Research on Development Method of MES based on Component and Driven by Ontology

LONG Wen
Guangdong Provincial key lab of Electronic Commerce Marketing Application Technology
Guangdong University of Business Studies, Guangzhou, Guangdong, 510320, China
Email: lw509@163.com

***Abstract*—Manufacturing Execution System (MES) links plan management and workshop control in an enterprise, which is an integrative management and control system of workshop production oriented to manufacturing process. The paper analyzes the function of MES software and the traditional software development flow as well as the defects of the traditional software development method. To overcome the difficulties of traditional software development method, development of MES based on component and driven by ontology is put forward to prompt development efficiency and performance of MES, which can be more reconstructing, reuse, expansion and integration, and the key technologies of MES development based on component and driven by ontology are investigated in detail. Development method of MES software based on component and driven by ontology is feasible and efficient through developing a pharmaceutics MES which applied in a pharmaceutics manufacturing factory.**

***Index Terms*—MES, component, ontology, architecture, retrieval**

I. INTRODUCTION

With the demand of customers become more and more great for improvement on product, enterprises need adopt more agile manufacturing mode to satisfy customers. Manufacturing Execution System (MES) links plan management and workshop control in an enterprise, which is an integrative management and control system of workshop production oriented to manufacturing process. The traditional MES software causes great development difficulty, long developing period, high cost, low reliability as well as bad reconstructing and integration ability. In view of all defects, the paper puts forward a developing mode of MES software based on component and driven by ontology and makes a research on the key technologies of the development.

II. MES SOFTWARE AND THE TRADITIONAL DEVELOPMENT OF MES

The three tiers mode of enterprise integrating has been put forward by Advanced Manufacturing Research, as shown in Fig. 1 [1]. From the mode we can catch the conclusion that MES is a bridge between plan management and workshop control which fills up the gaps of them. In the one hand MES can decompose the production management information from MRPII/ERP and transform the work instructions to workshop controls. In the other hand MES can monitor the running state of workshop equipments in time and gather the status data of equipments and appearances, after analysis, computing and handling, touch off a new event, then link the control system and information system conveniently and reliably, and feedback the production status to plan system timely.

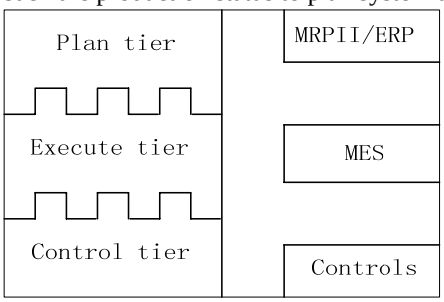

Figure 1. AMR three tiers enterprise integration model

As a computer aided production management system, MES includes a lot of function modules. The primary MES function modules have been concluded by Manufacturing Execution System Association from practice, which include working procedure detailed schedule, resource distribute and status management, production unit distribute, process management, human resource management, maintenance management, quality management, document management, output track and bill management, performance analysis and data collection, as shown in Fig. 2.

From MES function mode we can know that MES software is very great and complex, and exchanges information with many information systems in enterprise.

Traditional development method of MES is the waterfall development method, which separates software development into several steps: requirement analysis, system analysis, software design, software realization, software test, software maintenance.

Fig. 3 shows the traditional software development flow.

doi:10.4304/jsw.5.11.1228-1235

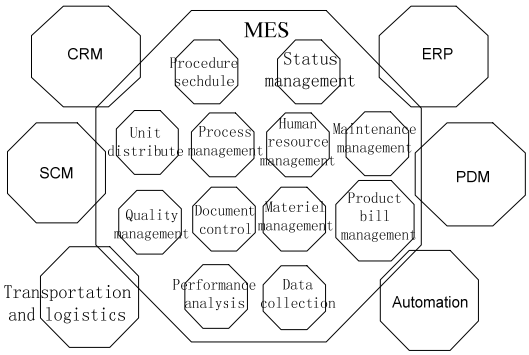

Figure 2. MES function model

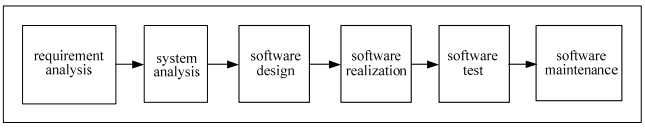

Figure 3. The traditional software development flow

(1) requirement analysis: it includes acquiring the requirement of consumer and definite information as well as the most clear description, such as comprehending business environment and restrictions of consumer, function of the system, performance of the system and the outside system to be compatible.

(2) system analysis: also called domain analysis, which solves what can be done in the system by decomposing the system into components and confirming the logic of the system under the instruction of general layout through detail and embedded research of the system.

(3) software design: including the definition of hardware and software architecture, the design of modules and user interfaces, confirming the performance and the security parameter, the design of database, choice of integrate development environment and the program language.

(4) software realization: according to design instruction to construct system by development team which involve programmer, interface designer and other experts in this phase.

(5) software test: modules and the integrated system will systematic validate to ensure no error and completely accord with demands established in phase (1).

(6) software maintenance: including the correction of the system and modules to promote performance which can be changed by consumer or system bugs.

The traditional software development method exists problems as follows:

(1) In requirement analysis phase, there are many persons such as experts, consumers, system analyzers and development workers, who have different knowledge background and different degree of domain knowledge. The diversity made bad communication of persons with different understanding of the system and cause the different demands.

(2) The system model derived from system analysis is deficient in standardization, conception and formalization, which is difficult to share, reuse and less guide to software design.

(3) Software is made up of modules to form tighten coupling system which is difficult to reconstructing, reuse, expansion and integration, the development of software is long period, difficult and expensive.

MES is a very giant software system, which relates to much knowledge, so the traditional development of MES need expensive fare to develop and maintain and a new software development method should be put forward.

In order to solve the problem(1) and (2) subsistent in the traditional development of MES, domain ontology technology is introduced [2], to solve the problem (3), the reuse technology of software component is introduced too [3]. This paper integrates the reuse technology of software component and the MES domain ontology technology, and puts forward a new development method based on component and driven by ontology, which can improve the development efficiency of MES software and the quality of MES software such as reconstructing, reuse, expansion and integration.

III. DEVELOPMENT OF MES BASED ON COMPONENT AND DRIVEN BY ONTOLOGY

The development method of MES software based on component and driven by ontology is illustrated in Fig. 4.

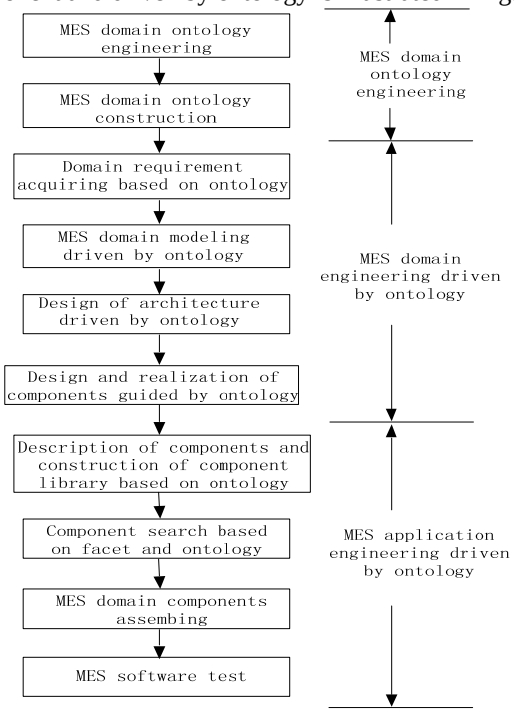

Figure 4. MES development flow based on component and driven by ontology

From Fig. 4 we can know that the development method of MES software based on component and driven by ontology is made up of three major phases: MES ontology engineering, MES domain engineering driven by ontology and MES application engineering driven by ontology. MES ontology is constructed in the phase of MES ontology engineering, MES domain components are developed in the phase of MES domain engineering driven by ontology and MES components are retrieved to assemble MES software in the phase of MES application engineering driven by ontology. When a new domain component should be developed in application engineering phase, the development flow should return to domain engineering phase to develop the domain component and put it into component library. MES ontology plays an important role to prompt the development efficiency and performance in the development process of the MES software.

In this paper, the MES ontology is derived from multi-view MES model, described with OWL and constructed by Protege 2000 editor in the phase of MES ontology engineering. In the phase of MES domain engineering driven by ontology, MES domain analysis method driven by ontology is adopted, MES domain knowledge is described and presented in standardization and formalization making use of domain ontology model, MES domain analysis and domain modeling are carry through driven and guided by MES ontology, a MES architecture is set up based on component and driven by ontology, and MES domain components are analyzed and designed based on ontology. In the phase of MES application engineering driven by ontology, MES components are described base on facet and ontology, a MES domain components library is established, multiply search methods including facet-based and ontology-based to pick up MES components, and the method of assembly and deploy of MES components based on frame is adopted to realize MES software.

IV. Key Technologies of MES Development Based on Component and Driven by Ontology

A. Construct MES ontology with OWL

To improve the efficiency of development and application of MES, ontology is introduced to play an important role in the process of development and application of MES. Ontology is a formal specification of conceptualization in a domain which includes concepts, relations, properties, axioms and instances. MES ontology is the ontology about MES and general knowledge of MES.

1) MES Model

A model is an abstract of actual affair or system, whose expressive form is various, and can be describe by mathematics expressions, physical models or diagrams and words. Enterprise model is description abstracted from enterprise in order to realize enterprise by people. Because of the complexity and uncertainty of practical problem of enterprise as well as the existence of subjective factors of human, more applications are described by diagram models and word models in CIM research.

Enterprise modeling constructs expressive method of practical system from various views orient to problems which enterprise should resolve. Enterprise modeling is a process which achieves an abstract model with a series of steps and definite methods to analyze and simply the practical enterprise object after getting rid of many details which have less influence to modeling motive.

MES is a subset of enterprise information management system, which is an enterprise information management system of workshop and embodies the activities of production and management. So many theories and methods of enterprise modeling can be applied to MES modeling.

The familiar enterprise models are CIM-OSA, ARIS, PERA, GRAI/GIM, GERAM and IEM [4], referenced by enterprise models this paper puts forward a multiple views MES model which core is business process and includes process model, organization model, resource model, product model, information model and function model.

The MES model of multiply views is illustrated in Fig. 5.

(1) Process model. It describes production about project names, process, mechanism of inter function of each activity in the process and exterior relation among activities.

(2) Product model. It describes structure of product, design information of product and correlative document, and embodies synthetically product information.

(3) Organization model. It describes groups, roles, organization structure and privilege of workers around production.

(4) Resource model. It describes the relation of creation, employment and release for resource such as human powers, equipments, tools and materials involved in production.

(5) Information model. It describes the relation of data using by MES.

(6) Function model. It describes the function of MES and confirms the logic structure of function for enterprise business.

The former five modes describe respectively five aspects of business activities around production in the workshop, and function mode is a function embodiment of five modes described formerly. The views of these modes have internal relation which restrict and integrate each other.

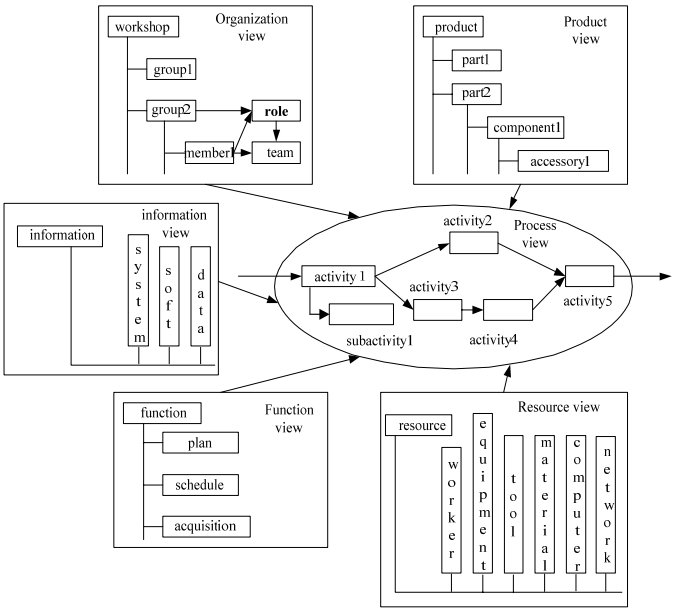

Figure 5 The MES model of multiply views

2) Ontology engineering and OWL

At first ontology is a branch of philosophy, which studies on the essence of being for objective thing. In 1990s ontology is introduced to computer science to overcome lots of problems such as knowledge presentation, information organization and software reuse.

There is a similarity between software development and ontology construct and the process of constructing the ontology is called ontology engineering. To construct an ontology five rules should be complied which are clarity, coherence, extendibility, minimal encoding bias and minimal ontology commitment.

General the method of ontology construct includes six steps: (1) identify purpose and scope, (2) ontology capture, (3) ontology coding, (4) integrating existing ontology, (5) evaluation and (6) documentation.

Ontology is made up of modeling primitives: (1) class or concept, (2) relations, (3) functions, (4) axioms and (5) instances.

Ontology can be described by nature language, frame, semantic network or logic language. With the development of Web a lot of ontology description language based on Web, such as SHOE, XOL, RDF(S), OIL, OIL+DAML and OWL, are applied to application.

Web Ontology language (OWL) is a standard of ontology description language recommended by W3C organization, which is developed based on DAML+OIL and expanded from RDF(S) [5].

The construct functions of OWL are intersection of, union of, complement of, one of, all values from, some valves from, max cardinality and min cardinality, by which classes and properties can be constructed.

In order to describe more clearly the characteristics and relations of classes as well as properties OWL defines axioms as follow: subclass of, equivalent class, disjoint with, same dividual as, different from, subproperty of, equivalent property, inverse of, transitive property, functional property and inverse functional property.

OWL is made up of three sub language such as OWL Lite, OWL DL and OWL Full to meet different demands.

3) Construct MES ontology with OWL

From the MES model of multiply views we can obtain the MES ontology model, then decompose the model into concepts, relations, properties and instances by consulting experts as well as referring to documents.

According to the MES model of multiply views MES ontology can be divided into process ontology, production ontology, resource ontology, organization ontology, information ontology and function ontology. Each ontology can be divided separately into smaller ontology, for example, process ontology can be divided into task ontology, plan ontology, schedule ontology, technical ontology, quality ontology, and so on, and resource ontology can be divided into human resource ontology, equipment ontology, tool ontology, material ontology and so on, as illustrated in Fig. 6.

The equipment ontology is illustrated in Fig. 7.

Fig. 8 shows the properties of equipment and the relations of equipment concept and other concepts.

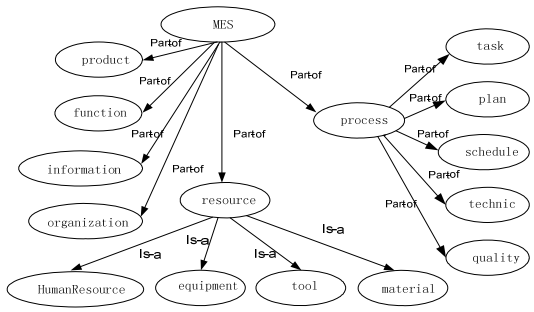

Figure 6. The relations of MES ontology

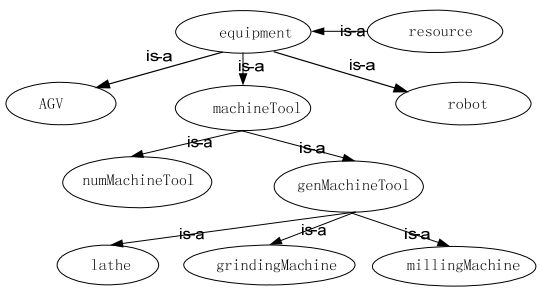

Figure 7. The equipment ontology

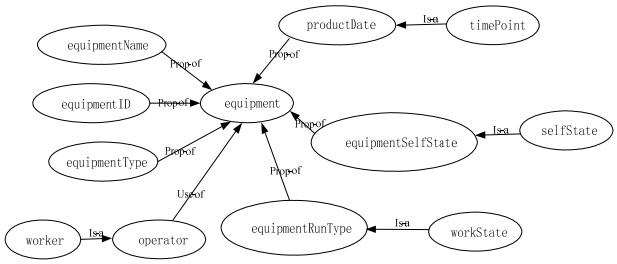

Figure 8. The relations of equipment concept and other concepts

After analyzing the concepts, relations, properties and instances of the MES ontology we can use OWL to describe the OWL ontology and apply protégé 2000 editor to develop the owl ontology [6]. A segment of equipment ontology as part of the MES ontology is described with OWL as follows [7].

```
<?xml version="1.0"?>
<rdf:RDF
   xmlns:rss="http://purl.org/rss/1.0/"
xmlns="http://www.learninglab.de/~dolog/fsm/fsm.owl#"
   xmlns:jms="http://jena.hpl.hp.com/2003/08/jms#"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-
ns#"
   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
   xmlns:owl="http://www.w3.org/2002/07/owl#"
   xmlns:vcard="http://www.w3.org/2001/vcard-rdf/3.0#"
   xmlns:daml="http://www.daml.org/2001/03/daml+oil#"
   xmlns:dc="http://purl.org/dc/elements/1.1/"
xml:base="http://www.learninglab.de/~dolog/fsm/fsm.owl
">
  <owl:Ontology rdf:about=""/>
  <owl:Class rdf:ID="Guard"/>
  <owl:Class rdf:ID="Event"/>
  <owl:Class rdf:ID="Condition"/>
  <owl:Class rdf:ID="Simple">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="State"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Action"/>
  <owl:Class rdf:ID="Transition">
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:someValuesFrom>
          <owl:Class>
            <owl:complementOf>
              <owl:Class rdf:about="#Final"/>
            </owl:complementOf>
          </owl:Class>
        </owl:someValuesFrom>
        <owl:onProperty>
          <owl:ObjectProperty rdf:about="#Source"/>
        </owl:onProperty>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:someValuesFrom>
          <owl:Class>
            <owl:complementOf>
              <owl:Class rdf:about="#Initial"/>
            </owl:complementOf>
          </owl:Class>
        </owl:someValuesFrom>
        <owl:onProperty>
          <owl:ObjectProperty rdf:about="#Target"/>
        </owl:onProperty>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Final">
    <rdfs:subClassOf rdf:resource="#State"/>
  </owl:Class>
  <owl:Class rdf:ID="Initial">
    <rdfs:subClassOf rdf:resource="#State"/>
  </owl:Class>
  <owl:Class rdf:ID="Composite">
    <rdfs:subClassOf rdf:resource="#State"/>
  </owl:Class>
  <owl:ObjectProperty rdf:ID="Entry">
    <rdfs:range rdf:resource="#Guard"/>
    <rdfs:domain rdf:resource="#State"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="Target">
    <rdfs:domain rdf:resource="#Transition"/>
    <rdfs:range rdf:resource="#State"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="fsm_Slot_0">
    <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#Annotatio
nProperty"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="GuardAction">
    <rdfs:domain rdf:resource="#Guard"/>
    <rdfs:range rdf:resource="#Action"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="Exit">
    <rdfs:range rdf:resource="#Guard"/>
    <rdfs:domain rdf:resource="#State"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="Source">
    <rdfs:range rdf:resource="#State"/>
    <rdfs:domain rdf:resource="#Transition"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="TransitionGuard">
    <rdfs:domain rdf:resource="#Transition"/>
    <rdfs:range rdf:resource="#Guard"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="GuardCondition">
    <rdfs:domain rdf:resource="#Guard"/>
    <rdfs:range rdf:resource="#Condition"/>
```

```
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="GuardEvent">
    <rdfs:range rdf:resource="#Event"/>
    <rdfs:domain rdf:resource="#Guard"/>
  </owl:ObjectProperty>
......
  <Condition rdf:ID="Cond1">
  </Condition>
  <State rdf:ID="S1"/>
  <Transition rdf:ID="T2"/>
……
</rdf:RDF>
```

B. The hierarchical architecture of MES based on component and driven by ontology

With the increase of scale and complex of software system the architecture design of system is more important than the choice of arithmetic and data structure, so the favorable architecture is most important for the success of software system. Architecture is the design of whole system structure including the whole frame and control structure, the protocol of communication among components and data access, the distribution of function and so on, and serves as an important basis for software designer to understand components and their effects of software system.

An usual software architecture are made of components, connectors and constrains, namely:

SA＝{components, connectors, constrains}

Components are defined as the carriers of computation, connectors are defined as interactions among components and constrains are defined as the state and condition of components connection.

In modern enterprise the manufacturing mode by distributing and network is the trend of development, so MES should be integrated with other information systems of enterprise and can share information with other information systems timely.

Components-based MES enables application program of customs to access MES database through public medial business logic tiers rather directly. MES components of medial tiers can be exchanged if business logic changes. Information systems such as ERP can access MES database from different tiers to realize communication and interaction with MES from higher tiers [8].

In order to realize the complicate system function and reuse the application systems as well as improve the intelligence of MES, the paper put forwards the hierarchical architecture of MES based on component and driven by ontology which adopt B/S and C/S structure mixed [9].

The hierarchical architecture of MES based on component and driven by ontology which adopt B/S and C/S structure mixed is illustrated in Fig. 9 [10].

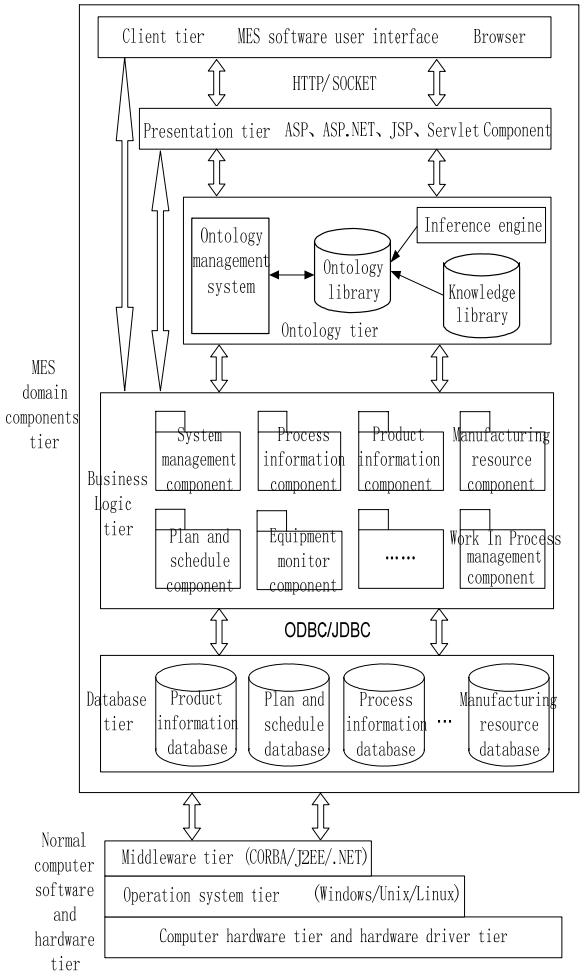

Figure 9. The hierarchical architecture of MES based on component and driven by ontology

The MES based on component and driven by ontology is made up of two great tiers, one is normal computer software and hardware tier and the other is MES domain components tier. Normal computer software and hardware tier is made up of computer hardware tier and hardware driver tier, operation system tier, middleware tier. MES domain components tier is made up of database tier, business logic tier, presentation tier, client tier and ontology tier.

The hierarchical architecture of MES based on component and driven by ontology supply an uniform environment of software and hardware for the development and deployment of MES domain component in which MES domain components can be develop, deploy, add, revise and delete to identify the application environment of domain components and prompt development efficiency, performance of MES, reuse level and system intelligence.

C. Retrieval of component based on ontology

To improve the efficiency of development and application of MES, component is introduced to play an important role in the process of development and application of MES. For the sake of huge component library of MES software the traditional retrieval method of MES software component causes great difficulty such as long retrieval time, low efficiency of retrieval and loss of components. In view of all defects, the paper puts forward a retrieval method of MES software component based on ontology.

1) MES components and MES components library

Traditional MES software is made up of modules to form tighten coupling system which is difficult to reconstructing, reuse, expansion and integration, the development of software is long period, difficult and expensive.

In order to solve the problems subsistent in the traditional development of MES the reuse technology of software component is introduced, which can improve the development efficiency of MES software and the quality of MES software such as reconstructing, reuse, expansion and integration. Mass software reuse needs support of sufficient components which is managed related to description and retrieval method of components.

Components library management system mainly realizes the functions such as storage, retrieval, management, security and maintenance of components. Description, storage, query and retrieval of components are key technologies of components reuse [11].

2) Description of MES components based on facets and ontology

A facet describes a feature of a component and corresponds to several facet values， so a MES component can be described by a series of facets and sub-facets to obtain best understanding of the component， as shown in Fig. 10 [12].

MES Domain Component	
ID:	
Name:	
Reuse Level:	Design Component、Source Code Component、Deploy Component
Sub Donain:	Discrete Manufacturing、Flow Manufacturing、Normal
Component Granularity:	Class、Distributed Component、compound Component
Function:	Manufacturing Resource、Sechdule and Plan、Product Status and Track、Document Control、Technology Management、Quality Management、Data Collection、Report Forms、Others
Development Language:	C/C++、Java、VB、Others
Number of Interface:	<5、5-9、>9
Requirement of Platform:	Windows、Unix、Linux、Others
Type of Component:	CORBA、COM/DCOM、JavaBean/EJB、Web Services、Others
Description of Component:	
Version:	
Developer:	

Figure 10. Description model of MES domain component

It is important to notice that for the steady facets such as using environment and application domain components can easily be described and searched, but for service and function facets of components it bring great influence for precision ratio and recall ratio of component retrieval because the facets is described by non-formal and nature language.

To solve the problem of non-formal description for components, MES ontology and MES components ontology are suggested to be applied for retrieval of MES components.

Ontology is a formal specification of conceptualization in one domain, MES ontology is the ontology about MES and general knowledge of MES, and MES components ontology is the ontology about MES components and general knowledge of MES components.

Fig. 11 shows a sample of ontology of MES components.

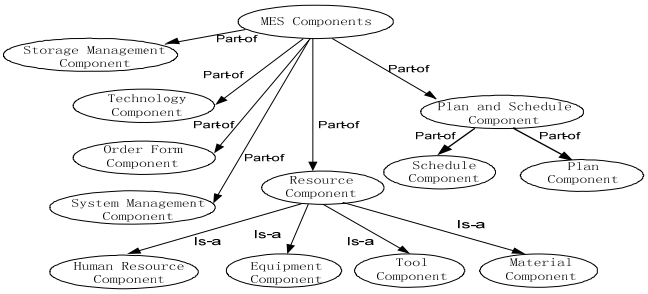

Figure 11. A sample of ontology of MES components

3) Retrieval of component based on ontology

Traditional retrieval method based on keywords may not get components needed and reduces the recall ratio of component retrieval if the query terms do not match keywords whereas they are same semantic [13].

In order to solve the problem of fuzzy and expansibility of query, ontology technology can be applied to components retrieval because domain ontology can reform fuzzy query and expand query scope to improve the recall ratio of component retrieval. Retrieval of MES component based on ontology is illustrated in Fig. 12.

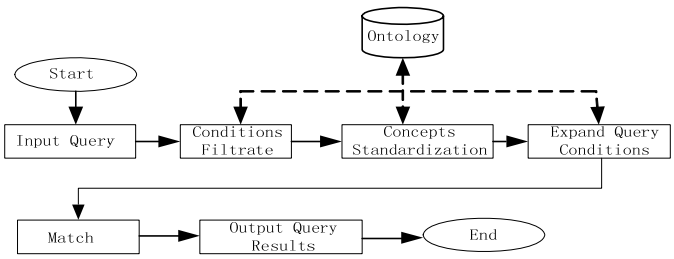

Figure 12. Retrieval of component based on ontology

According to the description model of MES components, function facet is a facet in which term is fuzzy. In MES domain components there are many functions which presence many relations, for example, in the facet of manufacturing resource there is a sub-facet as equipment resource, and in the ontology model, the relation between manufacturing resource and equipment resource is inclusion relation. If a user search a component which

function facet is equipment resource, then he can match a component of manufacturing resource.

In domain ontology the most important items are domain lexicon and domain concept model. Domain concept model describes definitions, relations, rules of each basic conception, and domain lexicon defines the conceptions of domain terms and relations among domain conception. During the search of MES components based ontology the similar relation, inheritance relation, inclusion relation and other relation are mainly applied to retrieval.

Retrieval of component using domain ontology is illustrated in Fig. 13.

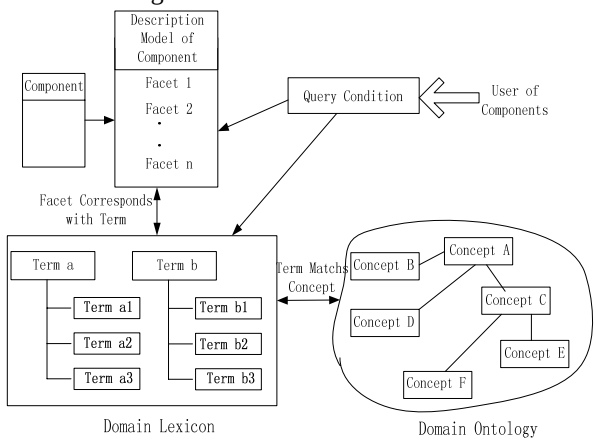

Figure 13. Retrieval of component using domain ontology

V. CONCLUSIONS

Because MES can improve productivity of engineering it is important to improve the development and application of MES. There are many problems in traditional MES software development method such as great development difficulty, long developing period, high cost, low reliability as well as bad reconstructing and integration ability, development of MES based on component and driven by ontology can overcome the difficulties of traditional MES software development method to prompt development efficiency and performance of MES, which can be more reconstructing, reuse, expansion and integration. The development method of MES software based on component and driven by ontology is feasible and efficiency through developing pharmaceutics MES which applied in a pharmaceutics manufacturing factory [14] [15].

REFERENCES

[1] MESA International (1997), MES Explained: The Benefit of MES: A Report from the Fiedl, *White Paper1* （Pittsburgh： Manufacturing Execution Systems Assoc）, 1997.

[2] Nicola Guarino (1998), “Formal Ontology and Information System”. *Proceedings of FOIS’98*, Trento, Italy, 6-8 June 1998, Amsterdam, IOS Press, pp. 3─15.

[3] Yu T. Morton, Douglas A, Troy, “An Approach To Develop Component-based Control Software For Flexible Manufacturing System”. Proceedings of the American Control Conference, Anchorage, 2002, 8, pp.4708─4713.

[4] FAN Yushun, WANG Gang, GAO Zhan, Introduction to enterprise modeling theory and methodology. Beijing: Tsinghua Univ. Press, Springer, 2001 (in Chinese).

[5] Smith MK, Welty C, McGuinness D, (2003). OWL Web Ontology Language guide. W3C Working Draft.

[6] LONG Wen (2005), “Construction and application of ontology for pharmaceutical enterprise based on OWL”. Computer and Applied Chemistry, Vol. 22, No. 9, pp.787─792 (in Chinese).

[7] Peter D. http://www.learninglab.de/~dolog/fsm/fsm .owl.

[8] Ivar Jacobson, Martin Griss, and Patrik Jonsson, Software Reuse: Architecture, Process and Organization for Business Success, Addison-Wesley, 1997.

[9] Ren Shougang (2005). Research on key technology of MES software product line based on component. *Doctor Dissertation of Nanjing University of Aeronautics and Astronautics*, 2005 (in Chinese).

[10] John-Luc Bakker, Harold J. Batteram, Design and evaluation of the Distributed Component Framework for Distributed Communication Architectures, Lucent technologies, Bell labs innovations.

[11] Steven Atkinson, A Formal Model for Integrated Retrieval From Software Libraries, the Proc of Technology of Object-Oriented Languages and Systems (TOOLS 21), 1996, pp.153─167.

[12] R. Prieto-Diaz, Implementing Faceted Classification For Software Reuse, Communication of ACM, 1991, Vol 735, No 5, pp. 89─97.

[13] Bernd Fischer, “Specification-based Browsing of Software Component Libraies”, Journal of Automated Software Engineering, 2000, Vol 7, No 2, 179─200.

[14] LONG Wen (2005), “Research on production plan management of Chinese herbal medicine enterprise”, Computers and Applied Chemistry, 2005, Vol 22, No 8, pp.667─670 (in Chinese).

[15] LONG Wen (2005), “Research and development of ERP system for pharmacy”, Computers and Applied Chemistry, 2005, Vol 22, No 7, pp.517─520 (in Chinese).

LONG Wen obtained his Ph.D. and Master degree in college of Mechanical and Electrical Engineering from the Nanjing University of Aeronautics and Astronautics, Nanjing, China respectively in 2009 and 1995.After his Ph.D., he worked as a research fellow at Guangdong Provincial key lab of Electronic Commerce Marketing Application Technology, Guangdong University of Business Studies, Guangzhou, Guangdong, China. Currently, he is a senior lecturer at GDEC, Guangdong University of Business Studies, Guangzhou, Guangdong, China. He has published around 20 articles in refereed journals and conferences. His research areas include CIMS, MES, ERP, EC and EB.