

Projektowanie systemów informatycznych - konspekt do zajęć z tematów:

**Strukturalna metodyka projektowania SI.
Narzędzia szczegółowe - diagramy przepływu danych, słowniki danych, specyfikacje procesów (pseudokod).**

Literatura podstawowa:

- Robertson, J., Robertson, S. (1999) Pełna analiza systemowa. WNT, Warszawa
- Yourdon, E. (1996) Współczesna analiza strukturalna. WNT, Warszawa

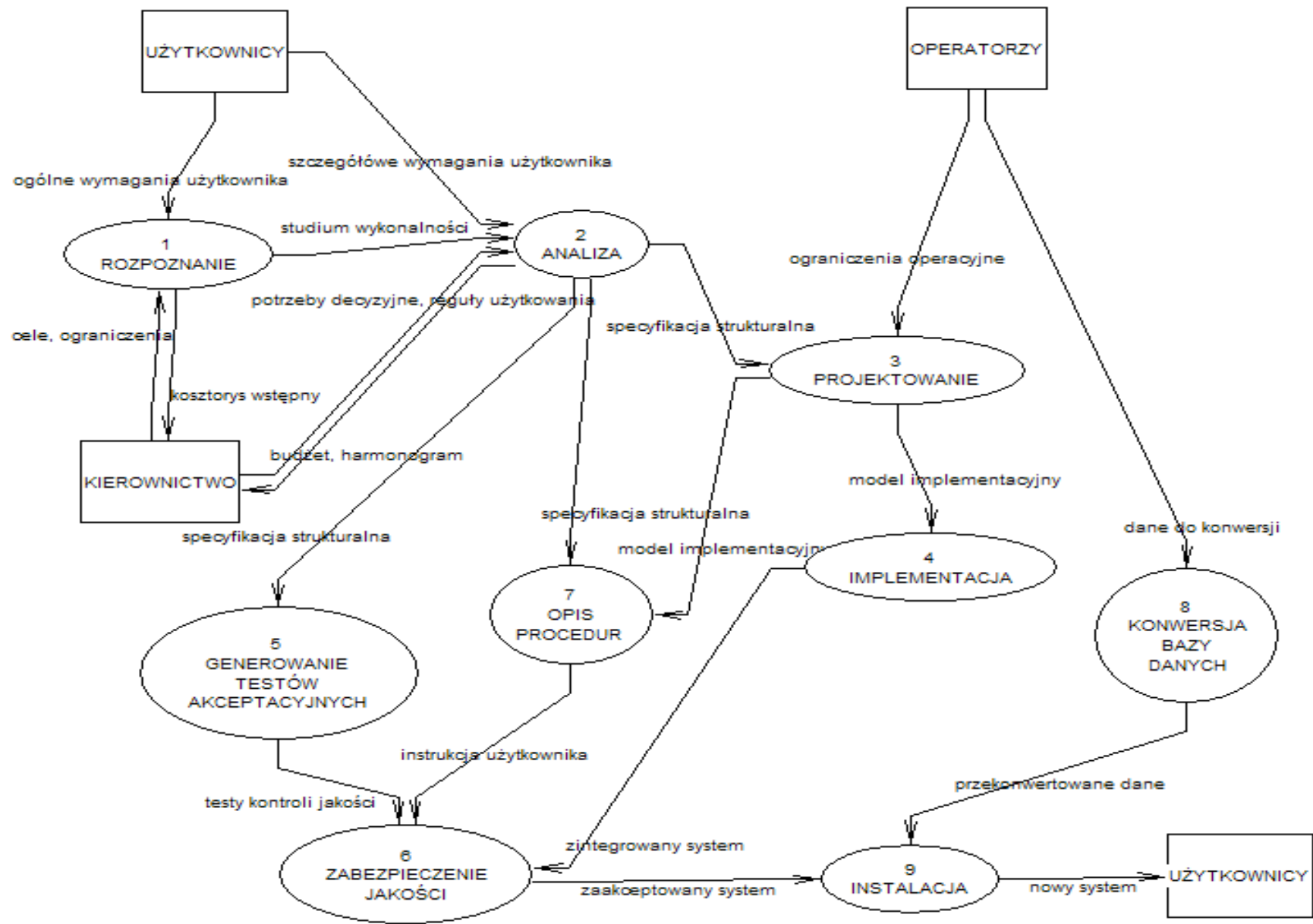


Diagram przepływu danych - Strukturalny cykl życia projektu tworzenia SI (Yourdon. Współczesna analiza strukturalna)

Metodyka tworzenia SI to spójny i logicznie uporządkowany zestaw **modeli, narzędzi i procedur** o charakterze technicznym i organizatorskim pozwalającym zespołowi projektowemu realizować cykl życia systemu

metodyka to dlaczego? jakimi modelami ? w jakiej kolejności ? jakimi narzędziami?

Dokumentacja projektowa w metodyce strukturalnej

- Proces analizy strukturalnej – Model podstawowy
- Proces projektowania - Model implementacyjny

Model podstawowy - za jego pomocą przedstawiamy, **co** powinien robić system, aby spełnić wymagania użytkownika

Uwaga!! Nie odpowiada na pytanie jakimi metodami należy go zaimplementować

Nie zwracamy uwagi na takie elementy jak:

- granice automatyzacji
- interfejs użytkownika
- konfiguracje sprzętowe
- narzędzia implementacyjne

Model podstawowy składa się z:

- modelu środowiskowego
- modelu zachowania

Model środowiskowy definiuje **granice** między analizowanym systemem a otoczeniem - środowiskiem

Model środowiskowy - elementy

1. Cele systemu
2. Lista zdarzeń
3. Diagram kontekstowy

Model zachowania (behawioralny) opisuje wymagane zachowanie wewnętrzne systemu niezbędne do poprawnej interakcji ze środowiskiem

Model zachowania - trzy aspekty

4.przetwarzania

model funkcjonalny

5.danych

model danych

6.zachowania systemu w czasie

model zmian stanu

systemu

Narzędzia stosowane w Modelu Zachowania

Aspekt przetwarzania

- Diagramy przepływu danych (DFD - dataflow diagrams)
- Słowniki danych (DD – data dictionary)
- Specyfikacje procesów (PSPEC – process specifications)

Aspekt danych

- Diagramy związków obiektów/encji (ERD – entity relationship diagrams)

Aspekt zachowania systemu w czasie

- Diagramy sieci przejść (STD – state transition diagrams)

Diagramy przepływu danych

Diagram przepływu danych – ilustruje procesy, które musi realizować system. Pozwala pokazać system jako sieć procesów funkcyjnych połączonych potokami zasilającymi i zbiornikami danych.

Synonimy DFD: model procesów, diagram bąbli, model funkcji

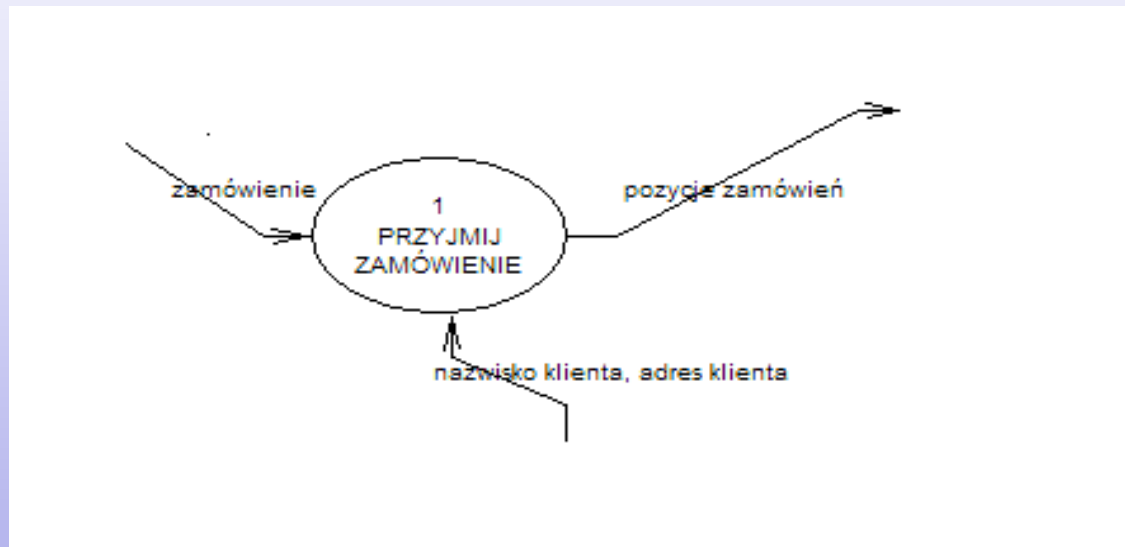
Elementy DFD:

- Procesy
- Przepływy
- Magazyny danych (Składnice)
- Terminatory

Procesy – rozmaite funkcje, które realizuje system.
Funkcje/procesy przekształcają wejścia na wyjścia

Odpowiadają tym składnikom systemu, które **operują**
na danych

- Dokonują transformacji przepływów wejściowych w przepływy wyjściowe



Zasady stosowania:

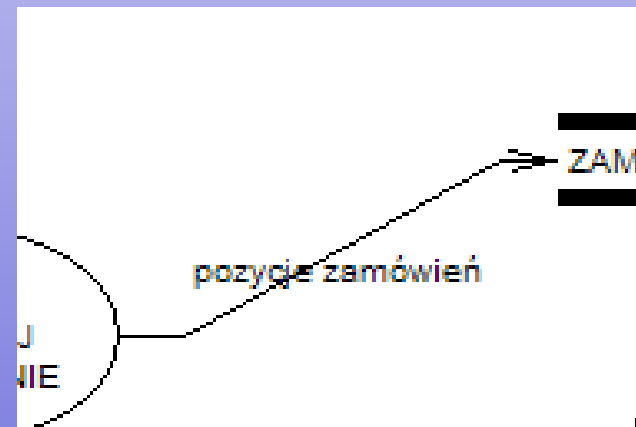
Zasada asynchroniczności - proces jest uaktywniany przez dane wchodzące i nie ma informacji o innych procesach

Zasada zupełności danych – proces musi otrzymać wszystkie potrzebne dane do poprawnego wyprodukowania wyników

Przepływy danych - tworzą związki pomiędzy procesami (funkcjami systemu) i reprezentują informację wejściową i wyjściową, wygenerowaną przez proces. Służą do przenoszenia jednostek lub pakietów informacji z jednego fragmentu systemu do drugiego - **dane w ruchu**

Opisują strumienie danych (niekoniecznie w formie elektronicznej) o określonej zawartości przepływające pomiędzy dwoma składnikami DFD:

- terminatorami a procesami,
- procesami a procesami lub
- procesami a składnikami danych

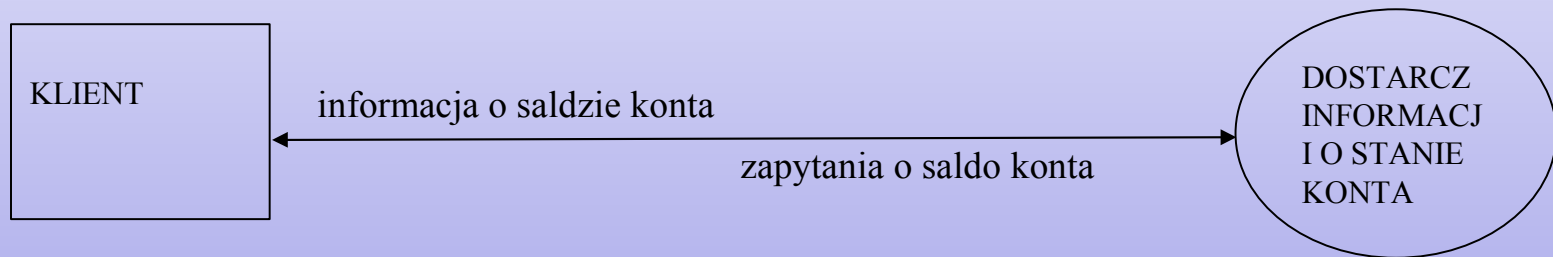


Rodzaje przepływów:

1. przepływy rozbieżne



2. przepływy dialogowe

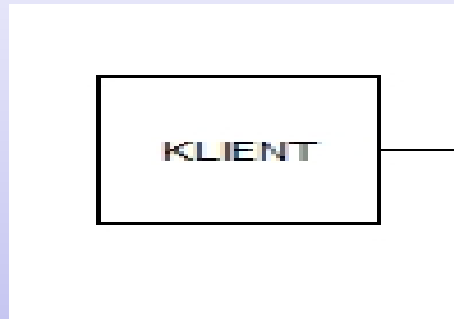


Zasady stosowania

- przepływ nie udziela informacji o harmonogramie pobierania danych
- przepływ nie udziela informacji o sposobie przesyłania danych

Terminatory - obiekty zewnętrzne, z którymi komunikuje się system

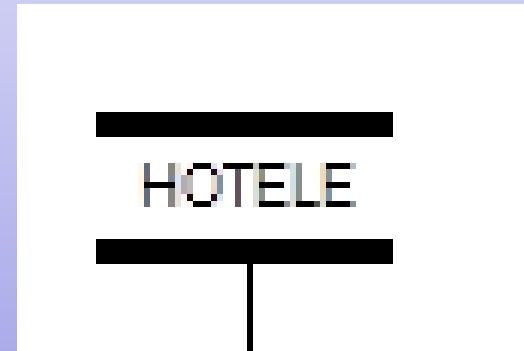
- Reprezentują źródła lub miejsca przeznaczenia informacji, które są zewnętrzne w stosunku do systemu
- Mogą nim być: osoba, instytucja, przedsiębiorstwo - odbiorca, system informatyczny, dział wewnętrzny firmy (tej dla której tworzymy system)



Zasady stosowania

Nie można pokazywać w diagramie związków pomiędzy terminatorami

Składnice (magazyny) danych – pokazują zbiory danych, które system powinien przechowywać przez pewien czas. **Dane w bezruchu**



Zasady stosowania

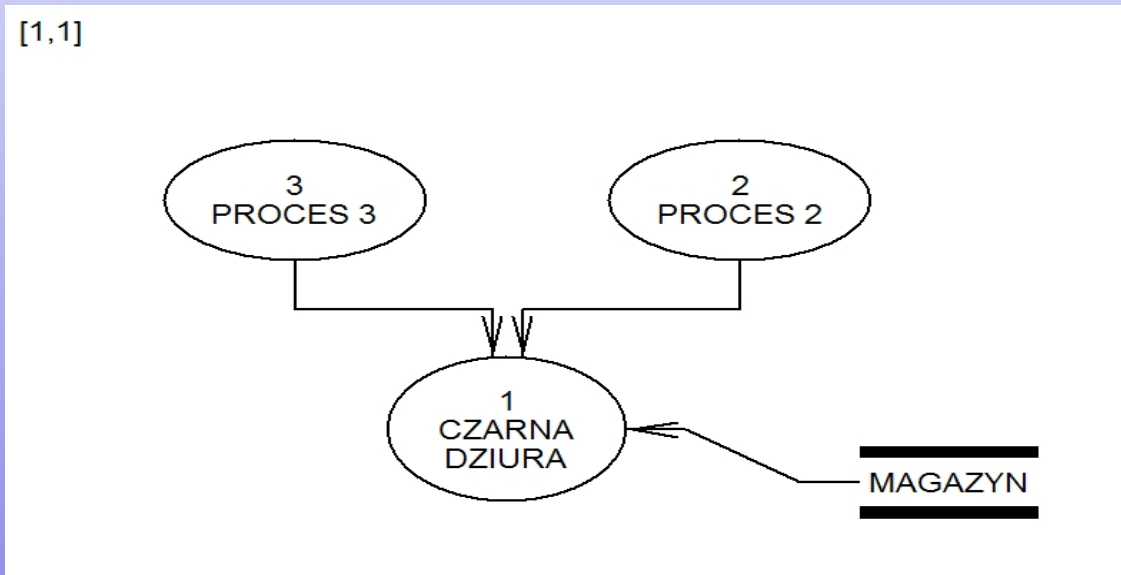
Magazyny są dostępne tylko dla procesów co oznacza, że magazyn danych nie może się łączyć bezpośrednio z terminatorem

Wskazówki dotyczące konstruowania DFD

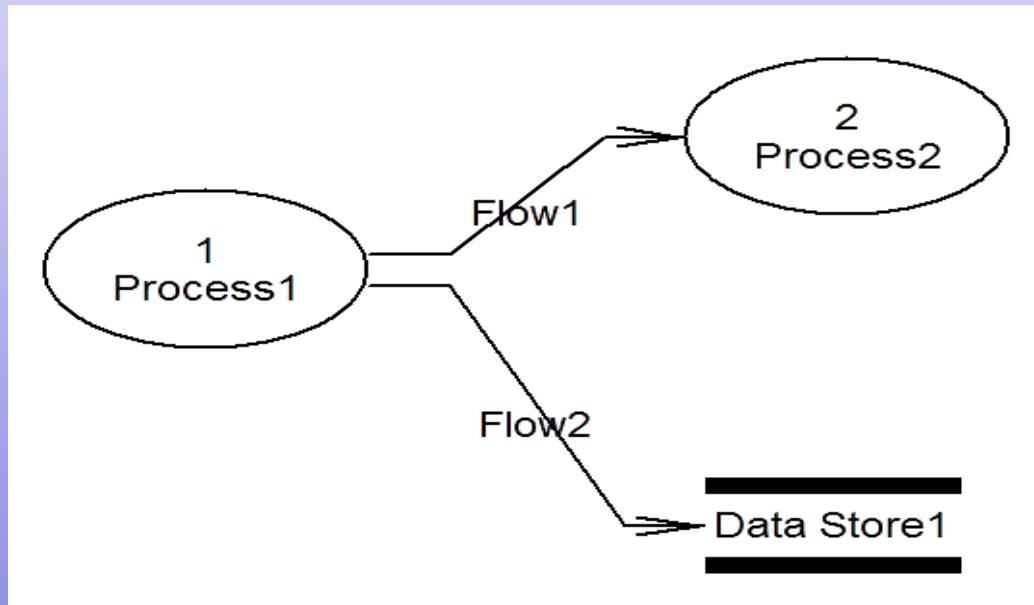
3. Wybieraj znaczące nazwy dla procesów, przepływów, magazynów i terminatorów
4. Numeruj procesy
5. Przerysowuj DFD – pamiętając, że użytkownik, projektant i programista muszą je zrozumieć
6. Unikaj nadmiernie złożonych DFD
7. Unikaj logicznych sprzeczności DFD

Zasady badania niesprzeczności diagramu

1. unikaj nieskończonych studni, czyli procesów które mają same wejścia a żadnych wyjść - **Czarne dziury**



2. unikaj procesów spontanicznej generacji, czyli takich które mają jedynie wyjścia



3. strzeż się nieetykietowanych procesów i przepływów
(wyjątek stanowią zapisy i odczyty pełnych pakietów danych
do i z magazynów)

4. strzeż się magazynów tylko-do-zapisu lub tylko-do-odczytu

Praktyka projektowania - prezentacja projektu Systemu Informacyjnego Biblioteki Publicznej (SIBP)

Model środowiskowy definiuje nam **granice** między systemem a otoczeniem/środowiskiem

Dodatkowym zadaniem modelu jest określenie interfejsów pomiędzy projektowanym systemem a otoczeniem

Elementy modelu środowiskowego

- Cele systemu
- Diagram kontekstowy
- Lista zdarzeń

Model środowiskowy Systemu Informacyjnego Biblioteki Publicznej - SIBP

Cel systemu:

Celem systemu SIBP jest gromadzenie informacji o zbiorach bibliotecznych oraz o operacjach ich udostępniania.

Wprowadzenie systemu powinno doprowadzić do:

- skrócenia czasu obsługi operacji wypożyczenia i zwrotu książki o połowę
- skrócenia czasu udzielenia informacji o książkach o 85% (3 min – 0,5 min)

Diagram kontekstowy to narzędzie służące do poznania zakresu działania systemu – prezentuje projektowany system jako jeden proces

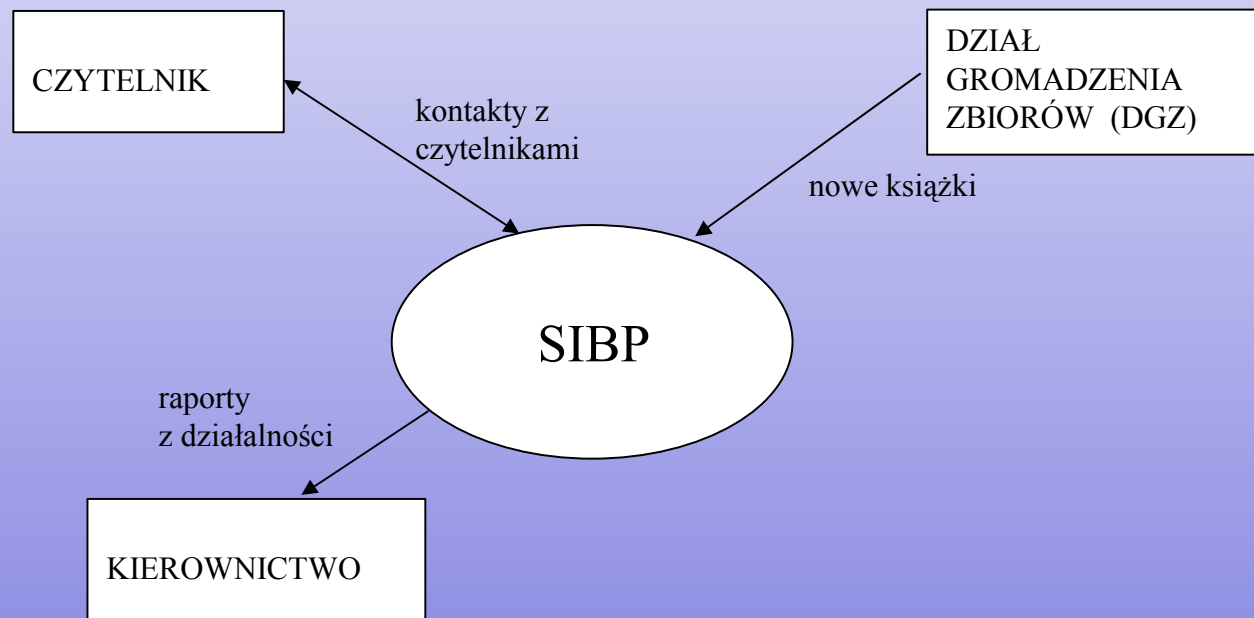


Diagram kontekstowy Systemu Informacyjnego Biblioteki Publicznej

Lista zdarzeń to tekstowa lista „bodźców” występujących w świecie zewnętrznym, na które musi odpowiadać system

Lista zdarzeń SIBP

2.Czytelnik chce się zarejestrować

3.Czytelnik prosi o informację o książce

4.Czytelnik wypożycza książkę

5.Czytelnik zwraca książkę

6.Czytelnik prosi o informacje o aktualnym stanie konta

7.Czytelnik zgłasza informację o niemożności zwrotu książki

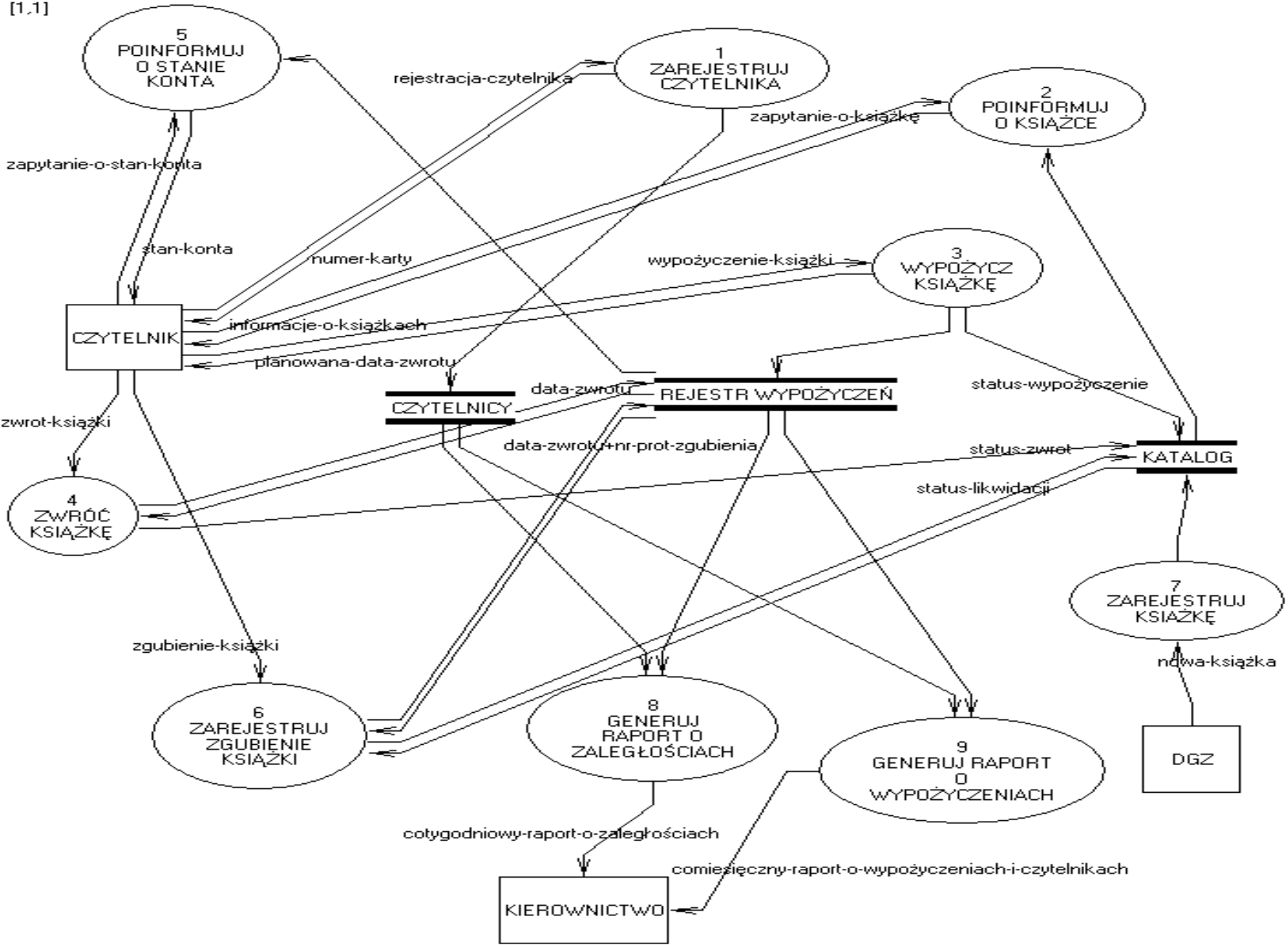
8.Dział Gromadzenia Zbiorów (DGZ) przysyła nową książkę

9.Kierownictwo potrzebuje (cotygodniowy) raport o bieżących zaległościach (T)

10.Kierownictwo potrzebuje (comiesięcznych) raportów o ilości wypożyczonych książek oraz o ilości nowych czytelników (T)

Taktyka tworzenia pełnego DFD (w metodzie Yourdona):

1. Wstępny DFD - diagram odpowiedzi na zdarzenia
2. Pełny DFD powstaje poprzez procesy **kompozycji** i **dekompozycji** diagramu odpowiedzi na zdarzenia



SIBP wstępny DFD - diagram odpowiedzi na zdarzenia

Aby tworzyć modele złożonych systemów (a więc o dużej ilości procesów) i nie łamać zasad złożoności należy zastosować **kompozycję i dekompozycję diagramu** - utworzyć diagram wielopoziomowy

Zasady kompozycji i dekompozycji

W górę

- Łączymy procesy operujące na wspólnych zasobach danych lub wspólnych obszarach wynikających z kontekstu systemu. Pamiętamy zasadę -maksymalna ilość procesów na jednym diagramie to siedem

W dół

- Dekomponujemy DFD do poziomu procesów elementarnych

Procesy elementarne to takie, których nie trzeba już rozpisywać na niższe poziomy – ich specyfikację można przedstawić na pojedynczej stronie A4 – (minispecyfikacja)

- Procesy mogą być nierównomiernie rozbijane na podpoziomy. Jedne procesy mogą być do opisanie za pomocą dwóch poziomów inne czterech (na tym samym pełnym DFD) i jest to sytuacja poprawna

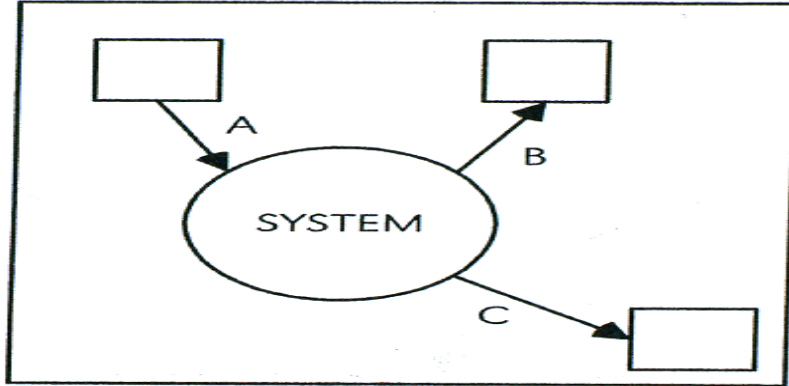


DIAGRAM KONTEKSTOWY

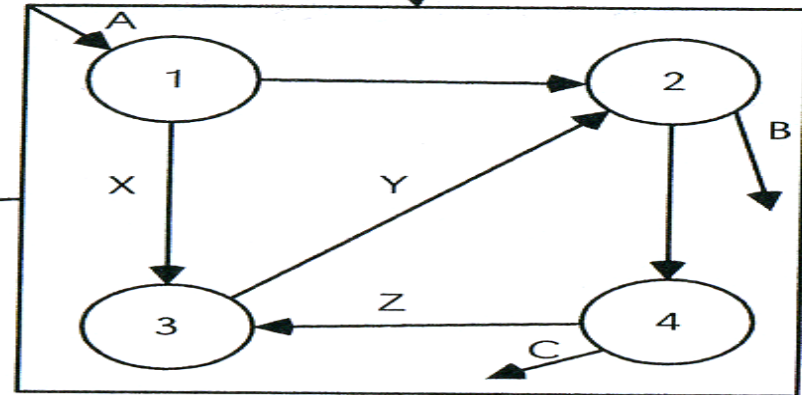


DIAGRAM 0

PROCES 3

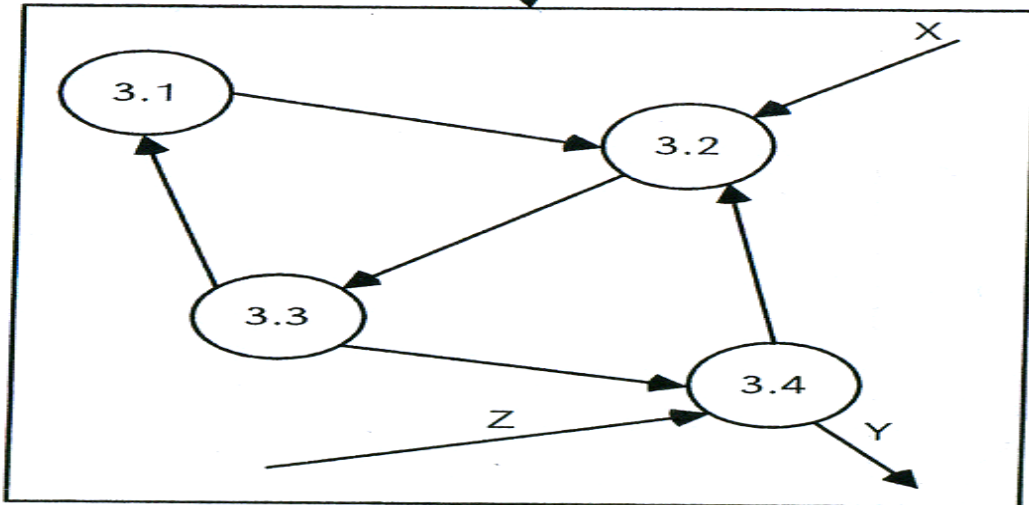


DIAGRAM 3

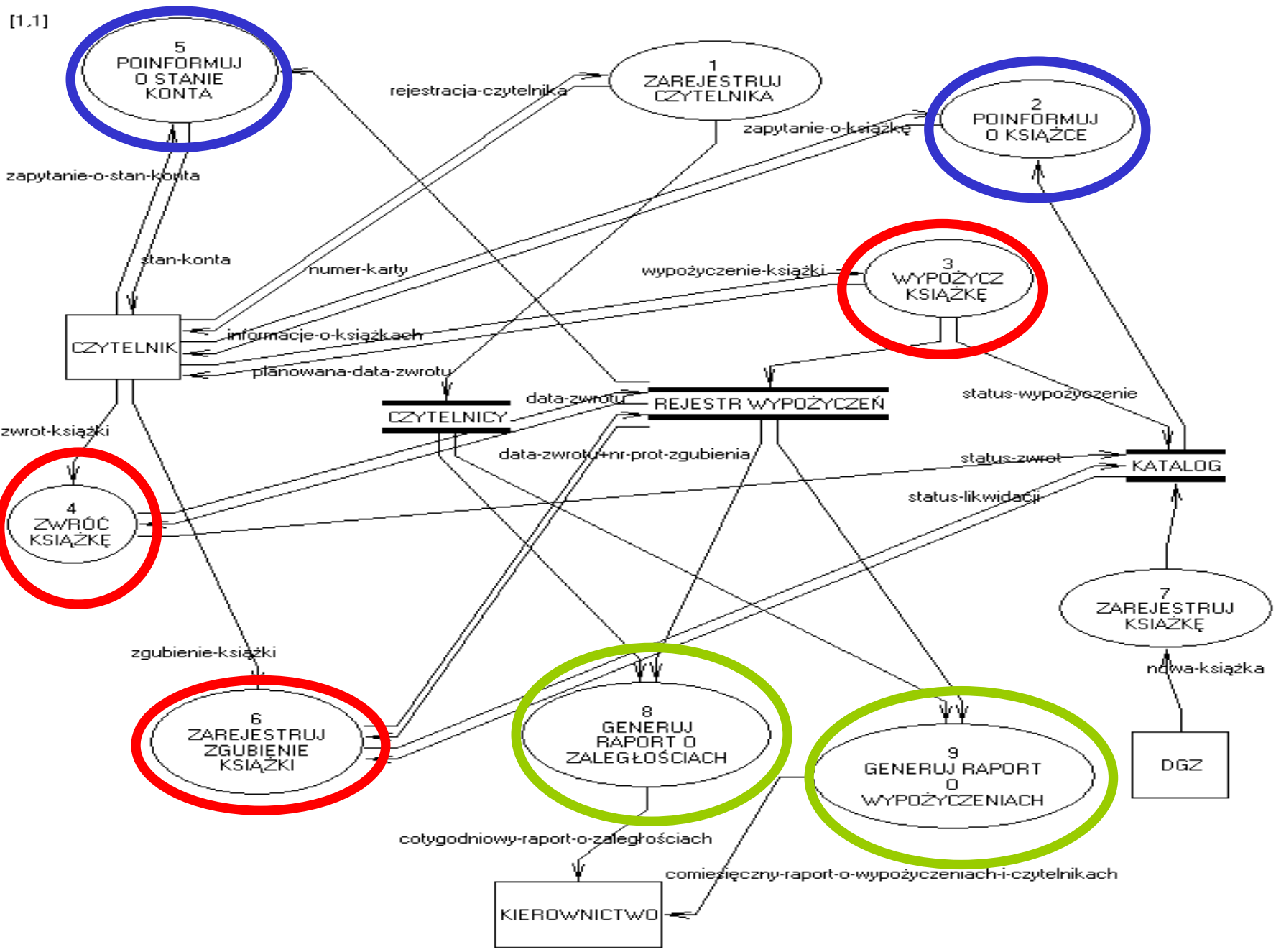
Fragment pełnego DFD (Yourdon.
Współczesna analiza strukturalna)

Równowazenie wewnętrzne DFD

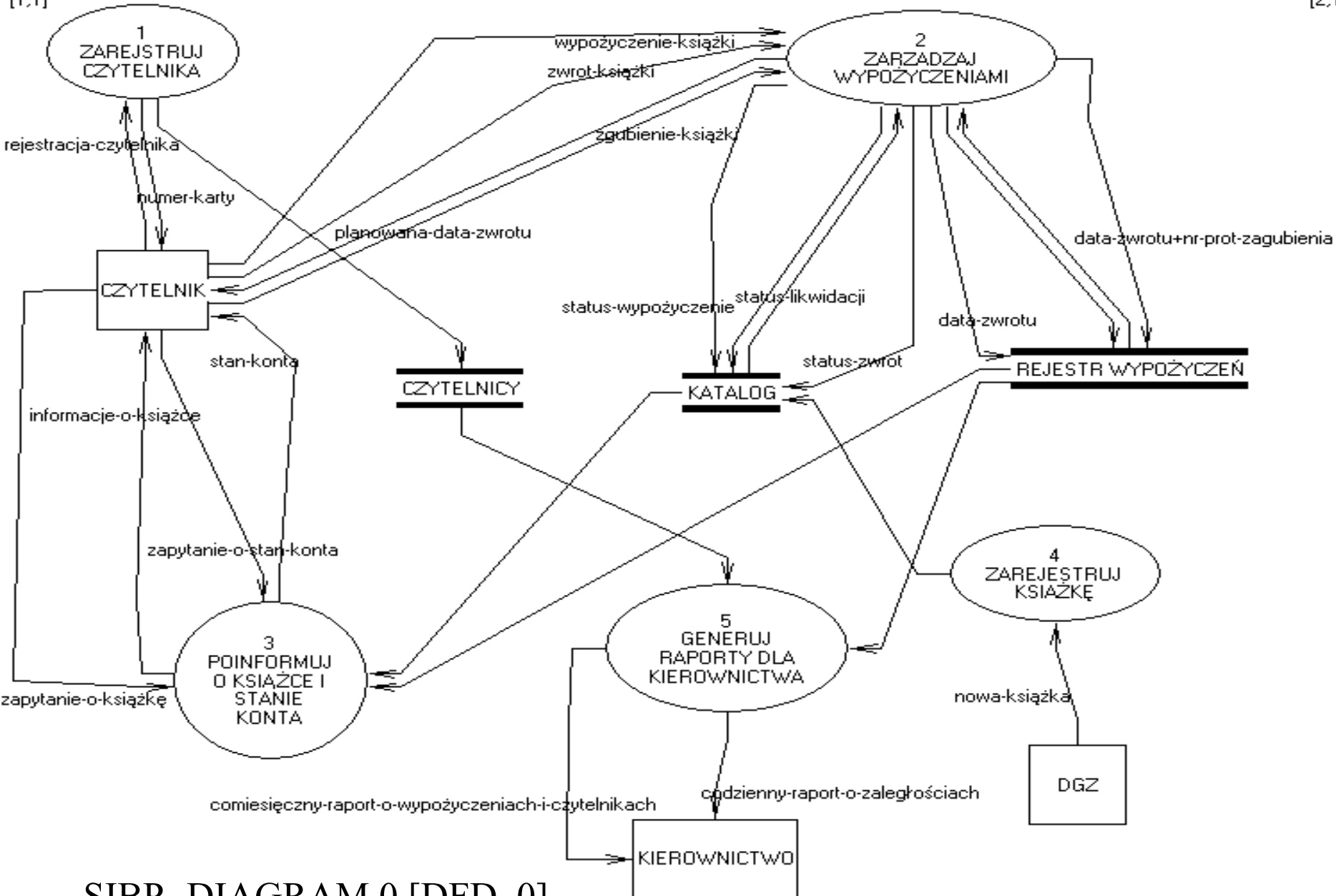
Zasada zachowania zgodności pomiędzy poziomami diagramu (Yourdon) - przepływy wchodzące i wychodzące z procesu na danym poziomie powinny być zgodne z diagramem reprezentującym ten proces na niższym poziomie lub

Zasada równowagi modelu (Roberstsonów)

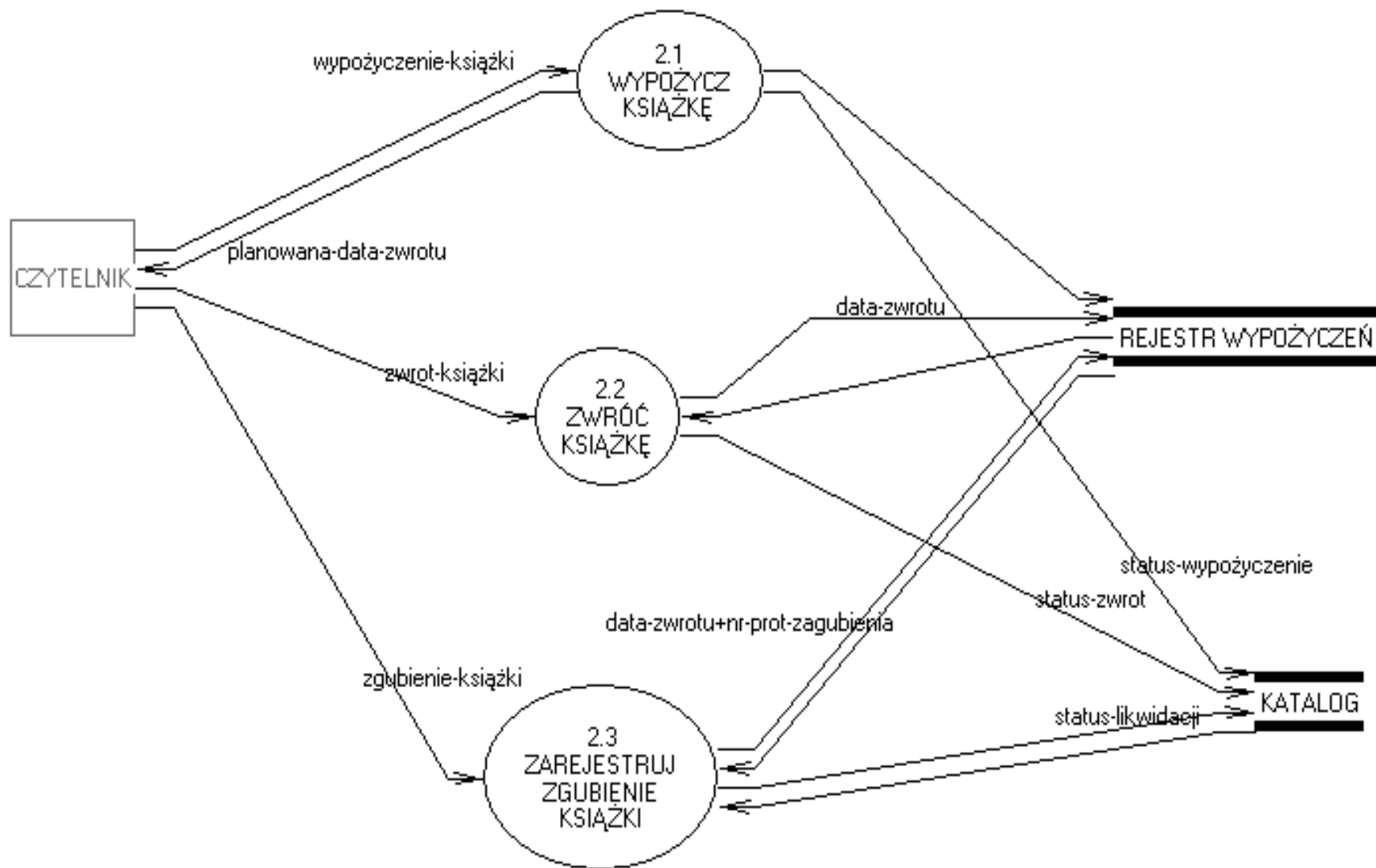
Diagram dziecko przetwarza te same dane co diagram ojciec



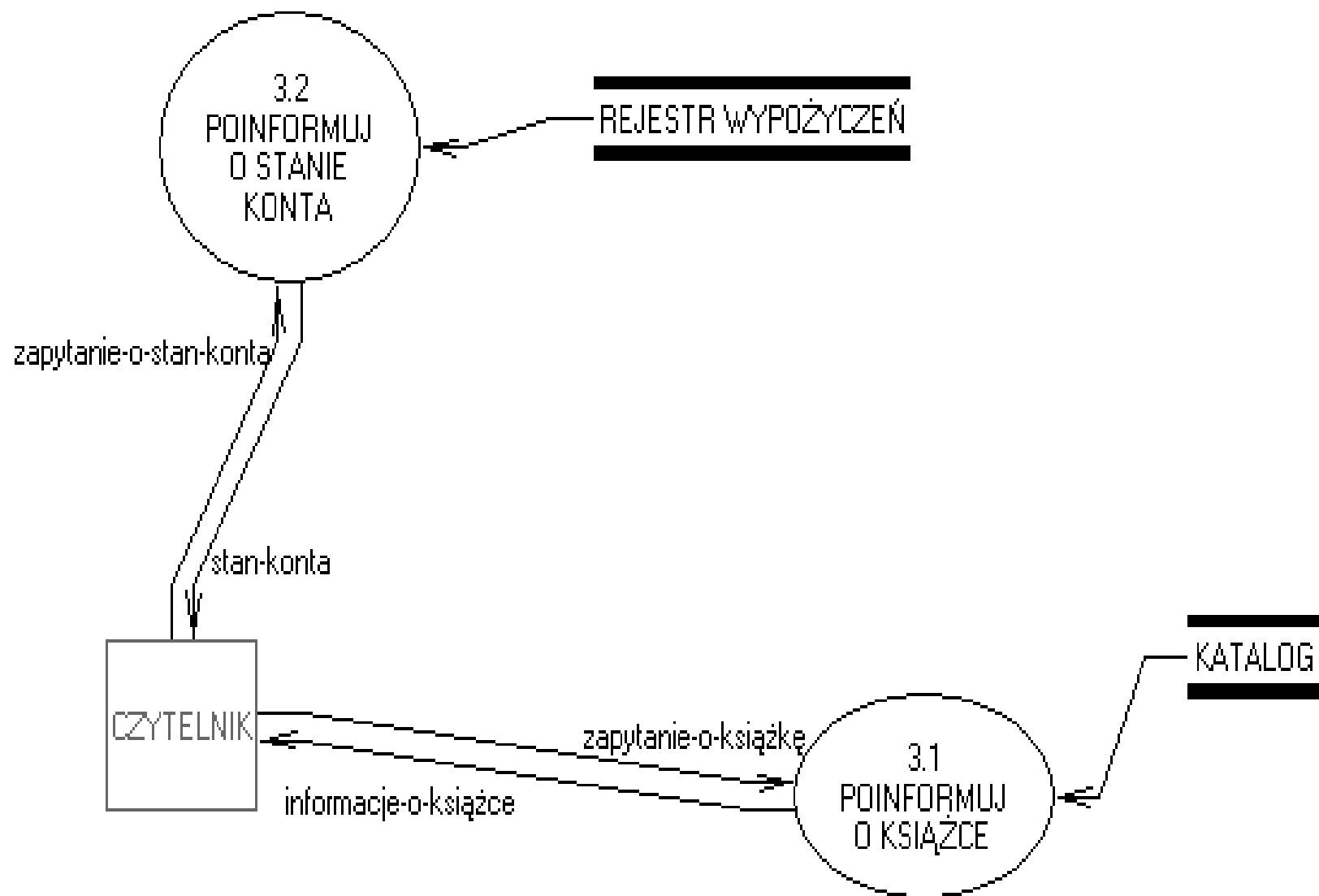
SIBP wstępny DFD - diagram odpowiedzi na zdarzenia. Łączenie procesów.



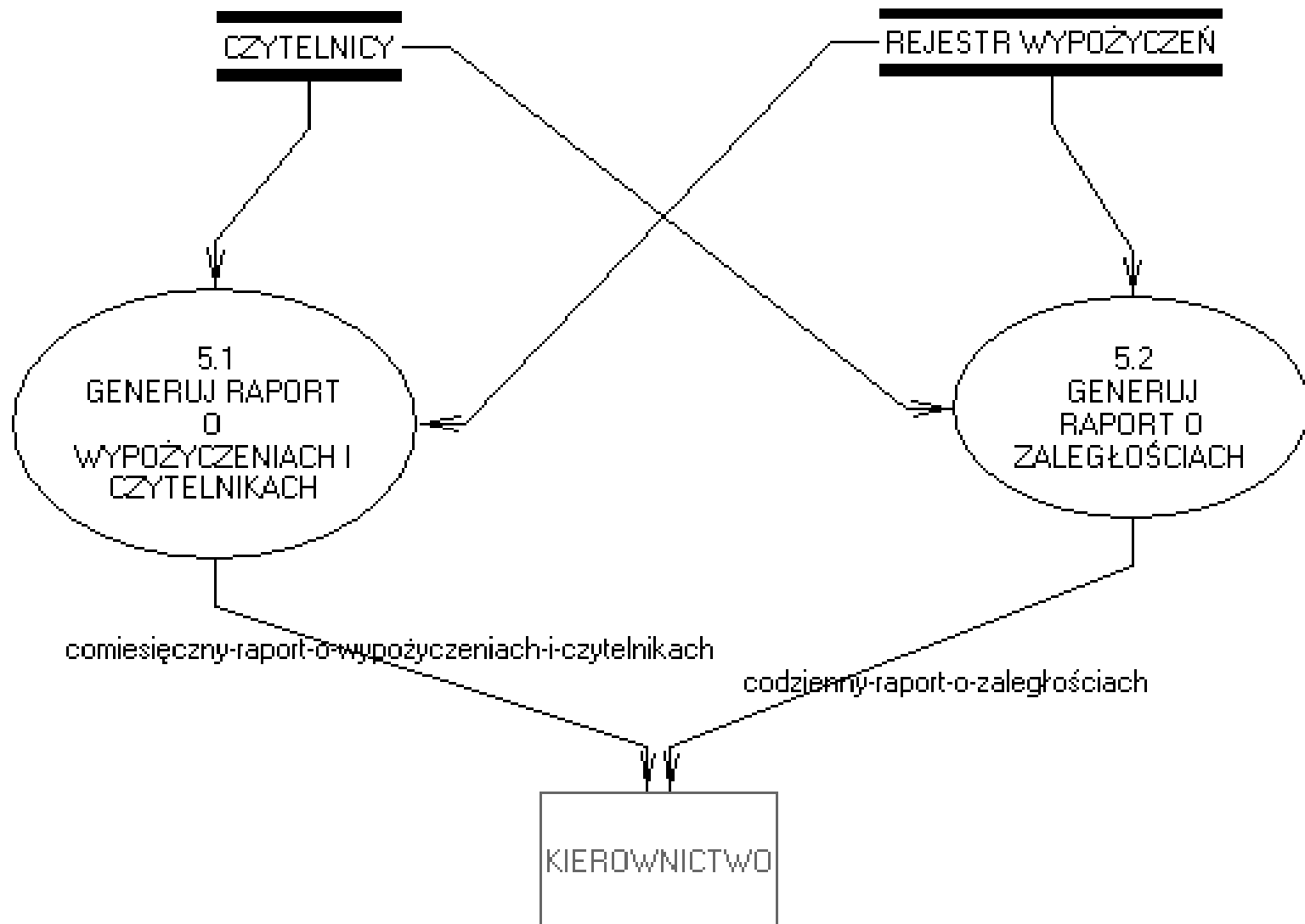
SIBP- DIAGRAM 0 [DFD_0]



SIBP- DIAGRAM 2 [DFD_2] ZARZĄDZAJ WYPOŻYCZENIAMI



SIBP- DIAGRAM 3 [DFD_3] POINFORMUJ O KSIĄZCE I STANIE KONTA



SIBP- DIAGRAM 5 [DFD_5] GENERUJ RAPORTY DLA KIEROWNICTWA

Słowniki danych (DD – data dictionary)

Słownik danych to uporządkowany wykaz wszystkich elementów danych mających związek z systemem, wraz z ich precyzyjnym określeniem.

Po co?

Abyś ty analityk i każda osoba korzystająca z dokumentacji projektowej (w szczególności z diagramów: DFD i ERD oraz specyfikacji) jednakowo rozumieli wszystkie wejścia, wyjścia, składniki magazynów oraz obliczenia pośrednie.

Rodzaje elementów występujących w słowniku danych

- **Dane elementarne to takie, które nie podlegają dalszemu rozbiciu w kontekście użytkownika**

np.

nazwisko-autora

sygnatura

numer-karty

- **Pakiet to zbiór danych elementarnych - najczęściej opisuje budowę magazynów i przepływów**

pakiet = dana-elementarna+dana-elementarna+inny-
pakiet+dana-elementarna

np.

adres-klienta=miasto+kod-pocztowy+ulica+nr-domu

Ogólna zasada budowania słownika danych

Złożone pakiety definiuje się za pomocą danych elementarnych, a te opisuje się w terminach zakresów wartości i jednostek miary.

Poprawna definicja elementu danych musi zawierać:

- znaczenie elementu danych w kontekście aplikacji użytkownika - notacja komentarza
- budowę elementu danych, jeśli składa się ze składników elementarnych
- wartości, jakie może przyjmować element danych, jeśli jest składnikiem elementarnym

Notacje

1. = +

= to kombinacja składników tworzących przepływ/magazyn

+ tym znakiem łączymy kolejne składniki

czytelnik = numer-karty+nazwisko+imię+typ-dokumentu+numer-dokumentu+data-urodzenia+miejsce-urodzenia+kod+miasto+typ-ulicy+ulica+telefon+data-rejestracji

2. {}

{to co w nawisach klamrowych może się powtarzać}

Zamówienie = id-klienta+data-zamówienia+{nazwa-towaru+ilość}

Granice iteracji: 1 {nazwa-towaru+ilość} 1 1

3. ()

(elementu w nawiasie okrągłym może nie być – jest opcjonalny)

czytelnik = nazwisko+imię+(telefon)+(email)

4. []

[w nawiasy kwadratowe ujmuje się elementy wyboru. Poszczególne możliwości oddziela się znakiem „|”]

typ-dokumentu = [”Dowód osobisty”|”Paszport”|”Legitymacja studencka”]

5. **

to jest komentarz

6. @

@ przed składnikiem elementarnym oznacza, iż jest on kluczem identyfikującym kolejne wystąpienia pakietu w składnicy danych

czytelnik = @numer-karty+nazwisko+imię+data-urodzenia+miejsce-urodzenia+kod+miasto+ulica+telefon+data-rejestracji

7. alias (synonim)

Zamawiający = *synonim dla klienta*

Klient = *patrz zamawiający*

Specyfikacje procesów (PSPEC – process specifications)

Specyfikacja procesu definiuje (opisuje), co należy zrobić w celu przekształcenia wejścia na wyjścia.

Metody (niektóre) specyfikacji procesów:

- strukturalizowany opis w języku polskim
- tablice decyzyjne
- drzewa decyzyjne
- pseudokod

Podstawowym wymaganiem poprawnej specyfikacji procesów, jest to iż musi być **zrozumiała i weryfikowalna przez zespół projektowy**

Cztery zasady tworzenia zdań w pseudokodzie

1. Czasowniki należy dobierać z następującej listy:

• operacje na danych z przepływów i magazynów

WCZYTAJ (WYBIERZ, POBIERZ),

USUŃ, USTAW, TWÓRZ, ZAPISZ, DOŁĄCZ, SORTUJ

• operacje arytmetyczne

OBLICZ, DODAJ, ODEJMIJ

POMNÓŻ, PODZIEL

• operacje komunikacyjne

WYŚWIETL

WYŚLIJ

2. Elementy znajdujące się w dopełnieniach zdań najczęściej określają dane, które przetwarza proces – a więc muszą się one znajdować w **słowniku danych**

przykład

WCZYTAJ rekord *wypożyczenia* z *REJESTRU WYPOŻYCZEŃ* z *sygnaturą=sygnatura* ze *zwrot-książki* i *datą-zwrotu=pusta*

3. Konstrukcje warunkowe

- IF-THEN-ELSE-ENDIF

```
IF warunek THEN
  zdanie-1
ELSE
  zdanie-2
ENDIF
```

przykład

```
IF dochód-pracownika<30000 THEN
  podatek=20%
ELSE
  podatek=40%
ENDIF
```

•CASE

DO CASE zmienna=wartość-1

zdanie-1

CASE zmienna=wartość-2

zdanie-2

.

.

CASE zmienna=wartość-n

zdanie-n

OTHERWISE

zdanie-n+1

ENDCASE

przykład

DO CASE dochód-pracownika<20000

 podatek=10%

CASE dochód-pracownika>20000 i dochód-pracownika<30000

 podatek=20%

OTHERWISE

 podatek=50%

ENDCASE

•DO WHILE

DO WHILE warunek-1

zdanie-1

ENDDO

przykład

suma-wypożyczeń=0

DO WHILE istnieje więcej wypożyczeń w REJESTRZE WYPOŻYCZEŃ z
datą-wypożyczenia w raportowanym miesiącu

DODAJ suma-wypożyczeń+1

ENDDO

WYŚWIETL suma-wypożyczeń

4. Terminy lokalne - definiowane na potrzeby procesu

przykład

suma-dzienna=0

DO WHILE istnieją zamówienia w ZAMÓWIENIA z datą-faktury =
obecnej dacie

 WCZYTAJ następne zamówienie z ZAMÓWIENIA

 DODAJ do raportu-zamówień numer-faktury+suma-całkowita-
zamówienia

 suma-dzienna=suma-dzienna+ suma-całkowita-zamówienia

ENDDO

DODAJ do raportu-zamówień suma-dzienna