Adding adaptive assessment capabilities to an e-learning system

Ioannis Hatzilygeroudis, Constantinos Koutsojannis, Nikolaos Papachristou*
Department of Computer Engineering & Informatics, School of Engineering
University of Patras, Greece
{ ihatz, ckoutsog, papaxris}@ceid.upatras.gr

Abstract

In this paper, we present EX-COFALE, an extension to an existing open-source, web-based adaptive e-learning system, namely COFALE. COFALE, although offers facilities for adaptive content presentation, adaptive use of pedagogical devices and adaptive communication, it lacks facilities for adaptive student assessment. EX-COFALE remedies this deficiency of COFALE by allowing for automated test creation and assessment based on the students' knowledge information. To this end, COFALE has been modified to allow for representation of associations between test questions and learning concepts. Also, assessment is made at two levels, the concept and the goal level, taking into account the difficulty level of the questions. To technically achieve the above, expert systems technology is used.

Keywords
Intelligent Web-Based Systems, Intelligent E-Learning, Adaptive Student Assessment, Personalized Learning.

1. Introduction

Recently, there has been a large research activity on web-based intelligent educational systems (WBIESs) [10]. WBIESs use Artificial Intelligence (AI) Techniques in order to adapt mainly to student needs for self-study. As WBIESs we consider either web-based intelligent tutoring systems (ITSs) [9] or adaptive hypermedia education systems (AHESs) incorporating intelligent techniques [2]. On the other hand, e-learning environments provide facilities mainly for helping course generation and management and refer to both the tutors and the students. Adding facilities (intelligent or not) for tutors in WBIESs make them a kind of intelligent e-learning systems (IELSs) [6], [11].

COFALE (Cognitive Flexibility in Adaptive Learning Environments) is an open-source adaptive e-learning system [4]. COFALE, apart from supporting common

adaptive techniques, such as student modelling and adaptive content presentation, also provides means for adaptive use of pedagogical devices and adaptive communication support.

According to [3], a system, in order to facilitate adaptive support, should be designed to meet the following operational criteria for adaptability:

- *Adaptive presentation of learning content*: At any given time during the learning process, the learner is provided with learning contents that are appropriate to his/her present competence.
- *Adaptive use of pedagogical devices*: At any given time during the learning process, the learner is encouraged to do learning activities that are appropriate to his/her present competence.
- *Adaptive communication support*: For any learning discussion, the learner is suggested with peers who are appropriate to help him/her overcome his/her own difficulties.
- *Adaptive assessment*: At any given time during the learning process, the learner is provided with assessment problems and methods that are appropriate to his/her present competence.
- *Adaptive problem-solving support*: For any problem solving session, the learner is supported with appropriate feedback (e.g. appropriate hints) to solve the problem effectively.

COFALE manages to successfully accommodate the first three requirements [4], [5]. So, COFALE lacks functionalities related to adaptive student assessment/evaluation. In order to help tutors to be able to create courses with adaptive assessment capabilities, we modified and extended it to provide such functionalities.

The paper is organized as follows. In Section 2, a short overview of COFALE with an emphasis to its adaptive capabilities is presented. Section 3 deals with the proposed extensions to COFALE, mainly concerning adaptive assessment, whereas Section 4 deals with implementation aspects. Section 5 presents related work and finally Section 6 concludes the paper.

---

*The order is alphabetical.

2. COFALE

COFALE is an adaptive e-learning environment supporting cognitive flexibility. Cognitive flexibility is a learning theory which emphasizes a case study based approach to learning, involving context-dependent and realistic situations. COFALE is based on ATutor, an open-source, web-based learning content management system (LCMS) designed and maintained by ATRC [1]. Compared to contemporary adaptive learning systems, COFALE seems to fulfill all the needed criteria for cognitive flexibility.

COFALE gives the tutor the ability to implement various types of student models. For example, one can implement two types of student model, 'novice' and 'expert'. Also, the learning content in COFALE can be decomposed into quite primitive content (or learning) units, so that the system can present each student different content units. For example, simpler examples for a 'novice' learner and advanced for an 'expert' one. In a course about 'recursion', COFALE introduces simpler concepts and proposes simpler situations to 'novice' students (e.g., recursive methods, base cases, recursive part, Fibonacci numbers) than those to 'expert' ones (e.g., recursive thinking, iterative thinking, partition) [4]. This implements adaptive presentation of learning content.

At the end of each content page, the student is encouraged and guided to do a number of learning activities, depending on his/her current "mental model" about the concept of study. This means that COFALE allows for a second level of student modeling, that of "mental models" of the students, which are related to the type of the concepts to be taught. Given the type of first-level student model (novice, expert), certain types of mental models may be excluded. So, COFALE may suggest activities based on simpler mental models to a "novice" learner, but based on more complex models to an "expert" one. In the course about recursion COFALE suggests 8 activities to 'expert' but only 5 "advanced" tasks to 'novice' learners. The 'novice' learners are versed in the use of COFALE, so COFALE does not present them with these three learning activities: "Next Page", "Related Topics", and "Learning History" [4]. This implements adaptive use of pedagogical devices.

Moreover, while learning with COFALE, students can use a tool to search for peers who could help them to overcome difficulties about acquiring the concept of recursion. For example, COFALE may suggest some "expert" students to a "novice", so that he/she can ask them questions about problems, or may suggest an "expert" student to another "expert" student, so that they can exchange ideas about advanced concepts or activities. That implements adaptive communication support.

3. Extending COFALE

To make COFALE meet our needs, we made some modifications and extensions to it, presented in the following sub-sections. We call the extended system EX-COFALE (EXtended COFALE).

3.1 Domain Knowledge

COFALE uses a tree-like structure to represent domain knowledge. COFALE can alter the domain knowledge tree presented to the user depending on the user model. In Figure 1, a partial such domain knowledge tree, which we constructed, implementing a course on 'radio safety' from the health care domain, is presented. Actually, what it can do is to hide certain subtrees, which are not appropriate for a certain user. EX-COFALE goes a step further. It can re-arrange the branches of the tree, based on the user's model, thus achieving something like concept sequencing.
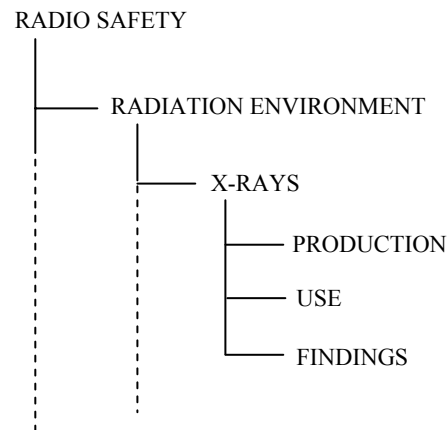

Fig. 1. Part of a Domain KnowledgeTree

3.2 Test Creation

We also modified the test manager of COFALE. We added more functionality as far as test construction is concerned: (a) The tutor can associate a test to a specific learning goal (set of concepts) and (b) The system can now automatically create a test. The tutor should only create and store questions in the system's database. Also, he/she may define the number and the difficulty levels of the questions to be included in a test for each concept. This is done via a rule-based expert system.

As far as creation of test questions is concerned, we added the capability of defining associations between a learning concept and corresponding questions. This way a test has each question associated with a specific learning concept. More than one question may refer to the same

concept. Questions may have different difficulty levels assigned to them. The tutor is able to insert, delete or modify all the parts/attributes of a question (i.e. the body of a question, its answers, possible help hints, the associated concept, the difficulty level etc). We must note here that two types of questions, multiple-choice and true-false, can be automatically marked. There is a third one, open-end questions, which are manually marked. The questions are created once, while in COFALE the tutor had to create the same question for different tests more than one time.

The process of a test creation by the tutor for a specific concept or chapter, assigned to different student models is quite straightforward. For example, in the course of "radio safety" there are different questions, with different difficulty levels associated with the sub-concepts of 'Production', 'Use', and 'Findings' of concept 'X-Rays'. Creation of a test for the concept 'X-Rays' is possible through random selection of the questions concerning the above sub-concepts. The only thing the tutor has to do is to select the topics for which the system will collect questions from the database and generate a test. He/she also has to assign the test to a specific student model. A revision test can be made by the tutor, after he/she selects all the concepts he/she wants the system to generate a test for. For example, a revision test for the chapter of 'nuclear medicine', in the course of 'radio safety', is generated after selecting the concepts of 'PET', 'Radiation Therapy', 'Radiosurgery', 'Proton Therapy' and 'Brachytherapy'.

3.3 Student Assessment

One of the most important functions of an intelligent e-learning system is student evaluation (or assessment). Student evaluation refers to the evaluation of the knowledge level of a student after having dealt with a learning goal. In other words, how well a student has learnt the concepts related to a learning goal. Student evaluation is important for both the students and the tutor.

COFALE allows for evaluation of students based on the tests provided for each learning goal. It actually allows for a learning goal level evaluation based on tests with predefined indistinguishable questions. By "indistinguishable" we mean that there is no explicit representation of which concept each question refers to (or examines). So, the system cannot be aware of those associations to use them.

In EX-COFALE, a student is evaluated at two levels: (a) the concept-level and (b) the goal-level. The concept-level evaluation deals with the level of understanding of the individual concepts of a learning goal test, whereas the goal-level evaluation deals with the level of understanding of a learning goal as a whole. Furthermore, EX-COFALE allows for on-line test creation, even if a student has not completed the study of all of the concepts related to a

learning goal. This is achieved via the above mentioned rule-based expert system.

The knowledge level of a student, as far as a concept is concerned, is classified in one of the following three categories: (a) low (0-49), (b) medium (50-70) and (c) good (71-100), whereas, as far as a learning goal is concerned, in one of the following five categories: (a) low (0-30), (b) average (31-49), (c) good (51-70), (d) very good (71-85) and (e) excellent (86-100) (within the parentheses are the corresponding ranges of the marks to be achieved).

The knowledge level of a student for a concept, called *concept level* (*CL*), is calculated via the following formulas:

$$CL = \sum_{i=1}^{3} Qm_i * qw_i$$

$$Qm_i = \frac{\sum_{j=1}^{n} Qm_j^i}{n}$$

where i represents the difficulty level (1→ easy, 2→ medium, 3 → difficult), Qm_j^i represents the answer to question j (which is 100 if it's correct and 0 otherwise), n is the number of the questions of the same difficulty associated with the concept, Qm_i is the average mark of the answers to questions of the same difficulty level related to the concept and qw_i is the question's weight. The weight of a question is related to the difficulty level of a question and the composition of the set of questions used for testing the concept. Table 1 presents the corresponding weights. It is assumed that at least two questions for each examined concept exist in a test.

If $CL \geq 0.5$ then the student has an acceptable level of knowledge about the corresponding concept.

Table 1. Question weights

Easy (E)	Medium (M)	Difficult (D)	Question (s) weight(s) (/100)
1	1	1	20 (E), 50 (M), 30 (D)
1	1	0	40 (E), 60 (M), 0 (D)
1	0	1	40 (E), 0 (M), 60 (D)
0	1	1	0 (E), 60 (M), 40 (D)

As an example, suppose we have four questions that are used to evaluate the knowledge of a student about a concept (e.g. X-Rays Production) and that two of them are 'easy' and from the other two one is 'medium' and one is 'difficult'. If the student gives right answers to one of the two easy questions and the medium one, but wrong to the other easy question and the difficult one, then the $Qm_1 = (100+0)/2 = 50$, $Qm_2 = 100$ and $Qm_3 = 0$. So, the corresponding CL= 50*0.2 + 100 * 0.5 + 0 * 0.3 = 60.

The knowledge level of a student for a learning goal, called *goal level* (*GL*), is calculated from the following formula:

$$GL = \frac{\sum_{i=1}^{n} CL_i}{n}$$

where CL_i is the value of the knowledge level of the student for concept i (i.e. the achieved mark before it is classified in one of the three levels) and n is the number of concepts that constitute the learning goal. Again, if GL $\geq$ 0.5 then the student has an acceptable level of knowledge about the corresponding learning goal, given that each CL_i $\neq$ 0, that is there is no concept not studied at all.

4. Implementation Architecture

Figure 2 shows the implementation architecture of EX-COFALE (as a modification of that of COFALE). The functionalities of the architecture are as follows:

- The user uses a Web browser to log into the system and make a request.
- Taking into account the user's request, the browser sends a HTTP request to the Web server in which a set of PHP scripts were installed.
- Depending on the kind of the HTTP request, the Web server creates new data or update existing data or retrieve existing data by connecting the MySQL database server in which all data about the user, learning content, tests, forums, and so on are stored and indexed. Then, the Web server formulates an HTML file including a CSS format and sends it back to the browser.
- On the basis of the HTML file and the CSS format received from the Web server, the browser creates a Web page and presents it to the user.
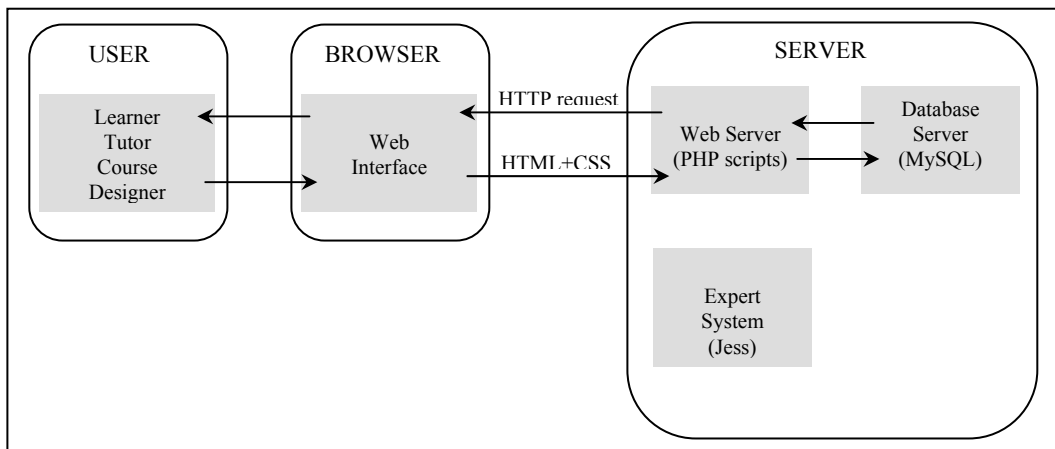
- For the tests' evaluation a rule-based expert system is used, which is implemented in Jess, a java based expert system tool [8]. PHP scripts make all the appropriate communication between the expert system and the browser. PHP scripts also bridge the expert system with the MySQL database server to store the results of the tests evaluation.
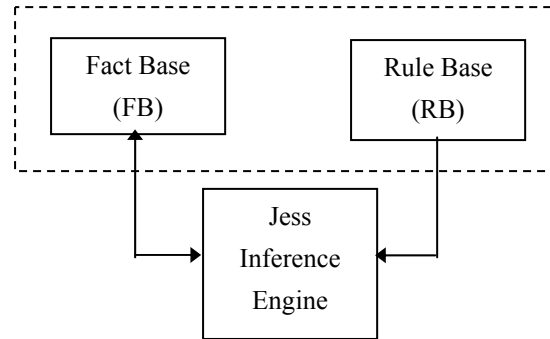

Figure 3. The structure of Expert System

The structure of the expert system is illustrated in Figure 3. It consists of the fact base (FB), the rule base (RB) and the Jess inference engine (JESS IE). FB contains facts, which are created from the problem data, whereas RB contains the rules used by the IE. The expert system processes the facts via the rules of RB, according to the Jess IE instructions, to (a) select the questions from the database according to the students' knowledge levels and tutor's settings and (b) deduce the knowledge level values of the students for the concepts involved in the delivered test, based on the test results.

A core prototype of the system has been implemented, which does not offer at the moment all the functionalities.

Figure 2: EX-COFALE Architecture

5. Related Work

There have been a number of e-learning systems (or environments) that can be used to produce adaptive courses. However, a few of them provide facilities for adaptive student assessment.

ALE (Adaptive Learning Environment) is an e-learning environment (implemented in the context of the WINDS project [15, 16]) that integrates an intelligent tutoring system, a computer instruction management system and a set of cooperative tools [12]. ALE can produce individualized courseware for students, based on their current state of knowledge, their preferences and learning styles. So, it supports adaptive content presentation. However, it does not support any facility for student assessment management. EX-COFALE provides tutor with the ability to evaluate the learner's performance and monitor his/hers comprehension of the concepts and chapters independently.

aLFanet (Active Learning For Adaptive interNET) is an e-learning platform created in the context of an IST project [13, 14]. It integrates adaptive techniques from intelligent tutoring systems (ITSs), adaptive hypermedia systems (AHSs) and computer supported collaborative learning (CSCL). It consists of three subsystems. The authoring subsystem provides facilities for the creation of instructional-based courses and optionally the possibility to define adaptive questionnaires. Course administration subsystem includes the users' management, learners/tutors assignment to the courses, permissions management and users' data privacy. It also provides facilities for the definition of new presentation layouts. Finally, LMS eLearning instruction subsystem includes an adaptive course presentation depending on the learner profile and preferences, dynamic user modeling (learner profiles refinement) and tools for learning process (collaborative frameworks). Again, aLFanet, although provides facilities for test/questionnaire creation, it does not seem to provide any facilities for student assessment. It does not provide any sensible reports to authors to help them to evaluate how learners are dealing with course activities.

In [17] a system that integrates course authoring and student assessment capabilities is presented, focusing on the assessment issue. The presented assessment method is based on SPC (Student-Problem-Course) table. Assessment is based on the correct answers and the time taken to deliver it. Although it offers automatic assessment, through the SPC table, it does not seem to offer automatic test creation. An on-line exam will be held after instructor select problems in the exam scope. Also, it uses weights associated with course units, which are taken into account in student evaluation. However, it does not take into account neither the difficulty of the questions nor the student level. All students must attend the same course and the questions' difficulty is calculated by the average performance of all the students to each of these questions. Also, students get a secondary evaluation by the total study time of a course unit compared to the minimum requirement time which was preset by the instructor for that particular course unit. Except that, the object navigation frequency must be over the threshold which was defined by the instructor. We implemented a more flexible and tolerant technique for student evaluation, where more experienced learners might have good test results without having to have a thorough attendance to some learning concepts. Finally, the system in [17] does not provide any collaborative tools for students inter-assessment, in contrast to EX-COFALE, which provides a number of such tools, like e-mail, forums, chat rooms and learners' hyperspace area for team solving situations.

The system presented in [6, 7] is a really good ILES. It provides content presentation and student assessment adaptivity alongside extensive authoring facilities. It uses three levels of assessment (paragraph, sub-chapter, and chapter). Tests can be adaptively (i.e dynamically) created. It uses only multiple choice questions, where wrong answers are taken into account for the assessment. Questions are not distinguished by difficulty level; all are considered as equally contributing to the final mark. Finally, there is a mark threshold associated with each learning item that should be overtaken by the student in order to consider that he/she has successfully passed it. The system is rather examination-oriented. Although it does not luck a tool for communication of the tutor and the learners, EX-COFALE offers a variation of such tools. EX-COFALE offers the tutor a more efficient tool, to understand how different types of learners comprehend and what difficulties they face in studying each learning concept.

6. Conclusion

In this paper, we present EX-COFALE, an extension to an existing open-source, web-based intelligent e-learning system called COFALE. COFALE, although offers facilities for adaptive content presentation, adaptive use of pedagogical devices and adaptive communication, it lacks facilities for adaptive student assessment, one of the requirements for such systems. In EX-COFALE, we introduce automated test creation and assessment based on the students' knowledge information. To this end, COFALE has been modified to allow for representation of associations between test questions and learning concepts. Also, questions are distinguished in three levels of difficulty. Assessment is done at two levels, the concept and the goal level. In the assessment process, the difficulty level of questions is taken into account, which is not the case in existing systems. To technically achieve the above,

expert systems technology is used. Very few e-learning environments provide facilities for adaptive assessment. In this vein, extending existing open source tools seems to be an interesting idea.

Although EX-COFALE at its present design offers capabilities for adaptive assessment, it does it in a degree that can be improved. For example, answers to questions are marked as correct or wrong, i.e. by two concrete values. This may not assess correctly the knowledge level of students in all cases. A more fine grained marking would improve it. To this end, other factors related to student interaction could be taken into account (e.g. number of tries, whether system help were used etc). Also, assessment is not parameterized as far as expert system rules are concerned. An authoring unit related to expert system rules could be another direction for improvement. Finally, adapting the difficulty level of the questions of a test to student's current knowledge level is another aspect for strengthening assessment adaptation.

7. Acknowledgements

8. References

[1] Adaptive Technology Resource Center, "*ATutor learning content management system*" (http://www.atutor.ca/), 2004.

[2] Brusilovsky, P., "Methods and Techniques of Adaptive Hypermedia". In: Brusilovsky, P., Kobsa A. & Vassileva, J. (Eds.): *Adaptive Hypertext and Hypermedia*, Kluwer Academic Publishers, 1998.

[3] Brusilovsky, P., "Adaptive and Intelligent Technologies for Web based Education", In C. Rollinger and C. Peylo, *Special Issue on Intelligent Systems and Teleteaching, Künstliche Intelligenz, 4*, 1999, 19–25.

[4] Chieu, V.M. and Milgrom, E., "COFALE: An Adaptive Learning Environment Supporting Cognitive Flexibility", The Twelfth International Conference on Artificial Intelligence in Education, 2005, 491–498.

[5] Chieu, V. M., Anh, D. T. V. and Hung, P. K., "An Operational Approach for Analyzing ICT-based Constructivist and Adaptive Learning Systems", 4th IEEE International Conference on Computer Sciences: Research, Innovation & Vision for the Future (RIVF'06), February 12-16, Hochiminh City, Vietnam, 1-10.

[6] Christea, P. D., Tuduce, R., Savescu, I. A., Grogorin, C. A., Tomozei, D.-C., Gradinescu, V. R. and Rangu, C. M., "Prototype Implementation of an Intelligent E-Learning System", Proceedings of the IASTED International Conference on Web-Based Education (WBE-04), Febr. 16-18, 2004, Innsbruck, Austria, Acta Press, 441-446.

[7] Christea, P. D. and Tuduce, R., "Test Authoring for Intelligent E-Learning Environments", First International Workshop on Authoring of Adaptive and Adaptable Educational Hypermedia, 2004. (http://wwwis.win.tue.nl/~acristea/WBE/416-805_WBE-PCristea_RTuduce_6pg.pdf)

[8] Friedman-Hill, E., *Jess in Action: Rule-Based Systems in Java*, Manning Publishing, 2003.

[9] Freedman, R, "What is an Intelligent Tutoring System?", *Intelligence* 11(3), 2000, 15–16.

[10] Hatzilygeroudis I. (Guest Editor), Special Issue on AI Techniques in Web-Based Educational Systems, International Journal on AI Tools (IJAIT), 13(2), 2004.

[11] Kazi, S. A., "A Conceptual Framework for Web-Based Intelligent Learning Environments Using SCORM-2004", Proceedings of the IEEE ICALT-2004, Aug. 30-Sept. 1, 2004, Joensuu, Finland, IEEE Computer Society, 2004, 12-15.

[12] Kravcik, M and Specht, M., "Authoring Adaptive Courses-ALE Approach", Proceedings of the IASTED International Conference on Web-Based Education (WBE-04), Febr. 16-18, 2004, Innsbruck, Austria, Acta Press, 2004, 396-401.

[13] Santos, O. C., Boticario, J. G. and Barrera, C., "ALFANET: An Adaptive and Standard-Based Learning Environment Built upon DotLRN and Other Open Source Developments", Calvo, R.A., Ellis, R.A. and Peters, D. Internationalisation and ElearningSystems: .LRN Case Studies. In Delgado Kloos, C. and Boticario, J.G. (eds) Proceedings of Foro Hispano .LRN, Madrid 2005.

[14] Santos, O. C., Gaudioso, E., Barrera, C. and Boticario, J. G., "ALFANET: An Adaptive E-Learning Platform", 2nd International Conference on Multimedia and ICTs in Education (m-ICTE2003), 2003.

[15] Hóttenhain. R., Klemke. R., Kravcik. M., Pesin. L., Specht. M., "Adaptive Learning Environment in WINDS", In Proceedings of ED-MEDIA 2002, Denver Colorado, AACE Press, Charlottesville, VA, 2002, 1846–1851.

[16] Roland Klemke, Milos Kravcik, Leonid Pesin, Marcus Specht, "Authoring Adaptive Educational Hypermedia in WINDS", Online Proc. ABIS 2001 Workshop (Adaptivitat und Benutzermodellierung in interaktiven Softwaresystemen), Dortmund, Germany, 2001. (Available at http://www.kbs.uni-hannover.de/~henze/ ABIS_Workshop2001/ABIS_2001.html).

[17] Hsuam-Pu Chang, Nigel H. Lin, Timothy K. Shih, H.-P., "An Intelligent E-Learning System with Authoring and Assessment Mechanism", Proceedings of the 17th International Conference on Advanced Information Networking and Applications (AINA'03).