

엔터프라이즈 엔지니어링 기반의 정보시스템 개발방법 진화 경향 및 특징

국방대학교 이태공

1. 서 론

1946년 컴퓨터가 개발된 후 정보시스템 개발을 위한 수많은 방법(론)[1,2]이 제시 되었으며, 이러한 방법(론)들에 의해 개발된 정보시스템은 조직 업무의 효율성, 효과성 및 생산성 향상 등에 크게 기여하였다. 그러나 기존의 방법(론)에 의해 개발된 정보시스템들은 중복투자, 과도한 유지보수비, S/W컴포넌트의 재사용, 정보시스템의 통합 및 상호운용성 등의 문제점을 안고 있다.

1990년도부터 개발된 도메인 엔지니어링 기반 정보시스템 개발방법(론)[3,4]은 특정 도메인 내의 중복투자, 유지보수비, S/W컴포넌트의 재사용이나 정보시스템의 통합성 및 상호운용성 문제를 부분적으로 해결하였다(2). 그러나 이 방법(론)도 엔터프라이즈 범위의 S/W컴포넌트의 재사용 또는 정보시스템 간 통합 및 상호운용성 등의 문제를 해결하지 못한 결과 중복투자 및 과도한 유지보수비 등의 문제가 있다.

이러한 문제점 해결을 위해 2000년도부터 엔터프라이즈 엔지니어링 기반 정보시스템 개발방법론이 제시 되어[5,6] 성과가 가시화 되고 있으며, 이 분야 연구가 가속화 되고 있는 실정이다.

본 고에서는 첫째, 엔터프라이즈 엔지니어링 기반 정보시스템 개발방법의 동향을 이해하기 위하여 지금까지 개발된 중요한 소프트웨어 개발방법과 그 특징을 분석한 후 둘째, 제시한 방법의 특징을 몇 가지 기준에 의해 비교하고자 한다.

2. 정보시스템 개발방법 진화 경향 및 특징

정보시스템 개발방법은 조직의 업무 및 정보기술의 변화에 따라 진화하고 있다. 즉, 1세대(1950-1990)인 소프트웨어 엔지니어링 기반에서, 2세대(1990-2000)인 도메인 엔지니어링 기반을 거쳐, 지금은 3세대(2001-)인 엔터프라이즈 엔지니어링 기반으로 진화하고 있다.

2.1 1세대: 소프트웨어 엔지니어링 기반 정보 시스템 개발 방법 및 특징

소프트웨어 엔지니어링의 IEEE 정의는“소프트웨어의 개발, 운영, 유지보수를 함에 있어 체계적이고, 훈련이 잘 되어 있고, 정량적인 접근 방법을 적용하는 것을 의미하며, 이는 곧 소프트웨어 개발을 공학 개념으로 접근하는 것”이다(7). 이 개념을 적용한 대표적인 방법이 폭포수 모델이다.

이러한 방법에 의해 개발된 정보시스템들의 특징은, (i)단위기능 최적화를 목표로 하고, (ii)단위기능 관점이며, (iii)시스템 간 상호운용성이 미흡하고, (iv)시스템 간 통합은 인터페이스에 의해서 하며, (v)재사용가능한 컴포넌트 개발이 극히 어려우며, (vi)TCO관점보다 개발비 절약 개념으로 개발하여 유지보수비가 기하급수적으로 증가하고, (vii)정보화 분야 중복투자가 과도하게 발생한다.

2.2 2세대: 도메인 엔지니어링 기반 정보시스템 개발방법 및 특징

도메인 엔지니어링은 도메인 내의 어플리케이션 개발 시 사용될 수 있는 중요한 산출물들을 개발하기 위한 프로세스로, 도메인내의 유사점과 차이점, 기획 등의 분석을 통해 재사용 가능한 컴포넌트를 확인하고 개발하는 것이다.

그림 1은 1996년에 소프트웨어 기술 컨퍼런스(8)에서 제시한 방법으로, 도메인 엔지니어링 단계는, (가)도메인 분석에 의해 도메인 모델을 생성하고, (나)도메인 모델을 기반으로 도메인 소프트웨어 아키텍처를 개발하며, (다)도메인 소프트웨어 아키텍처를 기반으로 재사용 가능한 컴포넌트를 개발한다.

어플리케이션 엔지니어링 단계는 도메인 엔지니어링 산출물을 기반으로, (가)새로운 요구사항이 있을 시 도메인 모델을 기반으로 하여 어플리케이션 성능 명세를 작성하고, (나)어플리케이션 성능 명세서와 도메인 아키텍처를 기반으로 어플리케이션 소프트웨어 아키텍

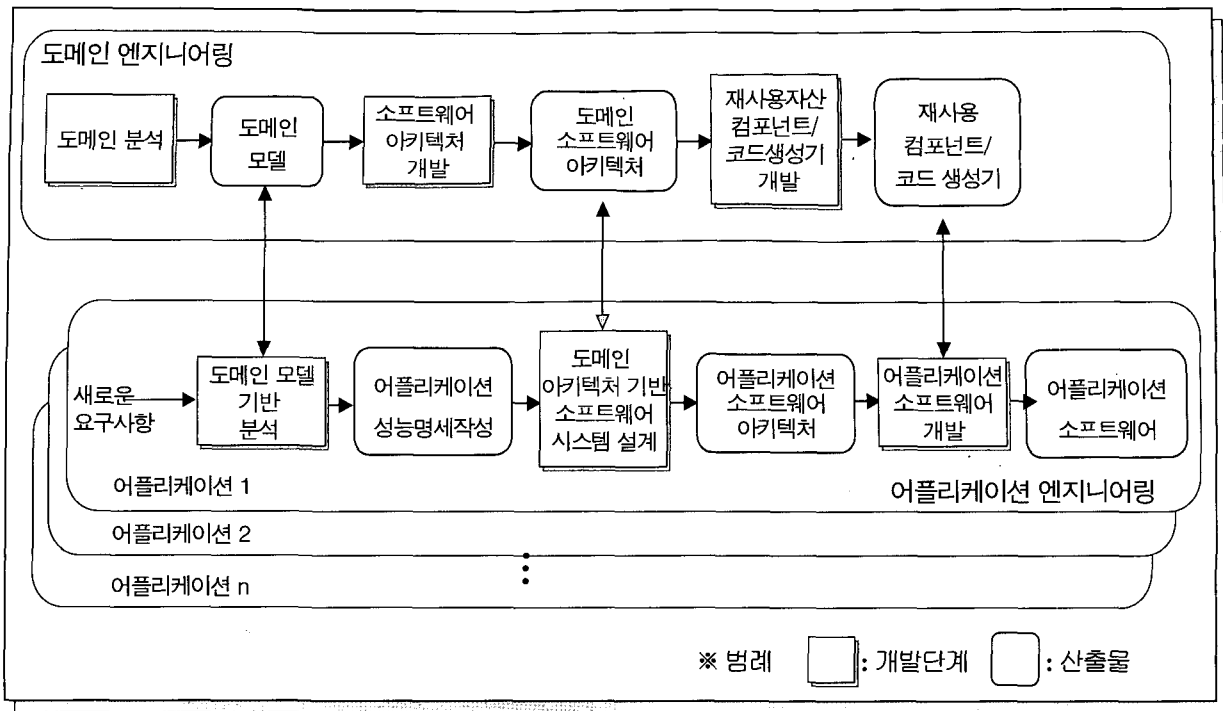


그림 1 도메인 엔지니어링에 의한 시스템 개발방법

처를 개발하며, (다)어플리케이션 소프트웨어 아키텍처 및 재사용 가능한 컴포넌트를 기반으로 어플리케이션을 개발한다.

이 방법에 의해 개발된 정보시스템들의 특징은, (i) 단위 도메인의 최적화를 목표로 하고, (ii)단위 도메인 관점이며, (iii)도메인 내 시스템 간 상호운용성이 있고, (iv)도메인 내 시스템 간 통합은 표준 및 아키텍처 개념으로 할 수 있으며, (v)도메인 내 재사용 가능한 컴포넌트를 개발할 수 있고, (vi)미비하지만 TCO관점

비용관리를 하며, (vii)도메인 내 정보화 중복투자를 방지할 수 있다.

2.3 3세대 : 엔터프라이즈 엔지니어링 기반 정보시스템 개발방법 및 특징

엔터프라이즈 엔지니어링이란 “전체 조직을 어떻게 설계하고 연관된 모든 요소를 개선시킬 것인가를 고려하고 엔지니어링과 분석방법, 도구 등을 이용해서 조직의 목적과 목표를 어떻게 효과적으로 달성시킬 것인가를 고려하는 것”이라고 정의한다[5].

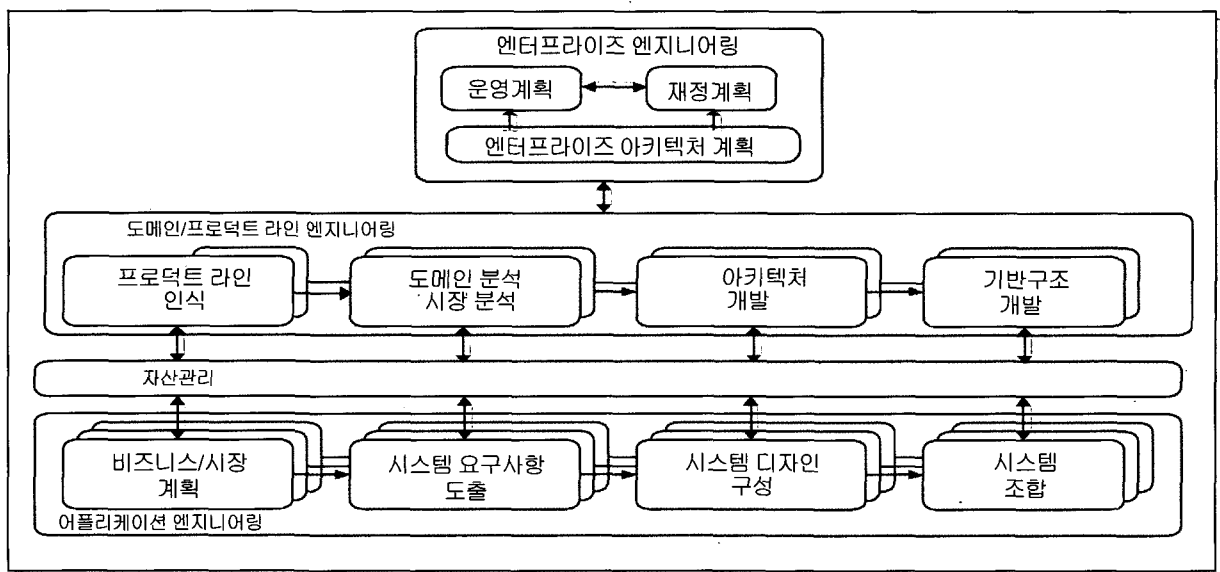


그림 2 Tri-Lifecycle 엔지니어링 모델

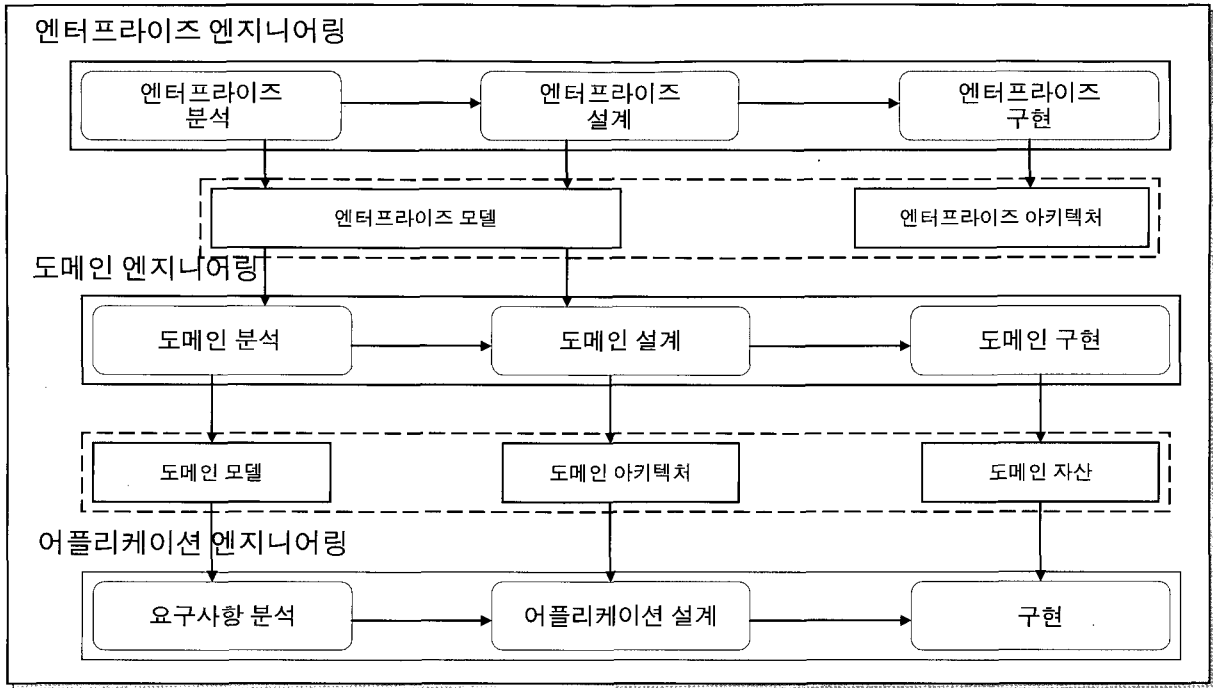


그림 3 GSAM 엔지니어링 모델

2.3.1 Tri-Lifecycle/GSAM 엔지니어링 모델

엔터프라이즈 엔지니어링 개념을 기반으로 개발된 모델은 Lockheed Martin사의 Tri-Lifecycle 그림 2(9)와 미 공군의 GSAM(Guidelines for Successful Acquisition and Management) 그림 3(10)이다.

이 두 모델의 엔터프라이즈 엔지니어링 단계는 전체 엔터프라이즈를 대상으로 상위수준의 데이터와 운용요구를 수렴하여 엔터프라이즈 참조모델과 엔터프라이즈 아키텍처를 개발하는 것이고, 도메인 엔지니어링 단계는 엔터프라이즈 엔지니어링의 산출물을 기반으로 도메인 아키텍처를 개발하게 된다. 어플리케이션 엔지니어링 단계는 도메인 엔지니어링을 기반으로 정보시스템을 개발한다.

이 프레임워크에 의해 개발된 정보시스템들의 특징은, (i)엔터프라이즈 범위의 최적화를 목표로 하고, (ii)엔터프라이즈 관점이며, (iii)엔터프라이즈 내 전 시스템 간 상호운용성이 있고, (iv)엔터프라이즈 내 시스템 간 통합은 아키텍처에 의해 가능하며, (v)엔터프라이즈 내 재사용 가능한 컴포넌트 개발이 가능하고, (vi)엔터프라이즈 정보화 중복투자를 방지할 수 있다.

2.3.1.1 LL-엔터프라이즈 엔지니어링 기반 정보시스템 개발 프레임워크

이 프레임워크는 엔터프라이즈 엔지니어링을 도메인 과 어플리케이션 엔지니어링으로 구분하였고 각각의 엔지니어링 단계별 세부 프로세스를 그림 4과 같이 정

의하였다(11).

이 프레임워크의 첫 번째 단계는 엔터프라이즈 엔지니어링을 통해 엔터프라이즈 범위 내의 모든 아키텍처 개발 시 공통으로 참조해야 하는 참조모델을 개발할 뿐 아니라, 개발된 엔터프라이즈 참조모델을 기반으로 부-엔터프라이즈 아키텍처를 개발한다.

엔터프라이즈 엔지니어링의 두 번째 단계인 도메인 엔지니어링은 부-엔터프라이즈 아키텍처 개발 프로세스의 목표아키텍처 개발 단계에서 결정된 도메인 범위(부-엔터프라이즈 또는 부-엔터프라이즈 범위)내의 재사용 가능한 소프트웨어 컴포넌트를 구축하는 과정이다.

세번째 단계는 어플리케이션 엔지니어링으로 부-엔터프라이즈 아키텍처 개발 과정의 구현계획에서 결정된 프로젝트를 어플리케이션으로 개발하는 것이다. 이 단계에서는 도메인 엔지니어링을 통하여 개발된 재사용 가능한 컴포넌트를 이용한다.

이 방법에 의해 개발된 정보시스템들의 특징은, (i)엔터프라이즈 범위의 최적화를 목표로 하고, (ii)엔터프라이즈 관점이며, (iii)엔터프라이즈 내 전 시스템 간 상호운용성이 있고, (iv)엔터프라이즈 내 시스템 간 통합은 아키텍처에 의해 가능하며, (v)엔터프라이즈 내 재사용 가능한 컴포넌트 개발이 가능하고, (vi)TCO 관점 투자 관리이며, (vii)엔터프라이즈 내 정보화 중복투자를 방지할 수 있고, (viii)엔터프라이즈 아키텍처에 의해 프로젝트가 식별되고, 우선순위 결정된 전환 계획에 시스템을 개발할 수 있다

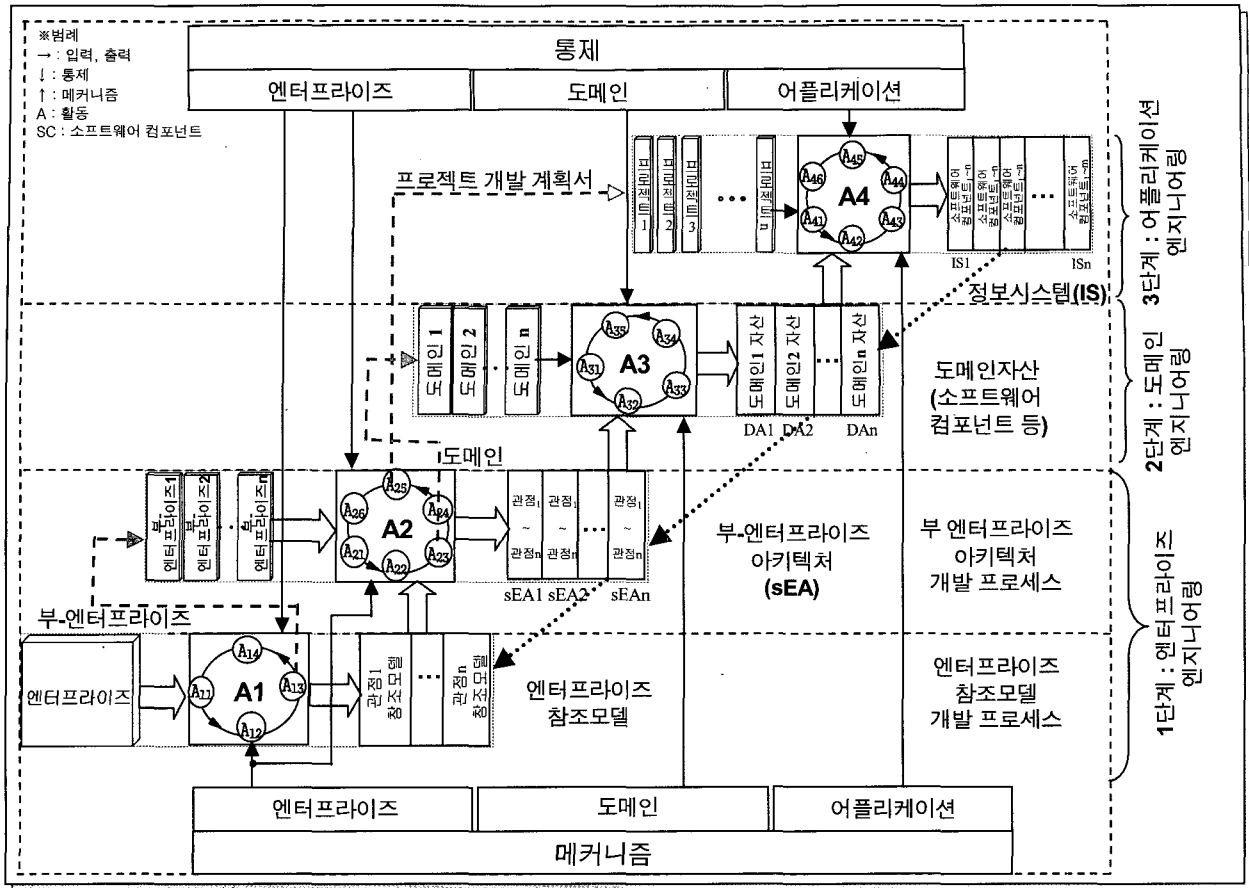


그림 4 LL-엔터프라이즈 아키텍처 기반 정보시스템 개발 프레임워크

더구나 이 프레임워크는 기존 엔터프라이즈 엔지니어링 보다 프로세스 뿐 아니라 관련성 정도가 보다 정형화 되었다.

2.3.1.2 모델 및 프레임워크 평가

표 1은 소프트웨어 엔지니어링, 도메인 엔지니어링 과 엔터프라이즈 엔지니어링 기반 정보시스템 개발 방법을 평가한 것이다. 평가 항목은 총 7개로, 관점적용, 범위, TCO, 정형화 정도, 프로젝트개발 우선순위, 엔지니어링 목적 달성 정도이다.

관점요소를 명시적으로 적용한 모델은 LL-엔터프라이즈 엔지니어링 기반 정보시스템 개발 프레임워크 뿐이며, 기존의 엔터프라이즈 엔지니어링 기반 모델은 명시적으로 관점요소가 적용되었다고 판단된다.

범위 측면에서도 엔터프라이즈 엔지니어링 기반의 모델 및 프레임워크를 제외한 기타 모델에서는 엔터프라이즈 전체 측면 범위가 아니다.

TCO의 개발비 측면에서는 엔터프라이즈 엔지니어링을 기반으로 한 정보시스템 개발방법은 초기 엔터프라이즈 아키텍처 구축에 따른 예산 및 개발 기간은 더 소요될 것이나, 유지보수비 및 교육훈련비가 절감된다.

표 5 LL-엔터프라이즈 엔지니어링 기반 정보시스템 개발 프레임워크와 기존 모델과의 비교

※범례 : ○ : 우수, △ : 보통, - : 미흡

항목	소프트웨어 엔지니어링	도메인 엔지니어링	엔터프라이즈 아키텍처 기반 정보시스템 개발모델		
			Tri-Life Cycle	GSAM	LL-F/W
관점적용	-	-	△	△	○
대상 범위	엔터프라이즈	-	△	○	○
	도메인	-	○	○	○
	어플리케이션	○	○	○	○
TCO	개발비	○	△	-	-
	유지보수비	-	△	○	○
	교육훈련비	-	△	○	○
	전체 합계	-	△	○	○
모델의 정형화 정도	어플리케이션 개발기간	-	△	○	○
	엔지니어링간의 연계성	-	△	△	○
	입출력요소의 명확성	○	○	-	○
프로젝트 개발 우선순위 고려	산출물 간의 연계성	○	○	-	○
	프로젝트 개발 우선순위 고려	-	-	△	○
엔지니어링 목적(정렬성, 통합성, 상호운용성, 재사용성, 이식성 등)	-	△	△	○	○

어플리케이션 개발기간 측면에서 비교해 볼 때도 엔터프라이즈 엔지니어링 기반 정보시스템 개발모델에서는 재사용 자산을 이용하여 시스템을 개발하므로 개발기간 및 비용이 단축된다.

모델의 정형화 정도에서 고려하면, 먼저 엔지니어링간의 연계성 측면에서 볼 때 도메인 엔지니어링은 도메인과 어플리케이션간의 연계성은 우수하나 엔터프라이즈 엔지니어링과의 연계가 없다. 또한 Tri-Lifecycle 모델 및 GSAM 모델은 구체적인 연계에 대한 언급이 없음을 알 수 있다.

입출력 요소의 명확성 측면에서 볼 때도 소프트웨어 엔지니어링과 도메인 엔지니어링 모델은 개발모델이 오랜 기간을 거쳐서 정형화 정도가 높으나, Tri-Lifecycle 모델 및 GSAM 모델은 프로세스가 명확하지 않음을 알 수 있다. 산출물간의 연계성 측면에서도 이와 유사하다.

엔지니어링의 목적 달성정도 평가는 엔터프라이즈 전체의 최적화를 고려한 LL-엔터프라이즈 엔지니어링 기반 정보시스템 개발 프레임워크이 우수함을 알 수 있다.

3. 동향 및 결론

지금까지 살펴본 바와 같이 컴퓨터가 탄생하고 난 후부터 지금까지 효율적이고 효과적인 정보시스템 개발을 위한 수많은 엔지니어링 방법들이 개발되어 적용되어왔으며, 각각의 엔지니어링 방법은 저마다 장점과 단점을 가지고 있는 그 시대의 최선의 방법이었다.

그러나 제시된 대다수 엔지니어링 방법은 단위 기능 또는 프로세스 최적화를 위한 방법이어서, 엔지니어링의 목표인 정렬성, 유연성, 상호운용성, 재사용성, 이식성, 통합성, 유지보수비 등의 문제를 해결할 수 없는 실정이다.

이러한 문제 해결을 위한 최선의 접근방법이 엔터프라이즈 또는 도메인 전체의 최적화를 목표하는 엔지니어링 방법이다. "엔터프라이즈 아키텍처 기반 정보시스템 개발 방법"이 이러한 문제 해결을 위한 엔지니어링 방법이다.

국외에서는 지금 많은 학자들이 "엔터프라이즈 엔지니어링기반 정보시스템 개발방법"을 연구하고 있을 뿐 아니라 이러한 방법을 적용하여 많은 조직이 엔지니어링의 문제점을 해결 하였다.

국내에도 정부 및 기업에서 이러한 개념을 가진 방법론을 일부 적용하고 있으나 "엔터프라이즈 엔지니어링기반 정보시스템 개발 방법"연구는 미흡한 실정이다.

참고문헌

- [1] Royce W. W., "Managing the development of large software systems: concepts and techniques," IEEE WESTCON, IEEE Computer Society Press, 1977.
- [2] Sommerville, "Software Engineering ver.7," Addison Wesley, 2004.
- [3] Carnegie Mellon Software Engineering Institute, "Domain Engineering & Domain Analysis," 2004.
- [4] Booch G., Rumbaugh J., and Jacobson I., "The Unified Modeling Language User Guide," Addison-Wesley, 1999.
- [5] D. H. Liles, B. L. Huff, K. J. Rogers, "A manufacturing Reference Model for the Enterprise Engineer," Univ Texas at Arlington, 2000.
- [6] 임춘혁, "정보기술구조 개발 프레임워크에 관한 연구", 국방대학교 2000.
- [7] Institute of Electrical and Electronics Engineers. IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries. NY:1990.
- [8] Foreman, John. Product Line Based Software Development-Significant Results, Future Challenges. Software Technology Conference, 1996.
- [9] Frea A. Maymir-Ducharme, PhD, "Extending Enterprise And Domain Engineering Architectures to Support the Object Oriented Paradigm," Lockheed Martin, Mission Systems, 1998.
- [10] Department of the Air Force Software Technology Support Center "Guidelines for Successful Acquisition and Management of Software-Intensive Systems: Weapon Systems Command and Control Systems Management Information System," 1999.
- [11] 이성호, 이태공, "엔터프라이즈 아키텍처 기반 정보시스템 개발모델", 정보기술아키텍처 연구 제2권 제1호, 2005.
- [12] US DoD, "DoD Architecture Framework Version 1.0," 2003.

이 태 공



1976 공군사관학교 전자과(학사)
1986 Naval Postgraduate School,
Monterey California,
U.S.A. (체계관리석사)
1991 Wayne State University, Detroit
Michigan, U.S.A. (전산학박사)
2003~현재 한국 ITA 학회장
2004~현재 국방대학교 정교수
2005~현재 국방대학교 직무연수부장
E-mail : tglee@kndu.ac.kr
