

컴포넌트 유통환경을 위한 컴포넌트 메타데이터 레지스트리 구축 : C_MDR

(A Construction of the C_MDR(Component_MetaData Registry)
for the Environment of Exchanging the Component)

송치양[†] 임성빈^{**} 백두권^{***} 김철홍^{****}
(Chee-Yang Song) (Sung-Bin Yim) (Doo-Kwon Baik) (Chul-Hong Kim)

요약 21세기 지식기반 사회로의 정착화와 글로벌 인터넷화 추진으로 소프트웨어는 대형화 및 복잡화 되어가고 있으며, 그 수요는 폭주하는 실정이다. 이에, 표준화된 컴포넌트의 개발 및 유통을 통한 재사용의 활성화가 최근 산업계와 학계에서 중요한 이슈로 부각되는 실정이다. 현재, 컴포넌트의 재사용을 위하여 해외 컴포넌트 판매 마켓에서는 자사별 판매하는 제품의 특성에 따라 정보를 제공하고 있으나, 상이하게 정보를 정의하고 있으며, 그 정보의 수준이 미약하며, 표준화되지 않은 메타정보를 제공하고 있다. 즉, 국제표준 ISO 11179에 기반한 컴포넌트 데이터 레지스트리의 구축이 이루어지고 있지 않다. 국내에서는 2001년도에 공용 컴포넌트의 출시를 추진하고 있다. 따라서, 개발된 컴포넌트의 정보 공유와 유통을 위한 지원도구로서 표준화된 컴포넌트의 메타정보를 서비스 해주는 시스템이 필요하다.

본 논문은 컴포넌트 재사용 활성화를 일환으로 체계적인 정보공유와 정보유통을 위하여, 제품화된 공용 컴포넌트에 대한 표준화된 메타정보의 등록 및 관리를 제공해주는 ISO 11179 표준에 준거한 컴포넌트 유통환경의 도구로서, 컴포넌트 메타데이터 레지스트리(C_MDR) 시스템을 제시한다. 이를 위해, 컴포넌트에 대한 메타정보의 명세 플랫폼을 제시하고, 이 플랫폼에 따라 메타정보를 정의하고, 또한 타 시스템과의 정보의 호환성 증진을 위해 XML을 이용해 표현한다. 그리고 시스템 디자인을 위해 3계층 아키텍처 표현방식을 적용하여 단순하고 이해성있는 시각화 모델링을 제공한다. 시스템 구현은 웹 상의 인터넷을 통해 컴포넌트 메타정보를 서비스를 할 수 있도록 시스템은 구축하며, ASP 개발언어와 PC용 RDMS 오라클을 사용한다. 이로써, 제품화된 컴포넌트에 대한 유통 메타정보의 표준화를 기할 수 있고, 재사용을 위한 유통 지원도구로서 지원이 가능할 것이다.

Abstract As the information-intensive society in 21c based on the environment of global internet is promoted, the software is getting more large and complex, and the demand for the software is increasing briskly. So, it becomes an important issue in academic and industrial field to activate reuse by developing and exchanging the standardized component. Currently, the information services as a product type of each company are provided in foreign market place for reusing a commercial component, but the components which are serviced in each market place are different, insufficient and unstandardized. That is, construction for Component Data Registry based on ISO 11179, is not accomplished. Hence, the national government has stepped up the plan for sending out public component at 2001. Therefore, the systems as a tool for sharing and exchange of data, have to support the meta-information of standardized component.

In this paper, we will propose the C_MDR system: a tool to register and manage the standardized meta-information, based upon ISO 11179, for the commercialized common component. The purpose of this system is to systemically share and exchange the data in chain of acceleration of reusing the component. So, we will show the platform of specification for the component meta-information, then

† 학생회원 · 고려대학교 컴퓨터학과
songyang@kt.co.kr

** 비회원 : (주)한국유니시스 IS 컨설팅팀
blucsol@swws2.korea.ac.kr

*** 종신회원 : 고려대학교 컴퓨터학과 교수
baik@swws2.korva.ac.kr

**** 비회원 : 한국전자통신연구원 컴퓨터소프트웨어기초연구실 연구원

kch@etri.re.kr

논문접수 : 2001년 3월 7일

실사완료 : 2001년 8월 28일

define the meta-information according to this platform, also represent the meta-information using XML for enhancing the interoperability of information with other system. Moreover, we will show that three-layered expression make modeling to be simple and understandable. The implementation of this system is to construct a prototype system of the component meta-information through the internet on www, this system uses ASP as a development language and RDBMS Oracle for PC. Thus, we may expect the standardization of the exchanged component metadata, and be able to apply to the exchanged reuse tool.

1. 서론

21세기 지식기반 사회로의 정착화와 초고속망의 조기 구축에 따른 글로벌 인터넷화 추진으로 소프트웨어는 대형화 및 복잡화 되어가고 그 수요는 폭주하는 실정이다. 이에 소프트웨어 품질향상, 개발기간 단축 그리고, 소프트웨어 수요 충족을 위해 컴포넌트 기반의 재사용을 통한 조립생산이 그 해결책으로 부각되고 가시화 및 상용화가 활성화되고 있다. 현존 실용화된 컴포넌트 개발도구인 MS사의 COM/DCOM, OMG의 CORBA, 그리고 SUN사의 EJB를 이용하여 컴포넌트 마켓 판매 제품으로 OLE, ActiveX, VBX 및 JavaBeans 등의 형태로 유통채널을 통하여 무료 또는 유료의 형태로 통용되는 상황이다.

컴포넌트 정보유통, 정보공유와 재사용 율의 극대화를 도모하기 위해서, 초기 컴포넌트를 개발할 때부터, 표준화된 참조 모델과 CASE 도구(tool)에 의거한 컴포넌트를 생산하고, 기 명세된 자료로부터 컴포넌트의 메타정보를 추출하여 컴포넌트 메타정보 데이터베이스를 구축하고 이를 사용자에게 정보를 제공함으로써, 결국에는 원하는 컴포넌트를 획득할 수 있도록 지원해주는 서비스가 중요한 이슈로 부각되고 있다. 이를 위한 참조 모델, 지원도구 체제, 분류체제 및 유통체제에 대한 표준화와 이의 실용화에 대한 연구에 많은 진전을 보이고 있다.

컴포넌트 유통을 위한 국제적 표준화 추진에 활발한 활동을 하고 있는 국가는 일본의 CBOP[1]이며, 그 외의 국가는 미진하다. CBOP은 컴포넌트(비즈니스 객체) 정보 공유 및 유통의 활성화를 위한 일본 컨소시엄 단체로서, 컴포넌트 벤더로부터 컴포넌트를 받아, 인증하고, 등록하고, 유통시키는 전 과정의 프로세스, 관련 기술에 대한 구축 및 표준화를 도모한다. 컴포넌트 정보서비스를 제공하는 대표적 컴포넌트 마켓의 업체로 미국의 ComponentSource사, ImagicCom사, 그리고 Flashline사 들 수 있다. 이들은 자사별 판매하는 제품의 특성에 따라 정보를 제공하고 있으나, 상이하게 정보를 정의하고 있

으며, 그 정보의 수준이 미약하며, 표준화되지 않은 메타정보를 제공하고 있다.

MDR에 대해서 환경분야, 교통분야 및 보건분야 등에 활발한 진전을 보이고 있으나, 컴포넌트 부문에 대해서는 상기와 같이 미진한 실정이다. MDR의 구축 사례로서, 미국 환경보호청의 EDR(Environmental Data Registry)[2], 미국 교통부의 ITS(Intelligent Transportation), 미국 보건복지부의 USHIK(the United States Health Information Knowledge base) 그 외 인구 경제 데이터 레지스트리, 그리고 노동 통계 데이터 레지스트리 등이 있다.

결국, 컴포넌트 마켓에서 자사별 취급하는 제품의 특성에 따라 컴포넌트 정보서비스를 제공하고 있으나, 표준적 측면 즉, 데이터 요소 및 데이터 레지스트리 구축 기능 등이 가미되지 않은 미성숙의 상태라 할 수 있다. 이것은 국제표준 ISO 11179[3]에 근거한 메타정보의 추출을 통한 메타정보 데이터베이스 구축 및 효과적인 정보교환에 대한 구축이 이루어지고 있지 않다. 미래에, 컴포넌트 시장의 급성장이 예견됨으로써 관련 표준화 작업과 지원도구가 필요하다. 이를 해결하기 위해, 개발된 컴포넌트의 정보 공유와 유통을 위한 지원도구로서 표준화된 컴포넌트의 메타정보를 서비스 해주는 시스템이 요구된다.

본 논문에서는 컴포넌트의 재사용을 활성화하기 위해, 체계적이고, 효율적인 정보관리와 정보공유를 목적으로, 제품화된 공용 컴포넌트에 대해 표준화된 메타정보서비스를 제공해주는 ISO 11179 표준에 준거한 컴포넌트 유통환경의 도구로서 컴포넌트 메타데이터 레지스트리(C_MDR : Component MetaData Registry)를 제시한다.

이를 위해, 제2장에서는 MDR 관련 기술과 상용서비스 현황을 분석하고, 제3장에서는 컴포넌트의 메타정보를 정의한다. 컴포넌트의 의미적 속성정보와 정보저장소 내 관리정보들을 분석하여, 컴포넌트 제품에 대한 필수적이고, 표준적인 메타정보를 추출한다. 컴포넌트 메타정보 속성 정의 플랫폼을 제시, 이 플랫폼에 의거하여 선정된 메타정보를 정의한다. 제4장에서 추출된 메타정

보를 관리할 컴포넌트 메타데이터 레지스트리 시스템을 구축한다. 즉, 컴포넌트 저장관리시스템, 변경관리시스템 그리고 검색관리시스템을 UML 기반하의 3계층 표현방식에 의해 설계하고, 시스템을 구현한다. 그 외 분류코드 정보관리 서비스 등을 구축한다. 제5장에서는 본 시스템에 대해 비교평가에 대해 다루었다.

2. 관련 연구

본 단원은 데이터의 이해성을 증진시키고, 공유할 수 있도록 만들기 위한 MDR, 일본의 CBOP과 상용 컴포넌트에 대한 판매 및 정보서비스를 제공하는 각 사별 현황을 분석한다.

2.1 MDR(MetaData Registry)

MDR에 대해서는 ISO 11179에서 정의하고 있다. ISO 11179는 메타데이터 즉, 데이터요소의 명세와 표준화를 위하여 데이터의 요청과 등록을 유용화하고, 데이터의 접근과 사용을 촉진하기 위하여 데이터를 이해할 수 있고, 공유할 수 있도록 만들기 위한 표준과 등록에 대한 내용을 설명하고 있다.

MDR은 메타데이터의 등록과 인증을 통하여 표준화된 메타데이터를 유지 관리하며, 메타데이터의 명세와 의미의 공유를 목적으로 한다. 따라서, 이것은 데이터 공유를 위한 기본 틀로서 공유를 위한 기본요소로 데이터 요소를 사용하고, 이러한 데이터 요소가 그 내포적인 의미를 충분히 나타내어 줄 수 있는 구조 내에 등록하여 각 응용에 활용토록 제공한다. 상이한 시스템간 또는 조직간의 데이터 공유를 제공하기 위하여 어떤 특정 분야의 데이터 요소에 대한 특성 정보로 추출된 메타데이터를 등록하여 모아 놓은 참조 정보의 집합체이다.

MDR의 구조는 다음과 같은 부분으로 되어있다.

- ① 관리부분(stewardship region)
- ② 명명과 식별부분(naming & identification region)
- ③ 분류부분(classification region)
- ④ 데이터 요소 관리부분(data element administration)
- ⑤ 데이터 요소 개념 관리부분(data element concept administration)
- ⑥ 개념 또는 값 영역 관리부분(concept or value domain administration)

2.2 CBOP

CBOP(Consortium for Business Object Promotion)은 컴포넌트 마켓과 컴포넌트 개발자간의 컴포넌트 공유를 목적으로, 컴포넌트(비즈니스 객체) 정보공유 및 유통의 활성화를 위한 일본 컨소시엄 단체이며, 비즈니스

스 컴포넌트 공유를 위한 기반 플랫폼을 제시하여, 국제 표준을 추구하고 있다. 2000년 1월 ISO/IEC JTC1/SC32/WG2 SAF25에 제안된 NWI Proposal은 ISO/IEC 11179 명세를 확장한 기업 비즈니스 객체를 분류하고 구조화를 위한 메타데이터의 표준화에 초점을 두고, 기업 객체 아키텍처의 공용 참조모델 구축 방법과 비즈니스 객체들을 대상으로 하고 있다.

초기 개발시부터 일원화된 컴포넌트 제품 생산을 위한 아키텍처 참조모델과 기준은 물론, 컴포넌트 벤더로부터 컴포넌트를 받아, 인증하고, 등록하고 및 유통시키는 전체적 과정의 프로세스 및 관련 기술에 대한 구축 및 표준화를 도모한다. 개발 CASE 도구와 연계된 레포지토리를 통한 컴포넌트의 메타정보, 패턴정보의 라이브러리화 그리고 실제 객체를 XML 정보교환에 의거한 광범위한 서비스를 도모하고 있다. 이 모델 및 시스템은 구축 중에 있다고 할 수 있다.

2.3 상용 컴포넌트 정보서비스 현황

국외의 경우, 소프트웨어 생산기술, 컴포넌트 마켓과 경험이 우리나라에 비해 풍부한 외국은 기존 객체 기반의 산출물, 이의 컴포넌트화를 위한 표준화와 관련 지원 체계에 대해 다각적인 연구가 수행중이다. 지원체계의 일부로서, 컴포넌트의 정보서비스 즉, 컴포넌트 레지스트리는 상용화된 컴포넌트를 대상 정보로 일부에서 활발히 마켓에서 서비스 중이다.

해외 컴포넌트 정보서비스를 제공하는 컴포넌트 마켓의 업체별 현황을 살펴보면, ComponentSource사[4]는 컴포넌트 레지스트리 시스템을 구축, 카테고리별로 분류하여, COM, 액티브엑스, 자바, 델파이, 비즈니스 컴포넌트의 상용 컴포넌트를 판매한다. ImagicCom사[5]는 컴포넌트 레지스트리를 제공하지 않으며, 타입 및 공급자별 인덱스의 정보를 제공한다. Flashline사[6]는 컴포넌트 레지스트리의 형태를 갖추고, ActiveX, COM 및 자바 컴포넌트를 대상으로 판매하는데, 이용자로 부터 컴포넌트 요구를 받아, 이것을 개발하여 제공한다. Vbxtras사는 컴포넌트 레지스트리라 부르나, 카테고리 수준의 정보서비스이며, VBX를 제공한다.

국내의 경우, 컴포넌트 유통서비스 업체로 Component Bank사 및 아이메카사를 들 수 있으며, MDCOM사는 시작단계에 있다. 현존 비즈니스 객체를 대상으로 컴포넌트의 유통체계 구축에 초점을 두고, 개발방법론, 품질평가 그리고 시장 판매를 포함하는 광범위한 유통시스템을 추진 중에 있다.

이와 같이, 일부 시스템에서 컴포넌트 데이터 레지스트리라 칭하고 있으나, 컴포넌트에 대한 표준화된 데이

타요소의 메타정보로 구성되어 있지는 않다.

3. 컴포넌트 메타정보 질의

C_MDR은 개발된 컴포넌트에 대한 제품규격을 나타내는 기본적인 메타정보 혹은 약식정보를 등록관리하고, 검색을 통하여 정보를 서비스하는 시스템이다. 이 시스템은 저장되는 데이터요소가 잘 정립되어야 내실화된 정보의 구축, 정확한 컴포넌트의 탐색 그리고 타 시스템과의 원활한 정보의 공유를 보장할 수 있다. 즉, 컴포넌트에 대한 표준화된 메타데이터로 구축되어야 한다.

현존, 컴포넌트 마켓 시장에서 서비스하고 있는 컴포넌트 정보서비스 시스템은 ISO 11179에서 정의한 데이터요소의 구조인 객체클래스(실 세계의 생각, 사물의 집합), 속성(객체 부류내의 모든 구성원을 특징짓는 성격), 그리고 표현(값, 영역, 데이터형, 측정 단위 등의 표현형태)에 준한 각 데이터요소에 대한 정의가 미약하다. 즉, 표준화된 컴포넌트의 메타정보를 서비스하고 있지 못하다.

일반적으로 사람들은 그 특징을 통하여 사물을 인식한다. 이러한 사물들의 특징을 나타내는 것이 데이터이다. 표현이 일관되고 정확한 실세계 항목을 보장하고, 각각의 데이터 요소와 관련된 특징들을 기록하기 위해 데이터의 기본단위인 데이터 요소에 대한 명세가 필요하다. 데이터요소는 여러 조직에 걸쳐 공유할 만한 가치가 있는 데이터들을 말한다. 따라서, 효과적인 컴포넌트의 정보공유와 관련 시스템간의 상호 연동을 제공하기 위해서 컴포넌트 메타정보의 각 데이터요소에 대한 속성들이 잘 정립되어야 하고, 표준화가 요구된다.

본 장에서는 컴포넌트의 메타정보의 선정기준을 정립하고, 이 기준에 의거하여 메타정보를 추출하고, ISO 11179 데이터요소의 속성에 의거하여 속성 명세 플랫폼을 제시하고, 정의하고, 추출된 메타정보의 XML 표현에 대해 기술한다.

3.1 컴포넌트 메타정보 선정기준

유용한 메타정보를 추출하기 위해서, 컴포넌트 정보를 대표할 수 있고, 공통성과 표준성을 지닌 메타정보를 선정해야 한다. 이에, 표준성, 기술성 및 사용성 측면에서 메타정보 추출의 선정기준은 다음과 같다.

가. 표준적 속성 측면

- 공통적이고 범용적으로 사용중인 데이터요소 고려 현재, 상용화하여 컴포넌트 정보서비스를 제공하는 시스템의 관리 정보들과 CASE 도구와 연계된 정보저장소의 관리정보
- ISO 11179 표준에 준한 데이터요소 명세의 속성

에 부합한 메타정보

컴포넌트 메타정보별 명료한 데이터요소의 정의

- 컴포넌트의 재사용을 극대화 할 수 있는 표준적 및 필수적 정보

나. 기술적 속성 측면

- 컴포넌트의 자체적 명세특성 요소 : 컴포넌트가 지닌, 기능정보와 분류별(다섯 부문)에 의한 체계적, 계층적 정보 구성
- 체계적 이력관리 위한 형상정보 및 컴포넌트 인터페이스 속성 정보
- 상호 운용성의 환경정보 및 신뢰성을 제공하기 위한 품질관련 메타정보

다. 사용적 속성 측면

- 사용시에 필요한 정보 부문을 강화
사용자가 컴포넌트를 이용하고, 획득하기 위해 요구되는 운용정보, 인터페이스정보, 그리고 제공자 정보를 강조
- 명세정보 부문을 한눈에 컴포넌트를 파악할 수 있도록 제공
- 미래 컴포넌트 정보의 확장성을 고려

3.2 컴포넌트 메타정보 추출

컴포넌트의 메타정보 선정기준에 의거, 컴포넌트 제품에 대한 메타정보를 추출한다. 추출된 정보가 C_MDR 시스템의 정보관리 메타정보가 되는 것이다. 컴포넌트에 관한 속성들로부터, 컴포넌트의 기본 명세를 의미하는 일반정보와 컴포넌트의 사용환경 측면에서의 상세정보로 그룹화한다. 이 두 가지의 그룹 속성은 특성에 따라 각각 2개와 3개의 세부속성들로 분류되어진다. 따라서, 컴포넌트의 일반 정보는 명세 정보(specification)와 형상 정보(configure)로 나누었고, 상세 정보는 환경 정보(environment)와 인터페이스 정보(interface), 그리고, 사용 정보(use)로 분류한다. 추출된 메타정보는 48개이다[7][8].

컴포넌트의 메타정보의 구성은 <그림 1>과 같으며, 분류별 세부 내역은 다음과 같다.

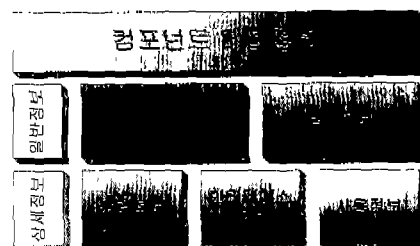


그림 1 컴포넌트 메타정보 분류체계

가. 일반정보

컴포넌트의 기본적 명세속성과 형상관리 정보로된 추상적이고, 일반화된 정보로 구성한다. 명세속성을 통해서 만으로도, 컴포넌트에 대한 컴포넌트의 기본 기능, 영역 및 상태 정보를 알 수 있다.

① 명세 정보(component)

명세정보는 컴포넌트에 대해서 가장 중심이 되고, 기본이 되는 속성들을 나타내는 것으로, 컴포넌트의 기능, 분류 및 상태 정보를 표현하며, 분류시에 사용되는 주요 키(primary key)와 같은 역할을 하는 속성들을 포함하고 있다.

구성 메타정보는 컴포넌트 아이디(identification), 컴포넌트 명칭(name), 컴포넌트 목적(purpose), 컴포넌트 설명(specification), 제품종류(product kind - COM, JavaBeans, CORBA, ActiveX, DLL 등), 관련범주(related scope-기술 컴포넌트, 비즈니스 컴포넌트, UI 컴포넌트 등), 컴포넌트 내용 대분류(category large), 컴포넌트 내용 중분류(category middle), 컴포넌트 내용 소분류(category small), 컴포넌트 단가(cost), 제공 결과물(results supported-실행 컴포넌트, 소스 코드, 개발보고서(명세, 설계, 시험 등), 최종수정일(final updated date), 컴포넌트 제작일(create date), 컴포넌트 등록일(register date)로 되어있다.

② 형상 정보(configure)

형상정보는 컴포넌트를 사용하기 전에, 컴포넌트가 어떤 도구나 타입으로 개발이 되어졌는지, 사전에 요구사항을 알 수 있도록 컴포넌트의 형상에 관련된 사항들에 대한 속성을 나타낸다. 이것은 컴포넌트의 개발 이력과 품질에 관련된 메타정보로 구성된다.

구성 메타정보는 컴포넌트 버전(version), 상태(status), 재사용기록(reuse history), 개발언어(development language), 개발표준(development standard), 개발툴(development tool), 품질평가결과(quality result), 품질평가기관(quality associate), 품질평가기준(quality criterion), 품질인증번호(quality number)로 되어있다.

나. 상세정보

컴포넌트에 대한 상세정보를 제공하기 위해서, 좀더 깊은 관점에서 세부적으로 표현되어 질 수 있는 사항들을 나타낸다. 이 정보는 컴포넌트를 이용하고, 획득에 관련된 정보로 환경정보, 인터페이스정보와 사용정보로 구성된다.

① 환경 정보(environment)

환경정보는 컴포넌트가 결합하기 위해서, 이 기존의 컴퓨터나, OS 등의 다른 환경의 컴포넌트와 연결할 때의

호환 가능한 정보에 대해서 속성을 나타낸다. 즉, 컴포넌트에 대한 결합적인면의 특성을 지원해주는 부분이다.

메타정보의 구성은 CPU 타입(cpu type), OS 타입(ostype), 필요 디스크량, 필요 메모리량 (memory space), 미들웨어 타입(middletype), DBMS(dbms), 프로토콜(protocol), 저장 타입(save type), 파일크기(file volume), 이식 제한(portability restriction)로 되어있다.

② 인터페이스 정보(interface)

인터페이스 정보는 컴포넌트의 인터페이스에 대한 속성의 서술을 표시하는 부분이다. 컴포넌트는 블랙박스의 형태를 띠고 있으므로 컴포넌트간의 대화나 외부적인 정보전달을 위한 유일한 방법이 컴포넌트가 소유하고 있는 인터페이스를 이용하는 것이다.

구성 메타정보는 인터페이스 이름(interface name), 메소드(method), 타입 (parameter type), 초기값(parameter init value), 허용범위(permit value), 널 사용여부 (permit null)로 되어있다.

③ 사용 정보(use)

사용정보는 본 컴포넌트에 대해 구매를 원할 경우에 필요한 공급사정보, 특히 정보와 관련 계약 정보를 담고 있다.

구성 메타정보는 공급사명(supplier name), 공급자 홈페이지(home page), 공급자 이-메일(e-mail), 공급자 주소(supplier address), 보안 제한(security restriction), 라이선스 제한 (license restriction), 특기사항(special facts), 관계컴포넌트(related component)로 되어있다.

3.3 컴포넌트 메타정보 명세 플랫폼 및 정의

컴포넌트의 메타정보(메타요소)의 속성을 정의하는 표준 플랫폼을 제시하고, 단원 3.2에서 추출된 각 메타정보별로 ISO 11179 표준에 준거하여 속성을 정의한다. 컴포넌트의 각 메타정보를 정의하기 위하여, 관련되는 속성들을 기반으로 메타정보에 대한 속성과 메타정보로 구성된 엔티티(그림 1)에서 5개의 그룹의 속성들을 실제 적용될 수 있도록 정의한다.

가. 컴포넌트 메타정보 속성 명세 플랫폼 및 정의

컴포넌트의 메타정보를 ISO 11179라는 메타데이터 정의 표준에 의해서 컴포넌트 메타요소라 정의하며, 이 표준에 의해서 컴포넌트 정보를 저장하기 위한 메타정보에 대한 속성을 크게 식별형, 정의형, 관계형, 표현형, 그리고 관리형 속성으로 <표 1>과 같이 분류한다. 각 5개 그룹형별로 정의를 하고, 다시 그룹내 속성항목별로 정의를 한다.

예시로서, 컴포넌트 메타정보 속성 정의 플랫폼에 의거하여 선정된 메타정보 중 컴포넌트 명세 부분의 "컴포넌

표 1 “컴포넌트 아이디”의 컴포넌트 메타정보 속성 명세서

*** 컴포넌트 메타정보 속성 명세서 ***		
식별형 속성	(1) 명칭(name)	컴포넌트 아이디
	(2) 식별자(identifier)	comp.spec.001
	(3) 버전(version)	V1.0 - 2000
	(4) 이명(alias)	id, 식별번호
	(5) 문맥(context)	컴포넌트 명세
정의형 속성	(6) 정의(definition)	컴포넌트의 고유 식별의 의미를 가지는 코드
관계형 속성	(7) 출처(origin)	관련사항 없음
	(8) 확인자(key)	관련사항 없음
	(9) 분류체계(classification scheme)	컴포넌트 명세 정보
	(10) 핵심어(keyword)	아이디, ID
	(11) 사용법, 사용빈도, 사용책임자(usage)	컴포넌트를 접근할 때마다 주요키로서 매번 사용을 한다. 접근 횟수가 가장 많은 데이터이다.
	(12) 특기사항(special item)	컴포넌트를 다루는 컴포넌트 요소의 주요키다.
표현형 속성	(13) 타입(type)	정수형
	(14) 형태(format)	integer
	(15) 최고길이(maximum length)	10
	(16) 최저길이(minimum length)	4
	(17) 허용범위(permit value)	1 ~ 9,999,999,999
	(18) 디폴트 값(default value)	9,999,999,999
	(19) 유일성(uniquness)	Yes
	(20) 널 허용(permit null)	No
관리형 속성	(21) 등록일자(registration date)	2000. 5. 31
	(22) 변경일자(updating date)	관련사항 없음
	(23) 변경사항(change facts)	관련사항 없음
	(24) 주석(comment)	표준적이고, 체계적인 코드
	(25) 책임자명(responsible user)	관련사항 없음
	(26) 책임자 전자우편(e-mail)	관련사항 없음

트 아이디”에 대해 정의한 것이 <표 1>이다. 또한,속성 정의에 대한 예시로서, 식별형 그룹에 대한 정의와 그룹 내 명칭에 대한 정의를 <표 2>에서 보여주고 있다[9].

◇ 식별형 속성

컴포넌트를 사용하기 위해서 표현하거나, 다루기 위한 식별자로서 이런 대표 기능들을 나타내는 메타정보들이다.

표 2 명칭(name)의 속성 정의

명칭	명칭(name)
정의	하나의 데이터 요소에 할당된 단일 혹은 다중 단어 명칭
의무	필수적
데이터유형	문자열

나. 컴포넌트 엔티티 속성 명세

메타정보들로 구성된 엔티티를 기술하기 위하여 다음 <표 3>와 같이 명세하여 사용하며, 속성별 정의내역은 다음과 같다.

- (0) 엔티티명 : 하나의 엔티티에 할당된 단일 단어의 이름을 나타낸다.
- (1) 정의 : 한 엔티티의 근본 성질을 표현하고 다른 엔티티들과의 차이점을 기술하는 문구
- (2) 동의서 : 주어진 명칭과는 다르지만, 동일한 엔티티의 개념을 나타내는 단일 혹은 다중 단어 호칭
- (3) 책임 : 엔티티를 관리하는 책임기관 및 책임자
- (4) 크기 : 엔티티가 실제 발생될 수 있는 양을 조사하여 나타내는 엔티티 발생 사이즈
- (5) 성장율 : 실제 엔티티를 적용할 때 발생될 수 있

표 3 “명세 정보”의 컴포넌트 엔티티 속성 명세서

*** 컴포넌트 엔티티 속성 명세서 . component ***				
1. 정의	컴포넌트의 기본정보를 가진다.			
2. 동의어				
3. 책임	전자통신연구원 소프트웨어기술연구소 컴포넌트연구팀			
4. 크기	최대:	최소:	평균:	
5. 성장률	추가:	삭제:	기간:	
6. 프로세싱 규칙				
7. 속성ID	8. 키사용	9. 발생수	10. 필수	11. 비고
id	PK	1	Yes	컴포넌트 아이디
name	AK	1	Yes	컴포넌트 이름
purpose		1	Yes	컴포넌트 목적
descript		1	Yes	컴포넌트 설명
product_kind		1	No	컴포넌트 종류
relate_scope		1	No	관련범주
l_category		1	Yes	내용대분류
m_category		1	Yes	내용중분류
s_category		1	Yes	내용소분류
cost		1	No	단가
version		n	Yes	버전
output		1	No	제공결과물
fupdate_date		1	No	최종 수정일
create_date		1	No	컴포넌트 제작일
regist_date		1	Yes	컴포넌트 등록일

- 는 추가량이나, 삭제량, 사용기간 등을 나타냄
- (6) 프로세싱 규칙 : 엔티티내에서 사용되어지는 운영 규칙에 대해서 기록을 한다. 예) 널 규칙, 삽입규칙, 삭제규칙, 제약사항 등
- (7) 속성 ID : 테이블의 레코드가 필수 있는 애트리뷰트(attribute)에 관한 ID
- (8) 키 사용 : 애트리뷰트가 키(key)의 역할을 하는지에 대한 유무사항
- (9) 발생수 : 애트리뷰트가 발생하는 횟수
- (10) 필수 : 데이터 입력이 필수적인지, 선택적인지에 대한 사항
- (11) 비고 : 기타 등의 주석내용

3.4 컴포넌트 메타정보의 XML 표현

본 컴포넌트 메타데이터 레지스트리 시스템(C_MDR)은 인터넷 웹 환경에서 컴포넌트 공유 뱅크를 위한 하나의 정보서비스 도구이다. 따라서, 인터넷상에서 데이터를 주고받기 위해서 다양한 응용을 지원하고 타 관련 시스템과의 연동성이 좋은 언어가 필요하다. 이에, 현재

호환성이 뛰어나고, 범용적으로 사용되는 XML을 사용한다. 이로서, 타 관련 컴포넌트 유통시스템과의 연동을 원활히 할 수 있다.

단원 3.2에서 추출된 컴포넌트 데이터요소(메타데이터, 메타정보)에 기반하여, 컴포넌트 레지스트리를 구축하기 위해, XML을 사용하여 컴포넌트 요소에 대해 DTD로 작성한 것이 (그림 2)이다[10][11].

```

<!-- XML DTD for Component Registry -->
<ELEMENT Component (GeneralInformation, SpecificInformation) >
<!-- General Information -->
<ELEMENT GeneralInformation (IdentificationProperty, ConfigurationProperty) >
<!-- Identification Property -->
<ELEMENT IdentificationProperty (ID, Name, Purpose, CreatedDate, ...) >
<ELEMENT ID (#PCDATA) >
<ELEMENT Name (#PCDATA) >
<ELEMENT Purpose (#PCDATA) >
.....
<!-- Configuration Property -->
<ELEMENT ConfigurationProperty (Version, DeveloperLang, DeveloperTool, Update, ...) >
.....
    
```

그림 2 컴포넌트 메타정보의 XML DTD 정의

(그림 2)에서 작성한 컴포넌트 요소에 대한 DTD 정의에 의거하여, (그림 3)에서는 하나의 컴포넌트를 대상으로 적용한 사례를 보여주고 있다.

```

<GeneralInformation>
  <IdentificationProperty>
    <ID> Comp278 </ID>
    <Name> Payroll Eval Component </Name>
    <Purpose> Payroll and Evaluation for employee </Purpose>
    <CreateDate> 2000-7-20 </CreateDate>
    .....
  </IdentificationProperty>
  <ConfigurationProperty>
    <Version> 2.1 </Version>
    <DevelopLang> EJB </DevelopLang>
    <DevelopTool> JBuilder 3.0 </DevelopTool>
    .....
  </ConfigurationProperty>
  .....
</GeneralInformation>
    
```

그림 3 컴포넌트 메타정보의 DTD 적용 사례

4. C_MDR 시스템 디자인 및 구현

본 장에서는 C_MDR 시스템의 분석, 설계 및 구현에 대해 기술한다. 먼저, 3계층 아키텍처 표현방식을 제시하고, 이에 따라 시스템 아키텍처의 모형화 분석, 시스템 모델링 그리고, 구현된 시스템에 대해 다룬다. 분석된 서브시스템은 컴포넌트 메타정보서비스, 컴포넌트 분류포드서비스, 운용자서비스, 부가서비스로 구성되어 있다.

4.1 3계층 표현방식 아키텍처

3계층 표현 방식(three-layer expression formula)의 아키텍처는 3개의 계층으로 구성된 표현 형태를 가진 아키텍처를 의미한다. 즉, 소프트웨어를 모델링 하는 경우에, 모델에 대해서 3개의 특징적인 수평적 영역을 주어서 모델구성을 구분하여, 보다 체계적이며, 효율적으로 관계를 가질 수 있도록 모델링을 제공하기 위한 표현방식이다. 기존의 모델링에서는 하나의 뷰(view)에 모든 내용을 한꺼번에 다 집어넣거나, 스테레오 타입(stereotype)이라는 것을 이용해서 구성원을 구분하였던 것을 보다 합리적인 분할 구조인 3계층으로 구분, 분할하여 표현할 수 있는 아키텍처를 제시한다. 3계층 표현 방식의 아키텍처의 3가지 계층으로는 사용자와 연관이 되는 사용자 관련 계층(user related layer), 주요기능을 담고 있는 주요 비즈니스 계층(primary business layer), 그리고 추가적인 기능이나, 종속적인 기능을 가진 보조 비즈니스 계층(secondary business layer)으로 구성된다. 제시된 표현방식이 시스템 모델링에 적용된다[12].

가. 사용자 관련 계층

실제 시스템이 외부 환경과 접촉하는 외부 인터페이스

스와 관련이 되는 부분을 표현하기 위한 계층이다. 주로, 비즈니스의 외부 환경과 통신을 하는 작업체들로 구성되어 있고, 일반적으로 사용자나 운영자가 다루는 사용자 인터페이스, 윈도우, 폼, 메뉴, 그리고 버튼 등의 그래픽에 관련된 컴포넌트 또는 모델 구성요소를 담고 있는 계층이다.

나. 주요 비즈니스 계층

시스템이 제공하고자 하는 주요 서비스나, 기능에 대한 내용을 담고 있는 계층이다. 실제 시스템에서 가장 중요한 비즈니스의 로직과 같은 주요한 기능의 역할들을 모아 놓은 계층이라고 생각할 수 있다. 하나의 소프트웨어나 시스템을 분석, 설계를 하는 과정에서 요구사항에서 요구하는 핵심적인 기능을 표현하고, 내부적으로 갖추어야하는 중심기능, 그리고 양쪽의 사용자 관련 계층과 보조 비즈니스 계층 사이를 관리해주는 중심기능의 표현 계층이다. 즉, 사용자 관련 계층을 통해서 들어온 데이터나 정보를 가지고, 실제 요구사항에 맞게 핵심기능이나, 중앙 제어 기능을 가진 컴포넌트를 담고 있는 기능 중심의 계층이다.

다. 보조 비즈니스 계층

이 계층은 주요 비즈니스 계층에서 작성한 시스템의 주된 기능에 대해서 보조적인 기능을 모아 놓은 계층이다. 이것은 시스템의 외부적으로 보여지지 않고 내부적으로 특정한 알고리즘이나 기능을 가지고 있으면서, 여러 비즈니스에 연관되어 사용되어지는 모델이나, 컴포넌트를 표현하는 계층이다. 즉, 요구사항에서 요구하는 시스템에서 주요기능은 아니지만, 여러 부분에서 사용되는 기능들로 사용자 관련 계층이나, 주요 비즈니스 계층과는 다르게 사물의 형성을 실현화하여 정적인 생산물을 만들어 내는 표현 계층이다. 예를 들어, 데이터베이스에 저장을 수행하는 컴포넌트나, 암호화 및 복호화 등 여러 가지로 핵심 기능을 도와주는 보조기능을 가진 컴포넌트 등을 들 수 있다.

4.2 C_MDR 아키텍처 분석

가. 시스템의 행위분석

컴포넌트 메타데이터 레지스트리를 사용하는 행위자 관점에서 그 동작되어지는 행위를 (그림 4)에 나타내었다. 사용자는 크게 세 부류로 분리한다. 관리자는 컴포넌트 레지스트리를 효율적이며 체계적으로 관리하기 위한 행동(behavior)이고, 일반고객은 컴포넌트에 관한 정보제공의 동작을 수행하며, 그리고 개발자는 일반고객의 기능에 추가적인 기능을 수행한다. 본 시스템은 개발자 중심으로 관리, 저장 및 유통을 위한 도구로서 설계한 것이다[13].

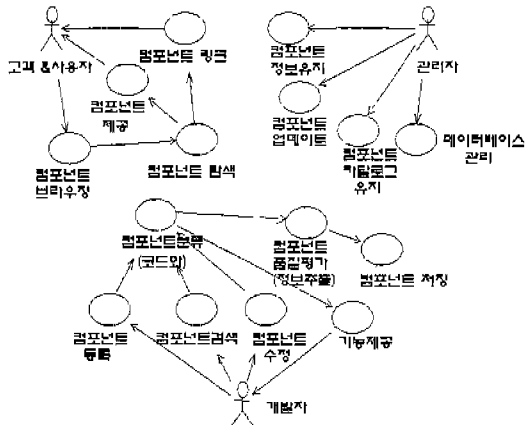


그림 4 MDR 시스템의 행위자별 동작과정

나. 시스템의 구조분석

(그림 4)에서 수행에 요구되는 기능 중심의 구조적인 관점에서 작은 컴포넌트들로 아키텍처를 설계한 것이 (그림 5)이다. 구조는 사용자 인터페이스 부분, 처리부분, 데이터베이스 연결부분으로 나누어진다.

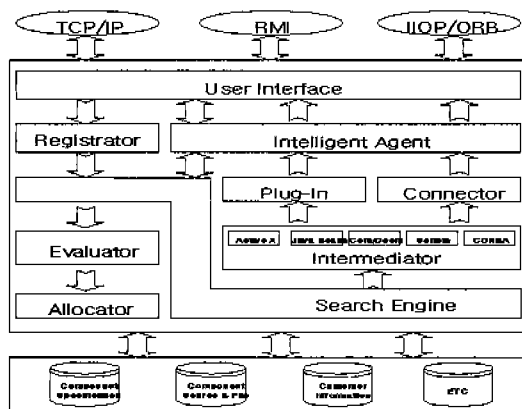


그림 5 MDR 시스템의 기본 아키텍처 구성

사용자 인터페이스는 외부 분산환경이나, 인터넷 환경에 접속을 가능하게 하는 부분이다.

컴포넌트 처리 부분은 실제적으로 컴포넌트를 다루는 부분으로 컴포넌트 레지스트리의 가장 핵심적인 기능을 수행한다. 이를 위해, 컴포넌트의 등록을 담당하는 등록기(registrator), 컴포넌트의 품질을 평가하여 코드화시키고, 분류해주는 평가기(evaluator), 추출된 정보를 데이터베이스와 연계해서 저장하는 기능을 갖는 할당기

(allocator), 시스템의 전 부분에서 사용되어지는 컴포넌트 검색을 지원하는 검색엔진(search engine), 그의 중간관리기, 지능형 에이전트 등으로 구성된다.

데이터베이스 연결부분은 컴포넌트 요소라는 컴포넌트 메타 단위를 이용하고, ODBC를 이용해서 컴포넌트 레지스트리와 데이터베이스의 연결을 한다.

다. XML과 MDR

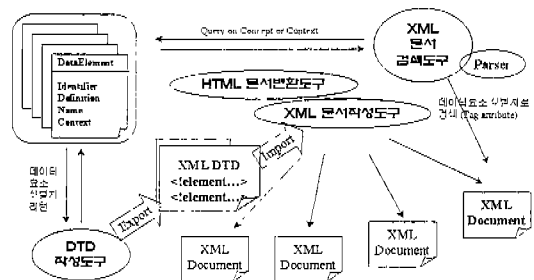


그림 6 XML과 MDR간의 처리과정

4.3 시스템 디자인

본 단원에서는 시스템 아키텍처 모형화를 기반으로 3계층 아키텍처 표현방식과 UML을 이용하여 시스템을 디자인한다[14][15]. 먼저, 쓰임새(use case) 다이어그램을 이용하여 기능 모델링하고, 클래스 다이어그램과 순차 다이어그램을 이용한 정적 및 동적에 의한 객체 모델링하며, 컴포넌트 다이어그램과 배치 다이어그램에 의한 컴포넌트 모델링 그리고 ER 다이어그램에 대해 기술한다.

가. 기능 모델링

C_MDR에서 사용하게 될 기능들에 대해서, 전체적 업무 및 흐름을 파악하기 위해, 개념적(conceptual)으로

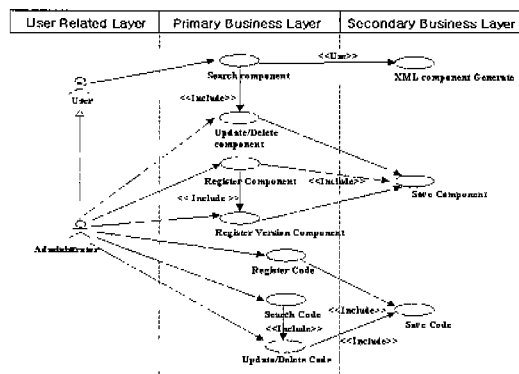


그림 7 쓰임새 모델

표 4 “Register Component” 쓰임새 명세

쓰임새 명	Register Component
개요(overview)	신규 등록을 요구하는 컴포넌트에 대한 메타 정보의 등록 기능을 처리하는 부분
사전요구사항 (pre-condition)	[P-1] 사용자 인증을 받은 사용자 [P-2] 대상 컴포넌트에 대한 일반정보와 상세정보의 구비 요구 [P-3] 동일한 컴포넌트명의 등록은 허용치 않으나, 동일한 컴포넌트의 다수의 버전 등록은 허용 [P-4] 가능한 선택적 입력토록 사용의 편리성을 제공
주요흐름 (main-flow)	[M-1] 관리자(운영자)는 컴포넌트 등록 메뉴를 선택한다. [M-2] 입력화면이 도시되면, 일반정보[S-1]로 명세정보와 형상정보를 등록하고, 상세정보의 등록을 원할 경우, 별도 윈도우로 상세정보[S-2]인 환경정보, 인터페이스정보 및 사용정보를 입력한다. [M-3] “저장” 버튼[S-3]을 통하여 DB 저장을 요구한다. (“save component” 쓰임새) [M-4] 상기 과정을 반복하여 계속적으로 등록을 수행한다
선택흐름 (select-flow)	[S-1] 일반정보 [E-1] -명세정보 : 아이디, 이름, 목적, 설명, 제품종류, 관련범주, 대중소분류, 제공결과물, 단가, 제작일, 등록일, 최종수정일 -형상정보 : 버전, 상태, 제사용기록, 개발언어, 개발불, 개발표준, 품질평가결과, 품질평가기관, 품질평가기준, 품질인증번호 [S-2] 상세정보 [E-1] -환경정보 : CPU타입, OS타입, 필요디스크량, 필요메모리량, 미들웨어, DBMS, 프로토콜, 저장타입, 파일크기, 이식 제한 -인터페이스정보 : 이름, 메소드, 타입, 초기값, 허용범위, 널여부 -사용정보 : 공급사명, 홈페이지, 이메일, 주소, 보안제한, 라이선스제한, 특기사항, 관계 컴포넌트 [S-3] 저장 : 입력된 5가지 정보를 DB에 저장한다. [S-4] 취소 : 초기 등록 화면으로 복귀한다.
예외상황 (exception situation)	[E-1] 입력 각 항목별 DB에서 정의하는 형식의 비 부합시와 물리적 DB 저장에러 발생시, 이에 적합한 오류 메시지를 발생한다.
참조기능 (reference function)	[R 1] 해당사항 없음
기타(besides)	해당사항 없음

필요할 것으로 예상되는 기능업무나 처리적인 기능들에 대해서 3계층 아키텍처 표현방식에 의해 쓰임새 다이어그램으로 모델화한 것이 (그림 7)에서 보여주고 있다.

(그림 7)의 쓰임새 모델에 대해, 각 기능별로 명세를 하는데, 사례로서 “Register Component” 쓰임새에 대해 나타낸 것이 <표 4>이다.

나. 객체 모델링

각 쓰임새별로 구축될 객체를 추출하기 위해 정적 및 동적 객체 모델링을 한다. 정적 모델링으로 클래스 다이어그램과 클래스 명세서(specification)를 작성한다. 동적 모델링으로 순차 다이어그램과 협력 다이어그램을 작성한다. 그리고 총괄 객체 모델링을 한다. 객체 분석의 예시로서, “Register component” 쓰임새에 대해 (그림 8)와 및 (그림 9)에서 보여준다[16].

여기서, 객체 모델링은 3계층 어플리케이션의 프로그램 방식으로 표현하기 위해, 객체 구성의 측면에서, 크게 엔티티 객체(entity objects, secondary business layer), 컨트롤 객체(control objects, primary business layer), 인터페이스 객체(interface objects, user related layer)의 3계층으로 나누어서 분석한다. 각각의 객체에 대한 설명은 다음과 같다.

* 엔티티 객체 : 특히, 관계형 데이터베이스상에서 데

이블이 되는 엔티티 데이터들을 의미한다.

* 컨트롤 객체 : 객체들 사이의 메시지에 대한 설명으로써 주로 인터페이스에서 사건이 발생하여, 이 사건이 관련된 객체 내의 메소드들을 활성화 시킴으로서, 어떤 사건의 흐름이 전개되어 질 수 있는 사건의 흐름과 방향을 정해주는 객체를 말한다.

* 인터페이스 객체 : 주로 사용자 인터페이스가 되는 화면이나 윈도우에서 작업선택 버튼이나 선택번호를 말한다.

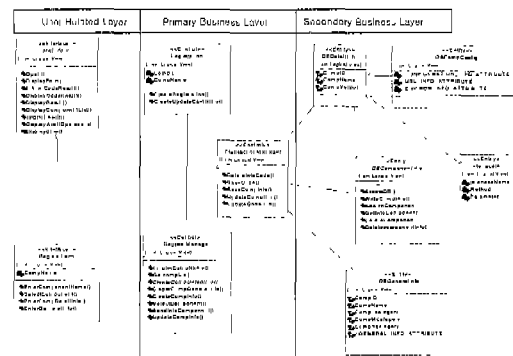


그림 8 “Register Component” 쓰임새에 대한 정적 객체 모델링

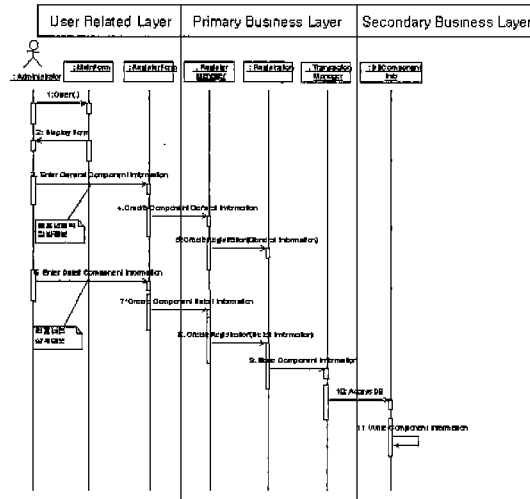


그림 9 "Register Component" 쓰임새의 동적 객체 모델링

각 쓰임새별로 작성된 클래스 다이어그램에 대해, 통합된 클래스 다이어그램을 (그림 10)에 나타내었다.

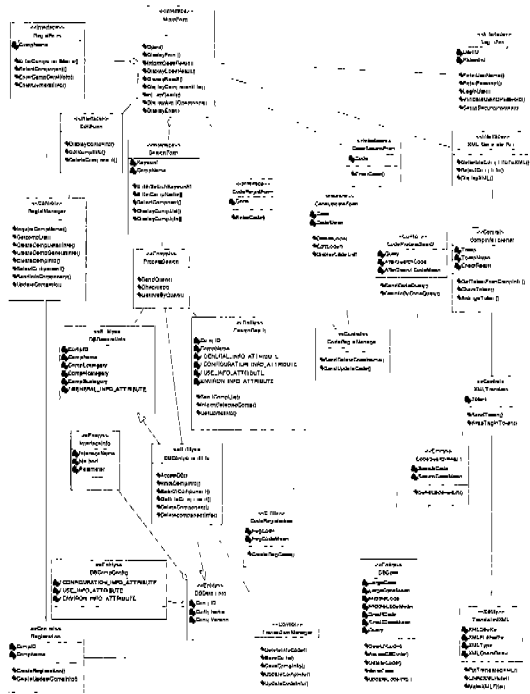


그림 10 통합 클래스 다이어그램

다. 컴포넌트 모델링

본 단원에서는 객체 분석을 통하여 추출된 객체에 대해서, 객체들을 그룹화하여 단위 독립성을 지니도록 컴포넌트화 한다. 역시, 3계층 표현방식의 아키텍처를 이용해서 컴포넌트 다이어그램을 작성한다. 쓰임새와 시나리오의 내용을 가지고, 가장 비슷한 레벨로서 컴포넌트 다이어그램을 나타낸 것이 (그림 11)이다[17].

배치 다이어그램은 시스템의 환경에 관련된 배치 상태를 나타내주는 배치 다이어그램을 3계층 표현방식을 이용해서 (그림 12)와 같이 나타내었다.

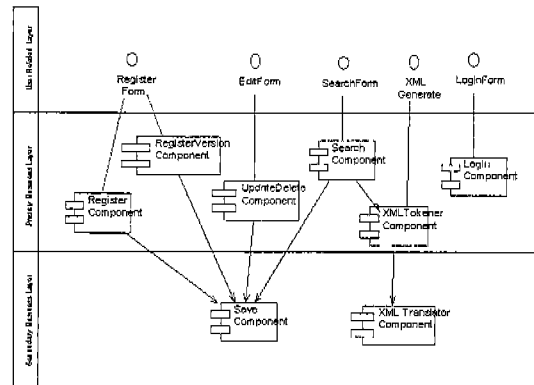


그림 11 C_MDR 시스템의 컴포넌트 다이어그램

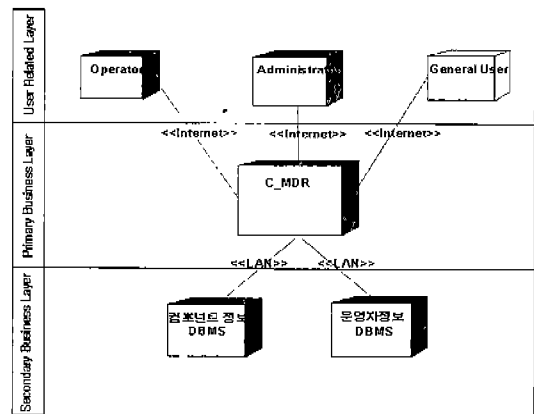


그림 12 C_MDR 시스템의 배치 다이어그램

라. DB 모델링

제3장에서 추출된 컴포넌트의 메타데이터에 대한 데이터베이스 스키마 설계도는 ER 표기법에 의해 (그림 13)에 나타내었다.

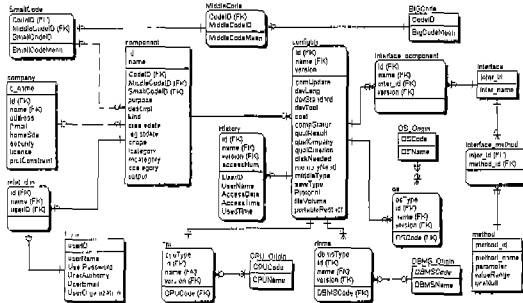


그림 13 데이터베이스에 적용된 ER 모델

4.4 시스템 구현

C_MDR에서 요구되는 서비스인 컴포넌트 메타정보관리, 분류코드 정보관리, 운용자 정보관리 및 게시판 정보관리에 대해 시스템을 구현하였다. 개발환경 및 구현언어는 (그림 14)와 같다. 개발언어는 ASP를 주로 사용하였으며, PC에서 손쉽게 이용할 수 있도록 제공하기 위해, 구동되는 운영체제는 Windows 2000과 Windows NT이며, DBMS는 RDB 형태를 지원하는 Oracle 8i를 사용했으며, C_MDR은 웹 상에서 PC용 웹서버를 통하여 시스템이 동작된다.

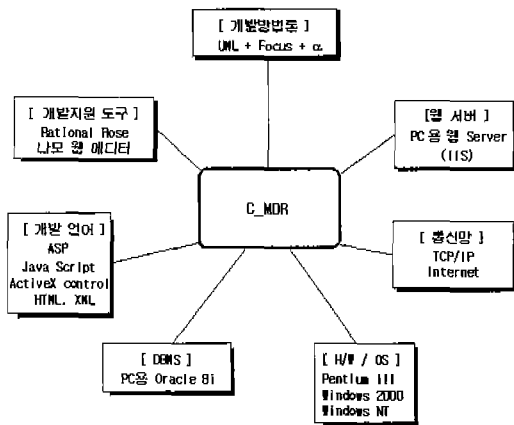


그림 14 C_MDR의 개발 및 운영환경

단원 4.3에 적용된 개발방법론은 UML, 전자통신연구원에서 제시된 Focus 방법론, 그리고 3계층 표현방법 등의 우리의 개발방법을 혼합하여 적용하였다.

시스템은 운용자용시스템과 일반사용자용시스템으로 구현한다. 운용자용은 모든 서버 시스템에 대한 등록, 변경 및 검색의 기능을 수행할 수 있으나, 일반사용자

은 제공 서비스에 대한 검색의 기능만을 갖는다. 이의 구분을 위해 URL 사이트를 개별적으로 주어서, 운용자용은 운용자 로그인 과정의 과정을 거친다.

각 서비스시스템별 구현 내역은 다음과 같다.

컴포넌트 메타정보 서비스의 경우, 신규 컴포넌트에 대한 분류코드체계에 의하여 메타정보를 등록하고, 변경하고, 그리고 검색하는 기능을 구현하였다. 컴포넌트 등록은 신규 컴포넌트 등록은 물론, 기존 컴포넌트에 대한 추가적 다른 버전 등록이 되도록 제공한다. 그리고 컴포넌트의 변경은 컴포넌트에 대한 버전별 수정과 삭제가 되도록 한다. 컴포넌트의 효과적인 검색이 되도록 다양한 검색 기능을 두었는데, 최신 컴포넌트, 일자별, 분류별, 그리고 주제어별 검색을 구현하였다.

사용방법은 운용자용시스템의 로그인 과정을 거치면, 좌측에 4개의 메인 서비스와 우측에 배경화면이 나타난다. 컴포넌트 메타정보서비스의 동작과정은 다음과 같다. 등록의 경우, 컴포넌트 메타정보에 대한 신규등록을 제공하는 “등록하기”와 기 등록된 컴포넌트의 버전별 신규등록을 제공하는 “버전추가”의 서버 메뉴를 제공한다. 컴포넌트 서비스의 “등록하기” 서버 메뉴를 선택하면, (그림 15)와 같이 컴포넌트 메타정보 등록화면이 나타난다. 먼저, 명세정보, 형상정보 및 환경정보를 입력하고, 저장하고, 필요시 하단의 상세정보의 등록을 위해 “상세등록” 버튼을 통해, 잔여 인터페이스정보와 사용정보를 입력 후, 저장의 과정을 거친다. 상세정보의 입력에서, 인터페이스의 경우, 하나의 인터페이스는 여러개의 메소드를 가지며, 메소드는 여러개의 메타정보를 입력토록 구현하였다. 환경정보의 경우, 컴포넌트의 OS별, CPU별 및 DBMS 등에 대해 기 등록된 정보의 경우, 재사용토록 선택, 입력하도록 하였다. 또한 컴포넌트 정보서비스의 “버

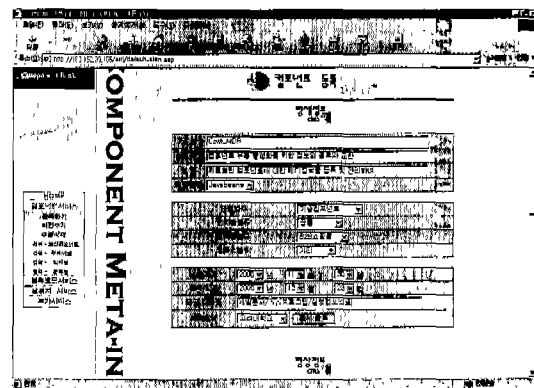


그림 15 C_MDR의 컴포넌트 메타정보 등록 화면

전추가"는 하나의 컴포넌트에는 여러 버전이 존재하기에, 기 등록된 명세정보는 그대로 두고, 버전별 상세정보를 신규등록하는 것으로서, 잔여 정보를 등록할 수 있도록 제공한다. 필수입력 항목에 대해서는 빨간글자색의 레이블로 되어 있어, 사용의 편의성을 제고하였다.

변경의 경우, 컴포넌트 서비스의 "수정/삭제" 서브메뉴를 통해 기 등록된 컴포넌트 메타정보에 대한 수정과 삭제를 한다. 수정은 대상 컴포넌트명에 의해 검색하여 상세정보의 내역이 도시되면, 등록과정과 유사하게 그 내용을 수정하게 된다. 검색의 경우, "최신컴포넌트", "검색-일자별", "검색-주제어별", 그리고 "검색-분류별"을 제공한다. 먼저, 검색서비스에 대해 검색조건을 입력하면, 1차 검색결과로 컴포넌트의 목록을 보여주는 약식 컴포넌트 목록(컴포넌트명, 컴포넌트설명, 등록일)이 도시된다. 여기서 도시된 컴포넌트 목록에서 원하는 컴포넌트 명칭을 클릭하면, 2차 검색결과로 상세정보에 해당하는 등록된 모든 메타정보가 도시된다. "최신컴포넌트" 검색은 최신 등록된 일자순에 의해 바로 1차 검색의 컴포넌트 목록을 보여준다. "검색-일자별"은 등록기간을 선택, 입력하면 1차 검색의 컴포넌트 목록을 보여준다. "검색-분류별"을 선택하면 (그림 16)과 같이 디스플레이 된다. 대.중.소의 분류에 대해 선택하면, 조합의 과정을 거쳐 검색결과가 도시된다. (그림 16)은 대분류와 중분류를 선택했을때, 1차 검색결과를 보여준다. "검색-주제어별"은 컴포넌트 명칭 또는 내용에 의해 검색할 수 있도록 제공한다.

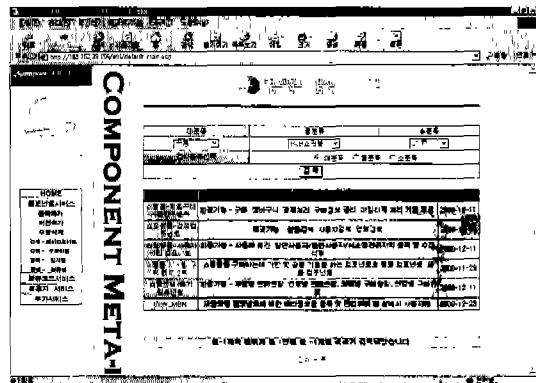


그림 16 C_MDR의 컴포넌트 메타정보 분류별 검색 화면

컴포넌트 분류코드 서비스의 경우, 가변적으로 확장해가는 컴포넌트 분류체제에 대한 분류코드의 등록, 변경 및 검색의 기능을 구현하였다. 분류코드는 전자통신

연구원에서 2000년 9월에 제시된 공용 컴포넌트 개발 관련 분류코드를 적용하였다. 분류코드의 식별성을 증진하기 위해, 분류별 선택시 ActiveX Control을 사용해 전체를 한눈에 볼 수 있도록 디렉토리 형태로 분류코드가 도시된다. 소분류코드 등록의 경우, 먼저 대분류 검색후, 다시 중분류 검색후, "소분류코드추가" 버튼을 선택하여 별도의 창을 띄워 소분류를 등록한다. 수정의 경우, (그림 17)과 같이, 해당 분류명을 선택하면 분류명, 분류 설명 및 분류 의미를 수정할 수 있다.

이하, 운용자 정보관리 및 부가서비스로서 게시판 정보관리의 기능을 구현하였다.

시험을 위해 각 부문에 대한 30개의 모의 컴포넌트 메타정보를 등록하여 시스템의 기능을 시험하였다.

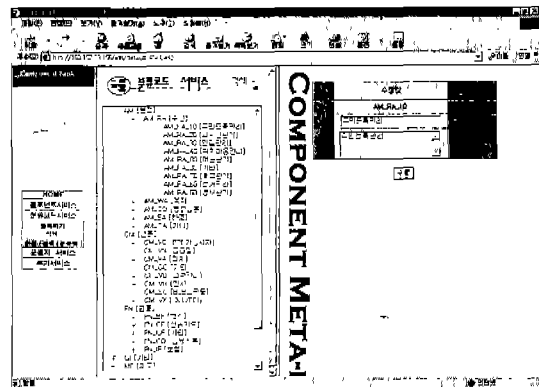


그림 17 C_MDR의 컴포넌트 분류코드 변경 화면

5. 비교 평가

지금까지, 컴포넌트에 대한 유통환경을 위한 지원도구로서, 정보의 표준성과 타 시스템과의 연동성을 갖는 컴포넌트 메타데이터 레지스트리(C_MDR)에 대해 메타정보를 정의하고, 3계층 아키텍처 표현방식에 의해 설계하고, 시스템을 개발하였다.

본 단원에서는 제시된 C_MDR에 대한 평가로서, 기존 상용시스템과의 비교, 데이터 레지스트리에 대해 정의하고 있는 ISO 11179와의 부합성, 본 시스템의 특징과 그 활용방안에 대해 기술한다.

가. 기존 상용시스템, CBOP과의 비교

대표적 컴포넌트 및 라이브러리 판매사와의 비교평가를 나타낸 것이 <표 5>이다.

일본의 CBOP은 비즈니스 컴포넌트의 공유를 목적으로 컴포넌트의 개발에서 유통에 이르는 인프라 개발을

표 5 기존 상용시스템과의 비교

구분	Component Source	ImagiCom	FlashLine	C_MDR
MDR 제공성	△	△	○	○
ISO 11179 준수성	△	△	△	○
메타정보 체계성	△	△	△	○(5개의 그룹화)
제공 정보량	△(제공자정보, 분류, 명칭, 벤더, 개발 볼 등)	△(컴포넌트 타입, 파일 타입, 제공자 등)	△(개요, review, 기능, 문제점, 목록, 판매정보등)	○(48개의 메타정보)
XML 제공성	X	X	X	○
기타 특징	메뉴방식 서비스 소스코드 조건부보관서비스 관련 기술 무료제공	인덱스 서비스 웹의 아이콘화	주문생산 방식 증계시스템 운영	표준 메타정보 타 시스템과의 연동

추진하고 있는데, 표준화된 비즈니스 객체 및 패턴의 개발에 역점을 두고 있다. 유통부문에서는 유통 및 사용의 활성화를 위한 하부 구조로서, 컴포넌트 배포, 등록, 평가, 검색 및 판매, 거래모형, 그리고 지침 및 교육 등에 대한 표준화 및 구축을 추진중이다. 정보공유 매체로서, 개발 CASE 정보저장소와 연계하여 자료 수집 및 관리를 지원하는 레포지토리와 패턴 라이브러리가 있다. 또한, 정보교환을 위해 XML을 사용한다. CBOP은 패턴과 객체 관점에서의 메타데이터에 초점을 두고 있으며, 근간에 ISO 11179를 고려하고 있으며, 추진중으로 그 실체를 볼 수 없다. 따라서, CBOP에 비해, C_MDR이 메타데이터 분류체계 및 정의, 그리고 지원도구 제공 측면에서 발전되었다고 할 수 있다.

나. ISO 11179와의 비교

먼저, 메타정보(메타요소)의 표현구조의 부합성 측면에서, C_MDR는 아래의 ISO 11179의 속성을 준수한다. 이 부분은 제3장 컴포넌트 메타정보의 속성 정의에서 보여주고 있다.

- 객체 클래스(object class) - 실 세계의 생각, 추상, 사물의 집합
- 프로퍼티(property) - 객체 부류내의 모든 구성원을 특징짓는 성격
- 표현(representation) - 값, 영역, 데이터 형, 측정 단위 등의 표현 형태

MDR의 기능과 구조적 측면에서, C_MDR는 명명파식별부분, 분류부분, 데이터 요소 관리부분, 데이터 요소 개념 관리부분, 개념 또는 값 영역 관리 부분에 대해 제 4장에서 그 기능을 수용하고 있다.

따라서, ISO 11179에서 제시한 표준화된 데이터 요소 추출과 정의, 그리고 데이터 레지스트리의 기능을 C_MDR에 구현했다. 이를 통해 컴포넌트 개발시, 생성 정보에 포함되어야 하는 데이터요소 등의 표준화를 유

도할 수 있다.

다. 컴포넌트 저장소와의 관계성 비교

컴포넌트 유통을 위한 정보제공 측면에서의 지원 도구로서는 컴포넌트 메타데이터 레지스트리(C_MDR)와 컴포넌트 저장소(C_R) 혹은 컴포넌트 라이브러리를 들 수 있다. 이 둘간은 목적, 용도 및 보유 정보수준 등에 따라 비교될 수 있다.

목적 측면에서, C_MDR은 여러 유관기관간의 필수적 정보의 유통에 있으나, C_R은 단일기관에서 제품의 판매 및 관리에 그 초점을 둔다. 따라서, C_MDR은 어느 정도의 공통적 표준성을 필요로 하나, C_R은 기관내 표준성을 갖고 있다. 용도 측면에서, C_MDR은 공용적 표준 메타정보 유통용 성격이 강하나, C_R은 기관내(유통업체별) 사내제품 판매용이라 볼 수 있다. 보유 정보수준 측면에서 볼 때, C_MDR은 카탈로그 정보(혹은 약식정보)를 가지는 컴포넌트의 실 데이터에 대한 기본적인 대표적이고, 필수적이고, 그리고 공통적인 데이터를 모아놓은 집합체이다. 이에 비해, C_R은 컴포넌트 제품에 대한 모든 정보를 갖는 실 데이터들 모아놓은 집합체이다. 그래서, 상세정보의 수준을 제공하는 것으로 디자인된 모델 및 명세 내역, 실행 컴포넌트, 소스, 판매 및 구매내역, 형상관리, 컴포넌트 다운로드, 그리고 유통 관련 모든 정보들을 관리한다.

이들 간에는 상호 연동성은 각 C_R에 저장된 정보로부터 C_MDR에 제시된 표준정보 플랫폼에 부합토록 메타정보를 추출하여 C_MDR의 정보를 구축하고, 사용자는 C_MDR을 통해 1차 컴포넌트에 대한 약식정보를 검색하고, 2차 상세정보 및 구매를 위해 C_R을 통해 얻을 수 있다. 따라서, C_MDR과 C_R은 다소의 데이터 중복을 갖고 있다. 이외, 검색 측면에서, 검색대상이 C_MDR은 표준적 메타데이터를 대상으로 하는 사전적 검색이나, C_R은 많은 제품정보에 대해 다양한 검색을 갖는다. 점

색기법 측면에서, C_MDR은 적은 데이터량 이기에 타 시스템과의 호환성 요구와 단순한 검색기법이 요구되나, C_R은 다양하고(디자인 모델, 바이너리 파일), 많은 정보량을 관리하기에 인덱싱 및 체계적 저장관리의 좀더 제고된 검색기법을 갖는다.

라. C_MDR의 특징

(1) ISO 11179의 MDR에 부합한 컴포넌트 메타데이터 레지스트리 구축

ISO 11179 표준에서 제시한 메타정보의 표현 및 기능을 준수함으로써, 이질적 시스템간의 원활한 상호 연동성을 제공하며, 데이터요소의 표준성과 유일성을 보장하고, 또한 데이터의 이해성과 신뢰성의 증대하였다.

(2) 컴포넌트 메타정보 속성 명세 플랫폼 제시 및 정의 컴포넌트 정보에 대해, 엔티티와 메타정보에 대한 속성을 정의해주는 플랫폼을 제시했다. 메타정보의 속성 명세서의 경우, 식별형, 정의형, 관계형, 표현형, 관리형의 5개의 그룹으로 분류하여 26개의 속성을 분류한다. 이 플랫폼에 따라 48개의 컴포넌트 메타정보를 정의하였다.

(3) 3계층 아키텍처 표현방식에 의한 디자인

3계층 표현방식에 의해, 기존의 모델링 방식보다, 일관성있고, 간단하고, 계층적이고, 상호 연관성을 비주얼하게 보여주고, 이해가 용이도록 하였다.

(4) XML 기반의 정보검색

XML이 전자상거래 및 EDI의 개선된 정보교환 포맷으로 적용이 예상되는데, 이를 데이터 통신방식으로 사용함으로써, 타 유통시스템과의 원활한 정보유통을 기할 수 있다.

마. C_MDR의 활용방안

컴포넌트 유통을 위한 인증된 컴포넌트의 표준정보제공 서비스 도구로 그 활용이 가능하다.(그림 18)은 컴포넌트 유통환경에서의 C_MDR의 역할을 보여주고 있다.

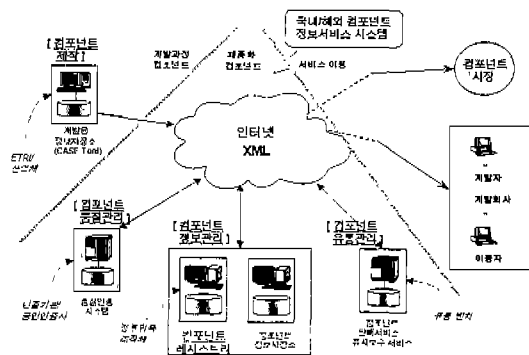


그림 18 컴포넌트 유통환경에서의 MDR 시스템의 위상

6. 결론

컴포넌트들을 조립하여 원하는 소프트웨어를 생산하는 체계가 소프트웨어 산업의 이상형이며, 꿈의 소프트웨어 자동화 실현에 다가설 수 있는 것이다. 이에, 업계에서는 각 사별로 컴포넌트의 상품화에 많은 노력을 기울이고 있다. 그러나, 컴포넌트 제품에 대한 정보관리 지원도구들은 그 제공정보의 서비스가 부실하며, 또한, 표준화되어 있지 못하다. 따라서, 컴포넌트 재사용의 활성화와 정보공유에 상당한 저해요소가 되고 있다. 이러한 해결책으로서, 본 연구에서는 컴포넌트의 활용 시대를 맞이하여 컴포넌트 기반 유통 및 관리 도구의 하나로 통합된 컴포넌트 공유환경 구축을 위해 사용하는 하나의 도구로서 컴포넌트의 메타정보를 웹 상에서 서비스 해주는 컴포넌트 메타데이터 레지스트리 시스템(C_MDR)을 디자인하고, 시스템을 구축하였다.

본 연구에서는 컴포넌트 메타정보의 중요성을 인식해, 정보기술 데이터요소의 명세와 표준화를 제시한 ISO 11179와 상용 컴포넌트 정보서비스 시스템들의 현황 분석을 기반으로, 메타데이터 성격의 표준화되고 캡슐화된 컴포넌트 데이터 요소를 추출하고, 이의 속성들을 정의하였다. 즉, 컴포넌트의 정보를 효과적으로 관리, 유통할 수 있는 데이터요소의 프레임워크를 표준화하여 제시하였다. 또한, C_MDR의 디자인을 위하여, 3계층 아키텍처 표현 방식에 의거하여, 일원화되게 아키텍처 모형화, 객체 및 컴포넌트 모델화를 하였다. 시스템 구축은 컴포넌트 정보 관리, 분류코드 정보관리 및 시스템관리를 구현하였다.

C_MDR 시스템의 구축을 통하여, 표준화되고, 내실화된 메타정보로 컴포넌트의 정보서비스를 제공할 수 있으며, 컴포넌트 재사용 개발의 지원도구로서 컴포넌트 정보공유는 물론, 국내외 타 시스템과의 연동을 원활히 할 수 있는 소프트웨어 정보유통체계의 구축에 일조할 것이라 사려된다.

향후 연구로서, 본 컴포넌트 메타정보서비스 시스템의 고도화 개발이 요구된다. 이를 위해, 시스템의 지속적 안정화 작업, 효과적 검색제공을 위한 시소러스 검색 제공, 기 저장된 메타정보의 분석처리를 통한 가공된 정보서비스 등의 작업이 필요하다. 또한, 컴포넌트의 공유와 교환에 관련되는 타 시스템 즉, 컴포넌트 정보저장소, 품질인증시스템, 판매시스템, 그리고 국내외의 관련 시스템과의 연동화 작업이 요구된다. 끝으로 컴포넌트 메타정보에 대한 범용적 C_MDR의 구축을 위해 분류별에 따라 계층적 체계화를 하고, 각 메타정보의 명확한 정의 그리고 국내외 표준화 추진에 대한 후속 작업이 요구된다.

참 고 문 헌

[1] CBOP OMG Joint Workshop, Hilton Tokyo Bay, May 20, 1999.

[2] EPA, "Environment Data Registry Introduction," <http://www.epa.gov/edr>

[3] Specification and standardization of data elements, "Framework for the Specification and Standardization of Data Element," ISO/IEC 11179 part 1 ~ 6, CD 1998.

[4] ComponentSource(<http://www.componentsource.com>)

[5] ImagiCom (<http://www.imagicom.com>)

[6] Flashline (<http://www.flashline.com>)

[7] CARDS(Central Archive for reusable Defence Software), "Software Architecture Seminar Report," STARS-VC-B008/001/00, January, 1994.

[8] Robert C. Scacord, "Software Engineering Component Repositories," ICSE 1999, 1999.

[9] 김승훈, 박대하, 나홍석, 백두권, "ISO 11179 기반 데이터 레지스트리에서 데이터 요소간 값 사상", 98 춘계 학술발표논문집(B), 한국정보과학회, 제26권 1호, p48-50, 1999.4.

[10] Paul Spencer, *XML Design and Implementation*, Wrox Press, 1999.

[11] W3C, "XML Schema," <http://www.w3.org/XML/Schema>

[12] 임성빈, 송치양, 문창주, 백두권, "3계층 표현 방식 아키텍처에서 UML 기반 컴포넌트를 이용한 시스템 모델링 기법", KISS2000(추계), 한국정보과학회, CD발간, 2000.

[13] Chee Yang Song, Sung Bin Yim, Chang Joo Moon, Doo Kwon Baik, "A Design and Implementation of a Component Registry Using XML," ISFST 2000.

[14] Grady Booch, James Rumbaugh, Ivar Jacobson, *The Unified Modeling Language User Guide*, Addison-Wesley, 1999.

[15] Philippe Kruchten, "Modeling Component Systems with the Unified Modeling Language," ICSE 1998, 1998.

[16] Desmond F. D'souza and Alan C. Wills, *Objects, Components, and Frameworks with UML*, p91-234, Addison-Wesley, 1998.

[17] Family-Oriented Component System Methodology, 김수동, 숭실대학교, 1999.



송 치 양

1985년 한남대학교 전산학과 학사. 1987년 중앙대학교 전산학과 석사. 1990년 ~ 현재 한국통신 연구개발본부 선임보 연구원. 1999년 ~ 현재 고려대학교 컴퓨터학과 박사과정. 관심분야는 UML 모델링 기법, 메타모델, 정형명세 및 분석, 메타데이터 레지스트리



임 성 빈

1999년 숭실대학교 컴퓨터학부 학사. 2001년 고려대학교 컴퓨터학과 석사. 2001년 5월 ~ 현재 한국유니시스 IS 컨설턴트. 관심분야는 S/W공학, CBD, OOAD, UML, CASE, 컴포넌트 저장소



백 두 권

1970년 ~ 1973년 고려대학교 수학과. 1974년 ~ 1976년 고려대학교 산업공학과(석사). 1981년 ~ 1983년 Wayne State Univ. 전산학 석사. 1984년 ~ 1985년 Wayne State Univ. 전산학 박사. 1986년 ~ 현재 고려대학교 컴퓨터학과 교수. 1989년 ~ 현재 한국정보과학회 평의원/이사. 1991년 ~ 현재 ISO/IEC JTC1/SC32 국내위원회 위원장. 관심분야는 소프트웨어 공학, 데이터 공학, 메타데이터, 컴포넌트 시스템, 데이터베이스



김 철 홍

1981년 충남대학교 문리과대 학사. 1993년 성균관대학교 정보공학과 석사. 1983년 ~ 현재 ETRI 컴퓨터소프트웨어기술연구소 S/W공학연구부 선임연구원. 관심분야는 컴포넌트 기반 개발방법, 소프트웨어 테스트, 재사용