

# 데이터관리론 Homework#2

201823869 조성우



## INDEX

<b>1. Contrast the following terms (p.80 1-3)</b> .....	<b>1</b>
a. data dependence / data independence .....	
b. structured data / unstructured data .....	
c. meta data / data .....	
d. repository / database .....	
e. entity / enterprise data model .....	
f. data warehouse / data lake .....	
g. personal databases / multi-tiered databases .....	
i. enterprise data model / conceptual data model .....	
 <b>2. Problems and Exercises 1-49 (p.82)</b> .....	<b>2</b>

## 1. Contrast the following terms (p.80 1-3)

### a. data dependence / data independence

File processing systems에선 File descriptions(metadata)가 각 Database application 프로그램내에 저장된다. 이에 따라 파일 구조의 어떤 변경이 이루어 질 시에는, 그 파일에 접근하는 모든 개별 프로그램들을 위한 File description(metadata)의 변화를 요구하는데.

이러한 시스템의 데이터 의존속성을 Data dependence(데이터 의존성)이라고 하며

이러한 의존적 데이터구조를 가진 시스템에서의 데이터 변화는 불필요하고 중복적인 개별적 변화반영 작업을 요구하며, 해당 작업중의 실수로 인해 데이터의 통일성을 저해할 수도 있다.

이러한 File processing system의 데이터 의존성(data dependence)의 단점을 극복하기 위해 DBMS(Database Management System)가 등장하였다.

DBMS에서는 Data descriptions(metadata)들과 그 Data를 쓰는 Application programs들이 분리되는데 이 분리된 Data descriptions들은 repository라고 불리는 중앙장치에 저장된다.

이러한 DBMS는 데이터 독립성(Data independence)를 보장하며 이에 따라 DBMS의 데이터 독립적 구조 하에선 데이터베이스 내의 데이터와 이들을 사용하는 응용 프로그램이 서로 영향을 받지 않게 된다.

따라서 DBMS에선 그 데이터를 처리하는 응용 프로그램의 수정 없이도 해당되는 데이터의 변화를 가능하게 하며 이를 Data가 independent하다고 말한다.

### a. structured data / unstructured data

데이터베이스에 저장되는 데이터는 Structured data와 Unstructured data를 포함한다.

Structured data는 기록,저장될 수 있는 객체,사건에 관련한 사실들로 말해지는 데이터를 얘기하며 가장 중요한 Structured data는 정수형,문자형,날짜형 이다.

Structured data는 표의 형태로 저장되며 전통적 Database와 Data warehouse에서 찾아

볼 수 있다.

Unstructured data란 서류, 이메일, 트윗, 페이스북, GPS정보, 지도, 이미지, 소리, 비디오 등 뉴미디어의 대두와 함께 주목받기 시작한 비정형화 된 데이터를 얘기하며 Multimedia data로서도 말할 수 있다.

오늘날 Structured data와 Unstructured data는 True multimedia environment를 창조하기 위해 종종 같은 데이터베이스에 결합되기도 한다.

## **b. metadata / data**

역사적으로, Data란 용어는 컴퓨터 미디어에 기록, 저장될 수 있는 객체나 사건들에 관한 사실을 말하는 것이었다.

이러한 데이터의 정의는 현시대의 새로운 현상을 반영하기위해 확장돼야 했는데 , 이는 전통적 data로 취급하던 structured data에 더해 각종 unstructured data를 더한 것을 포괄하며 이를 반영하여 확장된 현재 data의 정의는 'user의 환경에서 의미와 중요성을 가진 객체와 사건들의 저장된 대표성'이다.

이러한 데이터는 몇몇 적절한 맥락에 위치될 때에만 유용해지는데. 이 때 end-user data의 특성, 속성과 그 data가 사용되는 문맥을 기술하는 데이터를 meta data라고 한다.

일반적으로 metadata는 data의 이름, 정의, 길이(또는 크기), 그리고 그 data에 허용가능한 value를 포함한다.

metadata는 데이터의 특성을 기술하지만 그 데이터로부터 분리되며, 데이터베이스의 파트로서 저장되고 데이터나 정보에 접근할 때 그 데이터에 해당되는 메타데이터가 불러들여질 수 있다.

명료한 의미를 가지지 않은 데이터를 관리하는 것은 혼란스러우며 오해를 불러일으켜 문제의 가능성을 가질 수 있기 때문에 메타데이터를 관리하는 것은 매우 중요하다.

## **c. repository / database**

database와 repository는 database environment의 구성요소이다.

repository는 모든 데이터 정의,데이터 관계, 화면과 보고서 포맷과 다른 시스템 구성요소들을 위한 중앙화 된 지식베이스로 ,3번에서 설명한 것과 같이 각 데이터의 등

을 기술하는 모든 metadata가 저장되는 장소이다.

Database란 논리적으로 관련된 데이터들의 조직화된 하나의 컬렉션으로, 일반적으로 한 조직내의 다수의 사용자들의 정보요구를 충족시켜 주기위해 디자인된다.

Repository는 데이터의 정의의 기술을 포함하는 반면에, database는 data의 존재자체를 포함한다.

#### **d. Entity / enterprise data model**

Data를 모델링 할 때 현실세계에 존재하는 objects중 End user가 사용하게 될 대상들을 Entity와 relationship으로 식별하는데

이때, Entity는 공통된 성질을 가지며 서로 구분할 수 있는 현실세계의 객체들을 대상으로 해야 하고, 세부적으로 사람, 장소, 객체, 사건 또는 개념을 말하기 때문에 한 가지 명사와 비슷한 개념이라 볼 수 있다.

이러한 Entities 사이의 relationship와 성격에 따라 데이터를 논리적으로 구성하기 위해 사용되는 Graphical system을 Data model이라고 하고, 이 Data modelling 프로세스의 가장 첫번째 단계가 거시적인 관점에서 앞으로 다루고자 하는 데이터들의 범위와 entity와 relationship으로 식별하고자 하는 대상을 특정하는 enterprise data model 단계이다.

#### **e. data warehouse / data lake**

data warehouse란 사적, 업무그룹, 부서, ERP 등 분산적으로 운영되는 각각의 다양한 데이터베이스로부터 축적된 데이터들을 공통의 형식으로 통합적으로 변환, 저장하여 조직내 의사결정을 종합적으로 지원하는 정보관리 시스템을 얘기한다.

Data lake는 big data의 맥락속에서 소개된 상대적으로 최신의 enterprise level 개념으로 정형/비정형의 가공되지 않은 원형 데이터들을 모아놓는 통합 데이터 저장소이다.

data lakes는 전통적인 data warehouses로부터 차별화되는 다양한 성격들을 가지고 있다.

data lakes는 전형적으로 미리 정의된 데이터 모델 또는 스키마에 기반하지 않으며 대개 '모든 것'들을 수집하는데, 전통적 data warehouse는 특정 목적에 따라 특정 사용자에게 사용되기 위해 미리 정의된 데이터모델에 기반하여 데이터를 수집하는 데에 반해 data lake는 미리 정해지지 않은 다목적의 다수의 사용자를 가정하고 데이

터를 수집하는 개념으로, data item간의 새롭고 창의적인 연결성을 확인하기에 특히 좋은 개념이다.

#### **f. personal databases / multi-tiered databases**

personal database는 한명의 사용자(single-user)를 지원하기 위해 디자인된다. 주로 개인컴퓨터에 존재해왔으며 현재 스마트폰 태블릿 패블릿 등등 사용가능 장치가 증가해가는 중이다.

이 데이터베이스의 목적은 효율적 방식에서 작은 양의 데이터를 관리할 능력을 유저에게 제공하는 것에 있으며, 개인적 생산성을 개선시킬 수 있기 때문에 널리 쓰여왔다.

그러나 personal database는 타 사용자와 데이터를 공유하기 어려운 문제 때문에 매우 작은 조직같이 personal database 데이터의 사용자간 공유 이슈가 발생하지 않을 특별한 상황에 국한되어 사용되어야 한다.

이러한 single user 대상의 personal databases의 공유적 한계를 극복할 필요가 있는 다수의 사용자를 대상으로 하는 대부분의 현대 장치들은 multi-tiered architecture을 사용하게 된다.

여기서 multi-tiered는 말 그대로 여러 tier로 구성되었다는 뜻이며, Client tier, Application/web tier, Enterprise tier 로 구성될 수 있다.

이 multi-tiered databases는 비즈니스 내의 한 부서나 한 division같은 일반적으로 한 업무그룹보다 큰 조직을 지원하도록 의도되며 personal databases와 달리 사용자간 데이터의 공유가 용이하도록 설계된 데이터베이스이다.

#### **I. enterprise data model / conceptual data model**

Enterprise data model이란 거시적 관점에서 관계자가 관심있어 하는 area에 존재하는 모든 객체, 사건 등을 Entity와 relationship으로 식별하여 구조화 시키는 data model이며 데이터베이스 개발의 가장 첫번째 단계이다.

해당 단계에서 차후 개발할 조직적 데이터베이스의 일반적인 범위와 내용물을 구체화한다.

Conceptual data model 이란 데이터 모델링 프로세스에서 향후 논리적 데이터 모델(logical data model)과 물리적 데이터 모델(physical data model)의 데이터 구조적 정렬을 지원하기 위해 필요한 기반적 모델링 단계로,

실세계에 존재하는 객체(데이터베이스에 필요한)를 추상적인 개념인 개체(Entity)와 관계

(Relationship)로 식별하는 과정이며 추상화 수준이 높고 포괄적인 수준의 모델이다.

## 2.Problems and Exercises 1-49 (p.82)

Table for Problem and Exercise 1-49

	All Entities?	All Attributes>	Technology independent?	DBMS Independent?	Record Layouts?
Enterprise	Y	N	Y	Y	N
Conceptual	Y	Y	Y	Y	N
Logical	Y	Y	N	Y	Y
Physical	Y	Y	N	N	Y