A Study on Framework and Realizing Mechanism of ISEE Based on Product Line

Jianli Dong
HuaiHai Institute of Technology / School of Computer Engineering, Lianyungang, 222005, China
Email: dongjl1019@sina.com

Ningguo Shi
Lanzhou Resources & Environment Voc-Tech College/President Office, Lanzhou, 730021, China
Email: sng@lzre.gs.edu.cn

***Abstract*—Using product line automatic production procedure and the management system of modern manufacturing industry for reference, a new model of integrated software engineering environment based on product line is put forward, and framework and realizing mechanism of the new model is mainly analyzed in this paper. The new model takes product line core asset (resource) components as the agent bus. The upper of it supports the product line development environment realizing the assembling production of software products, while the lower is traditional common software development environment implementing the development of source codes and documents of product line core resource components. Compared with the present available product line development environment models, the developing and realizing ability of the new one is completely similar to production mode of automatic product line and management system of current manufacturing industry, will be likely to become an ideal software production environment that the future software engineering industry develops.**

***Index Terms*—software product line, core assets, ISEE-integrated software engineering environment, software architecture, software component**

I. Introduction

In recent years, with the gradual mature and application of the new techniques such as software architecture, software components, software large granularity reuse etc., software engineering method based on product line have aroused broad attention of software engineering filed and already become the hot spot and priority of the current studies in software engineering field. The core of software product line studies is to apply the techniques of software architecture, software components, software large granularity reuse and realize the industrialization and automatic production of mass custom-made software products in a specific field, which is similar to product line automatic production modes of modern manufacturing industry products (such as cars , TVs etc.). This is the ideal software production mode pursued for 40 years of software engineering development, which has the very important effect on the formation and development of modern software estate, and will produce the gigantic society and economic effect[1-3].

II. The Current Situation of Software Product Line Studies

Owing to software engineering method based on product line there has been the fundamental change in software development from the traditional one-time, manual programming way, "algorithm + data structure + manpower code", into the industrialized production mode, " software architecture + software components +product line assembly (systematic reuse)", which is characterized by the production system of modern manufacturing industry (such as cars, TV sets etc.) Software product line approach must rely on the achievement of integrated development environment based on product line with the features of product line engineering and the production capacity. Such environment is called integrated software engineering environment based on product line Therefore, the research, realization and application of the integrated software engineering environment based on product line, play a very important role on the automation of software production and the industrialization of the software industry, has become strategic initiatives for the world to occupy the field of information industry, and promote rapid and sustainable economic and social development.

However, we should see clearly that the formation and development of the software product line engineering approach has used product line automatic production mode of modern manufacturing industry for reference based on engineering domain, software architecture, software components and software reuse technology. Its objective is to establish software product line and realize the industrialized production of software products by using the technologies of software architecture of specific field and system-level reuse of large granularity. Obviously, the integrated software engineering environment based on product line is essentially different from the current common software development environment which is structure-based or object-oriented software engineering approach (called the traditional development environment for short). The former is to realize the industrialization and automatic production of mass custom-made software products in a specific field in accordance with "software Architecture + components + assembly" of a specific field, While the latter is in accordance with "data structure + procedures + manual

doi:10.4304/jsw.5.10.1077-1083

encoding" meaning the development of one-time, from scratch software programs, and the general environment also determines the applicable extent of environment, modes of production, productivity and update transformation will be subject to many limitations. Moreover, it is fully different from the production mode and development direction of modern manufacturing industry, as a common product line can not be used to manufacture cars, airplanes, television and other products in different fields.

It is a pity that in the current process of research and development of integrated software development environment based on product line, people haven’t recognized the essential difference between traditional software development environment and modern integrated one. The studies and development of product line software engineering methodologies and integrated development environment is still carried out according to the traditional software engineering methods and ideas, which is not correct. As the CMU / SEI and a number of literature [4-5] have pointed out: so far, there is not the true sense of the integrated software engineering environment based on product line. In other words, the actual status of research and development is that a number of software companies introduce some concepts of components on the basis of its existing software development environment and add the appropriate number of controls, then think that it is transformed into the so-called integrated software engineering environment based on product line. For example, IBM's Rational, SUN's J2EE development environment, etc., are of this type, they are far below the true sense of automated assembly line with production capacity and features of software product line integrated development environment of modern manufacturing industry. Of course, for the interests of enterprises and several decades of software engineering development models, especially the system software and tools for software development process, it is natural and understandable, because enterprises have to take into account their existing products and interests. It is impossible to completely discard the existing large-scale and widely used software products, regardless of the risk to pursue a new product despite the wide range of applications with great prospects and economic benefits.

The paper focuses on a real sense of the software product line development environment model with product line production mode and capacity of modern manufacturing industry, and mainly studies its architecture and the realization of mechanism.

III. Software Engineering Process and Life-Cycle Model Based on Product Line

Essentially, software engineering environment based on product line is a kind of product line which similar to the automatic production line of modern manufacturing industry. It is also a new software engineering method and process to carry out mass customization production of software products in specific domain based on standard component of core resources such as software architecture, component, connecting piece, production plan, specification, constraint, documents and so on. Therefore, what the most important for research on the product line software development environment is to set up software development process model and life-cycle model which suitable for the characteristics and the production methods of product line. It is used to describe the whole process of products development based on product line systematically, and then take this as a guide to determine the message-based application, tool configuration and production process.

Its goal is to describe the sequence of activities, workflow, the task framework, product submission and standards of software engineering process based on product line completely, clearly and specifically. And the guidelines to action and behavioral norms to implement the software product line engineering and software products would be the prerequisite and an important foundation for the research on the integrated software development environment. In recent years, there have been some preliminary research results on the research of product line engineering process model. For example: software product line double life cycle model and SEI model[7-8]. But there simple models can hardly meet the requirement of the whole process expressing ability of modern software management system, mode of producing, evolution of e-Learning, quality control and so on, such as the Multi-level upper and lower layer organization and management system of international, national, industry, domains and application and so on which owned by product line project, the engineering process characteristics and mode of multi-level iterative production methods and the evolution of multi-dimensional product.[13].

On the research and creation of the product line engineering process models and life-cycle, we firstly propose a kind of opened "N-life cycle model" suitable for software engineer based on product line. This model contain the whole process of product line software engineer, each operational phase division of inter process ,the customization of task framework, product quality standards, the entire process of monitoring the completion steps, management and technical characteristics completely. Compared with product line double life cycle model, SEI model and so on, N-life-cycle model, an open process model, which use for reference of the modern industry process and management system and has been proved to be more features and manage spatial of modern industry, meet the product line software project process modeling and expression ability[14].

IV. A NEW ISEE-MODEL BASED ON THE PRODUCT LINE CORE ASSET AND AGENT COMPONENTS BUS

With the research and development of software engineering methodology, the study of integrated software engineering environment has become very active in the field of software engineering. Integrated software engineering environment model is the foundation and prerequisite for integrated software engineering environment .The research purpose is to create the environmental framework and its implementation mechanism, technologies and methods adapted for specific software engineering methods and the development process (life cycle). Now the integrated reference model generally accepted by software engineering field is the three-dimensional integrated model of web-based distributed computing environment proposed by NIST / ECMA, that is, three-dimensional integration of interface, tools and data [9]. However, the research and development of integrated software engineering environment based on the NIST / ECMA model are about the realization of a common software development environment and one-time, from scratch software product development process which is almost all limited to the traditional software engineering methodology. The ability of such an environment model, "Architecture + component + pipeline assembly", can not satisfy the software production environment and development model based on product line, which is the main basis for almost the blank about CMU /SEI's evaluation of software product line development environment.

Integrated software engineering environment model based on product line not only has low-level and source code-level program development capabilities of traditional software engineering environment, but also the basic characteristics such as " field, abstraction, publicity, and scalability, reusability and variability," owned by core resources (the system architecture and components, etc.) included in product line development environment and automated assembly capacity of system-level components characterized by mass custom-made production of product line. This is the nature difference between traditional software development environment and product line development environment, and also the key of the establishment of product line integrated development model.

The research and establishment of product line integrated software engineering environment model must be based on a correct and complete software product line engineering process model and a life-cycle model. Because product line software engineering process and the life cycle model is used to define and describe the production process and demands in the process of software development such as activities sequence, tasks framework, technical methods, management measures, submitted products quality, which is regarded as the guideline and norm of the implementation of software product line engineering and the production of software products, and become the important foundation and the prerequisite for studies of integrated software engineering environment based on product line.

In the current studies of product line engineering process model, there have been some preliminary research results, for instance, the most representative of process models are the dual life cycle model of software product line and SEI (CMU / SEI) model [10-12] However, these simple models are difficult to express and describe the management system, production mode, the evolution of the process, quality control and other features and capacity needs that software product line engineering environment should have, such as international and domestic, industries and enterprises multi-level organization and management system of product line, multi-level iterative production mode and multi-stage products evolution mode. Through the research, in the literature [13-14] an open "N-Life Cycle Model," possessing automatic production features and management mechanism of modern manufacturing industry is proposed, and based on this model, product line integrated software engineering environment model is designed and established.(shown in Figure 1).

As figure 1 shows, product line software engineering environment model is an open and multi-level architecture new model of software engineering environment based on the conceptual model of unified product line engineering, the data model of granularity and reuse of core assets, the behavioral model of components assembly production, the evolution model of the iteration between core assets development and software products manufacture.

The architecture of this new model is essentially a double-environment model taking the core assets component as the bus, the upper is integrated software development environment based on product line supporting software products' automatic assembly production of product line, while the lower is the traditional software development environment based on traditional software engineering methods supporting the development of core assets components' source code and common programs. It is obvious that this new model of integrated environment also includes software production environment and production mode similar to automatic assembly production mode and management mechanism of modern manufacturing industry. It is a real sense of the integrated software engineering environment based on product line.

Figure 1 Note: Filled boxes represent the products of the process or tools or phases. No filled boxes represent the process or tools.

Product Line Interface Integration and Interface Services (interface layer): AT- Analysis Tools Interface, DT-Design Tools Interface, AST-ASsembly Tools Interface.

Standard Engineering Tools: PLSC-Product Line Standard Classification, PLSD-Product Line Standard Design, PLSR-Product Line Standard Release; DA-Domain Analysis Standard, DD-Domain Design Standard, DI-Domain Implementing Standard.

Domain Engineering Tools: DA-Domain Analysis, DD-Domain Design, DI-Domain Implementation

Application Engineering Tools: AA-Application Analysis, AD-Application Design, AI-Application Implementation.

Core Assets Database Platform and Data Integration Services: NSC-National Standard Components, ISC-Industry Standard Components, ESC-Enterprise Standard Components, LAC-Local Agent Components.

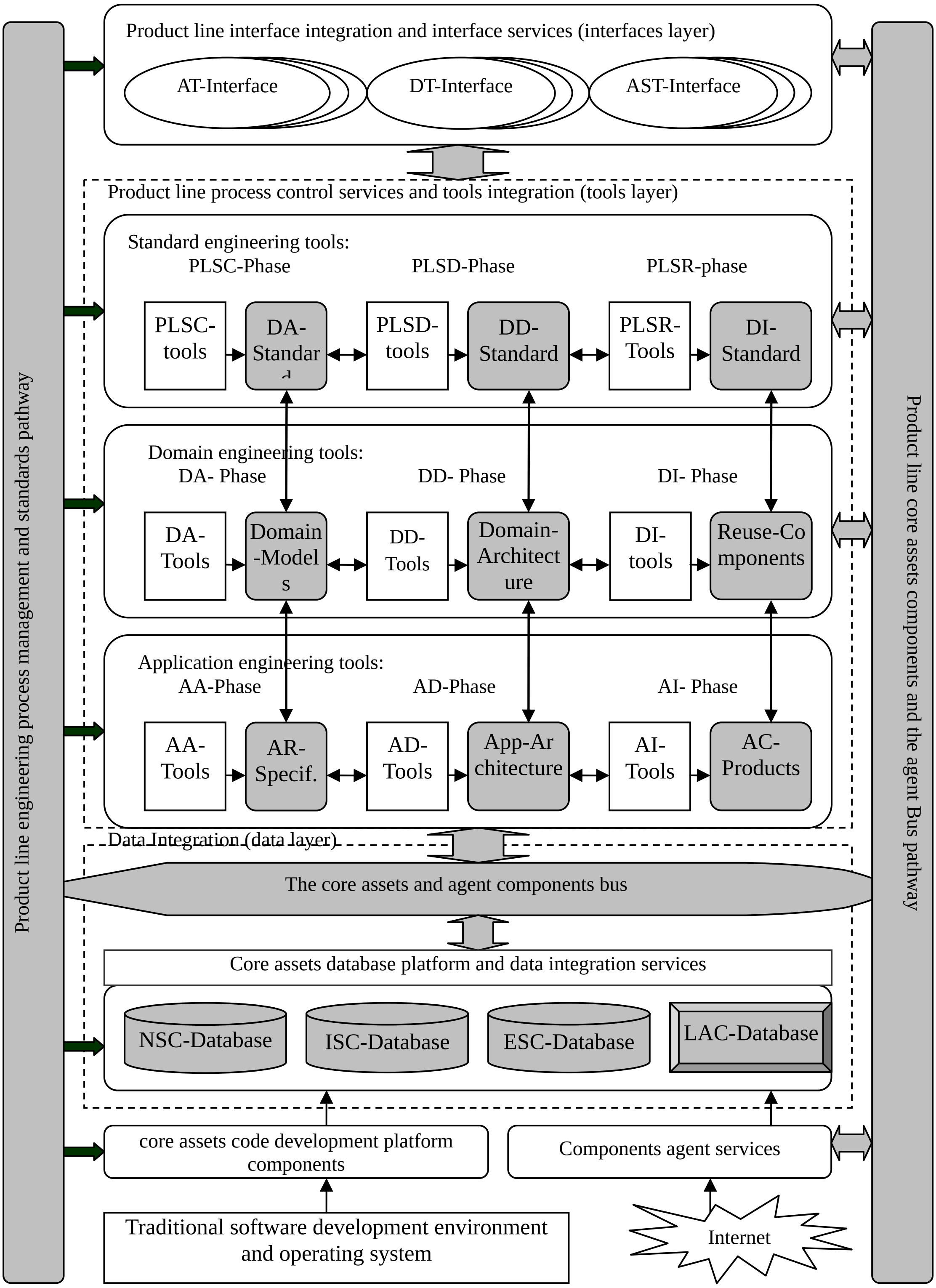
Product line interface integration and interface services (interfaces layer)
AT-Interface
DT-Interface
AST-Interface
Product line process control services and tools integration (tools layer)
Standard engineering tools:
PLSC-Phase
PLSD-Phase
PLSR-phase
PLSC-tools
DA-Standard
PLSD-tools
DD-Standard
PLSR-Tools
DI-Standard
Domain engineering tools:
DA- Phase
DD- Phase
DI- Phase
DA-Tools
Domain-Models
DD-Tools
Domain-Architecture
DI-tools
Reuse-Components
Application engineering tools:
AA-Phase
AD-Phase
AI- Phase
AA-Tools
AR-Specif.
AD-Tools
App-Architecture
AI-Tools
AC-Products
Data Integration (data layer)
The core assets and agent components bus
Core assets database platform and data integration services
NSC-Database
ISC-Database
ESC-Database
LAC-Database
core assets code development platform components
Components agent services
Traditional software development environment and operating system
Internet
Product line engineering process management and standards pathway
Product line core assets components and the agent Bus pathway

Figure 1. A New ISEE-Model Based on Product Line

V. Product Line Core Assets Database and Environment Database Platform

Software product line is a kind of new software engineering method and software development paradigm formed by domain engineering, software architecture, software component and software reuse technology. Software product line mainly consisting of two part of core assets and application products, while core assets includes large complex heterogeneous of domain-specific software architecture, component, connector, production planning, development document, test plan, use case, standard specification and constraint and so on and reusable software production resources. So an important basic research of integrated software engineering environment based on product line is product line core assets and the designed and realized of integrated environment database platform. Its main research contents includes: the study of the data model possess of the ability of core resource data expression and description, product line core assets database schema design, the research of operation mechanism and management ability on product line core assets storage, classification, retrieval, query, version, reuse and optimization and so on. That is, in the demand of capacity, except having the conventional database storage and management capabilities, core assets database should provide rich and reused production resource and convenient resource retrieval, query, reuse, assembly and configuration and so on and management ability for software products customization, assembly and production, this is also the basic goal of the research on product line core assets and environment database system.[15-16]

Currently, the most common and extensive database model is relational data model in the design of product line complex heterogeneous core resource database model, but for the capability defect of relation model such as poor expression ability and semantic fault, the establish of product line core assets database model must adopt object-oriented model that having ultra-intense expression ability and good mechanism, to complete the expression of complex heterogeneous resource data and the demand of modeling; in the design of core assets database, research and establish the multi-view core resource database mode which either has the features of three layer structure of product line architecture style, architecture framework and architecture component or mapping view, reuse degree view and relational view, to realize product line engineering characters and management requirements database, here not only provide management function of core assets such as storage, classification, retrieval, version, optimization and configuration, it also provide good mechanism and methods of data integration, tool integration and interface integration. Furthermore, to consider current network running environment, core assets database platform should offer Internet proxy function to realize localization and network of component integration. These are the basic characteristics and ability that environment database system must have.

VI. The Realizing Mechanism of Product Line Integrated Development Environment

Here it must be pointed out that as shown in Figure 1, the nature difference between the integrated software engineering environment model based on product line containing software development environment and traditional sense of the software (program) development environment is levels of architecture. The architecture of this new model is essentially a double-environment taking the core assets component as the bus, the upper is product line software engineering environment realizing high-level, system-level software products' automated assembly production; while the lower is traditional software development environment realizing low-level, source-level components of product line development.

In the process of the study and realization of software engineering environment model, the standard reference model accepted by the industry is three-dimensional integrated model and the realizing mechanism proposed by NIST / ECMA. In this model, the integrated software engineering environment consists of three aspects of data information including environment interface, environment tools and environment data, which is in accordance with the hierarchical structure of information from top to bottom known as interface layer, tools layer and data layer. Interface layer implements interface integration and management, whose function is receiving user's information and requests, and also dealing with tools call and the returning result data. Tools Layer carries out the integration and management of tools, whose upper part provides services for interface layer and lower part fulfills data access and sharing. Data layer is actually environment database platform realizing the integration, storage and management of the environmental data (resources).

In fact, the realizing mechanism and methods given by NIST / ECMA model is in line with people's day-to-day development, use and operation of computer. The common abstract model can be used for reference in any development and realization of software engineering environment. Only the interface, tools and data of the environment are of different.

As shown in Figure 1, in the study of the realizing mechanism of the integrated software engineering environment model based on product line, we still use the realizing mechanism and methods of NIST / ECMA widely accepted by the current software engineering industry, but interfaces in different environments, tools and data, as well as their behavior and ability demand change fundamentally. As far as the architecture of the new model is concerned, the realization of the integrated software engineering environment on the top of core resources component bus should be based on the provisions and requirements of product line engineering process models and life-cycle model and the management and development tools should be developed and deployed to meet all kinds of projects and internal tasks need at different stages, then the integration of interface, tools and data should be achieved as shown in figure 1 in accordance with the hierarchical structure from top to bottom known as interface layer, tools layer and data layer. [17-20]

Interface layer implements man-machine interface management, whose function is to receive user's operation information and call requests, and also dealing with tools call and the returning result data according to software product line engineering process and production process. The integration of interface should include vertical product line standards, field and application etc. (in Figure 1) and other various projects, as well as the related development tools and resource data interface of different stages of lateral various projects' internal analysis, design and realization etc.

Tools Layer carries out the integration and management of tools, whose upper part provides services for interface layer. In order to achieve tools integration shown in Figure 1, a complete understanding of tool layer standards and vertical iteration, the evolution and constraints of domain and applied engineering of the integrated environment should be possessed, as well as the horizontal iteration, the evolution and constraints of different phases of various projects' internal analysis, design and realization. Thus determines the demand for tools and call between tools, collaboration, communication and control interoperability.

As shown in figure 1, in accordance with product line development process and methods a variety of engineering tools are integrated in tool layer. Among them, the product line engineering should integrate: the standard classification tool of product line, the standard design tools and standard publishing tools etc. The domain engineering should integrate: the field of analysis modeling tools, domain architecture (style, framework and model, etc.) design tools, domain architecture and components implementation tools. Application engineering should integrate: application requirement analysis modeling tools, application architecture and components design tools, application production (components assembly, test analysis) tools, etc. As tool Layer it should ensure the realization of integrated engineering tools, at the same time, it is also necessary to achieve natural integration with data layer and interface layer. That is, tools integration not only ensures call, scheduling, communication, collaboration and interoperability, but also ensures that tools for data access and sharing, and provides users with comprehensive services.

The main function of data layer is realizing integration, storage and management of the environmental data. Data layer is composed of core assets component and the agent bus (including the agent component) as well as core assets component database platform. The component bus provides access interface for tools layer in accordance with a unified concept model and data model to implement tools layer's integration on data layer and have access to the component database and carry out management operations. Component database in accordance with national, industry and enterprise standards is divided into three categories in order to meet production standards demand of different levels of software product line. According to product line engineering process and the characteristics of assembly production, a core resources component and the agent channel have been set up on the right of Figure 1, the main function is to provides a good access mechanism to components database for interface layer and tools layer. Such a mechanism ensures that assembly line production of software products (family) is automatic assembly process and industrial production mode taking standard components as parts. The components bus and the channel is just the conveyor belt to provide standard components. In this product line integrated environment, product line production of software products also means that it is a high-level, system-level software component assembly process, rather than traditional low-level development environment and code-level software programming process.

In the realizing mechanism of environment data integration, the design and the realization of the environment database platform need to be particularly emphasized. The product line core resources mainly includes a large number of complex heterogeneous data resources such as domain-specific software architecture, components, connectors, production plan, development documents, test plans and use cases, standards and constraints etc. Therefore, data model of complex heterogeneous resources, database model of core resources and the design and creation of core resources database platform will become the key to achieve environment data integration. In the design of database model, now the most widely used database model is a relational data model, However, due to defects in the ability of the relational model (such as poor expression, semantic faults, etc.), the design of product line core resources data model must be the object-oriented model with super-strong communication skills and a good control mechanism in order to complete the expression and modeling needs of complex heterogeneous resources data. The designed and created core resources database schema need to have not only product line architecture style, architecture framework, the three-tier organizational structure of architecture components, but also the multi-view core resource database schema including the map view, reuse view, and relational view in order to achieve the characteristics of product line engineering and the database capacity of management needs. For the realization of the core resources database support platform, both management capabilities such as the storage of core resources, classification, search, optimization, version and configuration etc. and the good mechanisms and methods including data integration, tool integration and interface integration need to be provided. [21-23]

In the data layer is the development platform for core assets code-level components and the traditional general software development environment, mainly used to support the development of the source code programs and documents of core resources components. Of course, as product line core resources components, the third-party components can be directly invoked (COTS: Commercial Off-The-Shell) in the software product Line. Therefore, it is necessary to introduce the agent mechanism of components to the local agents and components bus as a special delivery mechanism. The lowest part of the integrated environment is the network environment and the operating system, and it will not go into details here.

VII. Conclusion

In summary, for the realization of product line engineering environment this paper first propose a new product line integrated software engineering environment model whose architecture and realization method are totally different from traditional one based on the recent theories, techniques and methods such as software architecture, components, reuse, domain engineering etc. in order to achieve the industrialization and automatic production of software products. The architecture and the realizing mechanism of integrated development environment based on product line are fully similar to the automatic product line environment and production mode of modern manufacturing industry reflecting a new idea of the research and development of integrated software engineering environment. It represents the forefront of the research and development of modern software engineering methodologies, and has the guiding significance for the formation and development of modern software industry.

Of course, we should clearly find that the studies, realization and applications of integrated software engineering environment based on the product line needed to be carried on compared with automated production line of modern manufacturing industry. We still need to persist and make tireless efforts to do some pioneering work for the early realization of industrialization and automatic production of software products.

ACKNOWLEDGMENT

This project is supported by Fund of Jiangsu University Natural Science Basic Research Project, Grant No. 08KJD520013.

REFERENCES

[1] YANG FuQing, “Thinking on the Development of Software Engineering Technology,” JOURNAL OF SOFTWARE, 2005, 16(1): 1-7.

[2] MEI Hong, SHEN Jun-Rong, “Progress of Research on Software Architecture,” JOURNAL OF SOFTWARE, 2006, 17(6): 1257-1275.

[3] Wang Zhijia, Fei YuKuai, Software Component Technology and Application, BeiJing: Science Press, 2005.4.

[4] Paol.Clements, Linda Nort-hrop, Software Product Lines: Practices and Patterns (SEI Series in Software Engineering), Addison Wesley/Pearson. 2003.

[5] Zhang YouSheng, Software Architecture (first edition), BeiJing: TSingHua University Press, 2004.1.

[6] Paol. Clements, Linda Nort-hrop(America, the Original Author), Zhang Li, Wang Lei(China, Translators), “Software Product Lines: Practices and Patterns”, Tsinghua University Press(Beijing), 2003.

[7] Samuel A. Ajila, Ali B. Kaba, “Evolution support mechanisms for software product line process”, Journal of Systems and Software, Vol81(10), pp.1784-1801,October 2008.

[8] Daniel Mellado, Eduardo Fernández-Medina, Mario Piattini, “Towards security requirements management for software product lines: A security domain requirements engineering process”, Computer Standards & Interfaces, Vol.30(6), pp.361-371, 2008.

[9] Minder Chen, Ronald J. Norman, "A framework for integrated CASE", IEEE Software, 9(2), 1992, 18-22.

[10] Zhang YouSheng. Software Architecture (second edition). BeiJing: TSingHua University Press, 2007.1

[11] Kang KC, “Issues in Component-Based Software Engineering[C/OL],” 99 International Workshop on Component Based Software Engineering, http://www.sei.cmu.edu/cbs/ icse99/ papers/ icse992papers. pdf, 1999, 207-212.

[12] Kwanwoo Lee, Kyo C Kang, Jaejoon Lee, “Concepts and Guidelines of Feature Modeling for Product Line Software Engineering,” In: Proc 7th Int’l Conf Software Reuse. London, UK: Springer-Verlag, LNCS2319,2002:62-77.

[13] Dong Jianli. “Research on software engineering process model based on software product line architecture”, Computer Engineering and Design, 2008,29(12):3016-3018.

[14] Dong Jianli, Jianzhou Wang. “The research of software product line engineering process and it’s integrated development environment model”, ISCSCT-2008 Proceeding, IEEE Computer Society, Vol.1, pp.66-71.

[15] Paul Brown, “Distributed component database management systems”, Component Database Systems, 2001, pp.29-70.

[16] IEEE, Inc. “Information Technology--Guideline for the Evaluation and Selection of CASE Tools (IEEE Std 1462-1988). New York, NY:IEEE Computer Society Press, 1998.

[17] P. Lempp, “Integrated computer support in the software engineering environment EPOS — Possibilities of support in system development projects,” Microprocessing and Microprogramming, Volume 18, Issues 1-5, December 1986, Pages 223-232.

[18] Ian Thomas & Brian A.Neijmeh, “Definitions of tool integration for environments,” IEEE Software, March, 1992.

[19] Brown, A.W. Carney, Ｄ.Ｊ., etc. “Principle of CASE tool integration,” Oxford, U.K.: Oxford University Press,1994.

[20] S Chen, J. M. Drake, W. T. Tsai, “Database requirements for a software engineering environment: criteria and empirical evaluation,” Information and Software Technology, Volume 35, Issue 3, March 1993, P149-161.

[21] M. P. Papazoglou, L. Marinos, “On integrating database modeling constructs for software engineering databases,” Microprocessing and Microprogramming, Volume 27, Issues 1-5, August 1989, P113-120.

[22] Andreas Geppert, Klaus R. Dittrich, “Component database systems: Introduction, foundations, and overview,” Component Database Systems, 2001, P1-28.

[23] Magnus Eriksson, Jürgen Börstler, Kjell Borg, “Managing requirements specifications for product lines – An approach and industry case study,” Journal of Systems and Software, Volume 82, Issue 3, March 2009, P435-447.

Jianli DONG was born in Shanxi province, China, in 1957. He got his Bachelor of Mathematics Science in, Northwest Normal University, Lanzhou, Gansu province, China, in 1988 and got his Master of Software Engineering in Beijing University of Aeronautics and Astronautics, Beijing, China, in 1995. He is now a professor at the School of Computer Engineering in HuaiHai Institute Technology, Lianyungang, China. He has published 40 papers, and completed many reseach projects, and won 6 times scientific and technological progress awards from the provance and military.

Mr. Dong current research interests include software engineering, integrated software engineering enviroment, software architecture, engineering database system, object-oriented Technology.

Ningguo SHI was born in Gansu province, China, in 1965. He got his Bachelor of Science in, Xi,an University of Mining and Technology, xi’an, Shanxi province, China, in 1988. He is now a professor and the president of Lanzhou Resources & Environment Voc-Tech College, Lanzhou, China. His current research interests include software engineering, database system, programming language.