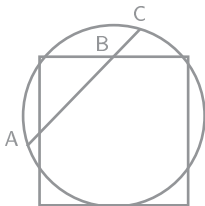


# Inżynieria oprogramowania

Radosław Klimek

2015-23

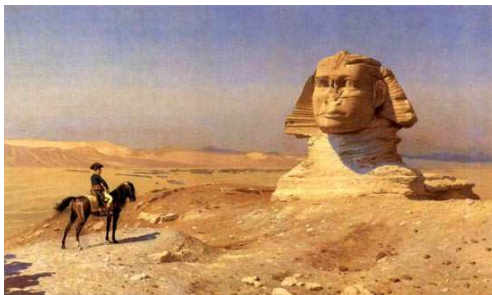


<http://home.agh.edu.pl/rklimek>

# 1 Analiza strukturalna: model logiczny

# 1 Analiza strukturalna: model logiczny

# Analiza strukturalna: model logiczny



Jean-Leon GEROME: *Edyp (lub Napoleon przed Sfinksem)*

# Diagramy DFD

Podstawowe fakty/wiadomości odnośnie diagramów DFD:

- ✓ elementarne narzędzie opisu funkcjonalnego systemu;
- ✓ pokazują proces/procesy przetwarzania danych wejściowych na dane wyjściowe;
- ✓ diagramy pozwalają zidentyfikować i zdefiniować przepływy danych (jednak nie są algorytmem), przy czym przepływy mogą być jedno- lub dwukierunkowe;
- ✓ definiują miejsca przechowywania danych (magazyny lub zbiory danych);

## Diagramy DFD (cd.)

- ✓ definiują niezbędny dostęp do danych jaki jest realizowany przez użytkownika, pokazują obiekty zewnętrzne oddziałujące z (analizowanym) systemem;
- ✓ obrazują przemieszczanie się danych pomiędzy poszczególnymi częściami systemu, nie jest jednak ściśle pokazane jak odbywa się przemieszczanie (postać danych);
- ✓ pokazują także jak dane są przetwarzane, przy czym nie pokazuje się jak dokładnie wygląda samo przetwarzanie.

# Dualizm w diagramach DFD

W diagramach DFD występuje **dualizm** w odniesieniu do informacji:

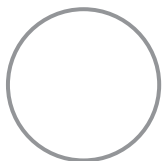
przepływ  
informacji

- co przychodzi?
- co się zmienia?
- co wychodzi?

struktura  
informacji

- jakie są poszczególne elementy?
- jak przedstawiają się razem?
- czy pewne ich grupy mogą być połączone?
- czy istnieją relacje pomiędzy grupami elementów?

## Składniki diagramów DFD (Yourdon, DeMarco)



proces, akcja



przepływ danych,  
połączenie



pamięć,  
magazyn danych



obiekt zewnętrzny,  
element terminalny



## Znaczenie poszczególnych elementów

- 1 Proces pokazuje fragment systemu przekształcający dane, czasem może to być pojedyncza (drobna) funkcja – „oblicz podatek”, a czasem może odzwierciedlać np. prace całego działu, departamentu czy grupy osób.

## Znaczenie poszczególnych elementów

- 1 Proces pokazuje fragment systemu przekształcający dane, czasem może to być pojedyncza (drobna) funkcja – „oblicz podatek”, a czasem może odzwierciedlać np. prace całego działu, departamentu czy grupy osób.
- 2 Przepływ danych pokazuje przemieszczanie się danych w systemie. Przepływ może być jedno- lub dwukierunkowy. Przepływy powinny być etykietowane, tak aby wskazywać na dane przepływające.

## Znaczenie poszczególnych elementów

- 1 Proces pokazuje fragment systemu przekształcający dane, czasem może to być pojedyncza (drobna) funkcja – „oblicz podatek”, a czasem może odzwierciedlać np. prace całego działu, departamentu czy grupy osób.
- 2 Przepływ danych pokazuje przemieszczanie się danych w systemie. Przepływ może być jedno- lub dwukierunkowy. Przepływy powinny być etykietowane, tak aby wskazywać na dane przepływające.
- 3 Magazyn danych oznacza dane pozostające w spoczynku.

# Znaczenie poszczególnych elementów

- 1 Proces pokazuje fragment systemu przekształcający dane, czasem może to być pojedyncza (drobna) funkcja – „oblicz podatek”, a czasem może odzwierciedlać np. prace całego działu, departamentu czy grupy osób.
- 2 Przepływ danych pokazuje przemieszczanie się danych w systemie. Przepływ może być jedno- lub dwukierunkowy. Przepływy powinny być etykietowane, tak aby wskazywać na dane przepływające.
- 3 Magazyn danych oznacza dane pozostające w spoczynku.
- 4 Obiekty zewnętrzne (terminatory) są na zewnątrz modelowanego systemu, a przepływy pomiędzy nimi a systemem tworzą interfejs systemu ze światem zewnętrznym. O terminatorach mówimy że są poza dziedziną zmiany! Analityk nie może modelować terminatorów jako takich!

## Inne oznaczenia składników (Gane, Sarson)



proces, akcja



przepływ danych,  
połączenie



pamięć,  
magazyn danych



obiekt zewnętrzny,  
element terminalny

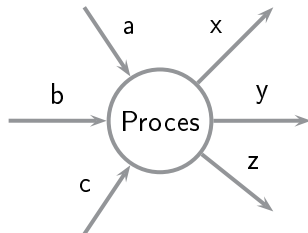
# Przepływy danych



Przepływ pokazuje przemieszczanie się informacji (danych) w systemie – przenoszenie się jednostek lub pakietów informacji z jednego fragmentu systemu do innego.

## Przepływy danych (cd.)

Przepływ nie odpowiada jednak na pytania proceduralne, tj. m.in. pytania ilościowe (liczba pakietów), oraz pytania o zależności pomiędzy samymi przepływami.

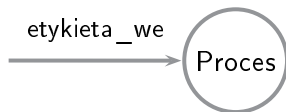


# Przepływy – uwagi

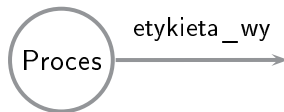
- Przepływy zasadniczo powinny być etykietowane.
- Przepływy w diagramach DFD ukazywane są bardzo ogólnie – nie są wskazywane zależności pomiędzy poszczególnymi przepływami, zarówno wejściowymi jak i wyjściowymi.
- Jeśli istnieje potrzeba dokładnego określenia przepływów, to należy modelować procedury wewnętrzne poszczególnych procesów.
- Mogą istnieć modyfikacje diagramów DFD, gdzie wprowadza się informacje odnośnie przepływów – nie jest to jednak zgodne z podejściem klasycznym, w gruncie rzeczy budzi wątpliwości.



# Kategorie przepływów danych



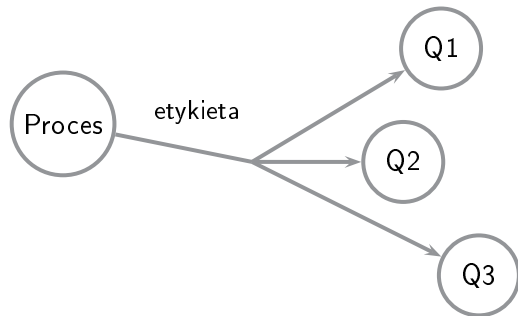
Przepływ wejściowy



Przepływ wyjściowy

# Przepływ rozbieżny

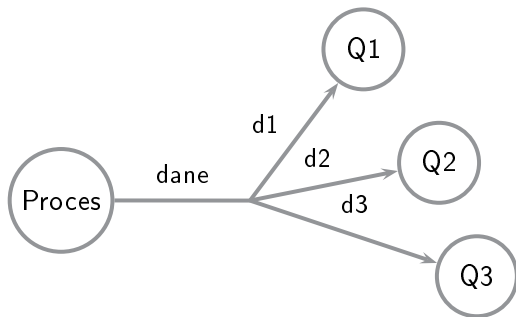
Przepływ może być rozbieżny, co oznacza rozchodzenie się pakietów danych do różnych części systemu. Rozchodzenie może odbywać się na różne sposoby, co pokazane zostało na dwóch przykładach.



Te same dane rozsyłane są do trzech części systemu.

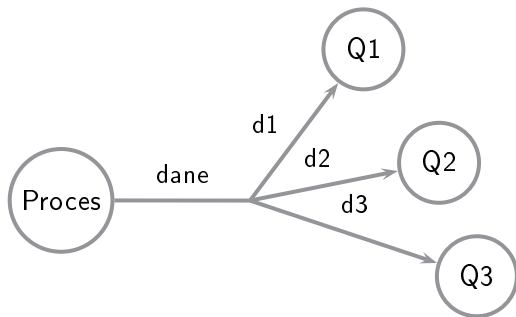
## Przepływ zbieżny

Podobne rozważania jak dla przepływów rozbieżnych także dla przepływów zbieżnych. Dane po rozdzieleniu na d1, d2 oraz d3 rozsyłane są do trzech różnych procesów, gdzie dane te są przetwarzane.



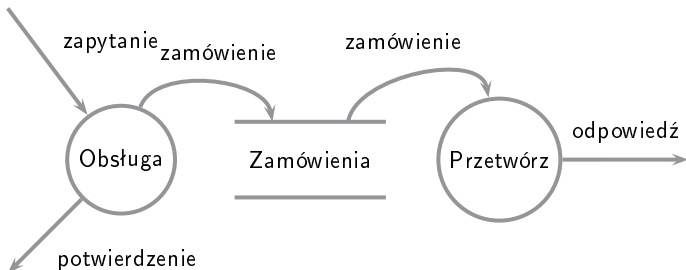
## Przeptyw zbieżny

Podobne rozważania jak dla przeptywów rozbieżnych także dla przeptywów zbieżnych. Dane po rozdzieleniu na d1, d2 oraz d3 rozsyłane są do trzech różnych procesów, gdzie dane te są przetwarzane.

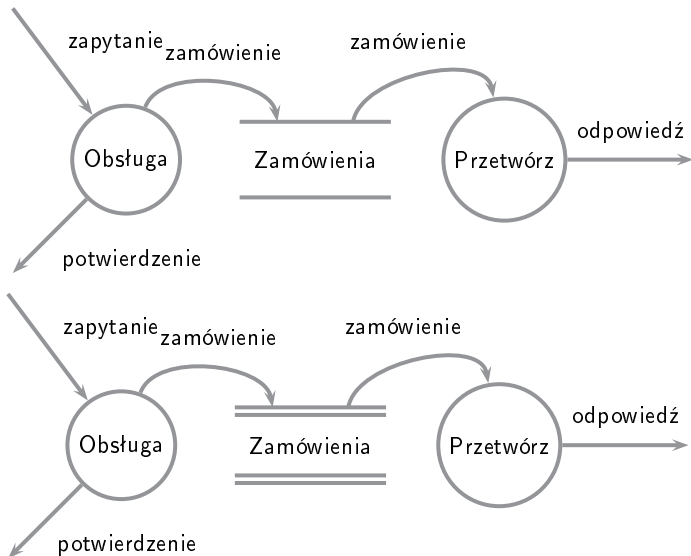


W obu jednak przypadkach nie są podawane informacje proceduralne, odnośnie sposobu przesyłania i pobierania danych.

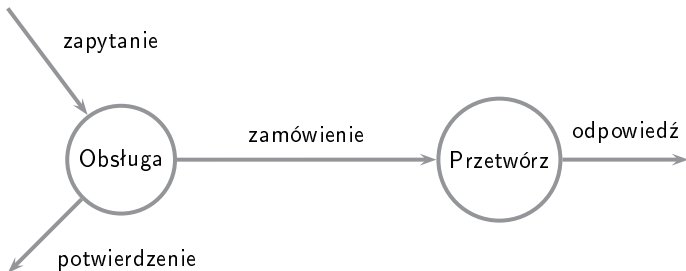
## Magazyny danych – wybrane aspekty



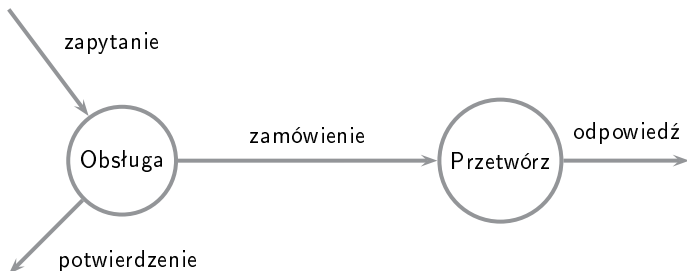
# Magazyny danych – wybrane aspekty



## Magazyny danych – wybrane aspekty (cd.)



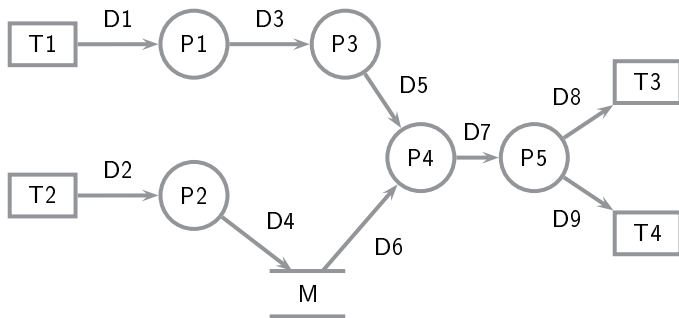
# Magazyny danych – wybrane aspekty (cd.)



- Powody modelowania magazynu mogą być różne i często uzależnione „lokalnie”/indywidualnie.
- W drugim przypadku mamy do czynienia z tzw. magazynem implementacyjnym.

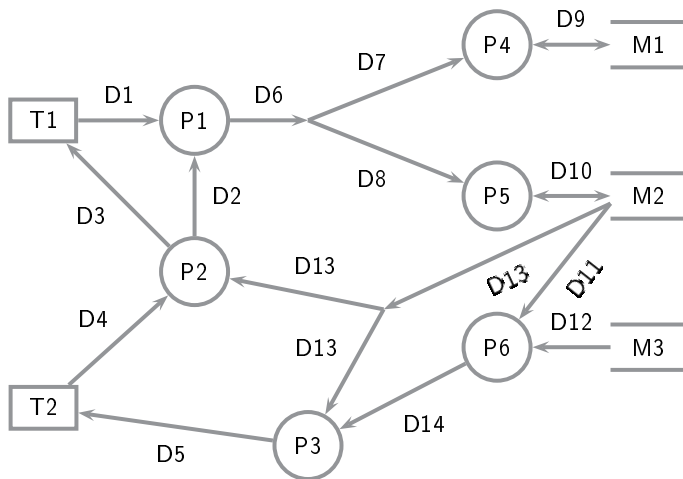


## Diagram DFD – przykłady (1)



## Przykład ogólny

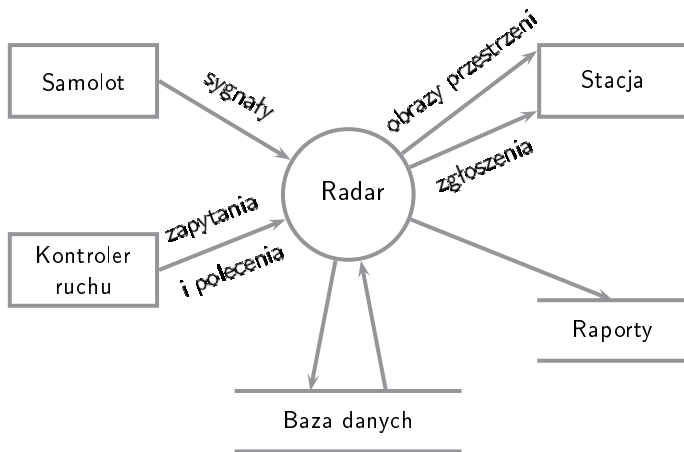
## Diagram DFD – przykłady (2)



Kolejny

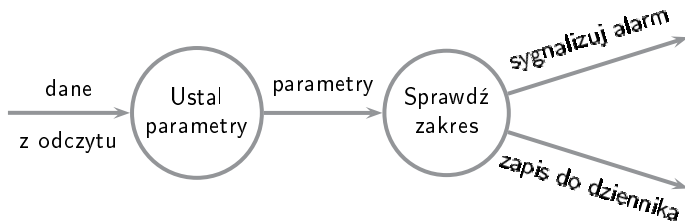
przykład ogólny

## Diagram DFD – przykłady (3)



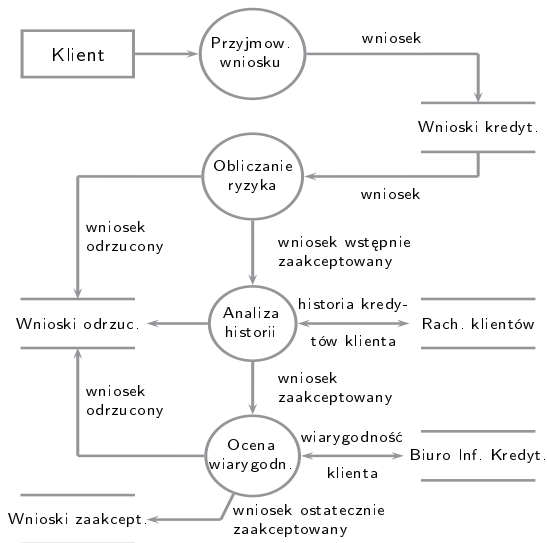
Stacja radarowa

## Diagram DFD – przykłady (4)

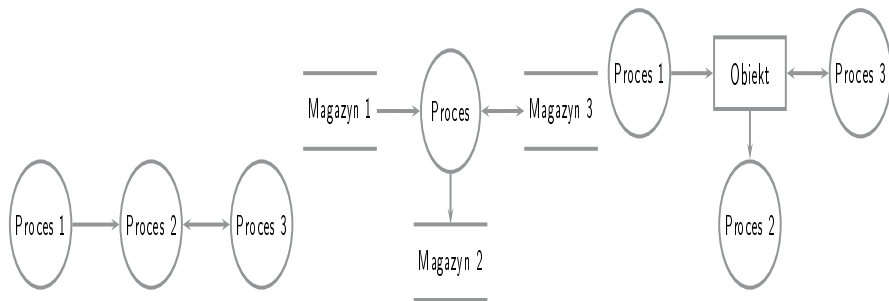


Prosty przykład  
(fragment większej całości)

## Diagram DFD – przykłady (5)



## Dopuszczalne przepływy danych

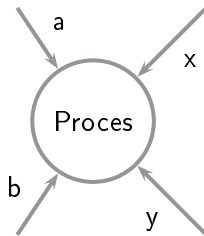


# Niedopuszczalne przepływy danych



Żle!

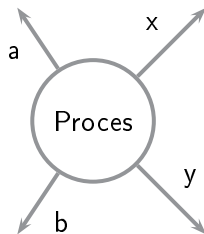
## Procesy specyficzne – studnia/generator



Studnia nieskończona



## Procesy specyficzne – studnia/generator (cd.)



Proces bez wejść

# Studnia/generator – uwagi

- Należy być ostrożnym w procesów określanych jako studnia nieskończona oraz proces bez wejść – z dużym prawdopodobieństwem są to procesy błędne.

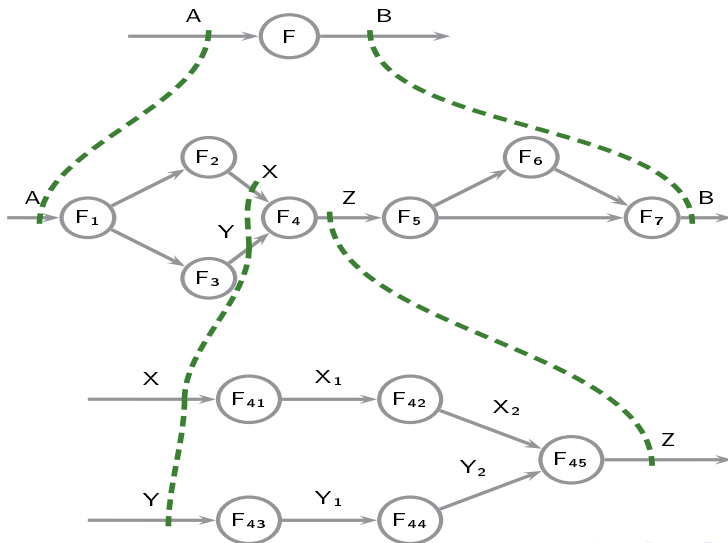
## Studnia/generator – uwagi

- Należy być ostrożnym w procesów określanych jako studnia nieskończona oraz proces bez wejść – z dużym prawdopodobieństwem są to procesy błędne.
- Czasem w przypadku procesu bez wejść możemy mieć do czynienia z generatorem liczb losowych – jest to jedno z nielicznych uzasadnień i trudno podać tutaj inne.

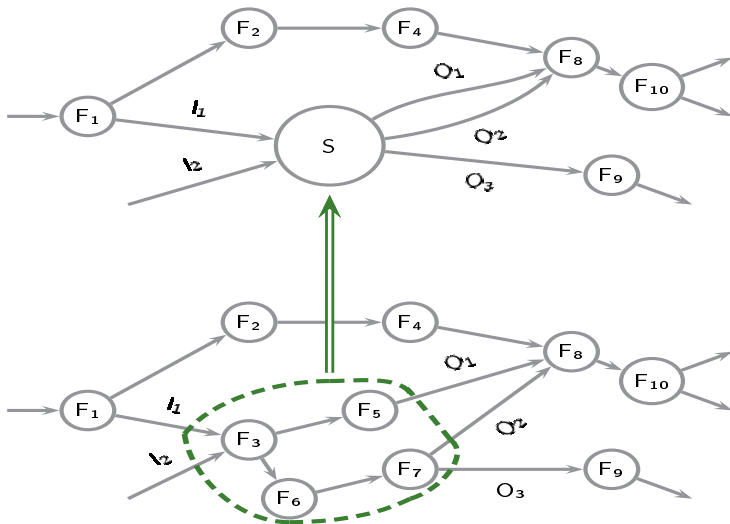
# Studnia/generator – uwagi

- Należy być ostrożnym w procesów określanych jako studnia nieskończona oraz proces bez wejść – z dużym prawdopodobieństwem są to procesy błędne.
- Czasem w przypadku procesu bez wejść możemy mieć do czynienia z generatorem liczb losowych – jest to jedno z nielicznych uzasadnień i trudno podać tutaj inne.
- W przypadku studni nieskończonej trudniej już o uzasadnienie – może tylko jakiś proces związany z rejestracją pewnej liczby ostatnich wydarzeń.

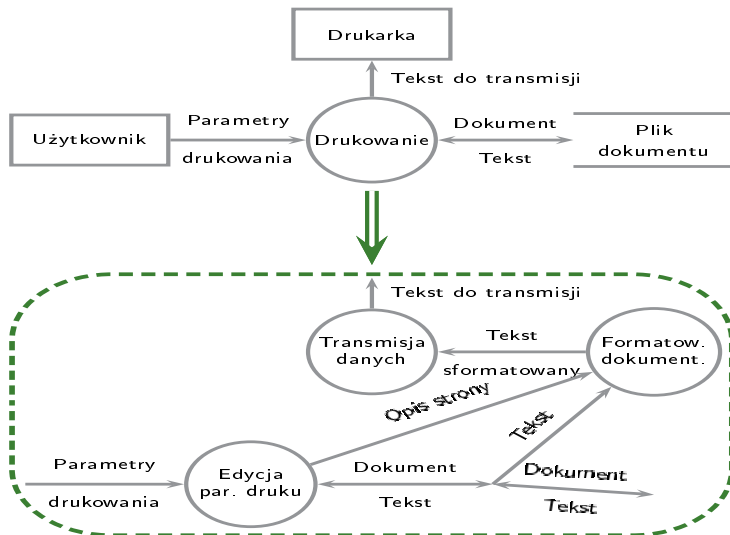
# Hierarchizacja w diagramach DFD



# Izolowanie w diagramach DFD

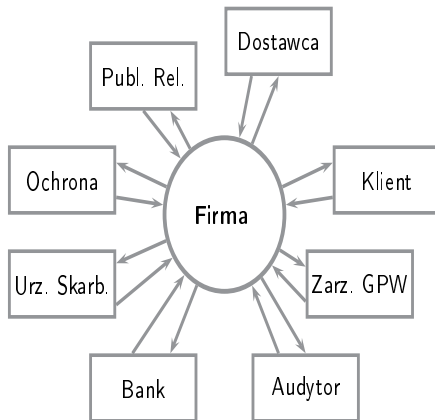


# Hierarchizacja/izolowanie – przykład



# Diagram kontekstowy

Diagram kontekstowy jest specyficznym rodzajem diagramu, o nieco niezrównoważonej strukturze – wprowadza się go zazwyczaj na początku modelowania systemu.





## Diagram kontekstowy (cd.)

Diagram kontekstowy zazwyczaj zawiera:

- jeden (lub niewiele więcej) procesów,
- dużą liczbę terminatorów (obiektów zewnętrznych),
- raczej nie zawiera magazynów danych, oraz
- posiada dość dużą liczbę zaznaczonych przepływów.

## Rozwijanie diagramu kontekstowego

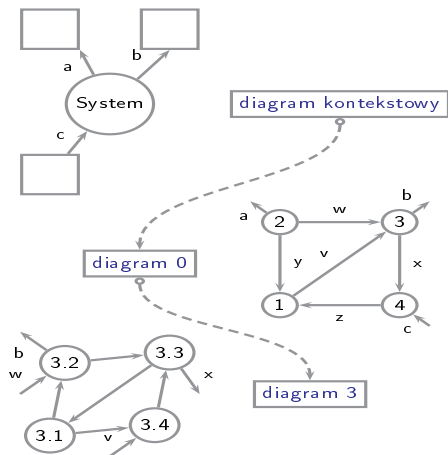
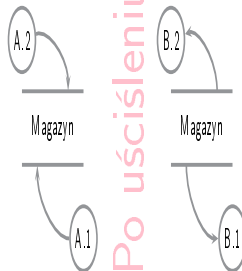


Diagram kontekstowy może być rozwijany. Diagram bezpośrednio po kontekstowym jest oznaczany jako 0. Następnie numeracja procesów służy do numeracji diagramów. Należy zwracać uwagę na to, aby diagramy były zrównoważone, aby zachodziła zgodność pomiędzy liczbą przepływów wejściowych i wyjściowych procesu uściślanego i procesów nowych powstałych na jego bazie.

## Magazyny danych a uściślanie procesów



Przed uściśleniem



Po uściśleniu

# Wskazówki konstrukcyjne dla diagramów DFD

## Jak konstruować diagramy DFD?

- zidentyfikuj wszystkie elementy zewnętrzne (terminatory), które mogą oddziaływać z systemem;
- narysuj (pojedynczy) proces pokazujący oddziaływanie z wybranym elementem;
- uściśl diagram poprzez dekompozycję pojedynczego procesu na kilka mniejszych;
- powtarzaj powyższe kroki dla pozostałych elementów zewnętrznych;
- podobnie jak dla terminatorów postępuj także dla magazynów danych (pewna liczba magazynów danych może się ujawnić w trakcie dokładnej analizy procesów);

## Wskazówki konstrukcyjne dla diagramów DFD (cd.)

- magazyny danych powinny być pokazywane – od momentu pojawienia się – na każdym diagramie niższego rzędu;
- stosuj nazwy znaczące dla procesów, terminatorów, przepływów oraz magazynów, tj. stosuj frazy mające samodzielne znaczenie semantyczne, ponadto dla procesów stosuj także numerowanie.

## Wskazówki konstrukcyjne dla diagramów DFD (cd.)

Nie ma ogólnego algorytmu łączenia elementów diagramu, ale istnieją pewne reguły:

- analizuj źródła/obiekty zewnętrzne i zastanów się, dla których procesów mogą być one wejściami;
- narysuj proces i zapytaj o wyjście jakie ono produkuje, a następnie uwzględnij to w połączeniach;
- powtarzaj powyższe kroki dla wszystkich elementów;
- nie rysuj pojedynczej warstwy z więcej niż ok. 5-7 procesami (niektórzy podają 3-9).

# Wskazówki konstrukcyjne dla diagramów DFD (cd.)

Przy dekomponowaniu można stosować następujące reguły:

- każdy poziom powinien zawierać nie więcej niż połowę ogólnej liczby procesów;
- pod rozważę: prosty system nie więcej niż 2–3 poziomów, średni 3–5 poziomów, a duży do 8 poziomów;
- nie wszystkie procesy muszą być rozwijane do maksymalnej liczby poziomów.

# Weryfikowanie diagramów DFD

Zagadnienia do uwzględnienia w trakcie procesu weryfikacji diagramów:

- czy wszystkie procesy rzeczywiście pokazują transformacje danych?
- czy wszystkie dane zostały pokazane w systemie (jednak tylko dane z którymi oddziałuje proces, bez pokazywania danych wewnętrznych)?



## Weryfikowanie diagramów DFD (cd.)

- czy każdy przepływ, analizowany pod kątem procesów i magazynów, ma swoje źródło i przeznaczenie na rozważanym diagramie – tzw. **bilans poziomy** ?
- czy na kolejnych diagramach dla procesu uściślonego zgadzają się wejścia i wyjścia (dane przed/po operacji) – tzw. **bilans pionowy** ?
- czy wejście do procesu i sam proces determinują wyjście?
- czy diagram może być przerysowany prościej?

# Weryfikowanie diagramów DFD (cd.)

- czy nie ma sytuacji takiej, że wejście procesu jest jego wyjściem?
- czy istotnie następuje przepływ danych?
- czy istnieją nieetykietowane przyptywy (należy ich unikać)?
- czy istnieją procesy typu studnia nieskończona oraz proces bez wejść?
- czy wszystkie przepływy „od” magazynu ilustrują czytanie lub dostęp do informacji w magazynie?
- czy wszystkie przepływy „do” magazynu ilustrują zapis, modyfikację lub usunięcie informacji w magazynie?
- czy istnieją magazyny typu tylko-do-zapisu oraz tylko-do-odczytu (należy je dobrze uzasadnić)?

# Diagramy DFD – podsumowanie

Podsumowanie odnośnie diagramów DFD:

- diagramy DFD są stosunkowo prostym, lecz jednocześnie bardzo mocnym narzędziem modelowania systemów (także np. biznesowych);
- diagramy DFD pokazują strukturę logiczną systemu;
- analiza strukturalna nie kończy się na diagramach DFD, lecz od nich się rozpoczyna – same diagramy nie są tu jednak wystarczające;

# Diagramy DFD – podsumowanie (cd.)

- zalety diagramów:
  - możliwość przeprowadzenia zarówno oceny poprawności składniowej (struktura diagramu, obecność poszczególnych elementów, charakterystyka przepływów, itd.), jak i poprawności semantycznej (szczegółowość specyfikacji, przydatność i realność poszczególnych elementów, celowość dekompozycji, itd.),
  - (wielokrotne) stosowanie strategii zstępującej (z góry na dół),
  - możliwość połączenia strategii zstępującej z wstępującą (z dołu do góry),

# Diagramy DFD – podsumowanie (cd.)

- zalety diagramów (cd.):
  - modelowanie nowych i odmiennych dziedzin zastosowań,
  - modelowanie procesów złożonych, skomplikowanych i nietypowych,
  - uwzględnianie specyfikacji procesów rutynowych i często spotykanych (biblioteki),
  - wskazanie interfejsu systemu wraz z procedurami dialogowymi (terminale a system),
  - czytelność diagramów ułatwia kontakty i dyskusje z klientem/użytkownikiem;

## Diagramy DFD – podsumowanie (cd.)

- wady diagramów:
  - pracowitość opracowania,
  - czasem pewna tendencja do nie korzystania z gotowych bibliotek,
  - mniejsza przydatność w przypadku pojedynczej i zintegrowanej bazy danych.