Framework for Evaluation and Development of Learning Objects in Codewitz

Wladimir Bodrow, Stefan Wolf, Irina Bodrow

Abstract: *The main focus of this paper is on the development of Learning Objects (LO) within the Codewitz Project (http://www.codewitz.net). This project is devoted to the implementation of the e-Learning tools to support the improvement of programming skills during academic and in the additional education. Here we evaluate the different methods, metaphors, and platforms used for developed LOs. The most important perspective for this research is the stringent didactical concept of knowledge transfer. According to it the understanding of the resources available in every modern programming language like Java, C++ or C# and the memorization of the facilities implemented in the different development tools like eclipse or Borland is of less importance. On the other side the growing/increasing the (cap)ability to solve business or other problems has the top priority. Based on this we formulate the decisive features toward the development of the common concept for Los.*

Key words: *Distance learning, programming education, knowledge transfer, knowledge visualization.*

INTRODUCTION

During the past decades the trend of learning supporting media went from a strict instructional design to Computer Based Training (CBT) and to Web Based Training (WBT). Today's claim of e-Learning is to achieve a maximum efficiency of knowledge transfer with a minimum effort. Using the perspective and/or concepts of knowledge management one can speak about the knowledge generation regarding the students.

In the last years almost 200 LOs were produced within the scope of the Codewitz Project by several international teams. A detailed overview of the categories and numbers of the produced LOs is available on the Internet site of the project [5]. Characteristic for Codewitz Project is the fact that there exist only a few basic rules for the design of the LOs which caused a huge variety of solutions that have been developed. Besides implementing different user interfaces and having divergent levels of user interaction all solutions have their specific understanding of knowledge transfer and a corresponding learning philosophy. All the experience gathered by producing and using these LOs need to be sorted and evaluated. Some of our evaluation results regarding in the project implemented LOs we presented before on several conferences [1, 2, 3, 4].

Accepted and mainly applied in distance education today the concept of blended learning allows the usage of various content available on the WEB. Just because the growing part of it is the open source and can be used for free [6] we observe the occurring wide integration of this open content into e-learning applications. But even if one can find in Internet encyclopedia like Wikipedia [7] well described and presented documents about all possible topics they are not a priori prepared/designed for learning as we defined it before – for empowerment to solve tasks or problems in particular situation or under special circumstances. Because of primarily orientation of Internet encyclopedia on scientific description of objects, events, processes, etc. they will never compete with FAQ which support the solution of real problems or tasks in the business and in the everyday life. From our point of view the acceptable distance education in general and e-Learning in particular has to combine both scientific and application oriented approaches. Especially important is it in such service oriented areas like programming skills.

2. LEARNING OBJECTS IN CODEWITZ

In this chapter we will briefly introduce the typical learning objects developed within Codewitz. This selection represents only our view on more than 200 learning

objects and just because of the continuously growing contributions to the project it is not comprehensive for all LOs in the project.

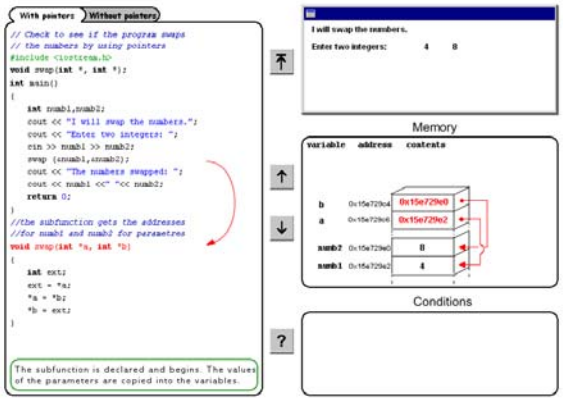

Fig.1. Example 1.

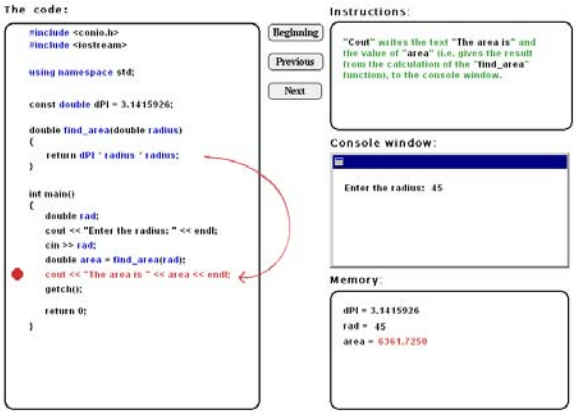

Fig.2. Example 2.

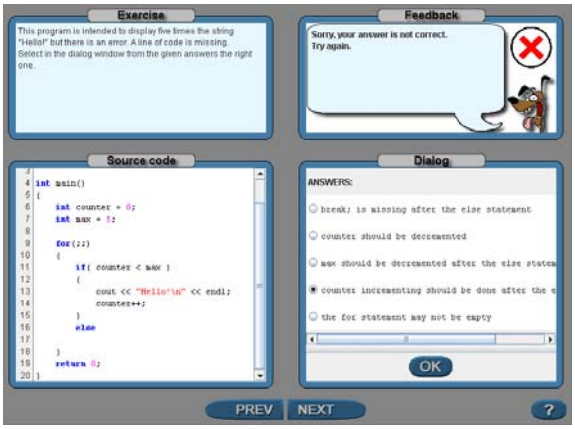

Fig.3. Example 3.

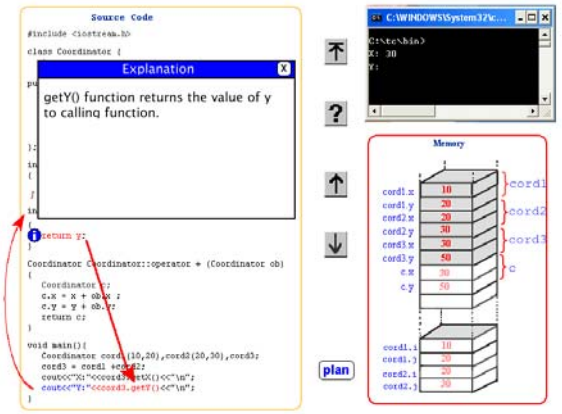

Fig.4. Example 4.

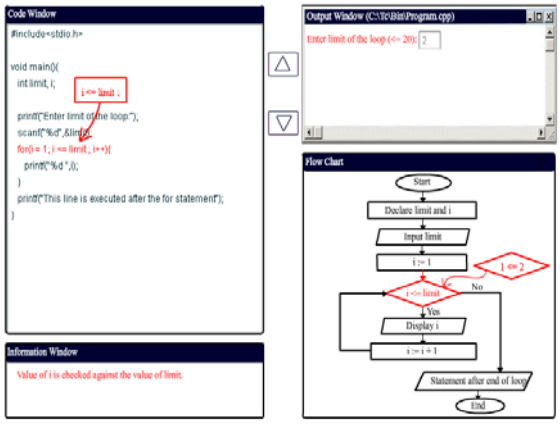

Fig.5. Example 5.

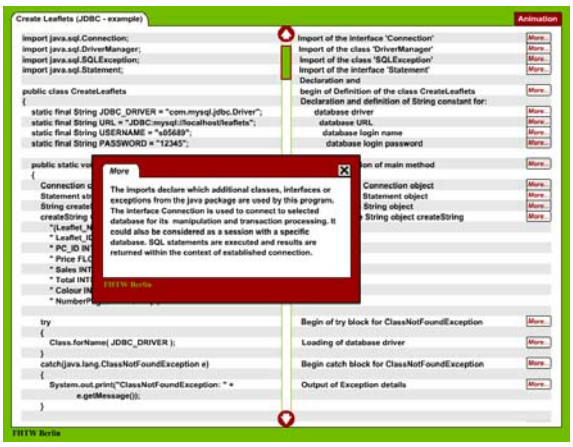

Fig.6. Example 6.

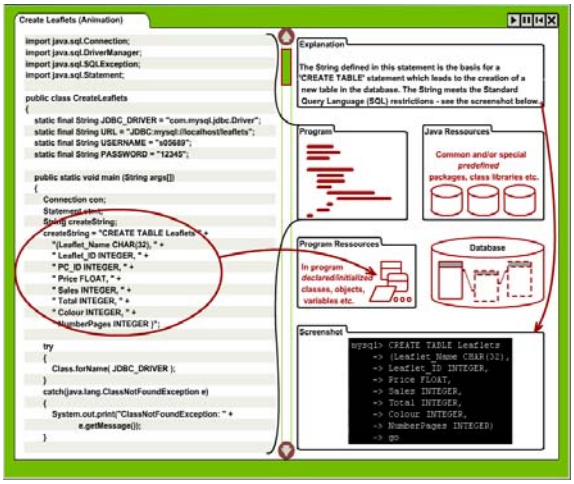

Fig.7. Example 6A.

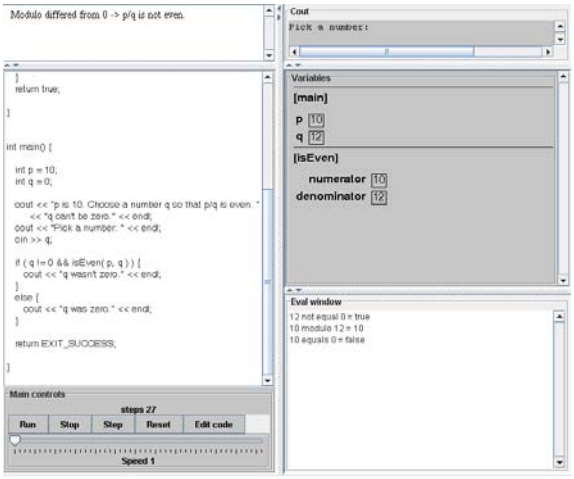

Fig.8. Example 7.

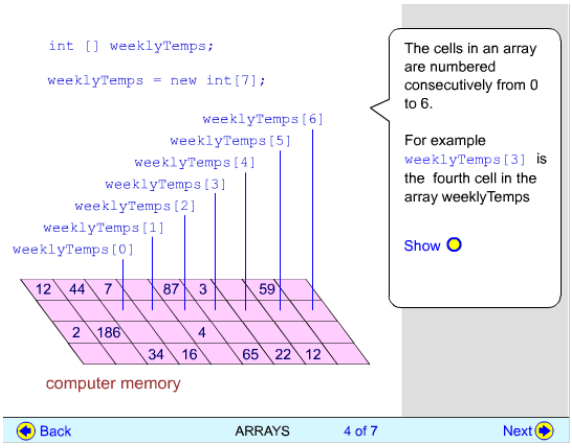

Fig.9. Example 8.

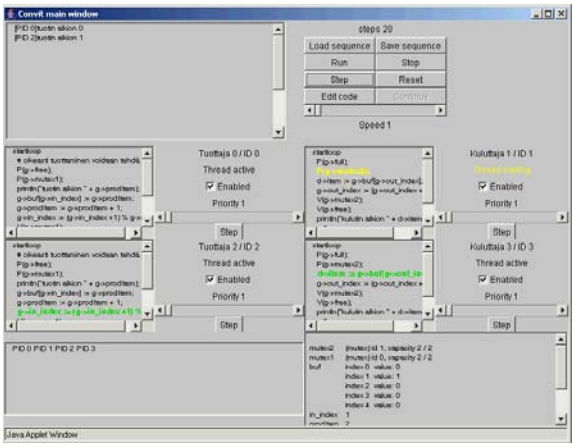

Fig.10. Example 9.

3. COMMON FEATURES

The analysis of pros and cons to presented LOs is reported in (Bodrow and Bodrow, 2006). Here we will figure out the common features and attributes to be able to find the similarities and discrepancies among these examples. These will be considered as the foundation to develop the new and to evaluate the existing Learning objects.

Firstly, the implemented LOs have very different look and feel:

- They differ in subdivision of the screen – in number, size, and placement of various windows, and in amount and quality of the content represented in them;
- Just developed to support the improving of the programming skills, some of them contain no windows with the source code of presented example;
- Some examples use different colours and graphic attributes like curves and arrows, where another are implemented with strong reduced colour palette;
- Very different is the usage of screenshots and animations.

Secondly, all (groups of) LOs follow different concept and corresponding structure for explanation, navigation, and interaction:

- Some Learning objects implement the instructional concept, where the others present the explanation of the knowledge to be submitted;
- Several solutions corresponds to constructivist approach and realize discovery learning;

- The explanations in many LOs are implemented to be used in the class, whether another solutions are prepared for non-supervised learning;
- Some LOs offer the two-level-explanations (short and extended) for new and experienced students. On the other hand, the short explanations could preferably be used in the class whereas the extended ones can be applied for the homework;
- The navigation concept also differs from one LO to another. Some provide flexible change from point-to-point while others implement step-by-step navigation;
- Some of the LOs require user input like numbers, reserved words or multiple choice selections.

Thirdly, various (groups of) LOs are prepared for different utilization:

- Main part of the implemented Learning objects are prepared for the teaching/learning in the auditorium. The (very) short explanation on the screen and the step-by-step navigation are the typical attributes of such solutions;
- There are several Learning objects which implement the task-metaphor. Here the students have to answer the question regarding the example presented on the screen to receive the feedback;
- Some implementations involve the input of numbers to explain e.g. the number of repetitions in some loop construction another present predefined examples.

Obviously the huge variety of LOs makes it difficult to figure out their comparable attributes. The analysis made allows the formulation of the following framework.

4. FRAMEWORK

To support the development of new and the evaluation of existing Learning objects following framework was defined:

1. Thematic orientation
2. Teaching/learning metaphor
3. Content visualization conception
4. Screen structure (corresponding to content visualization conception)
5. Navigation model
6. Interaction concept

All these elements have to be considered together accompanied by interdependencies to various environmental components. Because of the mentioned variety of the conceptions and resources used in the developed Learning objects the derived framework is very general and could be specified. To present the way of the further specification for this framework the development scenario for the Learning object will be considered in the next chapter.

5. DEVELOPMENT SCENARIO

In this chapter the scenario for the development of Learning object is presented. The most important decisions/steps are listed below including possible criteria.

- Decision about the topic/knowledge to be submitted
  a) Basic features/resources of programming languages (e.g. classes, objects, methods, variables, constants etc.);
  b) Standard concepts/algorithms with programming languages (e.g. loops, search or sorting algorithms, etc.);
  c) Special concepts/algorithms (graphical user interfaces, database programming, network applications, parallel computing, etc.);
- Decision about the metaphor for transferring the learning content:
  a) Supervised traditional classes versus non-supervised teaching/learning;
  b) Learning based on examples description;

c) Problem oriented explanation with focus on common problems;
d) Discovery learning (e.g. the exploration of the learning environment as part of the learning process to support adaptation of the learning process to the user requirements);

- Decision about screen design and placement of the screen elements:
  a) Important windows (e.g. source code, variables, control windows, memory, explanation, screenshot etc.);
  b) Labels of the windows on the screen;
  c) Placement of the windows on the screen;
  d) Consequent usage of the layout concept;
  e) Highlighting important parts of the screen (e.g. by different colours and fonts, etc.);
- Decision about navigation within the LO:
  a) Flexible change from point to point, according to user prompt;
  b) Simple step-by-step navigation through the learning content;
  c) Hyperlinks (e.g. for additional information);
- Decision about interaction during the session:
  a) Input of source code elements;
  b) Input of numbers and parameters that change the execution and the outcome of the LO;
  c) Selection of t
  f) Multiple choice test;
  g) Text input test;
  h) Test of reserved words;
  i) No test within
  d) he right answer (e.g. radio button);
- Testing the submitted knowledge:
  j) the LO (class based exams).

This list as mentioned before is not exhaustive and represents the basic level of the framework.

6. EVALUATION EXAMPLE

To demonstrate the usage of this framework we exemplarily apply the criteria to the before introduced LO example 6 from Codewitz shown in figures 6 and 7:

- Decision about the topic:

– The LO aims at supporting the understanding of database programming in Java. So it deals with special concepts of programming languages which requires the complex interaction of different resources.

- Metaphor for knowledge transfer:

– This example used the problem description at the beginning and the explanation of selected examples in the main part. Generally this LO is designed to be used in non-supervised learning.

- Screen layout and usage

– The screen is clearly subdivided into a source code window on the left and a brief explanation on the right. Within this explanation window one can find hyperlinks to additional information about the current line of code. These links open pop-up windows with text and sometimes visualizations aids.

– This particular LO offers a second view on the learning content. Besides the before described window with source code and short explanation one can browse it by starting an animation. The screen layout for this method of knowledge transfer also contains the source code on the left but on the right side it consists of several windows labelled "Explanation", "Program", "Java Resources", "Program Resources", "Database",

and "Screenshot". During the animation arrows underline the execution steps in selected lines.

- Navigation within the Learning object:

– The explanatory part of the LO empowers the user to navigate within the source code by scrolling buttons in the middle of the screen. This enables the student to focus on the lines of code to be learned in the session. It prevents from repetition of unnecessary code explanation. The second part, the animation, offers simple start and stop control buttons.

– In this solution the user can follow both the step-by-step navigation as well as the flexible and user oriented point to point browsing through the content.

- User interaction:

– In this LO there are no possibilities for the input of numbers or code fragments to change the execution or outcome of the example.

- Knowledge testing

– The LO contains no integrated tests of knowledge. For the usage as additional teaching material it should be combined with class based examination.

CONCLUSIONS AND FUTURE WORK

The proposed framework represents the basis for evaluation of developed Learning objects. In addition it can be utilized for the development of new solutions in the Codewitz and similar projects. The detailed specification of the framework and further analysis of Learning objects is scheduled as the upcoming research activity.

REFERENCES

[1] Bodrow, W., Bodrow, I., Wolf, S., 2006. Didactics and Implementation of Content for non-supervised Distance Learning. In: Tampere Polytechnic Univ. of Applied Sciences, MMT2006 - Methods, Materials and Tools for Programming Education, pp. 57-60, Tampere, http://www.codewitz.net/papers/MMT_57-60_Bodrow_Wolf_Bodrow.pdf

[2] Bodrow, W., Bodrow, I., 2006. Interaction Concepts for Learning Objects in Codewitz. In: e-learning conference '06 Computer Science Education, pp. 125 - 129, FCT (Fundação para a Ciência e a Tecnologia), Coimbra

[3] Bodrow, W., Bodrow, I., 2007. Knowledge Engineering and Visualization for Distance Education in Programming. In: Proceedings of the 10th IASTED Conference in Beijing (CATE 2007) pp. 176 – 179, ACTA Press Zurich

[4] Bodrow, W., Wolf, S., Bodrow, I., 2007. Challenge for the Learning Objects. In: R. Sadananda, E. Kujansuu (Eds.) Proceedings of the ITE2007, pp. 30 – 32, Bangkok

[5] http://www.codewitz.net

[6] http://www.microsoft.com/education/default.mspx

[7] http://en.wikipedia.org

ABOUT THE AUTHORS

Prof. Dr. Wladimir Bodrow, PhD, Business Informatics, University of Applied Sciences HTW-Berlin,

Phone: +49 (030) 5019-2478, E-mail: wladimir.bodrow@htw-berlin.de