

**Diagramy czynności  
tworzenie modelu przypadków  
użycia  
Wykład 2**

Zofia Kruczkiewicz

# Diagramy czynności- tworzenie modelu przypadków

- 1. Diagramy czynności UML**
- 2. Przykład diagramów czynności UML – modelowanie przepływu czynności**  
(wg Booch G., Rumbaugh J., Jacobson I., UML przewodnik użytkownika)
- 3. Przykład diagramów czynności UML – modelowanie operacji**  
(wg Booch G., Rumbaugh J., Jacobson I., UML przewodnik użytkownika)

# Diagramy czynności - tworzenie modelu przypadków

## 1. Diagramy czynności UML

# Diagramy UML 2 – część druga

Na podstawie

**UML 2.0 Tutorial**

[http://sparxsystems.com.au/resources/uml2\\_tutorial/](http://sparxsystems.com.au/resources/uml2_tutorial/)

## Dwa rodzaje diagramów UML 2

### Diagramy UML modelowania strukturalnego

- Diagramy pakietów
- Diagramy klas
- Diagramy obiektów
- Diagramy mieszane
- Diagramy komponentów
- Diagramy wdrożenia

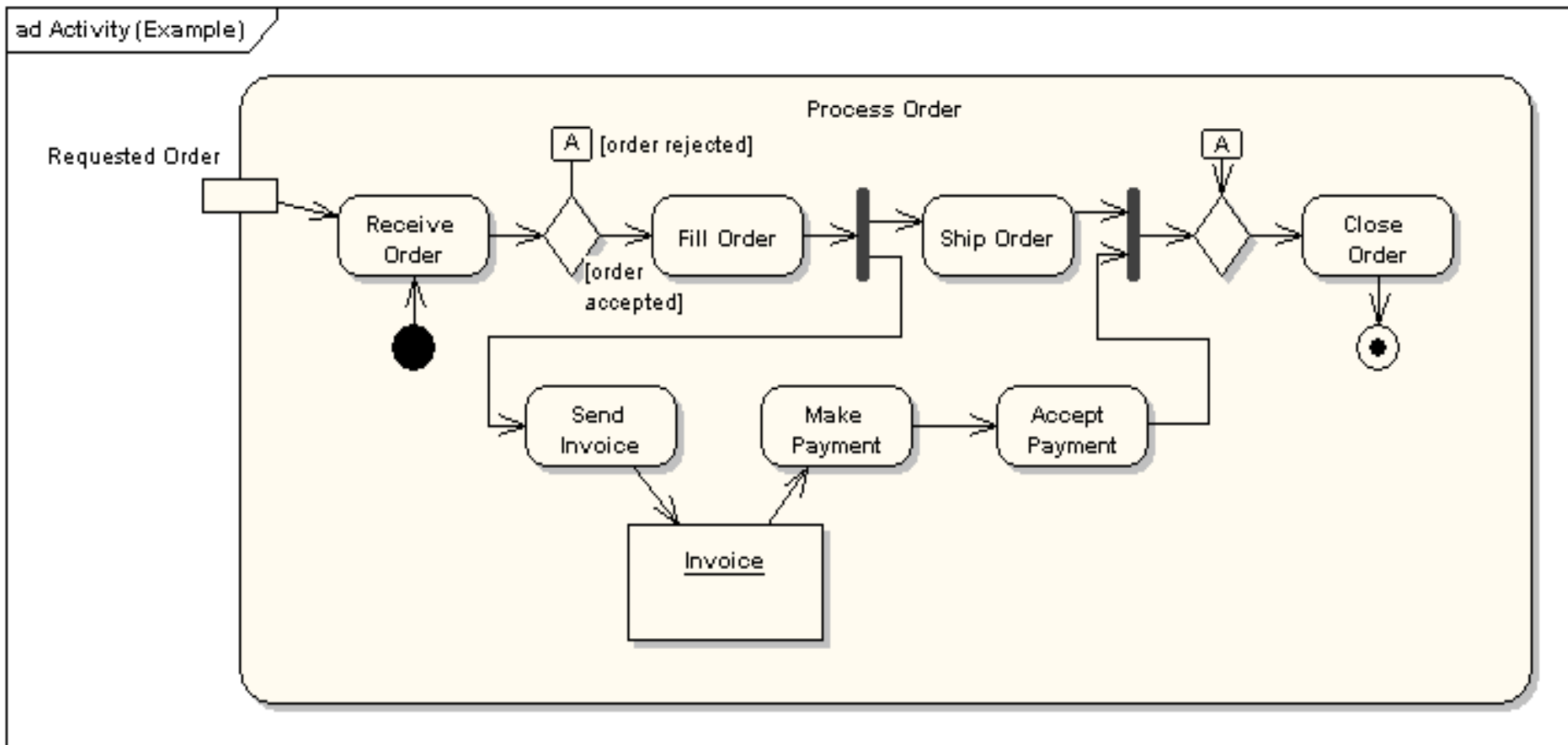
### Diagramy UML modelowania zachowania

- *Diagramy przypadków użycia*
- *Diagramy czynności*
- Diagramy stanów
- Diagramy komunikacji
- Diagramy sekwencji
- Diagramy czasu
- Diagramy interakcji

# Diagramy czynności

Diagram czynności opisuje interakcje między obiektami:

- **jak** pobierane są operacje,
- **co** operacje wykonują (zmiana stanu obiektu),
- **kiedy** operacje są wykonywane (sekwencje czynności lub akcji)
- **gdzie** są wykonywane.



ad Activity

Activity

**Czynność** - zawiera specyfikację sparametryzowanych zachowań:

- akcje
- przepływ sterowania

ad Action

Perform  
Action

**Akcja** – elementarny krok czynności

### ad Conditions

«localPreCondition»  
{A drink is selected that the vending machine contains}



«localPostCondition»  
{The vending machine dispensed the drink selected}

### ad Activity Edge



## Ograniczenia akcji:

Warunki przed akcją i po akcji

np.

**Warunek przed:** Wybór napoju w automacie, jeśli istnieje

**Stan:** akcja wydania napoju (*Dispense drink*)

**Warunek po:** Maszyna wydała wybrany napój

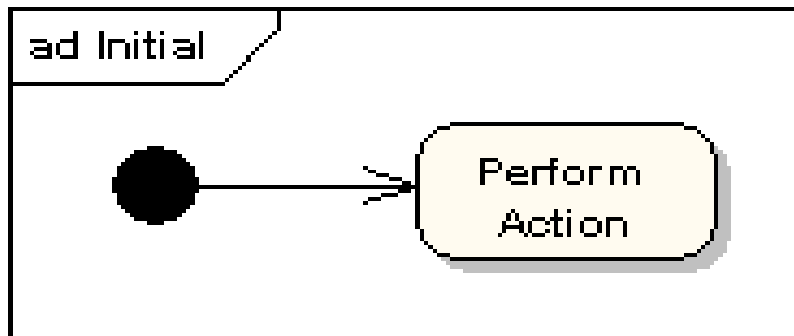
## Przeływ sterowania:

Przejście z jednej akcji aktywności do akcji drugiej aktywności

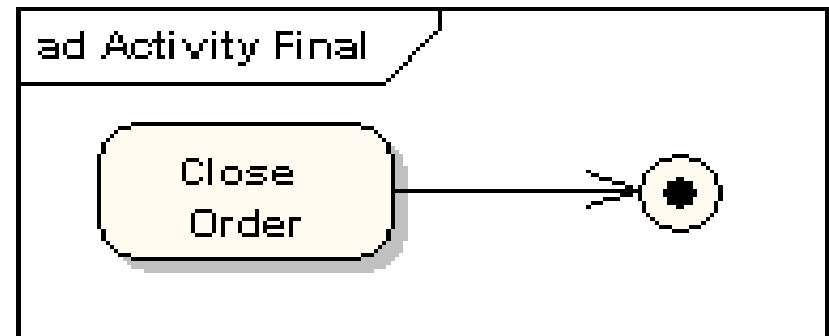
np.

wysłanie opłaty (*Send Payment*) i akceptacja opłaty (*Accept Payment*)

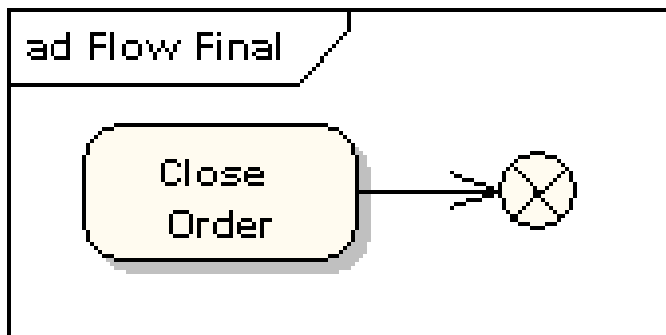




**Wierzchołek początkowy**  
(*Initial*) wielu przebiegów sterowania

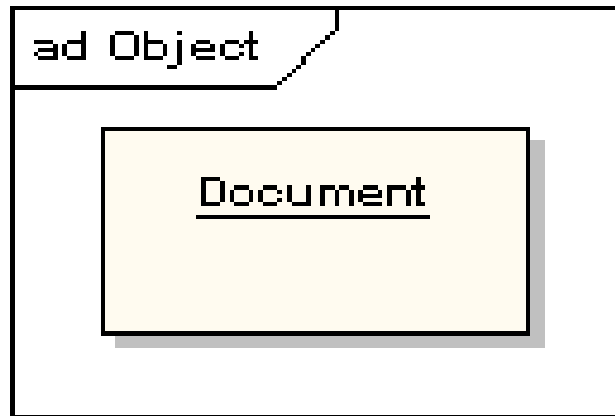


**Wierzchołek końcowy** (*Final*)  
wielu przebiegów sterowania  
związanych z jedną czynnością

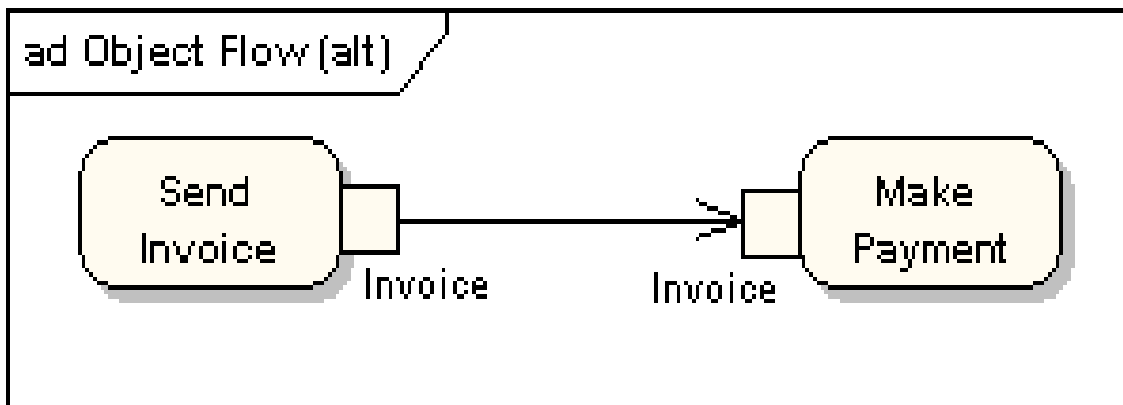
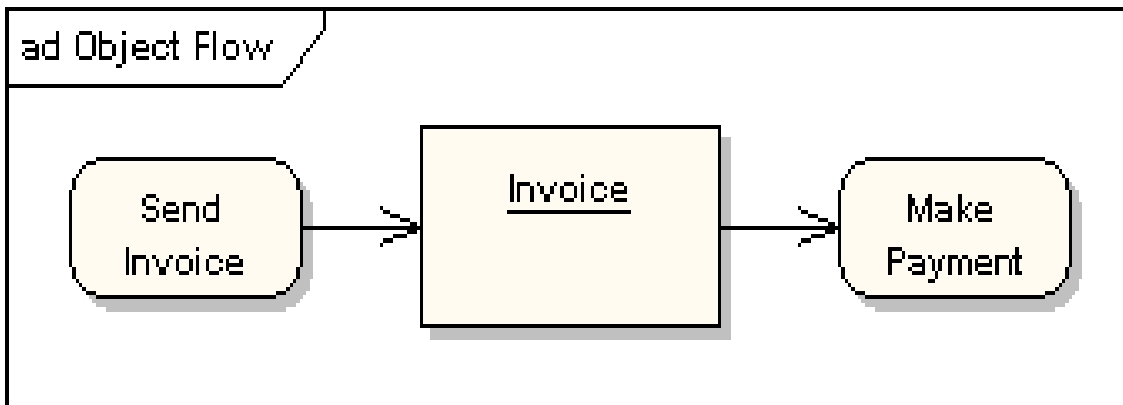


**Wierzchołek końca sterowania**  
(*Flow Final*) koniec pojedynczego przebiegu sterowania

## Obiekt



## Magazyn danych



## Przepływ obiektów

Wysłanie (*Send Invoice*) obiektu Faktura (*Invoice*) w celu dokonania opłaty (*Make Payment*)

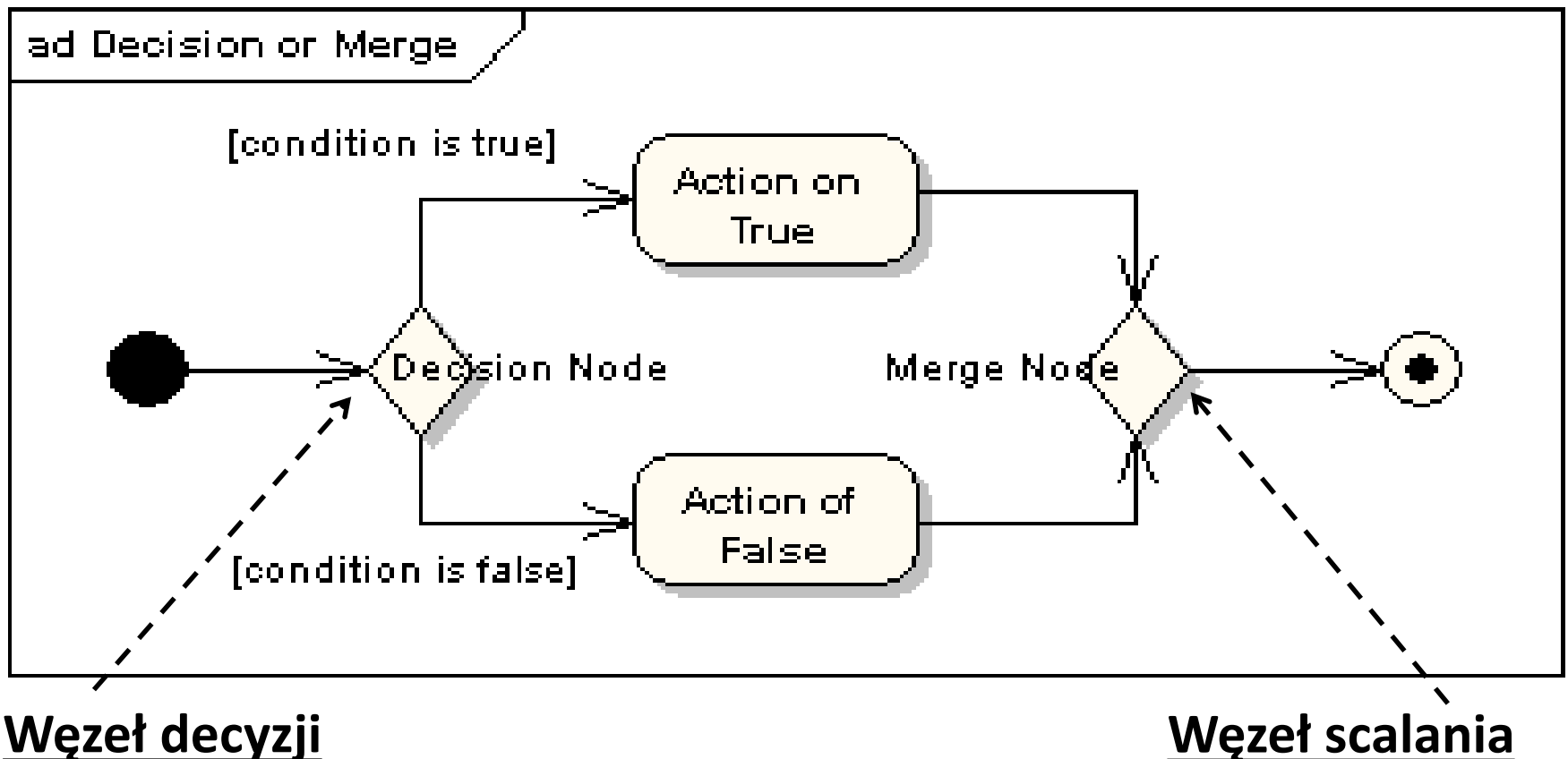


## Przepływ obiektów

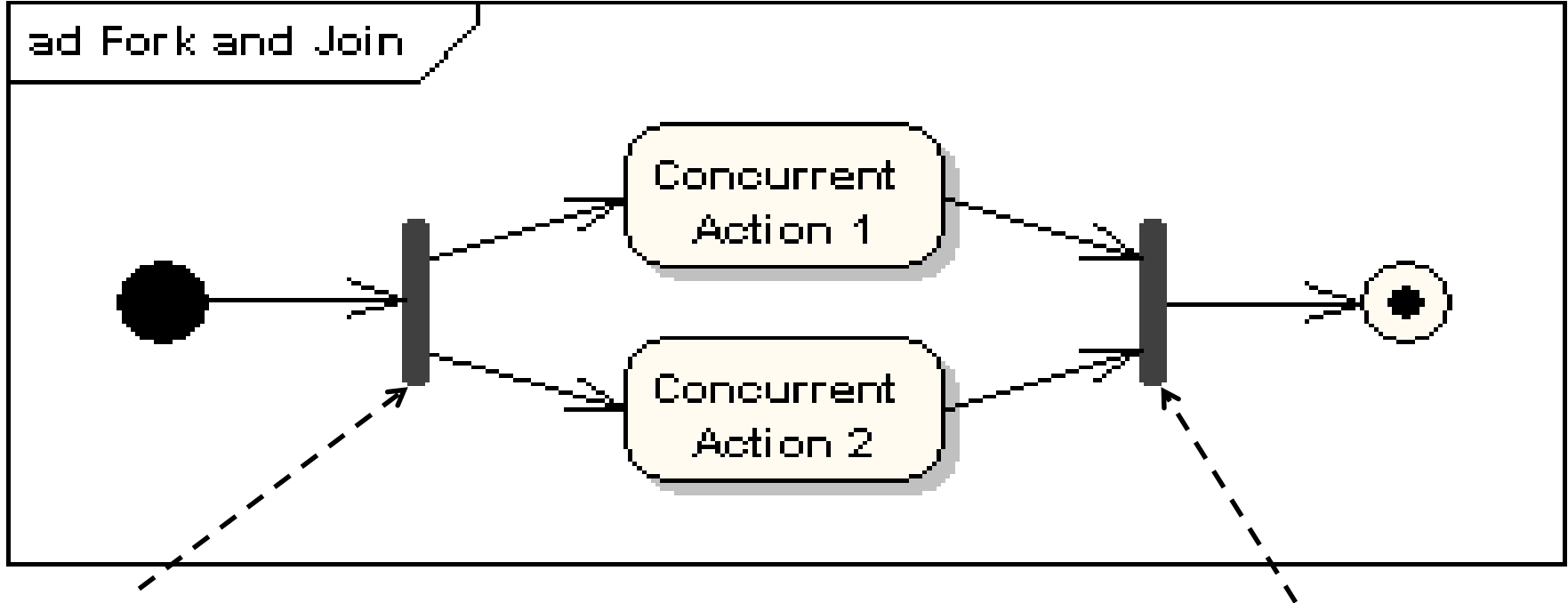
Równoważny diagram

## Węzły decyzji i scalania

Wybór przepływu sterowania w węźle decyzji (*Decision Node*) po zbadaniu warunku (*condition*) i scalenie z przepływem sterowania znajdującym się za węzłem scalania (*Merge Node*)



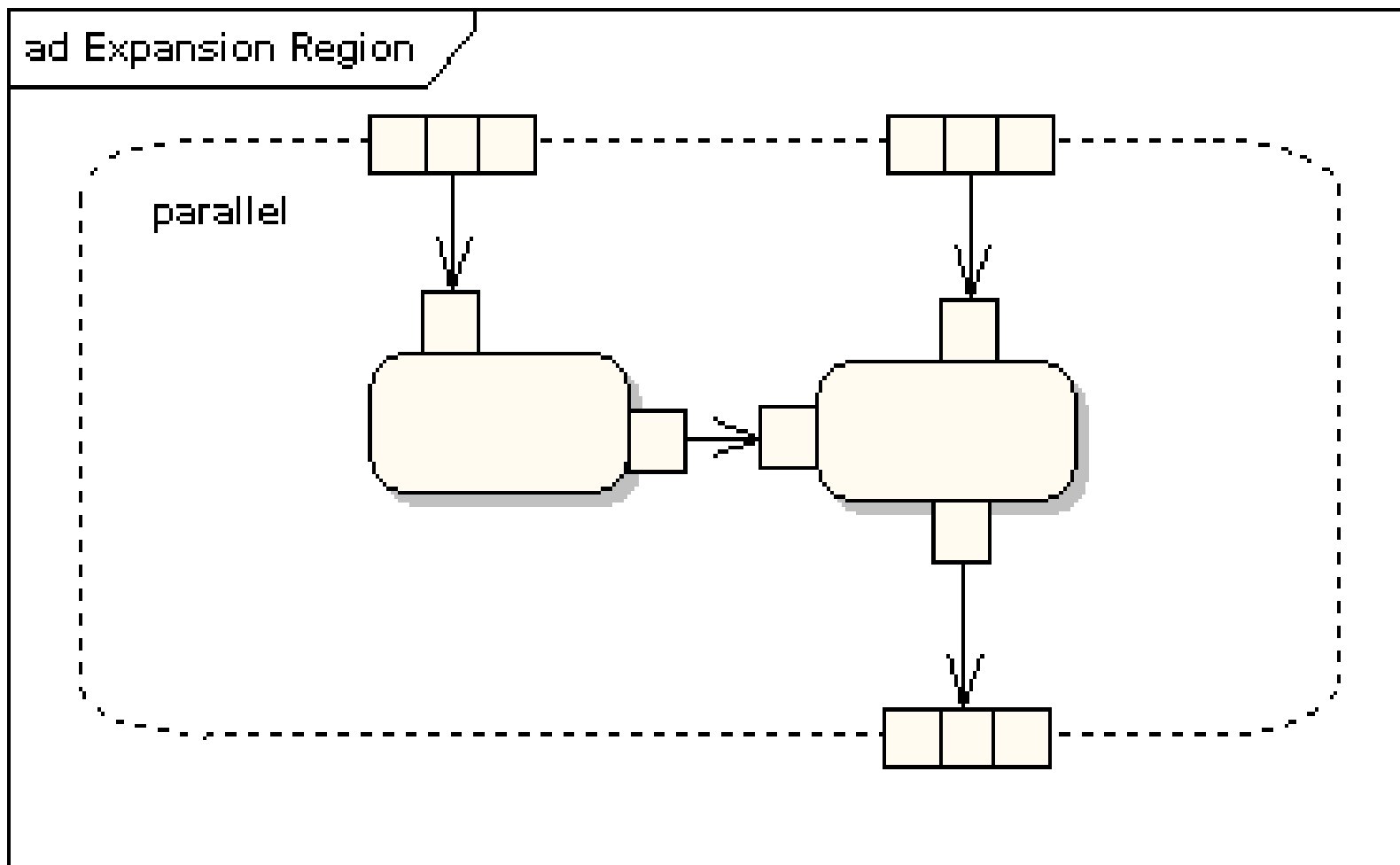
## Węzły rozdzielania i łączenia



**Węzeł rozdzielania** (*Fork* – pionowa lub pozioma linia) przepływu sterowania na kilka współbieżnie działających przepływów sterowania

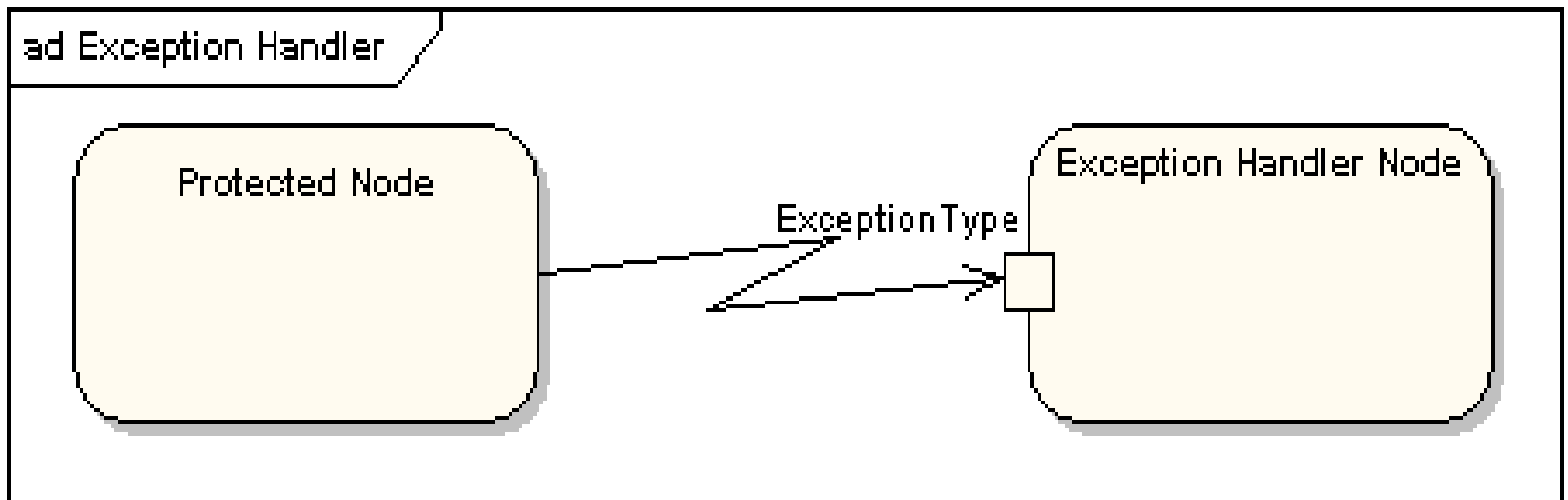
**Węzeł łączenia** (*Join* - pionowa lub pozioma linia) współbieżnie działających przepływów sterowania do jednego przepływu sterowania –po zakończeniu każdego z tych współbieżnych procesów

**Region rozszerzający - powtarzanie czynności:** iteracyjnie (*iterative*), równoległe (*parallel*) lub w postaci strumienia (*stream*) – nazwa sposobu wykonania regionu diagramu czynności

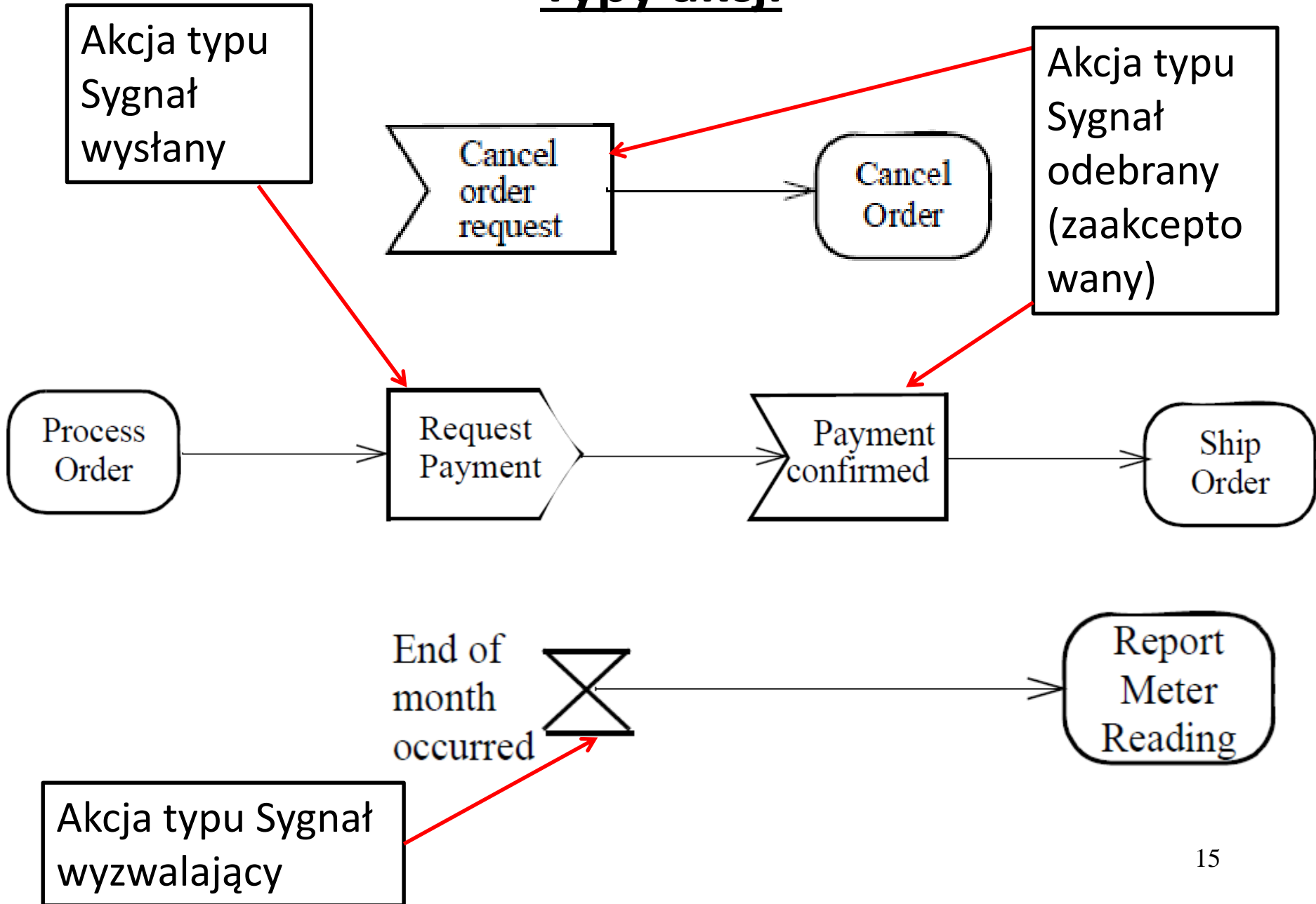


## Obsługa wyjątków

Np. Reakcja na błąd podczas wykonania akcji w czynności „*Protected Node*” – nastąpi przerwanie tych akcji i przejście do wykonania akcji w czynności „*Exception Handler Node*”

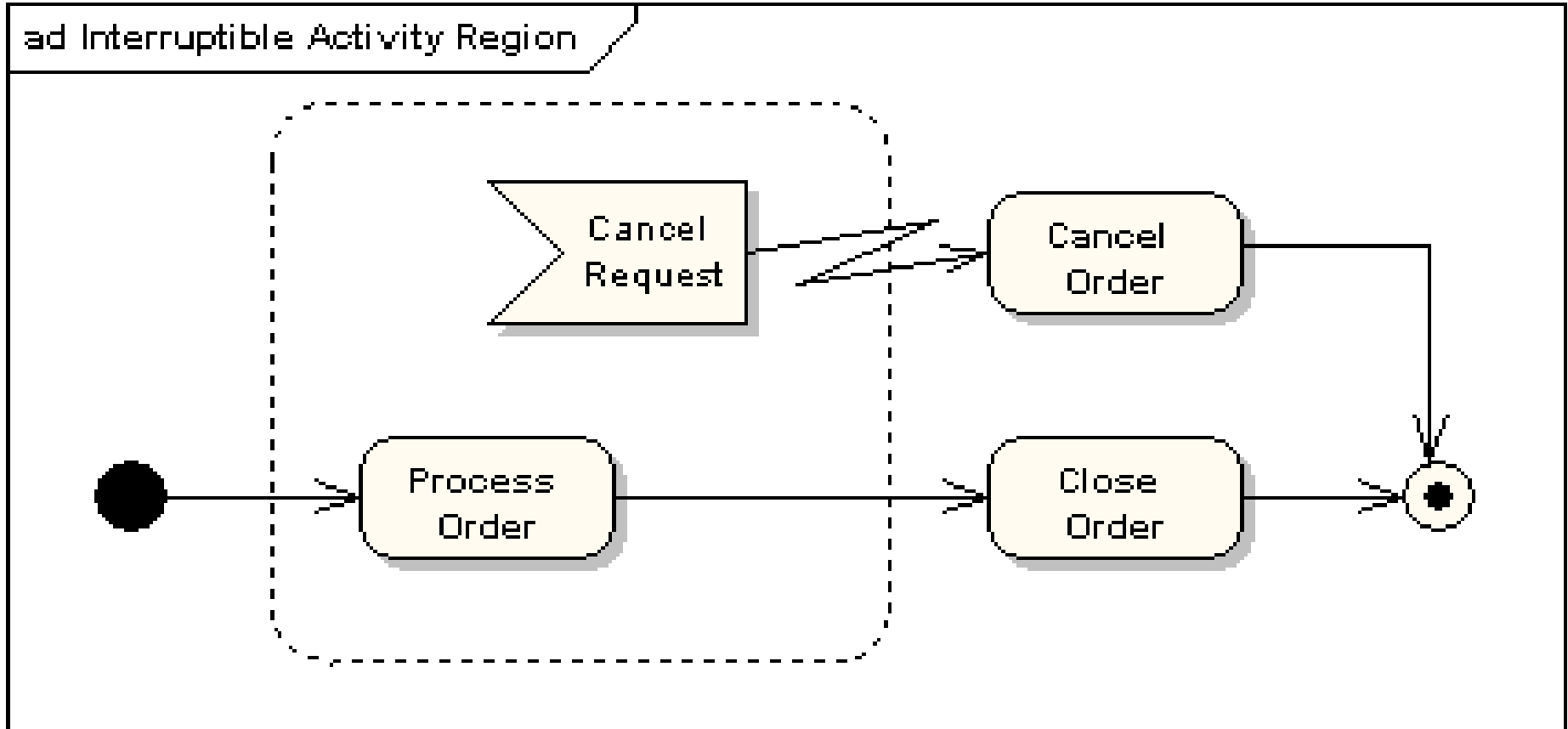


# Typy akcji



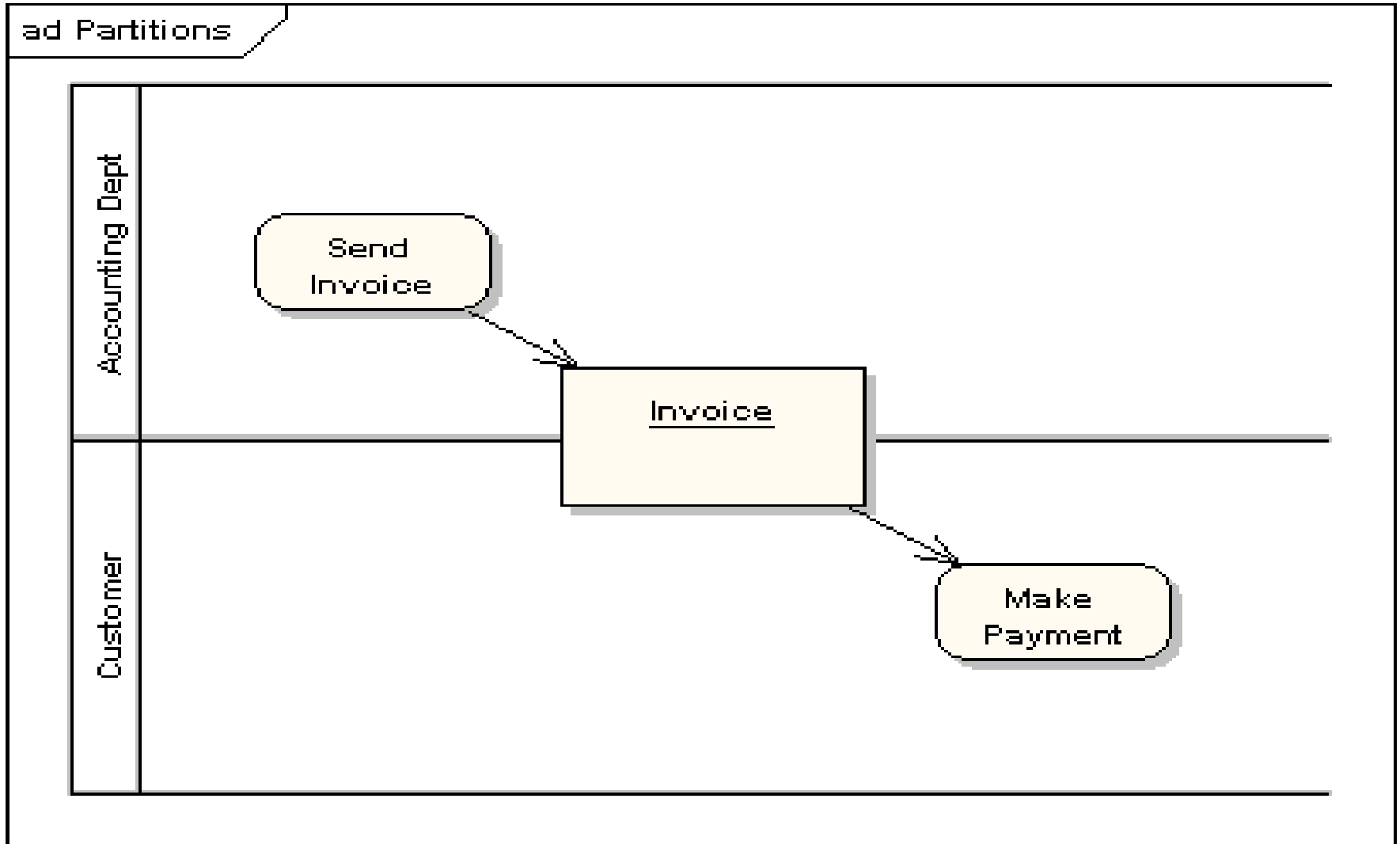
## Obsługa przerwania reakcja na inną akcję niż typowa

Np. Akcja "*Process Order*" aktywności zostanie wykonana i następnie można przejść do kolejnej czynności i wykonać akcję „*Close Order*” i zakończyć proces. Jednak w przypadku, gdy podczas akcji „*Process Order*” nastąpi przerwanie „*Cancel Request*”, zostanie wykonana akcja „*Cancel Order*” w innej czynności i nastąpi zakończenie procesu.





**Partycje(tory)** – np. podział czynności wykonywanych na obiekcie Faktura (*Invoice*) przez dwa różne obiekty reprezentowane przez partycje: Wydział Finansowy (*Accounting Department*) i klienta (*Customer*).



# Diagramy czynności - tworzenie modelu przypadków

## 1. Diagramy czynności UML

[http://sparxsystems.com.au/resources/uml2\\_tutorial/](http://sparxsystems.com.au/resources/uml2_tutorial/)

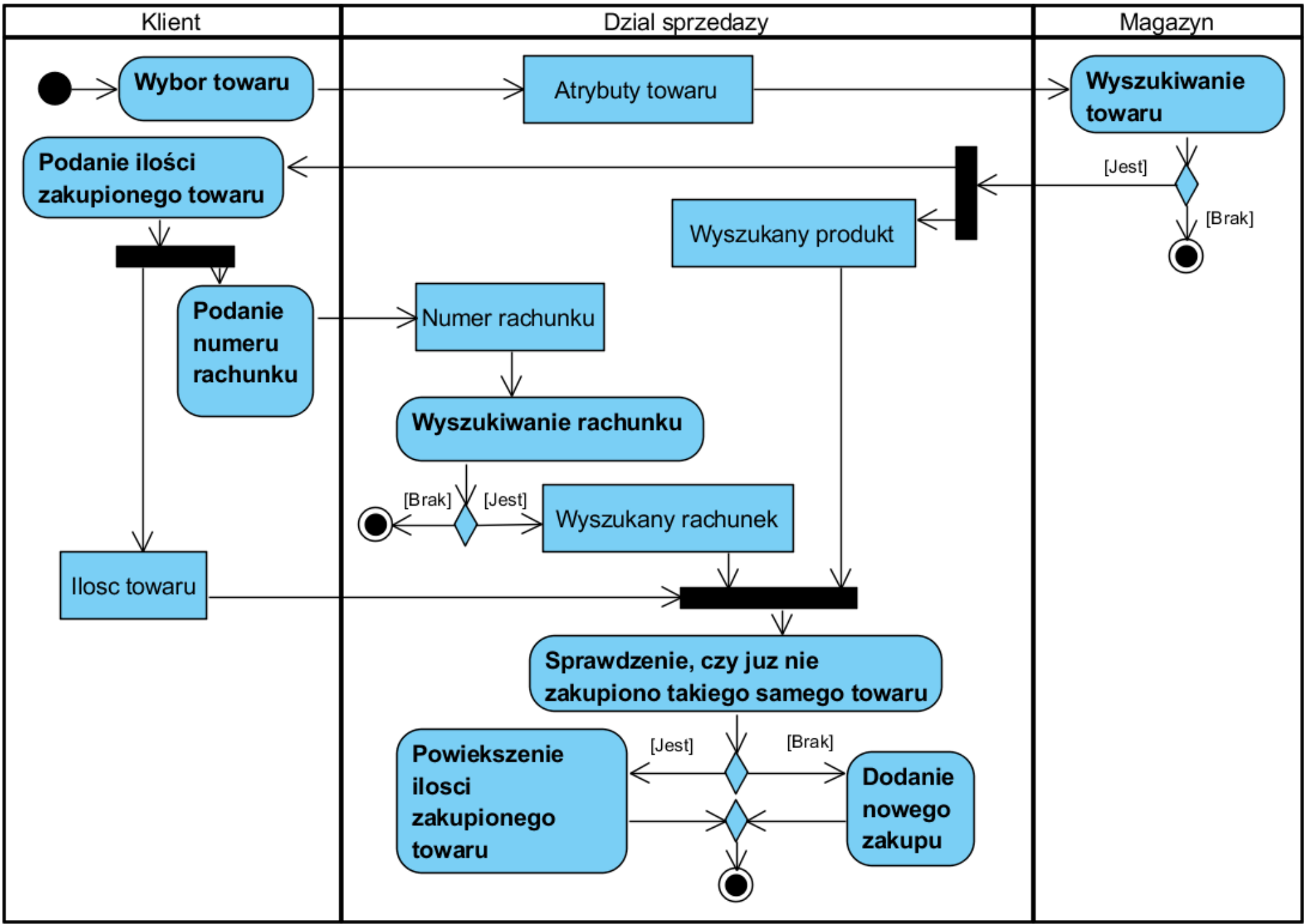
## 2. Przykład diagramów czynności UML – modelowanie przepływu czynności i obiektów

(wg Booch G., Rumbaugh J., Jacobson I., UML przewodnik użytkownika)

# Diagramy czynności - System sporządzania rachunków

## Kontynuacja (przykład 3 z wykładu 2)

**(1) Diagram czynności jako model biznesowy systemu sprzedaży towarów – proces zakupu**



# (1 cd) Obliczanie wartości rachunku

Sklep Spożywczo – Przemysłowy „ABC”

Jan Kowalski

ul. Leśna 1, xx-xxx Jakieś miasto

NIP xxx-xxx-xx-xx

Dn. 07r-09-24

nr wydr.8212

## PARAGON FISKALNY

XXXXXXXXXXXXXXXXXX

Nazwa produktu1 xxxxx

XXXXXXXXXXXXXXXXXX

Nazwa produktu2 xxxxx

Nazwa produktu3 xxx

XXXXXXXXXXXXXXXXXX

Nazwa produktu4 xxxxx

Sp.op.A 11.77

Sp.op.B 2.36

Sp.op.D 2.75

To jest cena brutto  
towarów z danej  
kategorii podatku

**RAZEM ZŁ**

To jest ilość  
zakupioneg  
o towaru

$1 * 6.79$

$4 * 0.59$

$0.6 * 4.59$

$2 * 2.49$

To jest cena  
jednostkowa brutto

A

B

D

A

To są kategorie  
podatków

PTU A = 22.00%

PTU B = 7.00%

PTU D = 3.00%

Razem PTU

16.88

2.12

0.15

0.08

2.35

To są kwoty  
tara  
wynikające z  
istniejących  
kategorii  
podatków

## **(2) Wykład 2 - Przykład 3. System sporządzania rachunków**

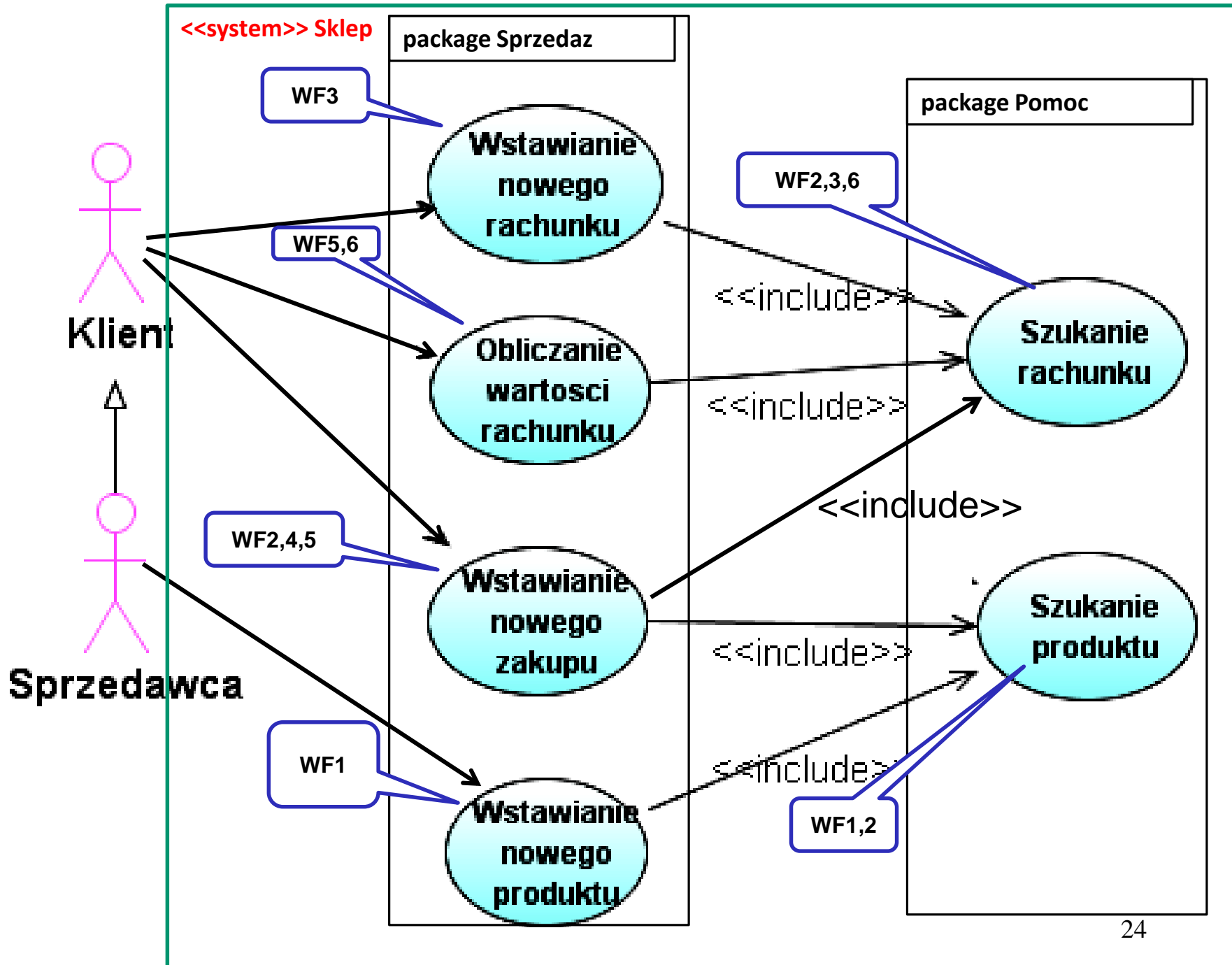
### **Lista wymagań funkcjonalnych**

1. System zawiera katalog produktów
2. Można zakupić trzy typy produktów różniące się sposobem obliczania ceny detalicznej: : bez promocji i bez podatku, z promocją i bez podatku, z podatkiem bez promocji, z podatkiem i z promocją,
3. Można wprowadzić wiele rachunków
4. Pozycje rachunku muszą zawierać produkty różne w sensie nazwy, ceny, podatku i promocji
5. Każda pozycja rachunku powinna podać swoją wartość brutto oraz dane produktu oraz ilość zakupionego produktu.
6. Na rachunku powinna znajdować się wartość łączna wszystkich zakupów oraz wartości zakupów należących do wybranych kategorii

### **Lista wymagań niefunkcjonalnych**

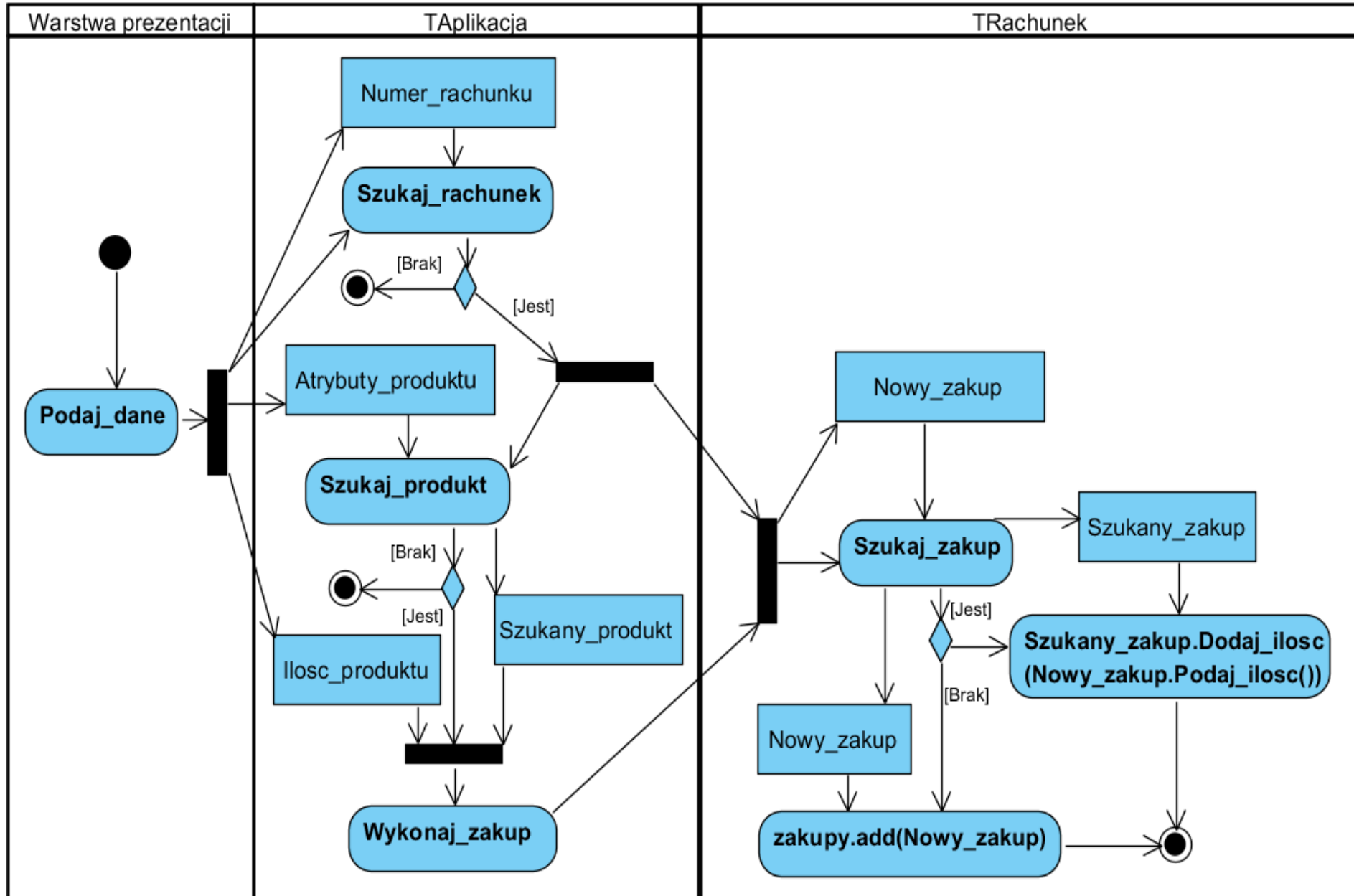
1. Wstawianie produktów może odbywać się tylko przez uprawnione osoby
2. Wstawianie nowych rachunków oraz wstawianie nowych zakupów jest dokonywane przez klientów
3. Zakupy mogą być dokonane przez Internet przez aplikację uruchamianą przez przeglądarkę lub bez jej pośrednictwa

AKTOR	OPIS	PRZYPADKI UŻYCIA
Klient	<i>Klient może dokonywać zakupów wybranych produktów przez Internet korzystając z przeglądarki lub z aplikacji</i>	<ul style="list-style-type: none"> <li>• Wstawianie nowego rachunku powiązane przez &lt;&lt;include&gt;&gt; z PU Szukanie rachunku</li> <li>• Obliczanie wartości rachunku powiązane przez &lt;&lt;include&gt;&gt; z PU Szukanie rachunku</li> <li>• Wstawianie nowego zakupu powiązane przez &lt;&lt;include&gt;&gt; z PU Szukanie rachunku oraz powiązane przez &lt;&lt;include&gt;&gt; z PU Szukanie produktu</li> </ul>
Sprzedawca	<i>Sprzedawca może dodatkowo dodawać nowe produkty</i>	<ul style="list-style-type: none"> <li>• Wstawianie nowego rachunku powiązane przez &lt;&lt;include&gt;&gt; z PU Szukanie rachunku</li> <li>• Obliczanie wartości rachunku powiązane przez &lt;&lt;include&gt;&gt; z PU Szukanie rachunku</li> <li>• Wstawianie nowego zakupu powiązane przez &lt;&lt;include&gt;&gt; z PU Szukanie rachunku oraz powiązane przez &lt;&lt;include&gt;&gt; z PU Szukanie produktu</li> <li>• Wstawianie nowego produktu powiązane przez &lt;&lt;include&gt;&gt; z PU Szukanie produktu</li> </ul>





(5) Diagram czynności przypadku użycia *Wstawianie nowego zakupu* (model przypadku użycia w warstwie biznesowej – partycje TApplikacja i TRachunek)



(6) Kod źródłowy metody *Wstaw\_zakup* obiektu typu *TAplikacja* – pierwsza część realizacji przypadku użycia *Wstawianie nowego zakupu*

```
//TAplikacja

public void Wstaw_zakup (int nr, int ile, String dane[])
{
    TRachunek rachunek;
    TProdukt1 produkt1 = fabryka.Podaj_produkt(dane);
    if ((rachunek=Szukaj_rachunek(nr)) != null)
        if ((produkt1=Szukaj_produkt(produkt1)) != null)
            rachunek.Dodaj_zakup(new TZakup(ile, produkt1));
}
```

**(7)** Kod źródłowy metody *Dodaj\_zakup* obiektu typu *TRachunek* – druga część realizacji przypadku użycia ***Wstawianie nowego zakupu***

```
//TRachunek

private ArrayList<TZakup> Zakupy = new ArrayList<TZakup>();

public void Dodaj_zakup (TZakup aTZakup)
{
    TZakup zakup;
    if ((zakup = Szukaj_zakup(aTZakup)) != null)
        zakup.Dodaj_ilosc(aTZakup.Podaj_ilosc());
    else
        Zakupy.add(aTZakup);
}
```

# Wytyczne dla tworzenia diagramów czynności

1. Należy ustalić najważniejsze czynności - nie można przedstawić na jednym diagramie wszystkich czynności
2. Należy wybrać obiekty przedsiębiorstwa, które są zobowiązane do realizacji bardziej ogólnego przepływu. Mogą to być elementy rzeczywiste, istniejące w słownictwie systemu, lub elementy abstrakcyjne. W obu przypadkach należy utworzyć tor dla każdego wybranego obiektu.
3. Należy zidentyfikować stan początkowy i końcowy modelowanego przepływu.
4. Przechodząc od stanu początkowego do końcowego należy modelować kolejne stany czynności lub stany akcji.
5. W przypadku złożonych akcji lub często występujących zbiorów akcji należy je połączyć w stany czynności. Z każdym stanem skojarz oddzielny diagram czynności, który przedstawia zebrane nim akcje.
6. Należy zobrazować przepływy czynności między stanami akcji i stanami czynności. Pierwsze powinny być brane pod uwagę przepływy sekwencyjne, potem rozgałęzienia, na końcu rozwidlenia i scalenia.
7. Jeśli w modelowanym przepływie czynności biorą udział istotne obiekty, należy je umieścić na diagramie. Należy uwzględniać zmieniające się atrybuty i stany tych obiektów, jeśli jest to konieczne do zrozumienia przepływu tych obiektów.

# Diagramy czynności - tworzenie modelu przypadków

## 1. Diagramy czynności UML

[http://sparxsystems.com.au/resources/uml2\\_tutorial/](http://sparxsystems.com.au/resources/uml2_tutorial/)

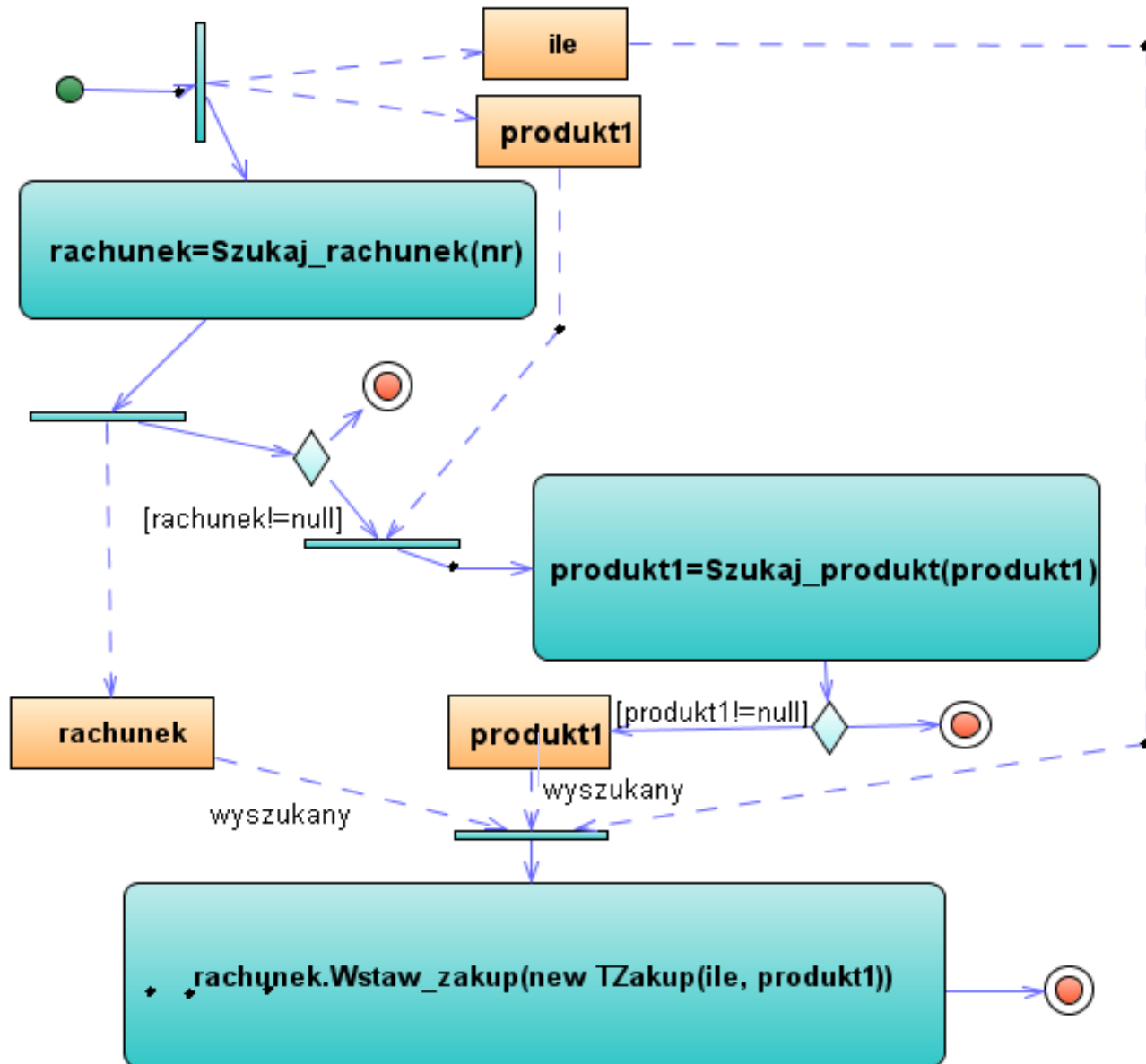
## 2. Przykład diagramów czynności UML – modelowanie przepływu czynności i obiektów

(wg Booch G., Rumbaugh J., Jacobson I., UML przewodnik użytkownika)

## 3. Przykład diagramów czynności UML – modelowanie operacji

(wg Booch G., Rumbaugh J., Jacobson I., UML przewodnik użytkownika)

# Modelowanie operacji *Wstaw\_zakup*



# Kod źródłowy projektowanej operacji za pomocą diagramu aktywności

```
public void Wstaw_zakup (int nr, int ile, TProdukt1 produkt1)
{
    TRachunek rachunek;
    if ((rachunek=Szukaj_rachunek(nr))!=null )
        if ((produkt1=Szukaj_produkt(produkt1))!=null )
            rachunek.Wstaw_zakup(new TZakup(ile,produkt1));
}
```

# Wytyczne dla tworzenia diagramów czynności do modelowania operacji

1. Należy uwzględnić abstrakcje uczestniczące w tej operacji. Są to parametry przekazywane do metody, atrybuty klasy metody oraz klas obiektów związanych z metodą.
2. Należy zidentyfikować warunki wstępne stanu początkowego i końcowego modelowanej operacji. Należy znaleźć wszystkie wartości, które nie mogą zmienić się podczas realizacji operacji.
3. Przechodząc od stanu początkowego do końcowego należy modelować kolejne stany czynności lub stany akcji.
4. Należy w razie potrzeby korzystać z rozgałęzień do modelowania ścieżek warunkowych i iteracji.
5. Jeśli operacja należy do klasy aktywnej (i tylko wtedy), należy wtedy korzystać z rozwidleń i scaleń, żeby określić równoległe przepływy sterowania.