

특집

MDA 기반 컴포넌트 생성 및 조립 기술*

김민정¹⁾ 윤석진²⁾ 권오천³⁾ 신규상⁴⁾

목 차

1. 서 론
2. 연구배경
3. MDA 기반 컴포넌트 통합 개발 도구의 특성 및 기능
4. 결론 및 향후 연구

1. 서 론

1990년대 중반부터 기존 객체지향 방법론의 한계점을 극복하고자 연구가 시작된 CBD(Component Based Development) 방법론은 재사용성, 적시성, 유지 보수성 등이 업체의 경쟁력으로 대두되면서 각광 받고 있다. 컴포넌트내의 코드나 데이터는 객체로 구성되어 객체 지향 프로그램과 유사하지만 컴포넌트는 구현으로부터 분리된 인터페이스를 갖고 있기 때문에 컴포넌트를 재사용하고자 하는 사람들은 컴포넌트의 인터페이스만을 이해하여 응용 시스템으로 합성시키면 된다. 따라서 소프트웨어 개발자들은 재사용성이 높은 컴포넌트들을 개발할 수 있고, 컴포넌트들을 조립하여 더 큰 규모의 합성 컴포넌트(Composite Component) 나 특정 어플리케이션

션을 보다 빠르게 개발할 수 있기 때문에 소프트웨어 개발의 생산성 및 품질을 높일 수 있다는 장점이 있다.

CBD가 성공적이기 위해서는 컴포넌트 자체의 재사용성이 높아야 하고, 서로 다른 요구사항에 따라 개조 가능해야 하며, plug-&-play 방식으로 조립이 가능해야 한다. 그러나 소프트웨어 기술이 급속하게 변화함에 따라 단일 기업 내에서도 다양한 플랫폼이 존재하게 되었고 B2B 환경으로 전환됨에 따라 이기종 플랫폼간의 컴포넌트의 이식성의 문제나 상호운용성 문제에 직면하게 되었다. 이러한 상황 하에서 OMG는 지금까지 추진한 여러 표준화 작업을 바탕으로 모델 중심의 시스템 명세와 개발 방법인 MDA(Model Driven Architecture)[1]를 제안하여 특정 기술에 기반한 시스템을 기술 변화에 맞게 통합, 변화, 유지 가능하도록 지원하고자 하였다. MDA 접근 방법은 시스템의 설계와 명세를 플랫폼 독립적인 모델인 PIM(Platform Independent Model)으로 작성하고, 매핑을 통해서 실제 구현과 관련된 모델

- 1) ETRI 컴퓨터, 소프트웨어기술연구소 S/W컨텐츠연구부 연구원
- 2) ETRI 컴퓨터, 소프트웨어기술연구소 S/W컨텐츠연구부 컴포넌트 공학팀 선임연구원
- 3) ETRI 컴퓨터, 소프트웨어기술연구소 S/W컨텐츠연구부 책임 연구원
- 4) ETRI 컴퓨터, 소프트웨어기술연구소 S/W컨텐츠연구부 컴포넌트 공학팀 팀장

* 본 연구는 2002년 과학기술부 국가지정연구실 사업으로 수행되었음.

PSM(Platform Specific Model)을 만들어냄으로써 시스템을 보다 효율적으로 유지 및 통합할 수 있다[2].

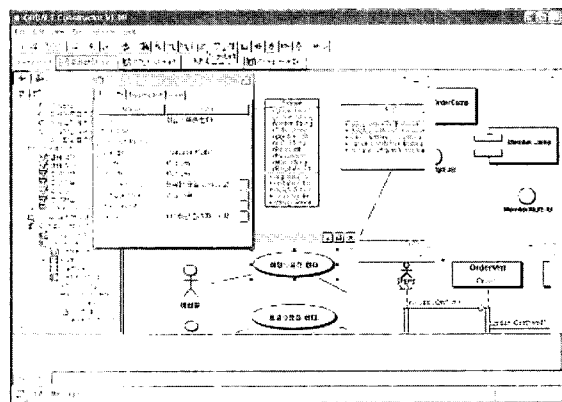
MDA가 성공적으로 이루어지기 위해서는 MDA 지원 도구에서 PIM에 플랫폼 종속적인 정보를 제외한 모든 정보를 모델링할 수 있어야 하고 PIM의 완전성을 검증할 수 있어야 한다. 또한 자동화된 매핑 기능을 제공하여 PIM을 각종 플랫폼에 종속적인 PSM으로 변환할 수 있어야 하고 설계 정보로부터 자동적으로 코드를 생성할 수 있어야 한다. PIM의 효과적인 재사용을 위해서는 도메인 아키텍처 재사용의 기능도 지원해야 한다. 본 논문에서는 MDA에 기반한 컴포넌트의 생성 및 조립을 지원하는 MDA 기반 컴포넌트 통합 개발 도구에 대해 기술하고자 한다. MDA 기반 컴포넌트 통합 개발 도구는 컴포넌트와 웹 서비스의 개발과 조립을 효율적으로 할 수 있도록 개발 과정에서 활용되는 개발 기법 및 절차의 자동화를 지원해 주는 통합 CASE(Computer Aided Software Engineering) 도구다. 본 논문의 구성은 다음과 같다. 2장에서는 연구 배경과 본 연구팀에서 개발을 완료한 컴포넌트 생성 및 조립 지원도구에 대해 살펴보고, 3장에서는 MDA 기반 컴포넌트 통합 개발 도구의 특징 및 구조를 설명한다. 마지막으로 4장에서는 결론 및 향후 연구방향을 제시한다.

2. 연구배경

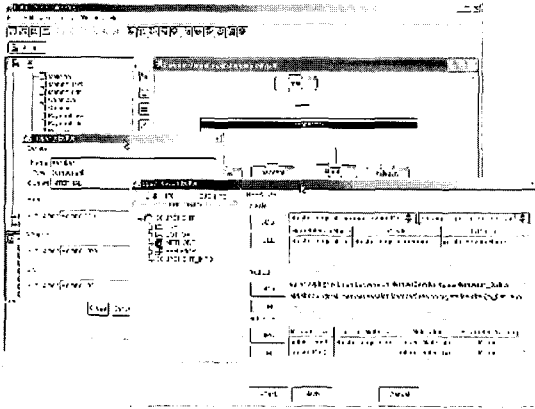
2.1 컴포넌트 생성 및 조립 지원도구 - COBALT Constructor & Assembler

본 연구팀에서는 기존의 CBD 접근 방식을 지원하는 CASE 도구를 개발하여 상용화를 완료하였다. 이 도구는 컴포넌트 생성 지원도구인 COBALT (COmponent-Based Application deVeLopment Tool) Constructor[7]와 컴포넌트

트 조립 지원도구인 COBALT Assembler[8]로 구성되어 있다. COBALT Constructor는 도메인 분석을 통한 컴포넌트 식별, 컴포넌트 단위의 설계 및 구현, 전개, 테스트의 컴포넌트 생성 프로세스 전 단계를 지원하는 도구며 (그림 1)은 컴포넌트 생성도구의 결과화면 중의 일부를 보여준다. 특히, 컴포넌트 생성 지원도구는 도메인 모델로부터 객체간의 관계 정보를 추출하여 자동으로 재사용 가능한 컴포넌트를 식별하는 기능을 제공한다. EJB 컴포넌트의 조립을 지원하는 도구인 COBALT Assembler는 시스템 아키텍처의 모델링, 컴포넌트의 개조와 합성, 전개, 테스트의 컴포넌트 조립 프로세스의 전 단계를 지원한다. 대상 어플리케이션 시스템을 구성하고 있는 컴포넌트들 간의 관계를 C2 아키텍처 스타일을 기반으로 Plug-&-Play 방식으로 쉽고 빠르게 아키텍처를 모델링해 주고, 이를 기반으로 컴포넌트를 합성하여 최종 어플리케이션 시스템을 생성하거나 합성된 큰 규모의 새로운 컴포넌트를 생성, 패키징, 전개해 준다. 또한 바이너리 컴포넌트를 사용자의 어플리케이션 요구사항에 맞도록 컴포넌트를 개조하여 사용할 수 있도록 한다. (그림 2)는 컴포넌트 조립 지원도구의 결과화면중의 일부를 보여준다.



(그림 1) 컴포넌트 생성 지원도구(COBALT Constructor)



(그림 2) 컴포넌트 조립 지원도구(COBALT Assembler)

컴포넌트 생성 및 조립 지원도구는 도메인 분석 정보로부터 컴포넌트 후보를 자동 식별할 수 있으며, 역공학을 통해 자바 어플리케이션으로부터 컴포넌트를 추출할 수 있고, 재사용을 위해 개발된 컴포넌트들을 Plug-&Play 방식으로 쉽고 빠르게 조립하여 응용 시스템을 구축할 수 있으며, 아키텍처 모델을 기반으로 컴포넌트들을 합성하여 최종 어플리케이션을 생성하거나, 합성된 새로운 컴포넌트를 생성할 수 있다는 장점이 있다.

이러한 컴포넌트 생성 및 조립 지원도구의 장점에도 불구하고 기존의 CBD 개발 도구들은 특정한 하나의 미들웨어 플랫폼만을 지원하는 도구로 개발될 수 밖에 없었다. 그러나 응용 시스템의 운용 환경은 J2EE, CORBA, .NET 등으로 점점 다양해지고 있으며, 이러한 요구를 만족시키기 위해서 새로운 개발 패러다임인 MDA를 지원하여야 한다.

2.2 MDA(Model Driven Architecture)

MDA는 OMG가 지속적으로 작업해 온 모델링 관련 표준화 작업의 성과를 바탕으로 모델 중심의 시스템 구현을 위한 기술 구조를 정의한 것이다. MDA의 핵심 사항은 시스템의 설계 및 명세를 구

현 기술에 독립적인 모델(PIM)로 기술하고 이를 자동으로 변환함으로써 실제 구현 환경에 배치하는 것이다. 다시 말하면 MDA는 시스템 통합과 유지를 위해서 PIM으로부터 기술 종속적인 모델(PSM)을 얻을 수 있도록 하는 기술 조건에 대한 정의 및 그에 대한 구조에 관한 것이라고 할 수 있다. MDA는 다음과 같은 네 종류의 OMG 표준에 기반을 두고 있다.

2.2.1 UML(Unified Modeling Language)(3)

객체 및 컴포넌트 시스템을 표현하기 위한 표준 언어다.

2.2.2 MOF(Meta Object Facility)(4)

모델 정보에 대한 표준적인 저장소를 제공하고 표준화된 방식으로 모델 정보를 접근하는 구조를 정의한다.

2.2.3 CWM(Common Warehouse Metamodel)(5)

데이터 저장소 통합에 대한 표준을 정의하고 데이터베이스 모델과 스키마 변환 모델, OLAP(On Line Analytic Processing), 데이터 마이닝 모델(Data Mining Model)에 대한 표준화된 표현 방법을 제공한다.

2.2.4 XMI (XML Metadata Interchange)(6)

UML로 기술된 모델 정보를 교환하기 위한 XML 방식의 표준이다.

이러한 표준을 통해서 시스템 설계를 표준적인 방법으로 수행할 수 있으며 모델의 생성, 배포, 변환, 저장 등의 모델 관리도 정형화된 방식으로 수행할 수 있다. 또한 미들웨어 공통의 디렉토리(Directory), 보안(Security), 분산 이벤트 핸들링(Distributed Event Handling), 트랜잭션(Transaction), 지속성(Persistence) 서비스 등

의 기본 서비스(Pervasive Service)와 공통 도메인 기능(Domain Facility)을 PIM 수준에서 정의함으로써 플랫폼 공통 서비스나 도메인 공통 모델이 특정 구현 기술에 종속되어 정의되는 것을 피하고 구현 환경에 독립적으로 최대한의 정보를 PIM 수준에서 기술할 수 있게 하였다.

2.3 기존 연구

MDA 기반의 대표적인 도구로는 Interactive Objects Software사의 ArcStyler[9]와 Compuware사의 OptimalJ[10], Objecteering Software사의 Objecteering/UML[11]이 있다. ArcStyler의 경우 분석된 비즈니스 모델을 기반으로 PIM을 생성하고 PIM 단계에서 플랫폼 정보 및 전개에 필요한 정보 등을 입력하여 소스 코드를 자동으로 생성시키는 기능을 제공한다. OptimalJ의 경우 클래스와 서비스로 구성된 도메인 모델을 PIM으로 간주하고 이로부터 패턴 기반 접근방식을 통해 J2EE 플랫폼 기반의 PSM을 자동으로 생성한다. Objecteering/UML의 경우 프로파일을 생성할 수 있는 Profile Builder와 실제 모델링 기능을 제공하는 모델러로 구성되며 프로파일의 집합인 모듈을 제공하여 PIM을 해당 플랫폼에 맞는 PSM으로 변환하고 코드를 생성하는 기능을 제공한다. 그러나 이들은 아키텍처에 대한 고려가 생략되어 있으며 PIM 단계에서 컴포넌트 기반의 분산 환경에 대해 고려하고 있지 않으며, PIM2PSM 변환과정에서 메타 모델 매핑에 대한 고려가 생략되어 일반적으로 적용하기엔 한계가 있다.

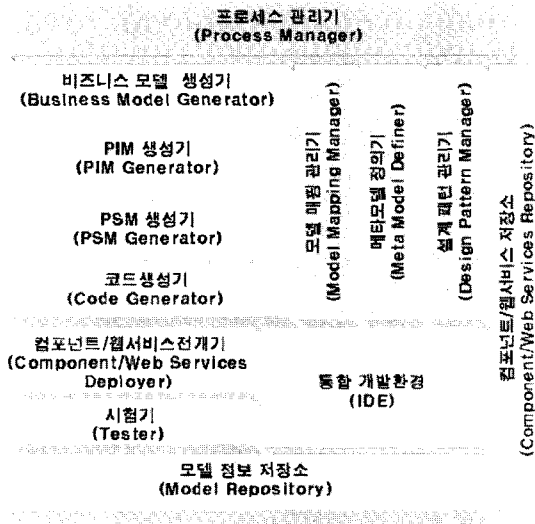
현재 OMG의 MDA를 기반으로 RM-ODP (Reference Model of Open Distributed Processing)[12]와 UML for EDOC을 채택한 COMBINE (Component-Based Interoperable Enterprise system development)[13] 프로젝트가 Opengroup을 중심으로

수행되고 있으며, 이에 따른 결과가 2003년에 발표될 예정이다.

3. MDA 기반 컴포넌트 통합 개발 도구의 특성 및 기능

본 논문에서는 당 연구팀에서 개발중인 MDA의 핵심 기술 및 추가적인 요소들을 이용해 MDA에 기반한 CBD 접근방법을 소개한다. 이 접근방식에서는 CBD 요소들 뿐만 아니라 도메인 아키텍처의 재사용, 플러그 앤 플레이 방식의 시각적인 컴포넌트의 조립 등의 추가적인 요소들이 포함된다. 이 접근 방식의 특성들은 다음과 같다.

- 아키텍처 기반의 접근방식 : 컴포넌트들로 구성된 시스템 아키텍처 내에서 컴포넌트 개발 프로세스와 컴포넌트 조립 프로세스를 통합한다.
 - 아키텍처 재사용 : 상위 추상화 레벨의 컴포넌트 아키텍처를 재사용하여 컴포넌트 재사용 과정의 생산성, 비용, 품질을 향상시킬 수 있다.
 - 자동화된 컴포넌트 식별 : 비즈니스 모델링 결과로부터 객체 모델과 유즈케이스 모델간의 관계, 그리고 객체 모델과 액티비티 모델들간의 관계를 이용해 자동으로 컴포넌트를 식별한다.
 - 다양한 플랫폼 기반의 컴포넌트들의 합성 (Composition) : 기존에 개발한 COBALT Assembler의 경우 J2EE/EJB 컴포넌트들의 합성만을 지원한 반면, MDA 접근방식에 의하면 J2EE, CORBA, .NET 등과 같이 여러 컴포넌트들의 합성을 지원할 수 있다.
 - 기존 CBD 지원도구와는 차별화된 요소 제공 : 플러그 앤 플레이 방식의 시각적인 컴포넌트 조립 및 마이너리 컴포넌트의 개조 기능을 제공한다.
- 위에 언급된 비교우위의 기능을 지원하는 MDA 기반 CBD 도구인 MDA 기반 컴포넌트 통합 개발 도구를 개발 하고 있다. MDA 기반 컴포넌트 통합 개발 도구는 컴포넌트와 웹 서비스에 대한



(그림 3) MDA 기반 컴포넌트 통합 개발 도구의 구조도

비즈니스 모델링, PIM 모델링, PSM 모델링, 구현, 시험 및 서버로의 전개 기능 등을 포함한다. 아울러 컴포넌트들의 조립을 통한 어플리케이션의 생성을 지원하며, 웹 서비스들의 조립을 통한 웹 서비스 기반의 어플리케이션의 생성을 지원하는 광범위한(Wide Spectrum) 도구다. (그림 3)은 13개의 서브시스템으로 구성된 MDA 기반 컴포넌트 통합 개발 도구의 구조를 나타내며 각 서브시스템의 기능은 다음과 같다.

3.1 프로세스 관리기

프로세스 관리기는 프로세스 단계를 정의하고 각 단계에 대한 적용 프로파일 및 제한 조건을 지정하여 새로운 프로세스를 생성할 수 있으며, 프로세스 진행 도중 생성/소멸/변경되는 모델들을 추적하고 관리하며, 보고서 생성기를 통해 프로세스에서 산출되는 보고서를 생성할 수 있다.

3.2 비즈니스 모델 생성기

비즈니스 모델 생성기는 시스템의 전체 프로세스를 보여주며 개발하는 시스템에 대한 기능적,

비기능적 요구사항을 분석한 후 컴포넌트를 식별한다.

3.3 PIM 생성기

PIM 생성기는 개발하는 시스템을 위한 아키텍처를 정의하고 시스템의 세부 요소들을 UML 1.5[3] 기반으로 모델링하며, 생성된 모델들을 검증한다.

3.4 PSM 생성기

PSM 생성기는 PIM을 특정 컴포넌트 플랫폼과 관련된 PSM으로 변환하고 이를 정제 및 검증할 수 있는 기능을 제공하며, 컴포넌트 및 웹 서비스를 조립할 경우 조립을 위한 모델을 정제할 수 있는 기능을 제공한다.

3.5 코드 생성기

코드 생성기는 PSM 정보 및 ASL(Action Semantics Language)[3]을 이용하여 실행 가능한 코드를 생성해내고, 생성된 코드 정보와 PSM 정보를 일치시키는 기능을 제공한다.

3.6 모델 매핑 관리기

모델 매핑 관리기는 MDA에서 시스템이 비즈니스 모델, PIM, PSM 등 다양한 관점 사이의 모델 변환을 위한 매핑 규칙을 정의하며, 매핑 엔진을 이용하여 모델 변환을 자동화 한다.

3.7 메타 모델 정의기

메타 모델 정의기는 UML 확장 규칙을 이용하여 새로운 UML 프로파일을 정의하고, 이를 MOF 규칙에 따라 검증한다. 또한 메타 편집기를 이용하여 UML 프로파일을 위한 모델러를 생성한다.

3.8 설계 패턴 관리기

설계 패턴 관리기는 비즈니스 모델 생성기, PIM 생성기, PSM 생성기에서 자주 사용되는 설계 정보들을 패턴화하여 정의할 수 있도록 지원하며, 관련 패턴들을 설계과정에서 적용할 수 있도록 지원한다.

3.9 컴포넌트/웹 서비스 전개기

컴포넌트/웹 서비스 전개기는 PSM 단계에서 모델링 되고 코드 생성기를 통하여 생성되고 정제되며 컴파일 과정을 거친 모든 컴포넌트 및 웹 서비스 코드에 대한 패키징과 서버에 전개하는 과정을 지원한다.

3.10 시험기

시험기는 단위 컴포넌트와 합성 컴포넌트의 인터페이스를 시험할 수 있는 기능을 제공하며, 또한 단위 웹 서비스와 합성 웹 서비스의 인터페이스를 시험할 수 있는 기능을 제공한다.

3.11 모델 정보 저장소

모델 정보 저장소는 도구 전체의 메타 모델 정보와 모델 정보를 저장, 관리하는 기능을 제공하며 모델 정보를 XMI 표준에 따라서 XML 문서로 변환하는 기능을 제공한다.

3.12 컴포넌트/웹 서비스 저장소

컴포넌트/웹 서비스 저장소는 생성된 컴포넌트와 웹 서비스를 저장소에 저장하고 저장된 컴포넌트와 웹 서비스의 정보를 검색하거나 수정, 삭제 및 관리할 수 있는 기능을 제공한다.

4. 결론 및 향후 연구

CBD는 현재 소프트웨어 개발에 있어서 보다 나은 방법으로 채택되었으나 기존의 CBD 개발 도구들은 특정한 미들웨어 플랫폼을 지원하는 도

구로 개발될 수 밖에 없었다. 그러나 개발 환경은 J2EE, CORBA, .NET 등의 다양한 미들웨어 플랫폼을 지원해야 하므로 플랫폼 간의 시스템 통합이나 상호운용성의 문제로부터 발생하는 요구를 만족시키기 위해서 새로운 개발 패러다임인 MDA를 지원하여야 한다.

본 논문에서는 기존에 수행했던 CBD 프로젝트의 결과를 기반으로 MDA 표준에 기반을 둔 MDD(Model Driven Development) 패러다임을 지원하기 위한 접근 방식을 지원하고자 한다. 본 논문의 접근 방식을 기존의 MDA 방식과 비교해보면 아키텍처 재사용, 컴포넌트 자동 식별, 컴포넌트의 개조, 컴포넌트 개발 프로세스와 조립 프로세스의 통합, 그리고 다양한 컴포넌트 플랫폼에서 개발된 컴포넌트들의 조립이라는 측면에서 기존의 MDA 방법과는 차별화된 기능을 제공한다. 현재 MDA 기반 컴포넌트 통합 개발 도구는 상세 설계를 기반으로 구현 중에 있으며, PIM2PSM 변환기 프로토타입이 완료되었고 사례 적용을 통해 모델 매핑 규칙을 검증할 것이다. 모델간의 일치성이나 완전성 체크는 정형화된 모델 검증 프로세스에 의해 지원될 것이며, 자동화된 코드 생성 기능을 제공하여 모델 정보와 ASL을 기반으로 실행가능한 코드를 생성하는 것을 목표로 하고 있다.

참고문헌

- [1] Object Management Group, "Model Driven Architecture", OMG document ormsc/01-07-01.
- [2] Jon Siegel and the OMG Staff Strategy Group, "Developing in OMG's Model Driven Architecture", <ftp://ftp.omg.org/pub/docs/omg/01-12-01.pdf>, November 2001

[3] Action Semantics Models, Unified Modeling Language Specification, Version 1.5, OMG document, formal/03-03-01, March 2003

[4] Object Management Group, "Meta Object Facility(MOF)Specification, Version 1.4", OMG document formal/2002-04-03

[5] Object Management Group, "Common Warehouse Metamodel (CWM) Specification, Version 1.0", OMG document ad/01-02-01

[6] XMI 1.1 Specification, OMG document <http://cgi.omg.org/cgi-bin/doc?ad/99-10-02>

[7] Woo-Jin Lee, Oh-Cheon Kwon, et al., "A method and tool support for identifying domain components using object usage information", ETRI Journal, April, 2003.

[8] Oh-Cheon Kwon, Gyu-Sang Shin, et al., "Technique for Adapting EJB Components according to Adaptation Patterns", ISE2001, Las Vegas, USA, June 2001

[9] Interactive Objects Software, ArcStyler, <http://www.io-software.com>

[10] Compuware, "Using OptimalJ", <http://javacentral.compuware.com>

[11] Objecteering Software, Objecteering/UML, <http://www.objecteering.com>

[12] ISO/IEC & ITU-T: Information technology - Open Distributed Processing - Part 1 Overview - ISO/IEC 10746-1 | ITU-T Recommendation X.901

[13] opengroup, <http://www.opengroup.org/combine/overview.htm>

저자약력



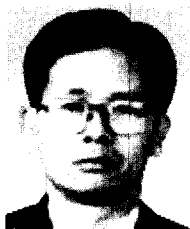
김민정

1998년 서강대학교 컴퓨터학과 졸업(학사)
 2000년 서강대학교 컴퓨터학과 대학원 졸업(공학 석사)
 2000년 - 현재 ETRI 컴퓨터.소프트웨어기술연구소 S/W컨텐츠연구부 연구원
 관심분야 : CBD, MDA, 방법론, 웹 서비스 등
 이 메 일 : minjkim@etri.re.kr



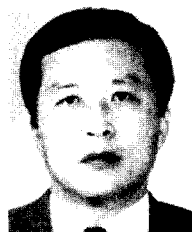
윤석진

1992년 중앙대학교 컴퓨터공학과 졸업(학사)
 1994년 중앙대학교 대학원 컴퓨터공학과 졸업(이학석사)
 1994년 시스템공학연구소 연구원
 1997년 ETRI 선임연구원
 현재 ETRI 컴퓨터.소프트웨어기술연구소 S/W컨텐츠연구부 컴포넌트공학팀 선임연구원
 관심분야 : CASE, S/W 컴포넌트 기술, 임베디드 시스템, Reverse Engineering
 이 메 일 : sjyoon@etri.re.kr



권오천

1994년 영국 Teeside 대학교 대학원 졸업 (소프트웨어공학 석사)
1998년 영국 Durham 대학교 대학원 졸업 (공학 박사)
1991년 미국 RTP IBM 연구소 객원 연구원
현재 ETRI 컴퓨터, 소프트웨어기술연구소 S/W컨텐츠연구부
책임연구원
관심분야 : 재사용, 컴포넌트 기반 개발(CBD), Product
Line, 웹 서비스 기술 등
이 메 일 : ockwon@etri.re.kr



신규상

1981년 성균관 대학교 통계학과 졸업(학사)
1983년 서울대학교 대학원 계산통계학과 졸업(이학석사)
2001년 충남대학교 대학원 컴퓨터과학과 (이학박사)
1983년 시스템공학연구소 연구원
1987년 시스템공학연구소 선임연구원
1997년 시스템공학연구소 책임연구원
현재 ETRI 컴퓨터, 소프트웨어기술연구소 S/W컨텐츠연구부
컴포넌트공학팀 팀장
관심분야 : CASE, S/W 컴포넌트 기술, 웹 서비스 기술, 멀티
미디어 시스템
이 메 일 : gsshin@etri.re.kr