OŚRODEK BADAWCZO-ROZWOJOWY INFORMATYKI

PROGRAMY OPEROWANIA DANYMI

Europejski Program Badawczy Diebolda

31

Warszawa 1973

OŚRODEK BADAWCZO-ROZWOJOWY INFORMATYKI

PROGRAMY OPEROWANIA DANYMI

Europejski Program Badawczy Diebolda

Wyłącznie do użytku na terenie PRL

31

Warszawa 1973

Tytuł oryginału: Diebold Research Program - Europe
Technology Series - October 1969
Document E 61

Tłumaczenie: Jędrzej Spychalski
Redakcja: Elżbieta Kołodziejska
Opiniodawca: Andrzej Idźkiewicz

Komitet Redakcyjny

Mieczysław Gula, Franciszek Haratym, Andrzej Idźkiewicz, Janina Jerzykowska /sekretarz/, Jerzy Kisielnicki, Stanisław Nelken /zastępca przewodniczącego/, Krzysztof Skulski, Zdzisław Zapolski /przewodniczący/

Wydawca:

Działowy Ośrodek Informacji, Warszawa, ul. Marszałkowska 104/122

OBRI, Warszawa 1973 r. Nakład: 850+162 egz. Objętość: ark.wyd.6,5; ark. druk. 25. Format A4. Papier offsetowy, kl. III, 80 g 61 x 86

"Zespół".z.523/I/73n.1000 R-13 Cena zł 92.-

SPIS TREŚCI

		s.
	STRESZCZENIE	5
I.	WPROWADZENIE	9
	A. Przegląd zagadnień operowania danymi	9
	B. Wnioski	12
	C. Zakres i metodologia niniejszego sprawozdania	15
II.	PODSTAWOWA TERMINOLOGIA I TECHNIKI	19
	A. Terminologia dotycząca struktury danych	20
	B. Terminologia dotycząca struktury pamięci	24
III.	PRZYKŁADY OPEROWANIA DANYMI	35
	A. SIK w National Brewing Company	35
	B. System OMNIBUS w Industrial Indemnity Company	41
	C. FMS w Pacific Gas and Electric Company	50
	D. DATA FILE TWO w International Minerals and Chemical Corporation	56
	E. System MARK IV w kilku korporacjach	62
	F. System GIM w korporacji BCD	65
	G. Streszczenie i ocena przebadanych przykładów	71
IV.	UZASADNIENIE I TECHNIKA OPEROWANIA DANYMI	81
	A. Elementy i funkcje systemu zarządzania danymi	84
	B. Płaszczyzna styku z użytkownikiem	95
	C. Porównywalne cechy charakterystyczne uogólnionych systemów zarządzania danymi	111
	D. Napotykane trudności	132
V.	OCENA UOGÓLNIONYCH SYSTEMÓW ZARZĄDZANIA DANYMI	173
	A. Wprowadzenie	173
	B. Wzorcowe przykłady ocen przeprowadzonych przez przedsiębiorstwa	178
	C. Analiza wzorcowych ocen	184
	D. Streszczenie porównawczych czynników oceny	191
	Wykaz odsyłaczy	199
	Bibliografia	200

STRESZCZENIE

Zastosowania dnia dzisiejszego z zastosowaniami w przyszłości łączy uniwersalny system operowania danymi /lub inaczej: system kierowania przepływem danych/. Tego rodzaju system jest niezbędny wówczas, gdy środowisko przetwarzania danych ma się uporać z rosnącymi złożonymi problemami zarządzania danymi. Wiele czytelników stawia sobie pytanie: jak przekroczyć pomost łączący te zastosowania i opanować umiejętność zarządzania danymi.

Obecnie na rynku dostępnych jest wiele pakietów z programami operowania danymi. Zaprojektowano je z myślą o łatwiejszym i bardziej ekonomicznym wykonywaniu funkcji przechowywania, manipulacji i wyszukiwania danych w sposób uwalniający użytkownika od żmudnego opracowywania tych funkcji dla każdego nowego zastosowania. Jednakże, jak to wykazuje niniejsze sprawozdanie, bardzo mało systemów spełnia wszystkie kryteria funkcjonalności, które w chwili obecnej są tak pożądane, jak i technicznie osiągalne. Na przykład większość z nich jest zaprojektowana wyłącznie dla jednego sposobu wykorzystywania, czy to przez ludzi czy przez programy, a jest sprawą oczywistą, że przydatniejsze są sposoby złożone. Co więcej, wiele programów nie uwzględnia różnorodnych nośników dla przechowywania danych spotykanych dziś z reguły w dużych instalacjach trzeciej generacji, lecz ogranicza się zaledwie do jednego czy dwóch no-

śników. Niniejsze sprawozdanie Programu Badawczego Diebolda poświęcono analizie pewnej liczby systemów operowania danymi, ich głównym zaletom i wadom. W omawianej analizie położono nacisk na pakiety uogólnione/o zastosowaniu ogólnym/, z intencją udzielenia pomocy instytucjom uczestniczącym w Programie w wyborze lub w zaprojektowaniu ich własnego uogólnionego systemu operowania danymi,

W trakcie badań nad programami operowania danymi zanotowano następujące spostrzeżenia. Zwrócono uwagę na:

. optymalną organizację zbiorów, na którą składają się następujące złożone i wzajemnie uzależnione czynniki: wielkość zbioru, zmienność danych, wymagania odnośnie czasu odpowiedzi, środki przechowywania informacji, a także powszechny brak zrozumienia tego zagadnienia, nie przekraczający poziomu intuicji,

. zespół problemów związanych z potrzebą przenoszenia systemów /niezależność od maszyny/ z potrzebą przenoszenia danych /współwymienność zbiorów/ oraz związanych z sieciami wielorakich baz danych,

. stosunkowo jeszcze prymitywny poziom uogólnionych systemów operowania danymi i jednocześnie - konieczność ich tworzenia,

. coraz więcej przedsiębiorstw uznaje wartość uogólnionych systemów zarządzania danymi, lecz rezygnuje z ich opracowania ich własnymi siłami ze względu na wygórowany koszt

takiego rozwiązania. Wybranie odpowiedniego systemu z dostępnych na rynku nadal nie jest łatwe, choć trzeba podkreślić, że niektóre z przedsiębiorstw przeceniają występujące tu trudności. Niniejsze sprawozdanie wypunktowuje czynniki, krytycznie oceniające systemy oraz proponuje trójetapowy tok postępowania: ustalenie wymagań charakterystycznych dla danego przedsiębiorstwa, wstępne "przesianie" dostępnych systemów z punktu widzenia ich podstawowych elementów i funkcji oraz bardzo krytyczne porównanie zakwalifikowanych do rozpatrzenia systemów.

Niniejsze sprawozdanie jest przeznaczone dla osób zajmujących się oceną systemów, dla projektantów systemów oraz dla ludzi wdrażających systemy. Jest ono także przeznaczone dla kierownictwa przetwarzania danych oraz dla nowej gałęzi fachowców: administratorów baz danych.

I. WPROWADZENIE

A. Przegląd zagadnień operowania danymi

Bieżąca literatura na temat komputerów poświęca wiele uwagi uogólnionym systemom wyszukiwania informacji, systemom zarządzania bazą danych, systemom zarządzania zbiorami itp. Rozszerza się również terminologia związana z powyższymi problemami. Wraz z powstawaniem nowych systemów rozszerza się techniczna terminologia. Pojawiają się również trudności z jej rozszyfrowaniem.

To terminologiczne zamieszanie jest z jednej strony niekorzystne, ale posiada i pewne zalety. Niefortunne jest ono dla czytelnika, który usiłuje porównać różne systemy lub usiłuje powiązać opis systemu z własnym doświadczeniem czy ze swoimi potrzebami. Z drugiej strony jest ono zdrowym wyrazem rosnącej świadomości, że potrzeby oddzielenia logiczne i fizyczne aspekty danych należy traktować oddzielnie. Jest ono również przejawem rosnącego niezadowolenia z dotychczasowej terminologii, która była pojęciowo prosta, lecz reprezentowała w najlepszym wypadku kategorie myślenia charakterystyczne dla drugiej generacji.

Oczywiście, pewien stopień zamieszania wprowadzono celowo, aby stworzyć wrażenie kompetencji tam, gdzie jej wcale lub prawie dotychczas nie było. Będziemy się starali uniknąć podobnego zarzutu w stosunku do przedstawionego materiału.

Istnieje wiele przyczyn rozpowszechniania się omawianych systemów lecz główne wymagania, które te systemy starają się spełnić - w różnym stopniu - są dwa: większa niż dotychczas, niezależność danych od programów i sprzętu oraz stworzenie wspólnego elastycznego narzędzia do wygodnego przechowywania, manipulowania i wyszukiwania danych. Wynikiem większej niezależności danych od programów jest zmniejszenie nadmiaru danych, jak również rozszerzone ich zastosowanie. Większa niezależność danych od sprzętu oznacza zwiększenie możliwości ich przesyłania między systemami.

> Termin "zarządzanie danymi" /w języku angielskim: data management/ powstał ... celem wyrażenia ujednoliconych zasad, które się rozwinęły w posługiwaniu się funkcjami przetwarzania zbiorów przechowywanych w komputerze. /Zawiera on/ ... pojęcie elastycznego pakietu użytkowego, mającego zastosowanie w kilku dziedzinach. Termin ten implikuje łatwość przystosowania, zorientowanie na użytkownika, niezależność zbiorów, niskie wydatki na zdefiniowanie problemu kosztem wydajności przetwarzania.[1]

.....

> Dawniej zbiory danych w komputerach były projektowane i programowane głównie w celu zadośćuczynienia wymaganiom użytkownika w zakresie przechowywania rekordów. Mówiąc skromnie, zwykle sprawia trudności zmiana takich programów i struktur zbiorów ... Tak więc opracowano systemy "zarządzania danymi", które mogą tworzyć zbiory danych

lub zmieniać ich strukturę w sposób znacznie łatwiejszy niż odbywało się to w przeszłości. Systemy te także z łatwością wykonują zwykłe funkcje przetwarzania zbiorów.[2]

System zarządzania danymi powinien zapewnić użytkownikom wygodne i elastyczne środki opisywania, przechowywania, budowy, aktualizacji, utrzymywania oraz wyszukiwania danych. W celu osiągnięcia wyżej wzmiankowanej niezależności powinien on izolować użytkowników od kaprysów i "przyzwyczajeń" poszczególnych urządzeń wyposażenia komputera. Jako dodatkowe udogodnienie może on zapewniać pewne powszechnie stosowane uogólnione funkcje, takie jak generatory wydruków sprawozdawczych. W czasie przeprowadzania ostatecznej analizy tego rodzaju funkcji należy uważać za pomocnicze w stosunku do zasadniczego pojęcia o systemie zarządzania danymi, które określa, że zadaniem takiego systemu jest:

> wydzielić te funkcje odnoszące się do danych, które są wspólne dla większości zastosowań i wdrażać je w pewien uogólniony sposób - czyniąc je niemal niezależnymi od charakteru zastosowań - tak, by mogły one być używane w szerokim wachlarzu zastosowań.

Ostatecznie, to właśnie dziedzina zastosowań gwałtownie się rozszerza. Ciągłe wymyślanie od nowa dla każdego zastosowania funkcjonalnie identycznego oprogramowania systemowego oznacza skierowanie wysiłku rozwojowego na niewłaściwą drogę.

Istotne jest, że niewiele zastosowań będzie zawierało swój własny system zarządzania danymi tak, jak obecnie niewiele z nich posiada swój własny system operacyjny lub kompilator. Poza nielicznymi wyjątkami, które rzeczywiście wymagają całkowicie odrębnego rozpatrywania /wyjątków tych może być znacznie mniej niż niektórzy z nas przyjmują/, na system zarządzania danymi należy potrzeć, jako na narzędzie w skali całego systemu, zapewniające płaszczyznę styku z funkcjami systemu, na których może się opierać wiele różnych zastosowań. Na przykład, na podstawie systemu zarządzania danymi można by zbudować system informacyjny dla marketingu. Tu wadą jest, tak jak ma to miejsce dla kompilatorów i wielu innych narzędzi na skalę całego systemu, pewne zmniejszenie wykorzystania zasobów maszyny lub inaczej mówiąc wydajności w celu bardziej wydajnego wykorzystania pewnych innych zasobów /np. ludzi/.

B. Wnioski

Najważniejsze wnioski sformułowane w niniejszym sprawozdaniu można podsumować następująco:

. Największym problemem dnia dzisiejszego w dziedzinie operowania danymi jest optymalna organizacja zbiorów, ponieważ wchodzą tu w grę takie złożone i wzajemnie powiązane czynniki jak wielkość zbioru, zmienność danych, wymagania odnośnie czasu odpowiedzi, środki przechowywania informacji, a także powszechny brak zrozumienia tego zagadnienia, nie przekraczający poziomu intuicji. Rozwią-

zaniem tego problemu, które znajduje się w początkowym stadium realizacji, mogą być dynamicznie przystosowujące się struktury zbiorów /zautomatyzowany ciągły rozdział zasobów zbioru.

Tuż za tym zagadnieniem wyłania się zespół problemów związanych z potrzebą przenoszenia systemów /niezależność od maszyny/, z potrzebą przenoszenia danych /współwymienność zbiorów/ oraz z sieciami wielorakich baz danych. Cechę możności przenoszenia systemu można by zacząć wypracowywać przez wprowadzenie jego modularności oraz przez stosowanie specjalizowanych języków, a możność przenoszenia danych można by uzyskać przez stosowanie danych samodefiniujących się oraz uniwersalnego języka opisu danych. Najważniejszym nierozwiązanym zagadnieniem w zakresie sieci baz danych jest powszechnie występująca niemożność ustanowienia między nimi połączeń.

Znajdujemy się w okresie przejściowym. Funkcje skomputeryzowanego operowania danymi towarzyszą nam od chwili pojawienia się komputerów, lecz rola tych funkcji oraz ich wdrażanie ulega przesunięciu jeśli chodzi o kierunek realizacji. Wskutek tego powstają obecnie uogólnione systemy zarządzania danymi jeszcze - co prawda - stosunkowo prymitywne, ale szybko osiągające taki stopień złożoności, że będą mogły stać się wkrótce głównym czynnikiem w rozwiązywaniu tego problemu w większości przedsiębiorstwach dzięki zdolności tych systemów do przystosowania się do zachodzących zmian.

. Płaszczyzna styku z otoczeniem - zewnętrzna - /między systemem a korzystającymi z niego programami /lub ludźmi w większości uogólnionych systemów zarządzania danymi zawiera specjalny moduł języka użytkownika oraz moduł generatora wydruków sprawozdawczych. Jest to uzasadnione wygodą użytkownika, ale również zapewnia znakomity oddźwięk na rynku. W rzeczywistości bardziej pożądane byłoby posiadać dobrze zdefiniowane i opublikowane wymagania w zakresie płaszczyzny styku /między systemem a użytkownikiem/, z dostępnymi według wyboru użytkownika modułami, a to celem uzyskania sprzężenia z systemem dla każdego żądanego przez użytkownika modułu płaszczyzny styku.

. Istnieje rzeczywista potrzeba streszczania danych celem zaoszczędzenia pojemności pamięci przy przechowywaniu ogromnych baz danych. Można by uzyskać poważne korzyści, dzięki którym zmniejsza się koszt urządzeń pamięci a także, do pewnego stopnia, czas dostępu i wyszukiwania informacji. Jednak niewiele systemów posiada takie własności.

. Najkrótsze czasy wyszukiwania informacji zależą bardziej od struktury pamięci oraz od strategii dostępu do niej niż od środka przechowywania informacji.

. Przedsiębiorstwa, które opracowują swoje własne systemy, mogą mimo woli zostać związane na wiele lat z jakąś szczególną konfiguracją sprzętu, daje tu znać o sobie tendencja do sztywnych systemów ściśle przykrojonych wyłącznie dla potrzeb określonego zastosowania.

. Nie ma uniwersalnej metody pomagania przedsiębiorstwom w wyborze tego "właściwego" uogólnionego systemu operowania danymi. Nie ma także łatwej drogi obejścia tego problemu. Każde przedsiębiorstwo musi określić swoje własne specyficzne wymagania. Następnie dokonuje oceny porównawczej: w jakim stopniu dostępne systemy spełniają postawione wymagania. Dysponujemy pewnymi ogólnymi wskazaniami w tym zakresie, a w rozdziale V zidentyfikowano dziesięć krytycznych czynników porównawczych.

C. Zakres i metodologia niniejszego sprawozdania

Wszechstronne omówienie wszystkich form systemów przechowywania i wyszukiwania wykracza poza ramy niniejszego sprawozdania. Szczególnie wyłączono tutaj systemy zarządzania danymi o dowolnym układzie treści, znane pod nazwą systemów zbiorów nieukształtowanych /unformatted file systems, np. TIPS/. Są one właściwym tematem jednego z wcześniejszych sprawozdań Programu Badawczego Diebold'a /pt. Library Systems and Information Science, MC20/. Także wykluczono tutaj "czyste systemy wyszukiwania", na przykład programy typu RPG /Report Program

Generator - generator programów sprawozdawczych/. Z samego założenia niniejsze sprawozdanie jest poświęcone systemom, które zajmują się danymi ukształtowanymi /być może z drobną ilością danych nieukształtowanych/ w sposób typu przechowywanie-manipulowanie-wyszukiwanie. Tego rodzaju systemy są powszechnie zwane systemami zarządzania danymi lub systemami zarządzania bazą danych.

Ujęcie lub odrzucenie jakiegoś szczególnego systemu w niniejszym sprawozdaniu nie powinno prowadzić do jakichkolwiek wniosków na temat jego zalet. Rozdziały III oraz IV omawiają kilka systemów bardziej szczegółowo. Systemy te zostały wybrane, ponieważ są dobrze znane lub też dlatego, że ilustrują one wartościowe cechy charakterystyczne czy też reprezentują tendencje rozwojowe.

Całokształt sprawozdania uwzględnia następującą tematykę:

- podstawowe określenia i klasyfikację struktur danych oraz struktur pamięci, włączając w to podstawowe możliwości kompromisów,
- przykłady różnych struktur danych oraz struktur pamięci zaczerpnięte ze środowiska zastosowań, ich zasadnicze własności, możliwości oraz niedomagania,
- elementy i możliwości funkcjonalne uogólnionych systemów zarządzania danymi, istotne cechy charakterystyczne oraz różnice między dziesięcioma dostępnymi na rynku systemami,

- zagadnienia techniczne, które zakreślają granice dla umiejętności w tym zakresie,
- kluczowe aspekty oceny uogólnionych systemów zarządzania danymi dla specyficznych wymagań danego przedsiębiorstwa.

Metodologia stosowana podczas opracowywania niniejszego sprawozdania polegała na przeglądzie literatury, na wywiadach badawczych z przekrojem projektantów, użytkowników i pracowników naukowych zajmujących się zagadnieniem systemów zarządzania danymi, na jednodniowej sesji roboczej Programu Badawczego Diebolda poświęconej przede wszystkim potrzebom użytkowników w tym zakresie oraz na wszechstronnej analizie wniosków.

II. PODSTAWOWA TERMINOLOGIA I TECHNIKI

Instytut Normalizacji Stanów Zjednoczonych A.P. /United States of America Standards Institute, w skrócie USASI/ zdefiniował dane jako "jakiekolwiek przedstawienie czegoś w postaci symboli, takich, jak znaki lub ilości analogowe,do których może być przypisane jakieś znaczenie". Słownik pn. Webster's Seventh New Collegiate Dictionary mówi: " coś dane lub przyjęte szczególnie, jako podstawa dla rozumowania lub wnioskowania; materiał źródłowy /oparty na faktach/ użyty, jako podstawa dyskusji lub podejmowania decyzji; coś co jest wykorzystywane jako podstawa dla obliczeń lub pomiarów". /Należy tu zauważyć, że "informacja" może być wyprowadzana z danych lub z zależności między danymi/. Definicja podana przez Instytut Normalizacji Stanów Zjednoczonych A.P./USASI/ z pewnością jest wystarczająca dla celów niniejszego sprawozdania.

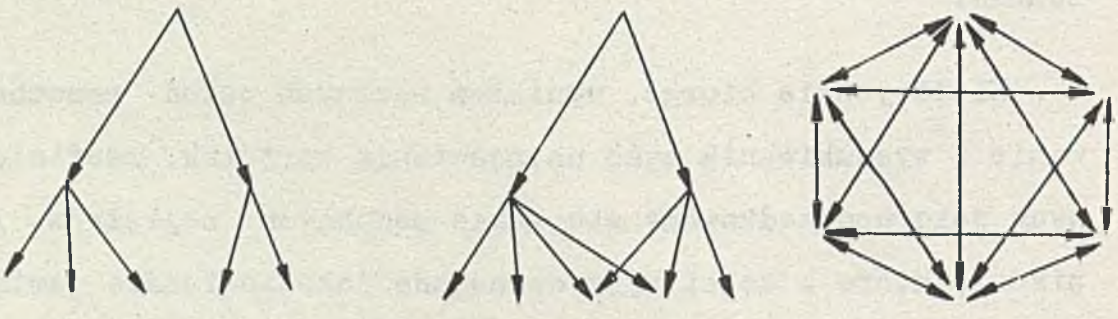
Historycznie biorąc, wynikiem ręcznych metod przechowywania i wyszukiwania było ustanowienie kartotek, zdefiniowanych jako uporządkowane skupienie podobnych rejestrów /zapisów/, które z kolei były określane jako logicznie powiązane skupienie zdarzeń źródłowych /faktów/. Rejestr /zapis/ był to zwykle arkusz papieru, na którym rejestrowano odpowiednie fakty. Zestaw rejestrów /zapisów/ ułożonych w pewnym porządku stawał się wówczas kartoteką /lub zbiorem/.

W naszych rozważaniach będziemy używać zarówno tych popularnych określeń, jak i szeregu nowych, dodatkowych terminów.[x/] Wprowadzamy następujące rozróżnienie między strukturą danych a strukturą pamięci. Struktura danych - jest to zewnętrzny wygląd danych, który służy do opisania logicznej organizacji danych. Struktura pamięci - jest wewnętrznym przedstawieniem danych w taki sposób, jak są one przechowywane w pamięci i zawiera dane, które opisują inne dane.

A. Terminologia dotycząca struktury danych

Większość roboczych zastosowań, dla których gromadzi się dane, można dogodnie przedstawić w postaci drzewek hierarchicznych/hierarchii/, drzewek odwróconych lub w postaci innych dających się logicznie wyrazić struktur.

Drzewko "czyste" Drzewko odwrócone Sieć



W celu wyszczególnienia struktur danych w niniejszym sprawozdaniu użyto następującej terminologii:

x/ Szerokie kręgi informatyków w Polsce używają także terminu rekord, jako równoważnika angielskiego terminu record /przyp.tłumacza/.

Bank danych	Poziom najwyższy
Baza danych	!
Zbiór	!
Wpis	!
Grupa	!
Pozycja	Poziom najniższy

1. Pozycja

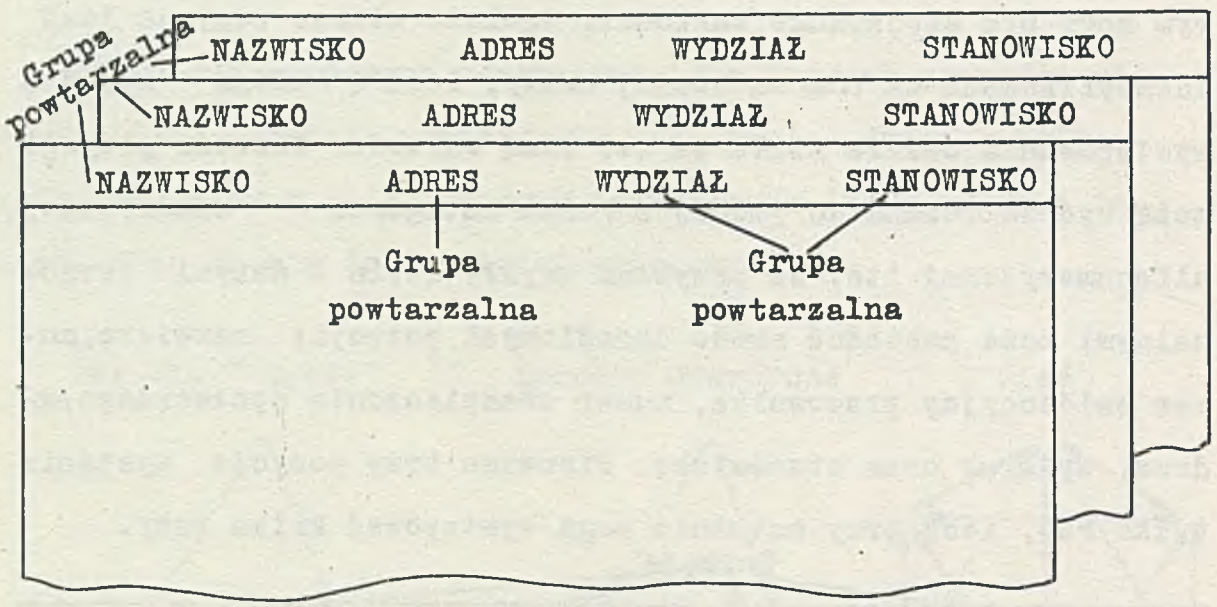
Pozycja jest najbardziej podstawową jednostką logiczną w strukturze danych i jest jedynym w tej strukturze poziomem, z którym mogą być skojarzone wartości. Ogólnie biorąc pozycja jest identyfikowana za pomocą swojej nazwy, lecz w każdym wypadku występowania będzie miała zwykle inną wartość. Wartość pozycji może być zaliczona do jednej z kilku kategorii - numerycznej, alfanumerycznej itd. Na przykład zwykły zapis z danymi personalnymi może posiadać sześć określonych pozycji: nazwisko, numer ewidencyjny pracownika, numer ubezpieczenia społecznego, adres, wydział oraz stanowisko. Pierwsze trzy pozycje wystąpią tylko raz, lecz trzy ostatnie mogą występować kilka razy.

Pozycje

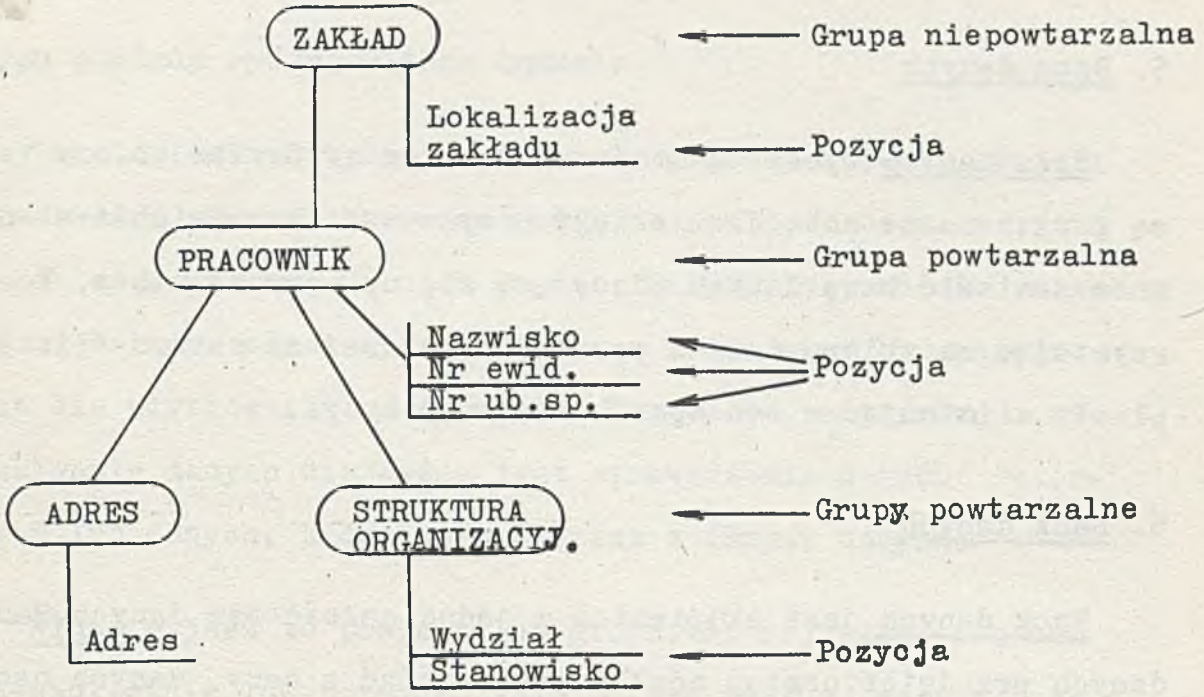
NAZWISKO	ADRES	WYDZIAŁ	STANOWISKO
Stanisław W.Nowak	Warszawa /6/62/ Marszałkowska 12	241	Starszy /6/62/ projektant
Nr Ewidencji	Milanówek /4/69/ Podleśna 3	241	Asystent/8/65/ dyrektora
4108		263	Dyrektor /3/69/
Nr Ubezpieczenia	Wartości		

2. Grupa

Grupa - jest skupieniem lub skojarzeniem jednej lub więcej logicznie powiązanych pozycji; grupa dopuszcza wielokrotne występowanie różnych wartości pozycji. Grupa może zawierać jakąś inną grupę lub też mogą występować zależności między grupami typu: grupa nadrzędna - grupa podporządkowana. W podanym wyżej przykładzie pozycje "wydział" oraz "stanowisko" są oczywiście powiązane ze sobą i tworzą grupę. Co więcej, jest to grupa powtarzalna, podporządkowana innej grupie powtarzalnej "nazwisko".



Hierarchię struktury danych w zbiorze wydziału personalnego w jednym z zakładów korporacji przemysłowej można by przedstawić następująco:



3. Wpis

Wpis jest zgromadzeniem grup użytych do sporządzenia zbioru; jest to zapis logiczny według "starej" terminologii. Powyższa ilustracja mogłaby przedstawiać strukturę wpisu w zbiorze danych personalnych dla korporacji o ile potraktujemy"zakład"jako grupę powtarzalną. W wyniku tego każdy wpis będzie ujmować personel w innym zakładzie produkcyjnym.

4. Zbiór

Zbiór jest to kompletowanie wszystkich powiązanych logicznie wpisów w jeden zestaw.

5. Baza danych

Baza danych jest zgromadzeniem zbiorów. Zwykle zbiory te są powiązane ze sobą dla jakiejś przyczyny. Przedsiębiorstwo może zestawić bazę danych odnoszącą się do pracowników, korzystając ze zbioru danych personalnych oraz ze zbioru "lista płac", eliminując w ten sposób zbędne pozycje.

6. Bank danych

Bank danych jest skupieniem w jedną całość baz danych. Bank danych przedsiębiorstwa mógłby się składać z bazy danych osobowych płacowych, bazy danych dotyczących produkcji-zamówień zapasów, bazy danych dotyczących zbytu-marketingu-produkcji, i tak dalej.

B. Terminologia dotycząca struktury pamięci

Celem organizacji zbioru jest łączenie pozycji, które "prawdopodobnie" będą rozważane łącznie... Tam gdzie zastosowanie nie jest dokładnie zdefiniowane lub gdzie wielorakie wykorzystywanie nie pozwala na zorganizowanie zbioru w kategoriach poszczególnych możliwych do przewidzenia potrzeb. Organizacja zbioru musi stać się zespołem /kompleksem/ struktur. Celem każdej podstruktury w takim zespole jest zapewnienie dostępu z punktu widzenia jakiegoś rodzaju wpisu, dla którego została ona zorganizowana w taki sposób, że odpowiada mu optymalnie - zatem celem jest łączenie pozycji, które dla danego rodzaju do-

stępu powinny być rozważane łącznie.[3]

Terminologia dotycząca struktury pamięci opisuje, w jaki sposób dane są przechowywane wewnątrz pamięci oraz w jaki sposób zapewnia się do nich dostęp. Struktury pamięci mogą być niezależne od struktur danych, a także mogą być niewidoczne dla użytkownika. W celu przechowywania, a później wyszukiwania danych niezbędne jest wprowadzenie danych "wtórnych" lub danych, które są powiązane z innymi danymi.

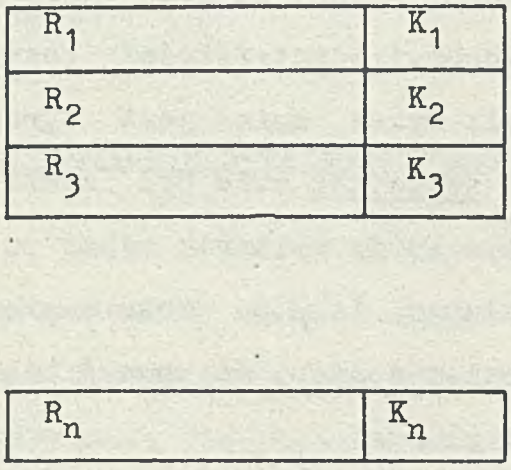
Klucz - jest to pozycja lub grupa, której używa się do przeszukiwania lub porządkowania zbioru według wartości tego klucza.

Łącznik lub wskaźnik /wskazówka/ - jest to dana wtórna, która określa adres /lub jakoś inaczej identyfikuje/ pozycji, grupy lub wpisu. Indeks /czyli wykaz, spis/ jest zgromadzeniem wskaźników. Słownik lub skorowidz może być zdefiniowany jako zestaw indeksów, gdzie każdy wskaźnik jest powiązany z jedną tylko wartością kluczą, tak, że może nastąpić proces przeglądania tablicy celem dostępu do pewnej danej na podstawie zadanej wartości klucza.

Dla uproszczenia termin "zapis" będzie w niniejszej dyskusji używany dwuznacznie, lecz zwykle można uważać, że oznacza on "wpis" o ile nie zaznaczono inaczej. Przykłady i odmiany będą stosunkowo oczywiste i proste dla osiągnięcia zwięzłości.

1. Struktury sekwencyjne

Prawdopodobnie najbardziej znaną metodą organizacji zbiorów jest struktura sekwencyjna ponieważ w całości wywodzi się ona z ręcznych środków obliczeniowych. Każdy zapis jest przechowywany logicznie i fizycznie w pozycji, która posiada powiązania z innymi zapisami według ściśle określonej sekwencji. Ta sekwencja jest określona za pomocą klucza używanego do porządkowania zbioru, który to klucz jest zwykle używany w celu uzyskania dostępu do zbioru. Taki klucz może być utajony, jak to ma miejsce w przypadku zbiorów uporządkowanych według kolejności napływających wpisów.



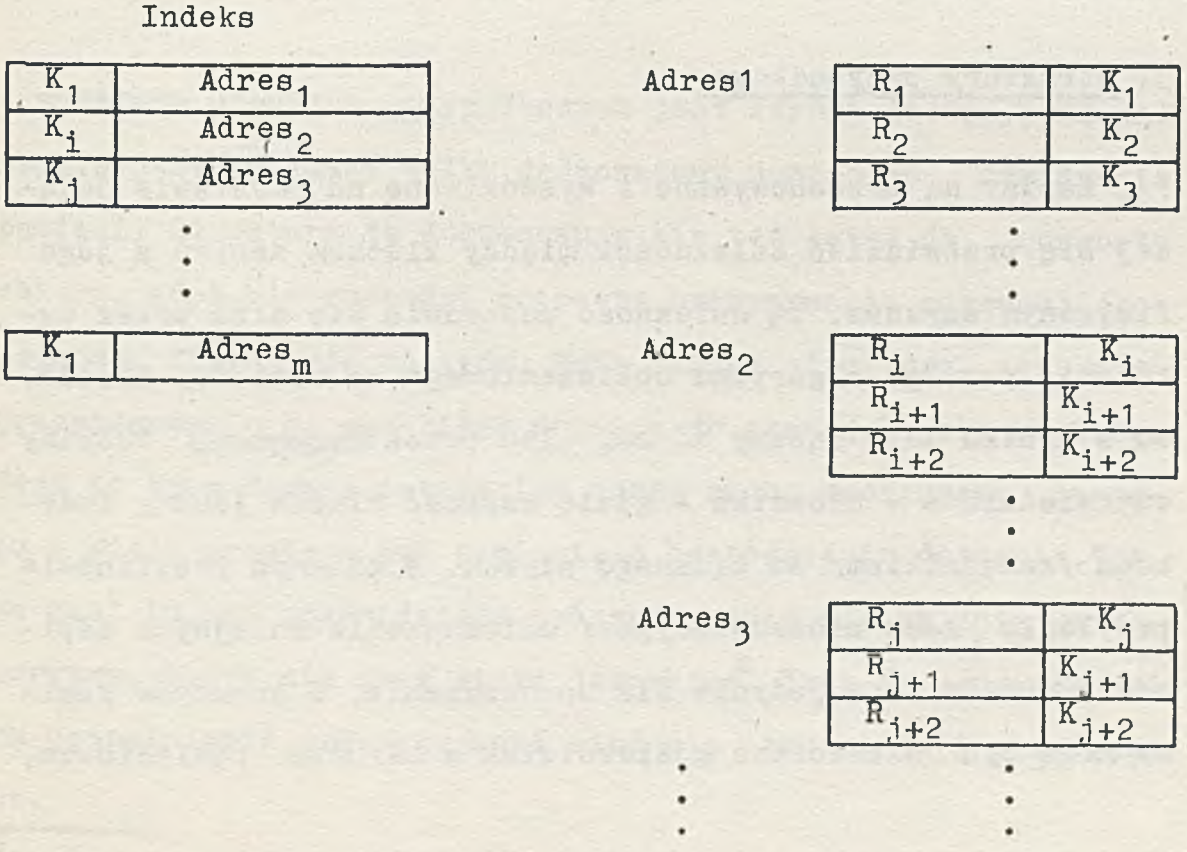
R_i = i-ty zapis
K_i = klucz sekwencji gdzie i jest numerem zapisu lub innym wskaźnikiem porządkowym

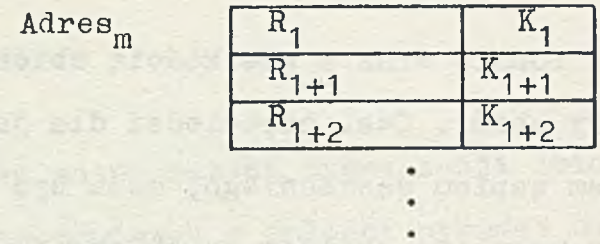
Zaletą struktury sekwencyjnej jest oszczędność pojemności pamięci, gdyż nie są wymagane żadne wskaźniki. Zwykle środkiem przechowywania informacji jest taśma magnetyczna. Jest ona stosunkowo niedroga. Kiedy mechanizm dostępu jest ustawiony na pozycji zapisu i, to następny zapis /i+1/ osiąga się prosto i szybko. Wadą tego systemu jest to, iż czas odpowie-

dzi rośnie wraz z wielkością zbioru, jeśli trzeba przeszukiwać cały zbiór. Czas odpowiedzi dla jakiegokolwiek zapisu z wyjątkiem zapisu sąsiedniego, może być dosyć duży. Zamieszczanie lub usuwanie zapisów ze zbioru jest dosyć kłopotliwe i wymaga kopiowania całego zbioru w celu jego aktualizowania.

2. Struktury sekwencyjne indeksowane

Jeden z wariantów organizacji sekwencyjnej jest zwany organizacją sekwencyjną indeksowaną. W takim wypadku ustanowiony indeks służy jako "tabela zawartości" zbioru. Podobny wariant, tutaj nie przedstawiony, zwany jest organizacją sekwencyjną wydzielaną; podstawowa zasada jest identyczna.





Zaletę struktury sekwencyjnej indeksowej, w porównaniu ze zwykłą strukturą sekwencyjną, stanowi szybszy dostęp do zapisu w urządzeniu pamięci o bezpośrednim dostępie. Szybszy dostęp uzyskuje się przez podział zbioru na oddzielnie adresowane podzbiory, zazwyczaj przyrównane do ścieżek i cylindrów na dysku. Wadę tego rozwiązania stanowi dodatkowa pojemność, która ma służyć do przechowywania indeksu, dostęp wewnątrz podzbioru jest w dalszym ciągu sekwencyjny. Także aktualizacja zbioru może wymagać również aktualizacji indeksu.

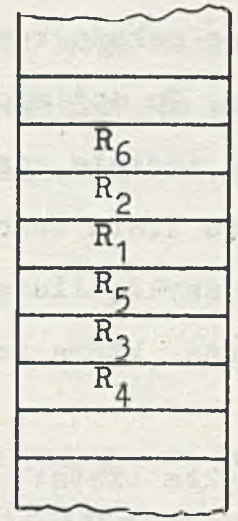
3. Struktury przypadkowe

Zapisy są przechowywane i wyszukiwane na podstawie dającej się przewidzieć zależności między kluczem zapisu a jego fizycznym adresem. Tę zależność ustanawia się albo przez wykonanie pewnego algorytmu obliczeniowego na wartości klucza, co w wyniku daje żądany adres, albo przez utrzymywanie tablicy odpowiedników - słownika - gdzie wartość klucza jest indeksem /wskaźnikiem/ do żądanego adresu. W podanym przykładzie przyległe /lecz niesekwencyjne/ umieszczenie kolejnych zapisów przedstawiono jedynie dla uproszczenia. W praktyce zapisy mogą być umieszczone gdziekolwiek w ośrodku pamięciowym,

którym zazwyczaj jest urządzenie pamięci o bezpośrednim dostępie.

Słownik

Wartość klucza	Adres zapisu
1	i+2
2	i+1
3	i+4
4	i+5
5	i+3
6	i



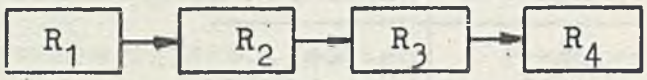
Zaletą struktur przypadkowych jest szybki dostęp, ponieważ zwykle jest wymagany tylko jednorazowy dostęp do urządzenia pamięci. Struktura ta dostosowuje się też łatwo do rozrostu zbioru, gdyż nie zachodzi potrzeba utrzymywania sekwencji/kolejności fizycznej/ zapisów. Wadą takiej struktury jest nieprzystosowanie do szybkiego dostępu do pewnej liczby zapisów. Jest to spowodowane istnieniem czasu ruchu mechanizmu dostępu w wielu urządzeniach pamięci, o bezpośrednim dostępie. Także musi tu być przewidziane zabezpieczenie dla wyjścia z algorytmu, które nie jest stale jednakowe[x]. Wraz ze wzrostem zbioru uzupełnianie słownika może stać się utrudnione i nieporęczne.

[x] Chodzi tu o różne rozmiary rezultatów wykonania algorytmu, co może spowodować nie umieszczenie się w jednym elemencie struktury pamięci.

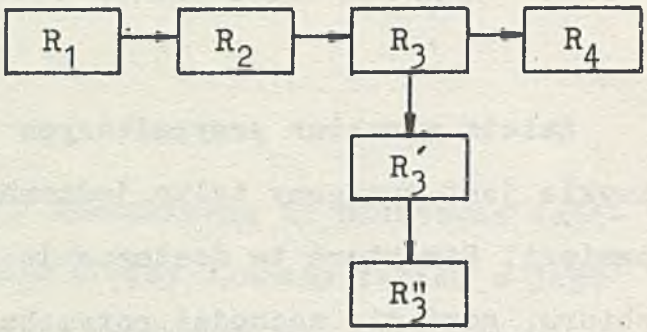
4. Struktura listująca

Lista jest skupiskiem zapisów, które są ze sobą powiązane za pomocą wskaźników - niezależnie od ich fizycznego umiejscowienia. Tu najistotniejszą rzeczą jest to, że polu klucza w każdym zapisie zostaje przypisany łącznik i powstała w wyniku tego lista zwana jest łańcuchem. Poniżej przedstawiono zarówno zwykłą listę, jak i strukturę listującą; ta ostatnia jest listą, która zawiera co najmniej jedną inną listę.

Zwykła lista:



Struktura listująca:

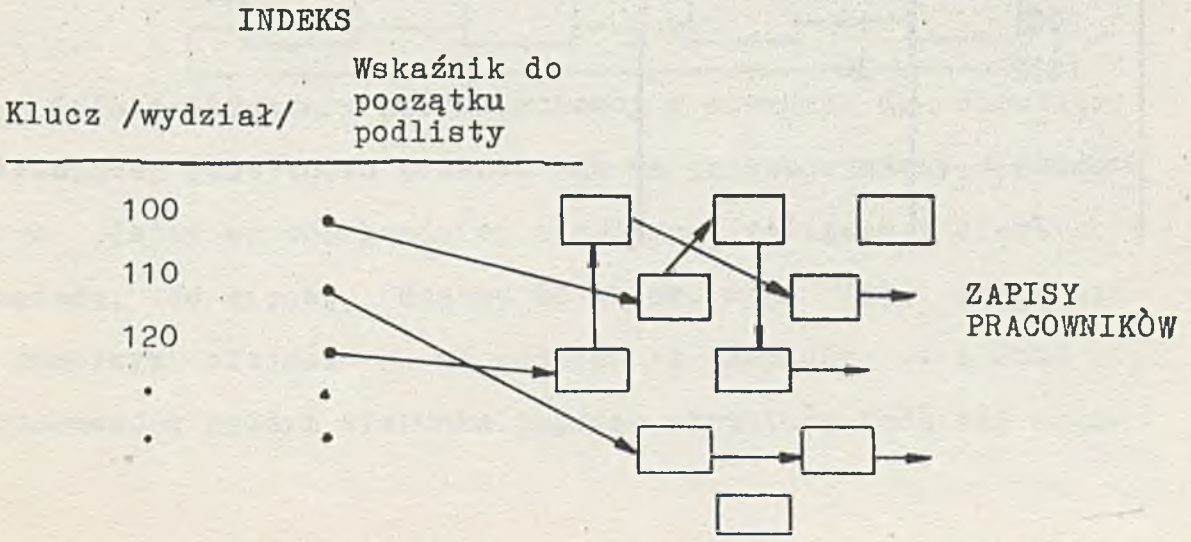


Zaleta struktur listujących polega na tym, że oddzialają one organizację struktury od problemów organizacji fizycznej osiąga się to przez stosowanie wskaźników Pozwalają one na to, by zapis należał do kilku list, ponieważ każda pozycja może być użyta jako klucz, z którym może być powiązany wskaźnik. Struktury listujące pozwalają na łatwą aktualizację zbioru ponieważ zapis można zamieścić, usunąć lub przesunąć w inne miejsce przez zwykłą zmianę wskaźnika. Wadą jest tutaj to, iż

sekwencyjne przeszukiwanie listy dla długich list może się stać bardzo czasochłonne, wielkość zapisu /a stąd i niezbędna pojemność pamięci/ wzrasta z powodu konieczności umieszczenia w nim wskaźnika. Dodatkowe wskaźniki mogą okazać się niezbędne dla zapisów wielolistowych /umieszczonych w wielu listach/ w celu przemierzania list w obu kierunkach podczas ich aktualizacji.

5. Odwrócone struktury listujące

Jedna z odmian list w swej istocie polega na tworzeniu podlist /list cząstkowych/ przez budowanie indeksu na jednym lub kilku kluczach z listy. Częściowo odwrócona lista będzie posiadała indeks dla co najmniej jednego klucza, w pełni odwrócona lista będzie indeksowana przez wszystkie klucze. Wcześniej wykorzystany przykład z zapisami dotyczącymi personelu przedstawia częściowo odwróconą listę jeśli ustanowimy indeks według numeru wydziału.



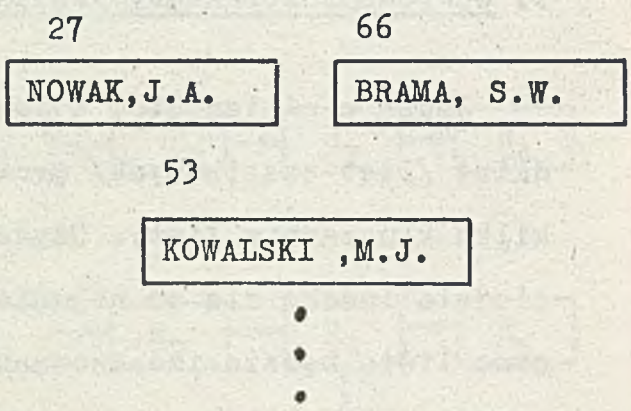
Lista w pełni odwrócona będzie indeksowana przez wszystkie klucze - z wszystkimi wskaźnikami zawartymi w indeksie - tak, że każda podlista będzie zmniejszona w swej długości do jednego zapisu.

INDEKS

KLUCZE WSKAŹNIKI

WYDZIAŁ	ADRESY
100	106,27, ...
110	53,72, ...
.	.
.	.
.	.
STANOWISKO	**ADRESY**
Analityk	46,53, ...
St.analit.	125,66, ...
.	.
.	.
.	.
NR EWIDENCJI	**ADRES**
1001	14
1002	36
.	.
.	.
.	.

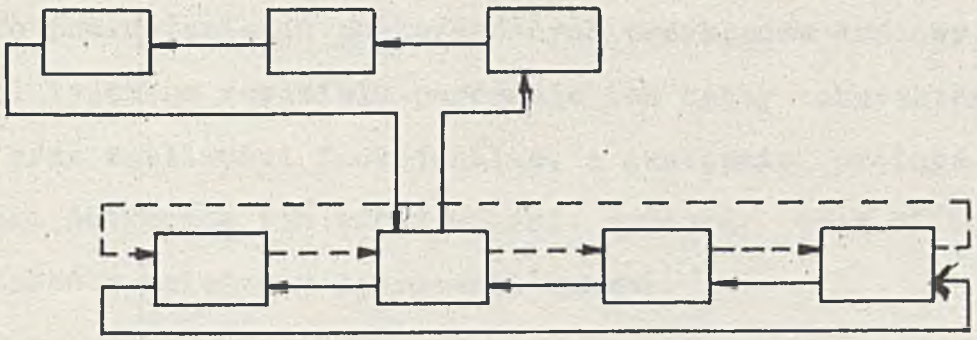
.
.
.



Zaletę odwróconych struktur listujących, w odróżnieniu od struktur listujących, jest szybszy dostęp do żądanej/indeksowanej/ danej. Wymagane jest wyszukanie jedynie danych mających związek z problemem. Wadą takich struktur jest utrudnione i czasochłonne przechowywanie i aktualizowanie spowodowane koniecznością utrzymywania bardzo dużych słowników.

6. Struktury pierścieniowe

Struktura pierścieniowa jest łańcuchem zapisów, w której ostatni zapis wskazuje na pierwszy. Mogą być także użyte dodatkowe wskaźniki celem wskazywania w kierunku wstecznym wzdłuż łańcucha zapisów, albo wskaźniki pokazujące każdemu zapisowi "zapis początkowy" w łańcuchu. Każdy z zapisów w łańcuchu może być zapisem początkowym dla innego pierścienia.



Zaletą struktury pierścieniowej w stosunku do struktury listującej jest to,że pozwala ona na przechowywanie w stosunkowo łatwy sposób bardziej złożonych powiązań i hierarchii zapisów, co więcej, dostęp do zbioru może się rozpocząć w dowolnym miejscu - od każdego z zapisów - i może się przesuwać w każdym kierunku poprzez strukturę.Wadą tej orga-

nizacji jest zazwyczaj wymagana dodatkowa pojemność pamięci na przechowywanie wskaźników.

Następny rozdział bada kilka systemów, w których wdrożono niektóre z powyżej zdefiniowanych koncepcji.

III. PRZYKŁADY OPEROWANIA DANYMI

Przykłady dają więcej, niż tylko fizyczną ilustrację koncepcji omówionych w poprzednim rozdziale sprawozdania. Zakreślają one również takie kluczowe obszary rozważań jak: czy nabyć uogólniony system zarządzania danymi czy też opracować go własnymi siłami, wielkość bazy danych i towarzyszące jej problemy środków przechowywania informacji oraz struktury pamięci, niezależność danych i programów, zintegrowane wielorakie bazy danych, a także systemy informacji kierownictwa przedsiębiorstw. Te oraz inne zagadnienia zakreślają w pewnej części ramy pozostałych rozdziałów i będą ponownie poruszane po zapoznaniu się z doświadczeniami przedstawionych poniżej przedsiębiorstw.

Po przedstawieniu poszczególnych przykładów końcowy odcinek niniejszego rozdziału porównuje ich cechy charakterystyczne oraz możliwości funkcjonalne, a następnie wyciąga pewne wnioski dotyczące ich efektywności, wymagań oraz kilku doświadczeń z dziedziny operowania danymi.

A. SIK[x] w National Brewing Company /w przemyśle piwowarskim/[4]

1. Tło i informacje ogólne

Przedsiębiorstwo National Brewing Company, z siedzibą w

---

[x] SIK = System informacji kierownictwa /za dr. Andrzejem Targowskim/ - w języku angielskim MIS = Management Information System.

Baltimore, Maryland, opracowało system informacji kierownictwa, dla którego ustanowiono następujące wytyczne:

Zbierać w jednej zintegrowanej bazie danych wszystkie informacje do celów zarządzania i na czas podejmowanych decyzji gospodarczych, syntetyzując tę informację raz dziennie dla kierownictwa, by odzwierciedlić wpływ działalności w poprzednim dniu na zestawienie zysków i strat.

Opisać postęp w realizacji celów przedsiębiorstwa i poszczególnych jego wydziałów, przyjmując jako podstawę pomiaru dzień, miesiąc i rok wstecz od dnia bieżącego tak, by ujawnić wahania w czasie w celu podjęcia odpowiedniej akcji korygującej; stąd wpisy dotyczące jakiegoś działania powinny następować w czasie jego wystąpienia, o ile tylko jest to możliwe.

Zapewnić bieżącą informację w formacie, który nadaje się do zdalnego przetwarzania zapytań ze strony kierownictwa i który zdradza tendencję do wzrostu, o ile przedsiębiorstwo rozszerzyłoby swoją działalność gospodarczą.

Kierownictwo NBC nie czuło się skrępowane w utorowaniu drogi w tej dziedzinie oczekując, że uzyskane doświadczenie i zalety systemu wyrównają, jeśli nie przewyższą, ryzyko rozpoczęcia tego rodzaju pionierskich prac. Proces projektowania rozpoczęty w 1965 roku dał w wyniku zdefiniowanie nas-

tępujących siedmiu funkcjonalnych podsystemów:

- system danych finansowych /księgowość ogólna/,
- system analizy sprzedaży i zysku /łącznie z kontami dłużników oraz wpływem zamówień/,
- system sprawdzania i uzgadniania sprzedaży ruchomej /dla sprzedaży rozprowadzanej własnym transportem/,
- system zautomatyzowania zakupów i płatności rachunków /dla dostawców/,
- system sterowania i sprawozdawczości dla zapasów,
- system planowania produkcji i sprawozdawczości z jej wykonania,
- system zatrudnieniowo-płacowy.

Z wszystkimi wyżej wymienionymi systemami posiada płaszczyznę styku ósmy podsystem zdalnego przetwarzania zapytań i odpowiedzi.

Po roku prac nad projektem przedsiębiorstwo NBC zdecydowało się nie nabywać uogólnionego systemu operowania danymi, lecz opracować i wdrożyć swój własny, wewnętrzny. Chociaż istniało wówczas niewiele tego rodzaju uogólnionych systemów, to jednak decyzję tę oparto w pierwszym rzędzie na opinii kierownictwa NBC, że przedsiębiorstwo i sposób, w jaki prowadzi swoją działalność gospodarczą, jest jedyne w swoim rodzaju i z tego powodu żaden z uogólnionych syste-

mów nie mogłby zadowolić stawianych przez nie wymagań. W trzy lata po podjęciu tej decyzji w dalszym ciągu uważano, że była ona słuszna.

Do chwili obecnej w opracowanie systemu zainwestowano około 30 osobolat /trzy lata programowania/i system jest, biorąc z grubsza, gotowy w 40 procentach. System danych finansowych, którego strukturę logiczną pokazano na tablicy 1, był pierwszym wdrożonym modułem funkcjonalnym.

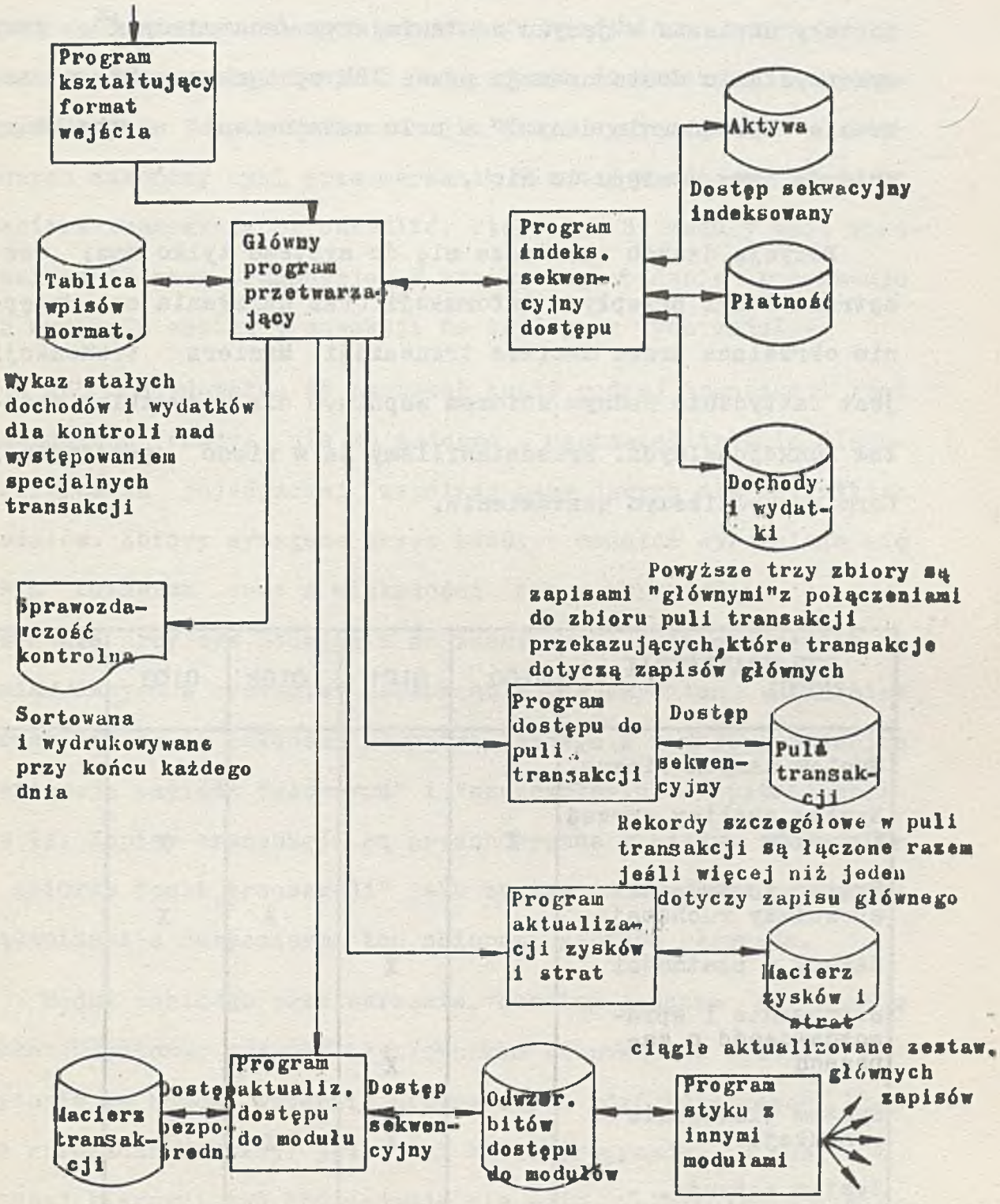
2. Cechy charakterystyczne oraz właściwości specjalne

Kierownictwo NBC, posiadając 4.000 klientów, dzienny cykl wprowadzania informacji i sprawozdawczości oraz / w chwili obecnej/ 2314 /stacja pamięci dyskowej IBM/ dla przechowywania bazy danych, doszło do wniosku, że jedyną drogą spełnienia swoich celów jest wykonanie ściśle "przykrojonego" własnego systemu zarządzania bazą danych. Co więcej, zdecydowano się wcielić operowanie danymi - jako integralną część - do całkowitego systemu, nie wyróżnialną, jako wydzielony element.

W rzeczywistości SIK w NBC jest systemem wielorakich baz danych, pomimo nakreślonego celu w postaci pojedynczej zintegrowanej bazy danych, przy czym każdy obszar funkcjonalny utrzymuje swoją własną bazę danych. Podczas projektowania trzymano się zasady "z góry - w dół" /w skali hierarchii przedsiębiorstwa/ w celu wyodrębnienia i zdefiniowania modułów fun-

Tablica
E 61-1 SYSTEM DANYCH FINANSOWYCH W NBC

Wejścia: Zlecenia dzienne , zawiadomienia debet/kredyt ,zlecenia czekowe



kcjonalnych i ich wymagań w zakresie wewnętrznej łączności. Jak dotąd, praktycznie biorąc, wszystkie ukończone programy zostały napisane w języku zestawiającym /montującym/[x], przy wykorzystaniu dostarczanego przez IBM oprogramowania w zakresie "zarządzania danymi" w celu ustanowienia struktury zbiorów oraz dostępu do nich.

Pozycję danych wprowadza się do systemu tylko raz; wewnętrzne drogi przepływu informacji oraz działania są wstępnie określane przez macierz transakcji. Macierz transakcji jest faktycznie jednym zbiorem wspólnym dla wszystkich modułów funkcjonalnych. Przedstawiliśmy ją w nieco uproszczonej formie w poniższym zestawieniu.

KOD TRANSAKCJI MODUŁ	0100	0101	0102	0103	...
System danych finans.	X				
System analizy sprzed i zysków	X			X	
System sprawdzania sprzedaży ruchomej			X	X	
Zakupy i płatności		X			
Sterowanie i sprawozdawczość o zapasach		X			
System planowania produkcji		X	X	X	
System zatrudn. - płac.					

x/ czyli typu assembler /przyp. tłumacza/

Przypuśćmy dla przykładu, że jakaś transakcja została wprowadzona do systemu danych finansowych. Program kształtujący format wprowadzanych danych przypisuje wyróżniony numer kodu transakcji - uzależniony od jej rodzaju do czterech pierwszych bajtów "zapisu" transakcji. Skoro system danych finansowych zakończy cykl przetwarzania dla tej transakcji,bada on macierz transakcji by określić, które dalsze moduły mają przetwarzać tą samą transakcję. W wyniku tego badania przekazuje on wyciąg z zapisu transakcji do każdego z tych modułów.

NBC zdecydowało, że ponieważ każdy rodzaj transakcji jest zasadniczo różny, nie ma żadnego usprawiedliwienia dla utrzymywania pojedynczej wspólnej bazy danych dla wszystkich modułów. Zbiory wymagane przez każdy z modułów wyróżniają się swym formatem oraz w większości swą zawartością.
Jednakże przy tym podejściu do zagadnienia istnieje pewien nadmiar danych w systemie. Usiłowano zmniejszyć ten nadmiar przez tworzenie połączeń wewnątrz każdego z modułów stosując koncepcję zapisów "głównych" i "szczegółowych" / patrz tablica 1/. Zapisy transakcji są przechowywane w każdym z modułów w zbiorze "puli transakcji" jako zapisy szczegółowe powiązane łącznikami z dotyczącymi ich zbiorami zapisów głównych.

Moduł zdalnego przetwarzania, chociaż jeszcze nie jest w pełni wdrożony, pozwoli użytkownikom kierować do systemu zapytania za pomocą wstępnie uformowanych zdań pytających, które wprowadzą do akcji jeden lub kilka programów. W chwili obecnej kierownictwo NBC zajmuje się m.in. ustaleniem,jakie py-

tania będą najczęściej skierowywane do systemu.

B. System OMNIBUS w Industrial Indemnity Company
/ubezpieczenia w przemyśle/

1. Tło i informacje ogólne

Przedsiębiorstwo Industrial Indemnity of San Francisco posiada system pn. OMNIBUS, którego celem jest:

- zapewnić obsługę w zakresie przechowywania i wyszukiwania informacji bezpośrednio programom dla poszczególnych zastosowań, przy czym czas wyszukiwania ma być dostatecznie krótki, by utrzymać krótki okres oczekiwania na odpowiedzi z kilkudziesięciu zdalnych terminali.

- zmniejszyć wielkość bazy danych. Poprzedniczka bazy danych realizowana na sprzęcie II generacji, przedstawiała się w postaci dwóch zbiorów /polisy oraz żądania odszkodowania/ zajmujących w sumie 35 krążków taśmy magnetycznej; lecz nowa baza danych, która musiała sprostać znacznie zwiększonym potrzebom, miała się zmieścić na pojedynczej stacji komórek danych typu IBM 2321 /data cell drive/ o pojemności około 390 milionów bajtów.

- utrzymać wysoki stopień integracji danych, z protek-

cją wdrożoną w co najmniej trzy sposoby: wykrywanie niewłaściwych usiłowań zmiany zawartości bazy danych, zabezpieczenie przed nieprawidłową aktualizacją oraz odzyskanie informacji w wypadku awarii programu, maszyny lub zniekształcenia danych, przy minimum poświęconego na ten cel czasu i przy wysokim stopniu zaufania do odnowionej bazy danych.

pozwolić na szeroki zakres różnorodności struktur danych, zarówno przy dostępie sekwencyjnym jak i bezpośrednim na różnorodne sposoby, aby przystosować się do zastosowań w zakresie badania zbiorów, wydajnej ich aktualizacji oraz w pewnym zakresie zastosowań zapytaniowych i sprawozdawczych. Omawiany system powinien być na tyle elastyczny, by pozwolić na wdrożenie nowych wymagań w zakresie danych w przyszłości przy minimum zakłóceń zarówno w programach zastosowań eksploatacyjnych, jak i w samym systemie zarządzania bazą danych.

Kiedy przeszło dwa lata temu przedsiębiorstwo IIC stanęło przed problemem wyboru czy nabyć uogólniony system, czy opracować swój własny, wybrało tę drugą możliwość. Decyzja ta podyktowana była kilkoma czynnikami - wszystkie były jednakowo ważne. Żaden z wówczas dostępnych systemów nie zapewniał niezbędnej zwartości danych, dużej prędkości wyszukiwania lub przechowywania danych o dużym znaczeniu.

W wyniku tej sytuacji IIC opracowało szereg procedur przeszukiwania wskaźników, procedur wejść-wyjść oraz funkcji zagęszczenia i rozszerzenia danych, przemieszczania, sterowania oraz funkcji pomocniczych, które wzięte razem stały się systemem OMNIBUS.

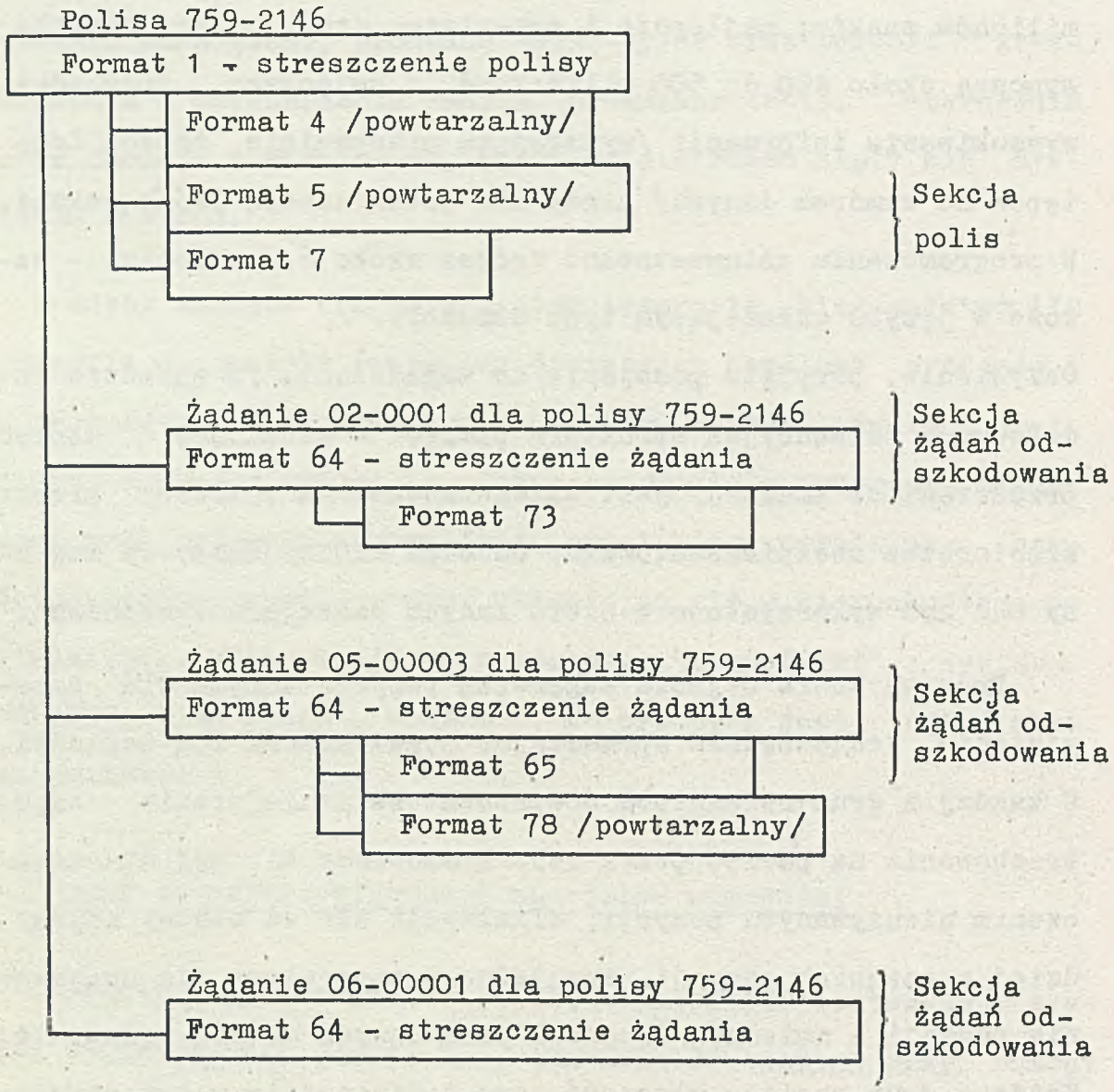
Gdyby musiano ten cały system przerobić, kierownictwo IIC ponownie dokonałoby przeglądu dostępnych handlowo systemów i w przypadku znalezienia systemu, który w przybliżeniu spełniałby zbiorowe wymagania, pokuszono by się o kompromis między zamiarami projektu, a możliwościami takiego cokolwiek bardziej uogólnionego systemu, skłaniając się w kierunku tego ostatniego. Ludzie z IIC są w zasadzie "szczęśliwi" z systemu OMNIBUS, lecz odnoszą wrażenie, że być może jest trochę zbyt skomplikowany.

2. Cechy charakterystyczne i specjalne własności

Omawiany system jest w chwili obecnej eksploatowany w trybie przetwarzania partiowego, a możliwość zapytań przy pomocy terminala znajduje się w stadium opracowania. System utrzymuje około 450.000 polis ubezpieczeniowych oraz - równą tej - liczbę żądań odszkodowania /więcej niż połowa polis nie posiada żadnych żądań odszkodowania, lecz z kolei niewielka liczba polis dla bardzo dużych instytucji gromadzi takie żądania w tysiącach./. W chwili bieżącej zapełniona jest informacją około

połowa całkowitej ilości komórek danych, co wynosi około 200 milionów znaków; najlepsze i przeciętne czasy wyszukiwania wynoszą około 400 do 500 milisekund, a najgorsze przypadki wyszukiwania informacji /wymagające maksymalnie trzech dostępów do komórek danych/ nigdy nie przekraczają dwóch sekund. W programowaniu zainwestowano ogółem około 6,5 osobolat - całość w języku składającym typu assembly.
Oczywiście, przyjęte podejście do zagadnienia /w zasadzie indeksowana sekwencyjna struktura pamięci - strukturę danych przedstawiono poniżej/ jest ukierunkowane na potrzeby przedsiębiorstwa ubezpieczeniowego, chociaż IIC wyczuwa, że mogłoby ono być wykorzystane w wielu innych rodzajach zastosowań.

Przetwarzanie wejścia zagęszcza pozycje danych dla zapamiętania, jednocześnie sprawdzając prawidłowość ich wartości. W każdej z grup osiągnięto 60-procentowe zmniejszenie zapotrzebowania na pamięć przez użycie macierzy bitowej dla oznaczenia nieużywanych pozycji, eliminacji zer od strony najbardziej znaczących pozycji, specjalnego separatora dla upakowania pozycji o zmiennej długości oraz innych metod kodowania. Każda z grup posiada własność samo-definiowania w tym sensie, że niesie ze sobą numer formatu, który może być przeglądany /przez system/ w katalogu formatów. Każdy z tych formatów opisuje zawartość i strukturę grupy danej kategorii. Formaty zanumerowane od 1 do 63 określają "sekcje polis ubezpieczeniowych", a formaty zanumerowane od 64 do 127 określają " sekcje żądań odszkodowania". Formaty 1 i 64 są specjalnymi formatami streszczającymi dla swych odpowiednich sekcji.



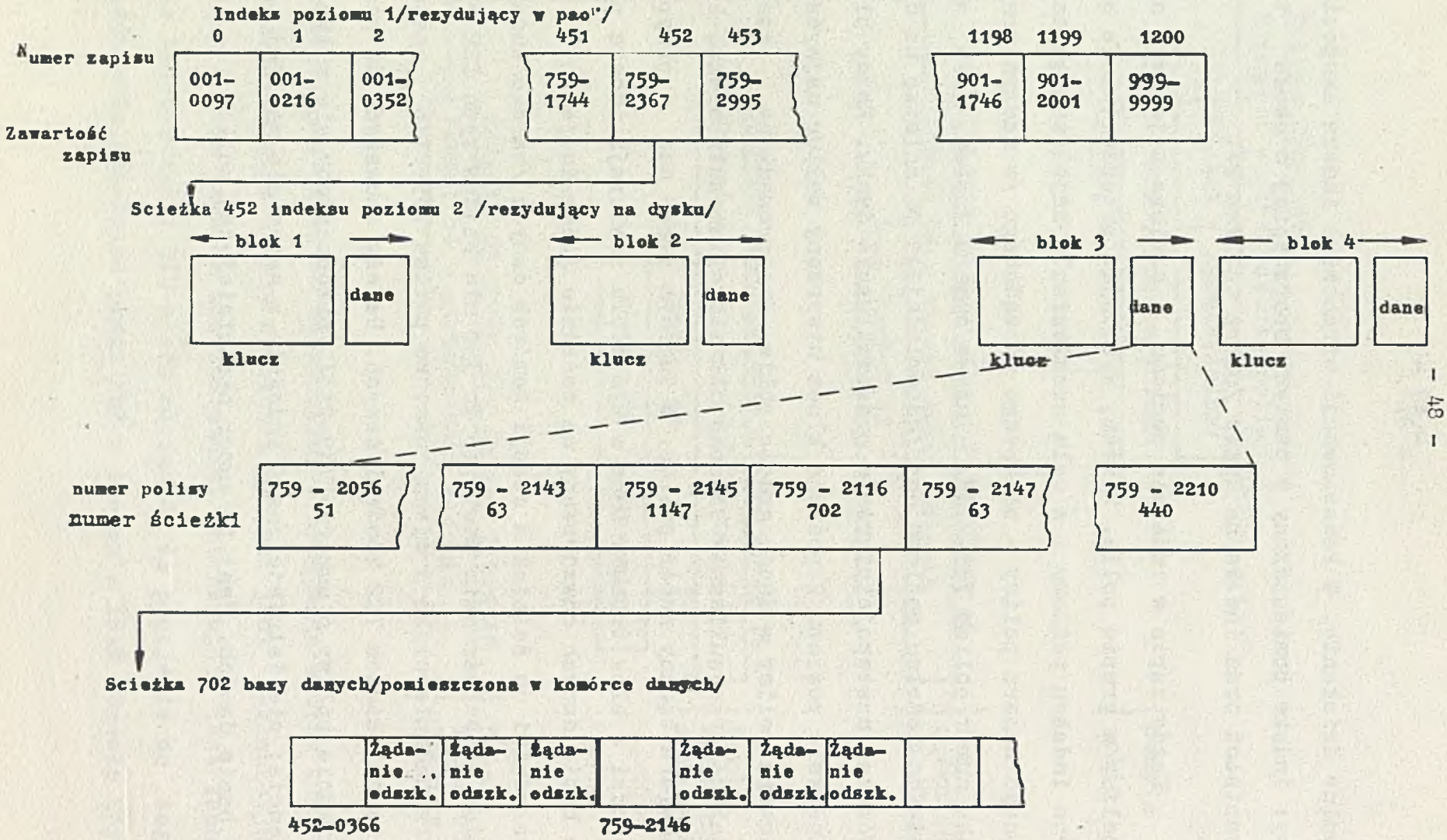
Struktura pamięci jest sekwencyjna - zbudowana według numeru polisy; wewnątrz polisy struktura pamięci jest sekwencyjna według numeru żądania odszkodowania. Interesująca jest technika indeksowania; IIC miało odczucie, że podejście do tego zagadnienia stosowane przez IBM w systemie ISAM dawało zbyt

wolne działanie. W konsekwencji opracowano indeks dwupoziomowy: indeks pomieszczony w pamięci operacyjnej o około 1.200 zapisach oraz indeks na dysku /patrz tablica 2/.

Każdy zapis w indeksie poziomu 1 zawiera najwyższy numer polisy w grupie polis. System, wykonując przeszukiwanie binarne indeksu poziomu 1 w celu odnalezienia najbliższego następnego numeru polisy - większego od żądanego /w naszym przypadku numer polisy 759-2146/ - używa numeru zapisu, jako wejścia do indeksu poziomu dwa /odpowiadającą mu ścieżkę na dysku/. Wówczas następuje pierwsze przeszukiwanie dysku. Każdy blok w indeksie poziomu 2 posiada klucz utworzony według najwyższego numeru polisy w danej grupie polis. Przeprowadzone zostaje dalsze przeszukiwanie binarne dla zlokalizowania bloku, który zawiera żądany numer polisy. W podanym przez nas przypadku chodzi o blok trzeci. Dane w tym bloku zawierają indeks polis w bazie danych odwzorowany na zasadzie jeden-do-jeden oraz wskaźniki do ścieżek w stacji komórek danych /bardzo duża pamięć o bezpośrednim dostępie do 390 mln bajtów typu IBM 2321/. Przebadanie bloku 3 ujawni nam, że polisa 759-2146 znajduje się na ścieżce 702 komórki danych. Należy zauważyć, że poszukiwanie polisy o numerze 759-2144 zakończyłoby się w tym momencie; nie istnieje ona w indeksie a zatem nie ma jej wcale w bazie danych. Jest to numer nieistniejącej polisy.

Tablica
E 61-2

INDEKS DWUPOZIOMOWY W IIC



W wyniku tego rodzaju techniki indeksowania najczęściej w celu odnalezienia jakiejkolwiek polisy wymagane jest jedno szukanie na dysku i jedno w stacji komórek danych. Wystarczy raz poszukać na dysku by określić, że dany numer nie istnieje w bazie danych. IIC przypuszcza, że "jest to znacznie szybciej, niż miałoby to miejsce w przypadku stosowania do bazy danych tej wielkości systemu opracowanego np. przez IBM pn. ISAM, który wymagałby co najmniej trzech lub czterech czasochłonnych przeszukiwań w stacji komórek danych celem ustalenia, czy jakaś polisa znajduje się w bazie danych".

Jeśli chodzi o dążenie do scalania, to system OMNIBUS ustawia w kolejkę żądania zamieszczenia, usunięcia i aktualizacji dla każdej sekcji polis ubezpieczeniowych lub sekcji żądania odszkodowania; po zagęszczeniu i kontroli prawidłowości w stosunku do katalogu formatu, żądania te są wykonywane jako jedna partia zadań obliczeniowych. Kiedy aktualizuje się jakąś sekcję, program aktualizujący ma wyłącznie do swej dyspozycji bazę danych i zapytania są wyłączone aż do momentu zakończenia aktualizacji sekcji /na dysku/. Archiwizacja jest dokonywana przez taśmy magnetyczne. W każdej chwili, kiedy należy zmienić zawartość ścieżki, zapisuje się pojedynczy blok na taśmie zawierający adres ścieżki, datę i czas zmiany oraz poprzedni i nowy obraz ścieżki. Indeks pomieszczony na dysku zostaje po prostu wypisany na taśmę przed każdym przebiegiem aktualizacyjnym.

Elastyczność systemu osiąga się przez użycie katalogu formatów. I tak, jeśli potrzebna jest nowa struktura grupowa, wówczas systemowi OMNIBUS należy zadeklarować nowy format. Żaden z programów eksploatacyjnych nie musi być zmieniony ani przekompilowany. Jeśli zaistnieje potrzeba zmiany już zawartego w katalogu formatu, to wówczas modyfikuje się jego skatalogowany opis i następnie należy przekompilować każdy program, który używa tego formatu.

C. FMS w Pacific Gas and Electric Company[6] /zbyt energii/

1. Tło i informacje ogólne

Przedsiębiorstwo Pacific Gas and Electric z siedzibą w San Francisco opracowało system zarządzania zbiorem - File Management System /FMS/ - który służy jako podwalina dla jego "banku danych scalonego systemu informacyjnego". Od tego systemu - współpracującego bezpośrednio z wieloma użytkownikami - przedsiębiorstwo wymaga, by zapewniał on:

scentralizowaną kontrolę nad definicjami danych, by zapewnić zwięzłość logiczną, wydajność oraz łatwość ich utrzymania. /Niezależność danych od indywidualnych programów - by uniknąć znacznych kosztów przeprogramowania - została osiągnięta przez oddzielenie wejścia-wyjścia, manipulacji na tablicach oraz procedur przeszukiwania od programów dla poszczególnych zastosowań.

Ochronę i zabezpieczenie zbiorów na szczeblu każdej pozycji, kontrolę nad bieżącą aktualizacją, zdolność odzysku danych po awarii systemu oraz wydajne wykorzystanie przestrzeni pamięci zajętej przez zbiór. Zestawia się statystyki wykorzystania dla każdej pozycji.

Zaangażowana przez przedsiębiorstwo PG&E firma doradcza w lipcu 1969 doszła do wniosku, że żaden z handlowo dostępnych uogólnionych systemów zarządzania danymi nie zadowoli jego wymagań. Zanim zaprojektowano system FMS, przeprowadzono podobny przegląd, który dał identyczny wynik. Przegląd ten przeprowadzono z powodu szybkiego rozpowszechniania się tego rodzaju systemów; PG&E chciało potwierdzenia słuszności swojej decyzji budowy takiego systemu własnymi siłami. Największe opory przed użyciem gotowych pakietów z przejrzanych systemów wynikły z tego, iż w chwili obecnej nie mogą one spełnić następujących wymagań:

- zagęszczenia danych i wymagań w zakresie posługiwania się tablicami;
- umożliwienia użytkownikom specyfikacji struktury zbiorów dla poszczególnych podsystemów /zastosowań/ - wielorakich baz danych;
- wymagań w zakresie rozległych i zróżnicowanych płaszczyzn styku w przetwarzaniu zdalnym;

. potrzeb uwzględnienia adaptacji istniejących systemów.

Z pewnością, każdy z pakietów posiada jedną lub dwie z wymienionych własności, lecz żaden z nich nie realizował wszystkich.

2. Cechy charakterystyczne i specjalne własności

Wydajne wykorzystanie pojemności pamięci można uzyskać stosując zagęszczanie danych i w ten sposób poważnie redukując wymagania w stosunku do potrzebnej pojemności pamięci, tak jak to pokazano poniżej.

Sprzęt IBM	Stacja dyskowa 2311	Szereg dysków 2314	Komórki danych 2321
Liczba wymaganych jednostek dla bazy danych o 5 mld znakach /netto/	1.000	30	17
Dzierżawa miesięczna	$600.000	$180.000	$60.000
Czas ładowania sekwencyjnego	62 h	15 h	170 h

Powyższy wykaz usprawiedliwia wybór jednostek 2314 jako środek przechowywania bazy danych gdyż uznano, że jednostka typu 2321 będzie za powolna. PG&E zamówiło dwadzieścia jednostek typu 2314 dla całego zestawu wszystkich zastosowań i może się jeszcze okazać, iż jest to za mało. Poniżej podany wykaz przedstawia pojedyncze zastosowanie - zbiór główny klientów. Widoczna tutaj jest opłacalność zagęszczania danych.

Przyjmując liczbę konsumentów na 3,5 miliona, 75% pojemności - czyli 175 milionów bajtów na jednostkę 2314/pozostałość to: indeksowanie, tablice itd./ - 30 minut na załadowanie każdego dysku oraz dzierżawę w wysokości około $ 6.000 miesięcznie dla każdej jednostki 2314, otrzymujemy:

Typ zapisu	Przec. liczba bajtów na zapis	Potrzebna liczba mld bajtów pamięci	Liczba jednostek 2314	Sekwencyjny czas załad.	Dzierżawa mies. tys. $
Stałej długości bez zagęszczania	4.500	15,8	90	45 h	540
Zmiennej długości, bez zagęszczania	700	2,5	15	7,5 h	90
Zmiennej długości, z zagęszczaniem	400	1,4	8	4 h	48

System FMS stopniowo wzrastał w ciągu ostatnich dwóch lat. Do chwili obecnej zainwestowano w niego blisko 10 osobolat prac w zakresie oprogramowania. Skoro tylko powstają nowe wymagania w przedsiębiorstwie szczególnie w zakresie sprzętu dla zdalnego przetwarzania /nabycie różnych terminali, nowe zastosowania itd./ system modyfikuje się w celu przystosowania go do nich. Czas odpowiedzi na bezpośrednie zapytania wynosi przeciętnie około siedmiu sekund; po godzinach system jest eksploatowany w trybie przetwarzania partiowego.

Wzajemne oddziaływanie systemu i użytkownika odbywa się za pomocą jednego z trzech poziomów języków zapytaniowych. Poziom najprostszy, do użycia przez urzędników i innych minimalnie przeszkolony personel, reprezentuje sztywno zdefiniowany język tabelaryczny o małej elastyczności. Poziom pośredni reprezentuje bardziej uogólniony język zapytaniowy, w którym pytania są zdefiniowane wstępnie w swej formie, lecz nie w swej zawartości. Ostatni z poziomów przedstawia język zapytaniowy skonstruowany dla największej elastyczności w stosunku do użytkownika, z zestawem rozkazów lub słów, które mogą być łączone na różny sposób i w różnej sekwencji.

Użytkownik buduje strukturę swoich danych według własnego życzenia; dostępne są manipulacje na poziomie pozycji, grupy oraz na poziomie wpisu. Struktura pamięci składa się z pierścieni wewnątrz zapisów, przy czym zapisy są przechowywane niesekwencyjnie z poszukiwaniem za pomocą tablicy/przypadkowa

struktura zapisów/. System składa się z trzech podstawowych modułów: program zarządzający sterowaniem danymi ustawia w kolejkę wszystkie zadania w zakresie wejścia-wyjścia, zarządza pulami buforów We/Wy, otwiera i zamyka wszystkie bezpośrednio współpracujące z systemem zbiory oraz obsługuje skorowidze zbiorów i tablic. Program zarządzający pojemnością pamięci masowej tłumaczy /interpretuje/ zawartości bazy danych programom z zakresu zastosowań, rozgęszcza i zagęszcza dane w czasie wyszukiwania i przechowywania oraz konstruuje różne struktury pamięci; program zarządzający tablicami dynamicznymi ładuje, usuwa, przeszukuje i obsługuje centralny zespół tablic /katalogów/ dla wszystkich programów zastosowań.

W pozycji centralnej w stosunku do tych trzech modułów oraz w stosunku do całego systemu, znajduje się tablica opisująca strukturę zapisów. Zawierając jeden wpis dla każdej pozycji w bazie danych służy ona jako płaszczyzna styku dla wszystkich działalności związanych z utrzymaniem i przetwarzaniem zbiorów. Każdy wpis składa się z informacji sterującej oraz ze "współrzędnych" danych. Informacja sterująca zawiera statystykę wykorzystania, kod zabezpieczenia /przed niepowołanym dostępem/, format i organizację pozycji, umiejscowienie w strukturze zapisu oraz procedurę potrzebną do zgęszczenia względnie rozgęszczenia pozycji. Współrzędne danych specyfikują długość i umiejscowienie pozycji przed i

po rozgęszczeniu; opisują one pozycję tak, jak ją widzi struktura danych oraz tak, jak ją widzi struktura pamięci.

PG&E przeznaczyło stały zespół ludzki do obsługi i rozwoju swojego systemu. Przedsiębiorstwo uważa, iż jest to praca o charakterze ciągłym szczególnie dlatego, że jego 3,5 milionowy zbiór zapisów głównych jest w stanie ciągłej płynności. Uważa się, że poprzez stosowanie tablic system pozostanie dość elastyczny dla przyszłych wymagań ze strony zastosowań.

D. DATA FILE TWO w International Minerals and Chemical Corporation[7] /przemysł chemiczny i wydobywczy w zakresie minerałów/

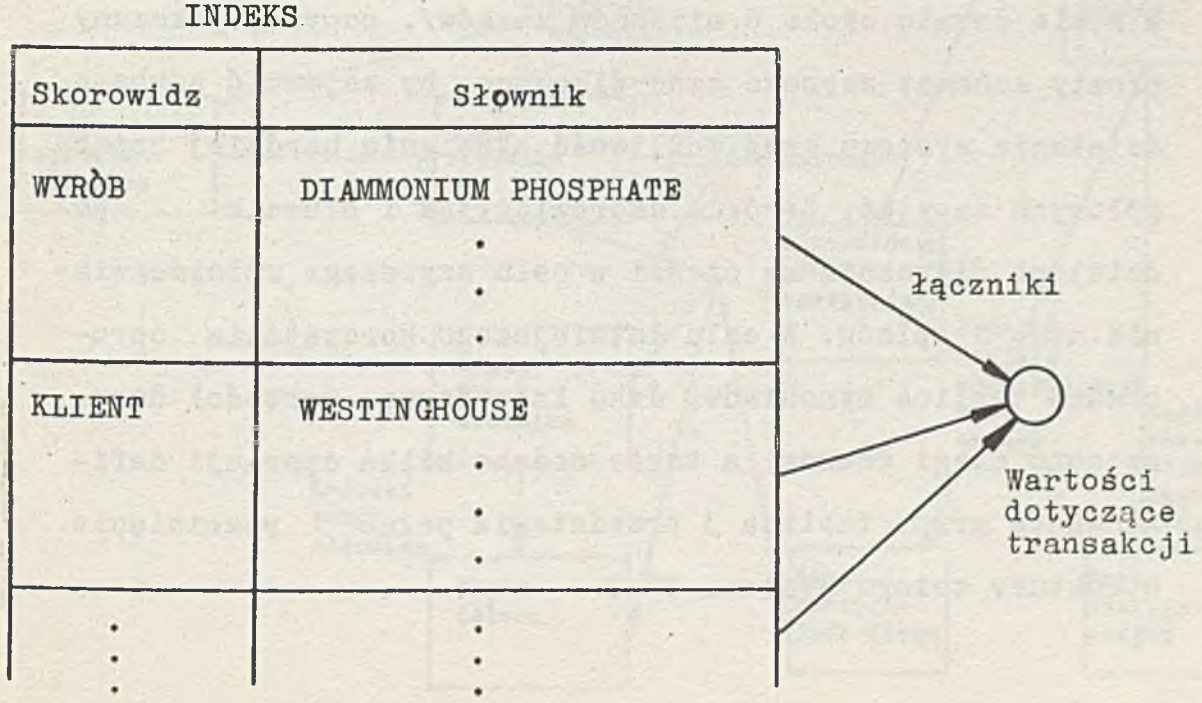
1. Tło i informacje ogólne

System DF-2 /Data File Two/ zaprojektowano w International Minerals /Skokie, Illinois/, lecz wdrożono jako doświadczalny system prototypowy jedynie w General Electric. System DF-2 przedstawia sobą zorientowany na użytkownika system przechowywania i wyszukiwania informacji, zbudowany na podwalinach zapewnionych przez system pn. Integrated Data Store opracowany w General Electric /IDS/. System IDS jest zbiorem rozkazów manipulowania na bazie danych i odpowiadających im procedur kompilacyjnych i wykonawczych, które służą jako rozszerzenie języka programowania COBOL.

Baza danych DF-2 była początkowo zaprojektowana dla przechowywania miesięcznych danych o sprzedaży za okres ostatnich dwóch lat, zbioru głównego klientów, bieżących miesięcznych zamówień i sprzedaży, umiejscowienia wagonów, planu zysków dla bieżącego roku, kartotek personalnych, księgowości finansowej oraz danych na temat jednorocznego i pięcioletniego planu zysków; łącznie miało to wynosić około 400.000 pozycji.

2. Cechy charakterystyczne i specjalne własności

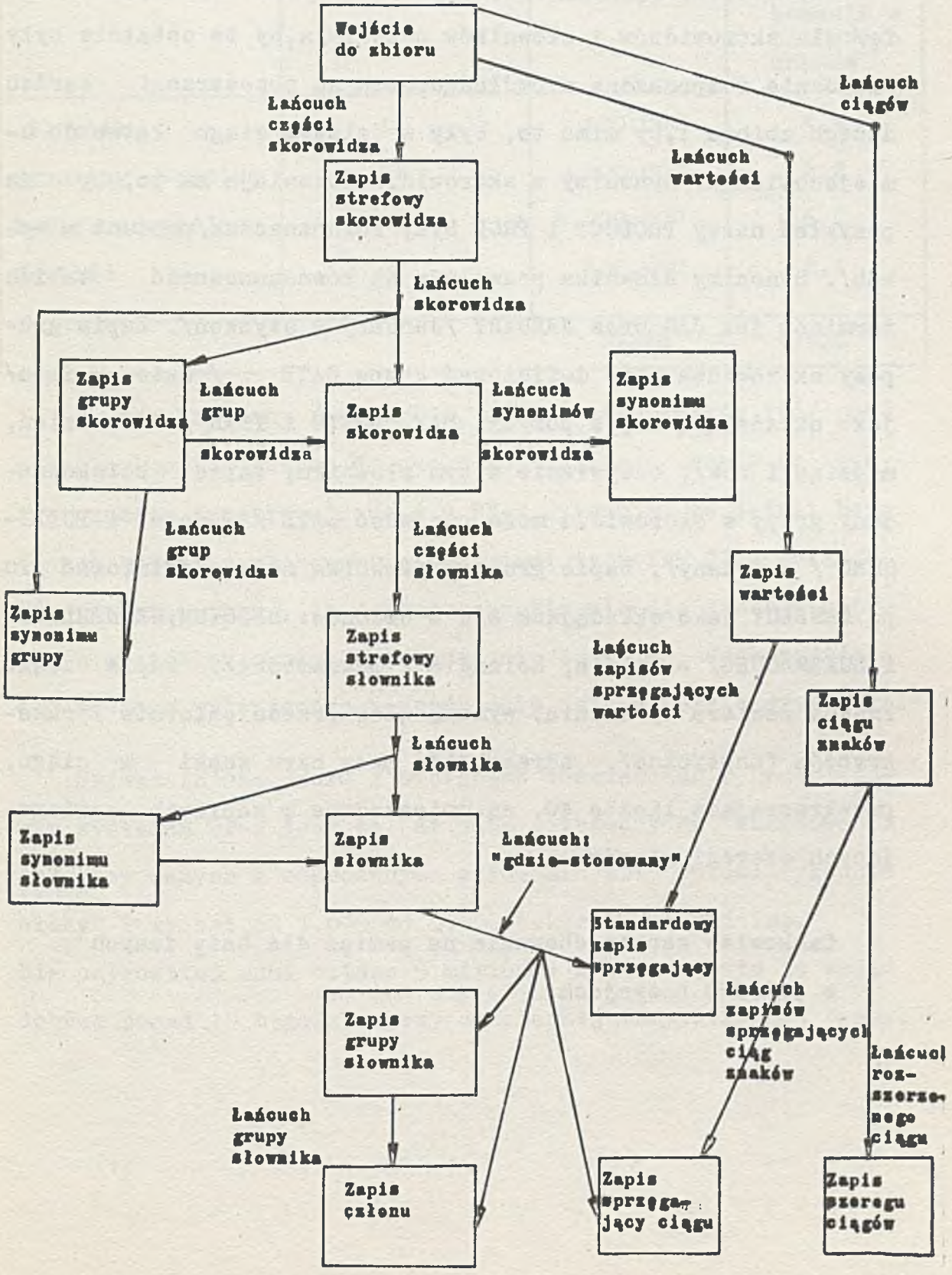
Chociaż system DF-2 nie był nigdy używany w środowisku "eksploatacyjnym", dostarczył on kilku raczej interesujących doświadczeń. Ponieważ system IDS jest zorientowany na struktury listujące, DF-2 posiada strukturę przechowywania w postaci listy odwróconej, która w prostej formie wygląda następująco:



Indeks składa się z dwóch list: skorowidza i słownika. Zapisy skorowidza zawierają nazwy kluczy, według których podzielono słownik, takie jak WYRÓB, KLIENT, WYSŁAĆ Z, PRZE AĆ DO, NAZWISKO, DATA itd. Są one rozprzestrzenione w całym zbiorze w celu ułatwienia fizycznego ich wyszukania. Zapisy słownika zawierają wartości kluczy / szczególnie deskryptory - czyli słowa opisujące/, takie jak DIAMMONIUM PHOSPHATE, WESTINGHOUSE oraz JANUARY /, czyli: nazwa artykułu, nazwa klienta i nazwa miesiąca - styczeń/. Są one umieszczone w pobliżu zapisów skorowidza.Zapisy opisujące wartość zawierają tylko dane numeryczne, umiejscowione w regularnych odcinkach wzdłuż zbioru.Łączniki są zebrane w listę zapisów sprzęgających umieszczoną blisko zapisów opisujących wartość.

Ponieważ teoretyczna baza danych, dla której przewidywano system DF-2, była bardzo duża /400.000 pozycji, co w sumie dawało około 8 milionów znaków/, powyżej pokazany prosty schemat szeroko zmodyfikowano, by zapewnić szybsze działanie systemu oraz możliwość stawiania bardziej szczegółowych zapytań. Zarówno skorowidz,jak i słownik podzielono fizycznie na części w celu szybszego umiejscawiania nazw i opisów. W celu łatwiejszego korzystania opracowano tablice synonimów. Jako inną formę wartości dopuszczono ciągi znaków, a także dodano kilka operacji definiowania grup. Tablica 3 przedstawia pełne rozwinięcie struktury zbioru systemu DF-2.

Tablica
E 61-3 STRUKTURA ZBIORU SYSTEMU DF-2 W PAMIĘCI



Jak to pokazuje tablica, zapisy strefowe /opisujące strefę/ dla skorowidzów i słowników pozwalają,by te ostatnie były fizycznie rozproszone w podłańcuchach na przestrzeni bardzo dużego zbioru i,by mimo to, były w dalszym ciągu łatwe do umiejscowienia. Synonimy w skorowidzu pozwalają na to, by na przykład nazwy PRODUCT i PROD były równoznaczne/product = wyrób/. Synonimy słownika pozwalają na równoznaczność takich terminów jak JAN oraz JANUARY /January = styczeń/. Zapis grupowy skorowidza może definiować grupę DATE / date = data/ jako składającą się z pozycji DAY, MONTH i YEAR / = dzień, miesiąc i rok/, oczywiście w tym słowniku; zapis bliskoznaczny grupy w skorowidzu może porównać DATE / = data/ i PUBLISHED / = wydany/. Zapis grupowy słownika może zdefiniować grupę BENELUX jako składającą się z członów: BELGIUM,NETHERLANDS i LUXEMBOURG/ = Belgia, Holandia i Luksemburg/. Zapis ciągu znaków zawiera wyrażenia, wymowę nazw przedsiębiorstw /transkrypcja fonetyczna/, adresy itd, przy czym znaki w ciągu, przekraczające liczbę 40, są umieszczone w zapisach zawierających szeregi ciągów.

Całkowite zapotrzebowanie na pamięć dla bazy danych o 400.000 pozycjach:

Rodzaj zapisu	Przeciętna liczba znaków	Liczba rekordów	Całkowite zapotrzebowanie w milionach znaków
Słownik	44	100.000	4,4
Synonimy słownika	31	200.000	6,2
Zapis sprzęgający	17	8.000.000	136,0
Wartość	21	400.000	8,4
		RAZEM	155,0

Dalsze łańcuchy /skorowidz, grupy, ciąg znaków itd/ podnoszą całkowitą liczbę do około 160 milionów znaków.Obliczenia czasowe przeprowadzone dla DF-2 wykazują, że jeżeli baza danych byłaby przechowywana w pamięci typu IBM 2314 /dla celów porównawczych/, to czas przeszukiwania dla jednego zapytania wyniósłby około 20 sekund; nie jest to doskonałością dla systemu procującego bezpośrednio, ale na razie wystarcza.

Najważniejszą nauką z bieżących doświadczeń z prototypowym systemem DF-2 jest to, że czas potrzebny do załadowania tej bazy danych z odwróconymi zbiorami jest "niewiarygodnie słaby" i wynosi od 1.000 do 5.000 rekordów na godzinę.
Dla całkowitej sumy blisko 9 milionów rekordów może to zająć dobrze ponad 10 tygodni, przy codziennej trzyzmianowej pracy.

E. System MARK IV w kilku korporacjach

1. Tło i informacje ogólne

W tym wypadku chodzi o zakres zastosowania tego pakietu zbadanego w kilku dużych przedsiębiorstwach, włączając w to Ford Motor Company, Johnson and Johnson, Standard Oil of Indiana oraz Mobil Oil Corporation. Typowe wykorzystanie systemu MARK IV różni się nieco wśród użytkowników, lecz jest na tyle podobne, by usprawiedliwić zebranie faktów do wspólnego mianownika i przedstawienie ich jako pojedynczego złożonego przypadku.

Dyrekcja przedsiębiorstwa nabyła około rok temu MARK IV, uogólniony system zarządz nia zbiorami, opracowany przez przedsiębiorstwo Informatics. W korzystaniu z systemu przeszkolono kilka setek personelu - nie tylko samych programistów. Przedsiębiorstwo już posiadało mnóstwo oddzielnych zbiorów danych w głównym ośrodku obliczeniowym, włączając w to między innymi dane o zbycie i zamówieniach na wyroby, dane o klientach i personelu sprzedaży; dane o dystrybucji oraz dane o ruchu frachtów, przeznaczaniu i środkach transportu; a także dane o produkcji i o zapisach.

Chociaż przedsiębiorstwo w końcowym wyniku chce posiadać bezpośrednio dostępną zintegrowaną bazę danych, to jednak uważa, że w chwili obecnej istnieje za dużo nierozwiązanych problemów w tym zakresie. Istnieją trudności w zde-

finiowaniu elementów danych faktycznie potrzebnych w bazie danych, w określeniu optymalnej struktury zbiorów w świetle częściowo nieznanych wymagań ze strony zastosowań,w zaprojektowaniu systemu, który musi działać na różnorodnym sprzęcie i prawdopodobnie poprzez rozległe obszary geograficzne /sieć baz danych/ oraz w rozwiązaniu bolączek wielorakich użytkowników w zakresie zabezpieczenia danych /przed utratą, nieupoważnionym dostępem/. Jednocześnie należy przedsięwziąć pośrednie środki w celu zaspokojenia bieżących potrzeb kierownictwa w zakresie sprawozdawczości, jak również trzeba podjąć się eksperymentów służących sprostaniu jutrzejszym potrzebom.

2. Cechy charakterystyczne i specjalne własności

Wiele wydziałów u użytkowników początkowo stosowało MARK IV do wypróbowania różnych formatów sprawozdań; w niektórych przypadkach zdarzało się, że pisano następnie program w języku COBOL /lub jakimś innym języku/ używając go w charakterze programu produkcyjnego do wytwarzania tych sprawozdań w trybie regularnym. Sam system MARK IV nie jest używany do celów produkcyjnych,ponieważ jego wadą jest niska wydajność. Użytkownicy zgłaszają, że programy MARK IV mają dwa do czterokrotnie dłuższe przebiegi na centralnym procesorze niż odpowiadające im programy w języku COBOL.

System jest jedynie używany w operacjach partiowych, gdyż nie może on pracować w trybie bezpośrednim /programiści z przedsiębiorstwa Forda opracowali prostą, ale użyteczną metodę dla sprawozdawczości "pseudo-bezpośredniej": użytkownik siedzący przed terminalem wyposażonym w monitor ekranowy może oddziaływać na fordowski system zapytaniowy pn. DISPLAYALL do chwili aż określenia, jakiej żąda informacji; wówczas formułuje on zadanie obliczeniowe dla systemu MARK IV - system ten później wykonuje wydruk sprawozdania na papierze w trybie pośrednim /off-line/.

Do celów operowania danymi z systemu MARK IV korzystano niewiele - w sensie tworzenia i obsługi zbiorów. Jest on używany w pierwszym rzędzie do celów wyszukiwania informacji oraz jako "szybki i zgrubny" generator wydruków sprawozdawczych. Z powodu tych jego cech uzyskano korzyści w formie stosunkowo łatwego i szybkiego produkowania jednorazowych sprawozdań. Na przykład, nagle był potrzebny specjalny raport na najbliższe posiedzenie naczelnego kierownictwa; raport ten wykonano - używając MARK IV - w ciągu dwóch tygodni. Programiści oświadczają, że zajęłoby blisko dwa miesiące napisanie i przetestowanie programu w COBOL'u, by wygenerować taki raport. Byłoby to już za późno.

W chwili obecnej MARK IV działa tylko na sprzęcie IBM 360, jest on dopasowany do systemu operacyjnego tych maszyn i wykorzystuje ich metody dostępu do danych - "operowanie danymi".

Wszystkie rozkazy i zapytania użytkownika są przesyłane do systemu za pomocą uprzednio zaprojektowanych specjalnych form nie wymagających, przynajmniej w teorii, żadnego doświadczenia w programowaniu. Po wczytaniu i zweryfikowaniu przez MARK IV wejścia od użytkownika, wybiera on odpowiednie moduły programów i podprogramów ze swojej biblioteki, generuje kompletny program, mający spełnić wyspecyfikowane funkcje, wykonuje ten program i następnie produkuje żądane raporty lub uaktualnione zbiory. Wygenerowany program może być skatalogowany i przechowany do późniejszego użycia.

Szczegółowy opis cech i możliwości systemu MARK IV można znaleźć w publikacjach wyszczególnionych w punktach 7, 8b lub 18 spisu bibliografii, chociaż mało jest tam informacji dotyczących faktycznych pomiarów wydajności - to stwierdzenie jest prawdziwe dla wszystkich uogólnionych systemów obecnie dostępnych na rynku.

F. System GIM w korporacji BCD

1. Tło i informacje ogólne

Korporacja BCD, podana w niniejszym przykładzie, w rzeczywistości nie istnieje;lecz system GIM / Generalized Information Management = uogólnione zarządzanie informacją/ istnieje. Opracowało go przedsiębiorstwo TRW Systems dla potrzeb własnego zestawu zastosowań i okazał się on dosta-

tecznie użyteczny i elastyczny tak, że po pewnych modyfikacjach, rzucono go na rynek. Prawnie zastrzeżony charakter systemu zabrania przedstawienia faktycznie istniejącego przykładu. Sam system jednakże posiada kilka istotnych cech charakterystycznych, które mu zapewniają zamieszczenie w tym rozdziale, tym bardziej, że na jego temat ukazało się mało publikacji na forum publicznym.

Wydział produkcyjny w BCD nabył system GIM niemal rok temu na trzyletnią dzierżawę, której koszt, jak się mówi,ma wynieść w przybliżeniu $ 100.000. Wydział przeprowadził przegląd uogólnionych systemów i stwierdził, że z dostępnych systemów jedynie GIM będzie spełniał postawione wymagania. Wymagania te przedstawiają się następująco:

- zbieranie w jednej zintegrowanej bazie danych całej informacji niezbędnej dla wspomagania zautomatyzowanego systemu sterowania produkcją i zapasami, przy szybkim wyszukiwaniu i uaktualnianiu informacji z wielu bezpośrednio podłączonych zdalnych terminali,

- posiadanie systemu zarządzania bazą danych zdolnego do przyswojenia wielorakich baz danych w celu przyszłego rozszerzania na inne dziedziny funkcjonalne przedsiębiorstwa i uzyskania w ostatecznym wyniku scalonego systemu informacji kierownictwa, bez względu na to jakiby on się okazał,

posiadanie na tyle elastycznego systemu, by mógł on łatwo adaptować zmiany w wymaganiach ze strony zastosowań /być może stwarzające konieczność reorganizacji istniejącej struktury danych/, jak również potencjalne zmiany w konfiguracji sprzętu /to jest, system powinien być jak najbardziej niezależny od typu maszyny/.

Po nabyciu systemu GIM personel systemowy w BCD spędził w sumie pięć miesięcy /około trzech tygodni na nauczenie się jak używać system i trochę ponad cztery miesiące na zaprojektowanie i przeprojektowanie bazy danych/ nad pełnym wdrożeniem systemu, inwestując w sumie w tę działalność 15 osobomiesięcy. W chwili obecnej jedna osoba jest odpowiedzialna za obsługę bazy danych dotyczących produkcji i sterowania zapasami, a dwie osoby przeznaczone do skonstruowania dalszych niezależnych baz danych. Dyrektor przetwarzania danych przedstawił sprawę następująco:"to co straciliśmy w wykorzystaniu zasobów maszyny przez przejście do uogólnionego systemu /nie wydaje mi się, by to było dużo/ zostało z nawiązką skompensowane przez bardzo szybkie wdrożenie i otrzymanie w krótkim czasie produktywnego systemu w eksploatacji". Narzeka on tylko, że "nie zapewniono sekwencyjnego przetwarzania zbiorów. Chociaż przedsiębiorstwo TRW System bada obecnie pseudosekwencyjną strukturę pamięci, między innymi w celu otworzenia możliwości dołączenia tej cechy w przyszłości do GIM, w chwili obecnej musimy wypisy-

wać zbiór na taśmę magnetyczną i sortować go, by wykonać przetwarzanie sekwencyjne. Zabawne było,gdy odkryliśmy, że przetwarzanie sekwencyjne nie jest tak ważne jak myśleliśmy".

2. Cechy charakterystyczne i specjalne własności

System GIM wdrożono na komputerach Systemu/360 IBM, Modele 30, 40, 50 i 65, przy czym przetwarzanie zdalne funkcjonuje na wszystkich modelach z wyjątkiem Modelu 30. System GIM jest dopasowany do systemów operacyjnych IBM: DOS lub OS, i wykorzystuje metody dostępu z systemu operacyjnego takie, jak BTAM i QTAM /oczywiście IBM/, lecz może także wykorzystywać swoje własne procedury dostępu napisane w języku maszynowym. GIM jest napisany w dwóch językach: w makrojęzyku niskiego poziomu i w języku składającym[x] /typu assembly/ zależnym od systemu operacyjnego. Ten ostatni język jest zwany pakietem styku i musi być wymieniony z chwilą przeniesienia systemu na sprzęt innego producenta. Pierwszy z języków tworzy ogromną większość zawartości GIM i mógłby być ponownie składany bez modyfikacji przez jakikolwiek makrojęzyk składający /makro-assembler/ po zdefiniowaniu nowych makrorozkazów.

Struktura pamięci w GIM jest swobodna, przy czym adresy zapisów są wyliczane przez algorytm przy uwzględnieniu artystycznych cech bazy danych. Teoretycznie nie istnieje granica

---

[x] lub montującym.

wielkości zbioru; system jest ograniczony jedynie przez wielkości fizyczne. Słownik systemu opisuje każdy zbiór i jego korelacje /patrz niżej/ z innymi zbiorami. Zawiera on także elementy języka użytkownika /swobodna forma języka angielskiego z pewnymi ograniczeniami/. Dodatkowo, oprócz zapewnienia możliwości tworzenia synonimów, słownik umożliwia zmianę opisów zbiorów bez wpływania zarówno na oprogramowanie GIM, jak i na programy użytkownika.

Jedną z najbardziej interesujących osobliwości systemu GIM jest to, co w TRW Systems jest zwane "współzależności" /correlatives/. Jeśli użytkownik opisuje swoje zbiory dla systemu GIM, może on również określić pewne powiązania danych albo ich zależności. Ustala on współzależności, by utrzymać te powiązania w bazie danych. Pozycje w zbiorze mogą być zdefiniowane w ten sposób, że wchodzące do nich wartości danych będą automatycznie użyte do wykonania funkcji arytmetycznych, pośredniej aktualizacji innych zbiorów lub do wpływania na inne pozycje w tym samym zbiorze. Zbiór może być także zdefiniowany w taki sposób, że zależności typu nadrzędny-szczegółowy mogą być wyszukane automatycznie. Stąd współzależności są faktycznie instrukcjami proceduralnymi, zdefiniowanymi przez użytkownika w celu ustanowienia połączeń lub skojarzonych zależności wśród danych. Rezydujące w słowniku współzależności są przywoływane automatycznie skoro tylko dokonuje się dostępu do powiązanych zbiorów, zapisów lub pozycji. Wśród współ-

zależności, które może określić użytkownik, znajdują się następujące:

- automatyczne wyszukiwanie powiązanych danych przechowywanych w innych zbiorach,
- automatyczna aktualizacja powiązanych danych w innych zbiorach,
- automatyczna aktualizacja zbiorów indeksowanych poprzecznie /wskaźniki dwustronne/,
- automatyczne obliczenie wartości danych jako funkcji innych przechowywanych wartości,
- automatyczne łączenie pozycji w sekwencje posiadające pewien związek,

Czas odpowiedzi oraz czas aktualizacji są przede wszystkim zależne od złożoności bazy dąnych /od liczby współzależności i powiązanych zbiorów/ oraz od wielkości bazy danych jeśli jest wymagane pełne przeszukiwanie szeregowe w przypadku, gdy jakaś pozycja nie została zdefiniowana, gdyż użyto dla niej klucza o wartości jeden. Takie nieprzewidziane wykorzystanie nie występuje często, ale czasem się zdarza. Zapytanie angażujące, powiedzmy, cztery powiązane zbiory, zajmuje około dwóch sekund dla pełnego wyszukania informacji, nie licząc czasu potrzebnego do wydrukowania względnie wyświetlenia odpowiedzi. Czas dla aktualizacji angażującej trzy

powiązane zbiory, odczytanie i wpisanie nowej zawartości do objętych nią pozycji, włączając w to wykonanie współzależności - wynosi 4,8 sekundy.

Ostatnią cechą godną zainteresowania jest zagęszczanie danych. Użytkownik może zastosować funkcje wydawnicze do danych wprowadzanych w celu odrzucenia znaków nie posiadających żadnego znaczenia. System GIM stosuje także rekordy o zmiennej długości, dopuszczają odstępy i odmiany w danych.

G. Streszczenie i ocena przebadanych przykładów

Poniżej podano w streszczeniu najbardziej interesujące i posiadające największe znaczenie aspekty podanych powyżej przykładów:

1. System informacji kierownictwa w NBC

Jest to system informacji kierownictwa na szczeblu korporacji stosujący wielorakie niezależne bazy danych z automatycznym przetwarzaniem transakcji. System jest przystosowany do dziennego cyklu działalności gospodarczej przedsiębiorstwa. Wdrożenie oprogramowania systemu będzie kosztować około $ 1,5 miliona. Zdolności zarządzania danymi są w pełni przystosowane do systemu informacji kierownictwa i wcielone do niego tworząc system, który jest do-

pasowany do tego właśnie przedsiębiorstwa i jego szczególnego sposobu prowadzenia działalności gospodarczej. Środowisko zastosowań musiało być bardzo jasno zdefiniowane; w wyniku tego powstał system, który jest wyspecjalizowany i nie przejawia elastyczności.

2. System OMNIBUS w IIC

Jest to system przeznaczony do eksploatowania jednego zbioru /zarządzanie danymi dotyczącymi polis ubezpieczeniowych i żądań odszkodowania/. Wymaga on zagęszczania danych, dużych szybkości wyszukiwania i archiwizacji danych ważnych dla przedsiębiorstwa. Zastosowano specjalne i zawiłe indeksowanie z samodefiniowaniem danych za pomocą katalogu formatów danych. Wdrożenie oprogramowania systemu będzie kosztować blisko $ 200.000. Dane w zbiorze są dobrze zdefiniowane i tu znów powstający system jest tak dalece specjalizowany, że nie może być użyty w innych zastosowaniach.

3. System FMS w PG&E

Jest to system informacji kierownictwa na szczeblu korporacji z podsystemem zarządzania zbiorami, stosującym wielorakie zintegrowane bazy danych, wymagające zagęszczania danych, względnie dużych szybkości wyszukiwania oraz zdolności operowania tablicami /by uzyskać integrację/. Wdrożenie oprogramowania systemu kosztuje około $ 200.000, i wzro-

śnie w miarę wzrostu systemu. Dane w bazach danych nie są tak dobrze zdefiniowane, co w wyniku daje system bardziej uogólniony i elastyczny; do umieszczania nowych struktur i formatów, aby dodać je do systemu, używa się tablic definiowania danych.

4. System DF-2 w IMSCC

System ten był planowany jako system przechowywania i wyszukiwania informacji dla kierownictwa na szczeblu korporacji. System stosuje odwrócone struktury listujące. Wdrożono go eksperymentalnie poprzez IDS przy nieznanym koszcie. Wypracowane struktury słowników pozwalają ku wygodzie użytkowników na stawianie zapytań w języku bardzo podobnym do gramatycznego języka angielskiego; zapewniają one także zdolności tworzenia synonimów, które mogą - między innymi - przystosowywać istniejące programy zastosowań. Czas ładowania zbiorów jest wygórowany, a system nie jest dobrze dostosowany do szybkiej odpowiedzi w pracy bezpośredniej /z terminali/. Bardzo wypracowane mechanizmy łączenia, nieodłączne dla tego typu struktury, także stwarzają sporo trudności w oszczędzaniu pojemności pamięci oraz w obsłudze zbiorów, chociaż to właśnie one pozwalają na ustanowienie wielorakich zależności i powiązań wśród elementów danych.

5. System Mark IV stosowany w kilku przedsiębiorstwach

Jest to uogólniony system zarządzania zbiorami na szczeblu korporacji /i niżej/. Używano go początkowo do wyszukiwania produkcji rzadko potrzebnych raportów oraz do eksperymentowania w zakresie formatu dla często potrzebnych raportów. Całkowity początkowy koszt wyniósł $ 30.000, przy czym stracono niewiele czasu między zakupem a wdrożeniem. System ten działa dla jakiejkolwiek struktury zbioru, która jest rozpoznawalna przez system operacyjny komputera i dlatego jest użyteczny dla niemal każdego zastosowania. Język użytkownika pozwala na stosowanie szerokiej różnorodności i mieszaniny podstawowych rozkazów lub żądań, przy czym użytkownik nie potrzebuje być zaznajomiony z technikami organizowania zbiorów. Kosztem takiego uogólnienia jest spadek wydajności w eksploatacji wygenerowanych programów.

6. System GIM w kilku bliżej nieokreślonych instytucjach BCD

Jest to uogólniony system operowania danymi dopuszczający wielorakie bazy danych, zagęszczanie danych, automatyczne przetwarzanie transakcji oraz zapewniający niezłe szybkości wyszukiwania informacji. Całkowite zobowiązania finansowe tytułem trzyletniej dzierżawy wynoszą nieco ponad $ 100.000, włączając w to przeszkolenie, przy czym opóźnienie między nabyciem a wdrożeniem jest minimalne. System jest

gotów do adaptacji w przypadku zmian w zastosowaniach lub w strukturach danych i celowo został zaprojektowany jako pakiet modularny, w dużym stopniu przenośny. Językiem użytkownika jest swobodny w formie język angielski z umiarkowanymi ograniczeniami; ponieważ sam język jest przechowywany w słowniku systemu na równi ze zdefiniowanymi przez użytkownika zbiorami, synonimami oraz z funkcjami współzależności. Jakikolwiek inny język może być podstawiony jako zastępczy na płaszczyźnie styku z użytkownikiem.

W czterech pierwszych przypadkach przedsiębiorstwa zdecydowały się na zbudowanie swoich systemów własnymi środkami, ale w różnym stopniu. Trzy z tych czterech wewnętrznych systemów było wdrożonych na poziomie języka składającego, w ten sposób ograniczając je do danej generacji sprzętu producenta; czwarty system był wdrożony na poziomie COBOL'u, gdyż język ten jest nieco bardziej przenośny między systemami. Dwa z tych systemów wykorzystują procedury dostępu i przechowywania właściwe dla systemu operacyjnego dostarczonego przez producenta /np. ISAM/; dwa dalsze używają specjalnie opracowane wewnętrzne procedury - są one bardziej wydajne lub bardziej odpowiednie dla pewnych formatów danych i struktur zbiorów. Przedsiębiorstwa uzasadniały opracowanie systemów we własnym zakresie z tym, że dostępne na rynku pakiety zarządzania danymi nie są w stanie spełnić surowych wymagań systemu, zwykle z uwagi na zagęszczanie ba-

zy danych, wymagania wielorakich baz danych lub szczególne właściwości zastosowań. Jednak ostatni przykład obala ten argument.

Co najmniej trzy z tych czterech przykładów przedstawiają sobą rozległe zainwestowanie siły ludzkiej w rozwój oprogramowania systemowego, sięgając od około dziesięciu osobolat do około 75 osobolat. W tych samych trzech przykładach zaprojektowanie podsystemu zarządzania danymi traktowano jako część w projekcie całego systemu. Zamierzano przeprowadzić rozpoznanie projektowe w celu włączenia bezpośredniego zdalnego przetwarzania oraz możliwie dużej elastyczności, stosownie do wymagań przyszłych zastosowań - na ile można to było przewidzieć - oraz na ile mogłoby to być w sposób rozsądny wdrożone do raczej sztywno zdefiniowanego zestawu problemów.

Podano przykłady różnorodnych struktur zbiorów w pamięci. Jeden z systemów wykorzystuje słownikopodobne odwrócone struktury listujące, dwa stosują struktury przypadkowe tworzone za pomocą algorytmu lub badania tablic, a reszta stosuje pewną kombinację struktur sekwencyjnych, indeksowanych sekwencyjnych i przypadkowych. Trzy z systemów, które nie stosują struktur listujących, wykorzystują wypracowane techniki definicji danych i zagęszczania, najwidoczniej nie poświęcając tak dużej ilości czasu, jak jej można by się spodziewać, znając czas potrzebny dawniej na poczynienie oszczędności na pojemności pamięci.

Wnioski. Na podstawie wyżej naszkicowanych przykładów poniższe wnioski wydają się nieuniknione:

a. Można stwierdzić, że w gospodarce istnieją zarówno pojedyncze, jak i wielorakie zastosowania baz danych. Zazwyczaj uogólnione systemy zarządzania bazą danych są zdolne obsłużyć wielorakie zbiory, lecz rzadko wielorakie bazy danych. Mała wydajność wykorzystania zasobów maszyny okazuje się być jednym z czynników odstraszających przed szerszym rozpowszechnieniem się takich systemów.

b. Istnieje bardzo realna potrzeba zagęszczania danych celem zaoszczędzenia pojemności pamięci przy przechowywaniu bardzo dużych baz danych. Tu korzyści ekonomiczne mogą się okazać znaczne: zmniejszanie kosztu urządzeń pamięci o bezpośrednim dostępie oraz do pewnego stopnia zmniejszanie czasu dostępu i wyszukiwania. Niewiele z uogólnionych systemów posiada zdolność zagęszczania danych, chociaż powinno być łatwym wdrażanie do systemu tej cechy na specjalne żądanie użytkownika /jako cecha opcjonalna/.

c. Oddzielenie opisów danych od programów eksploatacyjnych /operacyjnych/ przyczynia się do zwiększenia elastyczności systemu. Ta koncepcja, czy zwana "katalogami formatu" czy "tablicami deskryptorów struktury rekordu" lub też inaczej, jest zarówno prawidłowa jak i wartościowa, gdy do-

daje się nowe zastosowania lub modyfikuje stare. Więcej niż dziesięć lat temu system SAGE udowodnił wartość takiego podejścia stosując tablice opisu danych /pn. COMPOOL/.

d. Istnieją co najmniej trzy rodzaje języków na płaszczyźnie styku z użytkownikiem: jeden - dla wdrażania, obsługiwania i modyfikowania samego systemu operowania danymi; drugi - pozwalający użytkownikowi zdefiniować dla systemu formaty i struktury danych; oraz trzeci - pozwalający użytkownikowi kierować systemem w przechowywaniu, manipulowaniu i wyszukiwaniu danych.

e. Największe szybkości wyszukiwania zależą bardziej od struktury pamięci i strategii dostępu do danych niż od środka przechowywania. W przedstawionych przykładach najszybszy dostęp zapewniała wysoce wyspecjalizowana struktura indeksowana sekwencyjna z bazą danych przechowywaną w komórkach danych /pamięć masowa IBM/ w porównaniu do mniej złożonych struktur przechowywanych na nominalnie szybszych stacjach pamięci dyskowej.

f. Te przedsiębiorstwa, które poszły drogą opracowania swojego własnego systemu, są obecnie na dłuższy okres czasu związane z ściśle określonym środowiskiem sprzętu.

Wdrożyły one sztywne, ściśle "przykrojone" systemy, które są dopasowane,wyłącznie do wyznaczonych docelowo zastoscwań. Takie podejście jest z pewnością do przyjęcia dla przedsiębiorstw, których zastosowania są ściśle określone i pozostaną stosunkowo stałe w ciągu dłuższego okresu czasu; jest to jednak błędne podejście dla przedsiębiorstw, których potrzeby nie są /nie mogą być/ zbyt ściśle określone lub dla takich, które są dynamiczne i będą podlegać zmianom w czasie.

Byłoby nierozumne przyjąć, że dzisiejsze uogólnione systemy zarządzania danymi są receptą na wszystkie bolączki. W dalszym ciągu istnieją braki w tych systemach,lecz są one ulepszane w miarę jak projektanci i użytkownicy uczą się coraz więcej na ich temat, tak jak to miało miejsce w stosunku do ewolucji systemów operacyjnych i kompilatorów. Wydaje się rzeczą nieuniknioną, że do tych pakietów systemowych w końcu wcielone zostaną własności wymagane przez większość przedsiębiorstw i będą one daleko mniej kosztowne od opracowań we własnym zakresie. Już same naciski ekonomiczne wymuszą ewolucję w tym kierunku.

W niniejszym rozdziale naszkicowano niektóre problemy zarządzania danymi oraz kilka rozwiązań, włączając w to niektóre techniki organizacji zbiorów, metod dostępu, zagęszczania danych, wymagania pracy w systemie bezpośrednim /on-line/, wielorakie bazy danych, techniki opisu danych oraz

języki użytkownika. Wielu problemów nie zidentyfikowano bezpośrednio; pełniejsza klasyfikacja powinna uwzględnić optymalną organizację zbiorów, możność przenoszenia między różnym sprzętem /zarówno w zakresie generacji, jak i między producentami/ sieci wielorakich baz danych, niezależnie od procedur języki w stosunku do języków uwypuklających procedury oraz języków definicji danych, wielu jednoczesnych użytkowników oraz bezpieczeństwo ich zbiorów, a także wcielenie lub oddzielenie opisów danych od samych danych.Następny rozdział dalej rozwija pojęcia o systemie operowania danymi oraz bada bardziej szczegółowo przedstawione wyżej zagadnienia i obecny stan umiejętności w tym zakresie.

IV. UZASADNIENIE I TECHNIKA OPEROWANIA DANYMI

Istnieje jedna i tylko jedna, bezwzględna zasada w dziedzinie zarządzania danymi: zmiana. Wymagania się zmieniają i możliwości się zmieniają, co zresztą często wiąże się ze sobą. Większość zagadnień, które nazywaliśmy problemami na końcu poprzedniego rozdziału, stanowi dziedziny, w których jeszcze szukamy sposobów, jak uporać się ze zmianą.

Niniejsze sprawozdanie zakłada, że uogólniony system zarządzania danymi jest najbardziej praktycznym rozwiązaniem większości tych problemów, głównie z powodu jego teoretycznej zdolności do przystosowania się do zmiany. Idealny uogólniony system zarządzania danymi powinien być dostatecznie ogólny, by obsłużyć szeroki zakres zastosowań i nadawać się bez trudności do rozszerzenia oraz powinien być dostatecznie modularny, by pozwalał na niezależne opracowanie tych rozszerzeń, jak również nabycie przez indywidualne ośrodki obliczeniowe. Mówiąc o "teoretycznych" i "idealnych" systemach zakładamy istnienie granicy pojęciowej, która z samej definicji nie została osiągnięta przez żaden z istniejących uogólnionych systemów zarządzania danymi, chociaż zbliżają się one do tej granicy w różnym stopniu. Dlatego rozsądnym było by przyjąć, że nowe i rozwojowe systemy są coraz mniej oddalone w czasie.

> Pojęcie uogólnionego operowania danymi jest wynikiem obserwacji. Wiele funkcji wykonywanych przez systemy informacji kierownictwa lub dowodzenia jest wspólnych dla znakomitej większości tej kategorii systemów. O ile procedury mogą być napisane w ten sposób, że są dosyć uogólnione, by wykonywać te wspólne funkcje, to wówczas programista może się niemal wyłącznie skoncentrować na tych funkcjach, które są specyficzne dla jego problemu.[9]

Definiujemy program zastosowań jako zestaw przekształceń wykonywanych na zbiorach danych i strukturach danych, niezależnie od własności struktury pamięci czy od własności danych w powiązaniu ze strukturami i środkami pamięci. Celem uogólnionego systemu zarządzania danymi jest zapewnić taką niezależność oraz centralizować konieczne funkcje i możliwości. Zaletą tej centralizacji jest to, że zmiany sprzętu, zmiany struktury pamięci oraz inne zmiany mające wpływ na dane mogą być przyswojone przez jeden system bez modyfikacji /w idealnym przypadku/ wszystkich programów z dziedziny zastosowań.

Przykładem innego uogólnionego pakietu oprogramowania jest zawsze obecny /w komputerze/ system operacyjny. W pewnym sensie jest to zły przykład, ponieważ większość systemów operacyjnych wydaje się być zbyt uogólniona /w usiło-

waniu, by być wszystkim dla wszystkich ludzi/. Niemniej przeto istnieje stosunkowo mało instalacji komputerowych,dla których opracowuje się własny system operacyjny. IBM zużyło conajmniej 5.000 osobolat na opracowanie systemu operacyjnego OS/360; w czasie tych przygotowań inwestowano ponad $50 milionów rocznie. Dlaczego więc każdy użytkownik systemu 360 miałby powtarzać ten wysiłek?

Czy wielu w ogóle na to stać?

Ta sama analogia może mieć zastosowanie do kompilatorów COBOL'u /dziedzina programów dla zastosowań jest im tutaj bliższa/ lub do jakiegokolwiek innego języka programowania wyższego poziomu. Unika się powtarzania wysiłku w dziedzinie narzędzi systemu, z wyjątkiem konkurencyjnych ulepszeń i tu właśnie system zarządzania danymi jest jednym z narzędzi.

W tym rozdziale zajmiemy się elementami i funkcjami systemu zarządzania danymi, zwracając uwagę na uogólnione systemy zarządzania danymi oraz przebadamy rozpowszechnione obecnie zagadnienia technologiczne tak, jak to nakreślono w poprzednim rozdziale. Jako podstawa do opracowania niniejszego rozdziału służyły następujące uogólnione systemy /niektóre z nich nie są zbyt uogólnione/:

Disk FORTE	Burroughs
DM-1	Auerbach
GIM	TRW Systems Group
GIS	IBM
IDS	General Electric
IMS-DL/1	IBM
ISL-1	Information Systems Leasing Corporation
MARK IV	Informatics
TDMS	System Development Corporation
UL/1	RCA

Wszystkie one są eksploatowane co najmniej w swej podstawowej wersji, z wyjątkiem ISL-1, który znajduje się w trakcie opracowania i zaplanowano jego sprawdzenie u klienta pod koniec 1970 roku. Także DM-1 jest faktycznie obecnie niedostępny jako kompletny pakiet oprogramowania; chociaż wdrożony dla co najmniej dwóch klientów, jest on raczej narzędziem koncepcyjnym dla opracowania systemów operowania danymi. Obydwa te systemy włączono, by dać przykład pewnych interesujących koncepcji.

A. Elementy i funkcje systemu zarządzania danymi

Widzieliśmy już, że system zarządzania danymi zajmuje się strukturami danych, strukturami pamięci oraz wartościami danych.

Można tu wydzielić sześć następujących ogólnych funkcji:

Wprowadzanie - system przyjmuje wartości danych lub informacje dotyczące struktur danych i struktur przechowywania /struktur pamięci/.

Przeszukiwanie - system bada opisy struktur danych i struktur pamięci /oraz może mieć dostęp do bazy danych/, by się upewnić czy istnieje i gdzie jest umieszczona pewna wartość danych.

Przechowywanie - system dokonuje dostępu do bazy danych w celu dodania, zamieszczenia, zmodyfikowania lub usunięcia wartości danych. Ta funkcja jest również znana pod nazwą aktualizacji zbioru.

Obsługa - system tworzy lub modyfikuje opisy danych, struktury danych oraz struktury pamięci, by je przystosować do zmiany.

Wyszukiwanie - system dokonuje dostępu do bazy danych celem uzyskania wartości danych uprzednio przechowanych.

Wyprowadzanie - system emituje wartości danych lub informację dotyczącą struktury danych lub struktury pamięci.

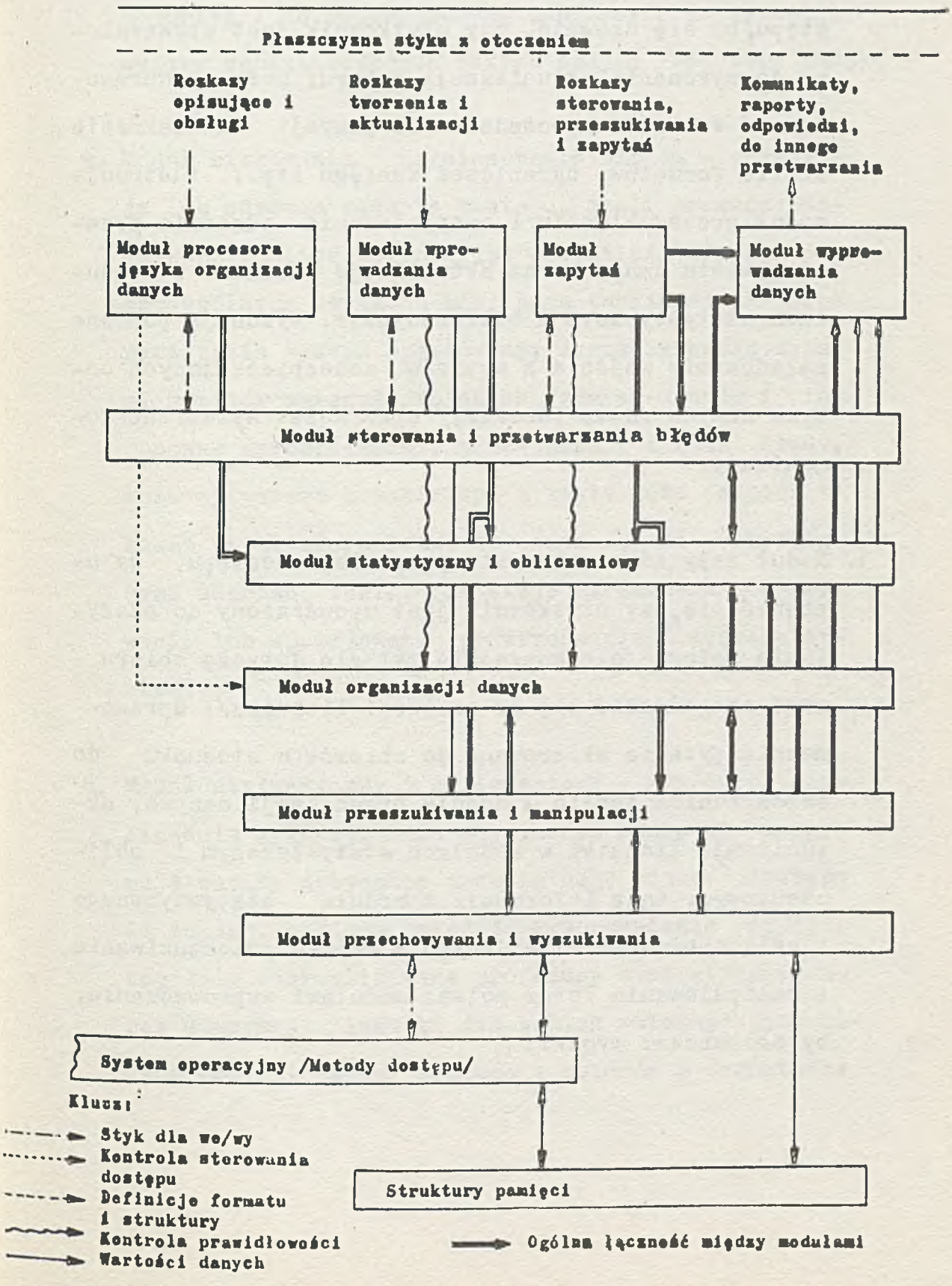
Wprowadzamy hipotetyczny model systemu operowania danymi oraz definiujemy jego elementy. Żaden z obecnie dostępnych systemów nie posiada przedstawionej tutaj struktury, chociaż niektóre z uogólnionych systemów zarządzania danymi przypominają taką strukturę. Przedstawiona struktura nie jest potencjalną konstrukcją takiego systemu, lecz jest pomocą w ilustrowaniu działania i możliwości uogólnionego systemu zarządzania danymi. Języki użytkowników nie są bezpośrednio przedstawione; są one uwzględnione później, kiedy omówi się różne rodzaje przedstawionych rozkazów.

Moduły przedstawione w tablicy 4 są następujące:

1. Moduł procesora języka organizacji danych - ustanawia przywileje dostępu, w celu sprawdzenia czy użytkownik jest upoważniony tworzyć, modyfikować lub usuwać wchodzącą w grę ściśle określoną strukturę danych lub strukturę pamięci na żądanym poziomie; interpretuje rozkazy opisujące i obsługi oraz wytwarza nowe wpisy w module organizacji danych albo modyfikuje istniejące; także aktualizuje liczniki w modułach statystycznym i obliczeniowym. Działalność tego modułu może generować komunikaty typu administracyjnego lub o błędach.

Tablica
E 61 -4

ELEMENTY SYSTEMU ZARZADZANIA DANYMI



2. Moduł wprowadzania danych - ustanawia przywileje dostępu, by się upewnić, czy użytkownik jest upoważniony do wykonania aktualizacji zbioru; może on uprawomocnić wartości wprowadzane do pozycji w zakresie tablic formatów, ograniczeń zasięgu itp.; instruuje moduł przeszukiwania i manipulowania odnośnie przechowywania danych oraz aktualizuje liczniki w modułach statystycznym i obliczeniowym. Wykonuje ponowne załadowanie wejścia z wypisów zabezpieczających odzysk danych /dump recovery/ oraz może wytwarzać komunikaty.

3. Moduł zapytań - ustanawia przywileje dostępu, by upewnić się, czy użytkownik jest upoważniony do odczytania zbioru /nie wszystkie pytania dotyczą zbioru - mogą one odnosić się do wartości licznika/; uprawomocnia pytania skierowane do zbiorów w stosunku do odpowiednich tablic w module organizacji danych, aktualizuje liczniki w modułach statystycznym i obliczeniowym, żąda informacji z modułu statystycznego i obliczeniowego albo danych z modułu przeszukiwania i manipulowania oraz poleca modułowi wyprowadzania, by dostarczał wyniki.

4. Moduł wyprowadzania danych - przesyła komunikaty i odpowiedzi do odpowiedniego odbiorcy oraz wykonuje wypisy zabezpieczające odzysk danych /recovery dump/.

5. Moduł sterowania i przetwarzania błędów - przyjmuje lub odrzuca żądania dostępu, jeśli prowadzi dodatkowe tablice definiujące przywileje dostępu poszczególnych użytkowników, poza tablicami określonymi przez system operacyjny; zarządza urządzeniami wypisów zabezpieczających /odzysk danych i ich ponowne wprowadzenie/, przetwarza i zlicza błędy, może wytwarzać komunikaty, a także może zapoczątkować działania, zależne od czasu lub od określonych zdarzeń, takie, jak akcje okresowego wymazywania lub spowodowane "przekroczeniem zakresu systemu".

6. Moduł statystyczny i obliczeniowy - wykonuje rozliczenia wykorzystania pojemności pamięci, utrzymuje zapisy dotyczące przeciętnego czasu dostępu do zbioru, oblicza częstość wykorzystania pozycji, zapewnia znormalizowane procedury statystyczne, oraz utrzymuje liczniki dla takich zmiennych, jak liczba pozycji, grup, zapisów i zbiorów w strukturze

danych, liczba bieżąco przechowywanych wartości pozycji, liczba zapytań itp. Może zaalarmować moduł sterowania i przetwarzania błędów o nadchodzeniu pewnych zdarzeń oraz o stanie pewnych zmiennych sterujących.

7. Moduł organizacji danych - zawiera wszystkie informacje na temat znajdujących się w systemie tablic, skorowidzów, słowników, indeksów, formatów, sposobów zagęszczania /opakowania/,itd.

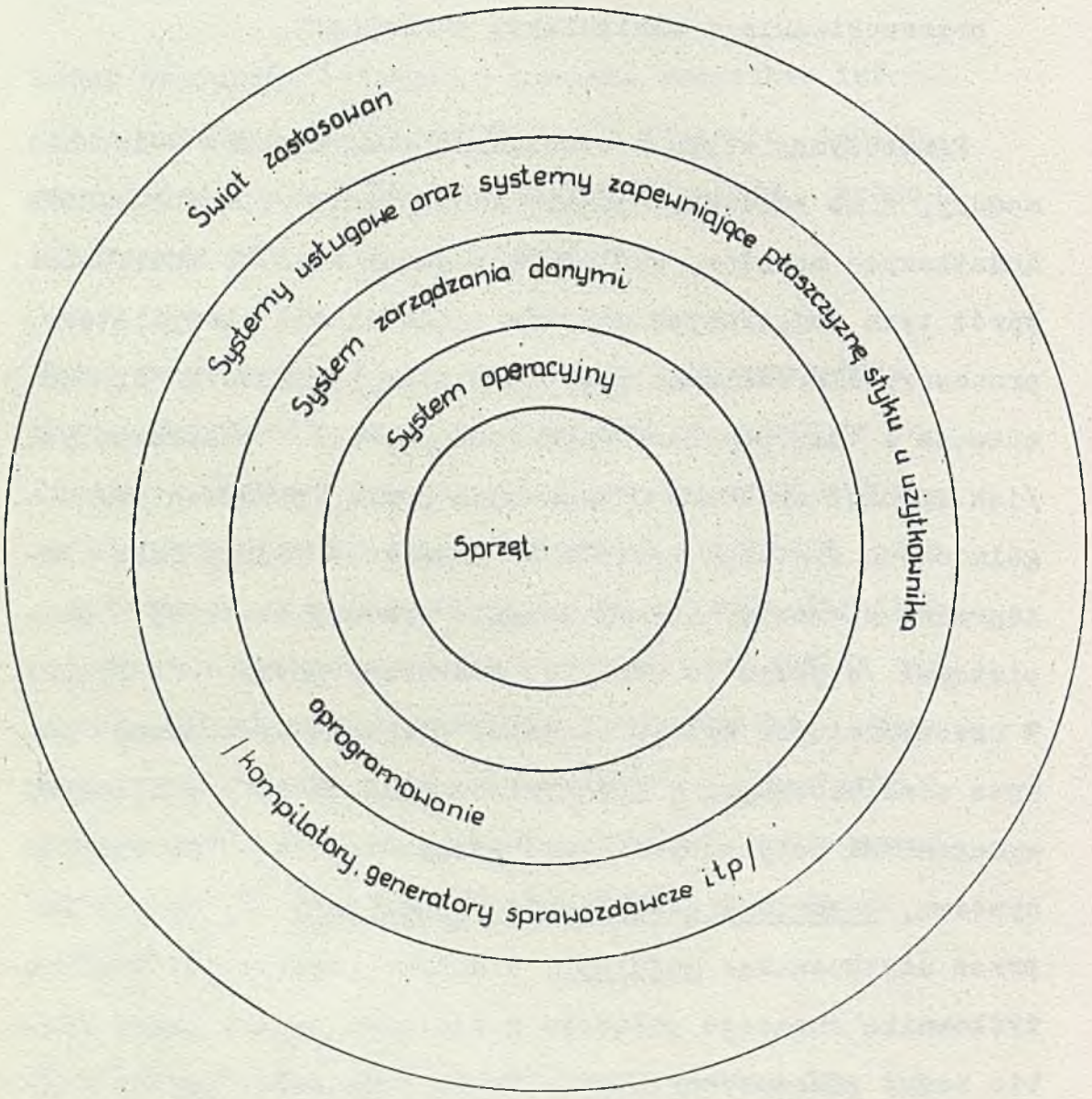
8. Moduł przeszukiwania i manipulacji - wykorzystując moduł organizacji danych, moduł ten przeszukuje indeksy i struktury pamięci, wytwarza obszary buforowe dla tymczasowego przetrzymywania zbiorów w celu dokonania na nich manipulacji, aktualizuje liczniki w modułach statystycznym i obliczeniowym,wydaje rozkazy modułowi przechowywania i wyszukiwania, przekazuje informacje o awariach modułowi sterowania i przetwarzania błędów, może zagęszczać i rozgęszczać dane, a także może wykonywąć funkcje sortowania i scalania.

9. Moduł przechowywania i wyszukiwania - używając modułu organizacji danych tworzy płaszczyznę styku z

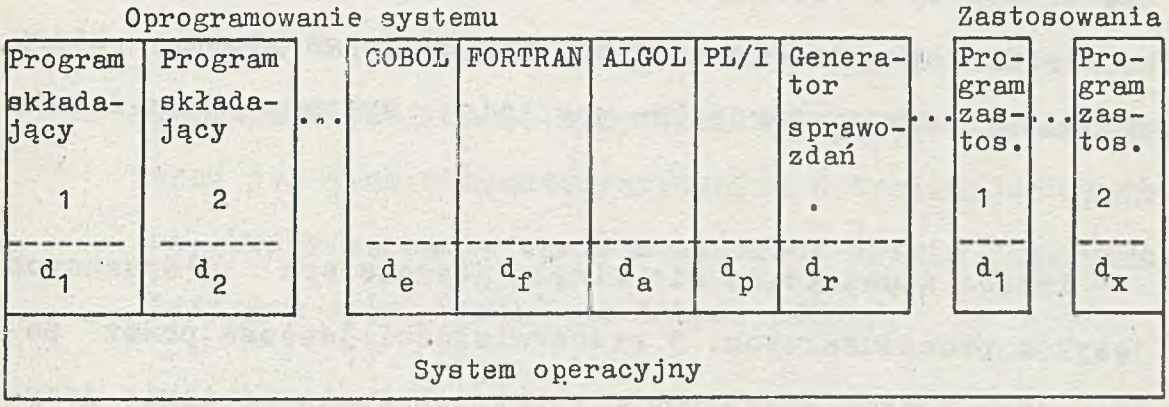
metodami dostępu, jakimi dysponuje system operacyjny, /a także może używać swoich własnych technik dostępu/, by umieszczać, przechowywać i wyszukiwać wartości danych. Moduł ten używa swoje własne lub systemu operacyjnego procedury naprawy błędów oraz zawiadamia moduł przeszukiwania i manipulacji o awariach.

Płaszczyzna styku z otoczeniem albo zawiera dodatkowe moduły, albo zapewnia "punkty zaczepienia" dla dołączenia dodatkowych modułów, względnie posiada obie te możliwości. Wśród tych dodatkowych modułów znajdują się kompilatory, procesory dla zdalnego przetwarzania, generatory sprawozdawcze i temu podobne. Większość prawnie zastrzeżonych /jak również systemów oferowanych przez producentów/ uogólnionych systemów posiada te dodatkowe moduły jako integralne elementy, tak by uczynić systemy bardziej kompletnymi /a przez to bardziej konkurencyjnymi na rynku/. W rzeczywistości byłoby bardziej pożądane dysponować dobrze zdefiniowanymi i podanymi do wiadomości publicznej wymaganiami dotyczącymi płaszczyzny styku z otoczeniem systemu, łącznie z dostępnymi opcjonalnymi / do wyboru przez użytkownika/ modułami, w dobrze pojętym interesie użytkownika chcącego połączyć z systemem żądany przez siebie moduł płaszczyzny styku. Takie podejście, pojęte najbardziej skrajnie, dałoby w wyniku przedstawione poniżej

powiązane ze sobą warstwy oprogramowania,gdzie każdy z poziomów posiada łączność jedynie z najbliższymi przyległymi poziomami.



Przeciwstawny pogląd, w szczególności utrzymywany przez orędowników ulepszonych języków proceduralnych, takich jak opracowany przez General Electric Integrated Data Store/IDS/, który jest rozszerzeniem języka COBOL, głosi, że funkcjonalne aspekty zarządzania danymi przenikają do wszystkich poziomów programowania i przetwarzania informacji, co można symbolicznie przedstawić na poniższym wykresie.



Tutaj każde d_i /gdzie i = 1,2,.....o/ przedstawia osobną zdolność zarządzania danymi dołączoną do każdego programu. System operacyjny zapewnia wspólne procedury wejścia -wyjścia, lecz programista musi zaprojektować strukturę swoich zbiorów i umieścić konieczne kodowanie w programie; to samo odnosi się do oprogramowania systemu.

Każdy z tych poglądów przedstawia różne etapy rozwoju. Ulepszone języki proceduralne /a stąd dodatki do każdego programu/ zostały opracowane w odpowiedzi na wymagania - nie spełnione przez projektantów konwencjonalnych języków i

kompilatorów,by w wygodny sposób ustanowić złożone zależności między danymi oraz manipulować nimi. Takie podejście do omawianego zagadnienia przedstawia bieg wypadków przed ukazaniem się "uogólnionych systemów zarządzania zbiorami",które usiłowały wypełnić nagle powstałą próżnię przez zebranie omawianych akcesoriów oprogramowania do wspólnej puli narzędzi budowy struktury i manipulowania zbiorami. Stoimy obecnie na pograniczu następnego ewolucyjnego etapu, który przyniesie samowystarczalne uogólnione systemy zarządzania danymi.

Trzeba zauważyć,że nie minęła jeszcze era ulepszonych języków proceduralnych. W rzeczywistości jeszcze przez pewien czas będzie występowało zapotrzebowanie na takie języki i wiele rozwinięć tych języków będzie utrzymanych w mocy jako naturalne rozkazy, które powinny się w takich językach znajdować od dawna, służąc jako łączniki do uogólnionych systemów zarządzania danymi.

Ta pozorna dygresja była konieczna, by wykazać,że znajdujemy się w stanie płynnym, w etapie przejściowym. Funkcje skomputeryzowanego zarządzania danymi towarzyszą nam tak długo jak sam komputer, lecz kierunek ich wdrażania ulega przesunięciu. W wyniku tego powstaje obecnie uogólniony system zarządzania danymi - jeszcze stosunkowo prymitywny - ale już kształtujący się.

B. Płaszczyzna styku z użytkownikiem

Użytkownik porozumiewa się z systemem zarządzania danymi, wymieniając informacje całymi partiami, co pewien czas lub w trybie bezpośrednim /on-line/, albo posługując się obiema tymi metodami, za pomocą języka, który pozwala na:

- wstępne definiowanie i późniejsze modyfikowanie lub ponowne definiowanie struktury danych, jej elementów, oraz jej cech charakterystycznych - "podzbiór" języka, który zapewnia takie możliwości będzie nazywany językiem opisu i obsługi danych /JOOD/.

- wstępne generowanie i późniejsze aktualizowanie lub manipulowanie wartościami bieżąco przechowywanymi w bazie lub bazach danych - podzbiór języka, który zabezpiecza takie możliwości będzie zwany językiem manipulacji i aktualizacji danych /JMAD/.

- kierowanie zapytań do bazy danych lub struktury danych, żądających albo bieżących wartości danych albo informacji na temat danych - podzbiór języka, który zapewnia takie możliwości będzie zwany językiem zapytań i wyszukiwań danych /JZWD/.

Te trzy podzbiory języka łącznie tworzą język dla celów operowania danymi /płaszczyznę styku systemu zarządzania danymi z otoczeniem/. W interesie zwiększenia wygody użytkownika wiele współczesnych systemów zarządzania danymi na tę płaszczyznę styku systemu z użytkownikiem nakłada wymaganie specyfikacji zadań, która pozwala użytkownikowi na odwołanie się do uprzednio przechowanych procedur lub na dodanie w bibliotece zadań systemu nowych procedur. Często używane procedury aktualizacji lub procedury wydruków sprawozdawczych mogą być raz opracowane i następnie zachowane dla wykorzystania w przyszłości. Znów rozszerzono koncepcję makrorozkazów, tym razem do powiązania w jedną całość zadań zarządzania danymi, które wówczas formują ciąg zadań obliczeniowych użytkownika.

Każdy z omawianych podzbiorów językowych posiada formę /sklasyfikowaną jako tabelaryczna, rozkazowa lub swobodna/ oraz rodzaj /sklasyfikowany jako proceduralny albo nieproceduralny/. Języki tabelaryczne używają uprzednio zdefiniowanych formatów, języki rozkazowe zwykle rozpoczynają każde określenie czasownikiem, po którym następują parametry, a języki posiadające formę swobodną zbliżają się w swym zasięgu do elastyczności i potencjału języka angielskiego. Język proceduralny wymaga od użytkownika, by ściśle określił w jaki sposób należy coś wykonać. Poniżej przedstawiono ogólne cechy charakterystyczne języków kilku współczesnych systemów zarządzania danymi.

Rodzaj / Forma	Proceduralne	Nieproceduralne
Tabelaryczne	Disk FORTÉ	MARK IV
Rozkazowe	IDS	TDMS
	GIS	ISL-1
Swobodne		UL/1
		GIM

Ogólnie biorąc, język proceduralny zawiera w sobie możliwość kontroli struktury pamięci przez użytkownika, podczas gdy język nieproceduralny zawiera niewidzialne struktury pamięci, których użytkownik nie ma możności kontrolować. Pierwszy z języków jest korzystny dla użytkownika, który wie, jak chce mieć zorganizowane struktury pamięci, by uzyskać "optymalną" wydajność, natomiast drugi język jest korzystny dla użytkownika, który nie wie w jaki sposób zmienia się strukturę pamięci /lub nie troszczy się o to/. Także zazwyczaj wzrasta złożoność i elastyczność przy przejściu od języka tabelarycznego do rozkazowego i dalej do swobodnego, przy czym jednocześnie wzrasta trudność w przeszkoleniu i początkowym użyciu. Forma tabelaryczna jest najłatwiejsza do nauczenia, można ją najszybciej w pełni wykorzystać, lecz jednocześnie jest ona najbardziej nieelastyczna. Forma swobodna języka jest najtrudniejsza do nauczenia, potrzebuje najwięcej czasu do pełnego

jej wykorzystania, a jednocześnie jest najbardziej skuteczna. Forma rozkazowa języka ze swymi zaletami i wadami wypada gdzieś pośrodku między tymi dwiema skrajnościami.

Poniżej reprodukujemy ten sam wykaz z przypisami przedstawiającymi "najlepsze" dopasowanie między rodzajami języka i jego formami, a poszufladkowanymi kategoriami użytkowników /co zawsze jest ryzykownym przedsięwzięciem/:

Rodzaj / Forma	Proceduralne	Nieproceduralne
Tabelaryczna	Wyuczeni urządnicy i programiści	Niewyuczeni urzędnicy i programiści
Rozkazowa	Wyuczeni programiści i analitycy	Niewyuczeni programiści analitycy i kierownictwo
Swobodne	Wyuczeni analitycy i kierownictwo	Niewyuczeni analitycy i kierownictwo

Użytkownik powinien móc porozumieć się z prawdziwie uogólnionym systemem zarządzania danymi za pomocą różnorodnych języków. Procedury tłumaczące systemu /translatory/mogłyby rozpisywać oryginalne języki źródłowe /FORTRAN IV czy

IPL-V, COBOL wzajemnego oddziaływania: wersja zapytaniowa tego języka; względnie CALL/360 BASIC/ na przyjmowane przez system podzbiory języków zarządzania danymi lub podprogramy biblioteczne, których wywołania są zamieszczane przez kompilatory do wygenerowanych kodów i mogą być zastąpione przez podprogramy, które "mówią" językiem systemu zarządzania danymi. Niewiele systemów, między innymi IMS-DL/1, pozwala programom wykonywać odwołania do podprogramu w języku zarządzania danymi. W każdym wypadku wdrożenie nie jest zagadnieniem, które tutaj rozpatrujemy kontynuowana dalej dyskusja pomija istotę łączności u źródła i koncentruje się na aktualnie zapewnianych użytkownikowi funkcjach przez wcześniej nazwane trzy hipotetyczne podzbiory języków. Należałoby tu zauważyć, że zachodzi rzeczywista różnica między JOOD i dalszymi dwoma językami, podczas gdy różnica między JMAD i JZWD jest raczej arbitralna. /Pierwszy z nich jest przede wszystkim narzędziem służącym do opisywania danych, a dwa ostatnie są przede wszystkim narzędziami służącymi do przemieszczania danych/. Tak samo nie będzie się próbować dyskutować na temat cech /własności/ pozwalających na wytwarzanie raportów /sprawozdań/, ponieważ ta możliwość na wyjściu z systemu odpowiada opisanej wyżej sytuacji dla wielorakich źródeł wejściowych.

1. Język opisu i obsługi danych /JOOD/

JOOD zapewnia środki, za pomocą których użytkownik może systemowi utożsamiać strukturę i przymioty danych, lecz nie w przypadku bieżąco przechowywanych wartości. Informacje wprowadzane przez użytkownika są tłumaczone przez moduł procesora języka organizacji danych, a następnie są przechowywane w module organizacji danych /patrz tablica 4/. Zakres wyspecjalizowanych możliwości jest różny w różnych systemach - głównie zależy on od tego, czy użytkownik może kontrolować w jakikolwiek sposób strukturę pamięci. Tam gdzie to ma miejsce, jak w przypadku systemów GIS lub MARK IV, struktura musi zostać wyraźnie określona. Tam, gdzie użytkownik nie posiada tych możliwości, jak na przykład w przypadku systemów TDMS i UL/1, system z samej istoty rzeczy oznacza strukturę pamięci. W poniższej dyskusji będą opisane bez jakiegokolwiek wyróżnienia zarówno funkcje, które musi wyraźnie określić użytkownik, jak i funkcje, które wynikają samoczynnie z działania systemu.

JOOD pozwala na wyspecyfikowanie:

identyfikacji struktury danych; użytkownik przydziela nazwy zbiorom, zapisom, grupom i pozycjom. Może on wskazać synonimy dla nazw i zależności hierarchiczne lub skojarzeniowe /powiązania logiczne/ dla określonych elementów.

przedmiotów fizycznych; użytkownik określa środki przechowywania zbiorów /taśma, dysk lub bęben/, reprezentację pozycji /stałoprzecinkowa, zmiennoprzecinkowa, alfanumeryczna, kodowana itp./oraz rozmiar każdej pozycji, grupy, wpisu i zbioru / o stałej lub zmiennej długości/. Te możliwości są zazwyczaj zawarte wewnątrz systemu; nie pozwala on na kontrolę struktury pamięci ze strony użytkownika.

bezpieczeństwa; użytkownik dostarcza zwykle informacji dotyczącej uprawnienia do dostępu do danych na poziomie zbioru i na poziomie pozycji.Nawet jeśli zbiór jest "ogólnie dostępny", niektóre pozycje w nim mogą mieć zastrzeżony dostęp,jak to może mieć miejsce w przypadku informacji na temat zarobków w kartotece personalnej.

uprawomocnienia; użytkownik może wskazać dla każdej pozycji pewne kryteria kontrolne, które muszą być spełnione przez wprowadzane wartości danych. Tu najbardziej powszechnymi są kryteria,takie jak wartość progowa, zakresy, porównania, tylko ściśle określone znaki, tylko ściśle określone formaty. Na przykład pozycja NUMER EWIDENCYJNY PRA-

COWNIKA może być ograniczona tylko do czterech znaków numerycznych. NAZWISKO będzie przyjmować tylko znaki alfabetyczne.

Wydawnictwa użytkownik może wskazać dla każdej pozycji, jakie przetwarzanie należy zastosować względem wprowadzanych wartości danych dla tej pozycji. Najbardziej powszechnymi rodzajami wydawnictwa jest zagęszczanie oraz zakodowanie z odpowiadającymi informacjami wydawniczymi w odniesieniu do wyjścia. Zagęszczanie zazwyczaj polega na usuwaniu znaków, które przedstawiają małe znaczenie albo nie przedstawiają żadnego znaczenia, przy czym faktycznie przechowuje się tylko posiadającą znaczenie część danych. Zakodowanie zwykle polega na konwersji z jednego formatu na inny /przeważnie numeryczny /. Numer ewidencyjny ubezpieczenia społecznego dla przechowania w pamięci może zostać pozbawiony łączników /znaków pisarskich: - /; takie proste zagęszczenie zaoszczędzi około 20 procent potrzebnej pojemności pamięci. Znajomość języka obcego /np. FRANCUSKIEGO/ może być zamieniona na jedno - względnie dwucyfrową liczbę, zakodowaną do użytku jako indeks do tablicy z takimi nazwami.

transakcji asocjacyjnych /kojarzeniowych/ użytkownik może określić, że jedna z pozycji ma wpływ na jedną,względnie kilka innych pozycji, ilekroć jakaś wartość ukaże się na wejściu tej pozycji. Najbardziej powszechne rodzaje transakcji to cechowanie dodatnie lub ujemne z wykonaniem lub bez wykonania obliczenia; to jest, wartość jakiejś pozycji podlega działaniu według pewnej formuły i wynik tego działania zostaje dodany lub odjęty od wartości innej pozycji. Tego rodzaju właściwość może wywołać reakcję łańcuchową rozszerzającą się po bazie danych jako wynik jednego wejścia /wprowadzenia jednej wartości/.

JOOD jest także środkiem, za pomocą którego użytkownik może na nowo zdefiniować i przerobić strukturę danych. Język ten zwykle pozwala mu dodawać, usuwać i modyfikować opisy zbiorów, wpisów, grup i pozycji, włączając w to wszystkie wymienione wyżej funkcje. Jednakże należy zachować ostrożność; większość systemów nie formuje automatycznie nowego zbioru, gdy zachodzi zmiana definicji starego zbioru. Zwykle użytkownik musi zdefiniować nowy zbiór, wywołać procedurę tworzenia zbioru /gdzie stary zbiór jest wejściem a nowy - wyjściem/,a następnie usunąć starą definicję zbioru. Taka kolejność postępowania wymaga użycia języka

manipulacji i sktualizacji danych dla etapu pośredniego. Zmiany ośrodka pamięciowego będą wymagały fizycznego przemieszczenia zbioru /znów nie automatycznie/, natomiast zmiany w nazwach, kodach bezpieczeństwa, uprawomocnieniu lub transakcjach prawdopodobnie nie będą wymagały zmian w jakichkolwiek wartościach danych.

2. Język manipulacji i aktualizacji danych /JMAD/

JMAD zapewnia środki, za pomocą których użytkownik może wywołać procedury systemu mające na celu zmianę wartości w bazie danych. Wartości są uprawomocnione i wydawane przez moduł wprowadzania danych /tablica 4/.Ogólnie biorąc, JMAD pozwala na:

generację; użytkownik dostarcza wartości i system generuje zbiór w zgodności z jego opisem uprzednio wprowadzonym za pomocą JOOD i przechowywanym w module organizacji danych.

tworzenie użytkownik określa jeden lub więcej zbiorów źródłowych oraz jeden lub więcej zbiorów docelowych oraz system odwzorowuje wartości na drodze od źródła do punktu docelowego.Na przykład, użytkownik może chcieć sporządzić kopię zbioru lub stworzyć podzbiór celem przeprowadzenia pew-

nego eksperymentu bez wprowadzania niezamierzonych zmian do oryginalnego zbioru. Może on też chcieć spróbować zastosowania innej struktury zbioru do tych samych danych celem ustalenia użyteczności różnych skojarzeń w zbiorze. Na koniec, użytkownik może życzyć sobie przejść od zewnętrznego formatu zbioru do wewnętrznego i na odwrót, jak to ma miejsce w zbiorach w COBOL'u przekształcanych dla dalszego przetwarzania.

rewizję użytkownik może aktualizować zbiór przez dodawanie lub modyfikowanie wartości pozycji lub przez usuwanie poszczególnych elementów pozycji, grup i wpisów. Rewizja uniwersalna obejmuje modyfikację wszystkich przypadków występowania jakiejś pozycji /np. dodanie 5 procent do zarobków każdego pracownika/, lecz dla każdej rewizji, która nie jest uniwersalna, użytkownik musi wyspecyfikować poszczególną pozycję lub zestaw pozycji, które należy zmienić /np. zmienić stan rodzinny pracownika nr 4108 z kawalera na żonatego/. Dodanie jakiejś wartości zwiększa liczbę przypadków występowania poszczególnej pozycji, usunięcie zmniejsza liczbę przypadków jakiejś pozycji /lub grupy czy wpisu/, a modyfikacja zamienia jakąś wartość na inną. Rewi-

zja pozwala,by nowe wartości były stałymi wynikami obliczeń, wartościami innych pozycji lub kombinacją wszystkich trzech wymienionych.Możliwości obliczeniowe są zwykle identyczne z tymi, jakie zapewnia język zapytań i wyszukiwań danych.

manipulacje; użytkownik może chcieć posortować zbiór albo rozdzielić względnie scalić zbiory.Dla sortowania specyfikuje on pozycje, które mają służyć jako klucze sortowania oraz specyfikuje porządek sortowania. Jeśli system dopuszcza złożone klucze sortowania /wynikające jedne z drugich/, użytkownik prawdopodobnie musi wskazać, które klucze są pierwotne, a które wtórne; dla wielorakich kluczy sortowania porządek /rosnący lub malejący/ może być ten sam lub różny. Na przykład, sortowanie alfabetyczne według nazwisk pracowników w kartotece personalnej może jako klucz pierwotny mieć nazwisko, a jako klucze wtórne pierwsze i drugie imiona, wszystkie w rosnącym porządku i w tej kolejności. Struktury pamięci, w których porządek jest zawarty w fizycznym umiejscowieniu /takim jak sekwencyjne lub sekwencyjne indeksowane/ powinny zostać zreorganizowane fizycznie, co oznacza tworzenie nowego zbioru z istniejącego, gdzie oba zbiory będą posiadały

tę samą nazwę i to samo umiejscowienie. Struktury pamięci, do których nie jest przypisany pewien porządek /takie, jak struktury listujące lub swobodne/ będą wymagały jedynie zmodyfikowania łączników lub indeksów. Rozdzielanie i scalanie są oczywiście funkcjami tworzenia zbiorów.

JMAD zapewnia listę zmian - dokument rewizji - tak, że można nadzorować ewolucję bazy danych lub można przekształcić zbiór o ile dokonano w nim mylnych zmian.

3. Język zapytań i wyszukiwań danych /JZWD/

JZWD pozwala użytkownikowi zażądać, by system wyprowadził pewne dane na zewnątrz /wyświetlił, wydrukował itd./ lub odpowiedział na zapytania dotyczące danych. Wejścia do systemu ze strony użytkownika są tłumaczone przez moduł zapytań oraz przetwarzane przez moduł statystyczny i obliczeniowy, względnie przez moduł przeszukiwania i manipulacji /tablica 4/. JZWD jest językiem zorientowanym na wykonywanie porównań i obliczeń - wykorzystuje on wyrażenia porównawcze, arytmetyczne i logiczne. Dla porównywania wartości pozycji z wartościami innych pozycji, ze stałymi lub z wynikami obliczeń używa się tu operatorów porównawczych. Operatory arytmetyczne określają, jakie mają być wykonane

obliczenia, a operatory logiczne składają wyrażenia proste w wyrażenia złożone. Poniżej przedstawiono typowe operatory:/oczywiście dla języka angielskiego - dla języka polskiego poza międzynarodowymi symbolami matematycznymi kod mnemoniczny operatorów w przypadku opracowania w kraju JZWD powinien być inny, być może taki, jaki się tu proponuje po znaku równości/

Porównawcze

LT / less than/ = MN /mniej niż/, mniejszy-a-e niż/

GT / greater than/ = WN /więcej niż, większy niż/

LE / less than or equal to/ = MR /mniejszy lub równy/

GE / greater than or equal to/ = WR /większy lub równy/

EQ / equal to/ = R /równe/

NQ / not equal to/ = RO /różne od/

Arytmetyczne

+ /dodawanie/

- /odejmowanie/

* /mnożenie/

/ /dzielenie/

E /exponentiation/ = P /potęgowanie/

SQR /o/ /square root/ = PK /pierwiastek kwadratowy/

LN /o/ /logarytm naturalny/

SIN /o/
.
.
.
} /funkcje trygonometryczne/

Logiczne

AND = I

OR = LUB

NOT = NIE

Zwykle język typu JZWD zapewnia następujące możliwości:

wyszukiwanie wybiórcze użytkownik specyfikuje warunki wyboru, jakie mają być spełnione przy wyszukiwaniu żądanych danych. Na przykład, wprowadzenie do systemu następującego ciągu wyrażeń:DISPLAY ALL EMPLOYEE NAMES FOR AGE GT 32 AND SEX EQ MALE AND /DEPENDENTS LT 4 OR MARITAL STATUS NQ MARRIED/, co w polskiej wersji JZWD mogłoby wyglądać tak:WYPISZ WSZYSTKIE NAZWISKA PRACOWNIKÓW WIEK WN 32 I PŁEĆ R MESKA I /RODZINA MN 4 LUB STAN RODZINNY RO ŻONATY/, spowoduje,że system wylistuje według nazwisk wszystkich mężczyzn starszych niż 32 lata albo nieżonatych albo z rodziną mniejszą niż 4 osoby. Nawiasy użyto w celu uniknięcia dwuznaczności logicznej.

wyszukiwanie bezwarunkowe; użytkownik specyfikuje bezwarunkowe wyszukiwanie danych w jakimś ściśle określonym zbiorze. Na przykład, program obliczania płac przejdzie przez cały zbiór płac, wyciągając te pozycje dla każdego pracownika, które są niezbędne dla obliczenia zarobków, potrąceń, sum zbiorczych i.in. dla każdego terminu wypłaty.

wyszukiwanie warunkowe; użytkownik stosuje słowa takie jak IF...ELSE /= JEŻELI...INACZEJ/ w celu sprawdzenia pozycji na pewne określone wartości wyznaczające alternatywne przebiegi akcji.Na przykład program przetwarzający rachunki dłużników badając konta każdego klienta na zadłużenie może otrzymać deklarację takiego rodzaju: IF / CURRENT DATE - BILLING DATE/ LT 90, CONTINUE, ELSE SET DELINQUENT = 1 AND .../ tu następuje akcja korygująca/; w wersji polskiej mogłoby to wyglądać następująco: JEŻELI /DATA BIEŻĄCA - DATA PŁATNOŚCI/ MN 90, KONTYNUUJ, INACZEJ USTAW PRZEKROCZENIE = 1 I... Spowoduje to wyróżnienie tylko tych kont dla których przekroczenie terminu płatności wynosi co najmniej 90 dni.

redukcja danych; użytkownik może zapytywać system o dane. Na przykład obliczenia statystyczne dla wszystkich przypadków jednej pozycji będą zawierały wartość maksymalną, wartość minimalną, wartość przeciętną, wartość środkową, wartość modulną odchylenie standardowe oraz całkowitą liczbę przypadków. Użytkownik może także żądać takich własności pozycji, jak ilość lub częstotliwość dostępu.

C. Porównywalne cechy charakterystyczne uogólnionych systemów zarządzania danymi

W niniejszym odcinku przyjrzyjmy się nieco bliżej systemom wymienionym na początku rozdziału oraz funkcjom i koncepcjom opisanym dla hipotetycznego modelu uogólnionego systemu zarządzania danymi w celu sporządzania zestawienia porównawczego. Omawiane zestawienie uwypukla niektóre z najbardziej interesujących lub niezwykłych możliwości systemów oraz przedstawia istotne różnice między systemami. Oczywiście nie jest to wyczerpujące porównanie wszystkich po kolei cech; takie szczegółowe informacje dla większości przedstawionych systemów można znaleźć w bibliografii, pozycja 7 i 18.

Kilka z terminów w zestawieniu użyto nieco przedwcześnie; wystąpią one później w tym rozdziale. Z drugiej strony, załączono opisane w rozdziale III systemy opracowane na WŁASNE POTRZEBY.

Która analiza zestawienia dostarcza kilku interesujących obserwacji. Niżej następuje jedna klasyfikacja systemów według niewielu istotnych cech charakterystycznych.

1. Disk FORTÉ oraz IDS - oba systemy dostarczane przez producenta komputerów - są zorientowane na programistę o najbardziej podstawowym poziomie. Niemal wszystkie cechy i możliwości innych systemów w tych dwóch systemach muszą być zaprogramowane. Są to jedyne systemy, które nie zabezpieczają niezależności opisów danych. Jednakże pozwalają one na stosowanie zarówno hierarchicznych, jak i sieciowych struktur danych /oczywiście zaprogramowanych przez użytkownika/; umożliwia to tworzenie bardziej złożonych powiązań między danymi.

2. DM-1 oraz IMS-DL/1 są także systemami zorientowanymi na programistę - lecz wyższego poziomu. Pozwalają one, by programista komunikował się z nimi w różnorodnych językach programowania, lecz nie dopuszczają sieciowych struktur danych. Ponieważ system DM-1 nie

został opracowany przez producenta, jest tak zaprojektowany, by był niezależny od maszyny,nawet gdyby musiał być całkowicie przepisany na nowo dla każdej serii maszyn.

3. GIS, MARK IV, TDMS i UL/1 są zorientowane na zwykłego użytkownika, który nie potrzebuje być programistą. Systemy te nie dopuszczają sieciowych struktur danych, chociaż zarówno MARK IV, jak i UL/1 dodają co najmniej częściową możliwość stosowania sieci,pozwalając na stosowanie łączników między zbiorami. Ani jeden z tych czterech systemów nie był przewidywany do przeniesienia na inne typy maszyn.

4. GIM oraz ISL-1 są najbardziej skomplikowane z systemów reprezentowanych w zestawieniu. Są one zorientowane na powszechnego użytkownika, dopuszczają sieciowe struktury danych, są niezależne od typu maszyny i zapewniają automatyczne, asocjacyjne wyszukiwanie i transakcje. Z przedstawionych tutaj systemów ISL-1 jest jedynym, który zrobił krok w kierunku dynamicznych struktur pamięci;jest bardzo ważny w rozwoju omawianych systemów.

Dalsze klasyfikacje można sporządzić na podstawie porównania innych cech i możliwości przedstawionych w omawianym zestawieniu; kilka z dalszych interesujących aspektów to: czy struktura pamięci jest uwidaczniana użytkownikowi oraz czy zabezpieczono możliwość tworzenia wielorakich baz danych, samodefiniujących się zapisów oraz zgodność /współwymienność/ zbiorów. Wiele z tych zagadnień omawia się w następnym odcinku. Występują one także w procesie oceny przedstawionym w rozdziale V.

Najważniejsze różnice i cechy charakterystyczne USZD

UOGÓLNIONE SYSTEMY POCHODZACE OD PRODUCENTA

SYSTEM	DISK FORTÈ
1. Przedsiębiorstwo	Burroughs
2. Sprzęt lub możliwość przeniesienia /wstępne wdrożenie/	B-2500 B-3500
3. Zdolność tworzenia wielarakich baz danych	Programowana przez użytkownika
4. Moduły dopasowania z otoczeniem/external interface modules/	Nie mają zastosowania
5. Zdolność zagęszczania danych	Zależna od języka użytkownika
6. Czy opis danych oddzielony od danych i programów	Nie
7. Język wdrażania systemu	COBOL
8. Język użytkownika	COBOL z rozszerzeniami
9. Dopuszczalne struktury danych /zdefiniowane przez użytkownika/	Hierarchiczna lub sieciową
10. Struktury pamięci /pod kontrolą użytkownika jeśli dlań widoczne./	Sekwencyjna indeksowana, swobodna, listująca /uwidoczniona dla użytkownika/
11. Środki pamięci	Urządzenie bezpośredniego dostępu typu głowica/ścieżka
12. Zgodność zbiorów	Zgodne na poziomie języka użytkownika

13. Wyszukiwanie asocjacyjne /łączniki wewnątrz i między zbiorami/	Programowane przez użytkownika
14. Transakcje asocjacyjne /automatyczne procedury/	Programowane przez użytkownika
15. Elementy danych fizycznie przechowywane jako zapisy	Wpisy
16. Wiązanie danych za pomocą	Indeksów, wskaźników
17. Ograniczenia w stosunku do zapisu	Stałej długości
18. Podstawowe techniki dostępu	Przegląd tablic, algorytm swobodnego dostępu, łańcuchowanie
19. Uwagi na temat środowiska operacyjnego	Zadanie systemu operacyjnego: wstępna kompilacja COBOL'u

UOGÓLNIONE SYSTEMY POCHODZĄCE OD PRODUCENTA

SYSTEM GIS	IDS
1. IBM	General Electric
2. IBM 360/40 i wyżej	GE 600-400-200
	Datanet 30
3. Tylko wielorakie niezależne zbiory	Tak[x] - programowana przez użytkownika
4. Język użytkownika GIS i generator raportów	Nie ma zastosowania
5. Wydawnictwo	Zależna od języka użytkownika
6. Tak	Nie
7. Język składający	Modyfikacje kompilatora języka użytkownika
8. Język opisu danych GIS, język wyszuk.i aktualizacji GIS	Język składający oraz COBOL, FORTRAN tylko na GE-600
9. Hierarchiczna	Hierarchiczna lub sieciowa
10. Sekwencyjna i sekwencyjna indeksowana /widoczna dla użytkownika/	Pierścienie /widoczne dla użytkownika/
11. Każde urządzenie wspomagane przez OS/360	Urządzenia bezpośredniego dostępu
12. Zgodne z OS	Zgodne na poziomie języka użytkownika
13. Nie	Programowane przez użytkownika
14. Nie	Programowane przez użytkownika

15. Wpisy	Grupy w stronnicy
16.	Wskaźniki łańcucha
17. Tylko stałej długości dla zbiorów sekwencyjnych indeksowanych, kiedy indziej mogą być stałej lub miennej dł.	Stronnice stałej długości /zdefiniowane przez użytkownika/
18. Przeszukiwanie sekwencyjne lub sekwencyjne indeksowane do zapisu wejściowego	System operacyjny: algorytm swobodnego dostępu bezpośrednio do stronnicy, sekwencyjne przeszukiwanie w kartotece głównej, łączniki łańcuchowe do kartoteki szczegółowej
19. Praca OS/360 w przetwarzaniu partiowym lub zdalnym	Praca systemu operacyjnego

UOGÓLNIONE SYSTEMY PODZOZĄCE OD PRODUCENTA

SYSTEM IMS-DL/1	UL/1
1. IBM	RCA
2. IBM 360/40 i wyżej	Spectra 70/45 i wyżej
3. Niezależne bazy danych	Tylko wielorakie niezależne zbiory[x]
4. BAL, COBOL, FORTRAN, procedury PL/I używające parametry DL/1	Język użytkownika UL/1 i generator raportów
5. Nie	Zdefiniowane przez użytkownika przeglądanie tablic celem odszukania zakodowanych pozycji, upakowanie wpisów
6. Tak	Tak na życzenie użytkownika mogą być przechowywane razem z danymi
7. Język składający	Język składający /procedury ponownego wchodzenia/
8. BAL, COBOL, FORTRAN, PL/I	Język użytkownika UL/1
9. Hierarchiczne	Hierarchiczne /sieciowe planowane/
10. Sekwencyjne i sekwencyjne indeksowane /widoczne dla użytkownika/	Sekwencyjne i sekwencyjne indeksowane /niewidoczne dla użytkownika/
11. Urządzenia bezpośredniego dostępu	Urządzenia dostępu sekwencyjnego i bezpośredniego
12. Nie ma	Procedury konwersji w systemie dla języka COBOL i FORTRAN
13. Nie ma	Nie ma[x]
14. Nie ma	Procedury użytkownika

15. Wpisy /niekoniecznie nienaruszone/	Wielorakie wpisy
16. Wskaźniki zapisów	Słownik:wskaźniki wpisu
17. Zapisy o stałej długości z nadmiarem dla zapisów o zmiennej długości	Zapisy o stałej długości zawierające możliwie dużo całych wpisów:wskaźniki do następnych wpisów
18. Szukanie sekwencyjne lub indeksowane sekwencyjne do zapisu wejściowego,łącznik łańcuchowy do podporządkowanych pozycji	Szukanie zapisu sekwencyjne lub indeksowane sekwencyjne, łańcuch do wejścia
19. Praca dla OS/360: DL/1 jest systemem partiowym, a IMS zapewnia łączność z terminalami	Praca systemu operacyjnego w trybie przetwarzania partiowego i w trybie łączności z terminalami /obecnie tylko przetwarzanie partiowe/

UOGÓLNIONE SYSTEMY NIE POCHODZĄCE OD PRODUCENTÓW

	SYSTEM DM-1	GIM	ISL-1
1.	Auerbach	TRW Systems Group	Information Systems Leasing Corp.
2.	Niezależny od maszyny /IBM 360/40 i wyżej/	Niezależny od masz /IBM 360/30 i wyżej/	Niezależny od masz. / -- /
3.	Tylko wielorakie zbiory niezależne od siebie	Niezależne bazy danych	Niezależne bazy danych[x]
4.	BAL, COBOL, PL/I i "wstawki"	Język użytkownika GIM i generator raportów	.Język użytk.ISL i generator raportów
5.	Wydawanie	Wydawanie i upakowanie zapisu	Pod kontrolą systemu
6.	Tak	Tak	Tak
7.	Język składający	Specjalny język z makrorozkazami	COBOL i w małym stopniu język składaj.
8.	Programy użytkownika z rozkazami "DAMOL"	Ograniczony język angielski	Język użytkownika ISL
9.	Hierarchiczna /z możliwością odwracania drzewka/	Hierarchiczna lub sieciowa	Hierarchiczna lub sieciowa
10.	Sekwencyjna i swobodna /widoczna dla użytkow./	Swobodna /niewidoczna dla użytkownika/	Półdynamiczna[xx] /niewidoczna dla użytkownika/
11.	Jakiekolwiek urządzenie wspomagane przez system operacyjny	Urządzenia bezpośredniego dostępu	Urządzenia dostępu sekwencyjnego i bezpośredniego
12.	Przyjmowane przez programy w BAL, COBOL, PL/I, JOVIAL	Należy dokonać konwersji przez program użytkownika	Współzgodne z COBOL em

13.	Tak	Tak	Tak
14.	Procedury użytkownika	Tak	Tak
15.	Wpisy	Wpisy	Zależy od struktury pamięci i cech sprzętu
16.	Słownika	Słownika	Słownika
17.	Stała długość: wielkość zależna od systemu operacyjnego	Nie ma /wpisy samodefiniujące się/	Nie ma, jeśli chodzi o użytkownika
18.	Przeglądanie tablic, a następnie wprost /poprzez system operacyjny/ wejściowego zapisu	Algorytm swobodnego dostępu do wejściowego zapisu	System operacyjny i własne procedury przeglądanie tabli indeksowanie, algorytmy swobodnego dostępu itp.
19.	Praca systemu operacyjnego	Praca systemu operacyjnego zamierzona dla łączności z terminalem	System operacyjny użytkownika plus rozszerzenia i nowe cechy: planuje si przetwarz. partiowe i łączność z terminalami

UOGÒLNIONE SYSTEMY NIE POCHODZĄCE OD PRODUCENTÒW

SYSTEMY SPECJALIZOWANE

SYSTEM MARK IV	TDMS	MIS
1. Informatics	System Development Corporation	National Breving Company
2. IBM 360/25 i wyżej	IBM 360/50 i wyżej	IBM 360/-
3. Tylko wielorakie niezależne zbiory[x]	Tylko wielorakie niezależne zbiory[x]	Niezależne bazy danych
4. Język użytkownika MARK IV i generator raportów	Język użytkownika TDMS i generator raportów	Nie ma zastosowania
5. Wydawanie:opcjonalnie przez użytkownika zdefiniowany przegląd tablicy za zakodowanymi pozycjami	Upakowanie wpisu	Nie ma
6. Tak	Tak	Nie
7. Język składający	JOVIAL i język składający w małym zakresie	Język składający
8. Język użytkownika MARK IV	Język użytkownika TDMS	Nie ma zastosowania[x]
9. Hierarchiczna	Hierarchiczna	Niezależna od użytkownika
10. Sekwencyjna i sekwencyjna indeksowana/widoczna dla użytkownika/	W pełni odwrócona listująca /nie widoczna dla użytkownika/	Sekwencyjna, sekwencyjna indeksowana, swobodna /niewidoczna/
11. Urządzenia dostępu sekwencyjnego i bezpośredniego	Urządzenia o bezpośrednim dostępie	Urządzenia o bezpośrednim dostępie

12.	Zgodne z OS	Należy dokonać konwersji przez procedury tłumaczące systemu	Nie ma
13.	Planowane[x]	Nie ma	Tak, tylko wewnątrz każdej z baz danych
14.	Procedury użytkownika	Procedury użytkownika	Tak
15.	Wpisy	Wpisy	- -
16.	Słownik	Słownik	Programy systemowe
17.	Dla zbiorów sekwencyjnych indeksowanych - stałej długości; w innym przypadku - mogą być stałej lub zmiennej długości	Nie zachodzą /zapisy samodefiniujące się/	Ograniczenia z powodu OS/360
18.	Szukanie zapisu wejściowego sekwencyjne lub sekwencyjne indeksowane - poprzez system operacyjny	Przeglądanie tablicy a następnie wprost do zapisu wejściowego /przez system operacyjny/	System operacyjny: sekwencyjna, sekwencyjna indeksowana i bezpośrednia

SYSTEMY SPECJALIZOWANE

SYSTEM OMNIBUS	FMS	DF-2
1. Industrial Indemnity Company	Pacyfic Gas and Electric Company	International Minerals and Chemical: GE
2. IBM 360/-	IBM 360/-	Różne systemy General El.
3. Pojedynczy zbiór	Zintegrowane bazy danych/programowane/	Pojedyncza baza danych
4. Programy zastosowań /np. PL/I lub COBOL/	Języki użytkownika FMS i programy zastosowań /np.COBOL/	Języki użytkownika DF-2 /eksperymentalne/ i generatory raportów
5. Tak	Tak	Nie ma
6. Tak	Tak	Nie
7. Język składający	Język składający /procedury ponownego wejścia/	IDS/COBOL
8. Języki programowania wyższego poziomu	Trzy języki o stopniowanej trudności i elastyczności	Podobny do angielskiego język przechowywania i wyszukiwania danych
9. Hierarchiczna	Hierarchiczna	Hierarchiczna
10. Sekwencyjna indeksowana/niewidoczna dla użytkownika/	Pierścieniowa swobodna/widoczna dla użytkownika/	W pełni odwrócona listująca /niewidoczna dla użytkownika/
11. Urządzenia bezpośredniego dostępu	Urządzenia bezpośredniego dostępu	Urządzenia bezpośredniego dostępu
12. Nie ma	Nie ma	W połowie zgodne na poziomie COBOL'u

13.	Ograniczone do żądań wypłaty odszkodowań	Tak, wewnątrz każdego zbioru	Programowane przez użytkownika w IDS
14.	Nie ma	Nie ma	Nie ma
15.	Wpisy /"sekcje" polis lub "sekcje" żądań zapłaty/	Wpisy	Grupy w stronicy
16.	Słownik	Słownik /tablice/	Wskaźniki w łańcuchu
17.	Nie ma /wpisy samodefiniujące się/	Nie ma /wpisy samodefiniujące się o zmiennej długości z grupami i pozycjami o zmiennej długości/	Jak w systemie IDS
18.	Przegląd tablic przy pomocy przeszukiwania binarnego, następnie bezpośredni dostęp do komórki danych i sekwencyjne szukanie wejścia	Przeglądanie tablicy, bezpośredni dostęp do zapisu wejściowego, powiązanie łańcuchowe do pozycji/tablica/	Jak w systemie IDS
19.	Praca OS/360 w trybie partiowym, system zapytaniowy w opracowaniu	Praca OS/360 w trybie partiowym i łączności z terminalem	Jak w systemie IDS

UOGÓLNIONE SYSTEMY POCHODZĄCE OD PRODUCENTA

SYSTEM	DISK FORTÉ
20. Zabezpieczenie danych	Żadne
21. Statystyka wykorzystania	Programowana przez użytkownika
22. Rezerwa i odzysk danych	System operacyjny
23. Dokumentacja kontrolna	Programowana przez użytkownika
24. Komentarze i uwagi	Użytkownicy-programiści
SYSTEM GIS	**IDS**
20. Wyszukiwanie: na poziomie zbioru i pozycji; Aktualizacja: na poziomie zbioru i pozycji	Wyszukiwanie: nie ma Aktualizacja: na poziomie grupy
21. Sterowana przez użytkownika	Programowana przez użytkownika
22. System operacyjny	Pod kontrolą użytkownika
23. Tylko na poziomie zbioru; programu użytkownika dla poziomu pozycji	Programowana przez użytkownika
24.	Użytkownicy - programiści x Cały ośrodek pamięciowy jest "zbiorem IDS" - użytkownik może arbitralnie budować tyle zbiorów logicznych lub baz danych ile zechce
SYSTEM IMS-DL/1	**UL/1**
20. Dla wyszukiwania i aktualizacji:cała baza danych jest albo wspólna, albo zastrze-	Wyszukiwanie:na poziomie pozycji Aktualizacja: na poziomie po-

żona, albo zupełnie zablokowana	zycji /tylko dla wersji bezpośrednich - on-line/
21. Programowana przez użytkownika	Raporty systemu operacyjnego
22. Pod kontrolą użytkownika poprzez dokumentację kontrolną	System operacyjny
23. "Notatnik" na taśmie	Według wyboru użytkownika
24. Użytkownicy - programiści	Synominy nie są dozwolone x Późniejsze wersje pozwolą na łączniki między zbiorami i wewnątrz zbiorów

UOGÓLNIONE SYSTEMY NIE POCHODZĄCE OD PRODUCENTÓW

SYSTEM DM-1	GIM
20. Wyszukiwanie:na poziomach zbioru,wpisu, grupy, pozycji; Aktualizacja: na poziomach zbioru, wpisu, grupy,pozycji	Wyszukiwanie: na poziomach zbioru i pozycji; Aktualizacja:na poziomach zbioru i pozycji
21. Nie ma	Sterowana przez użytkownika
22. Nie ma	Pod kontrolą użytkownika
23. Nie ma	Jest
24. Użytkownicy -programiści	

SYSTEM ISL-1

20. Zarówno dla wyszukiwania,jak i aktualizacji na poziomach bazy danych, zbioru, wpisu i pozycji

21. Raporty o eksploatacji systemu; zapytania użytkownika

22. Pod kontrolą użytkownika

23. Jest

24. x "Powszechna" baza danych może być zdefiniowana jako część z każdej z niezależnych baz danych, przy zapewnieniu,że wspólne nie powtarzają się i są aktualne w każdej z tych baz danych

    xxOrganizację wybiera algorytm, celem osiągnięcia wyważenia między wymaganiami dla aktualizacji a wyszukiwania - oczekiwane wykorzystanie i zawartość są mierzone w czasie; użytkownik otrzymuje komunikat,jeśli struktura nie jest odpowiednia dla bieżącej działalności /sekwencyjna, sekwencyjna indeksowana i półsekwencyjna wieloindeksowa w pierwszej wersji/.

SYSTEM MARK IV	TDMS
19. Praca DOS lub OS/360 /IBM/: tylko w trybie przetwarzania partiowego	Praca systemu operacyjnego dla wielodostępu, posiadającego na zapleczu możliwość pracy w trybie przetwarzania partiowego
20. Na poziomie pozycji /przez użytkownika zapewniającego wielorakie definicje pozycji	System operacyjny
21. Raporty o eksploatacji systemu	Opcjonalne raporty o eksploatacji systemu
22. System operacyjny	Pod kontrolą użytkownika
23. Sterowane przez użytkownika	Nie ma
24. x "Zbiór ze współrzędnymi" - specjalna właściwość struktury znajdująca się w opracowaniu, celem umożliwienia powiązań między zbiorami - wdrażanie w III kw. 1969.	x Każdy zbiór jest bazą danych w systemie TDMS

SYSTEMY SPECJALIZOWANE

SYSTEM MIS	OMNIBUS
20. System operacyjny	Kontrola prawidłowości jedynie przez porównanie z katalogiem formatów
21. Nie ma	Nie ma
22. System operacyjny	Przed/po wypisaniu na taśmie obrazu ścieżek
23. Nie ma	Taka sama jak dla rezerwy i odzysku danych
24. x System dla zastosowań zawiera w pełni możliwości zarządzania danymi	Użytkownicy - programiści
SYSTEM FMS	**DF-2**
20. Na poziomie pozycji dla każdego rodzaju dostępu	Takie samo jak dla IDS
21. Sterowana przez użytkownika	Taka sama jak w IDS
22. System operacyjny	Taka sama jak w IDS
23. Nie ma	Taka sama jak w IDS

- . -

D. Napotykane trudności

Jednym z największych problemów dnia dzisiejszego w dziedzinie zarządzania danymi jest optymalna organizacja zbioru. Jest to zagadnienie, w którym w grę wchodzą takie czynniki jak wielkość zbioru, zmienność danych, czas odpowiedzi oraz środki przechowywania informacji. Problem ten komplikuje okoliczność, że niewielu użytkowników naprawdę rozumie zasady organizacji zbiorów.

Jeden z zapytywanych użytkowników skarżył się, że "nawet w naszych grupach programistów nie posiadamy ludzi, którzy naprawdę rozumieją problemy zarządzania zbiorami".

Inny z użytkowników powiedział, "personel produkcyjny chce otrzymywać dane zorganizowane według numeru części /chcą oni także otrzymywać rozwinięcia i zwinięcia/, inni chcą je mieć zorganizowane według kodu zamówienia materiałowego, a jeszcze inni chcą je mieć zorganizowane według numeru zamówienia dla analizy sprzedaży".

Jeden z badaczy podsumował ten problem słowami: nie można tak naprawdę powiedzieć, że uogólnione systemy posunęły naprzód teorię w zakresie organiza-

cji zbiorów; raczej można powiedzieć, że systemy te zmniejszyły zapotrzebowanie na szeroką specjalizację w tej dziedzinie".

Równie zawodne, z różnych przyczyn, okazało się zagadnienie możności przenoszenia systemów. Nowocześni dyrektorzy przetwarzania coraz więcej skłaniają się w kierunku "mieszanki" - korzystania z elementów systemu /głównie urządzeń zewnętrznych/ dostarczanych przez różnych producentów - a to celem osiągnięcia większych korzyści w układzie koszt-wydajność. A nad wszystkim rozpościera się cień następnej generacji, która nie wiadomo, jaką będzie. Zrozumiałe, że producenci komputerów kontynuują wytwarzanie oprogramowania, które działa /jeśli w ogóle działa/ tylko na ich sprzęcie. Jakie wówczas nadzieje może posiadać użytkownik, gdy usiłuje wcielić obce urządzenie do swojej puli środków przechowywania informacji lub gdy dokonuje zmian sprzętu albo w zakresie trzeciej generacji albo na inną generację?

Większego znaczenia nabierają problemy powstające przy ustanawianiu sieci lub wielu oddzielnych baz danych. Bez względu na to, czy są one podzielone funkcjonalnie czy geograficznie, użytkownik spotyka się z koniecznością znalezienia kompromisu między redundancją danych i kosztem łączności, bazami danych wzajemnie połączonymi w przeciwieńs-

twie do niezależnych baz danych samowystarczalnych opisami danych przechowywanymi łącznie lub oddzialnie od danych oraz dostępem zdalnym jednoczesnym lub przeplatanym. Zabezpieczenie danych jest problemem, który rzutuje na wszystkie te sprawy.

1. Organizacja zbioru

Jak dotychczas nie ma wystarczającej ilości danych empirycznych pozwalających sformułować teoretyczne rozwiązanie problemu optymalnej organizacji zbioru. Dzisiejsze systemy zarządzania danymi są tak zaprojektowane, że albo intuicyjnie albo arbitralnie stosują pewne techniki lub ewentualnie wybór techniki pozostawiają użytkownikowi.

Projektant musi uwzględnić w wyborze odpowiedniej organizacji zbioru szereg przeciwstawnych czynników. Wchodzą tu w grę dwie podstawowe przeciwstawne możliwości: /a/ zbiór zorganizowany pod kątem widzenia wydajnego przeszukiwania wymaga wysokich kosztów przetwarzania wprowadzania oraz obsługi /i na odwrót/ oraz /b/ przy innych rzeczach takich samych, dla szybszego przeszukiwania potrzebny jest większy nadmiar pamięci .... /przykładem :a: /jest wybór między zbiorem uporządkowanym a nieuporządkowanym. Pierwszy łatwiej przeszukiwać /co najmniej według

argumentu klucza-porządkowania/ a drugi łatwiej uzupełniać ..... /Przykładem :b:/ jest wybór między przechowywaniem tablicy funkcji trygonometrycznych a przechowywaniem programu w celu ich obliczania.[10]

Przystąpmy do zbadania tego, co wiemy rzeczywiście na temat wzajemnego oddziaływania czynników wpływających na organizację zbioru.

Cel, bardziej niż wszystko inne, określa charakter systemu informacyjnego - rodzaj przechowywanej informacji, istotę żądań, kategorię użytkowników. Cel określa nie tylko dlaczego informacja jest potrzebna, lecz także w dużym stopniu formę, w jakiej jest potrzebna.[11]

Cel jest podstawą do określenia wielkości zbioru, jego zmienności, wymagań w zakresie wyszukiwania, oraz charakterystycznych cech użytkownika. Zbiór danych osobowych dla przedsiębiorstwa zatrudniającego 200 pracowników nie jest oczywiście tak duży, jak podobny zbiór dla przedsiębiorstwa o 10.000 zatrudnionych. Zbiór zaprojektowany dla sterowania zapasami towarów szybko psujących się jest w większym stopniu zmienny niż zbiór spektografii gwiezdnej układów pozasłonecznych; ten pierwszy posiada bardzo krótki "czas obrotu", ostatni jest bardzo stały,z stosun-

kowo rzadkimi zmianami /chociaż stale rosnący/. Zbiory zamierzone do wykorzystania w bezpośrednim przetwarzaniu zapytań i odpowiedzi, jak to ma miejsce w systemach rezerwacji, wymagają szybkiego wyszukiwania danych dla rozproszonej zbiorowości jednoczesnych użytkowników. Z drugiej strony zbiory przewidziane do wykorzystania w przetwarzaniu partiowym na zapleczu, takim, jak zastosowania zatrudnieniowo-płacowe, powinny zostać zorganizowane pod kątem widzenia wydajnego wyszukiwania dla pojedyńczego "użytkownika". Złym duchem, rzucającym cień na prace w tym zakresie jest oczywiście niepewność. O ile /i kiedy/ postawiony cel zmieni się w pewnym okresie czasu? Jak to wpłynie na organizację zbioru?.

Dodatkowe rozważania należy poświęcić środkom przechowywania informacji. Wielkość zbioru i wymagania w zakresie wyszukiwania dyktują w pewnym stopniu wybór środka przechowywania informacji.Jednak w tym wyborze ważną rolę także odgrywa budżet. I na odwrót, istota środka przechowywania informacji może podyktować wybór metod dostępu, i stąd, wybór struktur pamięci.

Wielkość i zmienność zbioru. Zakres wielkości zbiorów może wynosić od niewielu pozycji ujętych w postaci tabelarycznej, rezydujących w pamięci centralnej / jak to ma miejsce w przypadku wielu zbiorów systemowych/, do setek

milionów znaków przechowywanych na wielu krążkach taśmy magnetycznej /jak to ma miejsce w przypadku zbiorów w towarzystwach ubezpieczeniowych/. Wielkość zbioru może być stała lub zmienna w czasie. Problemy występujące w małych, względnie stałych, zbiorach, są niemal bez znaczenia w porównaniu z problemami zachodzącymi w przypadku posługiwania się olbrzymimi zbiorami, szczególnie, gdy nie są one ustabilizowane. Z uwagi na wykorzystanie pojemności pamięci najbardziej ekonomiczne jest zazwyczaj stosowanie organizacji sekwencyjnej. Zbiory tego typu są najlepiej dostosowane do szeregowego przetwarzania partiowego przy niewielkiej ilości występowania aktualizacji. Każdy inny rodzaj organizacji zbiorów odbija się na zwiększonym koszcie pojemności pamięci za cenę albo zwiększenia szybkości wyszukiwania, albo zwiększenia szybkości aktualizacji.

Wielkość zbioru wpływa na koszt pamięci w większym stopniu niż każdy inny czynnik. Tablica 5 przedstawia koszt pamięci w powiązaniu z wielkością zbioru dla kilku urządzeń pamięci z wymiennymi dyskami. Jednostkowe pojemności pamięci są reprezentowane przez przyrosty poziome, liczba jednostek /oraz przybliżony koszt miesięcznej dzierżawy/ przez przyrosty pionowe. Nie przedstawiono tutaj szybkości przesyłania czasu znalezienia

pozycji oraz czasu opóźnienia obrotowego. Powierzchnia sali może stać się problemem dla instalacji używającej jedynie jednostek 2311 IBM, ponieważ byłoby potrzeba około 30 takich jednostek dla przechowania zbioru o wielkości 200 milionów bajtów; dla tego samego zbioru byłaby odpowiednia pojedyncza jednostka IBM 2314.

Wielkość zbioru także ma wpływ na złożoność struktury pamięci. Jeśli na przykład projektanci systemu zdają sobie sprawę, iż jest wymagana duża szybkość wyszukiwania z bardzo dużego zbioru, to wówczas są oni skłonni do zaprojektowania bardziej wypracowanych schematów indeksowania lub łączenia, niż miałoby to miejsce w przypadku albo mniejszych zbiorów, albo zbiorów, gdzie nie jest wymagana taka duża szybkość wyszukiwania. Ta dodatkowa złożoność przynosi ze sobą koszt dodatkowej pojemności pamięci, a także zazwyczaj obsługa zbioru zabierze więcej czasu ponieważ będzie wymagało więcej procedur gospodarczych /housekeeping/.

Jeśli w ogóle jest to możliwe, należy przewidzieć szybkość wzrostu zbioru; wiele bardzo dużych zbiorów prostu wyrosło z początkowo małych zbiorów. Jeśli początkowo wybrano sekwencyjną lub indeksowaną strukturę pamięci, to wówczas dodawanie nowych pozycji do zbioru będzie coraz bardziej czasochłonne ponieważ, by zamieścić nowy zapis tego rodzaju struktury muszą być fizycznie kopiowane. Nie jest

wykluczone, że przy dużej szybkości wzrostu może dojść do takiej sytuacji, kiedy zbiór będzie ciągle aktualizowany i na wyszukiwanie informacji pozostanie albo bardzo mało czasu, albo w ogóle go zabraknie.

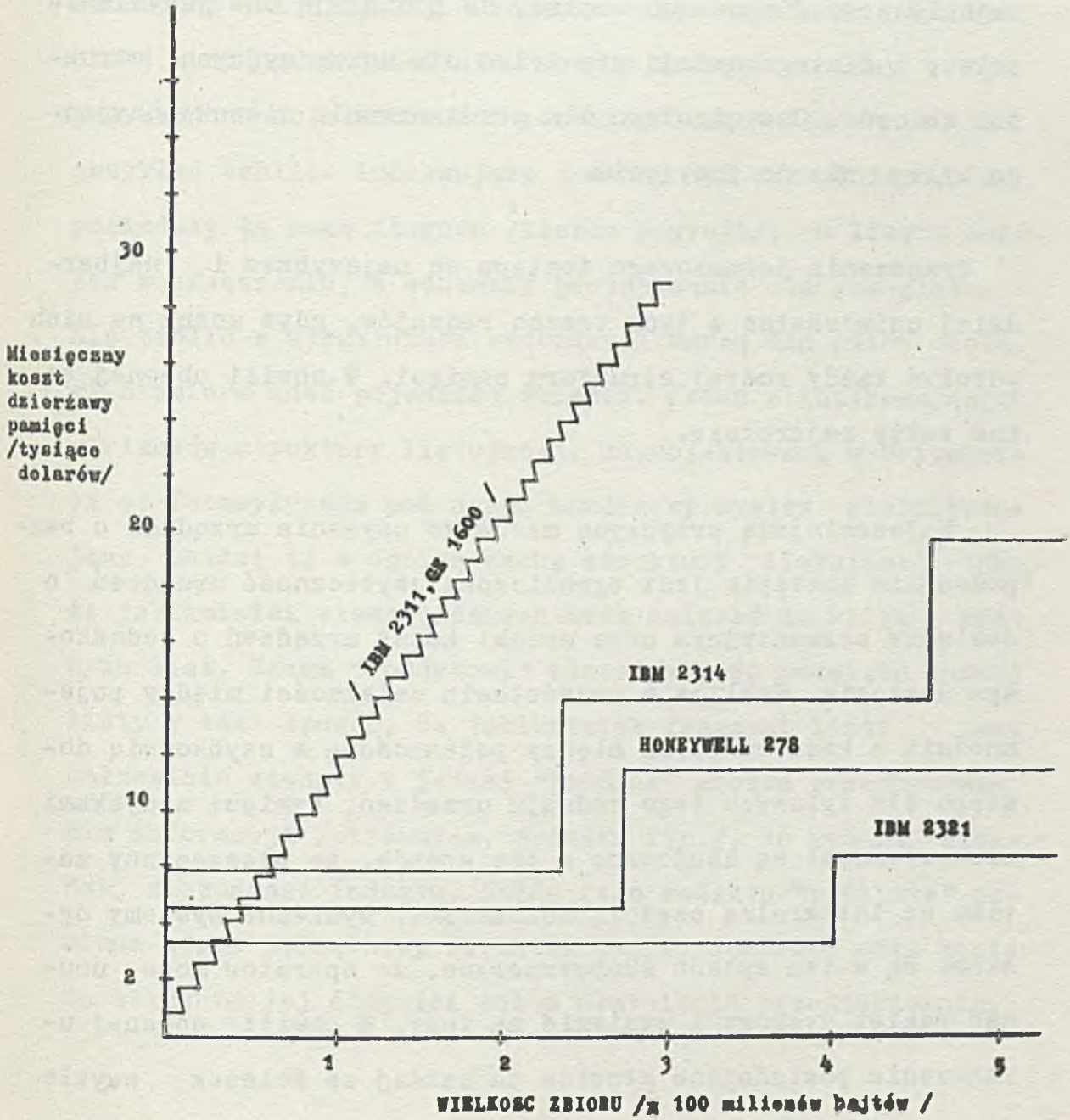
Podsumowując nasze rozważania można powiedzieć, że duży i ustabilizowany zbiór przewidziany do szeregowego przetwarzania partiowego celem zwiększenia wydajności winien być zorganizowany sekwencyjnie. Duży, lecz nieustabilizowany zbiór przewidziany do przetwarzania partiowego, powinien być zorganizowany w pewnej formie struktury listującej, a to celem łatwiejszej jego aktualizacji. Duży zbiór, ustalilizowany lub nie, przewidziany dla zastosowań w dziedzinie bezpośredniego wyszukiwania informacji prawdopodobnie powinien być zorganizowany jako swobodny, w dużym stopniu indeksowany lub o strukturze odwróconej. /Inne względy mogą dawać pierwszeństwo którejkolwiek z przedstawionych możliwości w tym zakresie/. Niestety, połączenie cech tego rodzaju, że zbiór jest bardzo duży, nieustabilizowany oraz zamierzony dla zastosowań zarówno w zakresie przetwarzania partiowego, jak i przetwarzania bezpośredniego /wyszukiwanie i aktualizacja/ wprowadza rzeczywiście przeciwstawne wymagania. W takim przypadku należy dokonać pewnego rodzaju oceny wartości różnych czynników celem rozwiązania na korzyść jednego z nich - zwykle kosztem innych czynników. Struktura swobodna może re-

prezentować najlepszy z kompromisów.

Środki przechowywania i metody dostępu. Istnieją trzy podstawowe rodzaje dostępnych w chwili obecnej na rynku urządzeń do przechowywania informacji. Pierwszy z tych rodzajów może być zwany urządzeniami o jednakowym dostępie: każde podane miejsce jest bez trudu dostępne jak każde inne, przy czym opóźnienia można pominąć, lecz dokonuje się dostępu tylko do jednego miejsca w jednej chwili czasowej. Każde miejsce posiada niepowtarzalny adres.
Centralne pamięci komputerowe /na rdzeniach ferrytowych lub inne/ oraz rozszerzone lub masowe pamięci na rdzeniach ferrytowych są urządzeniami jednakowego dostępu. Drugi rodzaj - to urządzenia bezpośredniego dostępu, dla których charakterystyczna jest technika dyskowa: dostęp do jakiejkolwiek ścieżki na dysku osiąga się przez bezpośredni ruch mechanizmu dostępu nad żądaną ścieżkę, bez badania po drodze danych. Z chwilą umiejscowienia się na żądanej ścieżce dostęp staje się sekwencyjny. Urządzenia z głowicami nad każdą z ścieżek są specjalnym przypadkiem, kiedy nie zachodzi opóźnienie powodowane umiejscowieniem mechanizmu dostępu nad żądaną ścieżką. Trzeci rodzaj - to urządzenia o dostępie sekwencyjnym, reprezentowane przez pamięci na kartach dziurkowanych, na taśmie papierowej lub na taśmie magnetycznej.

Tablica
E 61-5.

KOSZT PAMIECI JAKO FUNKCJA WIELKOSCI ZBIORU

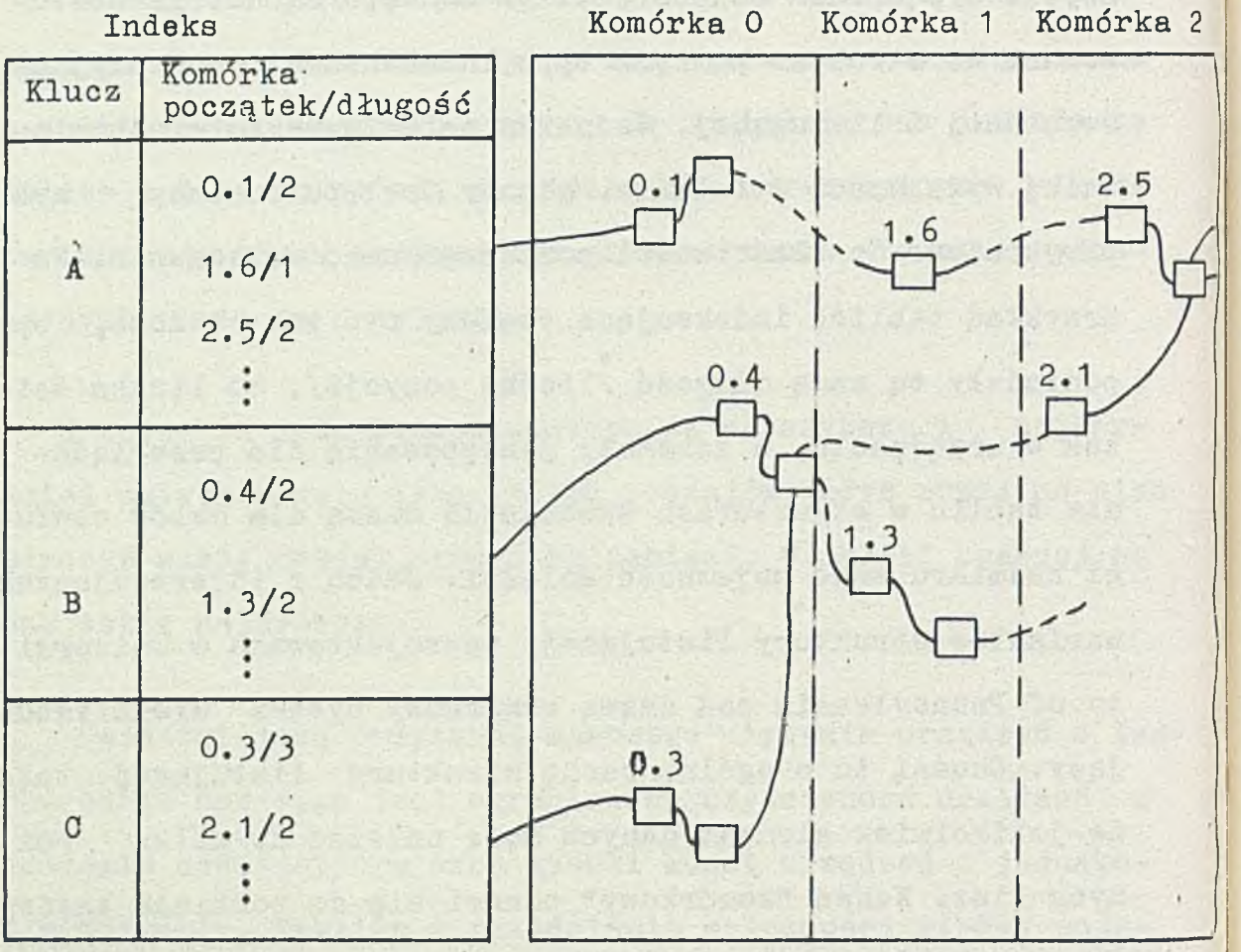


Możemy pominąć urządzenia o dostępie sekwencyjnym konstatując, iż są one dobre tylko dla zapisania lub odczytania najbliższego fizycznego zapisu, że posiadają one jedynie tę zaletę i dlatego nadają się tylko dla sekwencyjnych struktur zbiorów. Czas dostępu dla przetwarzania niesekwencyjnego byłby nie do przyjęcia.

Urządzenia jednakowego dostępu są najszybsze i najbardziej uniwersalne z tych trzech rodzajów, gdyż można na nich wdrożyć każdy rodzaj struktury pamięci. W chwili obecnej są one także najdroższe.

Najważniejszą przyczyną masowego używania urządzeń o bezpośrednim dostępie jest ograniczona użyteczność urządzeń o dostępie sekwencyjnym oraz wysoki koszt urządzeń o jednakowym dostępie. Tablica 6 przedstawia zależności między pojemnością a kosztem oraz między pojemnością a szybkością dostępu dla typowych tego rodzaju urządzeń. Pamięci z dyskami niewymiennymi są zbudowane w ten sposób, że płaszczyzny zapisu są integralną częścią mechanizmu. Wymienne systemy dyskowe są w ten sposób skonstruowane, że operator może usunąć pakiet dyskowy i wymienić na inny. W chwili obecnej urządzenia posiadające głowice na każdej ze ścieżek zwykle nie są wymienne /w zakresie pakietów dyskowych/. Przeciętny czas dostępu składa się z czasu ruchu ramienia z osadzoną głowicą lub z czasu przełączania głowic i z opóźnienia obrotowego.

Zwykle urządzenia bezpośredniego dostępu są dobrze dostosowane do struktur pamięci typu indeksowanej sekwencyjnej, swobodnej i listującej. Jednakże celem uzyskania odpowiedniej wydajności działania, metody dostępu powinny być przykrojone do właściwości poszczególnego urządzenia. Na przykład tablice indeksujące powinny być tak ułożone, by posiadały tą samą długość /liczbę pozycji/, co liczba ścieżek w urządzeniu, a schematy postępowania dla przeglądania tablic w strukturach swobodnych muszą dla celów obsługi nadmiaru znać pojemność ścieżki. Jeden z interesujących wariantów struktury listującej, zaprojektowano w University of Pennsylvania pod nazwą komórkowy system wielolistujący. Chodzi tu o ogólną cechę struktury listującej taką, że jakikolwiek element danych może należeć do kilku różnych list. Nazwa "komórkowy" odnosi się do podziału każdej listy w taki sposób, że jakikolwiek fragment listy jest całkowicie zawarty w jednej "komórce" środka przechowywania informacji /stronnica, ścieżka itp./, co wymaga większej złożoności indeksu. Każda tego rodzaju "podlista" posiada adres początkowy wewnątrz swojej komórki oraz posiada wskaźnik jej długości celem ułatwienia przeszukiwania.



W opracowaniu znajduje się kilka raczej interesujących nowych środków przechowywania informacji. Korporacja Ampex opracowała urządzenie o pseudobezpośrednim dostępie złożone z elementów taśmy wizyjnej o dostępie sekwencyjnym. Nazwano je terabit memory system /TBM/ - posiada ono pojemność dwa biliony bitów /2.10^{12}/. Jest to równoważne 330 miliardom sześciobitowych znaków lub 250 miliardom ośmiobitowych bajtów, czyli zgrubsza biorąc jest to pojemność tysiąc razy większa od pojemności pamięci dyskowej typu

IBM 2314. Czas dostępu jest nieco większy niż to zapewnią technika dyskowa.

Aby ulokować jakikolwiek blok danych potrzeba rzędu od 1 do 5 sekund. Bell Telephone Laboratories pracują nad pewną "bąbelkową pamięcią", której wielkość, koszt i zużycie mocy zapowiadają się mniejsze niż w konwencjonalnych urządzeniach elektronicznych. Przy pomocy takiej technologii ciała stałego na kryształach magnetycznych można wyprodukować urządzenie, które będzie mogło przechować 15 milionów bitów w pakiecie o pojemności jeden czy dwa cale sześcienne, przy zasilaniu energią wynoszącym 40 miliwatów. Z pewnością z czasem powstaną inne równie interesujące środki przechowywania informacji, co zwiększa potrzebę zdolności przystosowywania się systemów zarządzających wykorzystaniem urządzeń pamięciowych.

Istnieje jeszcze czwarty rodzaj urządzeń pamięciowych, znanych pod nazwą urządzeń o dostępie równoległym - nie są one jeszcze powszechnie dostępne. W tego rodzaju urządzeniach wszystkie miejsca pamięci są badane jednocześnie /wymaga to rozczłonkowanej logiki/. Umożliwiłoby to długo oczekiwaną "pamięć asocjacyjną", która dotychczas była symulowana na pamięciach tradycyjnych, za pomocą odpowiedniego oprogramowania. Rozumie się, że zadaniem pamięci asocjacyjnej jest przechowywanie i wy-

szukiwanie danych na podstawie ich zawartości, a nie na podstawie ich adresu. Techniki adresowania na zawartość za pomocą oprogramowania są reprezentowane przez metody swobodnego dostępu oraz mechanizmy wskaźników lub łączników. Większość obecnie prowadzonych badań wnioskuje, że wdrożenie pamięci asocjacyjnych w sprzęt jest wykonalne, ale jeszcze niepraktyczne, chociaż w ciągu najbliższych pięciu lat oczekuje się poważnych kroków w kierunku skonstruowania urządzeń bezpośredniego dostępu z logiką przydzieloną do głowic odczytująco-zapisujących umieszczonych nad każdą ze ścieżek.

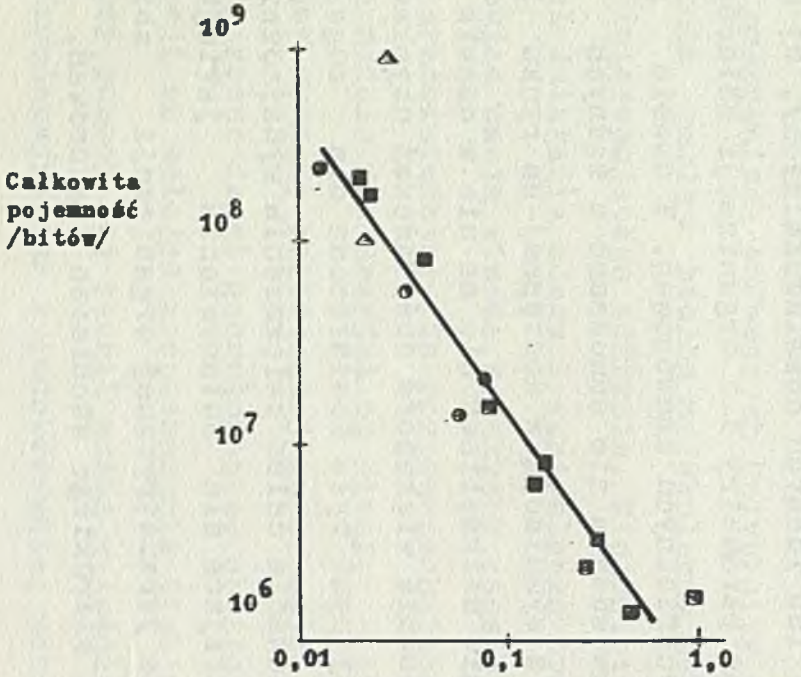
Na koniec, mogą powstać całkowicie odmienne rodzaje pamięci w ciągu najbliższych dziesięciu czy piętnastu lat. W pracowniach badawczych University of Southern California pewien biolog molekularny dokonał szczególnie interesującego odkrycia. Mianowicie skonstruował on model działania mózgu i centralnego systemu nerwowego stosując swoją wiedzę z zakresu genetyki molekularnej. W przeciwieństwie do rozpowszechnionego mniemania w swoim modelu nie przyjął on sieci przełączającej; wchodzące sygnały zakodowanej informacji po prostu otwierają różne drogi neurologiczne obniżając ich próg tolerancji na kolejne rozpoznawanie tych samych sygnałów. Przy pomocy tej teorii można wyjaśnić zarówno pamięci asocjacyjne, jak i indeksowanie; może to w konse-

kwencji prowadzić do urządzenia pamięciowego zdolnego do samouczenia się, ponownego wywoływania asocjacyjnego oraz innych funkcji /jak dotąd/ antropomorficznych/.

Wnioski. Istnieje bardzo duża różnorodność wariantów w wielkości zbiorów, ich stabilności i szybkości wzrostu. Zwykle wymagania w zakresie wyszukiwania informacji są przeciwstawne wymaganiom w zakresie jej aktualizacji. Ma to wpływ na struktury pamięci, środki przechowywania informacji oraz metody dostępu - co z kolei wpływa na wymienione wyżej cechy. W treści przedstawiliśmy pewne raczej proste wskazówki, lecz bardziej złożone sytuacje/jak to bywa w większości przypadków/ wymagają rzeczywiście pełnej symulacji lub /danych doświadczalnych/, by móc określić optymalne parametry dla organizacji zbiorów, przykrojonych do specyficznych zastosowań. W czasie pisania niniejszego sprawozdania nie donoszono o żadnych takich doświadczeniach z symulacją w dostępnej na rynku literaturze, ani też nie natknęliśmy się na nie w czasie naszych badań. Podobnie większość doświadczeń uzyskanych z systemami operacyjnymi była dostateczna dla sugerowania kierunków postępowania celem zwiększenia wydajności, lecz nie była wystarczająca dla ustanowienia jakiejkolwiek teorii w zakresie parametrycznej organizacji zbiorów - odpowiedniej dla szerokiego wachlarza zastosowań. Niemal

Tablica
E 61-6

CECHY CHARAKTERYSTYCZNE URZĄDZEŃ O BEZPOŚREDNIM DOSTĘPIE

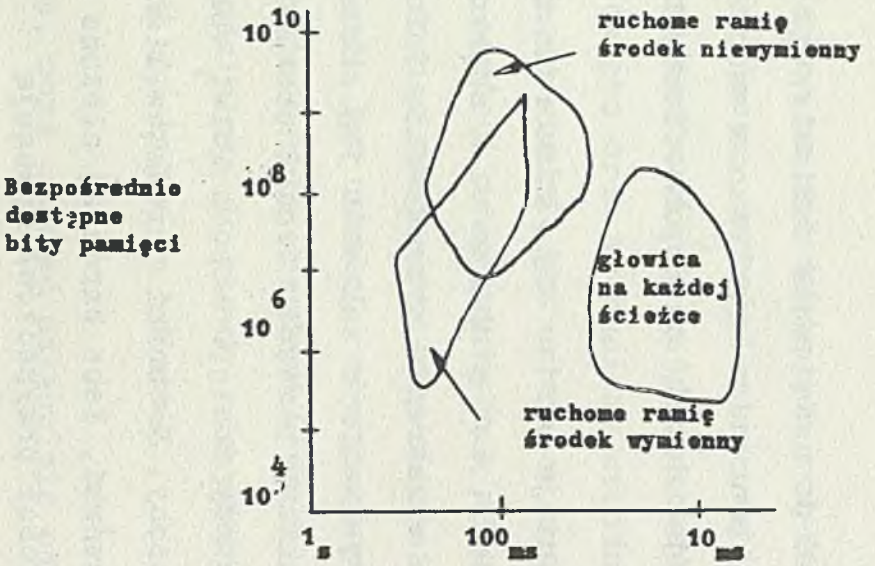


▲ środki niewymienne z ruchomym ramieniem

● środki wymienne z ruchomym ramieniem

■ głowica na każdej ścieżce , środek niewymienny

Zależność między całkowitą pojemnością pamięci a kosztem na bit



zależność między pojemnością pamięci a przeciętnym czasem dostępu według rodzaju systemu dyskowego

wszystkie argumenty za lub przeciw jakiejś szczególnej organizacji zbioru są wyłącznie oparte na założeniach logicznych. Jest to niebezpieczna prąktyka, gdy się nie zidentyfikuje i nie zatwierdzi wszystkich założeń.

Z powodu braku wypróbowanych technik optymalizacji tym bardziej jest potrzebne ogólne rozwiązanie problemu organizacji zbioru. Przedstawiamy tutaj rozsądne rozwiązanie:

> Potrzeba i uzasadnienie dla dynamicznie przystosowujących się struktur są proporcjonalne do ilości niepewności co do wielkości zbioru, zmienności lub stałości, oraz szybkości wzrostu.

Inaczej mówiąc, podane wcześniej proste wskazówki są odpowiednie i wystarczające, o ile problem organizacji zbioru jest dobrze zrozumiany, dobrze zdefiniowany oraz jest przedmiotem małych lub żadnych zmian /chyba, że są przewidywane i wzięte w rachubę/. O ile wzrasta brak zrozumienia lub brak zdefiniowania lub gdy rośnie prowdopodopodobieństwo zmiany, wówczas do systemu zbioru musi być wbudowane coraz więcej uogólnień. W przypadku skrajnym zbliżamy się do koncepcji dynamicznej struktury danych oraz pamięci : do automatycznego rozmieszczania i korygowania danych oraz pamięci celem osiągnięcia optymalnej wy-

dajności przez kontrolowanie bieżącej wydajności w stosunku do uprzednio wyznaczonych kryteriów w obliczu zmieniających się potrzeb.

Z systemów wcześniej porównywanych w tym rozdziale tylko system ISL-1 uczynił pewien krok w tym kierunku. Jednakże nie wykonuje on wymaganych korekcji automatycznie; po prostu informuje on administrację bazy danych, że wydajność systemu nie osiąga nakreślonych parametrów.

Niektóre z prawdopodobnych konsekwencji zastosowania dynamicznych struktur mogą być następujące:

/1/ Pozycje, grupy i wpisy o zmiennej długości, najprawdopodobniej osiągnięte przez rozszerzone definicje danych w słowniku systemu oraz używanie pól sterujących zapisem dla wpisów samodefiniujących się.

/2/ Automatyczne przydzielanie zbiorów o dużej aktywności do przechowywania w pamięciach o dużej szybkości. Skoro częstotliwość wykorzystania spada, zbiór jest przesuwany poprzez hierarchię pamięci do urządzeń o mniejszej szybkości.

/3/ Automatyczne określanie struktur pamięci, albo

poprzez wyspecyfikowane przez użytkownika charakterystyczne cechy oczekiwanego wykorzystania zbioru, albo poprzez pomiar wydajności za pomocą pewnych procedur optymalizacyjnych /np.za pomocą modułu programowania liniowego/ lub też poprzez wykorzystanie obu tych czynników. Automatyczna korekcja struktur pamięci następuje wówczas, gdy zmienia się wydajność.

Wszystko to jest niczym innym niż techniką rozmieszczania zasobów. Tak jak systemy operacyjne trzeciej generacji przydzielają zasoby maszyny, tak uogólnione systemy zarządzania danymi muszą przydzielać zasoby danych i struktury, by złagodzić brzemię, jakie obecnie ciąży na użytkowniku, który zbyt często i tak nie posiada dostatecznego zrozumienia istoty sprawy.

2. Możność przenoszenia

Zagadnienie przenoszenia jest rozpatrywane pod dwoma kątami widzenia: zdolność systemu zarządzania danymi do przenoszenia z jednej maszyny do drugiej lub do wchłaniania nowych urządzeń; oraz stopień zgodności /współwymienności/ zbiorów systemu do przetwarzania przez systemy inne niż system zarządzania danymi /i na odwrót/.

Stąd można mówić o przenoszeniu systemów i - oddzielnie - o możności przenoszenia danych. Istnieje nieuniknione nakładanie się tych zagadnień z trzecim większym obszarem problemowym, systemami sieci, lecz w niniejszym odcinku będziemy usiłowali zajmować się problemami przenoszenia w kontekście wydzielonych ośrodków przetwarzania danych.

John Gosden z MITRE Corporation ostatnio skonstatował, że zostały nam obiecane przez tych, którzy tym samym zobowiązali się do dostarczenia, pewne aspekty " zgodności oprogramowania", określone z definicji jako:

- wspólne płaszczyzny styku człowiek-system - np. zdolność do przesuwania użytkownika, programisty lub operatora od jednego typu komputera do drugiego bez dalszego szkolenia,

- wymiana programów - np. zdolność przenoszenia programu z jednego typu komputera na drugi bez jakichkolwiek zmian,

- wymiana danych - np. zdolność przesyłania danych z jednego programu do drugiego wyłącznie ze zmianami zautomatyzowanymi,

. składanie programów do jednej puli - np. zdolność składania lub łączenia razem różnych podprogramów z różnorodnych źródeł w jeden program,

. składanie danych do jednej puli - np. zdolność łączenia różnorodnych zbiorów źródłowych w zintegrowaną bazę danych.[12]

Zauważa on, że tendencją było obiecanie tylko zgodności "wewnątrz rodziny oprogramowania", gdzie,na przykład,rodziną oprogramowania mogą być "wszystkie programy napisane i zbiory opisane w języku COBOL". W większości wyżej przedstawionych dziedzin dokonano zadawalającego postępu, chociaż trzeba stwierdzić, że obecnie widzimy /i tego nam nie obiecano/ potrzebę zgodności "między rodzinami oprogramowania". Typowym przykładem dla takiej zgodności mogłyby być deklaracje w języku COBOL w zapytaniach do systemu zarządzania danymi, a także program napisany w FORTRANIE, używający zbioru w COBOLU.

Możność przenoszenia systemu. Skrajnym przypadkiem niemożności przenoszenia jest napisanie programów w technice kodowania wyłącznie przykrojonej do jakiegoś szczególnego komputera z daną wielkością pamięci operacyjnej, konfiguracją urządzeń oraz wersją systemu operacyjnego. Programy systemowe

/oprogramowanie systemu/ często zbliżają się do tej skrajności, by wykorzystać specyficzne właściwości jakiegoś modelu /np. struktura przerwań/lub,by uruchamiać jakieś specyficzne urządzenie wejścia-wyjścia lub też,by jak najwydajniej wykorzystać pamięć główną itd. Jednakże oprogramowanie systemowe traktowane jako element samodzielny winno się zbliżać do tej skrajności.

Najwyższa możliwość przenoszenia powinna być reprezentowana przez programy, które są całkowicie niezależne od cech charakterystycznych sprzętu lub od właściwości systemu operacyjnego; oczywiście,tego rodzaju programy w rzeczywistości nie istnieją. Taka zdolność umożliwiałaby programowi bez modyfikacji przetwarzanie na przykład na komputerach PDP-10, IBM 360/44, CDC 3500 oraz na KLUDGE $9^{1/2}$. Wiele przedsiębiorstw obecnie posiada podobnie zróżnicowaną mieszankę podstawowego wyposażenia plus pewien asortyment pamięci wtórnych i urządzeń zewnętrznych - to wszystko w jednym ośrodku przetwarzania danych.Przez powyższą zdolność w pełni ominęłoby się konwersje programów z trzeciej na czwartą generację.

Przyznając, że dla użytkowników pewien stopień możności przenoszenia posiada coraz większe znacze-

nie i to nie tylko dla samego systemu zarządzania danymi lecz dla niemal wszystkich zastosowań. Zbadamy najważniejsze środki do osiągnięcia tego celu: modułowość i uniwersalny język programowania.

Modułowość taka, jaka jest reprezentowana przez hipotetyczny uogólniony system zarządzania danymi z tablicy 4, zmniejsza wpływ zmian na cały system. Izolując i ujmując do jednego "worka" te zbiorowości funkcji systemu, które są zależne od takiego czy innego przymiotu zewnętrznego, uzyskujemy to, że tylko zbiorowisko, którego dotyczą zmiany zewnętrzne, musi być zmodyfikowane, by te zmiany odzwierciedlić. Na przykład moduł przechowywania i wyszukiwania powinien być jedynym modułem, który zajmuje się bezpośrednio fizycznymi urządzeniami pamięci. Wówczas, gdy zachodzi zmiana konfiguracji, ten moduł jest jedynym zestawem programów, który musi być przystosowany do tej zmiany. Moduł procesora języka organizacji danych powinien być tym jedynym modułem, który zajmuje się podanymi przez użytkownika definicjami danych. Nie chodzi tu o to, jakie różnorodne źródła zapewniają te definicje, czy specjalizowane języki użytkownika czy inne programy, nie powinno to wywierać wpływu na inne moduły systemu.

Tak zwany uniwersalny język programowania był wiecznym tematem akademickich debat przez wiele lat. Kulmina-

cja wystąpiła w koncepcji języka UNCOL /UNiversal Computer Oriented Language/ około dziesięciu lat temu.Obecnie istnieje kilka języków, które cieszą się dość rozpowszechnionym użyciem /wśród nich COBOL i FORTRAN/ na wielu różnych maszynach, tak, że zbliżają się one do UNCOL. Jednakże dla każdego języka w każdej maszynie musi być zapewniona specjalna płaszczyzna styku:kompilator.
Z trzech uogólnionych systemów zarządzania danymi,zamierzonych jako niezależne od typu maszyny, dwa były istotnie zakodowane w języku nie na poziomie języka składającego /montującego/, co z pewnością podnosi ich możność przenoszenia w stosunku do trzeciego systemu - z przedstawionych na karcie porównawczej systemów /ten ostatni będzie potrzebował ręcznego przepisywania na każdy nowy język składający/. Z pozostałych siedmiu zależnych od maszyny uogólnionych systemów, trzy zostały napisane w języku nie będącym językiem składającym, przypuszczalnie dla szybszej kompletacji i kontroli, lecz te właśnie trzy mogą być bardziej współwymienne z modelu na model w rodzinie maszyn, niż cztery napisane w języku składającym.

Możność przenoszenia danych. Powszechnym zmartwieniem użytkowników jest zgodność /współwymienność/ zbiorów. "Szeregi" w FORTRANIE i "zbiory" w COBOLU są w obu tych językach zbiorowiskiem danych, lecz program napisany w

jednym języku nie może przetwarzać zbioru wygenerowanego przez program napisany w drugim języku. Ośrcdki obliczeniowe, które w ciągu lat rozwinęły bardzo duże archiwa danych, napotykają na problem zgodności zbiorów za każdym razem, kiedy zmieniają maszyny, systemy operacyjne lub procesory języków. Obecnie ten sam problem kiełkuje w dziedzinie uogólnionych systemów zarządzania danymi - czy będzie można się posługiwać uprzednio wygenerowanymi zbiorami oraz czy będzie on tworzył zbiory odpowiednie dla przetwarzania za pomocą istniejących programów zastosowań?

Z karty porównawczej wynika, że współczesne systemy albo starają się osiągnąć pewien stopień zgodności albo są zamierzone, jako wyłączny pośrednik między wszystkimi programami i wszystkimi zbiorami. Te ostatnie mają prawdopodobnie być docelową funkcją uogólnionych systemów zarządzania danymi, i jest to właściwe, lecz należy znaleźć stosunkowo bezbolesną drogę do osiągnięcia tego. Jeden z pracowników badawczych powiedział: "już istnieje strasznie duża ilość zbiorów na świecie i dlatego nie możemy ignorować ich istnienia i ich użyteczności i przystępować do budowy systemów bazy danych w próżni". Chyba okaże się, że zapewnienie automatycznej konwersji zbiorów w <u>obu</u> kierunkach, jak to zrobiono w niektórych systemach jest najlepszą drogą do osiągnięcia zgodności

/współwymienności/ zbiorów w czasie, gdy zdążamy do osiągnięcia niezależnego od programów celu w zakresie danych.

Ostateczny cel, niezależności danych, wskazuje na potrzebę uniwersalnego języka opisu danych, języka będącego niezależnym od jakiegokolwiek innego. Byłby on bardzo pomocny w rozwiązywaniu zagadnień współwymienności zbiorów, a nawet jeszcze bardziej w rozwiązywaniu problemów związanych z systemami sieciowymi, przedmiotem, który odłożyliśmy na koniec tego rozdziału.

Wnioski. Możność przenoszenia jest cechą w sposób oczywisty pożądaną zarówno dla systemów, jak i dla zbiorów /bardziej ogólnie - dla programów i dla danych/. Nie jest prawdopodobne, by producenci komputerów chcieli dużo inwestować w celu zapewnienia tej cechy, ponieważ może ona spowodować ułatwienie ucieczki do konkurencyjnego sprzętu. Z drugiej strony, producenci oprogramowania stoją na stanowisku rozszerzenia swojego rynku zbytu przez produkowanie przenośnych pakietów, nawet jeśli początkowe wdrożenia są ukierunkowane na serię komputerów wytwarzanych przez jednego producenta. Dlatego można się spodziewać ukazania się większej ilości systemów niezależnych od maszyny pochodzących od wytwórców oprogramowania.

Zdolność przenoszenia i przystosowania systemów można otrzymać w sposób najbardziej wydajny przez stosowanie modułowych składników systemu oraz przez używanie języków wdrażania systemów innych niż indywidualne /dla danego sprzętu/ języki składające.

Możność przenoszenia danych daje się osiągnąć w sposób tymczasowy za pomocą automatycznych procedur konwersji zbiorów dla potrzeb przekazywania zbiorów między systemami lub międzysystemowego przetwarzania. Jak wskazano w pierwszym odcinku niniejszego rozdziału, jesteśmy jednak w etapie przejściowym, który doprowadzi docelowo do niezależności danych od programów, przy uogólnionym systemie zarządzania danymi służącym jako punkt styczny.

3. Sieci i wielorakość

Tak, jak obce urządzenia zewnętrzne tworzą " worek z mieszanką" sprzętu, tak wielorakość tworzy "worek z mieszanką" operowania danymi; wynosi ona małe dotychczas problemy na światło dzienne i bezpośrednio wpływa na problemy organizacji i przenoszenia zbiorów. Konfrontujemy się tutaj z licznymi użytkownikami, co oznacza wspólne korzystanie ze zbiorów, problem szczególnie ostry w środowisku wielodostępu tam, gdzie jest pożądany jednoczesny dostęp, oznacza to także liczne bazy danych, oddzielone funkcjonalnie lub geograficznie, połączone ze sobą lub niezależne.

Naturalnie, najgorszym z dających się wyobrazić przypadków jest sieć wielodostępna z rozdzielonymi bazami danych, z której korzysta wielu użytkowników jednocześnie, wszyscy żądający natychmiastowego dostępu z równie wielu różnych przyczyn. Jest to dosyć interesujące, że coraz większa liczba przedsiębiorstw i sieci usługowych zdąża bezpośrednio w tę strefę konfliktów. Jedno z ostatnich sprawozdań Programu Badawczego Diebold'a omawia zagadnienie wielodostępnego zarządzania danymi /patrz: Oprogramowanie Wielodostępu - Timesharing Software/ tak więc niniejsze sprawozdanie będzie poświęcone tym samym zagadnieniom tylko tak dalece, jak daleko wchodzi w grę uogólniony system zarządzania danymi.
Rozdzielanie tych problemów jest trudne i trochę nierealne, lecz takie podejście ułatwia ich analizę.

Wielorakość użytkowników. Wspólne korzystanie ze zbiorów lub z bazy danych rodzi problemy jednoczesnego dostępu użytkowników celem zapytań i aktualizacji, skojarzeniowych wyszukiwań lub transakcji oraz bezpieczeństwa.

Typowy uogólniony system zarządzania danymi, czy używany w trybie wielodostępu czy w trybie wieloprogramowego przetwarzania partiowego, nie pozwala na jednoczesne występowanie zapytań i aktualizacji jeśli odnoszą się one do tego samego zbioru /w niektórych systemach - do tego samego zapisu/. Zapytania są tak długo przetwarzane aż wystąpi

żądanie aktualizacji; przyjmując, że aktualizacja posiada wyższy priorytet, wyklucza się dalsze zapytania do zbioru lub zapisu, który jest właśnie aktualizowany i ustawia się je w kolejce /możliwe, że robi to system operacyjny/ aż do chwili zakończenia aktualizacji. Uogólniony system zarządzania danymi zorientowany na środowisko przetwarzania partiowego może nie kłaść dużego nacisku na szybkie wyszukiwanie lub aktualizację i zamiast tego preferować kroki zmierzające do maksymalnego wykorzystania pamięci. Uogólniony system zarządzania danymi pracujący w środowisku bezpośredniego przetwarzania na bieżąco musi z konieczności kłaść duży nacisk na szybkie wyszukiwanie i aktualizację, poświęcając wykorzystanie pamięci na korzyść lepszego wykorzystania czasu. Wówczas całe zagadnienie redukuje się do problemu organizacji zbioru.Skojarzeniowe wyszukiwania i transakcje mogą wpaść w konflikt z funkcjami aktualizacji. Rozważmy na przykład, ciąg powiązanych wyszukiwań dokonujących dostępu do pozycji A, B i C, przy czym każda z nich znajduje się w innym zapisie.Użytkownik zażądał pozycji A, i słownik systemu wskazuje automatyczne wyszukiwanie pozycji B i C poprzez łączniki, które zależą odpowiednio od zawartości B i C. System wyszukuje pozycję A, oblicza łącznik do B, i wówczas następuje przerwanie działalności systemu z powodu żądania przez innego użytkownika aktualizacji pozycji A.Następnie

system ponownie podejmuje czynność wyszukiwania umiejscawiając /odnajdując położenie/ pozycję B w oparciu o starą wartość pozycji A i tak dalej. Wynik tego jest taki, że pierwszy użytkownik otrzyma dane, które już nie są w pełni prawidłowe. Istnieją trzy sposoby uporania się z tym konfliktem: nic nie robić, wykluczyć aktualizację aż do zakończenia wyszukiwania lub ponowić wyszukiwanie po zakończeniu aktualizacji. Wybór jednej z tych możliwości powinien zależeć od charakteru zastosowań: na ile krytyczna jest konieczność posiadania w pełni aktualnych danych dla ściśle określonego ich skojarzenia? Użytkownik nie może na bieżąco informować o tym systemu ani sam system nie może tego rozpoznać, gdy wystąpi tego rodzaju przerwanie. Oczywiście potrzebne są pewne prace w tej dziedzinie, by rozwiązać tego rodzaju konflikty i zapewnić sposoby załatwiania ich w przyszłości. Żaden ze znanych uogólnionych systemów zarządzania danymi, które pracują w trybie wieloprogramowości, nie jest zdolny do rozwiązania tego rodzaju konfliktów. Wszystkie te systemy ignorują ich istnienie. Większość uogólnionych systemów zapewnia zabezpieczenie przed nieupoważnionym dostępem do danych. Większość z nich rozróżnia między dostępem w celu wyszukiwania a dostępem celem aktualizacji, rozpoznając różne kody dostępu lub hasła dla każdej z tych funkcji, często dla każdego z czterech poziomów /zbiór, wpis, gru-

pa, pozycja/. Ten rodzaj zabezpieczenia nie jest już problemem. Zabezpieczenie przed utratą danych przez przypadek lub złe funkcjonowanie systemu ma zastosowanie do wszystkich trybów operacji, nie tylko tam,gdzie istnieje wielu użytkowników i jest zwykle zapewniane przez główny system operacyjny czasem uzupełniany przez specjalne procedury zachowania i odzysku danych lub dokumenty kontrolne, które rejestrują łączne zmiany w bazie danych.

Wielorakość baz danych. Jest to zarazem szybko rozwijająca się czołówka stymulująca technicznie, jak i obszar pełen zasadzek. Rozdział traktujący o przykładach przedstawił funkcjonalnie odmienne bazy danych; przedsiębiorstwa także rozdzielają bazy danych geograficznie. Towarzystwa takie,jak np. Sears, biura kredytowe i inne głównie z powodu konieczności zmniejszenia kosztu łączności i prowadzenia sobie z rosnącym wolumenem danych inwentarzowych rozpraszają bazy danych w pewnym obszarze regionalnym lub w całym kraju. Organizacje usługowe podejmują takie same kroki. Na przykład Wydział Ośrodków Danych w Control Data Corporation ogłosił system Cybernet, który już częściowo jest wdrożony. Składa się on z połączonej sieci komputerów CDC 3300, z których każdy jest "węzłem" w sieci; każdy z węzłów jest podłączony do "centroidu" - komputera CDC 6600 - który służy jako "ujście"dla obliczeń lub jako "ujście" dla baz danych.

Zamierzeniem systemu Cybernet jest między innymi zezwalanie użytkownikowi na wejście do systemu przez lokalny węzeł i dostęp do baz danych gdziekolwiek w sieci bez potrzeby posiadania informacji na temat tego, który centroid zawiera potrzebne dane. W celu sprostania takim założeniom system faktycznie wykonuje bardzo dużo funkcji przełączania komunikatów, ponieważ żądania w zakresie przechowywania i wyszukiwania są automatycznie przekazywane do odpowiedniego ujścia danych w celu przetwarzania. Tak więc system nie przesyła właściwego zbioru do węzła, do którego należy użytkownik, by dokonywał on na tym zbiorze manipulacji. To ostatnie rozwiązanie byłoby zbyt kosztowne zarówno w kategoriach czasu, jak i łączy telekomunikacyjnych, a także prowdopodobieństwo wystąpienia błędu byłoby niepokojąco duże.

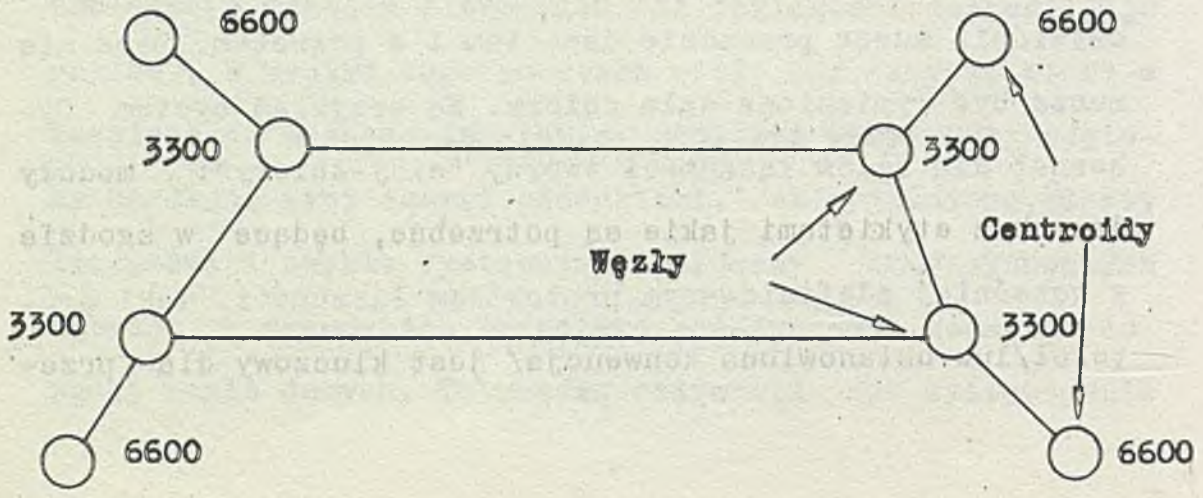
W zastosowaniach handlowych, takich jak system gospodarki zapasami w handlu detalicznym w skali całego kraju, każda wyróżniona geograficznie baza danych służy do kontrolowania zapasów i zamówień dla jakiegoś szczególnego regionu. W wyniku tego powstało wiele baz danych, które w zasadzie są podobne do siebie, ponieważ wszystkie regiony handlują tymi samymi produktami. Jak wyjaśniono, koszty łączności i zwykle występujące wolumeny zmagazynowanych zapasów są argumentem przeciwko pojedynczej scentralizowanej bazie danych. Tu rzeczą oczywistą jest występowanie

nadmiarów danych, które przyjmuje się, jako zło konieczne w tego rodzaju sieci.

Oba powyższe przykłady reprezentują jednolite sieci, jedną z podobnymi komputerami, drugą z podobnymi komputerami i z podobnymi bazami danych. Nie wszystkie przedsiębiorstwa mają szczęście posiadać takie jednolite sieci, które w znacznym stopniu upraszczają wcześniej omówione problemy przenoszenia. Niektóre z przedsiębiorstw nagle odkryły, że posiadają na przykład komputer Burroughs B-5500 w regionie Seattle, komputer GE 435 w Phoenix, a komputer IBM 360/50 w Atlanta - i wszystkie mają być połączone razem, by ułatwić utworzenie proponowanego zintegrowanego systemu informacji kierownictwa, lecz żaden z nich nie umie "mówić" językiem pozostałych. Dla sieci niejednolitych problemy przenoszenia oczywiście się zwielokrotniają.

Sieci z wieloma bazami danych, czy jednolite czy niejednolite, czy umiejscowione czy rozproszone, czasami rzeczywiście muszą przenosić dane tam i z powrotem. Lecz nie muszą być wymieniane całe zbiory. Na przykład system Cybernet dla celów łączności tworzy "niby-zbiory": moduły danych z etykietami jakie są potrzebne, będące w zgodzie z wcześniej zdefiniowanym protokółem łączności. Taki protokół/lub ustanowiona konwencja/ jest kluczowy dla prze-

noszenia danych w sieci i jest od lat praktykowany w wojskowych systemach dowodzenia i sterowania. Wymaga on dwóch elementów: samodefiniujących się modułów danych i słowników opisu danych. Słownik jest takim samym zbiorowiskiem opisów pozycji, grupy, wpisu i zbioru, opisów struktury pamięci oraz poprzednio przedyskutowanych zależności. Jednakże słowniki różnią się od siebie wśród systemów zarządzania danymi; w konsekwencji tego trzeba zbudować pomost dla tłumaczenia opisów danych między słownikami. Tym pomostem jest samodefiniujący się moduł danych. Jak to nasuwa sama nazwa, są to dane z dołączonym swoim własnym opisem /pełnym lub częściowym/. System "A" przygotowuje samodefiniujący się blok danych dla transmisji do systemu"B", przy czym każdy z systemów wykorzystuje dołączony opis dla automatycznej konwersji z innym systemem lub na swoje własne specyficzne wymagania słownika. Oczywiście, dołączony opis musi być rozpoznawany i dający się przetworzyć przez oba systemy, co oznacza stosowanie pewnego rodzaju norm /standardów/, które obecnie jeszcze nie istnieją.



Poprzedzająca dyskusja przedstawia jeden z poziomów uniwersalnego języka opisu danych. Chodzi tu o wcześniej wzmiankowany język dla potrzeb łączności między systemami. Komitet Systemów w CODASYL /Conference on Data SYstems Languages = Konferencja na temat Języków Systemów Danych/ zaproponował, że opis danych może być robiony na co najmniej czterech poziomach:

. opis koncepcyjny /pojęciowy/ dla celu porozumiewania się ludzi /np. stosujący wykresy/, niekoniecznie przewidziany do przetwarzania maszynowego,

. opis logiczny dla łączności człowiek-system, pozwalający użytkownikowi na opisanie struktur danych /np. nazwy i rodzaje pozycji, grupowanie pozycji oraz zależności między grupami/,

. opis fizyczny dla celów łączności człowiek-system, pozwalający użytkownikowi opisać struktury pamięci, /np. sposób przedstawiania długości pozycji, sposób przechowywania/,

. o p i s z a k o d o w a n y dla celów łączności między systemami i wewnątrz systemu, łączący wy-

ciąg w języku maszyny z opisów logicznych i fizycznych z dodatkowymi informacjami, potrzebnymi do przetworzenia ogółu opisanych danych.

Poziom opisu zakodowanego zawiera zarówno słownik, jak i samodefiniujący się moduł danych - jest to poziom, na którym dane przyjmują postać umożliwiającą ich przenoszenie.

Oczekuje się, że systemy powinny zapewnić płaszczyznę styku z jednym sposobem opisu danych dla celów wzajemnej wymiany oraz powinny one zapewnić procesy konwersji w takich elementach styku. Jeśli trzeba opracować odpowiedni element styku, nie będziemy wówczas chcieli normalizować formatów - co byłoby absurdalne - lecz chcielibyśmy znormalizować sposoby opisu formatów.... W dobrze zaprojektowanym systemie użytkownik będzie opisywał dane w formie przewidzianej dla lokalnego użytku, a system powinien ją tłumaczyć na konwencję przyjętą dla wymiany danych. Dla opisu danych /zewnętrznego/ przez człowieka będzie wymagana większa liczba standardów - tyle, ile będzie używanych języków programowania.[13]

Ostatnim zagadnieniem w niniejszej dyskusji na temat wielorakich baz danych jest, czy organizować wzajemnie połączone czy niezależne bazy danych. Zasadniczo zagadnienie to sprowadzasię do możności ustanawiania łączników

między bazami danych dla automatycznych skojarzonych /asocjacyjnych/ wyszukiwań, aktualizacji lub transakcji. Wiele z uogólnionych systemów zarządzania danymi jest w stanie wspomagać wielorakie niezależne, bazy danych, podczas gdy niektóre z nich mogą wspomagać wyłącznie pojedynczą bazę danych. Z tej ostatniej kategorii niektóre systemy mogą wspomagać tylko pojedynczą bazę danych, która nie jest zintegrowana /która nie dopuszcza łączników między zbiorami/. I rzeczywiście, niewiele musi istnieć systemów mogących wspomagać wielorakie wzajemnie połączone bazy danych, ponieważ w czasie sporządzania niniejszego sprawozdania nie napotkaliśmy na żaden tego rodzaju system poza takimi, które wymagają, by w każdym wypadku wszystko zaprogramował użytkownik.

Wydaje się, że istnieje zapotrzebowanie na wzajemnie połączone bazy danych, wspomagane przez uogólniony system zarządzania danymi lub przez sieć takich systemów. Być może, że najsilniejszym argumentem za tym jest tak bardzo dyskutowany system informacji kierownictwa. Z pewnością kierownictwo najwyższego szczebla nie będzie się troszczyć o to, która baza danych zawiera żądaną informację, ani nawet nie będzie tego uwzględniać, że informacja jest rozczłonkowana pomiędzy kilka baz danych; jeśli, na przykład, nie można w łatwy i wygodny sposób otrzymać sprawozdań zbiorczych o wynikach sprzedaży w danym regionie /stosując

skojarzone łączniki między regionalnymi bazami danych, by uformować złożony w jedną całość obraz/, to wówczas taki system nie posiada tak dużej jak powinien przydatności dla potrzeb kierownictwa.

Wnioski. Tam, gdzie istnieje wielu jednoczesnych użytkowników /albo ludzi, albo programów, względnie łącznie/, zapytywanie o informacje z bazy danych i jej aktualizacja nie może się odbywać jednocześnie, chyba że, jak to zapewnia większość systemów, wchodzą w grę niepowiązane ze sobą obszary bazy danych. Jednakże nawet wówczas skojarzone zależności mogą zostać w sposób nieodwracalny naruszone przez aktualizację pozycji, która jest częścią przerwanego powiązanego wyszukiwania lub transakcji.Bezpieczeństwo danych w systemach wielodostępnych będzie zawsze problemem, lecz już nie tak wielkim, jak dotychczas.

Rozproszone bazy danych charakteryzują się jako wydzielone funkcjonalnie albo geograficznie /albo posiadają jednocześnie obie te cechy/ i mogą być jednorodne lub niejednorodne. Możność przenoszenia systemu poprzez całą sieć zapewni jednolitość funkcji, zabezpieczając powtarzalne zdolności zarządzania danymi dla wszystkich baz danych. Możność przenoszenia danych poprzez sieć jest koniecznością chyba, że same bazy danych są całkowicie identyczne /są dokładnymi kopiami jedna drugiej/; lecz taka możliwość wymaga stosowa-

nia jednolitego protokółu łączności między systemami. Z tego wymagania powstała koncepcja standardowego / znormalizowanego/ lub uniwersalnego języka opisu danych. Tego rodzaju język wykorzystywałby samodefiniujące się dane tak, by była możliwa automatyczna konwersja między różnymi formatami i strukturami dla różnych systemów zarządzania danymi.

Ostatecznie może się okazać, że uogólniony system wykazuje braki skoro nie jest w stanie wspomagać wzajemnie połączonych baz danych przez zapewnienie między nimi automatycznych łączników. Łączniki między zbiorami są konieczne dla zintegrowania pojedynczej bazy danych /w znaczeniu tu używanym/ i niewiele systemów zapewnia w sposób zautomatyzowany choćby to. Trzeba jednak stwierdzić, że liczba tego rodzaju systemów wzrasta.

V. OCENA UOGÓLNIONYCH SYSTEMÓW ZARZĄDZANIA DANYMI

A. Wprowadzenie

Każde działające przedsiębiorstwo dnia dzisiejszego już operuje danymi, w większości za pomocą metod ręcznych. Skoro coraz więcej działań przedsiębiorstw opiera się na pracy komputera, to coraz więcej funkcji operowania danymi zostaje zautomatyzowanych. Ten historycznie oczywisty trend rozwojowy w końcu wymusza szereg decyzji umotywowanych pewną liczbą czynników ekonomicznych.

Pierwsza decyzja, którą należy podjąć dotyczy pytania: Czy w ogóle jest potrzebny system zarządzania danymi. Wydaje się, że istnieje pewnego rodzaju próg, poniżej którego nie jest potrzebny żaden system. Być może próg ten jest powiązany z liczbą zbiorów potrzebnych dla danego zastosowania lub ze złożonością zbiorów lub z liczbą programów, które współdziałają z pewnym zestawem wspólnych danych. Skoro już raz z jakiejkolwiek przyczyny zostanie przekroczony ten próg, to wówczas potrzeba stosowania pewnego rodzaju systemu zarządzania danymi staje się oczywista. W związku z tym następną decyzją, jaką należy podjąć, będzie określanie rodzaju tego systemu.

W dyskusjach na temat ewolucji systemów zarządzania danymi dawniej sugerowano istnienie ciągłości w czasie. "Na początku" istniała wspólna procedura wprowadzania-wyprowadzania, która eliminowała zbyteczne kodowanie w zakresie wprowadzania-wyprowadzania w każdym z programów. Te procedury były tworzone wokół parametrów, które pozwalały uzyskać uniwersalność. Wyniki posiadały charakter zdecydowanie ekonomiczny: bardziej wydajne wykorzystanie czasu programisty. Z tego niewinnego początku rozwinął się pewien rodzaj naturalnego procesu selekcji w zakresie opracowania coraz bardziej efektywnych narzędzi programowania. Niektóre z przedsiębiorstw opóźniały się tymczasowo lub stale w różny sposób na tej drodze ewolucji. Być może, że niektóre z przedsiębiorstw nie wymagają więcej niż wspólnych procedur wprowadzania-wyprowadzania, by zadowolić swoje potrzeby w zakresie zarządzania danymi; inne być może zadawalają się systemami zarządzania zbiorami; z kolei jeszcze inne być może korzystają z systemów wyszukiwania informacji i systemów sprawozdawczych. Niniejszy rozdział koncentruje się głównie na zagadnieniach, na które natknęły się przedsiębiorstwa stojące przed koniecznością podjęcia drugiej z kolei decyzji, a mianowicie jaki rodzaj systemu należy wybrać. Konieczność podjęcia tej decyzji doprowadziła wspomniane przedsiębiorstwa do przejrzenia pełnego zakresu współczesnych możliwości w dziedzinie operowania danymi.

Następna decyzja, którą trzeba podjąć, jest bardzo złożona. Podjęcie jej wymaga dokonania porównawczej oceny dostępnych systemów, ze szczególnym uwzględnieniem zdolności funkcjonalnych /w jakim stopniu każdy z systemów spełnia obowiązkowe wymagania oraz pożądane cechy charakterystyczne/; kosztów /koszt początkowy, koszty eksploatacji, wydajności/; wykorzystania / przewidywani użytkownicy, łatwość korzystania, elastyczność/ oraz innych względów /przewidywany okres życia, zdolność przystosowania się, obsługa/. Decyzja ta wymaga opinii czy dążyć do kompromisu, czy go uwzględnić oraz co jest korzystniejsze - oraz opracowanie wewnętrzne, oprącowanie zlecone w formie kontraktu /umowy/, czy bezpośredni zakup gotowego systemu.

Temat: przedmiot oceny porównawczej należy omówić szczegółowo w zakresie zdefiniowania krytycznych momentów. Istotne jest otrzymanie faktycznych liczb dotyczących wydajności każdego z rozważanych systemów. Tu najlepszą radą jest, jak to mówią ludzie z Buick Division w koncernie General Motors: "Pytać tych, którzy posiadają /użytkują/ taki system". Przyczyną jest fakt, że dostawcy uogólnionych systemów zarządzania danymi niechętnie mówią o ograniczeniach, brakach i złych wynikach swoich własnych systemów, natomiast chętnie omawiają w ciepłych słowach, rzekome mocne strony oferowa-

nego systemu /nawet jeśli nie są jeszcze wdrożone/. Inaczej mówiąc, dokonujący oceny powinni korzystać ze specyfikacji systemu jako wskaźnika wydajności tylko wówczas, gdy nie można uzyskać informacji na drodze doświadczalnej i jeśli się korzysta z takich specyfikacji, to należy to robić z dużą ostrożnością.

Dalszym krytycznym problemem jest sprawa ustalenia wielkości faktycznych potrzeb. Może nie być łatwą sprawą dla przedsiębiorstwa wyszczególnienie własnych specyficznych wymagań w zakresie operowania danymi i trzeba tu powiedzieć, że przeprowadzenie jakiejkolwiek oceny porównawczej bez pełnej znajomości wewnętrznych potrzeb nie jest niczym więcej niż akademickim ćwiczeniem. Nawet jest lepiej jeśli wymagania są stopniowane / zaznaczone jako obowiązkowe lub pożądane/, uszeregowane w porządku swej ważności lub tak wyważone, by ważnym wymaganiom nadać większy wpływ w porównaniu składającym się z " sumy pomiarów". Drugim możliwym podejściem jest pomiar zdolności systemu w stosunku do teoretycznego ideału, który niestety może wykazywać mało punktów odniesienia w stosunku do danych potrzeb przedsiębiorstwa. Opublikowane przeglądy uogólnionych systemów są pomocne jako źródło informacji /zależy to od wiarogodności takiego źródła informacji/ lub w zrozumieniu cech poszczególnych systemów lecz niestety zwykle można je zaliczyć do kategorii ćwi-

czeń akademickich ponieważ w dalszym ciągu muszą być interpretowane pod kątem widzenia specyficznych potrzeb. /Ćwiczenia akademickie mogą oczywiście służyć pożytecznym celom, takim jak szerzenie terminologii i technologii wśród większej zbiorowości/.

Mówiąc krótko, nie ma łatwego i dobrze utrafionego uniwersalnego podejścia do oceny uogólnionych systemów zarządzania danymi. Każde przedsiębiorstwo musi przebrnąć przez tę procedurę indywidualnej oceny swoich własnych potrzeb i być może napotka przy tym na duże trudności. Wysoka potencjalna opłacalność oraz przewidywane inwestycje usprawiedliwiają rozległą ocenę wielu różnych systemów; mały budżet lub mała potencjalna opłacalność prawdopodobnie spowoduje konieczność ograniczenia tego przeglądu do niewielkiej liczby proponowanych systemów. W każdym razie bez względu na zakres, każde przedsiębiorstwo powinno przebadać wchodzące w grę przedstawione w tym rozdziale najważniejsze zagadnienia w takim rozmiarze, na jaki pozwalają dysponowane środki.

W dalszym ciągu niniejszego rozdziału wydzielimy te czynniki, które na ogół mają największy wpływ na wynik omawianej oceny porównawczej. Przedsiębiorstwa, które chcą otrzymać szczegółową informację na temat ściśle określonych możliwości poszczególnych uogólnionych systemów za-

rządzania danymi, mogą je otrzymać w dziedzinie publikacji, które zawierają kilka bardzo obszernych sprawozdań przeglądowych /chociaż trzeba stwierdzić, że jest tam zawarte mało informacji na temat faktycznej wydajności tych systemów/: patrz np. bibliografia, pozycje 7 i 18. Przedsiębiorstwa, które w podejściu do oceny porównawczej chcą postępować według pewnego wzorca, mogą znaleźć dobry przykład w bibliografii pod pozycją 12. Czytelnicy, którzy są zainteresowani w zorganizowaniu wypunktowanego wykazu wewnętrznych potrzeb, w rozbiciu na kategorie potrzeb natychmiastowych, pożądanych i dalszoplanowych, powinni się zapoznać z publikacją wymienioną w pozycji 22 bibliografii. Ta ostatnia pozycja, będąca faktycznie zaproszeniem do złożenia ofert skierowanym do dostawców, by dowiedzieć się czy taki system mogłby być zbudowany dla klienta, jest doskonałym przykładem jednego ze sposobów sporządzania oceny porównawczej.

B. Wzorcowe przykłady ocen przeprowadzonych przez przedsiębiorstwa

Bieżący odcinek przedstawia dwa rodzaje podejścia do zagadnienia oceny, które były stosowane w przedsiębiorstwach. Naszym zamierzeniem jest przedstawienie czytelnikowi, które parametry były w trakcie oceny uważane za

krytyczne, a nie który z systemów okazał się w wyniku oceny lepszy. W następstwie tego omawiane dwa przypadki w zasadzie zawierają wykaz cech, które były porównywane dla wchodzących w grę systemów. Użyto tu terminologii takiej samej jak w przedmiotowych przedsiębiorstwach, nie dokonywano żadnych przeróbek na terminologię stosowaną w naszym sprawozdaniu. Właściwości oznaczone gwiazdką /x/ omówiono w następnym odcinku.

1. Przedsiębiorstwo A: filia producenta samolotów

Omawiane przedsiębiorstwo wybrało dla celów porównania 25 cech systemu, a ostateczny swój wybór ograniczyło do 12 systemów. Każdej z tych cech, jak widać, przypisano skalę oceny, przy czym liczbie "1" przypisano najwyższą ocenę, a wyższe liczby oznaczają niższy priorytet, z wyjątkiem pewnych parametrów, które mogą być tylko ocenione w skali "1" lub "2": albo system posiada daną cechę, albo jej nie posiada. Nie wykazano wagi parametrów, ani też przedstawiony porządek nie wskazuje na ich względną ważność.

x Niezależność od sprzętu	/1 do 3/
x Szkolenie programistów	/1 do 3/
x Szkolenie nieprogramistów	/1 do 4/
Dozwolone przetwarzanie partiowe	/1 lub 2/

Dozwolone przetwarzanie bezpośrednie	/1 lub 2/
Elastyczność w projektowaniu zbiorów	/1 do 7/
xWyszukiwanie zawartości /samorzutne/	/1 do 5/
Wyszukiwanie zaprogramowane uprzednio	/1 do 3/
Bezpośrednia aktualizacja	/1 lub 2/
xAktualizacja samorzutna	/1 do 3/
Niezależne definiowanie danych	/1 lub 2/
Wyszukiwanie wg kluczy wtórnych	/1 do 3/
Przetwarzanie wybranych części baz danych	/1 do 5/
Bardzo wielkie i złożone zbiory	/1 do 5/
Przeróbka struktury	/1 do 5/
xBezpieczeństwo	/1 do 5/
Odzyskiwanie zapisów i bazy danych	/1 do 5/
Sprawozdawczość systemowa /logging/	/1 do 3/
Ilość wybudowanych funkcji przetwarzania d.	/1 do 4/
Eksploatacja wieloprogramowości	/1 lub 2/
Dostępność	/1 do 4/
xMożność przykrojenia do specyficznych potrzeb	/1 do 5/
Zapewnienie utrzymania systemu	/1 do 4/
Zakres możliwości zdalnego przetwarzania	/1 do 5/
xKoszty wdrożenia i eksploatacji	/ - - /

Po dokonaniu oceny przedsiębiorstwo wysnuło wniosek, że żaden z omawianych 12 systemów nie zapewnia wystarczająco dobrego dopasowania do postawionych wymagań, lecz jeden z systemów był na tyle bliski spełnieniu tych wymagań, że wybrano go celem przeprowadzenia modyfikacji

2. Przedsiębiorstwo B: filia producenta przetworów naftowych

To przedsiębiorstwo wybrało 37 cech jako bazę dla porównania pięciu "półfinałowych" systemów. Każda z cech była po prostu zapisywana dla każdego z systemów "tak"lub "nie" i najlepszy wynik był ten, który posiadał największą liczbę odpowiedzi twierdzących "tak". Także każda z właściwości posiadała oznaczenie, czy jest traktowana jako obowiązkowa, czy jako pożądana, co przedstawiono odpowiednio jako /O/ i /P/ - patrz niżej. Nie używano żadnej innej klasyfikacji lub stopniowania, lecz właściwości systemów pogrupowano w pięć kategorii logicznych.

Języki użytkownika

System pozwala na łatwą specyfikację pracy /P/

Może pracować wielozadaniowo /P/

Posiada język zbliżony do angielskiego /P/

Posiada szeroką logikę przeszukiwania /O/

xMoże integrować inne przetwarzanie /P/

Posiada opcjonalnie format automatycznego wyprowadzania /O/

Posiada opcjonalne specyfikowanie formatu /O/

Pozwala na łatwą modyfikację zbioru /P/

Struktura zbioru

Używa skorowidzów /P/

Używa nieliniowych technik wyszukiwania /O/

Używa techniki indeksowania /P/

Pozwala na indeksowanie wybiorcze /P/

Pozwala na stałe i zmienne formaty /O/

Może posługiwać się podzbiorami zwrotnymi /recursive subsets/ /P/

Dopuszcza wielorakie zbiory /P/

xPozwala na stosowanie odsyłaczy /P/

Możliwości w zakresie przetwarzania

Używa operatorów logicznych /O/

Używa operatorów określających wzajemne zależności /O/

Dopuszcza operatory algebraiczne /P/

Wykonuje utrzymanie zbiorów /O/

Pozwala na przetwarzanie partiowe /O/

Pozwala na przetwarzanie swobodne /P/

Posiada elastyczne generatory wyjścia /O/

Sortowanie/O/

Pozwala na streszczenie, statystykę /P/

Pozwala na sterowanie przebiegiem przez użytkownika /O/

Zapewnione centralne We /Wy/O/

Zdolność akumulowania przebiegów /P/

Zdolność łatwego akumulowania nowych przebiegów /P/

Pozwala na przypisywanie priorytetów /P/

Płaszczyzna styku z programistą

xModuły łączące z językami proceduralnymi /P/

Stosuje bibliotekę systemową /O/

Automatycznie utrzymuje skorowidze /P/

Czas odpowiedzi

xPrzetwarza zbiory na taśmie /O/

Przetwarza zbiory na dyskach /P/

Stosuje techniki swobodnego dostępu /P/

Przedsiębiorstwo doszło do wniosku, że z szeregu przyczyn ekonomicznych i środowiskowych system, który zajął drugie miejsce, był w istocie lepszy, niż system zakwalifikowany jako pierwszy.

C. Analiza wzorcowych ocen

Obecnie przystąpimy do krytycznej analizy przed chwilą przedstawionych reprezentatywnych czynników oceny, z zamiarem określenia, które z nich są naprawdę ważne, a które są błahe lub nie przyczyniają się do ulepszenia systemu. Powyższe dwa wykazy czynników zostaną przeanalizowane w porządku ich występowania, z ominięciem jedynie tych czynników, które są wspólne dla niemal wszystkich współczesnych uogólnionych systemów zarządzania danymi już z samej ich definicji /niektóre z nich w powiązaniu z gospodarującym systemem operacyjnym/. Jasna sprawa, że jeśli niektóre z czynników już nie są użyteczne w rozróżnianiu systemów, nie będą one uwzględnione w procesie oceny porównawczej, chociaż mogą być użyteczne we wstępnym naświetleniu zagadnienia.

1. Czynniki przedsiębiorstwa A

Niezależność od sprzętu. Dla większości użytkowników /tj. użytkowników IBM/ jest to rozważanie raczej teore-

tyczne, ponieważ nie potrzebują oni możności przenoszenia poprzez typoszeregi maszyn /co nie jest tym samym co zgodność w górę/ jeszcze przez długi okres czasu. Stąd możność przenoszenia systemów w chwili obecnej musi być traktowana jako cecha pożądana, a nie jako natychmiastowa konieczność. Znacznie bardziej krytyczna i ważna już w tej chwili jest niezależność od sprzętu w zakresie różnorodnych środków przechowywania informacji.

Szkolenie programistów. Nie posiada ono znaczenia, jeśli uogólniony system zarządzania danymi jest zaprojektowany dla użytku przez nieprogramistów. Co więcej, nawet wówczas, gdy uogólniony system zarządzania danymi jest przewidziany do użytku przez programistów szkolenie programistów prawdę mówiąc nie ma dużego znaczenia ponieważ tego rodzaju systemy posiadają zazwyczej płaszczyznę styku z powszechnie stosowanymi językami programowania i należy się tylko nauczyć rozkazów systemowych, a dla tego celu będzie służyć odpowiednia dokumentacja.

Szkolenie nieprogramistów. Problem ten nie posiada żadnego znaczenia, jeśli uogólniony system zarządzania danymi został zaprojektowany do użytku przez programistów, a posiada jedynie niewielkie znaczenie w systemach zaprojektowanych dla użytku przez nieprogramistów, ponieważ takie systemy zazwyczaj są zaprojektowane pod kątem

"łatwego z nich korzystania". Być może bardziej prawidłowym kryterium dla celów porównawczych byłby zakres, w jakim system poucza użytkownika, zapewniając pomoc w trybie wzajemnego oddziaływania. Chodzi tu o pomoc dla użytkownika w zakresie nowych aspektów korzystania z systemu.

Wyszukiwanie zawartości /samorzutne/. Jest to bardzo kosztowna cecha, jeśli mamy na myśli pozycje adresowane według ich zawartości. Wówczas wchodzą w grę urządzenia pamięciowe o dostępie równoległym, które omówiliśmy w poprzednim rozdziale, chyba że czas wyszukiwania nie posiada znaczenia. Poszukiwanie nierównoległe wszystkich pozycji posiadających zadaną wartość jest procesem czasochłonnym. Czynnik ten jest bardziej zależny od cech technicznych sprzętu niż od techniki oprogramowania. Najbliższą analogią w dziedzinie oprogramowania jest zdolność omawiana w niniejszym sprawozdaniu jako wyszukiwanie asocjacyjne /skojarzeniowe/: określenie zależności strukturalnych lub wzajemnych zależności między danymi innych, niż formalne skojarzenia hierarchiczne. Dlatego ogólne problemy wyszukiwania według zawartości zostają odniesione do dziedziny przyszłych opracowań w zakresie sprzętu i oprogramowania. Jest to rzecz wykonalna, lecz z powodu dużych trudności mało realna. Jako prawidłowy i dający się pomierzyć czynnik pozostaje wyszukiwanie asocjacyjne/skojarzeniowe/.

Samorzutna aktualizacja. Jest kosztowna, jednakże także warta poważnego potraktowania. Jest to przełączona przez system aktualizacja /np. w systemie GIM aktualizacja"współzależna"/ lub skojarzeniowa /asocjacyjna/ transakcja, która automatycznie aktualizuje powiązane pozycje, gdy użytkownik aktualizuje pierwszą z nich. Działanie tego rodzaju aktualizacji wygląda jak rozchodzenia się "fali",jak uderzenie w szereg ustawiony z tabliczek do gry w domino. Jest to z pewnością czynnik decydujący.

Bezpieczeństwo. Czynnik ten nie ma dużego znaczenia,gdyż niemal wszystkie systemy zapewniają pewną formę zabezpieczenia, jakiego użytku zastosowano tę właściwość. System może paść ofiarą sprzeniewierzenia; żywotnymi środkami zabezpieczenia danych są sposoby i procedury administracyjne obmyślane dla właściwej kontroli dostępu do nich. Tutaj należy zaznaczyć, że każde przedsiębiorstwo powinno nie tylko ponownie ocenić jaki poziom i rodzaj zabezpieczenia jest faktycznie potrzebny, lecz powinno także zbadać logiczne i fizyczne aspekty kontrolowanego dostępu. Co się dzieje z każdą "drażliwą" wydrukowaną informacją? /Prawdopodobnie kładzie się ją na półkę w czyimś biurku lub bibliotece/. A przede wszystkim dlaczego jest ona poufna? Jeśli jest ona tak dalece poufna, to czy rzeczywiście musi się znajdować w skomputeryzowanej bazie danych?

Możność przykrojenia do specyficznych potrzeb. Jeśli oznacza ona utratę uogólnionych zdolności to powinna być całkowicie odrzucona. Z drugiej strony, jeśli oznacza ona możność przystosowania się systemu do nowych zastosowań, zmian w sprzęcie lub w innym oprogramowaniu systemowym, zmian w trybie wykorzystania /np. przejście od przetwarzania partiowego do wielodostępu/ lub do innych form zmiany, to wówczas jest to naprawdę ważny czynnik. Jest niemal nieprawdopodobne, by raz zainstalowany system nie był przedmiotem zmian. Modułowość systemu jest najlepszym zabezpieczeniem dla przedsiębiorstwa w tym, by uczynić system w większym stopniu przenośnym, łatwiejszym do obsługi, łatwiejszym do rozszerzenia. Przedłuży mu ona także okres użytkowania.

Koszty wdrażania i eksploatacji. Koszty wdrażania mają znaczenie w ciągu krótkiego okresu czasu. Wchodzą tu: koszt nabycia systemu i jego wybranych opcjonalnych cech /lub ich opracowanie/, koszt instalacji systemu, przyrost kosztu dla dodatkowych kopii systemu i ich instalacji, koszty dokumentacji i przeszkolenia /włączając w to czas szkolenia/, koszty dodatkowego wyposażenia lub urządzeń wymaganych przez system oraz koszty poniesione przez opóźnienie w dostawie lub instalacji, które odraczają korzystanie z systemu. Koszty eksploatacji mają dalej się-

gające następstwa; są one krytyczne lecz niezmiernie trudne do uprzedniego oszacowania. Oczywistymi elementami są tutaj: ciągłe szkolenie, kontrakty na usługi oraz potrzeby w zakresie personelu dla administracji oraz być może obsługi bazy danych. Inne koszty, mało uchwytne lub ukryte, mogą być w przybliżeniu oszacowane w okresie eksploatacji próbnej lub za pomocą szeregu prób przeprowadzonych metodą komparatystyczną/benchmark tests/ dla porównania takich okoliczności, jak czasy odpowiedzi i czas potrzebny do zdefiniowania, załadowania, aktualizacji lub przebudowy struktury zbioru.

2. Czynniki przedsiębiorstwa B

Może integrować inne przetwarzanie. Oznacza to zdolność integrowania wyjścia informacyjnego z innymi funkcjami przetwarzania. Nie posiada to zwykle znaczenia jeśli chodzi o zdolność tworzenia zewnętrznego modułu styku. Inne interpretacje są niejasne, chociaż jedną z możliwych jest automatyczne przełączanie funkcji asocjacyjnych /kojarzących/, które mogłyby być bardzo ważne.

Pozwala na stosowanie odsyłaczy. Jest to zdolność łączenia pozycji w hierarchicznej strukturze danych. Wszystkie systemy pozwalają na stosowanie struktur hierarchi-

cznych w postaci tzw. czystego drzewka co najmniej jednopoziomowego. Niektóre z nich pozwalają na stosowania bardziej złożonych struktur, sięgających od drzewek odwróconych do sieci. Zwykle nie ma dużego znaczenia liczba poziomów; znaczenie ma zdolność specyfikowania zależności wewnętrznych. Stąd, jeśli istnieje potrzeba wiązania pozycji z innymi pozycjami wewnątrz lub na zewnątrz grupy macierzystej /albo na tym samym poziomie albo między poziomami/, należy uwzględniać tylko te systemy, które pozwalają na stosowanie bardziej złożonych skojarzeń niż typu czystego drzewka. Zakłada to konieczność dysponowania co najmniej częściową zdolnością tworzenia sieci.

Moduły łączące z językami proceduralnymi. Ta cecha funkcjonalna ma znaczenie tylko wówczas, gdy z systemu będą korzystać programiści. Krytycznym zagadnieniem jest, czy system posiada płaszczyzny styku z odpowiednimi językami dla wszystkich jego użytkowników. Nieprogramiści /analitycy, technicy, urzędnicy, kierownicy/ wymagają jednego lub więcej języków specjalnie zaprojektowanych dla wzajemnego oddziaływania w układzie użytkownik-system. Ważną cechą języków dla użytkowników jest zdolność stosowania synonimów /przy założeniu wyszukanego poziomu możliwości słowotwórczych/, zdolność wertowania nieznanych danych /przy założeniu istnienia pomocy w zakresie konwer-

sacji i nauczania przez system/ oraz stosunkowe bogactwo języka - pozwalające na tworzenie złożonych raportów z automatycznie sporządzanymi tytułami, stronnicowaniem i tak dalej, za pomocą względnie prostych rozkazów.

Przetwarza zbiory na taśmie. Jest osobliwe, że wiele skądinąd skomplikowanych systemów zostało przewidzianych do wyłącznej pracy na urządzeniach o bezpośrednim dostępie. Skoro taśmy magnetyczne tworzą część puli środków przechowywaniá informacji, to wówczas system powinien wspomagać struktury pamięci odpowiednie dla całej hierarchii pamięci. Jest to więc czynnik krytyczny.

D. Streszczenie porównawczych czynników oceny

Powyżej omówione dwa wykazy oraz niektóre zagadnienia przedstawione w poprzednim rozdziale, dają narzędzie do wstępnego zakwalifikowania systemu, jako uogólnionego systemu zarządzania danymi, a także wyznaczają kilka najważniejszych czynników pomocnych w dalszych porównaniach. Podczas wstępnej selekcji przesiewa się "kandydatów" na podstawie porównania z cechami charakterystycznymi i własnościami uogólnionych systemów zarządzania danymi, które opisano w dwóch pierwszych odcinkach rozdziału IV.

W sumie istnieje dziesięć głównych czynników porównawczych i można je następująco przedstawić w streszczeniu:

możliwość przystosowania - W jakim zakresie każdy z systemów posiada budowę modułową? Jakie powstaną trudności przy przystosowaniu systemu do nowego sprzętu, nowych zastosowań itd.? /Czynnik zmienny/

Tworzenie odsyłaczy - Czy system pozwala na stosowanie łączników między zbiorami i w zbiorach? Czy zapewnia w sposób automatyczny wyszukiwania i transakcje asocjacyjne /skojarzeniowe/? /W zasadzie czynnik dwoisty/

zageszczanie danych - W jakim stopniu system automatycznie redaguje dane oraz uzwięźla struktury zapisów? /Czynnik zmienny/.

wydatki - Jaki jest koszt początkowego wdrożenia? A co ważniejsze, jakie będą koszty eksploatacji? /Czynnik zmienny/

organizacja zbioru - Czy system posiada dużą elastyczność w stosowanych rodzajach struktury pamięci i metod dostępu? /Czynnik zmienny/

niezależność od sprzętu - Czy system można "łatwo" przenieść na inny typ maszyny lub z jednej generacji na drugą? /Czynnik zmienny w długim okresie czasu/

niezależność od sprzętu - Czy system wspomaga całą hierarchię pamięci? Czy może on przydzielać zbiory na taśmy, dyski, bębny oraz na masowe pamięci ferrytowej/Czynnik dwoisty/

wielorakie bazy danych - Czy system wspomaga wielorakie bazy danych dla wielorakich zastosowań? /Czynnik dwoisty/

pomoc w nauczaniu - W jakim stopniu system zapewnia pomoc użytkownikowi? /Czynnik zmienny/

styk z użytkownikiem - Czy system zapewnia niezbędne języki /lub elementy podłączenia się/dla przewidywanego wachlarza użytkowników? /W zasadzie czynnik dwoisty/

Nie powtórzono tutaj pełnego zakresu czynników wstępnej selekcji lecz składają się one z takich cech charakterystycznych, jak:

- zdolność tworzenia "procedur obliczeniowych"dla powtarzalnego użytku,
- zdolność formowania złożonych logicznych pytań,
- zdolność przeszukiwania zbioru według więcej niż jednego klucza,
- zdolność wspomagania złożonych struktur danych,
- zdolność definiowania, tworzenia i manipulowania zbiorami,
- zdolność tworzenia raportów,
- zdolność przechowywania definicji danych oddzielnie od zarówno danych jak i programów,
- zdolność dodawania, zamieszdzania, modyfikowania lub usuwania wartości danych,

. zdolność weryfikacji wartości pozycji wprowadzanych,

. zdolność /poprzez współpracę z systemem operacyjnym/ działania w różnych trybach - przetwarzania partiowego, zdalnego itd.

Niektóre z czynników, wskazujące na ograniczenia systemu lub powszechnie uznane jako typowe, po prostu nie posiadają znaczenia w ocenie porównawczej. Mogą one być interesujące same przez się, lecz nie mają wpływu na wybór systemu. Ponieważ przykłady powinny posłużyć jako poparcie tego punktu widzenia.

1. Liczba pozycji w grupie /maksymalna/ - Czy stanowi różnicę, jeśli teoretycznie nie ma ograniczeń w tym względzie lub gdy granicą jest $2^{24}-1$?

2. Brakujące grupy - odnosi się to do możności definiowania logicznej struktury zbioru/wpisu/bez przydziału pamięci dla grup, które dotychczas nie posiadają wartości. Każdy z przebadanych uogólnionych systemów zarządzania danymi posiada tę zdolność; bez niej system musiałby rezerwować pamięć dla całego zbioru - wielkość sztywnie ustaloną - w momencie jego definiowania.

3. Maksymalna liczba wyrażeń logicznych połączonych za pomocą operatorów I, LUB, NIE w pojedynczą deklarację wyszukiwania. - W większości systemów nie określono tutaj teoretycznych ograniczeń; niektóre określają dużą, lęcz skończoną granicę w tym zakresie /np. GIS dopuszcza maksymalnie 253 wyrażeń/.Jaką mogłoby to stanowić różnicę? Użytkownik zaplątałby się całkowicie we własnych pytaniach, zanim by osiągnął ograniczenie postawione przez system.

4. Zapewnienie operatorów określających wzajemne zależności - Każdy system zapewnia w pewnej formie sześć podstawowych operatorów: równe,nierówne, większe niż, mniejsze niż, większe lub równe, mniejsze lub równe /ulepszone systemy języków proceduralnych korzystają z właściwości języka gospodarującego/. Ostatecznie, selektywne wyszukiwanie lub aktualizacja byłyby raczej trudne bez tych operatorów.

5. Maksymalna liczba zbiorów na jedną bazę danych - Tu znów większość systemów nie określa teoretycznych granic. System TDMS dozwala tylko jeden zbiór, co zrównuje zbiór z bazą danych.

Z tym tylko jednym wyjątkiem /faktycznie jest to kwestia nomenklatury/ czynnik ten nie posiada istotnego znaczenia.

6. Poziomy zawartości - odnosi się to do liczby zawartych jednych w drugich poziomów wyrażeń logicznych - stworzonych przez stosowanie nawiasów w celu zmiany logiki założonej przez nie zmienioną w inny sposób sekwencję operatorów. Niektóre systemy dopuszczają nieograniczoną liczbę poziomów. Inne posiadają ograniczenia w tym zakresie. /9 poziomów dla systemu MARK IV, 4 - dla systemu GIS/. Czynnik ten nie przyczynia się do oceny porównawczej w niczym innym, jak tylko we wskazaniu stopnia potencjalnego zamieszania, jakie grozi użytkownikowi.

Podsumowując należy stwierdzić, że procedura porównawczej oceny uogólnionych systemów zarządzania danymi powinna wymagać: 1/ określenia zamierzonego celu systemu, jego wykorzystania i użytkowników oraz określenia środowiska, w którym musi działać; 2/ wstępnej selekcji dostępnych systemów w celu przekonania się, które będzie można zakwalifikować jako zapewniające niezbędne

podstąwowe funkcje oraz 3/ porównania i przeciwstawienia zakwalifikowanych systemów w zakresie tych cech charakterystycznych, które się naprawdę wśród nich wyróżniają. Pierwszy krok poczyniony na tym polu będzie w dużym stopniu wskazywął czynniki, jakie mają być zawarte w drugim i trzecim etapie oceny.

WYKAZ ODSYŁACZY

/1/ Odwołanie się do pozycji 5 w bibliografii, strony 123-124.

/2/ Odwołanie się do pozycji 10 w bibliografii, grudzień 1967, strona 1.

/3/ Odwołanie się do pozycji 4 w bibliografii, strony 267-268.

/4/ Ten przykład jest częściowo oparty na pozycji 2 w bibliografii.

/5/ Ten przykład jest częściowo oparty na pozycji 1e w bibliografii.

/6/ Ten przykład jest częściowo oparty na pozycji 1h w bibliografii.

/7/ Ten przykład jest częściowo oparty na pozycji 1g w bibliografii.

/8/ GIM jest nazwą handlową prawie zastrzeżoną przez przedsiębiorstwo TRW Systems Group. Ten przykład jest częściowo oparty na pozycjach 1c i 21 w bibliografii.

/9/ Odwołanie się do pozycji 19 w bibliografii, strona 1.

/10/ Odwołanie się do pozycji 5 w bibliografii, strona 112.

/11/ Odwołanie się do pozycji 5 w bibliografii, strona 236-237.

/12/ Odwołanie się do pozycji 1f w bibliografii, strona 81.

/13/ Odwołanie się do pozycji 1f w bibliografii, strony 85-86.

BIBLIOGRAFIA

1. American Federation of Information Processing.Societies. Joint Computer Conference Proceedings. Vols. 31 and 33. Washington, D.C.: Thompson, 1967 and 1968.

2. American Management Association. Common Data Base Development for Information Systems of the Seventies. Notebook distributed at Administrative Services Briefing Session No. 6307-01, New York, March 12-14, 1969.

3. Auerbach Corporation. Associative Memory Investigations: Substructuring Searching and Data Organizations.Springfield, Va.: Clearinghouse for Fereral Scientific and Technical Information, November 1968 /AD-679-227/.

4. Becker, Joseph and Robert M. Hayes. Information Storage and Retrieval: Tools, Elements, Theories. New York: Wiley, 1963.

5. Climenson, W.Douglas , "File Organization and Search Techniques" In: Annulal Review of Information Science and Technology. New York:Wiley,1966. Vol.I,pp.107-135.

6. Conference on Data Systems Languages. Cobol Extensions to Handle Data Bases. Report to CODASYL prepared by the Data Base Task Group. New York: Association for Computing Machinery, January 1968.

7. _____. Systems Committee. A Survey of Generalized Data Base Management Systems, New York: Association for Computing Machinery. May 1969.

8. The Diebold Group, Inc. The Diebold Research Program. New York. Data Base: Powerful New Corporate Competitive Weapon. Applied Technology Research Studies, Technology Series /T7/ May 1967. Software Evaluation II. Nonmanufacturer Software Packages. Systems Methodology Studies, Software and Methodology Series /S7/, December 1968. Timesharing Software. Applied Technology Research Studies, Technology Series /TB20/ December 1968.

9. Dodd, George G. "Elements of Data Management Systems" ACM Computing Surveys. June 1969. pp. 117-133

10. EDP Analyzer December 1967, January 1968, and February 1968 issues. Wista, Calif., Canning Publications.

11. Eppstein, Anthony D. "The Technology of Disk Data Storage" Data Processing, September 1968, pp. 26-32.

12. Informatics, Inc. A Methodology for Comparison of Generalized Data Management Systems: PRGS /Parametric Evaluation of Generalized Systems/. By Anthony J. Dowkant, William A.Morris, and T.Dwight Buettell. Springfield Va.: Clearinghouse for Federal Scientific and Technical Information, March 1967 /AD-811-682/

13. Kent, Allen et al. eds. Electronic Handling of Information: Testing and Evaluation. Washington, D.C. Thompson, 1967.

14. Kisylia, Andrew Philip. An Association Processor for Information Retrieval. Springfield, Va.: Clearinghouse for Federal Scientific and Technical Information, August 1968. /AD-675-310/

15. Lefkovitz, David "File Structures for On-Line Systems" From lecture notes for ACM Professional Development Seminar, June 1968.

16. Martin, James. Design of Real-Time Computer Systems. Englewood Cliffs, N.J.: Prentice-Hall, 1967.

17. Meadow, Charles T. The Analysis of Information Systems. A.Programmer's Introduction to Information Retrieval New York: Wiley, 1967.

18. MITRE Corporation. Data Management Systems Survey, By J. P. Pry et al. Springfield, Va.: Clearinghouse for Federal Scientific and Technical Information, January 1969. /AD-684-707/

19. _____. Final Report - Joint AFLC/ESD/MITRE Advanced Data Management /ADAM/ Experiment. By B. F. CHAR and A.C. Foreman. Springfield, Ba.: Clearinghouse for Federal Scientific and Technical Information, February 1967. /AD-648-226/

20. Salton, Gerard. Automatic Information Organization and Retrieval. New York: McGraw-Hill, 1968.

21. TRW Systems, Inc. TRW Systems Group. GIM Personnel Accounting - A GIM Sample Application. /3181-B/ Also GIM System Summary. /3181-A/ Redondo Beach, Calif., n.d.

22. U.S. Departement of the Air Force. Headquarters Electric Systems Division /AFSC/. An Operational Capability Description for the World-Wide Military Command and Control System /WWMCCS/ Data Management System. Dedford, Mass., June 11, 1969.

Cena zł 92.-