

DFD – Diagram przepływu danych (Data Flow Diagram)



dr Tomasz Ordysiński

DFD - definicja

- Jedną z metod wykorzystywanych na etapie analizy i projektowania służących do modelowania funkcji systemu jest **Diagram Przepływu Danych** (ang. Data Flow Diagram - DFD).
- Przedstawia on, w jaki sposób dane przepływają w systemie oraz opisuje procesy przetwarzające dane.
- Tworzenie diagramu DFD opiera się na następujących kategoriach pojęciowych:
 - proces,
 - przepływ danych,
 - magazyn danych, terminator.
- Każda kategoria ma odpowiedni symbol graficzny.

Kategorie: proces, przepływ danych (1)





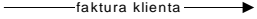
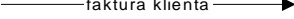
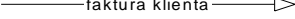

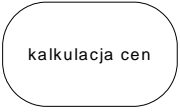
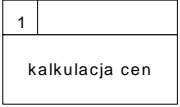
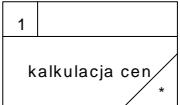
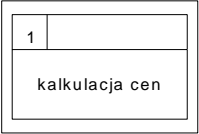
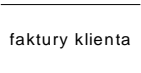
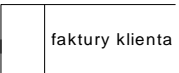
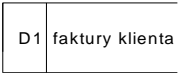

- ❑ **Proces** (ang. process) oznacza transformację danych wejściowych w wynikowe i odpowiada tym składnikom systemu, które przetwarzają dane. Procesy otrzymują i przesyłają dane za pośrednictwem przepływów danych. Kojarzą się one z procedurą, której specyfikacja jest przedstawiona przy użyciu innych technik strukturalnych. Nazwa procesu powinna opisywać czynność wykonywaną na określonym obiekcie, jak np.: *aktualizacja konta klienta, rejestracja danych klienta, wyliczenie stawki amortyzacji sprzętu*.
- ❑ **Przepływ danych** (ang. data flow) - opisuje zbiór danych przepływający pomiędzy dwoma obiektami w systemie. Przedstawia się go za pomocą linii ze strzałkami określającymi kierunek przesyłania informacji. Linie są skierowane najczęściej jednostronnie. Jeśli przekazywana informacja jest zwrotna używa się kolejnych linii lub strzałek dwukierunkowych. Nazwa przepływu to rzeczownik w liczbie pojedynczej. Przykładowe przepływy danych: *kwestionariusz osobowy, umowa, faktura dla klienta*.

Kategorie: magazyn, terminator (2)

- **Magazyn danych** (ang. data store) - inaczej składnica danych służy do przechowywania danych w postaci jednorodnych kolekcji. Zaistnienie magazynu danych w diagramie ma sens, gdy przechowywane dane służą do realizacji, co najmniej dwóch procesów. Charakter magazynu danych zależy od stopnia szczegółowości diagramu. Nazwa magazynu danych to rzeczownik w liczbie mnogiej, jak np.: *pracownicy, towary, wydawnictwa, faktury klienta*.
- **Terminator** (ang. terminator) - obiekt zewnętrzny w stosunku do systemu reprezentujący źródła lub miejsca przeznaczenia informacji. Terminatorami są obiekty, z którymi system komunikuje się. Nazwa terminatora, to rzeczownik liczby pojedynczej. Przykładowe terminatory to: *Szef* (odbierający raporty o niezapłaconych fakturach klienta), *Klient* (składający zamówienie na towary), *Moduł finansowo-księgowy systemu informatycznego* (korzystający z zestawień sprzedaży projektowanego systemu ewidencji sprzedaży, a nie będący przedmiotem projektu).

Notacja graficzna

Wielu autorów zdefiniowało różne notacje graficzne. Na potrzeby zajęć posłużymy się notacją Yourdon'a.

Nazwa	Yourdon-DeMarco	Gane-Sarson	SSADM
Terminator			 terminator  terminator powtórzony
Przepływ danych			
Proces			 proces danych  proces elementarny  proces wielokrotny
Magazyn danych			 magazyn danych  magazyn danych powtórzony

Reguły poprawności diagramów DFD

(1)

- ❑ diagramy DFD uporządkowane są hierarchicznie: diagram kontekstowy, diagram zerowy (systemowy), diagramy szczegółowe;
- ❑ diagram DFD nie może być większy niż format A4 oraz powinien zawierać od trzech do siedmiu procesów. Należy unikać nadmiernej złożoności DFD;
- ❑ magazyn danych (składnica danych) musi być wykorzystany przez co najmniej dwa procesy;
- ❑ nazwy kategorii określone w hierarchii diagramów są niepowtarzalne;
- ❑ wszystkie kategorie z poziomu podrzędnego muszą być pokazane na poziomie nadrzędnym;
- ❑ nazwy przepływów do i z magazynów danych (składnic danych) mogą być nienazwane, jeśli zapisywana lub pobierana jest pełna informacja;

Reguły poprawności diagramów DFD

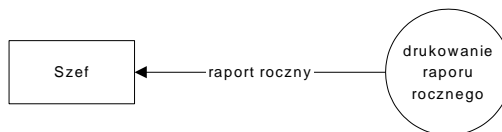
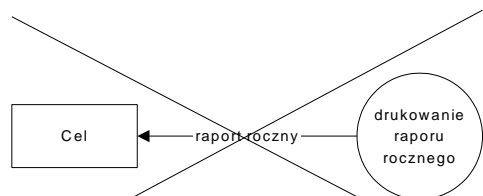
(2)

- ❑ strzałka do magazynu danych (składnicy danych) oznacza, że dokonuje się konkretna zmiana (wprowadzenie, aktualizacja, skreślenie), natomiast strzałka z magazynu danych (składnicy danych) oznacza, że dane są czytane;
- ❑ przepływ pomiędzy magazynem danych (składnicą danych) i terminatorem jest niedopuszczalny;
- ❑ diagram DFD zawiera zarówno ręczne, jak i automatyczne procesy;
- ❑ każdy proces musi mieć co najmniej jeden przepływ wejściowy i jeden wyjściowy;
- ❑ należy sprawdzić, czy DFD jest wewnętrznie niesprzeczny z innymi spokrewnionymi DFD.

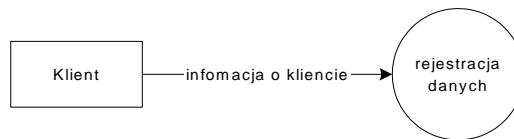
Błędne konstrukcje DFD

- ❑ Tworząc diagramy przepływu danych często zapomina się podstawowych zasadach ich konstrukcji.
- ❑ Typowe błędy wiążą się z:
 - użyciem niewłaściwych nazw komponentów,
 - niewłaściwym łączeniem komponentów
 - użyciem procesów duchów i procesów studni

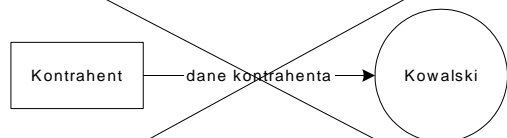
Niewłaściwe nazwy komponentów



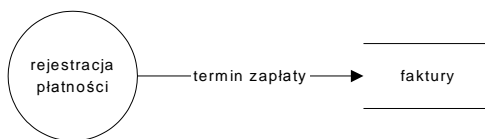
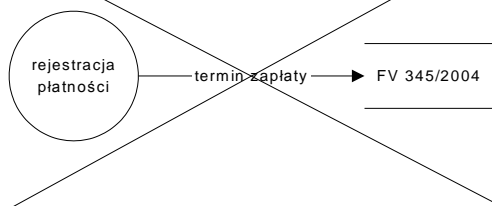
Błędna nazwa terminatora



Błędna nazwa przepływu danych

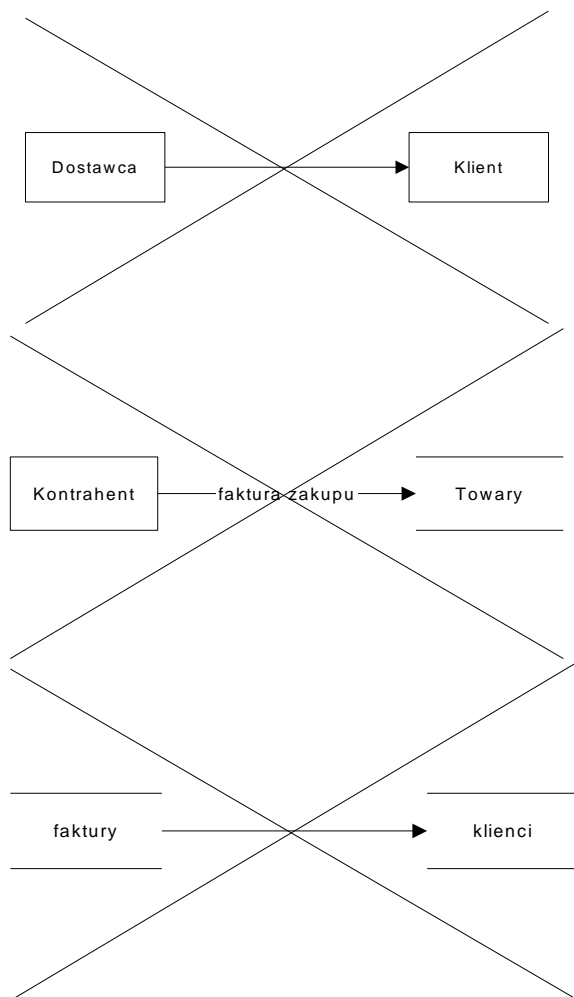


Błędna nazwa procesu



Błędna nazwa magazynu danych

Niewłaściwe łączenie komponentów

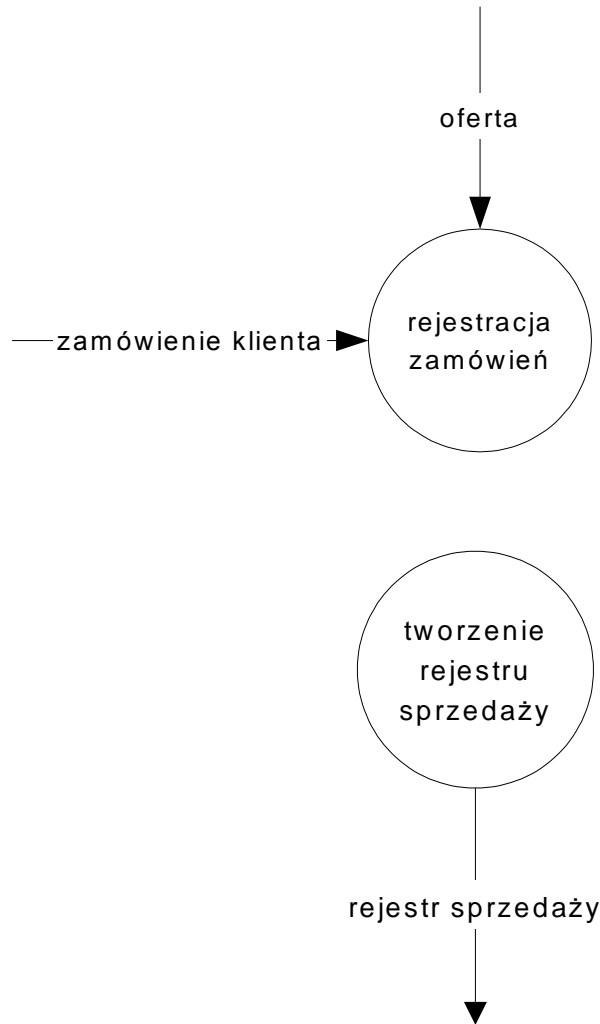


Niedopuszczalne połączenie dwóch terminatorów

Niedopuszczalne połączenie terminatora z magazynem danych

Niedopuszczalne połączenie dwóch magazynów danych

Użycie procesów duchów i procesów studni



Proces pochłaniający dane tzw. „proces studnia”

Proces „duch” – nie wiadomo skąd pobierane są dane

Dekompozycja funkcji

- **Diagram kontekstowy** (inaczej diagram DFD poziomu zerowego) jest specjalnym graficznym schematem przepływu danych, który definiuje zakres i granice systemu. System przedstawiony jest na diagramie jako pojedynczy proces powiązany bezpośrednio przepływami danych z terminatorami. Całość ma na celu przedstawienie powiązań systemu ze środowiskiem zewnętrznym. Nie występują magazyny danych.
- Następnie tworzony jest **diagram ogólny (poziomu „0”)** Zawiera:
 - wszystkie terminatory i przepływy, które były na wyższym poziomie
 - magazyny
 - kilka procesów (z numerami 1, 2, 3, ..., n)

Pokazuje jak pojedynczy proces z diagramu kontekstowego dzieli się na procesy bardziej szczegółowe – dekompozycja procesu

- Diagramy niższych poziomów to **diagramy szczegółowe**. Zbiór diagramów DFD dla systemu wraz z opisem elementów występujących na diagramie w słowniku danych stanowi model funkcji systemu. Jest to graficzna mapa procesów ukazująca przepływ danych między procesami w systemie oraz między światem zewnętrznym a systemem.

Diagram kontekstowy - przykład

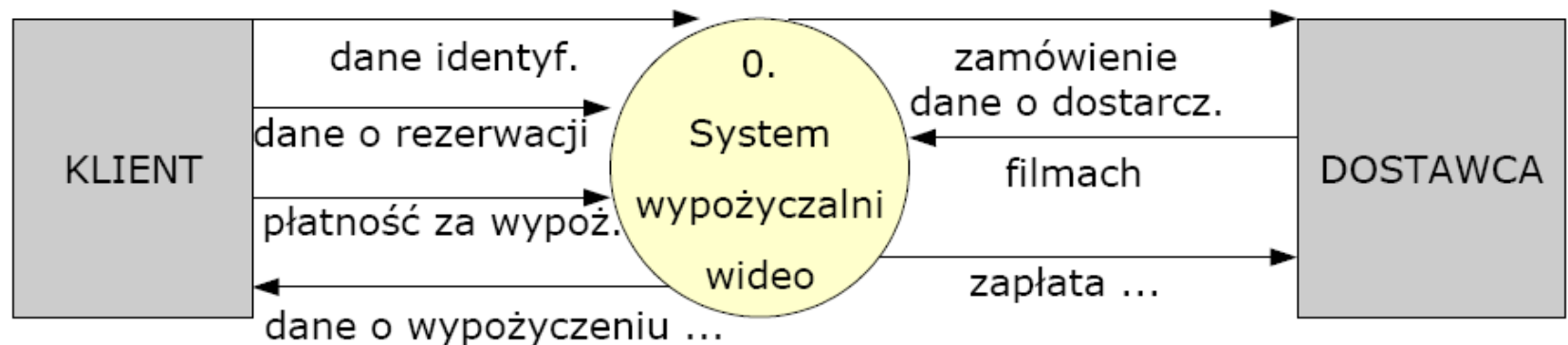


Diagram ogólny – przykład

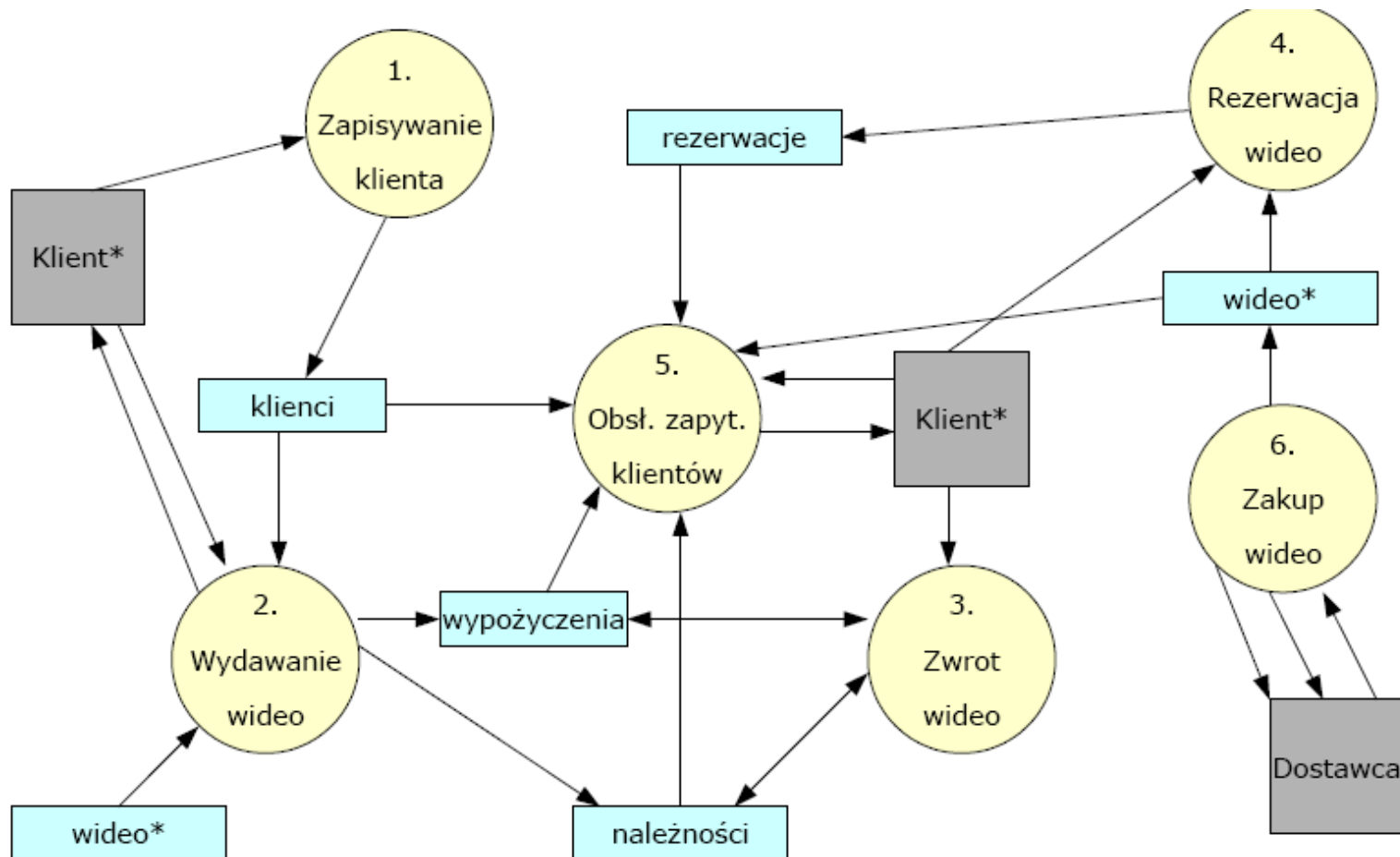
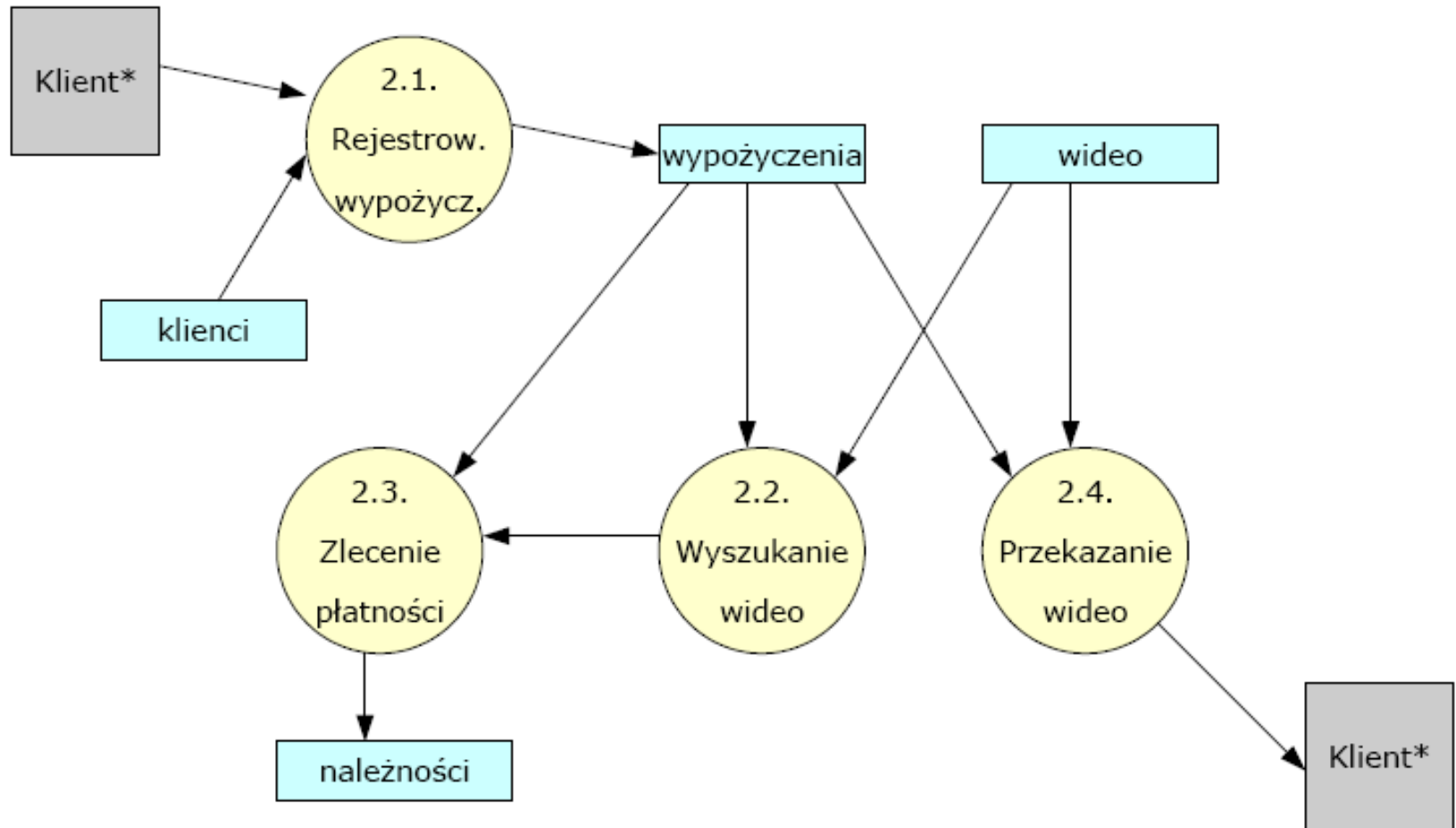


Diagram szczegółowy - przykład



Problemy dekompozycji

- ❑ Na diagramie nie widać szczegółów działania organizacji – procesy mają bardzo ogólne nazwy
- ❑ Rozwiązanie – dalsza dekompozycja
- ❑ Kiedy kończymy dekompozycję?
 - Nie ma sztywnych reguł odnośnie liczby poziomów
 - Nie dekomponujemy procesu, który wystarczająco precyzyjnie opisuje na czym on polega
 - ❑ jeśli specyfikacja procesu w pseudokodzie zajmie ok. jednej strony
 - ❑ jeśli schemat blokowy charakteryzujący proces zajmie ok. jednej strony

Jak tworzyć modele?

- Problem: Jak utworzyć model wielopoziomowy? Od czego zacząć?

- Rozwiązanie: 3 podejścia:
 - z góry na dół (*top-down*)
 - z dołu do góry (*bottom-up*)
 - hybrydowe

Podójście z góry na dół

- Diagram kontekstowy
- Definiuje się procesy bardzo ogólne
- Droga kolejnych uszczegółowień (dekompozycji procesów) dochodzi się do rozwiązania końcowego
- Wymaga doświadczenia – w dużym systemie trudne może być wyodrębnienie procesów na wysokim poziomie

Podójście z dołu do góry

- Tworzy się listę niezależnych procesów, które nie będą dalej dekomponowane
- Grupuje się procesy do postaci procesów wyższego rzędu
- ... aż na samą górę – diagram kontekstowy
- Problem – duża liczba procesów do przeanalizowania

Podejście hybrydowe

- Diagram kontekstowy
- Diagram ogólny
- Lista procesów, które nie będą dekomponowane
- Przypisanie procesów najniższego rzędu do odpowiedniej gałęzi z diagramu ogólnego
- Grupowanie procesów w ramach jednej gałęzi aż do utworzenia pojedynczego procesu (z diagr. ogólnego)
- ... dla wszystkich gałęzi

Narzędzia

- Power Designer 6.1
 - Process Analyst

Ćwiczenia

- ❑ **Przykład 1: *System ewidencji kart egzaminacyjnych***
założenia: Wykładowca po przeprowadzeniu egzaminu z danego przedmiotu powinien umieszczać uzyskane wyniki na tzw. protokóle egzaminacyjnym. Zakładamy, że system ewidencjonuje dane studentów, którzy podeszli do egzaminu i uzyskali wynik. System powinien umożliwiać generowanie protokołu egzaminacyjnego, który min. będzie przekazany do wiadomości studentom.
- ❑ **Przykład 2: *System ewidencji imprez***
założenia: Klub młodzieżowy organizuje różne imprezy, w czasie których występują różni wykonawcy. System powinien generować terminarz imprez informując o czasie, miejscu, celu imprezy oraz biorących w niej udział wykonawcach.