

UML 메타모델링과 모델의 변환을 통한 전자정부 표준 프레임워크 기반의 코드 생성 자동화

이승한*, 박재표²

¹엔키소프트, ²송실대학교 정보과학대학원

An Automatic code generation through UML Meta modelling and transformation of Model for electronic government framework

Seung-Han Lee^{1*}, Jae-Pyo Park²

¹Department of Research & Development, Enkisoft

²Graduate School of Information Science, Soongsil University

요약 UML 모델을 사용하는 다양한 소프트웨어의 설계 및 구현 환경에서 UML 메타 모델의 규칙을 준수하고, 이를 통하여 확장하면 많은 장점을 가질 수 있다. 하지만 UML 메타 모델은 자체 규모가 점진적으로 방대해지고 있으며 UML 메타 모델을 사용하는 다양한 곳에서 UML 메타 모델의 확장 및 변환을 위해서는 반드시 Profile의 정의를 통하여 다이어그램을 재정립할 필요가 있다. 즉, UML 메타모델을 확장하여 사용하고자 하는 대상에 대하여 요소들만을 추출하여 사용할 필요가 있다. UML 메타모델의 확장과 Profile을 메타 저장소를 기반으로 재정의하여 사용함으로써 UML 모델링 도구나 분석도구를 좀 더 쉽고 빠르게 개발할 수 있고, 이러한 도구를 활용하여 SW 산업에서의 개발 품질을 높일 수 있다. 본 논문에서는 UML 메타모델의 확장을 통하여 Profile을 재정의 하는 알고리즘을 제시하고, 전자정부 표준 프레임워크에 실제로 적용한 결과를 코드 사이즈와 복잡도를 비교하여 향상된 성능을 보여준다.

Abstract In the process of extending the UML model for a various domain, comply with the UML metamodel and it is possible to obtain a number of advantages. However, the UML metamodel is necessary to redefine the diagram must be defined via the Profile in order to expand and transformation the UML metamodel from a variety of sources using the UML metamodel is becoming increase massive scale. it is necessary to use only those extracts the element relative to the target to be used to extend a UML metamodel. Used to re-define the extension of the UML Metamodel and Profile based UML modeling tools and metadata repositories by analysis tools, can develop more quickly and easily, by utilizing these tools can improve the quality of development in the SW industry. In this paper, we present an algorithm that of the profile through the expansion of the UML metamodel and shows the results in actually applying e-government standards framework.

Key Words : Meta Model, UML, SW Quality, MDD, Electronic government standard framework

1. 서론

UML 메타모델은 UML 다이어그램의 표기법과

내용을 나타내는 UML의 상위 개념 모델로 제약이 포함된 UML의 클래스 다이어그램 형식으로 표현된다[1]. 이 UML 메타모델을 준수하면 UML을 사용하는 모든 곳에서

This work was supported by the ICT R&D program of MSIP/IITP. [10047178, Development a MDD Tool based on Open Source Software, for the expansion of E-Government Standard Framework usage]

*Corresponding Author : Seung-Han Lee(Enkisoft)

Tel: +82-2-419-6362 email: seunghan.lee@enkisoft.co.kr

Received March 2, 2015

Revised April 21, 2015

Accepted May 7, 2015

Published May 31, 2015

객체 정보들에 대한 호환이 가능하다. 즉, 서로 다른 UML 모델링 도구라도 UML 메타모델의 매핑 규칙을 따르면 쉽게 다이어그램의 내용을 서로 호환할 수 있다[2]. 하지만 UML 메타 모델은 점점 규모가 커지고 있으며 UML 메타모델을 사용하는 조직이나 도구에서는 UML 메타모델의 확장을 필요로 하는 경우가 있다. 예를 들어, 필요에 따라 UML 메타 모델 사용자 보안(Secure) 코드 규칙을 정의하거나 새롭게 정의되어진 컴포넌트 기반의 프레임워크 구성에 적용하기를 원할 수도 있다. 현재의 UML 메타 모델의 명세는 모든 다이어그램 요소를 표현하기 위해 260여 개의 모델요소(Model Elements)들을 포함하고 있다. UML 다이어그램들 중에서 특히 전자정부 표준 프레임워크 환경에서도 호환이 가능하게 하기 위해서는 기존의 UML 메타 모델에서 사용자가 필요로 하는 형태로 확장을 해야 할 필요가 있다[3].

본 논문에서는 UML의 특정 다이어그램 요소들 간의 확장 및 가시성을 고도화 하는 알고리즘을 사용하여 모델의 변환에 적용한 사례를 설명한다.

2. 관련연구

2.1 연구 배경

소프트웨어 구축 기술은 다양해지고 있지만, 이러한 기술의 통합을 요구하는 시스템은 점점 많아지고 있는 현실에서 이를 통합하기 위한 기법으로 OMG(Object Management Group)에서 모델 기반 아키텍처(MDA)/모델 기반 개발(MDD)을 정의하고, 이를 활용하여 다양한 시스템을 구축할 것을 권고하고 있다. 다양한 업무 도메인 뿐만 아니라 공공정보화에서 사용되는 전자정부 표준 프레임워크를 지원하는 모델 기반 개발(MDD)을 위한 절차, 기법 및 도구를 필요로 한다. 또한 비즈니스 패러다임의 변화에 따른 빠른 시장 대응(Time-to Market)을 위하여 요구 증가 및 비용 감소, 높은 생산성에 대한 필요성 증가, 재 사용성 증대를 통하여 유연한 SW를 구축할 수 있는 절차와 메커니즘에 대한 필요성이 점점 더 증가되고 있다.[4] 이러한 추세에도 불구하고,

IT서비스 공공분야 SW 공학 10개 주요 영역에서 그 수준을 조사한 바에 따르면 10개 영역에서 공공부문의 수준이 전체(공공, 제조, 금융, 정보통신, 유통서비스) 평균에 비해 낮게 나타나는 것으로 조사되었으며, 이의 개

선이 시급한 것으로 판단된다. 이러한 배경 하에 Fig. 1 과 같이 UML 기반으로 메타 모델을 새롭게 정의하여 다양한 산업 분야에서 활용될 수 있도록 모델링 도구를 설계하였다.

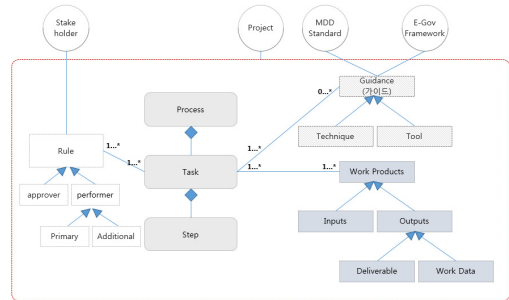


Fig. 1. Redefinition of UML Meta Model

2.2 코드 사이즈 및 복잡도의 측정

본 논문에서는 메타 기반의 객체 지향 모델 설계를 적용함으로써 향상된 결과를 검증할 수 있는 코드 사이즈 및 복잡도(Code Management)로 MDD 도구를 활용하여 작성된 소스코드의 사이즈와 복잡도를 자동으로 측정하는 N'SIQ Collector(오픈소스)의 Plug-in을 설치하여 CI 서버에서 복잡도(Cyclomatic Complexity)가 그림 Fig. 2 와 같이 30 이상인 함수(또는 메소드)가 검출되지 않도록 소스코드의 관리 측면에서 메타 기반 설계의 중요성을 증명할 수 있도록 측정하였다.

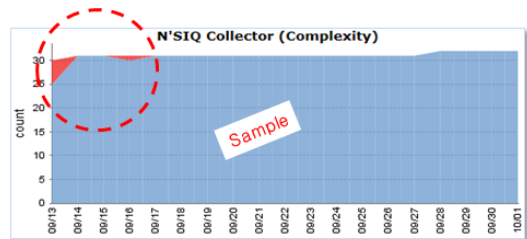


Fig. 2. N'SIQ Collector

3. 제안하는 모델 변환의 기술

3.1 전자정부 프레임워크 기반 절차 재정의

본 논문에서 제안하는 전자정부 표준 프레임워크 기반의 개발 절차는 MDD의 기본 개념과 OMG에서 발표한 MDA 가이드를 UML 메타를 기반으로 프로젝트 환

경에 적합하게 변경하여 적용하였다. 특히 비즈니스의 성과 향상을 위하여 Web-site 프로젝트 구축에 있어서 전자정부표준프레임워크를 활용함으로써 SW의 표준화, 품질 및 재 사용성을 향상시킴으로 서비스의 품질 향상 및 투자 효율성을 높일 수 있는 것이 가능해질 수 있었고, PIM과 PSM을 분리함으로써 PIM을 변경하지 않고도 기술 플랫폼의 변화나 요구사항에 빠르게 대처할 수 있게 되었다.

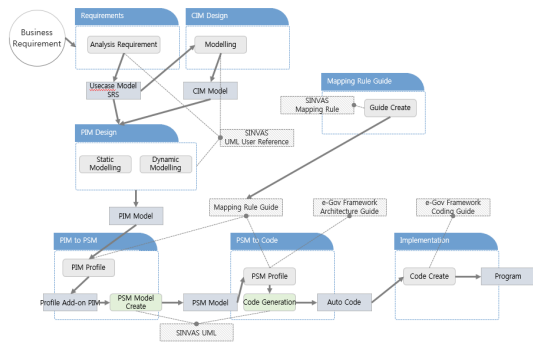


Fig. 3. Development of e-Government framework

UML 메타 정보 기반의 MDA/MDD 기술로 모델 변환의 구현뿐만이 아니라 통합 저장소 환경을 제공함으로써, 효과적인 협업 환경을 지원하여 대규모 프로젝트 및 신규 프로젝트 추진 시 재활용이 가능할 수 있도록 하였으며, 개발 조직의 개발 환경 표준화와 개발 생산성 향상에 크게 기여할 수 있음이 확인되었다.[5] 이러한 통합 저장소 환경은 IT거버넌스 프로세스를 제공하는 정보시스템의 활용이 확대되면서 인력, 프로세스 등 총괄적으로 관리할 수 있는 Enterprise Architecture, Compliance, 서비스관리, 투자대비효과(ROI)등의 업무 추진 환경에 크게 기여할 수 있을 것이다.

또한, 모델의 변환 규칙으로 매우 중요한 컴포넌트 구성에 있어서, 그림 Fig. 4와 같이 정의되어지는 PIM을 통한 PSM 자동 생성 시 Verify 기능을 활용하여 실행 가능한지에 대한 모델 검증은 함으로써, Boundary, Control, Entity의 모델 변환 컴포넌트로 구축되었으며, 이는 메타모델을 정의하여 가능할 수 있음을 확인할 수 있다.

Model	Description
PIM	Platform Independent Model: software engineering is a model of a software system or business system, that is independent of the specific technological platform used to implement
PSM	Platform Specification Model: Model of the solution containing elements for the specific platform. Several applications can share the same way for mapping into a specific platform

Fig. 4. Transformation to Component Model

3.2 데이터 입출력의 자동화

전자정부 프로젝트에서 가장 많이 사용되는 데이터 입출력 기능(Data Access Domain)구현에 활용하는 CRUD(Creation, Read, Update, Delete) 자동화 기능을 UML 도구와 통합하여 개발하여 전자정부 프레임워크의 표준 컴포넌트와 MDD 개발에 따른 상호 데이터 연계에 따른 관리 기능을 개선하였고, 데이터 입출력 기능을 위한 PIM 구성을 위해 Profile을 정의하고 구축하였다.

CRUD 자동화 도구 내에 데이터 입출력 기능을 그림 Fig. 5와 같이 런타임 수행 스크립트 엔진을 탑재하여, 사용자가 필요로 하는 데이터를 도구 내에서 활용할 수 있도록 정의하여 사용자가 원하는 정보를 별도의 추가 과정 없이도 정보를 추출하고, 가공하여 원하는 형태의 개발 산출물을 생성할 수 있도록 하였다.

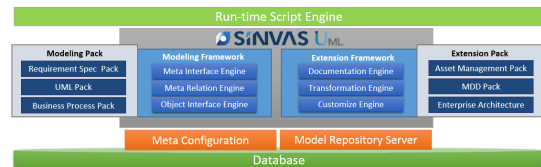


Fig. 5. Run-time Script Engine

이와 같이 모델링 환경에서 메타 모델로 정의되어지는 영역은 매우 광범위 하며, 이를 통하여 Database 설계, Business Process Management등의 SW 아키텍처의 분석, 설계 단계에서 중요한 환경을 효과적으로 제공할 수 있다.

3.3 Meta Profile의 정의

UML 메타 정보를 활용하기 위하여 본 연구에서는 UML 모델링 환경을 메타 기반으로 생성 및 가공이 가능할 수 있도록 하는 메타 Profile의 환경 구성을 제공하

여 사용자들로 하여금 그림 Fig. 6과 같이 필요로 하는 객체에 대한 다양한 재 정의의 가능성을 가능하도록 구성하였다.

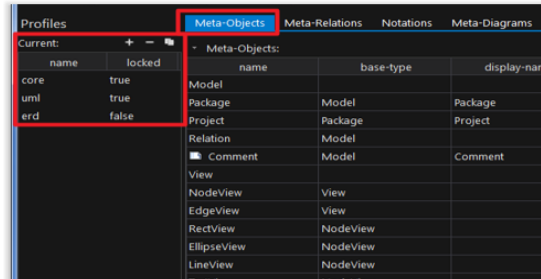


Fig. 6. Meta Rule Editor

통합 모델링 환경을 통해 제공되는 이러한 메타 정보들은 복잡한 비즈니스 환경에서 User Interface 구성 및 다양한 신규 프레임워크들에 대한 재 활용이 가능해짐으로서 프로젝트 환경의 분석/설계 단계에 큰 도움을 줄 수 있게 될 것이다.[6]

4. 성능평가 및 비교분석

본 장에서는 제안하는 UML 메타 기반의 MDA/MDD 기술로 기존의 프로젝트 코드 사이즈 및 복잡도와 비교하여 MDD를 적용한 사례 프로젝트의 자동 코드 생산성이 크게 향상 되었 음을 확인할 수 있었다.

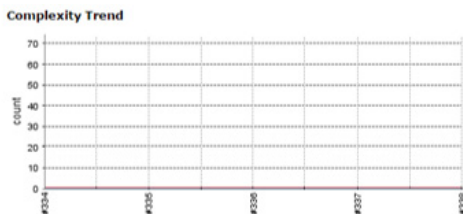


Fig. 7. PMD plug-in Inspection

자동 생성된 소스코드의 불필요한 부분을 찾아내어 성능을 높일 수 있도록 리포팅하는 PMD plug-in을 CI서버에 설치하여 제거해야할 수준에 따라 High/Normal /Low Priority 건수를 자동으로 검출하였고, 그림 Fig. 7과 같이 적용 결과는 0건이었고, 그 공식은 다음과 같다.

복잡도(Cyclomatic Complexity) = E - N + P이며 여기서 E는 변수의 개수이고, N은 노드의 개수를 의미하

며, P는 연결된 컴포넌트 (Connected Component)의 개수를 정의한다.[7]

본 연구 과제에서 목표로 하였던 복잡도가 30이상인 함수(또는 메소드)의 발견 건수가 전혀 존재하지 않았을 뿐만 아니라, 복잡도가 30 이상인 함수(또는 메소드)의 발견 건수도 없었기 때문에 메타 기반의 모델링된 정보들로 자동 생성된 소스코드들의 SW 품질이 기존 개발 환경 보다 우수함을 확인할 수 있었다.

5. 결론

UML 메타 기반의 MDA/MDD 기술은 모델을 기반으로 하는 시스템의 구현 시 주요 과업, 산출물, 가이드에 대하여 자동 생성하는 기술로서, 모델의 변환, 코드의 자동 생성, 코드의 역 변환, 산출물의 자동 생성 등 다양한 시스템 환경으로의 산출물 변환과 모델 검증 등의 분야에서 메타 기술을 적용하였다. 그 결과 개발 생산성 및 코드의 품질을 향상시키는 효과를 얻을 수 있었고, 코드와 산출물을 메타 모델을 통하여 자동 생성함으로써 설계와 코드, 그리고 산출물간의 일관성을 유지가 용이해졌으며 모델 검증의 자동화를 통하여 소프트웨어의 설계 품질을 향상시키고 체계적으로 수행하는 것이 가능해졌다.

또한, UML 메타 기반의 전자정부 표준 프레임워크의 적용을 위한 Rule 기반의 Editor를 제공하여 정해진 규칙에 따라 Model Repository 정보를 조회하고, 선택하여 전자정부 표준 프레임워크 개발 환경에서 소스 코드를 자동으로 생성할 수 있게 되었고, 다양한 분야로의 확장이 가능함을 확인할 수 있었다.

앞으로 이러한 결과들을 토대로 다양한 산업 분야의 프로젝트에 메타 기반의 UML 모델링 기술을 적용한다면 국내 IT 산업의 품질 강화 및 경쟁력 확보에 크게 기여할 수 있을 것으로 예상된다.

References

- [1] OMG(Object Management Group), Unified Modeling Language Specification, 2005.
- [2] Colombo, P., Pradella, M., and Rossi, M. A UML 2-compatible language and tool for formal modeling

real-time system architectures. In Proceedings of the 2006 ACM Symposium on Applied Computing, 2006.

- [3] Kagdi, H., Maletic, J. I., and Sutton, A. Context-Free Slicing of UML Class Models. In Proceedings of the 21st IEEE International Conference on Software Maintenance, 2005.

DOI: <http://dx.doi.org/10.1109/ICSM.2005.34>

- [4] Kai Pan, Sunghun Kim, E. James Whitehead, Jr., Bug Classification Using Program Slicing Metrics. Proceedings of the 6th IEEE International Workshop on Source Code Analysis and Manipulation, September 27-29, 2006.

DOI: <http://dx.doi.org/10.1109/SCAM.2006.6>

- [5] Sloane, A. M. and Holdsworth, J. Beyond traditional program Slicing. In Proceedings of the 1996 ACM SIGSOFT International Symposium on Software Testing and Analysis, 1996.

DOI: <http://dx.doi.org/10.1145/229000.226315>

- [6] Korel, B. Rilling, J. Program Slicing in understanding of large programs, Proceedings., 6th International Workshop on Program Comprehension, Jun 1998.

- [7] Louise A. Dennis Enhancing Theorem Prover Interfaces with Program Slice Information, User Interfaces for Theorem Provers (UITP 2006), 2006.

- [8] N. Kim, "Mobile Content Curation Service Based on Real-Time Request/Response Model", The Journal of The Institute of Internet, Broadcasting and Communication, Vol. 14 No. 4, pp.1-6, Aug. 31, 2014.

DOI: <http://dx.doi.org/10.7236/jiibc.2014.14.4.1>

- [9] S.-U. Lee, M.-B. Choi, "An Establishment of the Process System for Software Requirements Engineering", The Journal of The Institute of Internet, Broadcasting and Communication, Vol. 14, No. 1, pp. 37-49, Feb. 28, 2014.

DOI: <http://dx.doi.org/10.7236/jiibc.2014.14.1.37>

이 승 한(Seung-Han Lee)

[정회원]



- 2006년 2월 : 경희대학교 e-Business학과 학사 수료
- 2011년 2월 : 숭실대학교 정보과학 대학원 석사 수료
- 2015년 3월 ~ 현재 : 숭실대학교 컴퓨터학과 박사 과정
- 2014년 2월 ~ 현재 : 엔키소프트 개발 팀장

<관심분야>

SW Architecture, MDA/MDD, UML, SW 생명주기

박 재 표(Jae-Pyo Park)

[중신회원]



- 2004년 8월 : 숭실대학교 컴퓨터학과 공학박사
- 2008년 9월 ~ 2009년 8월 : 숭실대학교 정보미디어 기술연구소 전임연구원
- 2010년 3월 ~ 현재 : 숭실대학교 정보과학대학원 교수

<관심분야>

컴퓨터통신, 보안, 암호학, 멀티미디어 통신