By Soo Dong Kim

Lessons Learned from a Nationwide CBD PROMOTION PROJECT

Sharing insights gained from a large-scale initiative in an Asian context.

Time-to-market and productivity are key concerns in software industry as today's system complexity is sharply increasing. Component-based development is an appealing technology, providing increased productivity and reducing development efforts through larger-grained software reuse. Here, a unique nationwide CBD promotion project is discussed, with explanation and emphasis of the insights and lessons learned from the project. In late 1990s, the Korean Ministry of Information and Communication (MIC), was inspired by market research and promising forecasts on CBD, including the PITAC report [7], to begin a CBD intitiative. After a through investigation and planning, MIC launched a nationwide Component Industry Promotion (CIP) project [4] in January 1999 to promote the development of CBD technologies and commercial-off-the-shelf (COTS) components. The project has been conducted in four main venues: developing CBD core technologies, developing a library of COTS components, promotion and training in CBD, and developing relevant standards.

Different types of organizations have participated and played distinct roles in the project to make it successful. Figure 1 shows the overall CIP organization, which includes governmental offices, committees, consortia, academia, and industry. MIC is the head office for coordinating the diverse activities of the project and manages funds for developing core CBD technology and COTS components. The CBD advisory board under the Software Promotion Division of MIC consists of university professors and industry experts and oversees the CIP project to make recommendations to MIC.

The Institute of IT Assessment (IITA) is a governmental organization that executes the CIP program in the form of specific projects. IITA reviews industry proposals for CBD projects, allocates budgets to selected projects, and evaluates the project results. Most of projects sponsored by IITA are aimed at producing a large number of high-quality COTS components, initially focusing on banking and manufacturing domains. Each project team is a consortium of at least three companies, a mixture of component development companies and component utilization companies. Hence, each team has both producers and consumers of COTS components, to ensure the components produced are immediately utilized by consumer companies in developing component-based applications. Over 100 software companies have been supported by IITA during the first two years.

ETRI, the largest telecommunications and IT research organization in Korea, is responsible for developing core CBD technologies with academia. The first two years of ETRI research focused on CBD process and methodology, implementation platform, and tools. TTA is responsible for the development of component

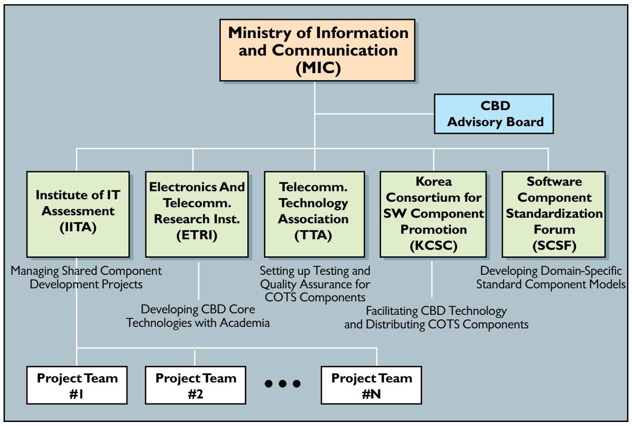

Figure 1. Organization of CIP project.

testing techniques and a quality certification system. TTA developed an ISO-compliant quality model for COTS components, and runs a component certification system that evaluates the quality of components. Hence, all components developed in CIP are to acquire this quality certification.

When the CIP project started, a new consortium on CBD, the Korea Consortium for Software Component Promotion (KCSC), was established to effectively cultivate CBD industry. KCSC consists of more than 150 member companies, each paying a yearly membership fee. The consortium is the center for sharing CBD technologies and experiences, and operates a Web-based system for distributing the components. The main role of the Software Component Standardization Forum is to develop a standardized domain model for certain business sectors, since components should capture the commonality among organizations within a business sector. Currently, an extensive set of component models are available in the areas of banking, credit card processing, manufacturing, e-commerce, ERP, and health care. The overall organization of the CIP has proven to be effective in conducting and harmonizing various managerial and technical activities. To facilitate the distribution and exchange of various artifacts and ideas, KCSC has been organizing seminars and conferences.

Technologies Applied and to be Developed

Developing core CBD technologies is considered the most crucial success factor for the CIP project. ETRI, in collaboration with academia, conducted an extensive survey of component-related research, projects, methodologies, organizations, and standards. Early in the project, the Object Technology Laboratory of Soongsil University conducted an experimental whole-life cycle CBD project with industry participants on a banking application in order to assess the current state-of-the-art CBD technologies [3].

The initial survey revealed the overall CBD technology was still largely immature to apply in industry. The immaturity of available CBD technology surprised all the CIP participants since several software companies were about to begin component modeling and construction for their commercial projects. As an example showing the immaturity, consider modeling variability among family members and customizing components that are advertised as key benefits of CBD. There existed no concrete methods for systematically modeling commonality and variability. It was not explored how the ideal components can be effectively implemented in component platform such as Enterprise JavaBeans.

We defined two technology sets: the first set for technologies currently available and applied to CIP; and the second set for technologies to be developed further in CIP. Hence, participating parties clearly understand what technologies they can utilize currently and what to expect in the future. The first technology set includes domain analysis, OO analysis and design methods, UML standards, Java, EJB, design patterns, and first-generation CBD methods such as Catalysis [2]. Domain analysis techniques were used to identify the essential and common requirements in each business sector. OO modeling was essential for all CBD activities since all the current component reference models have their roots in the OO paradigm. UML was an obvious choice for representing the model. Java was chosen as the implementation language for the CIP-supported projects since the majority of companies in Korea already adopted Web-based architectures for their applications.

EJB was a consequent choice for the CBD implementation platform for most of the CIP-supported projects, although the COM/DCOM platform was adopted in a few projects. It was not intended to designate a specific platform as the standard, but it was inevitable in order to make produced components easily integrated and interoperable. The set of known design patterns was applied to model components and applications to produce well-defined and robust components. Most of the CBD methods proposed by 1999 did not provide a commercial-level process and guidelines for conducting CBD projects. Hence, we utilized only the useful techniques from a small set of initial CBD methods such as Catalysis, Fusion, and ComponentWare.

As the second technology set, we identified technologies to be developed through CIP. The set includes: comprehensive CBD processes; domain analysis techniques for modeling commonality and

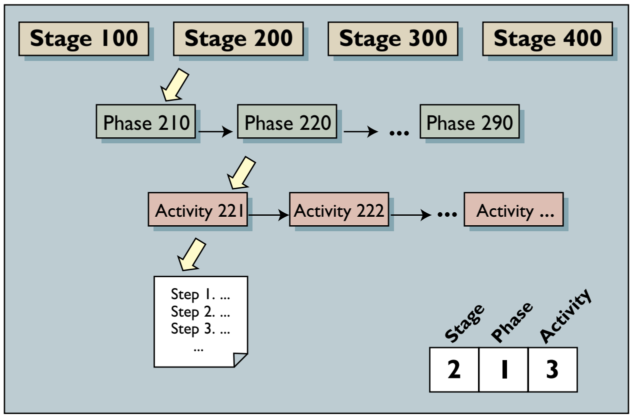

Figure 2. Process architecture of GREEN methodology.

variability; mature component reference models; component identification methods and customization techniques; performance metrics and engineering for CBD; component testing; quality assurance methods; specification and distribution of components and component CASE tools.

Essence of Having a Common Component Reference Model

Components are not just for reuse within an organization, but more for interorganizational reuse. To make components interoperable and allow third-party composition, components should adhere to certain standards and architecture. The very first technical problem encountered for the CIP project was the lack of a standard component reference model. Core technologies, tools, and COTS components should all be based on a single generic and standard component reference model to be practical. Representative CBD models, including CMU/SEI's [1], OMG CCM [6], and EJB differ in their presentation and scope, but they all are based on a conceptually single model. Hence, we defined a generic component reference model reflecting existing component models and architectures. Once this model was defined, we were able to develop further methods, techniques, tools, and standards.

Methodology and Tools

The next focus was to develop a well-defined CBD methodology for software companies participating in CIP. UML is a well-established standard for OO modeling, but alone it does not provide adequate modeling constructs for CBD projects. We conducted a technical comparison of published representative CBD methodologies and found almost all the methods provide activities and instructions only for the initial modeling phase. What is needed by the CBD project team is a commercial-level CBD methodology that covers a whole life-cycle process and provides practical guidelines.

By combining main features of existing methods and adding new techniques, we were able to produce a whole life-cycle CBD methodology. The initial version of the methodology, named FOCUS, was jointly developed in late 2000 by ETRI and Soongsil University. Two enhanced versions of FOCUS were developed in 2001: MaRMI and GREEN. Participating companies are not required to adopt the new methodologies, but these methodologies are widely used in CIP.

Figure 2 shows the overall process of GREEN methodology [5]. The Stage is the largest unit of work and there are four: Stage 100 is to plan the project and comprehend requirements; Stage 200 is to develop components; Stage 300 is to develop applications by using existing components; and Stage 400 is to deploy the component-based applications. Depending on the project goal, relevant stages can be selectively applied. For example, a project to produce

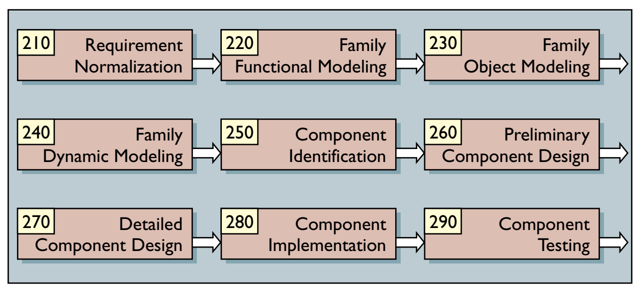

Figure 3. Phases of Stage 200.

COTS components for marketing should only apply Stages 100 and 200. Each stage has a set of phases, and each phase consists of less than 10 activities. Figure 3 shows the nine phases of Stage 200, which is to model and implement components; Phase 210 is to normalize requirements from diverse family members by using a standard glossary of terms; Phase 250 is to identify a set of components from the common domain; and Phases 260 and 270 provide activities and instructions for designing components. We found providing standard templates and forms in addition to the instructions greatly helped project teams to follow the methodology, and the standards are important for CIP-like national projects to produce a set of consistent CBD artifacts.

Use of a good CASE tool is essential to increase modeling effectiveness and to reduce the overall development time. Participating companies were looking for CASE tools designed to support their CBD projects, but most of the early C-CASE tools on the market adapted proprietary component architecture and models and require a runtime support

❁ To increase reuse and interchange of components among companies, it is important to extract a common functionality and behavior among family members of a business sector.

module to run the applications. In addition, the tools did not provide the practical strength and modeling constructs for essential CBD activities. Hence, a new project for developing a C-CASE tool was begun in late 1999; the initial version of the tool is available now for evaluation, and a commercial release should be available soon.

Architectures and Component Platforms. In the past three years, approximately 40 CBD projects were supported by the CIP program, producing a large number of COTS components. In order to effectively share components among organizations, the components from different vendors must be interoperable. However, component platforms are not interoperable with current CBD technology; EJB beans are not interoperable with COM components. Since the CIP project is to promote the reuse of components as much as possible at national level, we had to decide the certain component platforms and architectures on which components are developed and interoperate. EJB and COM/DCOM platforms were chosen for the CIP project.

Attracting Industry for Active Participation. One of the key success factors in the CIP project was to draw a large number of active participants from the industry sector. This is because COTS components would be developed and utilized by them. To increase reuse and interchange of components among companies, it is important to extract a common functionality and behavior among family members of a business sector. To accomplish this, each development project was conducted by a team of members from at least three different companies. One of these participating companies must be the potential client company that will utilize the developed components to build its own applications. In this way, we were able draw active industry participation.

Developing Standard Domain Models. Standardization on domain models is an essential prerequisite for successful development of shared components. Hence, we organized a few standardization working groups to model standard business process and functionality in each business sector. Initially, working groups on finance and manufacturing were active. A major problem in modeling the common business process was that participating companies were not willing to share their domain knowledge and experience acquired through projects over several years. Participating companies demanded some form of reward on providing their domain knowledge. Hence, research organizations and official software promotion agencies are now developing a new pricing and valuing system for components. It is expected that a substantial portion of the component license fee goes to the companies that provided domain knowledge in designing components.

Since components are meant to be reused by many organizations in same business sector, component developers must extract and implement the commonality among the relevant companies (family members). Participating companies presented their own requirement specifications, but an immediate problem encountered in modeling commonality and variability was the lack of standard terms and expressions in the set of requirement specifications. Different words and expressions are used for the same meaning, and different meanings are used for the same expression. Hence, it was unavoidable to normalize those requirement specifications by developing a glossary of standard terms before identifying commonality, which most developers found quite time consuming but essential.

Distribution of Components. During the first two years of the CIP project, participating companies developed a large number of commercial components. A component distribution system was developed in order to register and distribute newly developed components. To allow effective searches for correct components, we defined a template for specifying components, which includes component categories, ID, names, functional description, both provided and required interfaces, customization instruction, persistence requirements, and example client code. New components were documented in this standard specification form and submitted for certification and registration.

In order to manage a large number of components effectively, we defined a standard classification scheme. The classification of the CIP was based on six different views: domain, granularity, abstraction degree, generality, language, and platform, and each component was assigned a unique identification number according to the classification standards. We also found that selecting the correct COTS components is not a straightforward task but a task requiring careful inspection and testing of functional maturity and architectural conformance.

Other Activities in the Asian Region

For the last three years, there have been active exchanges of research results and ideas among Asian countries. The Asia-Pacific Software Engineering Conference (APSEC) is the main conference held in the region, focusing on software engineering issues. In conjunction with APSEC 1999, a workshop on component-based development and software architecture was organized. In addition to APSEC, an annual forum on CBD has been held in the region. The first forum on Business Objects and Software Components (BOSC) was held in Seoul in 2000, the second BOSC forum was held in October 2001 in Tokyo, and the next forum will be in China. This is a major conference specializing in CBD technology in the region. The forum is organized by three consortia: Korea's KCSC, Japan's CBOP, and China's CSCC, and provides an arena for exchanging research advances and industry practice reports. The third conference is scheduled to be held in China this November.

Due largely to the CIP project, CBD is rapidly becoming the main software development paradigm in Korea. Software development projects for government are strongly encouraged to utilize CBD technology, and IT companies have already started CBD projects. The CIP project was challenging initially, but now the regional IT industry is successfully integrating and utilizing the advances and standards of the project. C

References

1. Bachman, F. *Technical Concepts of Component-Based Software Engineering*. CMU/SEI Technical Report 2000-TR-008, May 2002.
2. D'Souza, D. and Wills, A. *Objects, Components, and Frameworks with UML*. Addison Wesley, Reading, MA, 1999.
3. Kim, S.D. *Report on CBD Case Study on Banking Applications*. Project Report, 1999.
4. Korean Ministry of Information and Communication. *Software Component Technology Promotion Plan*. July 1999.
5. Object Technology Laboratory. *GREEN Methodology Specification*. Technical Report, 2001.
6. OMG. *CORBA Component Model. Version 3.0*. June 2002.
7. President's Information Technology Advisory Committee. *Information Technology Research: Investing in Our Future, Report to the President*. Feb. 1999.

Soo Dong Kim (sdkim@comp.ssu.ac.kr) is an associate professor in the Department of Computer Science at the Soongsil University in Seoul, Korea, and an acting steering committee member of the CIP project and KCSC consortium.