A Bill of Material Transformation Method for Design-Manufacturing Integration

Wei MU¹, Chun-liu ZHOU²*, Hong-guang BO² and Jun-yi LIANG¹

¹Department of Production Management, CRRC Qingdao Sifang CO., Ltd., Qingdao, 266109, China
²Department of Management, Dalian University of Technology, Dalian, 116023, China
*Corresponding author

Keywords: BOM, DMI, Feature reorganization, ERP, PDM.

Abstract. In order to support design–manufacturing integration (DMI), a transformation method is proposed by recognizing the features of EBOM and MBOM, to effectively ensure data integrity, correctness and consistency in systems. By defining different components, the transformation processes are identified and a formal transformation model of BOM view is proposed. Then, rules for BOM transforming and conversion algorithms are described. Engineering change problems also are solved by processing rules. This research has achieved an automatic conversion of BOM. Finally, the practical application in the enterprise indicated that this method is effective for DMI.

Introduction

Design–manufacturing integration (DMI) activities can raise an organization’s ability of identifying and effectively addressing product process design interdependencies [1], in which computer systems and data integration plays a vital role. Bill of material (BOM) is initially defined in the product drawings and then converted into different BOM definitions and configurations, which are extended throughout the product life cycle. BOM is core data in production industry, which is the bridge of product design and manufacturing [2]. BOM is the technical description document of a product structure. During the implementation of computer integrated manufacturing systems, most manufacturing companies have applied Enterprise Resource Planning (ERP) system and Production Data Management (PDM) system. Each system is controlled independently in different department, which leads to inefficient information communication and interaction between two systems and exist reduplicative BOM data, and lack of data integrity and consistency [3].

Thus, manual intervention has to be used to coordinate consistency of BOM information. This phenomenon has seriously influenced integration of product design, manufacturing and management, even extended time for product development. In order to propel DMI, this paper proposes a transformation method based on feature recognition, which focuses on the transformation from Engineering BOM (EBOM) to Manufacturing BOM (MBOM).

BOM Views in PDM and ERP

Introduction of EBOM and MBOM

BOM information of a product in PDM system and ERP system respectively has two different BOM views, EBOM in PDM and MBOM in ERP.

EBOM is used to describe product design structure for organizing and managing list of parts in product design department. Designers design products according to customers’ requirements and design needs. After finishing design, information which derives from design drawings, including product name, product structure, and assembly relationship among parts, makes up EBOM.

MBOM is used to organize and manage lists of materials in actual production process of manufacturing department. MBOM generates from EBOM. Because manufacturing information is
more complicated than design information, MBOM is more complicated than EBOM naturally, which reflects assembly structure and manufacturing information of self-made parts. MBOM is significant for operations management, especially in computing MRP. Amounts and arriving time of all required materials can only be known from BOM, which is also used to make a plan for manufacturing and purchasing.

Thus, there exist essential differences between EBOM and MBOM. EBOM is a technical file while MBOM is a managerial file. More exactly, MBOM is a core business information model which synthesizes information of sales, manufacturing and supply.

Figure 1 shows the differences in structure between EBOM and MBOM. In the figure, letters represent parts and numbers shows assembly relationship in quantities between father-node and sub-node.

Math Model for EBOM and MBOM
In order to accurately define the mapping functions of EBOM and MBOM according to their characteristics, mathematical descriptions are shown respectively as follows.

**Definition 1.** A triple \((A, C, Q)\) is used to denote EBOM, noting for \(EBOM = (A, C, Q)\). Wherein, \(A\) is a set of components \(a_i (i = 1, 2, \ldots, m)\); \(C\) is a set of components assembly relation \(c_{ij}\), which represents the assembly relation between \(a_i\) and \(a_j\); \(Q\) is a set of the number of components in the assembly, \(q_{ij} = q(c_{ij})\), which represents that a parent component is made up of \(q_{ij}\) sub-parts in an \(c_{ij}\) assembly relationship. Thus, it is listed for figure 1 case as below:

\[
A = \{a, b, c, d, e, f\};
C = \{(A, b), (M, c), (M, d), (M, e), (M, f), (c, c), (e, e), (d, d), (d, d)\};
\]

Correspondingly, \(Q = \{2, 3, 3, 2, 2, 2, 1, 3, 2\}\)

**Definition 2.** A seven-tuple \((A^M, C^M, Q^M, Z, X, R, T)\) is used to denote MBOM, noting for \(MBOM = (A^M, C^M, Q^M, Z, X, R, T)\). Wherein, \(A^M\) is a set of components \(a_i^M\) in MBOM; \(C^M\) is a set of components assembly relations \(c_{ij}^M\) in MBOM; \(Q^M\) is a set of assembly number \(q_{ij}^M\); \(Z\) is a set of the process state \(z_{ij}\) of parts, which is the \(j^{th}\) process state description of components \(a_i^M\); \(X\) is a set of processes \(x_{ij}\), which represents the \(j^{th}\) process; \(R\) is a set of resources \(r_{ij}\), which means resource.
occupied by process $j$; $T$ is a set of work hours $t_{jk}$ which represents the time consumed by part $a_i^M$ in process $j$ with resource $k$. Thus, it is listed for figure 1 case as below:

$$A^M = \{M, g, a, b, c, e, c_2, d, e_1, f\}; C^M = \{(M, g), (M, c_1), (M, c_2), (M, d), (M, e_1), (M, f), (g, a), (g, b)\};$$

Correspondingly, $Q^M = \{1, 6, 3, 2, 2, 2, 3\}$

As it can be seen from the above definition, the elements contained in MBOM differ from ones in EBOM, which makes it more difficult to achieve conversion from EBOM to MBOM\(^6\). To ensure the consistency of BOM data conversion, this paper introduces PPBOM (Process Planning BOM) concept\(^7\). PPBOM information mainly comes from process sector, which includes: the product process design documents and assembly process design document.

**Definition 3.** A six-tuple $(A, S, Z, X, R, T)$ is used to denote MBOM, noting for $PPBOM=\langle A, S, Z, X, R, T \rangle$.

Wherein, $A$ is still a set of part $a_i(i=1,2,\ldots,m)$; $S$ is a set of parts production type properties, which describes not only the type of production of its components (homemade member, the association member, purchased parts), but also the state whether the component is virtual, intermediate or inherited parts, $S_i(i=1,2,\ldots,m)$ defines the production type attribute. As shown in figure 1,

$$A = \{M, a, b, c, e_1, c_2, d, d_1, e_2, f\}$$

Correspondingly, $S = \{l_i, M, V_i, V_d, C_i, C_d, P_i, P_d\}$

Wherein, $l_i$ denotes inherited component; $M_i$ denotes intermediate component; $V_i$ denotes virtual component; $C_i$ denotes outsourcing component; $P_i$ denotes purchased components. Other letters have the same meaning as described in EBOM and MBOM. In EBOM to MBOM conversion process, the $Z, X, R, T$ information of MBOM can be obtained directly from PPBOM\(^8\). $A^M$ can be re-set from the the production of the type attribute and the set A of parts in EBOM. $C^M$ and $Q^M$ both can be got from the $C$ and $Q$ of EBOM. Figure 2 shows the systematic transformation process.

**Transformation Rules from EBOM to MBOM**

In order to be convenient, $C_I$ is defined as a set of all the inherited components, $C_V$ is a set of all the virtual components, $C_M$ is a set of all the intermediate components, $C_C$ is a set of all the outsourcing components, and $C_P$ is a set of all the purchased components. Various types of processing rules in the transformation are given below.

**Transformation Rules of Inherited Components**

For product P, if component $a_i$ in EBOM is an inherited component, then the assembly relation is the same in MBOM with the relation in EBOM. The mathematical description is as follows:

![Diagram](image-url)
∀a_i∈C_v, exists \[
\begin{align*}
    a_i^M &= a_i \\
    c_i^M &= c_i \\
    q_i^M &= q_i
\end{align*}
\] (1)

In formula (1), \(c_i^M\) and \(q_i^M\) respectively represent assembly relationships and the number of assembly in MBOM. Here, \(c_i\) and \(q_i\) respectively represent assembly relationships and number of \(a_i\) in EBOM.

**Transformation Rules of Virtual Components**

If there is virtual component in EBOM / PBOM, then its sub-parts and the relations can be moved to its parent component in MBOM. The mathematical description is as follows: \(∀a_i∈C_v, a_i\) is the parent part of \(a_i\), \(a_j\) is the child part of \(a_i\), there exists

\[
\begin{align*}
    a_i^M &= a_i \\
    c_i^M &\notin A^M; c_i^M \in C^M \\
    q_i^M &= q_i \times q_j
\end{align*}
\] (2)

In formula (2), \(c_i^M \in C^M\) represents the assembly relationship between \(a_i\) and \(a_j\) in MBOM, \(q_i^M\) represents assembly number of \(a_i\) for \(a_i\), \(q_j^M\) assembly number of \(a_j\) for \(a_i\).

The algorithmic process of virtual component mapping rule is as follows:

Step1: Put all the virtual components into a “P-Stack (P means a part)”.
Step2: When P-Stack is not null, assign an element in the stack to P, or go to Step 9.
Step3: Search the father ID of P and its assembly number towards its father component in EBOM.
Step4: Search all the child parts of P, including information like child ID and assembly number, and then keep them in “C-Stack (C means child part)”.
Step 5: When C-Stack is not null, assign an element in the stack to C; or go to Step 6.
Step 6: Change the ID of C’s father to the “father ID”, assembly number is the result of multiply.
Step 7: Go to Step 5.
Step 8: Go to Step 3.
Step 9: Over.

**Transformation Rules of Intermediate Components**

If there is an intermediate component, then add the parent-child relations and the relevant process requirements to MBOM. The mathematical description is as follows: \(∀a_i^M ∈ C_v\), \(a_i^M\) is the child parts of \(a_i^M\), \(a_i\) is the parent parts of \(a_i\), there exists

\[
\begin{align*}
    a_i^M &= a_i \\
    c_i^M &\in C^M; c_i^M \in C^M \\
    q_i^M &= q_i \times q_j
\end{align*}
\] (3)

In formula (3), \(c_i^M \in C^M\) and \(c_i^M \in C^M\) represent respectively the assembly relationship between \(a_i^M\) and \(a_j^M\), and the relationship between \(a_i^M\) and \(a_j^M\).

The mapping process of intermediate components is similar to the mapping process of virtual components, which will not be mentioned here.

**Transformation Rules of Outsourcing/ Purchased Components**

Outsourcing components generally exists when enterprise outsources a process or a whole component to other manufactures due to its capacity and cost restrictions. For the process outsourcing components, the transformation rule is the same with inherited component, but the description is marked as outsourcing on the process. For the whole outsourcing component, only the component is described in MBOM, no information about its child parts are listed. Thus, this kind of outsourcing components can be dealt as a part in MBOM.
Since purchased component come from outside company, it can be dealt with in the same way with the whole outsourcing component.

The mathematical description is as follows: \( \forall a_i \in C_C, a_j \) is the child part of \( a_i \), there exists

\[
\begin{cases}
   a_i \in A^i; a_j \notin A^i \\
   c_i \notin C^i \\
   q_i \notin Q^i
\end{cases}
\]

In formula (4), \( c_i \notin C^i \) represents that the assembly relation between \( a_i \) and \( a_j \) is canceled in MBOM. In the same way, \( q_i \notin Q^i \) represents that the assembly number relationship between \( a_i \) and \( a_j \) is canceled in MBOM.

In summary, in accordance with the four rules above, transformation function from EBOM to MBOM is: \( MBOM = f_C(f_M(f_V(f_I(EBOM, PPBOM)))) \), where

- \( f_I \) — Transformation function of inherited components
- \( f_V \) — Transformation function of virtual components
- \( f_M \) — Transformation function of intermediate components
- \( f_C \) — Transformation function of outsourcing/purchased components

**Transformation Rules for Engineering Change in BOM**

In the enterprise actual production process, engineering change is an act frequently encountered, such as the introduction of new parts, upgrade older parts, modifications of the reference document. These changes may result in changes of BOM, leading to changes of EBOM and MBOM. Therefore, in the conversion process from EBOM to MBOM, we must consider the impact of engineering changes to the BOM conversion. Since most of the engineering changes are based on the BOM change, BOM change is divided into two categories, namely: non-structural BOM changes and structural BOM changes.

**DEFINITION 4:** Non-structural changes mean although the change exists in product, the description of production type in PPBOM to EBOM does not change. For this situation, MBOM can be getting via inherited mapping from EBOM.

**DEFINITION 5:** Structural changes generate when the description of production type in PPBOM to EBOM changes, leading to the assembly structure of virtual, intermediate, outsourcing or purchased components different in EBOM and MBOM. In this situation, the transformation from EBOM to MBOM by judging production type of parts and dealing in each rule respectively.

By defining transformation rules and BOM engineering changes, automatic conversion process is shown in figure 3.

![Diagram](image)

Figure 3. The process of transformation from EBOM to MBOM.
**Case Study**

Figure 4 shows an example of transformation from EBOM to MBOM of a locomotive production. In EBOM, according to actual assembly, the cab will be put on the locomotive body after it is assembled, thus Cab general view is a virtual component, which will not show in MBOM. In the assembly of Diesel generator installation, due to the requirement of process, flange (1) and flange (2) must be assembly first to be an assembly part- elastic flange assembly, and then it will be drilled and stored. So, the elastic flange assembly is an intermediate component. Besides, although windshield assembly is a sub-part of cab general view in EBOM, it is the last one to be assembled in actual operations, then its position changes in MBOM relative to EBOM.

**Conclusion**

This study built a transformation model by recognizing the features of EBOM and MBOM to solving the problem of overlaps and inconsistencies of BOM data in PDM and ERP systems for design-manufacturing integration. By introducing inherited, virtual, intermediate, outsourcing, purchased components and engineering change, as well as defining processing rules respectively, automatic conversion has achieved. Focusing on testing and feedback of product data in the conversion process ensures accuracy, completeness and consistency of BOM data. Practical application verified that BOM conversion technology based on feature recognition is reasonable and feasible. This method is hoped to be used in digital manufacturing environment and extend to maintenance data management area to realize production life time management.

Acknowledgement

This research was financially supported by a project supported by National Science and Technology Support Program, China (2015BAF08B02) and the State Key Program of NSFC (61533005).
References


