

소프트웨어 하드웨어 협동설계를 위한 통합모듈을 지원하는 제품자료모델

도 남 철*

A Product Data Model for the Integration Module for Supporting Collaborations on Hardware and Software Development

Namchul Do*

■ Abstract ■

Since software and hardware integration has become a strategic tool for companies to innovate their products, an information system that can comprehensively manage software and hardware integrated product development is critical for the current product development. This paper proposed a product data model that can support modules of related software and hardware parts in Product Data Management(PDM) integrated with Software Configuration Management(SCM). The model allows engineers to define software and hardware product structure independently, and support the integration module that can summon related software and hardware parts to build a comprehensive module for collaboration. Through the integration module, engineers can identify and examine the effectiveness of their design alternatives to other related parts form different disciplines. The product data model was implemented as a prototype PDM system and tested with an example robotics product.

Keyword : Product Data Management, Software Configuration Management, Software and Hardware Development

1. 서 론

신제품 개발에서 소프트웨어(Software)와 하드웨어(Hardware) 융합은 갈수록 가속화되고 있다. 최근 기업들은 특정 하드웨어에 종속된 소프트웨어 채용을 넘어, 범용 운영 체제(Operating System)를 하드웨어에 적용함으로써 보다 유연하고 혁신적인 제품을 개발하고 있다. 범용 운영 체제와 휴대 전화를 융합한 스마트 폰(Smart Phone)의 예에서 알 수 있듯이 소프트웨어와 하드웨어의 광범위한 통합은 시장을 근본적으로 바꿀 수 있는 파괴력을 지니고 있다. 그러므로 기업의 소프트웨어와 하드웨어 통합 개발에 대한 관심과 요구는 더욱 확대 되고 있다.

소프트웨어와 하드웨어의 통합 개발 지원은 소프트웨어 개발을 지원하는 Software Configuration Management(SCM)와 하드웨어 개발을 지원하는 Product Data Management(PDM)의 통합 연구로 진행되어 왔다. 하지만 기존의 SCM과 PDM 통합 연구는 소프트웨어 엔지니어링(Software Engineering)을 전문으로 하는 SCM 연구진에 의하여 진행되어왔다. 특히 SCM의 제품구성(Product Configuration)과 메타 자료(Meta Data) 관리 문제를 해결하기 위하여 PDM의 유사 개념을 받아들이고, 이를 소프트웨어 하드웨어 통합 제품개발에 적용하는 방식을 취해왔다. 이 접근 방식은 SCM과 PDM에서 공통으로 관리되는 제품구성이나 버전(Version)에 대한 고려를 하였지만, 하드웨어 개발 과정에서 중요시 되는 설계변경(Engineering Changes)을 비롯한 협동 설계를 충분히 고려하지 못하였다. 설계, 공정 계획, 그리고 고객 지원 정보와의 일관성 유지 및 협동 설계는 제품개발의 기본 요구이며, 이를 고려 하지 못한 SCM과 PDM의 통합은 일관된 제품개발 프로세스와 제품 자료 공유를 불가능하게 한다.

그러므로 본 연구는 SCM 관점의 통합 연구에서 고려하지 못하였던 소프트웨어와 하드웨어 협동 설계를 위한 인테그레이션 모듈(The Integration

Module)을 지원하는 제품자료 모델(Product Data Model)을 제안한다. 이 자료 모델은 제품구성과 제품구조(Product Structure) 측면의 통합에 관한 기존 연구[1, 2]를 확장하여, 협동 설계를 위한 소프트웨어 하드웨어 통합 모듈을 제공한다. 제안된 자료 모델의 유효성을 확인하기 위하여 모델을 구현한 시제품 PDM을 개발하고, 이 시스템에 소프트웨어 하드웨어 부품을 가진 로봇 설계 정보를 표현하였다.

본 논문의 제 2장에서는 관련 연구를 살펴본다. 제 3장에서는 소프트웨어 하드웨어 통합 설계를 위한 기존 모델[1]을 살펴본다. 제 4장에서는 기존 모델을 확장한 인테그레이션 모듈을 지원하는 제품자료모델을 설명한다. 제 5장에서는 제안된 모델을 구현한 PDM 시제품과 시제품을 이용한 예제 설계 표현에 대하여 설명한다. 제 6장에서는 결론을 도출한다.

2. 관련 연구

[3-6]은 소프트웨어 개발 방법론인 SCM과 일반 제품개발 방법론인 PDM(혹은 Engineering Data Management)을 비교하였다. 이 비교 연구들은 모두 소프트웨어 엔지니어링 관점에서 PDM을 다루고 있으므로, PDM에서 중요하게 관리되는 설계변경이나 제품구조 중심의 협동 작업에 관한 비교를 다루지 못하고 있다.

[7]은 웹 상의 Graphic Simulation 표준 기술을 이용하여 소프트웨어와 하드웨어를 포함한 다분야 통합이 가능하다는 주장을 하였다. 소프트웨어 하드웨어 통합 설계 지원은 단순한 사용자 환경의 문제가 아닌 제품 자료 표현과 운용에 관련된 기반 문제이다. 그러므로 제안된 가상현실 시뮬레이션 기술 기반 PDM SCM 인터페이스만으로는 통합 문제가 지원되기 어렵다.

[8, 9]는 소프트웨어와 하드웨어가 통합된 제품구성 모델(Product Configuration Model)을 제안하였다. SCM PDM 통합에서 제품구성 모델과 구

현은 중요한 문제이지만, 이는 고려해야 할 요소 중에 하나에 불과하며, 제품구성은 제품구조 및 설계변경과 연계되어야 한다. 그러므로 버전 관리, 메타 자료, 제품구조, 그리고 설계변경과 통합되지 않은 제품구성 모델만으로 소프트웨어 하드웨어 통합 제품 설계를 효과적으로 지원할 수 없다.

[10, 11]은 통신 기기 개발을 위한 PDM과 SCM의 통합을 보고하고 있다. 이 예에서는 상용 SCM인 ClearCase에서 소프트웨어를 생성 관리하고, 상용 PDM인 Metaphase를 이용하여 파일 정보와 관련 메타 정보를 함께 관리한다. 즉 PDM에서는 통합된 제품구조를 통하여 소프트웨어와 하드웨어 부품을 관리하며, 소프트웨어 소스 파일은 SCM 시스템과 인터페이스된 문서로써 표현하였다.

[12]는 소프트웨어 개발에 모델 기반 개발 방법론을 적용함으로써 PDM과 SCM을 통합하는 접근 방법을 소개하였다. 모델 기반 방법이란 PDM의 메타 자료 관리로 부터 영향을 받은 방법으로 소스 코드 기반의 개발을 지양하고 소프트웨어 모델을 설계하고 이를 개발 단계에서도 지속적으로 이용하는 접근 방법을 뜻한다. 이는 소프트웨어 엔지니어링 측면에서 PDM SCM 통합을 고려하는 대표적인 접근 방법이다.

[13]은 소프트웨어 하드웨어뿐만 아니라 전기, 전자 부분의 설계를 통합 지원 할 수 있는 정보 시스템 아키텍처를 제안하였다. 이 연구에서 PDM은 모델링에 강점이 있고 SCM은 개발(Building)에 강점이 있는 것으로 분석하고, 이를 이용하여 소프트웨어에 PDM의 메타 모델 개념을 추가한 아키텍처를 제안하였다. 저자가 언급하였듯이 이 연구는 소프트웨어 엔지니어(Software Engineer)를 위한 소프트웨어 엔지니어의 연구로써 PDM에서 요구하는 설계변경과 타 부분의 협동 설계에 대한 고려를 하지 못하였다.

현재 상용 SCM과 인터페이스 기능을 제공하는 상용 PDM 시스템이 존재한다. SIEMENS사의 TeamCenter PDM[14]은 IBM의 ClearCase와 인터페이스 객체를 제공한다. PTC사의 WinChill PDM[15]

도 상용 SCM 시스템과의 인터페이스를 제공한다. 그러나 상용 PDM의 SCM 통합의 문제점은 PDM과 SCM을 위한 독립된 데이터베이스를 기반으로 한 인터페이스라는 점이다. 복수의 데이터베이스는 자료의 정합성을 유지가 어려우며, 이는 공동 설계 작업 시 오류와 작업의 중복 등의 문제점을 야기한다. 또한 설계자들이 이질적인 작업 환경을 사용함으로써 협동 작업에 많은 애로를 겪게 된다.

그 동안 SCM과 PDM 통합 연구는 소프트웨어 엔지니어링 관점에서 소프트웨어 개발을 합리화하고 이를 통해 PDM과 통합하려는 노력이 주를 이루었다. 이러한 연구들은 SCM과 PDM에서 공통으로 필요한 제품구성 관리 등은 고려하였으나 PDM이 요구하는 설계변경을 포함한 협동 설계에 대한 고려를 하지 못하였다. 특히 소프트웨어와 하드웨어 제품은 각기 독립적인 제품구조를 구성해야 하는 이질적인 분야이다. 그러므로 협동 설계 과정에서 상호 설계변경의 영향을 쉽게 평가할 수 있는 일관된 소프트웨어 하드웨어 통합 모듈 제공이 필요하다.

3. 소프트웨어 하드웨어 통합 표현을 위한 기존 연구

본 연구는 기존 연구[1, 2]에서 제안한 제품자료 모델을 확장하여 소프트웨어 하드웨어 협동 설계를 지원하는 제품자료모델을 개발하였다. 기존 연구는 PDM 중심으로 SCM 기능을 흡수 통합함으로써, 소프트웨어 하드웨어 개발자에게 동일한 사용자 환경과 단일 데이터베이스에 기반을 둔 PDM 시스템 제공을 목적으로 하였다. 이를 위하여 확장된 PDM에 소프트웨어 개발자를 위한 SCM의 기본 기능을 추가하고, 소스 코드에 대한 다양한 메타 정보를 제공하였다. 또한 제품구조 표현 시 하드웨어와 소프트웨어 부품을 함께 사용할 수 있게 함으로써, 소프트웨어와 하드웨어 통합 제품구성과 설계변경 관리 기능을 제공하였다.

기존 연구에서 소프트웨어 하드웨어 부품이 합

게 표현된 제품구성과 설계변경이 제공됨에도 불구하고, 소프트웨어 하드웨어가 상호 독립적인 제품구조를 이루고 있어 관련된 부품 간의 상관관계를 표현할 필요성이 확인되었다. 특히 한 분야의 설계변경이 다른 분야에 미칠 영향을 고려해야 하는 설계변경 과정에서 다른 분야의 이질적인 제품구조와 정보를 확인하는 작업은 협동 설계에 많은 어려움을 야기하였다.

그러므로 본 연구에서는 소프트웨어 하드웨어 부품이 각 분야에 독립적인 제품구조를 유지하면서, 상호 부품 간의 연관 관계를 표현하는 인테그레이션 모듈을 추가하여, 소프트웨어 하드웨어 부분의 협동 설계를 지원할 수 있는 제품자료모델을 제안하였다. 이 모델은 기존 연구의 제품자료모델을 확장하여 개발하였으므로, 기존 연구에서 확인한 소프트웨어 하드웨어 개발 지원을 위한 이득을 모두 유지하고 있으며, 추가된 인테그레이션 모듈의 제품구조를 통하여 설계변경 과정을 포함한 소프트웨어 하드웨어 협동 설계에 필요한 일관되고 통합된 제품 정보 지원이 가능하게 되었다.

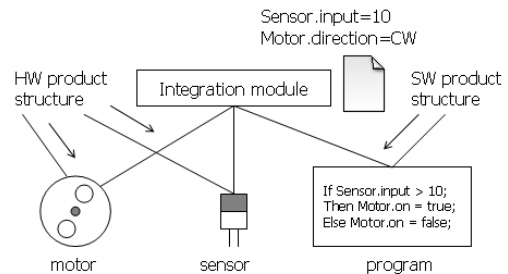
4. 제안 제품자료모델

4.1 인테그레이션 모듈 소개

인테그레이션 모듈(The Integration Module)이란 제품구조를 구성하는 부품의 일종으로 단위 기능을 구현하기 위한 부품의 모입인 모듈(Module)을 표현한다. 인테그레이션 모듈은 서로 다른 복수의 적용 분야(예로 하드웨어와 소프트웨어)에 속한 부품으로 이루어진 참여 부품들(Participating Parts)을 하위 부품으로 연결하고 있다. 참여 부품들은 단위 기능 구현에 꼭 필요한 요소 부품으로 구성되어 있다. 제안된 제품자료모델에서는 소프트웨어 하드웨어 제품이 독립된 제품구조를 가지도록 설계되어 있으므로, 참여 부품들은 각자 별도의 소프트웨어 하드웨어 제품구조에 포함되어 있다. 또한 인테그레이션 모듈에는 인테그레이션

에 필요한 인수(Parameter), 인수 값, 그리고 부품들 간의 정의(예로 Schema Diagram)등 모듈에 참여하는 부품들 간의 인터페이스 정보를 포함할 수 있다.

[Figure 1]에는 간단한 하드웨어와 소프트웨어 제품구조와 이들 사이의 인테그레이션 모듈의 예가 정의되어 있다. [Figure 1]의 하단 좌측에는 motor와 sensor를 포함하는 하드웨어 제품구조가, 우측에는 program 부품을 포함하는 소프트웨어 제품구조가 정의되어 있다. 3개의 참여 부품은 이를 하위 부품으로 가지는 인테그레이션 모듈([Figure 1]의 integration module 참고)에 연결되어 있다.



[Figure 1] Structure of Integration Module

이 시스템의 전체 기능은 sensor에 들어오는 입력 값이 10 이상일 경우 모터를 회전 시키고, 이외의 경우에는 모터 회전을 멈추게 된다([Figure 1]의 program 참조). 인테그레이션 모듈은 하드웨어 부품인 motor 및 sensor와 소프트웨어 부품인 program의 관계를 표현하고 있다. 인테그레이션 모듈에 연결된 문서에는 해당 인터페이스에 필요한 각종 기술 자료가 첨부될 수 있다([Figure 1]의 Sensor Input 값과 모터 회전 방향 정의 참조).

인테그레이션 모듈의 역할은 첫째, 특정 기능을 위하여 참여하는 하드웨어, 소프트웨어 부품(Participating Hardware and Software Parts)들을 하나의 모듈 단위로 관리할 수 있다. 예로 시스템 기능을 구현하기 위해 필요한 motor, sensor 그리고 program 소프트웨어 부품이 integration module을 통해 모두 모듈화 되었다. 이 때 참여 부품은 각각

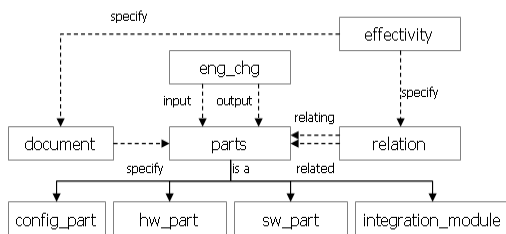
의 하드웨어 혹은 소프트웨어 제품구조에도 포함되게 된다.

둘째, 참여하는 부품에 설계변경이 일어날 경우 서로 다른 적용 분야의 관련 부품을 쉽게 검색할 수 있다. 예로 motor를 변경할 경우 integration module 부품을 통해 sensor와 program의 영향을 검토할 수 있다. 이 때 인테그레이션 모듈에는 속도와 방향 인수가 각각 정의되어 있으므로 검토 시 참고할 수 있다.

마지막으로 인테그레이션 모듈은 참여 부품 간의 인터페이스 정보를 정의할 수 있다. 예로 동일한 motor 부품에 대한 속도와 방향 인수는 다른 기능을 위한 부품 간 관계에서 다른 값을 가지며, 이는 각각의 인테그레이션 모듈에 정의할 수 있다.

4.2 제품자료모델

[Figure 2]의 제안된 제품자료모델은 기존 연구 [1]에서 제안한 모델을 확장하였다. 일반화된 객체인 parts의 하위 객체로 각각 제품구성, 하드웨어 부품, 소프트웨어 부품, 그리고 인테그레이션 모듈을 표현하는 config_part, hw_part, sw_part, 그리고 integration_module 객체가 존재한다. 하위 객체는 상위 객체와 is_a 관계로 연결되어 있으며, 상위 객체의 모든 속성을 상속받는다. 각 제품은 관련된 기술 문서를 연결하기 위하여 document 객체와 연관되어 있다.



[Figure 2] Proposed Product Data Model

각 제품(부품)은 다른 부품과 연관관계를 가질 수 있으며, 이를 표현하기 위하여 관련된 두 제품

을 연결하는 relating과 related 속성을 가지는 relation 객체를 이용한다. relating과 related 속성은 관계의 방향성을 나타내며, 예로 조립 관계에서 relating은 조립품을 related는 부품을 연결한다. 이 객체는 조립이나 구성관계를 주로 표현하며, 이 관계의 모임을 제품구조(Product Structure)라고 한다.

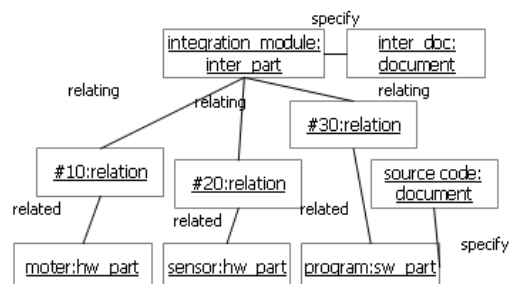
각 parts의 하위 객체는 다른 객체와 relation 객체로 연결할 때 다음 제약 조건을 가진다. relation 객체의 relating에 config_part가 연결될 경우 related에는 hw_part, sw_part만 연결될 수 있다. relating에 hw_part가 연결될 경우 related에는 hw_part만 연결될 수 있다. relating에 sw_part가 연결될 경우 related에는 sw_part만 연결될 수 있다. relating에 integration_module이 연결될 경우 related에는 hw_part, sw_part가 둘 다 올 수 있다.

유효성 객체(effectivity)는 제품구조가 유효한지를 결정하는 조건을 표현하고 있다. 일반적으로 유효성은 제품이나 제품구조에 적용되지만 제안된 모델에서는 소프트웨어 부품의 소스코드를 관리하는 문서(document)에도 적용된다[1, 2].

제품의 변경을 기록하는 설계변경을 표현하는 eng_chg 객체는 input과 output 속성을 이용하여 설계변경 전과 후 제품을 표현할 수 있다.

4.3 인스턴스 모델 예

[Figure 3]은 [Figure 2]의 제품자료모델을 이용하여 [Figure 1]의 예를 표현한 인스턴스 모델이다.



[Figure 3] Instance Model of the Product Data Model

[Figure 3]에서 인테그레이션 모듈인 integration module는 motor, sensor, 그리고 program 부품으로 이루어진 제품구조를 구성하고 있다. 제품구조를 구성하기 위하여 #10, #20, 그리고 #30의 relation 객체들이 정의되어 있다. 그림에서 알 수 있듯이 인테그레이션 모듈은 각기 독립적인 제품구조를 가지고, 서로 다른 개발자에 의하여 관리되는 하드웨어 소프트웨어 부품을 추가적인 제품구조로 연결시키는 역할을 한다.

그림에서 소프트웨어 부품(program : sw_part)은 프로그램 소스 코드 문서(source_code : document)를 가지고 있다.

5. 구현 및 적용 예

5.1 구현

제안된 제품자료모델은 상용 관계형 데이터베이스 관리 시스템을 이용하여 구현되었다. 구현된 데이터베이스를 기반으로 Java 기반의 웹 응용프로그램 개발 환경을 이용하여 웹 기반 PDM 클라이언트를 개발하였다. PDM 클라이언트는 PDM의 기본 기능인 부품, 문서, 제품구조, 그리고 설계변경 관리 기능을 제공한다.

5.2 적용 예

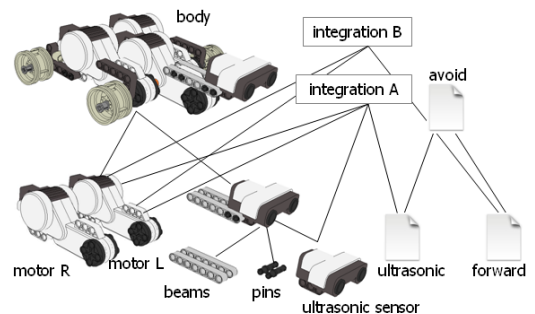
제안된 제품자료모델이 소프트웨어 하드웨어 협동 설계를 지원할 수 있는지 확인하기 위하여 구현된 PDM 시스템을 이용하여 교육 연구용 로봇 설계를 표현하였다. KEDB R5(Knowledge and Engineering DataBases lab. Robotics release 5)로 명명된 로봇은 레고 사의 NXT 로봇 시스템을 기반으로 2004년부터 다양한 연구와 교육 목적으로 개발되고 있다[16].

[Figure 4]의 KEDB R5의 구성 요소 중 본 연구와 관련된 부분은 로봇을 이동시키는 2개 모터(motor L, R), 전방 장애물을 회피하기 위한 초음

파 센서(ultrasonic sensor), 그리고 프로그램을 동작 시키는 NXT이다. 일종의 제어 컴퓨터인 NXT는 센서의 입력신호를 받고, 이를 처리하여 장애물을 회피하기 위한 출력 신호를 모터에 보내는 역할을 한다. NXT에서 실행되는 장애물 회피 프로그램은 센서로부터 입력을 받고 모터에 적절한 동작을 명령하는 역할을 한다. [Figure 5]는 관련 제품의 구조와 인테그레이션 모듈과의 관계를 표현하고 있다.



[Figure 4] KEDB R5 and Its Components



[Figure 5] integration Module for Robot Control

[Figure 5]의 좌측에는 하드웨어 제품구조가 정의되어 있다. 이 제품구조에는 로봇을 이동시키는 motor 부품과 장애물을 감지하는 ultrasonic sensor 부품이 포함되어 있다. [Figure 5]의 우측에는

로봇 구동을 위한 소프트웨어 제품구조가 정의되어 있다. 이 소프트웨어는 로봇이 장애물을 회피 하면서 이동하는 주요 기능을 수행한다. 이 제품은 avoid, forward, 그리고 ultrasonic 소스 코드 파일로 구성되어 있다. 이 중 forward 프로그램은 두 개의 motor를 같은 방향으로 구동시켜 로봇을 직진 시키는 기능을 구현하고 있다. 반면 ultrasonic 프로그램은 초음파 센서인 ultrasonic sensor에서 오는 신호를 감지하며 장애물이 있을 경우 두 개의 motor를 일정 시간 반대 방향으로 구동시켜 방향을 바꾸는 기능을 한다. 주 프로그램인 avoid는 두 개의 하위 프로그램이 동시에 실행되도록 하며, 이 중 장애물이 발견되었을 때 motor 프로그램을 멈추고 ultrasonic 프로그램이 실행되도록 조정하는 역할을 한다.

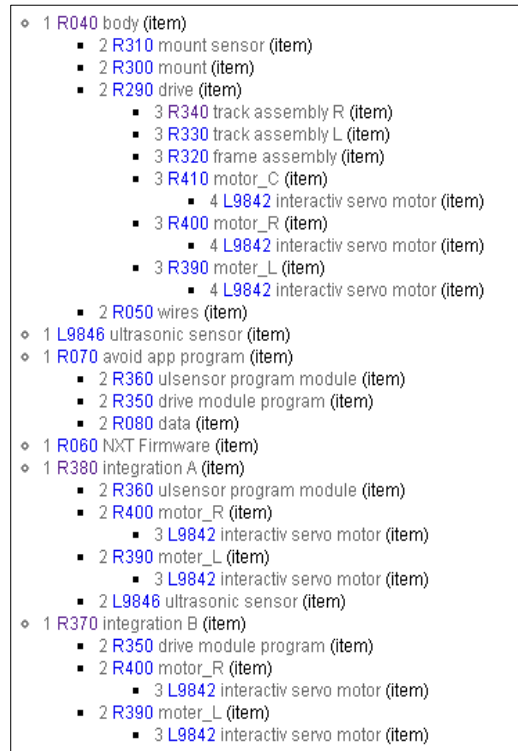
[Figure 5]에는 두 개의 인테그레이션 모듈이 정의되어 있다. 두 개의 모터, 센서, 그리고 ultrasonic 프로그램으로 구성된 Integration A는 센서의 신호가 있을 경우 모터를 반대 방향으로 회전 시켜 방향을 바꾸는 기능을 위한 통합을 정의하고 있다. 그러므로 이 인테그레이션 모듈에는 회피해야 할 센서의 신호 강도(혹은 장애물까지 거리), 두 모터의 속도, 회전 방향, 구동 시간 등이 인수로 정의될 수 있다. 또 다른 인테그레이션 모듈 Integration B는 직진 기능을 위한 통합을 정의하고 있다. 이 부품은 두 개의 모터와 forward 프로그램을 참여 부품으로 가지며 모터의 속도를 인수로 정의할 수 있다.

5.3 적용 결과

[Figure 6]는 PDM 시스템의 제품구조 탐색 기능을 이용하여 KEDB R5의 제품구조를 출력한 화면을 보여준다. 그림에서 motor, sensor, 그리고 소프트웨어 부품과 관련 제품구조를 확인할 수 있다([Figure 6] 제품구조상의 L9842 interactive servo motor, L9846 ultrasonic sensor, 그리고 R070 avoid app program 부품). 예로 L9842 interactive

servo motor의 경우 R040 body 아래 R290 drive 부품의 하위 부품으로 소프트웨어 제품구조와 독립적인 하드웨어 제품구조로 표현되어 있다.

각각의 소프트웨어 하드웨어 부품으로부터 R380 integration A와 R370 integration B 인테그레이션 모듈이 정의되어 있는 것을 알 수 있다. 예로 R380 부품의 경우, 참여하는 모터, 초음파 센서, 그리고 소프트웨어 부품이 포함되어 있는 것을 알 수 있다. 또한 인터페이스를 위한 인수가 R380에 속한 문서에 정의된다.

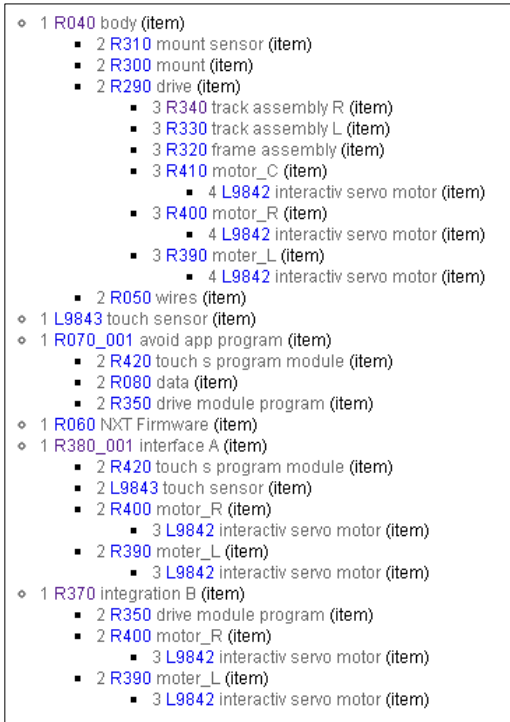


[Figure 6] Windows of HW and SW Product Structure of KEDB R5

[Figure 7]은 현재 로봇인 KEDB R5A의 초음파 센서를 터치 센서로 변경하여 KEDB R5B를 생성한 설계변경 관리 화면을 보여주고 있다. [Figure 7]의 좌측 화면에 설계변경 결과로 생성된 KEDB R5B의 제품구조를 보여주고 있다. 이 제품구조에

는 터치 센서(L9843 touch sensor 부품)가 추가되었으며, 이 변화로 인해 소프트웨어 부품과 관련 인테그레이션 모듈(R380_001 integration A)이 변경된 것을 확인할 수 있다.

이 설계변경을 완성하기 위하여 초음파 센서를 터치 센서로 변경할 필요를 확인한 하드웨어 설계자는 변경 대상인 하드웨어 부품(L9846 ultrasonic sensor)이 속한 제품구조를 확인하여 설계변경으로 영향을 받는 부품 들을 확인하게 된다. 해당 부품을 하위 부품으로 가지는 상위 부품을 찾는 Where Used 연산을 이용하여 관련된 하드웨어 부품 외에 인테그레이션 모듈(예에서 R380 integration A부품)을 확인할 수 있으며, 설계자는 인테그레이션 모듈의 하위 제품구조를 통해 설계변경이 2개의 모터와 1개의 소프트웨어 부품에 영향을 줄을 확인하게 된다. 인터페이스 인수와 값은 해당 인테그레이션 모듈 객체에 정의된 문서를 통하여 확인 할 수 있다.



[Figure 7] Windows of HW and SW Product Structure of KEDB R5 after the Change

이 영향은 소프트웨어 설계자에게 통보되며, 소프트웨어 설계자는 인테그레이션 모듈에 속한 부품들 간의 상관관계를 고려하여 새로운 소프트웨어 부품을 생성하게 된다(R420 touch s program module). 이 과정에서 소프트웨어 하드웨어 설계자는 상호 영향을 고려해야 할 대상을 함께 공유하며 협동설계를 진행하여 설계 대안을 완성하게 된다.

6. 결 론

기업들은 소프트웨어와 통합된 하드웨어를 경쟁력을 확보할 수 있는 효과적인 도구로 인식하고 이를 개발할 수 있는 방법과 지원 시스템을 요구하고 있다. 통합 개발 지원에 관한 연구는 소프트웨어를 위한 SCM과 하드웨어를 위한 PDM의 통합에 관한 연구로 진행되어 왔다. 하지만 기존 SCM PDM 통합 연구는 소프트웨어 엔지니어링 관점에서 접근하여 하드웨어 개발 중의 설계변경을 포함한 협동 설계 기능을 제공하지 못하고 있다.

본 연구는 기존의 SCM PDM 통합 표현 제품자료모델을 기반으로 서로 독립적인 소프트웨어 하드웨어 제품구조를 연결하는 인테그레이션 모듈을 제안하였다. 인테그레이션 모듈은 기능에 관련된 소프트웨어 하드웨어 부품을 모듈화하고, 이들 간의 영향 관계를 쉽게 찾을 수 있게 해주며, 인터페이스를 위한 인수를 PDM에 저장할 수 있게 해준다.

제안된 제품자료모델은 PDM 시제품으로 구현되었고, 그 유효성을 확인하기 위하여 로봇 설계 예를 표현하여 보았다. 로봇 설계를 적용한 결과, 설계변경 과정에서 소프트웨어 하드웨어 설계자는 변경해야 할 부품에 영향을 주는 소프트웨어, 하드웨어 부품과 영향 정보를 인테그레이션 모듈을 통하여 쉽게 확인할 수 있었다. 또한 다른 부분의 설계자와 관련 부품들을 쉽게 공유할 수 있었으며, 이를 통해 효과적으로 협동 설계를 진행할 수 있었다.

본 연구의 결과는 이전까지 암시적으로 관리되었던 소프트웨어 하드웨어의 인테그레이션 정보를 전사적 설계 정보 저장소인 PDM에 명시적으로

저장할 수 있도록 한다. 그러므로 이를 기반으로 변경 통지나 설계변경 자동화 시스템에 대한 추후 연구가 가능하다.

참 고 문 헌

- [1] 도남철, 채경석, “제품자료 관리와 소프트웨어구성관리 통합”, 『한국CAD/CAM학회 논문집』, 제13권, 제4호(2008), pp.314-321.
- [2] Do, N. and K. Chae, “A Product Data Management architecture for integrating hardware and software development”, *Computers in Industry*, Vol.62(2011), pp.854-863.
- [3] Estublier, J., J.-M., Favre, and P., Morat, “Toward SCM/PDM Integration?”, *ECOOP, SCM-8, LNCS 1439*, (1998), pp.75-94.
- [4] Wstfechtel, B. and R. Conradi, “Software Configuration Management and Engineering Data Management : Differences and Similarities”, *ECOOP, SCM-8, LNCS 1439*, (1998), pp.95-106.
- [5] The Association of Swedish Engineering Industries, *Product Data Management and Software Configuration Management-Similarities and Differences*, Special Report, 2001.
- [6] Svensson, D. and L. Crnkovic, “Information Management for Multi-Technology Products”, *International Design Conference-Design*, Dubrovnik, 2002.
- [7] Salmela, M., M. Kerttula, and J. Kaariainen, “Web-based Virtual Prototypes-an Enabling Technology for Collaborative PDM”, *7th International Conference on Concurrent Enterprising*, Bremen, 2001.
- [8] Krebs, T., L. Hotz, and A. Gunter, “Knowledge-based configuration for configuring combined hardware/software systems”, *PuK*, 2002.
- [9] Nguyen, T. N., “A Unified Model for Product Data Management and Software Configuration Management”, *21st IEEE International Conference on Automated Software Engineering*, (2006), pp.269-272.
- [10] Crnkovic, I., U. Asklund, and A. P. Dahlqvist, *Implementing and Integrating Product Data Management and Software Configuration Management*, Artech House Publishers, 2003.
- [11] Dahlqvist, A. P., “Product Data Management and Software Configuration Management Integration”, *Doctoral Thesis*, Dept of Computer Science and Electronics, Malardalen University, 2005.
- [12] El-Khoury, J., “Model data management-towards a common solution for PDM/SCM system”, *ACM*, Lisbon, Portugal, 2005.
- [13] Estublier, J. and G. Vega, “Reconciling software configuration management and product data management”, *Proceedings of the 6th joint meeting of the European software engineering conference and the ACM SIG SOFT symposium on The foundations of software engineering ESECFSE*, 2007.
- [14] Siemens PLM Software, “Team Center”, <http://www.plm.automation.siemens.com>, 2012.
- [15] PTC, “Windchill”, <http://www.ptc.com>, 2012.
- [16] KEDB, “KEDB Robotics”, <http://sites.google.com/site/kedblab/kedb-argos-robotics>, 2012.

◆ 저 자 소 개 ◆

**도 남 철 (dnc@gnu.ac.kr)**

포항공과대학교 산업공학학사, 산업공학석사, 산업공학박사 학위를 취득하였으며, 삼성중공업, 볼보건설기계코리아, 한국전자통신연구원에서 근무하였다. 현재 경상대학교 산업시스템공학부 부교수로 재직하고 있으며 관심 분야로는 Product Data Management, Product Lifecycle Management, Sustainable Product Development 그리고 Software Integrated Product Design 을 포함한다. 저서로는 PLM이해와 응용(생능출판사), 지속가능생산과 소비 미래를 저축하다(경상대학교 출판부)가 있다.