

## 第 1 講（人類の歴史と情報処理、紙の文化から計算機の文化へ、データベース・ネットワークの意義、データベース序論）

### 身近な情報

新聞、雑誌、本、時刻表、ラジオ、テレビ、電話、...

個人 衣、食、住 + 情報                  人としての幸せに必要な（最低限）なもの  
社会 物質、エネルギー + 情報  
情報が重要な柱になってきている

情報は 時として重大、親しい人の消息、適職探し、名医を求めて 会社（等の組織）  
会計処理、取引処理、在庫処理 企業・組織・国家の命運懸かる

### 情報とは：

事実、データ、概念、メッセージ、これらにかかわる手段も含めた総体

（データ 形式化された様式による事実、概念、指示の表現であって、人または自動的手段による通信、解釈、処理に適したもの。）

情報 データに適用した取り決めを使って、そのデータにつけられた意味 OECD セキュリティガイドライン （出典上園））

情報の種類 文字、音声、画像

情報の様相 記録、伝達、処理

### 情報処理発展の歴史

人が言葉を持ったのが、他の動物と分けた

記録

文字の発明 BC5000 年位 これ以降を有史という。人類の歴史（150 万年？）の大部分は文字による記録がない！

文字以前は語り部などしか記録手段がなかった

（現代でも文字の読み書きできない人はいる：文盲（無学文盲）、illiteracy ← → literacy（読み書き能力））

文字を記す手段として粘土板、石版、石碑など

やがて紙（羊皮紙 2C、パピルス 4C?）が生れ画期的に便利となる

書き写す手段として、筆写（奴隷を多数集めて写本作成など）が主流

(馬に載せて運ぶ本)

グーテンベルクの活字印刷術(1440年頃)これ以降現代文化につながる。宗教改革にも影響があったという

## 伝達

マラソンの語源 BC490年マラトンの丘 アテネ市民軍 vs 来襲したペルシャの大軍をマラトンの戦いで破った ある兵士がアテネまで約40kmを走りに走って<我が軍が勝った>と勝利を伝え、そのまま事切れたという これにちなんで近代オリンピックにマラソン種目が生れた(1986年第1回オリンピック(アテネ大会)より)。現行のマラソンは42.195kmである。記録は約2時間8分(1969年で2時間8分33秒6)。

伝令は軍隊などにも今でも存在する(?)

赤穂浪士など時代劇には早馬が登場する

のろし、太鼓など原始的伝達手段は古来からあった。手旗信号もこれに類するが現代でも使われている。

中世の腕木式信号は小高い山の上に置き情報を伝え合った

飛脚、郵便制度

電信(1833年)、電話(1876年ベル)、無線(1895年マルコーニ)

現代は衛星通信、計算機ネットワーク

## 処理

筆算、アバカス、算盤、パスカルの加算器、タイガー計算器

チャールズバベッジの機械式プログラマブル計算機(アナリティカルエンジン、未完)

は近代計算機の祖とされる

近来に至るまで座席予約などに原始的方法が採用されていた(回転式帳簿棚と人界戦術)手紙の宛名書きもつい先頃までは人界戦術が主流、現在では計算機を使わないのはむしろぜいたくになりつつある

現代計算機の直接の祖はENIAC(1945年。18,000本の真空管使用)、以後現代に至るまで計算機は飛躍的に進歩、単なる計算する手段を越えて情報に関わるあらゆる場面に使われるようになっている

データベース（様々なデータを蓄え必要に応じて取出せるシステム）、計算機ネットワーク（いつでもどこでもだれとでもものやり取り）の重要性

かつては生身の人間が全て関わらざるを得なかった情報の記録・伝達・処理は計算機（を含むシステム）で大部分処理されるようになってきた

これからも更にこの進歩はとどまることがない、大きな社会変革にもつながる可能性がある 産業立地、営業形態（テレショッピング、直接注文）、経済形態（電子取引）、職業形態（在宅勤務）、教育形態、都市の過密解消、社会形態（直接民主制？）

コンピュータリテラシーが重要 古くは読み書き、算盤と言われた 社会生活のための基盤的能力 コンピュータリテラシー、情報能力が個人の基盤を左右するようになりそう 現代を生抜く基盤能力はデータベース（必要な情報の獲得）とネットワーク（いつでもどこでも誰とでも）の有効活用である

古代から現代に至るまでの流れを概観するとこの重要性が分る

## 演習

情報の記録・伝達・処理等について発展の歴史を概観せよ

マラソンの語源となった故事と現代の通信との違いを定量的に考察せよ（例マラソン：1bit/2.0H を ISP（インターネット・サービスプロバイダ）との接続やパソコン通信で行われる電話回線とモデムの組みあわせ約 50kbps とを比較）

## 第1 講後半

（承前）

紙の文化から計算機の文化へ（アナログからデジタルへの流れもある）

紙の消費は文化のバロメータ

現代はいまだに紙の文化が主流

本、帳簿、決済文書、マニュアル、ビジネスレター、契約書、小切手、紙幣

紙と計算機の違い、サンプル化、量子化

電子化のメリット：

コピー、伝送、保存の容易性

検索の容易性

参考 計算機（機械）可読データ（マシンリーダブル）

大きな変革の時代にさしかかっている

当面重要なもの： データベース、計算機ネットワーク

## 第2 講 データベース

どんなことでも覚えておいて決して忘れず、すぐに正しく思い出せる人がいたら素晴らしい。このような夢の実現に関係しているのがデータベースである。個人の身の周りに比べると巨大な組織体（企業、官庁、学校など）を動かすのに必要な情報は多量で複雑

である。技術的にはこのようなものに対応できるのが目標となる。

データベース Database （古くは、または一部にはデータバンク）

（base 土台、基地、塁）

データベースは操作対象となるデータの集まり。

データベース管理システム（DBMS）は様々なデータを蓄え必要に応じて取出せるシステム

例 文献データベース、新聞記事データベース、化学物質データベース、判例データベース、従業員データベース、在庫データベース

データベース広義と狭義とある

広義：データを蓄え必要に応じて取出せるシステム一般、データ量が多いものを指すことが多い

狭義：多量のデータを多目的に用いる、相互に関連するデータを扱える、更新が容易などの性質を合せもったものを呼び、多量のデータを扱うものでも、単純なものはファイルと呼んで区別する（計算機の説明でよく使うファイルはここで呼ぶファイルと概念が異なるので、ここでのファイルを単ファイルと呼ぶことがある）。

マスコミ等で取上げられる場合、広義の意味のことが多い。

データベースは広い範囲で使われる新聞記事データベース等のもの（ただし実際のは単ファイルである場合が多い）と、一つの組織体を運営するのに必要な情報を管理するもの（ビジネス向けデータベース）がある。マスコミ等で取上げられる場合前者が多い（技術的にはデータベースの呼び方はあまり妥当でない場合が多い）が、後者は広範な需要を持ち技術的にも高度なものを要求される。この授業ではビジネス向けデータベースを主に勉学の目標とする。

歴史的にはファイルシステム → データベースと進んだ

ファイルシステムは単一目的、単一構造、特定プログラム

ファイルシステムではデータが変化するとプログラムも変更することがある（データ依存）→データ独立

データベース（仮の定義）

多量のデータを多目的に用いる、更新が容易などの性質を合わせもったもの（多量のデータでも単純なものはファイルシステムと呼んで区別する）

データベースは多量・多種類のデータを統一的（機械的）に処理することを目標とする。よって一定のルールに従ってデータを蓄えるようにする。計算機がいかに高性能でも雑

多なままではお手上げ。

例 紙切れ、ノート、ルーズリーフ、台帳 横書き、縦書き

様式の統一 縦横のマス目に区切って蓄えるのが自然。

第2 講（データベースの基礎概念、ファイルシステムからデータベースシステムへ、データベースに望まれる性格、データベースシステムの構成）  
歴史的にはファイルシステム → データベースと進んだ

（ファイルシステムとデータベースに共通する考え方）

簡単な例

名簿

| 名前   | 郵便番号 | 住所         | 電話番号    |
|------|------|------------|---------|
| 山田太郎 | 100  | 東京都千代田区... | 03- ... |
| 丸山花子 |      |            |         |
| 小山次郎 |      |            |         |

上の例で横の行をレコードと呼ぶことがある。各列を属性、フィールドと呼ぶことがある。（レコード、フィールドの用語は計算機内部の物理的管理単位として使われることが多い。慣用的に論理的説明にも使われることがある）

あらゆるデータを雑多に詰め込むのは難しい。この種の升目に書ける形式のデータが計算機には扱いやすい。各レコードは実社会の実体（例：人間、会社、品物...）を反映している。このような個別の実体の集りを反映したデータをファイル（単ファイル）と呼ぶことがある。

（注意： 黒板や教科書などでは例として簡単なものを上げることが多いが、レコード数が何万とあり、手作業ではとても処理しきれないものを想像して欲しい）

以後この種の形式を基本に考えると考えやすい。実際の人間は色々な属性（性質）をもっているが、この例のように一定範囲の目的に合わせて必要なものにしばってデータを蓄える。

データ構造 データベースがどのような要素から構成されてどのように関係づけられているか定めるもの。

ファイル名、ファイルの中の属性項目など、個々のデータの存在を規定するものをスキーマと呼び、個々の具体的データを実現値（インスタンス、オカレンス）と呼ぶことがある。

(注 スキーマという用語は CODASYL (後述) 由来)

このようなデータの集り、属性を定めることをデータ定義 (Data Definition) という。  
スキーマ設計と呼ぶこともある。データ定義の概要：ファイル名、属性名の定義、属性に許される値の範囲等

参考 レコード型 ある実体を反映した、いくつかの属性項目よりなるファイルを定義するとき、レコード型が定まったという (リレーショナル DB ではレコード型と呼ばない)。

属性 (フィールド) の内、各レコードを他から区別するために使われるものをキーという。レコードが異なるごとに必ず異なることが保証されているときプライマリ (主) キーと呼ぶ。学生在籍簿の学籍番号などがそれに当る。(c.f.複合キー 複数の属性によって各レコードを識別するとき、これを複合キーと呼ぶ)。キーはデータベースの設計者が定める。一般にはレコードを特定できる属性 (または複数の属性の組みあわせ) は一つに限らず、それらを候補キーと呼び、管理者が一つを特定するとき、プライマリキーと呼ぶ。プライマリキーは空値 (値が入っていないこと) にしてはならない。

人名は通常の社会生活で個人を識別するのに使われているが、プライマリキーとして使うことができない。なぜなら同姓同名が一定の率で必ず存在するから (同姓同名となることは何ら禁じられていないし、事実 500 人位人がいれば 10 人位は同姓がいるし、同姓同名も 1 組位ある)。同じ人は世の中に決して 2 人はいない。それをちゃんと区別するには、(使う範囲で) 決して重複しない番号をふる。これをユニーク IDという。

上の例のようなデータで想定される操作に：

レコードの新規入力、削除、修正、検索 (一致、大小など)、ソート

などがある。データ操作 (Data Manipulation) という。データを検索することをデータの問い合わせ (Query) という。

(データベースの有難み： 形式 (レイアウト) の変更が容易、検索、ソート)

デモ： 簡単な例 (受講者名簿)、レイアウト、検索、ソート、リレーション

## 演習

身近なものでデータベース化できるものを列挙し、それぞれのデータ定義を行え。

データ検索、データベースに関わる経験を述べよ

(単) ファイルは形式上相互に独立している。しかし、実質は互いに関連していることがある。

## 個人名簿

名前      所属      所属所在地

## 組織名簿

組織名      所在地      業種      代表者

( 学生データ

学籍番号   名前   学科   学年   指導教官

学科データ

学科番号   本部建物名   主任   連絡先

指導教官

教職員番号   氏名   連絡先

)

実質上データを重複して蓄えている

単純なファイルを複数個別々に蓄える場合：

無駄が多い（記憶容量、作業量において）

互いに矛盾しがち

相互に参照し合えない

これらのデータの重複や関連は、単独ファイルの集まりではすべて利用者が判断しなくてはならない。

（個人の引越の場合、様々なファイルを修正しなくてはならない

製造工場において何かの製品を作る場合、原材料が減り、製品が増える、従業員の作業が発生、光熱費が掛かる）

単純な名簿のようなデータ：（単）ファイルシステム

（狭義の）データベースシステムが対象とするのは様々な目的を果たす多数のファイルよりなるシステム

元来ファイルシステムは個別目的のため、個別プログラムが管理するものであった。同種のデータですら、部署が違えば別個に管理していることがある。例社員名簿データ、人事係、厚生係、給与係（別例：購買係、資材係、サービス係）で別々など。手書の帳簿で管理する時代には、もし異動事項（住所変更など）があるとき、全係に届けを出さなければならなかった。一カ所でまとめて処理できる場合でも、他の部署への通知は書類でおこなうので、最悪一ヶ月位掛り、矛盾することもよくあった。

これが計算機化されても、ファイルシステムである限り同じ事が起る。基本的にファイルシステムはデータ互換がなく、それぞれ個別のプログラムで読まなければならない。（同じ実体を対象としても、データ定義において個別目的に合わせて属性を決定している。）あるプログラムで読み出したデータを見ながら手作業で次のシステムに入力するなどのことが案外今でも行われたりしている。

同じデータを一箇所に維持し、他からはそれを参照するのがよい

（歴史的には（単）ファイルシステム → データベースと進んだ  
（単）ファイルシステムは単一目的、単一構造、特定プログラム

ファイルシステムではデータが変化するとプログラムも変更することがある（データ依存）→データ独立）

## 演習

2つ以上種類の異なるデータが互いに関わり合っている例を上げ、検索、データの更新などに例を取って問題点を述べよ

（狭義の）データベース

（単）ファイルシステムの発達したものと見ることができる。単独ファイルの集まりでは管理しきれない状況に対しシステムとして対処する。データの収集や整理は機械的に効率よく行えるようにしたい。

データベースに望まれる性格

### 1.データの共同利用

多数の利用者やアプリケーションからアクセスできるようにして、一つの組織体（利用者グループ）内部で役立てるようにする。

### 2.データの一元管理

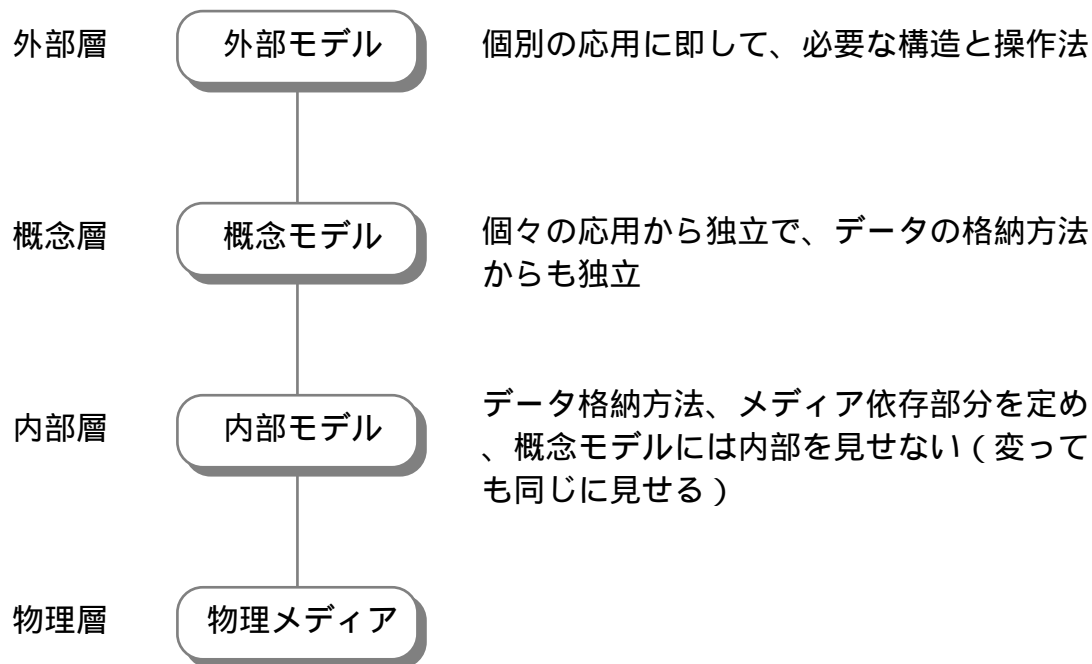
データの重複や矛盾がないように管理する。

### 3.データの独立性を確保

データは実体を反映したものであり、どのような媒体に蓄えられているか（物理的）、どのような構造で扱っているか（論理的）とは独立なものであるべきである。物理的・論理的蓄え方の変更があってもアプリケーションを変更したり利用者が個別的にデータを修正する必要がないようにする。

## データベースシステムの構成

望ましいデータベースシステムを実現するには階層構造とするのがよい（ANSI-SPARC 報告書に由来。関係型 DB では必ずしもこの構成を強く問題にしない）。



データは通常外部記憶（磁気ディスク等）に蓄える。

データベースに関わる人：

一般利用者（エンドユーザ）

応用プログラマ

データベース管理者

## システムの見え方

利用者インターフェース： 対話型コマンド、利用者言語

応用プログラマ向け： 開発言語、データ操作言語、コマンド

データベース管理者向け： データベース管理言語、コマンド 利用者登録、データ登録

データベースにはデータ辞書を備え、データ、利用者、プログラム等を管理する。

## データベースシステムの管理機能

データ一貫性チェック

機密保護

同時実行制御

障害回復