IOP Conf. Series: Materials Science and Engineering **327** (2018) 022042 doi:10.1088/1757-899X/327/2/022042

Means of storage and automated monitoring of versions of text technical documentation

S A Leonovets[1,2], **A V Shukalov**[1,2], **I O Zharinov**[1,2]

[1] St. Petersburg National Research University of Information Technologies, Mechanics, and Optics (ITMO University), 49 Kronverksky Pr., St. Petersburg, 197101 Russia
[2] Electroavtomatika St. Petersburg Scientific Design Bureau named after P.A. Efimov, 40 Marshala Govorova St., St. Petersburg, 198095 Russia

E-mail: ser2694@ya.ru

Abstract. The paper presents automation of the process of preparation, storage and monitoring of version control of a text designer, and program documentation by means of the specialized software is considered. Automation of preparation of documentation is based on processing of the engineering data which are contained in the specifications and technical documentation or in the specification. Data handling assumes existence of strictly structured electronic documents prepared in widespread formats according to templates on the basis of industry standards and generation by an automated method of the program or designer text document. Further life cycle of the document and engineering data entering it are controlled. At each stage of life cycle, archive data storage is carried out. Studies of high-speed performance of use of different widespread document formats in case of automated monitoring and storage are given. The new developed software and the work benches available to the developer of the instrumental equipment are described.

1. Introduction

The evelopment process of the on-board equipment is followed by preparation of the designer and program documentation at different design stages: sketch design; engineering design; technical sentences; design designer engineering. The list of the developed documents is regulated by the appropriate state standards and is defined by system complexity. This technical documentation has the life cycle throughout which it is necessary to track the status of the engineering data used in it. At the disposal of developers of the project organizations there are different softwares helping to exercise control of versions of documents [1-8]. But they do not consider specifics of operating state standards.

Therefore the purpose of operation was to write the software, which would give users an opportunity of automated preparation of technical documentation, and also to exercise control of versions.

2. Life cycle of technical documentation

For solution of this task, there was an environment of Microsoft Visual Studio. The program is written in the C# language with WPF use. The control system of versions which exercises constant control of data of the document during the entire life cycle, submitted in figure 1, is CAD part [9,10].

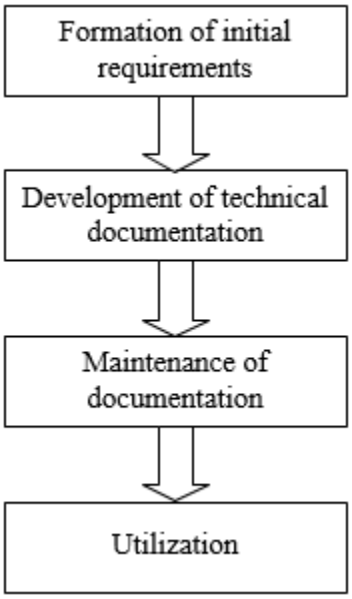

Figure 1. Life cycle of technical documentation.

During creation of the document, it automatically passes into a status "draft" with version 1.0. This status is used for editing. Upon termination of editing, the version of the document is exposed to the formal inspection for what "proposed" is transferred to a status. From this point, editing is forbidden. By the results of check, the version of the document either becomes the new basis version of the configuration unit (CU), passing into a status of "approved", or is sent back for revision, passing into a status "declined". This process is shown in figure 2.

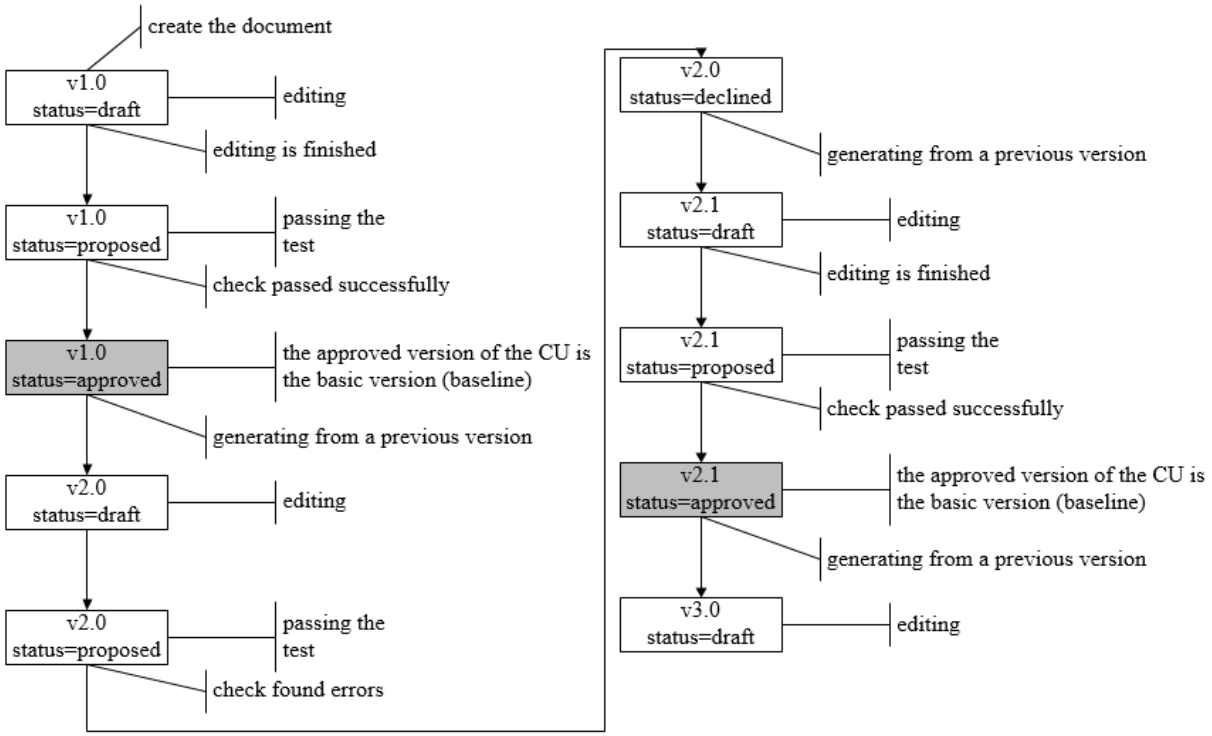

Figure 2. "Evolution" of the document.

In both cases the version of the document is fixed for history and its editing is not allowed. If it is

necessary to make changes, the result of upcoming version of the document is executed. At the same time, depending on a status of the latest version of the document (approved or declined) assignment, the version number is for the generated version of the document.

All versions of the document remain. The possibility of viewing any, selected by the user, versions of the document and also a possibility of comparing two any versions of one document are provided.

3. Research of speed of saving documents

In terms of the object-oriented approach, the class which objects will be used for storage of entities is delivered each object in compliance. In figure 3, the fragment of the developed formal information model in the form of the UML class diagram illustrating composition and an object hierarchy of design, their main attributes and operations in the context of control of life cycle of technical documentation is shown.

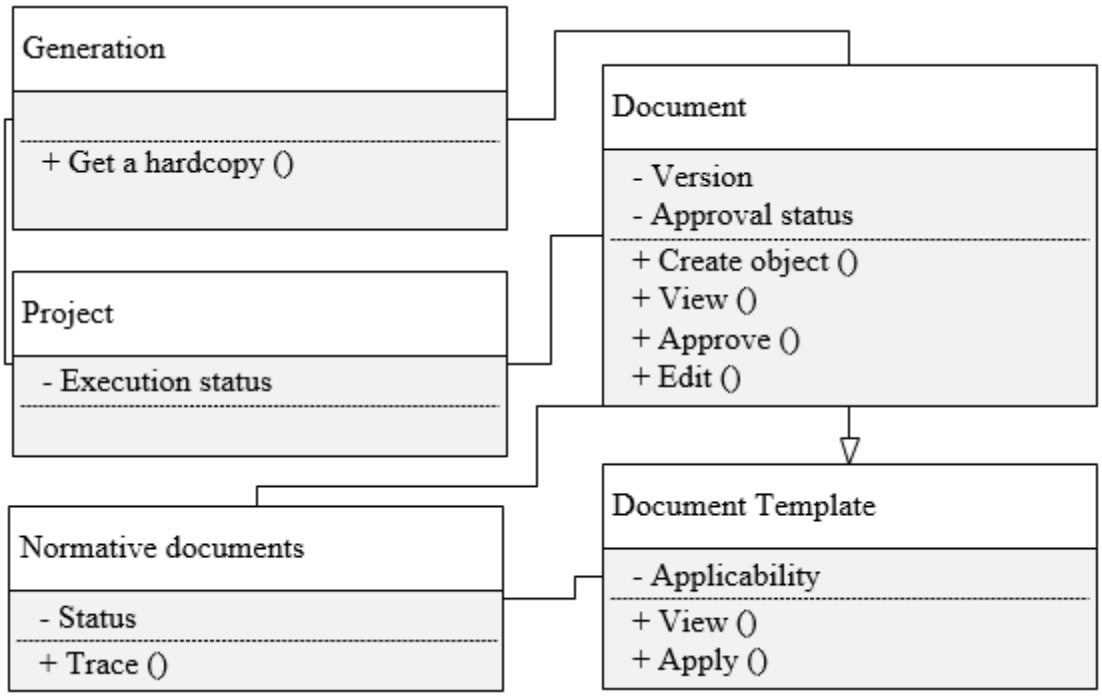

Figure 3. UML class diagram of subjects to design.

The group of several versions of different documents connected among themselves is carried out by means of configurations. For example, the configuration may contain those versions of each of documents of the project which entered release of a software product. Another example of a configuration is the combination of versions of documents at the time of completion of a certain stage of project lifecycle.

The configuration has no versions and is special EC with the simplified quantity of possible statuses: a “draft” copy, it is also “approved”.

Possible statuses of a configuration and transitions are provided in figure 4.

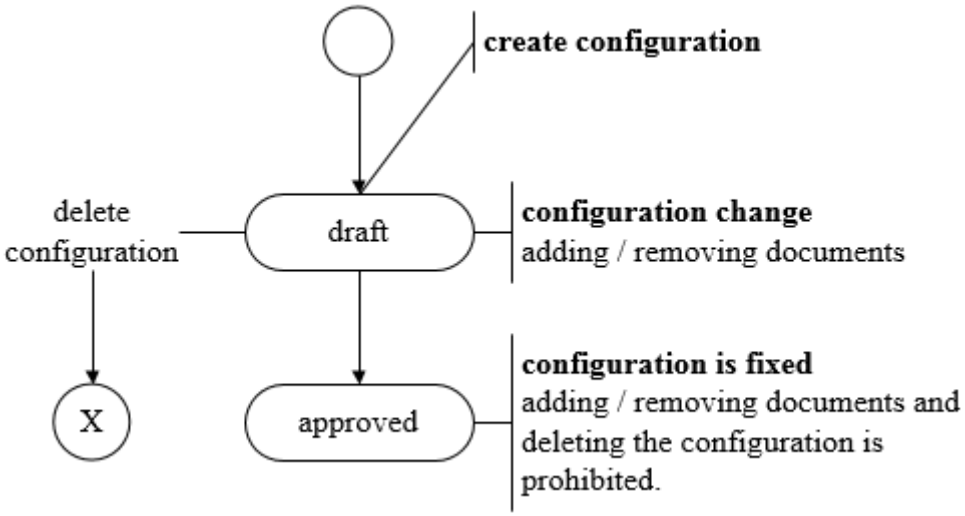

Figure 4. Configuration status.

The research of speed of saving documents in different formats with the use of Microsoft.Office.Interop.Word.dll library of version 14.0, as well as the C# language, was conducted. Testing of saving speed was made by sampling of runtime of the following code:

```
[1] Microsoft.Office.Interop.Word._Application appWord = new Microsoft.Office.Interop.Word.Application();
[2] // Заполнение документа контентом
[3] CreateAndFillActiveDoc(appWord);
[4] object fileName;
[5] object FileFormat =
[6] Word.WdSaveFormat.wdFormatRTF;
[7]     appWord.ActiveDocument.SaveAs(ref fileName, ref FileFormat);
    appWord.Quit();
```

Extension of the saved document is set by the FileFormat variable. The RTF format is selected from the example given above. Also the following formats were probed: PDF, XPS, DOC, DOCX, ODT, MHTML. The data were retained as follows: 3 pages A4, with a table, an image and a text. Results of the experiment are provided in figure 5.

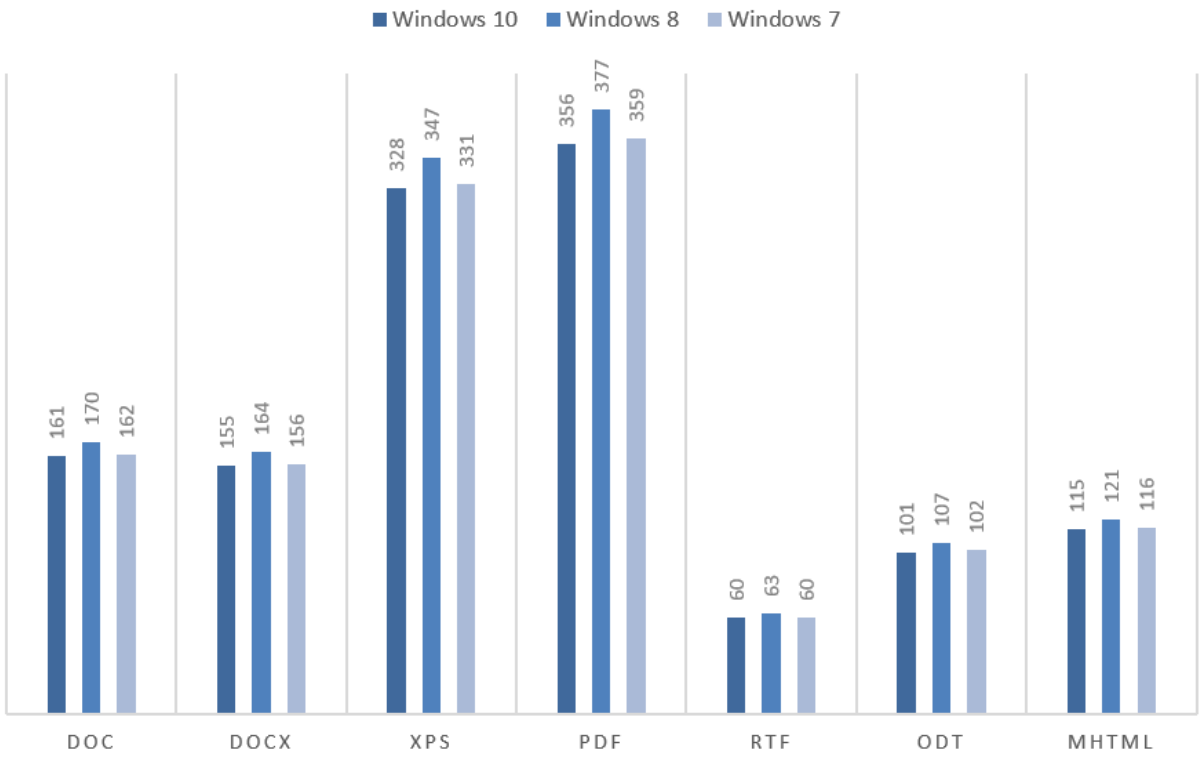

Figure 5. Time of saving data in case of different document formats in milliseconds.

Proceeding from the received results, the conclusion was drawn that it is expedient to save versions of each document in the RTF format, to allow one to spare the user's time when saving large volumes of data.

4. Conclusion

The authors offered a new component of the system of automatic generation of the designer and program documents – the subsystem of monitoring of versions providing the continuity of monitoring of life cycle of engineering data for the purpose of improvement of quality of documents. The user can trace the document at any moment who made certain changes and when, and can also return to the last version changed.

The received results can be used by developers of the project organizations which draw up technical documentation. Approbation of the work of this system can be performed with use of the standards existing in instrument making. Automation of the process of preparation of designer documentation significantly reduces labor input and design time that finally reduces prime cost of a development stage of a product in general.

References

[1] Rodriguez J, Thomas T N 2017 *ASEE Annual Conference and Exposition, Conference Proceedings*
[2] Sládek A, Pastirčák R, Brůna M, Remišová A 2017 *Manufacturing Technology* **17(5)** 842-847
[3] Fiorentino M, Uva A E, Monno G, Radkowski R 2016 *International Journal of Computer Aided Engineering and Technology* **8(1-2)** 56-79
[4] Alanen J, Vidberg I, Nikula H, Papakonstantinou N, Pirttioja T, Sierla S 2011 *VTT Tiedotteita - Valtion Teknillinen Tutkimuskeskus* **(2583)** 1-137
[5] Främling K, Rabe L 2006 *IFAC Proceedings Volumes (IFAC-PapersOnline)* **12(PART 1)**
[6] Simon M, Bee G, Moore P, Pu J-S, Xie C 2001 *Computers in Industry* **45(2)** 111-122
[7] Pfouga A, Stjepandić J 2015 *Advances in Transdisciplinary Engineering* **2** 575-584

[8] Mohr P, Kerbl B, Donoser M, Schmalstieg D, Kalkofen D 2015 *Conference on Human Factors in Computing Systems – Proceedings* 2015-April 3337-3346
[9] Marjanowski J, Tomasiewicz J, Fraczek W 2017 *Aircraft Engineering and Aerospace Technology* **89(4)** 579-589
[10] Ciecinski P, Pieniazek J, Swiech L 2014 *2014 IEEE International Workshop on Metrology for Aerospace* MetroAeroSpace 2014 - Proceedings 238-243