

# データベース設計

Hiroshima Institute of Technology

1

## 授業計画

- 第1回 ガイダンス・データベースの基本概念
- 第2回 データモデル
- 第3回 関係代数
- 第4回 データベース設計
- 第5回 リレーションの正規化
- 第6回 中間まとめ
- 第7回 関係データベース言語(SQL1)
- 第8回 関係データベース言語(SQL2)
- 第9回 計算機実習
- 第10回 データの検索機構 MySQL実習
- 第11回 トランザクション管理 MySQL実習
- 第12回 障害回復 MySQL実習
- 第13回 分散データベース MySQL実習
- 第14回 期末まとめ
- 第15回 応用技術と将来動向 MySQL実習

## 4. データベース設計

### 講義内容

- データベース設計の概要
- 概念データモデルのモデリング技法
- 論理データモデルのモデリング技法

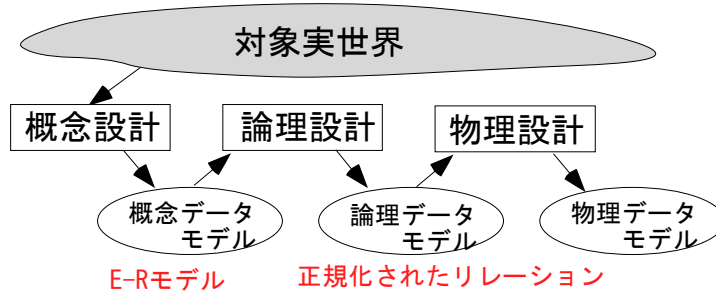
## 4. データベース設計

### 1. データベース設計の概要

- **データベース設計**は、データベースによってデータを管理できるように、**実世界を抽象化**してデータモデルを作成していく作業といえる。
- データモデルは、データベースをどのように構成するかということを定義したものである。
- このデータモデルを作成していく作業は、**概念設計**、**論理設計**および**物理設計**という3つの段階により行われる。
- 概念設計の出力が**概念データモデル**であり、論理設計はその概念データモデルを入力して**論理データモデル**を出力する作業である。
- そして、物理設計は論理データモデルを入力して**物理データモデル**を出力する作業である。

## 4. データベース設計

### 1. データベース設計の概要



### データベースの設計

Hiroshima Institute of Technology

5

## 4. データベース設計

### 1. データベース設計の概要

#### ■ 概念設計

- 概念設計は、データベースによって管理の対象とするものを**現実の世界から抽出して概念モデルを作成**する。
- 概念データモデルは、計算機への実装を前提としておらず、**特定のDBMSには依存していない**。
- 概念データモデルの作成にあたっては、**E-Rモデル**（実体関連モデル）がよく使用される。
- E-Rモデルは、その名の示すとおり、**実体と関連**によってモデルを作成する。
- 実体は現実の世界を構成する**実体そのもの**、関連は実体間の**つながり**を表現するものである。
- また、実体や関連は**属性**を持つことができる。

Hiroshima Institute of Technology

6

## 4. データベース設計

### 1. データベース設計の概要

#### ■ 論理設計

- 論理設計では、概念設計によって作成された概念モデルを、特定のデータモデルに対応した論理モデルに変換する。
- リレーショナルデータベースによってデータを管理するのであれば、E-Rモデルからリレーショナルモデルを作成していく。
- E-Rモデルからリレーション（表、テーブル）への変換は機械的に行うことができる。
- そのままりレーションに変換しただけでは、リレーショナルモデルとして適切な形式にならない場合がある。
- 論理設計ではテーブルをリレーショナルモデルとして適切な形式に変換する作業である正規化を行う。

Hiroshima Institute of Technology

7

## 4. データベース設計

### 1. データベース設計の概要

#### ■ 物理設計

- 物理設計の段階になって初めてデータベースとしての性能について考慮する。
- 具体的には、論理設計において正規化した表の定義を崩したり、インデックスを定義したりして性能が向上するようにモデルを修正していく。
- また、物理設計では使用するデータベースに依存する機能を使用することもある。
- 物理設計によって修正されたモデルを物理モデルと呼び、このモデルをもって実際にデータベースによって管理することができる形式となる。

Hiroshima Institute of Technology

8

## 4. データベース設計

### 2. 概念データモデルのモデリング技法

#### ■ E-Rモデルの表現要素

- E-Rモデルは、**エンティティタイプ**（実体型）というレコード型を**リレーションシップ**（関連）で結びつけて、現実世界のデータ体系を定義する。

#### ■ エンティティタイプ：

- エンティティタイプとは、現実世界の中で同じ性質を持つ個々のモノやデータを**抽象化したデータ型**のことである。
- 一般に、エンティティタイプにはその性質を表現する複数の属性が存在する。
- エンティティタイプをERDで表現するには、長方形でエンティティタイプを表し、その中にエンティティ名を記述する。属性も表現したい場合には長方形を2段に分割し、上段にエンティティ名を下段に属性名の並びを記入する。

Hiroshima Institute of Technology

9

## 4. データベース設計

### 2. 概念データモデルのモデリング技法

#### ■ エンティティタイプ：

- **主キー**を表す場合には、主キーを構成する属性名または属性名を組に**実線の下線**を付ける。
- **外部キー**を表す場合は、外部キーを構成する属性名または属性名を組に**破線の下線**を付ける。
- 主キーはそのエンティティタイプを**一意に特定するための属性名または属性名を組**であり、外部キーについては、他のエンティティタイプを一意に特定するための属性名または属性名を組である。

エンティティタイプ
属性名1, 属性名2, ...属性名n

エンティティタイプの記述方法

Hiroshima Institute of Technology

10

## 4. データベース設計

### 2. 概念データモデルのモデリング技法

#### ■ リレーションシップ：

- リレーションシップは、エンティティタイプ間の関連を示すものである。
- ERDで表現する方法は2つある。
- 属性を持つリレーションシップタイプは、ひし形で表現し、その中にリレーションシップ名を記入する。属性を記述したいときは横線を引き属性名を記入する。
- 属性を持たないリレーションシップタイプは、エンティティタイプを線で結ぶ。



## 4. データベース設計

### 2. 概念データモデルのモデリング技法

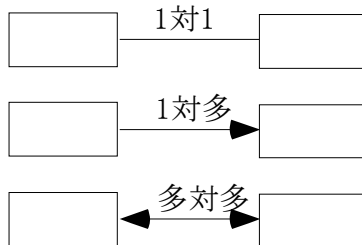
#### ■ カーディナリティ：

- エンティティタイプ間のリレーションシップにおいて、エンティティインスタンスの対応関係のことをカーディナリティという。
- ERDにおけるカーディナリティには、**1対1**、**1対多**、**多対多**の3つがある。
  - ✓ **1対1**：「発注」と「納品」のように同じ単位で行われる関係は、1対1の関係になる。
  - ✓ **1対多**：一つの商品が繰り返し発注される場合、「商品」と「発注」は、1対多の関係になる。
  - ✓ **多対多**：一人の顧客が複数の商品を購入し、一つの商品は複数の顧客に販売される場合、「顧客」と「商品」は多対多の関係になる。通常は新たなエンティティタイプを生成し、1対多の関係になるように再設計されることが多い。

## 4. データベース設計

### 2. 概念データモデルのモデリング技法

#### ■ カードィナリティ：



#### ERDでのカードィナリティ

## 4. データベース設計

### 2. 概念データモデルのモデリング技法

#### ■ スーパタイプとサブタイプ

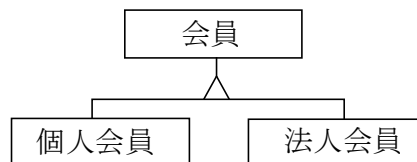
- スーパタイプとサブタイプは、**汎化－特化の関係（is-a 関係）**を表すために利用する。
- スーパタイプは、共通の属性、各サブタイプにはそれぞれに固有の属性を格納する。
- スーパタイプとサブタイプ構成となるか否かは、以下の条件による判断できる。
  - ① 汎化－特化の関係であること。
  - ② テーブルの主キーが一致すること。
  - ③ サブタイプ間に排他的関係が存在すること。
  - ④ サブタイプを個別に参照する要件があること。

## 4. データベース設計

### 2. 概念データモデルのモデリング技法

#### ■ 会員がスーパータイプとサブタイプ構成なる例:

- **会員と個人会員, 法人会員との関係**
- 会員は個人会員と法人会員があり, 会員番号は一意的な番号で管理され, 個人会員にはクレジットカード番号, 法人会員には請求書送付先などの排他的属性が存在している。



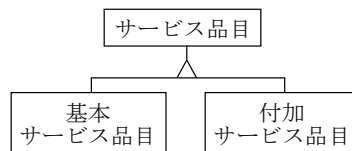
会員についてのスーパータイプとサブタイプ

## 4. データベース設計

### 2. 概念データモデルのモデリング技法

#### ■ サービス品目がスーパータイプとサブタイプ構成なる例

- **サービス品目と基本サービス品目, 付加サービス品目との関係**
- サービス品目は基本サービス品目と付加サービス品目があり, サービス品目番号は一意的な番号で管理され, ユーザからはいずれかのサービスを選択し, 各サービスには排他的属性が存在している。



サービス品目についてのスーパータイプ・サブタイプ



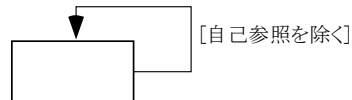
## 4. データベース設計

### 3. 論理データモデルのモデリング技法

- 概念データモデルが定義できたら、個々の業務特性や計算機システムの実装に適合する**DBMSを選定**し、論理データモデルを定義する必要がある。

#### ■ 階層データモデル

- 自己参照制約をもった階層モデルのERD
- 親エンティティと子エンティティが1対多のリレーションシップを自己参照を許さずに繰り返していることから、階層化していることを意味している。



自己参照制約をもった階層データモデルのERD

Hiroshima Institute of Technology

17

## 4. データベース設計

### 3. 論理データモデルのモデリング技法

#### ■ 階層データモデル

- 現実世界には、会社の組織のように階層構造として表現できるものが数多くある。
- しかし、これらを階層データモデルで表現すると、親子関係をもたせたアクセスパスしか検索できない等の欠点があり、階層データモデルを採用した**商用DBMSはほとんど存在していない**。

Hiroshima Institute of Technology

18

## 4. データベース設計

### 3. 論理データモデルのモデリング技法

#### ■ ネットワークデータモデルデータモデル

- アクセス経路を網構造とするネットワークデータモデルは、階層データモデルの親レコードを1つしか持てないという制約を撤廃したデータモデルである。
- レコード間に**ループを許容**しており、階層データモデルに比較してアクセスパスなどの自由度が高いデータモデルである。

## 4. データベース設計

### 3. 論理データモデルのモデリング技法

#### ■ 関係データモデル

- ERDで表現された概念データモデルを関係データモデルを用いて論理データモデルに変換する場合、**多対多のリレーションシップを排除**する必要がある。
- 関係の間に**新たにエンティティを挿入**して、1対多のリレーションシップに変更する。
- 実体を表すエンティティについては、**正規化**を行い、**主キー**でタブルを識別でき、**外部キー**で分割したリレーションのタブルを参照できるようにする。
- また、参照関係にあるリレーションの間には、**参照制約を維持するための規則を設定**する。

## 4. データベース設計

### まとめ

- データベース設計の概要
- 概念データモデルのモデリング技法
- 論理データモデルのモデリング技法