

기계 조립품 정보의 표현을 위한 XML 기반 공용문서 구조

정태형[#], 박승현^{*}, 윤성원^{*}

Development of Common Document Structure based on XML for Representing Mechanical Part and Assembly Information

Tae Hyong Chong[#], Seung Hyun Park^{*} and Sung Won Yun^{*}

ABSTRACT

In engineering design environment it is hard to link design data and systems because the types of them are disparate. Therefore, the importance of metadata has increased. Some researches have been executed to develop metadata. But they cannot interact with other metadata and are difficult to extend. The purpose of this paper is to develop a common document structure which represents the general information of mechanical part assembly using XML, and to use it as base documents in order to integrate design data and systems. It is composed of part, assembly and user documents. Part document represents the information of a part independently to part type. Assembly document represents the location of constituent part documents. User document represents user's information. Common documents can be used as a broker between design data and systems, and it can improve the interpretability and reusability of document. We applied the developed common document structure to 2-stage spur gear drive.

Key Words : Design Methodology (설계 방법론), Design Information (설계 정보), Metadata (메타데이터), XML (eXtensible Markup Language)

1. 서론

일반적으로 공학 설계 환경에는 다양한 설계 데이터와 상이한 설계 시스템들이 혼재 되어 있다. 이러한 설계정보들은 일반문서, CAD 파일 또는 PDM(Product Data Management) 등 다양한 형식으로 표현되어 있고, 이러한 정보를 사용하는 시스템들 또한 매우 상이해서 설계 데이터 또는 시스템 간의 연동에 어려움이 있다. 이러한 문제를 해결하

기 위해 여러 가지 연구들이 수행 되고 있는데 특히 표준 설계 데이터 형식의 개발에 관한 연구와 특정 분야의 설계 정보를 표현하는 메타데이터(metadata)의 개발에 관한 연구가 주를 이룬다.

표준 설계 데이터 형식의 개발은 STEP(Standard for the Exchange of Product Model Data)¹에서 이루어지고 있는 기계공학 전반에 걸친 표준 형식의 개발을 예로 들 수 있다. 이러한 연구들은 시스템 전반에 걸친 단일 표준 형식을 개발함으로써 설계

접수일: 2003 년 3 월 25 일; 게재승인일: 2003 년 7 월 11 일
교신저자: 한양대학교 기계공학과
E-mail: thchong@hanyang.ac.kr
* 한양대학교 대학원

데이터와 시스템 간의 통합 및 연동을 추구하고 있다. 이에 비해 특정 분야의 메타데이터 개발은 표준 형식만으로는 해결될 수 없는 특정 분야의 설계 정보를 기술 및 공유하기 위해 각각의 메타데이터 형식을 개발하는 것에 대한 연구가 주를 이루고 있다. 그러나 대부분의 연구들이 특정 기계 설계 활동이나 기계요소내 국한된 메타데이터를 개발하기 때문에 다른 메타데이터들과 상호운용에 문제가 있다. 즉 각각의 메타데이터가 상호호환성이 부족함으로써 새로운 기계요소의 정보를 표현하기 위해서는 새로운 메타데이터 형식이 개발되거나 기존의 메타 데이터를 확장해야 한다.

이에 본 연구에서는 일반적인 기계 조립품의 정보를 기술할 수 있는 일반화된 메타데이터 형식을 개발하여 메타데이터 간의 교환과 변환에 필요한 노력을 줄이고, 이를 이용하여 다양한 설계 시스템들의 통합을 위한 기반을 구축하는 것을 목적으로 한다.

2. 설계 정보의 메타데이터화

2.1 메타데이터의 필요성

일반적인 의미로 메타데이터는 데이터에 관한 데이터, 혹은 전자 자원을 기술하는데 사용되는 데이터요소로서 이용자들에게 효과적인 자원 사용에 도움을 줄 수 있는 지적인 요소로 정의 되어진다. 메타데이터의 대표적인 예가 도서관의 목록카드인데, 각각의 도서가 데이터라면 목록카드의 도서들을 여러 가지 형태로 검색을 가능하게 해주는 메타데이터에 해당된다. 이는 공학 설계에서도 비슷하게 적용될 수 있다. 특정 기계요소의 정보는 매우 다양한데, 그러한 정보를 기술하는 메타데이터는 대상 기계요소의 설계 정보들을 체계적으로 기술하고 접근할 수 있도록 함으로써 설계 활동지원을 위한 다양한 시스템에서 설계 정보를 쉽게 제공할 수 있도록 한다. 이와 같이 메타데이터의 도입은 컴퓨터를 기반으로 다양한 설계 정보들의 기술, 관리 및 이용을 자동화함으로써 설계활동의 자동화를 높이는데 기여할 수 있다.

메타데이터를 표현하는 방법에는 여러 가지가 있는데, 초창기에는 자신만의 형식을 직접 개발해서 사용하는 경우가 많았다. 그러나 XML²이 개발된 이후로는 편리성 및 상호운용성을 위하여 자신만의 형식보다는 XML 과 같은 마크업 언어들이

선호되고 있다.

2.2 XML의 개요

XML은 1998년 W3C(World Wide Web Consortium)에서 SGML(Standard Generalized Markup Language)보다 간결하면서 HTML(Hypertext Markup Language)보다는 구조적인 마크업 언어 개발의 산물로서 제안된 메타 마크업 언어이다. XML의 가장 큰 특징은 HTML과는 달리 사용자가 태그를 임의로 정의할 수 있어서 데이터의 구조와 의미를 그대로 유지할 수 있는 장점이 있다. 각 적용 분야에 맞는 태그를 정의함으로써 XML을 이용하여 복잡한 정보를 구조적으로 표현할 수 있다. 따라서 XML은 문서이면서도 데이터 컨테이너의 특성을 가진다. XML은 기본적으로 텍스트 문서이므로 플랫폼과 시스템에 독립적이다. 또한 XML은 유니코드를 사용하므로 다양한 문자를 표현할 수 있고 언어에 중립적이다. 따라서 XML은 상이한 플랫폼, 시스템간의 정보 교환에 매우 유리하여 높은 상호운용성을 가지고 있다.

XML은 자신만의 태그를 정의할 수 있으므로 같은 정보를 포함하고 있더라도 정의하는 방법에 따라서 여러 가지 형태의 XML 문서가 생성될 수 있다. XSLT(eXtensible Stylesheet Language Transforming)³는 XML 문서를 다른 문서 형태로 변환하기 위한 목적으로 개발된 XML 기반의 기술이다. XSLT를 이용하면 하나의 XML 문서를 다른 형태의 XML 문서, HTML, 심지어는 복잡한 형태의 텍스트 문서로 쉽게 변환할 수 있어서 서로 다른 형태의 문서를 사용하는 시스템간의 정보 교환을 보다 용이하게 한다.

2.3 관련연구

공학 설계환경에 XML을 도입하려는 연구들이 많이 수행되고 있다. 1999년 Dirk Hagemann⁴은 공학 설계 환경에 XML의 도입을 제안하였고, 무기체계 수명 주기 자료들에 대한 교환과 공유 전략인 PDI(Product Data Interoperability) 프로그램의 일환으로 제품 정보를 표현하기 위한 XML 기반의 PDML(Product Data Markup Language)⁵이 개발되었다. Dharmaraj Veeramani와 Ramarao Garrikapati⁶는 MML(Machining Markup Language)과 FML(Foundation Markup Language)을 정의하고 제작과 주조에 관한 메타데이터의 통합을 시도하였다. 본 저자⁷ 등은

웹 기반 환경에서 기어장치의 설계 정보를 표현하는 XML 문서를 개발하였다.

이와 같이 XML 을 공학 설계 환경에 적용하려는 연구들이 많이 수행되었지만 대부분의 연구들이 특정 기계 설계 활동이나 기계요소로 국한된 메타데이터를 개발했기 때문에 다른 메타데이터들과 상호운용이 어려운 문제를 안고 있다. 또한 이러한 메타데이터들은 낮은 수준의 확장성을 제공한다. 단일 XML 문서 안에 정보를 표현하는 것에 대한 연구가 주가 되기 때문에 새로운 설계 정보를 기술하기 위해서는 새로운 메타데이터를 개발해야 한다. 이와 같이 기존 연구들에서는 개발된 메타데이터들이 상호운용성과 재사용성의 한계를 가지고 있다.

3. 공용문서 구조의 개발

3.1 공용문서 구조

본 연구에서 개발되는 공용문서는 다른 메타데이터와의 상호운용성 및 문서의 재사용성을 높일 수 있도록 세 가지 연구 방향을 설정하였다. 첫째, 개발되는 공용문서는 기계요소에 중립적이어야 한다. 특정 기계요소에 국한되지 않고 일반적인 기계요소의 정보를 표현할 수 있어야 계속해서 새로운 기계요소들의 정보를 표현하는 것이 가능하다. 둘째, 개방적인 구조여야 한다. 하나의 XML 문서만을 이용한다면 파일이 크기 때문에 모든 정보를 표현할 수도 없을 뿐만 아니라 새로운 정보가 추가될 때마다 문서 구조를 새로 만들어야 하는 문제를 가지고 있다. 따라서 개발되는 공용문서는 정보가 변경되거나 추가되어도 문서 구조가 바뀌지 않으면서 확장이 될 수 있는 개방형 구조를 가져야 한다. 셋째, 다른 메타데이터와 상호운용성을 가져야 한다. 기존의 메타데이터와의 호환 및 연동을 위해서는 기존의 메타데이터로부터 정보를 가져오거나 반대로 내보낼 수 있도록 함으로써 기존의 환경을 지원하여 새로운 문서 구조로의 변경 시 필요한 노력을 최소화되어야 한다.

세 가지 요구조건을 모두 만족하기 위한 공용문서 구조를 Fig. 1 에서 보여주고 있다. 공용문서는 부품 문서(part document)와 조립품 문서(assembly document) 그리고 사용자 문서(user document)로 구성되는 하나의 문서 그룹으로서 각각의 문서는 물리적으로 분리된 XML 문서로 되어있다. 부품

문서는 단일 기계부품의 정보를 표현하고 조립품 문서는 조립품을 구성하는 부품 문서들의 정보를 가지고 있다. 사용자 문서는 부품과 조립품 문서에 반복적으로 사용되는 사용자의 정보를 따로 분리하여 만든 문서이다.

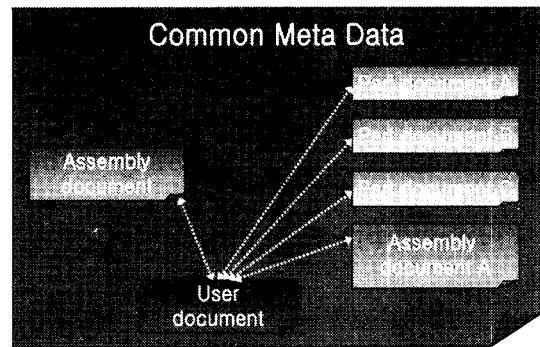


Fig. 1 Structure of common document

부품과 조립품 문서구조는 CAD 에서 부품과 조립품 파일의 구조와 유사한 형태를 취하고 있다. CAD 에서 여러 개의 기계부품을 조합하여 하나의 완성된 조립품을 만들듯이 설계 정보 역시 각각의 기계부품의 정보를 표현하는 XML 공용문서를 조합하여 조립품의 정보가 표현된다. 여기서 부품 하나의 기계요소가 XML 부품 문서에 대응되고 이러한 문서들이 조합되어 조립품 문서가 생성된다. 이러한 구조는 설계자가 개념적으로 이해하고 있는 조립품의 구조와도 일치하여서 설계자들이 빠르게 이해할 수 있다.

이와 같이 문서 구조를 정의한 것은 특정 기계요소에 국한되지 않는 일반적인 기계 조립품의 정보를 표현하기 위함이다. 단일 기계 부품은 일단 완성되면 그 자체가 확장되지 않는 하나의 개체로 생각할 수 있다. 한번 정해진 부품은 제원 정보 등이 바뀔 수는 있지만 새로운 정보가 추가되지는 않는다. 만약 새로운 정보가 추가된다면 그 부품은 기존에는 없었던 새로운 부품으로 간주되고 이 역시 하나의 개체로 생각될 수 있다. 따라서 하나의 부품을 단일 XML 문서로 표현하는 것은 개념적으로도 부합할 뿐만 아니라 확장성이라는 측면에서도 문제가 되지 않는다.

조립품 문서는 부품 문서들의 정보가 직접 포함하지 않고 단지 부품 문서들의 위치와 관계 및 관리를 위한 정보만을 기술함으로써 특정 조립품

에 국한되지 않는 일반성을 가지게 된다. 따라서 부품 문서들만 존재한다면 어떠한 조립품이든지 표현이 가능하고 조립품 문서가 다른 조립품의 부품으로도 사용될 수 있어서 여러 개의 조립품으로 구성된 복잡한 조립품의 정보도 표현할 수 있다. 또한 문서를 물리적으로 분할함으로써 중복되는 정보를 최소화하여 문서의 크기를 줄이고 검색 속도를 향상시키는 효과를 얻을 수 있다.

3.2 부품 공용문서 구조

부품 문서에 기술되는 정보들의 구조를 Fig. 2에서 보여주고 있다. 부품 문서는 일반적인 기계 부품들의 정보를 나타낼 수 있도록 정의되었기 때문에 모든 기계 부품의 부품 문서들을 동일한 방법으로 접근 및 사용할 수 있다.

부품 문서는 <doc_info> 엘리먼트와 <part_info>, <resources> 엘리먼트로 구성되어 있다. <doc_info> 엘리먼트에는 부품 정보를 관리하는데 필요한 정보들이 기술되고, <part_info> 엘리먼트에는 부품의 설계 정보가 기술된다. 부품 문서에서는 일반적인 기계 부품의 형상정보를 표현하기 위해 제원을 이용하는데, <dimensions> 엘리먼트 안에 형상정보를 “<dimension name=“형상정보이름” unit=“단위” value=“값”/>”의 형식으로 기술 하도록 하고 있어서 기계 요소의 종류에 상관없이 단일한 방법으로 기술 되도록 하였다. 예를 들어 기어의 형상 정보 중 치폭은 “<dimension name=“face_width” unit= “mm” value=“8”/>”와 같이 기술 된다. 부품문서에서 형상정보 표현을 위해 제원을 이용하는 것은 제원을 이용하여 형상을 재구성하는 것이 가능할 뿐만 아니라 제원을 이용하는 해석 시스템에게 필요한 정보를 제공함으로써 수식 기반의 해석 시스템들과 연동할 수 있기 때문이다. <type> 엘리먼트에는 레벨(level)이라는 속성을 두어서 다양한 기계요소를 분류할 수 있게 하였다. <resources> 엘리먼트는 해당 부품의 CAD 파일이나 문서 등의 위치 정보를 기술한다. 부품 문서에는 형상 정보와 관련하여 제원만이 기술되고 기하학적인 형상은 관련 CAD 모델에 위임함으로써 필요에 따라 제원과 기하학적 정보를 이용할 수 있게 하였다.

3.3 조립품 공용문서 구조

Fig. 3은 조립품 문서에 기술되는 정보들을 보여주고 있다. 조립품 문서는 부품 문서와 마찬가지로

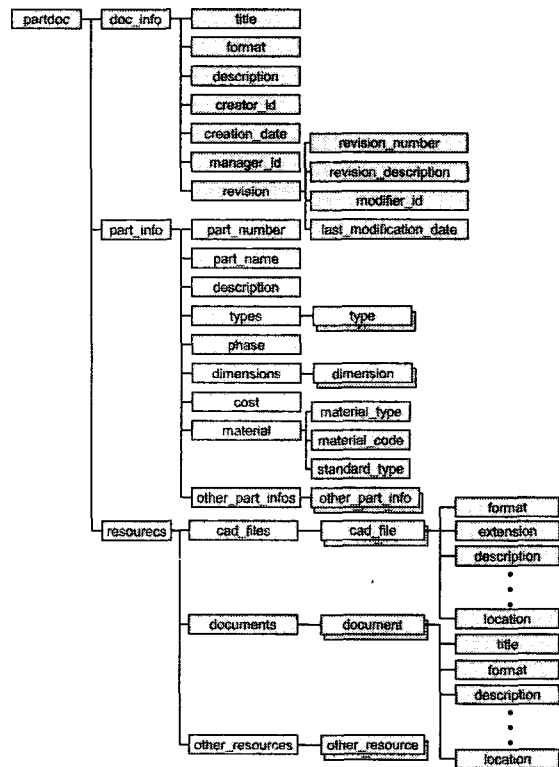


Fig. 2 Structure of part document

지로 조립품 문서를 관리하기 위해 필요한 정보들이 기술되어 있는 <doc_info> 엘리먼트와 조립품을 구성하고 있는 부품의 연결 정보가 기술되어 있는 <assembly_info> 엘리먼트 그리고 조립품에 관련된 CAD 파일이나 문서 파일의 위치를 알려주는 <resources> 엘리먼트로 구성되어 있다.

<doc_info> 엘리먼트는 부품 문서와 같은 구조를 가지고 있다. 조립품 문서의 <assembly_info> 엘리먼트는 <types>과 <dimensions>, <material> 엘리먼트가 생략되고 조립품 정보를 기술하는데 필요한 <connections>와 <subparts> 엘리먼트가 추가된다. <connections> 엘리먼트와 <subparts> 엘리먼트는 기계요소들의 관계를 나타내는데, <connections> 엘리먼트는 직접적으로 연결되어 있는 기계요소들에 대한 정보를 기술하는 부분이다. 형식은 <connections> 엘리먼트 안에 하위 요소인 <connection> 엘리먼트를 추가하고 그 엘리먼트 안에는 부품번호(part_number)와 타입(type) 그리고 공용문서의 위치를 기술하는 위치(location) 속성들이 기술되어 있다. 형식은 “<connection part_number =“부품번호” type=“타입”

location="공용문서위치" />"로 기술된다. 이렇게 함으로서 여러 개의 부품과 조립품의 연결 정보를 한번에 표현이 가능하여 높은 확장성을 가지게 된다. <subparts> 엘리먼트는 조립품을 구성하고 있는 하위 부품들의 정보를 기술하는 것으로 형식은 <connections> 엘리먼트와 비슷하나 number 라는 속성이 하나 더 포함되어 있다. 이것은 같은 부품이 하나의 조립품에 여러 개가 들어갈 수 있으므로 그런 부품들의 개수를 나타내는 것으로 부품의 정보를 중복으로 표기할 필요가 없어지게 된다. 조립품 문서 안에 직접 부품이나 조립품 정보가 포함 되는 것이 아니라 해당하는 부품이나 조립품의 위치정보만을 기술함으로써 문서의 크기를 줄일 수 있고 관련 부품들의 정보를 추적할 수 있다. <resources> 엘리먼트는 부품 문서의 <resources> 엘리먼트와 같은 역할을 하며 구조 또한 같다.

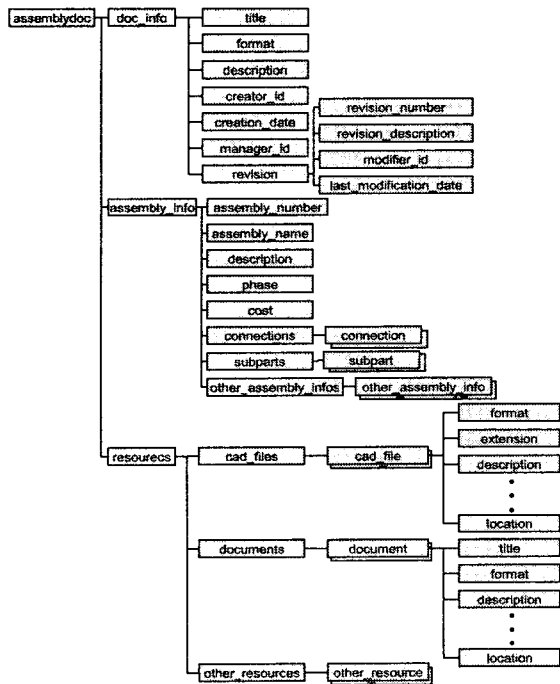


Fig. 3 Structure of assembly document

3.4 사용자 공용문서 구조

사용자 문서는 부품 문서나 조립품 문서에서 반복되는 인적 사항을 따로 분리하여 정의한 것이다. 사용자 문서에는 <users> 엘리먼트 아래에

<user> 엘리먼트들이 위치하고 이곳에 설계자의 구체적인 정보가 기술된다. <user> 엘리먼트에는 이름, 기관명, 부서, 전화번호 등 인적 사항에 필요한 정보들이 하위 엘리먼트로 정의 되어있고, 각각의 <user> 엘리먼트에는 고유번호(ID)를 가지고 있어 부품 문서나 조립품 문서에는 인적 사항이 모두 기술 되는 것이 아니라 고유번호로 표시하여 식별할 수 있게 하였다.

4. 공용문서의 역할과 활용방안

본 연구에서 개발한 공용문서는 Fig. 4 와 같이 시스템 및 메타데이터간의 중간자 역할을 수행한다. 기존의 환경에서는 각각의 시스템과 데이터가 직접 통신하는 구조를 취하고 있어서 새로운 시스템을 추가할 때마다 다른 모든 시스템과 통신할 수 있도록 어댑터를 개발하거나 기존의 시스템을 변경해야 하기 때문에 확장성과 변경 용이성이 매우 떨어진다. 그러나 공용문서를 기반 문서로 활용하면 모든 시스템과 메타데이터간의 중간자 역할을 수행함으로써 각 시스템간의 통신을 원활하게 하여 높은 확장성과 변경 용이성을 제공한다. 공용문서를 기반으로 개발되는 새로운 시스템들은 시스템의 추가 및 변경 시 변경의 영향을 최소화 하면서 설계 환경을 계속적으로 확장시킬 수 있다 또한 단일 형식으로 여러 가지 기계 요소의 정보를 기술할 수 있어 메타데이터 변환에 드는 노력을 최소화시킬 수 있다.

또한 공용문서 구조에서 부품 문서는 그 자체로 하나의 부품에 관한 필수 정보들을 기술하고 있는 단위 문서로 부품 문서들을 모아서 관리하던 공학 설계 환경에서의 표준 부품 데이터베이스로서도 사용할 수 있다.

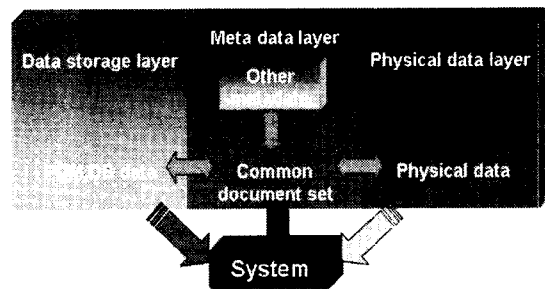


Fig. 4 Role of common document

5. 공용문서의 적용

개발한 공용문서 구조를 2 단 스피어 기어장치에 적용하였다. Fig. 5 는 적용하는 2 단 스피어 기어장치의 조립 구조로 기어, 기어 샤프트, 키, 축, 베어링, 기어박스로 구성되어 있다.

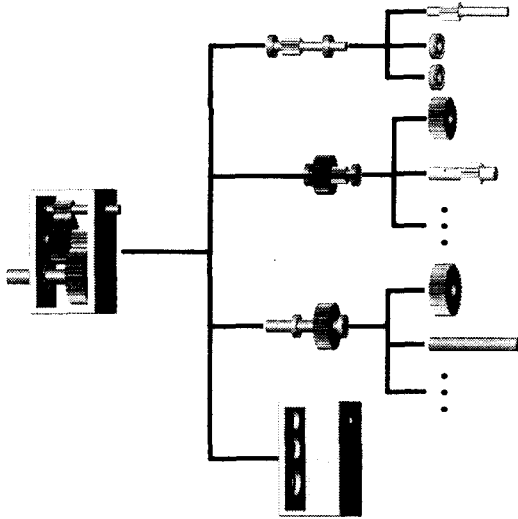


Fig. 5 Structure of 2-stage spur gear drive

5.1 2 단 기어장치 정보의 기술

Fig. 6 과 같이 문서들을 생성하여 부품 및 조립품 정보를 기술하였다. 여기서 Fig. 6 의 공용문서

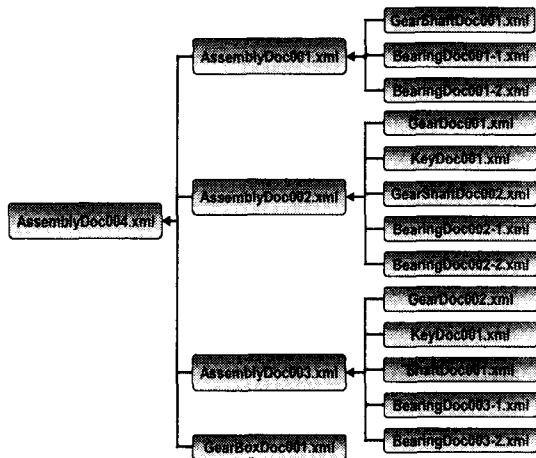


Fig. 6 Structure of common document for 2-stage gear drive

```
<?xml version="1.0" encoding="UTF-8"?>
<partdoc id="geardoc001" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="http://gearlab.hanyang.ac.kr/sungwon/part.xml">
  <doc_info>
    <title>First Stage Gear Part Document</title>
    <format>text/xml</format>
    <description>Gear of first stage</description>
    <creator_id>2000503338</creator_id>
    <creation_date>2002-10-1</creation_date>
    <manager_id>2000600831</manager_id>
    <revision>
      <revision_number>1.00</revision_number>
      <revision_description>first version</revision_description>
      <modifier_id>2001550854</modifier_id>
      <last_modification_date>2002-10-1</last_modification_date>
    </revision>
  </doc_info>
  <part_info>
    <part_number>gear001</part_number>
    <part_name>gear</part_name>
    <description/>
    <types>
      <type level="1" value="gear"/>
      <type level="2" value="spur gear"/>
      <type level="3" value="driven gear"/>
    </types>
    <phase>prototype</phase>
    <dimensions standard_type="ISO">
      <dimension name="nomal_module" unit="module" value="0.8"/>
      <dimension name="helix_angle" unit="deg" value="0"/>
      <dimension name="pressure_angel" unit="deg" value="25"/>
      <dimension name="addendum" unit="module" value="0.8"/>
    </dimensions>
  </part_info>
</partdoc>
```

Fig. 7 Content of GearDoc001.xml

구조가 Fig. 5 의 2 단 기어장치의 구조와 일치함을 알 수 있다. 2 단 기어장치의 표현을 위해 부품과 조립품 문서 및 사용자 문서가 사용되었고, Fig. 7 은 1 단 기어(gear)의 정보를 기술한 문서의 일부이다.

Fig. 7 에서 보여주고 있는 부품 문서에는 이 문서의 제목(title)과 형식(format) 등의 정보와 부품의 형상 정보 등이 기술되어 있다. 개발된 공용문서는 일반적인 모든 부품들의 정보를 표현하기 위한 구조로 만들어졌기 때문에 구조의 변화가 없이 각 부품의 값만 바꾸어 기술하면 된다. 2 단 기어 장치에 포함되는 모든 부품문서는 동일한 구조와 방법으로 표현하였다. 조립품 문서에는 조립품을 구성하는 부품이나 또는 직접적으로 접하고 있는 부품의 정보 등이 기술되었다. 이때 부품 문서의 위치 정보를 기술함으로써 관련 부품의 정보를 쉽게 가져올 수 있게 하였다. 2 단 기어장치를 위한 다른 조립품들도 같은 구조와 방법으로 표기하였

고, 사용자 문서에는 각각의 엘리먼트에 사용자 정보를 기술하고 있어서 반복되는 사용자 정보를 따로 분리하고 중복되는 정보를 최소화 하였다.

5.2.2 단 기어장치의 공용문서 활용

2 단 기어장치의 부품과 조립품 문서 정보를 기반으로 하여 시스템을 만들어 공용문서의 활용 방안을 확인하였다. 먼저 문서들 안의 필요한 정보를 추출하여 다른 형식으로 표현해 보고, 또한 공용문서를 가지고 기존 시스템과의 연동성을 확인하였다.

5.2.1 공용문서 에디터

공용문서 에디터는 공용문서 기반으로 만든 시스템으로서 공용문서를 한눈에 알아볼 수 있도록 XML 파일을 트리 구조로 만들어 보여주며, 기존의 공용문서의 값을 변경할 수 있는 에디터 기능까지 수행 할 수 있어서 쉽고 편리하게 문서의 편집 및 생성 작업을 수행할 수 있다. 또한 기존에 만들어진 공용문서에서 <resources> 엘리먼트에 표현된 CAD 와 문서 정보에서 부품과 조립품의 파일에 대한 위치 정보를 추출하여 공용문서 에디터의 브라우저를 통해서 직접 부품이나 조립품을 확인할 수 있다. Fig. 8 은 공용문서 에디터의 실행 화면이다. Fig. 9 는 2 단 기어장치에서 1 단 축과 피니언 일체형(gearshaft001)인 부품의 3D 모델링을 공용문서 에디터의 브라우저로 보여주고 있다. 이것은 기존에 있는 CAD 파일의 위치 정보만을 이용하여 공용문서의 브라우저를 통해서 파일을 열어 보여주는 것이다. 같은 방법으로 CAD 2D 도면이나 문서, 사진까지 확인할 수 있다.

5.2.2 정보 추출

여러 개의 부품과 조립품으로 이루어진 2 단 기어장치에서 필요한 정보만을 추출하여 사용할 경우 XSLT 를 이용하여 기존의 공용문서에서 필요한 정보만을 추출 할 수가 있다. Fig. 10 은 조립품 assembly003 의 구성 부품인 2 단 기어와 축, 키, 베어링의 정보를 기술한 부품 문서들에서 XSLT 를 이용하여 치수(dimension) 정보들을 추출하여 웹 상에서 나타내기 위해 HTML 문서로 변환하는 과정을 보여주고 있다. XSLT 는 HTML 형식의 파일로 변형할 수 있을 뿐만 아니라 다른 구조의

XML 이나 텍스트 문서 등으로도 변환 가능하다. 이와 같이 XSLT 를 이용함으로써 플랫폼(platform)과 프로그램 언어에 독립적으로 다른 형식으로 쉽게 변환 가능하다.

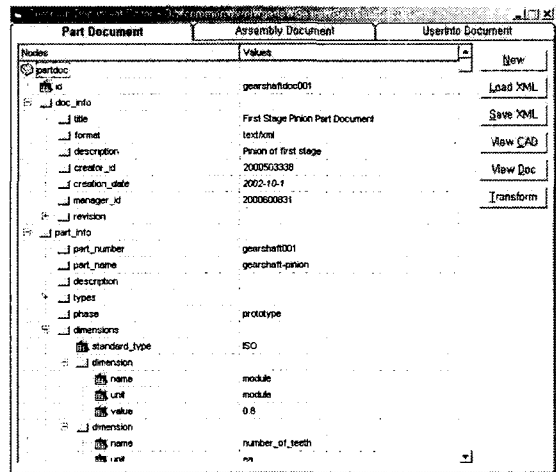


Fig. 8 Common Document Editor

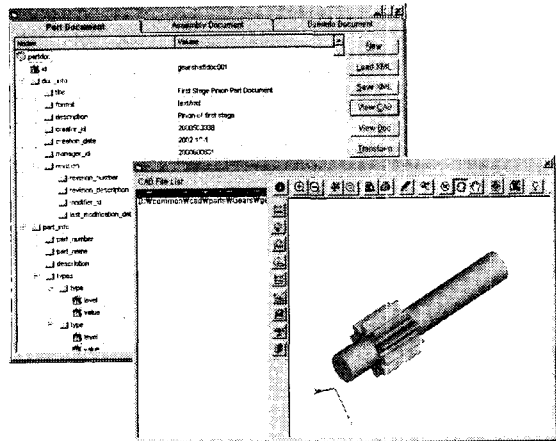


Fig. 9 3D Modeling of gearshaft001

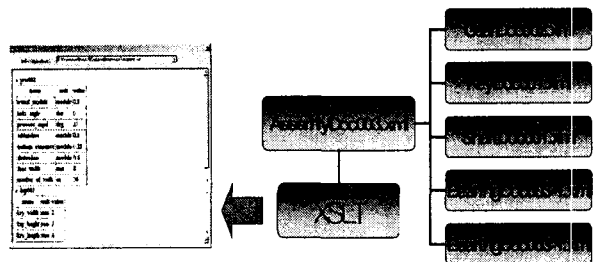


Fig. 10 Transformation of common document

5.2.3 기존 시스템과 연동

기존의 시스템과의 연동성을 보여주기 위해서 Fig. 11 과 같이 GMA(Gear Modeler Agent)⁸를 공용 문서와 연동하였다. GMA 는 기어 설계 정보를 위한 메타데이터 형식인 Gear XML 을 입력 받아 기어 모델(gear model)을 생성하는 시스템으로 저자들의 이전 연구에서 개발된 것이다. GMA 의 입력으로 Gear XML 을 받게 되어있는데, 예제로 사용된 2 단 기어장치의 공용문서 중에서 2 단 기어(gear)에 해당하는 부품 문서를 XSLT 를 이용하여 Gear XML 문서로 변환하고, 이를 GMA 에 전송하여 해당 기어의 3D 형상을 생성하였다. 이와 같이 공용문서는 다른 메타데이터 형식으로 쉽게 변환이 가능하므로 표준 형식으로서 기존의 메타 데이터를 대체하면서도 기존의 환경과의 연동성을 유지할 수 있는 장점을 가진다.

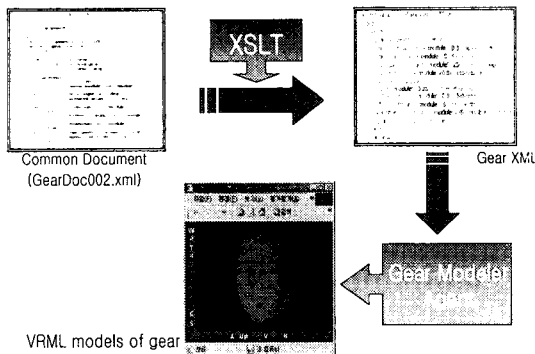


Fig. 11 System Integration

6. 결론

본 연구에서는 XML 을 이용하여 일반적인 기계 조립품의 설계 정보를 표현하는 부품 문서와 조립품 문서 그리고 사용자 문서로 구성되는 공용 문서 구조를 개발하고 이를 2 단 기어장치에 적용하였다.

개발된 공용문서는 실제 조립품의 부품 구성과 동일한 구조를 취하고 있어 특정 조립품에 국한되지 않고 일반적인 조립품에 모두 대응될 수 있는 특징을 가지고 있다. 특히 부품 문서는 기계 부품의 종류에 상관없이 동일한 구조로 설계 정보를 표현할 수 있어서 새로운 부품이 추가 되어도 문서의 구조가 변경되지 않고 표현이 가능하다.

공용문서는 제원을 이용하여 형상 정보를 기술하고 기하학적인 정보를 가지고 있는 CAD 모델과 연동될 수 있게 함으로써 기존의 PDM 및 CAD 시스템 이외에 수식을 이용한 해석 시스템들과의 연동도 가능하게 해준다. 개발된 공용문서를 설계 환경의 기반 문서로 사용하게 되면 기존의 시스템과의 연동을 확보하면서도 확장성과 변경의 용이성을 가지는 설계 환경을 구축할 수 있다. 또한 공용문서가 지속적인 확장이 가능하므로 차후 새로운 정보가 추가되더라도 이전 문서와 하위 호환성을 가질 수 있다.

참고문헌

1. STEP, Standard for the Exchange of Product Model Data, <http://cic.vtt.fi/links/step.html#parts>.
2. XML, Extensible Markup Language, <http://www.w3c.org/XML>.
3. XSLT, XML Stylesheet Language Transformations, <http://www.w3c.org/TR/xslt>.
4. Hagemann, Dirk, "XML and JAVA - Engineering Software Development Meets Internet Technologies," ASME Design Engineering Technical Conferences, Las Vegas, Nevada, September 12-15, 1999.
5. PDML, Product Data Markup Language <http://www.pdml.org>.
6. Veeramani, Dharmaraj and Grarikapati, Ramarao, "XML Based Markup Language for Knowledge Management in Manufacturing," ASME 2000 Design Engineering Technical Conferences and Computers and Information in Engineering Conference Baltimore, Maryland, September 10-13, 2000.
7. Chong, Tae Hyong and Ahn, June Young, "Structured Representation of Design Information for Gear Drives in XML," Proceedings of the 2000 Korean Society of Machine Tool Engineers Conference, Changwon, Kyeongnam, April 15, 2000.
8. Chong, Tae Hyong and Park, Seung Hyun, "Integration of Gear Design Data using XML in the Web-based Environment," Proceedings of the Korean Society of Precision Engineering Conference, pp 627-630, May 2001.