

# Learning Content and Software Evaluation and Personalisation Problems

Eugenijus KURILOVAS, Silvija ŠERIKOVIENĖ

*Institute of Mathematics and Informatics*

*Akademijos 4, LT-08663 Vilnius, Lithuania*

*Vilnius Gediminas Technical University*

*Saulėtekio 11, LT-10223 Vilnius Lithuania*

*e-mail: eugenijus.kurilovas@itc.smm.lt; silvija.serikoviene@gmail.com*

Received: July 2009

**Abstract.** The paper aims to analyse several scientific approaches how to evaluate, implement or choose learning content and software suitable for personalised users/learners needs. Learning objects metadata customisation method as well as the Method of multiple criteria evaluation and optimisation of learning software represented by the experts' additive utility function are analysed in more detail. The value of the experts' additive utility function depends on the learning software quality evaluation criteria, their ratings and weights. The Method is based on the software engineering Principle which claims that one should evaluate the learning software using the two different groups of quality evaluation criteria – 'internal quality' criteria defining the general software quality aspects, and 'quality in use' criteria defining software personalisation possibilities. The application of the Method and Principle for the evaluation and optimisation of learning software is innovative in technology enhanced learning theory and practice. Application of the method of the experts' (decision makers') subjectivity minimisation analysed in the paper is also a new aspect in technology enhanced learning science. All aforementioned approaches propose an efficient practical instrumentality how to evaluate, design or choose learning content and software suitable for personalised learners needs.

**Keywords:** e-learning systems, learning objects, repositories, virtual learning environments, personalisation, multiple criteria evaluation, optimisation, reusability.

## 1. Introduction: The Basic Notions, Principles and Methods Used in the Research

The problems of customisation/personalisation of Learning Object (LO) metadata as well as Learning Object Repositories (LORs) and Virtual Learning Environments (VLEs), their technological quality evaluation and optimisation are high on the agenda of the European research and education systems.

Different scientific methods are used for quality evaluation of learning software packages such as LORs and VLEs. The paper is aimed to consider the problems of expert evaluation and personalisation of LORs and VLEs technological quality criteria.

The basic notions, principles and methods applied in the paper are as follows.

Learning object is referred to as any digital resource that can be reused to support learning (Wiley, 2000). LORs are considered here as properly constituted systems (i.e.,

organised LOs collections) consisting of LOs, their metadata and tools/services to manage them (Kurilovas, 2007). Metadata is referred here as structured data about data (Duvall *et al.*, 2002). VLEs are considered here as specific information systems which provide the possibility to create and use different learning scenarios and methods (IMI, 2005). In ISO/IEC 14598-1:1999 quality evaluation is defined as the systematic examination of the extent to which an entity (part, product, service or organisation) is capable of meeting specified requirements.

Different scientific methods are used for quality evaluation of software. Multiple criteria evaluation Method is referred to as the experts' additive utility function presented further in Section 4.3 including the alternatives' evaluation criteria, their ratings (values) and weights.

Expert evaluation is referred to as the multiple criteria evaluation of the learning software aimed at the selection of the best alternative based on score-ranking results. According to Dzemyda and Saltenis (1994), if the set of decision alternatives is assumed to be predefined, fixed and finite, then the decision problem is to choose the optimal alternative or, maybe, to rank them. But usually the experts (decision makers) have to deal with the problem of optimal decision in the multiple criteria situation where the objectives are often conflicting. In this case, according to Dzemyda and Saltenis (1994), "an optimal decision is the one that maximises the decision maker's utility".

The authors apply here one of the software engineering principles which claims that one should evaluate the software using the two different groups/types of evaluation criteria – 'internal quality' and 'quality in use' criteria (further referred to as Principle). According to Gasperovic and Caplinskas (2006), 'internal quality' is a descriptive characteristic that describes the quality of software independently from any particular context of its use, and 'quality in use' is evaluative characteristic of software obtained by making a judgment based on criteria that determine the worthiness of software for a particular project or user/group (Gasperovic and Caplinskas, 2006).

The rest of the paper is organised as follows. Section 2 presents the comprehensive LORs technological quality evaluation model, Section 3 – the comprehensive VLEs technological quality evaluation model, Section 4 – implementation of the customisable LOs metadata schema and the method for the multiple criteria evaluation and optimisation of LORs and VLEs for the particular learner needs. Conclusion and results are provided in Section 5.

## **2. Learning Object Repositories Quality Evaluation Criteria**

The learning software multiple criteria evaluation Method proposed by the authors is based on the software quality criteria, their ratings and weights. The most difficult problem here is the analysis and proposal of suitable evaluation model (set of criteria).

First of all let us review and shortly analyse the literature on existing well-known LORs technological quality evaluation models (i.e., sets of evaluation criteria) and methods.

The main attention is paid to the sets of evaluation criteria (i.e., the models), but several evaluation methods concerning the application of ratings (values) and weights of the evaluation criteria are also provided.

### 2.1. SWITCH Learning Object Repository Quality Evaluation Grid

The first LOR quality evaluation tool presented here is SWITCH project tool (SWITCH collection, 2008) developed while evaluating DSpace and Fedora LORs in 2008 (see Table 1).

Table 1  
SWITCH LOR quality evaluation criteria

Criteria	Sub-criteria	Description
Architecture	Flexible and modular system	Flexible and modular architecture that allows to include various extensions in the future that are unknown now
	Possibility to use LOR system as part of a federation	
	API for storage engine	Basic operations accessible through well-defined, documented and designed API. API: SOAP, REST or comparable. Basic operations: Create new, edit existing, search, query (needed for LMS-LOR integration)
	API for user access rights	User access rights can be assigned programmatically
	API for federation functions	APIs for metadata harvesting and federated search
	Metadata search with heterogeneous schemas	Metadata search works with heterogeneous schemas across LOR federation and within a LOR. Applies metadata mapping or another best-effort approach
	Full-text search	Full text search: in federation, in objects of most common formats: HTML, XML, TXT, PDF, PPT, DOC, IMS-CP, SCORM
	Performance	A common single-server installation is able to reasonably deal with at least 50'000 objects
	Scalability	Possibility to support up to 1Mio objects, if necessary with clustering etc.
	Security	No intrinsic vulnerabilities. Bullet-proof access rights system
	Interoperability	OAI-PMH, federated search
	Persistent links	Possibility to add persistent link system like OAI, handle, URN
	Internationalisation	Possibility to support at least EN, IT, DE, FR
Metadata	Minimal metadata schema	Possibility to define a minimal metadata schema with overall mandatory items
	Predefined sets of metadata	Possibility to pre-define common metadata schemas (IMS, MPEG7, DC etc.) that can be used when necessary
	Customisable metadata schema	Institutions, groups of interest or individuals can extend the minimal metadata schema according to their needs
	Metadata mapping for metadata search	Possibility to map metadata items between schemas. For example "contributor" of mandatory schema to "author" of IMS schema
	Unicode support	
	Social tagging	Possibility to support dynamically defined social tags
Graphical user interface	Complete standard UI	Complete standard UI available that covers all important functions for administrators and end-users
	Extensible standard UI	The standard UI can be customised and extended
	Multiple standard UIs	Multiple standard UIs can be configured to run on the same repository system (needed to support multiple institutions)
	Custom UIs	Possibility to add special purpose UIs as needed, like stripped-down query interface, spec. video portal, etc.

(To be continued)

Table 1 (continuation)

Criteria	Sub-criteria	Description
	AAI authentication	Possibility to add AAI authentication system
	Associate copyright license	Pre-defined copyright licenses (like Creative Commons) can be easily associated
	Direct distribution	Objects can be directly accessed through a URL
	Direct streaming	Video/audio objects can be directly streamed
	Alternative protocols for data upload	Access trough https, WebDAV, (s)ftp, ...
Storage	Object can be of any format	
	Multi-part objects	One LO may consist of multiple elements: a CD consists of audio tracks and scanned booklet images
	Access rights	Possibility to define read and write access rights on 4 levels: world, institution, self-defined group, private
	Hierarchical organization	LOs can be stored in a self-defined hierarchical structure like Institution – Domain – Department – Teacher
	Property and metadata inheritance	Object properties (access rights) and metadata items can be inherited within hierarchy. Support for predefined properties and possibility to override it
	Versioning system	Re-uploading an object with the same ID does not overwrite the original, but create a new version.
	Large objects	Support for large objects like videos of several GBytes
Other	Strength of development community	
	Strength of users community	
	Code quality	
	Documentation quality	
	Ease of installation	

The results of the short analysis of this model are as follows.

There is no clear division of the criteria into ‘internal quality’ and ‘quality in use’ criteria in this tool. According to the Principle (see the Introductory Section), ‘internal quality’ criteria should be mainly the area of interest of the software engineers, and ‘quality in use’ criteria should be mostly analysed by the programmers and users taking into account the users’ feedback on the usability of software.

However we can notice that ‘Architecture’ group’s sub-criteria are mainly engineering criteria, and therefore they could be analysed as ‘internal quality’ criteria, and all other criteria are mainly users-related, and therefore they could be analysed as ‘quality in use’ criteria.

No accessibility and/or design for all criteria are mentioned in this model.

## 2.2. CatalystIT Technical Evaluation Tool for Open Source Repositories

The second LOR quality evaluation model presented here is the model developed by CatalystIT while evaluating DSpace, EPrints and Fedora LORs in Technical Evaluation of Selected Open Source Repository Solutions (2006). The model is quite complex and com-

binest several types of criteria: scalability, ease of working on code base, security, interoperability, ease of deployment, system administration, internationalisation, open source, workflow tools, and community knowledge base (see Table 2).

Table 2  
CatalystIT LOR quality evaluation criteria

Criteria	Sub-criteria	Description
Scalability	Scale Up	Ability for the Repository to scale higher by adding more resources (CPU, ram, etc.)
	Scale out	The repository supports caching, adding more instances, and other mechanisms to scale higher
	Architecture	The repository be separated into different local parts and put into different machines (e.g., separate the database, data directory, components from the repository to distribute to different machines)
Ease of working on code base	Add/change digital object type	The work involved in adding or changing a digital object type such as adding or changing metadata
	Documentation of code and	code consistency & style
Security	Data encryption	Supports encryption of data while transmitting the content, such as using SSL/https
	Server security	What does the repository require for installation? Does it follow good security practices, e.g., proper file permissions, secure database connection?
	Authentication	The authentication used by the repository to authenticate user
	Authorisation/access rights	Support for different roles to properly manage the content and administer the system
	Ability to restrict access at repository item level	E.g., view metadata but not content
Interoperability	OAI-PMH compliant (essential)	
	SOAP, UDDI	
	SRU/SRW	
	Bulk import and export	Support for batch/bulk import and export of digital objects
	Institution exit mechanism	to withdraw their content from the repository farm (essential)
	Authentication	Use an external authentication mechanism (ex. LDAP)
Standard metadata	Dublin core, METS, etc.	
Ease of deployment	Software and hardware requirements	The repository only requires common/basic software and hardware
	Packaging and installation steps	
	Separate repository and branding for each institution (essential)	
System administration	Ability to customise look and feel	Change the header, theme, footer
	Ease of publishing	Inexperienced users of the repository can easily publish content
Internationalisation	Localisable UI	
	Unicode text editing and storage	

(To be continued)

Table 2 (continuation)

Criteria	Sub-criteria	Description
Open source	Open source license (required)	
	Defined roadmap for the future	
Workflow tools	Workflow integration	Support to use different workflow tools
	Support for different workflows	
Community knowledge base	Quality and completeness of information on the product's web site	
	Size of and level of activity in the developer community	
	Size of and level of activity in the user community	
	Availability and use of a range of communication channels (email, forums, IRC, wiki, etc.)	
	Software release history for evidence of sustainability and vitality	
	Documentation on how to set up and manage a repository farm	One code base, many independent repositories

This LORs evaluation model also proposes the set of ratings (values) to assess the evaluation criteria. Each criterion in this tool is proposed to be given a rating (value) to be used when evaluating LORs. Major criteria (if needed) have to be broken down into sub-criteria with each sub-criterion also having a rating. The rating range is 0–4, with 0 being the lowest and 4 being of the highest value.

The results of the short analysis of this model are as follows:

There is no division of the criteria into ‘internal quality’ and ‘quality in use’ criteria in this tool. We can notice that ‘Scalability’, ‘Security’, ‘Interoperability’, and ‘Ease of deployment’ criteria are mainly engineering criteria, and therefore they could be analysed as ‘internal quality’ criteria, and the other criteria are mainly users-related, and therefore they could be analysed as ‘quality in use’ criteria.

Many useful evaluation criteria are missed in this model.

### 2.3. OMII Software Repository Evaluation Criteria

The next model presented here is “Software Repository – Evaluation Criteria and Dissemination” prepared by Newhouse (2005) from Open Middleware Infrastructure Institute (OMII).

This document has specified the three critical phases of the software repository process:

1. The information that must be captured when a product is created within the repository and a specific release submitted to the repository.
2. The assessment criteria that should be used to review the software contribution.
3. How the product and release information, coupled with the evaluation results, are presented within LOR.

The model combines 3 types of criteria: documentation, technical and management (see Table 3).

Table 3  
OMII LOR quality evaluation criteria

Criteria	Sub-criteria	Description/Ideal
Documentation		A high overall score here indicates that the user can be reasonably confident that the supporting documentation will answer the majority of their queries. How comprehensive and useful is the provided documentation? Ideal: A high score indicates that the documentation provides sufficient depth and coverage to be useful for those trying to utilise the product for its main purpose/function
	Introductory docs.	A concise summary of the software. Ideal: Information as to what the software does and how to quickly get started with it
	Pre-requisite docs.	Information relating to environment required for running this software. Ideal: Is the environment required for this software well described?
	Installation docs.	Information on how to install the software. Ideal: Clear instructions on how to install the software
	User docs.	Information on the API. Ideal: Clear simple user manual with usage scenarios and sample code
	Admin docs.	Information on how to administer the software. Ideal: Clear instructions on how to configure the software and maintain it in operation
	Tutorials	Details how to use the software. Ideal: Clear, simple step-by-step description how to use the software with code samples if appropriate
	Functional specification	Functional specification of the software. Ideal: Clear, simple description of product's functionality
	Implementation specification	Implementation details of functional specification. Ideal: This document should contain not only the implementation details but also the reasons
	Test documents	Details of product testing. Ideal: Details of the test plans, test code and results (with known issues) from running on various platforms and scenarios. Also describe how the user can repeat the same tests
Technical		The evaluator will use the provided documentation to try and use the software. Their success (or otherwise) in using the software will demonstrate if the contributed software provides useful functionality. An examination of the technical components of the software. Ideal: Can the product be deployed and does it run successfully from the provided documentation?
	Pre-requisites	Software and environment changes necessary to support the installation of the software. Ideal: Are the pre-requisites accurately described and sufficient to install and run the software?
	Deployment	Deployment of the product into the server or client environment. Ideal: How easy is the deployment of this software into the required environment?
	Verification	Evaluating the correct operation of the product. Ideal: Is it clear how you can verify that the software has been successfully deployed and operating correctly, e.g., post-installation tests?
	Stability	Determination as to the stability of the production. Ideal: Does the software run reliably under reasonable usage and are there tests to support this?
	Scalability	An assessment of the scalability of the software. Ideal: How well does the software respond to high levels of utilisation and concurrent client activity and are there tests to support this?
	Coding	An inspection of the code within the software

(To be continued)

Table 3 (continuation)

Criteria	Sub-criteria	Description/Ideal
Management		A decision to use software within a project is driven by many non-functional issues such as support, adoption, etc. – non-functional issues relating to the product. Ideal: Does the product appear to have a supported, sustainable future?
	Support	The support model provided by the contributors. Ideal: Is the software supported through the community or dedicated resources?
	Sustainability	Information about the products future. Ideal: Is there a roadmap with supporting funding defining the products technical future?
	Standards	Ideal: Does the software support mainstream specifications that are standards or becoming standards?

The results of the short analysis of the tool are as follows.

There is also no clear division of the criteria into ‘internal quality’ and ‘quality in use’ criteria in this tool. We can notice that ‘Technical’ group’s sub-criteria are mainly engineering criteria, and therefore they could be analysed as ‘internal quality’ criteria, and ‘Documentation’ and ‘Management’ criteria are mainly users-related, and therefore they could be analysed as ‘quality in use’ criteria.

Many useful evaluation criteria are also missed in this model.

#### 2.4. *Comprehensive Technical Evaluation Model for Learning Object Repositories*

The Principle presented in the Introductory Section claims that there exist both ‘internal quality’ and ‘quality in use’ evaluation criteria of the software packages (such as LORs). The analysis shows that no one of the models presented in Sections 2.1–2.3 has clearly divided the LORs quality evaluation criteria into two separate groups: LORs ‘internal quality’ evaluation criteria and ‘quality in use’ criteria. Therefore it is hard to understand which criteria reflect the basic LORs quality aspects suitable for all software packages alternatives, and which are suitable only for the concrete project or user, and therefore need the users’ feedback.

While analysing the LOR quality evaluation criteria presented in Sections 2.1–2.3 we can notice that several models pay more attention to the general software ‘internal quality’ evaluation criteria (such as the ‘Architecture’ group criteria) and the several – to the ‘customisable’ ‘quality in use’ evaluation criteria groups suitable for the concrete project or user: ‘Metadata’, ‘Storage’, ‘Graphical user interface’ and ‘Other’. In conformance with the Principle, the comprehensive LOR quality evaluation model should include both general software ‘internal quality’ evaluation criteria and ‘quality in use’ evaluation criteria suitable for the particular project or user.

The authors’ proposed LOR quality evaluation model is presented in Table 4.

This tool is mostly similar to the SWITCH tool (see Table 1) in comparison with the other tools presented in Tables 2 and 3, but it also includes criteria from the other presented tools (incl. the authors’ own research). The main ideas for the constitution of this model are to clearly divide LORs quality evaluation criteria in conformance with

Table 4  
LORs technological quality evaluation criteria

Internal quality evaluation criteria	Architecture	1. Flexibility and modularity of the LOR system
		2. Possibility to use LOR system as part of a federation
		3. Performance and scalability
		4. Security
		5. Interoperability
		6. Stability
		7. Ease of deployment
		8. API for storage engine, user access rights and federation functions
		9. Coding: an inspection of the code within the software
		10. Full-text search
		11. Internationalisation
Quality in use evaluation criteria	Metadata	12. Minimal metadata schema
		13. Predefined sets of metadata
		14. Customisable metadata schema
		15. Metadata mapping for metadata search
		16. Unicode support
		17. Social tagging
		Storage
	19. Access rights	
	20. Hierarchical organization	
	21. Property and metadata inheritance	
	22. Large objects	
	Graphical user interface	23. Complete standard UI
		24. Customisable and extensible standard UI
25. Multiple standard UIs		
26. Direct distribution		
Other	27. Strength of development community	
	28. Strength of users community	
	29. LOs retrieval quality: user able retrieve LOs in different ways	
	30. Ease of installation	
	31. Accessibility, design for all	
	32. Sustainability	
	33. System administration: ability to customise look and feel	
	34. Documentation quality	

the Principle as well as to ensure the comprehensiveness of the model and to avoid the overlap of the criteria.

The overlapping criteria could be ‘Accessibility: access for all’ (it could also be included into the ‘Architecture’ group, but this criterion also needs users’ evaluation, therefore it is included into ‘Quality in use’ criteria group), ‘Full text search’ (which could also be included into the ‘Quality in use’ criteria group), and ‘Property and metadata in-

heritance' (it could also be included into the 'Metadata' group, but it deals also with the 'Storage' issues).

We have mentioned 34 different evaluation criteria in this model (set of criteria), from which 11 criteria deal with 'Internal quality' (or 'Architecture'), and 23 criteria deal with 'Quality in use'. 23 'Quality in use' criteria are divided into four groups for the probably higher quality of practical evaluation and convenience reasons. There could be different experts (programmers and users) for different groups of 'Quality in use' criteria: 'Metadata', 'Storage' and 'Graphical user interface' criteria need different kind of the evaluators' expertise.

All new models need validation. It is planned to perform the proposed LORs quality evaluation model's validation procedure in autumn 2009 in Lithuania involving three researchers and software engineering experts to validate 'Internal quality' criteria, and 12 (3 for every of 4 groups) programmers and users to validate 'Quality in use' criteria.

We expect that the advantages of the proposed model could be its comprehensiveness and the clear division of the criteria.

Therefore, this model could provide the experts of e-learning sector the clear instrumentality who (i.e., what kind of experts) should analyse what kind of LORs quality criteria in order to select the best LOR software package suitable for their needs.

### 3. Virtual Learning Environments Quality Evaluation Criteria

Now let us review and shortly analyse the literature on existing well-known VLEs evaluation tools and methods.

The main attention is paid to the sets of evaluation criteria, but several evaluation methods concerning the application of ratings (values) and weights of the evaluation criteria are also provided.

#### 3.1. Methodology of Technical Evaluation of Learning Management Systems

Methodology of Technical Evaluation of Learning Management Systems – LMSs (or VLEs) is a part of the Evaluation of Learning Management Software activity undertaken as part of the New Zealand Open Source LMS project (Technical Evaluation Criteria, 2004).

The evaluation criteria here expand on a subset of the criteria, focusing on the technical aspects of VLEs (Kurilovas, 2005):

1. Overall architecture and implementation (suitable for technical evaluation): Scalability of the system; System modularity and extensibility; Possibility of multiple installations on a single platform; Reasonable performance optimisations; Look and feel is configurable; Security; Modular authentication; Robustness and stability; Installation, dependencies and portability.
2. Interoperability (suitable for technical evaluation): Integration is straightforward; LMS/VLE standards support.

3. Cost of ownership.
4. Strength of the development community: Installed base and longevity; Documentation; End-user community; Developer community; Open development process; Commercial support community.
5. Licensing.
6. Internationalisation and localisation (suitable for technical evaluation): Localisable user interface; Localisation to relevant languages; Unicode text editing and storage; Time zones and date localisation; Alternative language support.
7. Accessibility (suitable for technical evaluation): Text-only navigation support; Scalable fonts and graphics.
8. Document transformation.

### 3.2. Open Source Platforms Adaptation Evaluation Instrument

Graf and List (2005) presents an evaluation of open source e-learning platforms/LMSs with the main focus is on adaptation issues – adaptability, personalisation, extensibility, and adaptivity capabilities of the platforms.

An e-learning course should not be designed in a vacuum; rather, it should match students' needs and desires as closely as possible, and adapt during course progression. The extended platform will be utilised in an operational teaching environment. Therefore, the overall functionality of the platform is as important as the adaptation capabilities, and the evaluation treats both issues.

The work (Graf and List, 2005) is focused on open source products only. This research is focused on customisable adaptation only, which can be done without programming skills.

These LMSs adaptation criteria are (Graf and List, 2005):

1. Adaptability – includes all facilities to customise the platform/LMS for the educational institution needs (e.g., the language or the design).
2. Personalisation aspects – indicate the facilities of each individual user to customise his/her own view of the platform.
3. Extensibility – is, in principle, possible for all open source products. Nevertheless, there can be big differences. For example, a good programming style or the availability of a documented application programming interfaces are helpful.
4. Adaptivity – indicates all kinds of automatic adaptation to the individual user's needs (e.g., personal annotations of LOs or automatically adapted content).

The evaluation (Graf and List, 2005) is based on the qualitative weight and sum approach (QWS). QWS establishes and weights a list of criteria and is based on the use of symbols. There are six qualitative levels of importance for the weights, frequently symbols are used:

1. E = Essential.
2. \* = Extremely valuable.
3. # = Very valuable.
4. + = Valuable.

5. | = Marginally valuable.

6. 0 = Not valuable.

The weight of a criterion determines the range of values that can be used to measure a product's performance. For a criterion weighted #, for example, the product can only be judged #, +, |, or 0, but not \*. This means that lower-weighted criteria cannot overpower higher-weighted criteria. To evaluate the results, the different symbols given to each product are counted. Example results can be 2\*, 3#, 3| or 1\*, 6#, 1+. The product can now be ranked according to these numbers. But the results are sometimes not clear. There is no doubt that 3\*, 4#, 2| is better than 2\*, 4#, 2| but it is not clear whether it is better than 2\*, 6#, 1+. In the latter case further analysis has to be conducted.

In Graf and List (2005) its authors have adapted the QWS approach in a way where the essential criteria are assessed in a pre-evaluation phase. These minimum criteria cover three general usage requirements: an active community, a stable development status, and a good documentation of the platform. The fourth criterion incorporates the didactical objective and means that the platform's focus is on the presentation of content instead of communication functionalities.

At the beginning of the evaluation, Graf and List (2005) have chosen 36 platforms and evaluated these according to the minimum criteria have been selected earlier. Nine platforms (ATutor 1.4.11, Dokeos 1.5.5, dotLRN 2.0.3, based on OpenACS 5.1.0, Ilias 3.2.4, LON-CAPA 1.1.3, Moodle 1.4.1, OpenUSS 1.4 extended with Freestyle Learning 3.2, Sakai 1.0, and Spaghettilearning 1.1) meet the criteria. Next, these nine platforms were tested in detail. A questionnaire and an example of a real life teaching situation, covering instructions for creating courses, managing users and simulating course activities, were designed and applied to each platform.

Finally, Graf and List (2005) established eight categories: communication tools, LOs, management of user data, usability, adaptation, technical aspects, administration, and course management.

The evaluation results of the adaptation category are presented in Table 5.

Table 5  
Platform/LMS adaptation evaluation results (Graf and List, 2005)

	Adaptability	Personalisation	Extensibility	Adaptivity	Ranking
Max. values	*	#	*	*	
ATutor		#	#		3
Dokeos		0	*	+	2
dotLRN	+	+	*	0	2
Ilias	+	#	*	0	2
LON-CAPA	+	#	#		2
Moodle	#	+	*		1
OpenUSS	#	#	#	0	2
Sakai	0	0	*	0	3
Spaghettilearning	+	#	+	0	3

Examining the results from a vertical perspective, it can be seen that the adaptability and the personalisation subcategories yield a broad range of results. The majority of the platforms were estimated as very good with regard to extensibility. In contrast, adaptivity features are underdeveloped.

As a result, Moodle can be seen as the best LMS concerning adaptation issues. Moodle provides an adaptive feature called “lesson” where learners can be routed automatically through pages depending on the answer to a question after each page. Furthermore, the extensibility is supported very well by a documented API, detailed guidelines, and templates for programming. Also adaptability and personalisation aspects are included in Moodle. Templates for themes are available and can be selected by the administrator. Students can choose out of more than 40 languages (Graf and List, 2005).

### 3.3. *Comprehensive Technical Evaluation Model for Virtual Learning Environments*

While analysing the existing VLEs evaluation methods (see Sections 3.1 and 3.2) it has been necessary to exclude all evaluation criteria that do not deal directly with VLEs technological quality problems on the one hand, and to estimate interconnected/overlapping criteria on the other.

This analysis has shown that the both analysed VLE technological evaluation methods have a number of limitations:

1. The method developed in (Technical Evaluation Criteria, 2004) practically does not examine adaptation capabilities criteria.
2. The method proposed by Graf and List (2005) insufficiently examines general technological quality criteria.

Therefore, in the authors’ opinion, a more comprehensive tool/set of criteria for VLE technological evaluation is needed. It should include general technological evaluation criteria based on modular approach and interoperability, as well as adaptation capabilities criteria (Kurilovas and Dagiene, 2009). VLE adaptation capabilities criteria should have the same weight as the other criteria.

In conformance with the Principle, the comprehensive VLEs quality evaluation model/tool should include both general software ‘internal quality’ evaluation criteria and ‘quality in use’ evaluation criteria suitable for the particular project or user.

The authors’ comprehensive set of criteria (tool) for VLEs technological evaluation proposed earlier is presented in Table 6. It is suitable for the expert evaluation of both VLEs ‘internal quality’ criteria (see criteria 1–4) and ‘quality in use’ criteria (see criteria 5–8). This tool provides the experts (decision makers) the clear instrumentality who (i.e., what kind of experts) should analyse what kind of VLEs quality criteria in order to select the best VLE software suitable for their needs.

The main ideas for the constitution of this tool are to clearly divide VLEs quality evaluation criteria in conformance with the scientific Principle as well as to ensure the comprehensiveness of the tool and to avoid the overlap of the criteria.

Table 6  
VLEs technological quality evaluation criteria (Kurilovas and Dagiene, 2009)

Internal quality (General) evaluation criteria	1. Overall architecture and implementation	Scalability	
		Modularity (of the architecture)	
		Possibility of multiple installations on a single platform	
		Reasonable performance optimisations	
		Look and feel is configurable	
		Security	
		Modular authentication	
		Robustness and stability	
		Installation, dependencies and portability	
		2. Interoperability	Integration is straightforward
			VLE standard support
		3. Internationalisation and localisation	Localisable user interface
			Localisation to relevant languages
			Unicode text editing and storage
			Time zones and date localisation
		4. Accessibility	Alternative language support
Text only navigation support			
Quality in use (Adaptation) evaluation criteria	5. Adaptability (facilities to customise for the educational institution's needs)	Scalable fonts and graphics	
		Language	
	6. Personalisation aspects (facilities of each individual user to his/her own view of the platform)	Design	
	7. Extensibility	Good programming style	
		Availability of a documented API	
	8. Adaptivity (all kinds of automatic adaptation to the individual user's needs)	Personal annotations of LOs	
		Automatically adapted content	

#### 4. Learning Software Personalisation Approaches

##### 4.1. Reusability of Learning Objects and Customisation of Metadata

This Section is aimed to present one of the methods of customisation of the LOs metadata schema.

In the authors' point of view, one of the main criteria for achieving high LOs effectiveness and personalisation level is LOs reusability (Dagiėnė and Kurilovas, 2008).

The need for reusability of LOs has at least three elements:

1. Interoperability: LO is interoperable and can be used in different platforms.
2. Flexibility in terms of pedagogic situations: LO can fit into a variety of pedagogic situations.

3. Modifiability to suit a particular teacher's or student's needs: LO can be made more appropriate to a pedagogic situation by modifying it to suit a particular teacher's or student's needs (McCormick *et al.*, 2004).

There are two main conditions for LOs reusability elsewhere (Kurilovas, 2009):

1. LOs have to fit different countries national curricula.
2. Different countries' IEEE Learning Object Metadata (LOM) standard's Application Profiles (APs) have to be oriented towards quick and convenient search of reusable LOs.

The principle of ultimate increase of reusability of LOs is considered by the authors as one of the main factors of e-learning systems flexibility (Dagienė and Kurilovas, 2008).

It was analysed that the flexible approach to the e-learning system's creation and development should be based on the idea of LOs' partition to two main separate parts, i.e., LOM compliant small pedagogically decontextualised Learning Assets as well as LOM and IMS Learning Design compliant Units of Learning – UoLs (Dagiene and Kurilovas, 2007; Kurilovas and Kubilinskiene, 2007; 2008).

European Learning Resource Exchange (LRE) system's (LRE, 2009) validation in Lithuania performed by the authors while implementing FP6 CALIBRATE project (CALIBRATE, 2008) has shown that the teachers prefer LOs from national repositories which have the potential to 'travel well' and can be used in different national contexts. These reusable LOs preferred by the teachers are mainly the small decontextualised learning assets. Therefore in order to maximise LOs reusability in Europe LRE should consist mainly of the decontextualised learning assets (Kurilovas and Kubilinskiene, 2007; 2008).

The results of the teachers-experts survey also performed by the authors in CALIBRATE show that the teachers would mostly like to find pedagogically decontextualised ultimately reusable LOs and therefore to have a service for quick and convenient search of such LOs.

While searching for LOs in CALIBRATE/LRE portal the experts have used browsing by subject and advance search services. These advance search services have not contained any services to ease the search of reusable LOs in the portal. The LOs in the portal are described according to the partners' LOM APs, and these APs have not contained any services to simplify the search of reusable LOs. Therefore it took very much time for the experts to find and choose suitable reusable LOs for their UoLs (e.g., lesson plans).

According to Kurilovas (2009), the analysis of the existing and emerging interoperability standards and specifications shows that:

- The majority of standards and specifications are not adopted and do not conform to the educational practice.
- There exists a problem of complex solutions for the application of standards and specifications in education.
- Standards and specifications often do not cooperate.

First of all, in order to make it easier for educators to discover and use LOs that addresses the needs of their students, to maximise reuse of LOs and minimise the costs

associated with their repurposing, the good solutions are lacking for the specific application profiles of IEEE LOM (Kurilovas, 2009).

According to Duval *et al.* (2002), the purpose of an AP is to adapt or combine existing schemas into a package that is tailored to the functional requirements of a particular application, while retaining interoperability with the original base schemas. There are several principles described in Duval *et al.* (2002) providing “a guiding framework for the development of practical solutions for semantic and machine interoperability in any domain using any set of metadata standards”: modularity, extensibility, refinement and multilingualism.

One of the mechanisms for APs to achieve modularity is the elements’ cardinality enforcement. Cardinality refers to constraints on the appearance of an element. Is it mandatory or recommended or optional? According to Duval *et al.* (2002), “the status of some data elements can be made more stringent in a given context”. For instance, an optional data element can be made recommended, and a recommended data can be made mandatory in a particular AP. On the other hand, as an AP must operate within the interoperability constraints defined by the standard, it cannot relax the status of data elements (Duval *et al.*, 2002).

The authors have applied this cardinality enforcement principle in their research. It was analysed that the main LOM elements which vocabulary values could reflect the LOs ultimate reusability deal with structure of LO, its functional granularity (aggregation) level, educational type as well as the kind of relation of this LO with the others (Kurilovas and Kubilinskiene, 2007; 2008).

The results of the authors’ analysis of the last European LOM AP (LRE Metadata AP v3.0) have shown that it would be purposeful to improve it in order to provide more quick and convenient search possibilities for those searching ultimately reusable LOs (i.e., learning assets) by the means of changing (i.e., advancing/enforcing cardinality) the status of a number of LRE AP elements.

These proposals deal with changing the status of the following LOM AP elements from ‘optional’ to ‘recommended’ as well as from ‘optional’ and ‘recommended’ – to ‘mandatory’:

- 1.7 General. Structure;
- 1.8 General. Aggregation Level;
- 5.2 Educational. Learning Resource Type; and
- 7.1 Relation. Kind (see Fig. 1).

These elements should be included in the advanced search engine for those looking for reusable LOs to use them as ‘building blocks’ in their own lesson plans, modules or courses. The authors believe that the development of advanced search engine reflecting LOs reusability level based on this research would considerably reduce the time for the users to find and choose suitable LOs in the repositories.

There are more methods of personalisation of LOs metadata. They could be, e.g., based on the customisation of controlled vocabularies, implementation of the learners’ profiles or users’ tags to search for preferred LOs in the repositories.

The extended search and management of controlled vocabularies by desirable elements are also often implemented in the LORs to enhance the customisation of LOs for

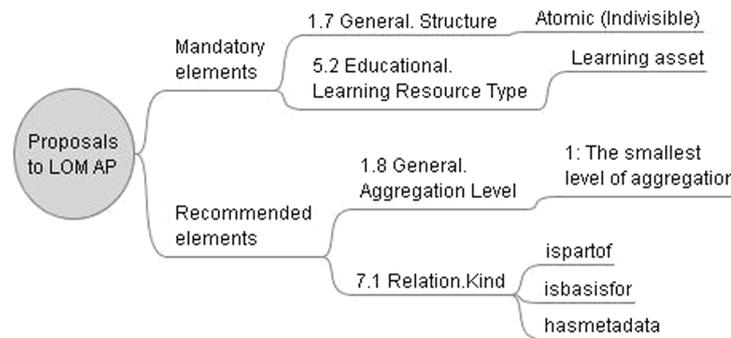


Fig. 1. Proposals on customisable metadata schema (Kurilovas, 2009).

the personal users needs. These possibilities are already implemented in the centralised LO metadata repository for general and vocational education in Lithuania. The teachers and learners can also use the other users' comments on the LOs in this repository and look at the LOs popularity based on the statistics of LOs downloads (Kurilovas and Kubilinskiene, 2008).

The detailed analysis of these approaches is out of the scope of this paper.

#### 4.2. Ratings of the Quality Evaluation Criteria

There are a number of the methods to explore the learning software customisation possibility level.

The authors propose to use the multiple criteria evaluation Method of the learning software quality expressed by the experts' utility function presented further in Section 4.3 and including the alternatives' evaluation criteria, their ratings (values) and weights.

The evaluation criteria used in this method should conform to the software engineering Principle based on the evaluation criteria division to 'internal quality' and 'quality in use' criteria.

Scientists who have explored quality of software consider that there exists no simple way to evaluate functionality characteristics of internal quality of software. According to Gasperovic and Caplinskas (2006), it is a hard and complicated task, which requires relatively high time and labour overheads. According to Zavadskas and Turskis (2008), each alternative in multi-criteria decision making problem can be described by a set of criteria. Criteria can be qualitative and quantitative. They usually have different units of measurement and different optimisation direction.

The comprehensive sets of evaluation criteria suitable for the expert multiple criteria evaluation (decision making) of LORs and VLEs have been proposed earlier in the Tables 4 and 6.

According to the multiple criteria evaluation method, we also need LORs and VLEs evaluation criteria ratings (values).

The widely used measurement criteria of the decision attributes' quality are mainly qualitative and subjective. Decisions in this context are often expressed in natural lan-

Table 7  
Linguistic variables conversion into triangular fuzzy numbers (TFNs)

Linguistic variables	TFN
Excellent	(0.700, 0.850, 1.000)
Good	(0.525, 0.675, 0.825)
Fair	(0.350, 0.500, 0.650)
Poor	(0.175, 0.325, 0.475)
Bad	(0.000, 0.150, 0.300)

Table 8  
Linguistic variables and QWS symbols conversion into non-fuzzy values  $E$

Linguistic variables and QWS symbols	Non-fuzzy values $E$
Excellent (E or *)	0.850
Good (#)	0.675
Fair (+)	0.500
Poor (l)	0.325
Bad (0)	0.150

guage, and evaluators are unable to assign exact numerical values to the different criteria. Assessment can be often performed by linguistic variables: ‘bad’, ‘poor’, ‘fair’, ‘good’ and ‘excellent’. These values are imprecise and uncertain: they are commonly called fuzzy values. Integrating these different judgments to obtain a final evaluation is not evident.

Therefore, Ounaies *et al.* (2009) propose to use fuzzy group decision making theory to obtain final assessment measures.

First, linguistic variable values are mapped into triangular fuzzy numbers  $(l, m, u)$  (see Table 7).

After the defuzzification procedure which converts the global fuzzy evaluation results, expressed by a  $TFN(l, m, u)$ , to a non-fuzzy value  $E$ , the following equation has been adopted by Ounaies *et al.* (2009):

$$E = [(u - 1) + (m - 1)]/3 + 1, \quad (1)$$

The non-fuzzy values  $E$  for all aforementioned linguistic variables calculated according to the Eq. (1) are presented in Table 8.

#### 4.3. Experts' Additive Utility Function

If we want to evaluate (or optimise) the technological quality of learning software (e.g., VLEs) for the particular learner needs (i.e., to personalise his/her learning process in the best way in conformance with the prerequisites, preferred learning speed and methods,

etc.), we should use the experts' additive utility function together with the weights of evaluation criteria.

The weight of the evaluation criterion reflects the experts' opinion on the criterion's importance level in comparison with the other criteria for the individual learner/user.

For example, for the most simple (general) case, when all VLE evaluation criteria are of equal importance, the experts could consider the equal normalised weights  $a_i = 0.125$  agreeably to the normalisation requirement

$$\sum_{i=1}^m a_i = 1, \quad a_i > 0, \quad (2)$$

for the VLEs quality evaluation criteria  $i = \{1, \dots, 8\}$  (see Table 6).

A possible decision could be to transform multi-criteria task into one-criterion task obtained by adding all criteria together with their weights. It is valid from the point of view of the optimisation theory, and a special theorem exists for this case.

Therefore here we have the experts' additive utility function:

$$f(X) = \sum_{i=1}^m a_i f_i(X), \quad \sum_{i=1}^m a_i = 1, \quad a_i > 0, \quad (3)$$

where  $f_i(X_j)$  is the rating (non-fuzzy value  $E$ ) (see Table 8) of the criterion  $i$  for the each of the three examined alternatives  $X_j$ :  $X_1$  – ATutor,  $X_2$  – Ilias, and  $X_3$  – Moodle.

Here  $i$  are the order numbers of the VLE quality evaluation criteria presented in Table 6. First four of these criteria are general 'internal quality' VLE quality criteria, and the other four – VLE adaptation 'quality in use' criteria (see Table 6).

The major is the meaning of the utility function (3) the better VLE meets the particular learner needs.

#### 4.4. Example of Evaluation of Virtual Learning Environments

In the general case all VLE evaluation criteria are of equal importance. The values of the function (3) where the non-fuzzy values  $E$  for all variables in Table 8 are calculated agreeably to the Eq. (1), and all VLE evaluation criteria are of equal importance, are presented in Table 9. According to the normalisation requirement (2), all  $a_i = 0.125$ .

These results mean that VLE Moodle meets 60.93% quality in comparison with the ideal (it is less than 'good'), ATutor – 54.37% (it is more than 'fair'), and Ilias – 50.00% (i.e., 'fair'). According to this experimental evaluation results, VLE Moodle is the best alternative (among the evaluated) from technological point of view in general case. This alternative has shown the highest ratings of both 'internal quality' evaluation (see General criteria ratings) and 'quality in use' evaluation (see Adaptation criteria ratings).

In more specific cases, e.g., if the experts (decision makers) would like to select the most suitable VLE for the students with special education needs/disabilities, they should choose higher weights for the particular criteria: Accessibility (e.g., measuring weight  $a_4 = 0.2$ ) and Personalisation (e.g., measuring weight  $a_6 = 0.2$ ).

Table 9  
VLEs technological evaluation summary

Technological evaluation criteria	ATutor	Ilias	Moodle
<i>General criteria ratings</i>			
Architecture and implementation	0.500	0.325	0.850
Interoperability	0.675	0.675	0.500
Internationalisation and localisation	0.325	0.500	0.675
Accessibility	0.850	0.325	0.500
<i>Interim rating</i>	<i>2.350</i>	<i>1.825</i>	<i>2.525</i>
<i>Adaptation criteria ratings</i>			
Adaptability	0.325	0.500	0.675
Personalisation	0.675	0.675	0.500
Extensibility	0.675	0.850	0.850
Adaptivity	0.325	0.150	0.325
<i>Interim rating</i>	<i>2.000</i>	<i>2.175</i>	<i>2.350</i>
<i>Total evaluation rating</i>	<i>4.350</i>	<i>4.000</i>	<i>4.875</i>
<i>f(X) (weights = 0.125)</i>	<i>0.5437</i>	<i>0.5000</i>	<i>0.6093</i>

Table 10  
VLEs technological evaluation summary for the learners with special needs (using different weights)

Technological evaluation criteria	ATutor	Ilias	Moodle
<i>General criteria ratings</i>			
Architecture and implementation $a_1 = 0.1$	0.0500	0.0325	0.0850
Interoperability $a_2 = 0.1$	0.0675	0.0675	0.0500
Internationalisation and localisation $a_3 = 0.1$	0.0325	0.0500	0.0675
Accessibility $a_4 = 0.2$	0.1700	0.0650	0.1000
<i>Interim rating</i>	<i>0.3200</i>	<i>0.2150</i>	<i>0.3025</i>
<i>Adaptation criteria ratings</i>			
Adaptability $a_5 = 0.1$	0.0325	0.0500	0.0675
Personalisation $a_6 = 0.2$	0.1350	0.1350	0.1000
Extensibility $a_7 = 0.1$	0.0675	0.0850	0.0850
Adaptivity $a_8 = 0.1$	0.0325	0.0150	0.0325
<i>Interim rating</i>	<i>0.2675</i>	<i>0.2850</i>	<i>0.2850</i>
<i>Total evaluation rating f(X) (different weights)</i>	<i>0.5875</i>	<i>0.5000</i>	<i>0.5875</i>

All the other criteria weights according to the normalisation formula (2) should be measured  $a_i = 0.1$ . In this personalised case the experts should find that, differently from the simple general case (see Table 9), both ATutor and Moodle are optimal alternatives for the learners with special needs (see Table 10).

These results mean that VLE ATutor and Moodle meet 58.75% quality in comparison with the ideal for special needs students (if is something between ‘fair’ and ‘good’), and Ilias – 50.00% (it corresponds the linguistic variable ‘fair’).

If we want to select, e.g., the most suitable LOR for the students with special education needs/disabilities, we should choose the higher weights for the particular ‘quality in use’ criteria such as ‘Customisable metadata schema’ (e.g., measuring the weight  $a_{14} = 0.05$ ), ‘Customisable and extensible standard UI’ ( $a_{24} = 0.05$ ), ‘Accessibility’ ( $a_{31} = 0.06$ ), and ‘Ability to customise look and feel’ ( $a_{33} = 0.06$ ) (see Table 4). In this case all the other criteria weights according to normalisation requirement (2) should be  $a_i = 0.026$ . The choice of the particular values of the weights usually depends on the experts (decision-makers).

In this case, if we would apply the formula (3), we would find out that Fedora is the optimal LOR for the users with special needs in comparison with DSpace and EPrint LOR packages due to its modular approach, metadata schema is extensible without restrictions, all UI-projects which are open source and can be adapted, its high ability to customise look and feel, etc.

Such approach has never been applied for solving the learning software evaluation and optimisation tasks before.

#### 4.5. Minimisation of the Experts Subjectivity

Another very complicated problem for such multiple criteria evaluation and optimisation tasks is minimisation of the experts’ (decision makers’) subjectivity. The experts’ subjectivity can influence the quality criteria ratings (values) and their weights.

There are some scientific approaches concerning this item.

One of them is formulated in Kendall (1979). According to Kendall (1979), in general, the experts influence importance is different, and therefore this importance should be assessed using the appropriate methodology. It is important to form the experts group purely by their competence. Furthermore, according to Kendall (1979), we should eliminate the extreme experts’ assessments of the ratings and weights. In order to pursue the compatibility of the experts’ assessments we should calculate so-called concordance rates  $W$  and distributions  $\lambda^2$ :

$$W = \frac{12S}{r^2(m^3 - m)},$$

where  $r$  – the number of experts;  $m$  – the number of the parameters under evaluation;  $S$  – the square sum of evaluated importance rates’ values deviations from the experts’ aggregate average.

In its turn,

$$\lambda^2 = Wr(m - 1) = \frac{12S}{rm(m + 1)}.$$

The compatibility of the experts’ assessments is considered sufficient if the value of concordance rate  $W$  is 0.6–0.7 (Kendall, 1979).

## 5. Conclusion and Results

Personalisation of learning content and software could be enhanced by the presented LOs metadata customisation method and the Method of multiple criteria evaluation of learning software such as LORs and VLEs.

The proposed LORs and VLEs multiple criteria evaluation Method represented by the experts' additive utility function is based on the transformation of the multiple criteria task into the one-criterion task obtained by adding all criteria ratings (values) together with their weights.

This Method provides the clear instrumentality how to choose suitable learning software for the personalised learners needs (e.g., for students with special education needs).

This Method together with the experts' subjectivity minimisation approaches is suitable to apply for the LORs and VLEs practical expert evaluation to meet the particular learner needs. Therefore, it is of practical importance for public and private sectors' experts (decision makers), software engineers, programmers and users.

## References

- CALIBRATE (2008). FP6 IST CALIBRATE (Calibrating eLearning in Schools) project web site. <http://calibrate.eun.org>
- Dagiene, V., Kurilovas, E. (2007). Design of Lithuanian digital library of educational resources and services: The problem of interoperability. *Information Technologies and Control*, 36(4), 402–411.
- Dagiene, V., Kurilovas, E. (2008). *Information Technologies in Education: Experience and Analysis*. Institute of Mathematics and Informatics, Vilnius (in Lithuanian).
- Duval, E., Hodgins, W., Sutton, S., Weibel, S.L. (2002). Metadata principles and practicalities. *D-Lib Magazine*, 8(4). <http://www.dlib.org/dlib/april02/weibel/04weibel.html>
- Dzemyda, G., Saltenis, V. (1994). Multiple criteria decision support system: Methods, user's interface and applications. *Informatica*, 5(1–2), 31–42.
- EU eContent plus Programme's EdReNe (Educational Repositories Network) Project Web Site (2009). <http://edrene.org/>
- Gasperovic, J., Caplinskas, A. (2006). Methodology to evaluate the functionality of specification languages. *Informatica*, 17(3), 325–346.
- Graf, S., List, B. (2005). An evaluation of open source e-learning platforms stressing adaptation issues. Presented at *ICALT 2005*.
- IMI (2005). *Institute of Mathematics and Informatics. Research on Digital Learning Tools and Virtual Learning Environments Implementation in Vocational Education*. Scientific Research Report (in Lithuanian). <http://www.emokykla.lt/lt.php/tyrimai/194>
- Kendall, M. (1979). *Rank Correlation Methods*. Griffin and Co, London.
- Kurilovas, E. (2005). Several aspects of technical and pedagogical evaluation of virtual learning environments. *Informatics in Education*, 4(2), 215–252.
- Kurilovas, E. (2007). Digital library of educational resources and services: Evaluation of components. *Informacijos mokslai*, 42–43, 69–77.
- Kurilovas, E. (2009). Interoperability, standards and metadata for e-learning. In: *Proceedings of the 3rd International Symposium on Intelligent Distributed Computing*, Agia Napa, Cyprus, October 12–14, 2009. *Studies in Computational Intelligence*. Springer.
- Kurilovas, E., Dagiene, V. (2009). Learning objects and virtual learning environments technical evaluation criteria. *Electronic Journal of e-Learning*, 7(2), 127–136. Available online at [www.ejel.org](http://www.ejel.org)
- Kurilovas, E., Kubilinskienė, S. (2007). Creation of Lithuanian digital library of educational resources and services: The hypothesis, contemporary practice, and future objectives. In: *Proceedings of the 1st International Workshop on Learning Object Discovery & Exchange (LODE'07) within the 2nd European Conference*

- on *Technology Enhanced Learning (EC-TEL'07)*, Sissi, Crete, Greece, 17–20 September, 2007, Vol. 311, pp. 11–15. <http://CEUR-WS.org/Vol-311/>
- Kurilovas, E., Kubilinskienė, S. (2008). Interoperability framework for components of digital library of educational resources and services. *Informacijos mokslai*, 44, 88–97.
- LRE (2009). European Learning Resource Exchange service for schools web site. <http://lreforschools.eun.org/LRE-Portal/Index.iface>
- McCormick, R., Scrimshaw, P., Li, N., Clifford, C. (2004). *CELEBRATE Evaluation Report*. [http://celebrate.eun.org/eun.org2/eun/Include\\_to\\_content/celebrate/file/Deliverable7\\_2EvaluationReport02Dec04.pdf](http://celebrate.eun.org/eun.org2/eun/Include_to_content/celebrate/file/Deliverable7_2EvaluationReport02Dec04.pdf)
- Newhouse, S. (2005). *Software Repository – Evaluation Criteria and Dissemination*. Open Middleware Infrastructure Institute, University of Southampton. <http://www.omii.ac.uk/dissemination/EvaluationCriteria.pdf>
- Ounaies, H.Z., Jamoussi, Y., Ben Ghezala, H.H. (2009). Evaluation framework based on fuzzy measured method in adaptive learning system. *Themes in Science and Technology Education*, 1(1), 49–58.
- SWITCH collection (2008). *The National Learning Object Repository Project Website*, Switzerland. <http://www.switch.ch/it/els/collection/evaluation.html>
- Technical Evaluation of Selected Open Source Repository Solutions* (2006). <http://eduforge.org/docman/view.php/131/1062/Repository%20Evaluation%20Document.pdf>
- Technical Evaluation of Selected Learning Management Systems* (2004). <https://eduforge.org/docman/view.php/7/18/LMS%20Technical%20Evaluation%20-%20May04.pdf>
- Wiley, D.A. (2000). *Connecting Learning Objects to Instructional Design Theory: a definition, a Metaphor, and a Taxonomy*. Utah State University. <http://www.reusability.org/read/>
- Zavadskas, E.K., Turskis, Z. (2008). A new logarithmic normalization method in games theory. *Informatica*, 19(2), 303–314.

**E. Kurilovas** is a doctor of technological sciences (informatics engineering). He is a research scientist in the Institute of Mathematics and Informatics, an associate professor in Vilnius Gediminas Technical University, and the head of International Networks Department of the Centre for IT in Education of the Ministry of Education and Science of Lithuania. The main areas of his scientific interest are: technology-enhanced learning problems analysis; e-learning systems architecture; metadata, interoperability and standards; usability of learning objects, their repositories and virtual learning environments; e-learning systems' components quality evaluation problems; personalisation and optimisation of the learning process.

**S. Šėrikovienė** has graduated from the Kaunas University of Technology as a specialist in the applications of mathematics and obtained the degree of master of mathematics and informatics in 1998. Since 2007 she is the Deputy Director of Panevezys Juozas Miltinis Secondary School, and since 2000 – a lecturer in Panevezys branch of Kaunas University of Technology. She is PhD student of informatics engineering in Institute of Mathematics and Informatics and Vilnius Gediminas Technical University since 2008. Her present research object is the analysis of learning software optimisation problems.

## **Mokomosios programinės įrangos vertinimo ir personalizavimo problemos**

Eugenijus KURILOVAS, Silvija ŠĖRIKOVIENĖ

Straipsnyje yra analizuojami keli mokslo požiūriai į mokomojo turinio ir mokomosios programinės įrangos vertinimą, parinkimą ir personalizavimą konkrečių naudotojų ir/ar besimokančiųjų poreikiams. Detaliau analizuojamas mokymosi objektų metaduomenų personalizavimo metodas, o taip pat mokomosios programinės įrangos daugiakriterinio vertinimo ir optimizavimo Metodas, kuris yra išreikštas ekspertų naudingumo funkcija. Ekspertų naudingumo funkcijos reikšmė priklauso nuo mokomosios programinės įrangos kokybės vertinimo kriterijų parinkimo, jų įverčių ir svorių nustatymo. Pagrindinė šio Metodo dalis yra informatikos inžinerijos Principas kuris nusako, kad mokomosios programinės įrangos kokybės vertinimui reikia naudoti dvi skirtingas kokybės kriterijų grupes – ‘vidinės kokybės’ kriterijus apibrėžiančius bendruosius programinės įrangos kokybės aspektus, ir ‘naudojimo kokybės’ kriterijus apibrėžiančius programinės įrangos personalizavimo galimybes. Šio Metodo ir Principo taikymas mokomosios programinės įrangos vertinimui technologijomis grįsto mokymosi teorijoje ir praktikoje yra naujoviškas. Ekspertų subjektyvumo mažinimo metodo taikymas išanalizuotas straipsnyje taip pat yra naujas aspektas technologijomis grįsto mokymosi moksle. Visi aukščiau minėti mokslo požiūriai siūlo efektyvias praktines priemones personalizuotam mokomajam turiniui ir mokomajai programinei įrangai sukurti ir/ar parinkti.