

# 応用技術と将来動向

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

## 1 2. 応用技術と将来動向

### 講義内容

- XMLデータベース
- OLAP (online analytical processing)
- ストアドプロシージャ
- オブジェクト指向データベース
- レプリケーション
- Google File System
- BigTable
- データベースの将来動向

## 1 2. 応用技術と将来動向

### 1. XMLデータベース

- 現在Webページの記述には、1989年に出現した**HTML** (hypertext markup language) という**マークアップ言語** (markup language) が主に用いられている。
- マークアップ言語は、文章の構造や見栄えなどの情報であるタグ (tag) を文書の中に付加したものである。
- さらに、1998年には**XML (extensible markup language)** が出現し、構造化文書の記述だけでなく、Webなどを介した一般的なデータの情報交換にも利用されるようになった。
- このXMLを扱うための機能を有するデータベースを**XMLデータベース (XML database)** という。

## 12. 応用技術と将来動向

### 1. XMLデータベース

- XMLデータベースには、XMLデータをそのまま既存のデータベースに格納する“XML Enableデータベース”と、XML文書をその構造のまま格納することができる専用のデータベースである“ネイティブXMLデータベース”の2つがある。
- また、現在ではXPathやXQueryで検索するデータベースをXMLデータベースと呼ぶことが多い。
- 現在、広く用いられているリレーショナルデータベースでは、一度作成されたデータ構造を運用中に変更することが一般的に困難なのに対し、XMLデータベースは非常に拡張性が高いと言われている。
- したがって、途中でデータ構造が変化することを前提としたシステム開発を比較的容易に行うことができる。

Hiroshima Institute of Technology

5

## 12. 応用技術と将来動向

### 1. XMLデータベース

#### XML Enableデータベースの例

EMP表

ENO	ENAME	AGE
0001	佐藤太一	33
0002	山川 洋	44
0003	浦野 正	53

```
<EMP>
  <ROW>
    <ENO>0001</ENO>
    <ENAME>佐藤太一</ENAME>
    <AGE>33</AGE>
  </ROW>
  <ROW>
    <ENO>0002</ENO>
    <ENAME>山川 洋</ENAME>
    <AGE>44</AGE>
  </ROW>
  <ROW>
    <ENO>0003</ENO>
    <ENAME>浦野 正</ENAME>
    <AGE>53</AGE>
  </ROW>
</EMP>
```

Hiroshima Institute of Technology

6

## 2. OLAP (online analytical processing)

- **OLAP (online analytical processing)** は、管理者などがデータを多角的に検索・集計して問題点や解決策を発見する機能であり、**オンライン分析処理**または**多次元分析処理**とも呼ばれる。
- OLAPは、地域、製品、時間などの項目を軸とする多次元空間でデータを捉えて、それらの軸に対する組合せをオンラインで行うことにより、**さまざまな集計データを視覚化しデータ分析**する処理である。

## 2. OLAP (online analytical processing)

- OLAPの主な手法としては、ダイシング、スライシング、およびドリルダウンがある。
- **ダイシング(ダイス)** は、サイコロを転がすように参照するデータ軸を変えることである。
- **スライシング(スライス)** は、2つの軸を選択して、特定の面だけを参照することである。
- **ドリルダウン** は、“穴をあけて下がっていく”意味から集約されたデータから詳細データへと対象を絞り込んでいくことである。

## 2. OLAP (online analytical processing)

- たとえば、企業の販売実績であれば、「地域別」、「製品別」、「月別」などの軸を設定し、
  - 「地域ごとの製品販売実績」、「製品ごとの地域別販売実績」のように軸を入れ替えて比較するダイシング、
  - 「地域別」や「製品別」を固定して「月別」の推移を比較するスライシング、
  - そして、ある地域におけるある商品の販売を月別ではなく、さらに細かな日別のデータを表示するドリルダウンなどの操作を繰り返すことで行われる。

## 3. ストアドプロシージャ

- **ストアドプロシージャ (stored procedure)** は、データベースに対する一連の処理をまとめたプロシージャ(手続き)にして、DBMSに保存(永続化)したものである。
- ストアドプロシージャは、**アプリケーションからSQL 文により呼び出されると、DBMSサーバ内で動作する。**
- 通常、複雑な SQL 文の呼び出しを、論理的に一つの処理単位にまとめて、簡単にその名前呼び出せるようになっている。
- 1つのプロシージャには、複数の SQL 文が含まれ、繰り返しや条件分岐などの制御構造をもつこともある。

### 3. ストアドプロシージャ

- ストアドプロシージャを利用することにより、次のようなメリットがある。
  - ① DBMS に1つずつ SQL 文を発行する必要がなくなる。
  - ② ネットワーク上のトラフィックを削減できる。
  - ③ あらかじめ処理内容が DBMS に登録され、構文解析や機械語への変換が済んでいるため、**処理時間が軽減される**。

### 4. オブジェクト指向データベース

- **オブジェクト指向データベース**(OODB : object oriented database)は、オブジェクト指向プログラミングで使うオブジェクトの形式で表現されるデータを格納するデータベースであり、オブジェクトデータベースともいう。
- OODBへの関心は、オブジェクト指向のシステム開発およびプログラミングの急速な浸透と発展において、アプリケーションプログラムが生成したオブジェクトを保存し、永続化する必要性とともに高まってきた。
- 1980年代後半には商用の**オブジェクト指向データベース管理システム**(OODBMS)が市場に投入されたが、オブジェクト管理システムの位置づけであり、CAD/CAMといったエンジニアリング分野が中心であった。

## 4. オブジェクト指向データベース

- 2004年頃からOODBMSが注目されるようになり, OODBMSの利点を残しつつRDBMSの使い易さを取り入れたOODBMS製品が出現し, より広い分野への適用性が高まっている。

## 4. オブジェクト指向データベース

### オブジェクト指向データモデルの概念

- まず, オブジェクト指向データモデルでは, 実世界のすべてを **オブジェクト(object)** として捉え, 各オブジェクトはシステム全体で一意的な **オブジェクト識別子(object identifier)** をもつ。
- オブジェクトは1つ以上の **属性(attribute)**, および1つ以上の **メソッド(method)** をもつ。
- 属性は **プロパティ(property)** とも呼ばれる。オブジェクト内の属性は, 値(複数でも可)をもち, その値もオブジェクトである。メソッドは属性の値に作用するものである。

## 4. オブジェクト指向データベース

- オブジェクト内の属性とメソッドは、そのオブジェクト内に**カプセル化(encapsulation)**され、それらにアクセスするためには、オブジェクトに**メッセージパッシング(message passing)**する必要がある、公開されたメッセージ送信インターフェースが用いられる。
- 同一の属性とメソッドをもつ全てのオブジェクトの集合が**クラス(class)**である。
- クラス間には、**クラス階層(class hierarchy)**という階層をもたせることができ、階層の下位のクラスである**サブクラス(subclass)**は、上位のクラスである**スーパークラス(super class)**の全ての属性とメソッドを**継承(inheritance)**する。

## 4. オブジェクト指向データベース

- サブクラスは、継承した全ての属性とメソッドの他に新たに属性とメソッドを追加できる。
- サブクラスの実現値である**インスタンス(instance)**は、スーパークラスのインスタンスでもある。
- サブクラスとスーパークラスの間には、互いに**特化(specialization)**と**汎化(generalization)**の関係がある。サブクラスはスーパークラスを特化したものであり、スーパークラスはサブクラスを汎化したものである。



### 4. オブジェクト指向データベース

- **OODBMS**は、オブジェクト指向システム開発、およびオブジェクト指向プログラミングの考え方をデータベースシステムに取り入れたものであり、複雑なデータ構造やマルチメディア・データ(文書／イメージ／画像等、多種多様なデータ)を対象とし、オブジェクト指向の概念により開発効率の向上も狙ったものである。
- オブジェクト指向データベースを設計する際に重要なことは、どのようにして共通の構造と振る舞いをもつ**クラス(class)**を見つけるかということである。

### 4. オブジェクト指向データベース

#### 銀行システムの当座預金クラスの例

```
class 当座預金
  inherit 銀行口座
  prosperities
    口座番号: Integer
    所有者: 顧客
    残高: Money
  operations
    create;
    calculate_changes(開始日,終了日: Date): 費用: Money;
    deposit(金額: Money);
    withdraw(金額: Money);
end当座預金;
```

## 5. レプリケーション

- **レプリケーション(replication)**とは, あるデータベースと全く同じ内容の複製(レプリカ)をネットワーク上に複数配置してDBMSの負荷分散やフォールトトレランス性を強化する手法である。
- ここで, **フォールトトレランス(fault tolerance)**とは, システムに障害が発生したときに, 正常な動作を保ち続ける能力のことである。

## 5. レプリケーション

- 分散データベースでは, 複数のデータベースを1つのデータベースと見なしてトランザクション処理が行われる。
- 分散データベースのレプリケーションは,  
**同期レプリケーション(synchronous replication)**と  
**非同期レプリケーション(asynchronous replication)**がある。
- 前者はデータの更新時に同じトランザクション内で重複データも更新し, 後者は独立した別レプリケーション機能によってトランザクション外で遅延して重複データを更新する。

## 6. Google File System

- **グーグルファイルシステム** (GFS : Google File System )とは、グーグルが独自に開発した分散処理ファイルシステムのことである。
- 検索エンジンで有名なGoogleには、クローラが集めてきた大量のコンテンツ、GMailなどの利用者が保存する大量のファイル、Google MapsやGoogle Documentなどが表示する大量のデータなど、大量のデータが保存されている。
- この大量のデータを安全に保存するための**分散ファイルシステム**がGFSである。

Hiroshima Institute of Technology

21

## 6. Google File System

- GFSは、管理するマスターサーバが1つ存在し、その管理下で大量のデータを保存する“**シングルマスター構造**”になっている。
- GFSは、Google内部においてクローラやインデックスといったバッチ処理に活用されている。
- GFSの特徴としては**同じファイルを異なるマシンに重複して持たせる**ことで、一部のマシンが故障してもファイルが失われないという点が挙げられる。
- Googleでは何万台ものサーバ用マシンが常時稼働しているので、必ずしも全てのマシンが完全な状態で稼働しているわけではない。
- 耐故障性の高い分散ファイルシステムを持つことは巨大データを扱う上で必須である。

Hiroshima Institute of Technology

22

## 6. BigTable

- **BigTable (ビッグテーブル)** は, Googleが開発したGFS上のデータベースシステムである。
- Bigtableは, Google検索をはじめ, YouTubeやGoogle Map, Google Earth, Google Analytics, Google App Engineなど, グーグルの数多くのプロジェクトの基盤として利用されている。
- Google などのWeb 検索サイトは, 事前に世界中のWeb ページを収集しておき, 利用者からの求めに応じ, 収集したWeb ページの情報を参照して適切なWeb ページを案内するが, 1 台のマシンでは取扱いが難しい大量のWeb ページデータの保存や処理が課題となる。
- この課題解決のために, **GFS, MapReduce, BigTable** という三つの技術が開発された。

## 6. BigTable

- **GFS** は, 多数のマシンのディスクを組み合わせ仮想的に一つのファイルシステムとして提供することで大量データの保存を可能にしている。
- **MapReduce** は, 多数のマシンを使って, 大量データを効率的に分散処理するための基盤を提供している。
- そして, Web ページなどの構造化された小さなデータを多数管理するために**BigTable** が提供された。

## 6. BigTable

## BidTableの概要を示す記事

## Bigtable: A Distributed Storage System for Structured Data

Fay Chang, Jeffrey Dean, Sanjay Ghemawat, Wilson C. Hsieh, Deborah A. Wallach, Mike Burrows, Tushar Chandra, Andrew Fikes, and Robert E. Gruber

## Abstract

Bigtable is a distributed storage system for managing structured data that is designed to scale to a very large size: petabytes of data across thousands of commodity servers. Many projects at Google store data in Bigtable, including web indexing, Google Earth, and Google Finance. These applications place very different demands on Bigtable, both in terms of data size (from URLs to web pages to satellite imagery) and latency requirements (from backend bulk processing to real-time data serving). Despite these varied demands, Bigtable has successfully provided a flexible, high-performance solution for all of these Google products. In this paper we describe the simple data model provided by Bigtable, which gives clients dynamic control over data layout and format, and we describe the design and implementation of Bigtable.

Appeared in:

OSDI'06: Seventh Symposium on Operating System Design and Implementation, Seattle, WA, November, 2006.

Download: PDF Version

(<http://labs.google.com/papers/bigtable.html>)

- 合計で数PB(ペタバイト)の天文学的規模のデータを, 全世界の**数10か所のデータセンター**に分散配置された**数10万台のサーバ**により, 各種サービスのスケーラビリティと高可用性を低コストで実現している。

Hiroshima Institute of Technology

25

## 6. BigTable

- Bigtableは, データの記憶のために一般的なリレーショナルデータベースを用いるのではなく, **分散KVS(Key-Value Store)**を用いるのが特徴となっている。
- KVSは, プログラミング言語のハッシュテーブル(連想配列)と同様に, **キー(key)**と**値(value)**の対からなるデータモデルに基づくデータの記憶方式である。
- したがって, KVSは**キー・バリュー型データストア(key-value store)**と呼ばれている。
- この技術が, 近年注目を集めている**クラウドコンピューティング(cloud computing)**の基盤技術として, その地位を確立しつつある。

Hiroshima Institute of Technology

26

## 12. 応用技術と将来動向

### 6. データベースの将来動向

- データベースの今後の動向を考える上で、**スケールアウト(scale out)**と**スケールアップ(scale up)**に着目することは重要であると考えられる。
- まず、**スケールアウト**とは、**サーバの数を増やすことで、サーバ群全体のパフォーマンスを向上させる**ことである。1台のサーバが仮に100人のユーザしか処理できないとしても、サーバを2台に増やして負荷を分散すれば200人のユーザに対応できるというのがスケールアウトの意味するところである。
- 一般的には、リソースの量に比例して全体のスループットが向上するシステムは**スケーラブル(scalable)なシステム**または**スケーラビリティのあるシステム**と呼ばれる。

Hiroshima Institute of Technology

27

## 12. 応用技術と将来動向

### 6. データベースの将来動向

- スケールアウトした場合、複数のサーバを連携して動作させることになるため、メンテナンスや障害発生時にもサービスを完全に停止させる必要がない点が利点である。
- しかし、サーバの台数が増えるために管理の手間が増大し、ソフトウェアのライセンス料金も高額になることが欠点である。

Hiroshima Institute of Technology

28

## 12. 応用技術と将来動向

### 6. データベースの将来動向

- 一方、**スケールアップ**は、既存のサーバを機能強化してパフォーマンスを向上させることである。
- **CPUやメモリなどを増強してサーバの能力を上げ、より高負荷に耐えられるように拡張することである。**
- スケールアップには、より強力な新しいサーバに交換することも含まれる。
- スケールアップした場合、1台のサーバでサービスを提供するために、複数台で運用するよりソフトウェアライセンスの料金が安く、構成が単純で管理しやすい点が利点である。
- しかし、その反面、複数台運用でないためスケールアップ時にはいったんサーバを停止する必要がある、障害が発生した場合の障害回復時間がかかることが欠点である。

## 12. 応用技術と将来動向

### 6. データベースの将来動向

- このように、スケールアウトとスケールアップには一長一短があるため、扱うデータによって適切な方法を選択することが望ましい。
- たとえば、複製や同期が容易であり、また複製しても問題の起きないデータを扱う場合には、スケールアウトが適しており、そうでなければ、スケールアップが適しているといえる。
- スケールアウトに着目してみると、データベース技術のトレンドが見えてくるようである。まず、複数のコンピュータを結合し**クラスター**（ぶどうの房）のようにひとまとまりとしたシステムである**コンピュータクラスター**(computer cluster)が登場したのが1983年である。

## 12. 応用技術と将来動向

### 6. データベースの将来動向

- クラスタ内の複数のコンピュータは、ネットワークを介して相互に接続され、1つのコンピュータシステムとして扱えるように制御されている。
- コンピュータクラスタは、気象予測などの**高性能技術計算 (HPTC : high performance technical computing)**に適用された。
- そして、低価格のPCを並列に用いてHPTCを実現しようとするのが**PCクラスタ (PC cluster)**であり、大学や研究所で盛んに採用された。
- このようなPCクラスタの考え方はさらに発展し、ネットワークに接続するPCを全て資源にできないかという発想につながって行った。

Hiroshima Institute of Technology

31

## 12. 応用技術と将来動向

### 6. データベースの将来動向

- そして、このPCクラスタは、世界中の遊休PCを活用しようとする**グリッドコンピューティング (grid computing)**に進化した。
- グリッドコンピューティングに成功例の1つに、地球外知的生命体を探索する**SETI(セチ)プロジェクト SETI@home**がある。
- このプロジェクトは、宇宙から飛来する電磁波を電波望遠鏡で受信し、その膨大なデータを分割してプロジェクトに参加している世界中のPCに送信して解析依頼をするものである。



Hiroshima Institute of Technology

32



## 12. 応用技術と将来動向

### 6. データベースの将来動向

- このような計算グリッドは、やがてデータ処理のグリッド化という**データグリッド(data grid)**へと進化した。
- このデータグリッドは、金融分野やライフサイエンス分野など多くの用途での利用への期待が高まっている。
- 近年、**クラウドコンピューティング(cloud computing)**という用語が流行するようになった。
- クラウドとは文字通り“雲”のことである。クラウドコンピューティングとは、ネットワーク上に存在するサーバが提供するサービスをそれらのサーバ群を意識することなしに利用できるというコンピューティング形態を表す用語である。

## 12. 応用技術と将来動向

### 6. データベースの将来動向

- ネットワークを図示するのに雲状の絵を使うことが多いことからきた表現と言われている。雲の中のハードウェアやソフトウェアの存在を気にしないで利用することが出来るというのが、クラウドコンピューティングのメリットである。
- このように、データベースに関連する様々な技術が提案され時代とともに進化している。
- 今後とも、“**ある特定の目的について、適切な判断を下し、行動の意志決定をするために役立つデータである情報**”を、信頼性をもって効率的にアクセスできる仕組みを如何に追及していくかにデータベース技術者の努力が払われることになるだろう。

## 1 2. 応用技術と将来動向

### 講義内容

- XMLデータベース
- OLAP (online analytical processing)
- ストアドプロシージャ
- オブジェクト指向データベース
- レプリケーション
- Google File System
- BigTable
- データベースの将来動向