

# 임베디드 시스템의 재사용 프레임워크를 위한 동적 메타모델 설계

김 철 진<sup>†</sup> · 조 은 숙<sup>††</sup>

## 요 약

임베디드 시스템은 다양한 분야에 적용되고 있으며 빠르게 변하고 있는 시장의 요구사항을 대응하기 위한 체계를 갖추어야 한다. 만약 이러한 빠른 구축 체계를 갖추지 못한 제품은 단종되고 말것이다. Time-To-Market을 위한 구축 체계는 기존의 제품을 재사용하여 새로운 제품을 생산해 낼 수 있는 재사용 체계 또는 프레임워크 이다.

임베디드 시스템은 하드웨어를 기반으로 하고 있기 때문에 쉽게 변경이 가능하지 않지만 소프트웨어의 부분을 이용하여 유연하게 시스템을 변경할 수 있다. 따라서 본 논문에서는 임베디드 소프트웨어를 변경할 수 있는 재사용 프레임워크의 메타모델을 제안한다. 본 논문의 메타모델에서는 재사용 프레임워크의 기능적인 흐름을 설계하기 위한 동적 메타모델을 제안한다.

키워드 : 임베디드 시스템, 프레임워크, 가변부, 메타모델

## A Design of Dynamic Meta-model for Reusable Framework of Embedded System

Chul Jin Kim<sup>†</sup> · Eun Sook Cho<sup>††</sup>

### ABSTRACT

Embedded system is applied in various fields and must equip system to respond market's requirement which is changing fast. Products that do not equip such fast construction system will have become sterilization. Construction system for Time-To-Market is reusability system or framework that have reused existent product and produce new product.

Because embedded system is being based on hardware, amendment is not available easily, but can change system softly using segment of software. Therefore, in this paper, suggest meta-model of reusability framework that can change embedded software. Meta model of this paper suggests dynamic meta-model to design functional flow of reusability framework.

Keywords : Embedded System, Framework, Variation Part, Meta-Model

### 1. 서 론

임베디드 시스템은 첨단 산업분야에서부터 일상생활에 이르기 까지 다양한 분야에 적용되고 있으며 다품종 대량 생산을 요구하고 있다. 예를 들어 2005년 1년에만 핀란드 노키아는 56기종의 휴대전화기, 마쓰시타 전기산업은 62기종의 PDP TV, 캐논은 17기종의 디지털 카메라를 세계 시장에 각각 출시했다[1]. 이와 같이 빠른 시간 내에 다품종의 제품을 생산해 낼 수 있었던 것은 새로운 품종을 생산하기 위해 처음부터 새로 개발하는 것이 아니라 기존의 제품을

수정하여 생산할 수 있는 재사용 체계에 의해 가능했다. 이러한 재사용 체계는 재사용 컴포넌트, 품질 검증 체계, 프로세스 효율화, 등과 포괄적인 체계들이 잘 조화를 이루어 가능했을 것이다. 그러나, 무엇보다도 중요한 요소로는 재사용 가능한 설계 자산과 이러한 설계 자산을 기반으로 재사용 구축 체계임을 강조한다.

본 논문에서는 이와 같이 임베디드 시스템의 생산성을 향상시켜 주기 위한 재사용 프레임워크의 메타모델을 제안한다. 메타모델 중에 재사용 프레임워크의 흐름을 설계할 수 있는 동적 메타모델을 제안한다. 본 동적 메타모델은 임베디드 시스템 재사용 프레임워크의 설계 및 구현에 기반을 제공할 수 있을 것이다.

본 논문은 다음과 같이 구성된다. 2장에서는 임베디드 시스템의 구성 요소, 임베디드 시스템의 재사용 프레임워크,

\* 이 논문은 2007년 한국학술진흥재단의 기초연구지원사업 지원을 받아 수행된 연구임(KRF-2007-313-D00694).

† 정 회 원 : 삼성전자 책임연구원

†† 정 회 원 : 서울대학교 소프트웨어과 조교수(교신저자)

논문접수 : 2008년 9월 19일

심사완료 : 2008년 10월 27일

그리고 메타모델에 대해 알아본다. 3장에서는 재사용 프레임워크의 동적 메타모델을 제안한다. 4장에서는 동적 메타모델을 기반으로 재사용 프레임워크가 적용된 동적 설계 사례를 수행한다.

## 2. 관련 연구

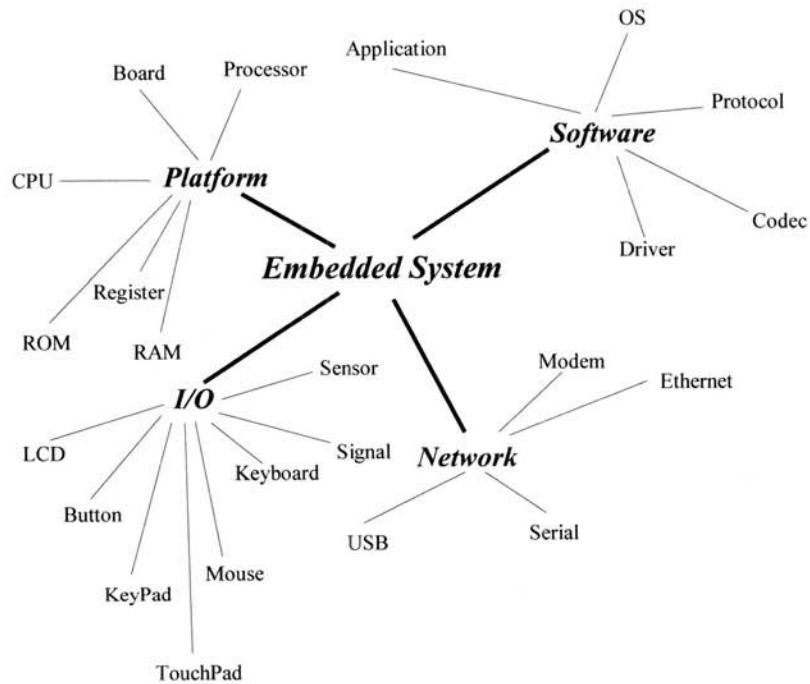
### 2.1 임베디드 시스템

임베디드 시스템의 재사용 프레임워크에 대한 메타모델을 설계하기 위해 임베디드 시스템을 구성하는 요소들을 분석한다. 임베디드 시스템은 하드웨어와 소프트웨어 요소가 함께 결합된 형태이기 때문에 크게 하드웨어 부분과 소프트웨어 부분으로 구별할 수 있으며, 더 상세하게 구분하면 (그림 1) 과 같이 소프트웨어, 하드웨어 플랫폼, 네트워크, 입출력 장

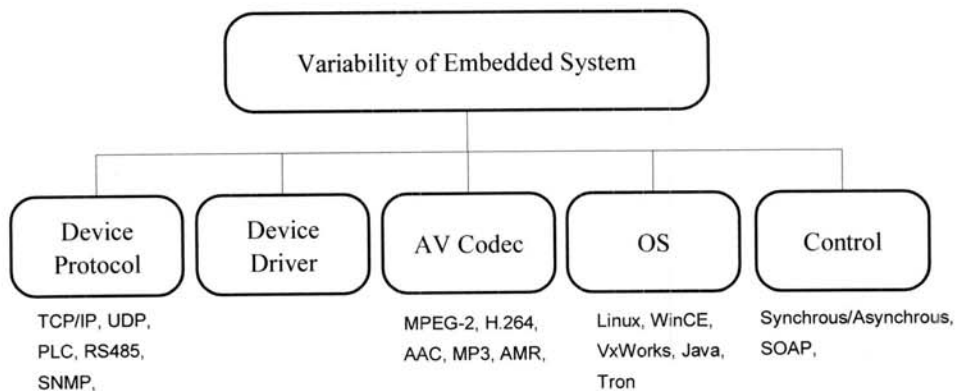
치로 크게 구분할 수 있다[2]. 이러한 임베디드 시스템 구성 요소들 중에 가변적으로 변경 가능하도록 구성할 수 있는 요소는 소프트웨어[3]이며 임베디드 소프트웨어를 구성하는 OS, 디바이스 드라이버, 코덱, 프로토콜 등이 임베디드 시스템의 재사용 프레임워크 대상이 되는 가변부가 될 수 있다.

### 2.2 임베디드 시스템의 재사용 프레임워크

다양한 도메인 과제 개발 시 특정 요구사항의 변경이 잦은 영역을 가변부(Variation Point)라고 정의한다[4][5]. 재사용을 향상시켜 주기 위해서는 이러한 가변부를 특정 요구사항에 대해 변경이 가능하도록 제공해야 한다. 다양한 가변부를 포함하여 임베디드 시스템의 재사용성을 향상시켜주기 위한 재사용 프레임워크는 (그림 2)와 같이 디바이스 프로토콜, 디바이스 드라이버, AV 코덱, OS, 제어(시그널)를 가



(그림 1) 임베디드 시스템 구성 요소



(그림 2) 임베디드 시스템의 가변부

변적으로 변경할 수 있도록 제공해야 한다.

임베디드 시스템의 재사용 프레임워크 구성은 가변부들을 중계하기 위한 어댑터 클래스들과 가변부 처리를 지원하기 위한 핵심 클래스들로 구성된다. 재사용 프레임워크의 어댑터는 홈 네트워크 시스템의 가변성인 디바이스 프로토콜, 디바이스 드라이버, AV 코덱, OS, 제어 가변성에 대한 대행 역할을 수행한다. 가변부 처리를 위한 핵심 클래스는 가변성 관리기(Variability Manager), 리플렉터(Reflector), 설정 관리기(Configuration Manager), 동적 전개기(Hot Deployer), 그리고 자원 할당기(Resource Allocator)로 구성된다.

재사용 프레임워크의 가변부를 설계하기 위한 기법으로 선택 기법과 플러그인 기법을 이용할 수 있다[6]. 선택 기법은 가변부에 대해 선택 가능한 기능을 설계하여 임베디드 시스템 외부에서 선택하는 기법이다. 선택기법은 매개변수화(Parameterization)에 의해 가능하다[7]. 플러그인 기법은 임베디드 시스템 내부로 변경 기능(디바이스)을 입력하여 임베디드 시스템 내부의 기능을 변경하는 기법이다.

2.3 메타 모델

메타모델이란 어떠한 개념들을 구성하는 주요 요소들과 그들 간의 관계를 보여주는 개념 맵이다. UML에서는 MOF(Meta Object Facility)를 통해 메타모델을 정의하고 있는데, 객체지향 모델 작성에 사용되는 UML 모델의 필수 요소와 문법, 구조를 정의하는 메타 모델로 제시하고 있다. 또한 MDA에서는 MOF를 CWM(Common Warehouse Metamodel)이나 UML(Unified Modeling Language)의 메타모델에 대한 공통 모델로 제공하고 있다[8].

메타모델은 특정 개발 플랫폼에 종속되지 않은 형태로 제공되기 때문에 메타 모델에 기반을 두고 모델을 개발할 경우

플랫폼에 따른 다양한 형태의 모델들을 확장하여 개발할 수 있을 뿐만 아니라 모델의 재사용성이 매우 향상되게 된다.

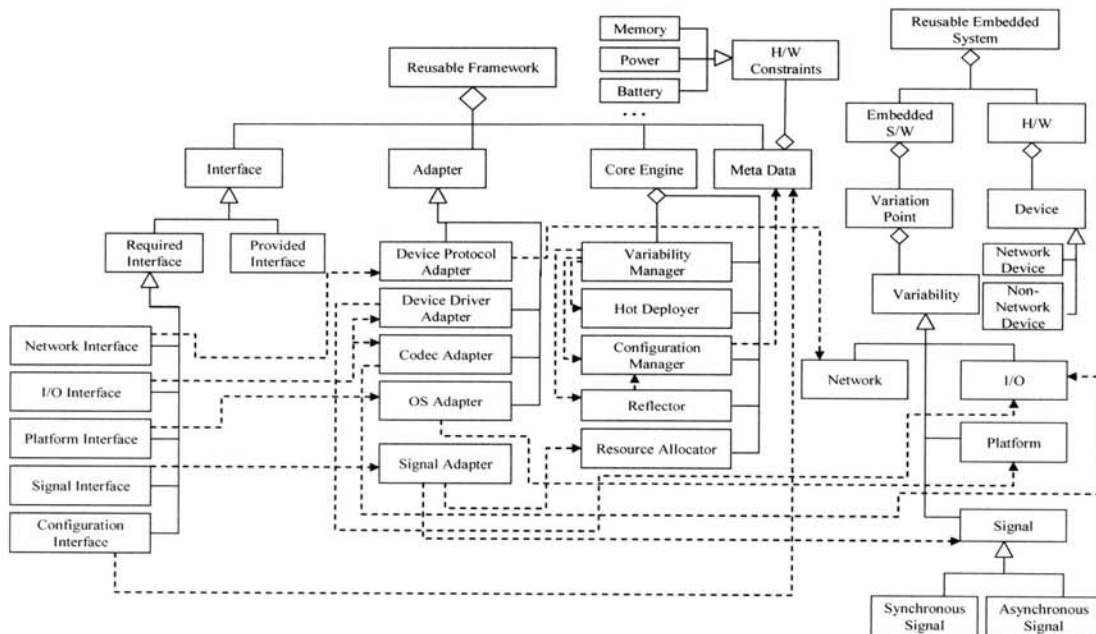
3. 재사용 프레임워크의 동적 메타모델 설계

본 장에서는 임베디드 재사용 프레임워크의 동적 설계를 지원하기 위한 메타모델을 설계한다. 동적 메타모델 설계는 정적 메타모델을 기반으로 하며, 재사용 프레임워크의 재사용성을 향상시켜 주기 위한 선택 기법과 플러그인 기법에 대한 동적 메타모델을 설계한다. 선택 기법과 플러그인 기법은 설정 측면과 서비스 측면으로 구분하여 설계되며 설정 시점에 선택되거나 입력된 가변 기능(디바이스)은 영속적으로 저장되어 서비스 시점에 해당 가변 행위를 호출하여 서비스를 제공한다.

정적 메타모델은 클래스 다이어그램(Class Diagram)을 이용하여 설계하며 동적 메타모델은 상태 다이어그램(State Machine Diagram)을 이용하여 설계한다.

3.1 동적 메타모델 설계를 위한 정적 메타모델

임베디드 시스템을 위한 재사용 프레임워크의 구조에 대한 정적 메타모델은 (그림 3)과 같다. 임베디드 시스템의 가변부들을 중계하기 어댑터와 가변부 처리를 지원하기 위한 핵심 클래스들로 구성된다. 재사용 프레임워크의 어댑터는 임베디드 시스템의 가변성인 디바이스 프로토콜, 디바이스 드라이버, AV 코덱, OS, 제어 가변성에 대한 대행 역할을 수행한다. 가변부 처리를 위한 핵심 클래스는 가변성 관리기(Variability Manager), 리플렉터(Reflector), 설정 관리기(Configuration Manager), 동적 전개기(Hot Deployer), 그리고 자원 할당기(Resource Allocator)로 구성된다.



(그림 3) 정적 메타 모델

임베디드 시스템의 가변성에 대한 어댑터들은 재사용 요구 인터페이스(Required Interface)를 통해 어떤 기능을 사용하지 설정하며, 재사용 프레임워크의 가변성 관리기(Variability Manager)는 설정 및 서비스 호출에 대한 대행 역할을 수행한다. 가변성 관리기에 의한 설정은 리플렉터(Reflector), 설정 관리기(Configuration Manager), 그리고 동적 전개기(Hot Deployer)를 통해 이루어지며, 설정된 서비스는 가변성 어댑터들에 의해 제공된다. 임베디드 시스템의 자원에 대한 관리하는 자원 관리기(Resource Allocator)에 의해 설정되며 가변성 어댑터에 따라 다르게 설정된다. 예를 들면 운영체제가 변 어댑터가 설정하는 운영체제(WindowCE, VxWorks, 등)에 따라 자원 관리기는 서로 다른 자원(메모리, 등)을 할당한다.

### 3.2 선택 기법의 동적 메타모델 설계

선택 기법은 임베디드 시스템 내부에 설계(또는 구현)되

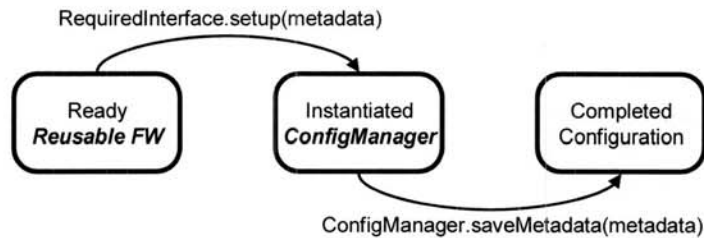
어 있는 가변부 중에 선택하여 변경하는 기법이다. 선택 기법은 설정 측면과 서비스 측면에서 재사용 프레임워크의 동적 메타모델을 설계할 수 있다.

선택 기법에 의한 가변부 설정은 (그림 4)와 같이 재사용 프레임워크의 설정 관리기가 요구 인터페이스로부터 메타정보를 입력 받는다. 이러한 메타정보는 임베디드 시스템 내에 설계(또는 구현)된 기능(디바이스, 프로토콜) 중에 하나의 정보이다.

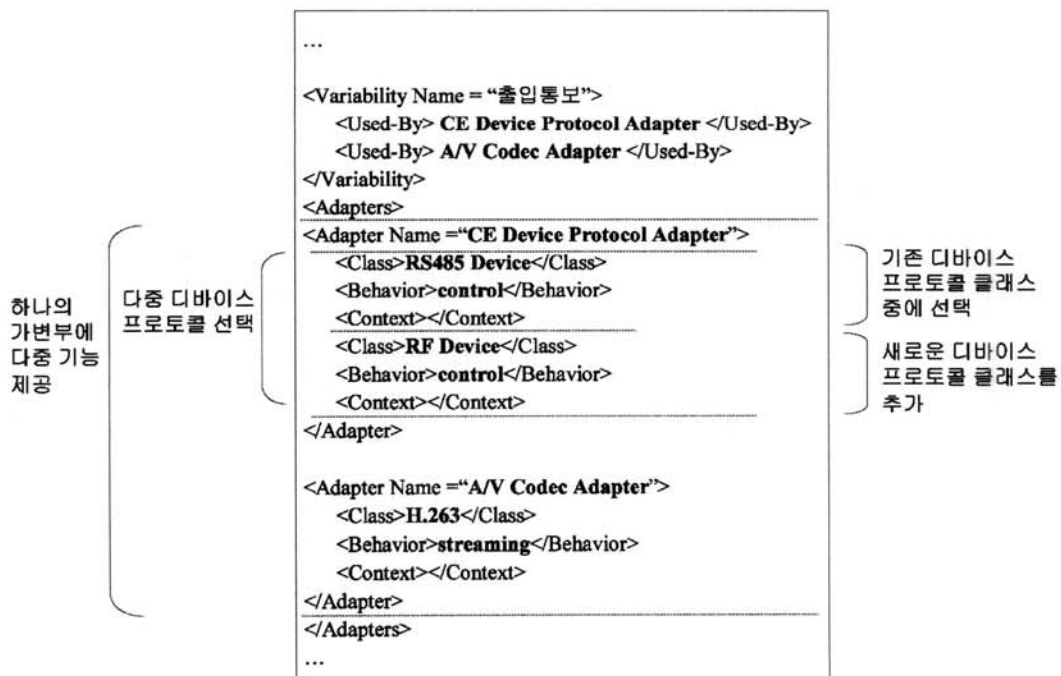
선택 기법의 의해 설정된 메타 정보는 (그림 5)와 같이 XML 형태의 구조를 가지며 가변부의 어댑터부터 클래스, 함수까지의 설정 정보를 포함하고 있다. 이 메타 정보는 서비스 시점에 호출되어 설정된 가변부의 기능을 제공할 수 있다.

(그림 6)은 홈 네트워크 시스템의 프로토콜을 설정하는 동적 메타모델을 나타낸다. 홈 네트워크 시스템은 다양한 디바이스 프로토콜을 통해 다양한 디바이스를 지원하기 때문에 디바이스에 맞는 프로토콜인 RS485나 PSTN(Public Switched

## • Configuration

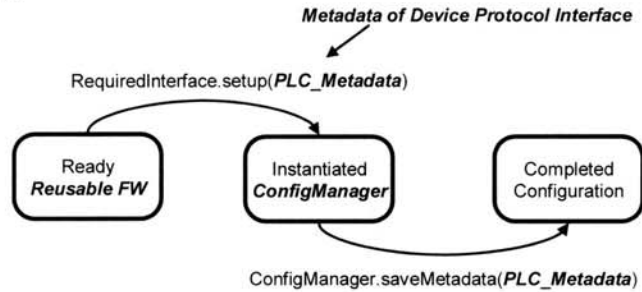


(그림 4) 가변부 설정 메타모델 (선택 기법)



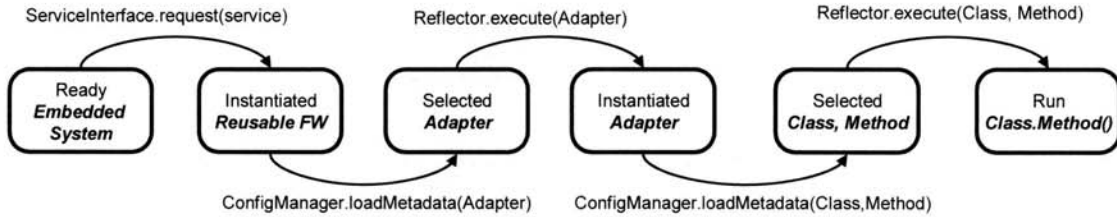
(그림 5) 가변부 메타 정보

• Configuration



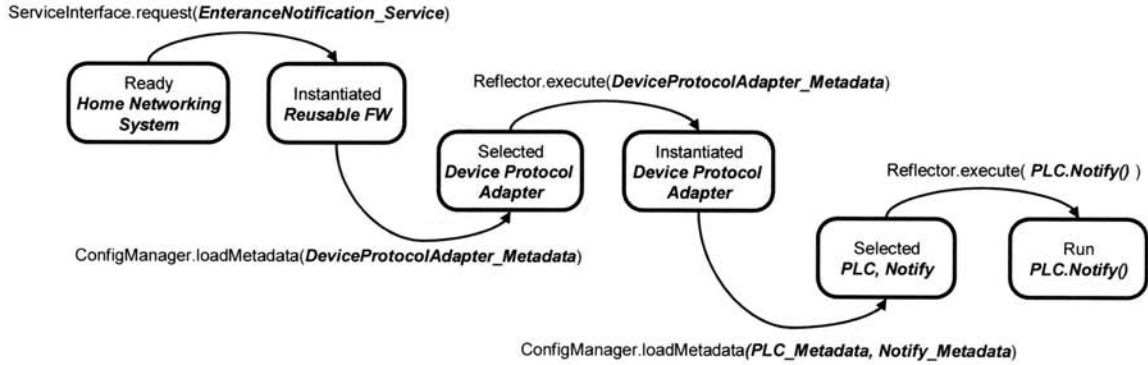
(그림 6) 프로토콜 설정 메타모델 (선택 기법)

• Service



(그림 7) 가변부 서비스 메타모델 (선택 기법)

• Service



(그림 8) 프로토콜 서비스 메타모델 (선택 기법)

Telephone Network), PLC(Power Line Communication) 등을 가변적으로 선택할 수 있도록 제공해야 한다.

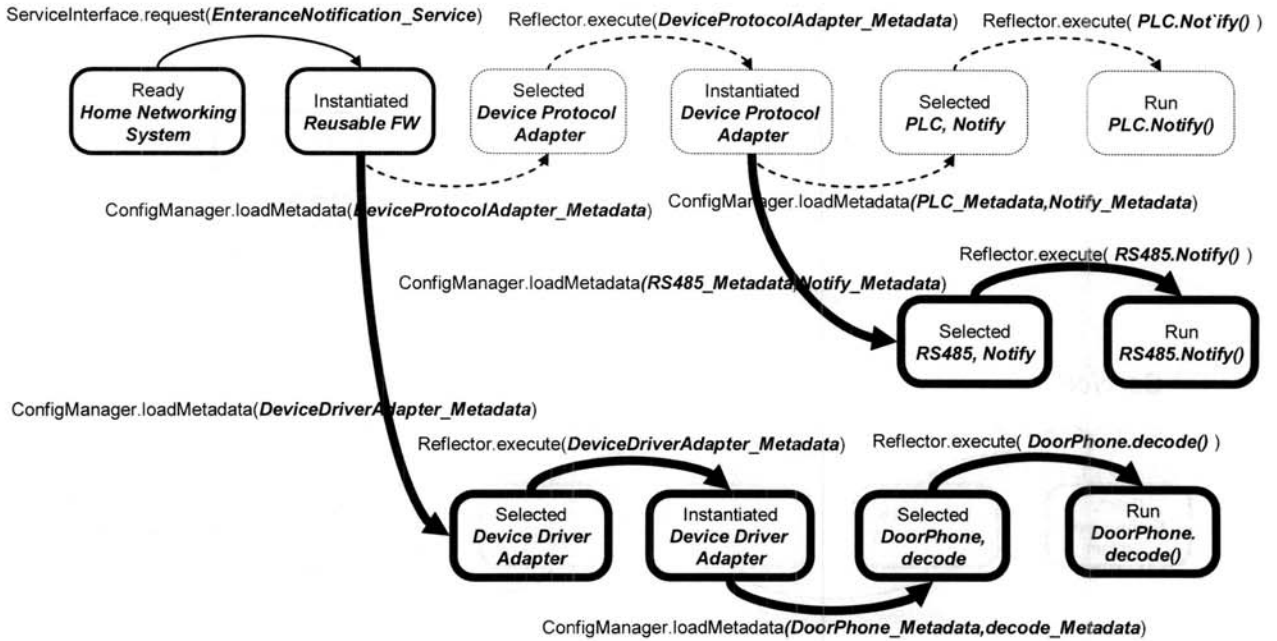
선택 기법에 의해 설정된 메타정보는 (그림 7)과 같이 재사용 프레임워크에 의해 서비스될 수 있다. 임베디드 시스템 외부에서 가변부와 관련되어 있는 서비스를 요청할 경우, 서비스 인터페이스는 재사용 프레임워크에 의해 가변부 중에 설정된 서비스로 제공해 줄 것을 요청한다. 재사용 프레임워크는 메타정보에 설정된 정보를 기반으로, 가변부를 대행(Delegation)하는 어댑터를 선택하고, 선택된 어댑터에 의해 클래스와 함수를 호출하여 서비스를 제공한다.

홈 네트워크 시스템에서 프로토콜 가변부에 대한 서비스는 (그림 8)과 같다. 현관에서 출입방문 서비스를 요청할 경우, 홈 네트워크 시스템은 프로토콜 서비스가 가변부와 관

련되므로 재사용 프레임워크를 호출한다. 재사용 프레임워크는 프로토콜과 관련된 설정 메타정보를 이용하여 프로토콜 어댑터를 선택한다. 프로토콜 어댑터는 다양한 프로토콜 중에 설정된 PLC 서비스를 호출하여 출입 방문 서비스를 제공한다.

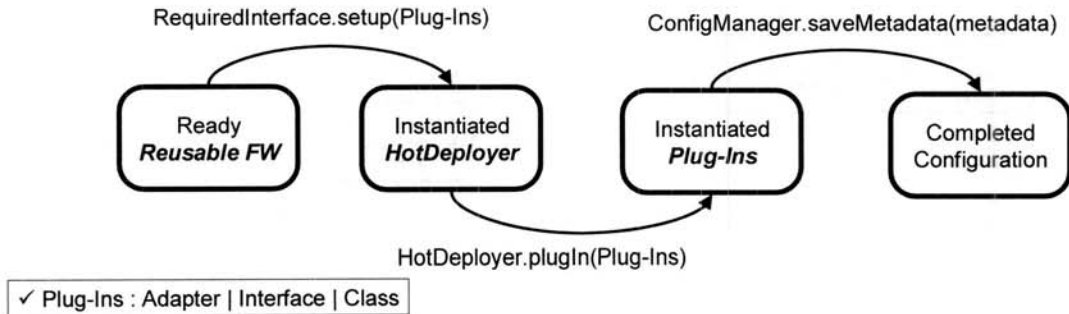
(그림 9)와 같이 재사용 프레임워크의 복합 서비스에 대한 동적 메타모델은 하나의 서비스에 대해 다양한 기능이 변경될 수 있음을 나타낸다. 출입방문 서비스에 대한 프로토콜 변경뿐만 아니라 디바이스 변경을 위해 드라이버를 변경할 수 있다. 이러한 변경은 전혀 다른 구조로의 변경이 가능함을 나타낸다. 기존 연구에서 동일 구조(또는 인터페이스) 내에서 변경이 가능하던 것과 다르게 전혀 다른 구조에서 변경을 제공하므로 더욱 다양한 요구사항을 수용할 수 있다.

• Service



(그림 9) 복합 서비스 메타모델 (선택 기법)

• Configuration



(그림 10) 가변부 설정 메타모델 (플러그인 기법)

3.3 플러그인 기법의 동적 메타모델 설계

플러그인 기법은 임베디드 시스템 내부에서 변경 요구사항을 제공할 수 없을 경우, 임베디드 시스템 외부에서 변경 기능(디바이스, 프로토콜)을 시스템 내부로 추가하는 기법이다. 플러그인 기법은 설정 측면과 서비스 측면에서 설계한다.

플러그인 기법에 의한 가변부 설정은 (그림 10)과 같이 변경하려고 하는 기능을 입력하여 설정한다. 입력되는 단위는 어댑터, 인터페이스, 클래스와 같이 메타정보가 아닌 실제 기능을 포함한 플러그인스(PlugIns)이다. 재사용 프레임워크는 플러그인스를 입력 동적 전개기로 전달하여 서비스를 제공할 수 있도록 활성화(Instantiation, Activation) 시킨다.

홈 네트워크 시스템은 한 기능에 대해 다양한 디바이스가 연결될 수 있으므로 (그림 11)과 같이 동적 전개기로 현관

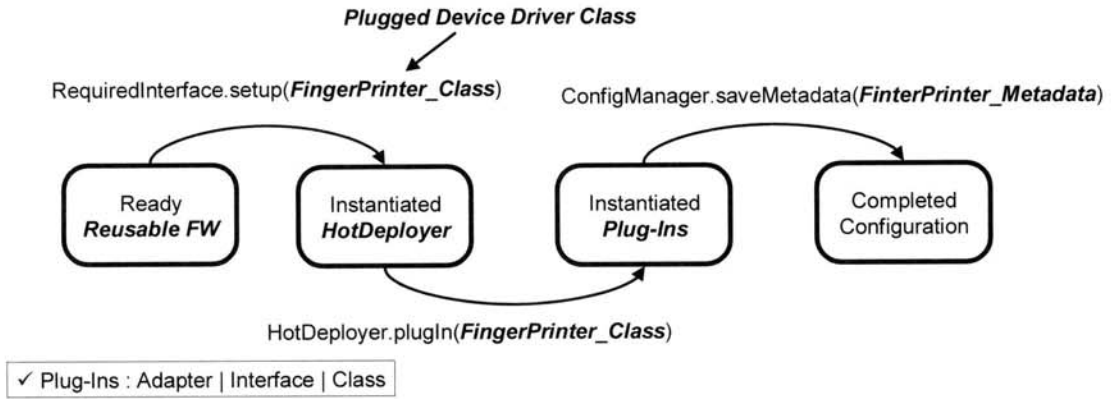
지문인식기의 드라이브를 추가하여 기존의 열쇠나 도어락에 의한 현관 출입 기능을 변경할 수 있다.

플러그인 기법에 의한 서비스의 동적 메타모델은 (그림 12)와 같이 출입 서비스가 도어폰에서 지문인식기로 변경될 경우, 디바이스 드라이브 어댑터가 설정 메타정보를 기반으로 지문인식기로 출입서비스를 호출한다. 새로운 기능(디바이스, 프로토콜)이 플러그인 된 이후의 서비스에 대한 동적 메타모델은 선택 기법과 동일하다.

4. 사례 연구

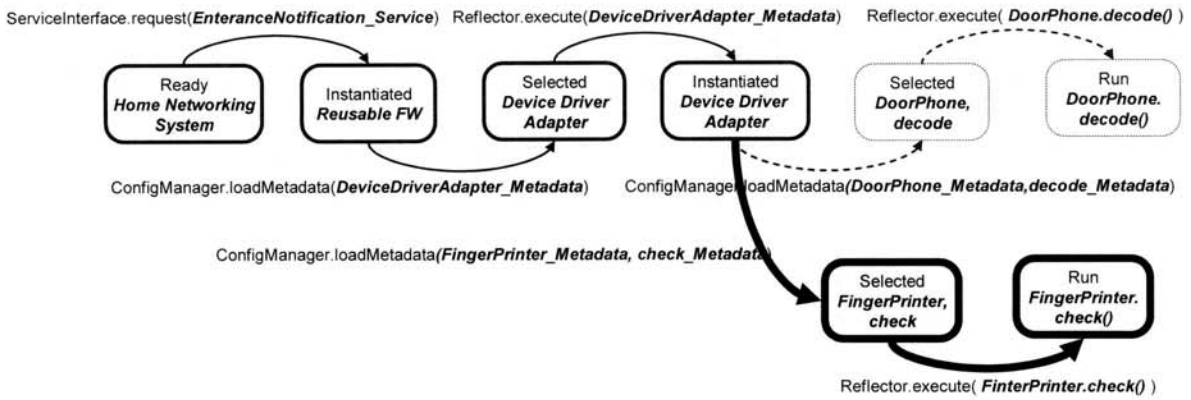
본 장에서는 선택 기법과 플러그인 기법에 대한 동적 메타모델을 기반으로 동적 설계를 수행하여 메타모델의 적합

• Configuration

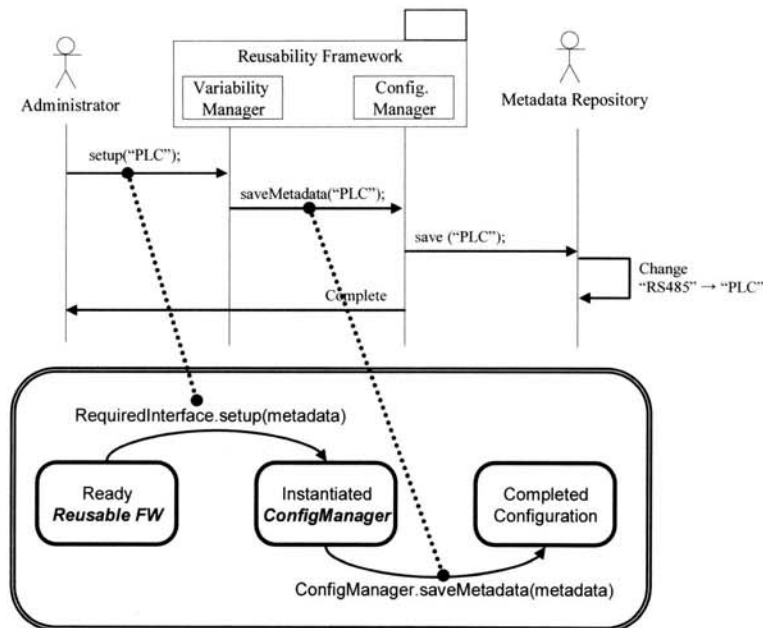


(그림 11) 드라이브 설정 메타모델 (플러그인 기법)

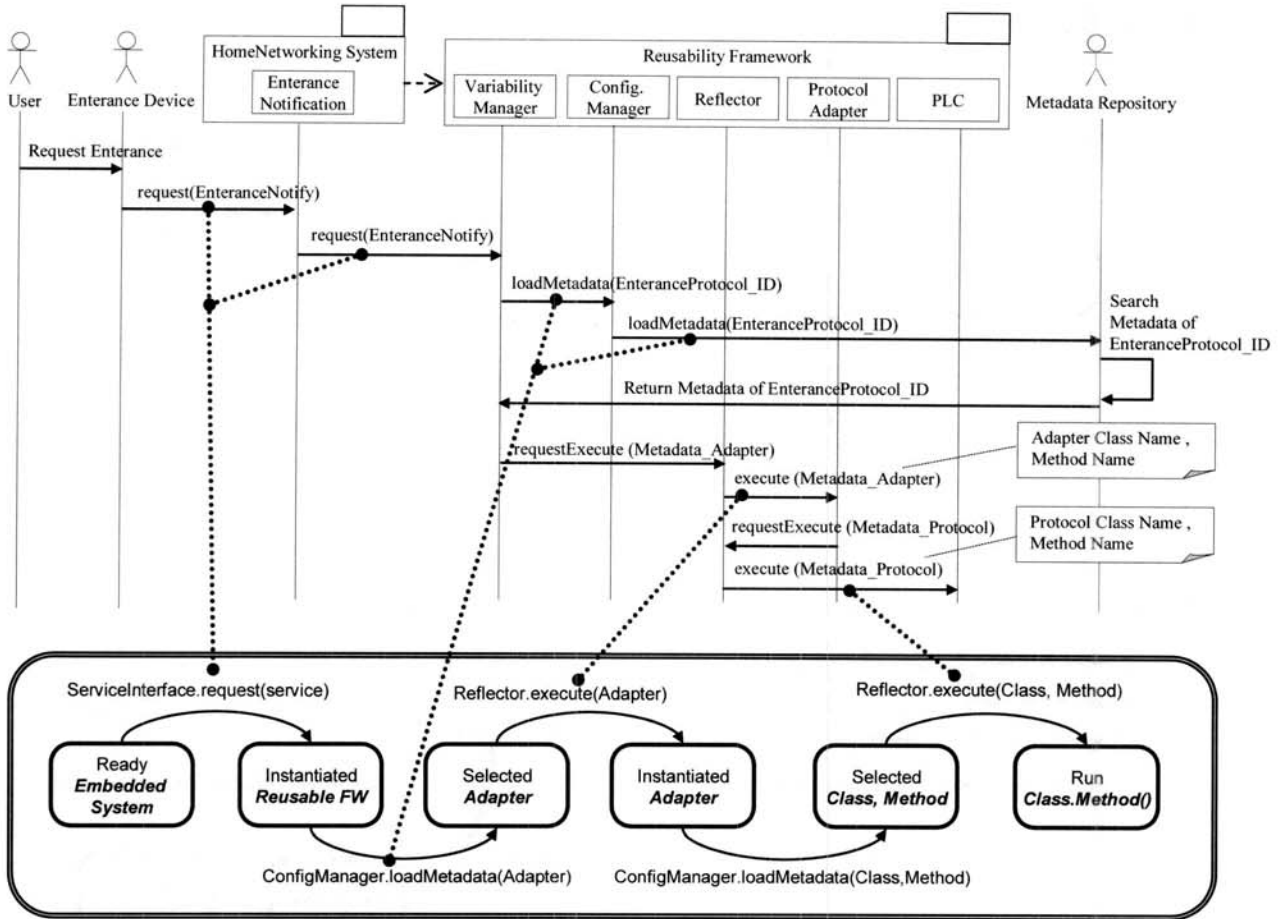
• Service



(그림 12) 서비스 메타모델 (플러그인 기법)



(그림 13) 가변부 설정 동적 설계 (선택 기법)



(그림 14) 가변부 서비스 동적 설계 (선택 기법)

성을 검증한다. 홈 네트워크 시스템의 현관 출입통제 기능에 대해 순차도(Sequence Diagram)를 이용하여 설계한다.

#### 4.1 선택 기법의 동적 설계 사례

선택기법에 의해 가변부를 설정하기 위한 설계는 (그림 13)과 같다. 출입통제를 위해 사용하는 프로토콜을 'RS485'에서 'PLC'로 변경하기 위해 재사용 프레임워크에 설정정보('PLC')를 입력하면 가변부 관리기와 설정 관리기에 의해 PLC 프로토콜로 변경된다. 이러한 동적설계는 (그림 13)과 같이 동적 메타모델의 기반으로 정의될 수 있다.

선택기법에 의한 가변부 서비스의 동적설계는 (그림 14)와 같다. 재사용 프레임워크는 설정정보를 기반으로 변경된 서비스를 제공할 수 있다. 설정 관리기에 의해 관리되는 설정정보를 기반으로 리플렉터는 변경된 기능을 호출하며, 위의 사례와 같이 리플렉터는 변경하기 위한 프로토콜 어댑터 및 PLC 클래스를 호출한다. 본 임베디드 시스템에 대한 동적 메타모델은 홈 네트워크 가변부 서비스를 설계하기 위한 중요한 동적정보를 제공한다.

#### 4.2 플러그인 기법의 동적 설계 사례

플러그인 기법에 의해 가변부를 설정하기 위한 설계는 (그

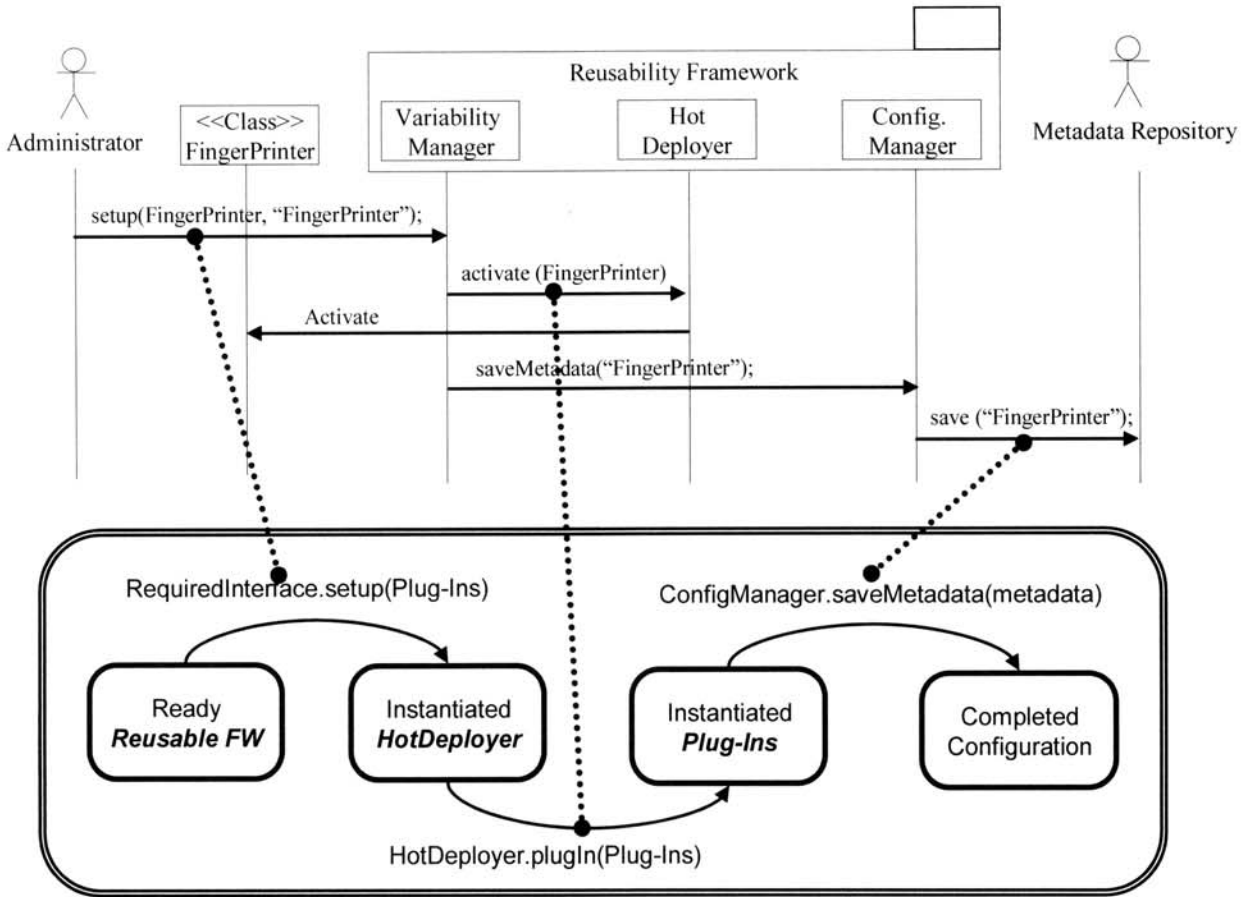
림 15)과 같다. 선택기법과 다르게 플러그인 기법은 외부에서 새로운 서비스를 정의하여 내부로 플러그인 하는 기법이다. 따라서 (그림 15)에서와 같이 서비스 클래스와 설정정보를 입력하면 서비스 클래스가 동적 전개기에 의해 활성화된다. 아래의 사례는 출입통제를 도어폰에서 지문인식기로 변경하고자 할 경우 지문인식을 위한 서비스를 플러그인하면 재사용 프레임워크에서 활성화시켜 주는 사례이다. 본 사례에서도 플러그인 기법의 동적 메타모델이 동적설계를 위한 중요한 메타정보를 제공함을 파악할 수 있다.

플러그인 기법에 의한 서비스 설계는 선택기법의 서비스와 동일하며, 단지 설정 방법에 차이가 있다. 또한 시스템 내부 클래스가 아닌 외부 클래스에 대한 서비스를 요청하는 것이 차이가 있다.

### 5. 결 론

지금까지 임베디드 시스템의 재사용 프레임워크를 설계하기 위한 동적 메타모델에 대해 알아보았다. 동적 메타모델은 재사용 프레임워크를 기반으로 동적설계를 위한 핵심 흐름을 제공하며, 동적 메타모델을 기반으로 재사용성 있는 임베디드 시스템의 설계나 구현에 적용할 수 있을 것이다. 향후에





(그림 15) 가변부 설정 동적 설계 (플러그인 기법)

는 본 동적메타 모델을 기반으로 임베디드 시스템의 재사용 프레임워크에 적용하여 재사용성을 향상됨을 연구한다.

Based Software Engineering(GCSE'00), Erfurt, October, 2000.

[8] UML Specification v2.0, OMG, Inc., January, 2006.

### 참고 문헌

- [1] Nikkei Electronics, "가격 하락을 밀친 삼아-세계를 얻을 다품종 대량생산", pp.77-110, April, 2006.
- [2] Axel J., Modeling Embedded System and SOCs, Mogan Kaufmann, 2004.
- [3] David E. S., An Embedded Software Primer, Addison Wesley, 1999.
- [4] Coplien J., Hoffman D., and Weiss D., "Commonality and Variability in Software Engineering," IEEE Software, pp. 37-45, November, 1998.
- [5] Anastasopoulos M. and Gacek C., Implementing Product Line Variabilities, Technical Report IESE Report No. 089.00/E, Version 1.0, Fraunhofer Institute for Experimental Software Engineering (IESE), November, 2000.
- [6] 김철진, 김수동, "컴포넌트 행위 커스터마이제이션 기법, 한국정보과학회 논문지(B), 제30권 제3.4호 2003년 4월."
- [7] Becker M., "Generic Components: A Symbiosis of Paradigms," 2nd International Symposium on Generative and Component-



김철진

e-mail : cjkim777@gmail.com

1996년 경기대학교 전자계산학과(학사)

1998년 숭실대학교 컴퓨터공학부

(공학석사)

2004년 숭실대학교 컴퓨터공학부

(공학박사)

2004년 가톨릭대학교 컴퓨터정보공학부 강의전담교수

2004년~현 재 삼성전자 책임연구원

관심분야: CBD, Component Customization, Embedded Software



조 은 속

e-mail : escho@seoil.ac.kr

1993년 동의대학교 전산통계학과(이학사)

1996년 숭실대학교 대학원 컴퓨터학과  
(공학석사)

2000년 숭실대학교 대학원 컴퓨터학과  
(공학박사)

2000년~2005년 동덕여자대학교 정보학부 강의전임교수

2005년~현 재 서일대학 소프트웨어과 조교수

관심분야 : CBSE, Embedded Software, Service-Oriented  
Computing, SOA