Assessing the Learning Path Specification: a Pragmatic Quality Approach

José Janssen
(Open University in the Netherlands, Heerlen, The Netherlands
jose.janssen@ou.nl)

Adriana J. Berlanga
(Open University in the Netherlands, Heerlen, The Netherlands
adriana.berlanga@ou.nl)

Stef Heyenrath
(Logica, Maastricht, The Netherlands
stef.heyenrath@logica.com)

Harry Martens
(Open University in the Netherlands, Heerlen, The Netherlands
harry.martens@ou.nl)

Hubert Vogten
(Open University in the Netherlands, Heerlen, The Netherlands
hubert.vogten@ou.nl)

Anton Finders
(Open University in the Netherlands, Heerlen, The Netherlands
anton.finders@ou.nl)

Eelco Herder
(L3S Research Centre, Hannover, Germany
herder@l3s.de)

Henry Hermans
(Open University in the Netherlands, Heerlen, The Netherlands
henry.hermans@ou.nl)

Javier Melero Gallardo
(Universitat Pompeu Fabra, Barcelona, Spain
javier.melero@upf.edu)

Leon Schaeps
(Open University in the Netherlands, Heerlen, The Netherlands
leon.schaeps@ou.nl)

Rob Koper
(Open University in the Netherlands, Heerlen, The Netherlands
rob.koper@ou.nl)

Abstract: Finding suitable ways to achieve particular learning goals is not an easy task, both in initial education and lifelong learning. To facilitate selection, personalisation and navigation of learning paths we propose to describe learning paths in a formal and uniform way by means of a learning path specification. This paper explains the rationale behind the Learning Path Specification. Based on a framework developed for the evaluation of the specification the paper describes a study that was carried out to establish pragmatic quality, i.e. whether stakeholders can understand and use the specification. The paper explores the relationship between the concepts *pragmatic quality*, *usability*, and *desirability*, and distinguishes first-order and second-order pragmatic quality, relating it to different stakeholders: software developers and end-users. First-order pragmatic quality of the Learning Path Specification was evaluated during the process of developing a tool that describes learning paths according to the specification: the Learning Path Editor. Second-order pragmatic quality was evaluated through workshop sessions with end-users involving some hands-on experiences with this tool. The paper describes adaptations made to the specification in the process of developing the Editor. End-user evaluations were quite positive, leading to one more adaptation.

Keywords: learning path specification, evaluation, pragmatic quality, usability, desirability
Categories: L.3, L.3.0, L.3.6

1 Rationale Behind the Learning Path Specification

Learning paths are defined as sets of one or more learning actions that lead to a particular learning goal. These learning actions can be formal, non-formal, informal or a combination of these and can vary from a relatively small activity like reading a book or taking a course to following an entire programme or curriculum [Janssen et al. 2008a, Janssen et al. submitted].

The number of learning opportunities available to lifelong learners has greatly increased in recent decades: educational institutions traditionally focusing on initial education have made a shift to target lifelong learners as well, the training market has expanded and more and more courses have become available through the Internet. Especially when learners seek to develop skills or gain knowledge in a relatively unknown field or when they are faced with numerous ways to learn something, they need help to chose a suitable way to reach their learning goals [Chen et al. 2006, Lea et al. 2003]. This problem exists not only in formal education, where increased modularization necessitates navigation support [Kilpatrick et al. 2007, Simpson 2004, Yorke 1999], but also in non-formal and informal learning [van der Klink et al. 2009]. The following example will illustrate the problem: a person who is interested in interior design and who would like to develop her competences in this direction might have a look to see what courses are available, for instance through a search on Internet. Deciding upon a course means that a particular learning path is chosen. The search entry ‘interior design course’ in Google presently (February 2010) results in over 70 million hits, referring to all kinds of interior design courses, at varying levels, some accredited others not, with different price tags attached, with varying study loads, etcetera. This clearly represents a case of information overload, even if to a novice course titles might offer a clue through words like ‘introduction’ or ‘basics’.

Comparison and selection of suitable learning paths can be facilitated if learning paths are described in a formal and interoperable way: i.e. a way that specifies a learning path’s constituent parts and characteristics in a language that is amenable to computer processing and intelligible across systems. Such a *Learning Path Specification* was developed within the context of the European FP6 TENCompetence Integrated Project [TENCompetence 2005]. Requirements for the specification were derived from a review of literature on the nature of formal and informal learning as well as an analysis of current practises that aim to support learner choice (Janssen et al., 2008a).

The Learning Path Specification distinguishes itself from related specifications in the field, which also aim at supporting learners in finding suitable learning opportunities but which focus on formal education [CDM 2004, CEN 2008, XCRI 2006]. The Pspex project [Oussena and Barn 2009] also focuses on formal programmes, more specifically on the rules for programme assembly and versions. In an earlier publication [Janssen et al. 2008a] we explained how the Learning Path Specification has clear links with the IMS-LD specification [IMS-LD 2003]. However IMS-LD provides a detailed description of the activities, assignments and materials involved in the learning process, whereas the Learning Path Specification is merely a vehicle to describe and connect learning actions, which might in fact be an IMS-LD unit-of-learning, but could also be a workshop, a manual, a video, a classroom course, a blog, etc.

In order to assess the quality of the Learning Path Specification an evaluation framework was developed, based on theories and research regarding conceptual model quality [Janssen et al. submitted]. The framework starts from the familiar distinction between *syntactic*, *semantic,* and *pragmatic* quality of conceptual models [Leung an Bolloju 2005]. The work presented in this paper focuses on the evaluation of *pragmatic quality* of the Learning Path Specification: how easy is it for stakeholders to understand and use the specification?

The remaining sections of this paper are structured as follows. [Section 2] of this paper briefly describes the Learning Path Specification. [Section 3] zooms in on pragmatic quality as a particular dimension of conceptual model quality and describes the connection between pragmatic quality and the related concepts of usability, and desirability. [Section 4] describes the method used to evaluate the pragmatic quality of the Learning Path Specification, including a description of the Learning Path Editor. [Section 5] describes the results of the study. Finally, [section 6] describes the conclusions and discusses the implications of the results for future work.

2 Model and Model Quality

The purposes of conceptual models can vary widely: enhance communication, document the current state of knowledge, guide system development, exploration, prediction, decision support [Beck 2002, Moody 2005]. The ‘immediate’ purpose of the learning path specification is to support the following processes:

1. Description of lifelong learning paths
2. Selection of suitable learning paths
3. Personalisation of learning paths (taking into account learners’ entry levels)
4. Navigation of learning paths (i.e. following the designated steps)

2.1 Learning Path Conceptual Model

The conceptual model of the Learning Path Specification [Janssen et al. 2008b] is shown in [Figure 1]. A learning path has a Start (prerequisites) and a Finish (learning goals) which are defined in terms of (a set of one or more) competences and associated levels of proficiency (CompetenceLevel). Competence is defined as the ability of a person to act effectively and efficiently in a particular situation, e.g. performing a job, hobby, sport, etc. Whereas specification of the path's finish is mandatory, specification of prerequisite competence levels by defining a start is optional. Both start and finish could be as elaborate as a job profile.

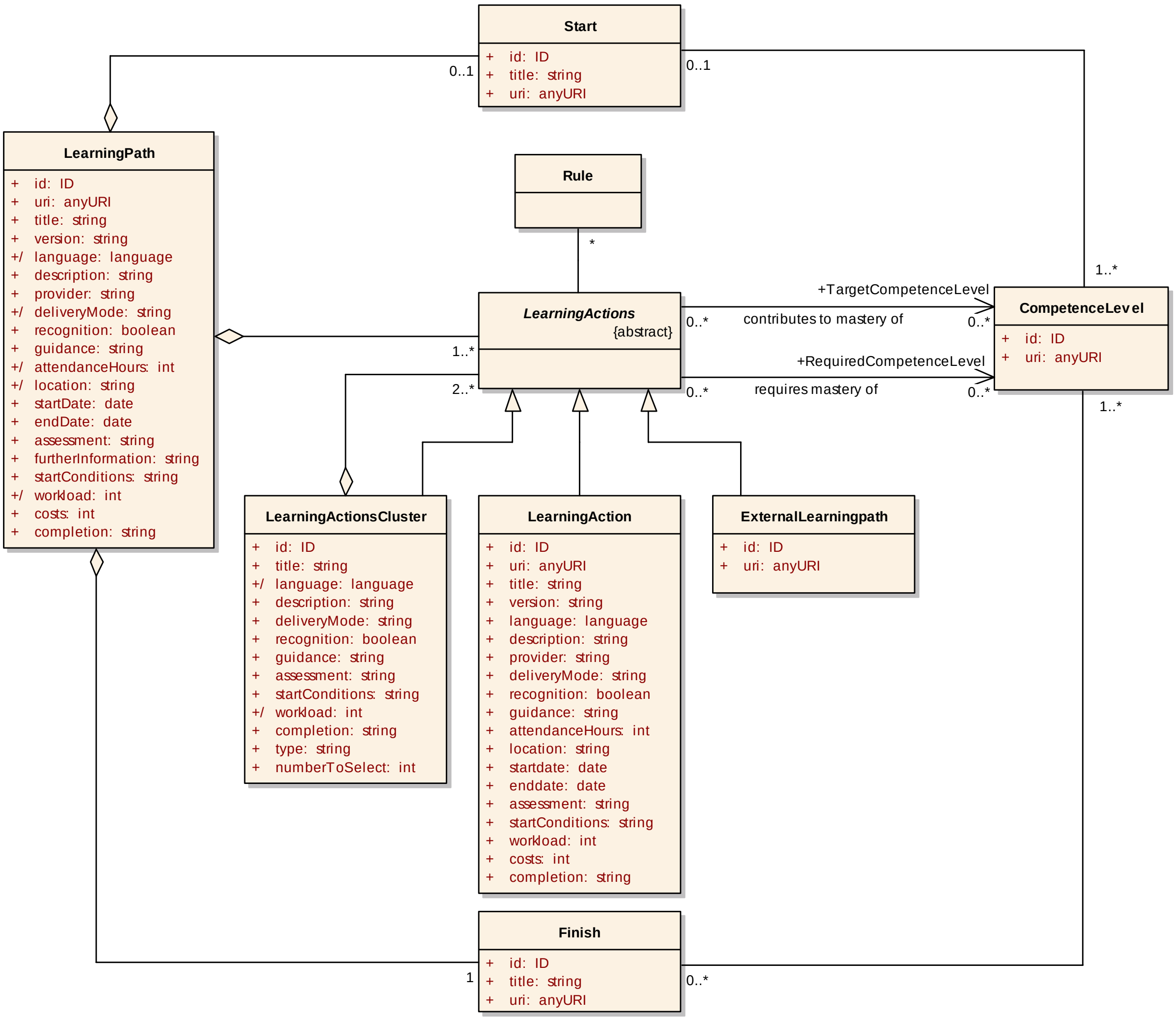

Figure 1: Learning path conceptual model

A learning path further defines the steps (LearningActions) that lead from the start to the finish, i.e. to attainment of specific competences at specific levels. These steps may involve:

- a single learning action (LearningAction: ‘workshop X’, ‘course Y’, ‘consult expert Z’, ‘read A’)
- a cluster of learning actions which are related (LearningActionsCluster: ‘chose one of the following actions’, ‘perform the following actions sequentially’)
- a reference to an existing learning path (LearningPathRef: this enables nested structures of learning paths, e.g. in formal settings one leading towards the Bachelor degree and the other leading to the Masters degree).

Each learning action may contribute to mastery of one or more competences and may require mastery of one or more competences at a particular level. Note that the relation between learning actions and competence levels is a very loose one: possibly even non-existent. This was done deliberately in order to avoid the specification to become too rigid. The methodical description of competences and associated levels of proficiency has deliberately been put out of scope for the Learning Path Specification. The model assumes that competences and their levels are described elsewhere in a standardised way that can be referenced [Kickmeier-Rust et al. 2006, TENCompetence 2006, Van Assche 2007]. A learning path is further described by a set of metadata that specify content, process, and planning information (e.g. title, description, assessment, tutoring, delivery mode, attendance hours), which are relevant to the process of choosing a learning path.

The learning path conceptual model presented in [Figure 1] was created using the Unified Modelling Language [UML 1997] for graphical representation in order to facilitate communication about the model. The implementation of the model in a binding was realised using XML, Extensible Markup Language [W3C 2008], so as to meet technical requirements of formality and interoperability.

The model was designed to meet a number of requirements related to the characteristics of learning paths that should be taken into account e.g. modular composition, nested composition, learning outcomes, optional parts, sequencing etc [Janssen et al. 2008a].

When learning paths and learning actions are described as proposed by the specification it becomes possible to support selection, navigation, and personalisation of learning paths in a (semi-) automated way: search engines can be developed that enable learners to specify criteria for the selection of suitable learning paths (e.g. costs, start date, delivery mode, location), learning paths can be automatically visualised (optional and mandatory parts, fixed orders) in support of navigation, and learning paths can be personalised by identifying learning actions as ‘completed’ when the learner already has attained the associated competence levels through prior learning.

[Figure 2] describes the processes to be supported by the specification, the required tools, and their outputs. The numbers of the tools pictured in the figure correspond with the numbering of the processes described earlier: 1. description, 2.

selection, 3. personalisation, and 4. navigation of learning paths. As [Figure 2] illustrates, the description of learning paths precedes all the other processes (search, personalisation, and navigation) as these follow-up processes require the availability of learning path descriptions in line with the specification. Hence, for the evaluation of pragmatic quality a tool that enables description of learning paths in line with the specification was the most likely choice.

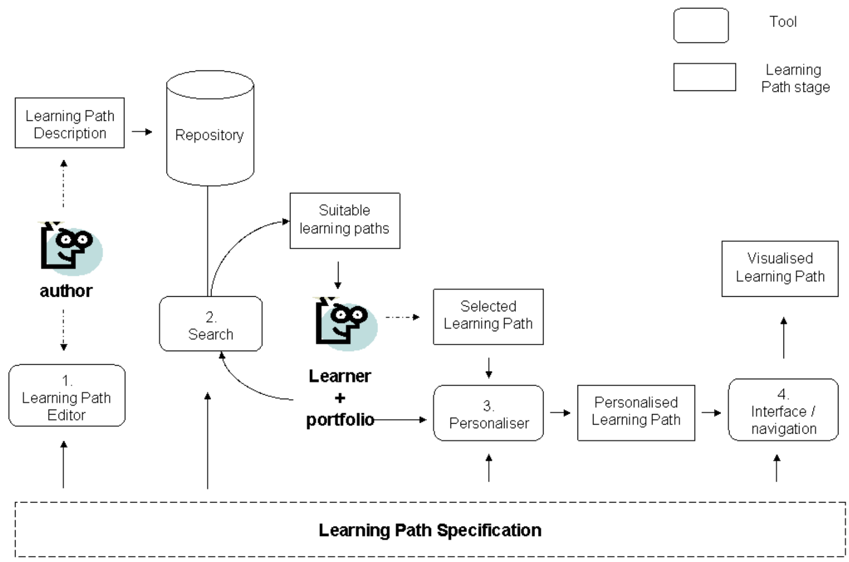

Figure 2: Tools building on the learning path specification

2.2 Model Quality

The framework developed for the evaluation of the learning path specification [Janssen et al. submitted] identifies three aspects of model quality:

1. Semantic quality: does the model represent essential features of the phenomenon under study?
2. Syntactic quality: does the model express what it intends to express in a correct way, i.e. in accordance with the syntax rules of the modelling language?
3. Pragmatic quality: is the model easily comprehended and used by the stakeholders in ways that are commensurate with its intended purpose?

Though ‘ease of use’ is often associated with the notion of usability, the concepts of pragmatic quality and usability are not synonymous, as will become clear later on. A specification is used in the first place by software programmers who develop tools that draw on the specification. They are the primary, direct users and their evaluation of the specification’s pragmatic quality is therefore referred to as ‘first-order’ pragmatic quality. However, ultimately they develop these tools for end-users who want to deploy the specification. This means we have to rely on users' evaluation of the Learning Path Specification *as conveyed by the tools that deploy the specification*, to assess second-order pragmatic quality.

The framework identifies a number of stakeholder groups: lifelong learners, learning designers, study advisors, educational providers, and software developers. The focus here is on pragmatic quality and the stakeholder groups of software developers and end-users. Even if the specification is to be used to describe informal learning paths as well, it is not evident that lifelong learners will do this themselves because of the reflective and learning design skills required to separate, for instance in workplace learning, the learning activities from actual work activities [Skule 2004]. We therefore expect learning paths to be described by trained professionals. These experienced users could be teachers, learning designers or study counsellors, who are employed by educational institutions or education and training brokers [Kilpatrick et al. 2007] to design curricula and provide study guidance to students. Or they could be human resource consultants and trainers employed by large companies and non-governmental organisations which make considerable investments in training and workplace learning. Finally, they could be professionals employed by local social services to advise the unemployed regarding opportunities for further professional or personal development. They all might want to document formal, non-formal and informal learning paths, which seem interesting or have proved successful, so that they become readily available for the purpose of recommending them to others.

2.3 Pragmatic Quality, Usability and Desirability

Pragmatic quality of a model is influenced by a number of characteristics (Janssen et al., submitted). Pragmatic quality is high when the model is:

a. unambiguous (i.e. concepts and relations have a clear single meaning);
b. internally consistent (i.e. the model does not contain contradictions);
c. general (i.e. concepts are as independent as possible from any specific application or domain).

[Krogstie 1998] defines pragmatic quality as "the correspondence between the model and the audience's interpretation of it". He further distinguishes social quality as "agreement among participants interpretation" (p. 87). However, this distinction between pragmatic quality and social quality appears rather artificial if we take into account that it is highly unlikely that an audience's interpretation fits the model when there is no agreement among the audience. We therefore prefer to consider agreement among participants as an indicator of the pragmatic quality of a model, more specifically of it being unambiguous. Others distinguish between technical pragmatic quality and social pragmatic quality, indicating whether the model is correctly (and easily) interpreted by tools and human users respectively [Nelson et al. 2005]. This distinction resembles the distinction we made between first-order and second-order pragmatic quality, with this difference that clearly in our view technical quality involves human users, i.e. software developers.

The criteria identified above as indicators of pragmatic quality apply equally to first-order and second-order pragmatic quality, e.g. creating learning path descriptions will be easier when concepts used in the model and subsequent tools are general, clear and applied consistently. However, assessing pragmatic quality is not merely a matter of asking users how they evaluate the specification regarding these criteria. Though software developers will understand these criteria and will be able to identify a specification's flaws related to these criteria at face value (i.e. merely by reading the specification), more flaws are likely to come to light in the process of building a tool that draws on the specification. Users involved in the assessment of second-order pragmatic quality can be expected to identify flaws in using the tool as well but, as was stated earlier, they cannot be expected to identify whether these flaws relate to the specification or the tool, nor to interpret them in terms of the rather abstract criteria of pragmatic quality in the same process. The model quality framework therefore suggests indicators like perceived ease of use, perceived usefulness, and intention to use as measurements of pragmatic quality. Indicators which are closely related to the concepts of usability and desirability [Benedek and Miner 2002, Hornbæk 2006, ISO 1998]. Whereas aspects like perceived usefulness and intention to use do not directly constitute an issue for the stakeholder group of software developers, they do become important with end-users. After all they may find it easy to use the specification, but if they see no added value in doing so, there is something amiss still with the pragmatic quality of the specification. The purpose of our study is not to evaluate the usability of the finished product (the Learning Path Editor), but rather the pragmatic quality of the Learning Path Specification. This then requires a more profound understanding of the rationale behind the Learning Path Specification: its intended use and desired effects. As [Moody 2005] points out, the evaluation of a conceptual model is less straightforward than the evaluation of a finished product (a software system). Whereas a finished product can be evaluated against initial requirements, evaluation of a conceptual model/specification also involves (tacit) needs, desires, and expectations. The concept of pragmatic quality is thus linked not just to 'usability' but to 'desirability' as well.

Taking into account the above considerations the evaluation of second-order pragmatic quality involves a number of successive steps as will be further explained in the Method section.

3 Method

The learning path specification's pragmatic quality was assessed distinguishing between first-order and second-order pragmatic quality [Figure 3].

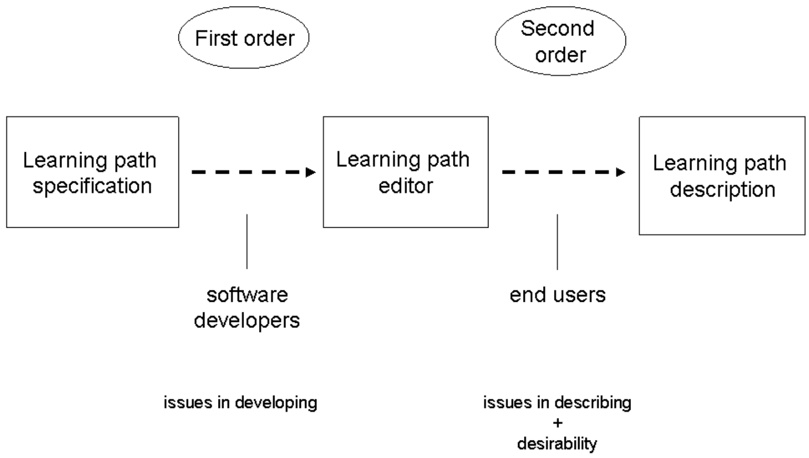

Figure 3: First-order and second-order pragmatic quality assessment

First-order pragmatic quality was assessed through the process of developing a reference implementation; in our case a tool that describes learning paths in line with the specification. The software developers involved in developing the tool had not been involved in the development of the specification. They had to rely on the information contained in the schema and the information model [Janssen et al. 2008b]. During this process some issues arose concerning the interpretation of the schema which lead to adaptations and a new release of the schema. These issues have been analysed and will be described in terms of the criteria for pragmatic quality in the Results section. Whereas first-order pragmatic quality was assessed basically by logging usability issues that arose during the software development process and analyzing these issues in terms of pragmatic quality criteria, second-order pragmatic quality was assessed beyond mere 'usability', incorporating the desirability of the approach and purposes implied by the Learning Path Specification. For this second-order quality assessment prospective users were invited to a hands-on workshop. The more detailed workshop proceedings are described in [section 3.2], following a brief description of the Learning Path Editor in [section 3.1].

3.1 The Learning Path Editor

The Learning Path Editor is a software programme that enables description of learning paths that are in line with the learning path specification [Melero et al. 2010]. The programme was developed to function as a portlet within the Liferay environment [Liferay 2000] and consists of three different 'views' that correspond to different tasks related to the description of learning paths:

1. Handling of learning paths, i.e. keep an overview, choose to change existing learning paths or to create a new learning path (Master view);
2. Describing the characteristics of a learning path (Metadata view);
3. Modelling a learning path (Design view).

In the Metadata view [Figure 4] the author enters a title and short description of the learning path and selects the competence levels, which are attained upon completion of the learning path. The competence levels displayed in the metadata view are predefined and made available through another portlet within the Liferay Environment.

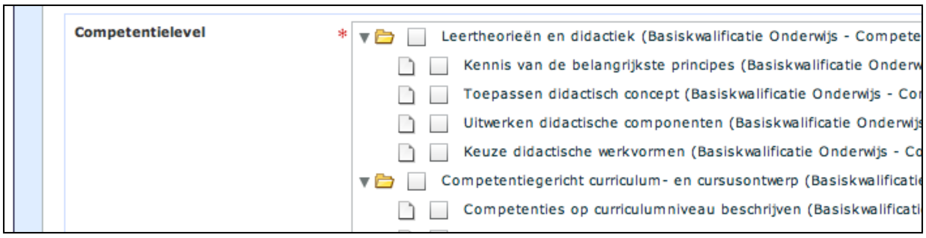

Figure 4: Learning Path Editor Metadata View: Competence Levels

Additional characteristics of the learning path used by learners in the process of searching a suitable learning path, e.g. language, costs, delivery mode etcetera, are specified through a form. When the necessary information has been provided the author clicks ‘save’ and returns to the Master view, which now includes the newly created learning path. For the actual modelling of the learning path the author clicks ‘Design’.

In the Design view [Figure 5] the author can add actions or existing learning paths and group them in clusters to specify particular subsets, e.g. sequential ordering, choice options, etc. Note that the user has to specify whether the learning path (top level) constitutes a sequence, free-order (Dutch: ‘selectie’) or parallel grouping of learning actions.

Figure 5: Learning Path Editor Design View

To add an Action the author clicks the “Add action” button. Next a dialogue box appears which asks to provide a title and a web address for the action. This web address may refer to all kinds of actions: a simple instruction to read a book, a complete course, a game or simulation, a test etc. The author can also choose to include existing learning paths in the design, for instance to express that a university degree is built up out of a bachelor programme and a master programme, each of which has been designed as a separate learning path (please note that we use an example from formal learning only because it makes for an easier explanation; this action applies to non-formal learning situations equally well, but is harder to describe as no commonly understood paths exist there).

To group actions and/or existing learning path descriptions into a cluster the author clicks “Add Cluster” and again a dialogue box appears, in which she can specify what type of cluster she wants to create: Free-order, Sequence or Parallel [Figure 6].

Once the cluster has been added the author can drag and drop the required actions to the cluster.

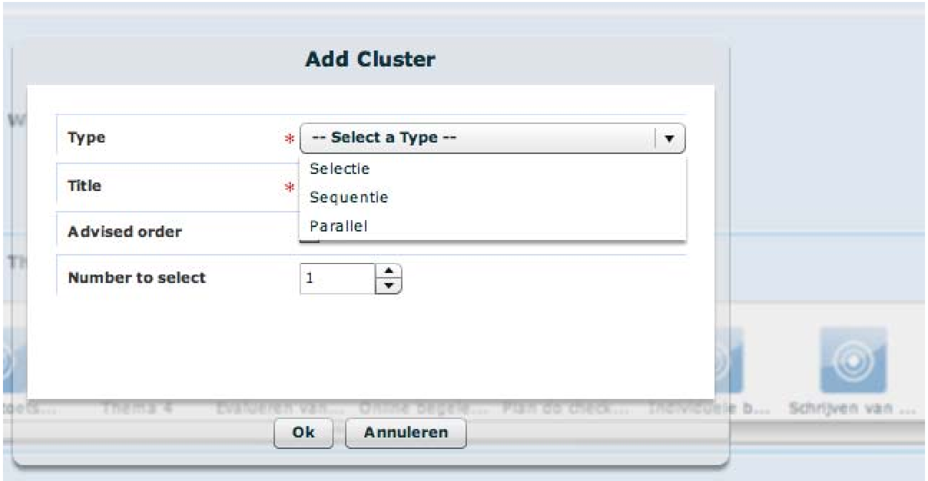

Figure 6: Learning Path Editor Design View: Add Cluster

As already mentioned, the Learning Path Editor was developed as part of the TENCompetence integrated infrastructure, which aims at supporting lifelong competence development. To this end, the infrastructure comprises a set of services through portlets like the Competence Editor, Profile, and Activity Navigator. The Learning Path Editor interacts and depends particularly on two portlets: the Competence Editor, for creating standard competence descriptions, and the Activity Navigator, for the presentation of learning paths to learners.

3.2 Evaluation

First-order pragmatic quality was assessed in the process of developing the Learning Path Editor. Eventually six software developers worked more or less closely with the specification; four of them directly in relation to the Learning Path Editor and two from the perspective of presenting learning paths to learners through the Activity Navigator. Clarifications were requested on a number of issues, often involving extensive discussions, which led to several minor and some more profound adaptations of the schema (binding).

The evaluation of second-order pragmatic quality required some specific instrumentation. Workshop sessions were organised to find out if the specification was easy to understand and to use by end-users of the Learning Path Editor. For the workshop sessions a demo was developed that offered a brief explanation of the learning path specification - its rationale and main features - followed by a demonstration of how an existing learning path can be described in the proposed way, using the Learning Path Editor [Janssen 2010]. As an example of a learning path, the Open University of the Netherlands Basic Teaching Qualification (BTQ) was used, because it seemed particularly suitable as an authentic case, which represents several requirements the specification is meant to meet (i.e. it represents a competence based, modular, nested learning path).

Participants (*n*=16) were recruited from two types of employees at the Open University of the Netherlands: those involved in advising students (study advisors), and those involved in designing learning (educational technologists). For the latter group an additional criterion was that they had no prior knowledge of Educational Modelling Language or IMS Learning Design, as such prior knowledge could help their understanding of the Learning Path Specification.

Participants were asked to watch the demo and to identify any unclear issues. Subsequently they were asked to use the Learning Path Editor to adapt the example learning path from the demo. The adaptations they were asked to make involved: 1. change the learning path so that an action that was part of a free-order cluster now becomes an action that can only be done after all the other actions of the cluster have been completed. 2. Add a last action ‘Final Assessment’. 3. Indicate that the final assessment can be taken either in location X or in location Y. Each workshop session involved two participants who were instructed to work individually. Requests for support or clarifications were noted down for later analysis.

At the end of the workshop participants were asked to evaluate their experiences using the Online Desirability Toolkit (ODT) [Storm and Börner 200]). The ODT is an online adaptation of the Desirability Toolkit developed by [Benedek and Miner 2002]. This Desirability Toolkit uses the product reaction cards methodology involving a set of 118 word cards, containing words like ‘relevant’, ‘exiting’, ‘useful’, ‘stressful’, ‘time consuming’, etc. Users were asked to pick a number of cards which most closely reflect their experience with the product. However, the particular selection of words is not that important: having selected the cards users were asked to explain their choice of cards in an interview. The cards merely function as prompts for these more in-depth explanations. This methodology allows users to describe and evaluate their experiences independent of any predefined notions, scales, criteria, etc. considered important by the researcher. The cards contain both ‘neutral to negative’ and ‘positive’ adjectives in proportions that reflect average results in usability studies of 60% positive and 40% neutral to negative feedback [Benedek and Miner 2002]. The Online Desirability Toolkit follows the same methodology: users were asked to select 6 cards from a total of 118 cards, which were now spread out on a screen rather than on a table. Having selected 6 cards users were asked to explain each choice in writing, rather than in an interview.

Though the Desirability Toolkit was designed for product evaluations, the particular features described above, e.g. leaving it to the users to decide which issues are relevant for the evaluation and stimulating negative feedback, make it a suitable instrument for our purpose of assessing pragmatic quality of the Learning Path Specification. In our study the Online Desirability Toolkit was used to ask participants to evaluate the 'approach of describing learning paths according to a specification as you have come to know it through the demo and the hands-on experience'. The data gathered during the workshop sessions were used as follows:

a. Questions and problems that arose during the introduction and hands-on session were recorded and interpreted in terms of the criteria clarity, ambiguity, and generality.
b. Online Desirability Toolkit data were used to assess the Learning Path Specification approach more generally: do prospective users consider it desirable to describe learning paths as proposed by the specification?

4 Results

4.1 First-order Pragmatic Quality

In the process of developing the Learning Path Editor several issues arose regarding the interpretation or use of the specification. These issues led to minor as well as more profound changes. Minor changes:

- For reasons of consistency the element LearningPathRef was changed into the element ExternalLearningPath and an ID was made mandatory for this element, as Actions and Clusters are also 'declared' separately and then referenced internally through an ID.
- In the first version of the schema metadata were grouped in a container element called Metadata. One of the software developers proposed to take the metadata element ID out of the container element, so that the related entity (Action, Cluster or Path) would become more directly accessible.
- The element Learning Actions as container element within the LearningActionsCluster element was removed because it appeared superfluous and therefore confusing. The schema now indicates that a LearningActionsCluster contains one or more LearningActionsRefs, LearningActionsClusterRefs, or ExternalLearningPathRefs.

Three more profound changes were made, of which the first two are related to the rendering of a learning path to students:

- The metadata element ‘Title’, optional in the first version of the specification, was made mandatory to enable proper rendering in the Activity Navigator.
- An attribute AdvisedOrder was added to the LearningActionsCluster, which provides information on how the Cluster should be presented to learners. The attribute specifies whether the order in which LearningActions are included in the cluster is mandatory or merely recommended, enabling the author to indicate to what extent navigation of the actions ought to be restricted.
- The restriction that a LearningActionsCluster contains at least two elements was removed. Though there is some logic to the restriction, e.g. it hardly makes sense to define a group of 1, it is not necessary and indeed undesirable to specify such a restriction, as the Cluster element is also used to specify the overall ordering of a LearningPath and it should be possible to create a LearningPath consisting of a single action.

What we called minor changes are changes concerning the translation of the specification into this specific schema rather than the specification itself. The term ‘specification’ is somewhat confusing in this respect, as it is used both for conceptual models (e.g. a UML model) and the technical implementation of these models in a schema using a particular syntax, in our case XML [cf. Klein et al. 2000]. As is the case in natural language, the things we want to express can be expressed in many different ways grammatically, representing different nuances perhaps, but still bearing the same message/meaning. Similarly, the minor adaptations made to the schema represent ‘grammatical’ or ‘syntactical’ changes. Nevertheless, even if we call these changes minor, they clearly brought about an important improvement in terms of pragmatic quality, in the sense that these changes made it easier for software developers to read, interpret, and deploy the specification.

4.2 Second-order Pragmatic Quality

Twelve participants thought the demo was clear and had no further questions concerning its contents. Four participants did have difficulties understanding the demo, more particularly the explanation of different ways learning actions can be grouped into clusters. One of them said she would need to see more examples. The other three felt it was just too confusing; that the terminology (e.g. ‘sequence’) was not clear, or that it was hard to make the mental switch of viewing things in this way. Interestingly though, one of these participants was actually the only one who correctly adapted the example learning path without any assistance. So although most participants considered the explanations provided through the demo satisfactory, they had difficulties making the adaptations to the learning path they had watched being modelled. Especially the first adaptation posed a challenge: re-design the learning path so that one of the learning actions included in a selection has to be done after the selection has been completed. Most participants moved the learning action so that it became the last action in the cluster, possibly confused by the demo which had explained that in a sequence the order the learning actions are shown in the cluster is the order in which they have to be completed. However, this is typically not the case for a free-order cluster. Adding an action ‘Final Assessment’ posed no problem, though several participants included the information on the location in the ‘description’ field, rather than in the designated field ‘Location’.

Following these experiences participants were asked to evaluate the approach of describing learning paths according to a specification, by selecting six adjectives they consider to best describe the approach and to further motivate their choice of words. Despite the large number of cards available, participants’ choice of cards showed some overlap.

[Table 1] offers an overview of ‘positive’ and ‘neutral to negative’ words that were selected.

Positive	*n*	**Neutral/Negative**	*n*
Usable	8	Organized	4
Useful	7	Complex	3
Helpful	5	Confusing	3
Professional	5	Time consuming	2
Timesaving	4	Hard to use	1
Efficient	4	Uncontrollable	1
Valuable	3	Too technical	1
Appealing	3	Frustrating	1
Customizable	3	Difficult	1
Understandable	3	Patronizing	1
Stimulating	2	Unpredictable	1
Attractive	2	Integrated	1
Accessible	2	Personal	1
Clear	2	Controllable	1
Relevant	2	Secure	1
Convenient	2		
Easy to use	2		
Effective	1		
Essential	1		
Flexible	1		
Advanced	1		
Low Maintenance	1		
Comprehensive	1		
Inviting	1		
Desirable	1		
Inspiring	1		
Innovative	1		
Impressive	1		
Friendly	1		
Simplistic	1		
Motivating	1		
Total	73 (76%)		23 (24%)

Table 1: Positive and Neutral/Negative Words Selected Overall

[Table 2] shows the selection of negative/neutral words across participants and indicates that the majority of participants selected no (*n*=3) or a single (*n*=7) negative/neutral card.

Number Negative/neutral	*n*
0	3
1	7
2	4
3	1
4	0
5	1
6	0
Total	16

Table 2: Number of Negative/Neutral Words Selected by Participants

However, rather than the exact choice of words, the motivations provided for these choices matters. The term ‘organized’, for instance, is a neutral term which was chosen by four participants, one of whom motivated the choice in a way that indicates the adjective was interpreted as a positive characteristic: ‘*well considered and set up*’. This explains why, for instance, the word ‘simplistic’ does not appear in the ‘neutral to negative’ column, as the motivation provided with this adjective was quite positive. The participant who chose this word, in fact provided two different positive motivations for this adjective: ‘*easy to maintain’* and ‘*making it easier to find suitable learning paths*’.

There are more incidences of participants giving more than one reason for choosing a particular adjective, so that numbers in the following analysis of motivations may diverge slightly from the numbers provided in [Table 1].

Analysing the motivations or explanations provided in words, it appears that participants have selected adjectives in connection with four different issues:

1. *General perception of the approach and its intended effects*
a. Positive evaluations (*n*=17): useful (3x), usable (2x) (‘*with respect to the goal*’), attractive (2x) (‘*many will easily understand the benefits*’), appealing (2x), advanced, time-saving, stimulating, relevant, innovative, convenient, flexible, stimulating;
b. Neutral/negative evaluations (*n*=4): patronizing (‘*less easy to customize because of use of standards*’), integrated (‘*everything is connected*’), organized, time-consuming (‘*the idea is good, but implementing it will be time-consuming’*).

2. *The impact of the approach on the provision of learning paths*
a. The provision of information on offerings becomes more structured (*n*=9). This particular effect was described as: organized (3x), time-saving, controllable, efficient, relevant, understandable, professional.
b. The information provided can be easily adapted (*n*=4): low maintenance, usable, customizable, and simplistic (‘*easy to maintain’*).

3. *The impact of the approach on learner experiences*
a. It will become easier for students to find suitable learning paths (*n*=15). Participants selected the following adjectives to express this: efficient (2x), useful (2x), appealing, accessible, usable, professional, time-saving, organized, usable, simplistic (*'easy to find required information'*), convenient, helpful, and valuable.
b. Learners get an overview of their learning path and how to follow it (*n*=24): helpful (4x), efficient (3x), useful (2x), organized (2x), understandable (2x), valuable (2x), effective, easy to use, motivating, customizable, usable, controllable, relevant, desirable, inspiring.

4. *Workshop experiences: demo and adaptation of learning path*
a. Positive evaluations (*n*=17): usable (4x), professional (3x), customizable (2x) (*'easy to manipulate', 'you can add a lot of information')*, easy to use (2x), friendly, accessible, understandable (*'it is easy to use'*), useful (*'easy to work with'*), inviting, comprehensive ('*everything I needed was more or less on the same screen*').
b. Neutral/negative evaluations (*n*=13): complex (3x), confusing (3x), uncontrollable, too technical, difficult (*'it takes time to learn'*), frustrating, hard to use, time-consuming (*'in the beginning'*), secure (*'requires precision'*).

The above categorization of motivations makes clear that end-users have not only understood the main purposes of the specification, but also find them desirable: expected impacts of the approach on provision of learning paths as well as learner experiences are solely described in positive terms. Though the workshop experiences were often evaluated in positive terms as well, participants were divided as to how easy it is to model learning paths using the specification's building blocks. The main problem concerned the interpretation and modelling of cluster types. Though it was regularly mentioned that it would be merely a matter of time to become acquainted with the approach and to develop some routine, we also concluded that the specification would become easier to understand if the term originally used to indicate a sequence of activities ('sequence') would be replaced by 'fixed order'. So the values of the attribute 'cluster type' were changed to 'free-order', 'fixed order', and 'parallel'.

5 Discussion

The Learning Path Specification's pragmatic quality was evaluated in two distinct steps. As always 'the proof of the pudding is in the eating', but it is the cook who gets to taste first. In the case of the Learning Path Specification software developers were the first to 'get a taste' of the specification in the process of developing tools that enable description and presentation of learning paths based on the specification. First-order pragmatic quality of the specification was assessed in the course of this process. Second-order pragmatic quality was evaluated through workshop sessions involving prospective users of the Learning Path Editor.

Adaptations made to the specification in the course of both evaluations illustrate how different aspects of quality are intricately connected: both syntactic and semantic quality issues may be discovered in the process of assessing pragmatic quality, and once resolved clearly contribute to improved pragmatic quality as well. As was stated already in developing the framework for the evaluation, syntactic quality and semantic quality precede pragmatic quality. If a specification lacks clarity or completeness or if it is poorly expressed in UML, XML or any other language, this will affect pragmatic quality, i.e. whether the specification is (easily) understood. In other words, semantic quality and syntactic quality are a prerequisite (but not sufficient) condition for pragmatic quality. However, paradoxically, some semantic and syntactic flaws are likely to become visible only in the process of deploying the specification. This means that the evaluation of model quality is inevitably an iterative process.

[Duval and Verbert 2008] formulate the rule of thumb that a specification should be implemented at least by two independent development teams and evaluated in at least two independent user studies before it is ready for a process of standardization. Our study clearly indicates that both perspectives (implementation and user studies) are invaluable in the development of a specification. First-order pragmatic quality assessment (implementation) and second-order pragmatic quality assessment (user study) provide quite different results and insights. As for the criterion that these implementations and user studies should be independent, the question arises what exactly is meant by ‘independent’. In our study some ‘independence’ was achieved by making sure that the software developers involved in the implementation of the specification had not been involved in the development of the specification. However, the evaluations and adaptations were instigated and carried out by the people who developed the specification. In practice it will be very hard to maintain a strict independence here as it requires a profound understanding of the specification and its aims. Third parties are not likely to engage in the process of implementing and evaluating the specification unless they have the same or similar aims.

A specification such as the Learning Path Specification is meant to be implemented in a number of different tools. Experiences with the development and deployment of tools designed to find, adapt, and present learning paths will lead to further evaluations that can enhance the quality of the specification, be it by independent third parties or by the team who developed the specification.

Acknowledgements

The work described in this paper has been sponsored by the TENCompetence Integrated Project that is funded by the European Commission’s 6th Framework Programme, priority IST/Technology Enhanced Learning. Contract 027087 (www.tencompetence.org).

References

[Beck 2002] Beck, B.: “Model evaluation and performance”; A. H. El-Shaarawi & W. W. Piegorsch (Eds.), Encyclopedia of Environmetrics, Vol. 3, Chichester: John Wiley & Sons (2002) 1275-1279

[Benedek and Miner 2002] Benedek, J., Miner, T.: “Measuring Desirability: New methods for evaluating desirability in a usability lab setting”; Paper presented at the Usability Professionals’ Association Conference (2002)

[CDM 2004] CDM: “A Specification of Course Description Metadata”; Proposal 20. Oct. 2004. Retrieved 30-01-2006, from http://cdm.utdanning.no/cdm/cdm-2.0.1/doc/courseDesc 201004.pdf

[CEN 2008] CEN: “Metadata for Learning Opportunities (MLO) – Advertising”; CEN Workshop Agreement CWA 15903. Retrieved 11-02-2010, from ftp://ftp.cenorm.be/PUBLIC/ CWAs/ e-Europe/WS-LT/CWA15903-00-2008-Dec.pdf

[Chen, Fan, Macredie 2006] Chen, S. Y., Fan, J. –P., Macredie, R. D.: “Navigation in hypermedia learning systems: experts vs. novices”; (Computer in Human Behavior), 22 (2006), 251-266.

[Duval, Verbert 2008] Duval, E., Verbert, K.: “On the Role of Technical Standards for Learning Technologies”; (IEEE Transactions on Learning Technologies), 1, 4 (2008), 229-234.

[Hornbæk 2006] Hornbæk, K.: “Current practice in measuring usability: Challenges to usability studies and research”; (International Journal of Human-Computer Studies), 64 (2006) 79-102.

[IMS-LD 2003] IMS-LD: “IMS Learning Design Information Model – Version 1.0 Final Specification”; retrieved 27-02-2004, from http://www.imsglobal.org/learningdesign/ index.cfm

[ISO 1998] ISO: “Ergonomic requirements for office work with visual display terminals (VDTs) - Part 11: Guidance on usability”; International Organisation for Standardisation: ISO 9241-11:1998(E).

[Janssen 2010] Janssen, J.: “De weg bereiden tot leren. Demo van de leerpad specificatie en editor”; Retrieved 08-04-2010, from http://dspace.ou.nl/handle/1820/2403

[Janssen submitted] Janssen, J., Berlanga, A. J., Koper, R.: “Evaluation of the Learning Path Specification: Lifelong learners' information needs”.

[Janssen 2008a] Janssen, J., Berlanga, A. J., Vogten, H., Koper, R.: “Towards a learning path specification”; (International Journal of Continuing Engineering Education and Lifelong Learning), 18, 1 (2008) 77-97.

[Janssen 2008b] Janssen, J., Hermans, H., Berlanga, A.J., Koper, R.: “Learning Path Information Model”; Retrieved 20-03-2010, from http://hdl.handle.net/1820/1620

[Kickmeier-Rust, Albert, Steiner 2006] Kickmeier-Rust, M. D., Albert, D., Steiner, C.: “Lifelong Competence Development: On the Advantages of Formal Competence-Performance Modeling”; International Workshop in Learning Networks for Lifelong Competence Development, TENCompetence Conference, Sofia, Bulgaria (2006)

[Kilpatrick, Fulton, Johns 2007] Kilpatrick, S., Fulton, A., Johns, S.: “Matching training needs and opportunities: the case for training brokers in the Australian agricultural sector”; (International Journal of Lifelong Education), 26, 2 (2007), 209-224.

[Klein, Fensel, Harmelen, Horrocks 2000] Klein, M., Fensel, D., Harmelen, F. v., Horrocks, I.: “The Relation between Ontologies and Schema-Languages: Translating OIL-Specifications to XML-Schema”; Proceedings of the Workshop on Applications of Ontologies and Problem-solving Methods, 14th European Conference on Artificial Intelligence ECAI-00. Berlin (2000).

[Krogstie 1998] Krogstie, J.: “Integrating the Understanding of Quality in Requirements Specification and Conceptual Modeling”; ACM SIGSOFT Software Engineering Notes, 23, 1 (1998), 86-91.

[Lea, Stephenson, Troy 2003] Lea, S. J., Stephenson, D., Troy, J.: “Higher Education Students' Attitudes to Student-centred Learning: beyond 'educational bulimia'?”; (Studies in Higher Education), 28, 3 (2003), 321-334.

[Leung, Bolloju, 2005] Leung, F., Bolloju, N.: “Analyzing the Quality of Domain Models Developed by Novice Systems Analysts”; Paper presented at the 38th Annual Hawaii International Conference on System Sciences, Hawaii (2005).

[Liferay 2000] Liferay: “Liferay Open Source Enterprise Portal”; www. liferay.com

[Melero et al. 2010] Melero, J., Van Stratum, B., Janssen, J., Heyenrath, S., Van der Heijden, S., Finders, A., et al.: “Learning Path Editor” (2010).

[Moody 2005] Moody, D. L.: “Theoretical and practical issues in evaluating the quality of conceptual models: current state and future directions”; (Data & Knowledge Engineering), 55 (2005), 243-276.

[Nelson, Poels, Genero, Piattini 2005] Nelson, H. J., Poels, G., Genero, M., & Piattini, M.: “Quality in conceptual modeling: five examples of the state of the art”; (Data & Knowledge Engineering), 55 (2005), 237-242.

[Oussena, Barn 2009] Oussena, S., Barn, B.: “The Pspex project: Creating a Curriculum management domain map [Electronic Version]”; Retrieved 29-07-2009, from http://www.elearning. ac.uk/features/pspex

[Simpson 2004] Simpson, O.: “Access, retention and course choice in online, open and distance learning”; The third Eden research workshop, Oldenburg, Germany (2004). Retrieved 05-07-2005, from http://www.eurodl.org/materials/contrib/2004/Ormond_Simpson.html

[Skule 2004] Skule, S.: “Learning conditions at work: a framework to understand and assess informal learning in the workplace”; (International Journal of Training and Development), 8,1 (2004), 8-20.

[Storm, Börner 2009] Storm, J., Börner, D.: “Online Desirability Kit”; Retrieved February 8th 2010, from http://desirabilitykit.appspot.com/.

[TENCompetence 2005] TENCompetence: “Building the European Network for Lifelong Competence Development”; Retrieved 05-01-2006, from http://www.tencompetence.org/

[TENCompetence 2006] TENCompetence: “TENCompetence Domain Model”; Retrieved from http://dspace.ou.nl/handle/1820/649

[UML 1997] UML: “Unified Modeling Language”; Object Management Group. Retrieved from http://www.uml.org/

[Van Assche 2007] Van Assche, F.: “Linking Learning Resources to Curricula by using Competencies”; Paper presented at the First International Workshop on Learning Object Discovery & Exchange (LODE'07). Retrieved from http://fire.eun.org/lode2007/lode11.pdf

[Van der Klink, Boon, Schlusmans, Boshuizen 2009] Van der Klink, M. v. d., Boon, J., Schlusmans, K., Boshuizen, E.: “Learn as you like. Research into informal learning of employees”; Paper presented at the AERA (2009).

[W3C 2008] W3C: “Extensible Markup Language (XML)”; World Wide Web Consortium

[XCRI 2006] XCRI: “eXchanging Course-Related Information”; Retrieved 16-10-2006, from http://www.elframework.org/projects/xcri

[Yorke 1999] Yorke, M.: “Leaving early. Undergraduate non-completion in higher education”; Falmer Press, London (1999)