

유도무기 통합최적설계를 위한 데이터베이스 설계 연구

Database Schema Design for Integrated Missile Design and Optimization

이 재 우† 이 승 진* 최 석 민** 조 국 현**
Lee, Jae-Woo Lee, Seung-Jin Choi, Seok-Min Cho, Guk-Hyun
김 광 식**** 이 정 육*** 정 갑 주*****
Kim, Kwang-Sik Lee, Jeong-Oog Jung, Karp-Joo
(논문접수일 : 2008년 11월 4일 ; 심사종료일 : 2008년 12월 12일)

요지

본 논문에서는 유도무기 통합 설계 프레임워크를 위한 데이터베이스를 설계하였다. 실제 개발에 앞서 유도무기 설계에 필요한 데이터베이스를 위한 요구도를 분석하였으며, 데이터베이스 설계 개념을 정립하였다. 또한 해석도구 간의 데이터흐름 등을 파악하였으며, 이를 정보를 토대로 테이블들을 구성하고 각 객체관계를 정의하여 데이터베이스 스키마를 설계 및 구축하였다. 개발된 DB는 유도무기 통합 설계 프레임워크에 통합되었으며 프레임워크내의 데이터 저장, 검색 및 관리 등의 역할을 수행하며, 유도무기 설계에 적용하여 그 효용성을 검증하였다.

핵심용어 : 유도무기, 데이터베이스, 데이터베이스 스키마, 객체관계

Abstract

In this paper, the missile database system for the multidisciplinary design and optimization framework has been designed and investigated. The requirements of the missiles database are analyzed and the database design concept has been defined first. The data flow among the analysis disciplines is investigated and the Entity-Relation diagram is constructed to develop the database table and to define the database schema. The developed database system is integrated into the missile design framework to perform the store, search and management of the design data during the design and analysis of the missile development, hence the efficiency and validity of the database designed in this study has been demonstrated.

Keywords : missle, database, database schema, object-relation

1. 서 론

유도무기는 현대전에서 그 중요도가 날로 증가해가고 있는 전술무기로서 이것의 보유 및 운용능력은 국가안보 유지능력의 주요 척도 중 하나이다(이재우 등, 2005).

유도무기는 고속/고기동이 요구면에서도 대부분 발사모체에 의해 크기와 무게에 대한 제약이 따르므로 그 설계가 까다롭다. 작은 크기와 무게를 유지하면서도 높은 성능을 달성하기 위해서는 최적화 개념의 설계가 이뤄져야 하며, 이는 어느 한 분

야에 대한 최적화 설계가 아닌 유도무기에 관련된 다양한 분야 간의 균형있는 최적화가 이뤄져야 한다. 이러한 부분들에 대한 해결방법으로 다분야 통합최적설계(MDO, Multidisciplinary Design Optimization)를 이용하기로 한다(전권수 등, 2001; 황진용 등, 2003; Salas 등, 1998).

성공적인 MDO를 위한 핵심기술로서 통합최적설계 기술과 함께 컴퓨터 소프트웨어 및 하드웨어 기술 및 데이터베이스 설계 기술이 필수적으로 요구된다. 다분야간 설계를 위해서는 각 분야간의 데이터 공유가 무엇보다 중요한 요소가 되

† 책임저자, 정회원 · 건국대학교 항공우주정보시스템공학과 교수
Tel: 02-450-3461 : Fax: 02-444-6670

E-mail: jwlee@konkuk.ac.kr

* LIG넥스원(주) 연구개발본부 기계연구센터 연구원

** 건국대학교 항공우주정보시스템공학과 석사과정

*** 건국대학교 항공우주정보시스템공학과 교수

**** 건국대학교 인터넷멀티미디어공학부 박사과정

***** 건국대학교 인터넷멀티미디어공학부 교수

• 이 논문에 대한 토론을 2008년 2월 28일까지 본 학회에 보내주시면 2009년 4월호에 그 결과를 게재하겠습니다.

고, 각 분야간의 설계데이터의 공유를 위해서는 데이터의 일관성, 무결성, 최신성 등이 요구된다(Ullman 등, 2007). 따라서 이러한 요구 사항을 만족할 수 있도록 데이터베이스가 효율적으로 설계되어야 하며, 이는 통합설계 환경구축을 위하여 대단히 중요한 요소이고 MDO의 성공여부를 결정짓는 핵심요소라고 할 수 있다.

본 연구에서는 유도무기의 통합설계환경 하에서 형상데이터, 해석데이터, 최적화 관련데이터를 포함하는 데이터베이스를 설계하고 그것을 효율적으로 관리할 수 있는 시스템 인터페이스를 개발하고자 한다.

2. 유도무기 프레임워크 데이터베이스 요구도 분석 및 설계 개념 정립

2.1 유도무기 통합최적설계 프레임워크

유도무기 통합설계 프레임워크에는 3개의 기본적인 해석코드를 사용하며, 이들을 토대로 최적화 기법을 적용하여 유도무기 설계를 수행한다. 유도무기 MDO 프레임워크는 6개 모듈로 구성되어 있다. 데이터베이스 시스템은 Oracle 9i를 사용하였으며, 사용자 인터페이스는 JAVA를 사용하여 개발하였다. 형상설계 모듈로는 I-DEAS를 선정하였고, 최적화 모듈은 구배기반의 DOT를 사용하였다. 해석자원은 MATLAB으로 작성된 wing 및 weight sizing code인 MOST 및 MOST의 초기 입력 값을 결정해주는 intcfg, CfgDesign, GUI 환경으로 제작된 DATCOM 및 동안정미계수 산출코드인(DYNDER) 통합 코드 그리고 C로 작성된 차원계수 산출 코드인 CODSAM을 사용하였다. 그림 1은 유도무기 통합설계 프레임워크의 시스템 통합구조도이다(이재우 등, 2005).

2.2 객체관계형 데이터베이스 (ORDB, Object-Relational Database)

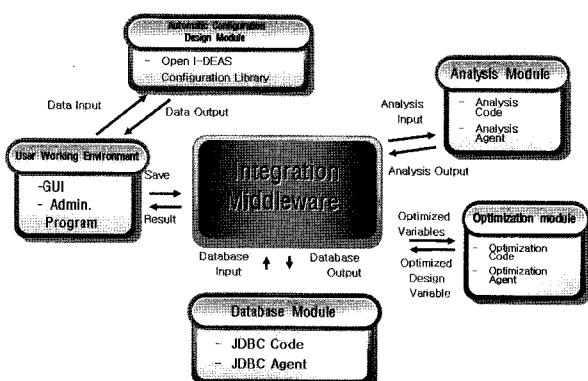


그림 1 프레임워크 시스템 구조

데이터베이스 관리시스템(DBMS, DataBase Management System)의 주목적은 자료의 효율적인 관리 및 다수 사용자의 자료 공유이다(Salas 등, 1998). 주로 사용되어온 DBMS 방식인 관계지향형 데이터베이스 RDB(Relational Database)는 사무용 데이터의 처리에 탁월한 기능을 갖고 있다. 그러나 최근 들어 RDB가 효율적으로 처리하기 곤란한 특성 데이터들 -CAD데이터, CAE데이터, 멀티미디어데이터, 하이퍼텍스트 데이터 등이 증가하고 있고 이러한 데이터의 관리가 중요한 문제로 대두되었다. 이의 해결을 위해 다양한 접근 방법들이 모색되고 있는데 그 중 가장 성공적인 접근 방법으로 객체관계형 데이터베이스(ORDB)가 있다(Ullman 등, 2007).

객체관계형 데이터베이스는 객체지향 개념을 데이터베이스 설계에 접목시키는데 가장 성공적인 평가를 받고 있다. 관계형 데이터베이스를 대체할 목적으로 개발된 객체지향형 데이터베이스는 표준화의 미비로 그 가능성에도 불구하고 상업적으로 성공치 못하였다. 객체관계형 데이터베이스는 객체지향과는 달리 기존의 관계형 데이터베이스 기술을 기반으로 하고 있기 때문에, 많은 부분이 잘 정의되어 있는 상태에서 출발하였다(Devarakonda, 2001).

객체관계형 데이터베이스는 관계형 데이터베이스가 갖고 있는 제약점을 해결하기 위해서 관계형 데이터베이스의 개념을 기반으로 객체지향형 데이터베이스의 일부 기능과 개념을 통합한 것으로 볼 수 있는데, 두 방식의 특성과 차이점을 요약하면 표 1과 같다(이재우 등, 2004).

표 1 객체지향형 데이터베이스와 객체관계형 데이터베이스의 개념 비교(이재우 등, 2004)

객체지향형 데이터베이스	객체관계형 데이터베이스
OO Database System Manifesto	Third-Generation DBMS Manifesto
복합 객체 지원(array, sequence, record, set, union을 탑입으로 지원)	유일키가 없을 때에만 객체 식별자 지원
유일한 객체 식별자 지원	다형성 지원
계산적(Computational) 완전성 지원	특별한 언급이 없음
	SQL 지원

2.3 유도무기 프레임워크 DB 요구도 분석

서론에서 언급되었듯이 유도무기는 공력 해석, 조종 및 안정성, 중량해석 및 형상 사이징 등의 분야가 상호 긴밀하게 연계되어 설계, 해석 및 개발되는 통합 시스템이며, 유도 무기의 개발을 위해서는 여러 설계/해석분야의 공동의 연구가 필수적이다.

유도무기 설계를 위한 데이터는 일반적으로 사용되는 관계

형 데이터베이스에서 요구되는 데이터 특성과는 다른 특성을 가지고 있으며 이를 요약하면 다음과 같다(이재우 등, 2004).

- 첫째, 객체들이 서로 연관성을 갖고 있기 때문에 다른 객체들의 집합으로 객체를 정의할 수 있어야 한다.
- 둘째, 동적인 관계를 처리할 수 있어야 하고 객체의 행동과 제약식도 표현할 수 있어야 하며, 다수의 응용프로그램에서 요구하는 새로운 자료형을 정의할 수 있어야 한다.
- 셋째, 설계변경에 따라 객체의 버전을 관리해서 변경과정에서의 서로 다른 상태라든가, 유효기간, 대안이 될 수 있는 객체, 가정 등의 정보들을 고려할 수 있어야 한다.
- 넷째, 유도무기를 표현하는 개체의 경우 다루어야 하는 데이터의 크기가 매우 클 수가 있으며 데이터의 도메인도 가변적인 것들이 있다.
- 다섯째, 여럿의 사용자가 설계에 참여하게 되므로 사용자끼리 효율적으로 통신을 할 수 있도록 객체의 상태 변화를 사용자에게 알려주고, 사용할 수 있는 객체를 알려주는 기능이 필요하다.
- 여섯째, 미사일 설계와 같은 공학 분야에서는 SQL과 같은 언어로 수행하기 어려운 복잡한 수학 데이터 조작을 필요로 할 수도 있다.

이러한 특성들을 고려할 때 일반적으로 널리 사용되고 있는 관계형 데이터베이스를 사용할 경우 원하는 성능과 기능을 구현하기가 어렵기 때문에 대용량 데이터 처리 및 높은 처리속도를 가지고 있는 객체지향 데이터베이스나 객체관계형 데이터베이스의 사용이 필요하다. 그러나 객체지향형 데이터베이스는 앞서 설명한대로 그 이론이 확실히 정립되어 있지 못하고 상용화된 프로그램이 적으로 개발의 위험도를 줄이기 위해 객체관계형 데이터베이스가 유도무기 설계를 위한 데이터베이스 구현에 가장 적합하다고 볼 수 있다. 본 연구에서는 Oracle 9i를 통해 객체관계형 데이터베이스를 구축하였다(Oracle 9i, 2002). ORACLE 9i은 다양한 O/S 및 하드웨어를 지원하며 원격통신이 가능하고 가장 중요한 특성 중 하나인 보안 및 권한 설정을 통해 데이터베이스 관리자가 일괄적으로 사용자를 통제할 수 있고, 허용되지 않는 사용자의 접근을 제한할 수 있다(Oracle 9i, 2002). 이러한 특성은 통합될 프레임워크가 분산 환경을 고려하였기에 다양한 환경에서 작동할 수 있으며, 보안성이 중시된다는 점을 고려 할 때, 본 연구의 목적인 유도무기 설계 데이터베이스 구축에 매우 적합한 DBMS로 볼 수 있다.

본 연구에서 개발한 데이터베이스는 유도무기 MDO 프레임워크에 통합되어 프레임워크 내에서 생성되거나 이동되는

데이터들을 관리하고 저장하며 검색하거나 불러오는 역할을하게 된다.

2.4 데이터베이스 설계 개념 정립

유도무기 설계 데이터베이스는 사용자가 필요한 데이터를 적시에 얻을 수 있어야 하므로 데이터의 무결성, 참조 무결성, 데이터 보안, 확장성과 설계 내력(Design History) 등을 고려하여 데이터베이스를 설계하여야 한다(이승진 등, 2005). 유도무기 설계는 해석 및 최적화 과정을 거치므로 설계, 해석 자료가 포함되어야 하고 설계 내력 역시 가지고 있어야 한다. 설계된 유도무기에 관한 해석 결과 및 최적화 결과 데이터는 사용자의 요구에 따라 일목요연하게 정리하여 자료의 검색, 저장의 효율성 및 사용의 용이성을 고려하여 설계/구현해야 한다. 데이터베이스 관리는 데이터의 보안과 데이터베이스 튜닝에 관련된 사항이다. 데이터베이스 보안의 방법으로는 사용자 생성 시에 각 사용자에 대한 보안등급을 정하고 보안등급에 따라 접근이 허용된 데이터만 볼 수 있도록 하며, 허용이 되지 않은 데이터는 그 존재 자체를 알 수 없도록 한다. 응용프로그램은 DBMS에 접근 할 때마다 신분확인과 인증을 요구하도록 구현된다. 또한 데이터 양의 증가에 따라 튜닝을 해줌으로써 데이터베이스가 가지고 있는 성능을 최대한 발휘할 수 있도록 하는데 설계의 주안점을 두었다.

3. 유도무기 프레임워크 데이터베이스 스키마 설계

3.1 유도무기 통합최적설계 프레임워크에서의 데이터 관리

데이터베이스에는 현재 설계가 진행 중인 유도무기에 대해 기본 요구조건인 비행속도, 사거리(공격반경), 고도의 정보가 프로젝트를 구분하기 위한 기본 조건으로 들어간다. 데이터베이스 설계 시 유도무기 설계과정은 프로젝트 별, task 별로 관리가 되는데, 한 프로젝트 당 하나 이상의 task에 대한 데이터가 입력 및 출력될 수 있다. 이러한 개념을 고려하여 데이터 관리의 계층을 표현하면 그림 2와 같다.

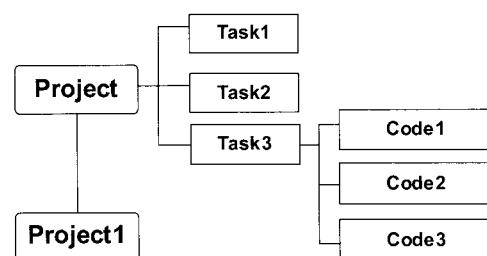


그림 2 데이터 관리 계층

프로젝트 관리 데이터는 유도무기 프레임워크를 이용하는 사용자의 기본 정보를 포함한다. ProjectID는 각 프로젝트를 구별할 수 있게 하기 위해 각 프로젝트별로 고유한 값을 부여하며 주키(Primary Key)로도 사용하였다. Task 관리 데이터는 전체 데이터 흐름에 대한 모든 정보를 가지고 있고 향후 사용자에 의한 과거 데이터 조회 시 검색이 원활하도록 복합키(Composite Key)로 구성하였다. 각 값은 ProjectID, TaskID로 구분되어 입력되도록 설계하였다. 데이터는 사용자의 목적에 따라 변화될 수 있는 데이터이므로 변화하는 데이터에 대한 일괄적인 관리를 통해 향후 설계자가 프로젝트 조회 시 해당 설계 임무를 구분할 수 있는 데이터로 관리된다.

3.2 해석도구 간 데이터 흐름 분석

주의 Sizing 및 성능 해석코드인 MOST가 Sizing하는 주의 외의 동체 및 미익의 데이터는 이미 설계된 것을 토대

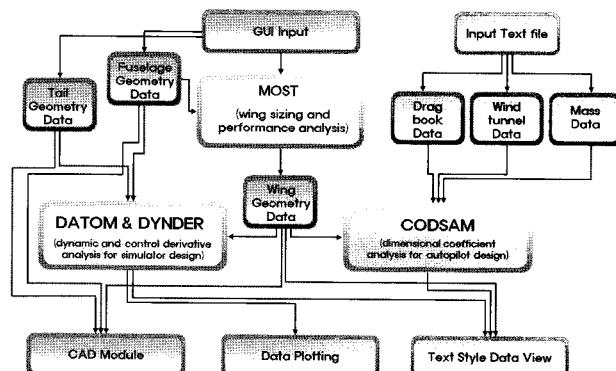


그림 3 해석도구 데이터 흐름도(Dataflow)(이재우 등, 2004)

표 2 데이터 변환 분석 예제

MOST output		변환과정	DATCOM input	
변수 번호	변수 명		변수 번호	변수 명
날개의 형상	MOST 결과는 사각형 wing만 생성	45	WGPNLF =1	
3-1	Cr(mm)	CHRDR =Cr*0.001*3.28	46	CHRDR (ft)
3-4	Ct(mm)	CHRDTP =Ct*0.001*3.28	48	CHRDTP (ft)
3-5	Span (mm)	SSPN=Span*0.5*0.0 01*3.281	49	SSPN (ft)
3-8	Swp QT (deg)	CODSAM은 LE와 QT 두 가지 Swp 값이 나온 CHSTAT=0.25 Swp QT=SAVSI	53	CHSTAT =0.25
			54	SAVSI (deg)
3-3	Cre(mm)	SSPNE =((Cre*0.001*3.28) -CHRDTP)*SSPN/(C HRDR-CHRDTP)	50	SSPNE (ft)

로 진행하며, 이 코드들을 위한 입력 값 이외에 필요한 경우 풍동 실험 데이터와 질량분포데이터, 항력계산을 위한 보정에 관한 데이터가 추가적으로 별도의 텍스트 파일 형태로 입력되게 된다. 프로그램 간의 데이터 흐름도는 그림 3과 같다 (이재우 등, 2004).

통합 해석 시 각 해석도구는 다른 해석도구에서 도출된 결과를 토대로 계산을 수행해야 하며 자동적인 통합 해석을 위하여 변수들의 상관관계를 분석하였고, 표 2는 그 한 예이다.

3.3 데이터베이스 구성체계(Hierarchy)

그림 4는 데이터베이스의 전체 구성체계를 보여주고 있다. 여기에는 사용자가 구성한 설계과정에 필요한 데이터, 사용자가 구성한 Data flow, 그리고 설계결과로써 획득한 데이터가 고려되었다. 본 연구의 유도무기 설계는 프로젝트 단위에서 이루어지며 각 프로젝트는 유도무기 설계 결과에 대한 데이터, 참고했던 풍동 데이터 및 형상데이터 그리고 최적화된 유도무기 형상에 대한 모든 설계자료를 포함하고 있다.

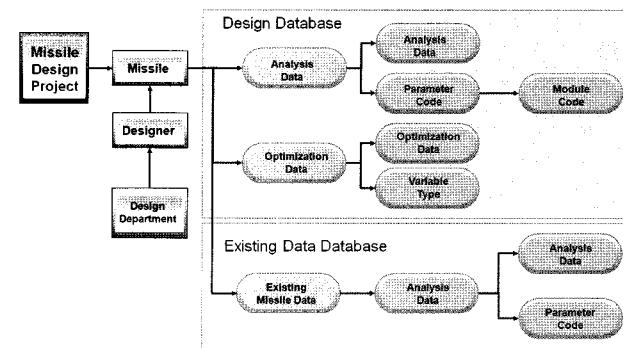


그림 4 데이터베이스 구성체계(Hierarchy)

3.4 데이터베이스 스키마 설계

다분야 통합최적설계는 해당 시스템에 대한 요구를 만족하기 위해 수많은 계산을 통해 설계 데이터를 얻고, 앞서 수행된 해석 결과를 다음 해석에 이용하는 하는 등 데이터의 이용률이 높기 때문에 효율적인 데이터베이스 설계가 요구되고, 이를 만족하기 위해서는 설계에 이용되는 데이터의 특성 및 데이터의 흐름에 대한 분석이 선행되어야 한다. 그림 3의 해석도구 데이터 흐름도에 기반하여 설계된 유도무기 통합설계를 위한 데이터베이스 스키마는 그림 5와 같다. 데이터베이스 스키마는 PROJECT_INFO, TASK_INFO 등 20개의 클래스로 구성되어 있다. 이들 클래스에 대한 설명은 다음과 같다.

- MODULE_INFO : 사용자가 필요로 하는 실제 작업이

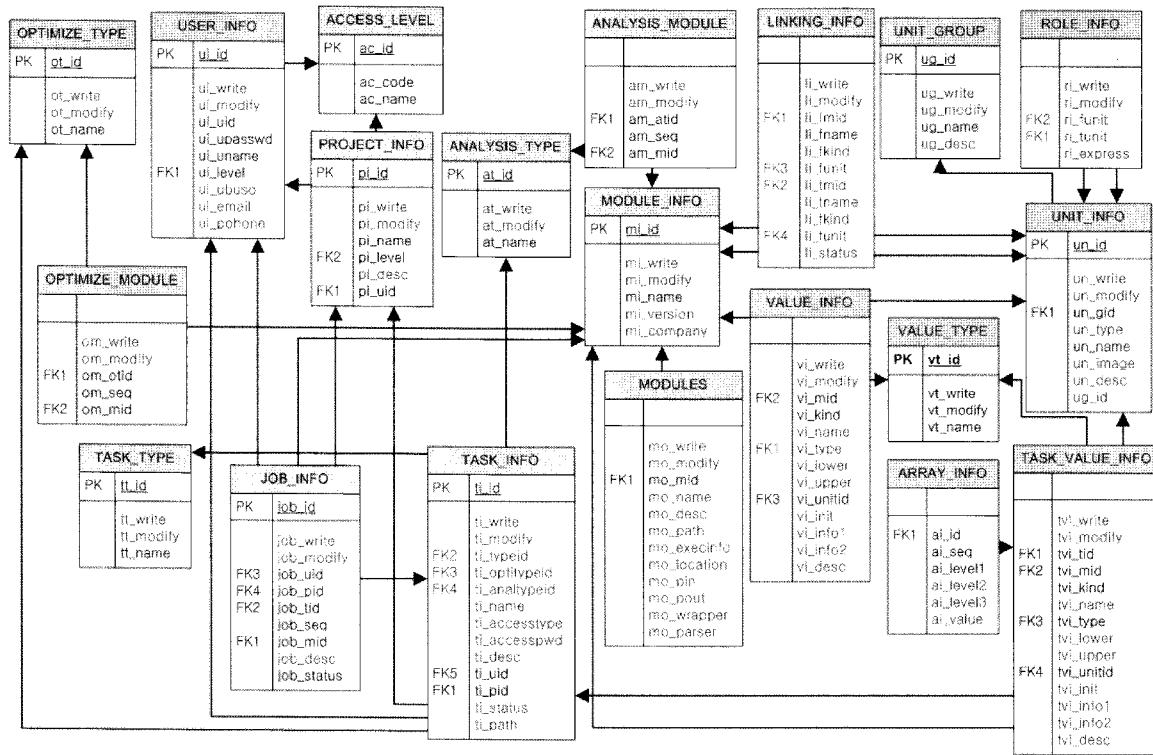


그림 5 클래스 다이어그램

수행될 모듈(Legacy 프로그램)에 대한 정보를 가지고 있다. 동일 모듈에 대하여 여러 개의 인스턴스로 구성될 수 있다. 분산환경이므로 동일한 모듈이 여러 시스템에서 동작할 수 있기 때문이다. MODULES 클래스에서 인스턴스에 대한 정보가 관리된다.

- **MODULES** : Legacy 프로그램에 대한 인스턴스에 대한 정보들을 담고 있다. 실행 방법과 위치, 필요로 하는 정보 등을 담고 있다. 실제 이 정보를 토대로 래거시 프로그램들이 수행된다. 사용자로부터 입력 받는 GUI화면에 대한 정보, 출력결과를 보여 주는 정보도 가지고 있다.
- **ANALYSIS_MODULE**, **OPTIMIZE_MODULE** : 해석과 최적화에 필요한 모듈들에 대한 정보를 가지고 있다. MODULE_INFO로부터 관련된 모듈들에 대한 정보를 저장한다.
- **ANALYSIS_TYPE**, **OPTIMIZE_TYPE** : 해석/최적화 모듈들의 작업 순서 정보를 저장한다.
- **TASK_TYPE** : 단독 해석 수행인지, 해석과 최적화를 동시에 수행하는지에 대한 정보를 저장한다.
- **ACCESS_LEVEL** : 프로젝트의 보안 수준을 결정한다. 외부 공개가 가능한 프로젝트인지 보안이 필요한 프로젝트인지를 결정한다.
- **VALUE_INFO** : 각 모듈들이 필요로 하는 기본적인 정보들이 들어있다. 레코드가 하나의 값을 나타내며 어

떤 모듈의 인풋/아웃풋 값인지를 결정하고 값의 형태 및 초기값, Boundary값을 설정할 수 있다.

- **VALUE_TYPE** : 값에 형태에 대한 정보를 나타낸다. 예를 들면 Integer형태 인지, Double 형태 인지 Filename 형태인지에 대한 정보 값을 가진다.
- **UNIT_INFO** : 값의 단위에 대한 정보를 나타낸다. 각각의 값들은 m, m^2, cm, km 등 다른 단위 형태를 가지고 있기에 각 값에 대한 단위 정보를 가지게 된다. 이 단위 정보는 값 변환 시에 예를 들어서 mft로 변환하는 과정 등에서 사용된다.
- **ROLE_INFO** : 값들의 변환 정보에 대한 테이블이다. m → ft 등 출력되는 모듈과 입력되는 모듈 사이에서 그 단위가 다른 경우에 그 변환 방법을 저장하는 테이블이다.
- **LINKING_INFO** : 모듈 간의 인풋/아웃풋 정보를 연결시켜 주는 역할을 한다. 각각의 모듈이 통합을 위해서 프로그래밍 되는 것이 아니기 때문에 어느 값이 어느 값으로 넘어가야 되는지에 대한 정의를 해준다.
- **USER_INFO** : 실제 프로젝트를 관리하고 사용할 사용자에 대한 정보를 가지고 있다. 각각의 프로젝트는 기본적으로 로그인한 사용자만 관리하도록 되어있다. 향후 보안 레벨에 따라 다른 사용자가 검색을 통해 유사한 해석 및 최적화에 관한 내용을 검색할 수 있도록 설계되어 있다.

- PROJECT_INFO : 사용자 별로 큰 해석 및 최적화 환경을 제공 하는 Grouping 역할을 한다. 유사한 해석 및 최적화 문제를 묶는데 사용된다.
- TASK_INFO : 실제 수행되어질 작업에 대한 정의를 한다. 해석과 최적화 등에 사용될 모듈과 각각의 초기값들을 세팅하여 진행한다. 이 작업들은 JOB_INFO 테이블에 등록되어 스케줄러에 의해 순차적, 자동적으로 수행된다.
- TASK_VALUE_INFO : 실제 수행된 모듈들에 의해서 생긴 인풋/아웃풋 값들을 저장한다. VALUE_INFO로부터 기본적인 정보를 가져오고 모듈이 수행한 결과를 통해 새롭게 값을 업데이트한다.
- JOB_INFO : 스케줄러에 의해서 수행되어야 할 모듈들의 정보를 담고 있다. 이 모듈들이 순차적으로 수행되면서 사용자는 결과가 나올 때까지 모니터 앞에서 기다릴 필요없이 자동적으로 수행된 결과를 나중에 확인하면 된다. 이 정보에는 현재 어떤 모듈이 수행되고 있는지 파악할 수 있다.

이들 클래스들을 특성별로 분류, 정리하면 표 3과 같다.

표 3 클래스 구조

기초 자료 관리 클래스	
모듈 정보	MODULE_INFO, MODULES
해석, 최적화 모듈 정보	ANALYSIS_MODULE, OPTIMIZE_MODULE
해석, 최적화 모듈 작업 순서 정보	ANALYSIS_TYPE, OPTIMIZE_TYPE
태스크 형식	TASK_TYPE
프로젝트 접근 권한	ACCESS_LEVEL
값에 대해 각 모듈이 요구하는 정보	VALUE_INFO
값의 형식	VALUE_TYPE
값의 단위	VALUE_UNIT
값의 변환	ROLE_INFO
연결 정보	LINKING_INFO
해석 및 최적화 자료 관리 클래스	
사용자 정보	USER_INFO
프로젝트 정보	PROJECT_INFO
태스크 정보	TASK_INFO
해석 및 최적화 결과	TASK_VALUE_INFO
해석 및 최적화 수행 관리 클래스	
작업 정보	JOB_INFO

3.5 데이터베이스 설계 결과

표 4는 설계된 데이터베이스 스키마의 한 예로써, VALUE_INFO 클래스의 스키마이다.

설계된 데이터베이스 및 유도무기 설계 해석코드를 통합하여 구성한 유도무기 통합설계 프레임워크를 그림 6에서 볼 수 있다(이재우 등, 2005).

표 4 VALUE_INFO 클래스 스키마

Name	key	type	Description	예제값
vi_write		DATE	해당 값의 작성날짜	SYS DATE
vi_modify		DATE	해당 값의 수정날짜	SYS DATE
vi_mid	FK	number(8)	Module ID	1
vi_kind		char(1)	해당 값의 종류	I
vi_name		varchar2(255)	해당 값의 이름	Input File1
vi_type	FK	number(8)	해당 값의 형식	5
vi_lower		number(10,4)	최소값(최적화 설계영역 정의에 사용)	0
vi_upper		number(10,4)	최대값(최적화 설계영역 정의에 사용)	0
vi_unitid	FK	number(8)	단위 정보	0
vi_init		varchar2(200)	초기값	input.txt
vi_info1		varchar2(200)	부가 정보	
vi_info2		varchar2(200)	부가 정보	
vi_desc		varchar2(2000)	Value 설명	MOST Input file

vi_mid : 1은 MOST 모듈의 고유 값이다.

vi_kind : I -> 입력, O -> 출력

vi_type : 값에 종류에 대한 고유 값이다. 5: 파일명

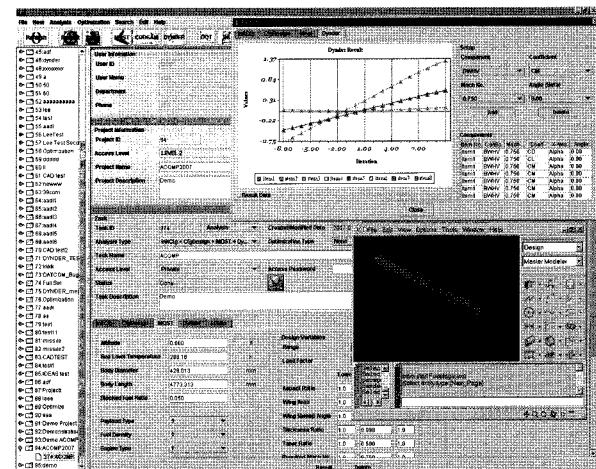


그림 6 유도무기 통합설계 프레임워크(이재우 등, 2005)

8. 결 론

본 연구는 유도무기 MDO프레임워크를 위한 데이터베이스의 개념을 정립하고, 이를 토대로 해석 및 최적화 코드 별 입출력 특성에 적합한 데이터베이스 스키마를 설계 및 구축하였다. 먼저 유도무기 개발에 적합한 데이터베이스에 관한 사용자의 요구조건을 분석하여 데이터베이스의 설계를 하였다. 설계 초기단계에서 분석된 사용자의 요구조건에 맞게 데

데이터베이스에서 관리되어야 할 데이터와 각 해석 코드에서 요구하는 데이터의 분석을 통해 데이터베이스에 저장해야 할 자료들을 정리하였고, 개념적 설계단계에는 앞서 분석된 결과를 토대로 데이터베이스의 구성체계를 정립하였다. 데이터베이스에 저장될 객체와 각각의 속성들은 클래스 디어그램으로 표현하였고, 논리적 설계단계에서는 실제 테이블에 대한 정규화 작업을 통한 테이블 사상을 거쳐 실제 테이블 구축을 하였다. 유도무기 설계에 사용되는 데이터의 특성을 고려하여 객체관계형 데이터베이스를 사용하였으며, 그 중에서 상용으로 가장 많이 사용되는 ORACLE 9i를 사용하여 실제 데이터베이스 환경을 구축하였다.

개발된 데이터베이스는 유도무기 통합설계 프레임워크에 통합되어 프레임워크 내의 데이터의 저장, 검색 및 관리 역할을 수행하며, 유도무기 설계에 적용하여 그 효용성을 검증하였다.

감사의 글

본 연구는 한국과학재단의 특정기초연구(2007년 특정연구, 과제번호 R01-2006-000-10744-0)와 BK21 ST·IT 융합공학인력양성사업단의 지원으로 수행되었으며, 연구비 지원에 깊은 감사를 드립니다.

참 고 문 헌

- 이승진, 도상윤 외** (2005) 유도무기 MDO를 위한 데이터베이스 개발, 13회 유도무기 학술대회 논문집.
- 이재우, 김종화, 정갑주** (2004) 유도무기 통합설계 환경 구축을 위한 DBMS 연구 최종보고서, 국방과학연구소.
- 이재우, 정갑주, 황호연** (2005) 유도무기 다학제간 최적설계 기법 연구 최종보고서, 국방과학연구소.
- 전권수, 이재우** (2001) 효율적 제품설계를 위한 다분야 통합 최적설계 기법 고찰, 항공우주학회지, 29(5).
- 황진용, 정주영** (2003) 다분야 통합환경에서의 데이터베이스 설계 연구, 한국항공우주학회지, 31(5).
- A. O. Salas, J. C. Townsend** (1998) Framework Requirements for MDO Application Development, AIAA Paper 98-4740.
- Jeffrey D. Ullman, Jennifer Widom** (2007) First Course in Database Systems, Prentice Hall.
- Oracle 9i: Application Developer's Guide - Object-Relational Features, Release 2 (9.2), Part No. A96594-01, March 2002.**
- Oracle 9i: Database Concepts, Release 2 (9.2), Part No. A96524-01, March 2002.**
- Ramakanth S. Devarakonda** (2001) Object-relational database systems-the road ahead, *Crossroads*, 7(3), pp.15~18.