

RESEARCH ARTICLE | MARCH 10 2015

Application of the two-hemisphere model supported by BrainTool: Football game simulation

Oksana Nikiforova; Uldis Sukovskis; Konstantins Gusarovs

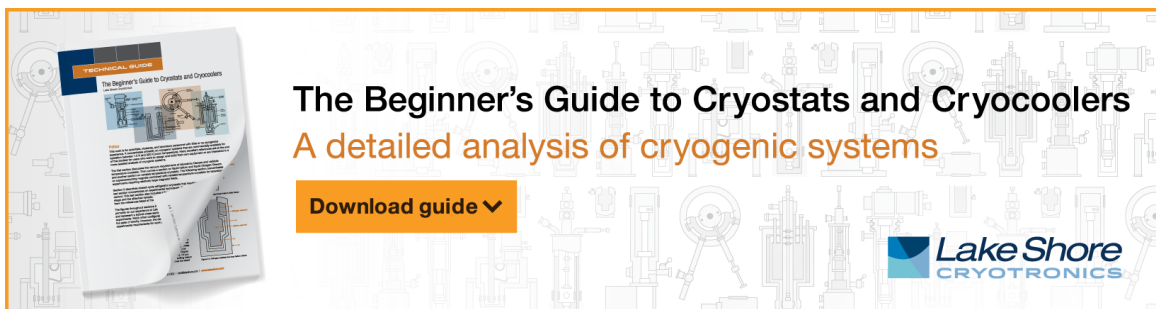



AIP Conf. Proc. 1648, 310004 (2015)

<https://doi.org/10.1063/1.4912557>




CrossMark






The Beginner's Guide to Cryostats and Cryocoolers

A detailed analysis of cryogenic systems

Download guide 



Application of the Two-Hemisphere Model Supported by BrainTool: Football Game Simulation

Oksana Nikiforova^{1, a}, Uldis Sukovskis^{1, b} and Konstantins Gusarovs^{2, c}

¹*Riga Technical University, 1 Kalku iela, Riga, Latvia, LV1658.*

²*Accenture Latvian Branch, Brivibas gatve 214, Riga, LV-1030.*

^a*Corresponding author: oksana.nikiforova@rtu.lv*

^b*uldis.sukovskis@rtu.lv*

^c*konstantins.gusarovs@gmail.com*

Abstract. The Unified Modeling Language (UML) is an industrial standard for object-oriented software specification, which offers notational conventions for system modeling at the initial stage of software development. An actual problem is to develop a tool available for automatic generation of UML diagram from some form of the problem domain description. Authors have been introduced such a tool, called BrainTool and developed by a research group in Riga Technical University. BrainTool supports transformation of the problem domain presented as two-hemisphere model into several kinds of UML diagrams and give an ability to export these diagrams for further software development into UML compatible modeling tools and IDEs. Present paper demonstrates the comparison results of the manual UML class diagram creation for an abstract example of the football game simulation to the UML class diagram received by application of BrainTool for the same game basic scenario. These comparison results give a basis for discussion about possible improvements of the transformations offered by two-hemisphere model and potentially “richer” usage of the model for generation of the UML diagrams suitable for further software development.

Keywords: Model drive software development, Model transformation, Two-hemisphere model, UML class diagram

PACS: 68U07 Computer-aided design, 68T35 Languages and software systems (knowledge-based systems, expert systems, etc.), 68P30 Coding and information theory, 68N19 Other programming technologies, 68R10 Graph theory (including graph drawing)

INTRODUCTION

The object-oriented approach is widely used in the software development. One of the tasks of software development is to present different aspects of the system for the implementation of the software solution for the required system. In solving this task, system modeling became an important activity in software development. The goal of system modeling is to represent the system graphically, in a form understandable to analysts, developers and at least partly or intuitively understandable for the customer. Unified Modeling Language (UML) [1] is an industry standard for software specification and modeling in an object-oriented manner. The UML class diagram is used to model class specification and serve as a “bridge” between the information about the problem domain and the information required for definition of the software components and their architecture. Currently, researchers are trying to achieve a high enough level of automation in creation of the UML class diagram and derivation of the diagram from information about the problem domain.

There exist a number of tools which generate the UML class diagram. Some of them enable to define several elements of class structure based on data presentation of the problem domain. Others generate a class diagram from existing source code, to display the structure of the developed system. However, the problem of automatic generation of the UML class diagram from the formal and still customer-friendly presentation of problem domain is not solved yet. Authors present the tool, called BrainTool [2] developed by researchers of the Riga Technical University, as a step forward in the area of automation of the modeling process. BrainTool supports generation of the UML class diagram from the so-called two-hemisphere model [3] of the problem domain, which presents information about processes, information flows between these processes and pre-defined types of these information flows (Figure 1). Present paper demonstrates the comparison results of the UML class diagram received by application of BrainTool for the simulation of football game to the UML class diagram generated from the software system developed for the same task using C++ programming language.

The paper is structured as follows. The next section describes the general functionality of BrainTool used for generation of the UML class diagram from the two-hemisphere model. Section 3 gives comparison results and approves the correspondence of the class diagram generated by BrainTool to the class diagram generated from the software solution. Several conclusions and directions for future research are stated in the fourth section.

APPLICATION OF BRAINTOOL FOR FOOTBALL GAME SIMULATION

The two-hemisphere model driven approach is based on the transformation of two interrelated models into the UML class diagram. These two initial interrelated models are: business process model (shortly – process model), which displays behaviour of the system and the model of conceptual classes (shortly – concept model), which displays a skeleton of system's static structure. The notational conventions of the two-hemisphere model are similar to ER diagrams and Data Flow Diagrams, wide used in software development starting from the end of 80-ies and still popular. The original idea offered by authors in 2004 [3] is to use both diagrams together, by assigning the data types for data flows from the concept model. The meaning of objects in an object-oriented philosophy gives a possibility to share responsibilities between objects based on the direct graph transformation, where the data flow outgoing from the internal process in the process model becomes an owner of this process for performing it as an operation in object communication and further is mapped into class responsible for this operation in class diagram.

Figure 1 demonstrates the two-hemisphere model created for the basic scenario of football game playing in BrainTool and import of generated UML class diagram into Sparx Enterprise Architect tool.

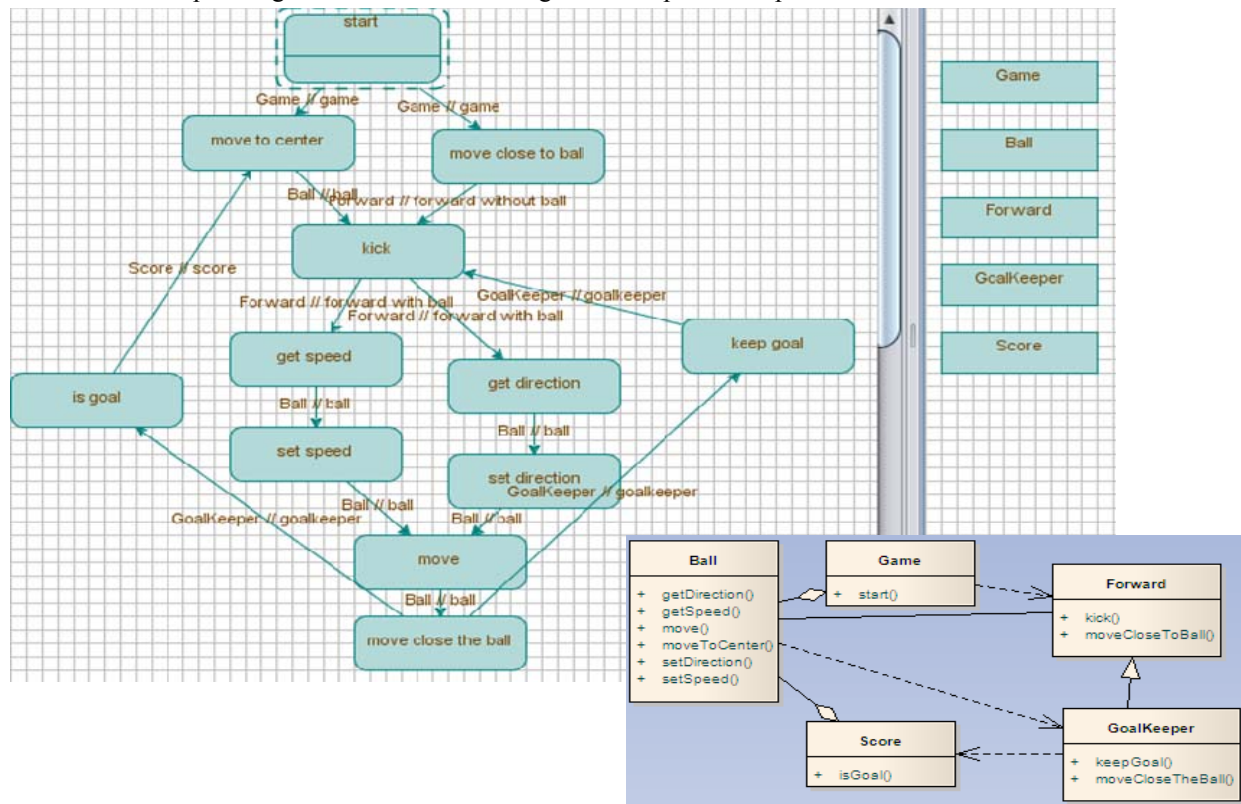


FIGURE 1. Two-hemisphere model developed by BrainTool and import of the generated UML class diagram for football game simulation into the Sparx Enterprise Architect tool

The current version of the implementation of the two-hemisphere model driven approach can be stated as a standalone tool titled as BrainTool in correspondence with the title of the approach, which in turn is derived from cognitive psychology by analogy with human brain consisting of two interrelated hemispheres. BrainTool gives a possibility to create the two-hemisphere model, to save it in the defined repository, to apply all the defined transformations for generation of the UML diagram and to export it in XMI format into some UML compatible tool. The papers giving explanations on different aspect for application of the two-hemisphere model are listed on tool's website [2].

EVALUATION OF THE GENERATED UML CLASS DIAGRAM

The class diagram for football game simulation shown in Figure 1 gives a possibility to approve the correspondence of the generated classes with the methods objects should perform to their practical usage for software system developments. In order to check this correspondence, authors have been generated the UML class diagram from the software solution developed in C++ for the same simulation of the football game. The generated classes and its specification is shown in Figure 2.

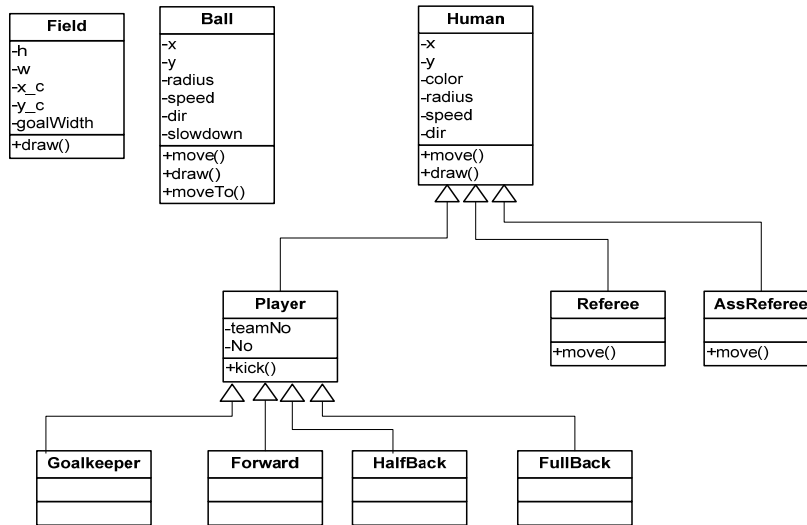


FIGURE 2. UML class diagram for football game simulation created manually during the development of the software system using general principles of object-oriented system development

To estimate the obtained result, these two models are mutually compared. Authors have been selected three comparison criteria describing the core set of the UML class diagram: (1) how many classes match in both models; (2) how many associations are identified, and (3) how many methods in each class are the same. The names that are in the obtained models may be different, so the comparison was done manually based on obtained elements' semantics. If the meaning of the different named elements is the same, in the comparison process these elements are used as the same elements.

As the result of the comparison, authors can claim, that BrainTool is able to generate quite good UML class diagram for further usage of it for software solution development. This means, that the resulting UML class diagram, produced by BrainTool, is quite close to data amount and data quality stored in the model of software solution developed in a set of iterations. The difference between diagrams is not critical and depends on the fact that only basic scenario was modelled and only for it the class diagram was generated by BrainTool. Authors suggest that further modeling of all the rules of the football game would give the complete structure of the classes. What is more, according to aggregations and dependencies BrainTool gives potentially more useful results as it is able to detect all the required associations.

Loniewski, Insfran and Abrahao in [4] describe the results of a survey about different approaches used for transformation of system requirements to system design and implementation. The survey shows the result of analysis of different approaches to transformation of the problem domain description into the UML class diagram during the last 10 years, published in four digital libraries (IEEEExplore, ACM, Science Direct, Springerlink). The survey states that there exist many approaches with different types of solutions for the generation of a UML class diagram. Moreover, the authors analyze the approach based on several criteria, one of them is tool support. Analysis of the automation level in these approaches shows that 25 out of 71 approaches described in corresponding papers are supported with a tool. However, [4] stress that these tools are academic tools and are not widely practically used as far as they are created to approve the automation level of the approach offered by their vendors. Therefore, authors can assume that two-hemisphere model driven approach supported by BrainTool is a step forward in Model Driven Engineering as an Engineering Science. Moreover, for the moment there is no equivalent tool for generation of UML diagrams, which would be wide used and convenient in software developer community.

CONCLUSION

Nowadays, the usage of model transformations has become a widespread practice and tools supporting such transformations have become increasingly popular. The main goal of the research presented in this paper was to demonstrate a tool, which can generate the UML class diagram from the initial presentation of problem domain and to export it to any UML compliant modeling environment supporting the XMI format for model interchange.

The main contributions of the research is a comparison of the generated UML class diagram to so called “As-is” diagram generated from “manually” developed software solution. It approves, that the generated diagram is not worse than the diagram created during the development project. At least for the only modeled basic scenario the generated structure has the same classes to support all the processes and conceptual classes defined in the problem domain. And the main benefit is that the structure is received automatically from problem domain presentation.

Secondary benefit of using the two-hemisphere model for the UML class diagram generation, is that modelers don't have to domesticate the new modeling notation. The two-hemisphere modeling is similar to time-tested ER and DFD diagrams just connected together. Therefore, BrainTool also can be positioned as a tool for developers, who don't like to draw UML diagrams but has a need to create them.

One more benefit of the two-hemisphere model is that it can be created and often already is created by the business analyst at the customer's side. A Standish group survey shows that about 83% of companies are engaged in business process improvement and redesign. This implies that many companies are very familiar with business process modeling techniques or at least they employ particular business process description frameworks [5], [6]. On the other hand, the practice of software development shows that functional requirements can be derived from the problem domain description as much as 7 times faster than if trying to elicit them directly from users [7]. Therefore, with minimal efforts created and intuitively understandable by customer two-hemisphere model can be used for automatically generating class diagram prototype that can be later reviewed and used in software development.

ACKNOWLEDGMENTS

The research presented in the paper is also partly supported by Grant of Latvian Council of Science No. 342/2012 "Development of Models and Methods Based on Distributed Artificial Intelligence, Knowledge Management and Advanced Web Technologies".

REFERENCES

1. OMG UML Unified Modeling Language Specification. Retrieved from: <http://www.omg.org>
2. Website of BrainTool. Available at braintool.rtu.lv/
3. O.Nikiforova and M. Kirikova. Two-Hemisphere Model Driven Approach: Engineering Based Software Development, *Conference on Advanced Information Systems Engineering*, 2004, pp. 219-233.
4. G. Loniewski, E. Insfran, and S. Abrahao, A Systematic Review of the Use of Requirements Techniques in Model-Driven Development, 13th Conference, MODELS 2010, Part II, Oslo, Norway, 2010, pp. 213—227.
5. P. Rittgen, Quality and perceived usefulness of process models, 25th Symposium On Applied Computing. 2010. ACM, pp. 65-72.
6. H. Peyret, Enterprise Strategy Represents A Compromise Among Three Dimensions, Not Two, Forester research. 2010
7. W. M. P. van der Aalst, H. M. W. Verbeek. Process Discovery and Conformance Checking Using Passages. *Fundam. Inform.* 131(1): 103-138, 2014.