Gradebooks as First Class Objects

Diptendu Dutta, Arup Kumar Ghosh, Sumanta Ray,
Srijan Roy, Chandan Kumar Konar
Aunwesha Knowledge Technologies Pvt. Ltd.
WEBEL Bhavan, Salt Lake, Sector V, Kolkata 700091
diptendu@vsnl.com

Abstract

Gradebooks are an integral part of all modern day Course Management Systems. Teachers spent lot of effort in designing and fine-tuning gradebooks. However, in most systems gradebooks exist as components of a course and may be used only in the context of that particular course. In this paper we propose promoting gradebooks to first class objects that may be reused across a number of courses. An implementation scheme that permits treating gradebooks as first class objects is presented. This scheme has been realised in the LearnITy™ Course Management System product that is in use at a number of sites.

1. Introduction

Powerful grading software can be one of the most effective teacher tools. The use of electronic gradebooks is on the rise and all Course Management Systems (CMS) today include a gradebook module. Gradebooks are complex objects and it takes a lot of effort by a teacher to develop and fine-tune a proper gradebook. However, in most cases gradebooks are closely associated with a particular course and are used only in the context of that course. In this paper we describe the design principles of a gradebook whose main motivation is to promote gradebooks to a status where they may exist by themselves (i.e., without being tagged to a particular course). This is what is meant by gradebooks as first class objects. The same gradebook may be used by a number of courses thus promoting re-use and adoption of institution wide standards.

The rest of the paper is structured as follows. In section 2 we present a generic architecture for gradebooks. Section 3 describes an XML binding for the proposed architecture. In section 4 we briefly describe a software implementation of the architecture that was presented in the paper. Section 5 concludes the paper with pointers to future work.

2. Generic Architecture of a Gradebook

In order to promote gradebooks to first class status we first need to come up with a generic architecture of a gradebook. This involves identifying the components of a gradebook and describing how these components may be combined to cater to the grading requirements of any course. In this direction we identify the following generic components of a gradebook:

- *Grading Category*
- *Grading Scale*
- *Grading Item*
- *Gradebook*
- *Course*

Elements of above types may have a number of attributes creating a very comprehensive and flexible framework that may be used to define various types of gradebooks.

2.1. Grading Category

This component is used to represent the type of the various grading items that make up a gradebook. Example of grading categories are

- Homework
- Class test
- Project
- Exam

2.2. Grading Scale

This component is used to represent the marking scheme used for grading items that make up a gradebook. Multiple grading schemes may exist and different grading items may be graded using different grading scales. Some examples of grading scale are given below:

Grading Scale I → A: 80-100, B: 60-79, C: 50 –59, D: 40-49, F: 0-39

Grading Scale II → A: 90-100, B: 60-89, F: 0 –59,

2.3. Grading Item

This component is used to represent the actual assessment item. Each grading item belongs to a grading category and has an associated grading scale. Examples of grading items are:

- 3rd Class test
- Summer Vacation Homework
- Midterm exam
- 2nd Physics assignment

Each grading item has the following properties:

- Grading Category
- Grading Scale
- Pass marks
- Number of attempts permitted

Other properties may be added based on additional requirements.

2.4. Gradebook

This component is used to represent the final gradebook. A gradebook is a collection of grading items as defined above. As grading items are added to a gradebook, each item may be given a <u>weight</u> to indicate how important the item is in the overall assessment and each item may also be attributed with a <u>scheduled date</u> indicating when that assessment is likely to happen. An example gradebook is shown below:

Undergraduate Gradebook

Item	weight	scheduled date
1st Class Test	10	19th July
2nd Class Test	10	03rd August
3rd Class Test	10	23rd August
Project	40	30th September
Final Exam	30	15th September

By making "weight" an attribute of the Gradebook--Item association rather than an attribute of a Grading Item, we get the flexibility of reusing the same item in different gradebooks with different weights.

2.5. Course

This component is used to represent any unit of instruction that has a gradebook associated with it. A course will have a number of other attributes associated with including:

- Syllabus
- Schedule
- Resources
- Announcements
- Assignments
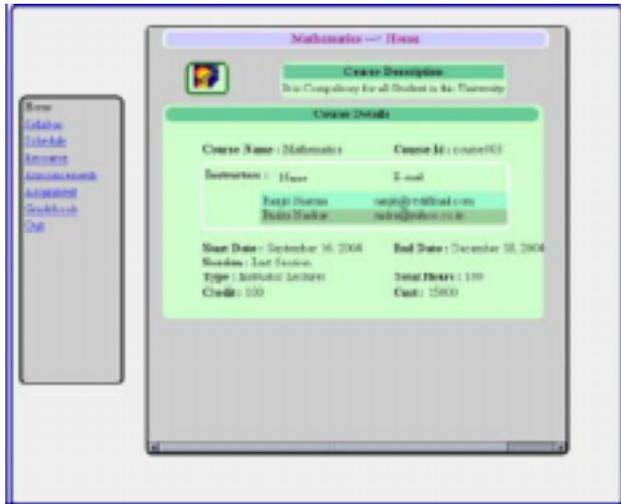- Gradebook

3. Computer representation

In order to promote reusability of gradebooks between faculty members and between institutions we need a representation mechanism that will permit easy exchange of gradebook definitions. We suggest using XML for this purpose.. This XML representation may be adopted by various CMS developers thus promoting interoperability of gradebooks in a manner similar to the adoption of the SCORM/IMS Content Packaging XML schema by LMS developers ([1], [2])enabling interoperability of courseware. Below we give an example of XML representation of a gradebook:

```
<COURSE-GRADE>
  <grade-book id="gb001" name="Grade Book-1"
scale="scale001">
   <category id="cat001" name="Exam">
    <item attempt="1" id="it001" max_marks="100"
        name="Monthly Test-1" pass_marks="30"
scale="scale001">
        <schedule_date>2005-05-18</schedule_date>
        <weightage>25</weightage>
    </item>
    <item attempt="1" id="it002" max_marks="100"
        name="Monthly Test-2" pass_marks="30"
scale="scale001">
        <schedule_date>2005-06-18</schedule_date>
        <weightage>25</weightage>
    </item>
    <item attempt="1" id="it003" max_marks="500"
        name="Final Exam" pass_marks="150"
scale="scale001">
        <schedule_date>2005-07-24</schedule_date>
        <weightage>50</weightage>
    </item>
   </category>
  </grade-book>
  <grade-scale>
   <scale id="scale001" name="UG">
     <grade lower="81" name="A" upper="100"/>
     <grade lower="51" name="B" upper="80"/>
     <grade lower="31" name="C" upper="50"/>
   </scale>
  </grade-scale>
</COURSE-GRADE>
```

4. Software Implementation

The design of a generic gradebook that was sketched above has been implemented in the LearnITy™ Course Management System developed by our company. The application has two components:

- The Student Interface – This is used by the students to log in and access the various facilities of the CMS.

The student view of the gradebook permits a student to view his/her own grades as well as the highest, lowest, and average of the entire class:

- The Administrative Interface - – This is used by the teacher/course administrator to log in and manage the various facilities of the CMS.

The administrator view of the gradebook permits a teacher to view grades of the entire class:

5. Conclusion

In this paper we presented gradebooks as first class objects that may exist by themselves and that may be used in association with multiple courses. We also presented an implementation scheme based on XML for realising this architecture. Adoption of such standard driven approach in the domain of gradebooks would promote interchange of these artifacts between faculties and between institutions.

6. References

[1] The SCORM Model,
http://www.adlnet.org/index.cfm?fuseaction=scormdown

[2] D. Dutta, C. Majumdar, and S. Majumdar, "Flexible Learning and the Adaptive Delivery of Online Content", Proceedings of Vidyakash 2002: The Conference for Online Leaning, Mumbai, Dec. 2002