INFORMATION AND SYSTEMS IN EDUCATION, Vol. 9, No. 1 2010
pp. 85–100

Adapting the Drill-and-practice-type e-Learning System SANNO KNOWLEDGE FIELD to SCORM Standards and Developing Its Contents

Kaoru Matsumoto[*], **Hiroshi Miyauchi**[*] **and Akihiko Koga**[**]

Abstract This paper describes redesign of the drill-and-practice-type e-learning system SANNO KNOWLEDGE FIELD (SKF) and its contents in order to make them compliant with SCORM 1.2 and SCORM 2004 international standards and while maintaining the learning functionalities. SKF was developed as an ASP-type WBT and its contents have been implemented with the proprietary specification. Such specification makes the system very suitable and has an advantage in not causing any compatibility problem. However, it is difficult to deliver the contents to other e-learning systems. Redesigning the contents to be compliant with the SCORM standards enables them to run on various types of learning management systems worldwide.

Keywords: e-Learning, internet, WBT, SCORM standards

1. Introduction

An e-learning system SANNO KNOWLEDGE FIELD (SKF) was released in 2001 by the SANNO Institute of Management(1), and it was in use by over 250 companies in 2006. The SKF is an ASP-type (Application Service Provider-type) WBT (Web Based Training) system on the Internet. Part of it is implemented with the proprietary specification.

At present, software for many LMSs (Learning Management Systems) has been released as WBT systems. They are based on SCORM 1.2 (Sharable Content Object Reference Model 1.2)(2–5), developed by the ADL (Advanced Distributed Learning) Initiative through the US Department of Defense.

There are several advantages in using internationally acknowledged standards such as SCORM for LMSs. These advantages include a broadened range of contents selection, reduced cost and improved quality by minimizing the scope of self-development and expanding the contents market, and assured quality by helping content vendors to provide contents, etc. The Standardization Promotion Committee in the E-Learning Consortium Japan is promoting the use of SCORM 1.2 by means of the program of SCORM Assessor and LMS-Contents certification.

On the other hand, when the SKF project started, the first version of SCORM had just been established. At that time, it was difficult to foresee the technology trends that would follow SCORM, and we were concerned about potential implementation problems that might arise by the technical restrictions of SCORM. Specific problems were lack of learning object sequencing and insufficient learning history management. We thought that these problems should be solved to implement the SKF concept. Therefore we designed and developed the original SKF system as an ASP-type WBT without SCORM to avoid its restrictions. The SKF contents were made with the proprietary specification.

As the number of systems based on SCORM 1.2 progressively increased, we got requests from customers to introduce these LMSs based on SCORM 1.2 to their situations. We found that, to a certain extent, SKF functions could be implemented with the SCORM 1.2 standard. In addition, we realized that most of the LMSs based on SCORM 1.2 were working stably and the market for compliant content was expanding quickly.

Thus we designed and developed the SKF contents based on the SCORM 1.2 standard. The developed contents were designed according to the original SKF system and contents with some restrictions on functions for choosing a way of learning, sequencing of learning objects and managing of a personal history.

When the latest version of SCORM, version 2004 (SCORM 2004) was developed by the ADL(6, 7), it had a definition for content sequencing and provided user interface control navigation, so it had sufficient functions to make a fully functional SKF system.

Then, we developed a content converter from the

[*] The SANNO Institute of Management, Japan
[**] School of Information-Oriented Management, SANNO University, Japan

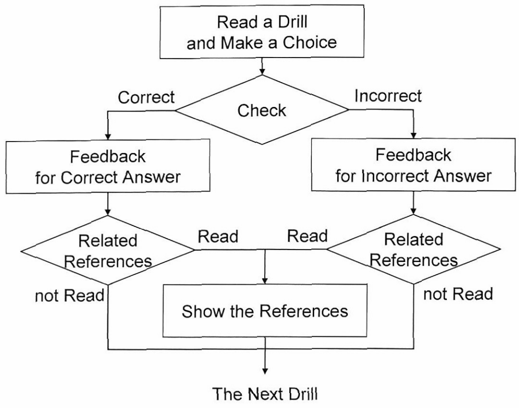


Figure 1. Flow of Response and Feedback Model for the SKF-ASP.

original SKF contents to contents for SCORM 2004, and we prepared the prototype SKF contents for SCORM 2004 using the content converter[8].

In this paper, we present how we realized the same learning as on the original SKF system using the SCORM international standards and we discuss various problems encountered in the realization.

Section 2 overviews the SKF system and contents with the proprietary specification. Section 3 and 4 describe the content design and implementation for SCORM 1.2 and SCORM 2004. Section 5 reports on our evaluation by an interoperability validation experiment. Section 6 gives our suggestions for standardization, and Section 7 presents our conclusions.

Hereafter, the SKF original system with the proprietary specification is referred to as SKF-ASP, the SKF contents for the SCORM 1.2 edition is SKF-SCORM 1.2, and the SKF contents for the SCORM 2004 edition is SKF-SCORM 2004.

2. Outline of SANNO KNOWLEDGE FIELD

First, we show the learning flow of the SKF-ASP, and then provide the design of the user interface, computer servers, software and content structure, and learner support.

2.1 Learning flow

The SKF-ASP was developed on the basis of programmed instruction theory[9,10] which is used in many computer-based training systems, and it satisfied the following requirements.

- Small step-sized contents
- Immediate feedback (both positive and negative) to the learner's response
- Immediate confirmation by the learner
- Self-paced learning by the learner

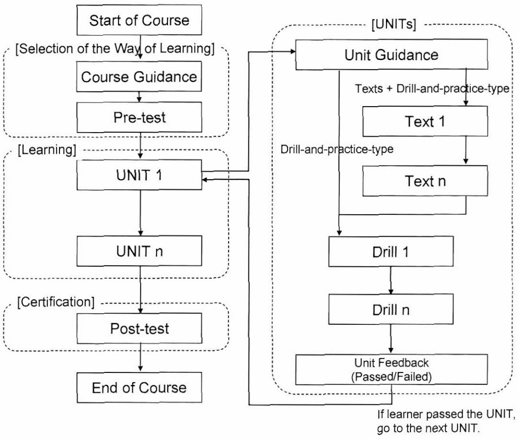


Figure 2. Learning Flow on the SKF-ASP.

Figure 1 describes the flow of the response and feedback model for the SKF-ASP. A learner reads a drill, thinks about an answer and makes a choice. Then, the flow shows an answer (positive or negative) and gives related references as the feedback. The learner can confirm the result of his/her choice immediately.

In the SKF-ASP, this flow is called a "UNIT." One UNIT has one unit guidance, texts, drills & answers, and related references. The learner can start, suspend and resume one UNIT independently.

Figure 2 shows the whole learning flow of the SKF-ASP. The SKF-ASP learning consists of three phases: "Selection of the Way of Learning," "Learning," and "Certification," The "Learning" phase has plural UNITs (generally, there are 10 to 16 UNITs).

First of all, in the "Selection of the Way of Learning" phase, the learner is shown one course guidance to introduce the course overview by the SKF-ASP system. After that, the learner is shown a screen to select the way to learn: "Texts+Drills" (the pattern of reading the texts first and then trying the drills) or "Drills" (the pattern of trying the drills only). If the learner chooses the "Texts+Drills," all of his/her UNITs will be initially set with the flow of reading the text first followed by trying the drills. If the learner chooses "Drills," all of his/her UNITs will be initially set with the flow trying the drills only. Then each learner is shown a Pre-test to identify the level of his/her knowledge.

The Pre-test consists of typically questions from each UNIT, and there is one test per one UNIT. If the learner passes the test, the SKF-ASP system judges that he/she has already understood the corresponding UNIT, and it sets the way to learn to the "Drill-and-practice-type" to allow the style of learning without reading the texts. However, if the learner cannot pass the test, the SKF-ASP system sets to the "Texts+Drill-and-practice-type" to force the learner to read the texts first. In other words, in accordance with the correct answer conditions, there can be mixed flow containing "Drill-and-practice-type" and "Texts+Drill-and-practice-type."

Next, the learner is shown a screen to set up his/her learning schedule. In this screen, the learner has to decide and input the deadline date for learning him/herself, and suitable dates for an end schedule for each UNIT are generated and shown by the SKF-ASP system.

After this procedure, the learner is moved to the second phase. In the "Learning" phase, the learner is shown a table of contents (TOC), and the learner learns these UNITs sequentially according to its order. If he/she chooses the "Texts+Drill-and-practice-type," the learner is shown the texts first and after that tries the drills. If he/she chooses the "Drill-and-practice-type," the learner tries the drills from the first. When he/she completes these drills and the score is above the passing score, the status of this UNIT is set as finished.

After the completion of all UNITs, the learner is moved to the third phase. In the "Certification" phase, the learner can try the Post-test. When the score of this test is above the passing score, the completion screen appears and the learner is able to download the certificate PDF file for the SKF course as evidence.

2.2 User interface

In consideration of limitation of expression by the web browser, we designed the user interface of the SKF-ASP accordingly, and displayed characters were not enumerated simply; the number of characters per page was restrained; and learners read the content by pressing a button as though they were turning the page of a textbook. We divided the layout of buttons into two areas so as not to confuse learners about the control. The first area is for controlling the contents and the second area is for the SKF-ASP system, and the related information are displayed at a short distance.

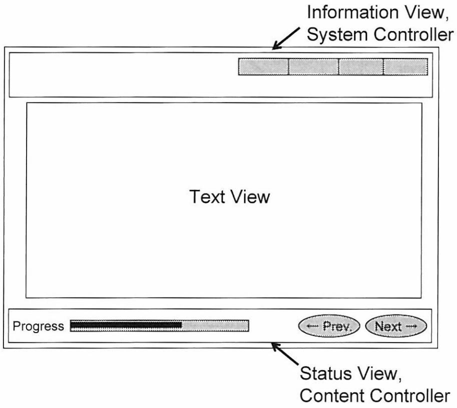


Figure 3. Overview of SKF-ASP User Interface.

Figure 3 shows an overview of the SKF-ASP user interface. The control buttons for learning (e.g. turning over the page; responding with an answer) are located at the bottom of the screen, and the control buttons for the system (e.g. asking a question; going back to the TOC; stopping the learning) are located at the top of the screen. In addition, all these control buttons are also located on the right side and the information views are on the left side. In addition, the text view is at the center of the screen. The user interface and screen design are common specifications of the SKF-ASP, therefore, all designs conform to them.

2.3 Server components

Figure 4 shows the server components of the SKF-ASP. It has three functions: contents delivery, database (DB) (contents structure, learner and mail information) and mail delivery (sending email to encourage the learner).

To deliver the contents, the SKF-ASP has three application servers (AP1 to AP3). When the learner is connected to the SKF network, the contents are delivered from the application servers via the load balancer.

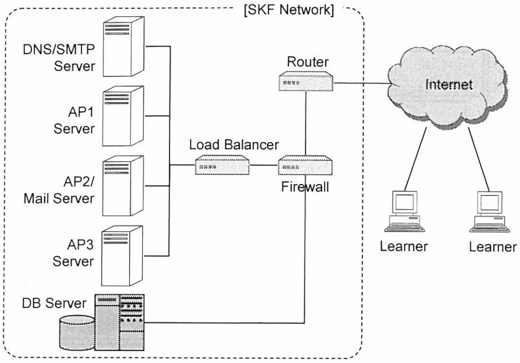


Figure 4. Server Components of SKF-ASP.

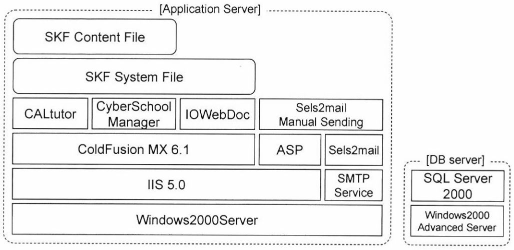


Figure 5. Software Components of SKF-ASP.

To handle the DB, the SKF-ASP has one DB server. The DB server is isolated from the Internet for security, and the application servers AP1 to AP3 can get information by connecting to DB server via the firewall.

Figure 5 shows the software components of the SKF-ASP. The application servers AP1 to AP3 run on the Windows2000 Server, and the web server software is IIS5.0. In addition, the SKF-ASP uses ColdFusionMX6.1 as web middleware, and CALTutor and CyberSchool are running on the middleware. CALTutor is testing learning platform software and CyberSchool is learning management platform software made by NTT Software Corporation. They were customized by NTT Software to work as one system. Furthermore, there is a module for making the PDF file which creates the certificate for completion of the SKF course.

To deliver emails, the SKF-ASP runs the appropriate software on the application server AP2. There is one function to send encouraging emails to the learner automatically or manually by the operators of the SKF-ASP, and another function to send question emails from the learner to the lecturer and to send emails about password reissue.

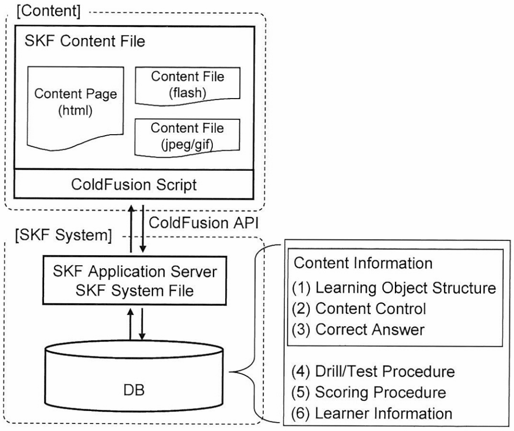


Figure 6. Content Architecture of SKF-ASP.

The DB server runs on the Windows2000 Advanced Server, and its DB system software is SQL Server2000.

2.4 Content architecture

Figure 6 describes the content architecture of the SKF-ASP. The information on content, the learner, and management are managed by the DB. Scoring and sequencing navigation are made by connection to the DB via ColdFusion API.

Each content file (including directories) is named by a fixed rule. Course Guidance, Pre-test, Text, Drill and Post-test are made as separate modules, and they are managed under another directory.

2.5 Learner support

The SKF-ASP learners are supported by a call center and email. They telephone the call center, making inquiries mainly about the learning procedure and operation of the PC. They get information by automatically sent emails or from the screen information they see.

Table 1 summarizes the automatically sent emails in SKF-ASP. The targets of the emails are the learners and training manager (this includes the person in charge of the training at the customer's facility and in SANNO). Warning emails about starting the training and about the approaching deadline are sent to not only

Table 1. Summary of Automatically Sent Emails in SKF-ASP.

Object	Target	Detail
Registration	Learner/ Training manager	User registration
		Training registration
		Password change
Alarm	Learner	Opening notice
		Warning to start the training
		Progress notice (days 10, 20, 30)
		Warning of the deadline
		Completion notice
	Training manager	Non-completion notice for a learner (Before the expiration)
		Warning to start the training
		Warning of the deadline
		Completion notice

the learner but also the training manager. Because of this, when the learning progress conditions are not smooth, the training manager can work on the learner to complete the training.

3. Content Design for SCORM 1.2

After a short outline of the SCORM 1.2 architecture, we discuss how to apply it in the SKF-ASP. Then we provide implementation methods of contents for the SKF-ASP based on SCORM 1.2.

3.1 SCORM 1.2 architecture

The SCORM 1.2 standard was designed for reuse and sharing of WBT-type e-learning content. SCORM 1.2 has two sub-specifications.

- The Content Aggregation Model (CAM)(4) specification which defines a data model of the hierarchical course structure and its binding to the XML.
- The Run-Time Environment (RTE)(5) specification for Sharable Content Object (SCO)/LMS communication which includes the JavaScript Application Program Interface (API) and the data model.

SCORM 1.2 defines functions of the LMS as “Show the TOC,” “Deliver the learning resources associated/corresponding to the learner’s choice from the TOC,” and “Receive and store the learner’s progress status, score, etc.”

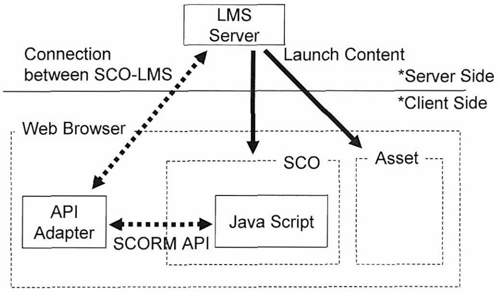


Figure 7. SCORM Content RTE (Drawn based on reference (5), p.3).

The roles of content in SCORM 1.2 are to “Display the learning screen to the learner,” “Decide the learner status (completed/incomplete/etc.),” “Send the status of learning to the LMS,” and “Score and send the result to the LMS.”

Figure 7 is an overview of the SCORM content RTE. In SCORM 1.2, content launched on the client side is called a learning resource. Learning resources are classified into assets and SCOs. Assets are the most basic form of learning resources consisting of html files, audio, still images, video, etc. SCOs are learning resources which communicate with the LMS via the SCORM API on the RTE. An SCO is a minimum unit of content which can be tracked by the LMS using the SCORM API.

The only data connections allowed between the LMS and SCOs are via the SCORM API, and SCOs must be independent from each other. Only one SCO can be executed on the system at one time, and execution of another is not allowed (nested SCO; moving SCO by SCO control).

Figure 8 is an overview of the SCORM CAM. It aggregates the learning units (course, chapter, section, module, etc.) and describes the order of learning contents and their structure.

Figure 9 gives the structure of the SCORM content package. The content structure of SCORM 1.2 is described by a manifest file (imsmanifest.xml), and the TOC of the course is generated and linkages of the physical files are established by the manifest. In addition, the order of learning can be described in the manifest. However, this function is not defined as mandatory in the SCORM 1.2 standard; therefore, it depends on the judgment of the LMS vendor whether or not this func-

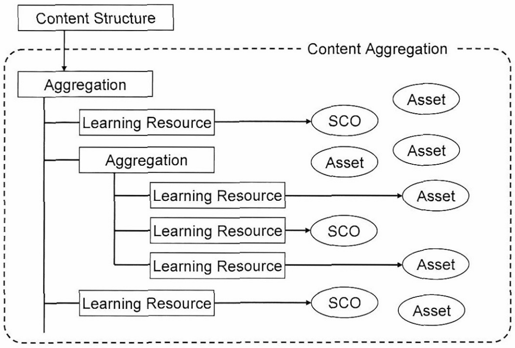


Figure 8. SCORM CAM
(Drawn based on reference (4), p.7).

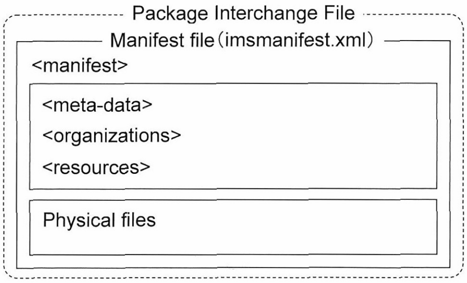


Figure 9. Structure of SCORM Content Package
(Drawn based on reference (4), p.111).

tion is supported.

More detailed information about the SCORM 1.2 standard is given in references (3)–(5).

3.2 Study of specifications

We considered how the LMS compliant with SCORM 1.2 is realized, based on the SKF-ASP system. Table 2 shows the requirements for implementation and the supported functions of SCORM 1.2.

The problems in implementation of the contents of the SKF-ASP on SCORM 1.2 are: inability to customize the navigation buttons; inability to control the learning sequencing; inability to implement the function for selection of the way of learning, because the sequencing function is not defined; inability to move between the UNITs directly; inability to set automatic sending of emails and various notices. Among these limitations, we considered the way to implement the contents of SKF-SCORM 1.2 with the following results.

Table 2. Requirements for Implementation and Supported Functions in SCORM 1.2.

Requirements for implementation	Supported functions in SCORM 1.2
Divert the content files.	Supported (It is based on the W3C standard.)
Reproduce the learning content structure.	Supported (It is possible to express an equal.)
Manage the learning personal history.	Partly supported (It depends on the LMS, because the mandatory and optional data model are mixed.)
Use navigation buttons.	Partly supported (It is impossible to navigate between SCOs from the content side.)
Control the learning order.	Partly supported (Supported situation depends on the LMS.)
Reproduce selection of the way of learning.	Not supported (Sequencing function is not defined.)
Allow movement between the UNITs.	Not supported (SCO cannot be called from another SCO.)
Email and various notice functions.	Not supported (Not defined. It depends on the LMS.)

- The SKF-ASP content files for texts, drills, certificate test are diverted to the SKF-SCORM 1.2.
- The included text and drill in one UNIT is shown as separate SCOs in the TOC.
- The course guidance shown the first time is shown as one SCO in the TOC.
- The lowest limited data model is used for the learning history information.
- The navigation buttons are displayed only inside the SCO.
- The learning order (per SCO sequencing) function is made available by the learner's choice.
- The "Selection of the Way of Learning" function is disabled. Only "Texts + Drills" is allowed.
- The function for automatic sending of emails and various notices is removed.

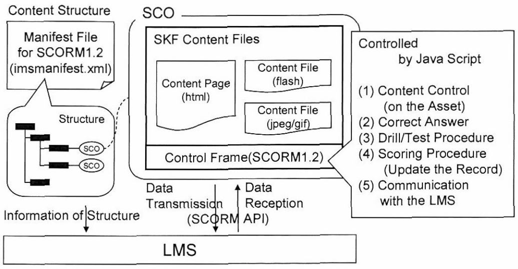


Figure 10. Content Architecture of SKF-SCORM 1.2.

Table 3. Functions of SKF-ASP Assigned to SKF-SCORM 1.2.

Function	SKF-ASP	SKF-SCORM1.2
Management of learning object structure	On the DB server specification	Manifest file (imsmanifest.xml)
Management of control information (scoring, content control, etc.)	On the DB server with the proprietary specification	Java script in learning object file (done by the asset)
Communication with the LMS	SKF application with the proprietary specification (ColdFusion API)	SCORM API

3.3 Implementation of content

The architecture of the SKF-ASP was changed in accordance with the principles discussed in the previous section and summarized in Figure 10. Table 3 describes the functions of the SKF-ASP assigned to the SKF-SCORM 1.2.

First, the learning object structure information that was managed by the DB server was defined in the manifest file. In addition, the learning object sequencing control function by the web middleware ColdFusion via the DB was exchanged with Java script control. The related information files for this control were located under the same directory with the content files. Furthermore, the Drill/Test procedure (to evaluate if the answer is correct) and Scoring procedure (to calculate the percentage of total score) were processed by Java script, and the files and the correct answer information were located under the same directory.

In addition, the procedure for transmitting these results to the LMS via the SCORM API was implemented. In the SCO of course guidance and texts which are judged to be completed by just reading, the status value of "completed" is set in the SCORM data model value of cmi.core.lesson_status and the value is transmitted to the LMS when the learner browses the last page of the text. In the SCO of drills, when the learner tries and finishes the drills, the learner's score is set as the value of cmi.core.score.raw, and if it is higher than the passing score, the status value of "passed" is set in the SCORM data model value of cmi.core.lesson_status. But, if it is lower than the passing score, the status value of "failed" is set. After that, the status value is transmitted to the LMS via the SCORM API.

In the SKF-ASP system, these procedures were realized by the server side script and the DB. However, in the SKF-SCORM 1.2, all these procedures were realized by the client side script. The LMS manages only the learning contents delivery and data connections to the SCOs.

The scope of the SCOs in SKF-SCORM 1.2 was designed as given in Figure 11, and the other contents (Pre-test and the learning pattern of "Drills" in the SKF-ASP) were removed. The content files of the SKF-SCORM 1.2 consist of a Course Guidance SCO, Text SCOs, Drill SCOs, a Post-test SCO and a manifest file for file binding and definition of TOC.

Figures 12 and 13 are examples of TOC screens for the SKF-ASP and the SKF-SCORM 1.2. In the SKF-ASP, both texts and drills were defined together as one item of the TOC. On the other hand, in the SKF-SCORM 1.2, both texts and drills were defined as two separate items. Furthermore, in the SKF-ASP, the course guidance was shown at the beginning of the first learning session, not in the TOC. However, in the SKF-SCORM1.2, the course guidance was handled as an SCO, and shown in the TOC.

More detailed information about the SCORM API procedure is given in Chapter 4 of reference (11).

3.4 Implementation of user interface

In the SKF-ASP, to provide a consistent user interface for learners, the user interface controls for changing pages and asking questions were unified. Also, the

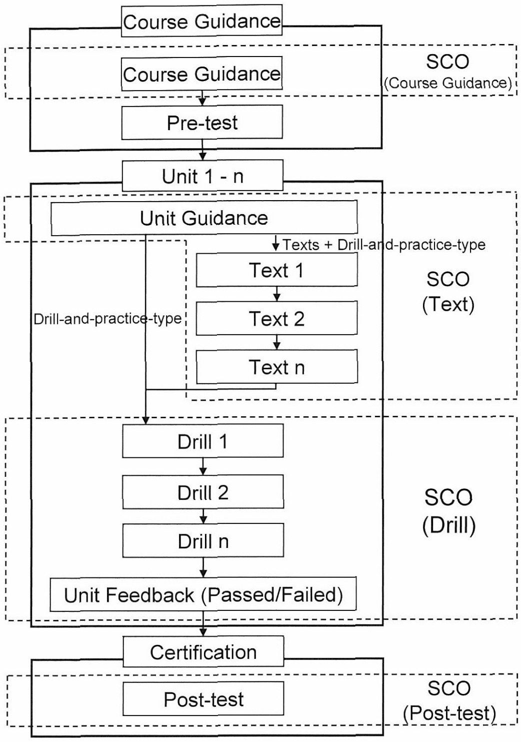
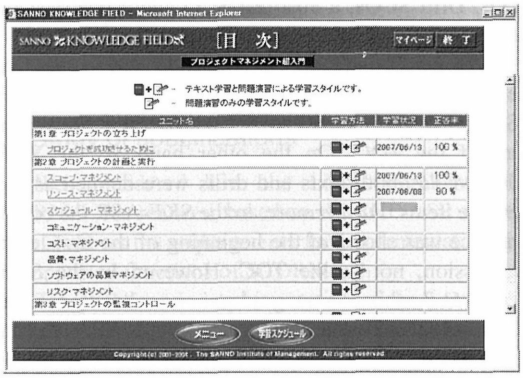


Figure 11. Scope of the SCOs in SKF-SCORM 1.2.



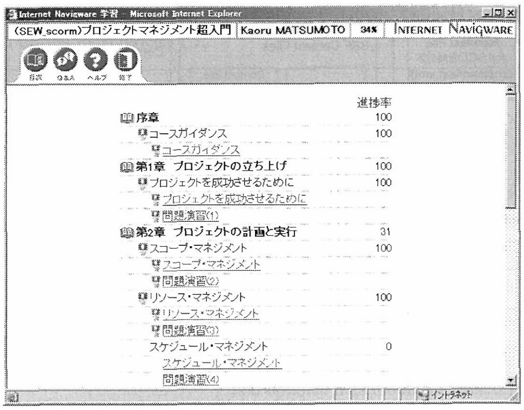
プロジェクトマネジメント超入門

■+✎ - テキスト学習と問題演習による学習スタイルです。
✎ - 問題演習のみの学習スタイルです。

ユニット名	学習方法	学習状況	正答率
第1章 プロジェクトの立ち上げ			
プロジェクトを成功させるために	■+✎	2007/06/13	100 %
第2章 プロジェクトの計画と実行			
スコープ・マネジメント	■+✎	2007/06/13	100 %
リソース・マネジメント	■+✎	2007/06/08	90 %
スケジュール・マネジメント	■+✎	▨	
コミュニケーション・マネジメント	■+✎		
コスト・マネジメント	■+✎		
品質・マネジメント	■+✎		
ソフトウェアの品質マネジメント	■+✎		
リスク・マネジメント	■+✎		
第3章 プロジェクトの監視コントロール			

Copyright(c) 2001-2004. The SANNO Institute of Management. All rights reserved

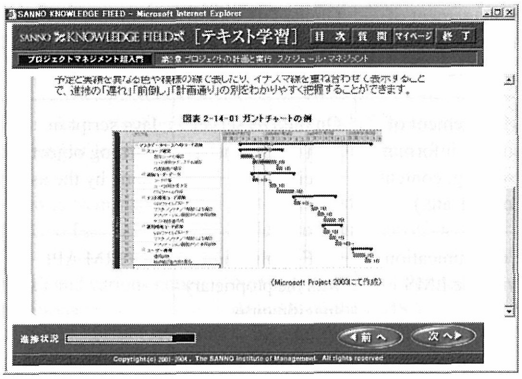
Figure 12. Example of TOC Screen for the SKF-ASP.



(SEW_scorm)プロジェクトマネジメント超入門 | Kaoru MATSUMOTO | 34%

	進捗率
序章	100
コースガイダンス	100
コースガイダンス	
第1章　プロジェクトの立ち上げ	100
プロジェクトを成功させるために	100
プロジェクトを成功させるために	
問題演習(1)	
第2章　プロジェクトの計画と実行	31
スコープ・マネジメント	100
スコープ・マネジメント	
問題演習(2)	
リソース・マネジメント	100
リソース・マネジメント	
問題演習(3)	
スケジュール・マネジメント	0
スケジュール・マネジメント	
問題演習(4)	

Figure 13. Example of TOC Screen for the SKF-SCORM 1.2 (by Fujitsu Internet Navigware7).



プロジェクトマネジメント超入門　第2章 プロジェクトの計画と実行 スケジュール・マネジメント

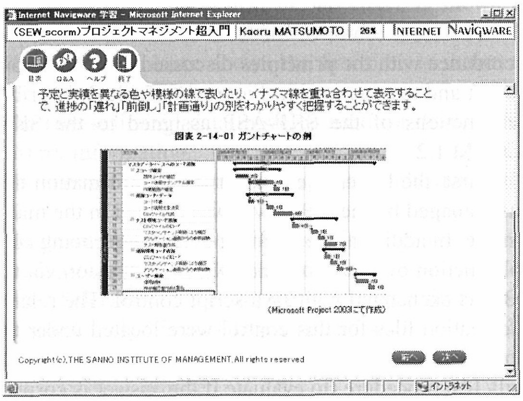
予定と実績を異なる色や模様の線で表したり、イナズマ線を重ね合わせて表示することで、進捗の「遅れ」「前倒し」「計画通り」の別をわかりやすく把握することができます。

図表 2-14-01 ガントチャートの例

(Microsoft Project 2003にて作成)

Copyright(c) 2001-2004. The SANNO Institute of Management. All rights reserved

Figure 14. Example of Text Screen for the SKF-ASP.



(SEW_scorm)プロジェクトマネジメント超入門 | Kaoru MATSUMOTO | 26%

予定と実績を異なる色や模様の線で表したり、イナズマ線を重ね合わせて表示することで、進捗の「遅れ」「前倒し」「計画通り」の別をわかりやすく把握することができます。

図表 2-14-01 ガントチャートの例

(Microsoft Project 2003にて作成)

Copyright(c),THE SANNO INSTITUTE OF MANAGEMENT.All rights reserved

Figure 15. Example of Text Screen for the SKF-SCORM 1.2 (by Fujitsu Internet Navigware7).

user interface of the SKF-SCORM1.2 was designed so that the learner can control the operation according to the SKF-ASP. Figures 14 and 15 are text screen views.

In the SCORM1.2 standard, no detailed specification about content navigation was defined. Therefore, every LMS had an original user interface. As a result, content designers must prepare navigation buttons on each of their contents to assure a certain movement if they want to use the screen changing function. For this reason, the "Next" and "Prev." buttons in the lower right corner on the SKF-ASP screen were embedded in the SKF-SCORM 1.2 contents and placed in the lower right corner. In addition, the functions to turn to the TOC, direct a question to the SKF, bookmark page and view the learner's progress that SCORM 1.2 does not support were removed from the screen of the SKF-SCORM 1.2.

More detailed information about the guidelines on the screen specifications is given in Chapter 4 of reference (11).

3.5 Problems and countermeasures

In this section, we discuss the problems of the SCORM 1.2 standard and countermeasures through the implementation of the SKF-SCORM 1.2 contents. The problems are about the optional data model, the control buttons, and the manifest file.

The first problem is the existence of optional data model elements. Nonconformity may exist between the LMSs and the contents, resulting from the differences in each LMS's support situation for optional items. If optional items are being used in the contents but the LMS has no support for them, the LMS may not operate correctly. For this reason, only mandatory data models in the SCORM content RTE were used in the SKF-SCORM 1.2 contents.

In the SCORM CAM, we decided to use the element ⟨adlcp:prerequisites⟩ which was defined as a "Precondition" as an ADL extension when necessary. This element was not mandatory in the SCORM 1.2 standard. Therefore the support situation on each LMS may differ. On the other hand, the contents have to use this element to customize the content flow according to specific requests from customers; for example, to use the contents from the beginning sequentially, or to try the Post-test after all UNITs are finished. For this reason, we decided to choose whether to use the element based on requests from the customer. We must rewrite the manifest file for each LMS because some LMSs do not support this function. However, it is not so difficult because the operation consists of only writing or removing the line of the element definition in the manifest file.

The second problem is about the navigation control buttons. Some LMSs constantly display navigation buttons. Therefore, if the navigation buttons are displayed on the content itself, the learner may become confused because multiple buttons with the same meaning will be shown by the LMS and the content.

In addition, there is a similar problem about the content control. On the last page of the SCO, the "Next" button does not appear on the screen. This may confuse the learner. In most cases, the correct operation will be to push the TOC button to go back to the TOC, and choose the next SCO. However, the function of the button operation is referred to the LMS in the SCORM standard. The content cannot recognize the screen design by itself. Therefore, the learner must find his/her way from the screen that is shown by the LMS. However, for most learners, it is difficult to decide based on intuition. The problem is caused by the screen composition difference between the LMS vendors, and the lack of the SCORM standard to identify the screen situation from the content side. The only solution to this problem is to customize the individual content to each LMS; for example, to display the correct operation on the individual content to suggest doing so to the learner.

The third problem is the existence of LMSs which do not work as intended, regardless of the description in the manifest file. For example, the element ⟨adlcp:masteryscore⟩ to define the passing score for decision of passed/failed cannot be used directly, because each movement differs between the LMSs. Therefore, in the SKF-SCORM 1.2 contents, the description is written in the manifest file so that problems will not occur on each LMS, and the Java script command is added for the irregular LMSs.

These problems are caused by the ambiguity in the definitions of the standard and they cause a heavy burden on the content developer. Therefore improvement of the standard is needed.

4. Content Design for SCORM 2004

Here we outline the SCORM 2004 architecture and discuss how to apply it to the SKF-ASP. After that, we provide implementation methods of contents for the

Table 4. Requirements for Implementation and Supported Functions in SCORM 1.2 and SCORM 2004.

Requirements for Implementation	SCORM 1.2	SCORM2004
Divert the content files.	Supported	Supported (It is based on the W3C standard.)
Reproduce the learning content structure.	Supported	Supported (It is possible to express an equal.)
Manage the learning personal history.	Partly supported	Supported (All of the data models are defined as mandatory.)
Use navigation buttons.	Partly supported	Supported (It is possible to navigate between SCOs from the content side.)
Control the learning order.	Partly supported	Supported (Supported by the sequencing function.)
Reproduce the selection of the way of learning.	Not supported	Supported (Supported by the sequencing function.)
Allow movement between the UNITs.	Not supported	Supported (SCO can be called from another SCO.)
Email and various notice functions.	Not supported	Not supported (Not defined. It depends on the LMS.)

SKF based on SCORM 2004.

4.1 SCORM 2004 architecture

SCORM 2004 is the latest version of the SCORM standard. The SCORM 2004 4th Edition Version 1.1 was released on the ADL website[6].

In SCORM 2004, definitions were added and improvements were made according to the problems encountered in the SCORM 1.2 standard. Although the basic concept and design were not changed, there were many important changes for implementation of the SKF-ASP contents. The significant changes are as follows.

- Adding the sequencing function
- Adding the function to issue navigation commands by the SCOs
- Adding the function to select display/hide the LMS navigation buttons
- Changing all data model elements of the Content RTE to mandatory
- Defining the specifications clearly, and supporting multiple languages

4.2 Study of specifications

In order to implement the SKF-SCORM 2004 contents in a way that it is similar to the SKF-ASP system, we considered the requirements and the presently supported functions. Table 4 shows the requirements for implementation and the supported functions of SCORM 1.2 and SCORM 2004.

The most significant change is the support for the sequencing function. By using this, the function of "Selection of the Way of Learning" that was not supported on SCORM 1.2 is supported on an environment other than the SKF-ASP. In addition, the standard for content navigation was not defined for SCORM 1.2. Therefore each LMS provided users with an original interface. As a result, learners had many problems with screen control. However, the SCORM 2004 standard supports content navigation, and SCORM 2004 can control the SCO from the content side. Furthermore, the function to display/not display the control button by the LMS is supported.

Hence, on the SKF-SCORM version, the following specifications were added to the SCORM 1.2 version.

- The Pre-test is shown after the Course Guidance.
- Each UNIT type is set based on the Pre-test results (Texts+Drills/Drills).
- On the last page, the SCO calls the next SCO.
- The navigation buttons by the LMS are not shown and they disabled while the contents are displayed.
- The learning order is controlled.

4.3 Implementation of content

In the SKF-SCORM 2004, we developed a specialized data conversion tool, and it makes the manifest file automatically from the SKF-ASP data composition table (MS-Excel file) as shown in Figure 16. In addition, the control frame inside the content is replaced by changing

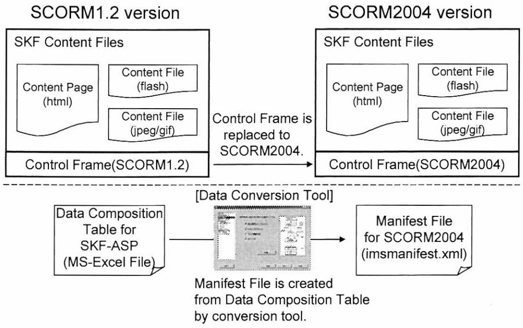

SCORM1.2 version
SCORM2004 version
SKF Content Files
Content Page (html)
Content File (flash)
Content File (jpeg/gif)
Control Frame(SCORM1.2)
Control Frame is replaced to SCORM2004.
Control Frame(SCORM2004)
[Data Conversion Tool]
Data Composition Table for SKF-ASP (MS-Excel File)
Manifest File for SCORM2004 (imsmanifest.xml)
Manifest File is created from Data Composition Table by conversion tool.

Figure 16. Mechanism of Content Conversion.

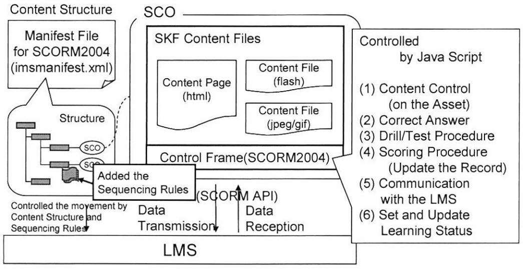

Content Structure
SCO
Manifest File for SCORM2004 (imsmanifest.xml)
Structure
SCO
SCO
Added the Sequencing Rules
Controlled the movement by Content Structure and Sequencing Rules
SKF Content Files
Content Page (html)
Content File (flash)
Content File (jpeg/gif)
Control Frame(SCORM2004)
(SCORM API)
Data Transmission
Data Reception
LMS
Controlled by Java Script
(1) Content Control (on the Asset)
(2) Correct Answer
(3) Drill/Test Procedure
(4) Scoring Procedure (Update the Record)
(5) Communication with the LMS
(6) Set and Update Learning Status

Figure 17. Content Architecture of SKF-SCORM 2004.

from the SCORM 1.2 standard to the SCORM 2004 standard, and other content files are used as is. We note that the SKF-SCORM 2004 in this paper was developed on the SCORM 2004 2nd Edition which was the latest version at the time.

Figure 17 shows the content architecture of the SKF-SCORM 2004. The basic mechanism is not changed much from the SKF-SCORM 1.2. The significant changes are: adding sequencing rules in the manifest file to describe the content structure; setting the function of learning goal by Java script; and adding the logical script to update the learning status.

The scope of the SCOs in SKF-SCORM2004 is designed as shown in Figure 18. The content files of the SKF-SCORM 2004 include a Course Guidance SCO, a Pre-test SCO, Text SCOs, Drills SCOs, a Post-test SCO and a manifest file for file binding, definition of the TOC and the sequencing rules.

In order to implement the Pre-test, we used the Shared Global Objectives that was defined additionally as a new function of the SCORM 2004 data model. The value of the objectives can be set and referred to from all SCOs, and that is important in making adaptive content. In the SKF-SCORM 2004, the Shared Global Objectives are used as shown in Figure 19.

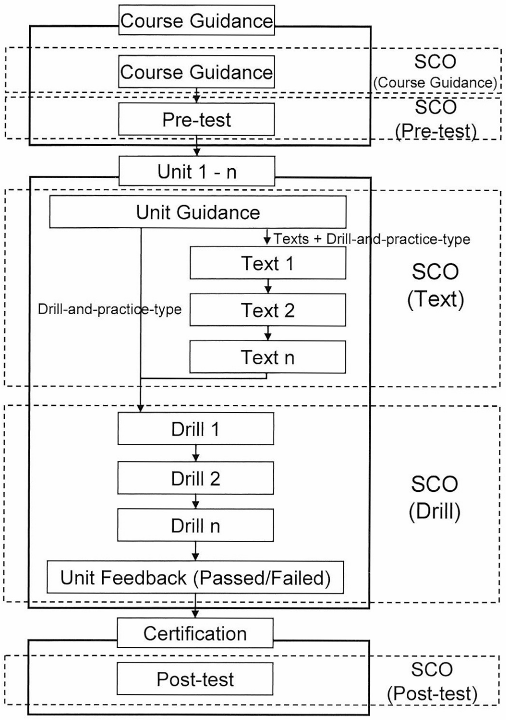


Figure 18. Scope of the SCOs in SKF-SCORM 2004.

At first, the learner takes the Pre-test. After that the corresponding learner status value (OBJ-x; defined as the Shared Global Objectives in Figure 19) is set as passed/failed based on the test result. Then learning is started and when the text SCO is called by the LMS, if the OBJ-x is passed, the text SCO will be skipped by the LMS control. The control is described as sequencing rules in the manifest file.

As a result, the values of Shared Global Objectives are set based on the result of the Pre-test, and either "Texts+Drill-and-practice-type" or "Drill-and-practice-type" for each unit will be chosen automatically.

Other sequencing rules are set as follows.

- If the Drills are passed, the UNIT will be set as completed (controlled by Rollup rule).
- If the UNIT is completed, the learner may proceed to the next UNIT (controlled by Stop Forward Traversal rule).

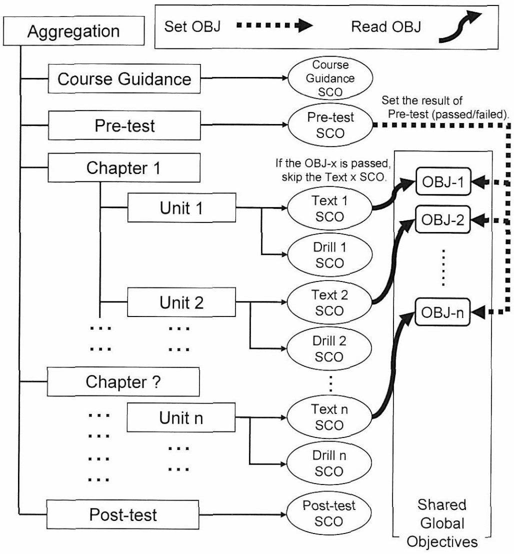


Figure 19. Content Aggregation and Learning Status.

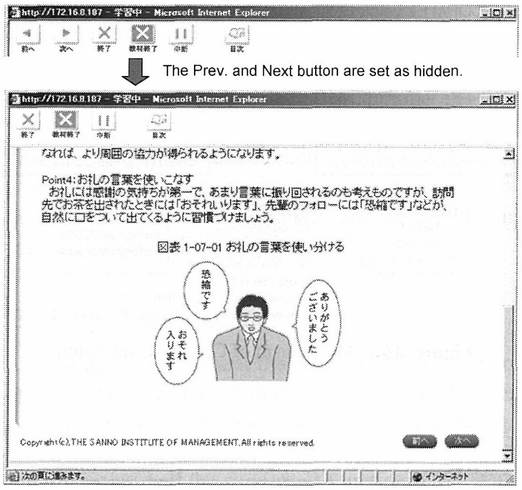
- When the UNIT is completed, the learner will be able to try the Post-test (controlled by Rollup rule).
- If the learner passes the Post-test, the course will be completed.

More detailed information about the Shared Global Objectives and the movement of sequencing rules in SCORM2004 is given in Chapter 3 of reference (12).

4.4 Implementation of user interface

In the SKF-SCORM 2004, we improved the user interface of the SKF-SCORM 1.2 to provide a consistent operating environment.

First of all, the navigation buttons of "Prev." and "Next" on the text screen were provided by the content side, and the navigation buttons of the LMS were hidden according to the manifest file. As a result, learners do not become confused by having two buttons with the same meaning provided by the LMS and by the content. Figure 20 is an example text screen.


http://172.16.8.187 - 学習中 - Microsoft Internet Explorer
前へ
次へ
終了
教材終了
中断
目次
The Prev. and Next button are set as hidden.
http://172.16.8.187 - 学習中 - Microsoft Internet Explorer
終了
教材終了
中断
目次
恐縮です
ありがとうございました
おそれ入ります
前へ
次へ
次の頁に進みます。
インターネット

なれば、より周囲の協力が得られるようになります。

Point4:お礼の言葉を使いこなす
　お礼には感謝の気持ちが第一で、あまり言葉に振り回されるのも考えものですが、訪問先でお茶を出されたときには「おそれいります」、先輩のフォローには「恐縮です」などが、自然に口をついて出てくるように習慣づけましょう。

図表 1-07-01 お礼の言葉を使い分ける

Copyright(c),THE SANNO INSTITUTE OF MANAGEMENT. All rights reserved.

Figure 20. Example of Text Screen for SKF-SCORM2004 (by Opensource LMS v1.04).

Second, in the SCORM 1.2 standard, the learner has to go back manually to the TOC from the last page of the Text SCO. In SCORM 2004, the button to move to the next SCO was implemented, and the SCO can call a navigation command to move to the next SCO. As a result, the learner can browse the content seamlessly without being forced to go back to the TOC.

More detailed information about the content navigation on SCORM 2004 is given in Chapter 4 of reference (12).

4.5 Problems and countermeasures

In the SKF-SCORM 2004, we were able to realize the functions of "Selection of the Way of Learning" and control of the learning object, which were implemented on the SKF-ASP. This was accomplished by the support of the sequencing function. In addition, the user interface was improved by the support of an additional learning navigation function and customization. As a result, in the SKF-SCORM 2004, we could virtually provide the same learning system and user interface as the SKF-ASP.

However, there are still problems. In the SCORM 2004 standard, although various kinds of learning content design became possible, deep understanding related to the standard in the stage of design and development of learning content is required. Especially, in the stage of setting the sequencing rules, we must consider the structure after understanding the system. Also, description of the manifest file to express the structure became complicated.

However, all of the SKF courses are designed and developed with the same structure, and learning resources are constructed independently for each UNIT. For this reason, unless a big change is made, we will be able to re-construct the content by re-using existent manifest files and file structures, and the changes will be minimized.

When we look at the SKF not as content but as a system, we are faced with other problems. For example, there are functions of automatic sending of email and some notices that were removed when the ASP-SCORM 1.2 was designed based on the SKF-ASP. These included functions of automatically sending emails to encourage the learner, to warn the learner to start the training, and to warn of the approaching deadline. They were implemented on only the SKF-ASP.

These problems are outside of the scope of the SCORM standards. However, if we broadly consider learning not only as learning content but also as a learning system, these problems will become more important.

5. Evaluation

Next, we turn to interoperability validation experiments done to confirm the interoperability condition between the SKF-SCORM 1.2 contents and the LMSs.

5.1 Interoperability validation experiments

The Standardization Promotion Committee in the E-Learning Consortium Japan is conducting a program for LMS-Contents certification for the SCORM 1.2 standard. In the program, conformance test tools are being used and LMSs or contents are being certified as SCORM 1.2-Conformant if they have no errors.

In addition, the experiments are providing real tests that check the certified contents on the certified LMSs to confirm the interoperability between certified products. We show the results of our interoperability validation experiment performed in 2005 and 2006. Table 5 shows the dates and objects of the experiment.

The experiment was conducted according to the following procedure.

(a) SCORM Assessors provided SCORM 1.2 certified contents to the Standardization Promotion Committee and the committee distributed them to LMS vendors.

(b) The LMS vendors tried the certified contents on their LMS, and confirmed the learning operations. They checked if any problems occurred.

(c) The LMS vendors reported the results of the confirmation to the Standardization Promotion Committee.

Table 5. Dates and Objects of Experiment.

Dates	June 24, 2005–July 8, 2005 , June 19, 2006–June 30, 2006
Objects	SCORM1.2 LMSs and contents certified by E-Learning Consortium Japan, LMSs: 10 products (in 2005)+4 products (in 2006), Contents: 5 products (in 2005)+5 products (in 2006)

5.2 Results

Table 6 shows the results related to the SKF. More detailed information is available on the website of the E-Learning Consortium Japan (http://elc.or.jp).

The contents of SKF-SCORM 1.2 worked without errors on all of the LMSs tested. However, it was found to be necessary to change the file format of the manifest file on two LMSs. The reason for this problem was differences in file encoding forms. The manifest files of SCORM 1.2 have to be written in UTF-8, but the file encoding form of the SKF manifest file was UTF-8 with Byte Order Mark (BOM), so it was not accepted on two LMSs. (The two LMSs could accept the UTF-8 file without BOM.) This problem was outside the scope of the SCORM standard. This problem was solved easily. After that we made the SKF manifest file on UTF-8 without BOM.

5.3 Discussion

The SKF-SCORM 1.2 contents were confirmed to work correctly on all of the LMSs tried in the experiment. The reason for this is that we limited use of the data model to use only mandatory items.

However, we found that the experiment was not conducted because unsupported optional items for the LMS were being used on the content; this occurred for another company's contents. This problem occurred from the differences in the supporting level of optional items on each LMS. We cannot make sweeping statements about SCORM 1.2 support, because the SCORM 1.2 standard has three LMS compliance levels as LMS-

Table 6. Results of Interoperability Validation Experiment.

Certification ID: Course name	Results	Certification ID: Vendor name, LMS name
C04-0008: Business manner	Good	P03-0004: Hitachi Electronics Services, HIPLUS on Web v8
	Good	P03-0004: Hitachi Electronics Services, HIPLUS on Web v8
	Good (*1)	P03-0005: NEC Corporation, CultiivaII v1.6
	Good	P03-0008: NEC Corporation, Cultiiva Enterprise Powered by SumTotal v6
	Good	P03-0009: COMPAC LTD, AcademicWare WBT v2
	Good	P03-0011: NRI Learning Network, NetTutorII v4
	Good	P04-0001: Wilson Learning WorldWide, eARTH-LMS v1.0
	Good	P04-0002: Lecwell, MSTeLMS v2.0
	Good	P04-0003: Koo Inc., Challenge Learning System v2
	Good	P04-0004: Panasonic Corporation, Let's learning v1
	Good (*1)	P04-0005: Fujitsu Ltd., Internet Navigware Server Enterprise Edition v8
C05-0016: Project management basic	Good	P03-0009: COMPAC Ltd., AcademicWare WBT v3.0
	Good	P04-0002: Liztech, MSTeLMS v2.0
	Good	P06-0001: KiBAN, eLearning Manager v3
	Good	P06-0001: KiBAN, eLearning Manager v3
	Good	(unofficial): NTT resonant, eLC, NTT, OpensourceLMS/Learning Engine for SCORM 2004 LMS v1.0

*1: It is necessary to change the file format of the manifest file

RTE1 to RTE3, and the LMS users have to identify the differences between the levels and choose the contents that are correctly supported on their LMSs.

Hence, LMS developers should support all optional items, and content developers should make all contents using only mandatory items to assure they work correctly. As a fundamental solution, the transition to SCORM 2004 in which all items have been defined as mandatory would be the simplest and most reliable way.

6. Suggestions for Standardization

Today, many LMSs are supporting the SCORM 1.2 standard, and it has become a content form standard for WBT-type e-learning systems. Although there are a few limitations for the standard, the SKF contents could be realized the same as with the original SKF-ASP version. We think it will be used as a de facto standard for some time to come.

The SCORM 2004 standard has not been popularized extensively, however the numbers of LMS products supported on the SCORM 2004 are increasing in recent years. There are reports of LMSs being developed as open source programs or as systems applied to the standards in the JSiSE and other organizations as well(13-16).

Although the description of movements of content with complicated learning design has become possible in the SCORM 2004 standard, the mechanism has become complicated accordingly. Therefore, we have to understand the standard deeply to design and develop the contents. At the present, not many easy tools for development of the SCORM 2004 contents have been released. In the future, the demands for these contents design and development tools will increase. Essentially, standards should be made not for system developers, but for content developers. In the future, standards should be developed on the basis of "how to design the learning" rather than "how content designers and developers should apply it."

Furthermore, a general support function for learners is often required when the learning is viewed not only as learning content but also as an educational system. Possible uses have been widened greatly by changing from SCORM 1.2 to SCORM 2004. There have been some research studies on expansion of the SCORM standards to support learners(17, 18) and also for a standard to describe learning design(19–21).

The SKF system sends emails automatically to encourage or prompt and warn the learner to start or complete the training. However, the email function is implemented only on the SKF-ASP, and it could not be implemented on the SKF-SCORM 1.2 and 2004 because it depends on each LMS's specifications. If we can describe the procedure for the email function on the content side by standardization, we may be able to design a more effective learning environment. This is a challenge left for the future.

7. Conclusions

In this paper, we described our original drill-and-practice-type e-learning system SANNO KNOWLEDGE FIELD that was developed by the SANNO Institute of Management. Then we described the adaption and implementation of the SKF contents to the SCORM 1.2 and SCORM 2004 standards, and considered their results and problems.

Although the SKF-SCORM 1.2 contents have some limitations caused by the specifications, we successfully realized the contents in a similar way to the original SKF-ASP version. Furthermore, we confirmed that they worked without problems on each LMS product.

The learning mechanism and user interface similar to those of the SKF-ASP were re-constructed on SKF-SCORM 2004 contents by using additional new functions of learning sequencing and navigation. These SKF-SCORM 2004 contents have not been released as commercial products yet. However we will be discussing this after analysis of market trends, and verification of the conformance with some LMSs.

At present, some products based on the SCORM 2004 standard have been released. For expansion of SCORM 2004, we think content development tools will become more important after the development of LMSs based on it.

In this paper, we described the interoperability validation experiment in the E-Learning Consortium Japan for confirmation of movement of SKF-SCORM contents. Nevertheless, not only the experiment on SCORM 1.2 but also the experiment on SCORM 2004 will be necessary, and we also think the experiments from another viewpoint such as user interface aptness in actual use will be necessary. We would like to work on this in our future studies.

Acknowledgements

We would like to express our sincere thanks to all those who have contributed to system design and development in the SKF project.

References

(1) Koga, A.: "Practical Use in Human Resource Development: e-Learning", in Pedagogy Using Digital Resources, Chapter 5, University of Tokyo Press (2010) (in Japanese).

(2) Fallon, C. and Brown, S.: e-Learning Standards, St. Lucie Press, Boca Raton, FL (2003).

(3) Advanced Distributed Learning Initiative: Sharable Content Object Reference Model (SCORM) Version 1.2—The SCORM Overview, (2001).

(4) Advanced Distributed Learning Initiative: Sharable Content Object Reference Model (SCORM) Version 1.2—The SCORM Content Aggregation Model, (2001).

(5) Advanced Distributed Learning Initiative: Sharable Content Object Reference Model (SCORM) Version 1.2—The SCORM Run-Time Environment, (2001).

(6) Advanced Distributed Learning Initiative: Sharable Content Object Reference Model (SCORM) 2004 4th Edition Documentation Suite, (2009).

(7) Nakabayashi, K., Shimizu, Y. and Yamada, T.: "New Trends in Development and Practice of e-Learning Technology Standardization", Japanese Society for Artificial Intelligence, Vol. 20, No. 1, pp. 92–98 (2006) (in Japanese).

(8) Miyauchi, H. and Ohta, M.: "Practical Example of Development of SCORM 2004 Content from ASP-typed WBT System with Proprietary Specification", The 30th Annual Conference of JSiSE, pp. 7–8 (2005) (in Japanese).

(9) Morimasa, Y.: "Programmed Learning" Psychological of Learning Methods, pp. 128–157. Yuhikaku Publishing, Tokyo (1993) (in Japanese).

(10) Kogo, C.: "Design, Practice and Evaluation of C Programming Course by Web-based Personalized System of Instruction", Transaction of JSiSE, Vol. 20, No. 3, pp. 293–303 (2003) (in Japanese).

(11) Asia e-Learning Network Working Group 2: SCORM ASSESSOR Workshop Guide, e-Learning Consortium Japan (2004).

(12) Miyauchi, H., Ohta, M. and Nakabayashi, K.: SCORM 2004 Content Development Guide, e-Learning Consortium Japan (2006).

(13) Nakabayashi, K., Nakamura, A., Kosaka, Y. et al.: "Implementation and Technical Issues of a SCORM 2004 Execution Engine", JSiSE Research Report, Vol. 20, No. 1, pp. 23–30 (2005) (in Japanese).

(14) Nakabayashi, K., Nakamura, A., Hoshide, T. et al.: "Open Source LMS Sui^2: Architecture and Practical Use Case", IPSJ-SIG-CMS1, pp. 63–70 (2005) (in Japanese).

(15) Nakabayashi, K., Hosokawa, M., Kawakami, T. et al.: "Design and Implementation of Mobile Learning Environment as an Extension of SCORM 2004 Specification", Transaction of IEICE, Vol. J91-D, No. 2, pp. 143–151 (2008) (in Japanese).

(16) Hoshino, T. and Toida, M.: "Development of opensource LMS; Attain 3 which focused on GUI design and SCORM 2004 conformant", The 32nd Annual Conference of JSiSE, pp. 310–311 (2007) (in Japanese).

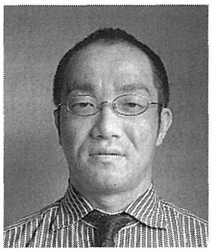
(17) Morimoto, Y., Ueno, M., Kikukawa, I. et al.: "SALMS: SCORM-compliant Adaptive LMS", Proc. of E-Learn 2007, pp. 2903–2912 (2007).

(18) Morimoto, Y., Maomi, U., Kikukawa, I. et al.: "Formal Descriptive Method of Behavior of an E-Learning System in Cooperation with Educational Meta-models", SIG-ALST-A702, pp. 13–19 (2007) (in Japanese).

(19) MS Global Learning Consortium: IMS Learning Design Information Model, (2003).

(20) IMS Global Learning Consortium: IMS Learning Design XML Binding, (2003).

(21) IMS Global Learning Consortium: IMS Learning Design Best Practice and Implementation Guide, (2003).



Kaoru Matsumoto received his B.Sc., M.Sc. and Ph.D. degrees from the Tokyo University of Agriculture and Technology in 1997, 1999 and 2004 respectively. From 2002 to 2004, he was a researcher at HRM research center in the SANNO Institute of Management. In 2005, he was a researcher and lecturer at SANNO University. Since 2006, he has been in Self Learning Systems Development Dept., at the SANNO Institute of Management. His current work is designing and developing learning contents for business person. He is a SCORM assessor in E-Learning Consortium Japan and a member of JSiSE, IPSJ, IEICE and JSAI.

Hiroshi Miyauchi has been working in the SANNO Institute of Management from 1995. His current work is designing and developing of learning program for company. At present, he is in Self Learning Systems Development Dept., at the SANNO Institute of Management. He is a vice-chairperson of Standardization Promotion Committee, a SCORM assessor in E-Learning Consortium Japan and a member of JSiSE.

Akihiko Koga has been working in the SANNO Institute of Management from 1988. He was a manager of the e-Learning Development Center, Self-Learning Systems Development Dept., at the SANNO Institute of Management. At present, he is an associate professor in School of Information-Oriented Management, SANNO University and a part-time lecturer in Graduate School of Instructional Systems, Kumamoto University. He is a member of JSiSE, LGESJ and CIEC.