

게임 소프트웨어 산출물 통합관리시스템 설계

김병철
서울대학교 정보문화학전공

Design of an Integrated Product Management System for Game Software Development

Byung-Cheol Kim

Information and Culture Technology Studies, Seoul National University

요약 게임과 같은 엔터테인먼트 분야의 소프트웨어 품질을 일정 수준 이상으로 보장하면서 개발을 진행하기 위해서는 소프트웨어의 생애주기 상에서 발생하는 다양한 종류의 개발 산출물들을 효과적으로 관리해야 한다. 게임 소프트웨어 개발 산출물들은 문서, 소스코드, 모델, 도면, 그래픽 리소스 등과 같이 다양한 형식과 종류를 갖게 되며 그 양이 매우 방대한 것이 일반적이다. 따라서 소프트웨어 개발 산출물들을 통합 관리하는 것이 품질보증을 위해 필수적이며, 이를 위해 본 연구에서는 서로 다른 종류의 산출물을 통합 관리할 수 있도록 하는 기능에 초점을 맞춘 소프트웨어 산출물 통합관리시스템을 설계하였다. 이 시스템은 프로젝트관리, 요건관리, 형상관리, 추적성관리, 소프트웨어공학도구 표준 인터페이스 기능 등을 제공할 수 있도록 설계되었다. 특히 이 통합관리시스템은 데이터웨어하우스 기술을 기반으로 함으로써 이종의 데이터를 통합 저장할 수 있는 중앙 저장소를 제공하고, 이를 바탕으로 한 다 측면 보고 및 분석, 추적, 검증 등의 개발 프로세스들을 효과적으로 지원할 수 있도록 설계되었다.

주제어 : 소프트웨어 품질, 산출물 관리, 개발 프로세스, 통합관리시스템, 데이터웨어하우스

Abstract This paper proposes an integrated management system for products of game software development. Those products generated over software life-cycle are generally huge in numbers, various in kinds and interconnected in relationships among them. Especially they are highly complex and tightly coupled each other because they are used to build another world which is composed of multi-modal data such as graphics, sound, logics and so on. Therefore it is critical to manage development products in an integrated manner to guarantee the quality of software. The proposed system is designed to provide project/requirement/configuration/traceability management capabilities and interfacing to CASE standards. To achieve the objective, it is based on data warehouse technologies which provide central repositories of heterogeneous data, multi-aspect reporting&analysis capabilities, traceability, verification, etc.

Key Words : Software Quality, Product Management, Development Process, Integrated Management System, Data Warehouse

Received 24 October 2015, Revised 27 November 2015

Accepted 20 December 2015

Corresponding Author: Byung-Cheol Kim
(ITCT Studies, Seoul National University)

Email: clorvie@gmail.com

ISSN: 1738-1916

© The Society of Digital Policy & Management. All rights reserved. This is an open-access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0>), which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. 서론

게임과 같은 엔터테인먼트 분야 소프트웨어의 경우 그 생애주기 각 단계 공정 활동으로부터 다양한 형태의 산출물이 발생하게 된다. 산출물은 워드 (MS-Word, HWP, Excel, PDF, 텍스트) 파일 형태의 문서는 물론, 각 단계별로 기존의 소프트웨어공학도구(Computer Aided Software Engineering, CASE)를 활용하는 경우, 각 도구에서 제공하는 형식의 설계문서, 검증문서 및 프로그램 코드, 그래픽 리소스 등의 형태가 혼재되어 다양한 형태로 생성된다. 이러한 상황은 설계 변경, 설계 오류 수정, 검증 과정에서 발생하는 정보 추적, 개정관리 및 품질관리에 상당한 어려움을 초래한다. 또한, 개발자 변경, 개발 및 검증도구 버전 변경, 운영체제 등의 개발환경 변경, 하드웨어 변경 등으로 인해 품질을 보증하는 과정에서 어려움이 발생하기도 한다. 결국 이러한 어려움이 곧바로 소프트웨어 품질의 저하로 연결되므로 소프트웨어의 품질을 일정 수준 이상으로 보증하기 위해서는 이에 대한 철저한 대책이 요구된다[1].

소프트웨어의 개발 및 품질보증 상에서 발생하는 어려움을 극복하고 위에서 언급한 대책을 수립하기 위해서는 요건관리 및 형상관리, 추적성관리 프로세스를 기반으로 한 통합정보관리 기술이 필요하며, 이를 지원하는 정보관리시스템을 활용하여 다양한 업무 프로세스를 적절히 지원해주어야 한다[2-6]. 현재 게임 등의 엔터테인먼트 서비스 시스템을 위한 소프트웨어 개발에서는 소프트웨어 요건관리, 형상관리, 품질관리 기법을 각각 필요에 따라 각 단계별로 이종의 개별 툴을 적용하여 소프트웨어의 품질을 유지하기 위해 노력하고 있으나, 통합관리를 통한 효율성 확보 수준에는 미치지 못하고 있다. 본 논문에서는 이러한 상황에 대한 대책의 일환으로서 서로 다른 종류의 소프트웨어 개발 산출물에 대한 통합 관리에 초점을 맞춘 산출물 통합관리시스템을 설계한 내용을 소개한다. 특히 본 통합관리시스템은 데이터의 통합 저장 및 관리 기능의 기반 기술로서 데이터 웨어하우스 기술을 채택하여 일관적으로 이종의 데이터들을 관리한다.

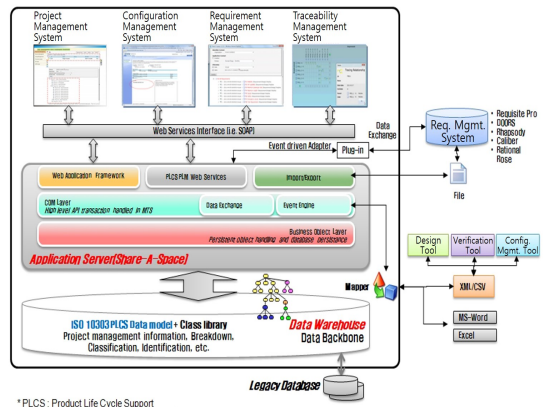
본 논문의 2장에서는 이러한 통합관리시스템의 설계에 대해 설명하고, 3장에서는 이의 활용 방안에 대해 논의한 뒤, 4장에서 결론을 맺는다.

2. 통합관리시스템 설계

게임과 같은 엔터테인먼트 분야 소프트웨어의 경우 본 연구에서 설계한 소프트웨어 개발 산출물 통합관리시스템은 데이터 웨어하우스 기술을 기반으로 각종 제품에 대한 생애주기 관리를 지원하는 Share-A-Space (SAS) 시스템을 기초로 하고 있다[7]. 본 논문에서는 데이터 웨어하우스 기술에 대한 소개는 생략하고 SAS 기반으로 아키텍처를 설계한 내용과 산출물 통합관리시스템의 기능 설계에 대한 내용을 기술한다.

2.1 아키텍처 설계

[Fig. 1]은 본 연구에서 설계한 통합관리시스템의 아키텍처를 보여준다. 본 통합관리시스템은 SAS라는 응용서버(Application Server)와 프로젝트관리 시스템, 형상관리 시스템, 요건관리 시스템, 추적성관리 시스템 등의 기능별 클라이언트 시스템들로 구성되며 이의 통합적 데이터 저장 및 처리 기반으로 데이터 웨어하우스를 활용한다.



[Fig. 1] The architecture of the Integrated Product Management System

클라이언트와 서버 간에는 웹 서비스 인터페이스인 SOAP을 통하여 인터페이스 하도록 하였다. 웹서비스를 위한 서비스 중심 아키텍처인 SOAP를 이용함으로써 다양한 웹서비스들을 포괄하면서 통일된 규격으로 서비스와 서비스 간, 서비스와 시스템 간의 정보 교류를 할 수 있다. [Fig. 1]의 오른쪽에 나타나는 바와 같이 본 연구에서 설계된 통합관리시스템은 기존의 상용 소프트웨어공

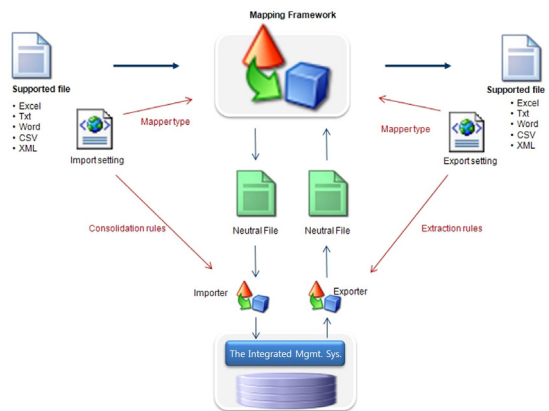
학도구(CASE)와의 연계를 제공한다.

본 시스템은 데이터 백본(data backbone)으로서 데이터 웨어하우스 기술을 활용한다. 데이터 웨어하우스 기술은 이종의 데이터(heterogeneous data)를 통합 저장할 수 있는 중앙 저장소(central repository)를 제공하고, 이를 기반으로 다측면(multi-aspect) 보고 및 분석, 추적(trace), 검증(verification) 등의 개발 프로세스들을 지원하는 기술이다[8,9,10,11,12]. 따라서 이를 연계해 활용하면 본 시스템이 제공하고자 하는 기본적인 서비스 기능들을 보다 효과적으로 설계할 수 있다.

또한 데이터 웨어하우스 기술은 기존의 데이터베이스(legacy database)들과도 연계하여 이들과의 교량(bridge) 역할을 함으로써 기존에 구축되어 있던 개별 관리시스템을 완전히 배제하지 않을 수 있으므로 연속적인 데이터 이력 관리를 가능하게 한다. 따라서 이를 위해 본 시스템에서 데이터 웨어하우스 기술을 응용할 때 기존 데이터베이스의 데이터 스키마 등을 통합하여 전체 데이터 처리 프레임워크를 설계한다.

응용 서버인 SAS는 이러한 데이터 웨어하우스와의 연계를 통하여 클라이언트 시스템들이 요청하는 데이터를 직접 제공하는 역할을 한다. 이는 크게 자료 공유 환경 제공, 협업 환경 제공, 변경 관리, 문서 관리, 업무 흐름 관리와 같은 형상관리 기본 환경 제공 등을 통하여 이루어진다. 특히 각 산출물의 생애주기 전반에 대해 그러한 기능들을 제공함으로써 일관적인 데이터 체계를 유지·확보할 수 있는 기반이 되며, 2.2절에서 설명할 각 관리 기능별 시스템과의 연동할 수 있다.

기존의 상용 CASE와의 인터페이스 또한 SAS가 제공하는 매핑 프레임워크를 활용하여 설계하였다. 이 때 BOM(Business Object Layer)이 서비스 단위별 데이터를 추상화하여 통일된 서비스 핸들링 기반으로 마련하고, COM(Component Object Model) 레이어는 데이터 교환 모듈과 이벤트 엔진을 탑재하여 응용 서버가 매핑 프레임워크와 이종의 데이터 및 요청들을 처리할 수 있는 기술적 기반으로 제공한다. 특히 매핑 프레임워크의 스키마를 따라 각 CASE의 산출물을 매핑하는 표준 인터페이스를 구축하고, 각 CASE의 산출물을 표준 인터페이스 형태로 변환하는 변환기를 개발하는 것이 상용 CASE와의 인터페이스 설계의 핵심 내용이다. [Fig. 2]는 이러한 표준 인터페이스 방식에 대한 설계를 보여준다.



[Fig. 2] A standard interface of the Integrated Product Management System

2.2 기능 설계

본 연구에서 설계한 소프트웨어 개발 산출물 통합관리시스템은 크게 프로젝트관리기능, 요건관리 기능, 형상관리 기능, 추적성관리 기능 및 소프트웨어공학도구 표준 인터페이스 기능을 갖는다. 각각의 기능에 대한 개략적인 설명은 다음과 같다.

- 1) 프로젝트관리(project management) 기능:
소프트웨어 요구사항분석, 설계, 구현, 시험 등 소프트웨어 생애주기 전체에 대하여 프로젝트관리 엔지니어링 프로세스를 통합 지원할 수 있는 기능
- 2) 요건관리(requirement management) 기능:
소프트웨어 요구사항분석, 설계, 구현, 시험 등 소프트웨어 생애주기 전체에 대하여 요건관리 엔지니어링 프로세스를 통합 지원할 수 있는 기능
- 3) 형상관리(configuration management) 기능:
소프트웨어 요구사항분석, 설계, 구현, 시험 등 소프트웨어 생애주기 전체에 대하여 형상관리 엔지니어링 프로세스를 통합 지원할 수 있는 기능
- 4) 추적성관리(traceability management) 기능:
요구사항 및 설계 변경 시 개발 데이터 및 검증 데이터에 대한 추적성을 관리하고 일괄적으로 변경 사항을 관리할 수 있는 기능
- 5) 소프트웨어공학도구 표준 인터페이스 기능:
상용 소프트웨어공학도구의 다양한 산출물들 사이의 자료 공유 및 연계 기술을 기반으로 하여 어떠한 상용도구가 개발에 적용 되더라도 통합 관리를

일관적으로 지원할 수 있는 표준 인터페이스 기능 일반적으로 소프트웨어의 설계, 개발을 위해서 사내/외 협력 및 아웃소싱이 필요하며, 각종 요구 및 의견/지시사항 전달, 산출물 제출 등 개발 프로젝트 범위 내에서 협업을 지원할 수 있는 환경이 필요하다. 프로젝트 관리 기능은 이러한 협업 환경의 구축을 위해서 업무프로세스 지원 기술, 협업 진행을 위한 기반 기술, 문서관리 기반기술, 기존 시스템 연계기술, 보안 기술 등을 제공하는 기능을 말한다.

요건관리 기능을 통해 소프트웨어의 수명주기 상 기획단계에서의 요구사항을 체계적으로 관리함으로써, 소프트웨어의 요구사항을 정리, 추적, 확인할 수 있으며, 이러한 요구사항이 설계 및 개발 단계에서 정확히 구현되고 있는지를 감시하여 요건을 충족시키는 결과물을 생산할 수 있게 한다. 요건관리 시스템은 소프트웨어 개발에 적용되는 각종 요구사항들을 일괄적으로 관리하는 것이 목적으로 요구사항에 대한 정의, 요구사항과 제품정보의 연계, 요구사항 충족여부 점검, 단계별 이력 관리, 문서관리, 시스템 구조에 따른 요구사항 관리 등의 기능을 제공한다.

소프트웨어 개발 프로세스 및 소프트웨어 수명주기를 관리하기 위한 형상관리 시스템은 개발 프로세스 상에서 생성되는 다양한 데이터 및 산출물 간의 일치성을 보장할 수 있는 기능이 요구됨. 이는 소프트웨어 코드에 대한 형상관리와는 상이한 형상관리 개념으로 모든 데이터 간의 연결성, 일치성을 보장하기 위한 다양한 관리기능을 제공한다. 형상관리 시스템은 식별자 관리 기능, 버전 관리 기능, 유효성 관리 기능, 속성 관리 기능, 분류체계 관리 기능 등을 종합적으로 제공하여 시간, 장소, 담당자가 변화하는 상황에서도 일관적 데이터 관리를 가능하게 한다.

추적성관리 시스템은 크게 추적성 가시화 기능 및 리포팅 기능을 제공하는데, 이는 요건 추적성을 가시화하여 연계 산출물과의 추적성을 용이하게 분석하도록 돕고, 요건별, 연계 산출물별 추적성 리포트를 생성하도록 하는 기능이다. 특히 단방향이 아니라 양방향의 추적을 가능하게 함으로써 단위 데이터의 변경이 전체 시스템에 미치는 영향뿐만 아니라 전체 시스템의 무결성에 문제가 발생하였을 때 이의 원인을 각 단위 데이터 레벨로 추적할 수 있는 장점이 있다.

3. 통합관리시스템 활용방안

이 장에서는 본 연구에서 설계한 소프트웨어 개발 산출물 통합관리시스템의 활용 방안에 대하여 고찰한다. 이를 위해 소프트웨어 개발 프로세스, 추적성 관리 프로세스, 검증 프로세스 각각에 대하여 본 시스템을 활용하는 경우를 고려하고, 이의 장단점을 논의한다.

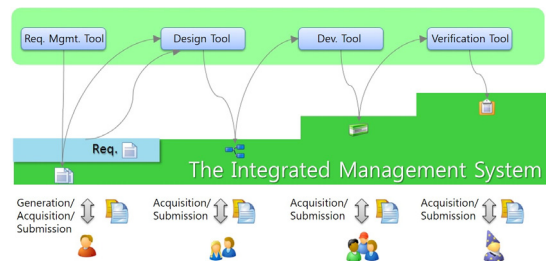
3.1 소프트웨어 개발 프로세스에서의 활용

[Fig. 3]은 통합관리시스템을 소프트웨어 개발 프로세스에 활용하는 방안을 도식적으로 보여준다. 해당 프로세스를 위해, 소프트웨어 요건을 통합관리시스템 상에서 생성하고 명세하거나, 기존 상용 요건관리도구 상의 소프트웨어 요건을 импорт(Import)하여 관리함으로써 다양한 참여자들의 요청사항을 일관적으로 전달할 수 있다.

통합관리시스템 상에서 관리되는 요건을 상용의 소프트웨어 설계도구로 익스포트(Export)하여 일관성이 유지되는 소프트웨어 설계가 가능하도록 한다. 이렇게 통합관리시스템에서 직접 익스포트 하는 경우 설계 아이템을 다시 통합관리시스템으로 импорт하여 관리할 수 있다.

마찬가지로 통합관리시스템 상에서 관리되는 설계 아이템도 상용 소프트웨어 구현 및 코드 형상관리도구로 익스포트하여 일관성이 유지되는 소프트웨어 코딩이 가능하도록 한다. 코드 및 코드 형상정보를 다시 통합관리시스템으로 импорт하여 관리할 수 있다.

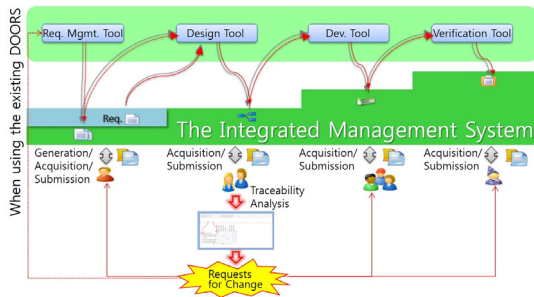
마지막으로 시스템 상에서 관리되는 설계 아이템을 상용의 검증도구로 익스포트하여 일관성이 유지되는 소프트웨어 검증이 가능하도록 한다. 검증 형상정보를 다시 통합관리시스템으로 импорт하여 관리할 수 있다.



[Fig. 3] The Integrated Product Management System applied on a software development process

3.2 추적성 관리 프로세스에서의 활용

[Fig. 4]는 통합관리시스템을 추적성 관리 프로세스에 활용하는 방안을 도식적으로 보여준다. 추적성 분석은 주로 Fagan Inspection 방법을 활용하여 수행된다[13]. 통합관리시스템을 활용하여 설계 단계에서 추적성 분석 결과 변경요청사항이 발생한 경우, 요건, 코드, 검증 단계 프로세스에 동시에 통보하여 각 단계 업무 담당자들이 일관되게 변경을 수행하고 해당 업무를 수행한다. 이 경우 일관된 추적성 관리는 물론 요건 관리, 형상 관리, 프로젝트 관리 프로세스와의 통합 관리가 용이하여 해당 업무를 효율적으로 수행할 수 있도록 한다는 이점도 얻을 수 있다.



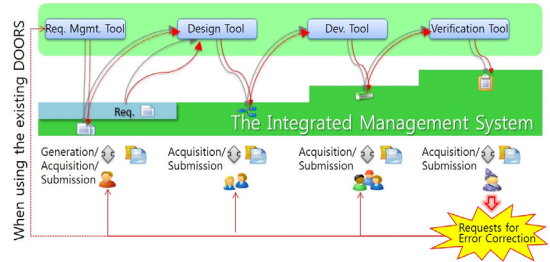
[Fig. 4] The Integrated Product Management System applied on a traceability management process

3.3 검증 프로세스에서의 활용

[Fig. 5]는 통합관리시스템을 소프트웨어 검증 프로세스에 활용하는 방안을 도식화한 것이다. 예를 들어, 검증 단계에서 테스트 상용 도구를 활용하여 시험을 수행하는 경우를 보자[14]. 각 단위 개발자가 작성한 모듈을 테스트한 결과, 만약 오류수정 요청사항이 발생하는 경우에는 이를 통합관리시스템을 활용하여 요건, 설계, 코드 단계에 모두 통보하게 된다. 따라서 각 단계 업무 담당자들은 해당 요청을 일관적으로 반영하여 전체 시스템의 무결성과 정합성을 유지할 수 있다.

이러한 방식으로 각종 소프트웨어 생애주기 프로세스에서 현재 적용되고 있는 DOORS, SCADE [15], Sub-Version, CodeScroll, TimeBounder 등의 소프트웨어공학도구와 본 통합관리시스템이 통합되어 운용될 수 있으며, 각 산업분야의 소프트웨어에 적용하고 있는 다

양한 소프트웨어공학도구가 본 연구에서 설계한 통합관리시스템과 함께 통합되어 운용될 수 있다.



[Fig. 5] The Integrated Product Management System applied on a verification process

4. 결론

본 연구에서 설계한 소프트웨어 개발 산출물 통합관리시스템은 소프트웨어 요구사항분석, 설계, 구현, 시험 등 생애주기 전체에 대하여 프로젝트관리, 요건관리, 형상관리, 추적성관리 등과 같은 각종 엔지니어링 프로세스를 통합적으로 관리하는 소프트웨어 엔지니어링 허브 구축을 가능하게 할 것으로 기대된다. 특히, 이종의 소프트웨어공학도구가 생성하는 다양한 산출물들에 대하여 자료 공유 및 연계 기술을 기반으로 하여 어떠한 도구가 소프트웨어 개발에 사용되더라도 통합 관리를 지원하는 표준인터페이스 기술은 상당히 실체적인 기술로서 많은 소프트웨어 프로젝트에 활용될 수 있을 것이다.

다만 이러한 통합관리시스템은 전체가 유기적으로 통합 연동되는 데 초점을 맞추기 때문에 개별 작업의 프로세스가 미세하게 바뀌는 것을 즉각 반영하기 힘들다는 한계가 있다. 한 단계 혹은 한 단위의 작업 상세(specification)의 변화가 전체 시스템의 여러 단계와 단위에 영향을 미치게 되는 경우, 이는 전체 관리 프로세스에 대한 재분석과 재조정 작업을 필요하게 만든다. 그렇지 않은 경우, 시스템의 무결성과 정합성을 유지할 수 없을 수도 있기 때문이다. 따라서 시스템 설계 및 구축 이후에 현장에서 발생하는 다양한 새로운 요건들과 조정을 실무적으로 변경하는 데 제한이 따르며, 이를 보완하기 위한 유연한 설계 방식이 추가로 요구된다.

본 연구에서 설계한 소프트웨어 개발 산출물 통합관

리시스템이 구현되어 게임 등의 엔터테인먼트 소프트웨어 개발 작업에 사용된다면 엔지니어, 아티스트, 기획자, 관리자 등 개발 참여자의 의사소통과 업무 협력 수준을 한 차원 끌어올릴 수 있을 것으로 기대된다. 또한 본 시스템은 엔터테인먼트 서비스 시스템 분야뿐만 아니라 원자력, 국방, 의료, 자동차, 철도 및 항공 분야에서도 활발하게 활용될 수 있을 것으로 사료된다. 이러한 분야에서는 대개 소프트웨어 생애주기 동안에 다양한 이종 도구들이 사용되고 있어 고품질 소프트웨어를 효율적으로 개발하는 것이 매우 어려운 상황임을 고려할 때 본 산출물 통합관리시스템의 효용성이 더욱 확실할 것으로 기대한다.

REFERENCES

[1] A. Labodová, "Implementing integrated management systems using a risk analysis based approach," *Journal of Cleaner Production*, Vol. 12, No. 6, pp. 571 - 580, 2004.

[2] O. Hermanns, "Data Access Protocols for Integrated Engineering Environments", *Proc. CompEuro '93: Computers in Design, Manufacturing and Production*, IEEE, pp. 350-357, 1993.

[3] K. Pohl and S. Jacobs, "Concurrent Engineering: Enabling Traceability and Mutual Understanding," *Concurrent Eng.: Research & Applications* Vol. 2, No. 4, pp. 279-290, 1994.

[4] A. Gunasekaran, "An integrated product development quality management system for manufacturing," *The TQM Magazine*, Vol. 10 Iss: 2, pp.115 - 123, 1998.

[5] M. Braglia and M. Frosolini, "An integrated approach to implement Project Management Information Systems within the Extended Enterprise," *International Journal of Project Management*, Vol. 32, No. 1, pp. 18 - 29, 2014.

[6] C. W. L. Hill et al., "Strategic Management: Theory: An Integrated Approach (11th Ed.)," Cengage Learning, 2014.

[7] DOI: <http://www.eurostep.com/>

[8] R. Kimball, "The Data Warehouse Toolkit: Practical Techniques for Building Dimensional Data Warehouses," John Wiley & Sons, Inc., 1996.

[9] T. Thalhammer et al. "Data warehouses: Complementing OLAP with active rules," *Data and Knowledge Engineering*, Vol. 39, No. 3, pp. 241 - 269, 2001.

[10] W. H. Inmon, "Building the Data Warehouse (4th Ed.)," Wiley Publishing Inc., 2005.

[11] A. Vaisman and E. Zimányi, "Data Warehouse Systems: Design and Implementation," Springer, 2014.

[12] J. E. Hajlaoui and N. Hamdani, "Active data warehouse: Review, challenges and issues," 2014 World Symposium on Computer Applications & Research (WSCAR), 2014.

[13] Y. S. Lim et al., "The Use of Fagan Inspection in Software Fault Detection and Comparisons with Other Methods: An Experiment Study," *Journal of the Korea Information Science Society(C)*, Vol. 2, No. 1, pp. 1-11, 1996.

[14] R. McLeod, "Software Tesing," John Wiley & Sons, Inc., 2007.

[15] DOI: <http://www.esterel-technologies.com/>

김 병 철(Kim, Byung-Cheol)



- 2002년 2월 : 아주대학교 정보 및 컴퓨터공학부(공학사)
- 2004년 2월 : 한국과학기술원 전자전산학과 전산학 전공(공학석사)
- 2011년 8월 : 한국과학기술원 전산학과(공학박사)
- 2011년 9월 ~ 현재 : 서울대학교 정보문화학 전공 강사
- 관심분야 : 가상현실, 컴퓨터그래픽스, 물리기반 시뮬레이션
- E-Mail : clorvie@gmail.com