

## PCイメージファイリング

1G-6

中里 竜 上田 昭彦 竹内 正行  
高山 佳文 佐藤 実 曾田 光雄

(株)東芝 青梅工場 ソフトウェア設計第二部

## 1. はじめに

近年のペーパーレス化に対応するため、大量の書類をイメージデータとして光ディスクに登録し、検索できる電子ファイリングシステムが開発された。これにより一部ではペーパーレス化は進んだが、電子ファイリング装置は専用機であるため、一般的には共有書類を入力することが多く、個人の書類と共有の書類の使い分けや共有の書類を各個人の観点で分類整理することができない。また、データ構造が変動している場合、一度決めた構造を変更することは困難である。本稿ではこの2つの欠点を克服し、データ構造の自由度を高めたファイリングシステムを提案する。

## 2. 従来の電子ファイリングシステム

主に省スペース化を目的とする従来の電子ファイリングシステムは、1枚の画像を「頁」とし、複数の頁をひとまとめにした「書類」を分類上の単位として扱っている。書類は、例えば図1のようにディレクトリのような階層構造によって分類されるとともにキーワードが付けられる。検索する際には、階層構造を上からたどって行くか、キーワードでの検索を行うことで目的の書類を探し出す。

このような書類の分類方法は、すでに分類体型が定まっている書類に対しては効率的にデー

タを格納することができる。しかし、1つの書類は1通りにしか分類されないため、違う観点からの複数の分類方法、例えば共有化してある文書に対してそれを使用する人が別の分類をするといったことは難しい。

また、登録を始めた後にキーワードの種類を増減・変更したり、分類体型をそのものを変更する頻度が高い場合、ファイルの大幅な書換が生ずるので、このようなデータを扱うのには向いていない。

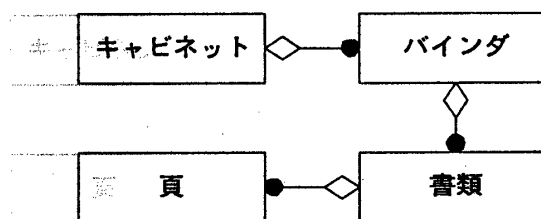


図1：従来の分類の構造例  
(オブジェクト図)

## 3. インデックス方式電子ファイリングシステム

2節での問題点を解決するため、インデックス方式と名付けた分類体型を提案する。

インデックス方式は図2に示すように、ボックス、グループ、インデックスと呼ぶ3つの階層で書類の分類を行う。

ボックスは従来通りの階層構造であり、まったく違った種類の書類を入力する場合にはこの階層で分類を行う。

グループ、インデックスが本方式での特徴となる部分であり、ボックス-書類間の関連とは独立に分類体型を構築できる。このことにより、

書類の登録を始めた後でも自由にグループやインデックスの追加、削除、変更ができ、複数の観点からの書類の分類を簡単に行うことができる。

書類から見た場合、インデックスに対して複数の関連を持つことができるので、付箋紙を付けるような感覚で分類を行うことができる。書類には約65000個までいくつでもインデックスを付ける可能である。また、インデックスを付けないまま未整理の状態ボックスに入れておき、後で分類することもできる。さらに、識別番号、日付など書類ごとに別々のキーワードが入れる必要がある場合には、キーワードとして入力することもできる。

検索をする際には、トップダウンにグループ、インデックスを選択することで書類を絞り込んでいく。また、絞り込み途中の段階からキーワード検索を組み合わせることもできる。

このように、従来の利点を継承しつつ、固定的な分類とあらかじめ型や個数の決められたキーワードの組み合わせより自由な分類・検索を簡単に行うことができる。

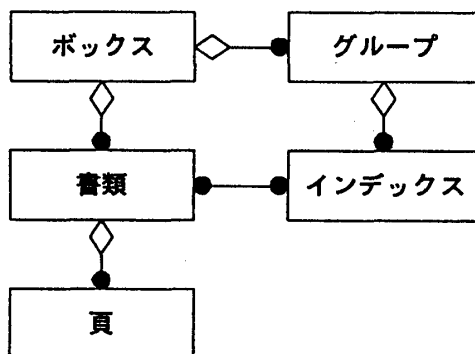


図2：インデックス方式の構造  
(オブジェクト図)

#### 4. データベースの構造と拡張性

上記のようなデータの構造をインプリメントし、さらに他のデータ構造にも対応できるよう

にするため、本システムではデータベース部分にオブジェクト指向を採用した。データベース部分の構造は図3に示すように3層に分かれており、上から順に三層スキーマアーキテクチャの外部スキーマ、概念スキーマ、内部スキーマに対応している。

ボックス、グループ、インデックス、書類については図4のように同一のクラスとし、属性の違いでオブジェクトの種類の違いを識別する。これにより、上位層のみを変更すれば本方式の利点を保ったまま、従来のデータ構造や他のデータ構造についても扱うことができる。

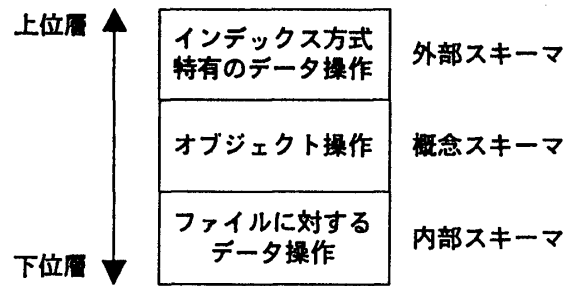


図3：データベースの構造

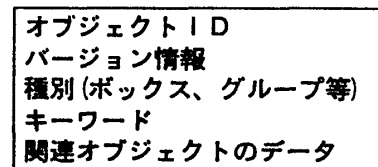


図4：データ構造

#### 5. おわりに

本稿では、オブジェクト指向データベースを使用する事により、従来の欠点を克服し構造を自由に変更できる電子ファイリングの分類体型を提案した。

また、多種多様なデータを扱え、再利用のしやすいクラスの構造を構築した。

今後は、異なったデータ構造間での効率的なデータのやりとりを性能の面から検討していく予定である。