ISSN No:-2456-2165

Modeling of the Authoring and Publishing Information System using xtUML

Denisa Starinská, Branislav Madoš
Department of Computers and Informatics,
Faculty of Electrical Engineering and Informatics, Technical University of Košice,
Košice, Slovak Republic

Abstract:- The paper deals with the problematics of scientific papers lifecycle and introduces model of integrated system for authoring and publishing of scientific papers using executable and translatable UML (xtUML), Object Action Language (OAL) and CASE system BridgePoint. The aim of integrated software solution for authoring and publishing of scientific papers is to simplify all processes that are connected with the scientific papers lifecycle e.g. formatting of the paper into templates and to prevent issues like bad citations of sources or confusions in names of authors. The model of an integrated information system for the authoring and publishing of scientific papers consists of a reference management tool linked to a typographic system LaTeX, supporting collaboration of multiple authors, along with the ability to generate quotes and bibliographies, also a tool for managing and publishing of created scientific papers with the possibility of the review process, a module for generating analyzes and statistics from published scientific papers. All of this unifies all processes related to scientific work, prevents mistakes and errors, delays and redundant work.

***Keywords**:- BridgePoint; LaTeX; OAL; Object Action Language; reference manager; Unified Modeling Language; UML; executable and translatable UML; xtUML.*

I. INTRODUCTION

An increasing number of professionals, not only from universities and research and development centers but also experts from different fields of industry, medicine, and others encounters the need to search for professional papers published in journals and conference proceedings, books and other various forms of professional literature that they use for creation of their own works. Nowadays, bibliographic resources that are spread in digital form are preferred because of the simple availability of information and possibility to find suitable sources literally using a few keywords in search engine and making a few clicks.

Many software systems have been developed to simplify work with resources that are then referenced in professional and scientific publications. These reference managers are primarily software products and on-line services that help the user to collect, organize and sort bibliographical sources. Many of them also have added features such as various statistics, export of documents to required formats, creation of lists of used literature according to a predefined templates and many others. Popular tools for the management of scientific knowledge, to reference only a few, include systems like Zotero [1], Mendeley [2], Colwiz [3] and EndNote [4].

In relation to these software tools, there are several comparisons of their properties and useful overview can be found in [5, 6].

Besides reference management software, there are many other tools used to create scientific papers. Writing and publishing is a complex process involving working with several software tools. In this process, tools such as the LaTeX typography system [7], the BibTeX reference management software for formatting lists of references [8], and conference management systems like EasyChair [9] and Confsys [10] are used.

There are abstract and citation databases of scientific journals, books and conference proceedings like Scopus [11] at the end of the professional peer reviewed publications lifecycle. Different scientometric tools are also used.

Usage of those separate tools brings problems with their interoperability and with the exchange of information that can be solved by integrating all the necessary software tools into a single entity, thus facilitating work related to the life cycle of authoring and publishing of scientific papers and other professional literature. That is why we are designing integrated tool that comprises all steps of the papers lifecycle and in this paper we are introducing modeling of this authoring and publishing tool using executable and translatable UML (xtUML), Object Action Language (OAL) and BridgePoint CASE tool.

II. PROBLEMS AND SHORTCOMINGS OF EXISTING SOLUTIONS

There are currently several problems with the use of software tools for creating and publishing of scientific papers with one simply addressable cause - fragmentation of scientific papers lifecycle into standalone and often incompatible software systems, which are not communicating to each other or are interchanging information by interfaces, but with incompatibilities and resulting inaccuracies and mistakes. Many times particular author of the paper sees his name and surname in different forms not only in different databases but also in one database, where name and surname can be altered, swapped, or because of the usage of full name vs. initials author can be registered in one database multiple times. There are many authors with the same initials and names and papers are assigned to the wrong authors. References in papers are often prepared with mistakes, misspellings and typos, with altered names of authors with missing diacritics and confused names and surnames of authors. Automatic scanning of papers is also the source of this type of confusions.

Another source of delays and extra work can be found in reformatting of the papers with the use of different templates, when changing target journals or conferences, because content of the paper is not separated from the form of presentation layout (template).

Exchange of information between different software systems can be based on standards e.g. BibTeX but sometimes it is based on extraction of information from digital files (pdf) or even printed versions of the papers, which is tedious and is the source of errors and delays. All of those problems can be solved in software system, which offers tools for all processes in all phases of the papers lifecycle.

III. POSSIBLE SOLUTION

Given the current problems and shortcomings, it would be advantageous to integrate existing tools into one unit to create a comprehensive system providing functionality assisting in the creation and publication of scientific work. Advantages would include, in particular, the removal of the shortcomings noted in the previous section, such as storing author's names in different databases and customizing work with different templates. In addition, the integrated system would facilitate passage through individual phases of the life cycle of a scientific paper and for example allows automatic database registration directly in the last phase of paper or conference proceedings publishing through the editorial board. Integrated system would also offer an option for application programming interface (API) for connection with other external solutions.

The aim of this paper is to design and model a complex program system to unify processes linked to the life cycle of creation and publication of scientific papers to make it easier for authors, editors and other stakeholders to save time and simplify the process. Model of an integrated information system for the creation and publication of scientific papers will consist of the reference resource management tool linked to a typographic system LaTeX with the possibility of collaboration of multiple authors, along with the ability to generate quotations and bibliographies. This system can be linked to existing online libraries and information database resources, what unifies a repository in which scientific work can be stored.

IV. XTUML, OAL AND BRIDGEPOINT

The BridgePoint Computer-Aided Software Engineering (CASE) tool (Fig. 1.) was used in the phase of the design and implementation of the model of designed authoring and publishing information system, to model the lifecycle of creating and publishing of the paper. BridgePoint comprises the Unified Modeling Language (UML) editor along with the Verifier, which is providing the possibility of execution of the models and different compilers to the target languages, which gives opportunity to translate models.

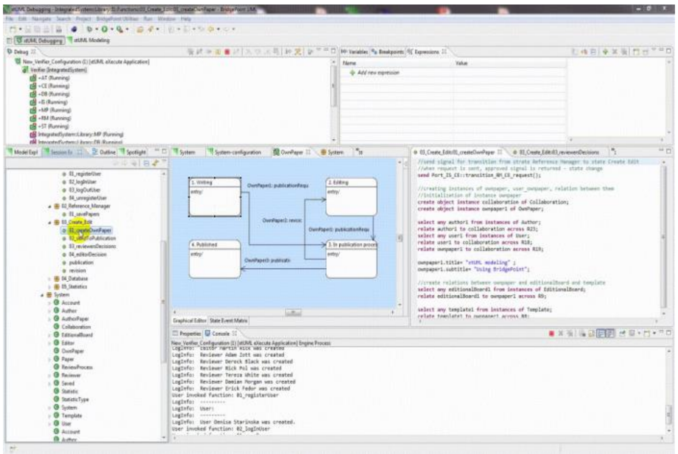

Fig 1:- BridgePoint CASE user interface.

Essential part of this tool is the Executable Translatable Unified Modeling Language (xtUML). It allows to model data, processing and control. Data and control are modeled using diagrams including component diagram or class diagram and control of the behavior is managed by using state machine diagrams. Processing is modeled using simple, abstract and model-aware Object Action Language (OAL) that can be inserted into the model as the source code. OAL is very similar to the target programming languages but is simpler.

Many methods and types of diagrams can be used to model systems using xtUML and OAL and for their correct selection, it is necessary to consider different factors such as the level of abstraction of the model of the system, its type or planned use. It is also possible to look at the model system from multiple perspectives, from external, which allows the system to see through the context and environment in which the system is located, from behavioral perspective, which displays behavior of a system or structural and architectural perspective of system or data. It is therefore possible to focus on particular aspects of the system, such as data in the system and their manipulation or modeling of system architecture as a complex. Usage of different types of diagrams can be advantageous in modeling of different systems, respectively one system from different perspectives.

V. METHODOLOGY

As the part of the research in the field we decided to establish and test the methodology of modeling the information system that comprises complex services which are integrated into one system. We found useful to go through the following six steps:

1. Component model - definition of integrated system as the reciprocal interconnection of components.
2. Sequence model - definition of interactions between actors and modeled information system respecting the timing of interactions.
3. Class model - definition of the class diagram with detailed information related to particular objects.
4. State machine model - implementation of finite state machines that are representing behavior and functionality of the information system.

5. Executable OAL code - Implementation of model dynamism using action language, to model processing within the information system.
6. Verification of the design – Last step is the verification of the design using Model Verifier from BridgePoint CASE tool.

VI. IMPLEMENTATION OF THE SYSTEM MODEL

The section of the paper deals with the implementation of the model of information system for authoring and publishing scientific papers linking all necessary software tools into one system. The section is divided into parts representing steps of gradual system modeling, from definition of components until verification of the complete model by executing it in the BridgePoint Verifier Tool.

A. Integrated system components

The first implemented layer is the component model. This is a set of components forming an integrated system, each providing different functionality. The individual modules of the system are interconnected using application interfaces serving to transmit data between them. The individual components of the system are:

- *IS Module (Information System)*- A central module which is linking all components together, using application interfaces for data transfer and allowing their interactions. Each connected component provides its interface for the integration of processes and data into the system. In total, it connects the remaining 6 components.
- *RM Module (Reference Manager)*- A component representing its own database of user information sources, respectively authors. Specifically, it is a storage collected knowledge to assist in the creation of scientific work, providing a readable organization of publications from scientific databases, and web sites serving this purpose. RM module in addition to the connection to the central system draws data from the official scientific database and provides data for creation of new works using the CE module.
- *CE module (Create & Edit)*- A part of the system designed for writing work in the LATEX typographic environment, allowing the separation of content of work from its form, tailored to a specific editorial board, enabling online collaboration of multiple authors, covering also generating the used literature using BibTeX. The CE component is, in addition to the central module, implemented with two interfaces serving for data communication with RM and MP components.
- *MP Module (Maintain & Publication)*- A module for maintain and publication of scientific papers, offers also functionality for managing conferences. Component includes implementation of the publishing and reviewing process and customizing the paper into the templates of the individual editorial boards. The MP component provides, in addition to the connection to the central module, another two application interfaces for data communication with other components. The first is to communicate with the database module DB, the second is data transfer interface between the MP module and the component CE.
- *DB module (Database)*- A database module representing the library that stores all published papers, allowing them to be shared, online access and conditional search. The database module is besides the connection to the central module interfaced with the application interfaces with three other components. The first is the RM module that provides its data needed for writing new work, the other is the MP module it provides the publication of new papers and the third is the ST module performing various data analyzes.
- *ST Module (Statistics)*- A tool that analyzes and processes all publications in the database to which it is linked, on which it is possible to draw up various statistics, e.g. number of citations of each work, etc. Module ST is implemented using an interface linked to the DB module, from which it draws the necessary data, for the production of statistics.
- *AT (Authentication)*- Authentication component for sensitive user data, stored in encrypted form on a remote server because of security reasons, linked from the central system using the application interface provided necessary data for authentication of registered users to system.

The component model contains also interactions with individual actors representing users of the integrated system, each interacting with another part of the system. Individual interactions and user roles in the system are:

- *Author*- Represents the user with the ability to interact with all components of the system, with each module is being able to do other activities. When writing a scientific paper, he collaborates with others authors, uses the database part of the system for searching the data for his scientific work, a RM module for storing and categorizing data obtained from the scientific database DB of published work, CE module for the creation of the paper itself and the MP components for its publication.
- *Editor*- Uses the MP module, primarily for provision templates given by the edition board to the author, and approval of the papers published in a specific editorial board. Its job is to decide which papers will be published, and thus has the ability to move the selected paper further to review process and then publish it.
- *Reviewer*- Uses MP to do reviews for potentially new scientific papers that will be published in the journal.
- *External system* - Represents other systems which they can retrieve data from an integrated system, e.g. saving published papers from the general database to its databases.

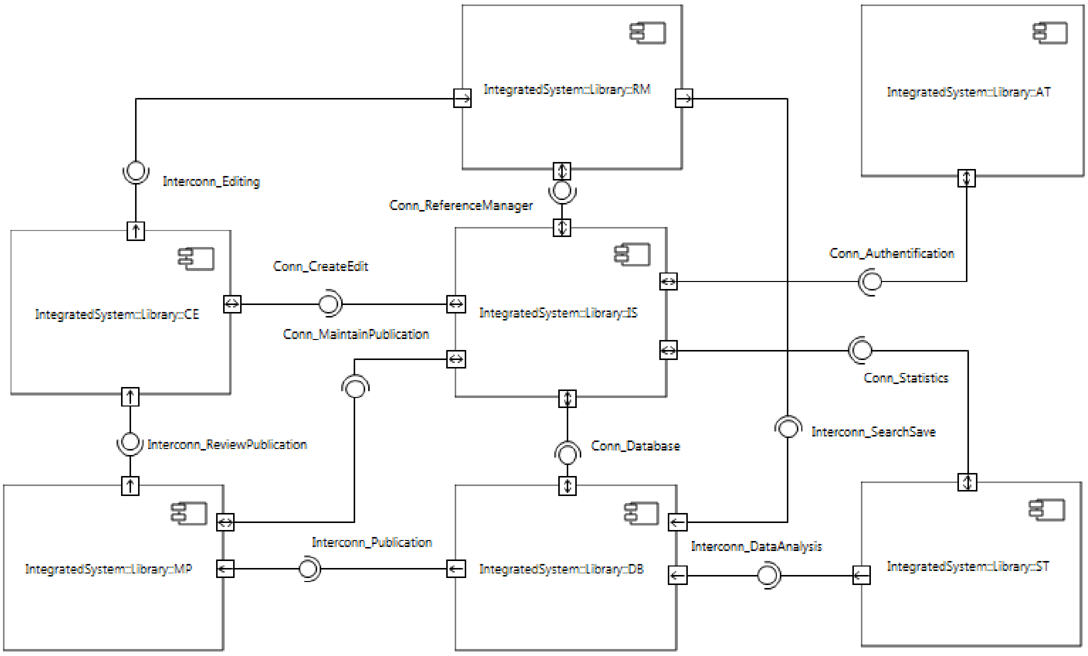

Fig 2:- Comprehensive configuration of components.

B. Comprehensive Configuration of Components

After creating the individual components and application interfaces, total system configuration model was implemented using created connections of application interfaces shown in Fig. 2.

C. Integrated System Components

The system model is implemented using two types of application interfaces. The first type are inter-component interconnections allowing data communication between individual components, the second ones are the connections to the central module, providing data for the overall system and processes associated with creation and the publication of papers. Each application interface of the system providing interconnection of the individual components of the integrated system is implemented through synchronous and asynchronous messages separated from implementation of components.

D. Finite State Machines

It is possible to implement finite-state machines to model the functionality of the system and verify its usability. Definition of finite-state machines is in individual classes since each state machine diagram represents changes in the status of a particular class (Fig. 3.).

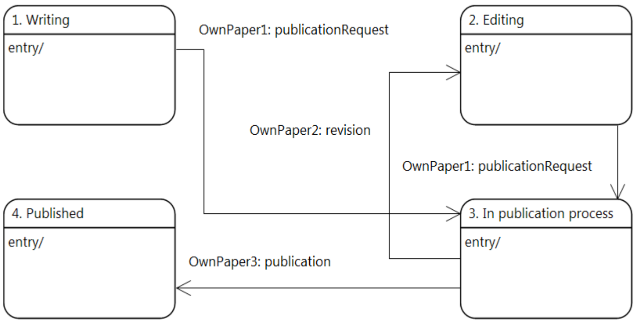

Fig 3:- State machine representing the progress of the paper creation.

The model of integrated system consists of two types of state machines. They are state machines based on classes or instances. Class state machines allow to define the static life cycle of a particular class and all its instances, as opposed to instances state machines allowing to model different states of individual instances at the same time. For the overall representation of the state of the system and currently used component is used class state machine that consists from seven states. Diagram of all states is implemented within the System class.

The initial state is when the system is not working. After running the model, the system is in this predefined inactive mode state. All other states are part of the system that are currently working. If user authenticates, the system is in the Authentication state, when saving papers to a custom database, the system goes into the Reference Manager state. When writing a new paper, also collaborating with other authors, the system is in the Create Edit state. Subsequently, when author submits a request for a paper publication to editorial board, the status will change to Maintain Publication. If it is necessary to edit the paper before it is published, it will go back to the Create Edit state, where the paper is edited, and a subsequent request for its publication is then made. At a successful publication, the system goes to the Database state and is published in public database from which it is possible to create various analyzes and statistics using the module specified when the system switches to Statistics.

E. System Usage Scenario

During the creation and publication, the paper goes through multiple phases. In the first phase, it is important to collect and categorize knowledge from the ones already created and published scientific papers relevant to the topic using the module DB. When writing a scientific paper through the CE system component it is also important part the collaboration between authors. After paper is written, if the author decides to publish the paper in a specific scientific journal, it must meet the form of a specific editorial board, so the paper needs to be matched to a matching template. After sending to editorial board, i.e. MP module, paper first goes through checking by the editor. The editor decides which paper to move further, there may also be a case of denial of work. Subsequently the paper is shifted to reviewers, who after reading, decide possible changes. Either a minor revision or a major revision, which is needed to recheck reviewers. Subsequently, the final check is complete and the paper is published. After publishing the journal, it is also registered in databases containing publications and can then serve as a source for other scientific work. The DB module then provides to the ST module all necessary data to produce various analyzes and statistics.

VII. SYSTEM MODEL VERIFICATION

Modeled system was verified based on the implementation of the OAL code representing the system functionality. Functions were created that sequentially triggered the lifecycle of the creation and publication of the paper. Examples are the initialization function creating the specific instances needed to work with the system, authenticating the user, storing papers in its own database, paper creation, publishing process, and process of analysis of published works. After triggering individual functions, it was possible to monitor the creation of individual entities and their interactions. Also, changes in finite state machines were tracked.

Process logic has been verified by simulation of created system performing through a BridgePoint Verifier tool. The execution of the model showed no unexpected outputs and behavior. Verification of the model has confirmed the efficiency of the processes by using the integrated system for creation and publication, which implies the possibility of implementation of the model in practice.

VIII. CONCLUSION

Benefits of the designed information system are in integration of standalone software tools that are used in different steps of the papers lifecycle. It allows easier and quicker manipulation with the paper and prevents occurrence of some mistakes and errors. System allows easier reformatting of the paper in case of changing template and quicker registration of papers in reference database which can be automatically performed when journal issue is published.

For the current complexity of processes related to the creation and publication of scientific papers a model of integration of several systems has been implemented providing functionalities needed for this process into one complex system offering currently the management of citation resources through a reference manager, collaboration of multiple authors in creating a new paper, simplifying customization of new paper, simplification of the publishing process by adding editors and reviewers as users of the system.

The program implementation of the integrated system model would allow simpler transition across the life stages of the creation and publication of the scientific paper which would be more transparent and more efficient. Possible ways to extend this work is to translate the modeled system into a specific implementation language that was not done in this work, because of the current higher abstraction of the system model. The complex system would therefore have the prerequisites to become used in real-life practice and to replace the various partial systems involved in the process.

IX. ACKNOWLEDGMENT

This work was supported by KEGA Agency of the Ministry of Education, Science, Research and Sport of the Slovak Republic under Grant No. 003TUKE-4/2017 ”Implementation of Modern Methods and Education Forms in the Area of Security of Information and Communication Technologies towards Requirements of Labour Market” and under Grant No. 077TUKE-4/2015 „Promoting the interconnection of Computer and Software Engineering using the KPIkit“ and by the Slovak Research. This support is very gratefully acknowledged.

REFERENCES

[1] A. K Mueen and B. Al Dhubaib, "Zotero: A bibliographic assistant to researcher", Journal of Pharmacology and Pharmacotherapeutics, 2(4), Oct-Dec, 2011, pp. 303-305, doi: 10.4103/0976-500X.85940.

[2] E. Mohammadi and M. Thelwall, "Mendeley readership altmetrics for the social sciences and humanities: Research evaluation and knowledge flows", Journal of the Association for Information Science and Technology, Volume 65, Issue 8, 12 March 2014, pp. 1627-1638.

[3] A. Hicks, “Review of “Colwiz””, Collaborative Librarianship, Volume 3, Issue 3, Article 10, 2011, ISSN: 1943-7528.

[4] F. A. Brahmi and C. Gall, "EndNote and Reference Manager Citation Formats Compared to “Instructions to Authors” in Top Medical Journals", Medical Reference Services Quarterly, Volume 25, Issue 2, 2006, pp. 49-57.

[5] R. Gilmour and L. Cobus-Kuo, "Reference management software: a comparative analysis of four products", Issues in Science and Technology Librarianship, 2011, pp. 63-75, DOI: 10.5062/F4Z60KZF.

[6] A. Butros and S. Taylor, "Managing information: evaluating and selecting citation management software, a look at EndNote, RefWorks, Mendeley and Zotero", Proceedings of the conference: International Association of Aquatic and Marine Science Libraries and Information Centers (IAMSLIC), January 2011.

[7] A. J. Bennieston, "Writing Scientific Documents Using LaTeX", Fifth edition, February 19, 2009.

[8] J. Fenn, "Managing citations and your bibliography with BibTeX", The PracTEX Journal, 2006, No. 4, ISSN: 1556-6994.

[9] Easychair, online: https://easychair.org/, cited: 25.6.2018

[10] M. Huang, Y. Feng and B. C. Desai, "CONFSYS: a web-based academic conference management system", Proceedings of the 2008 C^3S^2E conference, Montreal, Quebec, Canada, May 12-13, 2008, pp. 141-143, ISBN: 978-1-60558-101-9.

[11] Scopus, online: https://www.scopus.com/home.uri, cited: 05.6.2018.