Elup Editor Conceptual Guide and User Manual

Michèle Joab

▶ **To cite this version:**

Michèle Joab. Elup Editor Conceptual Guide and User Manual. 2007, pp.87. lirmm-00138492

HAL Id: lirmm-00138492
https://telearn.hal.science/lirmm-00138492

Submitted on 26 Mar 2007

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

e-Quality Project

110231-CP-1-2003-FR-MINERVA-M

UNIVERSITÉ MONTPELLIER 2
SCIENCES ET TECHNIQUES

Elup Editor

WP3 – D3.2
English version

Michelle Joab

15/9/2006
Version reviewed and validated by all partners

This project was carried out with the support of the European Community within the Action MINERVA of the Programme SOCRATES. The contents of this project do not necessarily reflect the position of the European Community or the National Agencies, nor does it involve any responsibility for their part.

This work is protected by copyright law as an unpublished work and cannot be used, reproduced or disseminated in any way without the previous consent of the authors. © 2003-2006 E-Quality

Acknowledgements

I would like to express my warmest thanks to everyone who collaborated on the eLup project: Jean-Marie Cazorla, DESS TNI, Montpellier 2 University, 2003-2004, Nicolas Gonzalez, DESS TNI, Montpellier 2 University, 2003-2004, Guilhem Legal, IUP 3rd year, Montpellier 2 University, 2004-2005, Rémi Legrand, Assistant Lecturer, Montpellier 2University, Odette Auzende, Assistant Lecturer, Paris 2 University, LIP 6 and Mireille Bachelot, multimedia designer.

I should like to extent my sincere thanks to those who nourished eLup: my colleagues at Tampere University, Juha Holma and Suvi Junes authors of deliverables D3.3 “Database of Best Practices” and D4 “Trainer’s and Professional’s Guide to Quality in Open and Distance Learning” and Nicolas Dunand at Lausanne University author of deliverable D3.1 annex 1 General Quality Process Charter. This work gave body to eLup, which would otherwise have been merely an empty shell.

Among the actors in the project, I would particularly like to thank Bernard Dumont for his invaluable help throughout the project. His thoughts on the Quality approach helped us to construct quality modelling for e-learning.

I would finally like to thank my colleagues on the e-Quality project, who managed to cope with the various eLup changes without complaining.

Table of contents

eLup Conceptual guide **4**

eLup User Manual **32**

eLup

Conceptual guide

Table of contents

1. INTRODUCTION 6

2. TO FAMILIARIZE WITH ELUP CONCEPTS 8

2.1. THE MODELLING LANGUAGE 8

3. DESCRIBING THE IMS LD LANGUAGE 11

3.1. OBJECTIVES 11

3.2. THREE LEVELS OF DESCRIPTION 12

4. COMPARISON BETWEEN ELUP AND IMS LEARNING DESIGN 17

4.1. LEVELS OF STRUCTURE IN ELUP 17

4.2. DYNAMIC ASPECTS 23

4.3. LANGUAGES DEDICATED TO EDUCATIONAL MODELLING. 25

4.4. SYNTHESIS 27

5. CONCLUSION 28

6. REFERENCES 29

7. TABLE DES FIGURES 31

1. Introduction

The « E-Quality» project aims at setting up a methodological guide to implement quality in Open and Distance Learning (ODL) in higher education in Europe. This guide is the outcome of the work of the E-Quality project. It is now available for use by ODL professionals to help them build the teaching aids with which to train ODL teams in quality (see [D4]). The guide is based on a conceptual model for ODL quality in Europe (see [D2.2] and [D3]).
The project partners analysed and compiled data on implementing quality in higher education in ODL in their various institutions and in the 5 countries concerned (Spain, Finland, France, Poland, Switzerland) (see [D2.1]).
The extracts on training materials in the guide were tested during training sessions (one in each country) (see D5). After training, a validation phase was used to evaluate the effective implementation of a quality approach by the trainees in the context of their work (see [D6]).

Ensuring the overall coherence of the data demanded a great deal of energy throughout the span of the project (October 2003-September 2006). Emphasis was placed on the conceptual model ([D2.2]), general information about quality processes ([D3.1]), the general quality process charter ([GQPC]), the best practices database ([D3.3]) and the methodological guide ([D4]).

The eLup editor: elearning Quality Process Editor was designed and produced to facilitate the integration of these data and to ensure their coherence. It is based on the conceptual model adopted by the E-Quality project.

The eLup software was developed within Montpellier 2 University. The purpose of this software is to enable the e-learning professional to structure and document the quality approach used in his/her organisation by making use of the approach and findings of the e-quality project. The software is a model editor coupled with a database using Java, MySQL and XML technologies. The eLup software allows the quality of e-learning processes to be modelled. The eLup interface is bilingual (French and English).

A user describes his/her quality approach on the basis of the activities, roles, artefacts and workflows featured in his/her institution. These concepts will be presented in section 2.

The designation « e-learning Unified Process (eLup) » was chosen as an analogy to the Unified Process, used in Software Engineering for software development [Jacobson et al.

1999, Jacobson et al. 2000, RUP 2003]. The Unified Process is based on Unified Modelling Language (UML) [Rumbaugh et al. 1999, UML 2004].

There were several ideas behind using this analogy. Firstly, the unified process models not only software but also the organisation model in which the software is run. The description of the process is very general: it involves the concepts of activities, roles, artefacts and workflows as listed in section 2. The activities make it possible to describe the procedures and artefacts step by step (the necessary concrete input and the results produced). The concepts make it possible to describe the concrete directives for implementing quality, underpinned by general approaches.

Secondly, the unified process is iterative and incremental. eLup software and the E-Quality project as a whole were designed in this spirit.

The functions of the eLup tool in the initial proposal were confined to navigating in an interactive document presenting the elearning quality model, norms, resources and best practices. eLup was originally intended to represent the formatting of the common model with the views and background specific to the various actors.

It became clear that the user would not be content to simply display the kernel of data resulting from the project. S/he needs to add and edit his/her own data. The user can thus work from the data produced by the project, adapt it to his/her needs and by adding his/her own data, develop his/her own model.

This idea led to one of a model editor instead of an interactive document.

The first version of eLup was a collaborative editor. After assessing this first version in terms of usability, the data validation process was found to be too cumbersome. The collaborative editor made it difficult to reach agreement about data. So the decision was made to produce software that could be used like an interactive notebook. The eLup software was modified to be able to do this.

2. To familiarize with eLup concepts

The modelling language provides the following concepts: role, activity, artefact and workflow. To summarize, an activity is performed by a role. An activity requires input artefacts and produces output artefacts. Activities are organised into workflows according to the formalism of activity diagrams of'UML (Unified Modelling Language). Each complex concept (activity or workflow) is related to quality features (quality criterion, indicator, commitments including measures to implement) and best practices. Thanks to these concepts, an internal view of the quality approach is given. In addition, an external view can be supplied by a role named "quality manager" by describing his/her activities in the institution.

2.1. The modelling language

Role

A role is recognized from behaviour and responsabilities of a group of people who work together in an organization.

For example (extracted from D4) : A tutor is a mediator between an institution and its students. This means that the tutor should be aware of the students´ progress, needs and problems to pass this information to other actors, and also to inform the students about the course. Tutoring in a virtual learning environment usually relies on textual support and there are no non-verbal clues to interpret. This requires reliable connection between the students and the tutor, which supports the questions, the expression of thoughts, and the feeling of easily approachable support.

Activity

An activity is performed by a role. It is described by steps, input and output artefacts.

For example (extracted from D4) : - Pedagogical planning - The pedagogical planning connects two processes of ODL: planning and student support. In the planning phase, the structure of the whole course is defined and the basis of pedagogy and methods are therefore decided. The teacher is responsible for the pedagogy of the learning event, that is to choose suitable pedagogical methods and to create activities and assignments to support the learning. After that the precise schedule and rules for working are created.

Artefact

Artefacts are the documents and templates that are modified, used, created as an input or output of an activity.

For example (extracted from CGPQ D3.1 Annexe 1) : *- Checklist - This tool is a way of ensuring that every Quality aspect is going to be (or has been) taken into account when planning the process. But it can also be used at the end of the process to re-evaluate the same Quality aspects in terms of how the process took place. It is thus both a preemptive and a subsequent measure toward the achievement of a Quality e-learning process.*

Workflow

A workflow is a sequence of activities which produces a result with an added value for each role involved in the workflow.

In the workflow, sequences of activities are grouped in workflow details. These workflow details are possibly performed in parallel or in sequence. Some of them may be performed under conditions. A workflow is a sequence of workflow details that may be separated by tests. Using *split* and *join nodes* indicates that the workflow details may be performed in parallel.

For example: the sub-process Student support is viewed as a workflow.

Workflow detail

The activities may be grouped in a workflow detail. Artefacts can be used as inputs or outputs of an activity. Workflow details are used to group activities in order to reach a higher level of abstraction which leads to a better understanding of workflows.

For example, Guidance is identified as a workflow detail.

Discussing the modelling language

Within the e-Quality consortium, the adequation between the modelling language and the description of the e-learning quality process was fully debated. Discussion focused on the difference between actors and roles. Actors are identified in each institution (teachers, students, administrative staff, quality manager, financial manager). They do not always play the same roles because they do not have the same responsabilities.

To reach an agreement on a common model, it is necessary to define the role independently from the organisational model of each institution, together with keeping links with effective actors.

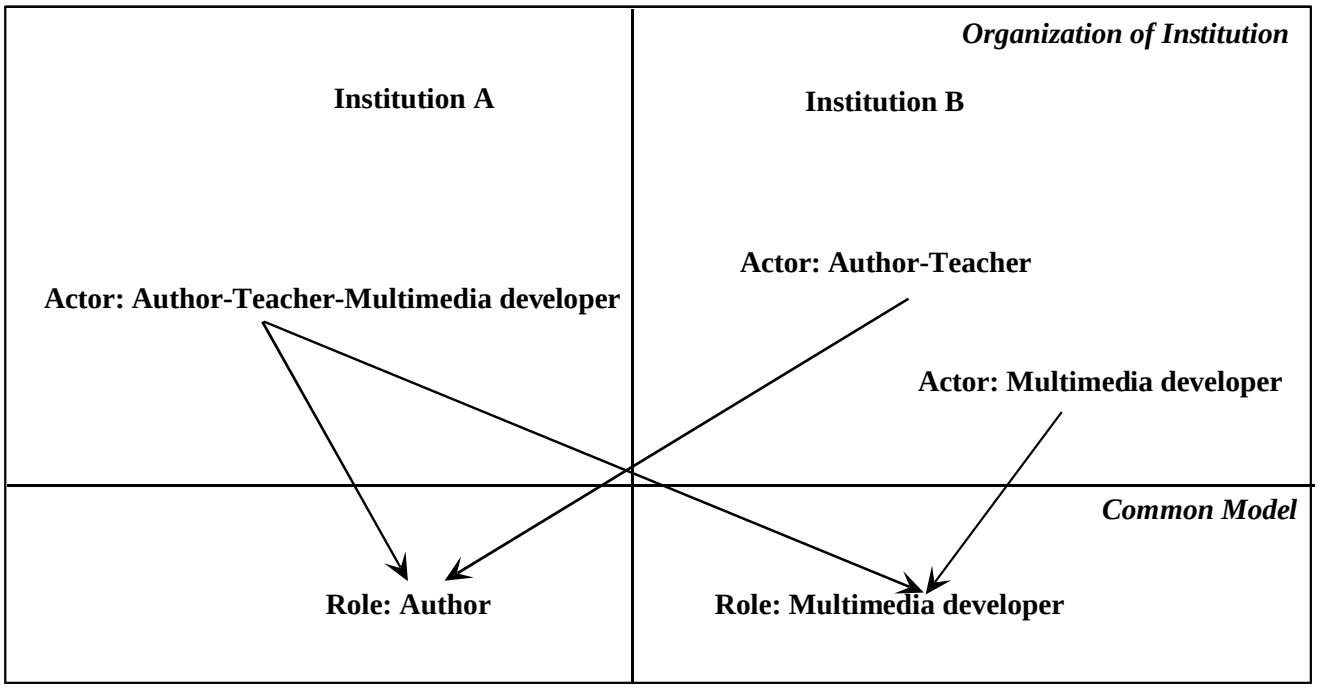


Fig. 2.1 Roles and Actors

3. Describing the IMS LD language[1]

This section will present a technical survey of the modelling language IMS LD in addition to the general presentation in to the [D3.1] deliverable of the e-Quality project entitled "General Information on Quality Processes.

IMS-LD is the result of the integration of the Educational Modeling Language (EML) (Open University of the Netherlands - OUNL) and existing IMS Specifications.
Reference documents (http://www.imsglobal.org/) are:

- IMS Learning Design Information Model
- IMS Learning Design Best Practice and Implementation Guide
- IMS Learning Design XML Binding

3.1. Objectives

What is « Learning Design » ?

Learning Design is a description of a method enabling learners to attain certain learning objectives by performing certain learning activities in a certain order in the context of a certain learning environment based on the pedagogical principles of the designer, specific domain and contexts variables

IMS Learning Design goal

To allow all designs to be included into e-learning modules, the approach of a meta-language is taken. enabling the description of all kinds of learning designs. A learning unit may include resources, instructions for learning activities, templates for structured interactions, conceptual models**,** learning goals, objectives and outcomes, assessment tools and strategies.

[1] Section 1 is extracted from the paper written by Odette Auzende published in the proceedings of the e-Quality Seminar held in Szczecin (september 2006). [Auzende 2006].

IMS-LD Specifications need:

- to describe and implement different kinds of learning approaches,
- enable repeatable and efficient units of learning,
- support multiple delivery models,
- support reuse and re-purposing of units of learning or of their component elements,
- leverage existing specifications and standard,
- be culturally inclusive and accessible (internationalization),
- support multiple learners and multiple roles in a learning activity,
- support reporting and performance analysis.

3.2. Three levels of description

Learning Design is divided into three parts, known as level A, level B and level C. Levels B and C each extending the previous level.

Level A

A Learning Unit is considered as a theatre play, set of acts. Each act consists in a time ordered series of activities. Each activity is performed by learners and teachers (*role*) within the context of an *environment* consisting of *learning objects* or *services.*

The conceptual model is presented in Fig. 3.1. The figure makes use of class diagrams expressed with the Unified Modelling Language (UML).

The major elements of the Learning Design Specification are hierarchically ordered. The method element contains a nested structure of play, act, and role-part elements; it specifies the dynamic aspects of the learning design. The play element (often only one) contains a number of act elements; these acts will be run in sequence.The play is complete when the last act is completed. Within an act there is a set of role-parts which are run in parallel ; this enables different roles to do different things at the same time. A role-part contains a reference to a role and a reference to an activity ; this effectively assigns the activity to the role for this act.

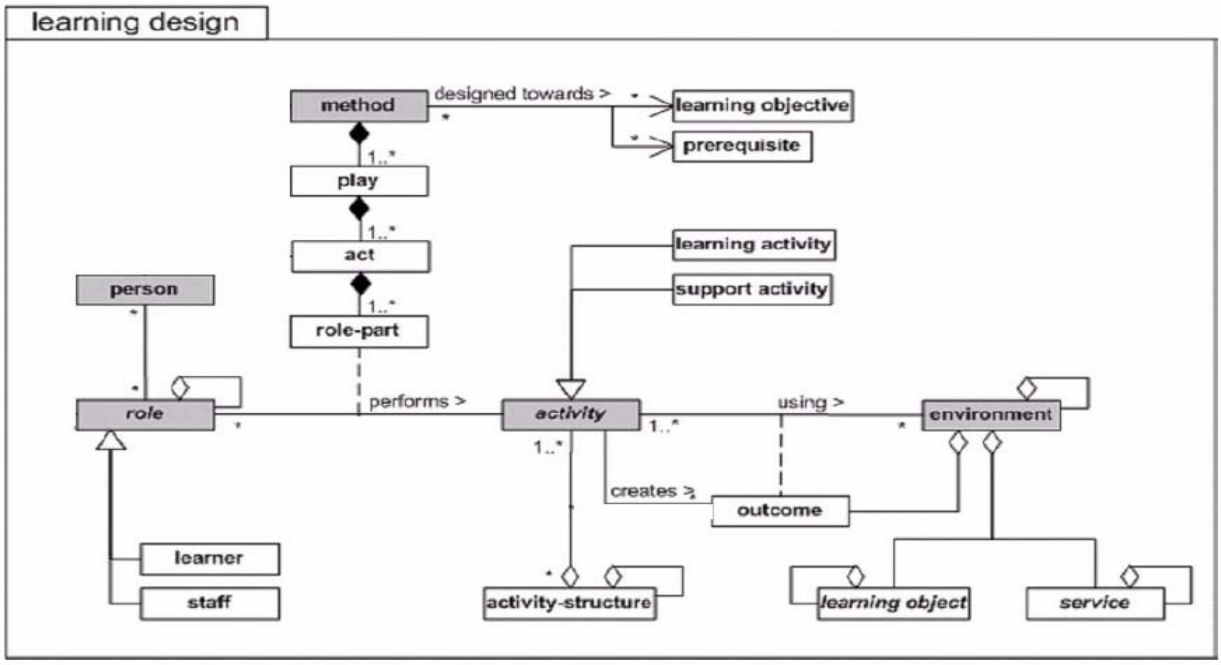


Fig. 3.1 IMS Learning Design Level A

Level B

Level B provides additional elements, which extend the ability of a learning designer to control the learning flow within a Unit of Learning. It provides for the inclusion of generic properties and conditions. To the single learner model, it adds learner personalization.

The grey marked classes are added to the model of level A:

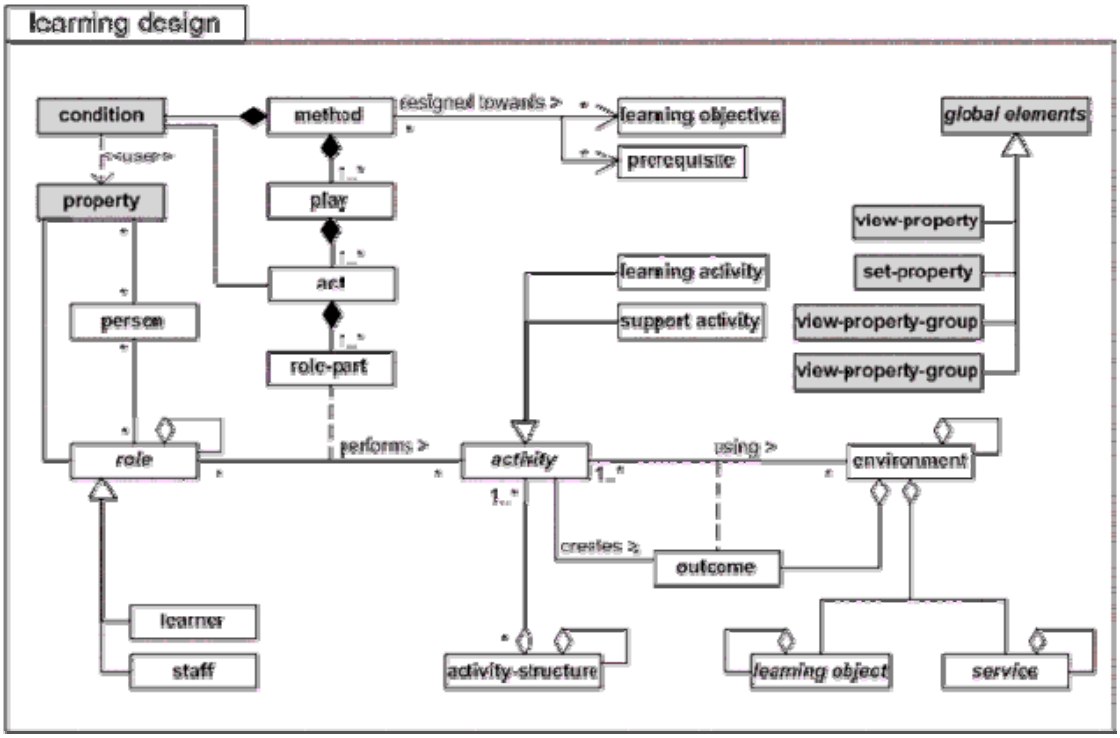


Fig. 3.2 IMS LD Level B

Level C

Level C introduces notification or "messaging" both between system components and between roles. A notification happens after an event, which is known by the runtime environment. This adds a new dimension by supporting real-time event-driven work/learning flow.

The grey marked class is added to the model of level B:

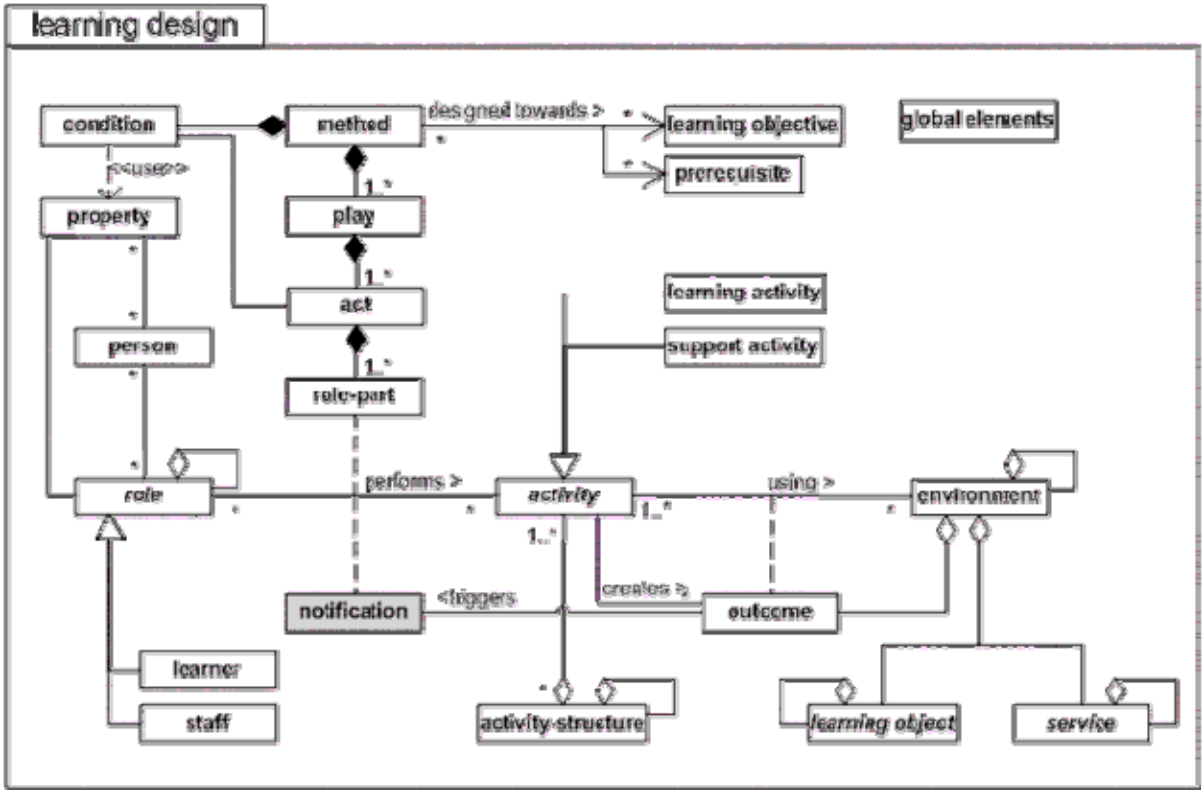


Fig. 3.3 IMS LD Level C

Unit of Learning = IMS Content Package + IMS Learning Design

The IMS Learning Design is preferably integrated into an IMS Content Package to create a Unit of Learning.

This perspective focusses more on the learning activities than on contents even structured into packages.

IMS-LD benefits

Completeness : the teaching-learning process is fully described in a unit of learning. This includes:

- integration of the activities of both learners and staff members.
- integration of resources and services used during learning.
- support for a wide variety of approaches to learning.
- support for both single and multiple user models of learning.
- support mixed mode (blended learning) as well as pure online learning.

Pedagogical Flexibility : the specification allows the description of all different kinds of pedagogies and doesn't proscribe any specific pedagogical approach.

Personalization : content and activities within a unit of learning can be adapted to users. In addition, the control over the adaptation process must be given, as desired, to the student, a staff member, the computer, and/or the designer.

Formalization : a unit of learning is described in a formal way, so that automatic processing is possible

Reproducibility : the learning design is described in such a way that repeated execution in different settings with different persons is possible.

Interoperability : the specification supports interoperability of learning designs.

Compatibility : the specification uses available standards and specifications where possible, mainly IMS Content Packaging, IMS Question and Test Interoperability, IMS/LOM Meta-Data and IMS Simple Sequencing.

Reusability : the specification makes it possible to identify, isolate, de-contextualize and exchange useful learning artefacts, and to re-use these in other contexts.

4. Comparison between eLup and IMS Learning Design

The comparison focuses on three subjects:

- The number of levels authorized in the structure; this could be related to the complexity of the modelling language.
- The way of specifying how to schedule the activities dynamically.
- Is the modelling language dedicated to education?

4.1. Levels of structure in eLup

The first point to compare between eLup and IMS LD is the hierarchical levels in the structure.

.

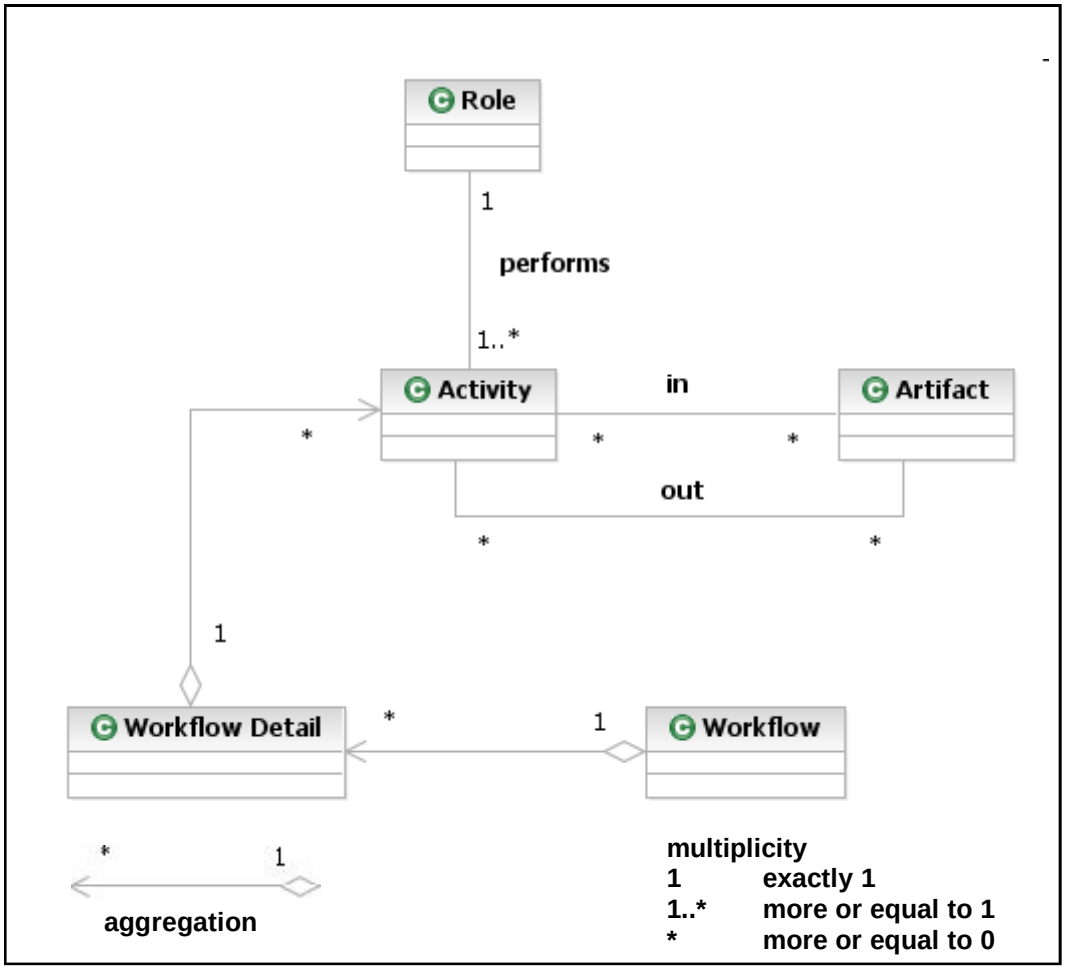


Fig. 4.1 eLup meta-model

Fig. 4.1 displays the structural relationships between the concepts used in eLup, expressed in the modelling language UML Unified Modelling Language and more precisely those of class diagrams.

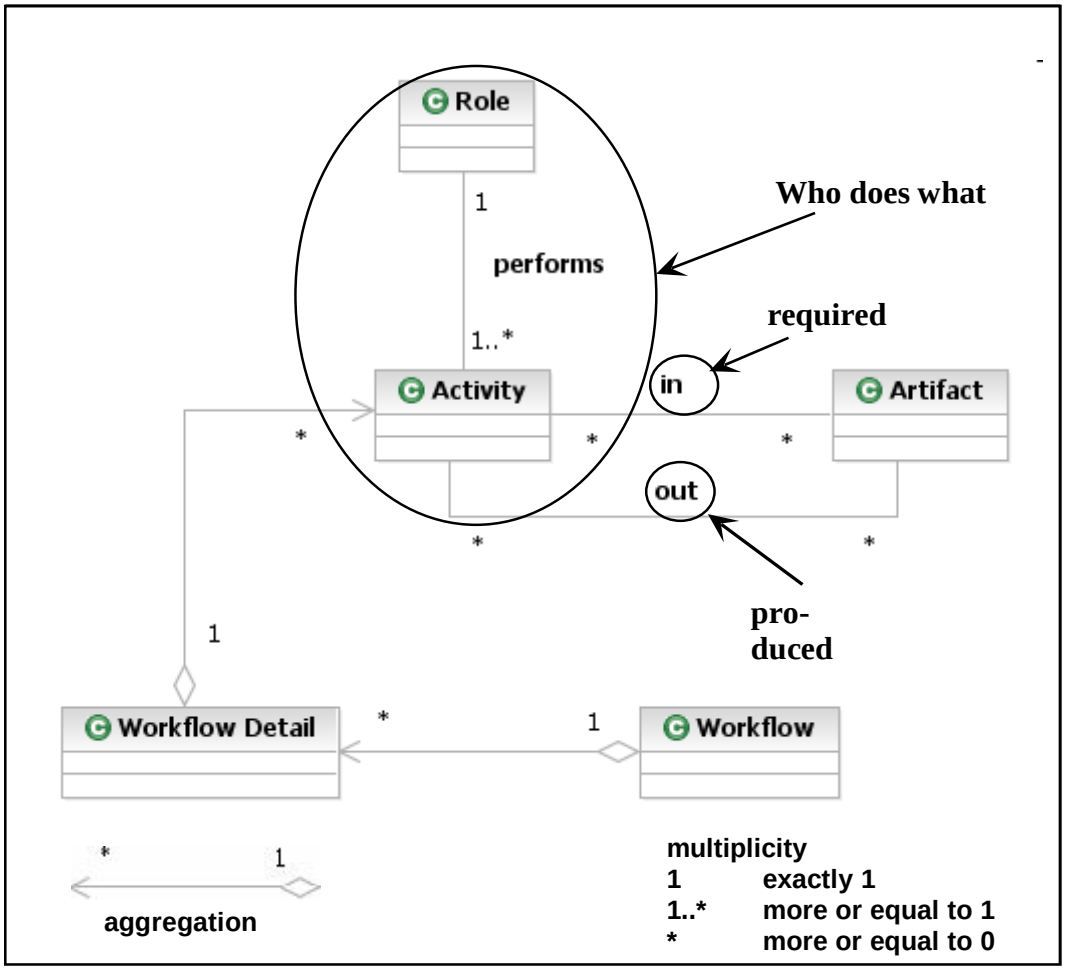


Fig. 4.2 Use of concepts

Fig. 4.2 displays the description of an activity with a role and one or several input and output artefacts.

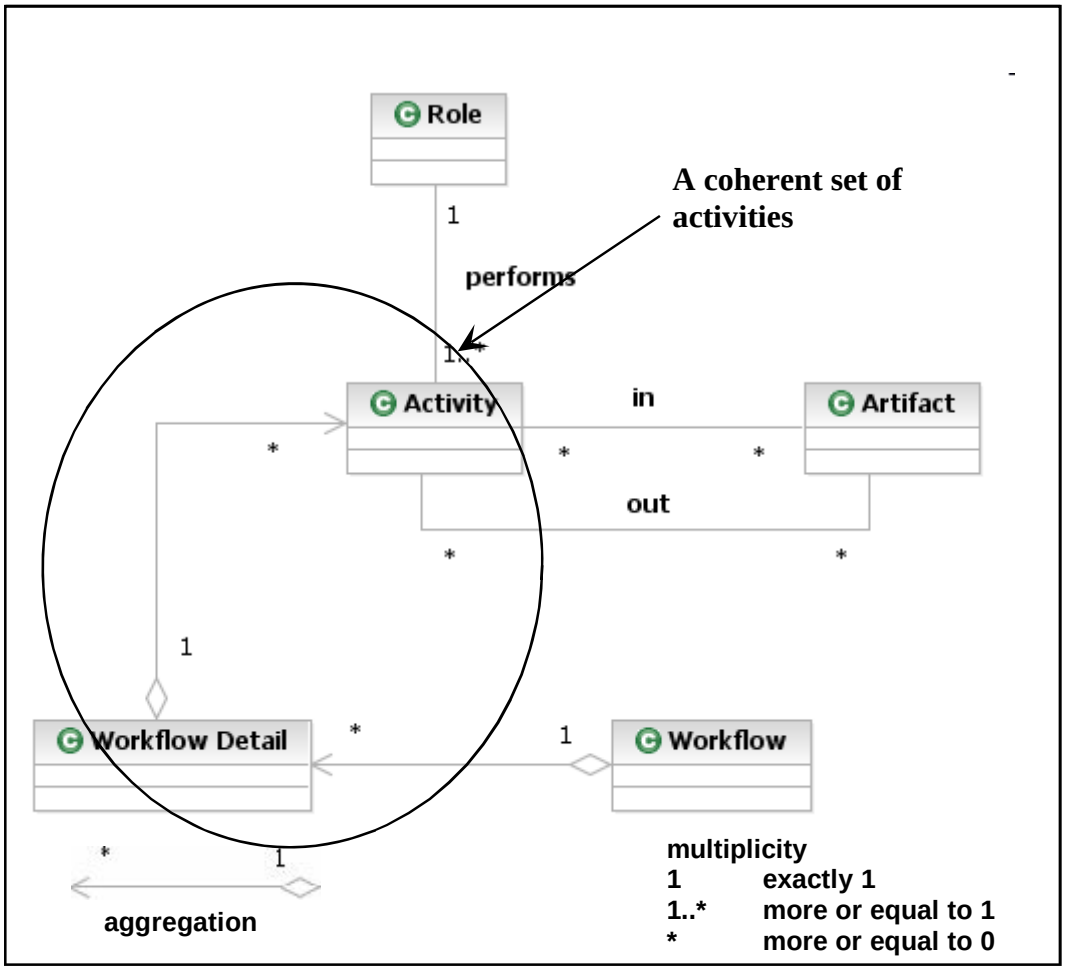


Fig. 4.3 Grouping activities into workflow details

Fig. 4.3 displays the aggregation relationship between a workflow detail and activities. The aggregation (part-of relationship) models a level of structural hierarchy between workflow detail and activity. We do not take into account the description of the steps of an activity as a level because these steps are not referenced outside the activity and thus are not reusable.

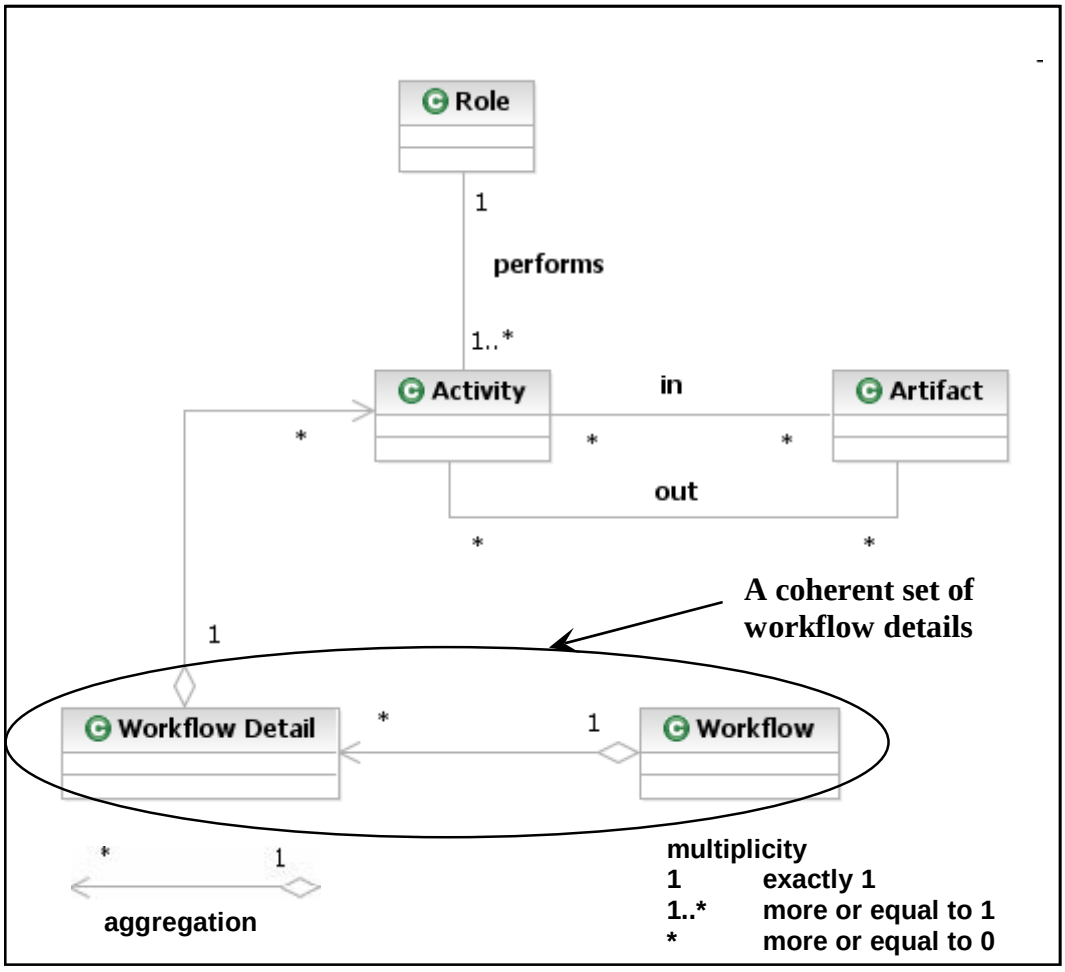


Fig. 4.4 Grouping workflow details into workflow

Fig. 4.4 displays an aggregation between workflow and workflow detail. The aggregation (part-of relationship) models a level of structural hierarchy between workflow and workflow detail.

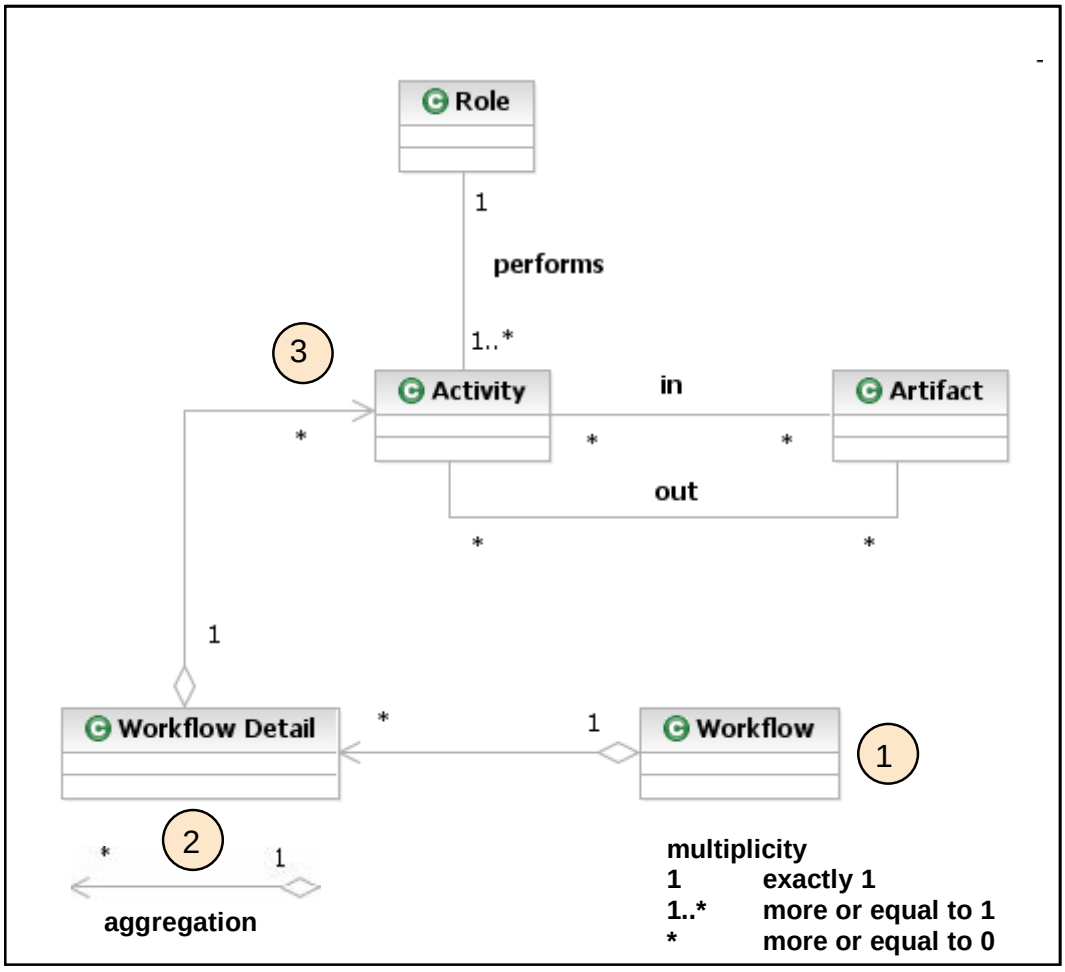


Fig. 4.5 Levels of structural hierarchy into eLup

Fig. 4.5 sums up the 3 levels of structural hierarchy into eLup: workflow, workflow detail and activity.

Levels of structure into IMS LD

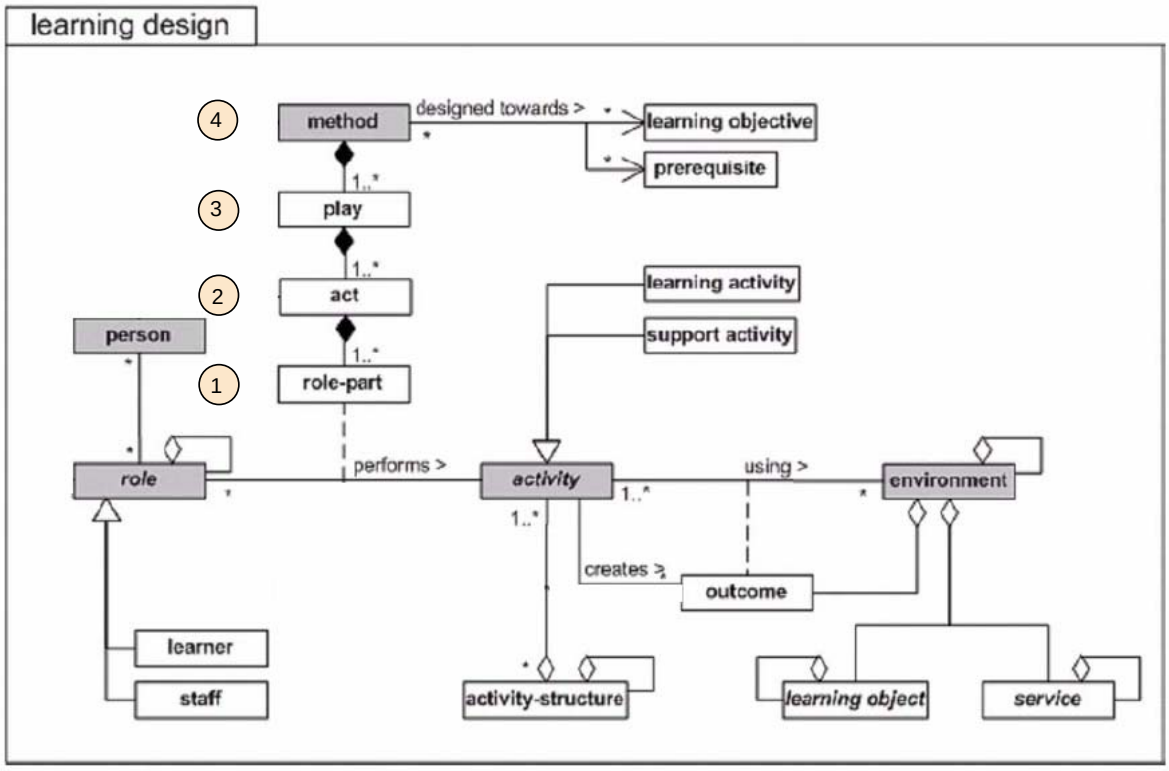


Fig. 4.6 Levels of structural hierarchy into IMS LD

Fig. 4.6 displays the 4 levels of structure into IMS LD level A: method, play, act, role-part.

Comparison between eLup and IMS LD from the structural viewpoint

In eLup, 3 levels are used in the structure. The 4th level, that of activity steps is not made explicit because it is not referenced. IMS LD makes use of 4 levels of structure.

Moreover, IMS LD uses a recursive structure for the *activity-structure*.

Elements will be defined in IMS LD at any level whereas in eLup they will be limited to the 3rd level. For eLup, we preferred simplicity on powerful language.This level fits user needs better.

4.2. Dynamic aspects

Dynamic organization in eLup

In eLup, the dynamic organization of activities is reflected in workflow diagrams.

The user can express whether workflow details are performed in parallel ou in sequence. He/she can express the conditions for transitions from a workflow detail to another one through a test node.

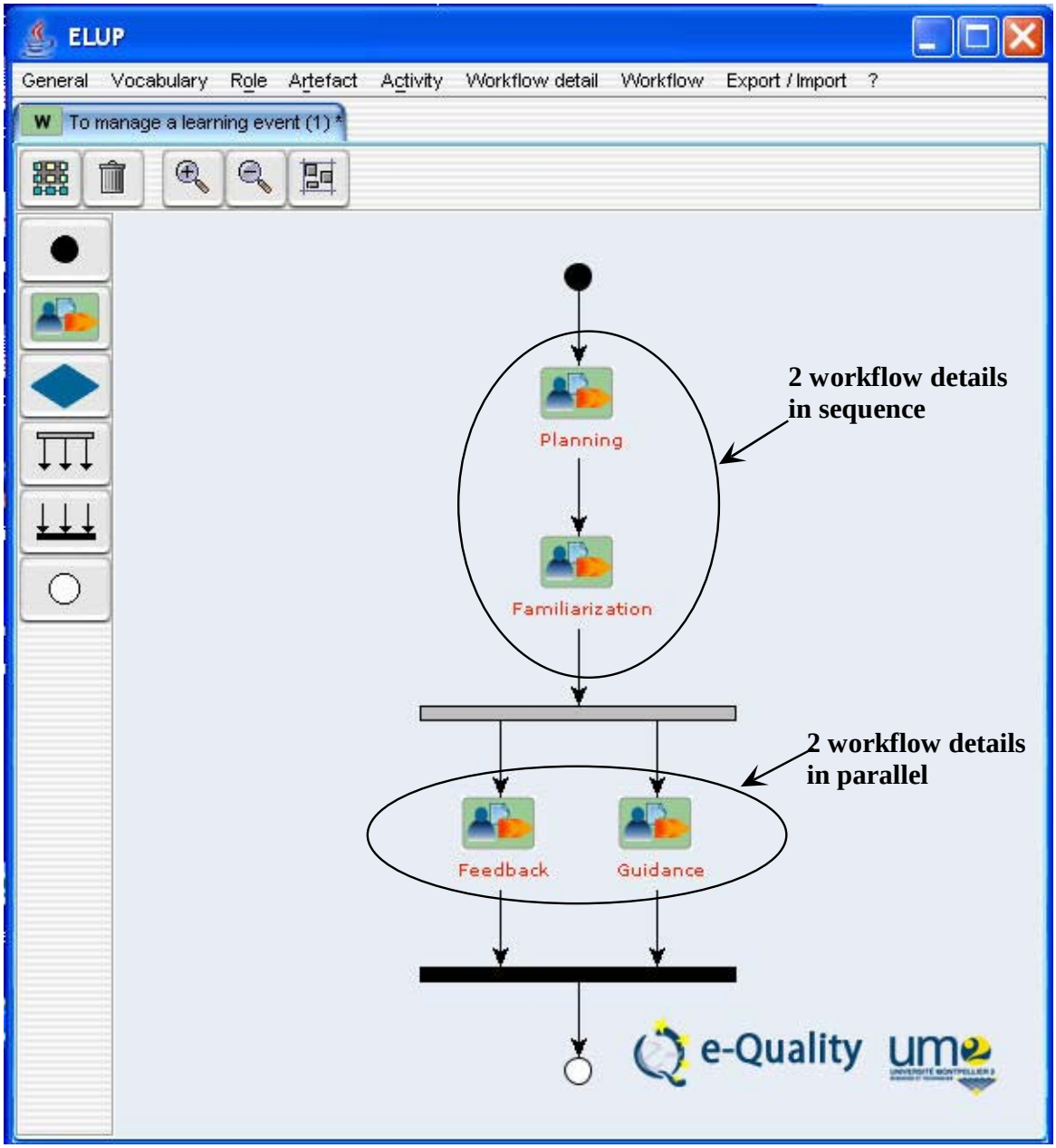


Fig. 4.7 Workflow diagram in eLup

Dynamic organisation in IMS LD

In IMS LD level B, properties related to roles can be added. There are conditions attached to the values of these properties, so conditional executions can be inserted into a sequence of activities (see Fig. 4.8).

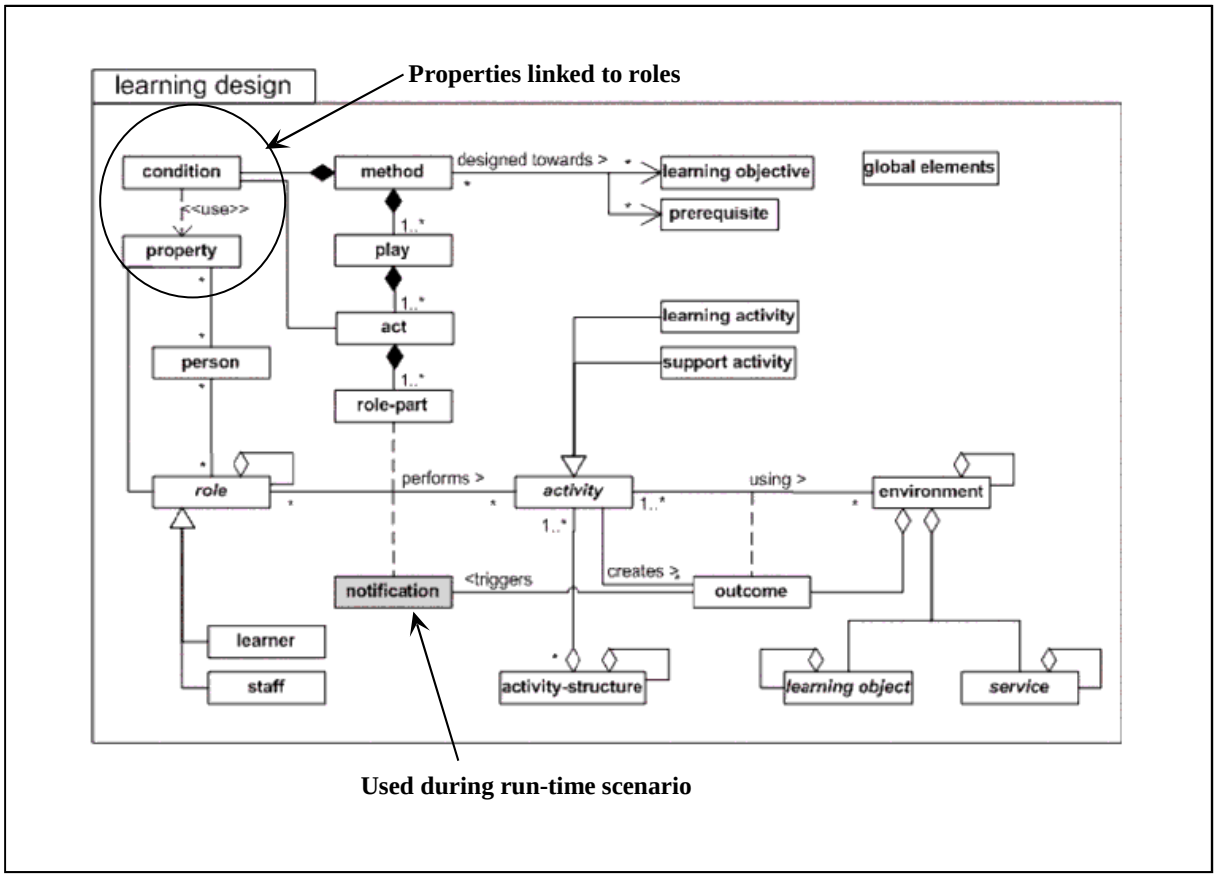


Fig. 4.8 Describing dynamics into IMS LD

In IMS LD level C, notifications have been added. A notification occurs after an event detected by the run-time environment. The learning sequence may be guided in real time by the occurring of events.

Comparison between eLup and IMS LD for the dynamic aspects

The dynamic aspects in eLup and IMS LD do not address the same constraints. eLup is a modelling language derived from Unified Modelling Language (UML). It is dedicated to describing the e-learning process. For this purpose, it makes use of tests ans activities performed in parallel or in sequence. It is not dedicated to being performed on the run-time

environment of a Learning Management System (LMS) as IMS LD. It does not need to specify complex running in real time.

4.3. Languages dedicated to educational modelling.

When the project started in 2003, IMS LD appeared among the educational modelling languages. Since then, its influence has gradually increased to become a standard comparable to SCORM. (see D3.1). eLup differs greatly from IMS LD by its general scope as opposed to the specialized educational scope of IMS LD.

eLup is independent from the educational domain

The concepts to be used in eLup are general and independent of the educational domain. Within UML, the concepts can be specialised by using stereotypes like [Laforcade et al.] did in the modelling language CPM.
In the same way, the concepts used to feature quality in eLup, best practices or quality criteria of the General Quality Process Charter, are not specific to the educational domain.

Features of IMS LD tied to educational domain

Fig. 4.9 highligths the elements dedicated to the educational domain:

- Roles are specialized into Student roles or Staff roles.
- Activities are specialized into Learning activity or support activity.
- The environment uses learning objects or services as resources.
- The method is specified according to learning objective and prerequisite.

No quality approach is made explicit.

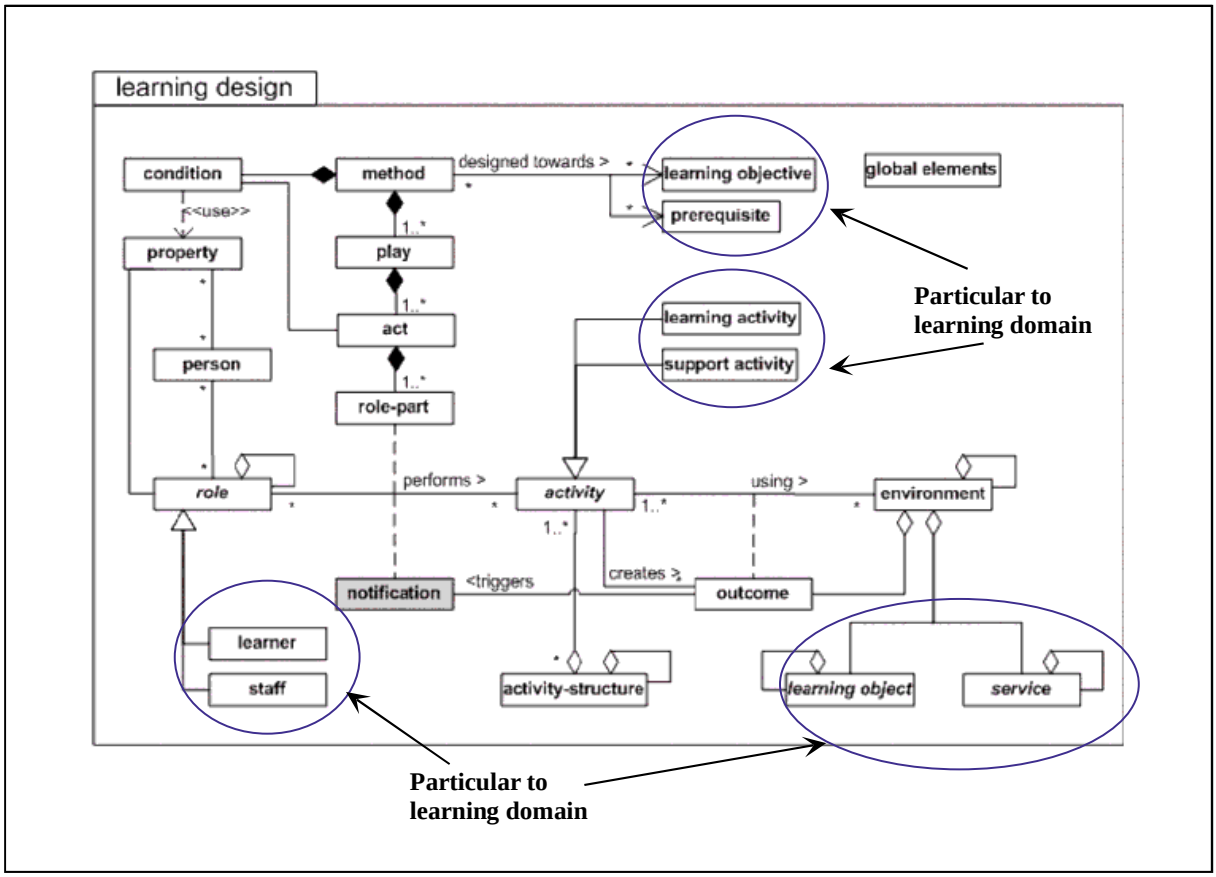


Fig. 4.9 Learning concepts in IMS LD

Comparison between eLup and IMS LD for adding in the quality approach

Our choice tended toward Unified Modelling Language from the outset, with RUP to represent the e-learning processes, and to integrate the quality dimension, because the concepts of activity, role and artefact were well suited to describing working modes and the organisation of e-learning actors, in concrete terms. The choice of these concepts facilitated the expression of quality criteria and best practices. The concepts are simple and should be easy for e-learning professionals to handle.

IMS LD is a rich language to which the quality approach can be added by attaching descriptive elements to an activity, an act and a full-length play.

4.4. Synthesis

IMS LD is a rich language designed for modelling pedagogical scenarios that can be run in a learning management system (LMS).

IMS-LD (IMS Learning Design) uses vocabulary borrowed from the theatre, it defines a play as a set of acts, and for each act there is a distribution of roles. The distribution of roles serves to indicate the activities to be carried out for each role.

A play contains a sequence of acts, corresponding to a workflow that contains a sequence of workflow details. But the workflow defines a sequence in an organised manner, with tests and parallel activity, whereas the play simply defines a linear sequence.

The act defines, through the distribution of roles, a set of activities produced by rules corresponding to a workflow detail. The activities and roles are defined in the same way for both formalisms. But the activities do not use the same vocabulary. In IMS-LD an activity uses an environment and creates an output, which can be assimilated to the input and output artefacts in eLup.

But the two formalisms are not entirely compatible. For one thing, IMS-LD does not define the structure present in workflows, leading to a loss of information in the passage from eLup to IMS-LD.

The translation work between the two languages has begun. eLup and IMS-LD have a common form of expression in XML language. Translation could be partial to recover elements of description from neighbouring concepts. Translation could ensure compatibility at levels A and B of IMS LD.

5. Conclusion

The reader will find a detailed description of eLup functions in the user manual. We hope that this tool, which has required considerable efforts in terms of software development, will be beneficial to the e-learning community.

It is hoped that there will be an ongoing effort to pool the data produced by the community. The eLup software will be widely distributed via the project web site and the DVD publishing its results. Feedback from users will point the way to developing the first collaborative version of the software toward a common construction of e-learning Quality in higher education in Europe.

6. References

[Auzende 2006] The IMS Learning Design specification. Proceedings of the e-Quality Seminar, Szczecin, Poland, on September 21 and 22, 2006, **http://www.e-quality-eu.org**

[Jacobson et al. 1999] Jacobson, I., Booch, G., Rumbaugh, J., The Unified Software Development Process, Addison-Wesley, 1999.

[Jacobson et al. 2000] Jacobson, I., Booch, G., Rumbaugh, J., Le processus Unifié de Développement, Eyrolles 2000.

[Laforcade 2005] Un langage de modélisation pédagogique basé sur UML. Pierre Laforcade, Thierry Nodenot, Christian Sallaberry. Revue STICEF (Sciences et Technologies de l´Information et de la Communication pour l´Éducation et la Formation) Volume 12 Numéro spécial : Conceptions et usages des plates-formes de formation. 2005http://sticef.univ-lemans.fr/num/vol2005/laforcade-05/sticef_2005_laforcade_05.htm

[RUP 2003] **http://www-306.ibm.com/software/awdtools/rup/index.html**

[Rumbaugh et al. 1999] Rumbaugh, J., Jacobson, I., Booch, G., UNIFIED MODELING LANGUAGE REFERENCE MANUAL 2ND EDITION. 2004.

[UML 2004] Rumbaugh, J., Jacobson, I., Booch, G., UML 2.0 - GUIDE DE REFERENCE. CAMPUS PRESS. 2004

[IMS LD] **http://www.imsglobal.org/learningdesign/**

[D2.1] Deliverable D2.1Quality issues in partners' countries – Synthesis Report. e-Quality public deliverable in English.

http://www.e-quality-eu.org

[D2.2] Deliverable D2.2: Conceptual model for ODL Quality process and evaluation grid, criteria and indicators. e-Quality public deliverable in English. **http://www.e-quality-eu.org**

[D3.1] General information on quality process. e-Quality public deliverable in French and English. **http://www.e-quality-eu.org**

[CGPQ] Charte Générale du Processus Qualité. e-Quality public deliverable in French, English, Spanish, Polish, German, Italian. **http://www.e-quality-eu.org**

[D3.3] Best Practices Database. e-Quality public deliverable in English. http://www.e-quality-eu.org

[D4] Deliverable D4. Trainer's and Professional's Guide to Quality in Open and Distance Learning. e-Quality public deliverable in English. **http://www.e-quality-eu.org**

[D5] Deliverable D5. Training sessions global evaluation report. e-Quality public deliverable in English. **http://www.e-quality-eu.org**

[D6] Deliverable D6. Training Session Evaluation. Methodology and results. e-Quality public deliverable in English. **http://www.e-quality-eu.org**

7. Table des figures

Fig. 2.1 Roles and Actors 10
Fig. 3.1 IMS Learning Design Level A 13
Fig. 3.2 IMS LD Level B 14
Fig. 3.3 IMS LD Level C 15
Fig. 4.1 eLup meta-model 17
Fig. 4.2 Use of concepts 18
Fig. 4.3 Grouping activities into workflow details 19
Fig. 4.4 Grouping workflow details into workflow 20
Fig. 4.5 Levels of structural hierarchy into eLup 21
Fig. 4.6 Levels of structural hierarchy into IMS LD 22
Fig. 4.7 Workflow diagram in eLup 23
Fig. 4.8 Describing dynamics into IMS LD 24
Fig. 4.9 Learning concepts in IMS LD 26

eLUP

User Manual

TABLE OF CONTENTS

1. INSTALLING THE SOFTWARE 35

1.1. HOW TO INSTALL 35

Step 1: Selecting the installation language 35

Step 2: Language options 36

Step 3: Preparing the installation assistant 36

Step 4: Starting the installation assistant 37

Step 5: Display of license for software use 37

Step 6: Selecting a target directory 38

Step 7: *Read me* displayed 38

Step 8: selecting the type of installation - standard or customized 40

Step 9: selecting the components in a customized installation 40

Step 10: Choosing 41

Step 11: Choice display 42

Step 12: Installation starts 42

Step 13: starting the Apache and MySQL servers 43

Step 14: Installation completed 43

2. LAUNCHING THE SOFTWARE 44

3. HOW ELUP APPLICATION IS ORGANISED 45

3.1. DEFINING A CONCEPT 47

4. ORGANISATION OF THE USER INTERFACE 48

4.1. LOGIN WINDOW 48

4.2. HOME WINDOW 49

4.3. GENERAL MENU 50

4.4. VOCABULARY MENU 52

4.5. ROLE MENU 53

4.6. ARTEFACT MENU 54
4.7. ACTIVITY MENU 55
4.8. WORKFLOW DETAIL MENU 56
4.9. WORKFLOW MENU 58
4.10. EXPORT/ IMPORT MENU 59
4.11. HELP MENU 60

5. EDITORS 61

5.1. DEFINITION EDITOR IN THE *VOCABULARY* MENU 61
5.2. ACTIVITY COMPOSER EDITOR 67
5.3. THE WORKFLOW DETAIL DIAGRAM EDITOR 68
5.4. WORKFLOW DIAGRAM EDITOR 70
5.5. DEFINING QUALITY 73
5.6. DEFINING THE QUALITY OF AN ACTIVITY 79
5.7. DEFINING THE QUALITY OF A WORKFLOW DETAIL 80
5.8. DEFINING WORKFLOW QUALITY 82

6. GENERATING DOCUMENTATION 84

7. TABLE OF FIGURES 86

User Manual

1. Installing the software

eLUP software is written in Java programming language (version 5). It was developed at Montpellier University 2. For the graphics editor, it uses *yFILES*, a graphics library developed by the firm yWorks (http://www.yworks.com/). It uses the data base management system *MySQL*.

To run it on a PC computer in a Windows XP environment, you first need to install the software EasyPHP 1.8 **(**http://www.easyphp.org/**)** including *MySQL* and to run the Java application, the J2SE environment JRE 5.0 (http://java.sun.com/javase/downloads/index_jdk5.jsp).

During standard installation, the eLup installer installs EasyPHP and JRE.

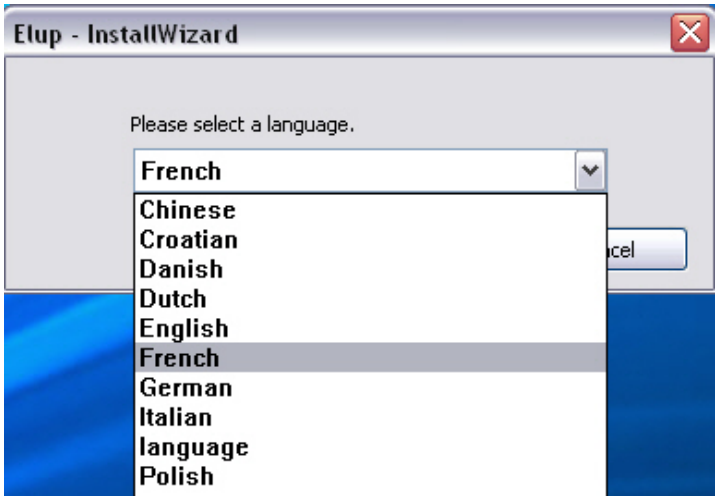
The eLUP software was implemented and tested with EasyPHP 1.8 and JRE 5. The eLUP software is not a commercial product. It is a prototype, developed in a university framework. Other configurations are potentially viable but have not been tested. Elup is configured to be used with at least 500 Mo RAM.

1.1. How to install

The following screens show detailed instructions for installation procedure.

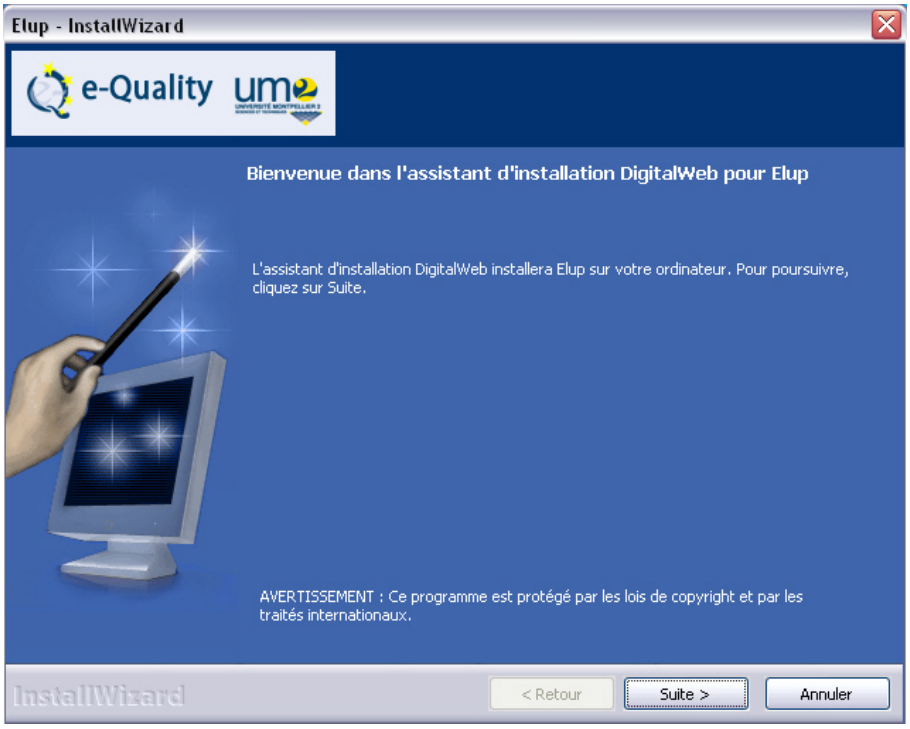
Step 1: Selecting the installation language

Step 2: Language options



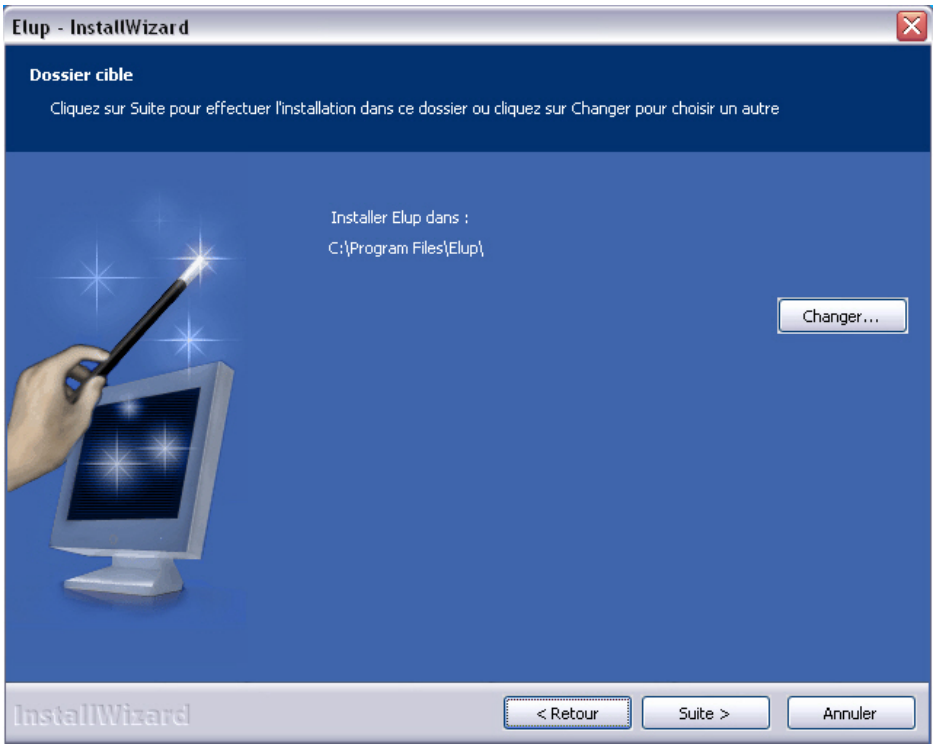
Step 3: Preparing the installation assistant

Step 4: Starting the installation assistant



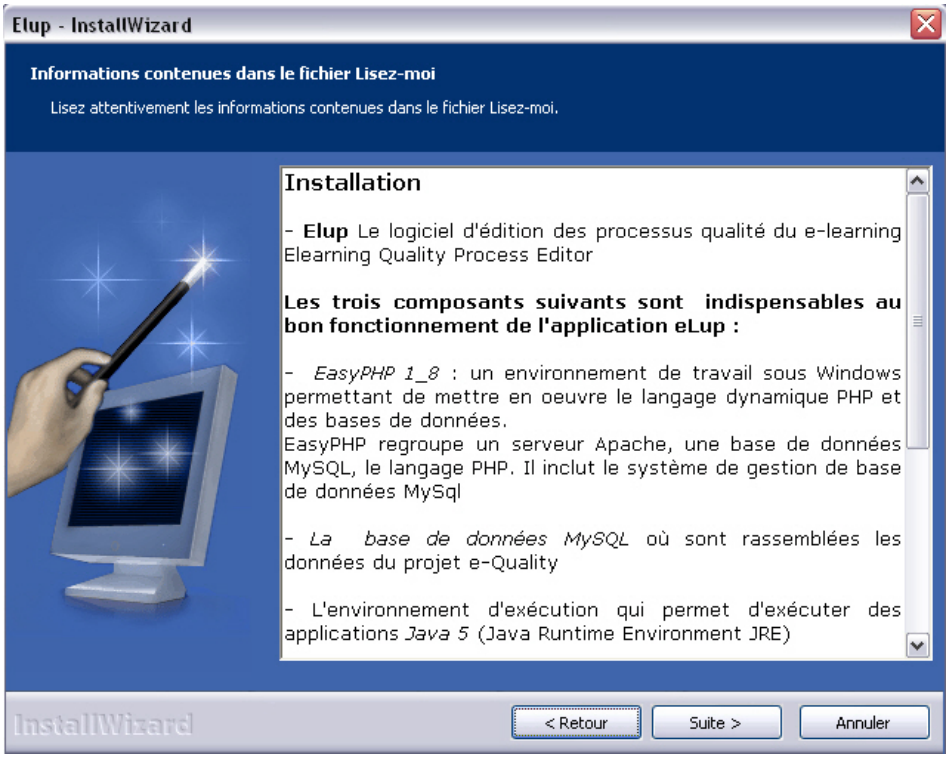
Step 5: Display of license for software use

Step 6: Selecting a target directory



The user selects the directory in which to install eLup. By default, the software is installed in directory c:\Program Files\Elup

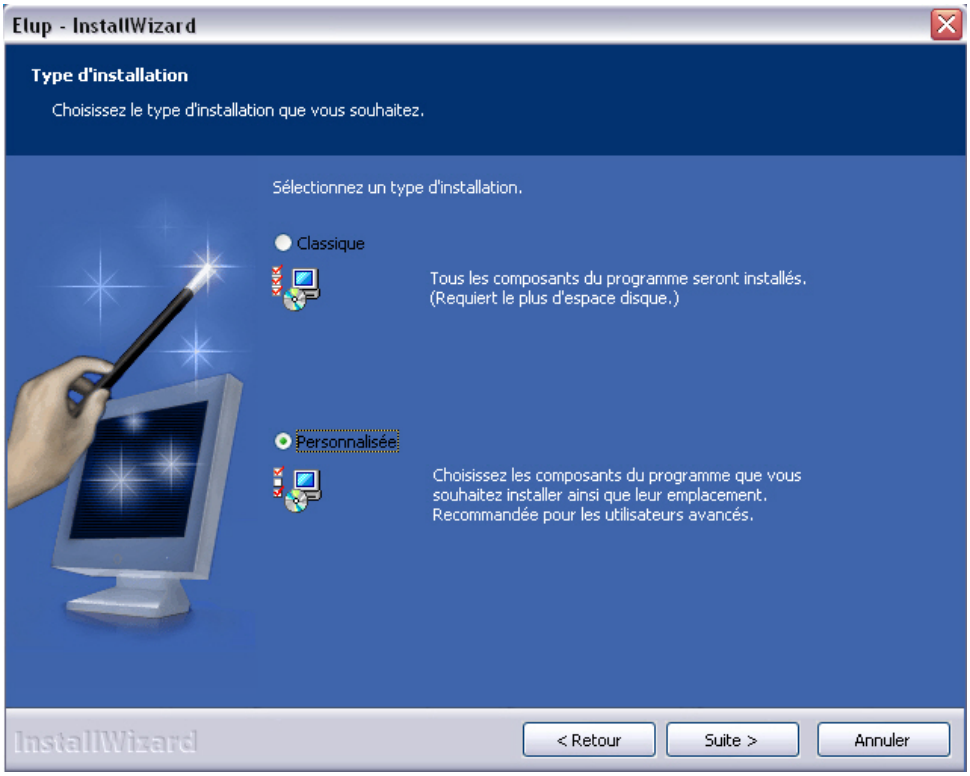
Step 7: *Read me* displayed



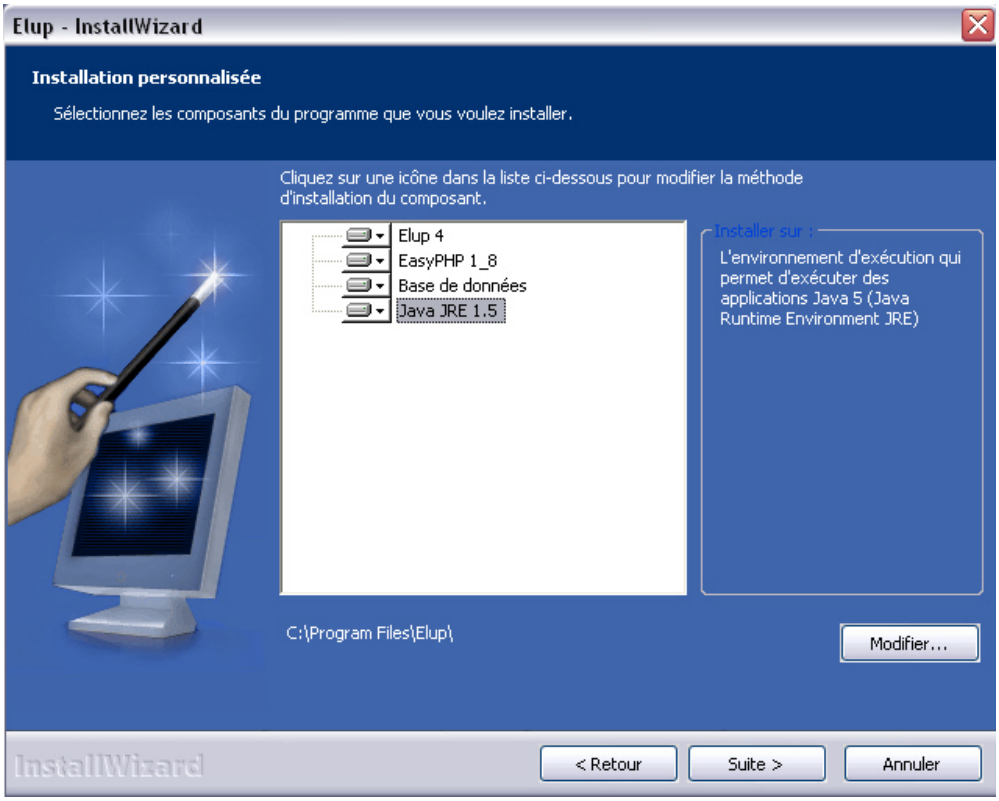
The user is informed of the software configuration required for eLup to function correctly.

eLup requires *EasyPHP1_8*, *MySQL*, *JRE*. These three components are essential for the eLup application to function correctly. If the user has already installed some of these components on his/her computer, s/he can select a personalised installation according to the components s/he wishes to install.

Step 8: selecting the type of installation - standard or customized

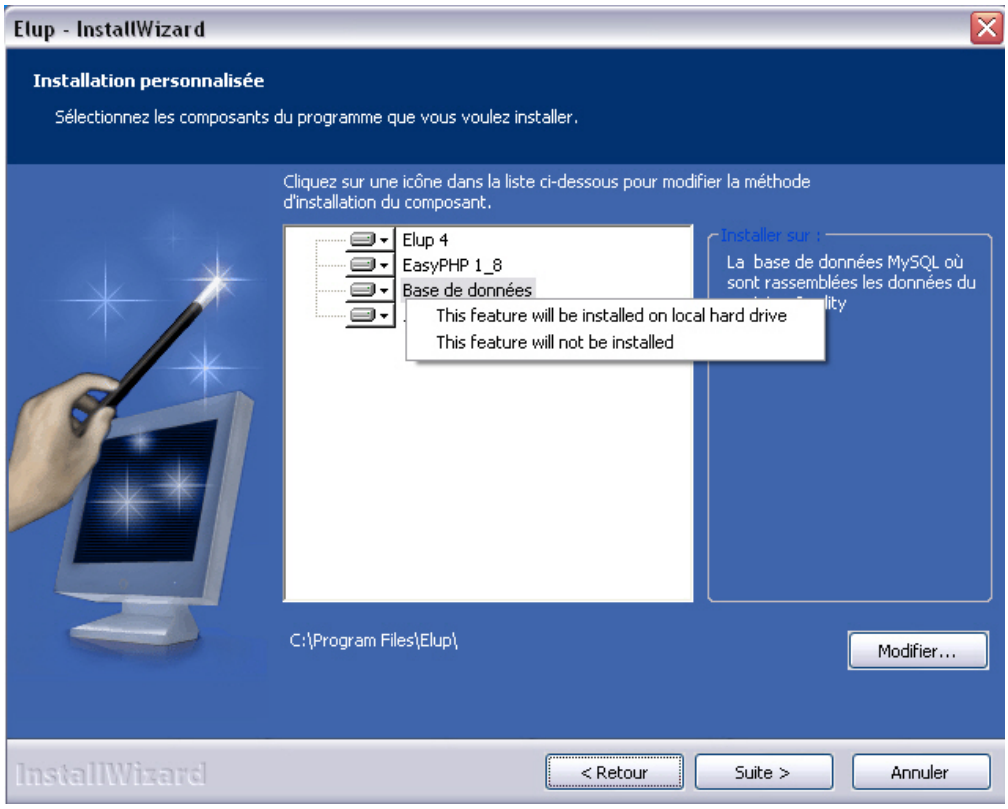


Step 9: selecting the components in a customized installation



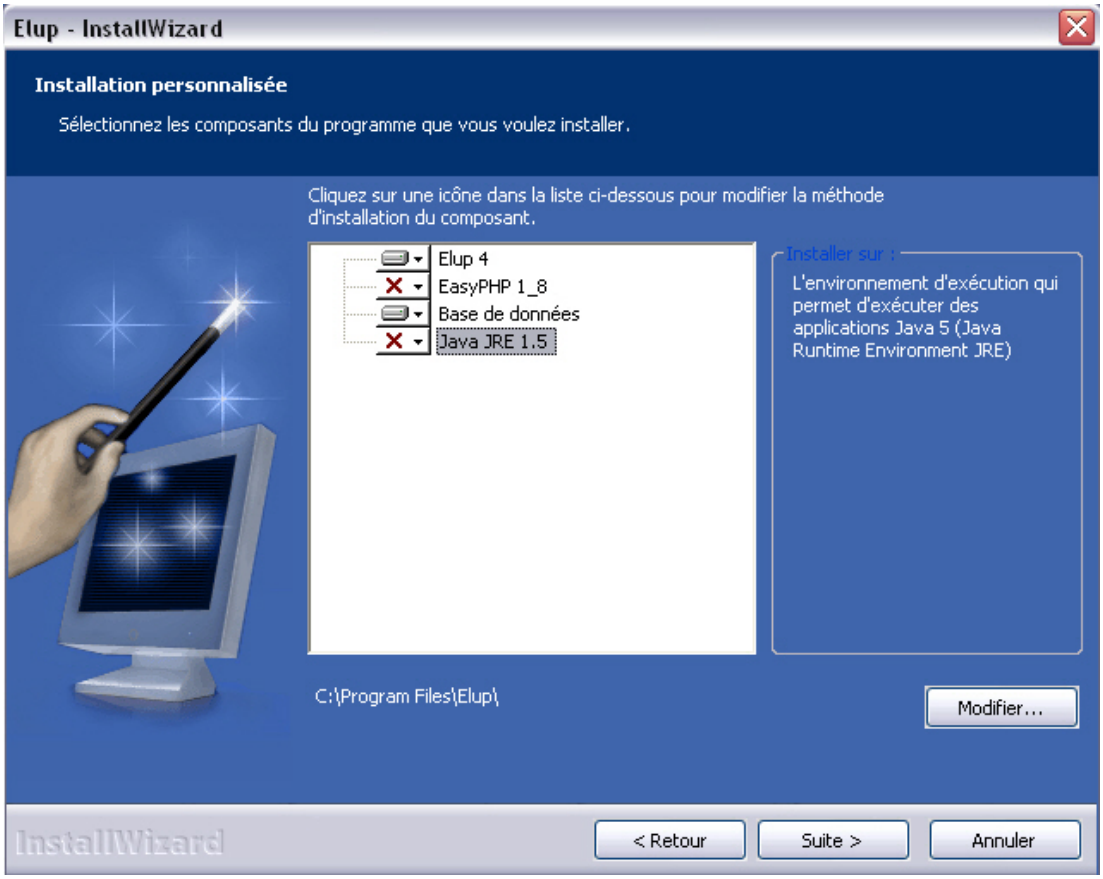
The user who already has the software component proposed, can choose to install only the complementary components. eLup software is a prototype and the proposed configuration has been tested. Any other configuration will be tested by the user.

Step 10: Choosing



To choose, the user must select the corresponding option.

Step 11: Choice display

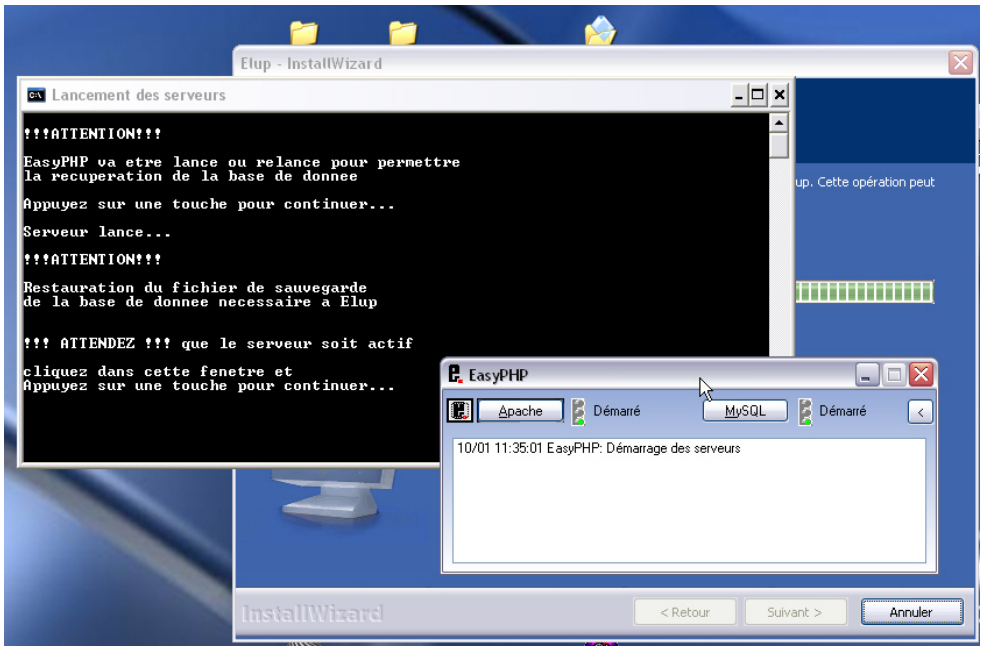


The options that are not selected will be marked with a red cross.

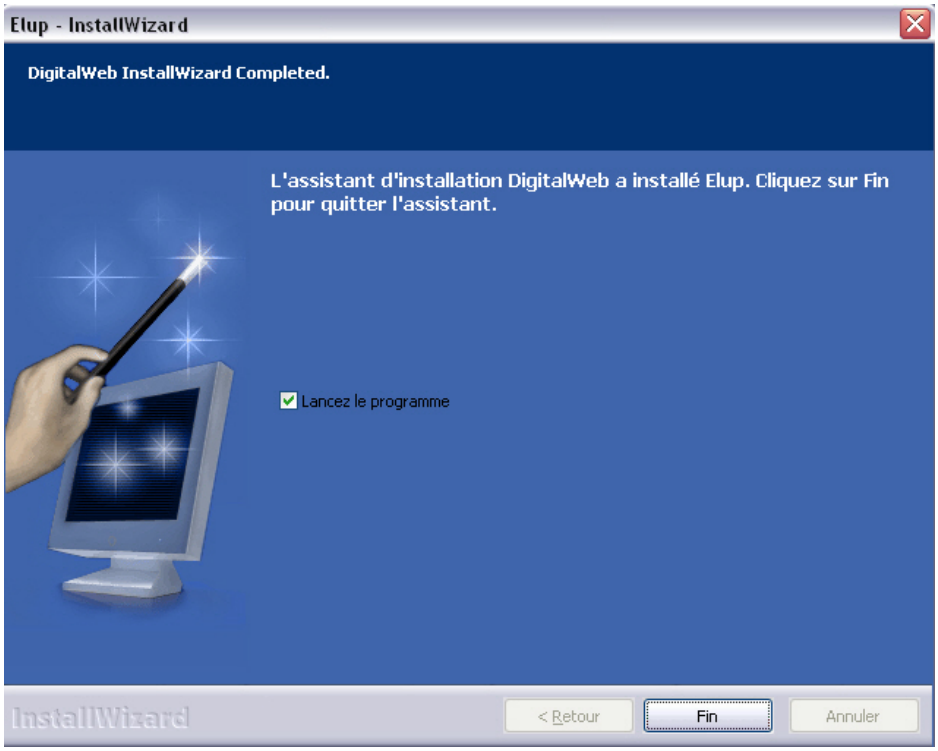
Step 12: Installation starts

To integrate the eLup database requires starting EasyPHP. The user will see the start message in a command prompt window.

Step 13: starting the Apache and MySQL servers



Step 14: Installation completed



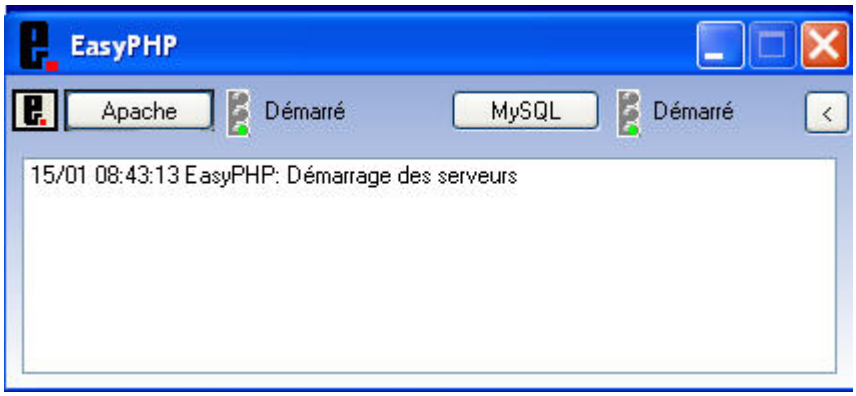
Installation completed, eLup is ready to go!

2. Launching the software

Step 1

In the menu *Start* follow the link *All programs* then click on *EasyPHP*.

Apache and MySQL are started.



Step 2

Launch *eLup* from the icon installed on the desktop

In the menu *Start* follow the link *All programs* then click on *eLup*.

3. How eLUP application is organised

eLUP software allows the user to manipulate the concepts described in the **Conceptual Guide**. These include roles, artefacts, activities, workflow details or workflows. These concepts will be represented in the software at level one: each role, artefact, activity, workflow detail or workflow will first be defined. By defining a concept the user can specify its meaning and apply any synonymous terms.

A role or an artefact are simple concepts and only require one definition.

An activity, a workflow detail or a workflow, are complex concepts. They must first be defined. They are subject to the following operations:

- For an activity once defined (Fig. 3.1 Steps to specify an activity):
  - The user can *compose* an activity, i.e. assign a role, input and output artefacts, define its steps.
  - The user can *define the quality* of an activity, i.e. attribute *Quality* characteristics to an activity: one (or several) best practice(s) or one (or several) documented quality criteria.

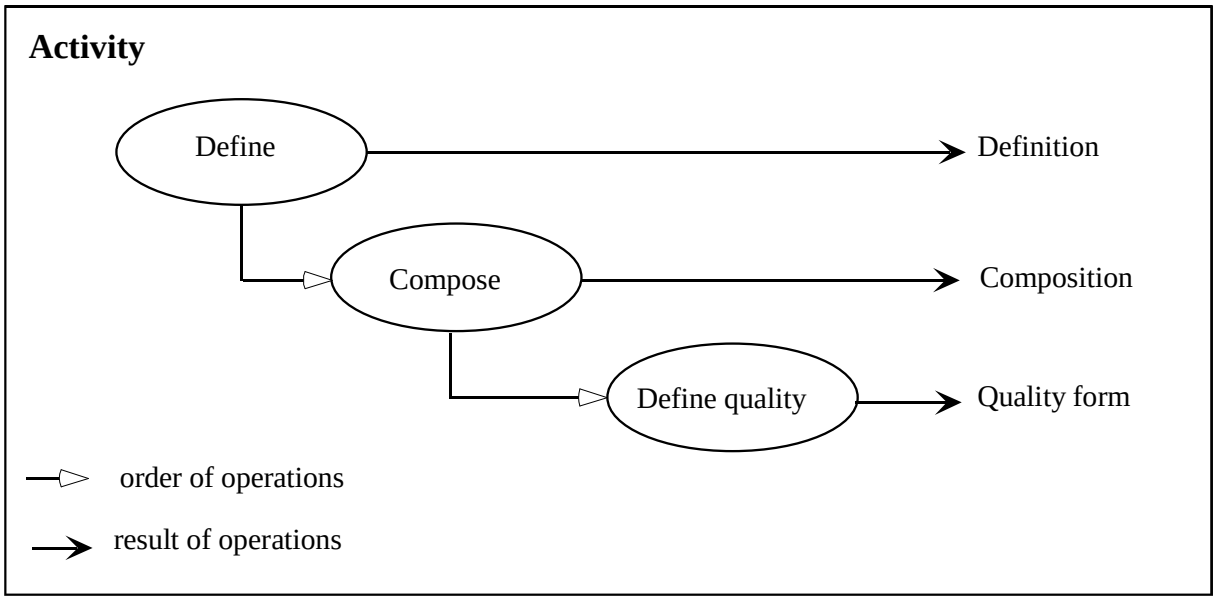


Fig. 3.1 Steps to specify an activity

- For a workflow detail once defined (Fig. 3.2):
  - The user can specify the list of activities in the workflow detail and view these in a graphics editor, *Workflow Details editor*. In this editor, s/he can view the activities that make up the set. The activities are grouped by role, and their input and output artefacts are displayed. Only composed activities can be included in the workflow detail; activities that are merely defined are not sufficiently specified for this.
  - The user can *define the quality* of a workflow detail, i.e. attribute *Quality* characteristics to a workflow detail: one (or several) best practice(s) or one (or several) documented quality criteria.

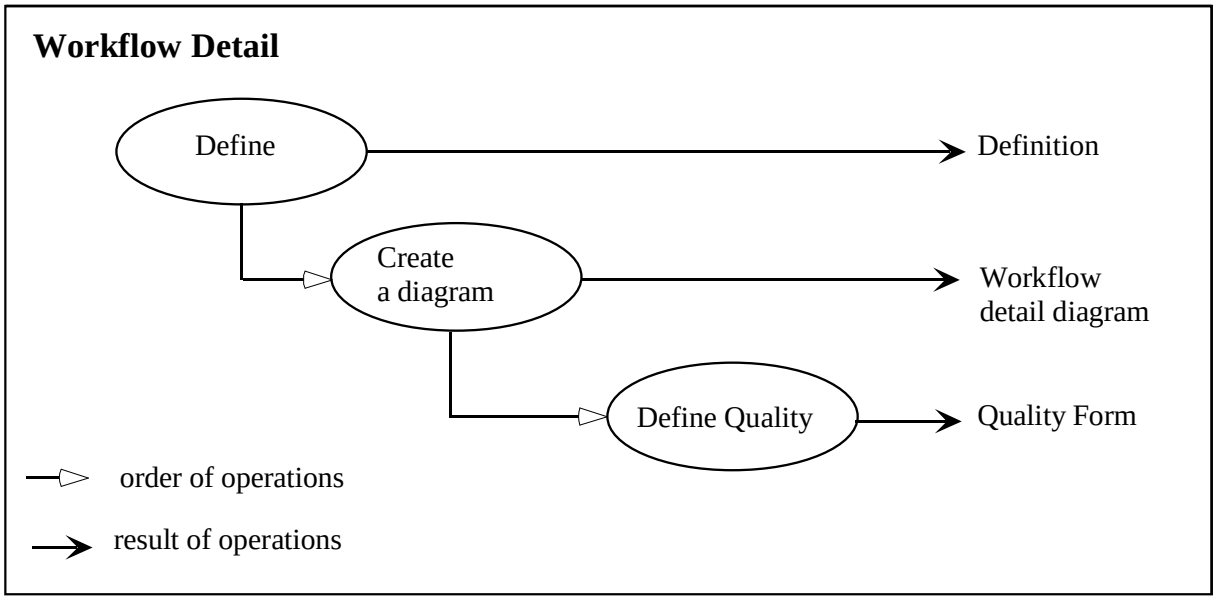


Fig. 3.2 Steps to specify a *workflow detail*

- For a workflow once defined (Fig. 3.3):
  - The user can specify the workflow details in a *workflow diagram*, and their relations in a graphics editor *the workflow editor*. In this editor s/he can indicate the initial state and final state of the workflow. S/he can link up the workflow details that make up the workflow by inserting tests or indicating that these workflow details can be run in parallel. The graphics editor uses activity diagrams from Unified Modelling Language (UML).
  - The user can *define the quality* of a workflow detail, i.e. attribute *Quality* characteristics to a workflow detail: one (or several) best practice(s) or one (or several) documented quality criteria.

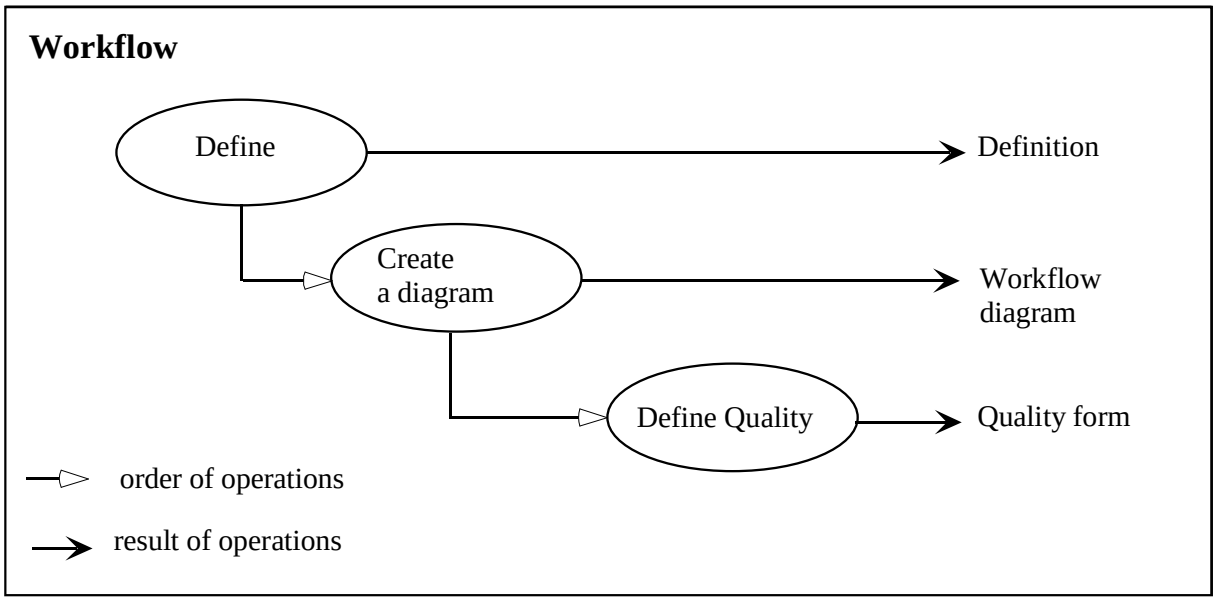


Fig. 3.3 Steps to specify a *workflow*

3.1. Defining a concept

Defining a concept can be accessed in the *Vocabulary* menu or any of the dedicated menus, *Role*, *Artefact*, *Activity*, *Workflow detail* and *Workflow*. In the *Vocabulary* menu, specify the category to which the concept belongs. Apart from this difference, the entry windows dedicated to each concept category, used to edit the concept definitions, are identical. By defining a concept the user can specify its meaning and apply any synonymous terms.

4. Organisation of the user interface

When starting up the eLUP software the user must first log in. His/her identification will distinguish his/her own data from that of the e-Quality project. The user can only modify copies of project data. This prevents the data being altered if the user makes a false move. Moreover if the user wants to exchange data with other users, this data will be identified by origin. On start-up s/he is invited to create a new account that will be automatically validated. The single-user version requires no prior authorisation for use.

4.1. Login window

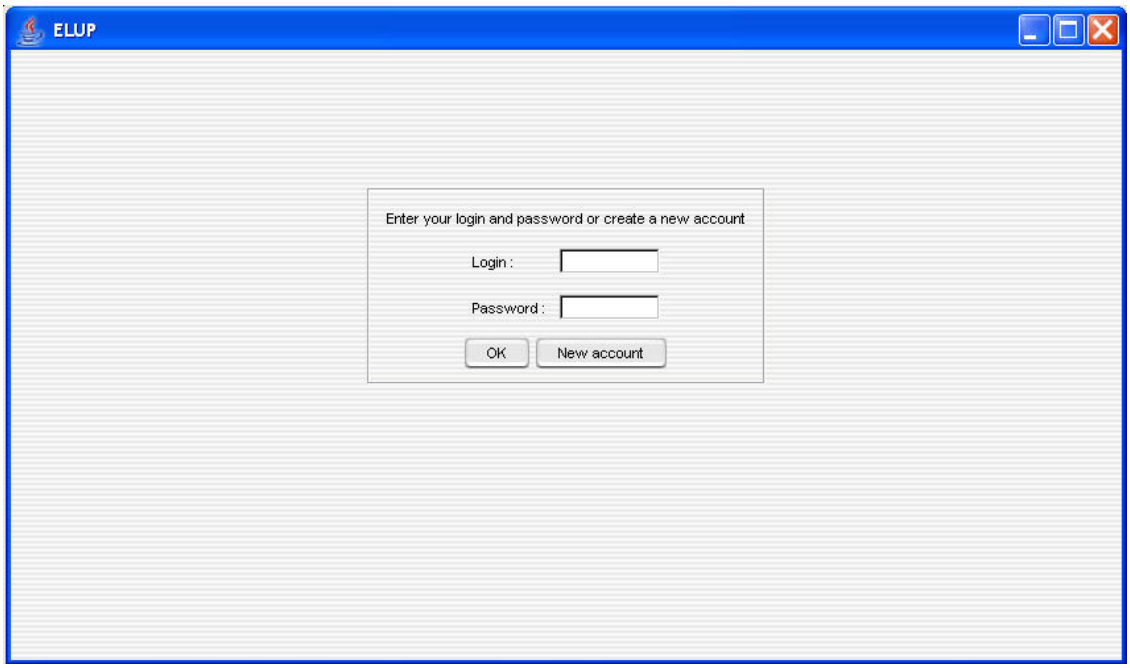


Fig. 4.1 Login window

The user logs in or asks to create a new account. In the latter case, s/he fills in an information form (Fig. 4.1).

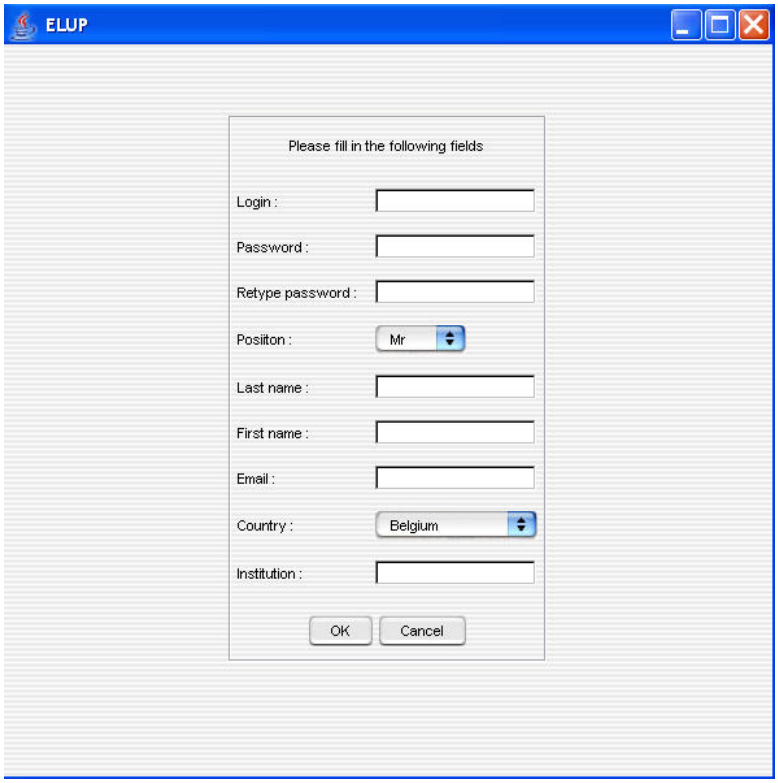


Fig. 4.2 Creating a new account

4.2. Home window

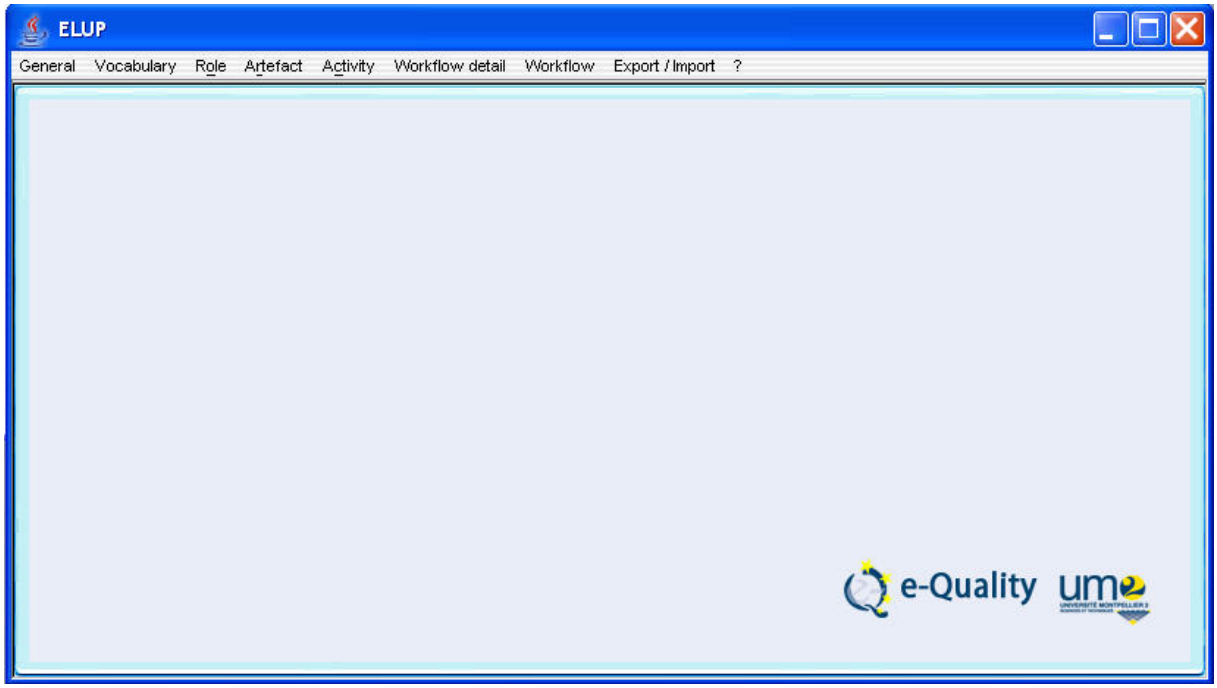


Fig. 4.3 Home window

4.3. General menu

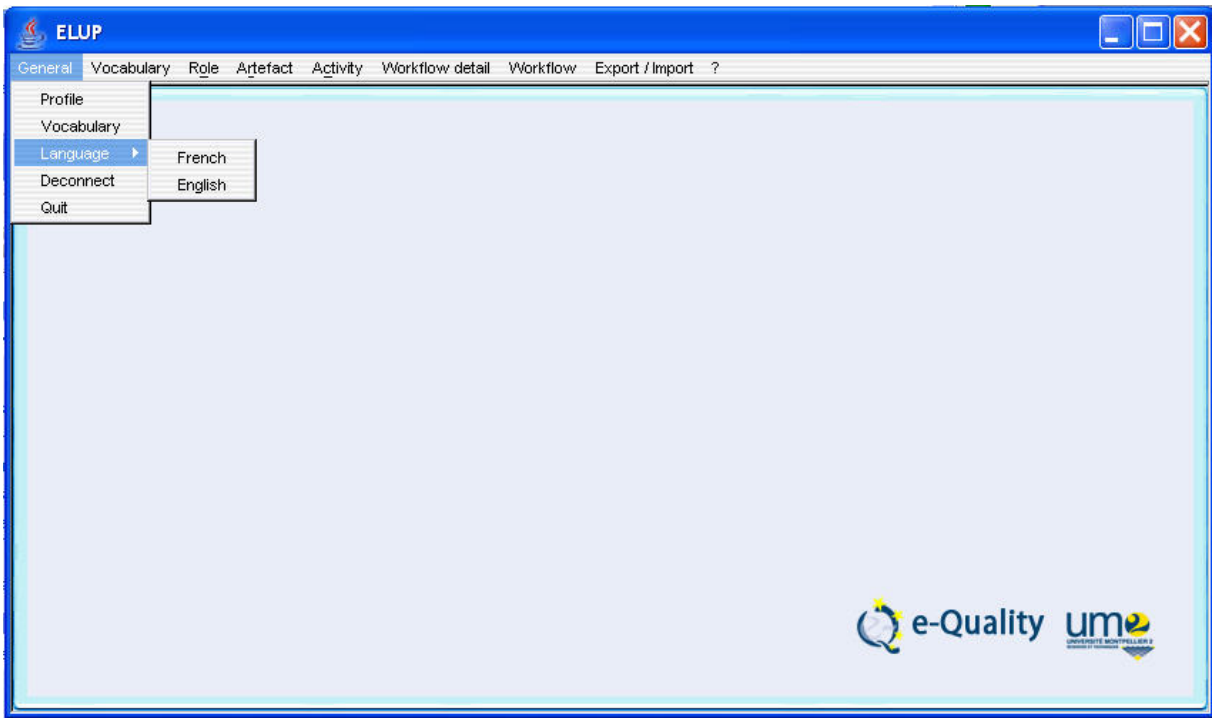


Fig. 4.4 *General* menu

Item	Action
Profile	To modify the initial user profile created when creating the account.
Vocabulary	To enter the definition of a concept of any category.
Language	To select the user interface language: French or English.
Log off	To allow the user to log off without quitting the software. The user can then log onto another account.
Quit	To quit the software.

Tab. 4.1 Action linked to an item in the *General* menu

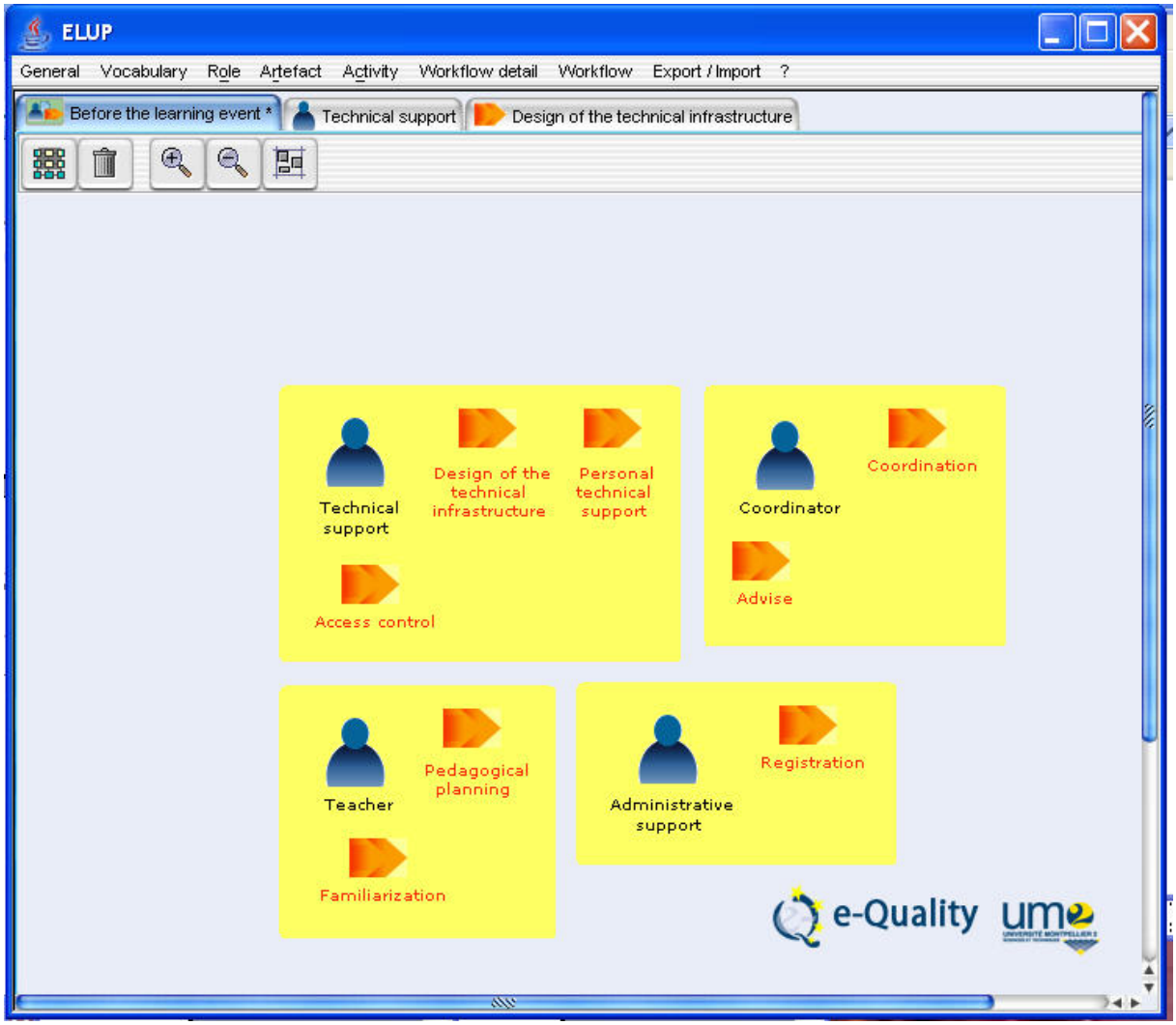


Fig. 4.5 Tabs

Comment: The eLup user interface allows several objects to be edited simultaneously. The windows open in different tabs (Fig. 4.5). The only restriction is that the same object cannot be opened in two different edition windows. Any modification of an object in a tab will update it in all the tabs where it is listed.

4.4. Vocabulary menu

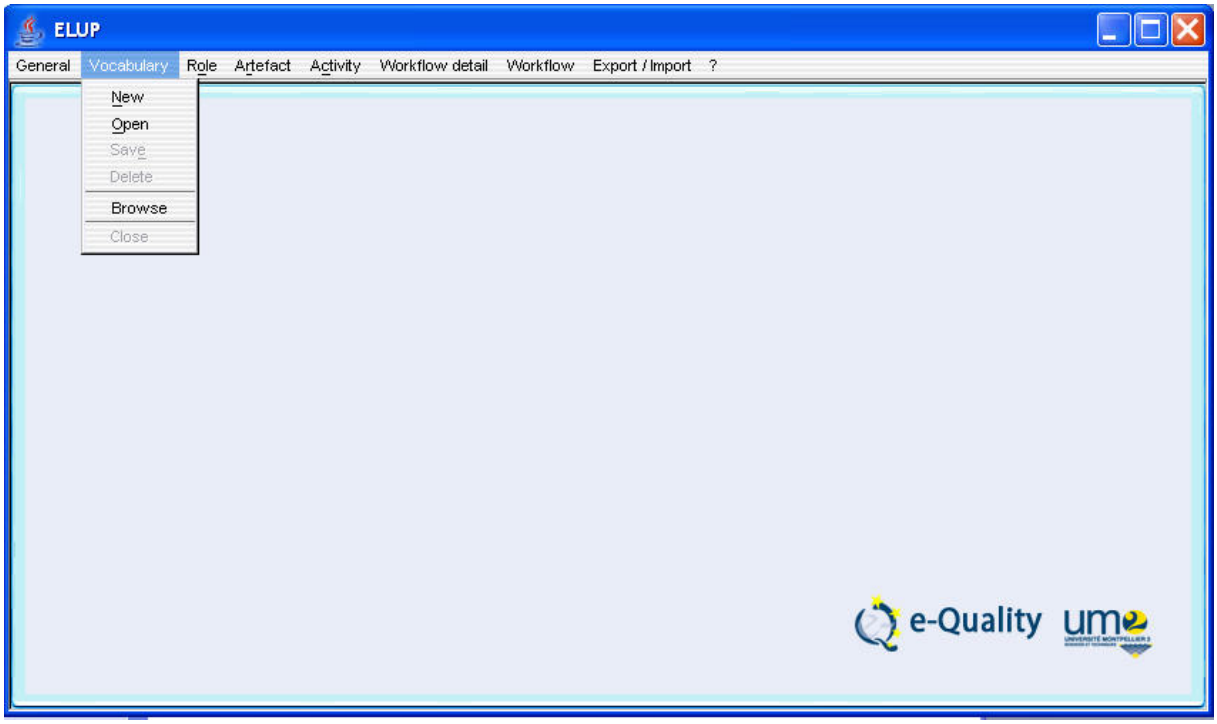


Fig. 4.6 Vocabulary menu

Item	Action
New	To create the definition of a new term by indicating its category.
Open	To edit a definition. If the definition comes from an author other than the connected user, eLup suggests creating a new version of the definition.
Save	To save the definition while editing.
Delete	To delete the definition while editing.
Browse	To browse the definition (read without modifying). eLup opens a dialogue box to select the definition to browse.
Close	To close the definition while editing. If the definition has been modified but not saved, eLup asks for confirmation.

Tab. 4.2 Action linked to an item in the *Vocabulary* menu

4.5. Role menu

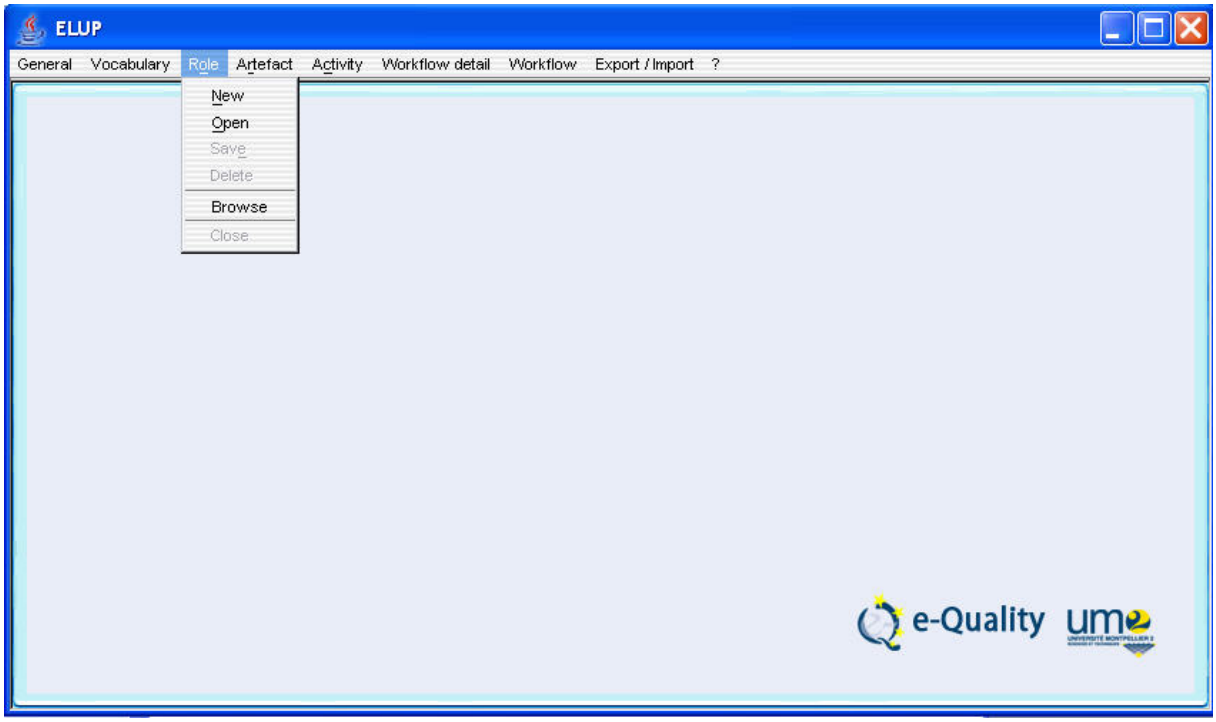


Fig. 4.7 *Role* menu

Item	Action
New	To create the definition of a new role.
Open	To edit the definition of a role. If the definition comes from an author other than the connected user, eLup suggests creating a new version of the definition.
Save	To save the definition of a role while editing.
Delete	To delete the definition of a role while editing.
Browse	To browse (read without modifying) the definition of a role. eLup opens a dialogue box to select the definition to browse.
Close	To close the definition of a role while editing. If the role has been modified but not saved, eLup asks for confirmation.

Tab. 4.3 Action linked to an item in the *Role* menu

4.6. Artefact menu

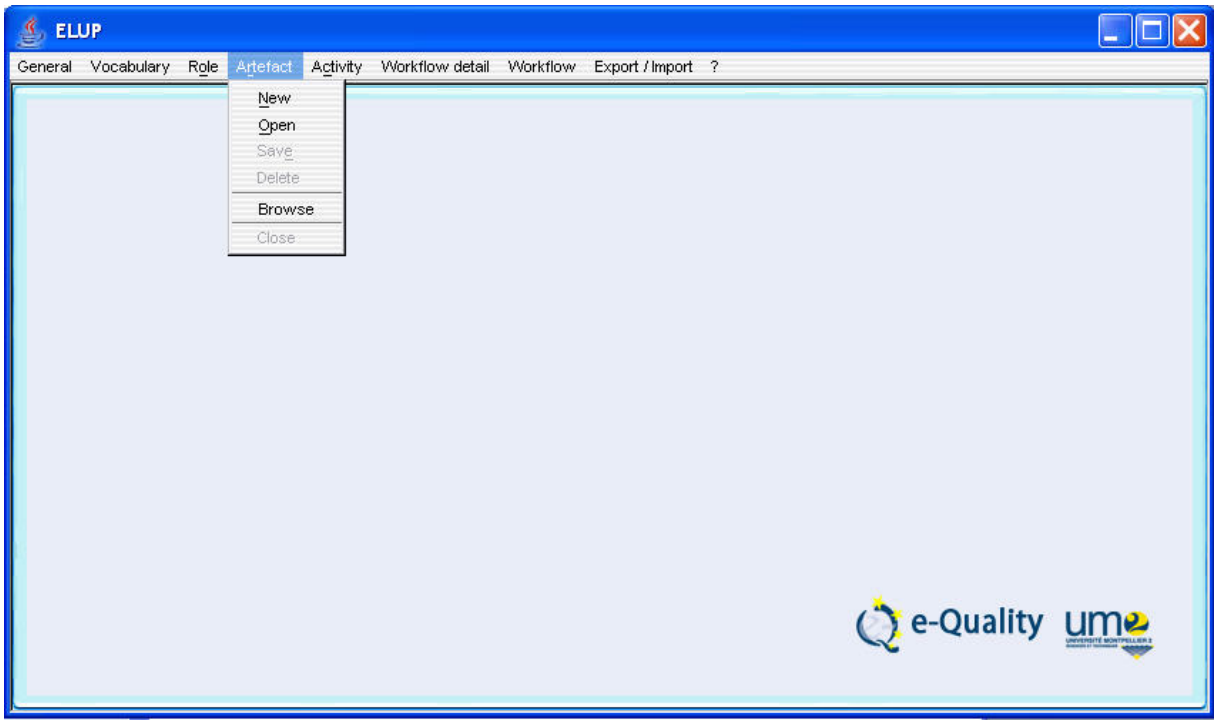


Fig. 4.8 *Artefact* menu

Item	Action
New	To create the definition of a new artefact.
Open	To edit the definition of an artefact. If the definition comes from an author other than the connected user, eLup suggests creating a new version of the definition.
Save	To save the definition of an artefact while editing.
Delete	To delete the definition of an artefact while editing.
Browse	To browse (read without modifying) the definition of an artefact. eLup opens a dialogue box to select the definition to browse.
Close	To close the definition of an artefact while editing. If the artefact has been modified but not saved, eLup asks for confirmation.

Tab. 4.4 Action linked to an item in the *Artefact* menu

4.7. Activity menu

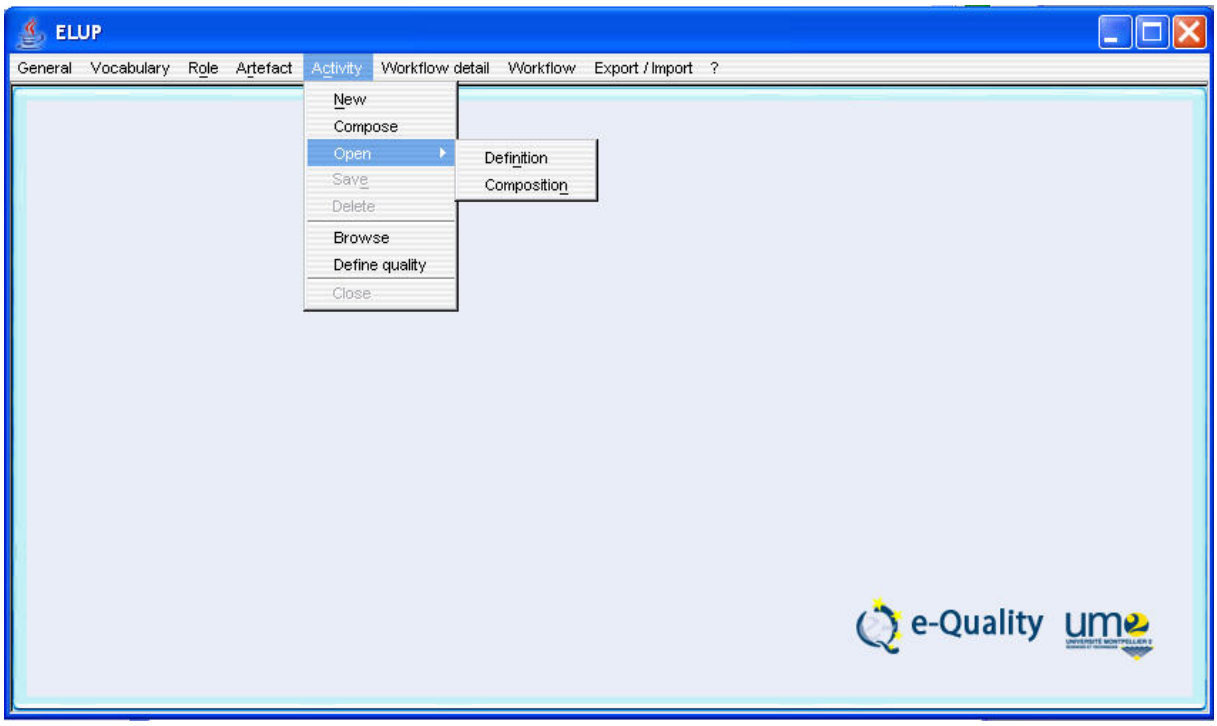


Fig. 4.9 *Activity* menu

Item	Action
New	To create the definition of a new activity.
Compose	To compose an activity. If this activity has already been composed, eLup suggests creating a new version of the composition or editing an existing composition.
Open Definition	To edit the definition of an activity. If the definition comes from an author other than the connected user, eLup suggests creating a new version of the definition.
Open Composition	To edit the composition of an activity. If the composition comes from an author other than the connected user, eLup suggests creating a new version of the composition.
Save	To save the definition or the composition of an activity while editing.

Tab. 4.5 Action linked to an item from the *Activity* menu (a)

Delete	To delete the definition or the composition of an activity while editing.
Browse	To browse (read without modifying). eLup opens a dialogue box to select the definition or the composition of an activity to browse.
Define quality	To create a quality form for an activity. Before doing this, the composition of the activity must already exist. The quality form has three panels: one panel summarising the definition and composition of the activity, one "Best practices" panel, and a "General Quality Process Charter"
Close	To close the definition or composition while editing.

Tab. 4.6 Action linked to an item from the *Activity* menu (b)

4.8. Workflow detail menu

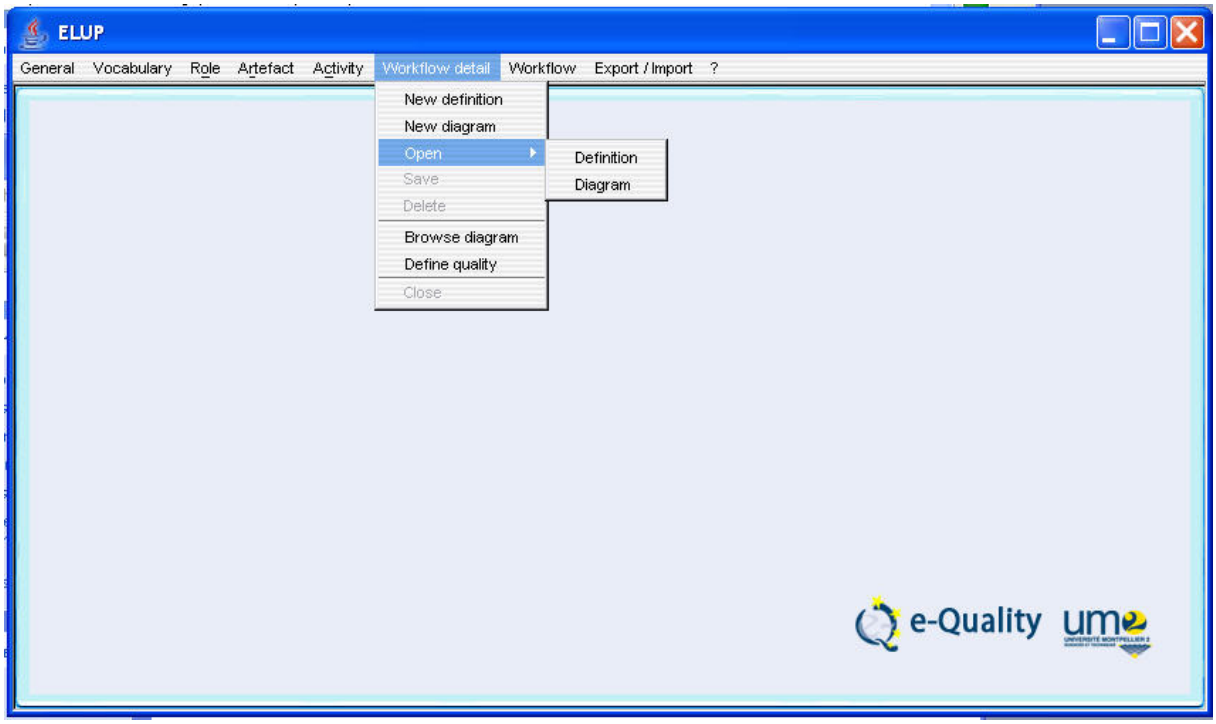


Fig. 4.10 *Workflow detail* menu

Item	Action
New definition	To create the definition of a new workflow detail.
New diagram	To create a new diagram linked to the definition of a workflow detail. eLUP opens the graphics editor dedicated to workflow detail diagrams with an empty panel.
Open definition	To edit the definition of a workflow detail. If the definition comes from an author other than the connected user, eLup suggests creating a new version of the definition.
Open Diagram	To edit a diagram linked to the definition of a workflow detail. eLUP opens the graphics editor of the workflow detail. If the diagram comes from an author other than the connected user, eLup suggests creating a new version of the diagram.
Save	To save the definition or the diagram of a workflow detail while editing.
Delete	To delete the definition or the diagram of a workflow detail while editing.
Browse diagrams	To browse (read without modifying). eLup opens a dialogue box to select the diagram to browse.
Define quality	To create the quality form for a workflow detail. Before doing this, the diagram must already exist. The quality form has three panels: one panel displaying the workflow detail diagram, one “Best practices” panel, and a “General Quality process chart”
Close	To close the definition or the diagram while editing.

Tab. 4.7 Action linked to an item in the *Workflow detail* menu

4.9. Workflow menu

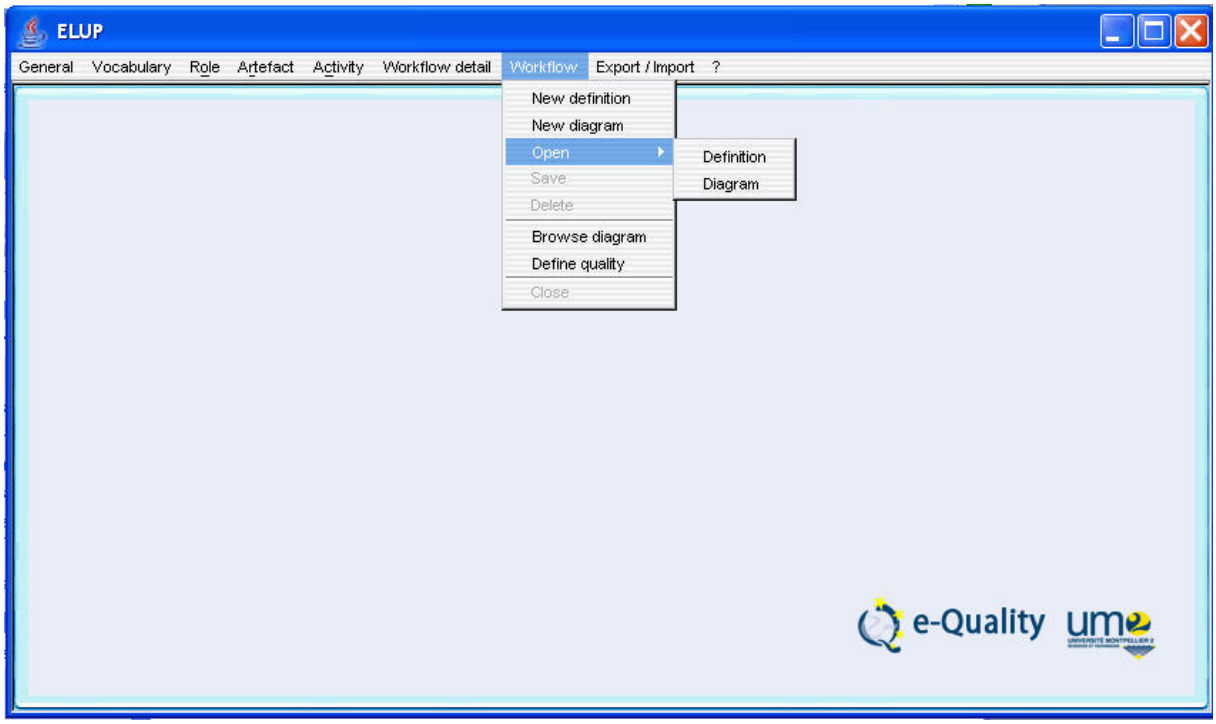


Fig. 4.11 *Workflow* menu

Item	Action
New definition	To create the definition of a new workflow.
New diagram	To create a new diagram linked to the definition of a workflow. eLUP opens the graphics editor of the diagram of the workflow with an empty panel.
Open definition	To edit the definition of a workflow. If the definition comes from an author other than the connected user, eLup suggests creating a new version of the definition.
Open Diagram	To edit a diagram linked to the definition of a workflow. eLUP opens the graphics editor dedicated to workflow. If the diagram comes from an author other than the connected user, eLup suggests creating a new diagram.
Save	To save the definition or the diagram of a workflow while editing.

Tab. 4.8 Action linked to an item in the *Workflow* menu (a)

Delete	To delete the definition or the diagram of a workflow while editing.
Browse diagrams	To browse (read without modifying). eLup opens a dialogue box to select the diagram to browse.
Define quality	To create the quality file for a workflow. Before doing this, the composition of the activity must already exist. The quality file has three panels: one panel recapping the definition and composition of the activity, one "Best practices" panel, and a "General Quality process chart"
Close	To close the definition or the diagram while editing.

Tab. 4.9 Action linked to an item in the *Workflow* menu

4.10. Export/ Import menu

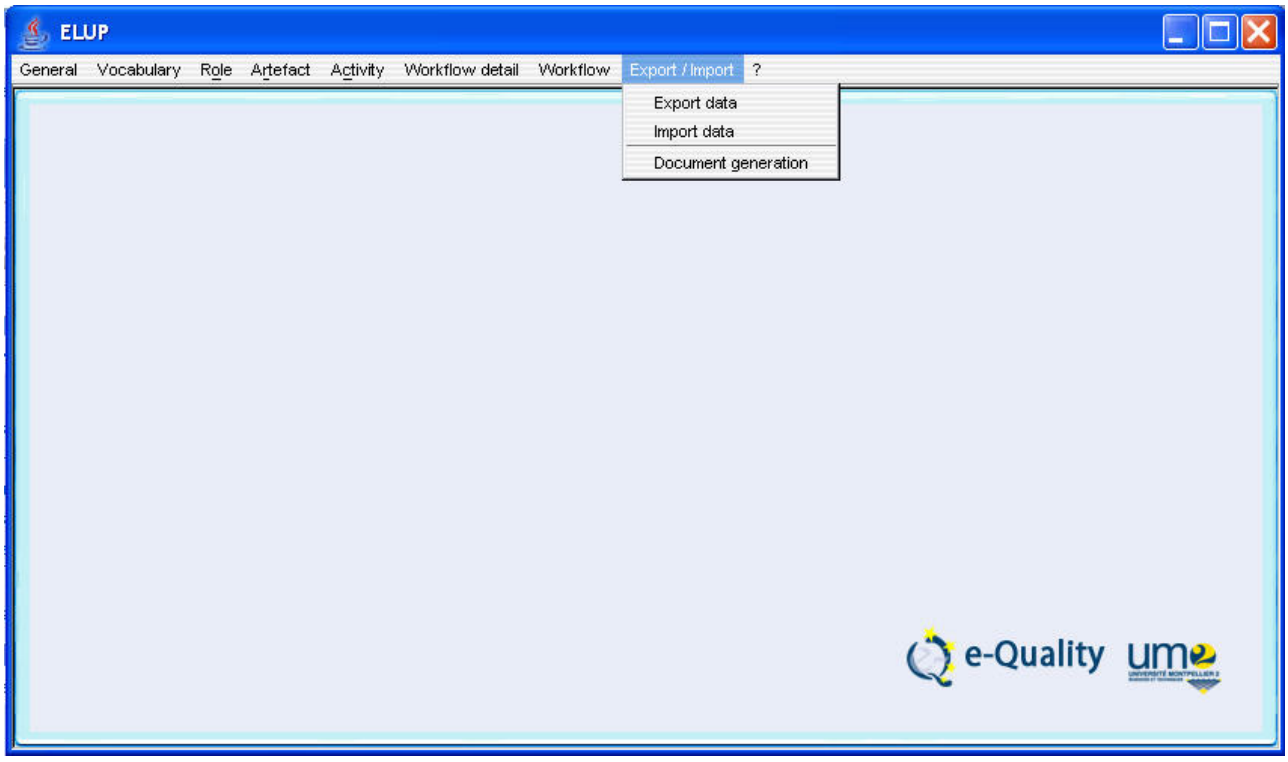


Fig. 4.12 Export/Import menu

Item	Action
Export data	To export data from eLUP to XML files. eLUP currently exports the vocabulary and compositions of activities
Import data	To import XML files into the eLUP editor. eLUP currently imports the vocabulary and compositions of activities
Generate documentation	To generate documentation in html and pdf format.

Tab. 4.10 Action linked to an item in the *Export/Import* menu

4.11. Help menu

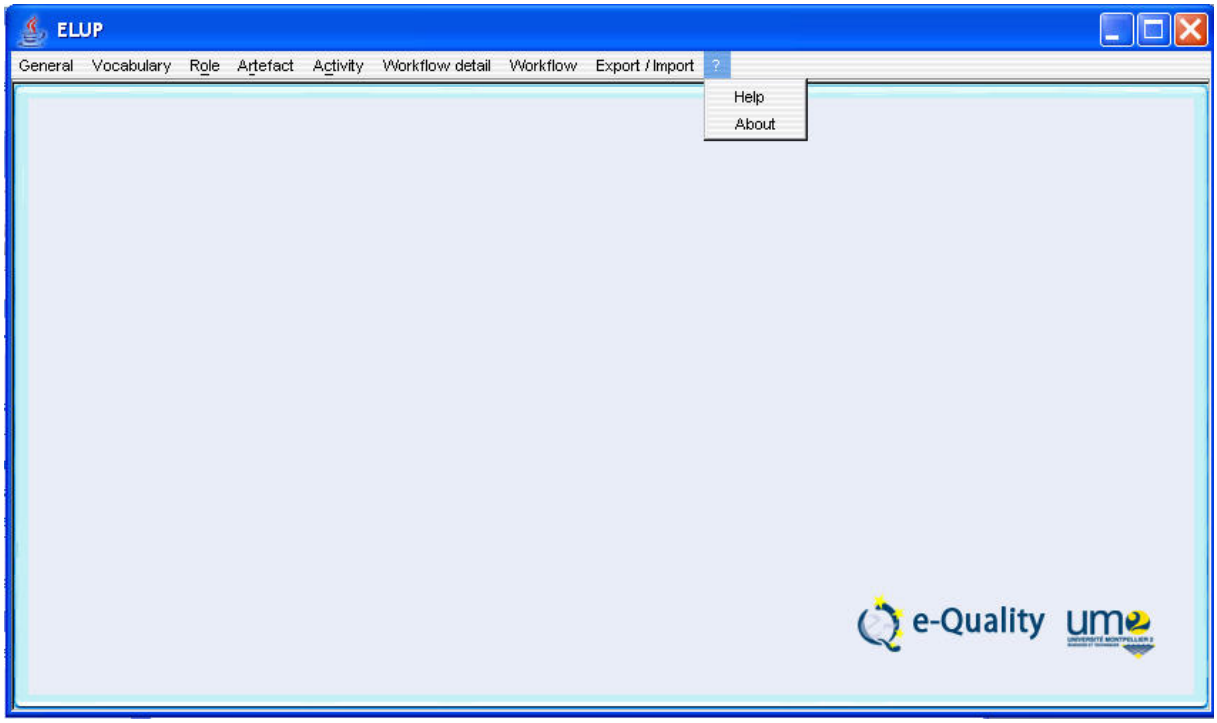


Fig. 4.13 Help menu

Item	Action
Help	Displays a summary of help for the software.
About	Displays information about the software.

Tab. 4.11 Action linked to an item in the *Help* menu

5. Editors

5.1. Definition editor in the *Vocabulary* menu

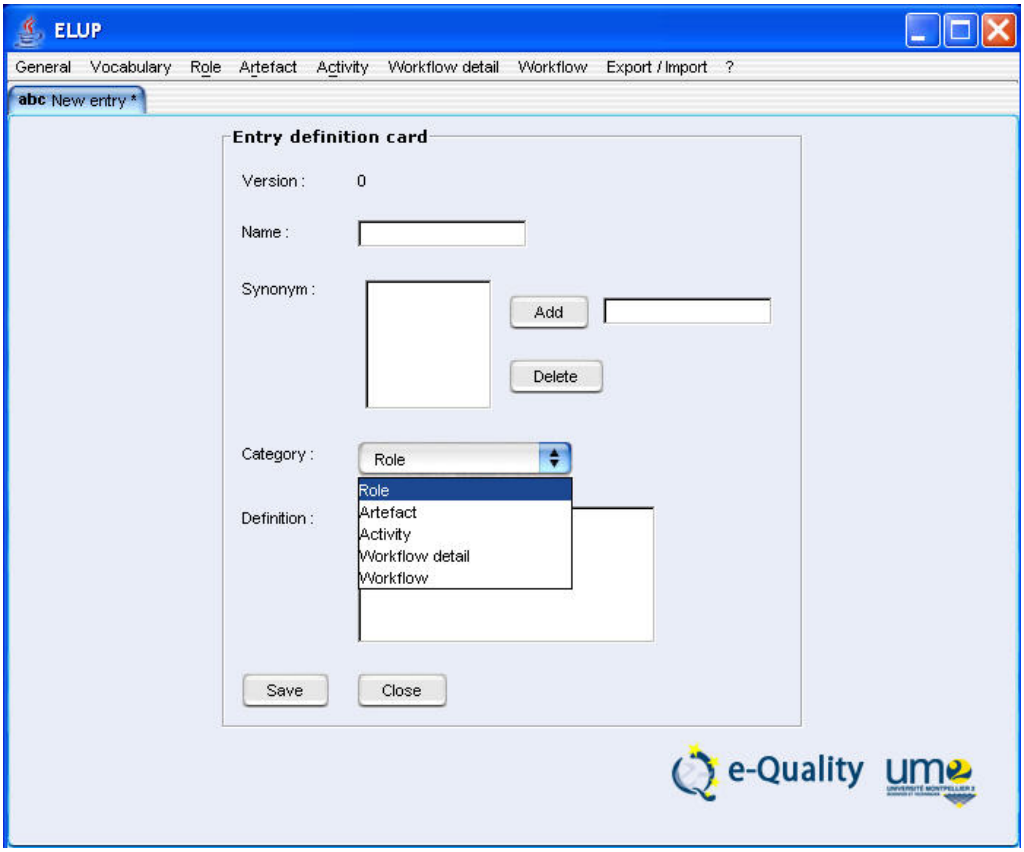


Fig. 5.1 Vocabulary definition editor

All the terms supplied with the software and produced by the e-quality project are in English. The interface is bilingual but the data entered are expressed in the user's language. eLup does not provide for a multilingual vocabulary.

Zone	Function
Version	Displays the version of the definition.
Name	Displays the term to be defined.
Synonyms	Specifies the synonymous terms.
Category	Selects the category.
Definition	Edits the definition of the term.
Save	Saves the definition while editing.
Close	Closes the definition while editing.

Tab. 5.1 Functions linked to the zones in the window

During data input, the editor displays the terms close to the vocabulary. If the user presses the Backspace key, all the terms in the vocabulary will be displayed.

The same applies for entering synonyms.

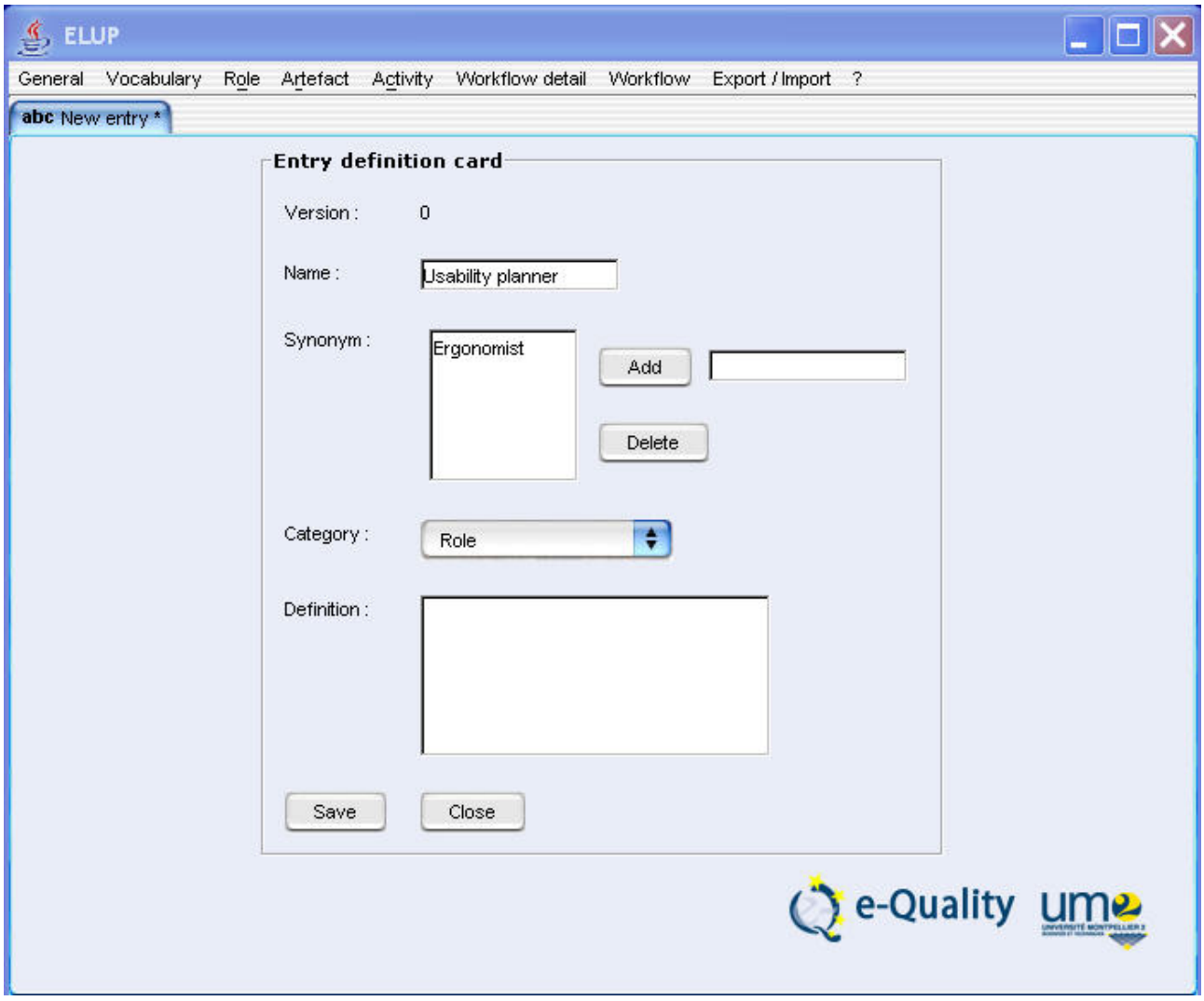


Fig. 5.2 Looking for similar words

When a word is entered, the editor displays the words it may be synonymous with.

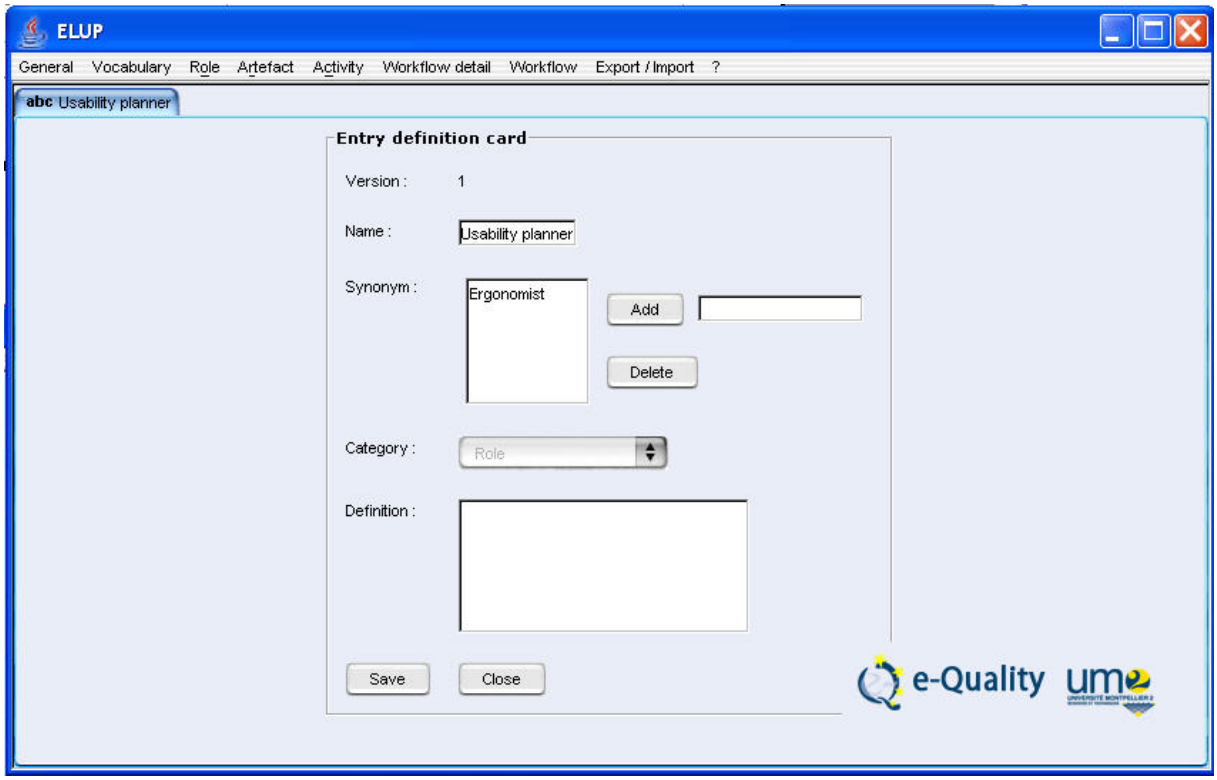


Fig. 5.3 Entering synonyms

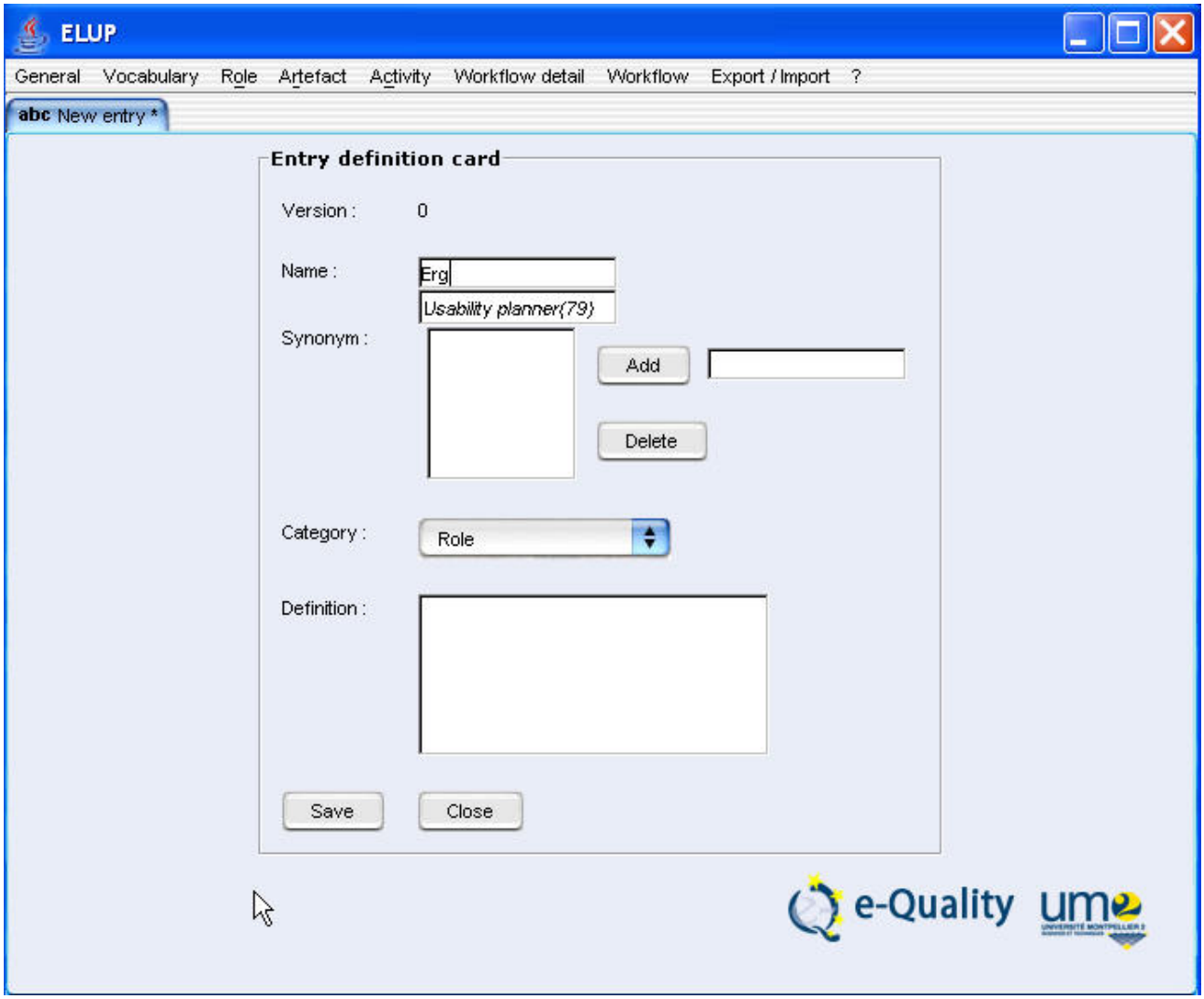


Fig. 5.4 Display of similar terms including synonyms (in italics)

When saving, the editor asks the user to specify the relationship: synonymy or homonymy (Fig. 5.5).

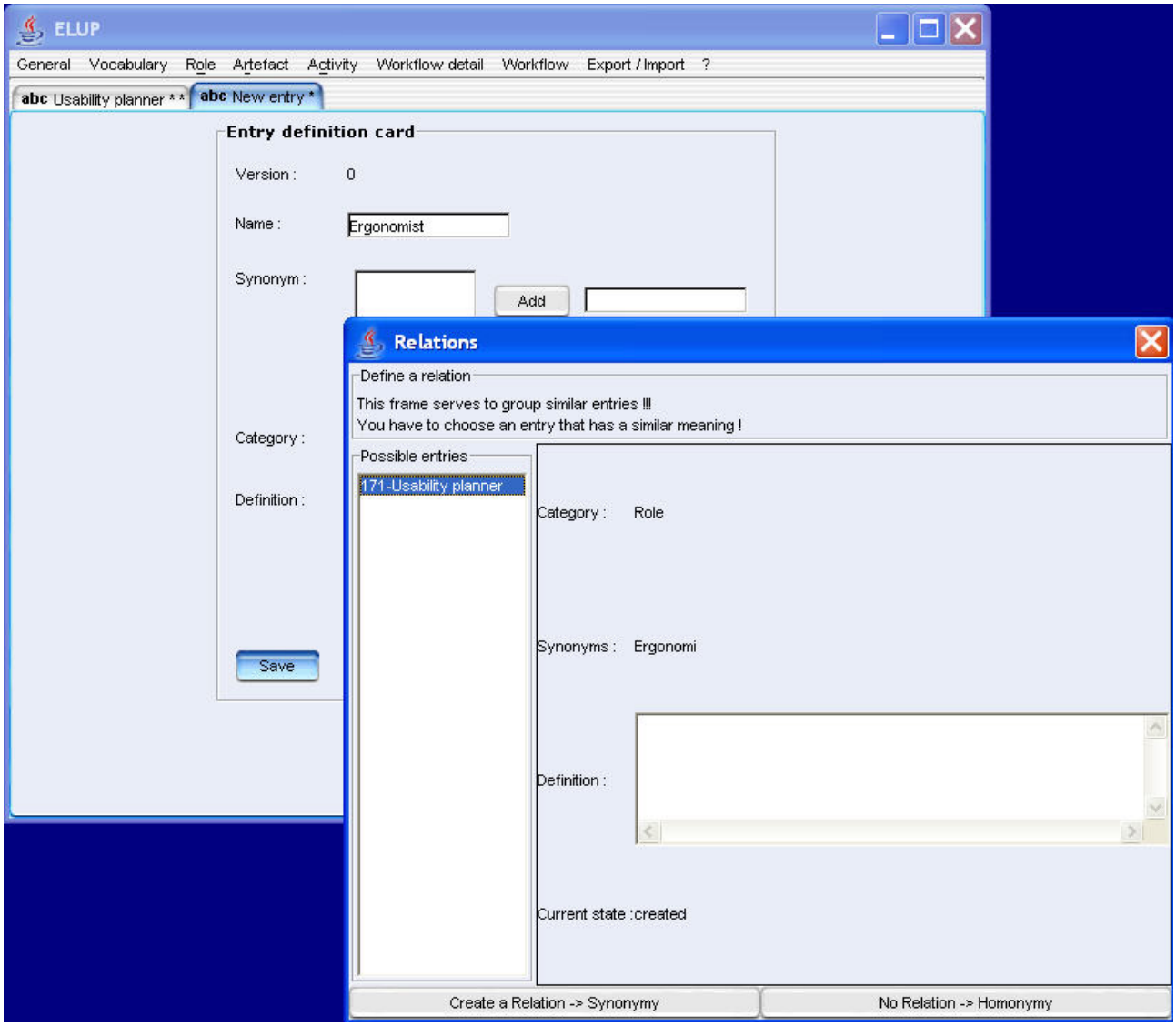


Fig. 5.5 Relation of synonymy or homonymy

All the other definition editors apply the same model (Fig. 5.6).

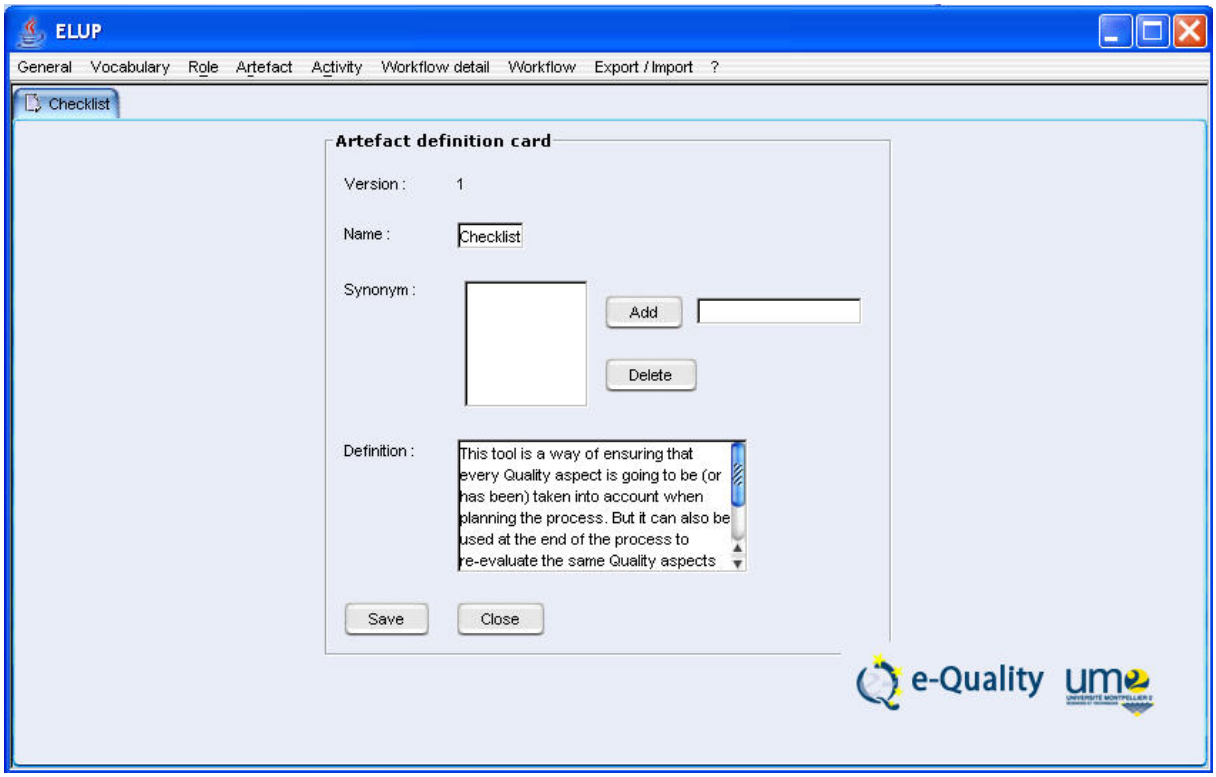


Fig. 5.6 Artefact definition editor

5.2. Activity composer editor

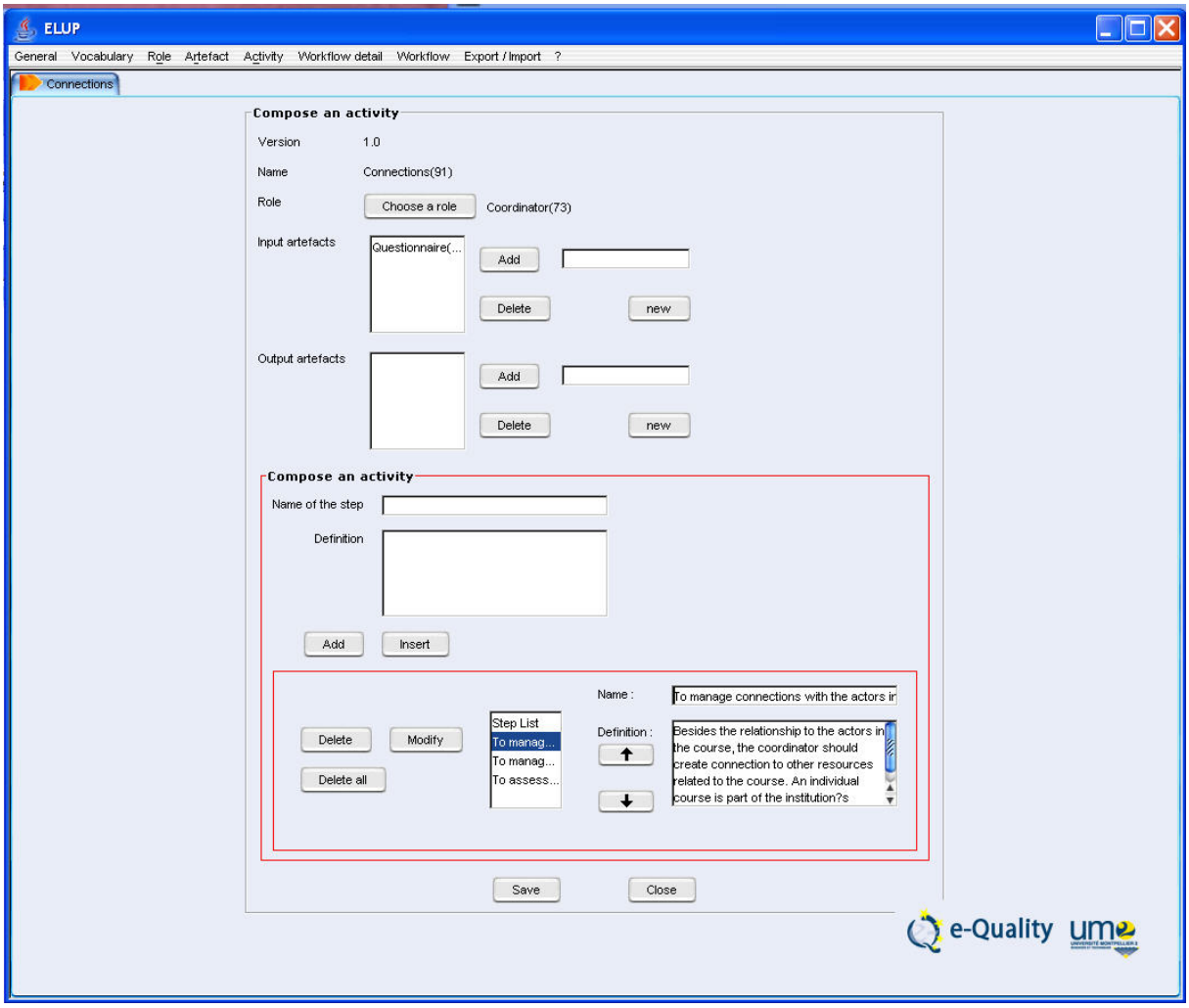


Fig. 5.7 Activity composer editor

The activity composer editor enables the user to select the role assigned to the activity (Who creates the activity), input artefacts for the activity (What the activity requires in order to be performed), output artefacts (What the activity produces) (Fig. 5.7).

The user can also specify the steps in the activity. These steps remain internal to the activity and cannot be referenced outside the activity.

An activity must be composed before it can be inserted into a workflow detail.

5.3. The workflow detail diagram editor

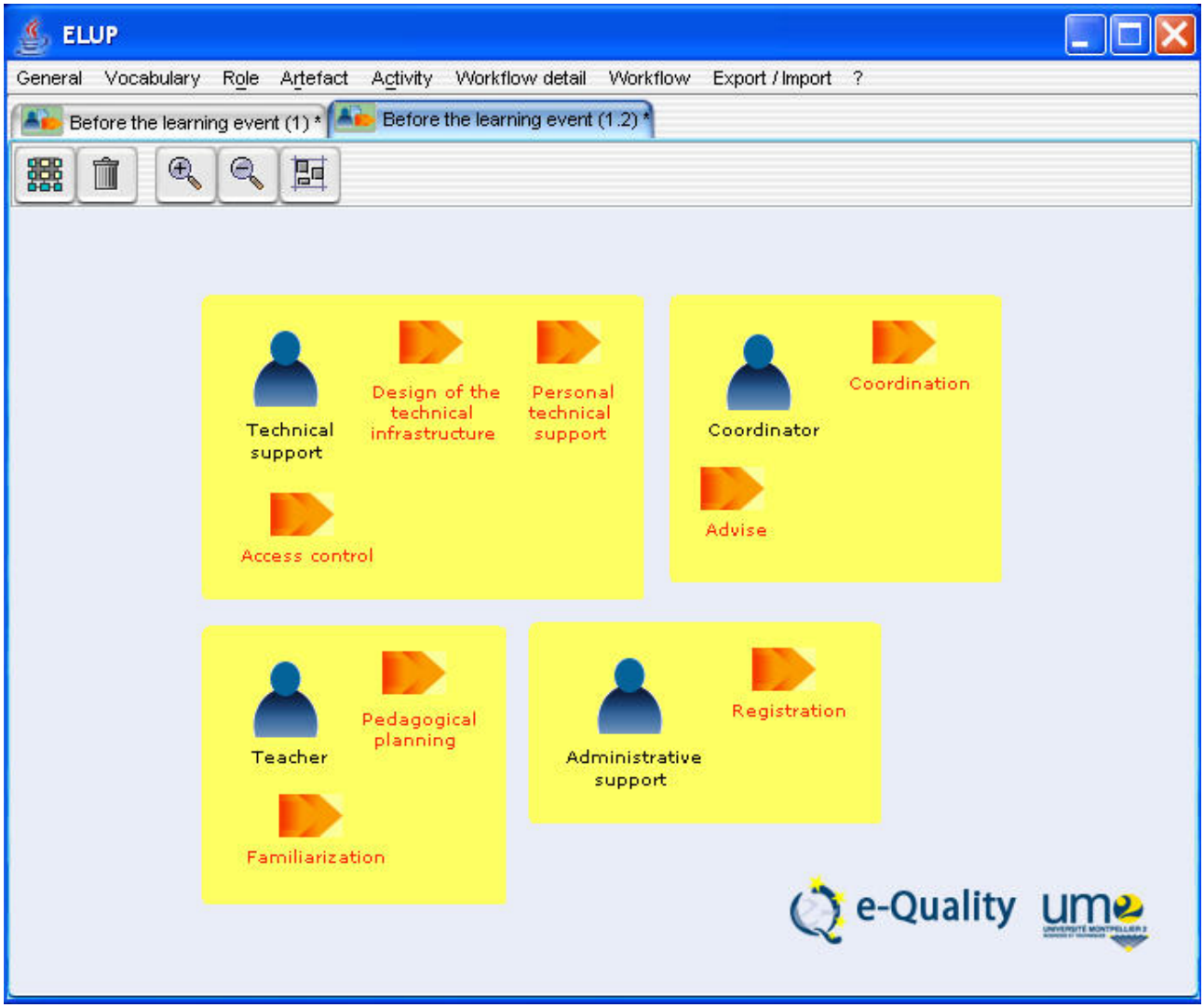


Fig. 5.8 Workflow detail editor

A workflow detail groups coherent activities together. It may represent a step in the e-learning process. For instance, in Fig. 5.8, the workflow detail represents a set of activities to be carried out under different roles, i.e. technical support, teacher, administrative support and coordinator before the learning event.

A workflow detail editor can only include composed activities.

Inserting an activity

To insert an activity, the user left clicks on the editor panel. A dialog box comes up to select the activity to insert.

Deleting an activity

To delete an activity from a workflow detail diagram, you simply select an activity by right clicking on it. The context menu allows you to delete the selected activity.

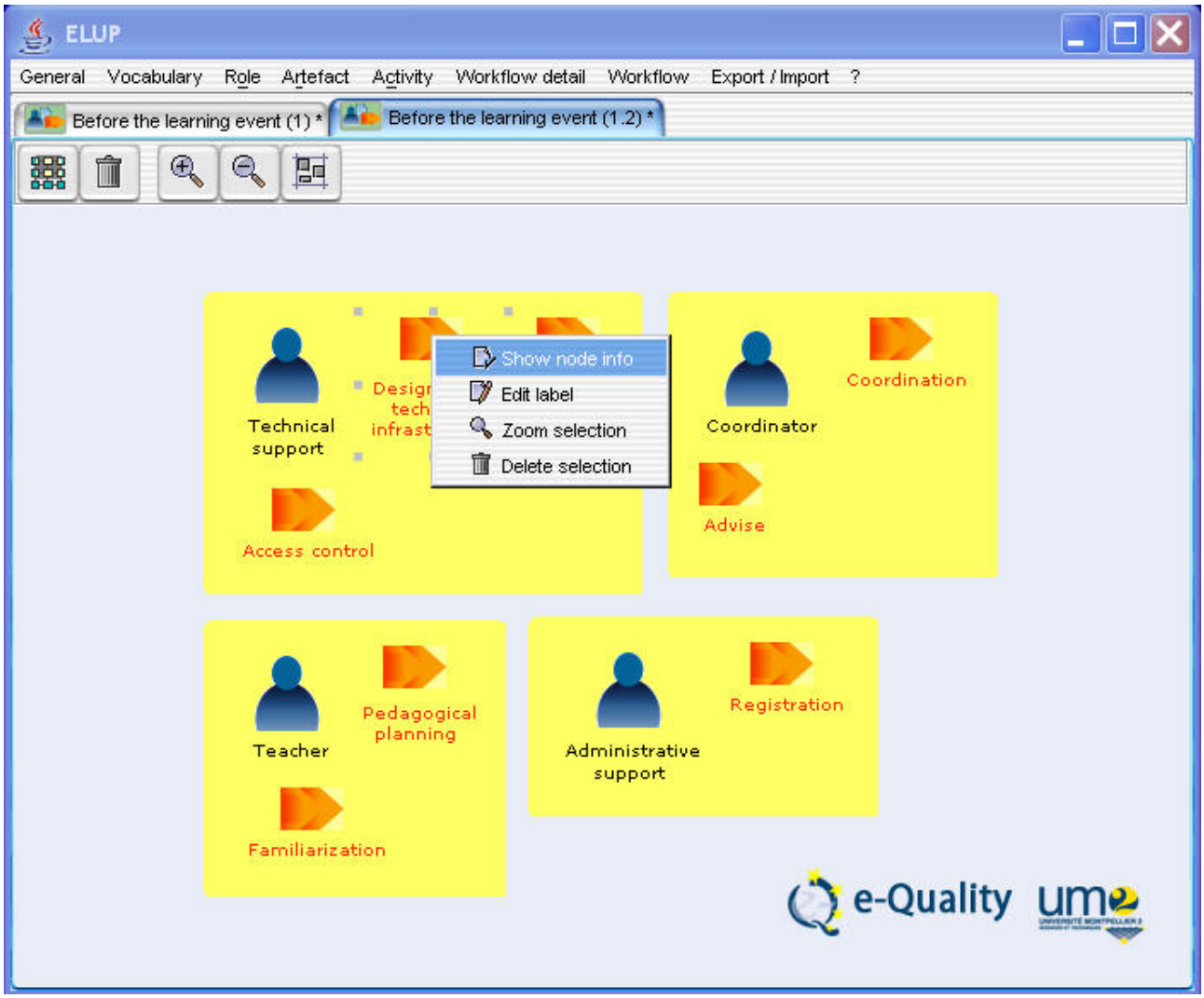


Fig. 5.9 Deleting an activity

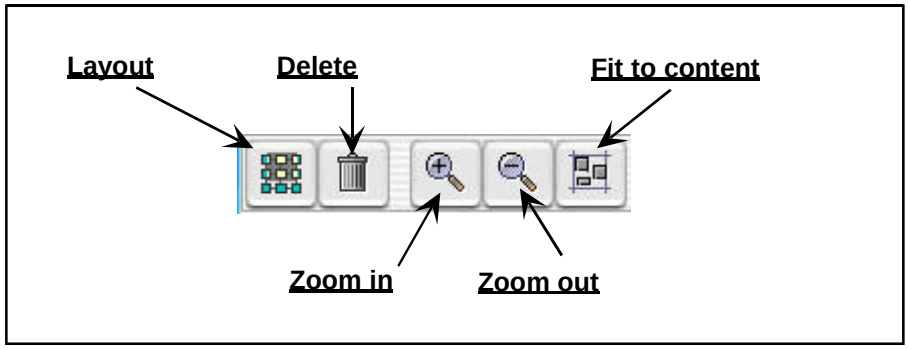


Fig. 5.10 Tools of the workflow detail diagram editor

5.4. Workflow diagram editor

A workflow diagram is used to organise the steps in the workflow details, which are themselves groups of activities. The sequence may be conditional (test nodes with exit conditions). Workflow details can be run in parallel (split and join nodes).

Inserting a workflow detail

To insert a workflow detail, the user selects the workflow detail icon in the left panel of the editor. S/he places the cursor on the background to position the workflow detail. A dialog box then allows the user to select the workflow detail to insert.

Deleting a workflow detail

To delete a workflow detail from a workflow diagram, simply select the workflow with a right click. The context menu allows you to delete the selected workflow detail.

Adding an arrow between two nodes

Left click on the first node and maintain pressure.

By releasing the left click an arrow will be drawn from where the cursor is placed. With each left click, the arrow is drawn from where the cursor is placed. The line stops when the cursor is placed on a node. The layout button correctly repositions the nodes and arrows in the diagram.

If the user wishes to cancel the arrow being drawn, s/he simply makes a double left click.

Comment: the user is in arrow insertion mode by default. To move a node, it must be selected with a left click and then moved.

Deleting an arrow

To delete an arrow, simply select it with a right click. The context menu allows you to delete the selected arrow.

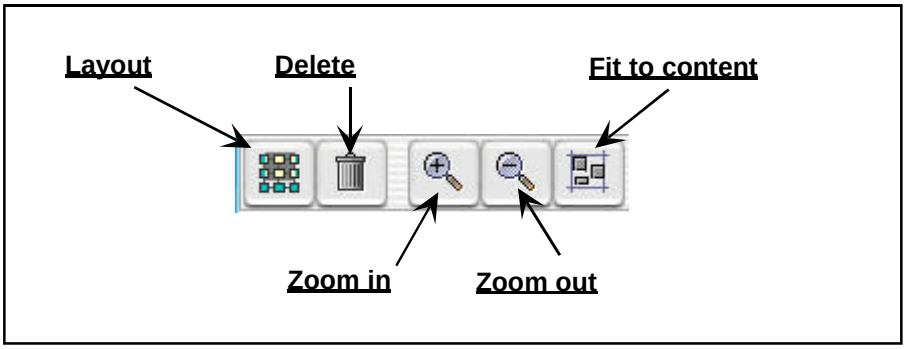


Fig. 5.11 Tools of the workflow diagram editor

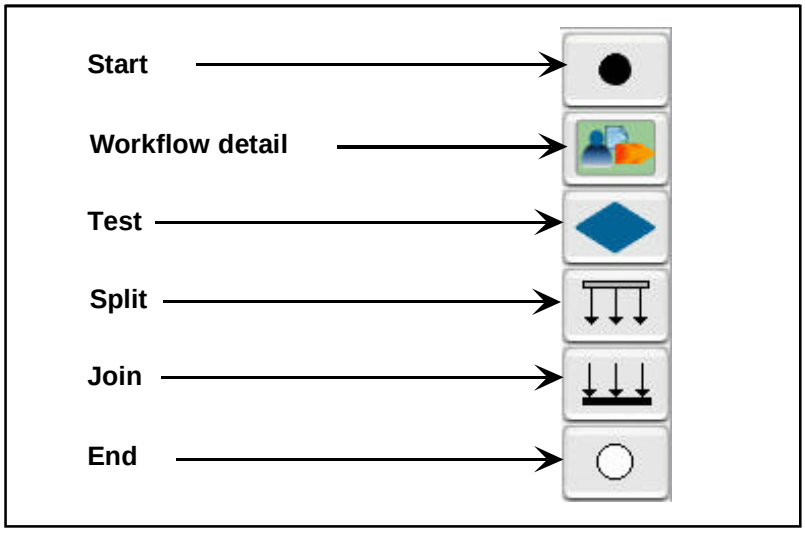


Fig. 5.12 Objects making up a workflow diagram

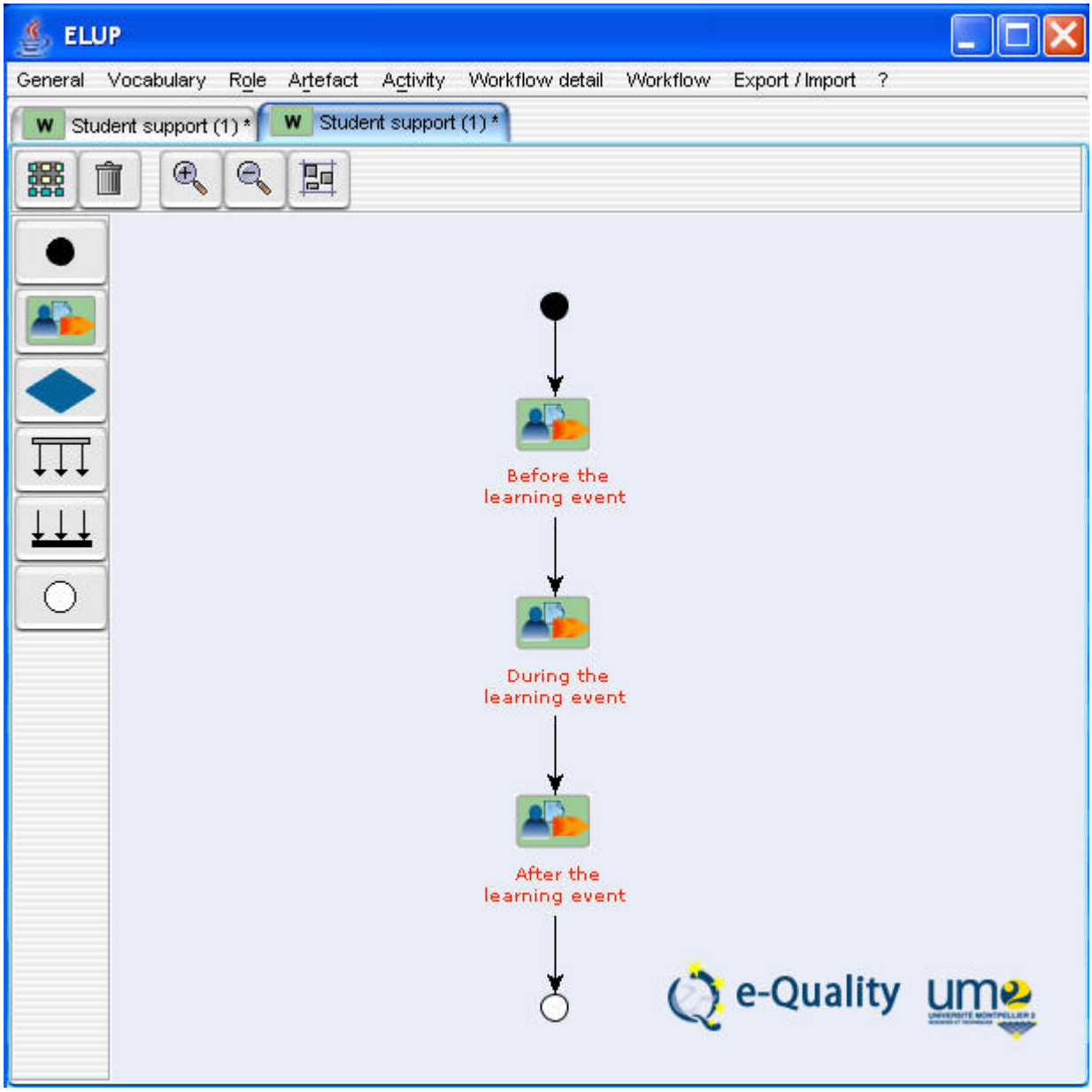


Fig. 5.13 Workflow diagram editor

5.5. Defining quality

Define quality in this context, means providing the elements giving concrete expression to the quality approach for an activity, a workflow detail or a workflow. These elements can be divided into two aspects studied in the e-Quality project: the best practices listed in the best practices data base (see [D3.3]) and the quality criteria set out in the General Quality Process Charter (see [GQPC]).

Entries to the General Quality Process Charter were imported to the eLup editor. Each entry describes a quality criterion that applies to one or a number of activities. For each Quality criterion, the charter shows a text presenting the criterion, comments, commitments to take this criterion into account, quality indicators related to the phase in the e-learning process to which they are relevant: before, during or after the learning event or no specific stage.

The quality criteria brought together in the GQPC were imported to the eLUP. Their origin is identified (e-Quality). The user can only modify a copy of this data, and not the data itself, in order to preserve its integrity. The user can also enter his/her own data in the quality criterion entry interface. The different sections presented in GQPC will be reproduced in the eLup interface.

In the GQPC, quality criteria were defined and then linked to one or a number of activities. In eLup, the activities are the entry point for these quality criteria. The user indicates an activity; eLup retrieves the related quality criteria. So when a user learns of a composed activity, (a role, input and output artefacts and the steps), s/he can display the effective implementation of a quality approach in the quality window, linked to a quality criterion. S/he will dispose of the commitments to respect and the indicators used to measure the implementation of his/her quality approach.

Fig. 5.14 shows, for the “Pedagogical Planning” activity:

- a quality criterion “A precise schedule of the work including rules has been established”,
- the associated commitments “Documents provided at the beginning of the course specify the work schedule, as well as each actor's commitments”,
- several indicators including “documents distributed at the beginning of the learning event”.

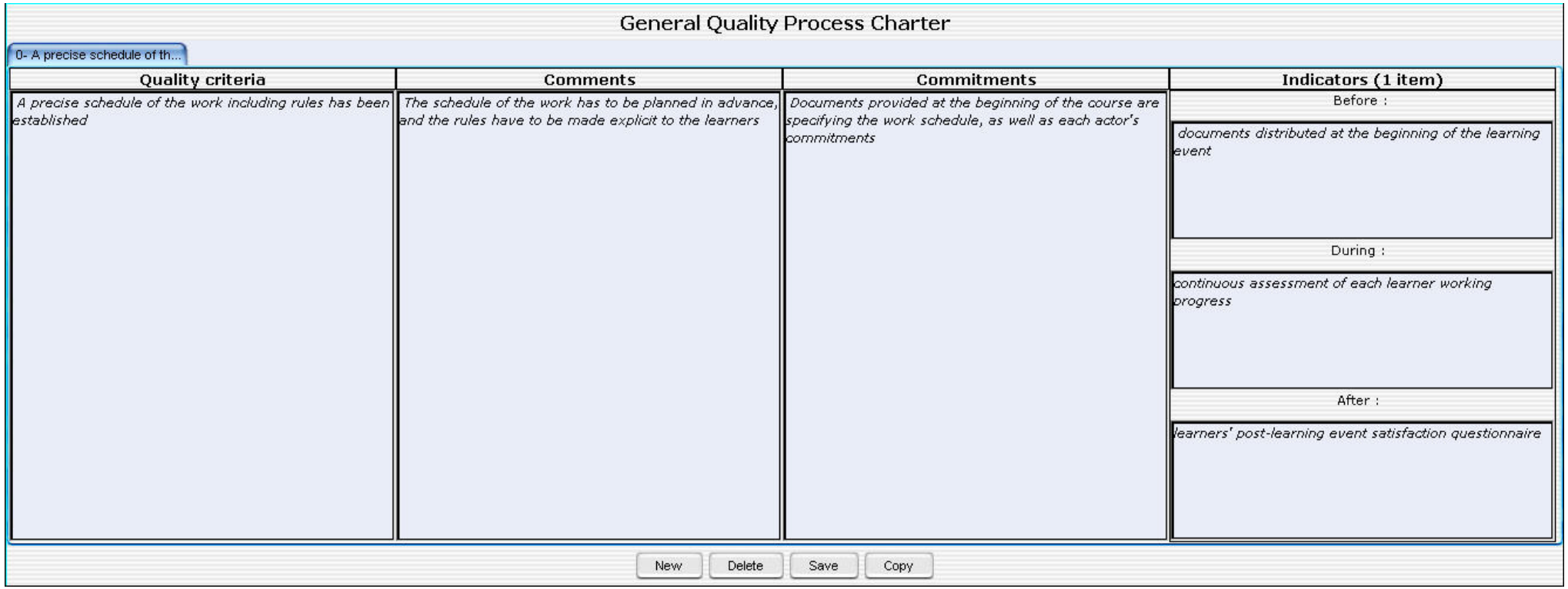


Fig. 5.14 Characteristics of a Quality criterion

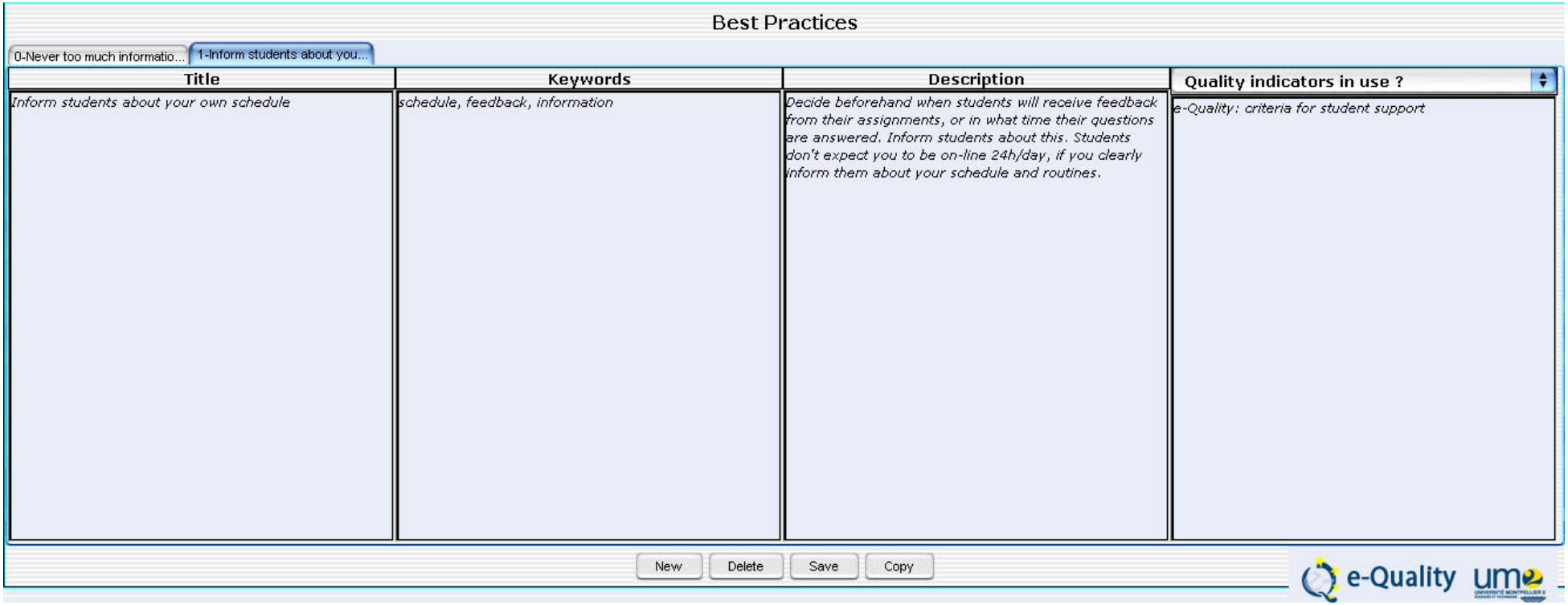


Fig. 5.15 Characteristics of a best practice

The best practices listed in the database at the end of September 2006 (http://e-quality.uta.fi/bpdb/) were imported to the eLUP software. Their origin is identified (institution of origin). The user can only modify a copy of this data, and not the data itself, in order to preserve its integrity. The user can also enter his/her own data in the quality criterion entry interface. The different sections filled in by the user will be reproduced in the eLup interface.

In the database of best practices, the best practices were defined and then linked to a role and to one or a number of activities. In eLup, the activities, workflow details and workflows are the entry points of best practices. The user indicates an activity; eLup retrieves the related best practices. So, when a user learns of a composed activity (a role, input and output artefacts and the steps), s/he can display a best practice in the quality window in order to benefit from the experience of other elearning actors and enter his/her own feedback experience of best practices. A best practice will be characterised by a title, a description and the optional fields (Fig. 5.15).
Fig. 5.15 shows the example of a best practice linked to a “Pedagogical Planning” activity. In this case, it is “Inform students about your own schedule” described as “Decide beforehand when students will receive feedback from their assignments, or in what time their questions are answered. Inform students about this. Students don't expect you to be on-line 24h/day, if you clearly inform them about your schedule and routines.” The best practice uses the keywords “schedule, feedback, information”. Its optional fields carry the value:

Name of field	**Value**
Quality indicator	Student feedback
Method of validation	Feedback from students
History and context	Student feedback from many courses has stressed the importance of this practice.
Still in use	Yes
Participants	Actors and students in course
Costs	Time resources: Informing all students at the same time takes little time compared to answering students’ questions separately.
Mode of calculation	By comparing processes from different courses
Resources	

Tab. 5.2 Criterion: ″Student feedback″

Fig. 5.16 shows, for the “Pedagogical Planning” activity: the set of descriptors. The eight optional fields are viewed one at a time. Each of these is accessible via a scroll menu. This interface was chosen to enable the user to work on a best practice without moving to another window – which would have been time-consuming. The drawback is that the user cannot view all the fields simultaneously.

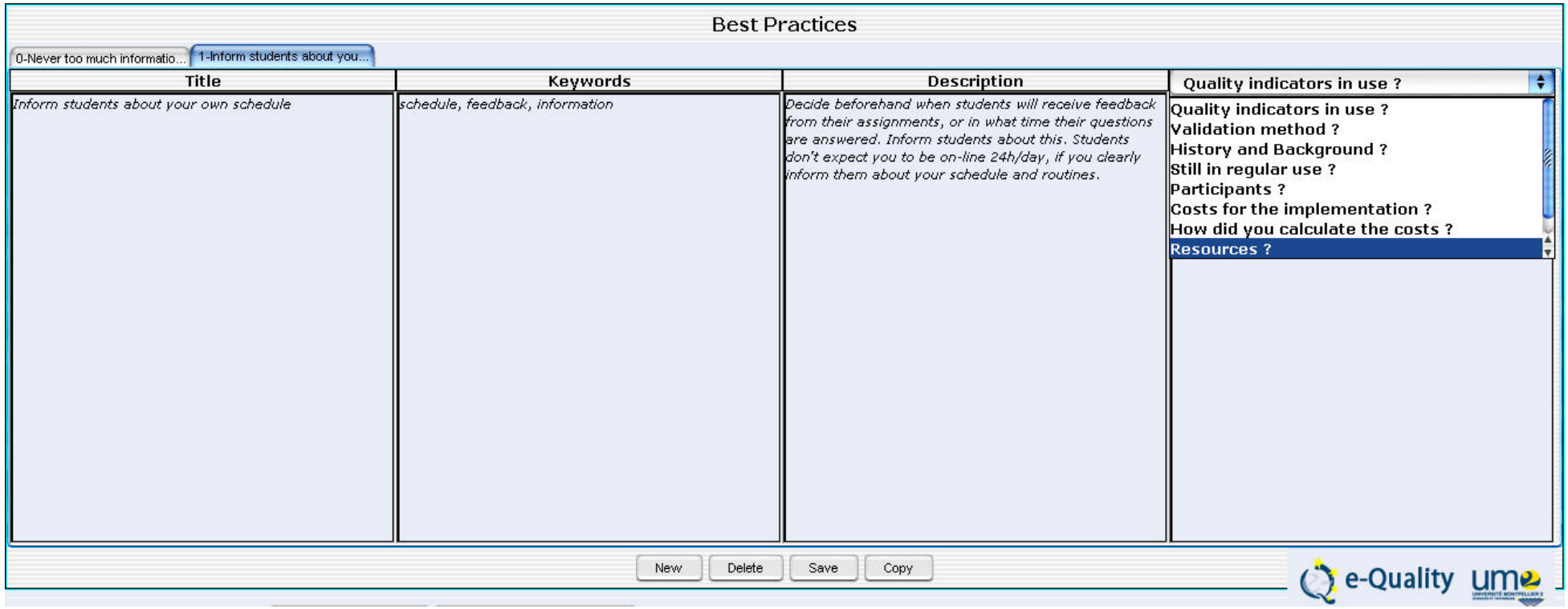


Fig. 5.16 Optional characteristics of a best practice

5.6. Defining the quality of an activity

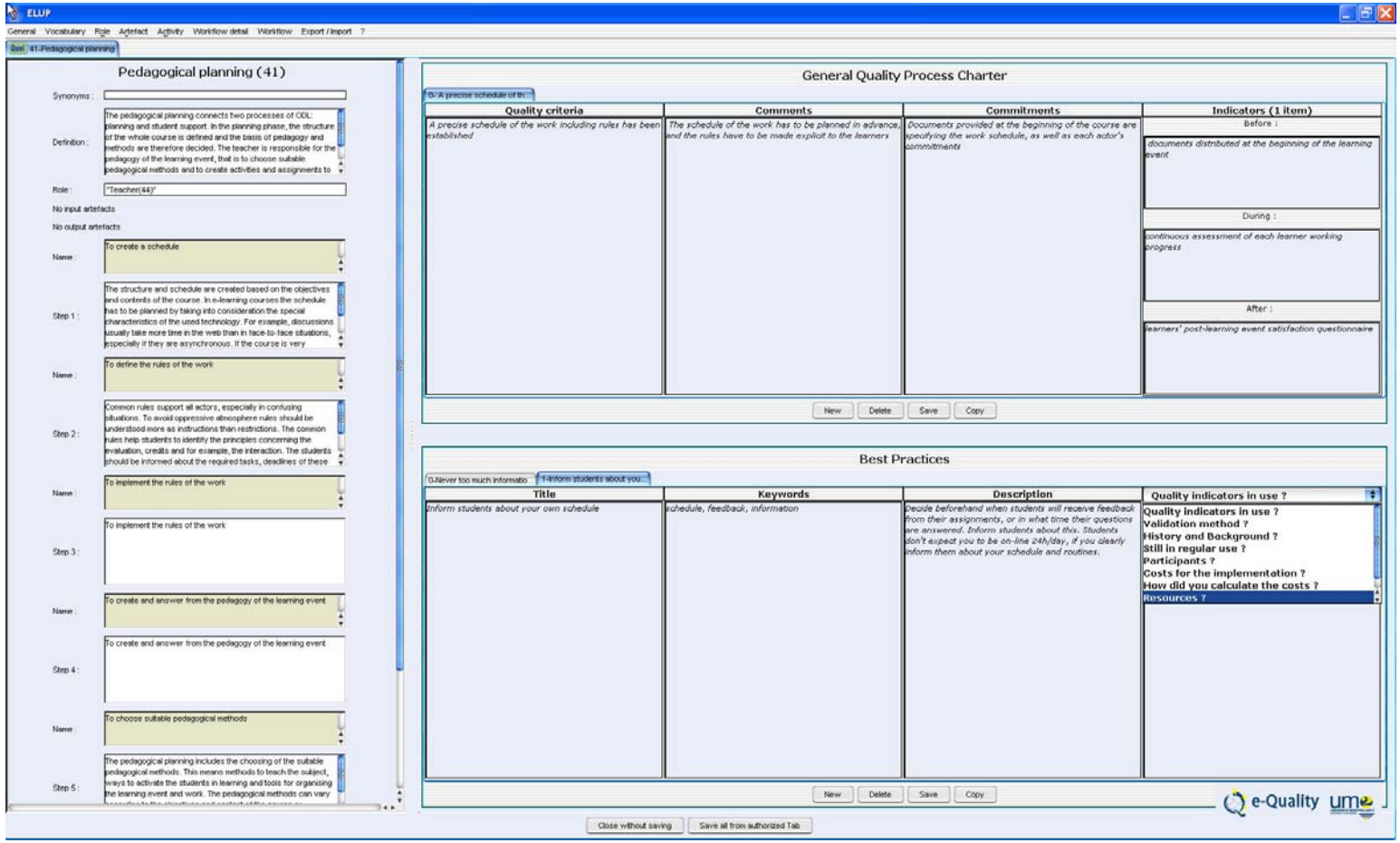


Fig. 5.17 ***Define the Quality of an activity*** **window**

The Define the quality of an activity window is divided into three panels: a panel recapping the definition and composition of the activity, a “Best Practices” panel and a “General Quality process charter” (see Fig. 5.17).

5.7. Defining the quality of a workflow detail

The window Define the quality of a workflow detail is divided into three panels: a panel displaying the diagram of the workflow detail, a “Best practices” panel and a “General Quality process charter” panel.

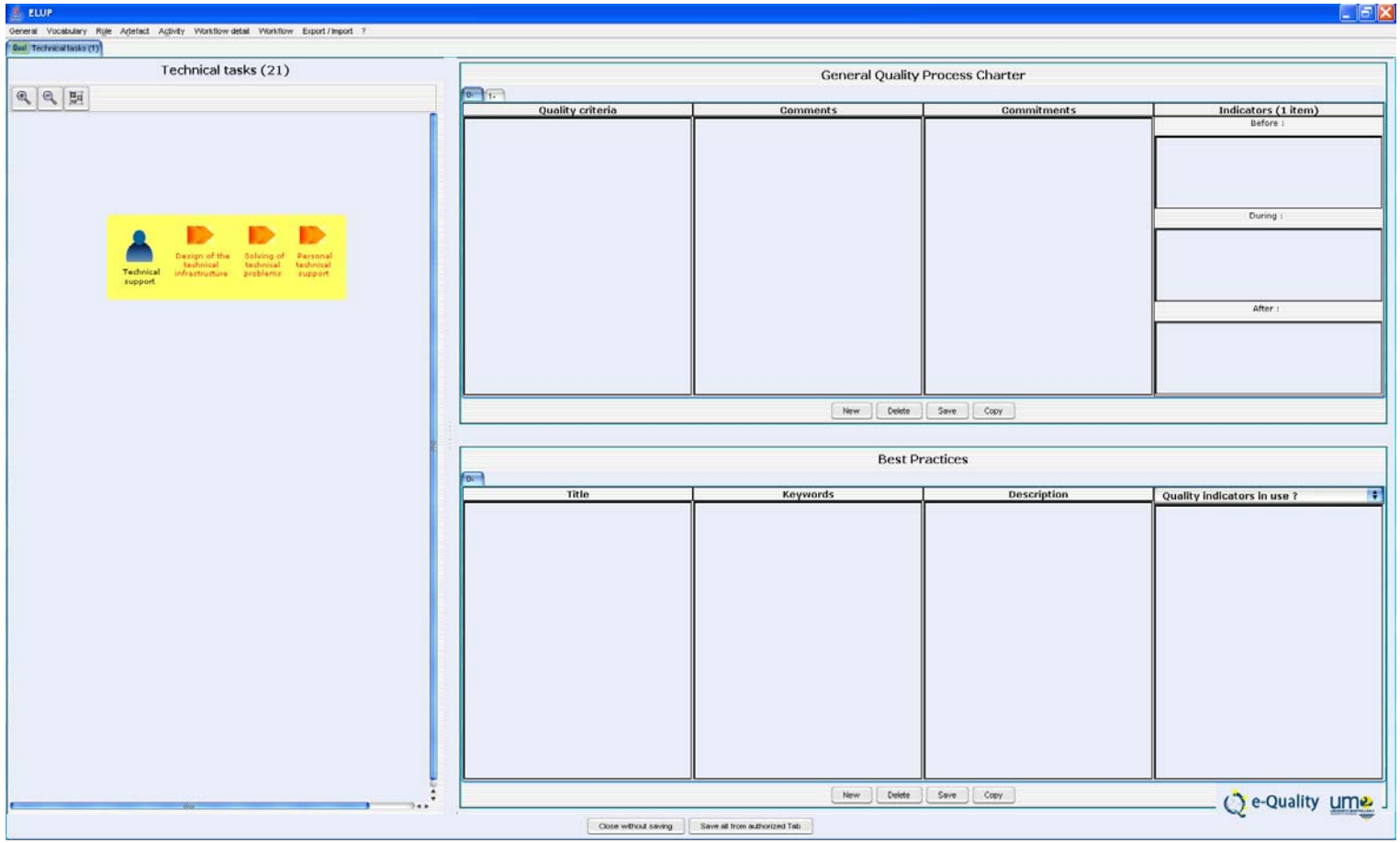


Fig. 5.18 Workflow detail quality window

5.8. Defining workflow quality

The window Define workflow quality is divided into three panels: a panel displaying the workflow diagram, a “Best practices” panel and a “General Quality process charter” panel

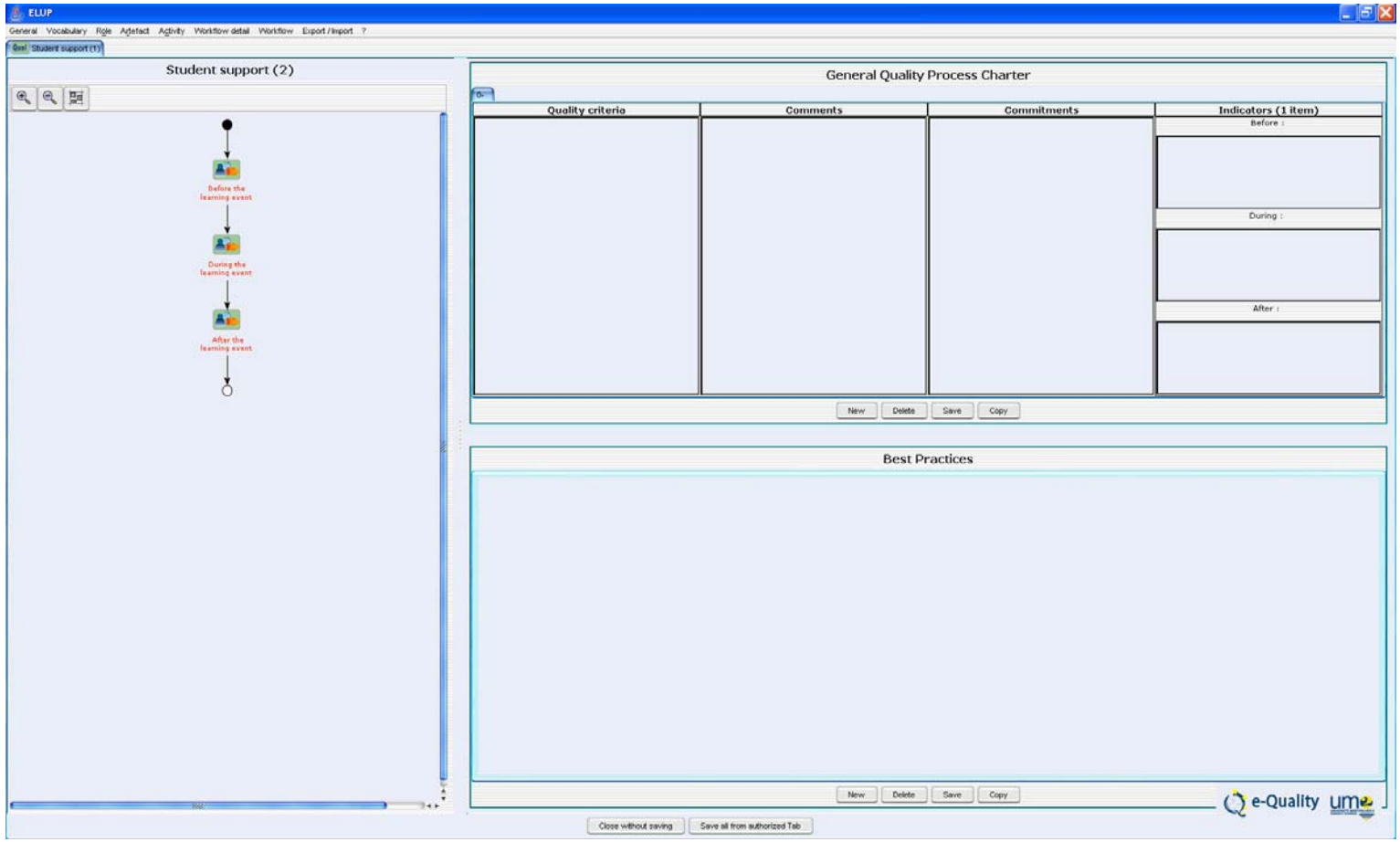


Fig. 5.19 Workflow Quality window

6. Generating documentation

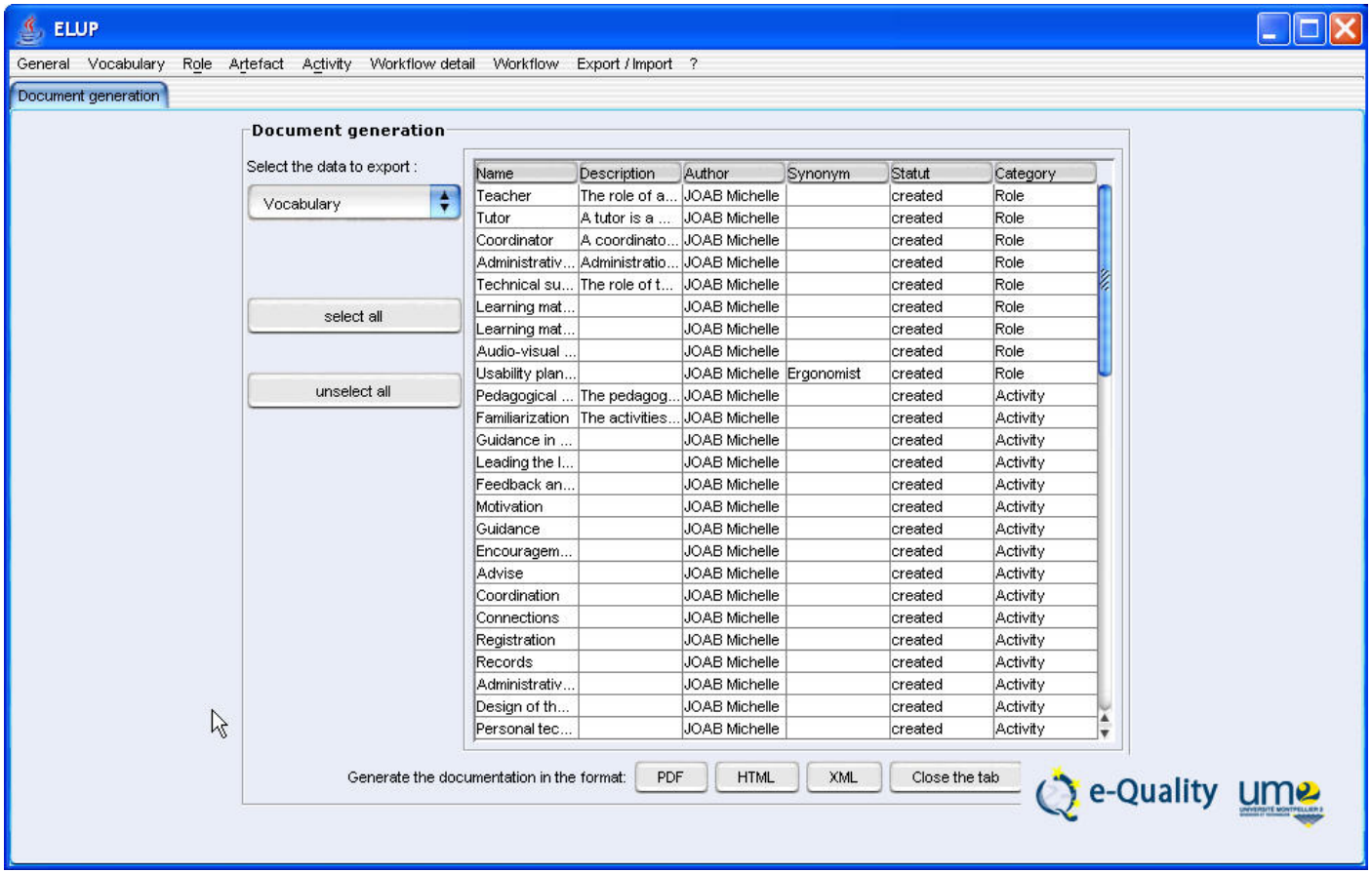


Fig. 6.1 Generate documentation

eLup allows elementary documentation to be generated in PDF, HTML or XML format. The user selects the objects to include in the documentation by category. For workflow details and workflows, s/he chooses to insert the matching diagrams.
S/he can also generate descriptions of the objects in the eLup base in xml format.

7. Table of figures

Fig. 3.1 Steps to specify an activity 45
Fig. 3.2 Steps to specify a *workflow detail* 46
Fig. 3.3 Steps to specify a *workflow* 47
Fig. 4.1 Login window 48
Fig. 4.2 Creating a new account 49
Fig. 4.3 Home window 49
Fig. 4.4 *General* menu 50
Fig. 4.5 Tabs 51
Fig. 4.6 Vocabulary menu 52
Fig. 4.7 *Role* menu 53
Fig. 4.8 *Artefact* menu 54
Fig. 4.9 *Activity* menu 55
Fig. 4.10 *Workflow detail* menu 56
Fig. 4.11 *Workflow* menu 58
Fig. 4.12 Export/Import menu 59
Fig. 4.13 Help menu 60
Fig. 5.1 Vocabulary definition editor 61
Fig. 5.2 Looking for similar words 62
Fig. 5.3 Entering synonyms 63
Fig. 5.4 Display of similar terms including synonyms (in italics) 64
Fig. 5.5 Relation of synonymy or homonymy 65
Fig. 5.6 Artefact definition editor 66
Fig. 5.7 Activity composer editor 67
Fig. 5.8 Workflow detail editor 68
Fig. 5.9 Deleting an activity 69
Fig. 5.10 Tools of the workflow detail diagram editor 70
Fig. 5.11 Tools of the workflow diagram editor 71

Fig. 5.12 Objects making up a workflow diagram 71
Fig. 5.13 Workflow diagram editor 72
Fig. 5.14 Characteristics of a Quality criterion 74
Fig. 5.15 Characteristics of a best practice 75
Fig. 5.16 Optional characteristics of a best practice 78
Fig. 5.17 *Define the Quality of an activity* window 79
Fig. 5.18 Workflow detail quality window 81
Fig. 5.19 Workflow Quality window 83
Fig. 6.1 Generate documentation 84