Journal of Engineering and Applied Sciences 12 (Special Issue 9): 8607-8613, 2017
ISSN: 1816-949X

System of Network Infrastructure Configuring for Project Learning

Galina O. Artemova, Artem D. Beresnev, Kseniya I. Buraya and Natalia F. Gusarova
Faculty of Information Technologies and Programming, ITMO University,
Saint Petersburg, Russian Federation

Abstract: An important area of modern pedagogy is active learning and project-based learning as one of its styles. Modern computer-based projects are usually connected with the network infrastructure which today means the use of network and cloud services. The existing typical solutions for configuring and debugging are costly, their interface is too complicated for the participants of the pedagogical process who have insufficient IT skills, they do not allow to take into account the requirements of poorly defined information. To overcome identified difficulties and support project-based learning, we have developed the system of network infrastructure configuring for project learning that is devoid of these shortcomings. The system works directly in the real network environment which means that development, modeling, debugging and verification of a real network configuration are implemented simultaneously. The results of experiments we conducted to empirically evaluate the effectiveness of the developed system in comparison with a standard approach. We aimed to answer the following question: How does runtime of the task change for different groups of participants using the developed system and standard approach? We developed the error's typology and researched it distribution depending on the qualification of participants. The research shows that the lower the skill level of the participant is the more pronounced the advantages of the developed system are. The maximum payoff for the task is 35% for the beginners, the minimum is 5% for the high level skills. Our researches show that user support in systems of network infrastructure configuring for project learning should have dynamic structure and depend on the initial qualification and the current behavior of the user. It completely conforms to the requirements of adaptivity and intellectuality of the environments for project learning.

Key words: Network configuration, modeling environment, design infrastructure, project learning, development, skills

INTRODUCTION

An important area of modern pedagogy is active learning and project-based learning as one of its styles. Active learning is commonly defined as an instructional method that engages students in the learning process (Bonwell and Eison, 1991). Frequently active learning requires from students to do experimental learning activities for solving problems of their own.

The review of published works shows that large numbers of studies for support active learning and student projects are presented. A significant body of research on active learning and project-based learning methods review improvement of approaches and tools for improved learning outcomes. In the tool support the main directions can be considered as: development of adaptive/intelligent environments (Riad *et al*., 2009; Nacheva-Skopalik and Green, 2011; Velusamy *et al*., 2013). Development of intelligent virtual agents which are able to adaptively interact with students (Boyer *et al*., 2011; Jackson, 2008; Miikkulainen, 2006). Studying the role of affect in the learning process, i.e., impact of the emotional factors in addition to cognitive on the learning outcomes (e.g., interest and involvement) (Shute *et al*., 2009, 2013) assessment of learning results in implicit mode, through analysis of user interaction with learning environment (Baker *et al*., 2010; Grafsgaard *et al*., 2012; Calvo and D'Mello, 2010).

One can allocate the main advantages of these approaches: flexibility, interactivity and supporting self-learning (Thomas, 2001).

Simulation in the e-Learning tools allows to activate learning process and to prepare students for applying their knowledge and skills in practice (Dori and Belcher, 2005; Janitor *et al*., 2010).

Visual models significantly improve the learning outcomes and understanding of abstract content by using a computer program that contains (simulates) the system, the process or equipment for training or experimentation (Dighe, 2003; De Jong and Van Joolingen, 1998).

Corresponding Author: Galina O. Artemova, Faculty of Information Technologies and Programming, ITMO University, Saint Petersburg, Russian Federation

Students research as a team creating an artifact presenting their gained knowledge. The nomenclature of projects is rapidly growing and more and more artifacts are performed in the computer environment. That's why designing of specialized infrastructure for each project is ineffective due to a lack of resources and specific IT specialists in the education sector and the united flexible software and hardware infrastructure is required. Such infrastructure should correspond to a wide variety of the contradictory requirements arising from different fields-pedagogical, technological, social, etc.

We show that in order to systematize these requirements in the implementation of pedagogical projects it is reasonable to use the approach proposed by Saracevic (1995). The approach proposes to divide the objective of the technical, computer-based system into 6 levels from the engineering level up to the social one. At the same time, it is crucial to redefine the content of each level with the active learning features. In this study, we confine ourselves to the first level where maintainability is the main characteristic of the effectiveness in pedagogical project groups.

Modern computer-based projects are usually connected with the network infrastructure which today means the use of network and cloud services. The existing typical solutions for configuring and debugging such infrastructure (for example, NetCracker, Cisco Packet Tracer) research in a model environment and then transfer the created configuration to real network environment. They are costly; their interface is too complicated for the participants of the pedagogical process who have insufficient IT skills; they do not allow to take into account the requirements of poorly defined information; the transfer of configuration from model to real environment can inspire errors that are difficult to detect with a lack of skills.

MATERIALS AND METHODS

Currently, service-oriented approach is dominant in the field of system integration and management of its infrastructure. Such documents as specific standards (ISO 20000), libraries of procedures (IT Infrastructure Library, ITIL) and corporate practices (Microsoft Operations Framework, MOF) are built upon the service-oriented approach.

The service is considered as a part of the collection of features and functions that enable a business process. The service is based on the process management of configuration items (Configuration Item, CI) which can be represented on high-level sub systems as specific technical elements, exemplified by the network configuration of compute node. But at the heart of the service are inter alia, the process management of configuration items (Configuration Item, CI) which can be represented and high-level subsystems and specific technical elements, exemplified by the network configuration of compute node. In our research we define network configuration as a set of TCP\IP stack's configuration data such as:

Table 1: The experiment's participants

Groups	Characteristic of participants	No. of participants
A	Students of technical specialty before studying a course "Administrating of information networks"	6
B	Students of technical specialty after studying a course "Administrating of information networks"	8
C	Teachers of disciplines of IT	2
D	System engineers	2

- The address information of the network layer (IPv4 or IPv6)
- Service recognition name system (DNS) configuration
- Firewall configuration
- Routing configuration, etc.

The process of developing network configurations of the services is described in the example of the MOF [https://technet.microsoft.com/en-us/solutionaccelerators/dd320379.aspx]. MOF is a process oriented technique, that consists of three phases (the plan phase, the deliver phase, the operate phase) and the manage layer. Each phase consists of a set of service management function (of the basic processes which in turn is made up of finite processes). We highlight the main processes and artifacts by certain stage of the life cycle from early design to implementation services (delivery) for the development of network configuration. Table 1 presents summarized information about functions, goals and results (artifacts) used in the development process of network configurations.

Using the requirements analysis the process of developing network configurations via. the MOF can be reduced to the following steps:

- Initial research with the customer (conceptualization of the decision)
- Development of a prototype, maybe in a model environment
- Testing and debugging of the prototype, release the configuration
- Transferring the configuration to the real system (equipment)
- Testing and debugging of real configuration

- Transferring the actual configuration to the customer for operation

These steps can be repeated iteratively to achieve the desired results or until receiving instructions for the completion of the management. The difficulties in implementing this approach are: on the stage of conceptualization of solution the design is carried out in the linguistic form without mandatory initial prototyping which can lead to inaccuracies in the portfolio project. There is an obvious gap when transferring configurations from the model environment to the real environment. The process of configuration migration involves errors caused by human factor and objective differences in platforms and modeling environments. Problems in the transferring solution process can occur even if the model environment and the real environment are the same by versions of the platforms, for example because it is often not possible to provide a quantitative similarity between them.

Commonly these difficulties are offset by better training of specialists in developing network configurations. For commercial projects, for example, a team of professionals is gathered and they implement the project according to the recommendations of the MOF. But in computer-based learning project it is difficult to employ highly qualified networking specialists for each experiment (learning project, startup, etc.), especially in those projects where the design and configuration of network infrastructure are auxiliary but retained the high requirements for the quality of the solution and/or ongoing support of their activities.

To overcome identified difficulties and support project-based learning, we offer an approach that suggests combiningthe steps of development, simulation, network configuration and the tool of its delivery in one environment.

We have developed the system of network infrastructure configuring for project learning that is devoid of these shortcomings. The system works directly in the real network environment which means that development, modeling, debugging and verification of a real network configuration are implemented simultaneously. It makes possible to avoid the errors in manual transfer of configurations as well as the problems caused by contrast in the modeling environment and the real systems.

The main difference between our and others systems lies in working with poorly defined information: other systems for networking configuration design (ex. Cisco Packet Tracer) ignore them (Bonwell and Eison, 1991; Janitor *et al*., 2010) but we have put them up as base properties of the model environment. The system consist of the following functional components (Fig. 1):

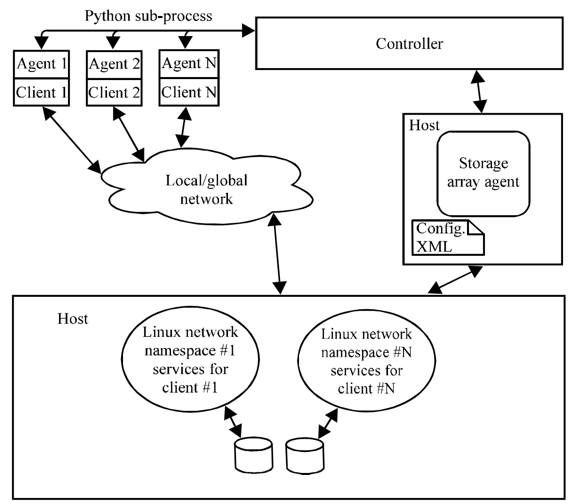

Fig. 1: Architecture of the developed system

- Controller main controlling element of the system
- Sends/receives/takes teams from all system entities

Agent is a separate system process that receives commands from controller via. TCP/IP. The agent can run its own sub-processes, create its own sub-entities and perform complex sets of terminal commands. In the beginning each computer node already runs agents at the test stand. However, the user may run them himself when needed, storagearray agent advanced entity type agent. The main goal of this entity is deployment network configuration. This agent may be located on the storage device and on a separate host. It is assumed that the agent is runningon the physical storage device. The system has the following functionalities:

- Edit (create new) the network configuration
- View the network infrastructure in *.xml file or graphically on the console screen
- Add tests for network infrastructure
- View reports on passage of tests

RESULTS AND DISCUSSION

This study analyzes the results of experiments we conducted to empirically evaluate the effectiveness of the developed system in comparison with a standard approach. In experiments 18 people were involved having different IT skills from high level to beginners (Table 1).

All participants were given the same task, namely to deploy the computer network in accordance with the specified functional requirements and resource constraints. The target scheme is shown in Fig. 2. The following functional requirements were set: nodes receive addresses by the following rule: routers-serially from the first address, hosts-serially from the 100th address. On servers the services specified on the diagram are launched. On client computers client programs of all services are launched.

All the participants performed the task sequentially in two systems in the typical system of network simulation cisco packet tracer and in the developed system. The task consisted of 2 stages prototyping and deployment, each stage included the following steps:

- Physical network creating
- Nodes addresses configuring, checking of accessibility on LAN
- Static routing configuring (on the LAN1 network the computer, upper on the diagram, uses the router, upper on the diagram for an output in an external network and similarly for couple "the lower computer the lower router"
- Checking of accessibility of all nodes from all nodes of a network
- Starting and configuring of services

During the experiment we fixed for each participant thr following data: the execution time of the task, the time for finding and correcting each error, the number and the typology of the committed errors. The error's typology which we used is shown in Table 2.

In the first experiment we aimed to answer the following question: how does runtime of the task change for different groups of participants using the developed system and standard approach? Results of an experiment are shown in Fig. 3.

The analysis of Fig. 2 shows that use of the developed system gives good effect for groups with feeble preparation (Table 2, Groups A, B) but in process of qualification growth (Table 2, Group D) this distinction is leveled.

Table 2: The error's typology

Error's type/Error's content	Error's example
1	
Errors of a human factor	Data entry errors
2	
Errors of equipment configuring	Errors of a choice of interfaces, physical lines, etc.
3	
Template errors	The errors caused by specifics of operation of the environments (a format of configuration files, standard of naming of controls, sequrity policy time scale, need of reset of elements of model, etc.)

Type 1 is matches with difficult b; type 2-difficult c; type 3-difficult a in method section

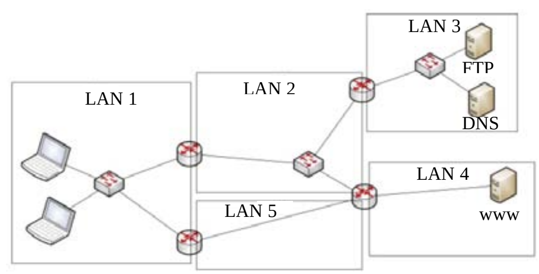

Fig. 2: The target scheme: LAN1-10.1.0.0/16, LAN2-10.2.0.0/16, LAN3-10.3.0.0/16, LAN4-10.4.0.0/16, LAN5-10.5.0.0/16

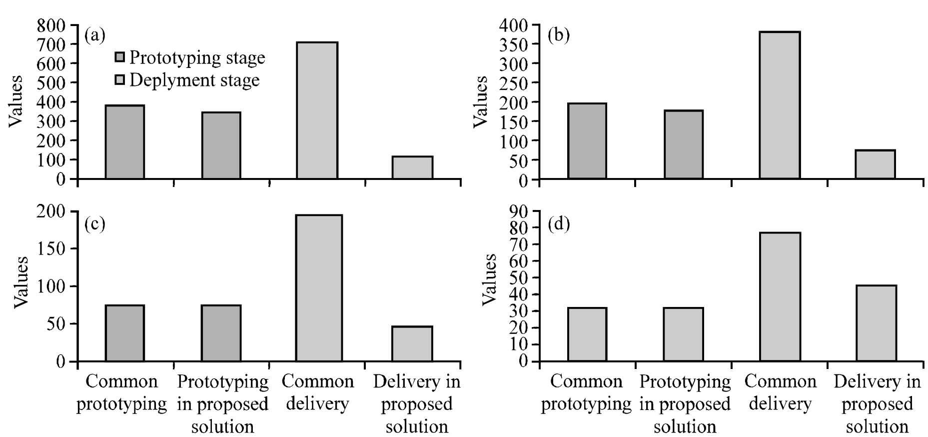

Fig. 3: Histograms of runtime of the task in the Cisco Packet Tracer and in the developed system for each group of participants; a) Group A; b) Group B; c) Group C and d) Group D

Table 3: Error distribution on types depending on qualification of participants. The average number of errors in the standard process, (Group A)

Error type	Steps 1	2	3	4	5	Sum
1	3.0	4	5	3	3	18.0
2	0.5	-	5		1	6.5
3	1.0	2	1	2	-	6.0
Sum	4.5	6	11	5	4	30.5

Table 4: Error distribution on types depending on qualification of participants. The average number of errors in the standard process (Group B)

Error type	Steps 1	2	3	4	5	Sum
1	1	5	3	2	3.0	14.0
2	1	-	5	2	0.5	8.5
3	1	1	1	-	3.0	6.0
Sum	3	6	9	4	6.5	28.5

Table 5: Error distribution on types depending on qualification of participants. The average number of errors in the standard process (Group C)

Error type	Steps 1	2	3	4	5	Sum
1	0	3	3	2	1	9
2	1	2	1	2	1	7
3	1	1	1	-	1	4
Sum	2	6	5	4	3	20

Table 6: Error distribution on types depending on qualification of participants. The average number of errors in the standard process (Group D)

Error type	Steps 1	2	3	4	5	Sum
1	1	-	-	-	-	1
2	-	1	1	-	2	4
3	1	1	1	-	2	5
Sum	2	2	2	0	4	10

Table 7: Error distribution on types depending on qualification of participants. The average number of errors in the proposed solution (Group A)

Error type	Steps 1	2	3	4	5	Sum
1	2	2	1	2.0	1	8.0
2	1	-	3	0.5	2	6.5
3	1	2	1	-	1	5.0
Sum	4	4	5	2.5	4	19.5

In the second experiment we researched typology of errors depending on qualification of participants. Results of an experiment are shown in Table 3-10. Data analysis of the table shows:

- In group of qualified users (Group D) summary quantity of errors is the same both in the developed system and in standard system. At the same time, in group of qualified users, errors of type 3 prevail irrespective of system used

Table 8: Error distribution on types depending on qualification of participants. The average number of errors in the proposed solution (Group B)

Error type	Steps 1	2	3	4	5	Sum
Type 1	2	2	1	4	2	11
Type 2	-	1	2	1	2	6
Type 3	1	2	1	-	1	5
Sum	3	5	4	5	5	22

Table 9: Error distribution on types depending on qualification of participants. The average number of errors in the proposed solution (Group C)

Error type	Steps 1	2	3	4	5	Sum
1	2	1.0	1	1	2	7.0
2		0.5	2	2	2	65
3	1	2.0	1	-	1	5.0
Sum	3	3.5	4	3	5	185

Table 10: Error distribution on types depending on qualification of participants.The average number of errors in the proposed solution (Group D)

Error type	Steps 1	2	3	4	5	Sum
1	1	-	-	1	-	2.0
2	-	-	-	1	1	2.0
3	-	0.5	2	1	2	55
Sum	1	0.5	2	3	3	95

- In groups of unskilled users (Group A, B) errors are distributed uniformly and don't depend on error's type

The received results can be interpreted as follows. Users of Group D have own behavior model in case of setup of a network and they act according to their understanding of the general principles of network functioning, ignoring the offered support (see item 1). Moreover, the offered model can even enter the conflict with their internal model and cause additional errors of simulation (see item 2).

Thus, our researches show that user support in systems of network infrastructure configuring for project learning should have dynamic structure and depend on the initial qualification and the current behavior of the user. It completely conforms to requirements of adaptivity and intellectuality of the environments for project learning.

The research shows that the lower the skill level of the participant is the more pronounced the advantages of the developed system are. The maximum payoff for the task is 35% for the beginners, the minimum is 5% for the high level skills.

CONCLUSION

An important area of modern pedagogy is active learning and project-based learning as one of its styles. Modern computer-based projects are usually connected with the network infrastructure which today means the use of network and cloud services. The existing typical solutions for configuring and debugging are costly, their interface is too complicated for the participants of the pedagogical process who have insufficient IT skills; they do not allow to take into account the requirements of poorly defined information.

To overcome identified difficulties and support project-based learning, we have developed the system of network infrastructure configuring for project learning that is devoid of these shortcomings. The system works directly in the real network environment which means that development, modeling, debugging and verification of a real network configuration are implemented simultaneously.

The results of experiments, we conducted to empirically evaluate the effectiveness of the developed system in comparison with a standard approach. We aimed to answer the following question: How does runtime of the task change for different groups of participants using the developed system and standard approach? We developed the error's typology and researched it distribution depending on the qualification of participants. The research shows that the lower the skill level of the participant is the more pronounced the advantages of the developed system are. The maximum payoff for the task is 35% for the beginners, the minimum is 5% for the high level skills. Our researches show that user support in systems of network infrastructure configuring for project learning should have dynamic structure and depend on the initial qualification and the current behavior of the user. It completely conforms to the requirements of adaptivity and intellectuality of the environments for project learning.

REFERENCES

Baker, R.S.J.d., S.K. Mello, M.M.T. Rodrigo and A.C. Graesser, 2010. Better to be frustrated than bored: The incidence, persistence and impact of learners cognitive?affective states during interactions with three different computer-based learning environments. Int. J. Hum. Comput. Stud., 68: 223-241.

Bonwell, C. and J. Eison, 1991. Active Learning: Creating Excitement in the Classroom AEHE-ERIC Higher Education Report No. 1. Jossey-Bass, Washington, D.C., USA.,.

Boyer, K.E., R. Phillips, A. Ingram, E.Y. Ha and M. Wallis *et al*., 2011. Investigating the relationship between dialogue structure and tutoring effectiveness: A hidden Markov modeling approach. Intl. J. Artif. Intell. Educ., 21: 65-81.

Calvo, R.A. and S. D'Mello, 2010. Affect detection: An interdisciplinary review of models, methods and their applications. IEEE. Trans. Affective Comput., 1: 18-37.

De Jong, T. and W.R. van Joolingen, 1998. Scientific discovery learning with computer simulations of conceptual domains. Rev. Educational Res., 68: 179-201.

Dighe, A., 2003. Understanding our Learners. In: Educational Multimedia: A Handbook for Teacher Developers, Usha, V.R. and M. Sanjaya (Eds.). Commonwealth Educational Media Centre for Asia, New Delhi, India, ISBN: 9788188770007, pp: 21-26.

Dori, Y. and J. Belcher, 2005. Learning Electromagnetism with Visualizations and Active Learning. In: Visualization in Science Education, Gilbert, J.K. (Ed.). Springer, Berlin, Germany, ISBN:978-1-4020-3612-5, pp: 187-216.

Grafsgaard, J., K. Boyer and J. Lester, 2012. Toward a machine learning framework for understanding affective tutorial interaction. Proceedings of the 11th International Conference on Intelligent Tutoring Systems, June 14-18, 2012, Springer, Chania, Crete, Greece, pp: 52-58.

Jackson, G.T., 2008. Learning Through Natural Language Dialogue that Adapts to the Cognitive and Affective States of the Learner. In: Recent Innovations in Educational Technology that Facilitate Student Learning, Robinson, D.H. and G. Schraw (Eds.). Information Age Publishing, Charlotte, North Carolina, USA., ISBN:978-1-59311-653-8, pp: 95-125.

Janitor, J., F. Jakab and K. Kniewald, 2010. Visual learning tools for teaching-learning computer networks: Cisco networking academy and packet tracer. Proceedings of the 6th International Conference on Networking and Services (ICNS) 2010, March 7-13, 2010, IEEE, Kosice, Slovakia, ISBN:978-1-4244-5927-8, pp: 351-355.

Miikkulainen, R., 2006. Creating intelligent agents in games. Bridge, 36: 5-13.

Nacheva-Skopalik, L. and S. Green, 2011. E-Assessment for adaptable personal learning. Proceedings of the International Conference on Information Technologies (InfoTech-2011), September 15-16, 2011, ACM, Sofia, Bulgaria, pp: 363-372.

Riad, A.M., H.K. El-Minir and H.A. El-Ghareeb, 2009. Review of E-learning systems convergence from traditional systems to services based adaptive and intelligent systems. J. Convergence Inf. Technol., 4: 108-131.

Saracevic, T., 1995. Evaluation of evaluation in information retrieval. Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, July 09-13, 1995, ACM, Seattle, Washington, USA., ISBN:0-89791-714-6, pp: 138-146.

Shute, V.J., M. Ventura and Y.J. Kim, 2013. Assessment and learning of qualitative physics in newton's playground. J. Educ. Res., 106: 423-430.

Shute, V.J., M. Ventura, M. Bauer and D. Zapata-Rivera, 2009. Melding the Power of Serious Games and Embedded Assessment to Monitor and Foster Learning. In: Serious Games: Mechanisms and Effects, Ritterfeld, U., M. Cody and P. Vorderer (Eds.). Taylor and Francis, Mahwah, New Jersey, ISBN:31-978-0-203-89165-0, pp: 295-321.

Thomas, R., 2001. Interactivity and simulations in E-learning. Multiverse Solutions, Noida, India, http://www.jelsim.org/resources/whitepaper.pdf.

Velusamy, B., S.M. Anouneia and G. Abraham, 2013. Reinforcement learning approach for adaptive E-learning systems using learning styles. Inf. Technol. J., 12: 2306-2314.