

ISTOTA I ZASADY PROJEKTOWANIA STRUKTURALNEGO

"Trzeba dokonać wyraźnego rozróżnienia pomiędzy zasadami a metodami stosowania tych zasad.

Jeśli zajmować się będziemy tylko zasadami, na pewno stworzymy bardzo piękne teorie, ale będą to teorie niezbyt skuteczne; będą też mało użyteczne i trudno zastosowalne.

Natomiast gdy ktoś pracuje nad metodami nie wiedząc dokładnie na jakich zasadach metody te się opierają, otrzyma w końcu metody, które nie przetrwają próby i nie dadzą prawidłowych wyników, a zatem metody, które nie zdadzą w praktyce egzaminu"

/J.D. Warnier/

Zarys istoty projektowania strukturalnego

W projektowaniu strukturalnym główny akcent skierowany jest na projektowanie elementów strukturalnych systemu i wiązanie ich w jednostki użytkowe odpowiadające potrzebom problemu zwane pakietami problemowymi.

W roli elementów strukturalnych występują takie komponenty wykonawcze jak procedury, zestawy danych i zestawy zdań sterujących systemem operacyjnym.

Zadanie polega na tym, aby poprzez kombinacje stosunkowo prostych komponentów realizować złożone zadania przetwarzania, a ściślej rzecz biorąc, aby za pomocą ustrukturalnionych elementów rozwiązywać niestabilne strukturalnie lub słabo ustrukturalnione problemy.

Strukturyzacja elementów polega na wyznaczeniu im względnie zamkniętego funkcjonalnie zakresu działania, wyposażenia ich w łącza montażowe i modyfikowalne mechanizmy wykonawcze.

Elementy oraz ich opis gromadzone są w specjalizowanej bazie danych zwanej prasystemem. Z elementów typowych uzupełnionych o algorytmy obliczeniowe specyficzne dla danego problemu tworzone są za pomocą narzędziowego oprogramowania eksploatacyjne pakiety problemowe.

Celem projektowania strukturalnego jest więc system adaptacyjny, który można rozwijać i modyfikować stosownie do potrzeb. Do osiągnięcia tego celu potrzebny jest określony aparat formalny, metodyczny i konstrukcyjny.

Narzędzia i techniki projektowania strukturalnego należy rozpatrywać w dwóch sferach /płaszczyznach/ działania:

- w sferze poznawczej, obejmującej analizę i dekompozycję problemu /lub sytuacji decyzyjnej albo po prostu potrzeb informacyjnych określonego stanowiska kierowniczego/,
- w sferze realizacyjnej, polegającej na projektowaniu komponentów wykonawczych i technologii montażu pakietów problemowych.

Sfera poznawcza charakteryzuje się stosowaniem wielowarstwowych technik dekompozycyjnych /top-down/ opartych na hierarchicznym podziale złożonego układu /bloku/ na elementy prostsze. Działanie w sferze poznawczej prowadzi do poznania istoty problemu oraz sformułowania potrzeb informacyjnych.

W sferze realizacyjnej stosowane jest podejście wielostrukturalne, obejmujące projektowanie struktur danych, struktur procedur, struktur użytkowych /reprezentowanych przez pakiety problemowe/ oraz narzędziowych struktur technologicznych. W sferze tej nie operuje się więc takimi konceptualnymi elementami strukturalnymi jak podsystemy, jednostki funkcjonalne i moduły projektowe, których użycie może być uzasadnione w sferze poznawczej w przypadku wystąpienia bardzo złożonego problemu. Natomiast w sferze realizacyjnej, w której pakiety problemowe wielokrotnie na siebie zachodzą /wskutek wykorzystywania tych samych komponentów/ wyróżnianie tak obszernych jednostek strukturalnych nie jest celowe.

W celu uproszczenia struktur technologicznych i zmniejszenia nakładów na konserwację pakietów problemowych zaleca się stosowanie w strukturach użytkowych spłaszczonych układów hierar-

chicznych, np. typu dwupoziomowego /moduł sterujący oraz grupa elementów podporządkowanych/.

Aby umożliwić modyfikacje tekstu procedury i swobodne wiązania sieciowe pomiędzy elementami, powinny być one wprowadzone do prasytemu w postaci uogólnionej /szkieletowej/, stanowiącej twórczo, z którego za pomocą narzędzi technologicznych generowane będą struktury użytkowe.

Dzięki dwu sferom realizacji w metodzie strukturalnego projektowania wykorzystuje się podejście z dołu do góry /na etapie montażu/, z góry do dołu /na etapie analizy rozpoznawczej problemu/ oraz swobodnych skojarzeń strukturalnych /na etapie budowy elementów prasytemu/.

Ramowe działania podejmowane w projektowaniu strukturalnym przedstawia rysunek 1. W sferze rozpoznawczej problem użytkowy może być identyfikowany wstępnie przez analizę zasobów /materiałnych, finansowych, kadrowych/ i procesów /zaopatrzenia, wytwarzania, zbytu/ oraz funkcji zarządczych /np. planowanie, analiza,.../. Powiązania pomiędzy nimi można przedstawić w postaci tablicy krzyżowej /tab.1/. Technikę sporządzania tych tablic omówiono w pracy /39/. Tablice te są uszczegóławiane podczas dekompozycji hierarchicznej problemu, którą omówimy w dalszej części pracy. Następnie problem jest rozpracowywany na poziomie struktur danych i algorytmów. Wówczas należy wziąć pod uwagę możliwości wykorzystania typowych elementów nagromadzonych w katalogach prasytemu. W przypadku stosowania języka specyfikacyjnego problem jest opisywany w sposób formalny, który tworzy wejście do sfery realizacyjnej poprzez przedprocesor. W powiązaniu z systemem zarządzania bazą danych działanie przedprocesora prowadzi do wygenerowania pakietu problemowego.

Przedstawiony obraz działań wydać się może skomplikowany w porównaniu z dotychczasową praktyką. Wyróżnienie sfer działania oraz poziomów strukturalnych ma na celu:

- dostosowanie zakresu działań i aparatu pojęciowego do możliwości i przygotowania grup zawodowych uczestniczących w procesie projektowania /użytkowników, analityków problemowych, projektantów, programistów zastosowań, technologów narzędzio-

	SPERA POZNAWOZA	SPERA REALIZACYJNA
Podstawowe marszalsia	<ul style="list-style-type: none"> • Technika top-down • Język specyfikacyjny do opisu problemu /lub podzbior ogólnego języka specyfikacyjnego/ 	<ul style="list-style-type: none"> • Język /podzbior/ specyfikacyjny do opisu konstrukcji systemu • Translator języka specyfikacyjnego • Przetwarzacz sterowania składowych tekstów procedur • Makrogeneratory • System zarządzania bazą danych do katalogowania i wyszukiwania opisu elementów prasytenu, w tym opisu fizycznych i logicznych struktur danych • Nieproceduralny język wyszukiwania informacji • Generatory danych
Zadania	<p>Projektowanie struktury problemu /funkcjonalnej/</p> <ul style="list-style-type: none"> • Dekompozycja problemu na elementy strukturalne • Charakterystyka problemu /głównie wg obiektów, procesów i funkcji/ • Projektowanie elementów strukturalnych: <ul style="list-style-type: none"> - logicznych struktur danych - algorytmów problemowych 	<p>Projektowanie struktury technologicznej /konstrukcyjnej/</p> <ul style="list-style-type: none"> • Konwersja struktury problemu na strukturę technologiczną • Wybór elementów prasytenu nadających się do realizacji problemu • Projekt fizycznych struktur danych oraz weryfikacja logicznych struktur danych /m.in. opracowanie pod-schematów baz danych/ • Opracowanie elementów uzupełniających specyficznych dla danego problemu • Ustalenie zasad montażu elementów oraz zasad modułu sterującego <p>Generowanie pakietu problemowego</p> <ul style="list-style-type: none"> • Modyfikowanie tekstów elementów prasytenu oraz wy-poseżenie ich w języku montażu • Montaż pakietu problemowego z modułów problemowych oraz zmodyfikowanych elementów prasytenu • Testowanie pakietu za pomocą generatora danych • Testowanie pakietu na danych rzeczywistych • Weryfikacja pakietu • Generowanie dokumentacji użytkowej pakietu problemowego

Rys. 1 b. Treść zadań w metodzie projektowania strukturalnego

wych, itp./, przy czym integrującą funkcję transferu rezultatów pracy poszczególnych grup zawodowych pełni między innymi język specyfikacyjny oraz oprogramowanie narzędziowe.

rozdzielenie w procesie budowy produktu informatycznego^{1/} następujących kategorii /typów/ elementów:

- konceptualnych

stanowiących jedynie ogniwa pośrednie w poznawaniu problemu,

- surowcowych wielokrotnego użytku

stanowiących tworzywo typowe, z którego budowany jest produkt /mamy tutaj na uwadze "standardowe" procedury i struktury danych utrzymywane przez system zarządzania bazą danych/,

- surowcowych indywidualnych

reprezentujących treści dodatkowe zależne wyłącznie od określonego problemu,

- usługowych /narzędziowych/

nie wchodzących do produktu, lecz umożliwiających jego otrzymanie z elementów surowcowych.

W projektowaniu niestrukturalnym powyższe kategorie elementów zacierają się wzajemnie, co utrudnia uchwycenie ich specyfiki i wyważenie wpływu na rozwiązania projektowe. Podejście strukturalne zmusza niejako twórców produktu do korzystania z nagromadzonego dorobku /w postaci elementów surowcowych wchodzących do prasyntemu/ i sprzyja tworzeniu adaptacyjnych systemów powtarzalnych, a ściślej rzecz biorąc, rodziny produktów opartych o elementy powtarzalne.

Zasady projektowania strukturalnego

Jako punkt wyjściowy do reguł projektowania strukturalnego przyjąć można szereg zasad ogólnych i szczególnych. Wśród zasad ogólnych wyróżniamy:

- i - elastyczność

czyli sprawność w zmiennych warunkach działania,

^{1/} W związku z rozważaniami rozdziału II w niniejszej pracy często zamiast terminu "system informatyczny" używany będzie termin "produkt informatyczny" lub pakiet problemowy.

- . ii - rozwijalność
czyli możliwość dołączania nowych elementów,
- . iii - przejrzystość strukturalną,
- . iv - optymalność stosunków wielkości związanych
/m.i.jako kryterium doboru wielkości elementów i głębokości struktur/,
- . v - ekonomiczność
/wyrażająca się w wykorzystaniu elementów powtarzalnych/.

Na tle zasad ogólnych sformułować można następujące zasady szczególne:

- . Zasada 1. Elastyczność i rozwijalność /składające się razem na adaptacyjność/ wywodzą się z technologii systemu oraz z właściwości konstrukcyjnych jego elementów strukturalnych.
- . Zasada 2. Elementami strukturalnymi są bloki realizacyjne a nie konceptualne. Do bloków realizacyjnych zalicza się opisy struktur danych oraz procedury, dane i zdania sterujące.
- . Zasada 3. Uogólnione elementy strukturalne charakteryzuje swoista konstrukcja w postaci łączy wejścia i wyjścia oraz szkieletowych /"zaślepionych"/ zwrotów wewnętrznych powiązanych z językiem specyfikacyjnym. Zestaw uogólnionych elementów strukturalnych nazywa się prasystemem lub "biblioteką możliwości".
- . Zasada 4. Element prasystemu powinien spełniać następujące wymagania:
 - posiadać sformalizowane łączy wejściowe i wyjściowe /to drugie opcjonalnie/,
 - stanowić całość logiczną /modyfikowalną i montowalną niezależnie od innych elementów/,
 - mieć przejrzystą strukturę sterowania wewnętrznego nastrajaną parametrycznie podczas przetwarzania lub generacyjnie podczas generowania pakietu problemowego,

- posiadać dostęp do danych wspólnych lub przekazywanych przez inne procedury,
 - zawierać treści "szkieletowe", które będą podczas nastrajania zamieniane na makrorozwinięcia lub modyfikowane przez przedprocesor języka specyfikacyjnego.
- . Zasada 5. W projektowaniu zasobów proceduralnych obowiązuje zasada ograniczonego zaufania do skoków GO TO, zaś w projektowaniu struktury montażowej /użytkowej/ zasada ograniczonego zaufania do struktur hierarchicznych /na rzecz struktur spłaszczonych/.
- . Zasada 6. Prasystem budowany jest jako utajona /potencjalna/ struktura sieciowa, co oznacza, że fizyczne /ręczyste/ powiązania pomiędzy elementami prasystemu nie muszą występować, zaś systemowo zabezpieczona jest możliwość ich realizacji.
- . Zasada 7. Produkt informatyczny tworzony techniką interpretacyjną /jako bezpośrednie użycie elementów prasystemu/ przedstawia sobą przejściowe sieciowe układy strukturalne. Produkt uzyskany techniką generacyjną zaleca się tworzyć w postaci struktur spłaszczonych /bez wielokrotnego zagnieżdżenia/, aby zapewnić przejrzystość strukturalną i ułatwić modyfikacje oraz utrzymywanie powiązań pomiędzy elementami.
- . Zasada 8. Zaleca się przeprowadzanie korekty elementu najpierw na poziomie prasystemu, gdyż wówczas łatwiej podlega ona wielokierunkowej kontroli na okoliczność spełnienia wymagań zrealizowanych dotychczas implementacji elementu. Ponadto w tym przypadku będzie istniała możliwość rekonstrukcji pakietu problemowego z elementów prasystemu /ewentualnie odrotnie/.
- . Zasada 9. Określony problem lub sytuacja decyzyjna jest głównym impulsem rozpoczęcia prac nad pakietem

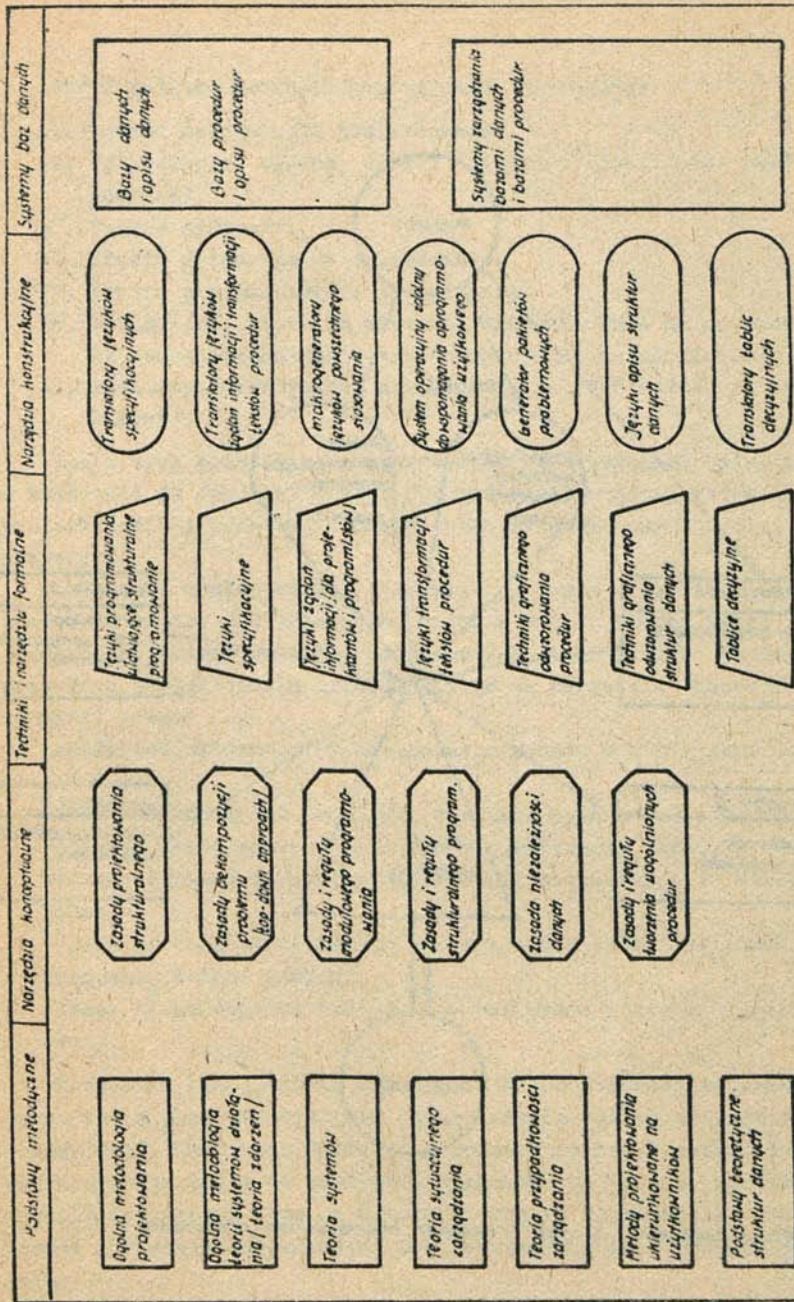
problemowym, tj. rozpoznania problemu lub uzupełnienia prasystemu, sformułowania parametrów, zwrotów specyfikacyjnych, itp. System informatyczny /rozumiany tutaj jako zmienny zestaw pakietów problemowych/ budowany jest więc nie tyle na podstawie harmonogramów strategicznych, ile według potrzeb "chwili". Strategia powinna dotyczyć głównie konstrukcji prasystemu względnie tematów stabilnych oraz przygotowania organizacyjnego.

- . Zasada 10. Aby "przypadkowość" w wyborze przedmiotu implementacji nie powodowała chaosu projektowego i dezorganizacji baz danych, może być ona stosowana jedynie w warunkach uprzednio określonej konstrukcji baz danych oraz przy pewnym stopniu napełnienia prasystemu uogólnionymi elementami realizacyjnymi /w tym elementami systemu zarządzania bazami danych/.
- . Zasada 11. Ważnym przedsięwzięciem przeciwdziałającym chaosowi informacyjnemu jest jednolity system kodowania zdarzeń /transakcji/ gospodarczych i informacyjnych w skali całego przedsiębiorstwa lub organizacji.
- . Zasada 12. Warunkiem strukturalnego projektowania jest posiadanie narzędzi konstrukcyjnych do wspomagania projektantów i programistów.

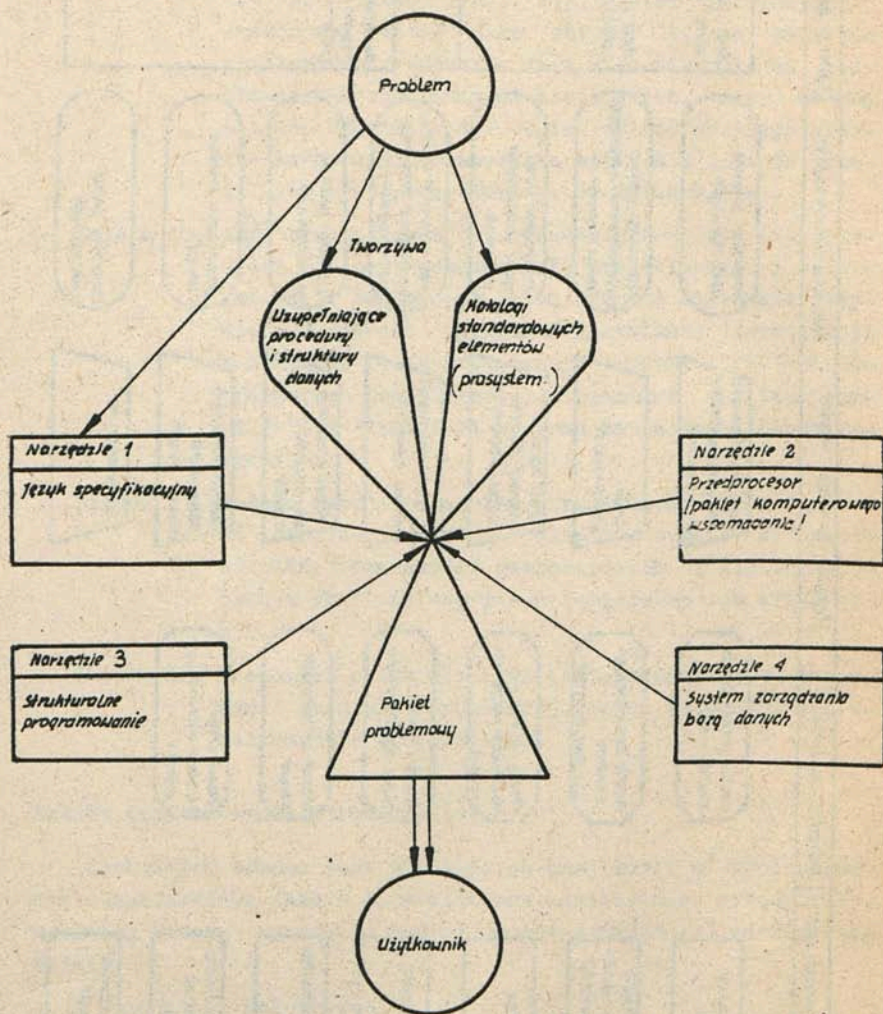
Działy projektowania strukturalnego

Aczkolwiek trudno jest w chwili obecnej mówić o projektowaniu strukturalnym jako o ukształtowanej dziedzinie metodycznej, niemniej jednak wyróżnić w niej można co najmniej następujące działy:

- . A - podstawy metodyczne,
- . B - narzędzia konceptualne,
- . C - techniki i narzędzia formalne,
- . D - narzędzia konstrukcyjne /komputerowo wspomagane/.



rys. 2. Ogólny schemat elementów metodycznych i wykonawczych metody strukturalnego projektowania



RYS. 3 Podstawowe narzędzia i tworzywa strukturalnego projektowania

Wśród podstaw metodycznych na uwagę zasługują:

- . A1. Ogólna metodologia projektowania,
- . A2. Metodologia ogólna teorii systemów działania /teoria zdarzeń/,
- . A3. Teoria systemów,
- . A4. Teoria sytuacyjnego zarządzania,
- . A5. Teoria przypadkowości zarządzania,
- . A6. Metody projektowania systemów ukierunkowane na użytkownika /socjo-techniczna, współangażująca, infologiczna/,
- . A7. Podstawy teoretyczne struktur /w szczególności struktur danych/.

Wpływ tych podstaw na projektowanie strukturalne omówiliśmy w rozdziale II /poza A1 i A7/. Uwzględnienie ogólnej metodologii projektowania i podstaw teoretycznych struktur wydaje się być oczywiste.

Narzędzia konceptualne stanowią pierwszą /ogólną/ część właściwej metody projektowania, dając w wyniku szereg zasad metodycznych i sformułowania podstawowych mechanizmów metody; w przypadku projektowania strukturalnego za narzędzia konceptualne uznać można:

- . zasady projektowania strukturalnego podane w niniejszym rozdziale,
- . zasady dekompozycji problemu w sferze poznawczej /podejście top-down występujące w metodach projektowania BISAD, HIPO/,
- . większość zasad i reguł programowania modularnego i strukturalnego,
- . zasada niezależności danych występujące w SZBD /systemach zarządzania bazami danych/,
- . zasada niezależności procedur /w przypadku procedur uogólnionych/.

Narzędzia konceptualne nie stanowią dostatecznej podstawy do prowadzenia prac projektowych /dają jedynie ogólne przygotowanie metodyczne/. Narzędzia formalne i konstrukcyjne /w sumie/ doprowadzają projektanta do uzyskania produktu. Za pomocą narzędzi i technik formalnych może on utworzyć projekt rozwiązania, zaś za pomocą narzędzi konstrukcyjnych przekształcić go w produkt działający.

Wśród narzędzi i technik formalnych wyróżnić można:

- języki specyfikacyjne /do specyfikowania elementów strukturalnych: ich treści i powiązań/,
- języki żądań informacji oraz transformacji treści elementów,
- techniki graficznego odwzorowania treści elementów /np. schematy blokowe/,
- tablice decyzyjne,
- języki programowania ułatwiające strukturalne programowanie.

Jako narzędzie konstrukcyjne występują:

- translatory języków specyfikacyjnych, wyszukiwawczych i manipulacyjnych,
- translatory języków programowania,
- systemy operacyjne ułatwiające technologię strukturalnie zaprojektowanych produktów /np. OS dla IBM 360 i JS RIAD/,
- translatory tablic decyzyjnych,
- makrogeneratory /w ramach języków programowania/,
- generatory pakietów problemowych,
- systemy zarządzania bazami danych.

Graficzne zestawienie elementów składowych projektowania strukturalnego przedstawiono na rysunku 2. Jak widzimy, projektowanie strukturalne selektywnie wykorzystuje dorobek wielu dziedzin informatyki. Szczegółowe omówienie wymienionych technik i narzędzi wymaga wielotomowego opracowania i jest zadaniem dla wieloosobowego zespołu autorskiego. W pracy niniejszej w dalszych rozdziałach omówimy wybrane aspekty niektórych technik i narzędzi, a w szczególności zajmiemy się techniką opisu struktur danych, techniką graficznego odwzorowania treści strukturalnego elementu proceduralnego oraz wymogami stawianymi systemom zarządzania bazą danych.

Kończąc rozdział, przypomnimy, że projektowanie strukturalne ma:

- ułatwić wykorzystanie dorobku projektowo-programowego,
- zwiększyć adaptacyjność systemu w drodze swobodnego montażu uogólnionych elementów projektowych,
- podnieść jakość rozwiązania projektowego, między innymi po-

przez jednoznaczny sformalizowany opis algorytmów i danych oraz przejrzysty układ strukturalny,

- znacznie przyspieszyć prace implementacyjne produktów informatycznych,
- zmniejszyć koszt opracowania produktów informatycznych /w stosunku do systemów indywidualnych/ dla użytkowników realizujących kompleksową informatyzację w skali przedsiębiorstwa, zjednoczenia lub zrzeszenia małych jednostek organizacyjnych.

Docelowo, metoda strukturalnego projektowania może przekształcić się w metodę strukturalnego użytkowania systemu, kiedy za pomocą łatwego nie proceduralnego języka specyfikacyjnego i manipulacyjnego sam użytkownik dokonywać będzie operacji montażowych i weryfikować jakość otrzymanego pakietu problemowego. W sytuacji tej, projektant i programista będą potrzebni głównie do tworzenia elementów prasytemu, budowy indywidualnych uzupełniających elementów składowych oraz szkoleniowego wspomagania użytkownika. Na razie jest to tylko perspektywa.