A C T A U N I V E R S I T A T I S L O D Z I E N S I S
FOLIA OECONOMICA 119, 1992

Gabriela Idzikowska*, Zofia Owczarek**

METODY KONTROLI FUNKCJONOWANIA SYSTEMÓW INFORMATYCZNYCH

1. Elementy systemu informatycznego

W ostatnich latach nastąpił burzliwy i wielostronny rozwój zastosowań informatyki w zarządzaniu. Ta wielostronność dotyczy takich elementów systemu informatycznego, jak sprzęt, oprogramowanie, technologia. W zarządzaniu coraz szerzej stosuje się informatykę, a w zbiorach gromadzi się dane związane z rzeczowymi, osobowymi i finansowymi zasobami przedsiębiorstwa. Warunkiem koniecznym właściwego zarządzania przedsiębiorstwem jest prawidłowość danych zawartych w zbiorach, a można ją zagwarantować poprzez poprawne funkcjonowanie systemu informatycznego, rozumianego jako zestaw następujących elementów [6]:

- sprzęt,
- dokumentacja systemu informatycznego,
- pracownicy,
- oprogramowanie systemowe,
- oprogramowanie użytkowe,
- zbiory danych.

Aby upewnić się, czy system działa poprawnie, należy skontrolować wymienione elementy i ich współdziałanie.

Współczesna technologia komputerowa osiągnęła wysoki poziom niezawodności sprzętu poprzez wbudowywanie wewnętrznych kontroli, które mają na celu wykrywanie nieprawidłowości jego działania. Najbardziej powszechnymi mechanizmami są [6]:

* Dr, adiunkt w Katedrze Informatyki.
** Dr, adiunkt w Katedrze Informatyki.

- podwójne czytanie (dual read),
- kontrola parzystości (parity check),
- opóźniona kontrola (echo check),
- czytanie po zapisaniu (read after write).

Programy diagnostyczne, zakupywane wraz ze sprzętem, dają pewność, że kontrole sprzętowe są realizowane prawidłowo.

Każdy wdrażany system informatyczny musi być udokumentowany. Ta dokumentacja spełnia szereg funkcji, m. in. stanowi pomoc przy kontroli pozostałych elementów systemu informatycznego. Kontrola prawidłowości tej dokumentacji stanowi przedmiot rozważań w odrębnym artykule niniejszego zeszytu.

Pracowników związanych z systemem informatycznym można w zasadzie podzielić na dwie grupy: pracowników ośrodka obliczeniowego i użytkowników systemu. Ten podział ma miejsce głównie w przypadku wykorzystywania komputerów stacjonarnych, natomiast zaciera się granica między wymienionymi grupami przy stosowaniu mikrokomputerów. Niezależnie od rozwiązań sprzętowych istotne jest rozdzielenie przynajmniej następujących funkcji: programowania, obsługi komputera i bibliotekarza zbiorów. Kontrola tego elementu systemu informatycznego odbywa się bez udziału komputera, poprzez wykorzystanie ankiet, wywiadów itd.[1]

Pozostałe elementy systemu informatycznego mogą być badane przy pomocy komputera. Stosowanym w tej dziedzinie metodom jest poświęcona dalsza część niniejszego artykułu.

2. Metody kontroli oprogramowania systemowego

Oprogramowanie systemowe, podobnie jak sprzęt, charakteryzuje się dość wysokim poziomem niezawodności, jest ono jednak bardziej podatne na błędy. Wiążą się z nim dwa rodzaje błędów [5]:

[1] Problem dotyczący zawartości ankiet i struktury wywiadów jest zbyt rozległy, aby można było go wyczerpać w ramach niniejszego artykułu. Bliższe informacje na ten temat są zamieszczone w pracach: V. R. V. C o o p e r, Manual of Auditing, Gee and Co. Publ., London; R. P. F i s h e r, Information Systems Security, Prentice-Hall Inc., Englewood Cliffs N. J. 1984. W ramach problemu badawczego CPBR 8.10. zostały opracowane wzory takich ankiet i zamieszczone w opublikowanym podręczniku pt. Prowadzenie i rewizja ksiąg rachunkowych przy technice komputerowej, SKWP, Łódź 1990.

- nie wykryte podczas projektowania i testowania oprogramowania (popełnione niecelowo),
- wynikające z niepowołanego oddziaływania na oprogramowanie.

Te oba typy błędów mają istotny wpływ na prawidłowość funkcjonowania systemu informatycznego.

W literaturze przedmiotu jest opisanych kilka metod kontroli poprawności działania oprogramowania:

- metoda włączania pakietu systemowego zabezpieczenia zasobów,
- metoda sprawdzania sum kontrolnych,
- metoda sprawdzania czasów utworzenia,
- metoda odwzorowania,
- metoda badania dziennika konsoli.

M e t o d a w ł ą c z a n i a pakietu systemowego zabezpieczenia zasobów [5] (resource security system) została eksperymentalnie wdrożona do uzyskania "bezpiecznej wersji" systemu operacyjnego OS/360. Do oryginalnego systemu dodano następujące oprogramowanie:

- obsługujące układy zabezpieczające pobieranie danych,
- zmodyfikowany program drukujący (writer), który liczy i drukuje numery stron oraz informacje dotyczące bezpieczeństwa na początku i końcu każdej strony w zbiorze wyjściowym,
- wymuszające zakończenie kroku zadania, który spowodował zagrożenie bezpieczeństwa, bądź próbował wykonać uprzywilejowaną operację,
- włączające brzęczyk konsoli w przypadku zagrożenia,
- automatycznie uruchamiające system zabezpieczenia zasobów,
- zerujące każdy zwracany do systemu blok pamięci operacyjnej,
- umożliwiające całkowite zamazanie wrażliwej pozostałości[2] w zbiorach o dostępie bezpośrednim,
- umożliwiające wyzerowanie buforów fizycznych, kiedy urządzenie jest wyłączane.

Programy systemowe powinny być modyfikowane tylko przez upoważnione osoby. Jedną z metod zapewniających bezużyteczność kopii programu po niepowołanej modyfikacji jest użycie sum kontrolnych [5]. Suma kontrolna może być automatycznie obliczana za każdym razem, gdy

---

[2] Wrażliwa pozostałość może być rozumiana np. jako istniejący fizycznie zbiór, który został skasowany w katalogu.

program jest legalnie modyfikowany. Wszystkie słowa chronionego obiektu mogą być poddane działaniu określonego algorytmu, a wynik zapamiętany w zaszyfrowanej postaci. Prawdopodobieństwo, że w wyniku nieupoważnionej modyfikacji uzyska się sumę kontrolną równą poprzedniej, jest bliskie zeru.

M e t o d a s p r a w d z a n i a c z a s ó w u t w o r z e n i a [5] polega na zapisywaniu daty i czasu dokonywania modyfikacji programu. System operacyjny może wówczas sprawdzać czas tworzenia programu przez jego porównanie z czasem zapisanym w oddzielnym zbiorze. Jeżeli w wyniku porównania zostaną stwierdzone różnice, będzie to oznaczało, że dokonano nieupoważnionej modyfikacji lub też, że użyto złej wersji programu. System operacyjny może tę metodę wykorzystywać także do kontroli legalności modyfikowania programów zastosowań i zbiorów danych. Wykryte różnice są sygnalizowane, a zbiory wyłączane z użycia.

M e t o d a o d w z o r o w a n i a [1] (mapping) polega na śledzeniu przebiegu programu poprzez wykorzystanie specjalnego oprogramowania (pomiarów, obliczeń). Podczas realizacji to oprogramowanie liczy, ile razy jest wykorzystywany każdy rozkaz programu oraz dostarcza statystyk dotyczących użytkowania zasobów.

D z i e n n i k k o n s o l i, zwany także dziennikiem kontrolnym lub śladem kontrolnym [1, 3, 5] jest jedną z najważniejszych i najbardziej skutecznych metod kontroli. Bez dokładnego dziennika nie jest możliwe zbadanie tego, co zdarzyło się w przeszłości. Może on być wyświetlany na ekranie monitora bądź drukowany, a jego zawartość może służyć do badań statystycznych. Zapisy występujące w dzienniku są utrwalane na nośniku magnetycznym i dotyczą:

- żądań użytkownika (identyfikator użytkownika, treść transakcji),
- czasu wystąpienia żądania,
- użytego terminalu,
- nazwy programu,
- statusu żądania.

Informacje te mogą być drukowane okresowo, w różnych układach, muszą jednak umożliwiać ustalenie:

- upoważnień związanych z każdym chronionym zasobem,
- odpowiedzialności za dokonane zmiany upoważnień,

- wszelkich przypadków dostępu do chronionego zasobu,
- wszystkich odmów udzielenia dostępu,
- wszystkich zmian dokonanych w pamięci przez operatora.

Tworzenie dziennika może działać odstraszająco na potencjalnych intruzów. Ponadto, dziennik jest wykorzystywany do ochrony i odzyskiwania zbiorów danych. Dane statystyczne z niego uzyskiwane pokazują kto korzystał z systemu oraz kiedy i jakie zasoby były użyte.

Kontrola oprogramowania systemowego [9] wymaga lepszego przygotowania informatycznego niż kontrola programów zastosowań. Zanim zostanie ona zrealizowana, należy przedsięwziąć następujące kroki:

1) określić ważność (istotność) oprogramowania systemowego,
2) określić żądaną częstotliwość jego kontroli,
3) uzyskać odpowiednią ekspertyzę,
4) ustalić program kontroli.

Ważność oprogramowania systemowego określa się z punktu widzenia integralności wszystkich danych. Niektóre części oprogramowania są zawsze ważne, np. systemy operacyjne czy systemy zarządzania bazą danych, inne natomiast są mniej istotne, np. programy generujące raporty.

W celu określenia częstotliwości przeprowadzania kontroli oprogramowania należy uwzględnić następujące czynniki:

- jego ważność,
- częstość dokonywania w nim modyfikacji (po każdej wprowadzanej zmianie powinny być realizowane procedury kontrolne),
- jego klasę, tzn. przynależność do oprogramowania standardowego (zwykle lepiej przetestowanego), bądź własnego (gorzej przetestowanego).

Efektywne kontrole oprogramowania systemowego wymagają profesjonalnej ekspertyzy. Należy podjąć decyzję, czy opracuje się ją we własnym zakresie, czy też zatrudni konsultantów, ale nawet w tym przypadku własny personel musi mieć dostateczną wiedzę informatyczną, by móc komunikować się z konsultantami.

Ustalony program kontroli oprogramowania systemowego stanowić powinien integralną część całego programu.

Podstawą przeprowadzenia badań są:

- dokumentacja dostarczana przez producenta oprogramowania,
- wywiady przeprowadzane z pracownikami ośrodka i użytkownikami,

- bieżąca dokumentacja eksploatacyjna dostarczana przez komputer (np. dziennik konsoli).

Ta manualna metoda badania umożliwia ustalenie słabych punktów oprogramowania systemowego, które powinny być poddane szczegółowej analizie. W tym celu można wykorzystać jedną z metod badania stosowaną do kontroli programów zastosowań.

3. Metody kontroli oprogramowania użytkowego

W historii eksploatowania systemów informatycznych wypracowano wiele metod kontroli wspomaganych komputerem.

Metody te mogą polegać na testowaniu systemów i programów (metody testujące) bądź na badaniu przebiegu działania systemu.

3.1. Metody testujące

Wszystkie metody należące do tej grupy wykorzystują dane testowe. Aby dobrze skontrolować system, istotne jest odpowiednie przygotowanie tych danych, ponieważ na ich podstawie wyciąga się uogólniające wnioski, dotyczące niezawodności całego programu lub systemu. Projektując dane testowe powinno się wykorzystywać podejście systematyczne, aby nie pominąć żadnej krytycznej ścieżki w programie lub w systemie. Osiąga się to poprzez stosowanie tablic decyzyjnych lub schematów blokowych. Po zaprojektowaniu danych testowych następuje tworzenie zgodnego z projektem zestawu danych. Można tu wykorzystać:

- dane rzeczywiste,
- dane testowe opracowane na etapie wdrażania systemu,
- dane testowe przygotowane specjalnie do celów kontroli.

Dane testowe można przygotować manualnie lub z wykorzystaniem automatycznych generatorów, względnie innych specjalistycznych pakietów, o czym będzie jeszcze mowa w tym artykule.

Metody, które wykorzystują dane testowe, były najwcześniej i najpowszechniej stosowane. Zalicza się do nich:

- metodę transakcji hipotetycznych,
- metodę zintęgrowanego testowania,
- metodę symulacji równoległej lub modelowania.

3.1.1. Metoda transakcji hipotetycznych [9]

Przetwarzanie danych testowych polega na wprowadzaniu transakcji hipotetycznych do systemu komputerowego, aby sprawdzić kompletność i prawidłowość przetwarzania systemu oraz procedur kontrolnych [2, 3].

Po przygotowaniu transakcji hipotetycznych kontroler powinien przejrzeć wydruk bieżącej wersji zbioru głównego lub zbiorów wykorzystywanych przez programy zastosowań oraz ustalić w sposób manualny, jakie powinny być oczekiwane wyniki tych transakcji. Dane testowe mogą być wprowadzone do systemu dwojako:

- jako dokumenty źródłowe i wówczas muszą przejść przez cały etap walidacji wejściowej,
- jako dane na nośnikach magnetycznych wcześniej skontrolowane.

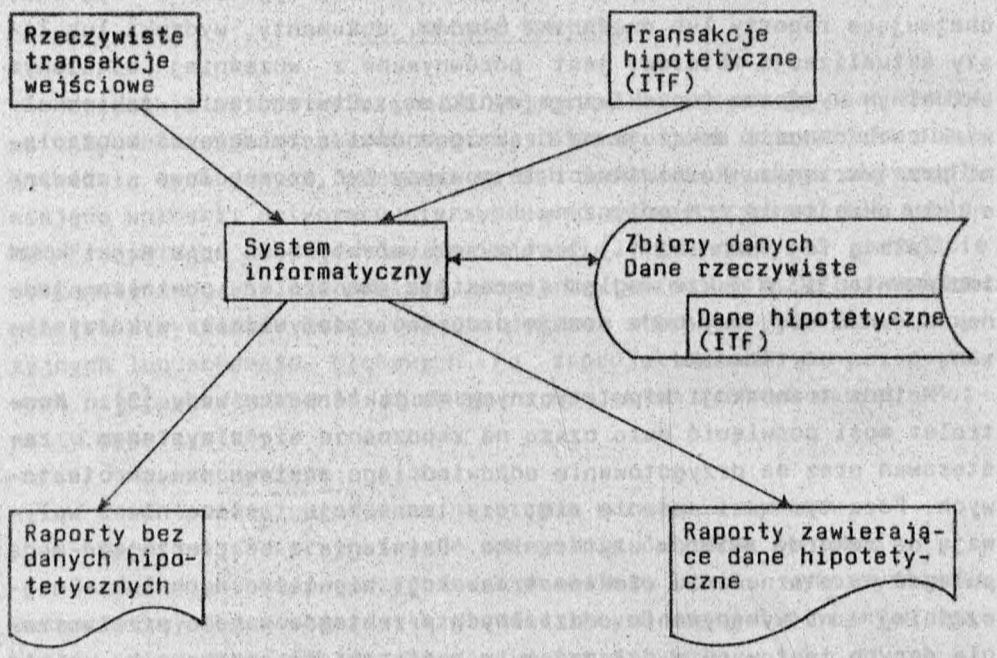
Po przetworzeniu danych testowych aktualne wyjście z systemu, obejmujące raporty lub dzienniki błędów, dokumenty, wydruki lub listy aktualizacji zbiorów, jest porównywane z wcześniej ustalonym aktualnym wyjściem (oczekiwanym wynikiem). Stwierdzenie jakichkolwiek rozbieżności wskazuje na brak zgodności z zalecanymi kontrolami przetwarzania. Rozbieżności te powinny być szczegółowo zbadane w celu określenia ich przyczyny.

Zaletą tej techniki [1] jest wysoka efektywność oraz niski koszt implementacji, a także względna prostota. Kontroler powinien jednak upewnić się, czy bada wersję programu rzeczywiście wykorzystywaną przez użytkownika.

Metoda transakcji hipotetycznych ma także pewne wady [3]. Kontroler musi poświęcić dużo czasu na zapoznanie się z systemem zastosowań oraz na przygotowanie odpowiedniego zestawu danych testowych. Poza tym musi upewnić się, czy transakcje testowe nie wpływają na rekordy zbiorów użytkownika. Działania zabezpieczające mogą polegać na stornowaniu efektów transakcji hipotetycznych lub - najczęściej - na wykonywaniu oddzielnych przebiegów w celu przetworzenia danych testowych w połączeniu z kopią zbioru użytkownika, a nie ze zbiorem rzeczywistym. Realizacja oddzielnego przebiegu pozbawia jednak kontrolera pewnej dozy "realności", którą uzyskuje on podczas przetwarzania danych testowych łącznie z transakcjami rzeczywistymi.

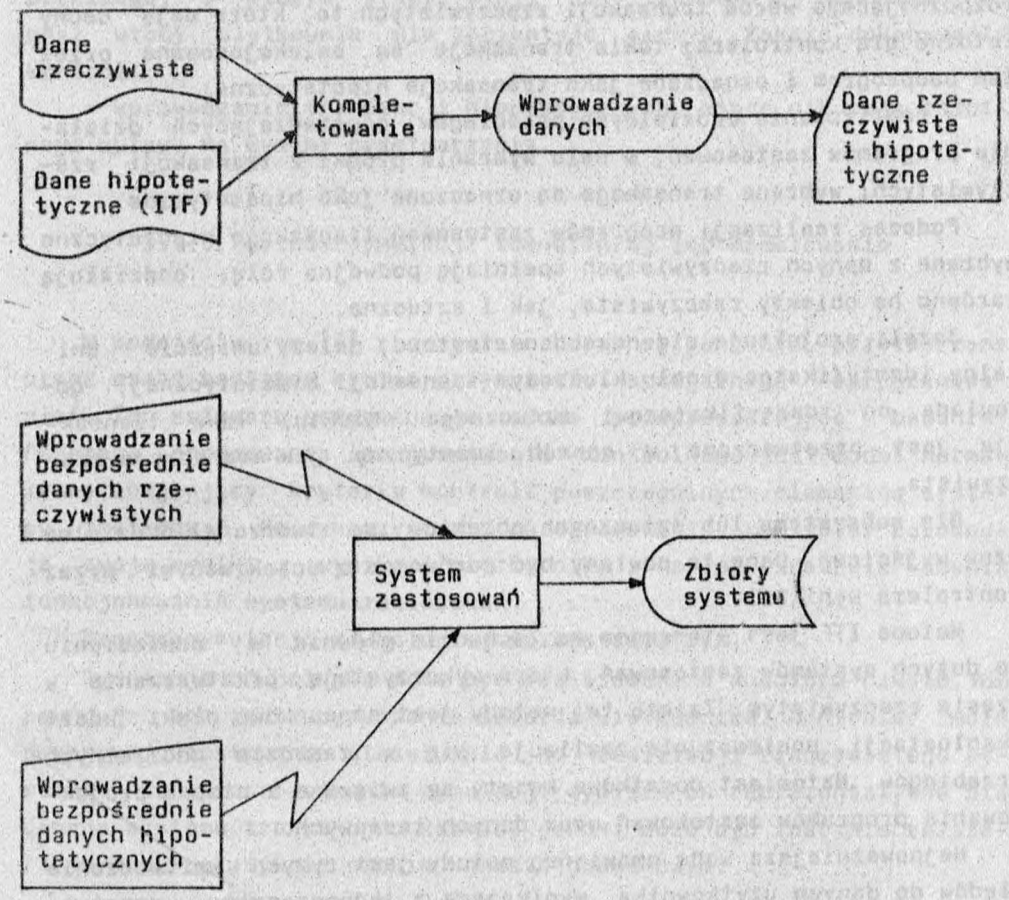
3.1.2. Metoda zintegrowanego testowania [1, 6, 7]

Metoda zintegrowanego testowania (ITF - Integrated Test Facility) jest rozwinięciem idei charakterystycznej dla metody transakcji hipotetycznych (wykorzystuje dane testowe). Polega ona na opracowaniu małego subsystemu w normalnie funkcjonującym systemie informatycznym. Osiąga się to poprzez utworzenie sztucznych zbiorów lub przez dodanie sztucznych rekordów do istniejących zbiorów użytkownika. Integracja polega na równoczesnym przetwarzaniu danych testowych i transakcji rzeczywistych w konfrontacji ze zbiorami głównymi, które zawierają zarówno rzeczywiste, jak i fikcyjne obiekty. Kontrole są więc wykonywane jako część normalnego cyklu przetwarzania, co daje pewność, że kontroluje się rzeczywiście wykorzystywane wersje programów. Metodę zintegrowanego testowania ilustruje rys. 1.



Rys. 1. Metoda zintegrowanego testowania (ITF)
Ź r ó d ł o: [1, s. 636]

Kontroler przekazuje dane testowe do systemu zastosowań taką samą drogą, jaką są przekazywane dane rzeczywiste. W przypadku systemów wsadowych kontroler kompletuje dane testowe, które następnie są włączane w normalny obieg dokumentów rzeczywistych. W przypadku systemów interaktywnych dane są wprowadzane w czasie działania systemu. Metody przekazywania transakcji hipotetycznych przedstawia rys. 2.



Rys. 2. Metody przekazywania transakcji hipotetycznych
Ź r ó d ł o: [8, s. 19]

Transakcje hipotetyczne i odpowiadające im obiekty sztuczne muszą być specjalnie oznaczone. Do oznaczenia transakcji hipotetycznych można wykorzystać jedną z następujących metod:

- zarezerwowanie dodatkowego pola identyfikującego, które oznacza, że dana transakcja jest transakcją hipotetyczną; w takim przypadku na dokumentach źródłowych zostawia się puste miejsce, a oznakowanie odbywa się w czasie palcowania (dziurkowania lub wprowadzania);
- włączenie do programu zastosowań odpowiedniego podprogramu, rozpoznającego wśród transakcji rzeczywistych te, które mają cechy istotne dla kontrolera; takie transakcje są selekcjonowane przez ten podprogram i oznaczane jako transakcje hipotetyczne;
- realizowanie oddzielnych przebiegów, poprzedzających działanie programów zastosowań, w celu wybrania próbki z transakcji rzeczywistych; wybrane transakcje są oznaczone jako hipotetyczne.

Podczas realizacji programów zastosowań transakcje hipotetyczne wybrane z danych rzeczywistych spełniają podwójną rolę; oddziałują zarówno na obiekty rzeczywiste, jak i sztuczne.

Jeżeli projektuje się nowe dane testowe, należy umieścić unikalny identyfikator w polu kluczowym transakcji hipotetycznej; odpowiada on identyfikatorowi sztucznego obiektu. Taka transakcja jest przetwarzana w sposób identyczny jak transakcja rzeczywista.

Dla subsystemu lub sztucznych obiektów są tworzone oddzielne dane wyjściowe. Dane te powinny być porównane z oczekiwanymi przez kontrolera wynikami.

Metoda ITF jest stosowana na Zachodzie głównie w odniesieniu do dużych systemów zastosowań, które wykorzystują przetwarzanie w czasie rzeczywistym. Zaletą tej metody jest stosunkowo niski koszt eksploatacji, ponieważ nie realizuje się w zasadzie oddzielnych przebiegów. Natomiast dodatkowe koszty są związane z etapem projektowania programów zastosowań oraz danych testowych.

Najpoważniejszą wadą omawianej metody jest ryzyko wprowadzenia błędów do danych użytkownika, wynikające z jednoczesnego przetwarzania transakcji rzeczywistych i hipotetycznych. Obecność transakcji hipotetycznych w systemie zastosowań może wpływać na otrzymy-

wane rezultaty (np. na sumy kontrolne). Efekty działania takich transakcji muszą być usunięte. Można tego dokonać przy użyciu jednego z następujących sposobów [9]:

- modyfikacja programów zastosowań po to, by rozpoznawały one transakcje ITF i nie uwzględniały ich, na przykład, w liczeniu sum kontrolnych;
- wprowadzanie transakcji przeciwstawnych do transakcji hipotetycznych; najlepiej, gdy jest możliwe ich wprowadzenie w tym samym przebiegu, w którym wykorzystuje się transakcje hipotetyczne, gdyż wtedy użytkownik nie zorientuje się w fakcie dokonywania kontroli;
- wprowadzanie transakcji hipotetycznych, które nie mają istotnego wpływu na wyniki przetwarzania.

3.1.3. Metoda symulacji równoległej lub modelowania

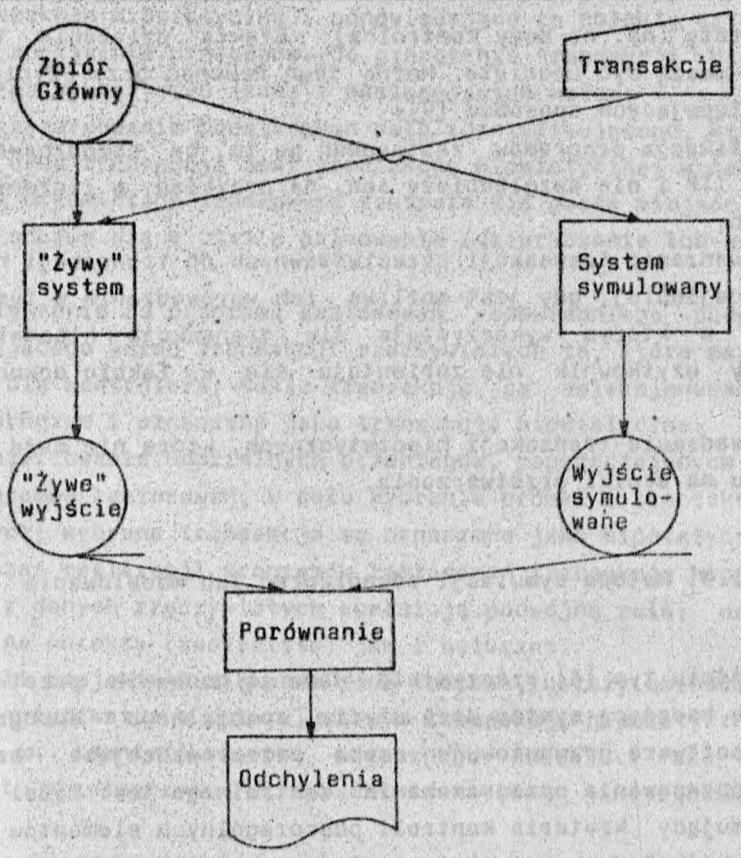
W podejściu tym [6] rzeczywiste dane są ponownie przetwarzane przez osobę badającą system przy użyciu specjalnego oprogramowania[3]. Ten software przygotowuje osoba przeprowadzająca badanie. Podstawą opracowania oprogramowania kontrolnego jest model normatywny obejmujący kryteria kontroli poszczególnych elementów systemu zastosowań. Po wykorzystaniu oprogramowania, kontroler porównuje swoje wyniki z wynikami użytkownika i na tej podstawie ocenia funkcjonowanie systemu zastosowań.

Proces symulacji równoległej przedstawia rys. 3.

Symulacja równoległa może być realizowana w dowolnym czasie. Nie powoduje ona zanieczyszczenia zbiorów użytkownika. Kontroler może dokonywać testowania niezależnie od realizacji rzeczywistego systemu. Do przeprowadzenia symulacji wybiera on reprezentatywne dla danego systemu transakcje. Rozmiar próbki może być znacznie rozszerzony przy relatywnie niskim koszcie dodatkowym.

---

[3] W artykule pojęcia "oprogramowanie" i "software" są używane zamiennie.



Rys. 3. Schemat procesu symulacji równoległej
Ź r ó d ł o: [6, s. 213]

3.2. Metody kontroli oparte na badaniu przebiegu działania systemu

W przypadku dużych lub skomplikowanych systemów informatycznych śledzenie różnych ścieżek logicznych jest niezwykle trudne, ponieważ liczba różnych rozgałęzień może być ogromna. Kontroler, który zamierza prześledzić drogę transakcji, staje wówczas przed trudnym problemem. Jedynym możliwym rozwiązaniem tego problemu jest włączenie komputera do przygotowania danych analitycznych na podstawie

przebiegu działania systemu informatycznego. Można tu wykorzystać jedną z następujących metod:

- śledzenie transakcji,
- włączanie modułów napisanych przez kontrolującego.

3.2.1. Śledzenie transakcji

Metody tej grupy tworzą ścieżkę kontrolną przebiegu transakcji w systemie informatycznym. Mogą one występować pod dwiema postaciami [9, 10, 11]:

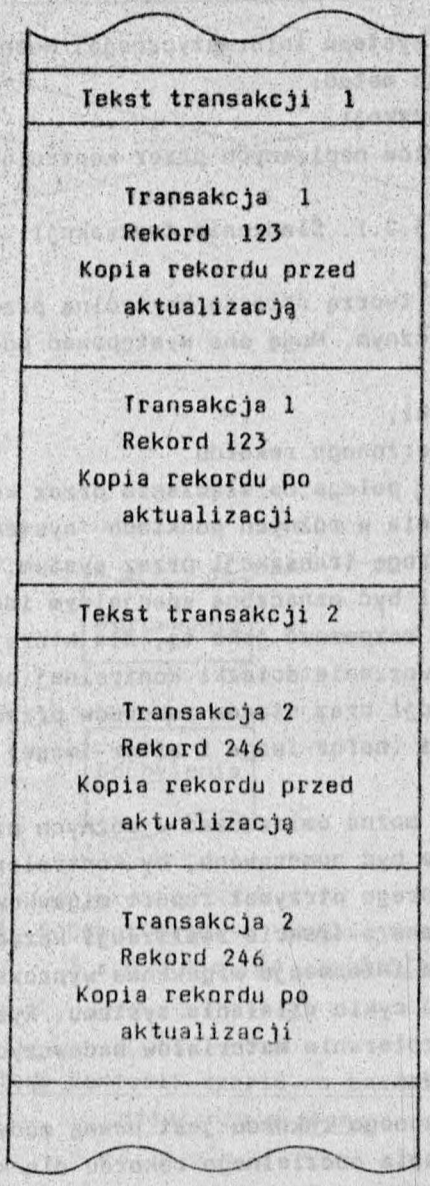
- techniki migawki,
- techniki rozszerzonego rekordu.

Technika migawki polega na włączaniu przez kontrolującego specjalnego oprogramowania w różnych punktach systemu zastosowań po to, by prześledzić drogę transakcji przez system. Wybrana transakcja (transakcje) musi być oznaczona specjalnym identyfikatorem, aby ów software mógł ją rozpoznać jako tę, dla której ma być tworzona ścieżka kontrolna. Tworzenie ścieżki kontrolnej polega na rejestrowaniu tekstu transakcji oraz stanów rekordów przed i po oddziaływaniu na nie transakcji (befor-image i after-image). Rysunek 4 przedstawia tę metodę.

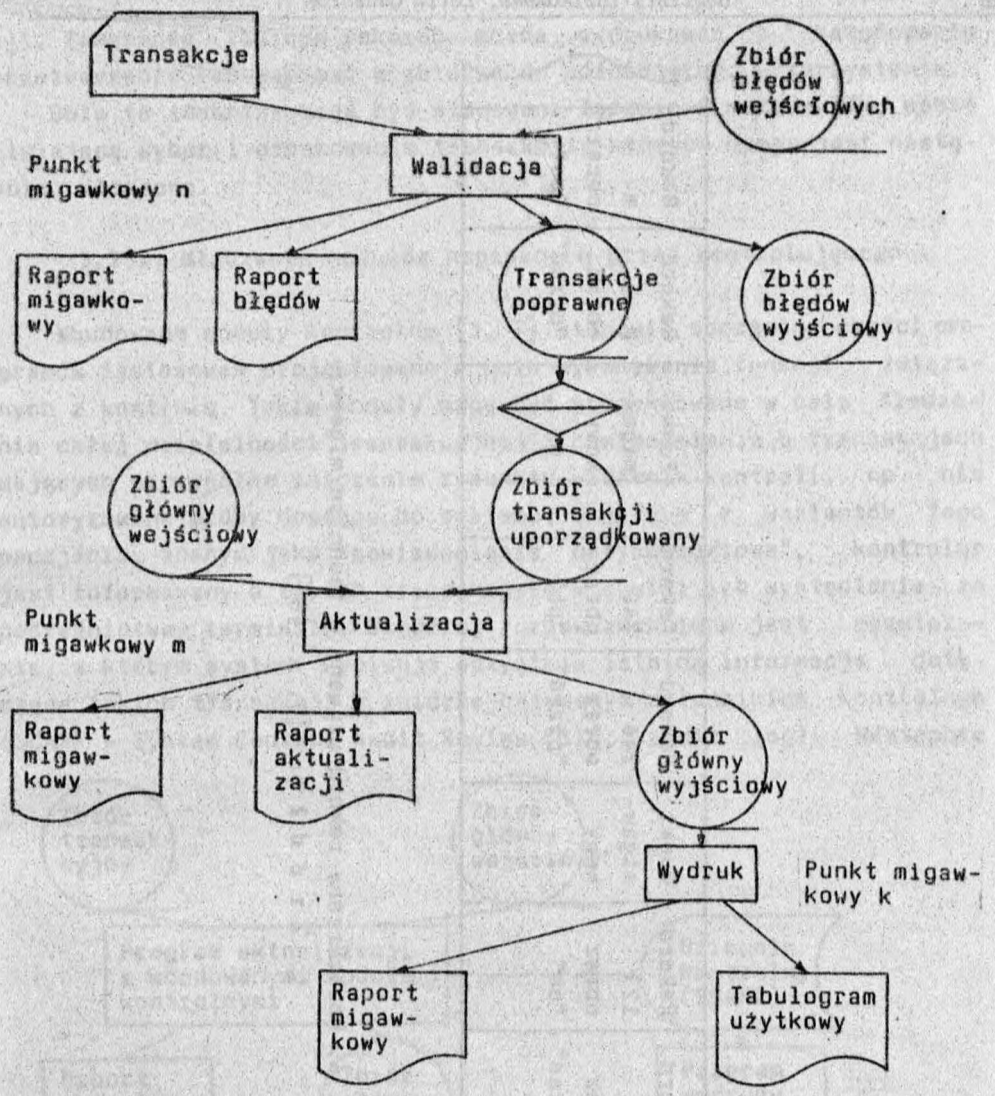
Punkty migawkowe można umieszczać w różnych przebiegach systemu zastosowań. Muszą one być numerowane, by kontroler mógł zidentyfikować przebieg, z którego otrzymał raport migawkowy. Raporty migawkowe mogą być drukowane w trakcie realizacji każdego przebiegu lub zgromadzone w zbiorze informacje migawkowe wyprowadza się na tabulogram po zakończeniu cyklu działania systemu. Rysunek 5 przedstawia schemat procesu zbierania materiałów badawczych w różnych punktach systemu zastosowań.

Technika rozszerzonego rekordu jest pewną modyfikacją techniki migawki. Zamiast pisania oddzielnego rekordu dla każdej transakcji w każdym punkcie migawkowym, tworzony jest dla każdej transakcji jeden duży rekord. Obejmuje on obrazy "przed" i "po" dla każdego punktu migawkowego, przez który przechodziła dana transakcja. Technikę tę przedstawiono na rys. 6.

Technika rozszerzonego rekordu ma tę zaletę, że gromadzi w jednym miejscu wszystkie dane migawkowe dotyczące określonej transak-



Rys. 4. Metoda śladu kontrolnego z rekordami "przed" i "po"
Źródło: [1, s. 363]



Rys. 5. Zbieranie materiałów kontrolnych w różnych punktach systemu zastosowań przy wykorzystaniu techniki migawki

Źródło: [9, s. 480]

Tekst transakcji 1	Punkt migawkowy 1	Rekord 123 obraz "przed"	Rekord 123 obraz "po"	Punkt migawkowy 2	Rekord 246 obraz "przed"	Rekord 246 obraz "po"	...	Punkt migawkowy n	Rekord m obraz "przed"	Rekord m obraz "po"

Rys. 6. Rozszerzony rekord używany w technice migawki

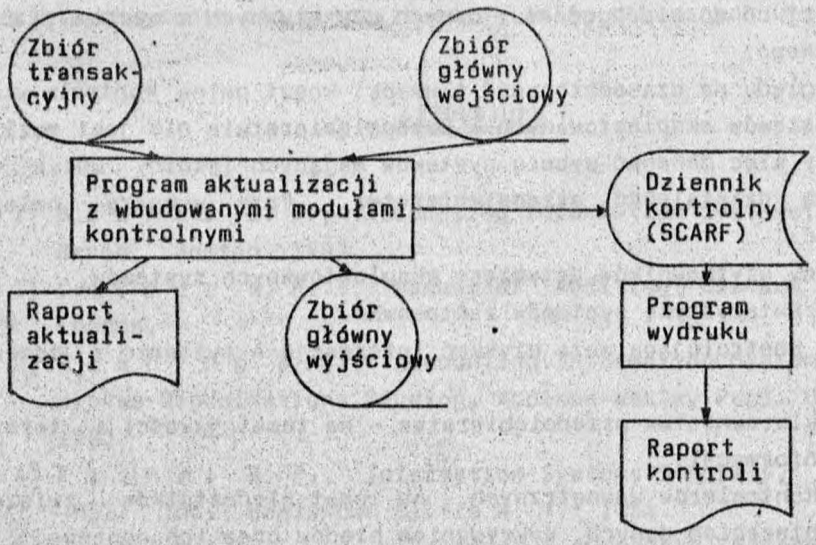
Ź r ó d ł o: [9, s. 481]

cji. Zawartość takiego rekordu można wydrukować po zakończeniu przetwarzania lub zapisać w zbiorze do późniejszego wykorzystania.

Obie te techniki mogą być stosowane łącznie z metodą ITF, umożliwiającą wybór i oznakowanie transakcji, których droga jest następnie śledzona.

3.2.2. Włączanie modułów napisanych przez kontrolującego

Wbudowane moduły kontrolne [3, 7] stanowią specjalne części programów zastosowań projektowane w celu wykonywania funkcji związanych z kontrolą. Takie moduły mogą być projektowane w celu śledzenia całej działalności transakcyjnej i powiadamiania o transakcjach mających szczególne znaczenie z punktu widzenia kontroli, np. nie autoryzowane próby dostępu do systemu. W jednym z wariantów tego podejścia, znanym jako "powiadamianie natychmiastowe", kontroler jest informowany o takich transakcjach w chwili ich wystąpienia za pośrednictwem terminalu. Bardziej rozpowszechnione jest rozwiązanie, w którym system wypisuje wszystkie istotne informacje dotyczące takich transakcji w zbiorze nazywanym dziennikiem kontrolnym (SCARF - System Control Audit Review File, audit log). Następnie



Rys. 7. Wykorzystanie wbudowanych modułów kontrolnych w programie aktualizacji zbioru głównego

Źródło: [9, s. 482]

kontroler okresowo analizuje informacje zawarte w tym zbiorze, aby zdecydować, czy jakieś aspekty systemu wymagają śledzenia. Wykorzystanie takiego zbioru przedstawia rys 7.

Metody oparte na badaniu historii działania systemu są szczególnie przydatne w sytuacjach, gdy użytkownik zleca przetwarzanie swego systemu wyspecjalizowanej instytucji lub gdy stosuje przetwarzanie rozproszone. Kontroler nie ma wówczas możliwości wykorzystania metod wymagających jego bezpośredniej obecności w ośrodku przetwarzania danych.

4. Kryteria wyboru systemu informatycznego do kontroli

Niezależnie od stosowanej metody, pełna kontrola systemu informatycznego jest czasochłonna i kosztowna. Ponadto, nie ma w Polsce tradycji przeprowadzania tego typu kontroli, a co za tym idzie, brakuje specjalistycznego oprogramowania kontrolnego; jego opracowanie wymaga również nakładów czasowych i finansowych. Konieczne jest jednak przygotowanie takiego software'u, a tymczasem należy dokonywać kontroli metodami manualnymi, które są czasochłonne i nie gwarantują pewności co do wiarygodności danych uzyskiwanych z systemu informatycznego.

Ze względu na czasochłonność i wysoki koszt pełna kontrola wszystkich systemów eksploatowanych w przedsiębiorstwie nie jest możliwa, należy więc dokonać wyboru systemów mających istotny wpływ na prawidłową działalność przedsiębiorstwa. Przy wyborze należy uwzględnić:

- oceny użytkowników dotyczące eksploatowanych systemów,
- charakterystykę systemów zastosowań.

Osoba kontrolująca może uzyskać informacje o systemie z trzech źródeł:

- od kierownictwa przedsiębiorstwa - na temat jakości i terminowości informacji,
- od kontrolerów wewnętrznych - na temat niedostatków związanych ze zbieraniem danych, wykrywaniem błędów oraz ich poprawą,
- od bezpośrednich użytkowników (brak zainteresowania systemem lub sabotowanie go).

Charakterystyka systemów zastosowań jako podstawa wyboru dotyczy pewnych aspektów systemów, mianowicie tych, które wpływają na prawidłową działalność przedsiębiorstwa. Szczególnie istotne są następujące systemy [9]:

- realizujące kontrolę finansowych zasobów przedsiębiorstwa,
- charakteryzujące się dużym ryzykiem, tzn. podatne na defraudację, czyli takie, których błędy mogą sparaliżować działalność przedsiębiorstwa lub spowodować błędy w innych systemach informatycznych,
- dotyczące tej dziedziny działalności przedsiębiorstwa, która stanowi poważne źródło jego dochodów,
- zaawansowane technologicznie, np. systemy z bazą danych, systemy rozproszone itp.
- kosztowne, ponieważ są one zwykle złożone i kryją wiele problemów dla kontroli.

Pełna kontrola powinna być zawsze przeprowadzana w odniesieniu do systemów wdrażanych lub funkcjonujących wadliwie, zwłaszcza w ocenie użytkowników. Zaleca się również warunkową kontrolę pozostałych systemów, szczególnie tych, które są modyfikowane, chociaż pozornie funkcjonują one poprawnie. Takie wyrywkowe kontrole umożliwiają wychwycenie nie autoryzowanych zmian dokonanych w systemach.

Literatura

[1] B o d n a r G. H., Accounting Information Systems, Allyn and Bacon, Boston 1983.

[2] C o o p e r V. R. V. Manual of Auditing, Gee and Co. Publ., London.

[3] C u s h i n g B. E., Accounting Information Systems and Business Organizations Reading, Addison-Wesley Publ. Comp., Mass. 1982.

[4] F i s h e r R. P., Information Systems Security, Prentice-Hall Inc., Englewood Cliffs N. Y. 1984.

[5] H o f f m a n L. J., Poufność w systemach informatycznych, WNT, Warszawa 1983.

[6] K e l l W. G., Z i e g l e r R. E., Modern Auditing, Warren, Gorham, Lamont, Boston-New York 1980.

[7] L e i t c h R. A., D a v i s R., Accounting Information System, Prentice-Hall Inc., Englewood Cliffs N. J. 1983.

[8] Prowadzenie i rewizja ksiąg rachunkowych przy technice komputerowej, SKWP, Łódź 1990.

[9] W e b e r R., EDP Auditing, MC Graw-Hill, N. Y. 1982.

[10] W i e d e r h o l d G., Database Design, Mc Graw-Hill, N. Y. 1977.

[11] Y o u r d o n E., Projektowanie systemów o działaniu bezpośrenim, WNT, Warszawa 1976.

Gabriela Idzikowska, Zofia Owczarek

METHODS OF CONTROLLING THE FUNCTIONING OF INFORMATION SYSTEMS

The article contains a review of the methods of controlling the functioning of information systems. The use of these methods guarantees correctness of data contained in files, which is an indispensable condition for effective management of an enterprise. There have been presented two groups of control methods:
- system software,
- application software.