Int J Adv Manuf Technol (2016) 84:197–212
DOI 10.1007/s00170-015-7815-6

ORIGINAL ARTICLE

A modular structure data modeling method for generalized products

Hao Li [1] · Yangjian Ji [2] · Guofu Luo [1] · Shanghua Mi [1]

Received: 29 May 2015 / Accepted: 7 September 2015 / Published online: 3 October 2015
© Springer-Verlag London 2015

Abstract Traditional product modeling methods mainly aim at physical products. However, with cloud manufacturing, the efficient management and utilization of big data under the support of cloud service platform are a challenge, and establishing a reasonable data model is the core of product life cycle data management for generalized products. Aiming at big data management of generalized product modules, the definition, description, and classification of general modules are given in this paper firstly. Then, modular master structure data models are built, including the composition of the generalized modular master structure, property description of generalized part variables, connection objects among the generalized parts, etc. Thirdly, modular instance structure data models of the generalized product are established, the formation process of the instance structure model is analyzed, and the structure mapping views in different phases of the product life cycle are also presented. Finally, based on the secondary development of a product life cycle management (PLM) system, the generalized product data models and the modular structure models of the transformer are carried out. The traditional integrated product models are enriched and improved so the demand of modular data management of generalized products is satisfied under the environment of cloud manufacturing.

✉ Yangjian Ji
mejyj@zju.edu.cn

Hao Li
lh9666@hotmail.com

[1] School of Mechanical and Electric Engineering, Zhengzhou University of Light Industry, Zhengzhou 450002, China

[2] Center of Industry Engineering, School of Mechanical Engineering, Zhejiang University, Hangzhou 310027, China

Keywords Generalized product · Product service system · Physics module · Service module · Data modeling · Big data

1 Introduction

Driven by pressures from globalization, environmental/resource concerns, advanced technologies, and customized requirements, transformation and upgrading of manufacturing enterprises are very urgent [1, 2]. To sustain profits, traditional manufacturing enterprises should provide products by using new technologies, such as cloud techniques and cloud manufacturing, and sell products with value-added services, provide solution packages of physical products and/or product services, known as "generalized products" [3–6], rather than selling pure physical products. In this field, some scholars also refer to it as product service system (PSS) [7, 8] and integrated product service system (IPSS) [9].

According to the different proportions of physical products and services in service solution packages, generalized products can be divided into three types: pure physical products, integrated service products, and pure service products [3]. Compared with traditional products, generalized products pay more attention to the role of service profit in product sales. Therefore, in view of the level of generalized products, pure physical products are the lowest level products, integrated service products are in the middle, and pure service products are in the highest level. Integrated service product is a hot research topic currently. It obtains additional profits through the sales of services; pure service product is a future trend in product solutions/provisions [5].

Cloud manufacturing is a computing and service-oriented manufacturing model developed from existing advanced manufacturing models (e.g., ASP, AM, NM, MGrid); enterprise information technologies under the support of cloud computing, IoT; virtualization and service-oriented technologies; and advanced computing technologies [10, 11]. Under the environment of cloud manufacturing, efficient management and utilization of big data in product design stage are a challenge. At the stage of product design, the big data involved can be traced from description of needs to specific product function description and, finally, detailed design specifications like drawings of product configurations [12]. The technologies of modularity would have an advantage in enhancing the speed of new PSS design and development [13]. Organizational modularity should be a part of the modularity of PSS, however, which is unsuitable to be carried out via a product platform [5]. In order to research the data model of a generalized modular product platform, we use the concept “the generalized product” but not the “PSS” because we pay more attention to modular description and data management of generalized products.

Modular design is the most important step for generalized product modularity. With cloud manufacturing, the realization of modular product design can be carried out based on the product platform. A product family includes some variant products, and each product is composed of lots of modules. If a universal data model for describing and managing modules stemming from the process of generalized product modularity is not presented, a huge number of variant products would be produced and lead to a large amount of product data redundancy. With lots of modules existing in the modular design platform, how to manage these modules becomes a key in the process of modular design.

For the reasons stated above, we need to design and build a comprehensive modular data structure model for generalized products, so as to reveal multiple attributions of the modules and the relationships between modules (including physical modules and service modules) [14]. Based on the modular master structure of a generalized product platform, lots of generalized product instances can be derived via the configuration design. However, the existing product data model only focuses on design data description and technical state changes of physical modules, which are difficult to describe. Building up a universal and reasonable data model for generalized products as a basis for generalized modular product design and product life cycle management under big data and cloud manufacturing is more meaningful.

In this paper, the authors intend to build universal and reasonable data models for generalized products and modules, which could meet the management requirement of generalized modules (including physical and service modules), reveal the relationship between functional service modules and physical modules, and give the relationships between different types of modules, such as fixed correlation, assembly, connection between “parts” objects, etc. The models proposed in this paper make the existing integrated product model more comprehensive and universal. This paper is organized with a structure as follows: The second section is a research review in related fields; the third section gives the definition, description, and classification of generalized modules; the forth section mainly builds the data model of generalized product modular master structures; the fifth section puts forward the data model of the generalized product modular instance structure; the sixth section uses a case to verify the key models; and finally, the summary shows the significance and future work of the described modeling methods.

2 Research review

In the field of PSS data management and big data management, Sakao proposed a new prototype of service CAD system called “Service Explorer” to support conceptual design in service/product engineering, with the core model representing critical concepts such as value, costs, functions of either products or service activities, and entities [15]; Li analyzed and concluded the big data involved in the three main phases (i.e., beginning, middle, and end of life) of product life cycle management (PLM) [12]; Tao proposed a computing and service-oriented manufacturing model under the environment of cloud manufacturing, and designed and presented a five-layered framework of manufacturing resource intelligent perception (including product design and manufacturing service resources) and access system based on IoT via cloud manufacturing [10, 16, 17]. Under the cloud computing environment, Tao put forward a case library and Pareto solution based on hybrid genetic algorithm (CLPS-GA), and also solved a multi-objective MGrid resource service composition and optimal selection problem based on the principles of particle swarm optimization [18–20]. The researches on resource management methods provide helpful guides on product and service design management. For the management of product design and service data, the key is in building a universal and reasonable data model.

The data model for physical products proposed in recent years mainly includes product data models based on product structure [21, 22], object-oriented product data models [22–24], and product data models for PLM [25–27]. However, these models have their limitations when describing physical products as data changes in modular product life cycle are difficult to describe accurately and comprehensively [28]. Targeting the above limitation for data models, German scholar Josef Schoettner proposed an integrated product model theory based on the product structure which is oriented to document description and organization, and object-oriented, and oriented to PLM, a perfect expression of information on physical product life cycle can be achieved [28–30]. The integrated product model theory originally proposed by Schoettner [29], and later enriched and perfected by Schoettner and QI, had been a classic product data model in the field of PLM [30–32]. In addition, some scholars such as Yu [33], Gu [34], Li [35], Yun [28], et al., have also improved on the model. Furthermore, in recent years, the application development of a PLM system (such as OpenEDM) conducted by the Germany Intellivate company is based on the integrated product model theory.

However, the existing integrated product model was designed solely for physical product life cycle. When considering integrated modular design containing both physical and service module design, the limited compatibility and scalability of traditional physical product data models would pose a challenge, and it would be difficult to manage a large number of service modules and their complex relations with physical modules effectively. Although Hu has proposed a product structure model based on service maintenance data of product life cycle [36], it was only in view of product repair and maintenance services instead of managing service modules in the design phase. We need to analyze whether traditional physical product models (such as integrated product model theory) could meet module data management requirements of generalized product and, if not, how the traditional model could be improved, and proposing some new models becomes a difficult problem for generalized product modular management, so it needs to be solved urgently.

3 The definition, description, and classification of generalized modules

3.1 The definition of generalized modules

(1) The physical module

Various scholars have attempted in defining the physical product module [37, 38], but there is still not a unified standard definition. The mainstream definition of the physical module was given by Masahi ko Aoki, who defined the module as a half self-disciplining subsystem with certain independent functions which can be interconnected with other half self-subsystems according to certain rules, and forms a more complex system. It is also an entity, independent functions and structures, with fixed interfaces between modules [39].

(2) The service module

Definition 1: Service module. The service module is an abstract object of tangible or intangible services with an independent function; it performs service functions through interactions between physical modules and service processes. For the implementation of service modules, manufacturers often provide for customers in the form of a service module package. The service module package is an aggregate of modules with similar properties and can be combined into a solution. The service module can be divided into functional modules and non-functional service modules based on the degree of coupling between the service module and the physical module [4, 40].

Definition 2: Functional service module. The functional service module is the result of modularity of functional service; it has close relationships with particular physical modules and is carried out with the support of related physical modules. For example, if the remote monitoring service is needed, an appropriate physical module needs to be added to the product to implement the monitoring service function, so the remote monitoring service is the functional service module in this case.

Definition 3: Non-functional service module. The non-functional service module is the result of modularity of non-functional service; it has no direct or strict relationships with physical products and can be carried out without depending on specific physical modules. Some modules such as installation services, transportation services, and financial services are non-functional service modules.

Definition 4: Generalized module. The generalized module can perform specific functions via independent physical structures or service processes; it includes physical modules and service modules. The similarities and differences between physical modules and service modules are shown in Table 1.

3.2 The attribute description of generalized modules

Generalized modules consisting of physical and service modules need to be analyzed separately in its attributes. A physical

Table 1 The comparison between physical modules and service modules

Types	Sub-types	Existence form	Stability	Composition
Physical module	Physical module	Tangible	Has good structure stability	Composed of the common module and special module
Service module	Functional service module	Tangible	Can be added or cancelled according to user requirements quickly	It needs the support of particular physical modules
	Non-functional service module	Intangible	Can be added or cancelled according to user requirements quickly	Composed by the common module and special module

module can be described by three attributes: feature size, material, and interface (reveals the assembly relations, see Fig. 1a). Feature size determines the specific shape of the physical module, material determines the material quality of the physical module, and interface defines the interactions among modules. For service modules, Xu proposed a project object could be described by four attributes: activities, executors, resources, and time. These attributes can be used to describe the implementation of a service object consisting of four units: activities, executors, resources, and time (see Fig. 1b) [41]. It is noteworthy that functional service modules can be carried out by physical modules so the attributes of the functional service modules not only requires the activities, executors, resources, and time, but also needs to describe the matching relations with physical modules.

In conclusion, the description of the generalized module attributes is a union set of the physical module attributes and the service module attributes (see Fig. 1c). It inherits the features of the physical module and the service module respectively, so the generalized module can be fully expressed through properties, such as activities, executors, resources, feature size, materials, interface, time, etc.

3.3 Different types of generalized modules

Physical module and service module are referred together as generalized module. The generalized module can be divided into common modules, mandatory modules, and optional modules according to user permissions in module selection and configuration.

Common module Common modules are those which need not be selected or customized according to different requirements but are necessary and default modules to ensure normal operation and use for a specific PSS. For example, the iron core and coils of a power transformer are designed by a manufacturer according to its optimal design, not customized by customer product warranty service (each module generally has a specific warranty period) or spare parts service. These modules are all basic configuration; we call them the basic modules.

Mandatory module Mandatory modules are those that must be selected by customers, but their instances should be customized according to different customer requirements. They are necessary, but not default, modules for ensuring normal operation and use of the generalized product. For example, seats in a car are mandatory modules, but their materials (leather or cloth) can be chosen by customers.

Optional module Optional modules are those that can be selected according to customer' personalized needs and are additional products or service modules for a generalized product. For example, a GPS navigation system is an optional module for a car.

Moreover, the common modules and mandatory modules are called the necessary modules, which must be configured in the generalized product. The optional modules are called the possible modules, and whether the optional modules are configured depends only on a customer's specific needs.

4 Modeling of generalized product modular master structures

4.1 Modular master structures for generalized products

The modular master structure data model is the most important model for generalized products. After analysis and

Fig. 1 The attribute description of the generalized module

The physical module	Attributes
1	Feature size
2	Material
3	Interface (express the assembly relations)

a attributes of the physical module

The service module	Attributes
1	Activities
2	Executors
3	Resources
4	Time

b attributes of the physical module

The generalized module	Attributes
1	Activities
2	Executors
3	Resources
4	Time
5	Feature size
6	Material
7	Interface (express the assembly relations)

c attributes of the generalized module

research, we find that the integrated product meta-model proposed by Prof. Josef Schoettner can still be used to describe service modules but needs to be improved [23, 26]. As shown in Fig. 2, the integrated product meta-model has four types of master records, which are PaMR (parts master record), DoMR (document master record), DrMR (draft master record), and MMR (model master record) [24–26]. The master records can describe a "parts" perfectly, including not only the physical modules but also the service modules.

PaMR PaMR includes components, parts, semi-finished products, raw materials, supplies, packaging materials, documents, and services [24]. Compared with the PaMR of traditional physical products, the PaMR of generalized products needs to increase service objects and its property descriptions.

DoMR DoMR describes document properties which relate to components and parts, such as documents of product and service orders, requirement specification, NC (numerical control) codes, and other business processes [32, 35].

DrMR DrMR describes the meta-data of engineering drawing and the properties of two-dimensional engineering drawings which relate to components and parts [32, 35].

MMR MMR shows the meta-data of three-dimensional models and describes properties of the three-dimensional model which relate to components and parts [32].

Physical modules are usually described by DoMR, DrMR, and MMR, while service modules are rarely described by drawings and are usually described by documents. Therefore, service modules are usually only described by DoMR.

4.2 Analysis of relationships among generalized product modules

(1) The relationship among physical modules

There are two types of relationships between physical modules in the modular master data structure of generalized products: assembly relationship and matching relationship.

The assembly relationship (close coupling properties among physical modules and very strict assembly relationship between modules): The combinations of structural relation, position relation, assembly dimensions, and assembly size between modules are realized through module interfaces; a complex modular tree structure forms after the assembly of physical modules. As shown in Fig. 4, the "and" relationship is the assembly relationship between physical modules.

The matching relationship: The modular master data structure of generalized products is a configurable structure, through which personalized products can be configured according to a customer's requirements. Mandatory modules and optional modules can be selected by

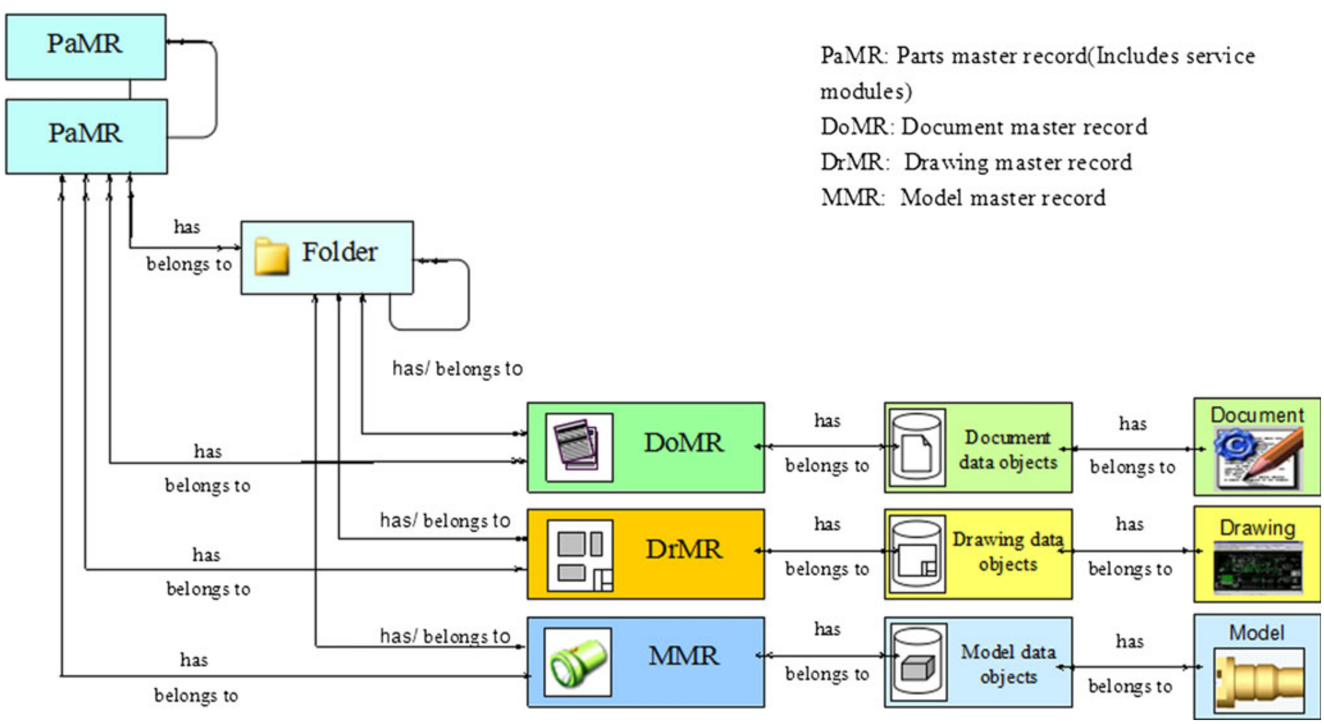


Fig. 2 The integrated product meta-model of generalized product [31, 32]

customers from similar module libraries according to their preferences and needs. For the similar modules, they generally have uniform interface sizes, but the size or functions of the modules are usually different; a customer can only select one module from the same type of module library. The relationships between these modules are matching relationships that are usually expressed through decision tables and configuration rules. As shown in Fig. 4, the “select relation” is a matching relationship between physical modules.

(2) The relationship among service modules

There are two types of relationships between service modules in the modular master data structure of generalized products: inclusion relation and matching relationship.

The inclusion relationship: There are no close relationships such as location and size relations between service modules. The inclusion relations between service modules are generally realized through classification, and the superior-subordinate inclusion relationships between modules are usually established through selection rules.
The matching relationship: In a matching relationship of physical modules, the generalized product modular master structure is a configurable master structure and users select a module to meet demand among functionally identical modules. The relation among these service modules with identical functions is a matching relationship which is generally expressed through the decision rule table.

(3) The relationship among service modules and physical modules

For non-functional service modules, there are no direct coupling relationships with physical modules, but their realization requires the support of related physical modules. Therefore, there is a N:1 ($N \geq 1$) relationship between physical module and functional service module. As shown in Fig. 3, the functional service is described to meet users’ service needs in the modular master data structure of generalized products (this figure omits the configuration rules). Meanwhile, the part of a physical product in the master data structure should be configured with a related physical module to support the realization of a service module. The relationship between a functional service module and a physical module are described through the properties shown by symbols in Fig. 4.

4.3 The composition of the modular master data structure of generalized products

According to different object types, the composition of the generalized product modular master structure can be divided into three categories: (1) the physical module and its business objects; (2) the service module and its business objects; and

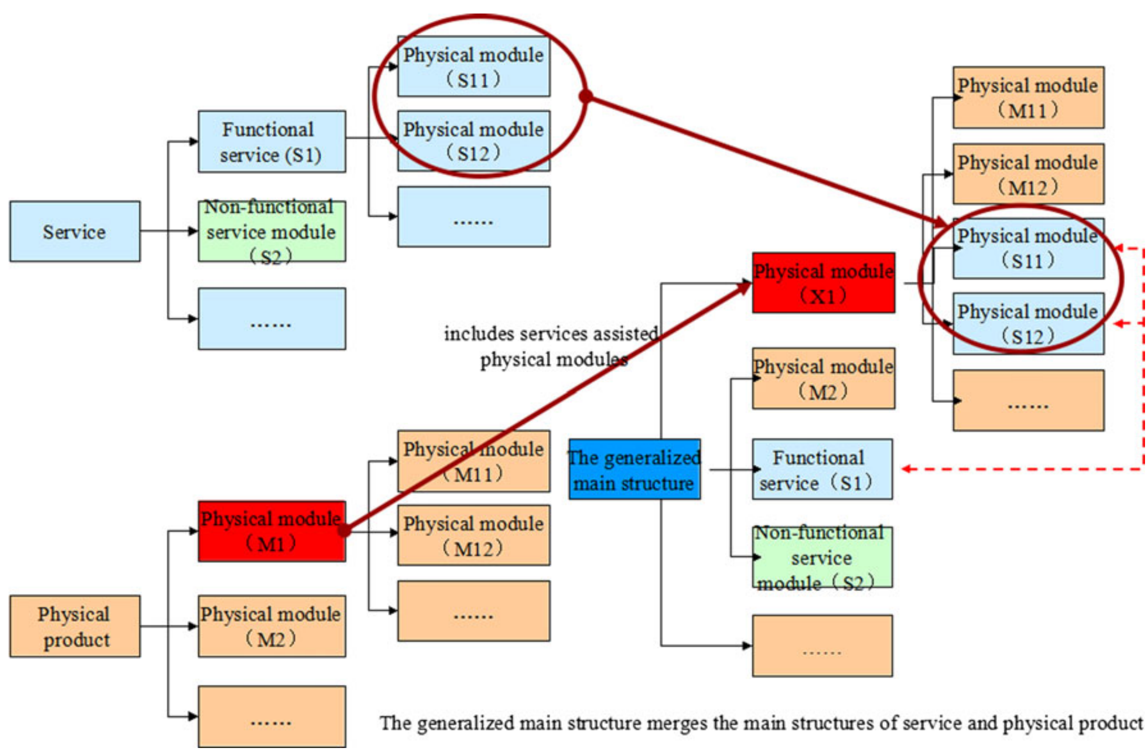


Fig. 3 The relationship between the functional service module and physical module

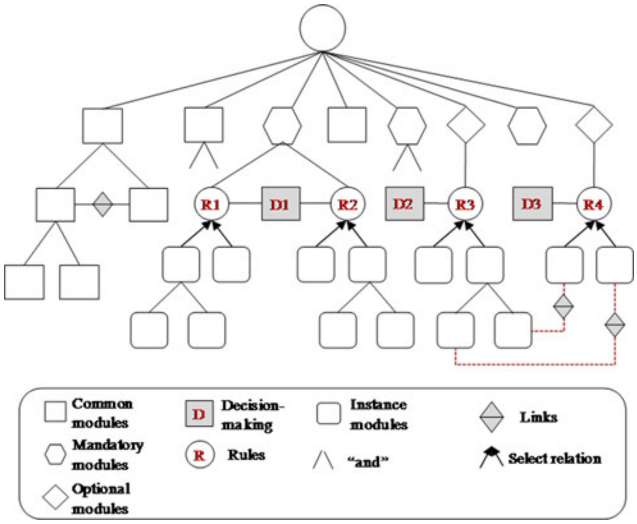


Fig. 4 The modular master data structure of generalized products

(3) the connection objects among generalized "parts" (it includes configuration rules and fixed connection, which respectively reflect in configuration rules (R), decision-making tables (D), and link relationships (◆), as shown in Fig. 4). The connection object describing the relationship between two objects includes the relationship between the two business objects, between the business object and data objects, and between the two data objects [28, 32].

As previously mentioned, according to different permissions for users in module selection, the generalized module can be divided into the common module, mandatory module, and optional module. The modular master data structure of generalized products is composed of generalized modules and the relationships between modules.

The modular master data structure of generalized products is a complex hierarchy tree, and the core of the master data structure is the generalized module. Along the hierarchy tree, following the parent node, are child nodes, as shown in Fig. 4. In the tree structure, nodes represent different levels and granularity of the business object and its connection objects; node types can be divided into the physical module business object, the service module business object, and the connection object between modules. Among the same-level module nodes, connection and select relations between modules can be established through connection objects such as decision-making table and connection relationship; among the different-level module nodes, fixed relations between physical modules and service modules can be established through the connection relationship. In Fig. 4, common modules, mandatory modules, and optional modules (instance modules include mandatory modules and optional modules) can establish a configurable master data structure through rules and connection relationships.

4.4 Property description of the generalized "parts"

In traditional physical product data modeling, the four types of master records are PaMR, DoMR, DrMR, and MMR, which can describe a product or module perfectly [30–32]. But when application objects change from a physical product to a generalized product, research shows the four master records can still describe a generalized product totally, but addition or modification of some properties of the master record is needed. When we describe a service module, the properties of DoMR, DrMR, and MMR can remain the same, but the property of PaMR needs to be modified. The PaMR is the master record of the generalized "parts," containing components, parts, semi-finished products, raw materials, supplies, packaging materials, documents, and services. Inheritance and derivation based on "parts" master records are needed when improving the data model (as shown in Fig. 5); sub-"parts" of the PaMR, such as the physical modules, service modules, supplies, packaging materials, and master record properties of physical "parts," are modified and established respectively. The master record properties of service "parts" are also established.

In the description of physical module property, not only the physical attributes (feature size, materials, and interfaces) are included, but also the special relations with service modules. Similarly, in the description of service module property, not only the service attributes (activity, executors, resources, and time) are included, but the special relations with physical modules are also reflected (as shown in Fig. 5).

4.5 Correlation objects among generalized "parts"

Correlation objects between "parts" in the modular master data structure of generalized products have another connotation; it includes not only assembly relationships and correlation relationships but also the rules between objects in product configuration design. Differentiating them by category, configuration rules are used to limit the combination of parameters while assembly relationships and object correlation relationships belong to the fixed correlation between business objects.

4.5.1 Configuration rules

Configuration relationships among generalized modules (physical modules and service modules) in the modular master data structure of generalized products can be established through constraint rules among modules. As shown in Fig. 6, there are two types of decision-making tables. The first is shown in D1 and the second is shown in D2 and D3. In the first type, the desired module or other types of master records can be selected according to conditions in the rule table. In the second type, there are some constraint relationships between

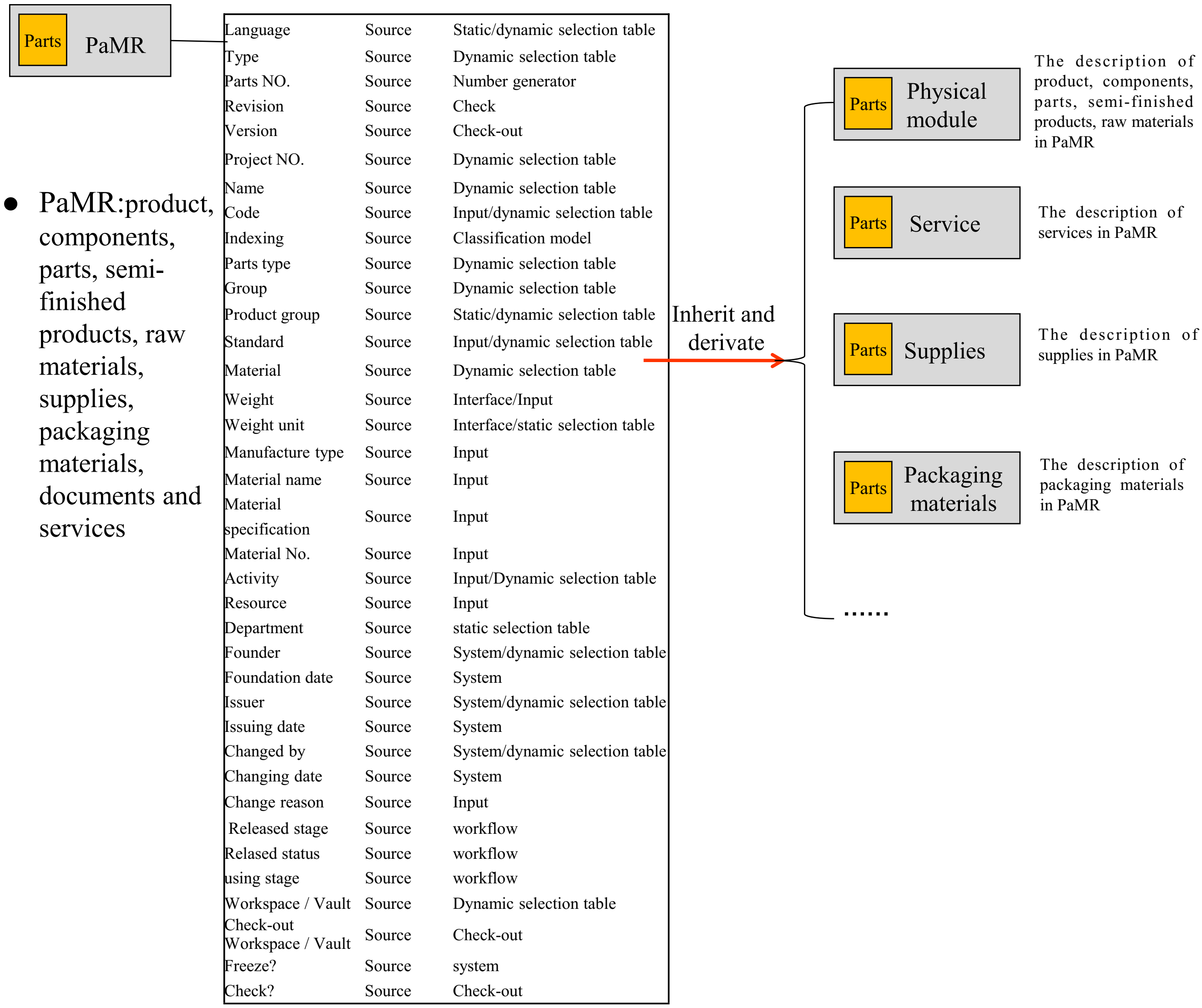


Fig. 5 Inheritance and derivative of master record attributes of generalized parts

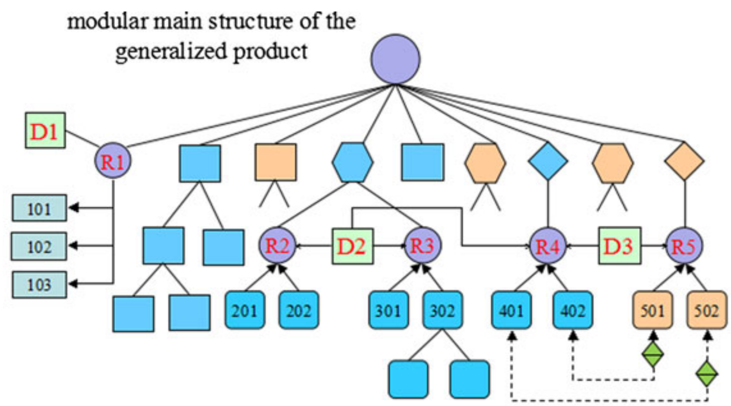
modules, such as the structure position or size; these constraint relationships among modules can be determined by parameter calculation or structure configuration. As shown in Fig. 6, D2 is a decision-making table between physical modules and functional service modules, and users can select functional service modules (501 or 502) according to their needs, and can also select matched physical module R4 (501 matches with 401, 502 matches with 402) according to service structure. Because the selection of 401 and 402 will affect other physical modules (R2 and R3) which have assembly relationships with them, so the type and size of physical module R2 and R3 can be selected and calculated through the decision-making table D2. Therefore, in the process of product modular configuration, the requirement for service modules should be determined first, then the physical modules are configured according to service requirements and service modules. This is the difference between purely physical product design and generalized product design in the product modular configuration process.

4.5.2 The fixed correlation relationship

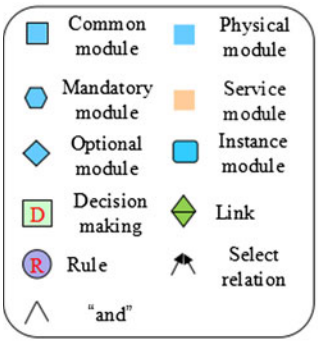
1. Correlation objects of assembly relationships between modules

   (a) Subordinate parts

The properties of correlation objects of subordinate parts mainly include number, materials, etc. and also describe the assembly relationship between superior-subordinate parts. The relationship between superior-



D1	R11	R12	R13	…
term 1	term 11	term 12	term 13	…
term 2	term 21	term 22	term 23	…
R1(No)	101	102	103	…



D3	R31	R32	…
term 1	service type11	service type12	…
term 2	parameter 21	parameter 22	…
R4(No)	401	402	
R5(No)	501	502	…

D2	R21	R22	R23	R24	…
term 1	Structure 11	Structure 12	Structure 12	Structure 12	…
term 2	parameter 21	parameter 22	parameter 23	parameter 24	…
R4(No)	401	401	401	402	…
R2(No)	201	201	202	202	…
R3(No)	301	302	301	302	…

Fig. 6 Modular configuration of generalized products based on configuration rules

subordinate parts is 1:1 or 1:N (N>1, as shown in Fig. 7).

(b) The assembly relationship

The properties of correlation objects in the assembly relationship mainly describe the name, version, and other information of superior parts. The subordinate objects have a 1:1 relationship with superior assembly parts. In Fig. 8, the front PaMR of correlation objects is a physical module, and the back PaMR is a superior part of the physical module. In the properties of correlation objects, the "parts ID" is the ID number of the physical part.

2. Connection object between "parts" objects

Connection relationship of "parts" objects refers to the relationship between service module and physical module. There is a 1:N (N≥1) relationship between the "parts" master record of service modules and the "parts" master record of physical modules. In other words, the realization of service function requirements generally needs one or more physical modules to support. Connection objects in Fig. 9 show the relationship between service module and physical module; the connection object includes the connection object code, the "parts" ID, and other required information of the physical module such as the number, the size, the version, etc.

5 Modeling for the modular instance structure of generalized products

5.1 Analysis of the formation process for the modular instance structure of generalized products

The configuration design process of generalized products is composed of two sub-processes, one is the optimal configuration of the service project and the other is the optimal configuration of the physical project. Thus, the formation process for the modular instance structure of generalized products logically consists of two phases. The first phase forms the modular architecture of the service solution. In this phase, service modules and physical modules will be configured and form a mixed instance structure of service modules and physical modules (component-level structure). The second phase is the formation process of the physical solution. In this phase, related physical modules for service modules and other physical modules (with smaller granularity) will be carried out, so it is mainly a realization process for physical modules.

5.2 Modular instance data structure of the generalized products

Figure 10 is the modular instance data structure of generalized products. The description of product instance mainly includes data records such as PaMR and DoMR. PaMRs are product

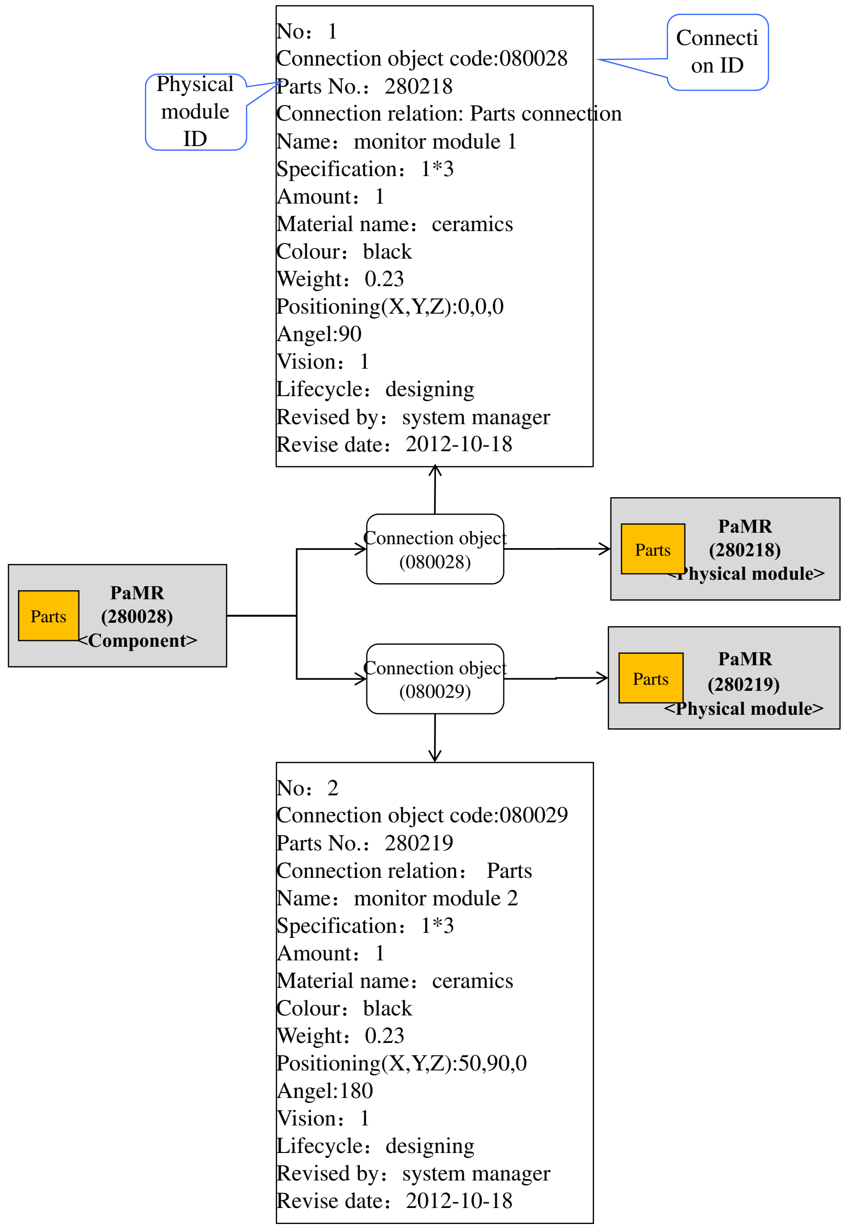


Fig. 7 The relationship between superior-subordinate parts

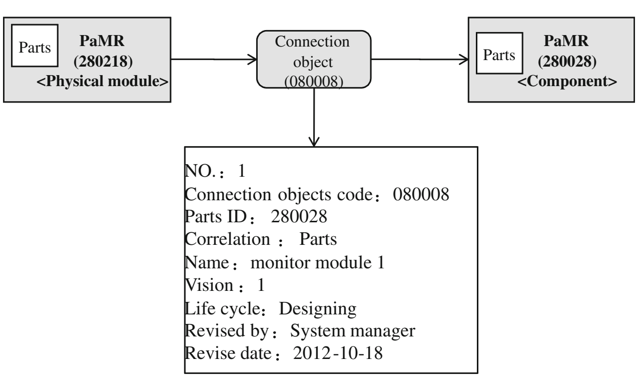


Fig. 8 The assembly relationship object between the module and superior assembly module

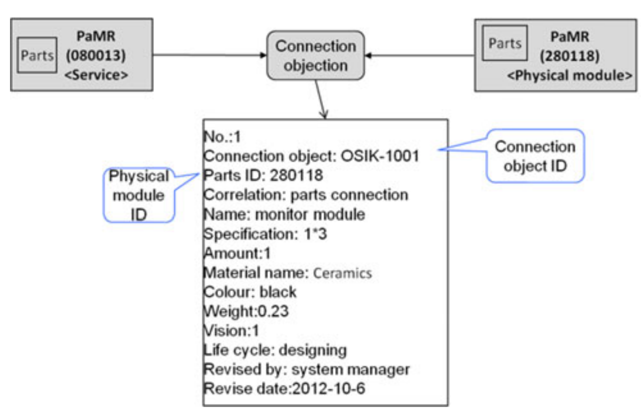


Fig. 9 Connection object between service module and physical module

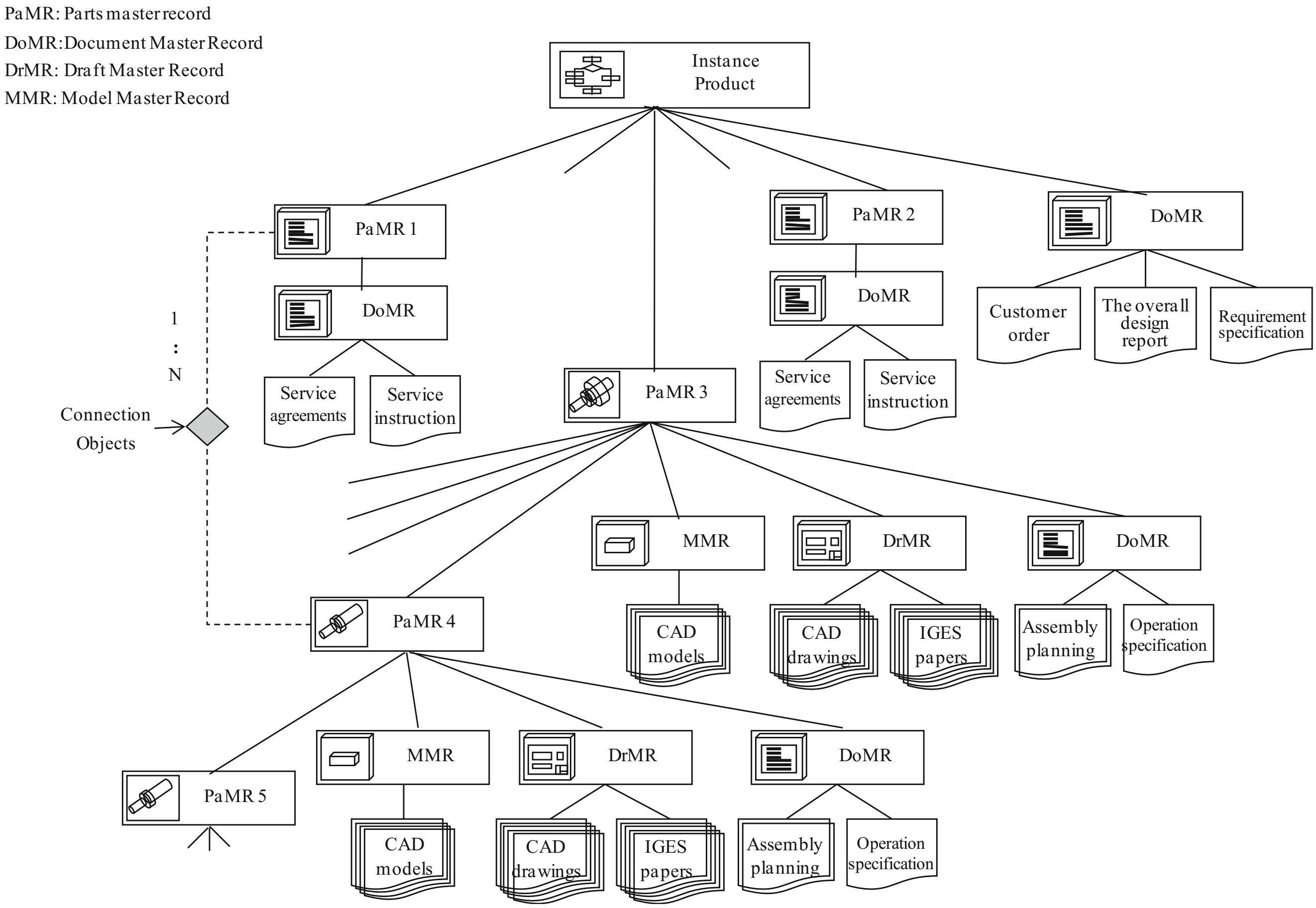


Fig. 10 Modular instance data structure of the generalized product

modules and service modules, and a PaMR is always described by MMR, DrMR, and DoMR. DoMRs are design documents, customer orders, service agreements, and service instructions. For each PaMR, it may have one or more sub-records (a part usually consists of a number of other sub-parts); such PaMR4 is the sub-part of PaMR4 in Fig. 10. In some special circumstances, there are direct relations between PaMRs. In Fig. 10, the need for a functional service module (PaMR1) is realized through the physical module (PaMR4). There is a 1:N ($N \geq 1$) relationship between PaMR1 and PaMR4, and this relationship should be revealed in instance data structure. In the data structure of generalized product instance, service modules are mainly described by DoMR, but some service modules may also need to be described by PaMR, which includes service drawings or three-dimensional model drawings (MMR or DrMR).

5.3 Different view models of the modular instance data structure of generalized products

Figure 11 shows the changing process of instance data structure of generalized products in different view stages. In Fig. 11, it reveals five different views in the product life stages, including development and design, manufacturing, assembly, sales, and service There are also certain relationships between two adjacent views. In each view, there is a product modular structure which is described by circles and lines. The solid line indicates a direct relationship between the modules and the dotted line represents an indirect relationship between modules. There is a mapping relationship for two adjacent views; this means that some modules would change in its special phase—add, merge, remove, replace, and so on. As shown in Fig. 11, in the development and design phase, the relationship between functional service and physical products is a solid line connection, because functional service is achieved by two physical modules; the dotted line means the two functional service modules belong to the physical product indirectly. In the manufacture phase, the relationship between the functional service and physical modules is also a solid line. In the assembly phase, all physical modules are assembled in the physical product, so the dotted line between functional service modules and physical modules changes into solid, and the solid line between the functional service and its sub-modules changes into a dotted line, because there is just one relationship logically. In the assembly phase, non-functional service modules and some random files are added because of product installation and operation in the workplace. In the service phase, some modules or sub-modules are

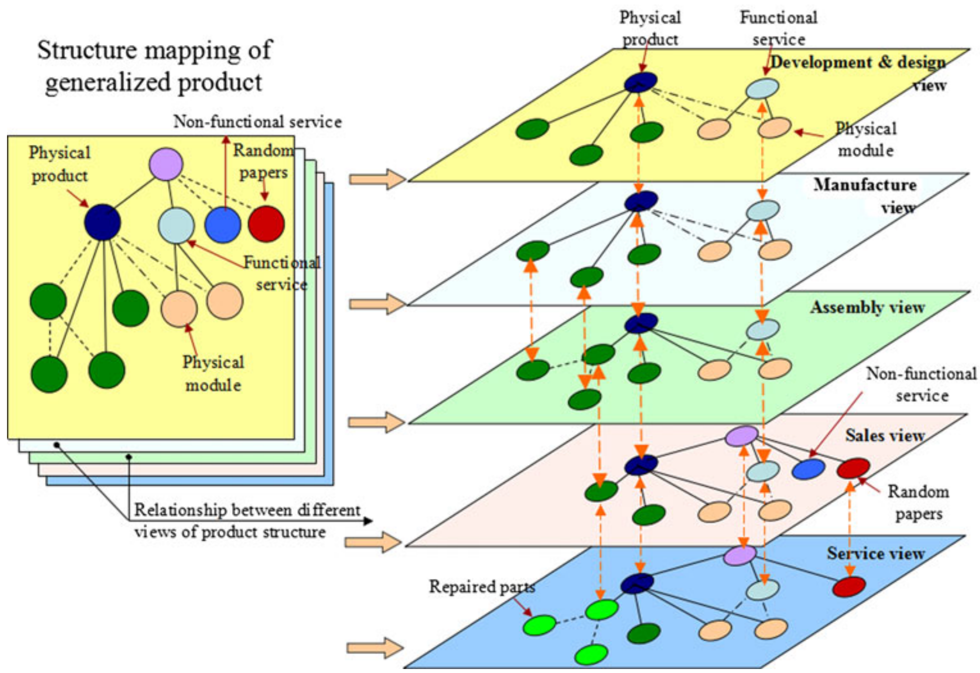


Fig. 11 Five different view mapping relationships of the generalized product

displayed in the modular instance data structure because of replacement or upgrading. Some non-functional service modules disappear in the structure because of the usage of modules.

6 The case study and analysis

The case is carried out based on the secondary development of an Extech product life cycle management (PLM) software system (Extech is the name of a software company); core data models of the generalized product modular structure are developed.

6.1 Property description of "parts" objects

When the "parts" object class is configured in the server of the Extech PLM software system, the properties of service objects need to be configured. It inherits all public properties of the "parts" object class and adds some needed objects. In the client of the Extech PLM system, a secondary development is carried out to realize business objects. The "parts" type is needed to be determined in the "parts" class, and there are two types of "parts" class in the secondary developed object class, which are some service module objects and physical module objects. Service module objects inherit public properties from the "parts" object class and derive some special properties. These specific properties are shown in Fig. 12a. In order to differentiate from the physical module descriptions, "types of objects" is added in service module objects. The difference of properties in the "parts" object class of physical modules between the generalized product and traditional physical products is whether "related service modules" exists in properties. This is used to build connection objects between physical modules and service modules. The property descriptions are shown in Fig. 12b.

6.2 Establishment of connection objects

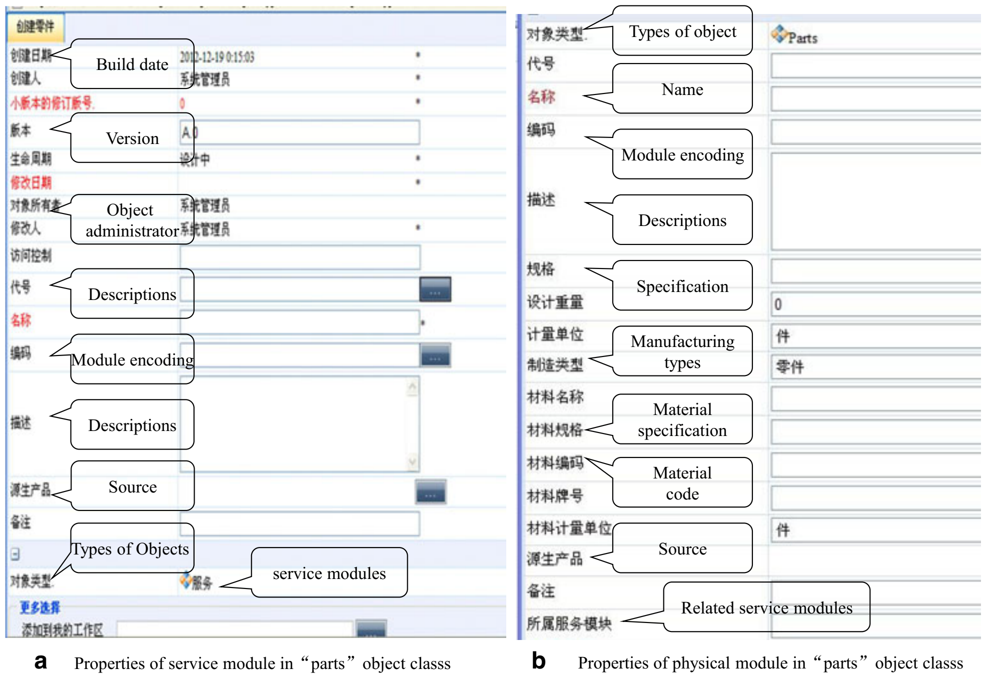
(1) Establishment of connection objects in the server (client/server working model)

In order to establish connection objects, the server side needs to be configured and developed first. Connection objects between objects are configured in the server of the Extech PLM software system. The properties of connection objects between the parent object and the child object are shown in Fig. 13. When establishing connection objects of the "service-related physical objects," the link between the parent object and the child object needs to be built. In Fig. 13, parent objects are service modules and child objects are physical parts. Basic properties of connection objects are inherited, which include name identification of the link, object and type identification of the link, name and solid identification of the parent object, name and solid identification for the child object, the version, serial number, quantity and update date, etc. Definitions of these properties in the server are prerequisite of the secondary development.

(2) Establishment of connection objects in the client

After the establishment of connection objects in the server, the client software also requires a secondary development. Connection objects of the assembly relationship between

Fig. 12 Property description of "parts" objects



modules also need to be developed, and this includes the development of connection objects of subordinate parts and connection objects of the subsidiary assembly (as shown in Fig. 14a, b). Secondary development for connection object relationship between "parts" objects is carried out, and connection objects between physical modules and service modules are established (as shown in Fig. 14c).

6.3 Modular instance structure model of generalized products for the transformer

After system configuration and secondary development are finished, the modular instance structure of generalized products for the transformer can be established (as shown in Fig. 15). The modular instance structure consists of physical modules and service modules, which has independent and complete property descriptions. Among physical modules and service modules, it has independent connection objects. The modular instance structure can express the module composition of the generalized product and the relationships among modules totally.

6.4 The result comparisons and analysis with other models

According to the case application based on the Extech software system, core data models are developed, such as the property description of "parts" objects, establishment of connection objects, and the modular instance structure model. These models can support the management of generalized modules very well.

In this field, the classic product data model is the integrated product meta-model proposed by Schoettner and QI. In comparison, the models proposed in this paper have three valuable aspects.

1) The models proposed in this paper expand the objects from pure physical products to generalized products, which include physical products and service products. The research scope also expands from master structure data models to modular instance structures.
2) The model reveals relationships between the functional service module and physical module and gives detailed descriptions of the service modules and their properties.

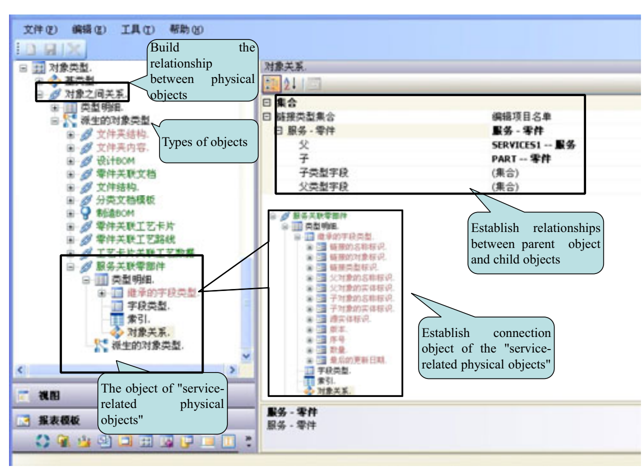
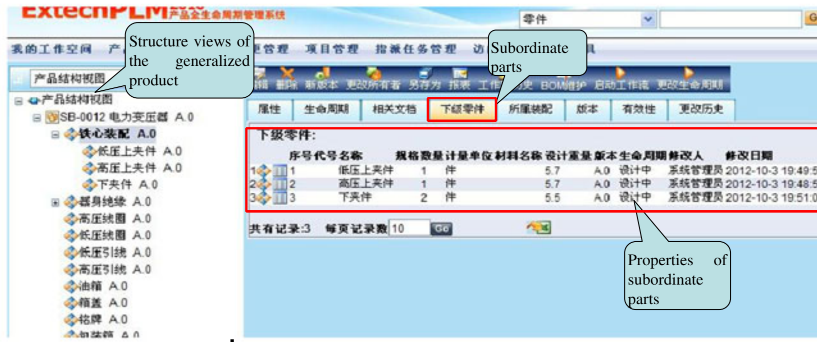


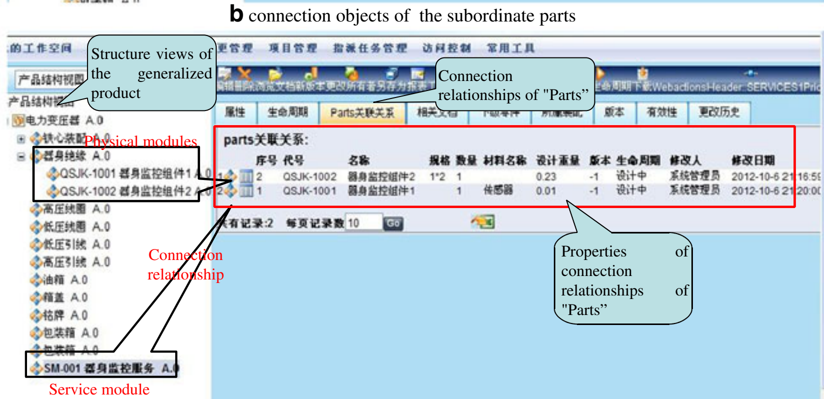
Fig. 13 Properties of connection objects

Fig. 14 Establishment of connection objects for "parts"

a connection objects of the subsidiary assembly



b connection objects of the subordinate parts



c connection objects between physical modules and service modules

3) The relationships among different types of modules, such as the fixed correlation relationship, assembly relationships, and connection object relationship between "parts" objects.

7 Conclusions

In this paper, aiming to meet the requirements of big data management under the environment of cloud manufacturing, modular master structure data models of generalized products are built; these mainly include description and classification of generalized modules, composition of modular master data structure of generalized products, property description of generalized "parts," and connection objects among generalized

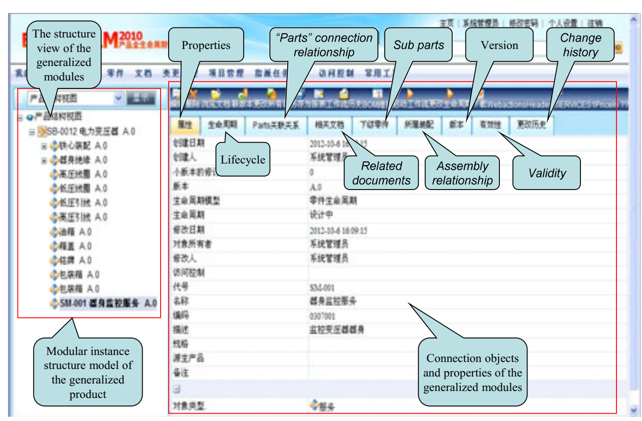


Fig. 15 Modular instance structure model of generalized products for the transformer

"parts." Then, the modular instance structure models of generalized products are built, the formation process is analyzed, and a multi-stage view model of the generalized product is given. Finally, the data model of generalized products and the generalized product modular instance structure of transformers are realized through secondary development of the Extech PLM software system. In conclusion, the traditional integrated product models are enriched and improved to better meet the demand of modular management of generalized products.

It is worth explaining that the property descriptions of "parts" objects and connection objects are the core models and most difficult models in the establishment of the modular data structure of generalized products. These core models have been realized and can explain and support the modeling methods proposed in the paper very well.

Acknowledgments This work has been supported by the National Natural Science Foundation of China (No. 51205372, No. 51275456), the Henan Science and Technology Project (No. 142102210079), and the Research Fund for the Doctoral Program of Zhengzhou University of Light Industry (No. 2013BSJJ032). The authors thank Mr. Yongji Wu and Zilong Wang, who participated in the secondary development of PLM software systems in this paper. Meanwhile, sincere thanks are also due to the Extech software company for providing the PLM development platform.

References

1. Qi GN (2009) Four big pressure ecbolic manufacturing services. Manufac Inform Eng Chin 1:14–15
2. Lee J, Abuali M (2011) Innovative Product Advanced Service Systems (I-PASS): methodology, tools, and applications for dominant service design. Int J Adv Manuf Technol 52(9-12):1161–1173
3. Li H, Ji YJ, Qi GN, Gu XJ, Tang RZ (2010) Connotation, theory and key technologies on the fusion of manufacturing and services. Comput Integr Manuf Syst 16(11):2521–2529
4. Li H, Ma J, Xiao Y Q, Luo G F (2011) Research on generalized product and its modularization process. Proceedings of ICSEM 2011, 22-23 Oct., Guiyang, China, pp290-293
5. Li H (2013) The key technologies of module partition and fusion for the generalized product. Dissertation, Zhejiang University
6. Tao F, Cheng Y, Zhang L, Nee A Y C (2015) Advanced manufacturing systems: socialization characteristics and trends. J Intell Manufac 1-16. DOI 10.1007/s10845-015-1042-8
7. Mont OK (2002) Clarifying the concept of product–service system. J Clean Prod 10(3):237–245
8. Tukker A (2013) Product services for a resource-efficient and circular economy—a review. J Clean Prod 97(15):76–91
9. Zhang F, Jiang P, Zhu Q, Cao W (2012) Modeling and analyzing of an enterprise collaboration network supported by service-oriented manufacturing. Proc Instit Mech Eng Part B: J Eng Manufac 226(9):1579–1593
10. Tao F, Zhang L, Venkatesh V C, Luo Y, Cheng Y. (2011) Cloud manufacturing: a computing and service-oriented manufacturing model. Proc Institution Mech Eng, Part B: J Eng Manufac 0954405411405575
11. Tao F, Zhang L, Liu Y, Cheng Y, Wang L, Xu X (2015) Manufacturing service management in cloud manufacturing: overview and research directions. J Manufac Sci Eng-Trans ASME. doi:10.1115/1.4030510
12. Li J, Tao F, Cheng Y, Zhao LJ (2015) Big data in product data management. Int J Adv Manuf Technol. doi:10.1007/s00170-015-7151-x
13. Aurich JC, Fuchs C, Wagenknecht C (2006) Life cycle oriented design of technical product-service systems. J Clean Prod 14(17): 1480–1494
14. Jiao JX, Ma QH, Tseng MM (2003) Towards high value-added products and services mass customization and beyond. Technovation 23(10):809–821
15. Sakao T, Shimomura Y, Sundin E, Comstock M (2009) Modeling design objects in CAD system for service/product engineering. Comput Aided Des 41(3):197–213
16. Tao F, Zuo Y, Da Xu L, Zhang L (2014) IoT-based intelligent perception and access of manufacturing resource toward cloud manufacturing. IEEE Trans Indust Inform 10(2):1547–1557
17. Tao F, Lai Li Y, Xu L, Zhang L (2013) FC-PACO-RM: a parallel method for service composition optimal-selection in cloud manufacturing system. IEEE Trans Indust Inform 9(4):2023–2033
18. Tao F, Feng Y, Zhang L, Liao TW (2014) CLPS-GA: a case library and Pareto solution-based hybrid genetic algorithm for energy-aware cloud service scheduling. Appl Soft Comput 19:264–279
19. Tao F, Zhao D, Hu Y, Zhou Z (2008) Resource service composition and its optimal-selection based on particle swarm optimization in manufacturing grid system. Indust Inform, IEEE Trans 4(4):315–327
20. Tao F, Cheng Y, Da Xu L, Zhang L, Li BH (2014) CCIoT-CMfg: cloud computing and Internet of things-based cloud manufacturing service system. Indust Inform, IEEE Trans 10(2):1435–1442
21. Tong BS, Li MJ (2000) Product data management. Tsinghua University Press, Beijing
22. Zhang HM, Xiong GL (2006) The product life cycle management in manufacturing enterprises. Tsinghua University Press, Beijing
23. Gou JH, Peng YH, Ruan XY (2000) Product data model in product data management. J Shanghai Jiaotong Univ 34(3):404–407
24. Eynard B, Gallet T, Roucoules L (2006) PDM system implementation based on UML. Mathematics and Computers in Simulation 70(1): 330-342.
25. Saaksvuori A, Immonen A (2004) Product life cycle management. Springer, Berlin
26. Fowler J (1995) STEP for data management, exchange and sharing. Technology Appraisals Ltd., Britain
27. Stark J, Yang QH, Yu N, Li RW (2008) Product life cycle management. China Machine Press, Beijing
28. Yun XD (2010) Research on the analysis and application of integrated product meta model. Dissertation, Zhejiang University
29. Schoettner J, Qi GN (2000) Product data management in manufacturing enterprises. China Machine Press, Beijing
30. Qi GN, Schoettner J, Gu XJ (2005) Figure interpretation for product data management. China Machine Press, Beijing
31. Schoettner J (2009) PDM/PLM seminar in Guilin. Report, Guilin
32. Qi G N, Schoettner J (2010) Integrated product model and its application. Report, Zhejiang University
33. Yu JH (2002) Research and application on virtual product model for product lifecycle. Dissertation, Zhejiang University
34. Gu Q X (2006) Research on key technologies of configuration identification for product lifecycle. Dissertation, Zhejiang University
35. Li X S(2007) Research on the implementation process and integrated product metal model of PLM. Dissertation, Zhejiang University
36. Hu H (2011) Key technologies of equipment maintenance status management for product lifecycle. Dissertation, Zhejiang University
37. Pahl G, Beitz W (1996) Engineering design: a systematic approach. Springer, Berlin

38. Tong SZ (1999) The principle, method and application of modular design. China Standard Press, Beijing
39. Aoki M, AnTeng FK (2003) Modular era: the essence of the new industrial structure. Shanghai Far East Publishers, Shanghai
40. Li H, Ji YJ, Gu XJ, Qi GN (2012) Module partition process model and method of integrated service product. Comput Ind 63(4):298–308
41. Xu QL, Jiao JX (2009) Design project modularization for product families. J Mech Des 131(7):1–10