

オフィス・ワークに用いるデータの統一の扱いについて

7Y-6

渡辺 豊英、桶谷 猪久夫
(京都大学 大型計算機センター)

はじめに 計算機の発展は様々な用途に適用効果を示し、情報化社会の必要不可欠な道具・手段として計算機を位置付けてきた。計算機は種々のデータを処理・管理・運用するのに適した機能・性能を提供している。しかし、目的用途別に開発されたソフトウェア・ツールは必ずしも相互に連携できる機構を提供していない。統合ソフトウェアの課題は、このような状況に対して提起された。各種ソフトウェア・ツールが有機的に連携できるためには、ツールのモジュール構造と、ツール間のデータ構造が重要である。

本稿では、オフィス情報システムにおける処理機能の連携の立場から、各処理機能のデータ構造について議論する。我々の方法は各処理機能で扱われるデータ構造を論理構造と物理構造に分離し、各データ構造から抽出された論理構造を階層的に関係付けることにより、処理機能の連携を図っている。

論理構造と物理構造 各処理機能が扱うデータには、少なからず処理機能に依存した管理構造が付帯し、それは一般に他の処理機能の管理構造と同じではない。処理機能間でデータを共用する場合、データ変換が必要になる。各処理機能が共通のデータベースを介してデータを共用する方法は、統合ソフトウェア対応の実際的な一つの解である。しかし、このアプローチはデータ構造が類似している場合、付帯的に処理機能を拡張する場合に有効であるが、全く対等で独立な処理機能に対して連携化の解を与えない。即ち、各処理機能に共通したデータ構造の捉え方が必要である。

各処理機能が扱うデータ構造を分析すると、データの並び・属性等が類似する場合が多い。このような類似する特性を機能に固有な構造として構成せずに、例えばデータベースにおけるデータ定義言語のように指示できれば、ユーザ操作の論理構造とシステム操作の物理構造を独立に扱える。抽出された論理構造の共通化を図れば、ユーザ・ビューとして統一的なデータの扱いが可能になる。

例えば、データベース管理、文章編集処理、文書作成処理を考える。データベース管理と文章編集処理では各処理機能、その用途に若干の相違があるものの、データ操作から捉えると同じ概念的な操作が多い。一方、文章編集処理と文書作成処理では、editorとformatterの枠組の下に連携の実例がある。この枠組ではformatterがeditorに従属しているが、我々の枠組で各々が独立に構成されなければならない。論理構造と物理構造を独立させた枠組を典型的な処理機能に適用すると、図1に示すようになる。文書作成処理では、従来のeditorとformatterのようにテキストと文書の従属関係では

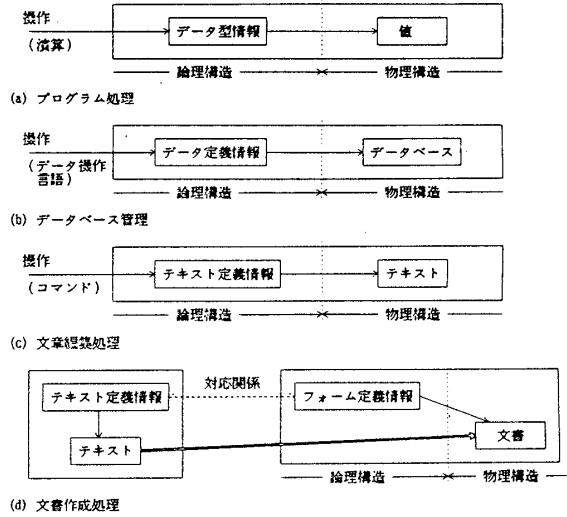


図1 典型的な処理における論理構造と物理構造

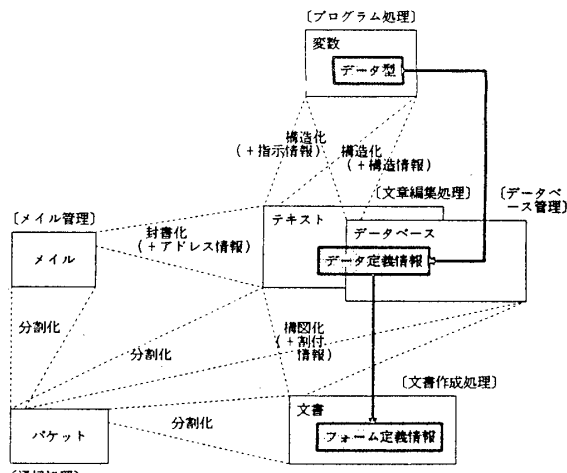


図2 各処理機能で扱われるデータの論理構造に関する階層性

なく、テキスト定義情報とフォーム定義情報が対応関係にある。

論理構造の階層性 各処理機能で扱われるデータの論理構造には、少なからず共通した枠組がある。データベースのカラムのデータ属性はプログラム言語のデータ型とほぼ同じであり、構造化された一つの定義体がデータベースの論理構造を定める。また、文章編集処理のテキストは文書作成処理の原データとして、文書内の位置情報と共に論理的に対応関係を保って文書を構成する。

オフィス情報システムのデータベース管理、文章編集処理、文書作成処理、メール管理、プログラム処理、通信処理等の各処理機

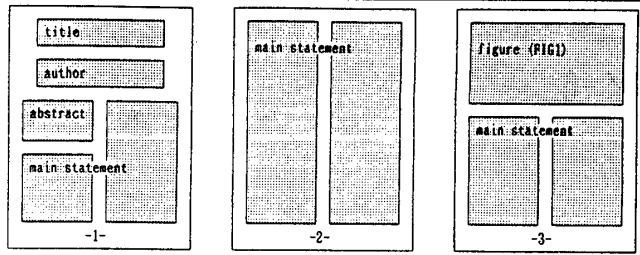
KEY	12345678	
TITLE	A PORTABLE POP-11 SIMU..	
AUTHOR	AUTHOR (1)	NAME DOYLE, J. K.
		ADDRESS ATT BELL LABORATORIES, ..
	AUTHOR (2)	NAME MANDELBERG, K. I.
	
TAKEN	SOFTWARE-PRACTICE AND ..	
PUBDATE	NOVEMBER, 1984	
PUBCOMP	JOHN WILEY & SONS, LTD.	
ABSTRACT	THIS PAPER DESCRIBES...	

(a) テーブルの論理構造

```
table B1B;
term KEY: integer 'KEY NO';
term TITLE: char(100) 'TITLE';
term AUTHOR(5): record 'AUTHOR';
term NAME: char(30) 'NAME';
term ADDRESS: char(80) 'ADDRESS';
end;
term TAKEN: char(200) 'TAKEN';
term PUBDATE: char(20) 'PUB-DATE';
term PUBCOMP: char(20) 'PUBLISHER';
term ABSTRACT: char(700) 'ABSTRACT';
end;
```

(b) テーブルの定義

図3 データベースにおける論理構造とその定義



(a) 文書構造

```
form PAPER;
region 60,80;
layout 6,50,5:1,39,2,39;
page 58,M: '-';
space 3;
footnote 'F:FOOT';
skip 5:M;
element (10,1), 'TI':OFF;
skip 2:N;
element (1,1), 'AU':OFF;
skip 4:N;
element (1,1), 'AB':ON;
skip 2:Y;
element (1,1), 'MA':ON;
block (20,70):FIG1;
block (40,39):TAB1;
block (40,39):FIG2;
.....
end;
```

(b) フォーム定義言語による記述

図5 文書における論理構造とその定義

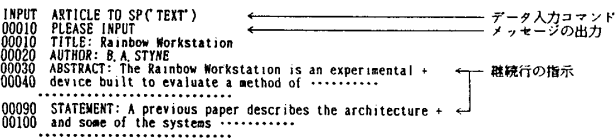
```
table ARTICLE;
input;
trap 'F: ';
cont 'F: ';
....
end;
term 'PLEASE INPUT';
term TI: char(200) 'TITLE';
term AU: char(200) 'AUTHOR';
term AB: char(1000) 'ABSTRACT';
term MA: char(20000) 'STATEMENT';
....
end;

table TABLE;
term TAB1(60): char(80);
end;

table FIGURE;
term FIG1(100): bit(400);
end;
```

(a) テーブルの定義 (本文関係)

(b) テーブルの定義 (図表関係)



(c) データ入力におけるデータ定義情報の利用

図4 テキストにおける論理構造とその利用

能で扱われるデータ構造を論理的な関係の下に捉えることができる。図2は各データ構造の論理的な階層性を表している。文書の論理構造(フォーム定義情報)はテキストの論理構造(データ定義情報)に割付情報等が付加され、テキスト(またはデータベース)の論理構造は変数のデータ型に構造情報等が付加されている。メールはテキストにアドレス情報等の付加の下に構成され、パケットはデータを物理的な分割処理により作成される。メールとパケットは処理操作により物理的に扱われるデータ形態である。

論理構造の記述 各データ構造に対する論理構造の記述について次に例を示す。文章編集処理とデータベース管理にはデータ定義言語を、また文書作成処理にはフォーム定義言語を用いる。

- (1) データベース: 記述子termの下にデータ属性等を指示し、termの並びで一つの定義体tableを構成する。例えば、図3(a)の論理構造は図3(b)の記述によって定義される。カラムのデータ属性はデータ型と同じであり、配列型、レコード型も含んでいる。
- (2) テキスト: データベースと同じデータ定義言語によって論理構造を定義する。例えば、論文テキストを扱う場合を考える。論文は標題、著者、内容要約、図表、脚注等から構成され、論理構造は図4(a)のように記述できる。図4(a)には図・表が定義されていないが、図4(b)の別テーブルで定義する。データベースが複数のテーブルから構成されるように、テキストは複数のサブテキス

トから構成でき、かつサブテキスト内の要素もtermにより全く独立に扱える。従って、操作対象が明確になり、扱いが容易になる。

図4(c)は図4(a)のテーブル定義をデータ入力時に指定し、データ・エントリ機能として動作させた例である。

- (3) 文書: 文書の論理構造は帳表等の紙面に対する設計書であり、テキストを原データとして文書を構成する。例えば、図4のテーブル定義で作成したテキストを、図5(a)の構造に構成するフォーム定義言語の記述を図5(b)に示す。基本的に、データ定義言語のtermとフォーム定義言語のelement、サブテキストとblock(spaceと追携)の対応の下に、文書は整形される。他に、物理的な出力域を定めるregion、論理的な出力域を定めるlayout、term間の空白行を制御するskip、ページ番号出力を指示するpage、脚注を制御するfootnote等が用いられている。

おわりに 本稿では、オフィス情報システムの各処理機能のデータ構造を、論理構造と物理構造に切り分けて、ユーザ・ビューとして論理構造を与え、統一的なデータ操作を可能にする方法を報告した。本方法はデータ構造について、ユーザに統一的なビューを与えるが、当然それを処理する機能においても、有機的なインタフェースを実現し、遠携性を保証する。しかし、本稿ではこの議論は紙面の都合で割愛した。

〔謝辞〕 日頃から研究・開発にご理解とご鞭撻いただいている長尾真センター長、星野聡研究開発部長を初め、研究開発部員の方々に感謝します。

〔参考文献〕 1) 渡辺: 学術情報ネットワーク形成のための計算機機能について、1985情報学シンポジウム講演要旨集、pp.3-4. 2) 渡辺他: パソコンの多機能を連携させたワークステーションの機能概要、情報処理学会第30回(昭和60年前期)全国大会講演論文集、3F-2. 3) 桶谷他: パーソナル・コンピュータ上のデータベース処理について、情報処理学会第30回(昭和60年前期)全国大会講演論文集、1F-2. 4) 桶谷他: 多機能を連携させたパソコン上のデータベース処理、情報処理学会データベース・システム研究会資料、47-4(1985). 5) 渡辺他: データベース処理と文章編集処理の連携について、情報処理学会第31回(昭和60年後期)全国大会講演論文集、7G-7. 6) 桶谷他: 文章編集処理と文書作成処理の連携について、情報処理学会第31回(昭和60年後期)全国大会講演論文集、7G-8.