

표준 제품 데이터 변환 방법에 관한 연구

안만진*, 유상봉*

Study on Standard Product Data Translation Method

M. J. Ahn* and S. B. Yoo*

ABSTRACT

Standardization for product data has not been well established and many different presentation methods are being used in CAD/CAM industry. In order to accomplish system integration or concurrent engineering in such situation, product data exchanges among heterogeneous systems are needed. This paper presents a data exchange system between IGES and STEP (AP 202 and 203). The schemata of those standard data formats are represented in EXPRESS and the relationship between corresponding entities are written in EXPRESS-X. Relationships among Non-geometric entities (such as color and annotations) as well as geometric entities are examined. Because the system implemented in this research uses high-level schema language and mapping language, it can be easily extended to support new data formats.

Key words : Product Data, STEP, IGES, EXPRESS-X, Data Exchange

1. 서 론

제품 데이터를 표현하는 표준이 정립되어 있지 않고 많은 종류의 표현 방법들이 사용되고 있다. 이러한 상황에서 시스템 통합 또는 동시 공학을 실현하기 위하여 이기종 시스템 간의 제품 데이터 교환이 필요하다¹⁾. 제품 데이터 교환을 위한 중립 파일로서 지금까지는 IGES, DXF 등이 많이 사용되어 왔으나 제품 정보의 범위를 설계, 제조, 유지 보수, 그리고 폐기 등을 포함한 전 생애 주기로 확장한 STEP이 국제 표준으로 개발되고 있고²⁾, 주요한 상용 시스템들이 이를 지원하는 모듈을 발표하고 있다.

본 논문에서는 기존의 도면 데이터 표준인 IGES 파일과 ISO에서 개발 중인 제품 데이터 표준인 STEP 파일간 데이터 변환시스템을 기술한다. STEP 파일로서는 AP(Application Protocol) 202와 AP 203을 사용하였다. 본 연구에서는 스키마 매핑 언어인 EXPRESS-X³⁾를 이용하여 서로 다른 파일 포맷 간의 변환 관계를 표현한다. IGES와 STEP 응용 프

로토콜 간의 제품 데이터 변환시스템을 구현하기 위해서는 세 가지 스키마 정의가 필요하다. IGES와 STEP 응용 프로토콜의 스키마 정의 그리고 이들 간의 매핑 관계를 정의하는 매핑 정의이다. IGES 스키마는 EXPRESS 언어⁴⁾로 사용하여 정의하였고, 매핑 관계는 EXPRESS-X 언어를 사용하여 정의하였다. 정의된 IGES 스키마와 STEP 응용프로토콜의 스키마, 그리고 매핑 관계는 EXPRESS 컴파일러와 EXPRESS-X 컴파일러로 각각 컴파일하여 변환 라이브러리를 구축한다. 이 변환 라이브러리를 통하여 IGES 파일, AP202 파일, 그리고 AP203 파일 간의 데이터가 변환된다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구에 대한 소개 및 본 연구와의 차이점을 설명하며, 3장에서는 본 연구에서 다루게 될 변환 대상 파일 포맷인 IGES, STEP 응용 프로토콜들(AP202, AP203), 그리고 응용 해석 구조체에 관해 설명한다. 4장에서는 IGES와 STEP 응용 프로토콜 간의 변환시스템과 응용 프로토콜 간의 변환 방법에 관해 설명한 후, 5장에서는 구현 시스템의 활용 예를 설명한다. 끝으로 6장에서는 본 연구의 결론과 향후 연구과제를 제

*인하대학교 자동차공학과

시한다.

2. 관련 연구

제품 데이터의 교환에 관한 연구가 전 세계적으로 활발히 진행되고 있다. 특히 매핑 언어에 관한 연구는 영국의 CIMIO와 미국의 STEP Tools사에 의해 주도되고 있다. CIMIO는 EXPRESS-M^[5] 컴파일러를 구현하여 상용화된 상태이고 현재 EXPRESS-X 컴파일러를 구현하고 있다. ST-Developer 1.5 제품에는 EXPRESS-V^[6] 컴파일러가 포함되어 있으며, 1997년 9월에 발표된 1.6 버전에는 EXPRESS-X 컴파일러도 포함하고 있다.

RaDEO(Radio Design Exploration and Optimization program)의 지원하에 FAMU-FSU(Florida A&M University-Florida State University)의 IPPD(Integrated Product and Process Development)에서는 'STEP AP203 to AP209 Mapper' 시스템을 구현하고 있다^[7]. 이 시스템은 AP203의 데이터 파일 중에서 부품 형상과 구성 관리 데이터만을 읽어 AP209 파일을 생성한다. 이 시스템은 AP203 스키마와 AP209 스키마를 STEP Tools사의 EXPRESS 컴파일러를 사용하여 컴파일하여 ST-Developer 라이브러리를 구성한 후, 이것을 통해 AP203 파일을 AP209 파일로 변환하며 CORBA를 사용하여 제품 서버와 클라이언트 환경을 구축하고 있다.

NIIP(The National Industrial Information Infrastructure Protocols)는 IBM을 주축으로 1994년에 구성된 컨소시엄으로 '산업체 가상 기업(Industrial Virtual Enterprise, IVE)'을 지원하기 위해 필요한 프로토콜을 정의하고, 인터넷 또는 인트라넷 상에서 IVE가 STEP 데이터베이스를 구축하고 사용할 수 있도록 인프라를 구축하고 있다^[8]. 이 연구에서는 AIM과 ARM 간의 매핑을 위하여 EXPRESS-X를 사용하였다.

이 영준은 IGES 데이터를 STEP 데이터로 변환하는 변환시스템을 구현하였다^[9]. 이 시스템에서는 EXPRESS 언어를 사용하여 IGES 스키마를 정의하였으며, STEP의 스키마는 파트 42번의 스키마들을 사용하였다. 그리고 이들 간의 데이터 변환을 위해 EXPRESS 언어를 사용하여 변환 함수들을 정의하였다. 스키마들과 변환 함수를 STEP 틀인 Fcdex+로 컴파일하여 변환 라이브러리를 구축하여 전체 변환시스템을 구현하였다.

이상의 기존 연구들과 비교할 때 본 연구의 특징

은 다음과 같이 요약된다. 본 연구에서는 EXPRESS-X 언어를 사용하여 변환관계를 정의하였기 때문에 EXPRESS 함수나 C와 같은 프로그래밍 언어를 사용하는 것보다 변환관계를 정의하기가 더 용이하다. 또한 본 연구는 형상 데이터 외의 비형상 데이터의 변환을 다루었다. 그리고 본 연구에서 구현한 변환 시스템은 필요한 구성 요소들이 EXPRESS와 EXPRESS-X 언어를 사용하여 모듈화 되어있기 때문에 변환 대상 파일 포맷의 추가 또는 삭제가 용이하다.

3. 변환 대상 파일 포맷

3.1 IGES(Initial Graphics Exchange Specification)

1980년에 발표되고 1981년에 ANSI 표준으로 채택된 IGES 버전 1.0은 2차원 형상 요소와 문자 정보를 정의하는 파일 형식이다. 1983년 2월에 발표된 버전 2.0은 형상 요소 정의 기능을 추가하였고, 유한 요소 데이터와 인쇄 배선 제품(PCB) 데이터의 정보 교환을 해결하였으며 이전 형식 표현을 도입하여 파일 크기와 처리 속도 문제를 해결하였다. 1986년 4월에는 버전 3.0이 발표되었다. 이 버전은 곡면과 3차원 와이어프레임 모델 형상을 지원하며 주석 스타일을 더 광범위하게 표현할 수 있고, 표준 부품 라이브러리를 위해 필수적인 사용자 정의 매크로의 성능을 향상시켰다. 또한 기존의 IGES 파일 크기를 1/3로 축소하는 압축 ASCII 형식을 도입하였다. 버전 4.0은 1988년 6월에 발표되었으며 CSG(Constructs Solid Geometry) 모델링 데이터를 다룰 수 있는 엔티티가 도입되었다. 1990년의 버전 5.0은 경계 표현법(Boundary Representation)에 의한 모델링을 지원한다^[10]. IGES는 버전 6.0을 마지막으로 ISO STEP으로 발전할 전망이다^[9].

IGES 버전 5.0^[11]의 엔티티들은 크게 기하학적(Geometry) 엔티티, CSG 엔티티, 주석(Annotation) 엔티티, 그리고 구조(Structure) 엔티티들로 구분된다. 기하학적 엔티티는 제품 형상 정의를 표현하며 곡선과 곡면을 포함한다. 기하학적 엔티티는 일반적으로 100~199의 엔티티 번호를 사용한다. CSG 엔티티는 솔리드 모델에서 사용되는 기본(Primitive) 엔티티들을 포함하며, 주석 엔티티로는 치수(Dimension), 치수 표기(Note), 화살표(Leader) 등이 있으며 200~299의 엔티티 번호를 사용한다. 구조 엔티티는 연관성(Associativity), 뷰(View), 부그림(Sub-figure), 외부 참조, 매크로(MACRO) 엔티티 등을 포함한다.

3.2 STEP 응용 프로토콜(Application Protocol, AP)

STEP의 응용 프로토콜은 통합 자원(일반 자원과 응용 자원)과 응용 해석 구조체(Application Interpreted Construct, AIC)로부터 각 응용 분야에 필요한 개념(엔티티)을 추출하여 구성되며, 필요시 제한된 규정에 의해 새로운 관계나 속성을 첨가할 수 있다^[1].

3.2.1 AP202

AP202는 조합 제도(Associative Draughting)로서 CAD 도면과 3차원 형상을 교환하기 위한 목적으로 사용되는 응용 프로토콜이다. AP202에서 도면의 전체적인 내용은 제품 형상의 기하학적인 표현과 관련이 될 수 있다. 제품 형상의 기하학적인 표현은 2차원 또는 3차원으로 표현될 수 있으며 모두 8개의 기하학적 형상을 지원한다^[12].

AP202는 Fig. 1처럼 크게 도면 구조, 주석, 제품 관계, 그룹핑, 연관성(Associativity), 기하학적 형상 등으로 나눌 수 있다. 도면 구조는 도면 승인, 책임 조직 등과 같은 도면의 관리에 해당되는 정보들이며, 주석은 치수, 텍스트 등 도면에 필요한 도면 요소 등에 관한 정보들이다. 제품 관계는 제품을 도면과 연관시키기 위해 필요한 관리 정보들을 포함하며, 그룹핑은 형상 모델, 또는 도면의 수정 작업에 도움을 주는 기하학적 요소, 혹은 도면 요소들의 집합이다. 연관성은 제품의 기하학적 모델과 치수, Callout, Fill areas와의 관계를 표현한다. 즉, 요소의 크기, 요소들 간의 거리, 요소의 표현 특징에 관한 정보 등을 포함한다. 기하학적 형상은 고등 경계 표현, 각진 경계 표현, 기본 경계 표현, 다양체 곡면, 위상 정보가 있는 와이어 프레임, 위상 정보가 없는 곡면 및 와이어프레임, 기하학적으로 유한한 2차원 와이어 프레임을 포함한다.

AP202에서 지원하는 제품 형상 모델들은 Fig. 2와 같다. 이 모델들은 모두 위의 8개의 기하학적 형상 모델들이며, 와이어프레임 모델 중에서 기하학적으로 유한한 2차원 와이어프레임 모델과 솔리드 모델의 기본 경계 표현 모델을 제외한 6개의 모델들은 AP203에서 지원하고 있는 모델과 일치하며 상호 데이터 교환이 가능하다. AP202는 제품 형상 모델 중에서 CSG 모델을 지원하지 않으며 IGES의 CSG 모델과 상호 교환될 수 없다. AP202는 형상 데이터 외에 2차원에서 정의된 주석과 3차원에서 정의된 평면상의 주석(Planar Annotation)을 포함한다. 그러나 제품과 관련 없는 도면, 3차원에서 정의된 Non-planar 주석, BOM 데이터 등은 포함하지 않는다^[13].

3.2.2 AP203

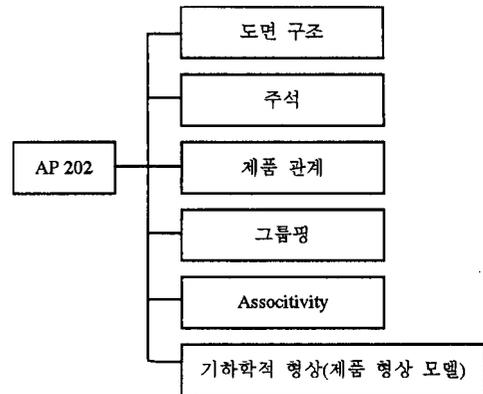


Fig. 1. AP202의 구성요소.

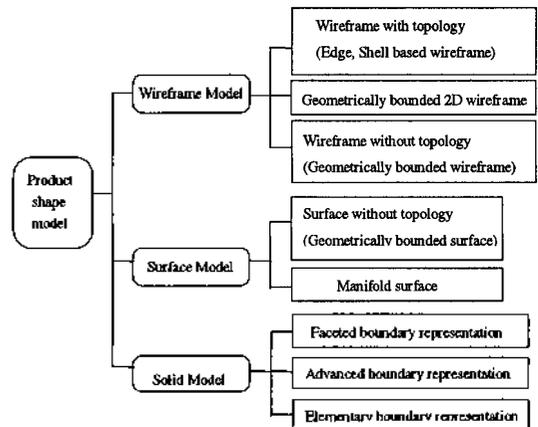


Fig. 2. AP202의 제품 형상 모델 분류.

AP203은 구성 제어 설계(Configuration Controlled Design)로서 기계 부품과 조립품에서 제품의 형상에 관계없이 구성 제어된 3차원의 설계를 위한 응용 시스템 간의 설계 정보의 교환을 가능하게 한다. AP203은 도면 관련 데이터는 지원하지 않으며 기계 부품과 조립품의 3차원 형상을 지원하며 모두 6개의 기하학적 형상을 지원한다^[12].

AP203은 Fig. 3처럼 구성 관리, 제품 구조, 규격, 기하학적 형상 등으로 구분할 수 있다. 구성 관리는 제품 데이터 승인, 제품 개발을 진행하기 위한 작업 지침, 제품 모델의 구성 부품의 계획된 사용에 관한 정보, 부품이나 부품의 설계를 생성하는 조직에 관한 정보 등을 포함한다. 제품 구조는 BOM 데이터, 조립품, 부품 등에 관한 정보 등을 나타내며, 규격은 부품이나 제품의 품질과 관련된 정의, 공정 및 규칙 등을 포함하는 문서이다. 기하학적 형상은 고등 경계 표현, 각진 경계 표현, 다양체 곡면, 위상 정보가

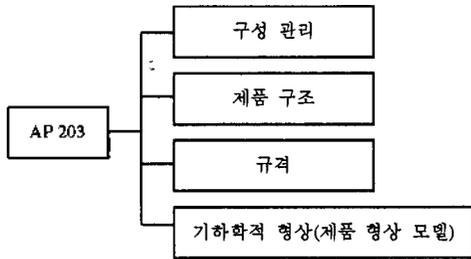


Fig. 3. AP203의 구성요소.

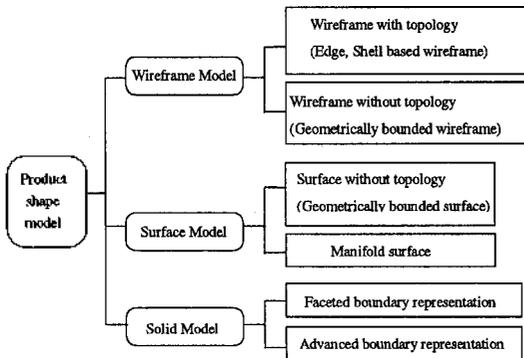


Fig. 4. AP203의 제품 형상 모델 분류.

있는 와이어 프레임, 그리고 위상 정보가 없는 곡면 및 와이어프레임 등 6개의 형상을 포함한다. 이들 형상 모델은 모두 AP202의 형상 모델 중에서 기본 경계 표현과 기하학적으로 유한한 2차원 와이어프레임을 제외한 형상 모델과 일치한다.

AP203에서 지원하는 제품 형상 모델들은 Fig. 4와 같다. 이 모델들은 모두 AP202의 제품 형상 모델의 일부분과 일치하며 상호 교환될 수 있다. AP203과 AP202의 가장 큰 차이점은 AP202처럼 도면과 관련된 주석 데이터 등을 지원하지는 않지만 제품 개발의 설계 단계에 속하는 제품 정의 데이터와 구성 제어 데이터, 그리고 BOM 데이터 등을 지원한다는 것이다. AP203 역시 CSG 모델은 지원하지 않는다¹¹⁾.

3.3 응용 해석 구조체(Application Interpreted Constructs, AIC)

AIC는 1991년 가을 휴스톤에서 개최된 STEP 회의에서 CADEX와 PDES사에 의해 처음으로 정의되고 쓰여졌으며, 현재 18개의 AIC가 개발 중이다. AIC는 여러 응용 프로토콜 사이에서 공유할 수 있는 데이터 모델로 적어도 두 개의 AIM(Application Interpreted Model)에 공통인 구조체이다. AIC는 통합 자원에서 정의된 정보 모델을 응용 프로토콜에서 이

용할 수 있도록 최소 단위로 나눈 것이다.

응용 해석 구조체의 목적은 서로 다른 응용 프로토콜 간의 정보 교환을 최대한 보장하기 위한 것이다. 각 응용 프로토콜 간 또는 새로운 데이터 응용 프로토콜과의 데이터 교환 등과 같은 응용 프로토콜 간의 상호 작용이 필요한 경우에 기존의 제품 형상 표현을 무시하고 새로운 제품 형상을 표현하는 것은 바람직하지 못하다. 따라서 기존의 응용 프로토콜의 일부분을 재사용해야 할 것이다. 이런 관점에서 볼 때, 일단 제품 형상이 모델링 된 후에 다른 응용 프로토콜들은 이 모델을 공유해야 하고 또 서로 공유할 수 있도록 모델링되어야 한다. 즉, STEP에 기초한 정보 모델에서 서로 관련 있는 부분의 중복을 허용하는 상호 작용성(Interoperability)을 AIC가 제공한다.

1996년에 국제 표준으로 인정 받은 AP202는 Table 1에서처럼 전체 18개의 AIC 중에서 12개의 AIC를 사용하고 있다¹³. 현재 AP203은 AIC를 사용하고 있지 않지만 앞으로 AIC가 국제 표준으로 승인을 받는다면 이들을 사용할 것으로 예상된다. Table 1은 AP202와 AP203에 포함되는 AIC를 분류하였다.

4. 변환시스템의 구현

4.1 변환시스템의 전체 구조

본 연구에서 구현한 제품 데이터 변환시스템의 전체 구조는 Fig. 5에서처럼 4개의 구조로 나누어져 있

Table 1. AP202와 AP203에 포함되는 AIC의 분류

| Part # | AIC(Application Interpreted Constructs) | AP 202 | AP 203 |
|--------|--|--------|--------|
| 501 | Edge-based wireframe | O | O |
| 502 | Shell-based wireframe | O | O |
| 503 | Geometrically bounded 2D wireframe | O | X |
| 504 | Draughting annotation | O | X |
| 505 | Draughting structure and administration | O | X |
| 506 | Draughting element | O | X |
| 507 | Geometrically bounded surface | O | X |
| 508 | Non-manifold surfacc | X | X |
| 509 | Manifold surface | O | O |
| 510 | Geometrically bounded wireframe | O | O |
| 511 | Topology bounded surface | X | X |
| 512 | Faceted boundary representation | O | O |
| 513 | Element boundary representation | O | X |
| 514 | Advanced boundary representation | O | O |
| 515 | Constructive solid geometry | X | X |
| 516 | Mechanical design context | X | X |
| 517 | Mechanical design geometric presentation | X | X |
| 518 | Mechanical design shared presentation | X | X |

주) O: AP에 포함됨, X: AP에 포함되지 않음

다. 즉, 변환 대상 파일 스키마, 이들 변환 대상 파일 간의 변환관계를 정의하는 매핑 스키마, 제품 데이터 매핑 라이브러리, 그리고 물리적인 변환 대상 파일 등으로 구성되어 있다.

첫 번째의 변환 대상 파일 포맷은 IGES 스키마, AP202 스키마, 그리고 AP203 스키마이며 이들은 모두 EXPRESS 언어로 정의되어 있다. IGES 스키마는 본 연구에서 직접하였고, AP202와 AP203는 STEP 표준을 이용한다. 이 스키마들은 EXPRESS 컴파일러를 거쳐 데이터 매핑 라이브러리에 추가된다. 두 번째의 매핑 스키마는 IGES-AP202 매핑 스키마, IGES-AP203 매핑 스키마, AP202-AP203 매핑 스키마들로 구성되어 있다. 이들 각각의 매핑 스키마들은 IGES 스키마와 AP202, IGES 스키마와 AP203, 그리고 AP202 스키마와 AP203 스키마 간의 엔티티 변환관계를 EXPRESS-X로 정의한다. 이들은 각각 EXPRESS-X 컴파일러를 거쳐 매핑 라이브러리에 추가된다. 세 번째는 제품 데이터 매핑 라이브러리의 첫 번째와 두 번째의 스키마들이 각각 컴파일러를 거쳐 생성된 라이브러리들로 구성되어 있다. 또한 이 라이브러리는 IGES의 물리적 파일을 읽고 쓸 수 있도록 IGES File Read/Write 라이브러리를 포함하고 있다. 네 번째는 물리적인 변환 대상 파일로 실제로 변환할 대상 파일들이다. 제품 데이터 매핑 라이브러리는 IGES 파일을 입력으로 받아 IGES-AP202 매핑 라이브러리, 또는 IGES-AP203 매핑 라이브러리를 사용하여 STEP 파일을 생성할 수 있으며, AP202 파일을 입력으로 받아 AP203 파일을 생

성할 수 있다.

4.2 스키마 매핑 언어(EXPRESS-X)

스키마 매핑 언어는 스키마들에 의해 기술된 응용 모델 간의 데이터의 변환을 용이하게 하기 위해 엔티티 인스턴스들이 스키마들 사이에서 어떻게 매핑되는지를 기술하는 언어이다. 일반적으로 매핑 언어가 필요로 하는 요구조건은 다음과 같다. 인간이 이해하기 쉬운 언어이어야 하며, EXPRESS 언어의 구조와 유사하여야 한다. 데이터 교환을 위해 명확한 규격이 정의되어야 하며, 한방향 뿐만 아니라 양방향 매핑, AIM과 ARM 간의 매핑을 할 수 있어야 한다^[8]. 매핑 언어는 AP 간의 매핑, CAD 시스템과 AP 간의 매핑, CAD 시스템 간의 매핑을 정의할 수 있어야 한다. 또한 매핑 언어의 규격 중에서는 엔티티 매핑과 속성 타입 매핑, 서로 다른 뷰 엔티티로의 조건 매핑, 수동 인스턴스와, 타입 변환과 캐스팅 등을 수행할 수 있어야 한다. EXPRESS-X는 매핑 언어로 위와 같은 요구조건을 만족시킨다^[15].

EXPRESS-X는 1996년 코베에서 개최된 STEP 회의에서 EXPRESS 사용자들이 필요로 하고 있는 매핑과 이미 존재하는 EXPRESS 매핑 언어들의 가장 좋은 특징들을 조합할 수 있는 방법을 조사하여, STEP과 EXPRESS 사용자들의 요구조건을 충족시키는 EXPRESS 매핑 언어를 개발하기 위한 프로젝트로 소개되었다^[16]. EXPRESS-X 언어에서 두 개의 응용 모델 간의 데이터 변환을 정의하기 위해서 3개의 스키마 정의가 필요하다. 첫 번째 스키마는 매

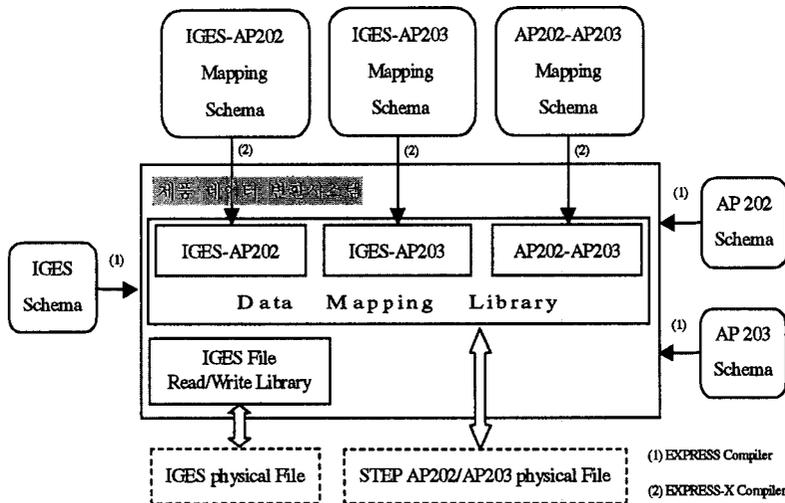


Fig. 5. 제품 데이터 변환시스템의 구조.

핑을 수행할 응용 모델에 대한 스키마이고, 두 번째 스키마는 매핑 스키마 정의에 의해 생성되는 응용 모델에 대한 스키마이다. 이 두 개의 스키마는 모두 EXPRESS 스키마를 사용하여 정의하며, 전자를 베이스 스키마라 하고 후자를 뷰 스키마라 한다. 세 번째 스키마는 매핑 스키마(예를 들어, IGES-AP202_mapping_shema)라 하며, 위의 두 스키마 내의 엔티티들 사이의 매핑관계를 정의한다.

EXPRESS-X 규격은 하나 이상의 매핑 스키마들로 구성되며 각 매핑 스키마 구문은 2 가지 경우를 제외하고는 EXPRESS의 스키마 규격과 유사하다. 즉 선언문과 글로벌 섹션이 필요하다. 구문은 SCHEMA_MAP으로 시작하며 END_SCHEMA_MAP으로 끝난다. schema_body는 글로벌 섹션과 선언문들로 구성된다. 글로벌 섹션은 GLOBAL로 시작하며 매핑 스키마에서 참조될 베이스 스키마와 뷰 스키마가 이 부분에서 선언되며 END_GLOBAL로 끝난다. 선언문들은 EXPRESS의 선언문과 유사하며 VIEW 선언문, COMPOSE 선언문, 그리고 MEMBER 선언문 등 3개의 선언문이 추가되었다.

<예 1> 데이터 변환을 보이기 위하여 먼저 매핑 스키마에서 사용될 베이스 스키마와 뷰 스키마의 정의가 필요하다. Fig. 6과 같이 베이스 스키마 이름은 Base_Schema이고 뷰 스키마 이름은 View_Schema이다. Fig. 7은 이들 베이스 스키마와 뷰 스키마들 사이의 데이터 변환을 정의하기 위한 EXPRESS-X 매핑 언어의 구문이다. 베이스 스키마의 point 엔티티 속성타입이 모두 실수형이며 뷰 스키마의 cartesian_point 엔티티의 속성타입도 모두 실수형이다. 이렇게 기본적인 속성 타입만을 갖는 엔티티의 변환은 Fig. 7과 같이 VIEW 선언문만을 사용하여 엔티티 변환관계를 정의한다.

Fig. 7의 Mapping_Schema는 매핑 스키마 이름이며, 글로벌 섹션에서는 매핑 스키마에서 참조되는 베이스 스키마와 뷰 스키마들이 각각 sdb와 tdb라는 유일한 이름으로 선언된다. 그리고 베이스 엔티티 cartesian_point와 뷰 엔티티 point를 각각 cp 그리고 bp라는 이름을 사용하여 이 매핑 스키마 내에서 참조된다. 이 VIEW 선언문은 베이스 엔티티 point로부터 뷰 엔티티 cartesian_point를 생성한다. FROM절은 베이스 모델의 point 엔티티 인스턴스로부터 생성될 것이라는 것을 지시한다. cartesian_point 인스턴스는 베이스 인스턴스 point로부터 WHEN절 내의 조건이 참일 경우에 한해 생성된다. Fig. 7처럼 WHEN절 내에 조건식이 없고 TRUE인 경우에는 point로부터 향

```

SCHEMA Base_Schema;
ENTITY point;
  x : REAL;
  y : REAL;
  z : REAL;
END_ENTITY;

ENTITY base_circle;
  point_1 : point;
  point_2 : point;
  point_3 : point;
END_ENTITY;

END_SCHEMA;

SCHEMA View_Schema;
ENTITY cartesian_point;
  x : REAL;
  y : REAL;
  z : REAL;
END_ENTITY;

ENTITY circle;
  center_point : cartesian_point;
  radius : REAL;
END_ENTITY;

END_SCHEMA;

```

Fig. 6. 베이스 스키마와 뷰 스키마의 예.

SCHEMA_MAP Mapping_Schema;

GLOBAL

```

DECLARE sdb INSTANCE OF Base_Schema;
DECLARE tdb INSTANCE OF View_Schema;

```

END_GLOBAL;

VIEW cp : tdb::cartesian_point;

FROM (bp : sdb::point)

WHEN (TRUE);

BEGIN_VIEW

```

  cp.x := bp.x;

```

```

  cp.y := bp.y;

```

```

  cp.z := bp.z;

```

END_VIEW;

END_SCHEMA_MAP;

Fig. 7. 매핑 스키마 정의의 예.

상 cartesian_point 인스턴스가 생성된다. 실제 인스턴스들의 속성값들은 BEGIN_VIEW 키워드와 END_VIEW 사이에서 할당문을 사용하여 계산된다. 즉, 베이스 엔티티 인스턴스의 속성인 cp.x, cp.y, cp.z의 속성값으로부터 뷰 엔티티 인스턴스의 속성인 bp.x, bp.y, bp.z의 속성값으로 각각 복사된다.

위의 세 개의 스키마들을 정의한 후, 각각 컴파일하여 실제 데이터를 변환할 수 있는 데이터 변환 라이브러리를 구성한다. Fig. 8은 Fig. 6의 Base_Schema를 바탕으로 구성된 실제 데이터이며 Fig. 9는 이 데이터들을 위의 데이터 변환 라이브러리를 거쳐 생성된 변환 데이터들이다.

4.3 제품 형상 데이터 변환

IGES 버전 5.0에서 CSG 엔티티들을 제외한 기하학적 엔티티는 총 22개로 조사되었다. 이들 엔티티 중에서 Copious Data(#106) 엔티티는 점들의 집합으로 형상을 표현하며, Parametric Spline Curve(#112)

```

ISO-10303-21;
HEADER;
FILE_DESCRIPTION(('IGES mapping file'),'1');
FILE_NAME('f202x.stp','some date','(Man Jin, Ahn)','(Information System Lab., Dept. of Industrial
Automation, Inha Univ.)','IGES2STEP MAPPING Ver 1.0','$,$');
FILE_SCHEMA(('Base_Schema'));
ENDSEC;

DATA;
#1=BASE_CIRCLE(#2,#3,#4);
#2=POINT(3.0,1.0,2.0);
#3=POINT(3.1,2.2,3.3);
#4=POINT(34.0,32.00,2.33);
ENDSEC;
END-ISO-10303-21;

```

Fig. 8. Base_Schema의 물리적 파일의 예.

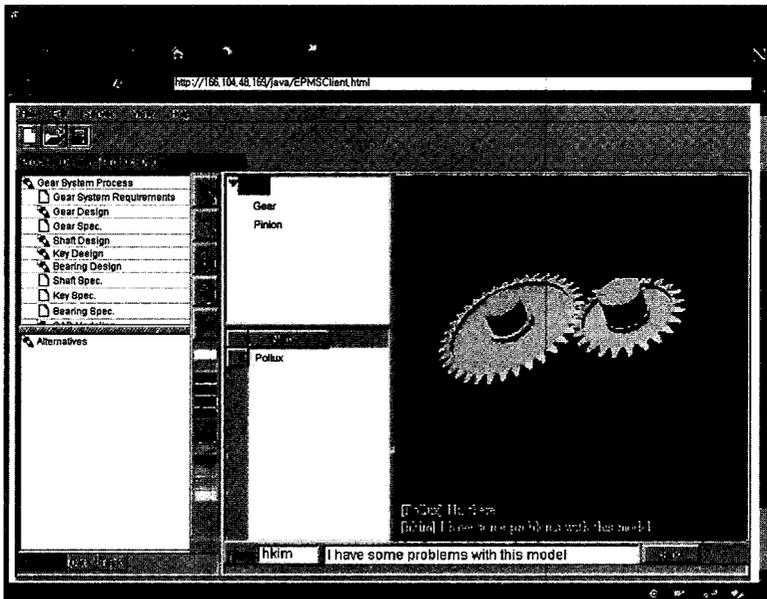


Fig. 9. 변환된 물리적 파일의 예.

는 곡선을 여러 개의 섹먼트들로 분할한 구간별 매개변수 다항식으로 표현한 것이며, Ruled Surface(#118)는 직선 또는 두 공간 곡선상의 해당 점을 연결하여 생성되는 곡면으로서 이들 엔티티는 STEP에 대응되는 엔티티가 존재하지 않으므로 변환되지 않는다. Boundary(#141)와 Boundary Surface(#143) 엔티티는 아직 완성되지 않은 엔티티들로 본 연구에서는 제외하였다. Boundary 엔티티는 곡면 위에 놓인 곡선들의 집합으로 구성된 곡면 경계를 나타내며, Boundary Surface 엔티티는 트리밍된 곡면들과 변수

전달에 사용된다. 따라서 22개의 기하학적 엔티티 중에서 위에서 설명한 5개의 엔티티를 포함하여 Parametric Spline Surface(#114)와 Flash(#125) 엔티티는 변환이 불가능하며, STEP 엔티티로 변환이 가능한 IGES의 기하학적 엔티티들은 15개로 조사되었다. Table 2는 변환 가능한 IGES와 STEP의 기하학적 엔티티의 변환관계를 나타낸다.

IGES 기하학적 엔티티 외의 CSG 엔티티는 AP 202와 AP203에서 지원하지 않으므로 이들 AP로의 변환은 불가능하다. 그러나 AP207(Sheet Metal Die

Table 2. IGES와 STEP의 기하학적 엔티티의 변환관계

| IGES Entities | STEP Entities(AP202) |
|--------------------------------------|--|
| Circular Arc (#100) | trimmed_curve circle |
| Composite Curve (#102) | composite_curve |
| Conic Arc (#104) | ellipse parabola hyperbola |
| Planc (#108) | plane |
| Line (#110) | polyline or line |
| Point (#116) | cartesian_point |
| Surface of Revolution (#120) | surface_of_revolution |
| Tabulated Cylinder (#122) | surface_of_linear_ex- trusion |
| Transformation Matrix (#124) | cartesian_transfor- mation_operator |
| Rational B-Spline Curve (#126) | b_spline_curve |
| Rational B-Spline Surface (#128) | b_spline_surface |
| Offset Curve (#130) | offset_curve_2d offset_curve_3d |
| Offset Surface (#140) | offset_surface |
| Curve on a Parametric Surface (#142) | curve_on_surface |
| Trimmed Parametric Surface (#144) | curve_bounded_surface |

Planning and Design)의 형상 정의 표현, AP210 (Electronic Assembly, Interconnection and Packaging Design)의 기하, 그리고 AP227(Plant Spatial Configuration)의 형상은 CSG 모델을 사용하고 있으므로 적절한 변환관계를 정의함으로써 변환이 가능하다.

AP202에는 총 345개의 엔티티를 갖고 있으며, AP203에는 총 252개의 엔티티를 갖고 있다. AP 202과 AP203 사이의 변환 가능한 엔티티는 178개로 조사되었다. 변환이 가능한 엔티티들 중에서 기하학적 형상에 관련된 엔티티들은 파트 42번의 geometry 스키마에 속한 엔티티들이다. 변환되지 않는 형상으로는 기하학적으로 유한한 2차원 와이어프레임과 기본 경계 표현 모델이다. 위의 Table 2에서 STEP AP202의 엔티티는 AP203에도 존재하며 이들 엔티티 간의 변환이 가능하다.

4.4 제품 비형상 데이터 변환

IGES 버전 5.0의 비형상 엔티티는 선 종류, 선 폭, 컬러와 주석 엔티티 등이다. 주석 엔티티는 총 14개로 조사되었다. 이들 엔티티 중에서 Flag Note(#208), General Label(#210), Point Dimension(#220) 엔티티는 STEP에 대응되는 엔티티가 존재하지 않으므로 변환되지 않는다. New General Note(#213) 엔티티는 아직 완성되지 않은 엔티티들로 본 연구에서는 제외하였다. 따라서 14개의 주석 엔티티 중에서 위에서 설명한 4개의 엔티티는 변환이 불가능하며, STEP

Table 3. IGES와 STEP AP202의 주석 엔티티의 변환관계

| IGES Entities | STEP Entities(AP202) |
|---------------------------|----------------------|
| Line weight number | curve_style |
| Line font pattern | curve_style |
| Color | color_rgb |
| Angular Dimension (#202) | angular_dimension |
| Curve Dimension (#204) | dimension |
| Diameter Dimension (#206) | diameter_dimension |
| General Note (#212) | text_string |
| Leader (Arrow) (#214) | leader |
| Linear Dimension (#216) | linear_dimension |
| Ordinate Dimension (#218) | ordinate_dimension |
| Radius Dimension (#222) | radius_dimension |
| General Symbol (#228) | symbol_style |
| Sectioned Area (#230) | fill_area_style |

엔티티로 변환이 가능한 비형상 엔티티는 선 종류, 선 폭, 컬러와 IGES의 주석 엔티티들 10개로 조사되었다. Table 3은 변환 가능한 IGES와 STEP AP 202의 비형상 엔티티들의 변환관계를 나타낸다.

AP202와 AP203 간의 변환 가능한 엔티티들 중에서 기하학적 형상 데이터 외의 비형상 데이터들 중에서 AP202의 제품 관계를 구성하는 정보 중 일부는 AP203의 구성 관리와 제품 구조의 일부분으로 변환된다. AP202의 도면 구조를 구성하는 정보의 일부는 AP203의 구성 관리와 규격의 일부분으로 변환된다. 이들 관계는 Table 4에서 보여준다.

AP202에서 제품관계(product_relation)를 기술하기 위해 필요한 정보들은 조직(Organization), 제품정의(Product_definition), 제품버전(Product_version)에 관한 데이터들이다. 조직은 제품을 디자인하고, 생산하고, 공급하는 사람들 또는 그룹들의 집합이다. 조직과 관련된 데이터는 조직이름(organization_name)과 주소(address)이다. 제품정의와 관련된 데이터는 캐드파일이름(cad_filename)과 discipline_id로 캐드파일이름은 CAD 시스템에서 디자인된 도면 형상 모델을 포함하고 있는 파일이름이며, discipline_id는 제품에 관한 정의가 속해 있는 어플리케이션 식별자이다. 제품버전과 관련된 데이터는 제품 식별자(product_id)와 개버전 식별자(revision_id)이다.

AP203의 UoF(Unit of Functionality) 중 authorization은 조직과 주소 데이터를 포함하고 있으며, 이들 데이터와 AP202의 조직이름과 주소 데이터 간의 변환이 가능하다. AP203의 part_identification은 Design_discipline_product_definition, 부품(Part), 부품 버전(Part_version)을 사용하여 정의된다. Design_discipline_product_definition과 관련된 데이터 중에 캐드파일이름과 discipline_id가 존재하며 AP202의

Table 4. STEP AP202-AP203 비형상 엔티티 변환관계

| AP202 | | | | AP203 | | | |
|--------------------------------------|-----------------------|-------------------------|--|-------------------------|--------------------------------------|---------------------|--|
| UoF | Application object | Application element | Entity | UoF | Application object | Application element | Entity |
| product_relation | Or-organization | organization_name | organization | authorization | Person-Organization | organization/person | organization.name/pers on. id |
| | | address | address | | | address | address |
| | Product definition | cad_filename | document. id | part identification | Design discipline_product_definition | cad_filename | document. id |
| | | discipline_id | product_definition_context. life_cycle_stage | | | discipline_id | product_definition_context. life_cycle_stage |
| | Product version | product_id | product.id | | Part | product_number | product.id |
| | | revision_id | product_definition_formation. id | | Part_version | revision_letter | product_definition_formation. id |
| drawing_structure_and_administration | Approval | date | date_and_time | au- thorization | Approval | date | date_and_time |
| | Approval | description | Approval.level | | Approval | purpose | approval.level |
| | Drawing Drawing_sheet | security_classification | security_classification_level.name | BOM/part identification | | security_code | security_classification_level |
| | | Drawing | drawing_specification | | document.id | part identification | Spec-ification |
| contract_reference | contract.name | | Part_version | contract_number | contract.name | | |
| draughting_shape_model | Sub_model | transformation | representation_map.mapping_origin/mapped_item.mapping_target | BOM | Component_assembly_position | transformation | representation_map.mapping_origin/mapped_item.mapping_target |

카드파일이름과 discipline_id로 변환이 가능하다. 부품과 관련된 데이터는 유사한 부품별로 구분하는 부품군, 조직 내에서 사용되는 부품이름, 부품을 식별할 수 있는 부품번호, 부품타입 등이다. 이들 중에서 부품번호가 AP202의 제품 식별자와 교환이 가능하다. 부품버전과 관련된 데이터 중에서 revision_letter는 부품의 특정 버전을 식별할 수 있다. 이 데이터는 AP202의 재버전 식별자와 교환이 가능하다. Table 4는 이것에 관한 엔티티 변환 관계를 보여준다.

위에서 예를 들어 언급한 데이터를 표현하기 위해 필요한 엔티티들은 Table 4에서처럼 AP202와 AP203에 모두 존재하며 구조가 일치한다. 이들 엔티티들은 AP202와 AP203에서 각각 사용되는 영역은 다르지만 데이터들이 담고있는 정보가 동일하고 이들이 사용하는 엔티티들도 각각 AP202와 AP203에 존재하므로 응용 프로토콜 간의 변환에서 교환될 수

있다.

4.5 엔티티 변환관계의 분류

엔티티 변환관계의 분류 기준은 다음과 같다. 첫 번째는 엔티티 속성타입의 종류에 따른 분류이며, 두 번째는 하나의 엔티티로부터 변환되는 엔티티의 개수에 따른 분류이다. 세 번째는 속성들 간의 변환식의 필요유무에 따른 분류이다. 먼저 첫 번째 분류 기준으로 분류한 후, 분류된 것을 다시 두 번째 분류 기준으로 분류하였으며, 이것을 또다시 세 번째의 분류 기준으로 분류하였다.

4.5.1 엔티티 속성타입의 종류에 따른 분류

IGES 엔티티는 실수, 정수 등과 같은 기본적인 속성타입만으로 구성되어 있는 엔티티와 속성 중에서 다른 엔티티를 가리키는 포인터를 속성타입으로 갖는 엔티티들이 있다. 첫 번째 경우에는 VIEW 선언

```

SCHEMA_MAP Mappint_Schema;
.....
VIEW cp : tdb::cartesian_point;
FROM (bp : sdb::point)
WHEN (TRUE);
BEGIN_VIEW
  cp.x := bp.x;
  cp.y := bp.y;
  cp.z := bp.z;
END_VIEW;

VIEW vc : tdb::circle;
FROM (bc : sdb::base_circle)
WHEN (TRUE);
BEGIN_VIEW
  vc.radius := SQRT (
    (bc.point_1.x-bc.point_3.x)
    *(bc.point_1.x-bc.point_3.x)
    +(bc.point_1.y-bc.point_3.y)
    *(bc.point_1.y-bc.point_3.y) );
END_VIEW;

COMPOSE c : tdb::circle;
WHEN (TRUE);
BEGIN_COMPOSE
  FROM (cp : tdb::cartesian_point)
  WHEN(TRUE);
  BEGIN
    c.center_point := cp;
  END;
END_COMPOSE;

END_SCHEMA_MAP;

```

Fig. 10. 다른 엔티티를 가리키는 포인터를 속성타입으로 갖는 엔티티의 매핑 스키마 정의.

문만을 사용하여 엔티티 사이의 매핑을 정의하며, 두 번째 경우에는 VIEW 선언문과 COMPOSE 선언문을 함께 사용하여 매핑을 정의한다.

기본적인 속성 타입만을 갖는 엔티티의 변환은 예 1에서 설명한 것처럼 Fig. 7과 같이 VIEW 선언문만을 사용하여 엔티티 변환관계를 정의한다.

다음은 각 엔티티가 자신만의 속성으로 어떤 형상을 표현하지 못하고 다른 엔티티를 가리키는 포인터를 속성타입으로 갖는 속성을 추가하여 어떤 완전한 형상 표현을 정의할 수 있는 엔티티들이다. 즉 여러 개의 엔티티들을 함께 사용하여 형상을 표현한다. Fig. 10은 이에 대한 EXPRESS-X 매핑 언어의 구문이다. 베이스 스키마와 뷰 스키마는 Fig. 6에 있는 것을 사용한다. Fig. 6에서 base_circle 엔티티는 각 속성의 속성타입으로 point 엔티티를 갖는다. 뷰 스키마 엔티티 circle은 베이스 엔티티 point와 base_circle로부터 생성된다. 하지만 circle 엔티티의 속성 중 center_point는 다른 뷰 엔티티인 cartesian_point를 속성타입으로 갖고 있기 때문에 이 엔티티 인스턴스가 생성된 후에 비로소 완전한 circle 엔티티 인스턴스가 생성될 것이다. 이 center_point의 속성값을 계산하기 위하여 COMPOSE 선언문이 사용된다.

4.5.2 하나의 엔티티로부터 변환되는 엔티티의 개수에 따른 분류

IGES 형상 엔티티는 일반적으로 하나의 엔티티가 하나의 형상만을 정의하며 STEP으로 변환될 때, 한 개의 STEP 엔티티로 변환된다. 하지만 형상의 수학적 정의를 비슷하여 각각의 형상을 서로 다른 엔티티로 정의하지 않고 하나의 엔티티만으로 정의하는 경우가 있다. 이럴 경우 엔티티를 구성하는 속성값에 따라 여러 가지 형상 정의가 달라진다. 따라서

한 개의 IGES 엔티티가 표현할 수 있는 형상들의 개수만큼 변환되어 생성되는 STEP 엔티티가 필요하다.

EXPRESS-X는 하나 이상의 베이스 엔티티로부터 한 개의 엔티티만을 생성할 수 있기 때문에 하나의 엔티티로부터 여러 개의 엔티티들 생성할 수 있는 방법이 필요하다. 이런 매핑을 구현하기 위해서는 동일한 IGES 엔티티로부터 각각 다른 AP202 엔티티를 생성할 수 있는 매핑 관계 정의가 생성될 수 있는 AP202 엔티티 개수 만큼 필요하다. 즉, 여러 개의 VIEW 선언문이 필요하며 각 VIEW의 WHEN절에 필요한 조건절이 포함되어야 한다.

하나의 엔티티로부터 변환되는 엔티티의 개수가 한 개인 경우에 사용되는 EXPRESS-X 매핑 언어의 구문은 Fig. 7과 동일하다. Fig. 12은 하나의 엔티티로부터 변환되는 엔티티의 개수가 여러 개인 경우에 사용되는 EXPRESS-X 매핑 언어의 구문이다. 베이스 스키마는 Fig. 11과 같이 정의하며 뷰 스키마는 STEP AP202 스키마를 사용한다. IGES의 circular_arc 엔티티는 x2, y2 그리고 x3, y3의 속성값에 따라 원 또는 호를 표현할 수 있다. 따라서 이들 각각에 대한 변환 스키마 정의가 필요하며 원인지 호인지를

```

ENTITY circular_arc;
  z1 : REAL;
  x1 : REAL;
  y1 : REAL;
  x2 : REAL;
  y2 : REAL;
  x3 : REAL;
  y3 : REAL;
END_ENTITY;

```

Fig. 11. IGES의 circular_arc 엔티티.

```

SCHEMA_MAP Mapping_Schema;
.....
VIEW cir : tdb::circle
FROM (cla : sdb::circular_arc)
WHEN ( (cla.x2 = cla.x3) AND
        (cla.y2 = cla.y3) );
BEGIN_VIEW
.....
END_VIEW;

VIEW tc : tdb::trimmed_curve
FROM (cla : sdb::circular_arc)
WHEN ( ( NOT(cla.x2 = cla.x3) AND
        NOT(cla.y2 = cla.y3) );
BEGIN_VIEW
.....
END_VIEW;

.....
END_SCHEMA_MAP;
    
```

Fig. 12. 하나의 엔티티로부터 변환되는 엔티티의 개수가 여러 개인 경우의 엔티티 매핑 스키마 정의.

구별할 수 있는 구문이 WHEN절에 정의되어야 한다. 원일 경우에는 STEP AP202의 circle로 변환되며, 호일 경우에는 trimmed_curve로 변환될 수 있도록 2개의 VIEW 선언문이 필요하다.

4.5.3 속성들 간의 변환식의 필요유무에 따른 분류

IGES 엔티티를 STEP 엔티티로 변환할 때, 두 엔티티의 구조가 동일한 경우에는 단지 IGES의 속성값을 STEP의 속성값으로 할당하면 된다. 하지만 만약 두 엔티티의 구조가 동일하지 않다면 하나 이상의 IGES의 속성값들로부터 한 개의 STEP 속성값으로 변환하기 위한 변환식이 필요하다. 위의 두 가지 경우에 대해 모두 할당문을 사용하여 엔티티의 속성값을 계산한다.

속성들 간의 변환식이 필요하지 않은 경우에는 단

순 할당문을 사용하여 매핑 스키마를 정의한다. 사용되는 EXPRESS-X 매핑 언어의 구문은 Fig. 7과 동일하다. 속성들 간의 변환식이 필요한 경우에도 할당문을 사용하지만, 이들 할당문들은 뷰 엔티티 인스턴스의 속성값을 계산하기 위하여 여러 개의 베이스 엔티티 인스턴스들의 속성값들을 변환식에서 사용한다. 사용되는 EXPRESS-X 매핑 언어의 구문은 Fig. 10에서 radius의 속성값을 계산하기 위해 사용된 할당문에서 변환식을 볼 수 있다.

지금까지 3가지 분류 기준에 따라 엔티티 변환관계를 분류하였고 각 분류에서 필요로 하는 EXPRESS-X 구문을 설명하면서 또한 예를 들어 설명하였다. 실제 IGES와 STEP AP202간, 그리고 STEP AP 간의 엔티티 변환관계는 Table 5와 같다.

Table 5. 엔티티 변환관계의 분류

| 분류기준 | | IGES → AP202 | AP202 → AP203 | |
|--------------------------------|---------------------------------|---------------|---|--|
| 기본 속성타입만을 갖는 경우 | 하나의 엔티티가 하나의 엔티티로 변환되는 경우 | 변환식이 필요 없는 경우 | Point(#116) ransformation_matrix(#124) | cartesian_transformation_operator_3d cartesian_point direction ... |
| | | 속성들간의 변환식 필요 | Line(#116) Plane(#108) Rational_b_spline_curve(#126) Rational_b_spline_surface(#128) | |
| | 하나의 엔티티가 여러 개의 엔티티로 변환될 수 있는 경우 | 변환식이 필요 없는 경우 | | |
| | | 속성들간의 변환식 필요 | Circular_arc(#100) Conic_arc(#104) | |
| 다른 엔티티를 가리키는 포인터를 속성타입으로 갖는 경우 | 하나의 엔티티가 하나의 엔티티로 변환되는 경우 | 변환식이 필요 없는 경우 | Composite_curvea(#120) Surface_of_revolution(#120) Offset_surface(#140) | composite_curve vector circle ... |
| | | 속성들간의 변환식 필요 | Tabulated_cylinder(#122) Trimmed_parametric_surface(#144) | |
| | 하나의 엔티티가 여러 개의 엔티티로 변환될 수 있는 경우 | 변환식이 필요 없는 경우 | Offset_curve(#130) Curve_on_a_parametric_surfac(#142) | |
| | | 속성들간의 변환식 필요 | | |

5. 예제 부품에 대한 형상 변환

변환시스템의 활용예를 보이기 위해 Pro/Engineer에서 IGES 파일을 생성하여 본 연구에서 구현한 변환시스템을 이용하여 STEP AP202 파일과 AP203 파일로 변환하여 이 변환된 STEP파일을 다시 Pro/Engineer에서 읽어 들이는 과정을 설명한다.

5.1 예제 부품

Fig. 13은 Pro/Engineer 버전15.0에서 생성한 IGES 파일로서 솔리드 모델로 설계된 부품이다. 본 변환시스템에서는 와이어프레임 모델만을 구현하였기 때문에 솔리드 모델이 아닌 와이어프레임 IGES 파일로 출력하여 이것에 대한 변환을 수행하였다. 예제 부품에 대한 와이어프레임 모델은 모따기(Chamfer), Protrusion, 홈(Slot) 등의 특징형상(Feature)들로 구성되어 있으며 이 특징형상들을 구성하는 엔티티들을 중심으로 변환 예를 설명한다. 이 특징형상들은 Line, Circular arc, Transformation matrix, 그리고 Rational B-Spline curve 엔티티들을 포함하고 있다. 이 예제 부품의 와이어프레임 모델에 대한 IGES 파일 크기는 4,920바이트이고 사용된 엔티티의 종류는 Line, Circular arc, Transformation matrix 등 총 3종류이다. Line 엔티티의 수는 14개이며, Circular arc 엔티티의 수는 2개, Transformation matrix 엔티티의 수는 2개이다. Transformation matrix 엔티티는 Circular arc 엔티티를 이동시키고 회전시키기 위해 사용된다. 솔리드 모델에서는 Rational B-Spline curve 엔티티가 사용되지만 와이어프레임 모델에서는 이 엔티티가 사용되지 않는다.

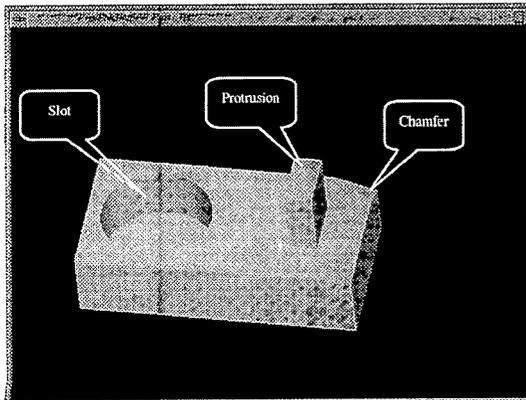


Fig. 13. 예제 부품(IGES 파일).

5.2 STEP AP202로의 변환

Pro/Engineer 버전15.0는 부품(Part) 단위 또는 조립품 단위로만 STEP 파일을 생성하고 읽어 들일 수 있으며 2차원 형상에 대한 STEP 파일을 지원하지 않는다. IGES 파일은 부품의 형상 정보 이외에도 주석, 컬러, 선의 종류, 선의 두께 등에 대한 정보도 포함하고 있으며 이들에 대한 정보들은 AP202로 변환될 수 있다. 그러나 형상 이외의 주석 정보는 AP203이 지원하지 않기 때문에 본 예제는 3차원 부품 형상만을 예를 들었다. 위에서 생성된 IGES 파일은 IGES-AP202간 변환시스템을 거쳐 STEP AP202파일이 생성된다.

예제 부품의 IGES 엔티티는 3종류로 이들 모두 STEP AP202의 line, circle, cartesian_transformation_operator_3d 엔티티로 변환된다. 또한 이들 엔티티를 나타내기 위해 필요한 direction, vector, axis2_placement_3d 엔티티 등이 변환과정에서 생성된다. 변환되어 생성된 STEP 파일의 크기는 3,224바이트이고 생성된 STEP 엔티티의 수는 총 76개이다. 생성된 STEP 엔티티의 수가 IGES 엔티티의 수보다 많은 이유는 IGES Line 엔티티 1당 4개의 STEP 엔티티가 생성되고, Circular 엔티티 1당 5개, Transformation_matrix 엔티티 1당 5개의 STEP 엔티티가 생성되기 때문이다.

5.3 STEP AP203으로의 변환

AP202와 AP203 간의 변환에서는 형상 데이터의 변환 이외에도 AP202의 제품관계를 기술하기 위해 필요한 정보들인 조직이름, 주소, 캐드파일이름, discipline_id, 제품 식별자, 재버전 식별자 등이 AP203의 구성 관리와 제품 구조를 위한 정보로 변환될

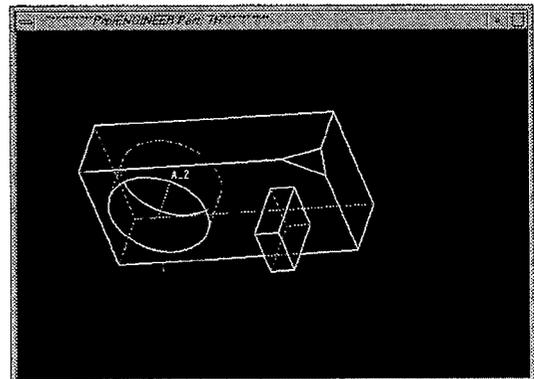


Fig. 14. Pro/Engineer에서 읽어 들인 예제 부품(STEP파일).

수 있다. 그러나 AP203은 제품 형상이외의 주석 또는 도면에 대한 정보를 포함하고 있지 않기 때문에 IGES 파일로부터 AP203 파일로의 변환은 형상 데이터에 대한 변환만을 수행하였다.

Fig. 14는 AP202 파일로 변환된 STEP 파일을 다시 AP203 파일로 변환한 후, Pro/Engineer 버전 15.0에서 읽어 들인 예제 부품이다. 위상 엔티티에 대한 변환을 하지 않았기 때문에 와이어프레임 형상으로 표시되어 나타난다. 생성된 STEP 파일의 크기는 3,223바이트이고 생성된 STEP 엔티티의 수는 AP202 파일의 엔티티의 수와 같은 76개이다. 이는 본 예제 부품에서 AP203에서 사용된 엔티티와 AP202에서 사용된 엔티티가 일치하기 때문이다.

6. 결 론

본 논문은 IGES 파일과 STEP 파일 간의 변환방법을 기술하였다. 변환 대상 파일들을 STEP의 제품 데이터 모델링 언어인 EXPRESS 스키마로 정의하였고, 변환 대상 파일들 간의 변환은 스키마 매핑 언어인 EXPRESS-X를 사용하여 변환관계를 정의하였다. 본 연구에서는 IGES와 STEP 간의 변환과 STEP 응용 프로토콜인 AP202와 AP203 간의 변환만을 제시하였지만 위와 같이 변환하고자 하는 교환 파일 포맷(예, DXF 등)을 EXPRESS 스키마로 정의한 후, EXPRESS-X를 사용하여 변환관계를 정의한다면 쉽게 데이터 변환을 할 수 있다. 또한 IGES의 데이터들 중에서 제품의 형상 데이터에 대한 변환을 구현하였으며 도면에 관련된 주석 데이터에 대한 변환관계를 포함하였다.

STEP 응용 프로토콜로서는 AP202와 AP203 간의 데이터 변환을 구현하였다. AIC를 중심으로 변환 가능한 형상 모델을 조사하였으며, 형상 데이터 변환 이외의 비형상 데이터들로 AP202의 제품관계를 기술하기 위해 필요한 정보들인 조직이름, 주소, 캐드 파일이름, discipline_id, 제품 식별자, 재버전 식별자 등과 AP203의 구성 관리와 제품 구조를 위한 정보들 간의 변환도 포함하였다. 본 연구에서 적용한 변환 방법은 3차원 객체의 디스플레이를 위한 VRML이나 3 차원 솔리드 모델과 빠른 프로토타이핑 시스템간의 인터페이스를 위한 포맷인 STL (STereoLithography)로의 변환에도 적용하여 다양한 제품 데이터의 요구를 만족시킬 수 있다.

향후 연구 과제로는 본 연구는 제품의 형상 데이터와 비형상 데이터인 선 종류, 선 폭, 컬러, 주석 데

이터 변환만을 다루었지만 제품 데이터의 완전한 변환을 위해 IGES의 위상 데이터, CSG 데이터, 다른 비형상 데이터에 대한 변환이 필요하다. 각 응용 시스템의 상호 데이터 교환을 위해 AP202와 AP203 이외의 다른 응용 프로토콜 간의 데이터 변환도 필요하다. 그리고 네트워크를 통해 이종 분산 시스템들 간의 데이터의 변환 및 공유, 그리고 교환이 이루어질 수 있도록 본 시스템을 확장시켜야 할 것이다.

감사의 글

본 연구는 1997년도 특장기초연구지원사업의 지원에 의하여 연구되었습니다(과제번호: 97-02-00-09-01-3).

참고문헌

1. Teeuw, W. B., Iicfting, J. R., Demkes, R. H. J., Houtsma, M. A. W., "Experiences with product data interchange: On product models, integration, and standardisation", *Computer In Industry*, Pages 205~221, Vol 31, Number 3, 30 November 1996.
2. ISO, TC184/SC4, 1994.
3. ISO TC184/SC4/WG5, EXPRESS-X Reference Manual, March 1996.
4. ISO, TC184/SC4, STEP Part 11: EXPRESS Language Reference Manual, 1994.
5. ISO TC184/SC4/WG5/N243, EXPRESS-M Reference Manual, August 1995.
6. ISO TC184/SC4/WG5/N251, EXPRESS-V Reference Manual, November 1995.
7. Yanping Zhang, IPPD, <http://chaos.eng.fsu.edu/~yzhang/cgi-bin/mapper.html>.
8. Martin Hardwick, David L. Spooner, Tom Rando and Morris, K. C., "Data Protocols for the industrial Virtual Enterprise", *IEEE Internet Computing*, Pages 20-29, Jan(Feb 1997).
9. 이영준, STEP을 이용한 CAD 데이터 변환시스템의 구현, 인하대학교, 자동화공학과 석사학위논문, 1995년, 8월.
10. Ibrahim Zeid, *CAD/CAM Theory and Practice*, McGraw-Hill, 1991.
11. Kent Reed, Dennette Harrod Jr., and William Conroy, *The Initial Graphics Exchange Specification (IGES) Version 5.0*, NIST, 1990.
12. STEP 연구회, 제품 모델 정보 교환을 위한 STEP, 성안당, 1996.
13. ISO, TC184/SC4, STEP Part 202: Application protocol: Associative Draughting, 1995.

- 14. ISO, TC184/SC4, STEP Part 203: Application protocol: Configuration Controlled design, 1994.
- 15. Ian Bailey, Martin Hardwick, Adrian Laud, David L. Spooner, "Overview of the EXPRESS-X Language", <http://www.cs.rpi.edu/~spooner/Paper/BHLS96.ps>, 1996.
- 16. CIMIO, "EXPRESS-M to EXPRESS-X incorporating EXPRESS-V", http://www.cimio.co.uk/STEP/exp_mxv.html.

안 만 진



1996년 인하대학교 자동화공학과 학사
 1998년 인하대학교 자동화공학과 석사
 관심분야 : STEP, CALS, CORBA, CAD/CAM

유 상 봉



1982년 서울대학교 제어계측공학과 학사
 1986년 Arizona 주립대학교 전기및컴퓨터공학과 석사
 1990년 Purdue 대학교 전기및컴퓨터공학과 박사
 1989년 AT&T Bell 연구소 연구원
 1990년 삼성전자 컴퓨터부문 선임연구원
 1992년 ~ 현재 인하대학교 자동화공학과 부교수
 관심분야 : 공학 데이터베이스, 시스템 통합, STEP, CALS, PDM, 동시공학