

# Using artificial intelligence methods and 3D graphics for implementation a computer simulator for ophthalmology

**V. V. Gribova, M. V. Petryaeva, L. A. Fedorischev**

Intelligent System Laboratory, Institute of Automation and Control Processes, the Far Eastern Branch of Russian Academy of Sciences  
Vladivostok, Russia

## **Abstract**

A software service of the computer teaching simulator for ophthalmology with virtual reality and its implementation are considered in the paper. The implementation of the service is based on the methods of artificial intelligence. The simulator is oriented for professional skills training of medical university students.

*Keywords: computer simulators, teaching systems, virtual reality, ontology*

## **1 Introduction**

Urgency of computer teaching systems is well-known and widely discussed in literature. Their main advantages are [1]: accessibility, improving the quality and effectiveness of training, and the possibility of practicing repetitive actions and scenarios that are needed in future professional activity, etc.

A process of development and especially maintenance of computer simulators is very laborious; it demands the high level of the programmer's qualification and concludes coding complicated scripts or programs using imperative programming languages with following their integration. Teaching simulators have a lot of domain knowledge, but domain experts can't directly (without

engineers and programmers) take part in the development of such simulators, because they don't understand how to use the special tools of development and maintenance. As a result, the life cycle of such kind of software is still very short. Modification and extension of functional features of them are so complicated processes.

In order to solve above mentioned problems, the authors have developed the cloud service for creation and maintenance of computer simulators using 3D graphics and artificial intelligence methods. The model and the implementation methods of the service are described in [2-4]. Using this service the computer teaching simulator for ophthalmology was developed. It includes teaching tasks on classic methods of examination in ophthalmology: examination of central and periphery vision, examination of an optical system of an eye, and binocular vision.

The aim of this paper is to describe: the architecture of the software service for creation and using teaching simulators, steps of simulator development, and fragments of the medical computer simulator for examination of visual acuity by Sivcev-Golovin's tables.

## 2 SOFTWARE SERVICE FOR DEVELOPMENT OF TEACHING SIMULATORS

The software service is intended for development, maintenance, and using teaching simulators with virtual reality [4]. It is created using the multi-agent cloud platform IACPaaS (Intelligent Application, Control and Platform as a Service), which provides controlled access and a unified system of administration for the development of intelligent services and their components represented by semantic networks and agents, and also supports launching and execution of services. [5-6].

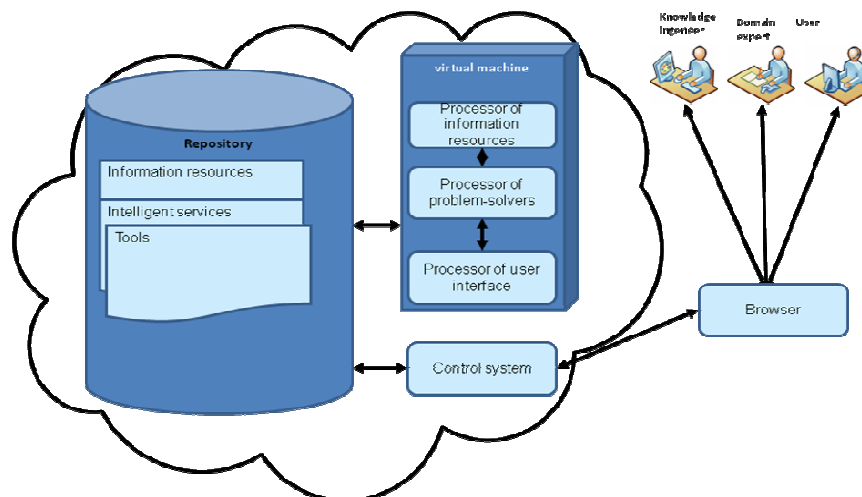


Fig.1 The architecture of the IACPaaS platform

The platform is based on the cloud computing technology and provides the remote access to intelligent systems to use them for users and developers for their development and maintenance (Fig. 1).

The software service for creation and using teaching simulators comprises informational and program components (Fig. 2) which are saved in the repository of the IACPaaS platform. The informational components are: the ontology of the virtual environment, a model of the teaching simulator, multimedia data and agents. The program components consist of the following services: the structural editor (for forming domain knowledge in the model of the teaching simulator), the graphical editor (for forming visual representation), the agent editor, and the interpreter of the virtual environment. The interpreter has two parts: the server application and the web-client.

Because the service has some different tools (the program components) therefore different groups of developers (domain experts, designers, and programmers) have to take part in a simulator development. The ontology describes the system of terms for development of a model of the simulator thus simplifies its creation. Developers do not need to study any special language for a model creation (it is important for domain experts because they do not know any program languages and technologies of development of program systems). The model is created in the terms of the ontology. Editors are controlled by the ontology and so also simplify labourness of development. The ontology is divided in two levels – the logical level and the presentation level. This is done to separate the work between developers – domain experts and designers [2].

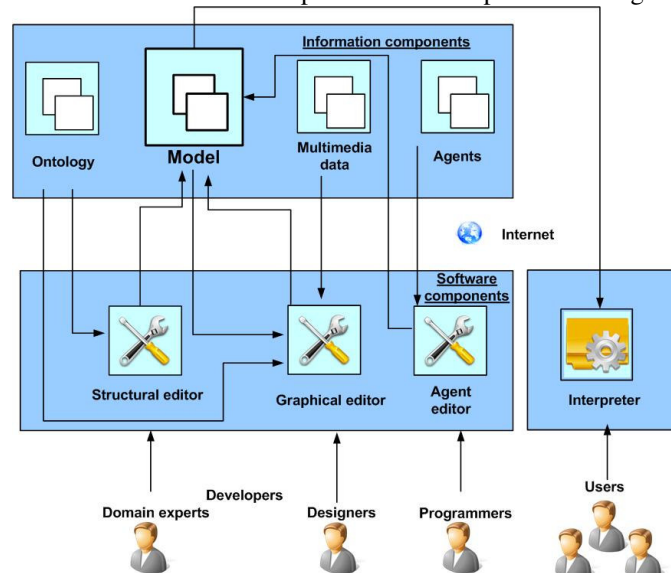


Fig. 2. Architecture of the program software for development and use of virtual simulators.

At the first step the domain expert forms the logical part of the model by the structural editor. He or she defines objects of the virtual world, their logical attributes, actions, and the scenario.

At the second step the logical part of the model is loaded to the graphical editor and the designer forms the graphical representation of the virtual world for the objects formed by the domain expert. The designer complements the logical part of the model with 3d-meshes and textures of the objects, creates graphical images of the objects, the full visual representation of the virtual scene, and then tests them.

At the third step if it is necessary programmers realize mathematical, string, and other specific functions. Users (students, teachers) launch the simulator using the administrative system of the IACPaaS platform.

The teaching simulators differ from other kinds of simulators and virtual environments by the scenario which describes the steps to be done by a student to achieve a goal. Users' actions and commands are compared with the scenario and the result is given. The student and the teacher may analyze the result.

### 3 IMPLEMENTATION OF THE SIMULATOR

The computer teaching simulator consists of teaching tasks on classic methods in ophthalmology: examination of central and periphery vision, examination of the optical system of an eye, and binocular vision. It developed in accordance with the technology above mentioned. A fragment of the teaching simulator on examination of the visual acuity by Sivcev-Golovin's table is presented.

#### 3.1 Logical part of the simulator model

**The non-formal description of the examination method.** The characteristic that a student must determine is the acuity of vision of a virtual patient. It is generated in a random manner. To determine it a student in the role of ophthalmologist ask the virtual patient letters in the ophthalmology table in accordance with the rules of examination (the student must know the rules and the order of examination). The virtual patient can give the following answers: "can't see", "name the letter" (correctly or incorrectly), or can give any answer with the delay. The answer of the virtual patient depends on the acuity of his vision. The student estimates the given answers and makes a decision either to continue the examination or to give the result (the acuity of vision), based on correlation between the number of wrong answers of the virtual patient and values of characteristics of the vision acuity.

**The description of the scene.** A lighted room, a chair for a patient near the window, the patient sitting on the chair, an apparatus of Rot is hanging on the wall opposite to the window on the distance of 5 meters from the patient and 1.2 meters from the floor, the Sivcev-Golovin's table in the Rot's apparatus, a desk for tools by the window or the wall, an occluder laying on the desk.

*Formal description of some objects of the scene*

Occluder

type: changeable, description: "a dashboard of an opacity material with a handle to cover one eye when the other eye is examined"

states set: ["lays on the table", "in the left hand of a patient", "in the right hand of a patient", "close the left eye", "close the right eye"]

Rot's apparatus

type: changeable, description: "the lighter for tables to examine the acuity of vision"

states set: ["off", "on"]

Sivcev-Golovin's table

type: table, description: "the table comprise 12 rows of optotypes "letter""

elements:

1: [«Ш», «Б»],

2: [«М», «Н», «К»],

....

12: [«И», «М», «Ш», «Б», «И», «Б», «М», «К»].

*Formal description of some actions*

Turn on a Rot's apparatus

type: interactive action

description: "a user clicks on a Rot's apparatus, it is turned on and lights a table",

objects' states: {

object "Rot's apparatus", state "on" }

Name letters on a row

type: interactive action

description: "a user clicks on a row of a table in order to a patient names letters on the row"

input parameters: [object "Sivcev's table", attribute "elements"]

result: define by number of row

estimation: ["0 errors", "1 error", "2 errors", ">2 errors", "not see"]

parameters of estimation:

acuity: [{object "right eye", attribute "acuity of vision"}, {object "left eye", attribute "acuity of vision"}]

row: {object "Sivcev's table", attribute "elements" }

sets of values:

acuity: 0.1

[estimation: "0 errors", row: 1],

[estimation: "1 error", row: 2-12],

...

### 3.2 Graphical presentation of the simulator

The graphical representation of the scene forms the designer. All objects described by the domain expert have a graphical representation. For every object its model (named "meshes") and texture is created by using external editors. Then they are loaded in the graphical editor of the software service and the

designer is positioning, rotating, and scaling them compared to other objects of the scene (Fig. 3).

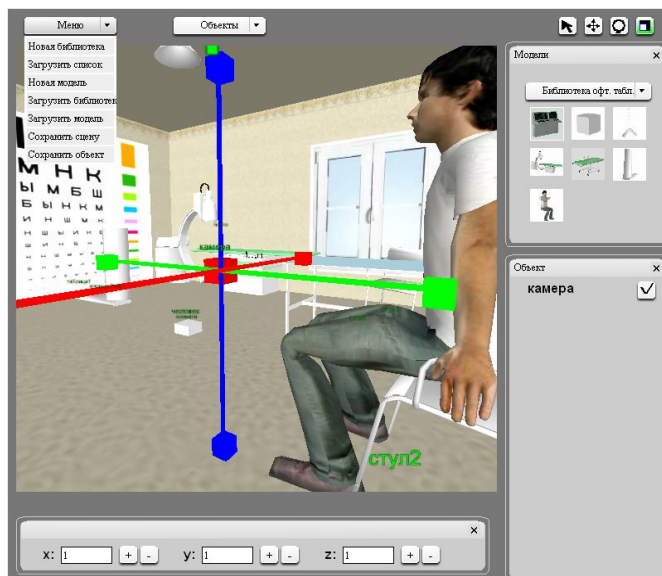


Fig. 3. Graphical representation of the scene in the editor.

### 3.3 Implementation of agents for the simulator

A programmer develops agents for processing specific functions of the simulator. For the simulator for ophthalmology the specific function is computing the row of the Sivcev-Golovin's table pointed by the student. The agent has information about a current state of the scene and about parameters of the completed action. The agent receives a description of elements of the table from the scene description, coordinates of a mouse, computes the area of the table, and returns the result to the service.

## 4 CONCLUSION

The paper describes a software system for development teaching simulators with virtual reality, steps of simulator development, and fragments of the medical computer simulator for examination of the visual acuity by the Sivcev-Golovin's table.

The software service is problem-independent and developed on the multi-agent cloud platform IACPaaS. Basic ideas, realized in the service are the following: development the model of the simulator and its interpretation instead of coding the simulator using programming languages, involving domain experts and designers in the development process, and development the model of a simulator using editors controlled by the ontology of the virtual environment.

The problem-independent software service can be used not only for creation teaching simulators but also for creation leaning courses, virtual laboratories, and animated videos. Using the software the virtual chemical laboratory and the virtual demonstration of a district of the city have been implemented.

## **Acknowledgment**

The research was supported by the Russian Foundation for Basic Research, the projects 14-07-00270a, 13-07-00024-a

## **REFERENCES**

- [1] V. V. Gribova, G. N. Osipenkov, and S. A. Sova, "Concept of development of diagnostic computer simulators on base of knowledge" *International Book Series "Human Aspectsof Artificial Intelligence"*. Varna, Bulgaria, 2009. P. 27-33
- [2] V.V. Gribova, M.V. Petryaeva, and L.A. Fedorischev, "Development of the virtual world for the medical computer simulator" // *Distance and Virtual Education* № 9, pp.56-66 (2011).
- [3] M. Yu. Chernyakhovskaya, V. V. Gribova, M. V. Petryaeva, and L.A. Fedorischev, "The model of a virtual world of a multimedia simulator for medical education" // *International Book Series "InformationScience & Computing"*, №22.- P. 140-148.
- [4] V.V. Gribova, L.A. Fedorischev, "The architecture of Internet software environment for creating teachware with virtual reality" // *Emerging Intelligent Computing Technology and Applications*. – Springer Berlin Heidelberg, 2012. Vol. 304, № 11. – Pp. 394-399.
- [5] V.V.Gribova, A.S. Kleshchev, E.A. Shalfeeva, "Control of Intelligent Systems" // *Journal of Computer and Systems Sciences International*. 2010. Vol. 49. No. 6. Pp. 952–966.
- [6] V.V. Gribova, A.S. Kleshchev, D.A. Krylov, P. Moskalenko, V. Timchenko, E.A. Shalfeyeva, M. Goldstein, "A software platform for the development of intelligent multi-agent internet-services" // *Proceedings of the Distributed Intelligent Systems and Technologies Workshop (DIST'2013)*. 1-4 July 2013. St. Petersburg, Russia. – Pp. 29-36.