLMS on the Windows Server 2008R2. 2) We setup RESTful base DLMS on Fedora LINUX. After finished setting up the system, then we tested by searching learning content based on Windows server 2008R2 across through DLMS based on Fedora LINUX server. Then, we tested again as same as last time but we will setup by using SOAP based DLMS which we have got the same result as RESTful base DLMS.

The last considers the design software. Because the working process of SOAP based Web service has a complicated process [SW10]. So, the design of the software application for DLMS will be more complicated, as well. For the working process of the RESTful based Web service has not complicated [FT02]. So, the design of the software application for RESTful base DLMS is easy and more flexibility than the design of SOAP base for DLMS.
4.4.2 Comparison for RESTful and SOAP based implementation for DLMS

Table 4.3 Comparison of implementation for DLMS

<table>
<thead>
<tr>
<th>No</th>
<th>Details</th>
<th>RESTful</th>
<th>SOAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Response time</td>
<td>Very good</td>
<td>Good</td>
</tr>
<tr>
<td>2</td>
<td>Turn-around process</td>
<td>Simple</td>
<td>Complicate</td>
</tr>
<tr>
<td>3</td>
<td>Across platform</td>
<td>Support</td>
<td>Support</td>
</tr>
<tr>
<td>4</td>
<td>Design software</td>
<td>Simple</td>
<td>Complicate</td>
</tr>
<tr>
<td>5</td>
<td>Message for exchange data</td>
<td>No need</td>
<td>Need</td>
</tr>
<tr>
<td>6</td>
<td>Bandwidth consumption</td>
<td>Consumes less bandwidth</td>
<td>Consumes much bandwidth</td>
</tr>
</tbody>
</table>

Table 4.3 summarizes the SOAP and RESTful base implementation for DLMS. They have capabilities to connect between systems with different platforms. RESTful base implementation for DLMS is better than SOAP base implementation for DLMS in five ways: 1) RESTful is less response time than SOAP and 2) Turn-around process of RESTful is less than SOAP base implementation for DLMS 3) The design software of RESTful is simple but the design software of SOAP is complicated 4) Message for exchange data of SOAP is needed but RESTful has no need for message and 5) RESTful consumes less bandwidth than SOAP for the same amount of data.
The research of Distributed e-Learning in the past used SOAP based web service. Also, there are some points that need to improve for algorithm in the term of accessing information which is very slow to access. In addition, SOAP base will be effected on consumes bandwidth. The working process is complicated and other as stated above. This thesis focuses on two main goals. 1) We proposed DLMS base RESTful implementation for learning content sharing. 2) In addition this research proposed the possibility of DLMS for student recommendation based on trajectory mining.

4.5 Conclusion

Chapter 3 presented the design concept for a data flow diagram for a Distributed e-Learning Management System based on the standard protocol and explained how DLMS was implemented. It first compared two standard protocols, i.e., the SOAP Web and RESTful Web services. It was proposed that the RESTful Web service be applied to the prototype implementation as a result of the comparison. Then, it presented Unified Modeling Language (UML) diagrams, i.e., class and data flow diagrams.

The performance of the proposed DLMS was evaluated 4 parts. 1) The average of response time for RESTful base implementation for DLMS is less than the average of response time for SOAP base implementation for DLMS. 2) DLMS based on RESTful implementation consists of 4 processes but DLMS base on SOAP implementation consists of 7 processes. 3) RESTful based implementation for DLMS and SOAP base implementation for DLMS was supported across platform. 4) The design of the software for RESTful base implementation for DLMS is easy and more flexibility than the design of the software for SOAP based implementation for DLMS.
Chapter 5
Possibility to DLMS for Learning Plan Support

5.1 Introduction

In the previous chapters, this thesis has focused on learning contents in DLMS and discussed how to share them transparently. Based on technology comparison, chapter 2 described why DLMS is required. Then chapter 3 clarified how DLMS could be implemented based on web service technology. Further, using knowledge sharing among science teachers, chapter 4 proposed how to use DLMS.

Then this chapter will discuss the different aspect of DLMS because each LMS can store not only learning contents but also learning log. Data mining has already been very popular in market analysis such as customer segmentation and product segmentation [TSTA13], and is also being applied to learning log analysis in LMS [RVG08]. The consideration of the growth prospects for student’s capability based on data mining is an example of earlier research in which a student capability growth structure is extracted from growth trajectories [KSIANT13]. The objective is to find growth patterns found within sample student groups.

In order to identify constraints on growth order among capabilities, the capability structure extraction from growth trajectory has been also proposed [NNIST12]. The formal concept of trajectory mining in the capability space, one of data mining which allows to implement an information system was also proposed [I12]. A practical application on software outsourcing companies’ growth demonstrated the proposed concept [TSTA13] which is based on Capability Maturity Model Integration (CMMI) [PWCC95]. In CMMI, capability assessment involves five
levels: Level 1 (the lowest) to Level 5 (the highest). In [SWTS09], another application for science teachers introduced in the previous chapter also demonstrated why growth analysis based on CMMI was important to identify the level for capability on sharing knowledge.

However, the research discussed above only proposed methods for finding pattern growth and for identifying constraints of growth. They did not clarify how the found patterns and identified constraints were available in the future growth. Then, to upgrade the idea on trajectory mining in LMS, diagnosis and planning is required. If students can identify his weakness or issue for improving his capability and set the direction for growth, they can achieve their final academic goals more easily, successfully and effectively.

Based on the concept of trajectory mining, this chapter proposes the method on planning support in capability growth. To guide students in a study plan, this chapter proposes a function that generates planning support to students based on their growth history (learning log of DLMS). The first section will review capability trajectory mining for learning log. Then the second section will describe the basic idea for proposal. Finally, the third proposes the algorithm of the method.
5.2 Capability Trajectory Mining for Learning Log

5.2.1 Capability trajectory mining

As shown in Figure 5.1, LMS has user interface, common learning engine, learning contents and learning log. This chapter focuses on leaning log which express growth history of students. Based on data mining for learning log, how to make learning plan is an issue of this chapter. The benefit of data mining, called trajectory mining, is (1) to clarify current position for student, (2) to find learning order and (3) to recommend next to do.

Let us define capability space and capability structure, by setting capability \( C_i: C_1, C_2, C_3, \ldots, C_n \). In Figure 5.2, Capability \( C_i \) is growth Y axis and capability \( C_j \) is growth X axis. Each has 5 levels which are used basic idea from CMMI: \( l_1, l_2, l_3, l_4, l_5 \). Unit \( U \) which is a growing object with includes both organizations and individuals. For example, student in Japanese class has an objective to improve their own skills, such as speaking skill, listening skill, writing skill, etc. We will consider growth units. Each capability should be evaluated in
one of five levels; poor, average, good, very good, excellent. In determination each level’s detail will depend on the appropriate supervisor.

On the other hand, we assume that there are more than three capabilities for one subject domain. For example, there are four capabilities in the English language study domain (speaking, listening, reading, and writing) where a unit is a student or learner.

Let us define the capability space. The basic idea arises from the comparison of knowledge space theory [PJ85], [M05] for knowledge acquisition. Capability space is a set of potential capability states. And capability subspace is a set of state vectors that consist of some capability axes. We take 2 axes $C_i$ and $C_j$ for example in Figure 5.2. $X_i(t)$ is a capability state for unit $U_i$ at $X_i(t) = (l_{i1}, l_{i2}, l_{i3}, \ldots l_{in})$. We suppose units only grow at one capability state for one change from state $(l_{i1}, l_{i2})$ to $(l_{i5}, l_{i5})$.

![Figure 5.2 Two axes based Capability Space](image-url)
Capability structure is structure of relation at least 2 capabilities. Figure 5.3 presents two capabilities in $C_i$ and $C_j$. Sometimes levels $l_{i5}$ in Figure 5.3 (a) can be achieved without any condition. However, achieving levels $l_{i5}$ in Figure 5.3 (b) may sometimes require another condition. For example, $l_{i5}$ should be achieved before a unit achieves $l_{i4}$ in other words; $l_{i3}$ and $l_{j3}$ are prerequisites for achieving $l_{i4}$.

![Figure 5.3 Example of capability structure](image)

An example of visualization is shown in the type of vector. Each path can reach the destination or goal depending on individual’s capability level. The current state can be plotted in capability space. We can visualize path by using data of student’s growth log. Also, we can assign capability level in each capability. For example, capability of Japanese reading level
could be in “poor”, “average”, “good”, “great” and “perfect”). The example is shown in Figure 5.4. Figure 5.4 also illustrates the growth trajectory.

Figure 5.4 Example of visualization path and growth trajectory

Capability structure extraction consists of three steps. First step is to record growth. This step will collect data by questionnaire from students (learning log). In detail, students will collect data of achievement year in each capability’s level (See Figure 5.5).
Second step is to create visualization path which traverses capability state. Each path can reach the destination depending on individual’s capability level. Third step is the process of generating structure. We call it capability structure extraction. For a part of algorithm in extraction, we use ISM Interpretive Structural Modeling and SPM Sequence Pattern Mining.

### 5.2.2 Possibility under DLMS

Let us consider the possibility of trajectory mining under DLMS. The concept of trajectory mining does not consider the IT environment whether it is stand-alone or distributed. Under DLMS environment, learning contents are stored by organization as shown in Figure 5.6. Then the goal of trajectory mining under DLMS can be to clarify the difference among organizations. For example, Figure 5.7 (a) and (b) show the difference between organization A and organization B. Therefore, there are two possibilities:
(1) Comparing current distributions of students in organization, the system can suggest diagnosis if an organization is going head or not.
(2) Checking capability structure in superior organization, the system can show the better direction for growth.

5.3 Basic idea for proposal

5.3.1 CMMI & SPICE

Based on the previous intuitive possibility of trajectory mining under DLMS environment, this section discusses the premise of our proposal: CMMI and SPICE. Using the capability levels and maturity levels, CMMI is an extension of CMM which is prepared for evaluating software organizations. Basic idea of CMMI is shown in Figure 5.8.

In the previous research, CMMI was embedded in SPICE (Spiral Enhancement Capability Support System) which is a learning support system. The structure of SPICE is shown in Figure 5.9. SPICE is able to store the growth log of students and to visualize their state transition. In order to store log for analysis the growth trajectory and capability structure, this chapter considers how to enhance SPICE.

![Figure 5.8 Capability Maturity Model Integration](image_url)
5.3.2 Plan–Do–See Cycle

As shown in Figure 5.10, Plan–Do–See is a well-known principle applied to improve the quality of general management [SD13]. For learning, this concept is also important. In this context for SPICE, “Plan” suggests next goal which is described by capability and its level. For example, let us improve capability $C_i$ for level $l_{i4}$ and capability $C_j$ for level $l_{j3}$ in six months.

“Do” is actual learning action in the domain. “See” is the evaluation work to PLAN. For example on a student, their levels of capabilities are evaluated: some capabilities have been achieved as plan but others have still stayed in lower levels than planned levels. Based on “SEE”, next “PLAN” should be set.
Chapter 5. Possibility to DLMS for Learning Plan Support

Figure 5.10 Plan–Do–See cycle

However, SPICE only supported “SEE” activity. It did not support “Do” nor “PLAN”. Table I shows the comparison between available functions of the process discussed in [I12] and this research: FMG and planning support. FMG (finding message generator) shall support the “Plan” function. The proposed planning support function supports both “Plan” and “See” functions in the PDS cycle as discussed later.

<table>
<thead>
<tr>
<th>No</th>
<th>Function</th>
<th>Plan</th>
<th>Do</th>
<th>See</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Visualization</td>
<td>-</td>
<td>-</td>
<td>●</td>
</tr>
<tr>
<td>2</td>
<td>FMG</td>
<td>●</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>3</td>
<td>Panning support</td>
<td>●</td>
<td>-</td>
<td>●</td>
</tr>
</tbody>
</table>
5.4 Algorithm for Functions on Message Generation

5.4.1 Process for planning support function under DLMS

Let us clarify the algorithm for function on message generation in SPICE. As shown in Figure 5.11, there are four steps for planning support. The first process is to record student growth logs. This process may collect data through student questionnaires in some cases (if digitized history data is not available). Students provide achievement data periodically reflecting their current capability levels.

The second process builds a visualization path for each capability level. Each path may indicate the planned destination depending on an individual’s current capability levels.
Chapter 5. Possibility to DLMS for Learning Plan Support

The third step consists of two functions for planning support: (1) FMG generates the extraction message from student learning logs. Interpretive Structural Modeling (ISM) [W74] was used to find dependencies between capability pairs. (2) IPG generates future direction as plan from extraction of a student’s learning logs.

Finally, the last step provides overall results. One result comes from the FMG function to recommend enrolling students in the types of student groups. Another result from the IPG function will recommend an individual study plan.

**Term Definition**

*IPG* is the plan that generated from the extraction of a student’s learning history. IPG function will recommend an individual study plan.

*FMG* is the extraction’s message from students’ learning histories. FMG function will recommend enrolling students in the type of student groups.

Chapter 5 presents the possibility of DLMS by learning log. Data analysis will use learning log of each DLMS by using Trajectory mining. FMG means the function of data analysis of learning log. FMG function will help to advise students as a group. IPG means the function of data analysis of learning log, as well. IPG will help to advise students as an individual. FMG and IPG mean functions of using data for learning log of DLMS in order to analyze and create studying plan for student.

FMG follows three steps as follows:

**5.4.2 Flowchart for FMG**

In Step f1 of Figure 5.12, the process collects the values of Relation and Necessary by setting all capability pairs in the object class. For this process, Loop (Legacy Calculation Loop)
is used to calculate the values of Necessary. Then, the process creates Object class, which is the agent for Relation, and then collects all Relations’ data (Axis i, Axis j, Necessary) in that page to create a list used in Step f2.

Figure 5.12 Flowchart for FMG (Step f1)
Step f2 of Figure 5.13 checks whether there is the Strong Dependency Relation or not as follows:

*Searching for Strong Dependency*: the first step in searching for Strong Dependency values, introduces the variable Max Relation Count for determining the maximum number of relationships of type Relation. An array will be packed only with Strong Dependency Relation values (the Strong Dependency Relation Array). The algorithm begins by setting Max Relation
Count equal to 1. After that, it performs the Loop to evaluate each Relation obtained from the previous step.

If the value of relationships of Relation in the Loop is higher than that of Max Relation Count, and then the algorithm performs the following three steps:

1) Increase Max Relation Count to equal to the number of relationships of Relation in the Loop.

2) Clear the Strong Dependency Relation Array.

3) Insert the Relation value in the Loop into the Strong Dependency Relation Array.

If the value of Relation’s relationship in the Loop is equal to Max Relation Count, it inserts the Relation value in the Loop into the Strong Dependency Relation Array. If the value of Relation’s relationship in the Loop is lower than Max Relation Count, no action is taken.

Searching for No Dependency: to confirm that there is No Dependency, the principle similar to that for searching for Strong Dependency is applied. In this case, the algorithm confirms whether the value of each Relationship’s Relation is equal or not equal to 0. If the number is equal to 0, the algorithm places the Relation into the No Dependency Relation Array. Step f3 simply displays the Strong Dependency and No Dependency.

5.4.3 Flowchart for IPG

This part describes the flowchart for the IPG function. The process begins by reading output data from the visualization function. Then, it sets advice or recommendation type to create planning support. Later, the student status is checked.
Chapter 5. Possibility to DLMS for Learning Plan Support

The process has three conditions that must be checked; the first condition checks whether a student is over focused on one side of the capability structure, and if so, it is recommend that the student should get back to studying other skills first. The second condition determines if a student has completed two capabilities in level 5, the result message is “You are excellent,” etc. The third condition check is to determine if a student’s data shows that both capabilities have not yet been completed at level 5, and then the process will check current student levels.

![Flowchart of IPG](image)

**Figure 5.14 Flowchart of IPG**

The process loops to check the conditions searching for paths that have the largest number of students and will choose that path and provide planning support advice for each path. Functions will continually check until the student’s status shows completion at level 5 in both skills. Finally, it will display the output from the IPG function with the FMG function as stated above.
5.4.4 Example

As case study, growth log data on eight capabilities of Thai-Nichi Institute of Technology are collected. They are undergraduate students from the information technology, business Japanese and computer engineer, and graduate students from IT. Total student number is 60. The defined capabilities are shown in Table 5.2. Each capability has capability levels. Due to the sample group’s institute has been taught in Japanese language which is a mandatory course. Figure 5.15 shows an example of questionnaire and Figure 5.16 shows an example of capability levels.

<table>
<thead>
<tr>
<th>No.</th>
<th>Detail of capabilities</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Japanese skill</td>
</tr>
<tr>
<td>2</td>
<td>English skill</td>
</tr>
<tr>
<td>3</td>
<td>Java programming skill</td>
</tr>
<tr>
<td>4</td>
<td>Study planning ability</td>
</tr>
<tr>
<td>5</td>
<td>Presentation skill (Power point)</td>
</tr>
<tr>
<td>6</td>
<td>E-mail skill</td>
</tr>
<tr>
<td>7</td>
<td>Web skill</td>
</tr>
<tr>
<td>8</td>
<td>Database skill</td>
</tr>
</tbody>
</table>
### Japanese skill

<table>
<thead>
<tr>
<th>Level</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Can use basic Japanese. (Ohayo, Arigato, Oishi, Sumimasen).</td>
</tr>
<tr>
<td>2</td>
<td>Can roughly use daily life Japanese. (shopping, travel)</td>
</tr>
<tr>
<td>3</td>
<td>In addition to daily life Japanese, you can read Hiragana and Katakana.</td>
</tr>
<tr>
<td>4</td>
<td>Can understand a wide range of contexts spoken in Japanese and write Japanese sentences.</td>
</tr>
<tr>
<td>5</td>
<td>Can enjoy TV program/Movie and write logical sentence.</td>
</tr>
</tbody>
</table>

### English skill

<table>
<thead>
<tr>
<th>Level</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Can use basic English. (Good morning, Thank you, Delicious, Excuse me)</td>
</tr>
<tr>
<td>2</td>
<td>Can roughly use daily life English</td>
</tr>
<tr>
<td>3</td>
<td>In addition to daily life English, you know more than 1,000 words.</td>
</tr>
<tr>
<td>4</td>
<td>Can understand a wide range of contexts spoken in English and write English sentences.</td>
</tr>
<tr>
<td>5</td>
<td>Can enjoy TV program/Movie and write logical sentence.</td>
</tr>
</tbody>
</table>

**Figure 5.15 Example of questionnaire**

**Figure 5.16 Example of explanation levels**
In Figure 5.17, the student is currently between DB skill level 4 and Java skill level 4. In this example, the student must achieve the Java skill level 5 in order to have a chance to achieve the DB skill level 5 at first. In addition, when the characteristics of all students are considered, a minimum Java skill of level 3 is required to reach a DB skill of level 5. From the group sample, a general statement can be made that students need to study and pass the test of Java skill level 3 before they can take the skill level 5 DB test.

5.5 Conclusion

To consider the possibility of future DLMS, learning trajectory mining was introduced. Under DLML environment, trajectory mining helps to find the difference of current position and
learning order among organizations. The algorithm for generating recommendation called FMG has been proposed.

This chapter proposed an automated function of planning support for students based on trajectory mining and have also presented a case study example. The goal of this part is to provide a method to guide students to select a study plan to further their academic growth. This chapter proposes the process of a planning support function comprising four steps, as previously discussed. We choose one student out of 60 to demonstrate a study plan to achieve success in two capabilities and show how the plan depends on the interrelationship between the two capabilities.

The current proposal focuses on planning support functions for students involving two functions. The FMG and IPG function is a part of two planning support functions. This study illustrates the finding message of dependency only for capability pairs. The result may be not enough to create a study plan for student groups in which capabilities with multiple dependencies exist. In future research, we will investigate other methods to accomplish the finding message function in order to guide students with effective group tailored study plans.
Chapter 6
Conclusion

This thesis has proposed a new concept called the Distributed e-Learning Management System (DLMS) to discuss the benefits of sharing of learning content in e-Learning systems. It presented problems and their solutions by explaining that e-Learning systems have three components of user interfaces, common learning engines, and domain oriented learning content. Moreover, the thesis also proposed the future possibility of DLMS based growth of capabilities in trajectory mining.

The main focus in the problems was how users could find and use learning content in LMSs that were distributed over networks in schools or organizations. There were four chapters (2–5) in the body of this thesis, which are described below.

Chapter 2 presented alternatives for sharing learning content in e-Learning systems to explain why DLMS was required. Then, it focused on the importance of DLMS, and why DLMS was required. First, the chapter introduced four alternatives for sharing learning content in e-Learning systems. The first alternative was a method of replicating shared learning content. The second alternative was a method of sharing centralized content. The third alternative was a method of sharing remote login content (multi-login). The fourth alternative was a method of sharing distributed content. This chapter also defined the term DLMS and clarified the requirements for the DLMS. Chapter 2 is also compared the performance between DLMS and LMS. Comparing the alternatives for sharing learning content in e-Learning systems, the chapter concluded that methods of sharing distributed content became more important as the number of organizations increased.
Chapter 6. Conclusion

Chapter 3 introduced an application model for the proposed DLMS to enable its practical use. It also proposed transparent sharing of digital content by science teachers. First, the background and maturity levels were presented based on the socialization, externalization, combination, and internalization (SECI) and CMMI models for digital content sharing by science teachers. Next, the chapter explained the evolutionary process for content sharing that allowed science teachers to appropriately share digital content. Finally, it provided four scenarios from initial to optimized states to demonstrate how the proposed framework worked.

Chapter 4 presented the design of data flows for DLMS based on standard protocols and explained how DLMS was implemented. The standard protocols that consisted of SOAP Web and RESTful Web services were first compared in this chapter and the latter was found to be more advantageous in the prototype implementation. In addition, chapter 4 have evaluated the performance of DLMS implementation for REST base and DLMS implementation for SOAP base. A description of Unified Modeling Language (UML) diagrams, i.e., class diagram and data flow diagrams were presented in this chapter. Finally, it was concluded that DLMS became more powerful as the number of LMSs that had sharable learning content increased.

Chapter 5 discussed the future possibilities of DLMS by introducing the capabilities of trajectory mining. Then, new added support functions for learning plans were proposed that could be separated into two parts. The first part was a finding message generator (FMG) function that was responsible for finding different features in groups based on students’ learning logs. The second part was an individual planning generator (IPG) that generated learning plans from students’ learning history logs for individual students. The chapter indicated the possibility of designing a questionnaire for a sample group (TNI students’ capabilities) in a DLMS and using an algorithm for message generation.

DLMS in practical use should be evaluated in the future. The two functions (FMG and IPG) should also be implemented and evaluated.
Acknowledgements

First I want to thank Professor Hiroshi Tsuji of Osaka Prefecture University and Professors Krisada Visavateeranon and Sura Lekhakul of Thai-Nichi Institute of Technology for their continuous support during my doctoral course and beyond. Their guidance and advice led this work to a successful outcome.

Furthermore, I want to thank Professors Koichi Kise, Hideki Tode and Kiyota Hashimoto for giving up their valuable time to be my supervisors and giving me helpful feedback. Without their encouragement, I may have had to give up on my thesis.

Huge thanks also go to all members of the Tsuji Laboratory of Osaka Prefecture University, especially to Professor Ryosuke Saga, for their immense support. I had a pleasant time while I was in Japan. I want to thank Ms. Miho Yoshihara for arranging the draft and documents. In addition, I would like to thank Mr. Mauricio Letelier for checking my draft. I am most thankful for the support of my wife, Ms. Benjaporn Kaewkiriya, my parents, Mr. Chub and Ms. Wilai Kaewkiriya, and all of my family during this demanding time.
Bibliography


Technology, Chengmai University, Thailand, 2005.


