Automating CAD for creating assembly structure from Bill of Materials

Nanang Ali Sutisna¹, Nasir Widha Setyanto²
1 Mechanical Engineering Department, Faculty of Engineering, President University
Jababeka Education Park, Jl. Ki Hajar Dewantara, Cikarang, Bekasi 17550, Indonesia
Email: nanang.ali@president.ac.id

2 Professional Engineer Study Program, Faculty of Engineering, Brawijaya University
Jl. MT Haryono 167, Malang 65145, Indonesia
Email: nazzyr_lin@ub.ac.id

Abstract. This paper presents the development of an add-on for CAD system generating assembly structure automatically based on a given Bill of Materials (BOM). The BOM in Excel format is scanned by the system and the system arrange 3D CAD files into assembly at in CATIA V5 CAD system automatically. This program has been tested and succeeded well at compiling 3D assembly based on 3D part file data that is available or not available. If the 3D file is not available then the 3D assembly is still generated according to the assembly structure listed on Excel file but its 3D representation does not appear.

Keyword: 3D CAD, CATIA V5, Product Data Management, Bill of Materials.

1. Introduction
Several manufacturing industries located in Indonesia, especially automotive industry, are mostly headquartered in Japan. Although they manufacture the vehicle at local factory, they do research and development at their own head office. In order to do manufacturing operation, the local company needs to gather all necessary data from its head quarter, including design data (CAD data), specifications, and other related document. The data management system in the head quarter and in local office are somehow not fully integrated and creates unproductive processing time because the local office should collect and put the CAD data in a certain way manually to form CAD assembly model.

2. Problem Statement
Product Data Management (PDM) is a software or system to control product-related data (Drawing, Technical Specification, Bill of Materials, etc.). Many of the world's leading manufacturing industries have implemented PDM, but the system at the head office and at local offices such as in Indonesia mostly have not been integrated, and sometimes even using different PDM systems. Therefore, data management is done conventionally by transferring data from the server at the head office to the local office.
Due to lack of integration, the design data, specifications, and assembly structure (called Bill of Materials or Structured Part List) are downloaded by the design and specifications section in local office. The design data is 3D CAD files while the assembly structure is usually in Microsoft Excel format. Structuring 3D assembly in CAD programs based on BOM data manually is really time consuming and potentially causing human error as the number of parts in the assembly is very large, which possibly reach around 6000 parts for motorbikes and even up to 30,000 parts for cars. To overcome the above problem, the author developed a program that automatically read assembly structure in Excel files and then arrange 3D CAD files into assembly in the CAD system, in this case CATIA V5.

3. Literature Review
In the global competitive market, companies are under constant pressure to improve their products, efficiencies and respond to continually changing market demands. They are also required to meet industry and government regulations. All of these has made enterprises to take reshape their product and process information. In many companies, product and process information is scattered across multiple division. In addition, a variety of software are used to create this information and there is little or no connectivity to bind these information assets together. As a result, it is difficult to coordinate and synchronize product and process information that is used to define the company’s product offerings, assemblies, parts, and their related systems [1].

A PDM system is used: to store and retrieve product design data; to keep track of all possible product configurations (product structure management); to control the release of engineering information (workflow management); and to record a detailed bill of material of every product (configuration management). PDM integrates and manages processes, applications and information that define products across multiple systems and media [2].

Based on The Tech-Clarity PDM Framework [3], there are three primary capabilities of PDM:
- Controlling and securing product-related data
- Improving the ability to quickly find and reuse information
- Sharing product knowledge and collaborating with other departments

Huhtala et al [4] highlighted the role of PDM in engineering design and its relation with PLM (Product Lifecycle Management) system, it can be said that the purpose of the PDM system is to manage the activities of the product during its lifecycle. Therefore, the system has to be flexible so that it is easy to use, easy to find and easy to modify. By implementing PDM system: the productivity can be improved, better access to information, and easier to manage data of products.

The previous work related to managing product data are mainly concern with the ideal PDM system which integrates all different related product data. The PDM system becomes a centralized data repository for all data related to all parts (product structure management, workflow management, and configuration management) as can be seen in Figure 1. However, due to some business arrangement a joint venture company can not implement a fully integrated PDM system between its head quarters and local companies. In addition, several components from local vendors, such as tyre, are also exists in the product but the design data is not available in the PDM system since the product design and development is done inside the vendor’s company [5].

In fact, CAD files and other related documents are downloaded separately from the head quarter server and are stored in local server for further processes. Later on the engineering people need to arrange all CAD data into a CAD assembly model based on information given in a Bill of Materials (BOM), and this job is very tedious and time consuming. Since there is no published report found regarding development of a software to automate this process, the author developed the software to automate the CAD assembly creation based on the provided BOM file.
The common practice in the field, CAD data 2D and 3D drawing files) and BOM (created in Microsoft Excel files) are imported into the PDM system separately. There are two methods of importing CAD data and related document into PDM system:

- The BOM files are imported into PDM system, and this will create BOM within PDM. Then, the CAD and specification files are attached and synchronized to BOM that has been formed earlier within the PDM. However, this method has disadvantage i.e the CAD model can not be viewed inside PDM system.

- In CAD system, create 3D assembly from existing CAD part files according to the given BOM files, then export the CAD assembly into the PDM. This method is meant to enable 3D CAD model be viewed inside the PDM Viewer both by the CAD user and non CAD users. Without this method, the 3D View object will not be formed inside PDM. The disadvantages of this method is: the time need to construct the CAD assembly manually is relatively long and having risk of human error.

With regard to the second method above, a program is needed automate the process of reading the BOM file and import 3D files into CAD according to the structure of assembly listed in the BOM. The development of this program is chosen by the author as a topic in this study.

4. Program Development Methodology
The program is developed using Visual Basic and having the task to import BOM (Bill of Materials) from Excel files and open 3D CAD files using CATIA V5 software, then arrange them in CATIA system according to the structure assembly in BOM.

The assumptions used in developing this program are: 3D CAD files have been created with CATIA V5 and BOM in the form of an Excel file contains information about: level in assembly, part number, part name, and CATIA 3D file location.
The method of developing this automatic CAD assembly program uses a waterfalls method [6], because the software is developed in a simple function and the definition of needs are clear and there are no complicated relationships with other systems as shown in Figure 2.

Figure 2. Waterfall model [7]

The program is named XL2CATIA and the process of developing the program is carried out with the waterfall model summarized in Table 1.

Table 1. Development Phase

| Development Phase          | Detailed process                                                                 |
|----------------------------|----------------------------------------------------------------------------------|
| Requirements engineering   | Requirements analysis is done by observation the need for software specifications, namely: |
|                            | Input: Excel file containing BOM                                                  |
|                            | Output: CATIA V5R24 3D file containing Assembly according to the structure of the BOM |
| Architectural design       | Making process flow diagrams that must be carried out by software in outline      |
| Detailed design            | Making flow charts for each stage the process that must be done by software in more detail |
4. Coding

| Coding          | Encoding with Visual Basic includes making interfaces and functions required |
|-----------------|--------------------------------------------------------------------------------|
| Unit testing    | Test each function unit/procedure in the program. |
| System testing  | Test the entire program including synchronization between Excel files as CATIA inputs and files as output. |
| Acceptance      | Perform a joint system testing process customer (UAT) and documented the results. If adjustments are needed, it will be done repairs as needed and testing reset it. |

4.1 Requirement Engineering Phase

The desired requirement is to read the BOM similar to Figure 3 and make a 3D assembly in CATIA V5 based on the BOM arrangement, as shown in Figure 4.

Figure 3. BOM in Excel format
4.2 **Program Architecture**

The XL2CATIA program architecture is designed as Figure 5, in the following order:

1) XL2CATIA reads BOM in Excel which contains information about Part Number, Part Name, Level in assembly, and location of CATIA files

2) XL2CATIA opens the CATIA V5 application, if it is already open it will load 3D files CATIA corresponds to the order in the BOM and forms a suitable 3D assembly arrangement BOM

4.3. **Detailed Design**

Detailed design includes interface design and design of functions needed in XL2CATIA operations.

1) **Interface Design**

Given its simple function, the interface design is made as simple as possible in accordance with its function, as can be seen in Figure 6a. There are only three command buttons for the "Continue", "Cancel" and "Customize" functions and an "About" menu that explains the version, function, maker, and year of manufacture.

The "Customize" function (Figure 6b) is to determine in what column information about Product Level, Part Number and Part Name for BOM input are contained in Excel file. The "Continue" button is pressed if you want to continue and the "Cancel" button to cancel the operation.
Figure 5. XL2CATIA Architecture

(a) Main Menu
(b) Customize Menu

Figure 6. XL2CATIA User Interface Design (a) Main Menu (b) Customize Menu
2) Program Flow
The program flow diagram consists of a main program (Figure 7a) and several sub-programs underneath (Figure 7b), consisting of:
- Main Program
- Customize Option
- Continue option

![Flowchart of Main Program and Continue Option](image)

(a) (b)

**Figure 7. Flowchart of Main Program and Continue Option**

4.4 Coding
Based on the flow diagram in Figure 7, a program is developed with Visual Basic 6. The use of Visual Basic is done to make it easier to call the declaration of the CATIA function through the API because CATIA uses VB Script for the macro / automation process. Coding phase includes writing program code, compiling, running programs, and debugging.

4.5 Unit and System Testing
Unit and system testing includes testing the success of entering data from the "Customize" form, the success of loading CATIA data and its suitability with BOM.

1) The first test on the unit "Customize" where the data entered in the Customize form will be stored in the Custom.xlssetting file. The data in the Custom.xlssetting file must be the same as the inputted data in Customize form. The result is the same, meaning this unit works well.

2) The second test is to test whether XL2CATIA successfully calls the CATIA application, if CATIA is not open, then it will open the CATIA application. If it is open then the process of
forming the product assembly structure will be carried out. This test was successful call
and open the CATIA application.

3) The third test is to ensure that the BOM data has successfully formed a structure product
assembly perfectly in CATIA.

In Figure 8, the CATIA 3D assembly has been formed, and after comparison with the input file,
which is BOM in Excel, turns out to be the same. Thus this test was successful.

![Testing result, the generated CAD assembly, compared to Excel BOM](image)

**Figure 8.** Testing result, the generated CAD assembly, compared to Excel BOM

4.6 Test Result Summary

XL2CATIA’s success in fulfilling its duties can be seen in the Report.log file. The log file will provide
information on how many CATIA files have been successfully loaded into CATIA and how many
unsuccessful, and shows which Part data and in what row is not successfully loaded. The log data
below shows that all products have been successfully uploaded.
Figure 9. The test result showing the successful rate of forming 3D assembly from BOM

The results of the test shown in Figure 8 is added to the Report.log file (Figure 9), it shows that XL2CATIA has successfully fulfilled its task very well.

5. Conclusion
The XL2CATIA program was built to form a CATIA 3D file assembly automatically based on the assembly structure created in an Excel file called BOM (Bill of Materials). This program is built using Visual Basic 6 with the consideration that compatibility with CATIA applications is better because scripting in CATIA V5 uses VB Script.

The software development process used in building XL2CATIA is a Waterfall model, which consists of Requirement Engineering, Architectural Design, Detailed Design, Coding, and Testing. The user acceptance is not included in this study since it involves the company that refuses to be published.

Test results on the success of XL2CATIA in uploading CATIA 3D files and arranging it into assembly has been done and the results show that XL2CATIA can do his job very well.

6. References
[1] Siemens PLM Software Inc. 2014 Getting Started with Product Data Management pp 4-6
[2] Siddiqui, Q. A, Burns, N. D and. Backhouse, C. J. 2004, Implementing product data management the first time International Journal of Computer Integrated Manufacturing, September 2004, Vol. 17, No. 6 520–533
[3] Tech Clarity Inc. 2015 The Facts About Managing Product Data: The Real Story on PDM Value and Accessibility pp. 4-5
[4] Huhtala, M., Lohtander, M., and Varis, J., 2013, The Role Of Product Data Management (PDM) In Engineering Design And The Key Differences Between PDM And Product Lifecycle Management (PLM) The 1st PDM forum for Finland-Russia collaboration, April 2013
[5] Sutisna, N.A and Gapsari, F. 2018 Rolling Resistance and Noise Estimation for Product Designand Development of Eco-Tyre using Finite Element and Numerical Method Proceeding of
International Conference on Mechanical Engineering Research and Application 2018 IOP Conf. Ser.: Mater. Sci. Eng. 494 012021

[6] Sommervile, Ian 2011 Software Engineering 9th Edition, Addison Wesley

[7] Pressman, Roger S. 2010 Software Engineering: A Practitioner Approach 7th Edition, McGraw Hill