Research on the CORBA Implementation Mechanism of a New Industrialized PL-ISEE

Jianli Dong

Huaihai Institute of Technology / School of Computer Engineering, Lianyungang, 222005, China Dongjl1019@sina.com

Ningguo Shi Lanzhou Resources and Environment Vocational and Technical College, Lanzhou, 730021, China. sng@lzre.gs.edu.cn Yanyan Chen Huaihai Institute of Technology / School of Computer Engineering, Lianyungang, 222005, China chenyanyan306@126.com

Abstract-Borrowing the automation production and management system of modern manufacturing industry production line, a new industrialized PL-ISEE (product line based integrated software engineering environment) architecture was proposed by us in the reference [6]. This kind of new architecture model is a hiberarchy double development environment model with the product line core assets as the software components bus. The upper part on the bus is the software product line development environment to achieve the assembly line production of software products; under the bus is a traditional software development environment to achieve the development of core asset component source program and related documents. For this new PL-ISEE model, its implementation framework and mechanisms as well as methods based on the CORBA are further proposed and studied in this paper. Such the implementation of the new PL-ISEE and own developing abilities are similar to the assembly line and management system of modern manufacturing industry, and it will be an ideal production environment of computer software industry and pattern of software engineering in the future.

Index Terms—software product line, core assets, PL-ISEE (product line based integrated software engineering environment), software architecture, software component bus, CORBA (Common Object Request Broker Architecture)

I. INTRODUCTION

In recent years, with the growing maturity and application of new technologies such as software architecture, software component, large-granularity software reuse and so on, software engineering methodology based on product line, which rises the widespread concern and attention, has become a hot and key topic in software engineering field. The research aim of software product line is, like to the automated assembly line of the modern manufacturing industry products (such as cars, television), to implement the automated and industrialized mass-customized production of domain specific software products by using the software architecture, core asset components and large-granularity system-level reuse techniques. This is the most ideal way of software production in the 40 years software engineering development, which would play a very important role for promoting the development and formation of modern software industry, and would result in enormous social and economic benefits [1-3].

The main research areas of the software engineering based on product line include the product line engineering methodology, engineering processes and life-cycle model, PL-ISEE, software assembly line, DSSA(domain specific software architecture), domain and application engineering, core asset components and its database system, standardization and specification and so on. The paper will research and discuss on the realizing mechanism and method of a new industrialized PL-ISEE based on CORBA (Common Object Request Broker Architecture).

II. RELATED RESEARCHES AND REPROBLEMS

Compared to the traditional software development methods, the software engineering methodology based on product line would make fundamental changes on software development methods and techniques. Software development would gradually be transformed into the industrialization production of "software architecture + components + assembly line" with the producing systems and features of modern manufacturing industry (such as cars, television) from the traditional one-off and starting from ground zero manual programming way of "algorithms + data structures +manual programming". The implementation of software product line production methods must depend on the supporting of product line integrated development environment with the product line engineering features and producing capacities. This kind of supporting environment is known as PL-ISEE (product line based integrated software engineering environment). Therefore, the research, implementation

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

Corresponding author: Jianli Dong. <u>dongjl1019@sina.com</u>

and application of PL-ISEE have very important role in promoting the automation and industrialization of software industry, which have been the strategic initiatives for countries in the world to seize the information industry and promote the rapid and sustainable development of economy and society.

But, we should clearly understand that the formation and development of software product line engineering methodology are on the basis of domain engineering, software architecture, software components and software reuse techniques and completely use the experience of the industrialized producing methods and ideas of modern manufacturing industry product line to build software product lines aimed at reusing the DSSA(domain specific software architecture) and system-level large-granularity final object is to achieve components. The industrialization and automation production of software products. Certainly, the study on the PL-ISEE is just to build an ideal supporting platform or environment for the software's assembling production. Clearly, there are essentially different between the PL-ISEE model and traditional all-purpose ISEE based on structured and object-oriented software engineering methodology. The former achieves the mass customization industrialization and automatic production the domain specific software products according to the method of "domain specific software architecture + components + assembly"; the latter achieves one-off and from scratch software development in accordance with the developing way of "data structure + algorithm + manual programming", and its commonality also determines that the environment's suitability, production mode, productivity, update and transformation would be subjected to many restrictions and differ with producing mode and development direction of the modern manufacturing industry. The reasons are the same as the modern manufacturing industry can not use a common assembly line to achieve the products production in different fields such as cars, airplanes, television and so on.

Unfortunately, the essential differences between PL-ISEE and traditional ISEE are not well understood and known in the research and development of current PL-ISEE as yet. It is not correct to apply traditional software engineering approaches and ideas in developing software engineering methodologies based on product line and integrated development environment. As shown in CMU/SEI (Carnegie Mellon University/Software Engineering Institute) and reference [4-5], there is not a true sense of PL-ISEE so far in the researches and developments of modern ISEE. In other words, the actual PL-ISEE research and development status are that the so-called PL-ISEE are formed by some software companies appropriately introducing the concept of software components and relative control facilities into their original software developing environment. The developing environment such as Rational of IBM, J2EE of SUN, etc. are all belong to this kind of type, and also far below the true sense of PL-ISEE with assembling line production capacities and features like modern manufacturing industries. Of course, analysis from the

benefits of software enterprises and their decades of development modes, especially system and tools software development processes, this approach is natural and understandable because companies must take the existing products and benefits into account. Despite of the wide range of applications and huge economic benefits, it is not impossible to abandon or overthrow the current large-scale and widely applied software products overnight completely to pursue the new products with regardless of the risk.

In this paper, based on a new industrialized PL-ISEE architecture model with assembly line producing mode and capacities as modern manufacturing industry shown Figure.1 in reference [6], we further research the implementing mechanism and method of the new PL-ISEE model according to the CORBA standard and technology. We hope to create a new way for the researches of product line based software engineering methodology and realizing the industrialized production of software products.

III. THE CORBA IMPLEMENTATION OF THE NEW INDUSTRALIZED PL-ISEE ARCHITECTURE

A. A New Industrialized PL-ISEE Architecture Model

With the development of software engineering methodology, ISEE (integrated software engineering environment) has been an active research branch in the software engineering field. However, the research on ISEE architecture and model is the foundation and prerequisite for achieving ISEE. The research purpose is to create the environment framework which fit for a specific software engineering methodology and its development process (life-cycle) as well as ability requirements. Currently, the ISEE reference model generally recognized by software engineering is the three-dimensional integrated model based on network distributed computing environment proposed bv NIST/ECMA (National Institute of Standards and Technology/European Computer Manufacturers Association), which means the three dimensions integration of interface, tools and data [6]. However, the ISEE developed by NIST/ECMA is almost the common software development environment limited to traditional software engineering methodology and one-off, from scratch development process. This kind of environment model can not support the software assembling producing process and development pattern of "architecture + components + assembly line" based on product line, which is also the main reason why the CMU/SEI estimated that software PL-ISEE is almost blank.

Besides having low-level and source code level programs development ability of traditional ISEE, the PL-ISEE should provide the core asset components included in software product line and the assembling ability of these system-level large granularity components as well as the mass customized producing ability of software products. These are the essential differences between the PL-ISEE and traditional ISEE, and the key researches on the PL-ISEE architecture model. For realizing a real PL-ISEE, we have studied and created an open product line engineering process model called "N-Life Cycle Model" with the industrialized production characteristics and management mechanisms as the modern manufacturing industry in reference [7]. In this paper, on the basis of the "N-Life Cycle Model", we have proposed a new industrialized PL-ISEE model shown the Figure 1 in reference [6]. This new industrialized PL-ISEE model is a multi-layer architecture model on the basis of unified product line engineering conceptual model, large-granularity reusable asset data model, components assembly behavior model, iterated evolution model of core assets developing and application software producing.

The framework of this new industrialized PL-ISEE is essentially a hiberarchy double-development environment with core asset components as the software components bus. On the bus, it is the assembly line of the PL-ISEE based on product line software engineering methodology, and realizes the automated and industrialized assembling production of application software product clusters; the asset components needed by assembly line are provided by the software component bus in the producing process of product-line software products; under the bus (or data layer), it is the traditional ISEE (integrated software engineering methods, which support the development of component program and documents of product line core assets.

Obviously, the new model is a real industrialized PL-ISEE architecture with the software assembling production mode and process similar to the automated assembly line and management system of modern manufacturing industry.

In the new PL-ISEE model shown in Figure 1in reference [6], the engineering process managements and controls related software product line would be implemented by "product line engineering process management and standard access" in the left side. This access would provide a comprehensive range of control and management for staff organization, activity sequence, operation behavior, product quality and release configurations of software product line engineering [8]. Managements would include organizational management, project management and quality management. Therefore, this access will provide a good software engineering management mechanism.

The CORBA implementation of the new industrialized

PL-ISEE model will be studied and discussed below.

B. Architecture and Development of the CORBA

OMG(Object Management Group) founded in 1989 has over 830 members now. CORBA(Common Object Request Broker Architecture) is an important standard released by the OMG. Its objective is mainly to enhance the characteristics of good reusability, portability and interoperability of the object-oriented software systems under heterogeneous distributed environment, and build heterogeneous distributed application systems more conveniently. To achieve this objective, OMG developed OMA (Object Management Architecture) reference model, whose architecture is as shown in Figure 1. The core of this model is the ORB (Object Request Broker) which also is called software bus [9-10].

The ORB is the core of CORBA architecture. Its responsibility is to transfer client's requests to the server object and returns the results provided by the server object to the client. It implements the remote communication mechanisms and invoking, scheduling tasks between objects (include client and server end objects), but object's locations, implementations and states are all shielded. ORB, which transfer information between application programs and remote networks under hiding the complexity communication operations of networks, is foundation and core of developing distributed applications based on CORBA.

CORBA is an object interoperability model unrelated to language under the heterogeneous platform. The services on CORBA are described by IDL (Interface Definition Language), and IDL can be further mapped into a programming language such as C + + or Java and distributed in both ends of client and server. In the client-end, it is called IDL Stub; in the server end, it is called IDL Skeleton. Both of them can be described and implemented by different languages. Server-end prepared object implementation on the basis of Skeleton. If customers want to access object methods in server-end, they must send the operation requests to ORB statically through client stub or send dynamically the requests to ORB through interface library and dynamic link library. ORB is responsible for the conversion and request of parameters and sends it to the appropriate object adapter. When object adapter receives the request, it determines whether there is a corresponding skeleton for the requested object implementation. If there is corresponding skeleton, object adapter will call the operations in object implementation library through the



Figure 1. CORBA Architecture

skeleton. Otherwise, object adapter will call the operations in object implementation library through the dynamic implementation routines in DFI (Dynamic framework interface). After executing the service implementation of server object, the results would be reversed back to client according to object request transmission and executed path.

With 20 years development, OMG developed and published criterion cluster--CORBA V1.0, CORBA V2.0 and CORBA V3.0, and defined and introduced the concepts and methods of CORBA object model, interface definition language (IDL), client stub, interface library (IR), object skeleton, Object Adapter, GIOP/ IIOP (General Inter-ORB Protocol/ Internet Inter - ORB Protocol), POA (Portable Object Adapter), CORBA Component model, CORBA Messaging, objects by Value and so on in the cluster. It makes the application and development of CORBA technology very well to satisfy a wide range of requirements of application software development under modern network distributed heterogeneous computing environment.

C. The CORBA Implementation of the New PL-ISEE Model

According to the requirements of PL-ISEE model as shown in Figure 1 in reference [6] and CORBA architecture standards and its function description above illustrated in Figure 1(in this paper), we design an implementation framework of the PL-ISEE model based on the CORBA, and this framework model is shown in Figure 2. In Figure 2, in addition to retaining ORB functions of CORBA, that is, retaining ORB to be responsible for the communication and interoperability between customers and server objects in heterogeneous environments, the component model and component assembly capability provided by CORBA V3.0 cluster could be used. Furthermore, "core asset component and its agent bus" in the PL-ISEE model shown in Figure 1 in reference [6] can be implemented by ORB. In fact, ORB acts a dual role in the framework of PL-ISEE based on CORBA, one is ORB role in original CORBA architecture, and another is product line core asset component and its agent bus which is also called CAB (Component Agent Bus) [9-10].

The essence of PL-ISEE is to realize the assembly production of software products with product line core asset components (such as software components, frameworks, connectors, etc.). Therefore, in the product line development environment shown in Figure 2, the left part is client-end, which is work environment of all developers (such as analysis, design, programming, testing and management personnel) who produce the application software products. The right part is server-end, its main function is to implement the services, organization, saving, distribution and management of software product line core assets in heterogeneous environments and generally complete by component manufacturers or the third-party component providers. The communication and interoperability between client and server end is achieved by object request broker (ORB) and ensure the transparency of the communication and interoperability between the computing objects distributed in heterogeneous environment.

The Figure 2 shows that client environment framework mainly contains various types of development tools using software product line core asset components to assembly application software products, such as product line analysis, design, implementation and management tools. These tools would be integrated on the CORBA architecture according to the three-dimensional integrated model with interfaces, tools and data. It must be emphasized that the integration of client data (core asset components) appears at client end in the way of agent, which only is the registration and identification of components data. And actual components data is all located in the server end which will provide the retrieval,



Figure 2. The CORBA implementing framework of the new PL-ISEE

query, classification and management of component data with corresponding server object. The interface between ORB and client should be expressed by the interface definition language (IDL) provided by CORBA architecture, mainly including CS (Client Stubs), DII (Dynamic Invocation Interface), and IR (Interface Repository). CS is static interfaces provided by object server to client, that is, client local agent of server end object services, and customers call the services provided by service object through these interface. DII (Dynamic Invocation Interface) and IR (Interface Repository) can be used to dynamically call services provided by service objects under running status. On the server end (the right part of Figure 2), object adapter, product line object skeleton. DFI (Dynamic Framework Interface). component object server and component its implementation library, architecture object server and its implementation library, management object server and management assets implementation library and other facilities are configured.

When the client users or developers use the services provided by component, architecture and management object server in server end, they could inquire and access the various services provided by service-end object implementation by two ways. One is that they could send the requests to object implement through client-end CS (Client Stubs) service agents of object implementation. Another is that they can use routines in IR (interface library) and send requests to service-end object implementation by calling DII (Dynamic Invocation Interface). Once object calling requests arrive at ORB, ORB is responsible for the requests and their parameters conversion, and sends the requests to the appropriate object adapter. After receiving the requests, the object adapter determines whether there is corresponding skeleton for the requested object implementation. If there is corresponding skeleton, object adapter call the operation in object implementation library through the skeleton. Otherwise, the object adapter calls the operations in object implementation library through the dynamic implementation routines in DFI (Dynamic framework interface). After finishing the service of object implementation, the results would be reversed back to client according to object request transmission and executing path. Here what must be noted is that object implementation or services of service-end mainly provide the search and acquisition of software product line core asset components and return to client-end developers through ORB in order to achieve the assembling production of product line application software [11-12].

In addition, the implementation of server-end product line core asset objects mainly complete the development, storage, classification and management tasks of the core asset objects(including product line architectures, frameworks, components and so on). These tasks may be completed and provided by different manufacturers. Generally, the product line core asset objects mainly originate from three aspects: firstly, the extraction and reuse of the excellent components in existing system; secondly, new components proposed in the designing of new system; thirdly, suitable modifications and using of components in core assets library. In the management of core asset component objects, standardization and classification of various components directly affects the performance and application scope of whole component database system. The standardization of components requires the consistency of component interfaces and the unified messages format, pattern and protocol. The classification of components can adopt hiberarchy classifying methods to ensure the flexibility of the system and the security and scalability of the assets data transmission.

IV. CONCLUSION

The PL-ISEE architecture shown in Figure 1 in reference [6] is first proposed a new industrialized PL-ISEE model with new techniques and features such as software architecture, components, reuse, domain engineering, application engineering and so on in recent years. Sure, The PL-ISEE developed and implemented by the model will help to achieve the industrialization and automation of software products production. And then, we further research and discuss the realizing mechanism and method as well as framework of the new PL-ISEE model based on CORBA architecture and ORB technology. The CORBA implementation of the new PL-ISEE will own the assembly production and management capacity of application software products like the assembly line of modern manufacture industries. addition, considering ORB as In development environment software bus could carry out the integration and operation of environment parts (such as interfaces, tools, software, etc.) on the software bus in the way of plug and play. This feature can guarantee the adaptability, flexibility and variability of the PL-ISEE.

Certainly, above the new PL-ISEE architecture and its CORBA implementation will be a new ideal ISEE which can realize software industrialization assembling production in future. But, it should be clearly understood that the research, implementation and application of the new PL-ISEE also has a long way to go compared with the automated assembly line of modern manufacturing industry.

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. 2006.11.

- [6] Jianli DONG, Ningguo SHI. "A study on framework and realizing mechanism of ISEE based on product line", Journal of Software, OCTOBER 2010, Vol.5, No.10, p.1077-1083.
- [7] Jianli Dong. "Research on software engineering process model based on software product line architecture", Computer Engineering and Design, 2008, 29(12):3016-3018.
- [8] 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.
- [9] YAN Juan, ZHANG Lifang, CAI Xuqing, "Research and Development of CORBA", Software Guide, Jun, 2009, Vol.8, No.6, pp.41-43.
- [10] CHEN Bo, LI Zhoujun, CHEN Huowang, "Research on Component Model s :A Survey", COMPU TER EN GINEERING & SCIENCE, Vol.30, No.1, 2008, pp.105-109
- [11] LI Yuling, ZHANG Yongmin, "Integrated Environment for the Software Enterprise Based on CORBA", Journal of Henan Agricultural University. Vol.39, No.1, Mar, 2005, pp.116-120.
- [12] HE Tianzhang, WANG Panqing, LI Xiaohui, "Research and Application on CORBA Component Assembly", Science Technology and Engineering, Vol.8, No.1, Jan.2008, pp.84-86.



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 60 papers, and completed about six research projects, and won 7 times scientific and technological progress awards from the province and military.

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



Ningguo Shi was born in Linxia of Gansu province of China on 11, 1963. He graduated from Xi'an Mining Institute majoring in mine survey in Bachelor Degree in 7, 1985. Now he is a president and a professor in Lanzhou Resources & Environment Voc-Tech College. Pro. Shi is one of first leading talents of

Gansu province. Also he is an expert who is entitled to special grants from the National Council of China, an evaluation expert of Ministry of Education for assessing the talent cultivating work in higher vocational colleges. He gets outstanding academic achievements: the first prize in Gansu Science and Technology Progress Award for one time; the Provincial Teaching Achievement prize for one time; the first prize in Science and Technology Progress Award of many provincial departments for two times.

Mr. SHI current research interests focuses on constructing and developing an excellent course--Mine Survey, and computer technology and application, software engineering, database system, information technology, etc.

Yanyan Chen was born in Shandong province, China, in 1972. She got her Bachelor of Computer Science in Southeast University, China, in 1997. She is a lectuer at the School of Computer Engineering in HuaiHai Institute Technology, Lianyungang, China.

Ms. Chen, her research interests mainly focus on software engineering, software testing, software quality metrics and so on fields.