

A Series of Development Methodologies for a Variety of Systems in Korea

Jihyun Lee

Electronics and Telecommunications
Research Institute
161 Gajeong-dong, Yuseong-gu,
Daejeon, 305-350, Korea
+82-42-860-1765

jihyun@etri.re.kr

Jin-Sam Kim

Electronics and Telecommunications
Research Institute
161 Gajeong-dong, Yuseong-gu,
Daejeon, 305-350, Korea
+82-42-860-5995

jinsam@etri.re.kr

Jin-Hee Cho

Electronics and Telecommunications
Research Institute
161 Gajeong-dong, Yuseong-gu,
Daejeon, 305-350, Korea
+82-42-860-6491

chojh@etri.re.kr

ABSTRACT

To meet the development condition of inside of the country, domestic development methodologies are made with the abbreviation of MaRMI (Magic and Robust Methodology Integrated) in a series of methodologies in South Korea. The MaRMI have the four different methodologies for developing information, object-oriented, component-based, embedded systems. In this paper, the authors describe the feature and structure of each methodology and show our movement for the methodology transfer.

Categories and Subject Descriptors

D.2.9 [Software Engineering]: Management – *life cycle, software process models.*

General Terms: Documentation, management.

Keywords: Methodology, process, system development, MaRMI.

1. INTRODUCTION

In the environment of software system development, problems related to productivity and quality have issued. To solve these problems, the developed countries have researched system development process for a long time. And they have systemized one development model and applied to real industry. Although software system can be developed without practical adoption and use of systematic development process, the speed of technology growth and the need from software markets make maintenance for previous software complex and labor cost high. The industries of South Korea have also encountered the problems such as an excess of development budget, delay of development schedule, low productivity, and unsatisfactory productivity. As an effort for these domestic problems, development methodology is suggested as a real solution. Development methodology is a collection of method, policy, procedure, rule, standard, technique, tool and language for analyzing, designing, implementing and testing software systems. Hence, South Korea started to adopt development methodology, which was developed from outdoor countries.

From the late 1980's, the adapted development process started to be applied to support development of indoor information systems. From the industry expansion in the field of system integration in 1990's, the activity for process standardization was accelerated and each business company started intra standardization satisfying business characteristics of its own. But the adopted methodology was inadequate for indoor environment so that a series of bottlenecks were raised to establish standardized process.

Examples of the bottleneck are summarized: the difficulty for expansion, modification, and coordination, the insufficiency of detailed guideline, the ambiguity of translation into Korean, the cultural inadequacy, and the need of consulting.

To solve these difficulties, software engineering research team of Electronics and Telecommunications Research Institute of South Korea has generated MaRMI (Magic and Robust Methodology Integrated) development methodology.

MaRMI is a series of methodology to help domestic system development, which organizes unsystematically spread and pragmatic concept, method, activity, procedure, technique and knowledge acquired by experience into well-organized framework. MaRMI-I, the first methodology, was generated to develop information engineering system in 1997. After that, MaRMI-II was developed for the purpose of developing information and communication system in 1998. MaRMI-III, the methodology with basis on component engineering concept, was developed in 2003. Currently, the product-line based MaRMI-EM methodology is in the process of development for embedded system areas. MaRMI has propagated to domestic development companies, system audit organizations, universities, etc.

In this paper, we will describe a series of development methodologies, MaRMI. The overall characteristics of MaRMI and general organization will be firstly described in section 2. The compared results for the features of MaRMI are described in section 3. In the section 4, we will finally represent conclusion.

2. THE OVERALL CHARACTERISTICS OF MaRMI

Until 1994, South Korea has been entirely dependent on Method/1 development methodology, adopted foreign country. At the end of 1994, MaRMI development project was started to secure technology of our own for development methodology in South Korea. MaRMI development methodology defines

systematically the technologies, which are fragmentarily utilized in software development. MaRMI has origin of phonetic meaning of cutting out in Korean.

As shown in figure 1, the development methodology covers all activities of development and maintenance, which are related to system design and maintenance, development process and method, techniques, and supportive tools for life cycle.

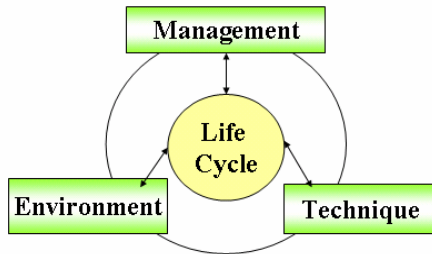


Figure 1. The concept of development methodology.

From our survey, users responded that the use of methodology should be easy. For consistency and easy use, a series of MaRMI share taxonomies, meta-models, artifacts, and project management activities of existing methodology and provide more detailed procedures than other methodologies. Therefore, industrial company which adopts this development methodology at first can develop systems without big trail-and-error. By continuous acceptance of the requirements from indoor developers, MaRMI overcomes difficulties and provides feedback on the new requirements from methodology users. On the contrary, many methodologies, which are expensively adopted from outdoor, do not improve with Korean user's requirements. MaRMI provides circumstances which share development knowledge from indoor individual developers or business companies.

2.1 The CHARACTERISTICS AND ORGANIZATION OF MaRMI-I

MaRMI has aimed to actively follow survey for actual condition, interview analysis, and detailed analysis for existing methodology. MaRMI follows international standard, ISO/IEC 12207-Software Life Cycle Processes [1]. MaRMI-I, the first methodology defines 7 phases based on 13 activities of ISO/IEC 12207 to support whole processes in system development. The phases for system development are hierarchically defined for development plan, requirement analysis, fundamental design, detailed design, implementation, test, and installation and transfer and provide simplified artifacts.

MaRMI-I was developed on the basis of structured development and information engineering concept and is organized with all viewpoints from developers, managers, and users. The organization of MaRMI-I is largely separated into MaRMI-D and MaRMI-P. MaRMI-D covers the development of information system from the viewpoint of developers and managers. And MaRMI-P facilitates the establishment for planning information such as project plan, management strategy, environment analysis, business design, and workflow analysis. The existing methodology was described by focusing on the viewpoint of only developers and managers, but on the other side MaRMI-I covered the business, which are known to users and performed by users for mission of systems.

2.2 The CHARACTERISTICS AND ORGANIZATION OF MaRMI-II

After developing MaRMI-I, MaRMI-II was developed on the basis of object-oriented paradigm and aimed to provide business companies with practical and concrete methodology to be used in organization of object-oriented development paradigm. MaRMI-II supports and aids management process as well as development process. The characteristics of MaRMI-II are summarized as follows.

According as MaRMI-II was designed to be applicable in information and communication industry, MaRMI-II use UML (Unified Modeling Language) which are focused on object-oriented modeling as a standard notation [2][3].

The adopting architecture to software development makes benefits to understand the requirements, select constituent modules, define interactions of the modules, decompose modules, and verify quality scenarios. Therefore, the architecture-centric approach makes benefits of increasing reusability of constituent elements and quality in the proposed system. MaRMI-II builds up initial architecture with use-cases, physical packages, hierarchical layers, implementation structures, constituent elements, and distribution topologies.

And the next, iterative and incremental development is executed on base of development plan in MaRMI-II. As mini-project is small-scale work unit, this is iteratively and incrementally developed with one constituent element or more, in a sequential or concurrent way and in a centralized or distributed, shown in figure 2. This makes developers possible to take development works stepwise.

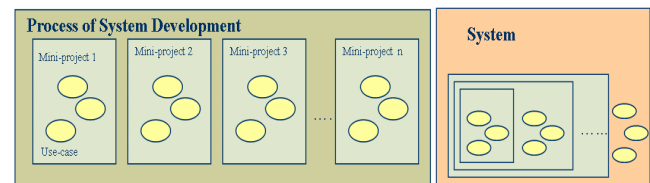


Figure 2. Incremental development of a system with mini-projects.

The software risks hidden in a project need to be identified and classified. From the phase of risk analysis, question lists are generally used to check the points with risks and examine the risks. The risks are classified by accordance with those are related to requirements, techniques, and business markets. For example, the staff risks are asked with questions as follows: "Is staff with good capability available?", "Is staff enough to accomplish this project?", "Are all participants fully educated?", and etc. In technical risks, latent risks with technical uncertainty and technical obsolescence exist through the every phase of design, implementation, verification, and management. The risk identification in every development process is essential to lessens impacts of the risks and the systematical risk management is required on the whole life cycle. In the risk management of MaRMI-II, the use-cases containing high risks are firstly developed as a pre-product and continuously managed with a document of risk analysis.

Moreover, MaRMI-II utilizes a meta-model and supports activities in MaRMI-I. The common parts are utilized with

consistency so that this consistency helps the users who use existing MaRMI to adopt MaRMI-II.

The process of MaRMI-II is defined as subsequent activities for 4 phases of planning, establishment, incremental development, and transfer. Additionally, MaRMI-II has concrete and detailed procedures by being organized with 25 activities and 77 tasks. The detail definition and difference between the activity and the task will be described in the part of meta-model of section 2.4. As brief description, phase means highest level of work and activity is composed of tasks. Each task is a smallest unit of work, accomplished by participants.

2.3 The CHARACTERISTICS AND ORGANIZATION OF MaRMI-III

In 1990's, component-based development (CBD) has been considered as software development method in all activities of development life cycle. CBD approach induces high quality and maximizes software reuse, therefore overcome development difficulties and risks in developing a large-scale system. The reuse of component causes benefits to enhance software reusability and maintainability, furthermore decrease time-to-market and increase productivity of software system.

At that time MaRMI-III was developed, CBD has chosen as a practical solution to increase competitiveness of business companies. MaRMI-III was developed for development of component-based systems and has following characteristics. The development process in MaRMI-III is composed of a process for component-development and for component-based application development. Besides, MaRMI-III provides 2 versions of methodology for J2EE and .NET component platforms.

MaRMI-III [4] is concerned about separation of development process. Firstly, in component development process, the generation of components is mainly executed. After then, in component-based application development process, applications are generated by assembling the already generated components from component development process. Hence, the component development process, lays focus on finding common components with high reusability and the application development process lays focus on the use of the existing components and development of application program.

MaRMI-III is organized with 4 different development phases, 31 activities, and 93 tasks. Development phases are classified into planning, architecture, incremental development, and transfer.

2.4 The CHARACTERISTICS AND ORGANIZATION OF MaRMI-EM

Since 2004, MaRMI-EM has been developed. MaRMI-EM is focused on developing product-line based systems of embedded systems. By reference, the functionality is accounted for 80 percent of software embedded in the fighters in 2000 versus 20 percent in 1970 [5].

Especially, in embedded system domain, we can find that the productivity is not high and the reuse of software and hardware is deficient. For this status, the concept based on product-line can increase productivity of common assets and final products.

MaRMI-EM has developed for development of product-line based embedded systems and has following characteristics. MaRMI-EM is focused on generating a series of products in concurrent way. MaRMI-EM is organized with meta-model compatible to SPEM 2.0. This has feature-driven approach and based on cross architecture and cross process for hardware and software. UML 2.0 standard is adopted for modeling and verification in early stage. The meta-model of MaRMI-EM is shown in figure 3. Meta-model represents that which element is chief constituents in MaRMI-EM.

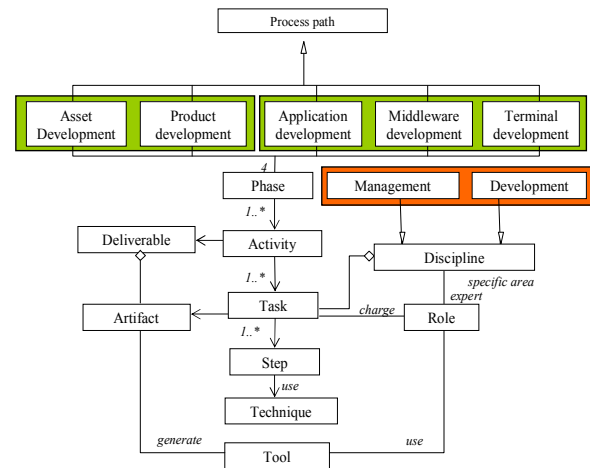


Figure 3. The meta-model of MaRMI-EM.

The definition of each element is described as follows.

1. Process path: A set of phases are defined for accomplishing a specific purpose of project.
2. Phase: As a set of activities for timing-flow in project, which are structured for finishing process path.
3. Activity: A set of tasks which are logically interconnected. At a specific point, the result, generated from the activity, is shared in artifacts from other activities.
4. Task: As a smallest unit, a task is accomplished by project participants. Task is a unit to establish plans in project. Task is composed of a set of procedures.
5. Step: A successively related set of detailed work, which is required to accomplish a task.
6. Discipline: As a process package, a discipline is constructed for a specific area, such as analysis, design, implementation, and configuration management, in software engineering.
7. Technique: Method, procedure, concept, and skill in order to accomplish task.
8. Tool: This is computer-based software for accomplish task efficiently.
9. Role: Organization or people who are responsible to conduct task.
10. Artifact: All kind of outputs, generated from finishing task. This is useful to developers.

11. Deliverable: The outputs to be handed in to users, such as user manuals and tool.

The process path is composed of five different development processes. At first, in processes of asset development and product generation, reusable assets are generated in asset development process and then a specific product is generated by reuse of pre-generated assets.

Next, in the development of applications, middle wares, and terminals, these process support non-product line in embedded systems. These two processes are provided to conduct product-line in the startup and non-product-line.

The concept of discipline, firstly introduced in MaRMI-EM, is defined to represent activities in special area of domain, architecture, design, implementation, verification, deployment, project management, and quality management. The work units which are contained in one area are tightly coupled into one discipline. And they have traceability with other works from other discipline. Accordingly, guidelines can not be overlapped in disciplines. Furthermore, the componentized methodology makes mechanism for complementation and tailoring. In the case of embedded systems, there are diverse kinds of systems, development organizations, and development environments.

To verify the processes of MaRMI-EM, two application cases were performed. The development processes of hardware and software are concurrently processed to generate core assets on the target of PDA platform of navigation systems. The other case is related to the POS system of smartcard.

3. THE COMPARISON OF MaRMI

Since 1994, MaRMI is developed as a series of development methodology and it has followed the development paradigm of structural, object-oriented, component-based, and product-line. The outline of MaRMI project is summarized in table 1.

Table 1. The overview of MaRMI development project

Item	MaRMI-I	MaRMI-II	MaRMI-III	MaRMI-EM
Duration	'94.11. ~'97.10.	'96.11. ~'98.10.	'00.09. ~'03.10.	'04.02. ~'07.01.
Fund Supply Organization	MOST	MIC	MIC	MIC
Leading Organization	SERI	SERI	ETRI	ETRI
Supporting Method	Structural & information engineering based	Object-oriented engineering based	Component engineering based	Product-line based

MOST (The Ministry Of Science and Technology of Korea); MIC (The Ministry of Information and Communication Republic of Korea); SERI (System Engineering Research Institute; incorporated into ETRI as an affiliate in 1996.

The below figure shows user manuals and CDs which are resulted from MaRMI-I, II, and III projects.



Figure 4. The MaRMI-manuals and CDs.

4. CONCLUSIONS

In Korea, since 1994, the research project for MaRMI has been executed. At that time, the technology of development methodology was fully dependent on itself imported from other countries. To procure independent technology especially including their culture and organization, MaRMI development projects have been continued until now. Through research project, MaRMI-I, MaRMI-II, and MaRMI-II are developed and utilized in organizations which bring that technology. Currently, the MaRMI-EM is in process to establish the development processes of embedded systems.

For the spread of technology, especially including their culture and organization, we found that the support from government is absolutely necessary. To distribute and let development methodology be smoothly utilized, the support of government is definitely required. For example, it is possible to increase the use of development methodology in the way of giving benefits to the products if the product followed the methodology and also made in Korea. In addition, it is important to consider that good domestic software can be utilized in the classes of universities.

5. REFERENCES

- [1] ISO/IEC 12207-1995, *Information Technology – Software Life Cycle Processes*, 1995.
- [2] G. Booch, J. Rumbaugh, and I. Jacobson, *the Unified Modeling Language: User Guide*, Addison-Wesley, 1999.
- [3] C. Kobryn, Modeling Components and Frameworks with UML, *Communications of the ACM*, Vol.43 (10), 2000, 31-38.
- [4] D. Ham, J. Kim, J. Cho, and S. Ha, MaRMI-III: A Methodology for Component-Based Development. *ETRI Journal*, vol.26, no.2, Apr. 2004, 167-180.
- [5] T. Genbler, O. Nierstrasz, and B. Schonhage, Components for Embedded Software, In *Proceedings of the 2002 international conference on Compilers, architecture, and synthesis for embedded systems*, 2002, 19-26.