Design of MBD Information Management Module Integrated with PLM Client Based on the UG/NX Integrated Environment

Kai Wang [1,4], Pengfei Zeng [2,4], Chunjing Shi [3,4], Weiping Shao [3,4] and Yongping Hao [4]

[1] School of Mechanical Engineering, Shenyang Ligong University, jkjkwk@163.com
[2] School of Mechanical Engineering, Shenyang Ligong University, pfzeng@163.com
[3] School of Mechanical Engineering, Shenyang Ligong University
[4] Key Laboratory of Advanced Manufacturing Technology and Equipment of Liaoning Province

Corresponding author: Pengfei Zeng, pfzeng@163.com

Abstract. Aiming at the application of model-based definition (MBD) in each link of product lifecycle, this study developed a MBD information management module of product lifecycle management (PLM) client in the Siemens UG/NX integration environment by utilizing secondary development technology of the UG/NX, and implemented a tight integration between the UG/NX and generic PLM system. As all functions of the module is implemented in the external mode of the NX Open, they can run completely independent of the UG/NX environment. In actual application, users are allowed to operate the MBD information of product directly through the PLM client without starting the UG/NX. This paper focus on describing the MBD information extraction process and integration process of the module, also shows an integration example of the module and evaluates the optimization effect by comparing the module function process with the previous management process of the MBD design information Based on three practical functional application scenarios. At the end of the paper, the advantages and limitations of the module are summarized, and the next stage of research is prospected.

Keywords: integration of CAD and PLM, MBD information, UG/NX Secondary Development, external mode of NX Open
DOI: https://doi.org/10.14733/cadaps.2023.614-627

1 INTRODUCTION

Model-based definition (MBD) is an information processing method that uses 3D models to express complete product information and realize paperless product design. It makes it easy for personnel in all production links to understand design intention, improves the efficiency of collaborative work and shortens the production cycle of products. At the same time, it exercises a great influence on the processing design, authorization and workshop production [1-4]. The MBD changes the expression of traditional product design information and product manufacturing information, enriches the information carried by 3D models to a large extent, but it also brings great pressure

© 2023 CAD Solutions, LLC, http://www.cad-journal.net

of data management. Product lifecycle management (PLM) system takes product data management (PDM) system as the core, integrates supply chain management (SCM), customer relationship management (CRM), computer aided process planning (CAPP) and other complex systems as well as computer aided design (CAD) and computer aided manufacturing (CAM) software [5-7]. For that matter, the PLM system has obvious advantages in management of the MBD information [8]. At the same time, the MBD information has been widely used in in various sub-integrated systems of the PLM system as source of data on their functions and business.

Due to the MBD uses the same model to express the peculiarities of different formats of information, the application of MBD is beneficial to improve the overall efficiency of the PLM [9]. In an integrated environment of the CAD and the PLM, the designer completes the definition of the MBD design information by attaching the information to the 3D model of product in the CAD environment. While in other links of the product lifecycle, the MBD design information will be extracted and reused according to business features to achieve business goals [10]. The application of MBD can effectively save the time cost of product design and manufacturing [11]. However, the introduction of the MBD into the PLM systems and effective utilization of the MBD also depend on a tight integration of the CAD and the PLM and a certain degree of information processing technology [12].

Based on the purpose of exploring the method to realize the tight integration between the CAD and the PLM system and optimizing the traditional management processes for the MBD information, this work designs a MBD information management module of PLM client based on client-server (C/S) architecture in the integrated environment of the UG/NX and generic PLM system. The core function of the module is to implement the extraction and integration of the MBD design information by using secondary development technology of the UG/NX and other computer technology, create the MBD design model and format the MBD information as XML to support the information transmission between the PLM client and the PLM server. After the MBD information is received by the server, it can be stored by the database system of the PLM, providing the necessary data support for the other production links and facilitating the collaborative work between personnel with different functions. Compared to the general application of the UG/NX integration technology, the MBD information management module can be completely separated from the UG/NX environment, and users can directly use the function of the module through the interaction with the PLM client.

2 LITERATURE REVIEW

This paper mainly discusses the process of the MBD information management based on the UG/NX and generic PLM system integrated environment. The relevant references are reviewed in the following two subsections.

2.1 Attachment Of the MBD Design Information in CAD Environment

The MBD design information can be divided into the geometric information and the non-geometric information. Geometric information includes the 3D model composed of geometric elements and expressed graphically as well as its assembly information. Non-geometric information includes the attribute information and the annotations attached to the 3D model, which include design dimension, roughness, geometric dimensioning and tolerancing (GD&T) as well as text annotation [13-15].

At present, the mainstream CAD software can support the attachment of the MBD design information. For example, the UG/NX, in addition to the original function of the MBD geometric information editing, also allows users to define the MBD non-geometric information. In the process of design, user can attach the attribute information by dedicated attribute modules of the UG/NX, and attach other geometry information as the product manufacturing information (PMI) annotation by the PMI tools of the UG/NX [16,17]. For the 3D model with the MBD design information

© 2023 CAD Solutions, LLC, http://www.cad-journal.net

attached, its MBD information can be extracted by means of secondary development of corresponding CAD software.

2.2 Application Process of the MBD Model in Product Lifecycle

In the lifecycle of product, the definition of the MBD information model of starts from the product design link. Firstly, the MBD design information is attach to the 3D model of the product by the CAD unit of the PLM to create the MBD design model. The CAPP system designs the processing of products based on design dimension, GD&T, roughness and other MBD information of the MBD design model, and then creates the MBD process model of the product by attaching the processing information to the 3D model [18-20]. In the process of inspection, according to the works mentioned in the literature [21], researchers extract the process information of the process model, and then get the inspection information by analyzing and processing the information extracted, finally complete the creation of the MBD inspection model. In addition, the MBD inspection model is used as the source data of the digital closed-loop control system based on MBD to realize the digital inspection of product quality. In the upstream of the product design link, SCM, CRM and other subsystems of the PLM use the lightweight model of products and the MBD design information to realize the redrawing of the product design models in the Web environment [22], which facilitates the transmission and communication of the product related information between enterprises and supply chains and between enterprises and customers.

3 IMPLEMENTATION PROCESS OF THE MODULE

3.1 Development Environment for the Module

The development environment of the MBD information management module is shown in Table 1.

Category	*Name*
Operating System	Microsoft Windows 10
CAD Software	Siemens UG/NX 8.5
IDE	Microsoft Visual Studio 2010
Language	C++
Library	NX Open C/C++
	MFC
	Win32 API
	tinyxml2

Table 1: The development environment for the module.

3.2 Core Support Technologies

3.2.1 Integration technology based on NX Open

As things stand, NX Open is the most commonly used application programming interface (API) library for the secondary development of the UG/NX. In order to adapt to the current mainstream computer languages, NX Open includes NX Open C (also known as UG Open or UFUN), NX Open C++, NX Open Java and Other subtypes of library [23]. Theoretically, NX Open C/C++ have the most comprehensive API, so they were used to implement the MBD information management module. Besides, NX Open defines two API invocation modes, internal and external. In the internal mode, the code is compiled into a dynamic link library (DLL), which needs to be loaded in the UG/NX environment when applied, and then functions of the DLL are executed. In the external mode, the code is compiled into an executable program with the ".exe" extension that can be accessed directly by user. Since the programs implemented in the external mode do not depend on

© 2023 CAD Solutions, LLC, http://www.cad-journal.net

the UG/NX environment, the development in the external mode can meet the requirements of the module for the running environment.

After the PLM client drives the MBD information module, the functional integration between the UG/NX and the PLM is implemented in the way that the API of the module was invoked by the PLM client. In practical application scenarios, users can use the functions of the module by interacting with the PLM client. And during the running of the module functions, users do not need to start the UG/NX or do other additional operations.

3.2.2 Integration of the MBD design Information

When extracting the MBD design information, it can be extracted by invoking the corresponding NXOpen C/C++ API according to the type of the information. For example, the MBD non-geometric information can be extracted by invoking the API defined by uf_Attr, uf_drf header files of NX Open C and the member functions of pmifeaturecontrolframebuilder, surfacefinishbuilder, datumfeaturesymbolbuilder and other classes defined by NX Open C++. Some MBD geometric information can be extracted by invoking the API defined by uf_assem header file and the member functions of Components, ComponentConstraints and other classes.

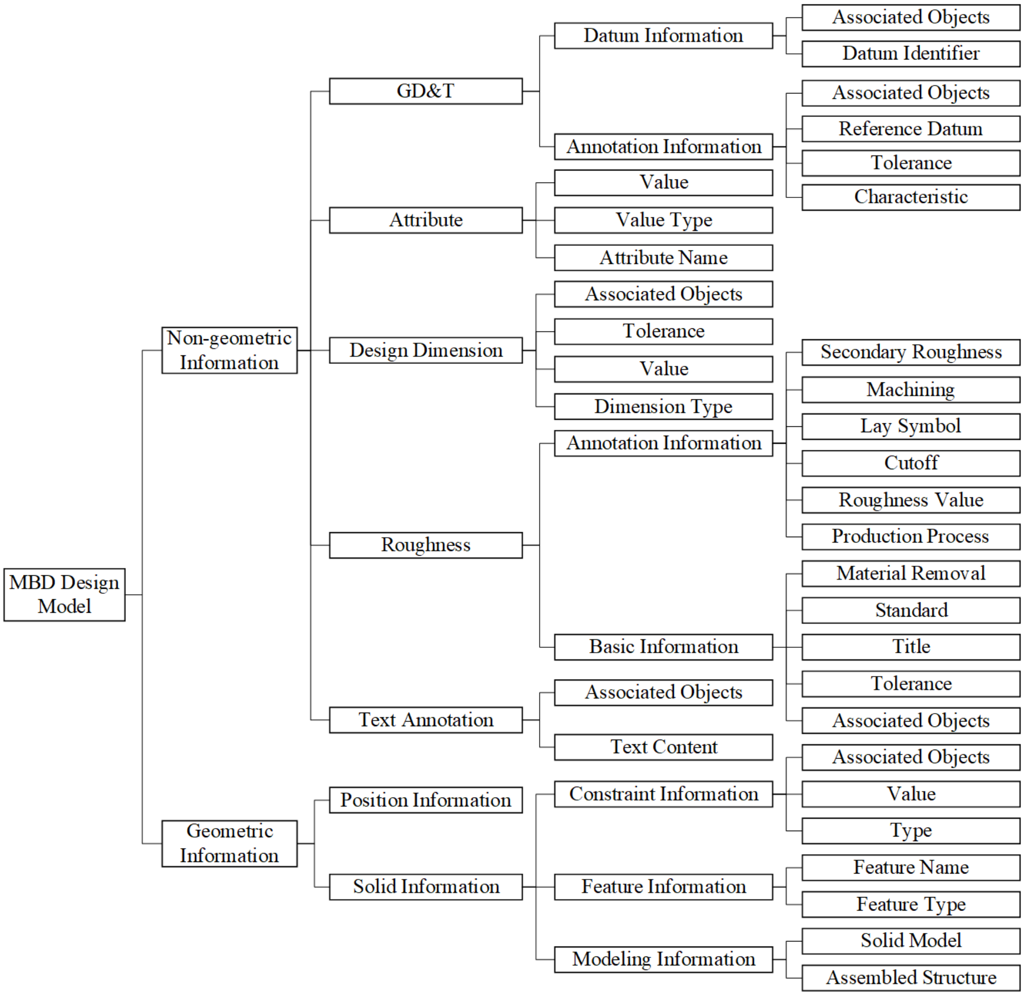

Figure 1: The MBD design model.

© 2023 CAD Solutions, LLC, http://www.cad-journal.net

However, the phenomenon of various API being invoked leaves the extracted MBD information in a discrete state, which is not conducive to the management and utilization of the information. To solve this problem, we first defined a information model according to the types of the MBD information that can be extracted by the MBD information management module, as shown in Fig.1. Then, based on this MBD design model, a data structure as shown in Fig.2 is designed to integrate and standardize the MBD design information extracted by the module.

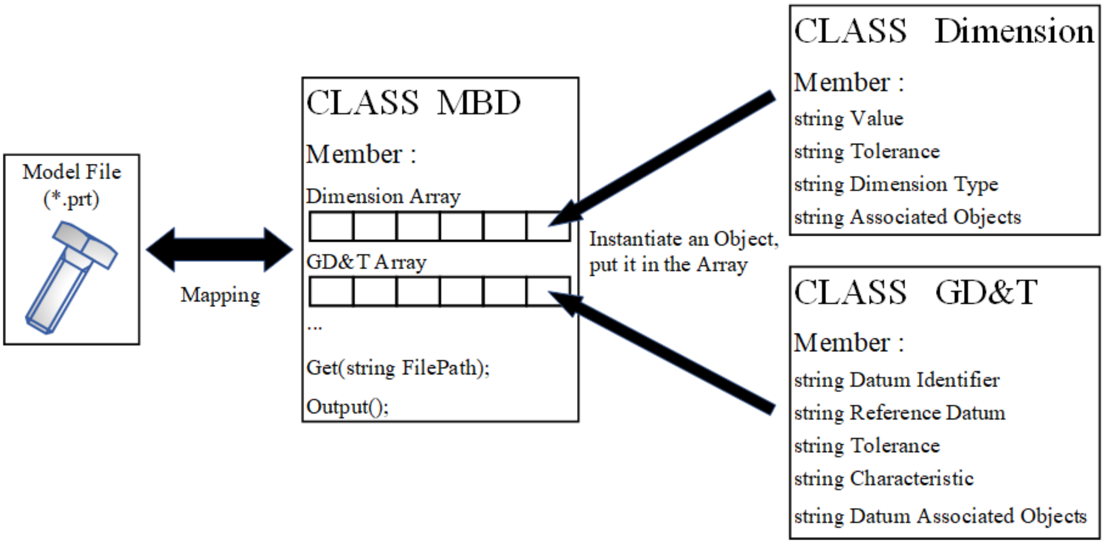

Figure 2: The data structure defined.

According to the idea of “object-oriented”, firstly several classes are defined, each corresponding to a type of the MBD information and taking the contents of the MBD information as its members. Then the “MBD” class is defined, whose members include several arrays with the instantiated object of the corresponding information classes as elements and two member functions named “Get” and “Output”. All API for extracting MBD design information is encapsulated in the “Get” function, whose tasks include establishing the mapping relationship between the design model and an instantiated object of the “MBD” class based on the storage path of the model file, extracting the MBD design information and assigning values to all arrays of the “MBD” object. After the "Get" function is successfully invoked, the MBD object can be used to express all the MBD design information. In other words, the integration of the MBD design information is completed. The integration process of module to MBD design information is shown in Fig.3. The “Output” function is used to send the MBD information that has been integrated to the PLM client.

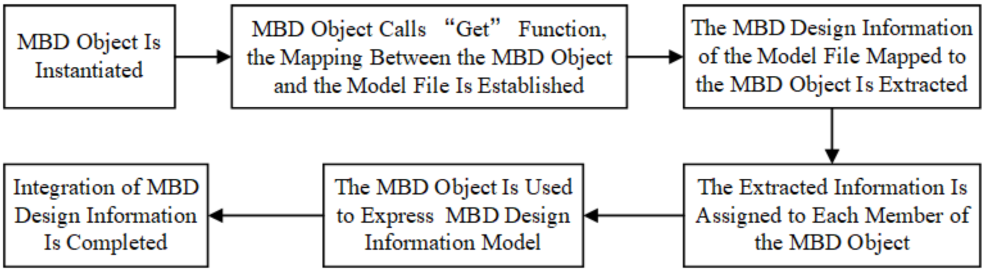

Figure 3: The integration process of the MBD design information.

© 2023 CAD Solutions, LLC, http://www.cad-journal.net

3.3 Implement Process of Core Functions

3.3.1 Construction of module framework

In the section3.2, the modeling method of MBD information based on the idea of "object-oriented" is described. However, the information transfer between the MBD information management module and the PLM client also needs to be implement in the integration process. In this work, the "memory-mapped file" is used to create a shared memory between the module and the PLM client as a communication channel between the two. The process of obtaining the MBD design model by the PLM client in the integration environment is shown in Fig.4.

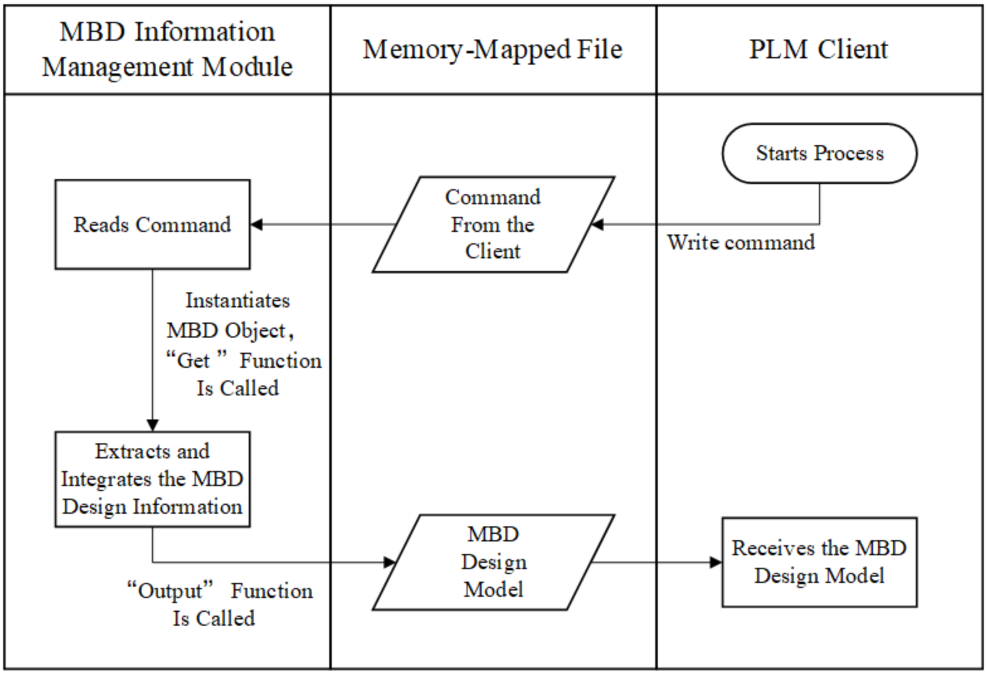

Figure 4: The MBD design information obtaining process of the PLM client.

Based on the "memory-mapped file" application, the MBD information management module can continuously and repeatedly provide full MBD design information during the PLM client run without generating any additional physical files.

3.3.2 Extraction of Non-Geometric Information

The MBD non-geometric information of the product includes attributes, design dimension, GD&T, roughness and text annotations. As for the extraction of attribute information, NX Open C/C++ have defined dedicated API, which can be directly invoked by the MBD information management module, so there is no need for additional discussion.

According to the section 2.1, in addition to attributes, the UG/NX uses PMI annotations attached by PMI tool to express other MBD non-geometric information, and it regards each annotation on the 3D model as an "object". Each type of the annotation object has corresponding type ID and subtype ID. In order to avoid confusion, the UG/NX provides a macro definition for each ID. For example, the type ID of the GD&T annotation is defined as UF_drafting_entity_type, and its subtype ID is defined as UF_draft_fpt_subtype. When extracting the MBD Non-geometric information, it is necessary to traverse all the annotation objects on the 3D model, at the same

© 2023 CAD Solutions, LLC, http://www.cad-journal.net

time use the type ID and the subtype ID to determine which type each annotation object belongs to and obtain the "tag" (a type of object identifier of the UG/NX) of each annotation object. The tags obtained are then passed to corresponding NX Open API as parameters to extract the detailed information of the annotation. The extraction process of the MBD non-geometric information is implemented by the "Get" function of the "MBD" class, and it is integrated after the extraction is completed. The process of extracting MBD non-geometric information is shown in Fig.5.

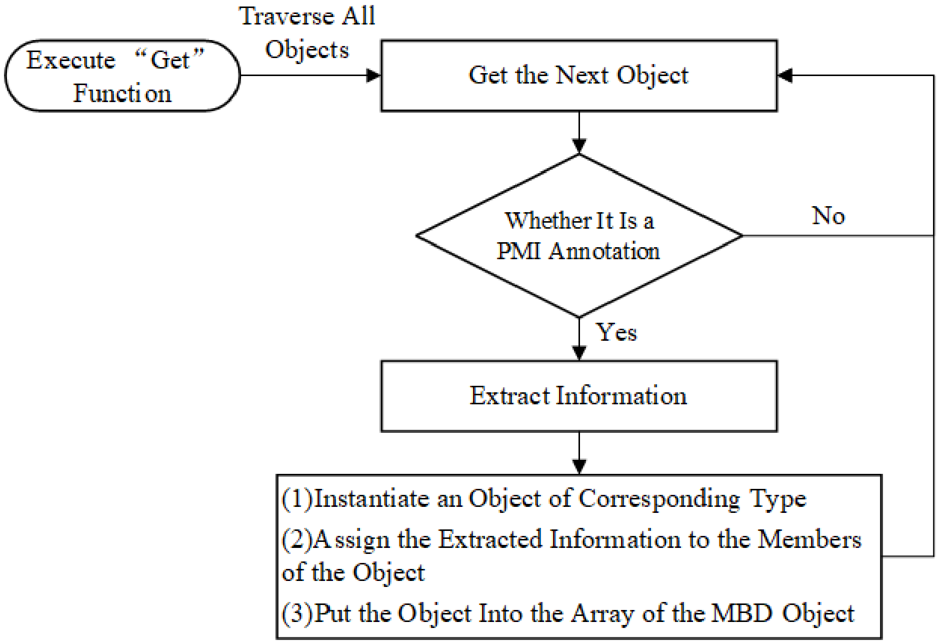

Figure 5: The extraction process of the MBD non-geometric information.

3.3.3 Extraction of Geometric Information

The extraction of the MBD geometric information is mainly aims at the 3D model of assembly. The information includes assembly structure, constraints on each component and assembly position coordinates of each component. The UG/NX typically uses an assembly tree to express the assembly structure, each node of the tree corresponds to a component. For that reason, all nodes of the tree should be traversed when extracting the MBD geometric information.

Since the assembly nodes of the model are distributed in a tree structure, the general traversal method is not suitable for reading the complete structure of multilevel complex assembly, so "recursion" algorithm is used to extract the MBD geometric information. The recursion requires that the function invoke itself inside its function body, so in each iteration of the loop, the parameters passed to the function and the return value are different [24,25]. The process of traversing the assembly tree with recursion just meets two basic rules of the recursion, "base case" and "making progress". During the recursion, if a node does not have any child nods, which can be regarded as the "base case", the traversal of this node is skipped. Conversely, if a node has child nodes, this situation requires "making progress", which includes three steps: traversing all child nodes of the node, extracting the MBD geometric information of each child node and finally doing the same traversal process for each child node until the "basic case" occurs. In this work, using the recursion algorithm to implement the traversal of all assembly nodes of the complex assembly and the extraction of the MBD geometric information. The extraction of MBD

© 2023 CAD Solutions, LLC, http://www.cad-journal.net

geometric information is also finished by the "Get" function of the "MBD" class. The process is shown in Fig.6.

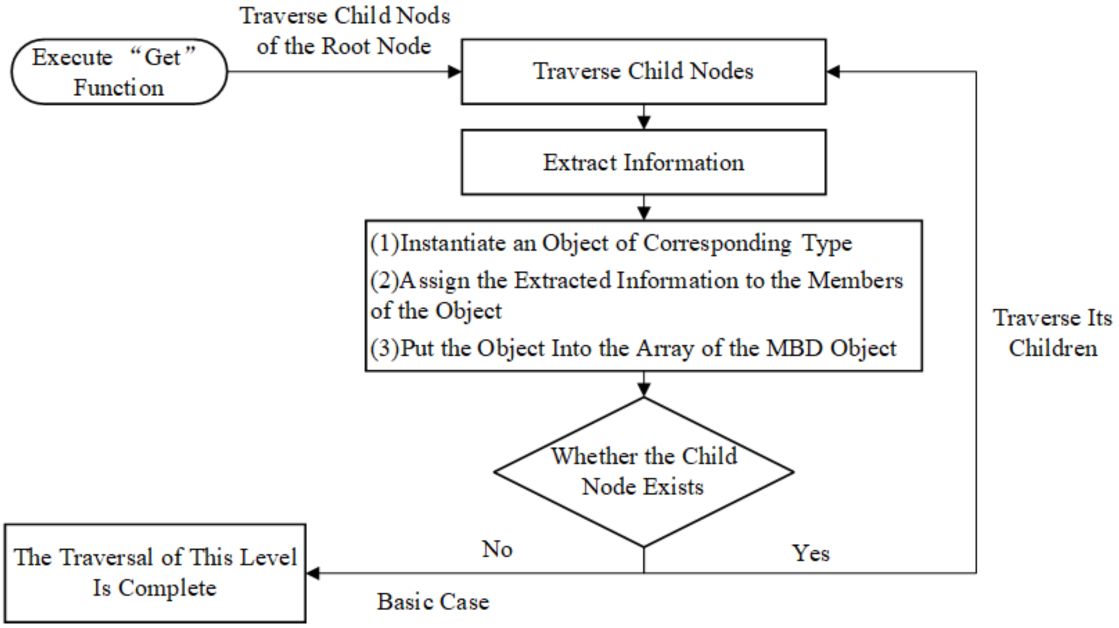

Figure 6: The extraction process of the MBD geometric information.

4 INTEGRATION EXAMPLE AND ANALYSIS

This chapter mainly presents an example of the integration between the MBD information management module and a PLM system developed by our team, shows some interfaces of the module in the PLM client environment and analyzes the optimization effect of the module in practical application scenarios.

The framework of the PLM client is implemented by using the Microsoft Foundation Classes (MFC) and run in the Microsoft Windows environment. The PLM server is developed and run in the CentOS environment. The PLM database system has made necessary adjustments.

4.1 Main Page of the MBD Information Management Module

User can access the main page of the MBD information management module by interacting with the PLM client. The MBD design information extracted is displayed in different sub-pages according to its type. Due to the JT2Go browser is integrated into the page, the user is allowed to check the JT format model corresponding to the design model. The style of the main page takes the structure/constraint information page shown in Fig.7 and the dimension information page shown in Fig.8 as examples. In the Fig.7, the tree control is used to express the structure of assembly, the user can extract and query the MBD design information of the corresponding component of by selecting the node of the tree. In the Fig.8, the list control is used to display the dimension information of the MBD design model.

© 2023 CAD Solutions, LLC, http://www.cad-journal.net

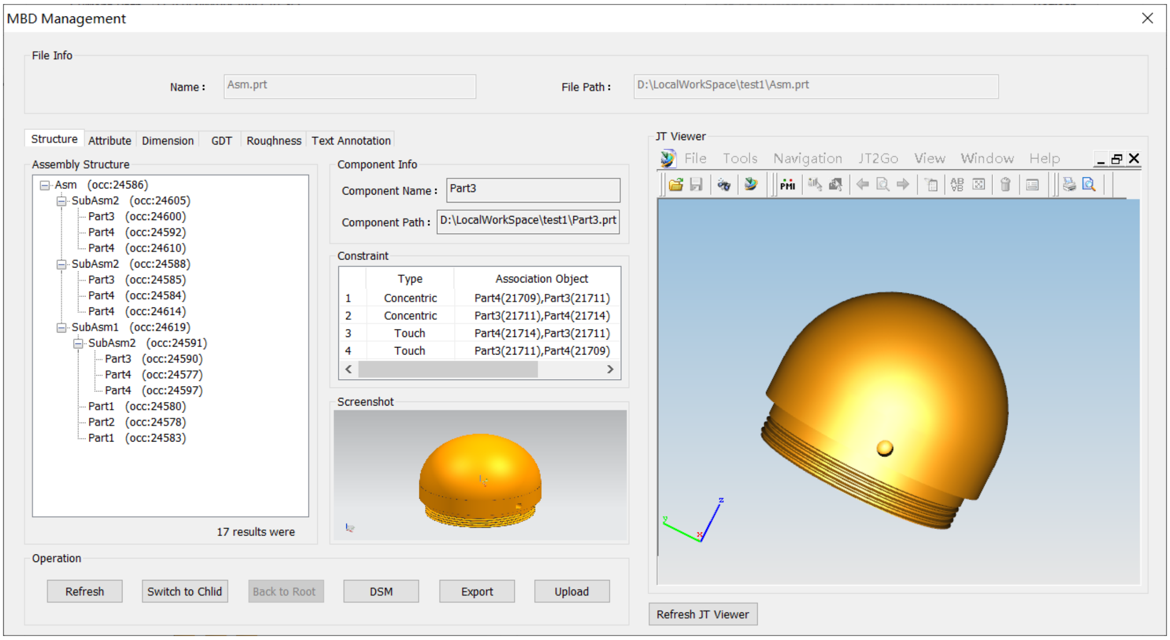

Figure 7: The structure/constraint information page of the module.

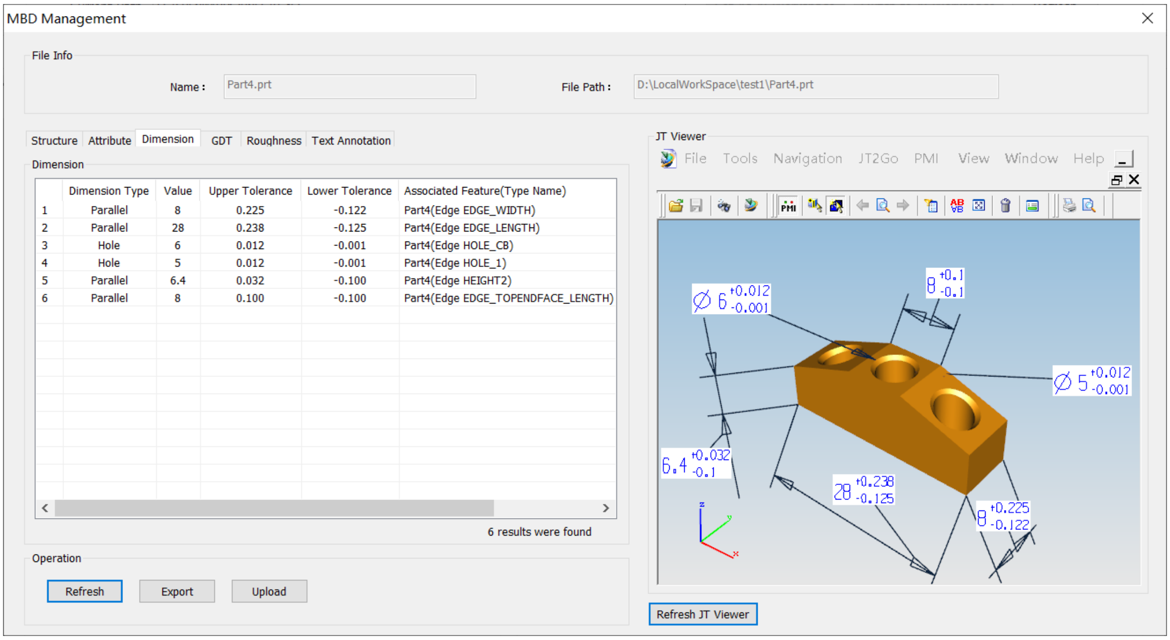

Figure 8: The dimension information page of the module.

The biggest advantage of the MBD information management module is that it optimizes the extraction process of MBD design information in the past, as shown in Fig.9. The traditional MBD information extraction process needs to be carried out in CAD environment, and several

© 2023 CAD Solutions, LLC, http://www.cad-journal.net

intermediate files are generated during the process. For the users, the process is tedious, and for the PLM system, dealing with the intermediate files will cause a waste of system performance. If the system does not clean the intermediate files in time, data redundancy and conflict will easily occur, affecting the system efficiency. In this work, the above problems are circumvented through the appropriate application of the UG/NX secondary development technology and the memory-mapped file.

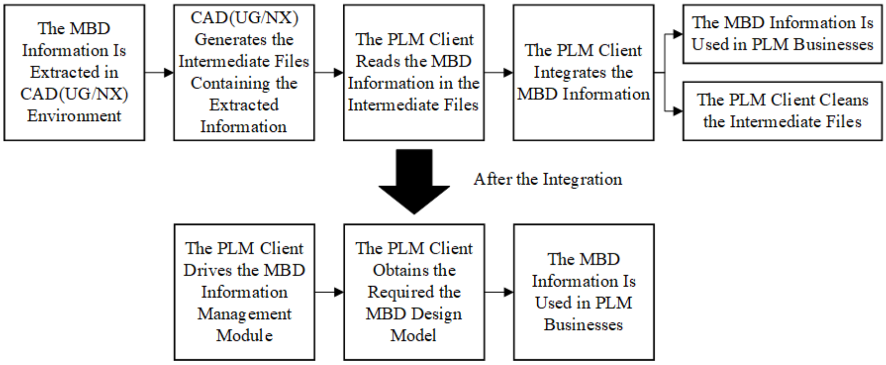

Figure 9: The optimization effect of the module on the MBD design information extraction process.

4.2 Upload the MBD Design Information

Based on the original upload process of the PLM client, the MBD information management module can automatically extract the MBD design information of assembly and components after the user starts the uploading process, and then format the information as XML, finally, the information is used for network transmission based on the network interface of PLM client. The relevant process is shown in Fig.10.

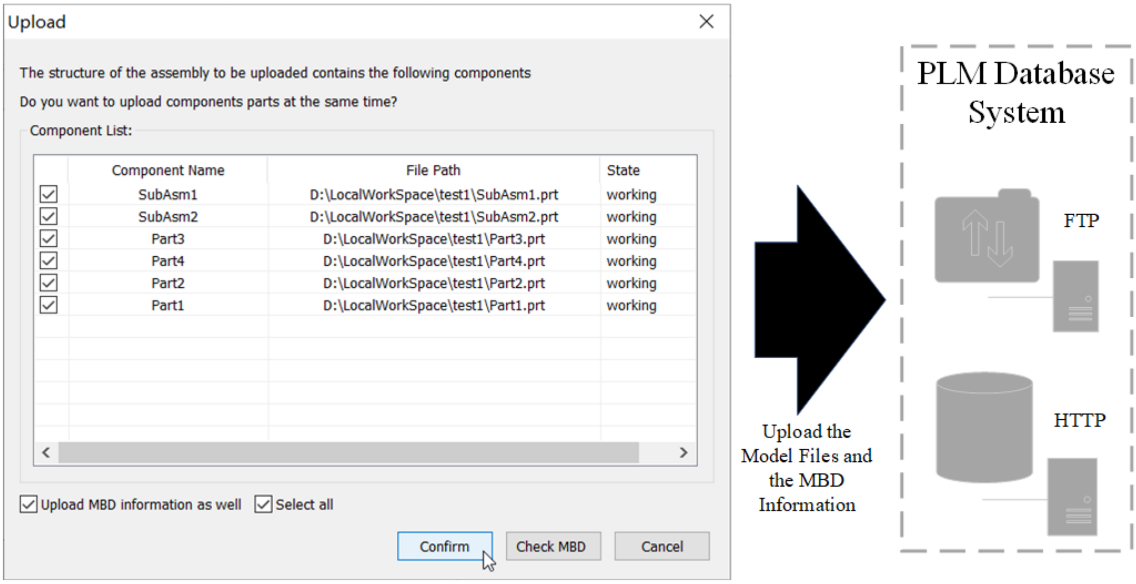

Figure 10: Synchronous upload the MBD design information.

© 2023 CAD Solutions, LLC, http://www.cad-journal.net

According to the section 2.2, the MBD design information will be reused in other links of product lifecycle. Due to the module's "allow synchronous upload the MBD design information" function, the application process of the MBD design information in other links is simplified to a certain extent, as shown in Fig.11. The MBD design information can be downloaded directly from the PLM database by other personnel. The repeated extraction of the information is avoided.

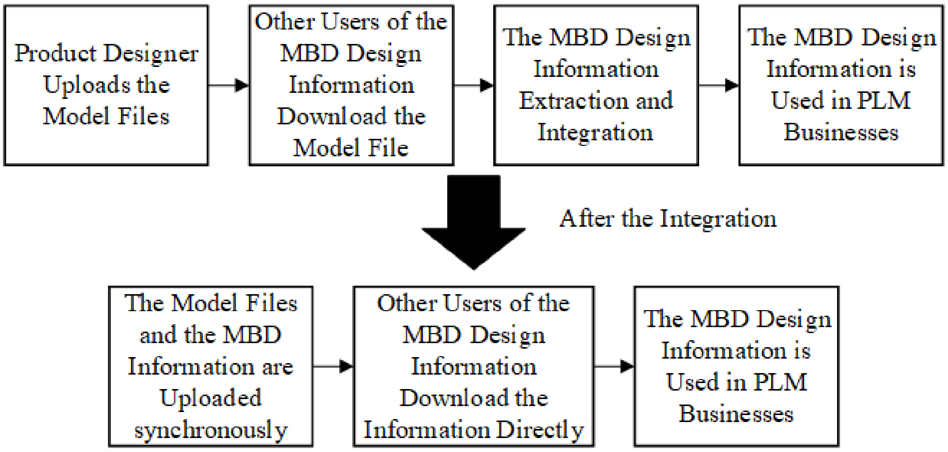

Figure 11: The optimization effect of the module on the MBD design information application process.

4.3 Parameter Driven Design Based on MBD

The MBD information management module also implements the parameter driven design based on MBD. The user can open the corresponding page in the PLM client environment and quickly generate the design model by setting the values of the key MBD dimensions. The related process and the generated model are shown in Figure is shown in Fig.12.

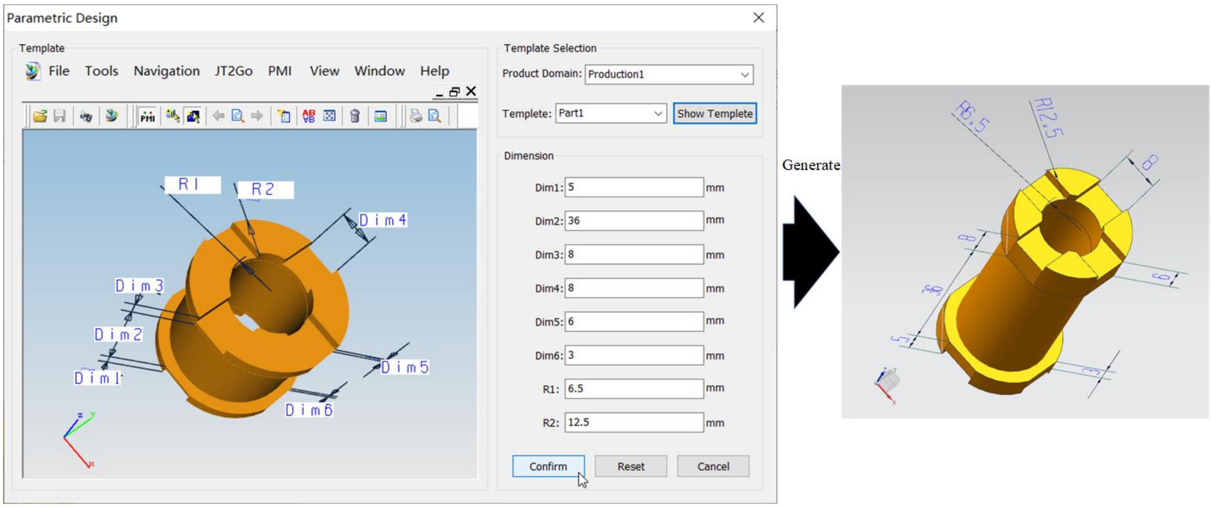

Figure 12: Process of parameter driven design.

© 2023 CAD Solutions, LLC, http://www.cad-journal.net

This function makes use of the MBD design information to simplify or even omit the design process of product, which provides convenience for the design link of product lifecycle and is conducive to shortening the product design cycle.

5 CONCLUSION

For the purpose of optimizing the traditional MBD information management process, this study developed the MBD information management module of the PLM client in the UG/NX integrated environment. It mainly implements the convenient extraction of the MBD design information, the creating of the MBD design model and formatting the MBD design information as XML used for the network transmission. In addition, the module also implements the parameter driven design based on MBD and batch output of product design information, lightweight format model, screenshot and other functions to support the management of the PLM system for the product design information.

From the actual integration effect, the main advantages of this study are reflected in the following aspects: (1) the module optimizes the extraction process of the MBD design information and avoids the generation of intermediate files in the integration process of the MBD design information. (2) The upload process of the MBD design information in design link and the application process of the MBD design information in other links of product lifecycle were optimized. (3) The module provides convenience for the product design work under the specific product domain. To sum up, MBD information management module has a certain extent of influence on the product lifecycle, which is conducive to saving time cost and improving the work efficiency of the PLM.

According to our team's investigation and research on PLM system, the limitations of the MBD design integration module mainly lie in that its function will be greatly restricted when facing other CAD software except the UG/NX. In order to solve this problem, we will explore the integration method of the module with other CAD software in the future research. Two solutions are proposed at present: (1) Firstly, converting the design model of product into a common intermediate format. Secondly, the extraction of the MBD design information is carried out. (2) Use the secondary development means of other CAD software to expand the function scope of the module, such as Pro/TOOLKIT tool of Pro/E, CAA function library defined by CATIA, etc. At present, we are conducting research on the above methods.

Based on the demand of modern manufacturing enterprises to expand production and promote digital production, the application of the MBD information will tend to be complicated in the future, and the information content of the MBD will also increase significantly in practical applications. In order to improve the data management performance of the MBD information management module, we plan to apply artificial intelligence (AI) technology in the next stage of research, which is used to process the MBD information under different business backgrounds and realize intelligent management and application of the MBD information.

ACKNOWLEDGEMENTS

This work was carried out at the Key Laboratory of Advanced Manufacturing Technology and Equipment of Liaoning Province at the Shenyang Ligong University, China. the research is funded by the Applied Basic Research Program of Liaoning Province (grant no. 2022JH2/101300254).

ORCID

Kai Wang, https://orcid.org/0000-0001-5735-4502

Pengfei Zeng, https://orcid.org/0000-0003-2994-9183

© 2023 CAD Solutions, LLC, http://www.cad-journal.net

REFERENCES

[1] Wan, N.; Mo, Rong.; Liu, L, l.; Li, J.: New methods of creating MBD process model: On the basis of machining knowledge, Computers in Industry, 65(4), 2014, 537-549. https://doi.org/10.1016/j.compind.2013.12.005.

[2] Zhu, W.; Bricogne, M.; Durupt, A.; Remy, S.; Li, B.; Eynard, B.: Implementations of Model Based Definition and Product Lifecycle Management Technologies: a Case Study in Chinese Aeronautical Industry, IFAC-PapersOnLine, 49, 2016, 485-490. https://doi.org/10.1016/j.ifacol.2016.07.664

[3] Corallo, A.; Vecchio, V. D.; Lezzi, M.; Luperto, A.: Model-Based Enterprise Approach in the Product Lifecycle Management: State-of-the-Art and Future Research Directions, Sustainability, 14(3), 2022. https://doi.org/10.3390/su14031370

[4] Quintana, V.; Rivest, L.; Pellerin, R.; Venne, F.; Kheddouci, F.: Will model-based definition replace engineering drawings throughout the product lifecycle? a global perspective from aerospace industry, Computers in Industry, 61(5), 2010, 497-508. https://doi.org/10.1016/j.compind.2010.01.005

[5] Prashanth, B. N.; Venkataram, R.: Development of Modular Integration Framework between PLM and ERP Systems, Materials Today: Proceedings, 4(2), 2017, 2269–2278. https://doi.org/10.1016/j.matpr.2017.02.075

[6] ruanfujia, https://www.ruanfujia.com/199906/, PLM product life cycle management connotation and technical framework.

[7] Saaksvuori, A.; Immonen, A.: Product Lifecycle Management, Springer Berlin Heidelberg, Berlin, Heidelberg, 2004. https://doi.org/10.1007/978-3-540-24799-9

[8] Stark, J.: Product Lifecycle Management (Volume 1), Springer International Publishing, Cham, 2022. https://doi.org/10.1007/978-3-030-98578-3

[9] Rinos, K.; Kostis, N.; Varitis, E.; Vekis, V.: Implementation of model-based definition and product data management for the optimization of industrial collaboration and productivity, Procedia CIRP, 100, 2021, 355-360. https://doi.org/10.1016/j.procir.2021.05.082

[10] Lan, G, L.; Zhang, Z, Q.; Shah, M.: Key Technology of CADDCAM Integration for Complex Parts of Marine Diesel Engine, Computer-aided Design and Applications, 19, 2021, 99-111. https://doi.org/10.14733/cadaps.2022.S2.99-111

[11] Varitis, E.; Rinos, K.; Kostis, N.: Model-based definition capabilities and its impact on industrial productivity, MATEC Web of Conferences, 318, 2020, 01048. https://doi.org/10.1051/matecconf/202031801048

[12] Huang, Y.: The formation and solution of enterprise "information island", China Computer & Communication, 04, 2018, 159-161.

[13] Zhou, Q, Z.; Cha, H, Y.: Digital product definition method based on 3D annotation technology, Journal of Machine Design, 01(28), 2011, 33-36. https://10.13841/j.cnki.jxsj.2011.01.020

[14] Zeng, P. F.; Hao, Y. Y.; Shao, W. P.; Hao, Y. P.: System integration method for assembly process information using model-based definition technology, Journal of Ordnance Equipment Engineering, 40(07), 2019, 190-195.

[15] Cicconi, P.; Raffaeli, R.; Germani, M.; An approach to support model-based definition by PMI annotations, Computer-aided Design and Applications, 14, 2016, 526-534. https://doi.org/10.1080/16864360.2016.1257194

[16] Gong, H. L.; Guo, Y. L.; Zhu, D. W.; wen, L. J.: Application of key technology of product process collaborative design based on MBD, Intelligent Manufacturing, 05, 2017, 45-49.

[17] Liang, X. G.; Zhou, J. B.; Weng, H. H.: MBD tolerance design international standard analysis and aerospace products application prospect, Aerospace Standardization, 02, 2021, 12-17. https://doi.org/10.19314/j.cnki.1009-234x.2021.02.005.

[18] Wu, R.; Zhang, D.; Wei, T.; Xu, F.; Zuo, D. W.: NC Machining process design system based on MBD, Aeronautical Manufacturing Technology, 07, 2017, 56-60. https://doi.org/10.16080/j.issn1671-833x.2017.07.056

© 2023 CAD Solutions, LLC, http://www.cad-journal.net

[19] Hao, B.; Jiang, B. B.: Technology of 3D-CAPP based on MBD, Journal of Sichuan Ordnance, 32(12), 2011.

[20] Deja, M.; Siemiatkowski, M. S.: Machining process sequencing and machine assignment in generative feature-based CAPP for mill-turn parts, Journal of Manufacturing Systems 48, 2018, 49-62. https://doi.org/10.1016/j.jmsy.2018.06.001

[21] Qu, L. G.; Sun, Y. X.; Yang, Y. G.; Ye, B. C.; Tian, J. Q.: Research on measuring points and their distributed strategy of inspection feature based on MBD, Aeronautical Manufacturing Technology, 17 ,2016, 97-102. https://10.16080/j.issn1671-833x.2016.17.097

[22] Mombeshora, I. M.; Dekoninck, E.: Web-Based Portal for Sharing Information through CAD/PLM Software during the Eco-product Development Process. In: Product Lifecycle Management for Society, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013. https://doi.org/10.1007/978-3-642-41501-2_38

[23] Wang, K.; Zeng, P. F.; Shi, C. J.; Shao, W. P.; Hao, Y. P.: A system integration method of product data management based on UG/NX secondary development, Intelligent Robotics and Applications: 15th International Conference, 2022, 466-476. https://doi.org/10.1007/978-3-031-13835-5_42

[24] Weiss, M. A.: Data structure and algorithm analysis C language description, China Machine Press, Beijing, 2019.

[25] Shaffer, C. A.: Data Structures and Algorithm Analysis in C++ (Third Edition), Publishing House of Electronics Industry, Beijing, 2020.

© 2023 CAD Solutions, LLC, http://www.cad-journal.net