

건설 프로젝트 생애주기 관리를 위한 정보교환 서비스시스템 개발방안

Development strategy for an information exchange subsystem as a part of
Construction Project Lifecycle Management System(CPLMS)

유 석 준*
You, Seok-Joon

요 약

본 연구는 건설 생애주기 관리시스템(CPLMS)에서의 데이터 교환기능의 구현에 관한 것으로 국제표준 건물정보모델(BIM)인 IFC에 기반하여 국내실정에 알맞게 개발중인 K-IFC의 기능 확장을 통해 이를 실현하고자 한다. 건설산업에서 사용중인 다양한 파일형식을 지원하기 위해 CPLM의 데이터 저장기능은 이들 파일을 입출력할 수 있어야 한다. 그러나 파일을 통한 PLM의 기능구현에는 한계가 있기 때문에 이를 BIM 개체에 기반한 방식으로 전환할 필요가 있으며, 이를 위한 단계적 개발방안이 제시되었다. 이를 통해 CPLM정보의 보안성과 무결성을 보장할 뿐 아니라 프로젝트 참여자 사이의 커뮤니케이션을 원활케 하여 CPLM 시스템의 전체적 효율성을 높일 수 있을 것으로 기대된다.

키워드: PLM, CPLM, IFC, BIM, 정보교환

1. 서 론

본 연구는 2007년부터 5년 예정으로 진행중인 건설 PLM 시스템 개발연구의 일부로 PLM시스템(Product Lifecycle Management System)이 관리하는 디지털화 된 BIM(Building Information Model) 기반 건물정보를 프로젝트 참여자 사이에서 원활히 교환할 수 있도록 하기 위한 서비스시스템을 개발하는 것을 목적으로 하고 있다. PLM은 건설산업 프로젝트의 기획단계에서부터 설계, 설계평가, 입찰 및 계약, 시공, 유지관리 및 해체에 이르는 전 단계에 걸쳐 대상물(건축물 또는 토목구조물)의 설계정보 및 프로젝트 수행 중 발생하는 모든 정보를 모든 참여자들이 공유토록 함으로써 이들 사이의 커뮤니케이션을 활성화 하고 이 과정에서 생길 수 있는 문제점을 조기에 파악하여 대처할 수 있게 해준다. 따라서 이의 핵심은 관련 정보를 효율적으로 공유하면서도 공유중인 정보의 무결성을 확보할 수 있게 해주는 정보교환체계라고 할 수 있다. 본 논문은 이 같은 정보교환 시스템의 목표를 충족하기 위해 고려해야 할 사항과 적용 가능한 구현 방식, 또한 앞으로 개발될 시스템의 구현방안에 대해 알아볼 것이다.

2. 건설 프로젝트 관련정보의 형태와 교환방식

PLM에서 관리하는 정보는 기본적으로 컴퓨터 소프트웨어에서 사용할 수 있는 디지털화 된 정보에 한정된다. 여기에는 CAD 시스템이나 기타 엔지니어링 소프트웨어에서 작성, 사용되는 건물설계에 대한 정보, 시공을 위한 공정 계획 및 관리를 위한 정보, 견적, 예산 및 지출 등에 관한 재무정보, 인허가 관련 서류, 사진, 비디오 등의 멀티미디어 정보 등 방대한 종류의 정보가 포함된다.

별도의 관리시스템이 없는 경우 프로젝트에서 사용되는 이들 파일의 체계적인 관리는 매우 어렵다. 때문에 파일을 손실하거나 엉뚱한 버전의 파일이 다른 파트너에게 전달되어 설계나 시공과정에서 심각한 문제가 발생하는 등으로 프로젝트 수행의 진행이 방해되는 경우가 다수 발생한다. 이를 관리하기 위한 CAD 도면관리 시스템 등이 다수 시판, 사용되고 있으나 이들 초보적 시스템이 관리할 수 있는 데이터의 종류는 아직은 일부 도면과 관련 문서파일에 머무르고 있는 실정이다.

이들 다양한 관련정보사이의 호환성을 확보하기 위해 이를 모두 포함할 수 있는 하나의 정보모델을 정의하고자 하는 시도가 계속되었고 IFC(Industry Foundation Classes)가 이를 가능케 하는 유력한 후보로 부상하고 있으며, 한편 건설업 워크플로우 전반에 걸쳐 관련 소프트웨어를 보유하고 있는 대형 개발사 중심으로 자체적인 정보모델을 역시 BIM이란 이름으로 홍보하고 있다. 그렇지만, BIM을

* 비회원, 조지아공대 건축대학 박사과정 (ABD)

전체 프로젝트 생애주기 단위에서 관리할 수 있는 시스템의 출현은 아직 요원한 것으로 보인다. 따라서 현재 사용중인 파일 형태의 정보라도 관리할 수 있는 시스템 소요 역시 제기되고 있다.

3. BIM 개체의 저장방식

3.1 파일시스템을 통한 파일단위의 BIM 저장

개별 소프트웨어에서 출력한 파일형태의 정보모델을 컴퓨터의 저장장치에 직접 저장하는 방법 (실제로는 컴퓨터의 운영체계에 의해 관리되는 파일 시스템을 통한)이다. 효율적인 관리를 위해서 프로젝트에 참여하는 다수의 구성원은 하나의 파일 서버를 공유하게 하고 (물리적으로는 다수의 서버에 분산될 수 있음) 서버로의 접근을 통제함으로써 해당 프로젝트에서 생성, 변경, 또는 참조하는 정보를 관리하게 된다. 다양한 형식의 개별 파일 내부에 관리에 필요한 정보를 기록하기는 어렵기 때문에 이들 파일에 관계되는 관련정보를 메타데이터의 형태로 기록해야 한다. 근래 사용되는 운영체제와 파일시스템은 대부분 파일과 관련된 메타데이터를 별도로 기록할 수 있지만 PLM에 이를 사용하기에는 부족하다. 과거 출시된 대부분의 상용 PDM 시스템은 메타데이터를 기록하기 위한 별도의 데이터베이스를 자체적으로 관리하는 방법으로 이 문제를 해결한다. 이 방식의 장점으로서는 시스템 부하가 적기 때문에 처리속도가 빠르다. 최신의 운영체제와 파일시스템은 상당한 수준의 데이터 안전성과 보안성을 갖고 있다.

3.2 BLOB (Binary Large Objects)을 통해 파일단위 BIM을 데이터베이스에 저장

BLOB은 데이터베이스 시스템의 페이지 크기를 초월하는 (> 4KB) 큰 사이즈의 비정형(불특정 양식과 크기)데이터를 수용하는 특수한 데이터 타입으로 데이터베이스에 사진, 음성, 동영상 등의 정보를 저장하는데 쓰인다. 이를 파일형태의 BIM 또는 다른 형식의 파일을 데이터베이스 시스템에 저장하는데 사용할 수 있다. 개별 BLOB은 다른 형태의 데이터에 비해 데이터베이스에서 다루는데 더욱 부하가 걸리기는 하지만 한번에 처리해야 하는 개체의 수를 줄일 수 있기 때문에 실제 처리 속도는 개체단위로 저장하는 것 보다는 빠르고 파일시스템 보다는 느리다. BLOB을 이용한 방식은 파일시스템과 마찬가지로 개체 단위의 저장방식에 비해 장된 정보에 대한 세밀한 관리가 어렵다는 등의 단점을 공유하지만 데이터베이스 시스템의 탁월한 안정성과 보안성을 활용할 수 있으며 파일시스템 위주의 저장체계를 개체 위주의 데이터베이스로 전환하기 위한 징검다리 역할을 할 수 있다.

3.3 BIM 개체를 직접 데이터베이스에 저장

다수의 개체가 하나의 저장단위에 기록되는 파일과 달리 데이터베이스 시스템(DBMS)의 레코드(record, 하나의 유

의미한 개체entity 또는 테이블의 한 열(列)을 형성하는 일련의 데이터 항목field이 저장되는 공간)에 개별 BIM 개체를 기록하는 방법이다.

제품별 차이가 있기는 하지만, 다수의 데이터베이스 시스템에서 쓰이는 정보 정의사상인 관계형 데이터 모델(relational data model)은 BIM을 정의할 때 쓰이는 모델링 방식(다수의 방식이 있으나 주로 Entity-Relationship model, object-oriented model과 여기에서 파생된 방식)과 바로 호환되지는 않는다. 때문에 이들을 중개해 주기 위한 매핑 규칙이 정의되어야 한다.

이 방식은 데이터베이스 시스템의 기능을 활용할 수 있다는 장점이 있다. 개별 개체의 정보를 참조 또는 변경하기 위해 전체 파일을 불러오지 않아도 개체의 속성에 바로 접근해 변경할 수 있으며 다양한 질의기능을 활용하여 필요한 개체를 쉽게 검색할 수 있다. 이를 통해 정보모델의 세세한 수준까지 관리할 수 있다

프로젝트 구성원 사이에 정보모델(i.e. BIM)이 공유되는 경우 이를 수정하려 할 때 파일은 한 구성원의 세션이 해당 모델파일을 독점적으로 점유해야 하지만 데이터베이스에 저장된 모델은 다수의 구성원이 담당하는 부분에 따라 공유모델의 일부분만을 점유하여 동시에 수정하여 시간을 절약할 수 있다. 파일 역시 복수의 사본을 만들어 여러 구성원이 동시에 작업할 수 있기는 하다. 그러나 이들 변경된 사본 사이의 내용이 불일치할 경우 (예를 들면 두 개 이상의 사본이 동일한 개체를 수정했을 경우) 이를 보정하기 어려운 반면 데이터베이스 시스템은 하나의 데이터 모델 사본을 다수의 사용자가 공유하는 형식을 취하기 때문에 동일한 개체가 둘 이상의 사용자에게 의해 변경되지 못하도록 한 사용자에게만 해당 개체에 대한 독점권을 주거나(locking) 관련 사용자에게 이를 알려 문제가 발생하지 않도록 하는 등의 관리가 가능하다. 이러한 기능은 PLM 시스템이 지원해야 하는 주요 기능이기도 하다.

이 방식의 문제점으로, 정보 모델을 데이터베이스 시스템이 수용할 수 있는 형태 (다수의 시스템이 관계형 데이터 모델에 기초함)로 매핑할 때 정보 모델의 의미(semantics)가 변형 또는 손상될 수 있다. 파일시스템을 이용하는 경우에는 네이티브 파일로 저장할 경우 매핑문제가 발생하지 않는다. 이종 소프트웨어간 네이티브 형식의 데이터를 교환할 때에는 한번의 매핑만으로 충분하다 (물론 개별 소프트웨어별로 각기 매핑을 해줘야 하기 때문에 전체 매핑의 수는 매우 커진다). 반면 이종 소프트웨어 사이의 매핑의 경우 설령 중간포맷을 사용하더라도 두 번의 매핑이 필요하기 때문에 정보의 오류 가능성이 그만큼 증대하게 된다.

이 방식은 외부에서 개발한 상용 데이터베이스 시스템을 사용하게 된다. 시중에서 구할 수 있는 데이터베이스 시스템들은 비록 동일한 데이터베이스 언어인 SQL(Structured

Query Language)을 대부분 지원하긴 하지만 각자 수용할 수 있는 데이터의 형식과 길이와 같은 세부적인 부분에 적지않은 차이가 있다. 때문에 한 시스템에 적용된 매핑을 다른 시스템에 사용하려면 상당부분 수정되어야 한다.

이 방식의 단점중 하나는 파일시스템 기반 또는 데이터베이스 시스템의 BLOB을 이용한 파일 단위의 관리방식에 비해 처리속도가 느리다는 것이다. 전형적인 BIM개체는 상대적으로 작은 크기의 수치 데이터 또는 단문 데이터의 집합으로 되어 있으나 (개별 개체의 크기는 보통 수 바이트 ~ 수백 바이트) 전체 모델은 매우 많은 수의 개체가 복잡한 참조관계로 얽혀있다. 예를 들어 3층 이내의 주택 건물의 IFC 모델이라 하더라도 STEP 클리어 텍스트 인코딩 파일 (.ifc 형식) 기준으로 7MB 정도의 크기를 가지며 개체수는 10만개 이상에 이른다. (You et. Al, 2004) 의 매핑 방법을 적용하였을 경우 전체 데이터베이스 레코드 수는 40만개 가량이 되어 이를 기록하기 위해서는 RAID구성의 데이터베이스 서버라 하더라도 상당한 시간이 소요된다 (4개의 400MHz 프로세서와 5 디스크의 Level 4 RAID 구성의 MS SQL Server 데이터베이스 시스템에서 약 7분 정도).

4. 개체단위 저장을 위한 데이터 모델의 선택

4.1 중립적 모델의 이용

IFC와 같이 특정 소프트웨어에 종속되지 않으면서 프로젝트에 필요한 모든 정보를 기록할 수 있는 정보모델을 선택하는 것이다. CPLM에 간여하는 이해주체(예: 프로젝트 주계약자, 설계자, PLM시스템 공급자, 개별 응용 소프트웨어 개발자 등)에 따라 BIM의 정의에 대한 관점에는 차이가 있긴 하지만 중립적 정보모델이야말로 BIM의 원래 취지에 가장 부합하는 것으로 볼 수 있다. 그러나 앞서 지적되었듯이 어떤 소프트웨어도 중립적 모델을 내부적 정보모델로 사용하지 않기 때문에 데이터 매핑에 따른 문제점(예를 들면 정보의 손실 또는 왜곡 가능성)이 항상 존재한다. 한편 Bentley社와 같이 자사의 소프트웨어 제품 라인업에 동일한 정보모델을 적용함으로써 이를 해결하려는 시도가 알려져있다. 그러나 이 경우 타사 제품의 데이터 모델을 수용하지 어렵기 때문에 (이유: 중립적 모델과 달리 이들 내부적 정보모델에 대한 정보가 외부에 공개되어 있지 않기때문) 적용 가능한 소프트웨어의 수에 제약이 있게 되고 다른 소프트웨어를 사용하는 타 프로젝트 참가자와의 협업이 어려워 질 수도 있다.

4.2 응용 프로그램별 복수의 모델 이용

이 방식은 정보 모델이 아닌 정보관리 시스템 (또는 데이터베이스)에 데이터를 정의, 기록, 검색및 입출력하는 방법을 표준화 하여 각각의 응용 소프트웨어가 자사의 정보

모델을 직접 이 표준을 구현케 하는 방법이다. 즉, 데이터베이스 내부에 다수의 정보모델 형식(스키마)이 존재하게 되며 동일한 설계대상물이 여러 개의 모델에 기록되어 있을 수 있다. 이렇게 함으로써 개별 소프트웨어에서 정보관리 시스템으로 정보를 전달할 때 생기는 문제 일부를 해결할 수 있다. 그러나, 파일 기반의 관리시스템과 마찬가지로 정보 교환 시 소프트웨어 별로 다수의 매핑방법을 정의해야 하는 문제가 있다. 그렇지만 업무 프로세스에 대한 분석을 통해 정보 교환이 필요한 업무 및 사용되는 소프트웨어를 식별함으로써 필요한 매핑의 수를 최소화 함으로써 이 문제를 최소화 할 수 있다. 개별 소프트웨어에서 정보저장 및 교환 시스템으로의 정보 왜곡문제를 줄일 수 있고 BIM 개체와 마찬가지로 데이터베이스 시스템을 통해 파일시스템 대비 수준높은 정보 관리가 가능해진다

5. 정보저장 및 교환 시스템의 구현방향

생애주기 관리시스템은 다수의 소프트웨어 통합은 물론 업무 프로세스의 재구축을 포함하는 대형 시스템으로 설계부터 제작, 설치(deployment) 및 보완에 이르기까지 길게는 수년의 시간이 소요된다. 새로운 시스템으로의 적용을 쉽게 하고 설치 후 보완 과정을 용이하게 하기위해 순차적인 시스템 개발이 바람직 할 수 있다. 정보의 저장 및 교환을 위한 서브시스템의 경우 앞서 다뤄진 구현방법 을 순차적으로 적용하는 방법이 이에 해당한다. 즉, 상대적으로 구현하기 용이한 파일시스템 방식을 먼저 구현하여 데이터베이스 시스템의 적용범위를 순차적으로 넓혀가는 방안을 생각할 수 있다.

5.1 파일시스템을 이용한 초기 시스템 구현방안

파일에 대한 메타데이터를 상용 데이터베이스 시스템에 저장함으로써 이후 시스템 개발과의 연계성을 갖게 할 수 있다. 메타데이터베이스를 확장함으로써 후속 시스템에 필요한 정보를 수용할 수 있어 개발에 필요한 자원을 절약할 수 있다. 이를 위해 메타데이터에 대한 구조 (데이터 스키마)를 정의해야 하며 데이터의 접근과 사용에 대한 권한체계를 정의해야 한다. 사용자에 대한 정보가 여기에 해당하는데 이를 동일한 데이터베이스 시스템에 저장함으로써 후속개발에 있어서 이들 정보의 마이그레이션 필요를 최소화 할 수 있다.

IFC와 같은 중립적 포맷을 사용할 경우 이들 포맷으로의 변환은 개별 응용프로그램에서 담당케 해야 하지만 PLM 시스템에서 변환된 파일의 입출력에 대한 로그(import/export logs)를 관리해야 한다.

5.2 개체레벨 시스템으로의 전이

전술한 메타데이터 데이터베이스에 BLOB 필드를 적용하여 BIM 파일을 저장한다. 후행호환성(backward

compatibility)을 확보하기 위해 IFC등의 중간 포맷뿐 아니라 응용 프로그램의 파일형식을 지원하는 것이 좋으나 중간 포맷으로의 전환을 고려하여야 한다. 전체 데이터를 개체단위로 저장할 때에도 변환에러에 의한 데이터손실을 최소화 하는 차원에서 응용 프로그램별 네이티브 데이터를 같이 저장하는 것이 데이터 안전을 확면에서 바람직하다.

6. 건설프로젝트 생애주기 관리를 위한 정보교환 프레임워크

K-IFC (International Foundation Classes ? Korean Localization and Extensions)은 중간포맷의 BIM 표준이라 할 수 있는 IFC모델을 건설 PLM시스템에서 활용하기 위해 개발 중인 정보 저장 및 교환 서브시스템이다. K-IFC에는 기존 IFC 정보모델 정의와 속성정의 (property sets)에 더해 건설 PLM시스템에서 부여하는 개체 식별자의 적용, 독자적인 전송프로토콜 및 IDM(Information Delivery Manual)에 기반한 소프트웨어의 호환성 확인 장치 (application conformance checker), 산업체 적용을 위한 테스트케이스와 표준사례(best practices)가 포함될 예정이다.

6.1 K-SDAI (Standard Data Access Interface)

K-SDAI는 기존 SDAI 인터페이스에 더해 건설 PLM 시스템에서 관리하는 각종 정보에 접근할 수 있도록 도와주는 역할을 하게 된다. K-SDAI는 서버에 저장된 BIM 정보에 대한 개체단위 검색 및 입출력을 가능케 하며, 설계대상물에 부여된 고유 식별자를 통해 IFC 이외의 다른 정보 모델에 저장된 개체 또한 사용할 수 있도록 할 계획이다. 또한 응용소프트웨어 개발자가 쉽게 자신의 정보모델을 건설 PLM 데이터베이스에 저장할 수 있도록 단순명료한 API (Application Programming Interface)를 제공하는 것을 목표로 하고 있다.

6.2 IDM을 응용한 호환성 확인장치

IDM은 IFC를 개발한 국제단체인 IAI(International Alliance for Interoperability)에서 연구 중인 소프트웨어 호환성 보조체계로 각 응용소프트웨어가 지원하는 업무프로세스와 데이터 내용을 IFC 스키마의 서브세트를 포함한 매뉴얼(설명서)의 형태로 서술토록 하고 있다. IDM은 컴퓨터가 인식할 수 있는 언어체계는 아니지만 K-IFC에서는 응용 소프트웨어 사이에 교환 가능한 정보를 자동으로 식별하는데 사용할 수 있도록 매뉴얼 형식을 개선, 확장하고 개선된 IDM을 입력받아 처리할 수 있는 호환성 확인장치 (conformance checker) 및 이들 호환 소프트웨어가 공유하는 데이터 모델을 IFC 서브셋 형식으로 출력하는 소프트웨어 (IFC subset extractor)를 개발 중이다.

6.3 데이터 교환 프로토콜(전송규약)의 개발방안

개체단위의 정보교환이 이루어지더라도 PLM서버와 소프트웨어 사이의 정보교환은 여러 개체를 한 묶음으로 처리하는 벌크방식으로 이루어진다. 이는 현존 소프트웨어가 개별 개체단위의 접근을 지원하지 않기 때문이기도 하고 데이터베이스 접근 시 필연적으로 발생하는 오버헤드(데이터베이스 내의 개체의 안전과 무결성을 확보하기 위해 취하는 일련의 조치)를 이에 대한 접근회수를 줄임으로써 최소화 할 수 있기 때문이기도 하다. 이러한 프로토콜은 다음과 같은 사항을 반영하여야 한다.

- * 입출력되는 개체의 불변성 식별자 및 사용자에 대한 정보를 포함하여야 한다. 여기에는 사용자 정보뿐 아니라 타임스탬프 등의 일부 메타데이터가 함께 전송되어야 한다.

- * 전송된 정보의 무결성을 확인할 수 있어야 한다. TCP/IP와 같은 하위 레벨 네트워크 프로토콜이 이 부분을 담당할 수 있다. 그러나 전송중인 데이터가 중간에 급작스럽게 끊어질 경우 전송된 데이터 중 어떤 부분이 온전한지 PLM시스템에서 확인할 수 있어야 하며 이를 위한 요소가 프로토콜에 포함되어야 한다.

- * 전송주인 정보에 대한 보안이 확보되어야 한다. 이 부분 또한 네트워크 프로토콜과 운영체제에서 지원하므로 프로토콜에서는 이를 반영해주기만 하면 된다. 다만 암호화 등이 포함될 경우 일부 요소를 프로토콜에서 직접 구현해야 할 수도 있다.

이러한 추가적인 요구사항 때문에 기존의 파일 인코딩 형식인 STEP Part 21 (clear text encoding)이나 IFCXML 등은 전송프로토콜로는 적합하지 않다. PLM을 위한 전용의 프로토콜이 정의된 사례는 알려진 바 없으나 XML 표준의 하나인 SOAP (Single Object Access Protocol) 을 응용한 사례로 SABLE이 있다. 다만 SABLE의 전송형식은 앞서 서술한 필요사항을 충족시키기에는 조금 미흡한 편이다.

6.4 고유식별자(invariant unique part identifier) 정의방안

구조부재와 창호, 마감 등의 설계대상체는 설계 시공의 프로세스를 거치면서 다수의 소프트웨어에서 동일한 개체에 대한 해당 정보를 공유하게 된다. 단일의 중간적 정보 모델을 사용하지 않을 경우 복수의 소프트웨어에서 사용하는 이질적 정보모델 사이의 공유가 실현되려면 이들 동일 개체를 식별할 수 있는 고유 코드가 정보모델 사이에서 공유되어야 한다. 예컨대, 기둥과 보와 같은 건축물의 구조체는 건축설계 소프트웨어에서는 박스와 원통이라는 3차원적 지오메트리의 집합으로, 구조해석 소프트웨어에서는 선과 절

점의 집합으로, 구조설계 소프트웨어에서는 형태에 더해 내부 배근의 레이아웃으로 정보모델의 형태에 차이가 있다. 이들이 동일한 구조체를 가리키고 있다는 것을 알려주려면 각 구조체별 고유코드가 정의되어 세가지 소프트웨어 모두 이를 변형 없이 입력, 저장 및 출력할 수 있어야 한다. 여기에는 다음과 같은 점이 고려되어야 한다.

* 식별자가 반드시 유일무이할 필요는 없다. IT업계에서 사용되는 GUID(globally unique identifier)의 경우 이론적으로 중복 식별자가 사실상 존재하지 않기는 하지만 유일무이성을 보장하기 위해서는 필연적인 데이터의 크기로 말미암아 (128바이트 = 영문 16글자에 해당) 전체 시스템의 처리속도를 저하시킬 수 있다. 또한 GUID는 일련번호가 아니기 때문에 색인 작업등이 매우 어렵다.

* 응용 소프트웨어에서 PLM 시스템으로부터 개체의 고유식별자를 전달받아서 정보를 다시 시스템에 되돌려 줄때 식별자 또한 함께 반환할 수 있어야한다. 대부분의 소프트웨어는 IFC의 GUID와 같은 식별자를 내부에서 보관하지 않는다. 다만 구조설계 등 일부 소프트웨어에서 부품코드(piece mark)의 보관 및 입출력이 가능하다

* 위와 같은 이유는 IFC에 사용되는 개체식별자(IfcRoot.id)는 이런 목적으로 사용하기에는 부적당하다. 대신 개체 속성 등에 정의된 부품 코드(piece mark)에 일련번호를 더하는 형태, 또는 property set (PSET)에 일련번호 속성을 추가하는 형태 등을 고려할 수 있다.

7. 결론

건설산업에서 사용되는 다양한 정보형식과 상대적으로 뒤늦은 IT기술 적용, 또한 새로운 시스템의 적용 및 보완에 소요되는 비용 및 교육에 대한 문제로 인해 건설 PLM 시스템을 처음부터 BIM개체기반으로 개발하기에는 어려움이 따른다. 이 같은 어려움을 극복하고 PLM의 이점을 현장에 빨리 적용하기 위해 본 연구에서는 일부 정보 저장부분을 포함한 정보 교환 서브시스템의 순차적인 구현방안을 계획하고 있다. 그러나 이것이 개체기반 데이터베이스 방식의 잠재성을 부정하는 것은 아니며, 이러한 순차적인 구현방안을 통해 당 프로젝트 후기에는 개체수준 정보의 고도의 관리를 가능케 할 것으로 기대된다.

참고문헌

1. 가상건설구축프로젝트 사업단, 가상건설 프로젝트 1차년도 연구보고서 (2007)
2. S You, D Yang and C Eastman. "Relational database implementation of STEP-based product models". CIB World Building Congress 2004. Toronto, Canada. 2004.5.1-7
3. C Eastman, F Wang, S You and D Yang.

"Deployment of an AEC industry sector product model". Computer-Aided Design 37 (2005) pp.1214-1228

4. Wicks. The Information Delivery Manual (2007).

<http://idm.buildingsmart.no/confluence/display/IDM/Home>
Liebich, T. IFC 2x2 Model Implementation Guide (2004).

5. International Alliance for Interoperability
BLIS-Project. SABLE: Simple Access to the Building Lifecycle Exchange (2003).

<http://www.blis-project.org/~sable>

Abstract

This paper discusses on implementation methods for providing data exchange capability of the Construction Project Lifecycle Management System (CPLM). Because most information media currently used in the construction industry are computer files, the system must be capable of handling various file types; however, The goals of PLM cannot be achieved if the contents of the files are communicated freely, which can only be achieved using a standardized information model that has comprehensive coverage of all information produced within the projects (i.e., a BIM). Several methods of adopting the model are investigated here, and a transitional implementation method for K-IFC is suggested, in which a file-based exchange backed by the BIM can evolve into object-level transactions of the BIM with not only advanced data integrity and security, but also improved communications between the project participants.

Keywords : EVMS, EAC, CPI, SPI
