

大韓造船學會論文集  
 第28卷 第2號 1991年 10月  
 Transactions of the Society of  
 Naval Architects of Korea  
 Vol.28, No.2, October 1991

## 관계형 데이터모델을 이용한 선체구조의 표현에 관한 연구

김용대\*, 이규열\*, 서승완\*, 류호율\*\*, 홍창호\*\*

### A Study on the Relational Data Model for the Representation of Ship Structure

by

Y.D.Kim\*, K.Y.Lee\*, S.W.Seo\*, H.Y.Ryu\*\* and C.H.Hong\*\*

#### 요 약

지금까지 선체구조 설계 관련 정보는 주로 도면을 통해서 전해져 왔으나 최신의 컴퓨터 관련 기술인 CAD 시스템이나 DBMS등을 이용하면 설계과정에서의 중복작업을 없애므로써 설계생산성을 크게 높일 수 있다. 본연구에서는 선체구조 설계 관련 정보를 데이터 베이스관리시스템을 통하여 일관화하기 위해서 선체 중앙부 구조부재들의 배치형상과 속성정보들에 대한 데이터 모델링을 시도하였으며, 그 적용예로서 산적화물선의 중앙부 구조형상을 컴퓨터 그래픽을 이용하여 가시화 하였다. 데이터모델의 구조는 관계형으로 하였으며 선체구조의 접속관계정보를 이용하여 구조부재의 배치형상처리를 용이하게 하였다. 결과로서 나온 데이터 관계형 데이터베이스 관리시스템 'ORACLE'을 통해 저장되고 질의어를 통해 다른 시스템에서 직접 사용될 수 있음을 보였다.

#### Abstract

The informations of ship structural design have been represented as the form of various drawing plans, but to enhance the design productivity, it is necessary to utilize the advanced technologies such as CAD system and database management system. In this study, efforts are focused on the data modeling of those informations of midship configuration and attribute for the integration of ship structural design programs through database system. As actual example, midship configuration of bulk carrier was visualized by computer graphics. Relational data model is employed and topological information is used for the best treatment of geometries representing structural arrangement. RDBMS 'ORACLE' was used for the implementation of the results.

발표: 1990년도 대한조선학회 추계연구발표회 ('90. 11. 10.)

접수: 1990년 12월 12일, 재접수일자: 1991년 4월 2일

\* 정회원, 한국해사기술연구소

\*\* 정회원, 충남대학교 선박해양공학과

## 1. 서론

선체구조설계는 복잡한 규정의 적용과 공학적 해석과정을 거치고 많은 인력과 시간이 소요되는 고도의 전문적인 업무로서 전산화에 의한 업무처리의 단순화 및 생산성 향상이 크게 기대되는 분야이다. 그러나 아직까지는 부분적인 업무의 전산화에 그치고 있는데 이는 선체구조가 갖는 복잡한 형상을 잘 처리할 수 없었기 때문이며 선박의 설계와 건조과정에서 요구되는 정보들을 의미있는 데이터 모델로 만들어 효율적으로 이용하는 기술과 이를 지원하는 컴퓨터의 뒷받침이 부족했던 때문이다. 이러한 이유로 인해 설계 각 작업단계에서 사용되는 시스템 사이에서 정보의 자동교환이 잘 이루어지지 않으며, 제품에 대한 각종 정보가 일관되게 관리되지 못하고 이용자에 의해 정보가 중복 입력되는 경우가 많았다. 그러나 최근 일반화되고 있는 workstation 수준의 컴퓨터는 형상처리 기능이 크게 강화되고 또 대용량의 데이터 처리를 쉽게 해결할 수 있게 해주는 데이터베이스 관련 소프트웨어 기술도 크게 발달되어 설계 업무의 전산화에 밝은 전망을 보여주고 있으며, 그를 이용한 설계 일관시스템의 개발이 활발히 추진되고 있다 [1-4].

본 연구에서는 지금까지 설계도면을 통해서 전달되던 선체구조 설계 관련정보를 관계형 데이터모델을 이용하여 서로 유기적인 관계를 갖는 전산 데이터화 함으로써 각 설계단계에서 일관성이 유지된 데이터 관리를 가능케하고 효율적인 업무처리를 이루고자 하였다.

## 2. 관계형 데이터 모델

### 2.1 일반

데이터 모델이란 데이터를 정의하고 데이터들 간의 관계(relation)를 규정하며 데이터의 의미 및 데이터에 가해진 제약조건을 기술하기 위한 개념적 요구들의 집합이라 할 수 있으며 이들은 세가지 집단으로 분류된다[5-6].

- 객체에 근거한 논리적 모델
- 레코드에 근거한 논리적 모델
- 물리적 데이터 모델

여기에서는 데이터 모델을 객체에 근거한 논리적 모델과 레코드에 근거한 논리적 모델로 개념을 한정시켜 정의하기로 한다.

#### (1) 객체에 근거한 논리적 모델

객체에 근거한 논리적 모델은 개체-관계형 모델(entity-relationship model)을 사용하여 표현하고자 하는 실세계를 개념적으로 나타내어 그들 사이의 관계를 보다 정확하게 정의하는데 사용되며 나중에 레코드에 근거한 논리적 모델의 설계 사양이 된다. 여기에서는 어떤 대상을, 즉 객체(object)를 표현하기 위해, 그것을 다른 객체로 부터 구분할 수 있도록 하는 개체(entity)와 그 개체를 설명해줄 수 있는 속성(attribute)으로서 구별하고 개체간의 관계를 분석하여 도식화 함으로써 실세계에서 존재하는 데이터 사이의 관계를 구체화 한다. 예를 들어 어떤 선박에 관한 자료를 그 소유주와 건조한 조선소에 대한 자료와 관련해서 나타내면 Fig.1과 같으며 그림에서 □는 개체를, ○은 속성을, 그리고 ◇는 개체간의 관계를 의미한다.

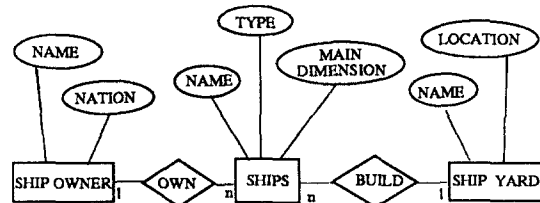


Fig. 1 Concept of E-R Diagram

#### (2) 레코드에 근거한 논리적 모델

레코드에 근거한 논리적 모델은 데이터를 개념적 설계 단계와 논리적 단계에서 기술하는데 사용되며 데이터 베이스 전체적인 논리적 구조를 지정하는데 이용된다. 레코드에 근거해 분류된 데이터 모델은 다음과 같다.

- 계층모델(Hierarchical model)

데이터와 데이터간의 관계가 Fig.2와 같이 각각의 객체를 나타내는 개체(entity)와 이들

을 연결하는 링크(link)로써 표현되고 트리(tree)형태의 구조를 갖는 모델로서, 데이터 구조를 쉽게 이해할 수 있으나 중복된 데이터가 나타나는 단점이 있다.

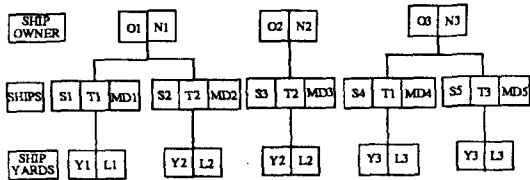


Fig. 2 Hierarchical Data Model

— 네트워크 모델(Network Model)

구별 가능한 객체를 나타내는 개체(entity)의 집합과 엔티티들의 관계를 나타내는 포인터(pointer)와 링크(link)로 구성되는 모델로서, 데이터는 Fig.3과 같이 레코드로 표현되고 데이터 사이의 관계는 링크(link)로 표현되며, 데이터베이스 내의 레코드들은 임의의 그래프 형태로 조직된다.

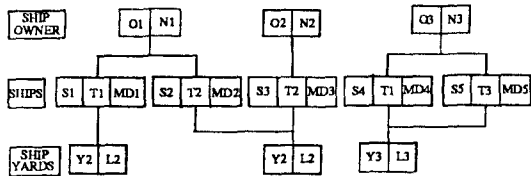


Fig. 3 Network Data Model

— 관계형 모델(Relational Model)

관계형 데이터 모델은 데이터와 데이터 간의 관계가 Fig.4와 같이 표(table)의 집단으로 표현되며 2차원 표와 같이 자연스러운 구조로 나타낼 수 있다. 또 데이터의 물리적 표현에만 사용되는 부가적인 상위구조가 필요 없으며 프로그램들 사이와 데이터의 논리적 표현 및 물리적 구성 사이에 극히 높은 독립성을 이룰 수 있어서 응용 프로그램과 데이터베이스의 운영비를 최소화 한다. 이 모델은 그동안 주로 사용되어져 왔던 계층구조(Hierarchical Structure)나 네트워크구조(Network Structure)모델이 갖는 결점들, 즉 데이터베이스 설계시 비용이 많이 들고, 사용자의 새로운 요구를 만족시

킬 수 없으며, 데이터베이스를 설계하고 응용하는데 고도로 전문화된 요원이 필요하는 등의 문제들을 해결해 주며 단순성, 유연성 및 논리성 등을 크게 향상시켰다.

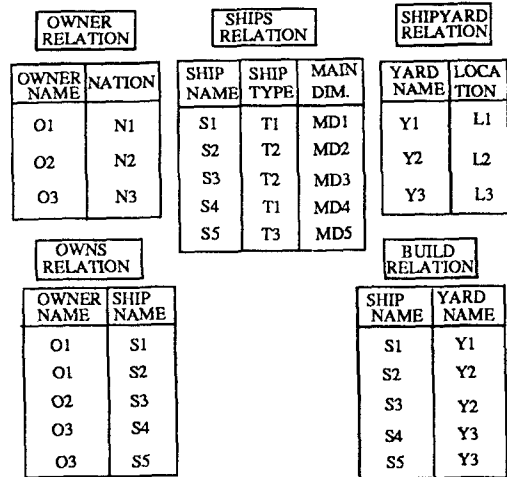


Fig. 4 Relational Data Model

2.2 관계형 데이터 모델의 구조

관계형 모델은 데이터 간의 관계가 표(table)의 집단으로 표현되는데 “관계(relation)” 또는 “표(table)”는 관계형 시스템에서 사용되는 유일한 구조이다. 데이터베이스에 포함된 모든 정보는 그러한 표 안에 표시된 자체의 제한 조건을 만족시키는 표로써 나타난다. 하나의 표는 행(row)과 열(column)로 이루어지는데 행(row)은 ‘튜플(tuples)’이라 하며 하나 이상의 속성집합으로 이루어지며 데이터베이스 안에 주어진 정보에 의해 표현된 세계의 실제 대상이 되는 개체(entity)를 나타내고 각각의 행(row)은 개체의 속성값들의 집합이다. 이와 같이 하나의 표는 실제세계의 객체(object)를 그 객체가 가지고 있는 특성과 관련지어 표현하고 있다.

— 개체(entity)에 대한 표현

관계형 데이터 모델은 대상이 되는 객체(object)와 이들간의 관계(relationship)를 표현한다. 이러한 표현은 데이터의 추상화(abstrac-

tion) 절차를 수행한 결과로써, 이들간의 관계를 데이터베이스로 모형화될 대상이 되는 전체 데이터들은 적합 데이터와 관계들로서 분류되어 데이터 모델내에서 정의된다. 이렇게 분류된 데이터 집단이나 관계들을 개체(entity)라고 정의하며, 개체 집합 E가 A1, A1, ..., An을 속성(attribute)으로 가지고 있다고 하면 E(A1, A2, ..., An)이라고 표시 하고 각각의 속성 Ai는 해당되는 영역(domain)Di에 속한다.

— 관계형 스키마

관계형 데이터모델에서 릴레이션들의 구조에 대한 명세를 관계형 스키마(relational schema)라고 한다. 관계형 스키마는 속성(attribute)과 그 속성의 영역(domain)들의 집합이다. 릴레이션 SHIP에 대한 관계형 스키마 SHIP(NAME: CHARACTER, TYPE: CHARACTER, LPP: NUMBER, B: NUMBER, D: NUMBER)로 표시된다. 다른 예로서, BUILD라는 릴레이션을 정의하여, BUILD(SHIP.NAME, YARD.NAME)라고 하자. 속성 SHIP과 NAME 이 두 릴레이션 스키마 SHIP BUILD에 공통적으로 나타나 있는 것은 서로 다른 릴레이션 스키마의 값들을 공통의 속성으로 연관시키기 위해서이다.

— 관계형 데이터 모델의 정규화(Normalization).

우리가 관계형데이터 베이스를 설계하는 목적은 정보를 불필요한 중복이 없도록 저장하고 또 정보를 쉽게 찾을 수 있는 릴레이션 스키마를 생성하는 것이다. 따라서 설계하고자 하는 데이터들에 대한 릴레이션 및 그들의 속성(attribute)들은 현실 세계를 가장 정확하게 표현할 수 있어야 하며, 이러한 표현들은 중복을 최소화하는 방향으로 결정되어야 하는데, 이와 같이 데이터 변경에 따른 릴레이션의 모순성을 피하고 사용자가 원하는 형태의 정확한 정보를 포함하는 릴레이션을 구성하기 위한 활동을 정규화(normalization)라 하며 이 과정에서 데이터들 사이의 종속성이라 불리는 제약조건을 이

용해야 한다.

앞의 예에서 OWNS 릴레이션과 BUILD 릴레이션은 정규화 과정을 거치면 다음 표와 같이 하나의 릴레이션으로 바꿀 수 있다.

O-B RELATION

SHIP NAME	OWNER NAME	YARD NAME
S1	O1	Y1
S2	O1	Y2
S3	O2	Y2
S4	O3	Y3
S5	O3	Y3

2.3 관계형데이터베이스

관계형 데이터베이스는 관계형 데이터 모델에서의 릴레이션들의 집합으로 구성되며, 시간에 따라 변화하는 실세계를 표현하는 것이므로 시간 의존적이다. 따라서 임의의 시점에서 데이터베이스의 내용을 적절히 변경, 첨가, 삭제할 수 있는 기능이 요구되어 지며 이때 사용자가 데이터베이스에 저장된 정보에 접근하기 위해서 질의어(query language)를 사용한다. 질의어는 릴레이션의 각 행들이 나타내는 정보를 동질의 집합체로 인식하고 수학적 집합연산을 통하여 사용자가 원하는 결과를 만들어 내며 선택(select), 추출(project), 카티션곱(cartesian product), 연합(union), 차집합(set difference)등의 기본 연산과 교집합(set intersection), 조인(theta join), 자연조인(natural join), 나누기(division)등의 연산자가 이용된다.

앞의 예에서 'Y1조선소에서 건조된 선박의 이름과 선종, 주요치수'를 찾는 질의를 관계형 대수식[5]으로 다음과 같이 나타낼 수 있다.

$$\sigma_{YARD\_NAME = "Y1"}(SHIPS \bowtie BUILD)$$

여기서  $\sigma$ 는 선택(selection) 연산을 의미하고  $\bowtie$ 는 자연조인(natural join)을 의미하며 수

학적으로는 두 릴레이션  $r(R)$ 과  $s(S)$ 가 있을 때

$$r \bowtie s = \prod_{R \cup S} (r \bowtie_{R.A_1 = s.A_1} \dots \cap r.A_n = s.A_n S)$$

단  $R \cup S = \{A_1, A_2, \dots, A_n\}$ 이다.

실제로 상용화된 DBMS를 사용할 경우, 위 질의어는 다시 다음과 같이 사용하기 쉬운 상용 질의어로서 나타낼 수 있다.

```
SELECT SHIP.NAME, SHIP.TYPE,
       SHIP.DIMENSION
FROM SHIPS, BUILD
WHERE YARD.NAME='Y'
```

이와같이 질의어를 사용하여 데이터베이스내의 정보를 처리하는 과정은 그 타당성이 입증된 수학적 원리를 바탕으로 하고 있으며, 사용자로 하여금 내부의 복잡한 처리 과정으로부터 벗어나 자유롭게 각종 작업을 수행할 수 있게 해주고 있다.

### 3. 선체 중앙부 구조부재의 접속관계 표현

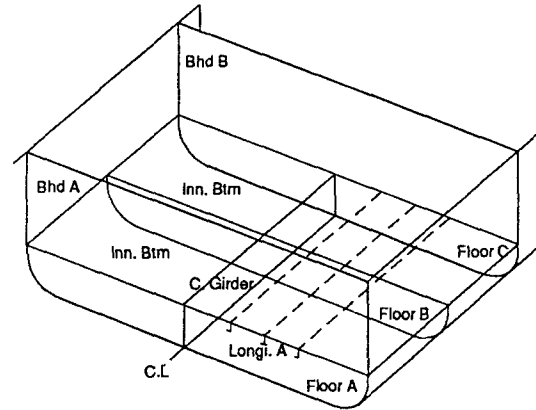
#### 3.1 구조부재의 의미론적 표현

선체구조 도면을 통해서 전달되는 정보를 전산 데이터로 바꾸기 위해서는 복잡하게 얽혀있는 구조부재의 배치형상을 나타내는 3차원 형상데이터와 그에 수반되는 부재치수와 같은 속성데이터를 다같이 나타낼 수 있는 3차원 설계 모델이 필요하게 된다. 여기서 말하는 모델은 단순히 기하학적 형상만을 나타내는 것이 아니라 선체구조를 스스로 완전한 의미를 갖도록 표현(semanticly perfect representation)할 수 있는 것이어야 한다[7]. 예를 들면 이중저 깊이가 설계과정에서 변경되었을 때 이중저 내저판에 부착된 증통보강재의 위치가 자동적으로 따라서 변경되게 하기 위해서는 보강재가 이중저의 부착(attached)되어 있다는 접속관계(topology)를 모델안에 정의할 수 있어야 한다.

#### 3.2 중앙부 구조부재의 접속관계

선체 중앙부 구조를 완벽한 의미로 표현하기 위해서는 앞에서 말한대로 부재간의 접속관계(topology)를 나타낼 수 있어야 한다. 이러한 접속관계 정보는 어느 한부재의 형상정보가 변경되었을 때 그에따라 주변부재의 형상정보가 자동적으로 변경되도록 하기 위해 필요하며 Fig.5는 그에 대한 개념을 보여준다.

이와같이 초기 구조설계 단계부터 각 구조부재간의 접속관계 정보를 그 부재가 갖고 있는 형상(geometry)정보와 분리하여 취급하면 설계과정에서 접속관계 또는 형상중 어느 한쪽이 변경되어도 다른 한쪽의 데이터를 변경하지 않아도 되기 때문에 데이터 처리의 양을 훨씬 줄일 수 있게 된다.



Bhd A	is LIMITED by HULL and DECK
Bhd B	is COPY of Bhd A
INN.BTM	is LIMITED by HULL and INN. BTM
C.Girder	is LIMITED by HULL and INN. BTM
Floor A	is watertight
Floor B	is copy of floor A, but non-watertight
Girder	is superpart of C.Girder
Longi A	is attached on Inn.Btm

Fig. 5 Concept of Topological Information

선체중양부를 구성하고 있는 각 요소들에 대한 접속관계는 우선 다음의 몇가지를 들수 있으며 형상정보의 세밀한 정도에 따라 보다는 많은 접속관계 정보를 추가할 수도 있다.

- Super part of: 구성하고 있는 단위요소 구조를 표현
- Limited by: 경계를 이루는 인접 구조요소를 표현
- Attached on: 보강재의 부착관계를 표현
- Identical with: 동일한 형상 또는 접속관계를 가진 구조요소를 표현
- is of type: 각 구조요소의 형 정의

이와같은 접속관계 정보를 이용한 구조배치 변환의 효율성을 검증하기 위한 2차원적인 실험예를 Fig.6에 보였는데 이들은 각각 원래 배치, 이중저 깊이를 변경시킨 것, 'limited'관계를 변경시킨 것(이중저 girder 일부의 제한 접속관계가 이중저 내저판에서 topside tank 경사판으로 변경됨), 'attached' 관계를 서로 바꾼 것(Topside tank 보강재와 Hopper tank 보강재의 부착위치를 서로 바꾸었음)등을 보여주고 있다.

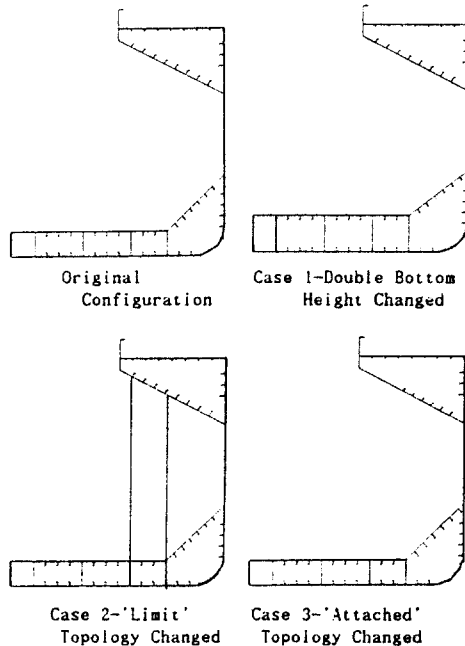


Fig. 6 2D Example of changed Topological Information

#### 4. 데이터 모델링

##### 4.1 설계모델

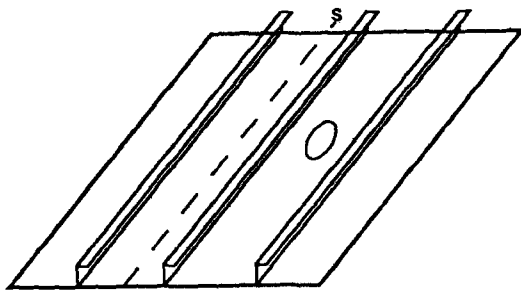
설계업무가 단계별로 진행됨에 따라 각종 설계정보가 점점 더 추가되어 마지막으로 생산과정에 이르게 되면 초기, 기본설계 단계에 비해서 훨씬 많은 정보를 다루게 된다.

따라서 초기, 기본설계 단계에서의 정보관리를 효율적으로 하기 위해서는 생산단계에서의 복잡하고 많은 정보를 처리할 수 있는 생산모델보다 단순화된 설계모델을 이용하는 것이 바람직 하다. 설계모델(design model)이나 생산모델(product model)은 궁극적으로는 모두 같은 형상을 표현하기 위한 것이지만 취급하는 업무 성격에 따라 정도(精度)와 용도가 약간씩 다르다고 할 수 있다. 다시 말해서 생산모델은 생산공정을 위한 것이므로 절단도를 만들기 위해서는 판재 두께와 용접부의 개선 형상까지 나타낼 수 있는 형상정의 기능이 필요하나 초기, 기본설계 단계에서는 molded line으로 나타나는 형상과 부재간의 접속관계 정도로도 충분하고 판재두께 같은 것은 속성정보로만 나타내면 된다. 대신 설계 모델에서는 각각의 구조부재가 어떤 설계조건을 만족해야 하는지에 대한 정보를 표현 할 수 있어서 부재치수를 결정하는데 이용할 수 있어야 한다.

##### 4.2 개체-관계형데이터 모델

본 연구에서는 초기, 기본설계 단계에서의 선체구조 정보를 전산 데이터화하여 데이터베이스에 저장하고, 구조, 진동해석이나 설계하류 단계에서 직접 불러다 사용할 수 있도록 설계 모델에 대한 데이터 구조를 개체-관계형 모델(entity-relationship model)로 나타내었으며, 각 부재간의 접속관계 정보를 표현할 수 있는 관계형 데이터 모델을 구성하였다. 선체구조의 설계 및 생산의 모든 단계별 과정에서 그 구조물의 일부를 각각의 필요에 따라 효과적으로 취급하기 위해서는 구조물을 이루고 있는 각

구조요소들을 분리해서 독립된 정보단위로 가지고 있는 것이 바람직하며, 여기에서는 그를 위해 '논리요소(logical element)'의 개념을 도입하였다. 논리요소는 순환반복적으로 구성될 수 있으므로 하나의 논리요소로서 단 한개의 보강재를 나타낼 수도 있고 선체의 매우 복잡한 구조물 전체를 나타낼 수 있으며, 각 논리요소에 존재하는 복합적 관계를 개체-관계형 데이터 모델 개념에 따라 정의함으로써 선체 전부 또는 일부를 나타내는 조직망을 만들 수 있다. Fig.7에는 선체구조를 이루고 있는 단위요소인 보강된 평판을 정의하는데 필요한 정보들을 나타내고 있으며 그와같이 단순화된 단위구조요소를 데이터모델의 기본요소로 하고 각 단위 구조요소간의 접속관계를 정의한 개체-관계 도표(entity-relation diagram)를 Fig.8에 보였다. 이 그림에서 나타내고 있는 단위구조요소, 즉 논리요소(logical element)는 단면형상에 굽힘(knuckle)을 가질 수 있는 단순평면에 보강재, 용접선(seam), 절취선(cutout)등의 정의가 가능하고 논리요소들 간에 '상위구조(superpart)'와 '제한(limited)'의 관계를 표현할 수 있다.



- 판재 재료, 판폭, 판두께
- 보강재 재료, type, 치수, 취부위치
- Cutout type, size, 위치
- 용접선 위치
- 접속관계
- 소속관계

Fig. 7 Structural Element of Ship

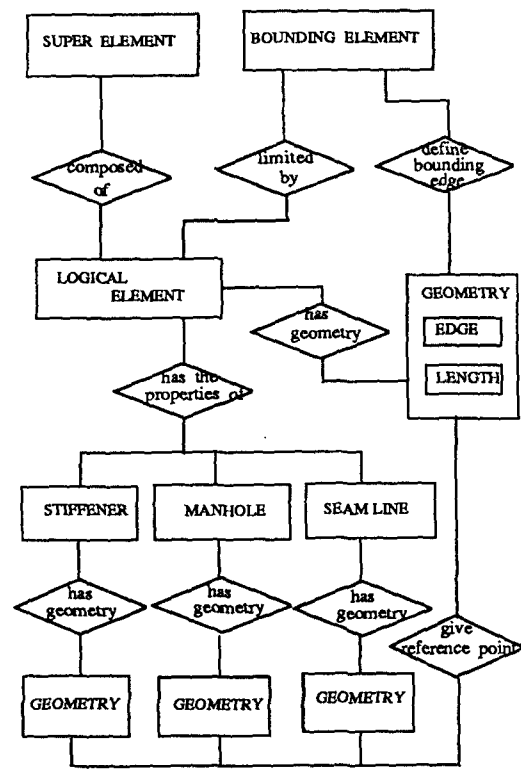


Fig. 8 E-R Diagram Representing Structural Element

이러한 개체-관계의 기본형에 따라 선체중앙부를 구성하는 각 단위 구조부재를 나타내는 데이터 집합을 분리, 정리한 예를 Fig.9에 보였으며 이들 데이터 집합들은 정규화 과정을 거쳐 각각의 스키마(schema)를 가진 표(table)로서 구성된다.

#### 4.3 결과 및 고찰

도면을 통해서 전달되던 정보들을 전산데이터화 하였으며 이를 구조해석이나 진동등 다른 분야에서 이용될 수 있도록 유기적인 관계를 갖는 데이터의 구조화, 즉 데이터 모델링을 시도했다. 데이터 구조는 각 부재간의 접속관계까지 표현된 개체-관계형모델을 만들고 그에 따라 선체중앙부 구조부재의 형상과 치수를 나타내는 각종 정보가 20여개의 표(table)로서 이루어지는 관계형 데이터모델을 구성하였다.

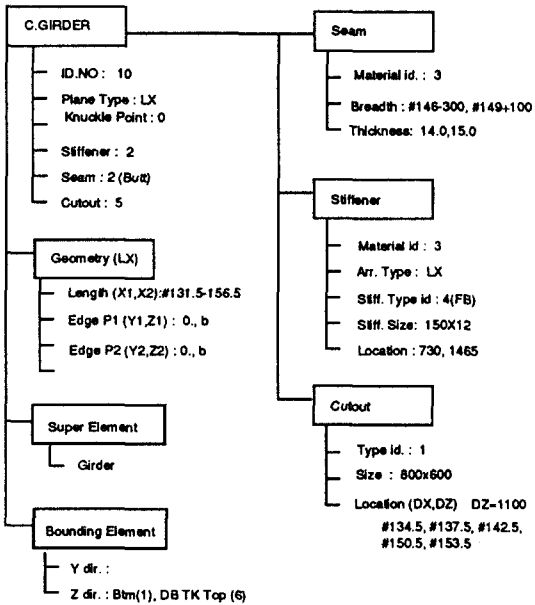
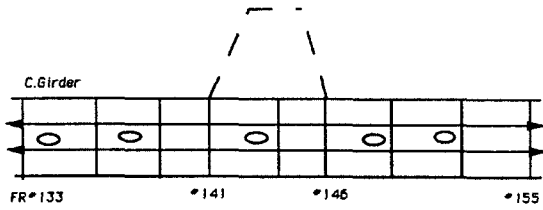


Fig. 9 Data Sets Representing Center Girder

구조배치 및 치수산정 프로그램에서 생성된 구체적인 데이터는 상용화된 관계형 데이터베이스 관리시스템(RDBMS) 'ORACLE'을 통하여 저장, 갱신, 회수되며, 각각의 표(table)는 독립적으로 혹은 특정한 목적에 맞추어 동시에 처리할 수 있는 장점이 있다. Fig.10은 표에 나타난 데이터 스키마의 한 예이고, Fig.11은 데이터의 실예(instance)에 따라 workstation의 그래픽 기능을 이용해서 가시화한 결과이다.

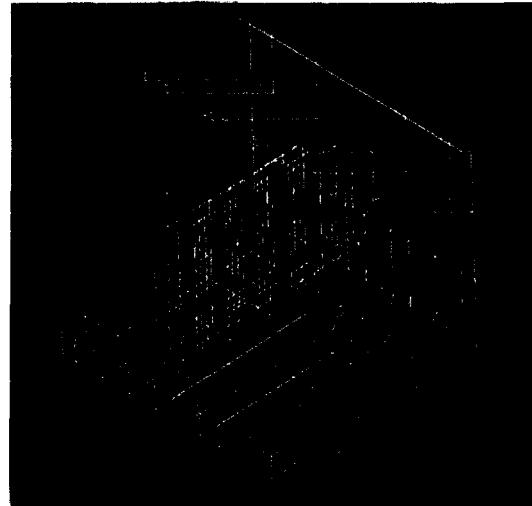


Fig. 11 3-D View of Midship Structure

1	LOELEM (Logical Element)						max N=30
N	ELEM(N)	JEL(1, N)	JEL(2, N)	JEL(3, N)	JEL(4, N)	JEL(5, N)	JEL(6, N)
I	C	R	R R	R	R	R	R
elem. no.	element name	plane type no.	num. of knuckle point	num. of longi. Stiff.	num. of fr. sect stiff.	num. of plate	num. of cutout

2	PLANELX (Longitudinal Plane)								max N = 30	
KLX	KLX	KLX	PLLX	PLLX	PLX	PLX	PLX	PLX	PLX	PLX
(1,N)	(2,N)	(3,N)	(1,N)	(2,N)	(1,N)	(2,N)	(3,N)	(4,N)	(5,N)	(6,N)
I	I	I	R	R	R	R	R	R	R	R
			Fr. #	Fr. #	m	m	m	m	m	m
elem. no.	bounding elem. no.		start	end	sect. edge point		P2		P3	
	Y dir	Z dir	fr. no.	fr. no.	Y1	Z1	Y2	Z2	Y3	Z3

Fig. 10 Data Schema for Midship Structure



## 5. 결 론

선체구조는 거대한 구조물의 한 전형으로서 수많은 부품이 여러단계의 공정을 거쳐서 건조되며 따라서 대량의 생산관련 정보, 즉 설계정보를 어떻게 효율적으로 관리하는가의 문제가 생산성 향상의 관건이 되고 있다. 그러나 지금까지 이와 관련된 전산화의 실적은 관련정보의 일원적 관리보다는 처리 절차에 더 치중된 단위 프로그램들이 주를 이루고 있으며 이는 아직도 공학적 문제에서는 데이터의 관리보다는 데이터가 처리되는 과정, 즉 알고리즘 중심으로 프로그램이 이루어 지고 있음을 뜻한다.

여기에서는 서로 다른 설계업무 처리 과정에서 도면을 통하여 전달되는 각종 정보의 일관화를 위해서 데이터 모델링을 수행하고 관계형 데이터모델이 갖는 특징, 즉 데이터 구조의 단순성, 유연성, 논리성 및 데이터간의 독립성을 검증하였다. 형상을 나타내는 데이터에 대해서는 형상이 처리되는 프로그램과 독립적으로 완전한 의미의 형상을 관계형데이터 구조로서 나타내기 어렵다는 점이 지적되고 있으나 이것은 기존의 계층식 또는 망 데이터 모델도 마찬가지 경우이며 실제로는 관계형 모델의 구조에 추가적인 기능을 부여하여 사용하는 방법이 여러가지 시도되고 있다[8,9].

## 후 기

이 연구는 과학기술처의 국책연구 사업인 CSDP-선체구조설계 시스템개발(II)'의 일환으로 수행된 것입니다[10].

## 참 고 문 헌

- [1] 野本敏治의, “造船 CIM을 위한 設計, 生産情報獲得 支援 시스템 構築”, 日本造船學會紙 第166號, 1989.
- [2] 梅崎 一夫, “造船의 設計, 生産設計의 컴퓨터에 의한 一貫處理에 對해서”, 日本機械學會紙 第794號, 1985.
- [3] 木村文彦, “형상모델링과 CAD/CAM”, 日本精密工學會紙 第53卷3號, 1987.
- [4] 小山健夫의, “統合化 CAD/CAM시스템構築을 위한 船體構造 表現의 基礎的 研究”, 日本造船學會紙 第164號, 1988.
- [5] Henry F.Korth and Abraham Silberschatz, *Database System Concepts*, Univ.of Texas, 1986.
- [6] 서승완의, *CSDP( I )—船體設計·生産 전용 데이터 베이스관리 시스템개발*, KIMM연구보고서, 1989.
- [7] R.Bronsart and E.Lehmann, “A Data Model for Ship Steel Structures”, ICCAS88, 1988.
- [8] A.Kemper and M.Wallrath, “An Analysis of Geometric Modeling in Database System”, *ACM Computing Surveys*, Vol. 19 No.1, 1987.
- [9] 木村文彦, “Object지향 CAD/CAM을 위한 모델링과 데이터베이스”, 日本情報處理學會紙 Vol.29 No.4, 1988.
- [10] 김용대의, *CSDP( II )—船體構造設計 시스템 開發*, 海士技術研究所, 1990.