International Journal of Computer and Information Technology (2277 – 0764)
Volume 01– Issue 01, September 2012

A Domain-Based Learning Object Search Engine to Support Self-Regulated Learning

Ali Alharbi, Frans Henskens, Michael Hannaford
School of Electrical Engineering and Computer Science
The University of Newcastle, Australia
Ali.H.Alharbi@uon.edu.au, Frans.Henskens@newcastle.edu.au, Michael.Hannaford@newcastle.edu.au

Abstract— **The number of learning resources available on the web has increased dramatically. However, it is a difficult task for students to locate the learning materials that are appropriate to their requirements and needs. This study proposes a custom search engine to help students find learning objects related to computer science topics. The custom search engine provides a unified interface to search different learning material repositories and filter the result using criteria such as the type of learning material and the topic under which these learning materials are classified. The custom search engine implements a term suggestion function to make it easy for students to choose relevant keywords for their search. The term suggestion function is based on the IEEE/ACM Computing Curriculum guidelines and the ACM Computing Classification System. An empirical evaluation of the proposed custom search engine with computer science students reveals that the system is highly effective in retrieving learning objects related to topics about programming languages. The students' responses to the evaluation questionnaire indicate that they consider the custom search engine easy to use and useful for finding computer science learning objects.**

Keywords-Learning objects; computer science education; self-regulated learning; microdata; rich snippets; custom search engine.

I. INTRODUCTION

Self-regulated learning is receiving increasing attention in educational research, especially in higher education, because of its importance to academic success and lifelong learning [1, 2]. In this paradigm, learners are responsible for regulating and controlling their own learning process using different learning strategies. Finding relevant learning material is essential to help students be self-regulated learners. Many learning materials are available on the web that can be used to enhance teaching and learning processes. Learning resources with pedagogical objectives that are intended for use and reuse in different learning contexts are typically called learning objects [3]. These learning objects are normally stored in online repositories to make them easy to find. Finding appropriate learning objects in these repositories, however, is a challenging task [4]. For these learning objects to be reused, they must first be located.

The main methods used to search for learning objects include using general-purpose web search engines or the search engine provided by the learning object repository. Most learners use general-purpose search engines such as Google to find learning materials related to their studies [5], and few learners consider investigating learning object repositories. In both cases, it is not easy, especially in large repositories, to find learning objects related to a specific topic that are compatible with the students' and teachers' preferences and pedagogical requirements [6]. There are many possible reasons behind this challenge. First, choosing the best keywords for the search query is not a straightforward task, and users tend to have difficulty choosing the suitable keywords [7]. Users tend to write short queries, use few keywords, and examine few of the results returned by the search engine [8]. Additionally, users tend to have difficulty evaluating the results of the search query to determine which to choose [9]. Second, general-purpose search engines search the entire web, which means a large dataset is returned each time a user performs a query; and many entries in this dataset may not be learning objects at all. Third, the information retrieved and shown to the user as a list of search results does not help the learner choose the suitable learning objects from the large dataset retrieved by the search engine.

The idea of general-purpose search engines must be revisited. Several search assistants exist to help users make travel plans or do online shopping. Helping users find relevant learning material is an area that needs more attention. The importance of such support becomes clear in self-regulated learning.

This paper argues for designing domain-specific learning object search engines instead of general-purpose ones. These domain-specific search engines should reflect advancements in educational research achieved in the domain of interest. Additionally, these learning object search engines should provide integrated services to help users search through many learning object repositories and the web. This paper therefore presents a custom search engine to help students search for computer science learning objects. The custom search engine uses a proposed metadata application profile for computer science learning objects as a consistent scheme to index and describe such objects. To demonstrate the value of the proposed approach, a prototype of a computer science learning object repository has been developed. The repository uses the new application profile and custom search engine to help students find suitable learning objects in the repository and in other repositories.

II. Problem Statement

Learning objects are stored in different online repositories, making it difficult for students and teachers to locate them easily and within a reasonable time frame. To support the discovery of learning objects, descriptive metadata must be created and linked to the learning object. These metadata follow specific standards so that learning systems can follow a consistent approach for populating, searching, importing, and exporting learning objects [4]. Several metadata standards have been developed to provide consistent schemes for learning object publishing and searching. Learning object metadata standards support interoperability between learning systems and discovering and selecting learning objects by humans [10]. Finding learning objects related to a specific topic is not an easy task because of the vague vocabulary used in metadata fields [11]. Metadata standards are general-purpose schemes that do not consider the contextual meaning of the learning object in a specific discipline or subject [12]. Previous studies have indicated that labels and keywords adapted from metadata standards are overwhelming and confusing for users when used to search for learning objects in a repository [11]. The metadata standards used to describe learning objects also focus on the technical aspect of the learning object and neglect the important pedagogical information related to the actual use of the learning object in context [13].

Learning object repositories do not apply a consistent classification system to organise the objects. An exploratory study of an online digital repository reveals that learning object classification does not reflect the actual areas of computer science courses [14]. Searching for learning objects is also not efficient in many such repositories. Most learning object repositories depend on special search services in each repository to search for the objects. Usability studies of these search services reveal that they are not easy to use [11] and that the information retrieved and shown to the user as descriptions for result entries does not help the learner choose suitable learning objects from the large dataset retrieved by the search services.

III. Related Work

This section reviews approaches proposed in the literature that can be used to support finding and retrieving learning material in digital libraries and the web.

A. State-of-the-Art Learning Object Metadata

To support discovering and retrieving learning objects, metadata must be available to describe and index learning objects in online repositories. Researchers have developed many standards to support learning object metadata. The IEEE LOM standard is a metadata scheme developed by the IEEE Learning Technology Standards Committee to index and describe learning objects [15]. The IEEE LOM, which has dominated the field of learning object metadata, comprises elements grouped into the following categories:

1. The *General* category contains information that generally describes the learning object, including the title and language in which the learning object is written.
2. *Life cycle* category group information is related to the history of the learning object and its evolution.
3. *Meta-metadata* category contains information related to the metadata instance.
4. *Technical* category contains information related to the learning object format.
5. *Educational* category describes the pedagogical characteristics of the learning object.
6. *Rights* category contains information related to licensing and copyrights.
7. *Relation* category describes the relationships between the learning objects and other learning objects, if they exist.
8. *Annotation* category contains information about the comments users provided after using the learning object.
9. *Classification* category describes where the learning objects fits in a specific classification system.

Search interfaces that are designed on the basis of metadata standards use terms and vocabularies that seem confusing and overwhelming to the search service users [11]. Each learning community has its own requirements for the learning material; metadata must therefore adapt to the community requirements, which results in what are called application profiles. An application profile is an adaptation of the original metadata standard to meet the specific requirements and needs of the target community [16].

B. Application Profiles and Learning Object Search Tools

Regarding the application profiles, [17] proposed an application profile to make searching for and retrieving science learning resources more efficient. Another application profile to describe agricultural learning resources is reported in [18]. Neither application profile has been empirically evaluated.

On the other hand, researchers have developed tools to support searching for learning materials. In [19], the MIT Libraries Catalogue was tested to identify confusing features that need modification and those that are useful for researchers. The evaluators found that the searchers had trouble using the search interface. Many users did not understand the structure of the search interface and had trouble finding the option to filter the results according to learning material format. The searchers also did not understand much of the terminology used in the search interface.

Dumais, Cutrell, and Chen [20] reviewed and evaluated several interfaces for displaying search engine results. The evaluation aimed to compare the traditional list view of results with results grouped into categories. The study found that participants could finish their search task in the categories-based approach compared to the traditional list view approach. The study also found that presenting short inline summaries for the search results is more effective than showing a summary when the user moves the mouse over an item in the result set.

Another method to improve searching for learning objects is reported in [21]. The authors applied text extraction techniques and machine learning to extract learning object metadata from the web, using the metadata to enhance search engine accuracy for learning resources. The study results showed that metadata extraction from actual web page content is a feasible approach for improving search queries, but it was still time consuming and expensive. The applicability of this approach was limited to text-based learning material; it would be difficult to extract metadata from animations and interactive simulations.

Mercury [22] is a metadata harvesting tool that is used to retrieve metadata from several external repositories and combine them to provide unified search results. Mercury supports different metadata standards, uses open-source technologies, and provides multiple search services. There is no empirical evaluation available to support the effectiveness of this tool in searching for learning objects, especially those related to computer science education.

Most search tools developed to support searching for learning materials are general-purpose search interfaces that are not adapted to community, users, or context requirements in which learning objects might be used.

C. Microdata and Rich Snippets

Microdata [23] is a standards-based approach to describe a specific type of web content, including persons, products, reviews, or events. Microdata is promising because it is simple and builds on existing web technologies that are already adopted by many websites and supported by major search engines [24]. Microdata uses the current web page mark-up and extends it with additional information to make it more machine readable. Microdata contains one or more key-value pairs, representing a property and value, and the syntax comprises HTML attributes. In contrast, learning object metadata are represented using metadata standards, which are mostly stored as XML-based files. With such an approach, it is difficult to discover learning objects that are stored in repositories because of the lack of rich information that can provide users with sufficient information to locate the relevant learning objet.

Rich snippets, on the other hand, are presentations of the search results that search engines generate automatically from structured metadata (Microdata, Microformat, or RDFs) stored in web pages [25]. Many search engines now generate rich snippets to make it easy for users to determine whether a result is relevant (Fig. 1).

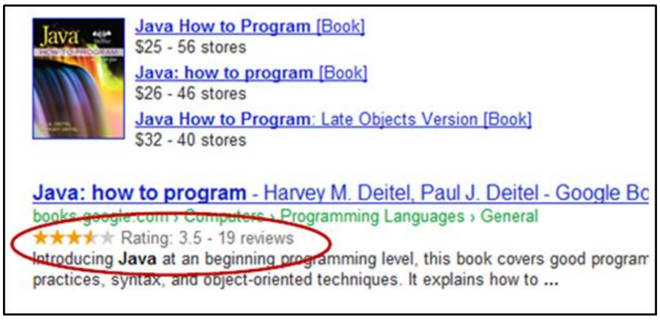

Figure 1. Example of a rich snippet for book search results

Developing e-learning standards seems to be isolated from the advances already achieved in web technology. Creating rich snippets is not directly possible from the existing learning object metadata because no search engines recognise the format. Most learning object repositories depend on special search services in each repository.

IV. Proposed Learning Object Standard-based Search Engine

This section describes a standard-based approach to enhance the effectiveness of searching for learning objects related to computer science education. This custom search engine helps students find learning objects related to their preferences, which leads them to become more self-regulated learners. First, a description of a proposed computer science learning object metadata profile is presented, followed by a description of the custom learning object search engine designed to help students locate computer science learning objects.

A. Computer Science Learning Object Metadata Application Profile

To solve the problem of the difficulty of finding and retrieving learning objects related to a topic in computer science courses, this paper proposes a Computer Science Learning Object Metadata (CS LOM) application profile as an extension to the original IEEE LOM application profile. The proposed application profile can be used to classify learning objects related to computer science education, which can help improve finding relevant learning objects for computer science topics. The profile also provides a consistent metadata scheme to facilitate searching for and retrieving computer science learning objects from large learning object repositories and the web. To design the application profile, the guidelines for designing application profiles [26] are applied.

According to the guidelines for creating application profiles [26], the first step in designing application profiles is to define a curriculum classification system that represents the community of practice and can extend the original IEEE LOM scheme. The ACM Computing Classification System (ACM CCS) [27] is a classification system that serves as the best starting point for classifying computer science learning objects. ACM CCS uses a hierarchical tree structure to organise research articles by subjects into 11 categories. In addition to the ACM CCS, the IEEE/ACM Computing Curriculum guidelines [28] are used as a curriculum classification system to provide guidelines for generating a controlled vocabulary for the keywords and terms in the application profile.

The second step is to determine the element in the original IEEE LOM hierarchy that can host the proposed classification system. The last category in IEEE LOM, *Classification*, is the relevant element to serve as the extension point for the original IEEE LOM scheme. Following the IEEE LOM specification [15], the *Classification* element is used to describe the learning object in terms of other classification schemes (Fig. 2). This element is thus the most suitable official extension to the original LOM scheme that does not compromise the metadata structure.

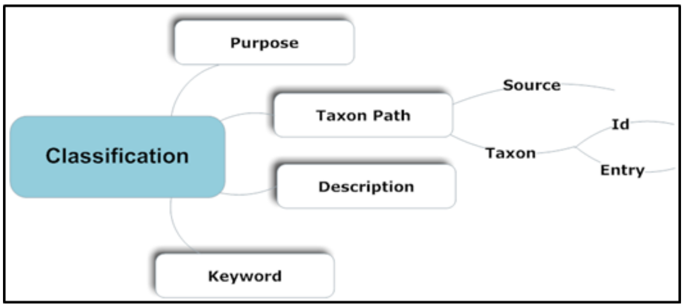

Figure 2. An IEEE LOM classification element

As the diagram shows, the IEEE LOM Classification category provides four sub-elements:

1) Purpose: This sub-element represents the reason behind the classification system. In our proposed application profile, we use the word "discipline" as the value for this sub-element.

2) Taxon Path: This sub-element represents a specific taxonomic path in the proposed classification system.

a) Source: This sub-element represents the name of the classification system. We use "IEEE/ACM Computing Curriculum" as the value for this sub-element.

b) Taxon: This sub-element represents a specific term within the taxonomy

c) Id: This sub-element represents a taxon identifier. For this sub-element, we can use the numbering provided by the ACM CCS or abbreviations used for computer science courses.

d) Entry: This sub-element represents a taxon label. For this sub-element, terms and vocabularies from the IEEE/ACM Computing Curriculum can be used.

3) Description: This sub-element represents a short description of the classification system

4) Keywords: This sub-element represents any keywords and phrases that can provide descriptive information about the classification system.

Table 1 and Fig. 3 show examples of using the IEEE LOM classification element to describe CS learning objects.

TABLE I. AN EXAMPLE OF AN IEEE LOM CLASSIFICATION ELEMENT FOR COMPUTER SCIENCE LEARNING OBJECTS

Purpose	Source	Taxon (Example)
Discipline	IEEE/ACM Computing Curriculum	*1. Programming languages* *1.1. Formal definitions and theory* *1.2. Language constructs and features* *1.2.1. Abstract data types* *1.2.2. Control structures* *1.2.3. Memory management* *1.3. Object-oriented programming* *1.3.1. Classes and objects* *1.3.2. Inheritance* *1.3.3. Polymorphism*

```
<classification>
<purpose> <langstring lang="en">Discipline</langstring> </purpose>
<taxonpath>
<source>IEEE/ACM Computing Curriculum</source>

<taxon>
<id>PL1</id>
<entry>
<langstring lang="en">Programming Languages</langstring>
</entry>
</taxon>

<taxon>
<id>PL1.2</id>
<entry>
<langstring lang="en">Languages Constructs and Features</langstring>
</entry>
</taxon>

<taxon>
<id>PL1.2.3</id>
<entry>
<langstring lang="en">Memory Management</langstring>
</entry>
</taxon>
</taxonpath>

<description>
<langstring lang="en-US">
principles of memory management in programming languages
</langstring>
</description>

<keywords>
<langstring lang="en">Heap,Stack,Static</langstring>
</keywords>
</classification>
```

Figure 3. An Example of an IEEE LOM Classification Element in XML Format

The CS LOM application profile introduces a controlled vocabulary as keywords that can be used to search for and describe the learning object in addition to the main taxonomy. Another type of metadata used in this profile is related to the type of the learning object. Such metadata can be used to help learners with different learning styles choose the learning objects that are appropriate for a specific context. Dynamic metadata can also be generated by reasoning about the learning objects using the usage information, including

- the average time spent using the learning object,
- the average rating given by students to the learning object, and

- the number of times the learning object is viewed.

For simplicity, this study uses and evaluates only the part of the application profile related to *programming languages and paradigms* to assess their educational usefulness in supporting locating and classifying learning objects related to teaching and learning of *programming languages and paradigms*.

B. Custom Search Engine for Computer Science Learning Objects

This paper presents a custom search engine to improve searching for computer science learning objects in large repositories. The custom search engine uses the learning object metadata application profile proposed in this study (Section IV.A). This proposed custom search engine uses a new approach to generate learning object metadata based on the Microdata technology and to build on the existing e-learning standards to make it easy for search engines to find relevant learning. Microdata are generated from the existing metadata associated with the learning objects stored in learning object repositories. This process is achieved using a conversion engine based on XSLT technology to convert metadata from the IEEE LOM schema into the corresponding Microdata format. Only a metadata set relevant to computer science students is used.

The custom search engine contains the following main components (Fig. 4):

1) Search Engine User Interface

The user interface in the custom search engine has four main parts (Fig. 5). The first part allows the learner to choose the learning object category in the computer science learning object taxonomy to help refine the search query according to a specific topic. This part is based on the CS LOM application profile proposed in Section IV.A. The second part selects the type(s) of learning objects (e.g., interactive animation, picture, or self-assessment).

The third part refines the search using programming language(s). The fourth part allows students to enter the keywords that best describe the learning object and uses an auto-complete function to minimise errors in entering keywords. Terms are suggested to the user based on the selected topics in the computer science learning object taxonomy and the selected programming language(s).

The user interface also contains an option to filter the results according to the average learning object ratings. This feature works only for learning object repositories that support user evaluations for learning objects.

The result of the query performed using the custom search engine is presented to the user as a list, and each entry in this list describes a learning object as a rich metadata snippet.

2) Learning Object Metadata Harvester

Metadata harvesting is a technique used to make local copies of metadata available in external data sources [29]. The most well-known protocol for metadata harvesting is the Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH), which enables discovering and collecting metadata in distributed archives [30]. OAI-PMH transmits data in XML format over an HTTP protocol. In this proposed custom search engine, the learning object metadata harvester extracts metadata related to computer science learning objects from external learning object repositories and sends them to the metadata filtering in XML format. The metadata filtering components are responsible for filtering the results returned from the harvester to exclude any learning object metadata that are not related to computer science topics.

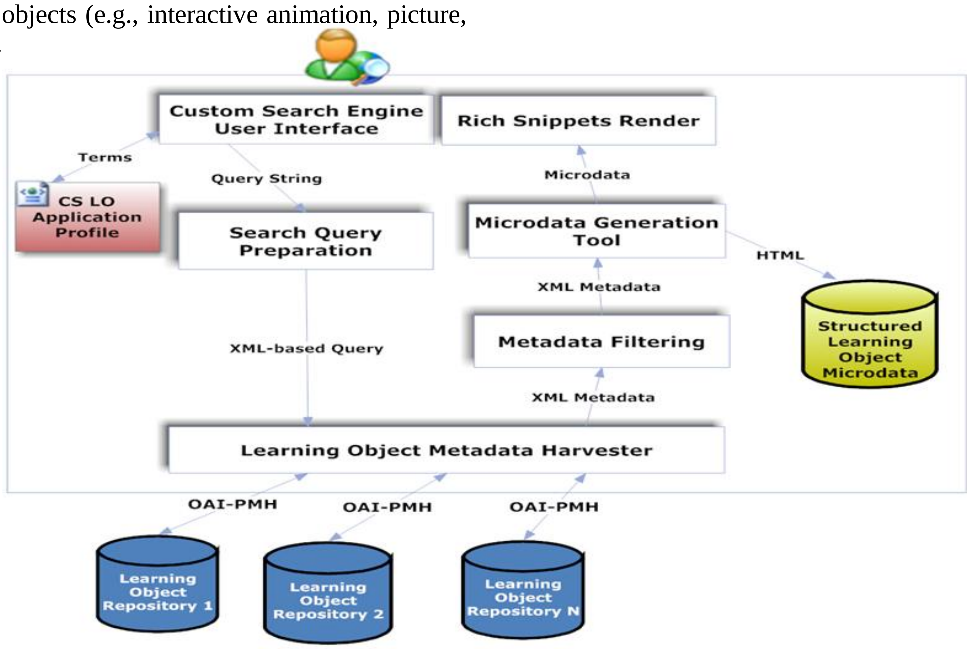

Figure 4. Structure of the Custom Computer Science Learning Object Search Engine

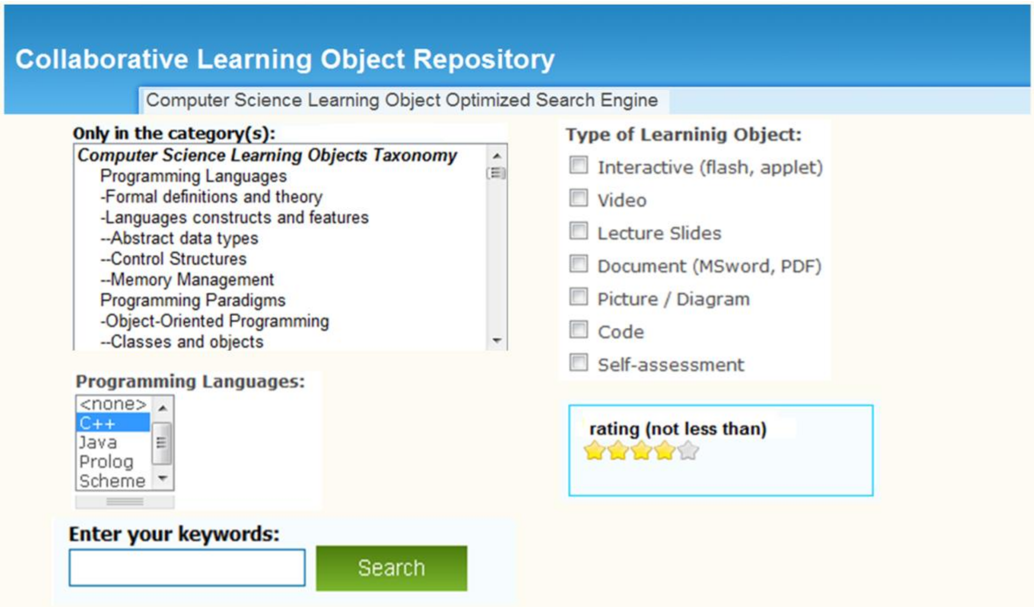

Figure 5. The user interface of the Custom Computer Science Learning Object Search Engine

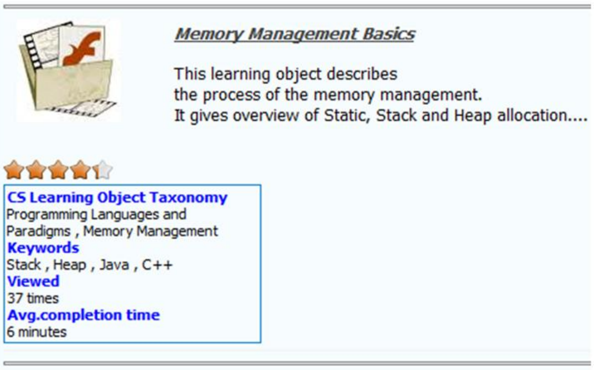

Figure 6. Learning object query result as rich metadata snippets

3) Learning Object Microdata Generation Tool

This component receives the raw metadata in XML format from the harvester after filtering. The learning object Microdata generation tool converts these metadata into structured Microdata records in HTML format. The resulting Microdata can be stored in a central repository so that other search engines can easily recognise and index them.

4) Rich Snippets Render

This component is responsible for interpreting the learning object Microdata and rendering them as rich metadata snippets in the query result list (Fig. 6). The rich snippets render receives the Microdata from the learning object microdata generation tool. The rich metadata snippets contain different information to help the learner choose relevant learning objects and filter the result more easily. This information includes the learning object's classification in the computer science learning object taxonomy, average ratings, a thumbnail, and keywords.

V. Evaluating the Computer Science Learning Object Repository

To demonstrate the effectiveness of the proposed method in searching for and presenting computer science learning object search results, a prototype of a computer science learning object repository has been developed. The repository uses the application profile (Section IV.A) and the custom search engine for computer science learning objects (Section IV.B) to help students find suitable learning objects in both the prototype repository and in certain external repositories. At this stage, the custom search engine can search in different learning object repositories, including MERLOT [31], ARIADNE [32], and Connexions [33].

The learning object repository is developed based on Drupal Learning Management System [34]. Drupal is an open source content management system written in PHP. Drupal provides features common to any learning management system such as content creation, publishing and user administration. Moreover, Drupal is customizable and flexible and it provides the ability for developers to integrate custom modules into the core system to extend its functionality. In this study, a number of modules have been created to provide the functionality required for the custom search engine.

A. Research Questions

(1) What is the accuracy of the proposed custom search engine in searching for learning objects related to computer science topics?

(2) Do students perceive the custom search engine as easy to use for finding computer science learning objects?
(3) Do students consider the custom search engine useful in supporting self-regulated learning?
(4) How do students perceive the rich metadata snippets used to describe learning object search results?

B. Participants

The participants in this study are students enrolled in the *Programming Languages and Paradigms* course at the University of Newcastle, Australia, in the second semester of 2012. The course covers the theory behind designing and implementing programming languages, recognised as an integral part of any computer science or software engineering degree (IEEE/ACM, 2005). The course covers different topics related to programming languages, including memory management, inheritance, and polymorphism. The participants use the learning object repository as a new approach introduced to improve course teaching and learning.

C. Data Collection Techniques

To evaluate the effectiveness of the proposed approach, data are collected using three main resources: query logs, questionnaires on students' perceptions, and students' qualitative comments.

1) Query Logs

The query logs store information about the search queries performed by the students inside the system. This information includes the keywords that students used in the search query and the items that they selected from the drop-down menus. The results retrieved by each query are also stored for further analysis. The query logs are used to study the custom search engine performance. Precision and recall are the two main metrics used to evaluate information retrieval system performance [35, 36]:

- Precision represents the fraction of retrieved elements from the entire set that are relevant to the search query.
- Recall represents the fraction of the relevant elements in the entire set that are retrieved.

	Relevant	Not relevant	Precision	Recall
Retrieved	tp	fp	tp/(tp+fp)	tp/(tp+fn)
Not retrieved	fn	tn		

Precision estimates the percentage of relevant elements in the retrieved dataset, whereas recall estimates the percentage of relevant elements in the repository that have been successfully retrieved. Precision and recall range from zero to one, where one denotes 100% precision or recall.

2) Questionnaire on Students' Perceptions

The students' perceptions of the search engine are measured using an online questionnaire completed after the students used the search engine in the computer science learning object repository. This questionnaire is part of the overall feedback questionnaire to evaluate the educational effectiveness of the entire learning object repository. The questionnaire uses a seven-point Likert scale, where one represents *strongly disagree* and seven represents *strongly agree.* The questionnaire contains two sections. The first section measures students' satisfaction with the custom search engine regarding its ease of use and usefulness in searching for learning objects. The second section measures students' perceptions of the usefulness of the rich metadata snippets used in the repository to describe the results of the learning object search queries.

The questionnaire is adapted from the Technology Acceptance Model (TAM), developed, tested, and revised by Davis [37], to predict user acceptance of a technology on the basis of its perceived ease of use and usefulness. TAM is a cost-effective and easy-to-use questionnaire for predicting users' acceptance of different systems [38]. Many researchers have extensively empirically tested TAM, proving its high reliability [39]. Many studies have used TAM to measure users' attitudes toward search engines [39]. Within TAM [37], perceived ease of use is defined as "the degree to which an individual believes that using a particular system would be free of physical and mental effort". Perceived usefulness is defined as "the degree to which an individual believes that using a particular system would enhance his or her job performance".

3) Qualitative Comments

Students can provide qualitative comments for any learning object in the repository, including the custom search engine. These comments are provided in an open-ended form at the bottom of the search engine page.

D. Method and Procedure

Using the learning object repository is optional for all students. Students are given accounts to log on to the repository and access the learning objects in the repository. The learning objects in the repository are categorised using the CS LOM application profile and are described using rich metadata snippets.

To evaluate the custom search engine, students are redirected to use the search engine and are provided with a full description of certain query tasks to perform. The tasks are simple search queries for certain learning objects related to the topics covered in the *Programming Languages and Paradigms* course. The search tasks are listed below:

1. Search for learning objects that describe memory management in programming languages.
2. Search for interactive animations (including videos) that describe inheritance in object-oriented programming.

3. Search for diagrams that explain multiple inheritances in C++.

All search queries are logged into the system for future analysis. To perform any search query, a student can select the topic from a drop-down menu. The search keywords can be entered into a specific text field that implements an auto-complete function using a controlled vocabulary obtained from the CS LOM application profile (Section IV.A).

After performing the initial search tasks, the search engine is available for students to use during the semester to search for learning objects in the repository. At the end of the semester, students are asked to provide feedback on the usefulness of the metadata rich snippets and the search engine used in the repository by answering questions in the final feedback questionnaire to evaluate the effectiveness of the learning object repository.

VI. Results

A. Performance of the Custom Search Engine for Computer Science Learning Objects

The first step to measure the custom search engine performance is to analyse search query logs for the search tasks performed by students when first using the custom search engine. Of the students, 17 completed the search tasks. Those students who did not complete the three tasks are discarded from the analysis of the query logs. There are a total of 51 search queries performed using the custom search engine and related to the search tasks given to the students. To calculate the precision and recall for each query task, the total number of learning objects relevant to the search task is predetermined by directly searching and browsing each repository before students perform any task. For each search task, the total number of learning objects retrieved by all search queries performed by all students to complete the search task is obtained from the query logs and subdivided according to relevancy to the search task. For the same learning task, the total number of relevant learning objects that are not successfully retrieved by search queries is calculated. **Error! Reference source not found.** presents the information obtained from the query logs and the average precision and recall calculated for each search task.

Table II shows that the custom search engine has high precision and recall levels. The levels of precision and recall for searching for memory management learning objects are 0.95 and 0.92, respectively. For this search task, approximately 95% of the learning objects retrieved by the search queries are relevant to the topic of memory management in programming languages. The custom search engines can also retrieve approximately 92% of the relevant learning objects in the repository. For all search tasks, the average precision is 0.91, and the average recall is 0.86. On average, 91% of the learning objects retrieved by the search engine are relevant to the search task. Using the custom search engine, the students could successfully retrieve approximately 86% of the relevant learning objects in the repositories.

TABLE II. PRECISION AND RECALL FOR THE CUSTOM SEARCH ENGINE FOR THE THREE TASKS

Search Task		Relevant	Not Relevant	Precision	Recall
Memory management learning objects	Retrieved	405	22	0.95	0.92
	Not retrieved	37			
Diagrams for inheritance in object-oriented programming	Retrieved	95	16	0.86	0.80
	Not retrieved	24			
Diagrams describing multiple inheritance	Retrieved	209	23	0.91	0.87
	Not retrieved	29			
Average	Retrieved	709	61	0.91	0.86

B. Ease of Use and Usefulness

Table III and IV present the results of the students' responses to the questionnaire items related to the ease of use and usefulness of the custom search engine for finding learning objects related to computer science topics. Table III shows that the mean response on the ease-of-use scale is 6.29 (out of 7), indicating that students strongly agree that the custom search engine is easy for them to use. For the usefulness scale in Table IV, the mean response is 5.98, indicating that students agree that the search engine is useful for them to find suitable learning objects.

TABLE III. STUDENTS' RESPONSES ON THE EASE OF USE OF THE CUSTOM SEARCH ENGINE

Item	Mean	SD
Learning to use the custom search engine is easy for me.	6.41	0.80
I find it easy to make the custom search engine do what I want it to do.	6.47	0.93
I find the custom search engine easy to use to look for learning objects.	6.18	0.89
I find the custom search engine flexible to interact with.	6.29	0.66
It is easy for me to become skilful at using the custom search engine to look for learning objects.	6.41	0.71
Overall, I find the custom search engine easy to use.	6.00	0.71
Overall (Ease of Use)	6.29	0.53

TABLE IV. STUDENTS' RESPONSES ON THE USEFULNESS OF THE CUSTOM SEARCH ENGINE

Item	Mean	SD
Using the custom search engine enables me to accomplish tasks more quickly.	6.24	0.44
Using the custom search engine would improve my learning.	6.06	0.83
Using the custom search engine makes it easier to search for learning material.	5.94	0.83
Using the custom search engine would improve my productivity.	5.88	0.86
Overall, I find the custom search engine useful in my learning.	5.82	0.81
Overall (Usefulness)	5.98	0.18

TABLE V. STUDENTS' PERCEPTIONS ON THE RICH METADATA SNIPPETS

Item	Mean	SD
1. Presenting the average ratings of the learning object helped me to determine the quality of the learning object.	6.24	0.75
2. Showing the number of times the learning object has been viewed helped me to choose the most suitable learning object.	5.94	1.09
3. The average usage time helped me to estimate how much time I may need to complete the learning object.	6.29	0.69
4. The thumbnail (the small icon) helped to me to determine the type of learning object (animation, self-assessment, etc.).	6.00	0.87
5. Overall, the metadata used in the repository described the learning objects well.	6.06	0.97
All items	6.12	0.51

Table V shows the mean responses to the questions related to evaluating the rich metadata snippets. The questionnaire contains 5 questions, asking students about their perception of different features used in the learning objects rich metadata snippets. The overall mean of the students' responses is 6.12, indicating that students strongly agree that the rich metadata are an effective way of presenting the results of the learning object search engine. Showing the average usage time and ratings are the features that received the highest evaluations by the students.

C. *Analysis of the Qualitative Comments*

A thematic analysis of the comments posted by students who used the custom search engine reveals that the students perceive the search engine to be useful. The thematic analysis is used to group the comments on the basis of the feature of interest to which the students refer. Table VI summarises the results of the qualitative comment analysis, categorised on the basis of the feature of interest and how frequently students referred to each feature in their comments.

TABLE VI. QUALITATIVE COMMENT ANALYSIS RESULTS CATEGORISED BY FEATURE OF INTEREST

Feature of Interest	Frequency
Controlled vocabulary based on CS topics	21
Filtering by type of learning object	13
Display of ratings	10
Result list including only CS learning objects	6

As Table VI shows, students consider the features related to the control vocabulary used in the search engine to be the most useful. The control vocabulary is used in the auto-complete function to suggest terms and keywords for students to minimise input and errors. The control vocabulary also provides terms in the taxonomy to allow students to determine the topic or subtopic to which the learning objects belong. Some students provided positive comments about restricting the suggested terms in the auto-complete feature to the terms related to computer science topics. This is followed by the ability to filter the search by learning object type. Some students indicated that restricting the search query results to animations and videos makes them like the search engine more.

VII. DISCUSSION

As mentioned above, finding relevant learning objects is not an easy task. Each learning object repository has its own search interface, and the usability of these interfaces is of high concern. These search services normally use terminology adapted to the metadata standard, not to the subject area or context in which the learning objects are used.

The results of this study show that the custom learning object search engine can effectively support the search and retrieval of computer science learning objects. The search engine precision exceeds 90% in the three tasks, which indicates that at least 90% of the learning objects retrieved by the search engine are relevant to the search task. This precision reduces the time a user needs to filter irrelevant results. The recall of the search engine exceeds 85%, which indicates that at least 85% of the relevant learning objects in the repositories are retrieved for each search query. Certain factors may affect the precision and recall of the search engine in the experiment. In some queries, students forget to specify the type of learning object; thus, they retrieved more irrelevant learning objects. The lack of interoperability between learning object repositories is a major influence on the custom search engine performance. Some of the computer science learning objects stored in these repositories also do not provide enough metadata information for the user to determine whether the learning object is relevant or not.

The results of the students' perception questionnaire reveal that students find the custom search engine to be easy to use and useful in supporting their self-regulated learning study. Students consider showing the results as rich metadata snippets useful in helping them choose the suitable learning objects retrieved by the query.

The results of the current study reveal that the learning object search tools must be adapted to users' needs and requirements, not to metadata standards. However, the metadata standards should still be used to store the learning objects in the repositories, although with application profiles that reflect the subject areas of these learning objects. General-purpose learning object search tools seem less useful because of the amount of time and effort students must devote to find learning objects related to their needs and preferences.

Learning object search tools should be simple and smart enough to detect students' preferences and needs from the students' interactions with the learning objects. This can be achieved by designing specialised learning object repositories that track students' use of different learning objects, detecting students' misconceptions and suggesting learning objects to overcome these misconceptions.

VIII. Conclusions and Future Work

This study addresses the challenges associated with searching for and retrieving computer science learning objects. The study presents the results of an empirical evaluation of a proposed method to improve finding learning objects to support self-regulated computer science learning. The study first proposes an application profile to categorise computer science learning objects using a more consistent and standard scheme that reflects the actual courses and topics in computer science programs. The application profile bridges the gap between the learning object metadata standard and computer science educational research. This study presents a custom search engine based on this application profile for computer science learning objects. The custom search engine helps students find learning objects related to computer science topics more easily and with high accuracy. An empirical evaluation by the participants in a computer science course reveals that the new search custom search engine has high precision and recall as an information retrieval system for computer science learning objects. The students' evaluations of the custom search engine also indicate that it is useful and easy to use for supporting their self-regulated computer science learning.

Future research will make this tool simpler and smart enough to detect students' preferences and needs from user interactions with learning objects. The computer science learning object repository in which the experiments in this study are conducted tracks students' uses of different learning objects, detects students' misconceptions, and suggests learning objects to overcome these misconceptions. These data can be used to improve the search engine by minimising the input data that students must provide and by suggesting learning objects from different repositories related to students' knowledge levels and preferences.

References

[1] G. Dettori and D. Persico, "Detecting self-regulated learning in online communities by means of interaction analysis," *IEEE Trans. Learn. Technol.,* vol. 1, pp. 11-19, 2008.

[2] S. Cassidy, "Self-regulated learning in higher education: Identifying key component processes," *Stud high Educ,* vol. 36, pp. 989-1000, 2011.

[3] M. Sosteric and S. Hesemeier, "When is a learning object not an object: A first step towards a theory of learning objects," *IRRODL,* vol. 3, 2002.

[4] E. Duval and W. Hodgins, "A LOM research agenda," in *Proceedings of the International Conference on World Wide Web*, 2003, pp. 1-9.

[5] A. Alharbi, D. Paul, F. Henskens, and M. Hannaford, "An Investigation into the learning styles and self-regulated learning strategies for computer science students," in *Ascilite 2011 Hobart* 2011.

[6] Y. Ouyang and M. Zhu, "eLORM: learning object relationship mining-based repository," *Online Inform Rev,* vol. 32, pp. 254-265, 2008.

[7] N. J. Belkin, "Helping people find what they don't know," *Communications of the ACM,* vol. 43, pp. 58-61, 2000.

[8] B. J. Jansen and A. Spink, "An analysis of web searching by European AlltheWeb. com users," *Inf Process Manag,* vol. 41, pp. 361-381, 2005.

[9] A. Walraven, S. Brand-Gruwel, and H. Boshuizen, "How students evaluate information and sources when searching the World Wide Web for information," *Comput Educ,* vol. 52, pp. 234-246, 2009.

[10] H. S. Al-Khalifa and H. C. Davis, "The evolution of metadata from standards to semantics in E-learning applications," in *HYPERTEXT '06 the seventeenth conference on Hypertext and hypermedia* 2006, pp. 69-72.

[11] J. Najjar, J. Klerkx, R. Vuorikari, and E. Duval, "Finding appropriate learning objects: An empirical evaluation," *Res Adv Technol Digital Libraries*, pp. 323-335, 2005.

[12] D. Sampson, "The evolution of educational metadata: From standards to application profiles," presented at the *Proceedings of the IEEE International Conference on Advanced Learning Technologies*, 2004.

[13] A. Alharbi, F. Henskens, and M. Hannaford, "Pedagogical framework to improve the quality of learning objects in collaborative e-learning systems," presented at the *3rd International Conference on Computer Supported Education,* Valencia, Spain, 2011.

[14] A. Alharbi, F. Henskens, and M. Hannaford, "Computer science learning objects," in *e-Education, Entertainment and e-Management (ICEEE), 2011 International Conference*, Jakarta, Indonesia, 2011, pp. 326-328.

[15] [15] LTSC. (2002, 4 July). *Draft standard for learning object Metadata*. Available: http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf

[16] E. Duval, N. Smith, and M. Van Coillie, "Application profiles for learning," in *Advanced Learning Technologies, 2006. Sixth International Conference on*, 2006, pp. 242-246.

[17] D. Sampson and P. Zervas, "Enhancing educational metadata with science education information," 2008, pp. 1081-1082.

[18] H. Stuempel, G. Salokhe, A. Aubert, J. Keizer, A. Nadeau, S. Katz, and S. Rudgard, "Metadata application profile for agricultural learning resources," *Metadata Semant,* pp. 499-507, 2009.

[19] N. Hennig. (2001, 15 June). *Test 1 Barton catalog usability tests*. Available: http://libstaff.mit.edu/webgroup/usability2001/barton/test1/overview.html

[20] S. Dumais, E. Cutrell, and H. Chen, "Optimizing search by showing results in context," presented at the Proceedings of the SIGCHI conference on Human factors in computing systems, Seattle, Washington, United States, 2001.

[21] C. Thompson, J. Smarr, H. Nguyen, and C. D. Manning, "Finding educational resources on the web: Exploiting automatic extraction of metadata," in *Proceedings of the ECML Workshop on Adaptive Text Extraction and Mining* , 2003, p. 79.

[22] R. Devarakonda, G. Palanisamy, B. Wilson, and J. Green, "Mercury: reusable metadata management, data discovery and access system," *Earth Sci Inf,* vol. 3, pp. 87-94, 2010.

[23] I. Hickson. (2012, 3 July 2012). HTML microdata. *W3C Working Draft 29 March 2012*. Available: http://dev.w3.org/html5/md/

[24] J. Ronallo, "HTML5 Microdata and Schema. org," *Code4Lib Journal,* 2012.

[25] T. Steiner, R. Troncy, and M. Hausenblas, "How Google is using Linked Data Today and Vision For Tomorrow," *Future Internet Assembly, Ghent, Belgium,* 2010.

[26] N. Smith, M. Van Coillie, and E. Duval, "Guidelines and support for building Application profiles in e-learning," *CEN/ISSS WS/LT Learning Technologies,* 2006.

[27] CCS ACM, "ACM Computing Classification System [1998 version]," ed: ACM, 1998.

[28] IEEE/ACM. (2005, 20 June). *Computing Curricula 2005, The Overview Report*. Available: http://www.acm.org/education/education/curric_vols/CC2005-March06Final.pdf

[29] J. Greenberg, K. Spurgin, and A. Crystal, "Functionalities for automatic metadata generation applications: a survey of metadata experts' opinions," *Int. J. Metadata Semant. Ontol.*, vol. 1, pp. 3-20, 2006.

[30] C. Lagoze, H. Van de Sompel, M. Nelson, and S. Warner, "Open Archives Initiative-Protocol for Metadata Harvesting-v. 2.0," 2002.

[31] [31] Merlot. (1997, 27 July). *Multimedia educational resource for learning and on-line teaching*. Available: http://www.merlot.org/merlot

[32] E. Duval, E. Forte, K. Cardinaels, B. Verhoeven, R. Van Durm, K. Hendrikx, M. W. Forte, N. Ebel, M. Macowicz, and K. Warkentyne, "The Ariadne knowledge pool system," *Communications of the ACM,* vol. 44, pp. 72-78, 2001.

[33] G. Henry, R. G. Baraniuk, and C. Kelty, "The connexions project: Promoting open sharing of knowledge for education," *Syllabus Technol High Educ,* 2003.

[34] B. Fitzgerald, *Drupal for Education and E-learning*: Packt Pub Limited, 2008.

[35] R. Baeza-Yates and B. Ribeiro-Neto, *Modern information retrieval* vol. 82: Addison-Wesley New York, 1999.

[36] C. D. Manning, P. Raghavan, and H. Schutze, *Introduction to information retrieval* vol. 1: Cambridge University Press Cambridge, 2008.

[37] F. D. Davis, "A technology acceptance model for empirically testing new end-user information systems: Theory and results," Massachusetts Institute of Technology, Sloan School of Management, 1985.

[38] M. G. Morris and A. Dillon, "The Influence of User Perceptions on Software Utilization: Application and Evaluation of a Theoretical Model of Technology Acceptance," *IEEE Software,* vol. 14, pp. 58-6, 1997.

[39] P. Legris, J. Ingham, and P. Collerette, "Why do people use information technology? A critical review of the technology acceptance model," *Inform Manage,* vol. 40, pp. 191-204, 2003.