Learning-by-doing approaches for skill acquisition

Ivo Hristov
Institute of Information Technologies – Bulgarian Academy of Science
Sofia 1113, acad. G. Bonchev st., bl. 2
E-mail: collin@iinf.bas.bg

Abstract
The paper discusses an implementation of learning-by-doing features allowing the learner to make exercises and projects, accessing external software – subject of study. An implementation based on Microsoft terminal services is offered and some results of usage are presented.

Keywords
E-learning, learning management system, sharable content object model (SCORM), learning-by-doing

1. Overview of architecture

The aggregate layered structure of the DTC /distant training center/ is shown on Figure 1. DTC functionality covers the basic functions and services for a generic learning management system /LMS/ with some specific features oriented to professional and continuous education forms [1]. The centre essentially provides common services, necessary to carry out working connection between the available courseware and the learners, including registration, following a course, assessment procedures, communication with instructors and peers, and some administrative procedures.

Blocks with dashed lines are discussed in the second part of the paper, which considers the organization of work with external software, subject of study, to allow the learners to make exercises and projects with it, acquiring and improving operational skills. Such learning-by-doing functionality is important for learning processes in the vocational e-training.

As seen on Figure 1, the architecture is based on layered approach. User interface is html based. DTC API is a set of modules that implement main subsystems: curriculum mapping, delivery, assessment, trainers support, authoring studio bases and student tracking. The DTC server hides the underlain physical data source complexity and presents standardized way for accessing resources.

2. Learning-by-doing by using simulations

Skills acquisition and mastering is an important learning aim in vocational e-training. The learner has to learn skills by interacting with the learning environment. This may be achieved only by incorporating pro-active courseware using learning-by-doing facilities and the vocational E-learning systems pay special attention to this functionality. The chapter discusses usage of
simulations to provide learning-by-doing facilities in LMS in order to permit the learners to acquire and train skills to work with ICT applications.

If all learning activities have to happen on-line, and blended learning is not affordable, many LMS use courseware with simulations to interact with learners. Typically simulation tasks are not delivered alone, but as a part of SCO or asset over the WEB by using http protocol. For this reason educational simulations have to easily integrate into html page and to have reasonable size to be delivered by the LMS. Macromedia Flash [2] might be used for preparing such simulations because of reach set of features available.

Simulation has advantages like:
- Using standard software for creation;
- Comparatively easy integration in SCOs within IMS content packaging specifications;
- Control of every learner’s action; can track learner’s most common faults,
- Emphasize learner’s activity in impressive way.

However, preparing courseware content in this situation is not conventional. Preparing content that integrate simulations lead to additional requirements:
- Making simulations requires courseware authors to learn additional simulation software or to use help of qualified specialists that work with software for simulation.
- Deeper knowledge of possible learners’ errors and misunderstandings is necessary in order to choose significant cases for simulation.

Besides, common reason considerations claim that no one could become a pilot by using only flight simulators. Another way we offer is learning by working with real software – subject of study.

3. Learning by working with real software subject of study in IT training

Most common situation is when learner has to be prepared to use a program or set of programs by doing different exercises and projects. An exercise in this content is an execution of a well defined task applying some functions of the software under study. Several consequent exercises may be chained to inherit intermediate results in order to construct at the end more complex object as a result of an exercise some files are created and have to be evaluated. Normally this evaluation goes well beyond the possibility for automatic processing and has to be made by the instructor. Therefore well designed and easy-to-use communication facilities between the learner and the instructor have to be included in the LMS in order to allow exchange of exercises results and their evaluations.

Making exercises by using the real software application subject of study might be realized by using two approaches depending where this software – subject of study is installed – on the local machine or on remote server. In both cases the learner uses his/her local computer for interacting with learning environment, but the workload is on the learner’s machine or on the remote server respectively. In other words two approaches are classified by the delivery process managed by LMS – local machine installation require download of exercise template and upload exercises being finished in contrary of remote machine working where LMS redirect learner to the appropriate exercise server with predefined settings.

If software, subject of study is installed on the local machine, LMS has no guarantee that all required software for learning can be installed and tuned correctly. Even though the second approach presents more requirements for the internet connection, it guarantees that learning environment is tuned like required by the courseware designer. For that reason we implemented the approach that utilizes working on remote machine.

The use case diagram on Figure 2 describes the general features that implement usage of remote machine for learning by doing. The main actors on Figure 2 are: learner, trainer, courseware author-writer and
For implementation of learning by working with real software - subject of study the courseware author have to prepare exercise file templates and to organise hints and recommendations how to do the exercises, delivered by the LMS. Trainers have to support learners while doing their exercises and projects with the real software and in many cases to evaluate the final results. The administrators have to prepare remote server(s) and software, subject of study corresponding to the courseware author and trainer specifications.

**Use cases on Figure 2 have the following meaning:**

- **UC1.0** describes doing exercises by the learner. He/she can start doing new exercise (UC1.1) or continue working on the same exercise (UC1.1), starting of the point of interruption or completely redo exercise (UC1.3) with clear author's template – loose current working on the exercise. It is common that learner has to announce which exercise is being finished and ready to be evaluated.

- **UC2.0** describes establishment of connection to the remote exercise server – automatically by LMS or by learner.

- **UC3.0** describes creation of exercises by the author. Exercise contains an exercise template – this is one or more physical file(s) (UC3.1), used as starting point when learner starts exercise. Template helps learners to save time and efforts of doing unimportant, and/or repeatable actions not directly connected with current task. UC3.2 contains supporting resources for doing exercise. Every exercise template can be implemented as a SCO in term of SCORM. The SCO might contain finished exercise or exercises steps, serving the need of assessment.

- **UC4.0** – describes OS and software policy required by the author and viewed and implemented (UC5.0) by the administrator and trainer. This is where courseware author specify operating system environment that are optimal for software being subject of training and version of the software being used in the exercises.

- **UC6.0** – describe requirements for exercise evaluation by the trainer. In many situation trainers have to had ability to view and annotate exercises that has not been finished yet for helping learners.

In order to be available for exercises and projects the software – subject of study has to be installed and tuned. In case of installation on local machine this may be a hard job for the learner even if on-line support by the trainer. If an incorrect version of the software is installed, the learner will be frustrated of differences between the courseware content, delivered by the LMS, and the behaviour of the used software. So the initial installation may require off-line support by the trainer or administrator. For this reason the local software installation of software under study is more appropriate for LAN environment. In case of using remote server for organising the work with the software - subject of study, the learners have access to predefined, ready-to-use environment.
4. An implementation by using Microsoft terminal services for accessing the software under study

We used Microsoft terminal services [3] for implementation of learning by doing exercises on remote machine. The terminal server is configured and started in application server mode. On the Figure 1 with dashed lines are presented building blocks of the offered solution that uses Microsoft terminal services – terminal console, exercises templates and exercises results. Terminal console is an internal part of LMS that manages learners and trainers activity related on usage of terminal services. At minimum the following functionality is supported:

• Integration with the learning environment – learners do not have to go out of the learning environment for doing exercises
• On each learner connection learner’s settings must be loaded and recovered as they was when learner exited the learning environment
• Support functions that help learner view state of every exercise, annotate exercise, mark exercise as finished and switch between different exercises
• Assist the learner with advice how to do the exercise on every stage
• Support instructor when he/she review exercises
• Support author when he/she develops exercise templates

Exercises templates are set of files that are used as starting point when learner starts doing exercise. The exercises files templates are developed by the courseware author, and are stored within directory structure managed by terminal service console within unique directory and files names. When learner tries to do the exercise for the first time, LMS creates directory structure for requested exercise within learner’s directory, maps it to the virtual drive and copies corresponding exercise file template. By this way every learner receives his own copy of exercise template. This set of files located on the learners’ virtual logical drive is exercises results marked on the Figure 1. Terminal console manages file names and locations and learner does not have to remember them. When learner leaves the learning environment, learner settings are automatically saved.

The implementation of the approach of learning by using real software – subject of study with Microsoft terminal services has following advantages:

• Managed by standard windows tools
• Ability to be accessed by remote desktop
• Possibility to be accessed by remote desktop Active X control, that can be integrated in the SCORM package

Most common disadvantages are:

• Requires software licenses for user’s connection
• Server part can be implemented on the windows machine only, with internet information server
• There is limitations of display resolution of the learner computer (client machines)
• Has some specific requirements for additional software installations

When a learner starts doing exercise the LMS opens connection to Microsoft terminal server and presents current learner’s settings. This connection stays open until the learner is connected.

5. Results and Conclusions

In the last years we have experimented implementations to allowing the execution of exercises with software systems – subjects of study in LMS framework [4], [5]. We got following difficulties when running the system that support learning by using software – subject of study:

• Trainers can evaluate only the final states of finished exercises and intermediate results cannot be tracked. That means that the exercises can be
done not in the most efficient way or by using wrong approaches.

- Exercises have to be relatively simple, because of the needs to be executed in single manner, defined by the trainer. Complex exercises, was restarted by the beginnings by the learners.

- Preparing simple exercises required extra work by the courseware authors and trainers. In many situations they were required to take into account different technical aspects of delivery not directly related to the subject of study.

- A connection to the terminal server was restricted by many proxies, because of the need of specific ports opening. Because of that, doing exercise was not possible at any time and place as we expected.

- If many learners were connected to the remote server or learners were doing complex exercises the remote terminal server loses performance and learners were not able to connect in reasonable time. Even LMS sends massages to learners to try connect later many learners were discouraged.

The results of these experiments shape the following possible future directions of investigations, oriented to extensions of the Bulgarian KNOSOS DTC in order to improve its learning-by-doing functionality:

- Practical ways to include exercise templates into SCORM assets and to realise the activation and data exchange with the real software application under study;

- Means for organising combinations of simulation and working with real software in order to allow the use in the courseware of more diverse proactive exercises of different types and levels;

- Means to monitor the learner actions on each session of working with the real software in order to track better his/her progress and give more feedback information to the trainers;

- Means to prepare load balancing solutions that handle learners’ connections in efficient way:

References


