

PALET-0のオフィスデータベース機能

鶴岡 邦敏 渡部 和雄 西原 義之
(日本電気(株)C&Cシステム研究所)

1. はじめに

PALET-0は、オフィスで使用する帳票や管理表など、表または表の集合として定義されるドキュメントを設計、処理、管理するソフトウェアである。従来、表処理用のソフトウェアとしては、関係データベースの問合せシステム、パソコン等の表計算ソフトなどが開発されている。しかしながら、これらのソフトウェアには以下のような問題がある。

- ① ファイル全体を一枚の表として直接見せており、利用者から見た処理単位である「ドキュメント」(一枚の帳票や管理表)を意識していない。
- ② 「帳票」のような、同一の形式を持つドキュメントの集合を扱えない。
- ③ 画面形式が一枚の表に固定されており、自由な形式を持つドキュメントが設計できない。
- ④ 画面上で矩形のフィールドを編集できないため、文章を内容とする表の処理には適さない。
- ⑤ 項目の長さの変更や項目の追加が難しい。

このため、一般のオフィス帳票の処理には、煩雑なデータベースの定義の上に、更に業務プログラムの作成が必要になる。しかしながら、部門単位で設計/管理するような少量多品種の帳票や管理表の場合には、とてもそれほどの工数はかけ得ない。

PALET-0は、こうした統合化されていない帳票や管理表の設計・処理・管理を目的としており、以下の特長を持つ。

- ① 自由な形式の「テーブル形ドキュメント」を設計できる。
- ② データベースの定義が不要である。画面上のドキュメント定義が、そのままデータ定義になる。
- ③ データが格納済みでもドキュメントの形式を自由に変更でき、かつデータベースの再構成が不要である。

本資料では、テーブル形ドキュメント管理システムPALET-0が持つオフィスデータベースとしての機能について述べる。

2. データ定義機能

PALET-0は、利用者に、「データ」ではなく「ドキュメント」を単位としたインタフェースを与える。また、利用者はドキュメントの外部形式(画面上の形式)のみを意識すればよく、論理形式(主記憶上の形式)や物理形式(二次記憶上の形式)はほとんど意識せずに処理を実行できる。

(1) 外部形式の定義

多くのデータベース(DB)システムでは、データ定義言語等でまずデータベースを定義し、次にレポート生成ツールや業務プログラムで画面上/プリンタ上にドキュメントを表示/印刷する。これは、データの集中化を重視した統合データベースのアプローチを取るためである。しかし、オフィス/個人DBの場合には、こうした煩雑な作業は好まれず、データの統合化よりも使いやすさが重要となる。

PALET-0では、利用者が画面上にドキュメントの外部形式を設計すれば、システム側がその構造を認識し、データの論理形式や物理形式を自動生成する(統合DBとは逆のアプローチ)。この機能により、一般利用者自身によるデータベース設計が可能となる。

初めに、図1. にドキュメントの外部形式の例を示す。このような画面上のドキュメントを「外部ドキュメント」とよぶ。ドキュメントには、データベースと同様に、タイ

社製品発注依頼書

発行日	83年7月8日		
要求元	C&C研CS研		
発注先	光通信応用装置		
事業部	事業部		

発注番号	6-5889		
オーダー	8G7140		

品名	付替価格	工事費	合計
光LAN-7777	540,000	20,000	560,000
光コネクタ	270,000	1,000	271,000
イメージディスク	875,000	19,000	894,000
分岐増設機種	198,000	1,500	199,500

担当者	鶴岡 邦敏		
届け先	A-306		

備考	発注品は、伝票到着後1ヶ月以内に納入されるように工場手配して下さい。		
----	------------------------------------	--	--

図1. 外部ドキュメント

プとオカーレンスとの区別が存在する（関係データベースではリレーションとタプルに相当する）。PALETTE-0では、ドキュメントタイプの外部形式を与えるものを「様式」とよぶ。図2. に様式と内容（オカーレンス群）の例を示す。

様式は複数個の「枠組み」から構成される。枠組みは「単項目」または「集団項目」またはそれらの結合したも
のから構成される。集団項目とは単項目の集合である。単項目または集団項目は、それぞれ繰返し項目であってもよい。図3. に枠組みと（集団）項目の例を示す。この図では、枠組み1は1個の単項目（繰返し数1）から構成され、枠組み2は1個の集団項目（2個の単項目から成り、繰返し数3）から構成され、枠組み3は1個の集団項目（3個の単項目から成り、繰返し数4）と2個の単項目（「合計残業時間」と「合計休出時間」で、各々繰返し数1）とから構成される。図のように、1個のドキュメントオカーレンス内に複数の表を定義することができ、従って1個のドキュメントタイプは3次元の表の集合と見ることができ
る。

PALETTE-0での外部ドキュメントの定義は、画面上に枠組みを配置し、それを適当に分割し、行/列を拡大または縮小し、項目名を与える、という手順で実行される。図4. に様式設計の例を示す。（a）は枠組みを生成したところ、（b）は枠組みを分割し行/列の拡大/縮小を

行なったところ、（c）は項目名を入力しているところを示す。この様式設計により、項目名・項目値の表示位置と表示領域（矩形領域が可能）の大きさ、及び繰返し項目の繰返し方向などの外部形式情報が定義され、更に後述する論理形式情報（データ構造）が暗黙のうちに定義される。項目名と項目値との関係は、縦でも横でもよい（縦型の表と横型の表とが可能）ので、種々の枠組みを組合せることにより、かなり複雑な様式を持つ帳票を設計することが可能である。

（2）論理形式の定義

PALETTE-0における様式定義は、単にドキュメントの画面上の形式を与えるだけではなく、暗黙のうちにドキュメントの論理形式（プログラムから扱える主記憶上の形式）をも定義している（ここで、論理形式を持つ主記憶上のドキュメントを「論理ドキュメント」と呼ぶ）。すなわち、画面上に様式を設計したことがデータ構造を定義したことになり、従来のシステムで必要だったデータベース定義を省略することが可能である。

図5. に、外部形式とそれに対応する論理形式の例を示す。図の上のような様式を与えると、システムはその構造を自動的に認識し、図の下のようなデータ構造が定義されたものとみなす。図のように、論理ドキュメントは、COBOLやPL/Iで言う「構造体」の形式をとる。即ち画面上での様式設計は、項目間の構造、項目のデータ長、繰返し項目の繰返し数、及び項目のデータタイプ（デ

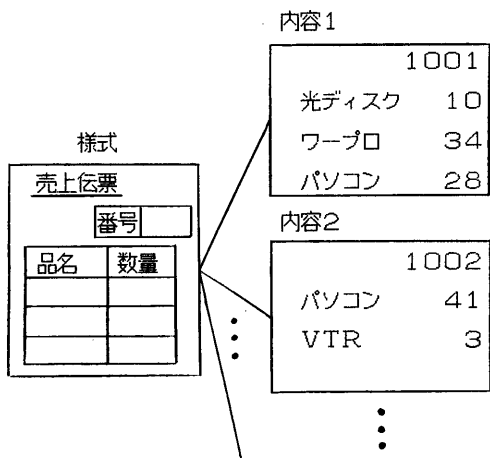


図2. 様式と内容

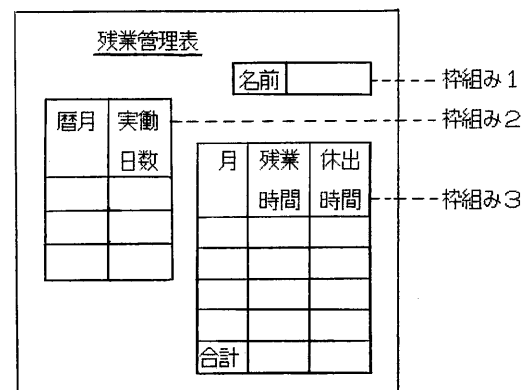


図3. 枠組みと（集団）項目

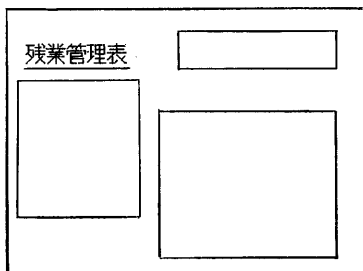
フォルト)等の論理形式情報をも定義している(更に、後述する仮想項目の定義も与えている)。

論理ドキュメントの構造体は、その構成要素として繰返し項目を含むことが可能である。従って、論理ドキュメントタイプは非正規形のリレーションとみなせる。図6に、図5.の論理ドキュメントタイプを展開した例を示す(横線の間が1オカレンス(タプル)に相当する)。

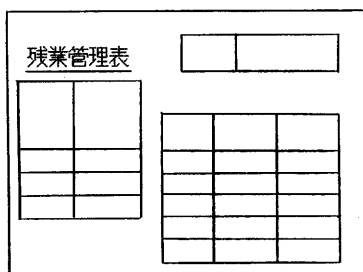
上記のように、PALETTE-Oでは論理形式の定義はほとんど不要であるが、利用者が望むいくつかの論理情報を与えることは可能である。その一つに、「仮想項目」の定義がある。仮想項目とは、他の項目から(一つのオカレンス内で)生成される項目を意味する。例えば、

3つの項目A, B, Cがあり、 $A=B+C$ が成り立つとき、Aを仮想項目と呼ぶ。PALETTE-Oでは、予約語「合計」、「平均」、「最大」、「最小」、「日付」、「時刻」、「利用者」に関しては、システムが自動的に仮想項目の定義(生成規則)を与える。ここで、初めの4つの予約語については、表上の位置によって2通りの生成規則が定義される。縦型の表を例にとると、表頭に予約語がある時は、集団項目の合計/平均/最大/最小となる(繰返し数 $n \rightarrow$ 繰返し数 n 、という生成; 図1参照)。また、表側に予約語がある時は、単項目の合計/平均/最大/最小となる(繰返し数 $n \rightarrow$ 繰返し数1、という生成; 図3参照)。

(a) 枠組みの生成



(b) 行・列の処理



(c) 項目名の入力

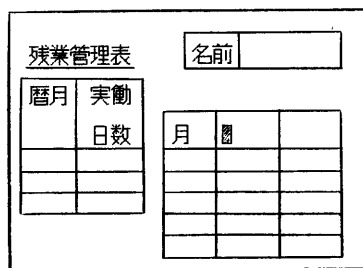
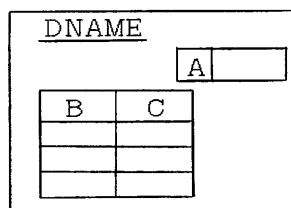
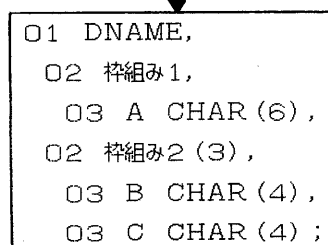


図4. 様式設計の手順



外部形式



論理形式

図5. 外部形式と論理形式

A	B	C
a1	b11	c11
	b12	c12
	b13	c13
a2	b21	c21
	b22	c22
	b23	c23
a3	b31	c31
	b32	c32
	b33	c33

図6. 論理ドキュメントタイプ

(3) 物理形式の定義

PALETTEの利用者は、ドキュメントタイプごとにファイルを作成する必要はなく、ユーティリティによりデータベースの生成のみを実行すればよい。また、物理的なレコード定義も不要で、様式の定義時に自動的に用意される。利用者が意識する必要のあるのは、インデックスの指定のみ（但しこれは性能に影響するだけで、通常は指定不要）である。利用者は、任意の項目に任意の時点でインデックスを作成することができる。

二次記憶上に格納されるドキュメントを「物理ドキュメント」と呼ぶ。物理ドキュメント定義は、論理ドキュメント定義から自動生成される。物理ドキュメントは、可変長・可変繰返し数の単項目の可変個の集合である。システムは、構造体として定義された論理ドキュメントを、フラットな物理ドキュメントに変換する。これを図7に示す。

(4) 形式の修正

PALETTEの大きな特長の一つは、データが格納済みでも、様式の修正が動的に行える点にある。ここで、様式の修正はドキュメントの外部形式（画面上の見え方）に関するものだけでなく、論理形式に関するものも可能で

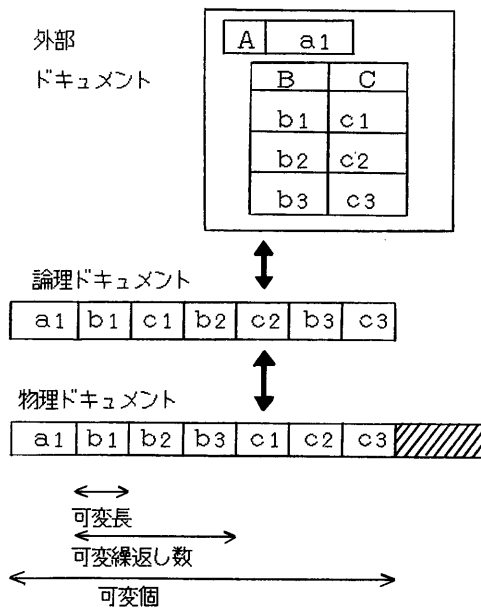


図7. 論理ドキュメントと物理ドキュメント

ある。利用者は画面上で、枠組みの生成/消去、行/列の拡大/縮小/追加/消去/置換、矩形の分割/併合などを実行し、外部形式の変更のみを意識していればよい。これに対して、システムは必要なら論理形式までもを変更する。即ち、ドキュメントの外部形式と論理形式とは、同時に修正される。

図8. に、外部ドキュメントの修正と、それに対応する論理ドキュメントの修正とを示す。図の例では、外部形式の変更は、行の拡大（Aの行）、列の拡大（Bの列）、列の追加（Dの列）、行の追加（枠組み2の行）、及び枠組みの生成（Eの枠組み）が行われている。これに対して論理形式の変更は、項目長の拡大（A, B）、項目の追加（D, E）、及び繰返し数の増加（枠組み2）が実行される。

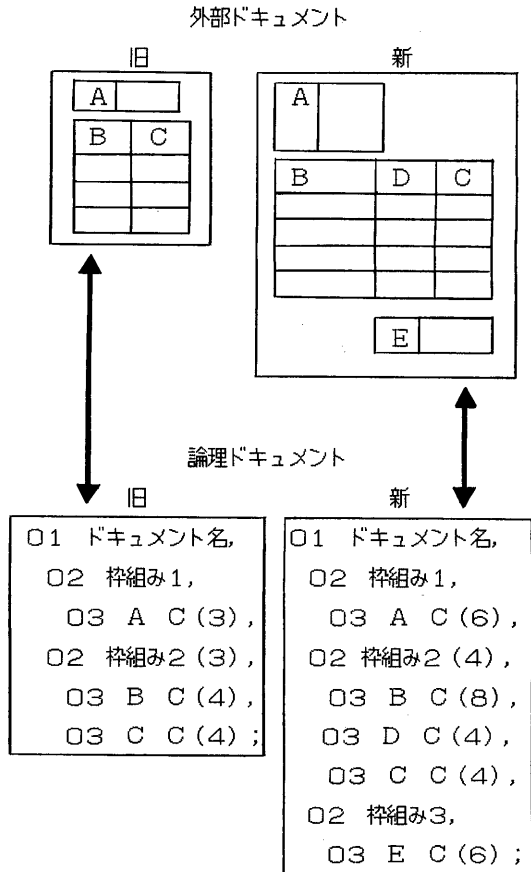


図8. 様式の修正と論理形式の変更

通常のデータベースでは、これはスキーマの変更にあたるため、データが格納済みの時は実行できない。なぜなら、ファイル中に固定形式で格納されている旧レコード群を新しい形式に変換（データベースの再構成）しなければならないからである。一般にデータベースで言うデータ独立の概念は、データの物理形式と一部の論理形式との変更に対して、業務プログラムが独立であることを意味し、データベース自身の独立性は保証されない。

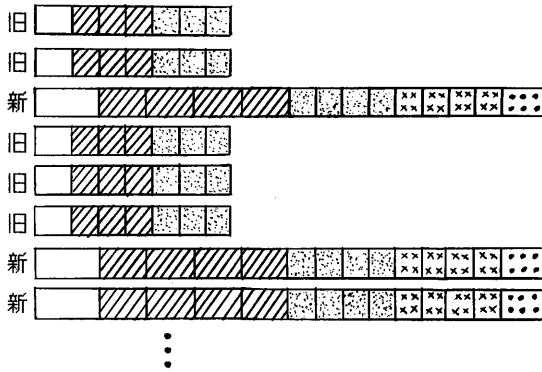


図9. 新旧物理ドキュメントの混在

PALETTEでは、外部形式と論理形式との変更に対して、データベースが独立であることを保証する。この機能は、外部 \leftrightarrow 論理 \leftrightarrow 物理という各ドキュメント間の交換機構により実現される。実際にデータベースの独立性を支えるのは、可変構造の物理ドキュメントであり、これを図9. に示す。図8. のような論理形式の変更に対して、二次記憶中では、図9. のように新/旧の物理ドキュメント群が混在する。これは、物理ドキュメントが可変長・可変繰返し・可変個の項目を持ち得るためである。

次に、図10. に、様式変更時の外部・論理・物理という各階層間での交換の様子を示す。図10の下（または図9）のように、物理ドキュメントは新・旧が混在する。旧物理ドキュメントを新様式で参照すると、物理 \rightarrow 論理変換により、主記憶上で新しい論理構造に変更される（図の左下から右上への矢印）。更にこれを（更新して）保管すると、旧物理ドキュメントは（論理 \rightarrow 物理変換、及び旧物理 \rightarrow 新物理変換により）新形式に変更される（図の右下から右下の矢印）。

以上のような交換機構により、格納済みの古いドキュメント群を新しい様式で参照/更新することができるため、様式を更新してもデータベースの再構成は不要になる。

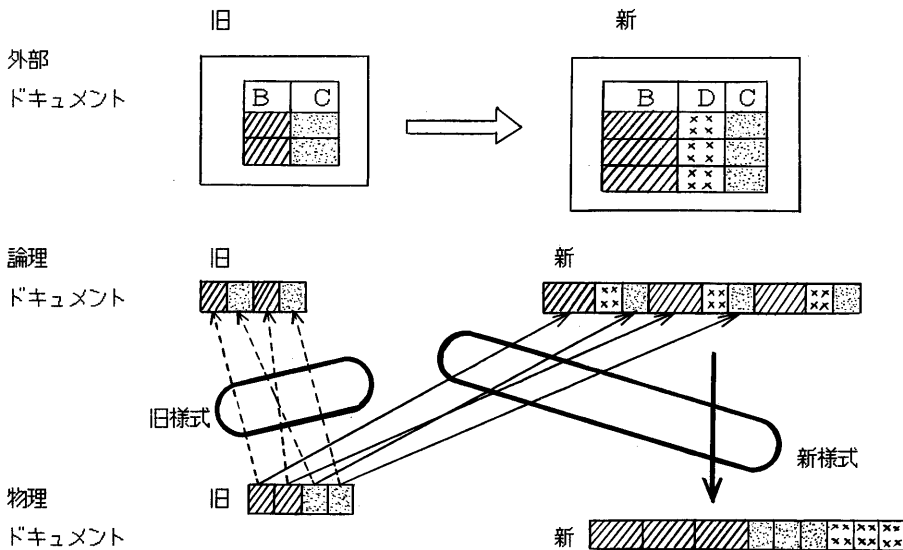


図10. 様式の修正と物理-論理ドキュメント変換

3. データ処理機能

PALETTEは、ドキュメントの様式に関して2. 節で述べた機能を持つ一方、ドキュメントの内容についても検索、修正（種々の表処理機能を持つ）、保管、印刷等、多くの機能を備えている。ここでは、それらのうちで、データベース的かつ特徴のある機能について述べる。

(1) 条件検索

PALETTEの検索機能は、一般のデータベースと同じく、あるドキュメントタイプ中でオカレンスを探す、という機能を基本とする。ここで、特異な機能として、「ドキュメント単位の検索」及び「行単位の検索」という二つのレベルの条件検索を可能としている。

① ドキュメント単位の検索

これは、特定のドキュメントタイプ中で、与えられた条件を満たすオカレンス群を求め、それらを一枚一枚表示

する機能である。検索の例を図11. に示す。図のように、利用者は、画面に表示された様式の中に検索条件を記入する。結果として得られるのは、図の下のようなドキュメントの束である。

ドキュメントには繰返し項目が許され、かつオカレンス内に複数の枠組み（表）が含まれるので、検索の意味は正規形のリレーションに対するそれと比較して、やや複雑になる。ある枠組みの繰返し項目（群）に対する条件の設定は、その条件を満たす「行」を一つでも含むようなオカレンスを探すことを意味する。それぞれの条件は、ANDで結ばれる。図11 (a) の例では、【品名=光ディスク & 数量>5】の行を少なくとも一つ含み、かつ【品名=ワープロ & 数量<10】の行を少なくとも一つ含むようなドキュメントが見つかる。一方、図11 (b) の例では、【品名=光ディスク】の行を少なくとも

(a)
条件

売上伝票		品名	数量
番号		光ディスク	>5
支店名		ワープロ	<10

結果

売上伝票		品名	数量
番号	23	プリンタ	2
支店名	東京第5	光ディスク	15
		ワープロ	8

(b)
条件

売上伝票		品名	数量
番号		光ディスク	
支店名			>5

結果

売上伝票		品名	数量
番号	16	パソコン	12
支店名	横浜第1	ワープロ	4
		光ディスク	2

図11. ドキュメント単位の検索

(a)
条件

売上伝票		品名	数量
番号			>10
支店名			

結果

支店名	品名	数量
東京第1	プリンタ	17
横浜第1	パソコン	12
千葉	ワープロ	11
東京第5	光ディスク	15
水戸	プリンタ	20

(b)
仮想的な検索対象

番号	支店名	品名	数量
01	東京第1	光ディスク	2
01	東京第1	プリンタ	17
01	東京第1	ワープロ	4
02	横浜第1	パソコン	12
02	横浜第1	FDD	5
02	横浜第1	コピー	8
03	千葉	パソコン	4
03	千葉	ワープロ	11

図12. 行単位の検索

一つ組み、かつ【数量>5】の行を少なくとも一つ含むようなドキュメントが見つかる。このように、同じ枠組み内の繰返し集団項目に対しては、条件を入れる位置によって検索の意味が異なってくる。

ドキュメント単位の条件検索機能により、例に示したように、繰返し（集団）項目を含むような一般の帳票の検索が可能になる。

② 行単位の検索

これはドキュメントオカーレンスを構成する各「行」に着目し、あるドキュメントタイプ全体にわたって、条件を満たす行の集合を求める機能である。ドキュメントタイプが繰返し項目を含まない（第1正規形）ときには、関係データベースで言う「制限」及び「射影」による問合せと同様の機能となる（この場合にはドキュメント単位の検索とも同じ結果となり、答えを一覧表で見せる点のみが異なる）。

繰返し項目を含むドキュメントタイプの場合には、各オカーレンスをフラットな表に展開し、これらをつないで検索対象とみなす。例を図12(a)に示す。図の例では、利用者が支店名・品名・数量を出力項目と指定している（ハッチング部分）ので、システムは（支店名、品名、数量）から成る行を考え、【数量>10】を満たす行集合を求める。この検索の意味を説明したのが、図12。

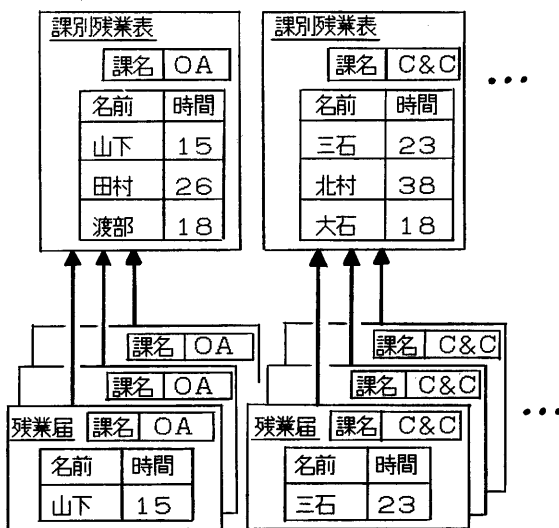


図13. 二次ドキュメントの例

(b)である。図のように、ドキュメントを構成する項目のうち繰返し項目でない項目を他の繰返し項目の数だけ展開して、各オカーレンスをフラットな表とし、それらをドキュメントタイプ全体にわたってつなげて1枚の仮想的な表を作る。次に、この表の各行に対して指定された条件を満たすか否かを調べ、条件を満足する行から利用者が指定した出力項目群を抜き出して出力する。

(2) 二次ドキュメント

PALETTE-0は、あるドキュメントタイプのおカーレンスをいくつかまとめて、別のドキュメントを作成する機能を持つ。ここで、生成元となるドキュメントを「一次ドキュメント」、新しく作られるドキュメントを「二次ドキュメント」と呼ぶ。例えば、「売上伝票」を一次ドキュメントとする時、「売上集計表」は二次ドキュメントとなる。

二次ドキュメントは、「二次ドキュメントタイプ」として定義される。例を、図13. に示す。図のように、二次ドキュメントタイプの各オカーレンスは、特定の項目（この例では「課名」）の値が等しい一次ドキュメントオカーレンス群をまとめたものである（単に全オカーレンスをまとめることもできる）。この際、二次ドキュメントは、ドキュメントタイプの登録時に一度だけ定義すればよく、これにより全オカーレンスを自動生成できる。

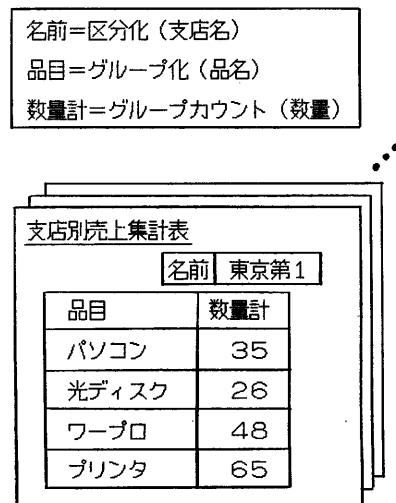


図14. 二次ドキュメントの定義と生成

生成された二次ドキュメントは、その実体を二次記憶に保管することも可能である。この意味で、二次ドキュメントタイプは、View集合と言うよりSnapshot集合と言える（「集合」としたのは、「タイプ」として定義されるためである）。保管された二次（物理）ドキュメントにより、次回からの二次ドキュメント参照を高速化することができ、また二次ドキュメント自体の修正も可能となる。

二次ドキュメントは、以下のような生成規則により定義される。

$$J_k = \text{区分化}(I_t) \quad \text{OR} \quad J_k = \Delta$$

$$J_s = F(I_1, I_2, \dots, I_m) \quad \text{OR}$$

$$= F(J_1, J_2, \dots, J_n)$$

I_t : 一次ドキュメントの項目 ($t=1, 2, \dots, m$) J_s : 二次ドキュメントの項目 ($s=1, 2, \dots, n$) J_k : 二次ドキュメントの識別キー F : 生成規則

ここで、 J_k は二次ドキュメントのオカレンスを一意に定めるキーである。「区分化」が指定されていると、項目 I_t の値が等しい一次ドキュメントオカレンス群がまとめられ、一個の二次ドキュメントオカレンスが作られる（二次のオカレンスは、識別キーの値ごとに存在する）。 $J_k = \Delta$ の時、 J_k は実項目であり、一次ドキュメントの全オカレンスがまとめられる（二次のオカレンスは一個である）。 F は仮想項目としての生成規則であり、一次ドキュメントから生成する時は、単純代入、四則演算、「合計/平均/最大/最小」、「グループ化」、「グループ合計（/平均/最大/最小/カウント）」などが指定可能である。

二次ドキュメント定義と生成の例を、図14. に示す（図11. のドキュメントを一次とする）。生成処理は、個々の一次ドキュメントオカレンスごとにまず演算を実行し、その結果（一次のオカレンス数分ある）を二次ドキュメントの繰返し項目に代入する、という形態が基本である。但し、上記した予約語に関しては、オカレンスにまたがった生成演算を行う。二次ドキュメント生成機能により、二次ドキュメントの上に三次ドキュメントを定義するというように、多次ドキュメントの定義・生成が行え、サマリドキュメントの作成が可能となる。

4. おわりに

テーブル形ドキュメント管理システムPALETTEの特長をまとめると、以下のようになる。

- ① 自由な形式のオフィス帳票・管理表を設計、処理、管理できる。
- ② 画面設計がそのままデータ設計になり、データベース定義が不要である。
- ③ データが格納済みでもドキュメントの様式を動的に変更でき、データベースの再構成が不要である。
- ④ ドキュメント単位と行単位という、二つのレベルの条件検索機能を持つ。
- ⑤ 二次ドキュメントの定義・処理機能を持ち、ドキュメントの集計処理が容易である。

PALETTEは、「ドキュメント単位のインタフェース」という基本概念のもとに設計され、利用者にとって自然で親しみやすいシステムとなっている。またその極めて柔軟なデータ管理機能のため、一般利用者でもデータベースの作成・保守が可能で、オフィス業務の変化にも容易に対応できる。

おわりに、本研究にあたって多くの貴重な助言を頂いた日本電気㈱C&Cシステム研究所コンピュータシステム研究部の服部光宏、金子朝男の両氏に感謝致します。

参考文献

- [1] 鶴岡、金子、西原「テーブル形ドキュメント管理システムPALETTEの情報管理方式」情報処理学会第25回全国大会、1J-10、57年10月
- [2] 渡部、鶴岡「テーブル形ドキュメント管理システムPALETTEにおける画面制御方式」情報処理学会第25回全国大会、3J-12、57年10月
- [3] 鶴岡、金子、西原「PALETTEにおける多階層化ドキュメントの管理方式について」情報処理学会第26回全国大会、3H-7、58年3月
- [4] 渡部、鶴岡「テーブル形ドキュメント管理システムPALETTEのユーザインタフェース」情報処理学会第27回全国大会、3F-2、58年10月
- [5] 渡部、鶴岡「テーブル形ドキュメント管理システムPALETTEのドキュメント検索方式」情報処理学会第28回全国大会、5C-8、59年3月