



Politechnika Wroclawska

Diagramy stanów w języku UML

Maciej Troć



Plan prezentacji

- Definicja
- Elementy diagramów stanów
 - Przejścia
 - Stany
 - Pseudostany
 - Stany złożone
- Generowanie kodu
- Podsumowanie



Definicja

Maszyna stanowa określa ciągi stanów przyjmowanych przez obiekt w odpowiedzi na zdarzenia zachodzące w czasie jego życia, a także reakcje obiektu na te zdarzenia.

[UML - przewodnik użytkownika, G Booch, J Rumbaugh, I Jacobson]



Zastosowanie

Modelowanie historii życia:

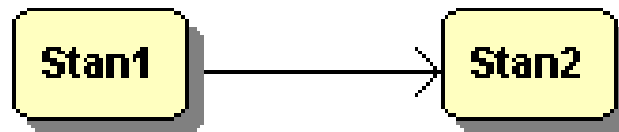
- Instancji klasy
- Przypadku użycia
- Całego systemu

Podstawowe elementy

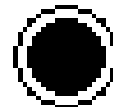
- Stan



- Przejście

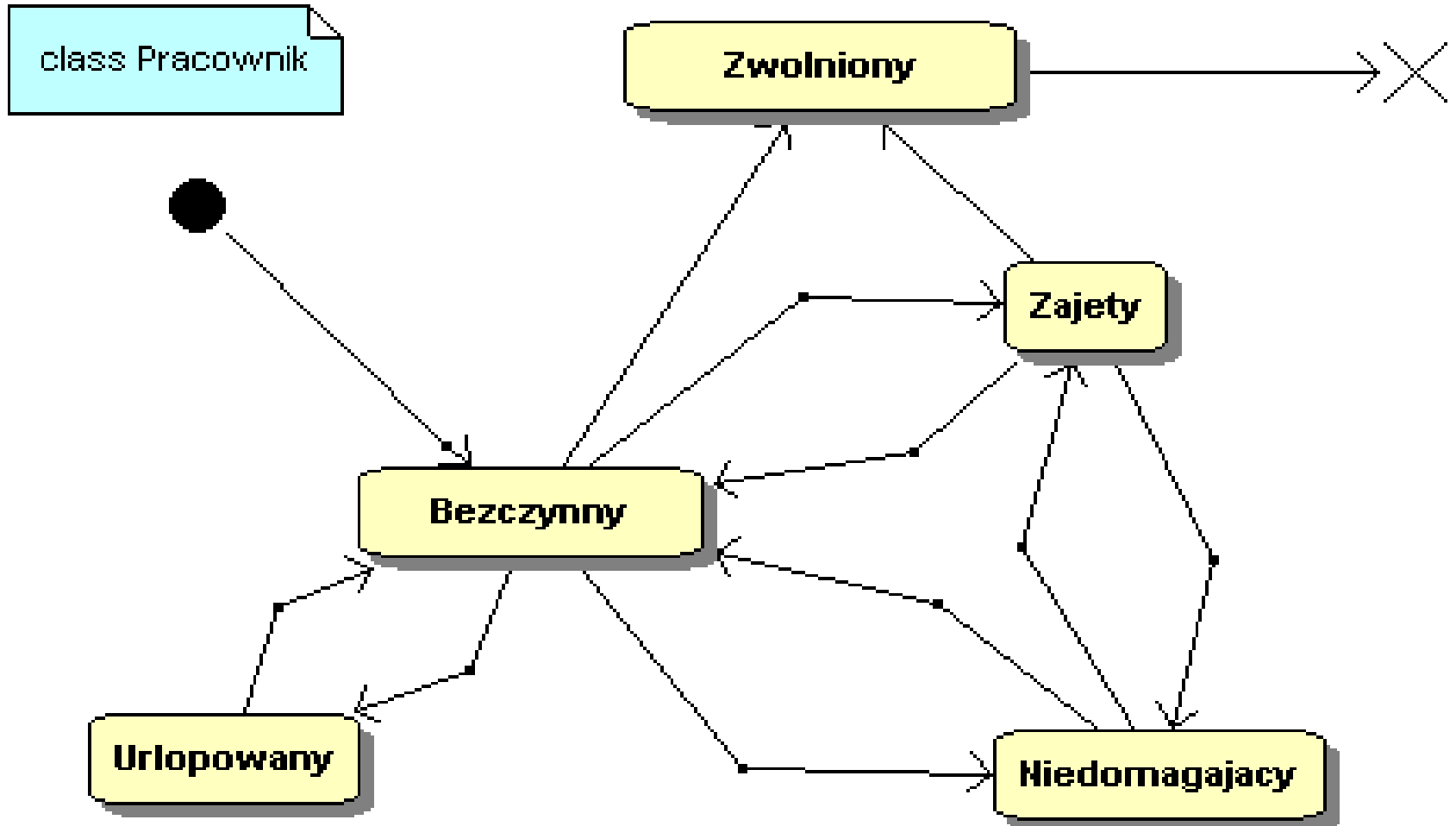


- Stan: początkowy, końcowy, zniszczenia





Prosty przykład





Przejścia

- Format

[<zdarzenie>] [[<warunek doz.>]] [/<akcja>]

- Przykład

Urlopuj(dni)[dni<=wolnych]/wyliczDatePowrotu(dni)



Zdarzenia

- **Sygnaty**
FloatingPointException
- **Zdarzenie wywołania**
obliczRate(cena, liczba_rat)
- **Zdarzenie czasowe**
after(1 h)
- **Zdarzenie zmiany stanu**
when(Day() == 7)
- **Zdarzenie odroczone**
wygenerujStatystyke() / defer

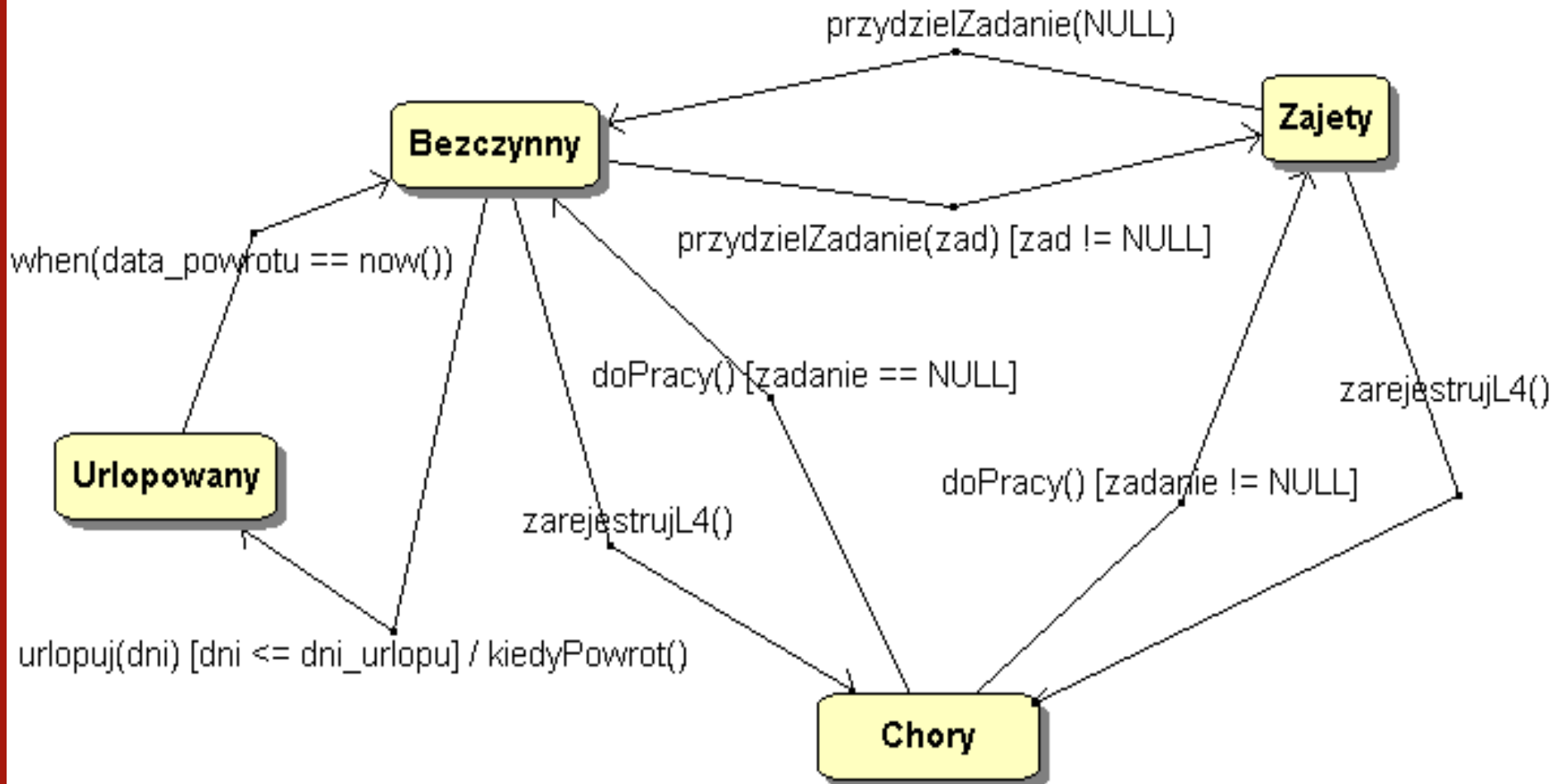


Akcje

- Wywołanie operacji
 - Własnej
 - Obiektu powiązanego
- Utworzenie / zniszczenie obiektu
- Wysłanie sygnału (*send daneWBuforze*)



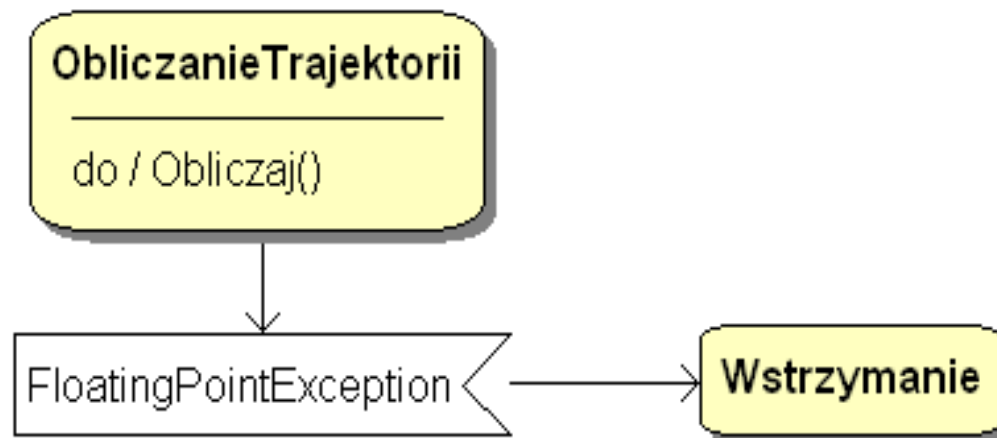
Zdarzenia i przejścia - przykład





Sygnały 1/2

- Komunikaty asynchroniczne, nie związane z przekazaniem sterowania
- Rzadko używane do modelowania obiektów pasywnych



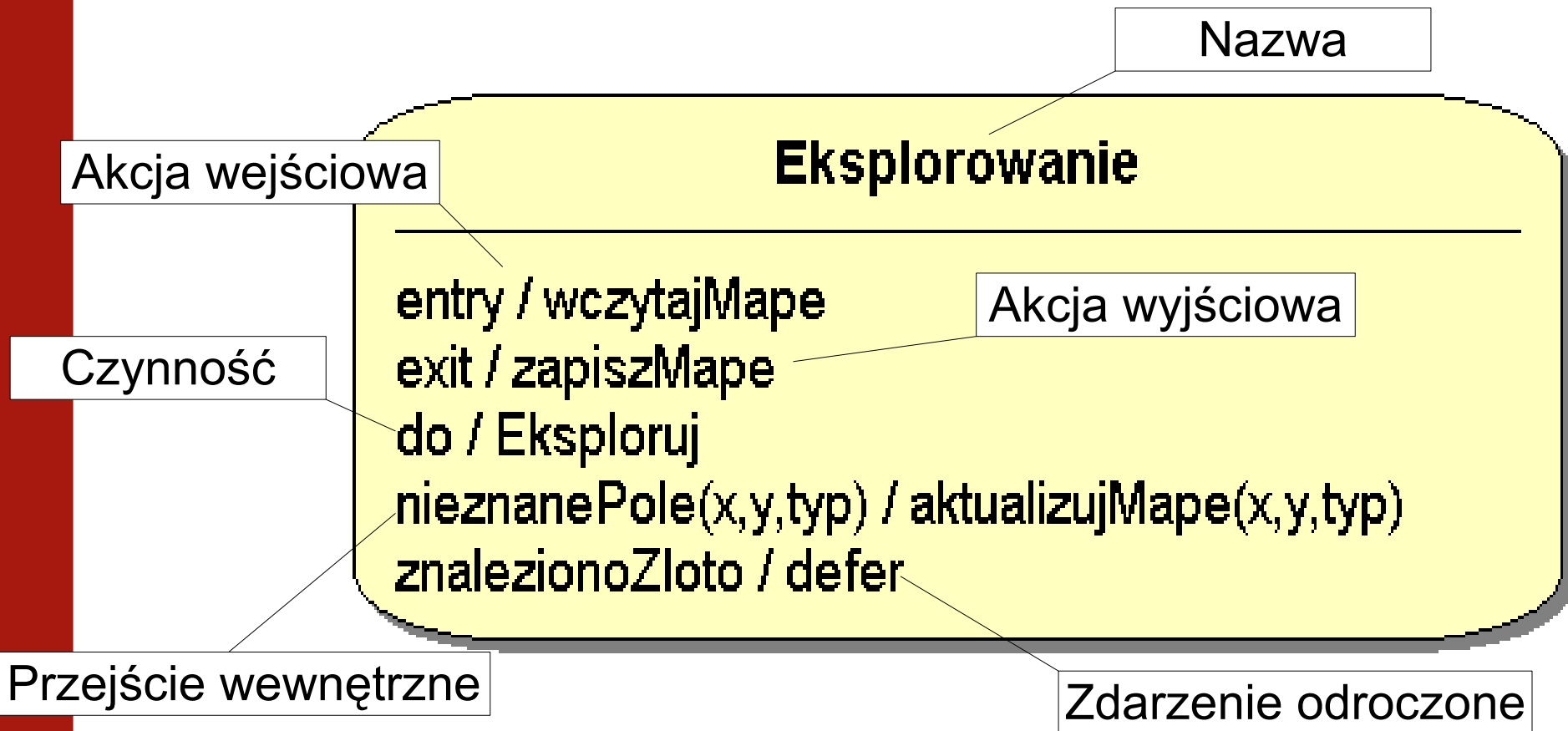


Sygnały 2/2 - zastosowania

- Komunikacja między obiektami aktywnymi
(wątki, procesy)
- Wyjątki
- Przerwania



Opis stanu



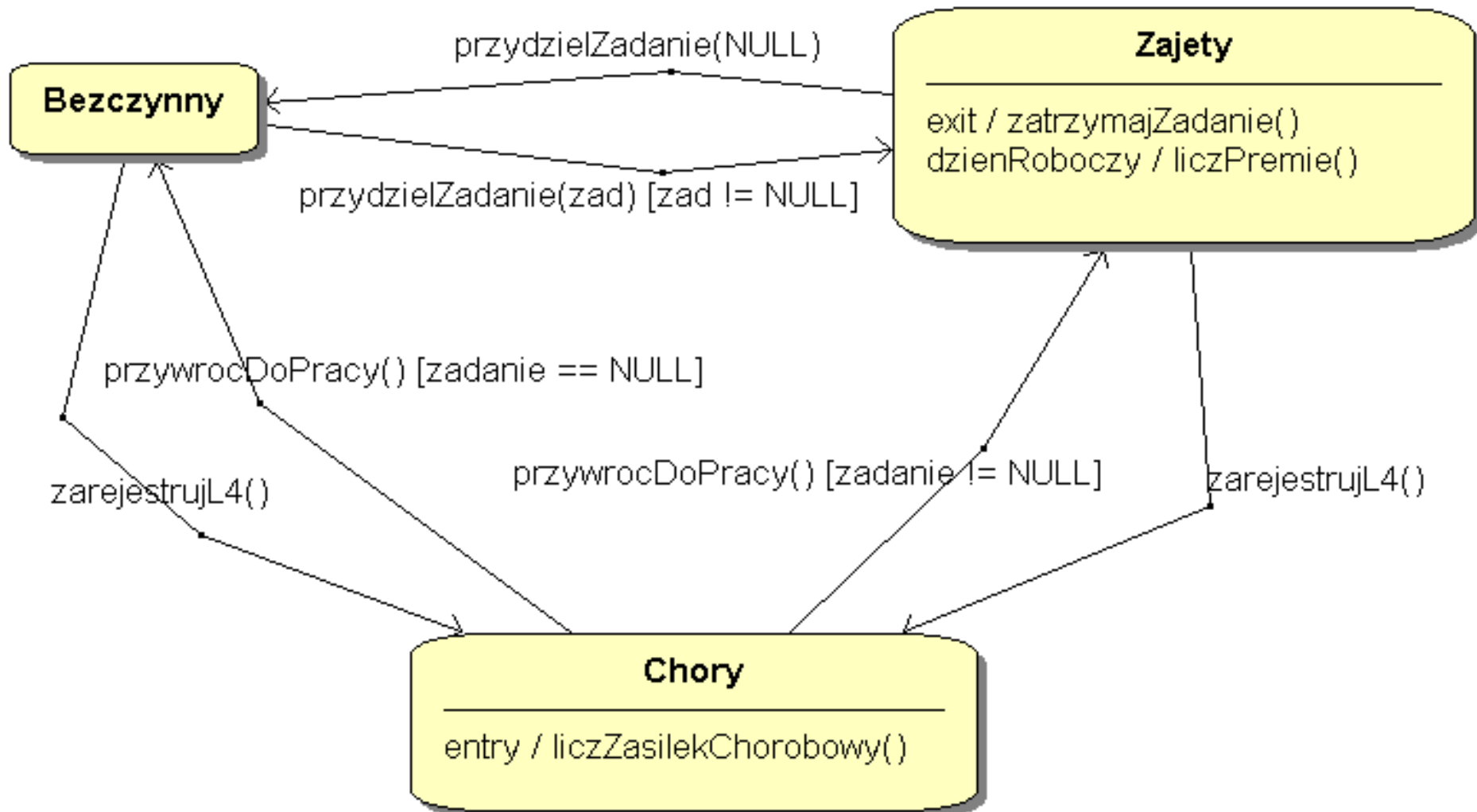


Czynności

- Zadania realizowane do momentu zajścia zdarzenia
- Podzielne
- Mogą być opisane przy pomocy ciągu akcji
do / akcja1();akcja2();akcja3()



Opis stanu - przykład



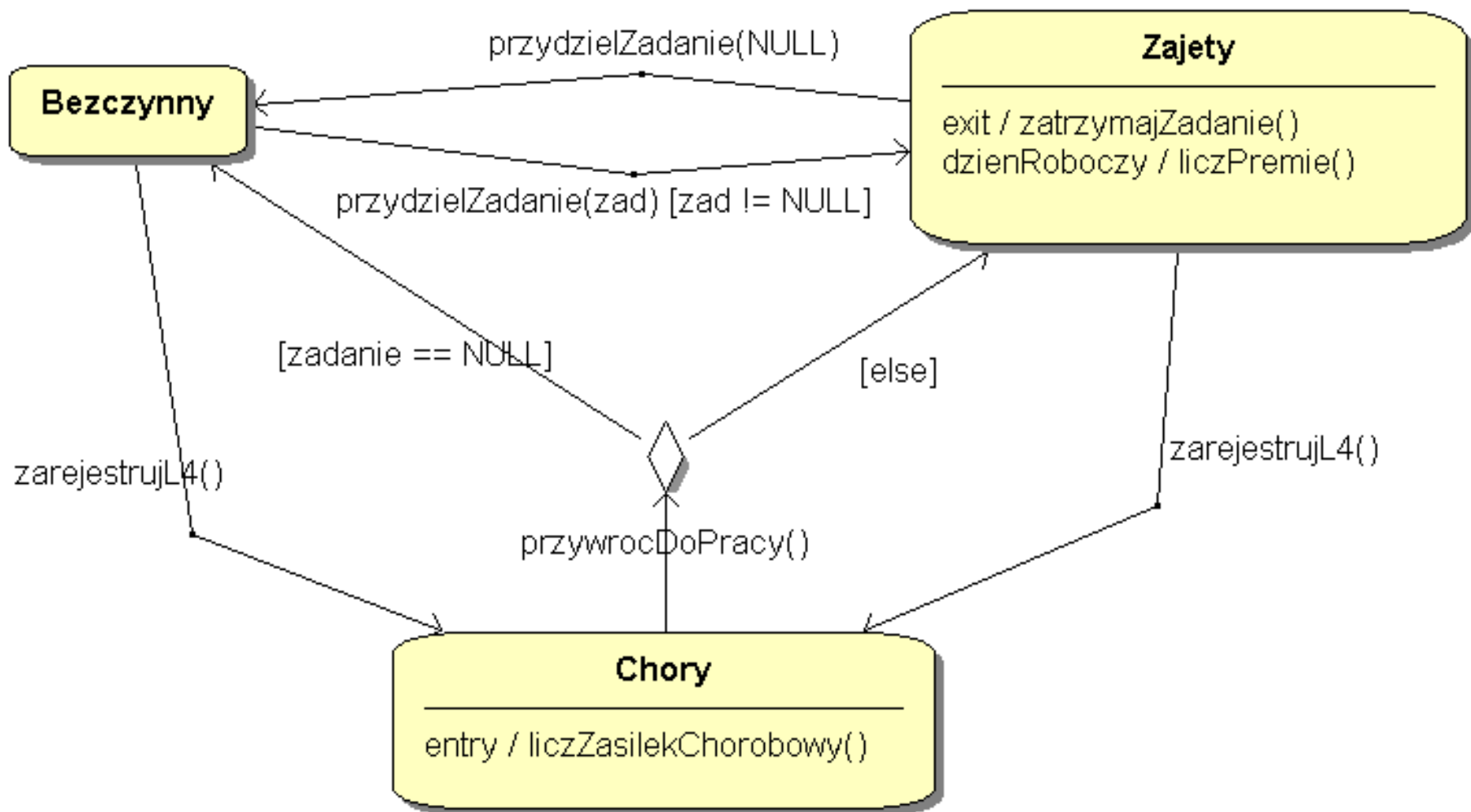


Pseudostany

- Podstawowe
stan początkowy, stan końcowy, punkt zniszczenia
- Rozwidlenie i scalenie
- Rozgałęzienia
decyzja, złączenie, punkt węzłowy
- Powiązane ze stanami złożonymi
płytkie wznowienie, głębokie wznowienie, punkt wejścia, punkt wyjścia

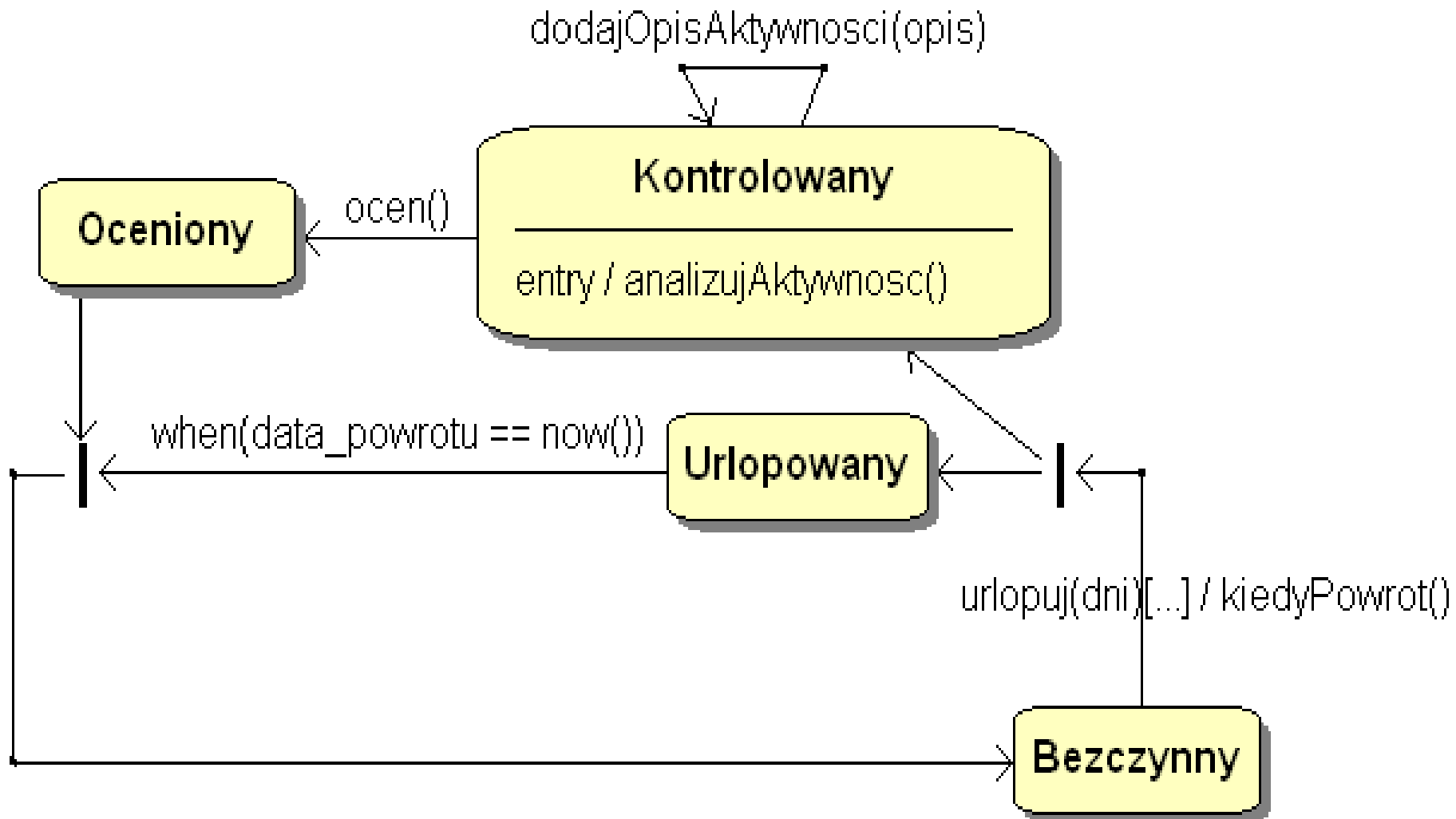


Pseudostany - rozgałęzienie





Pseudostany - rozwidlenie i scalenie



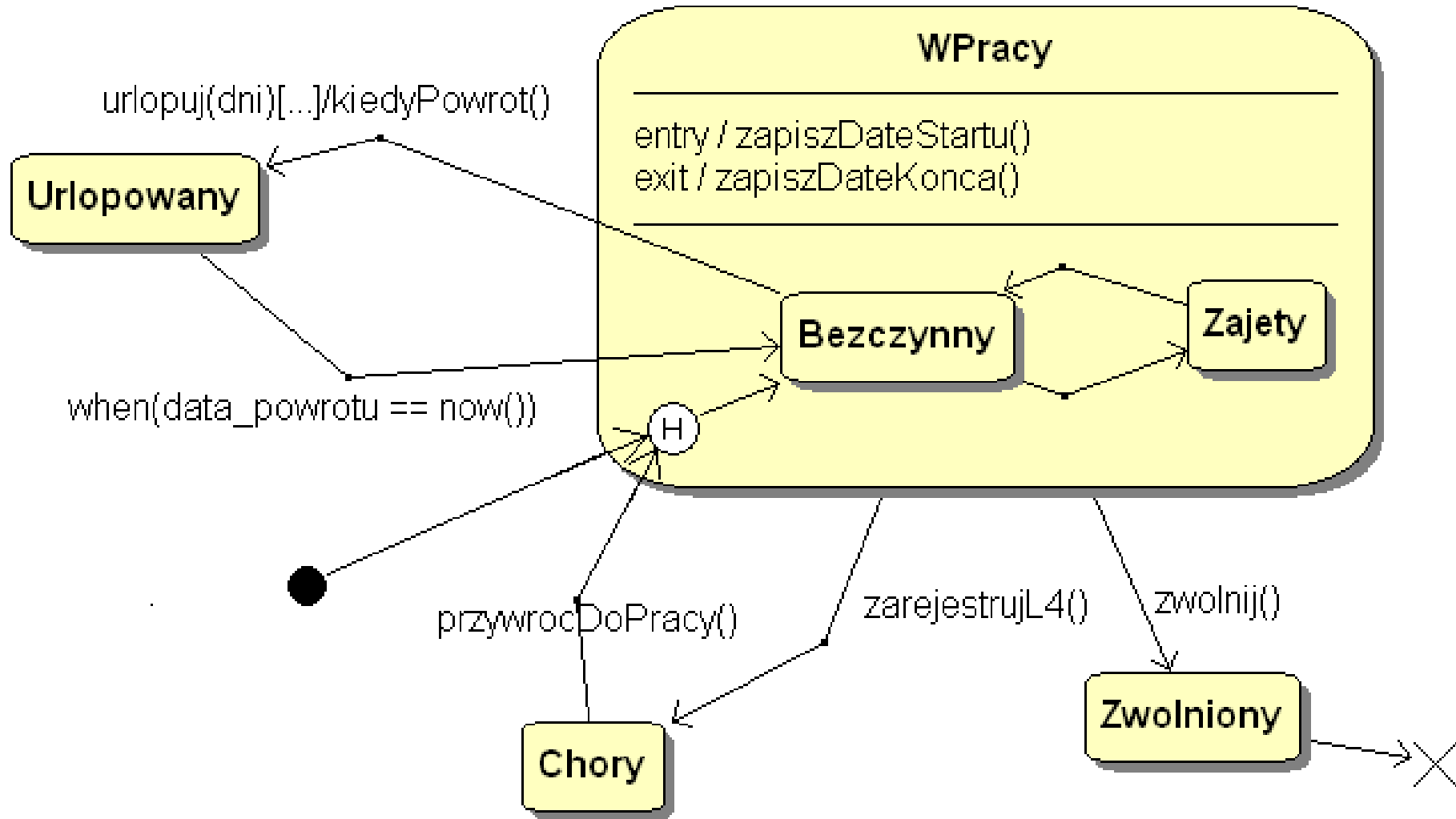


Stany złożone

- Stany, które zawierają:
 - Podmaszyny stanowe (*podstany sekwencyjne*)
 - Obszary współbieżne (*podstany współbieżne*)

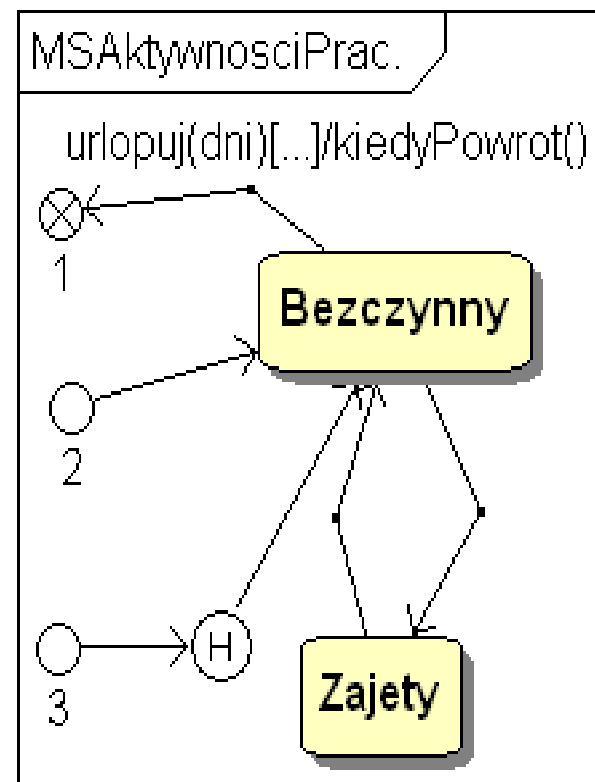
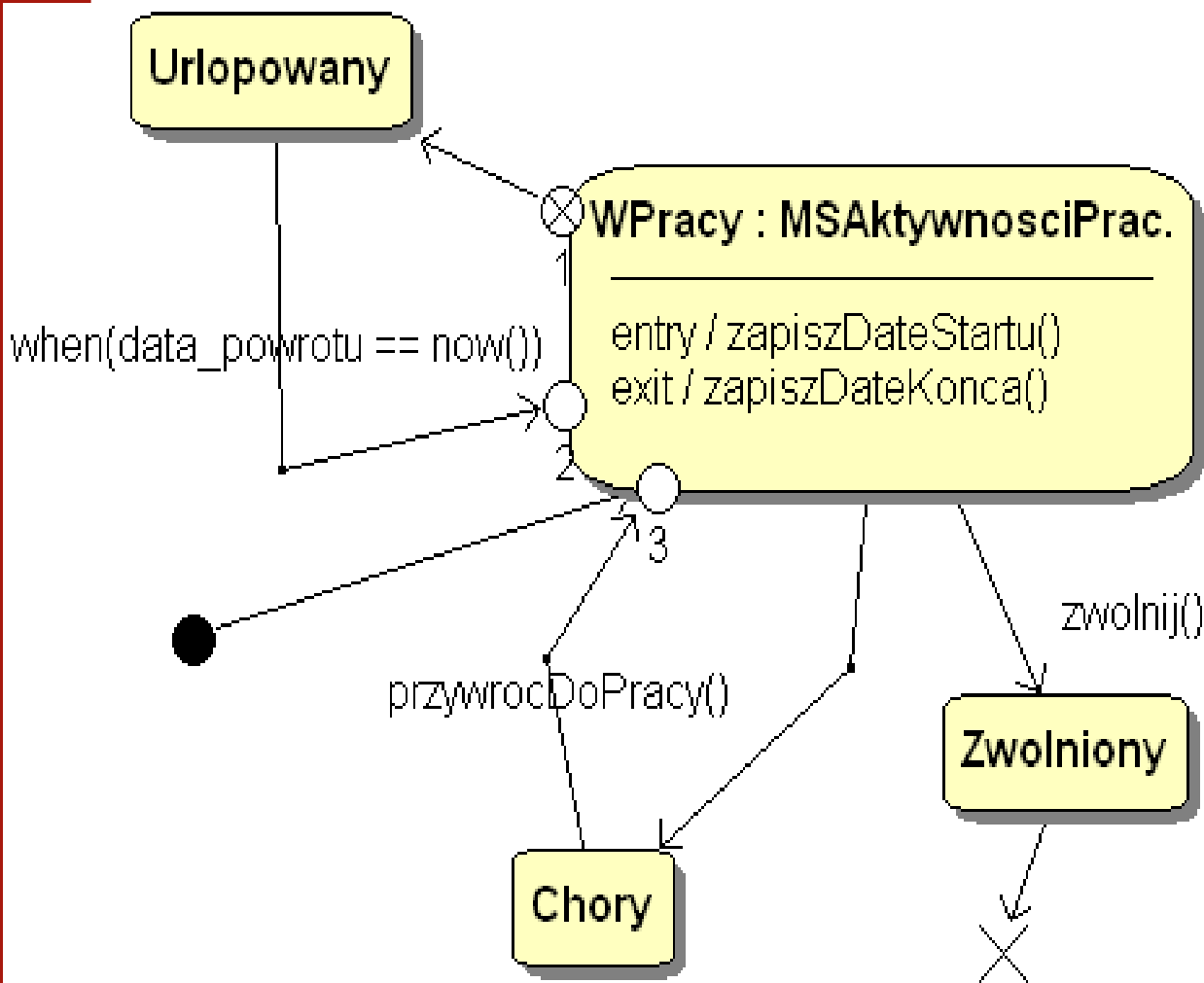


Sekwencyjne podmaszyny stanowe



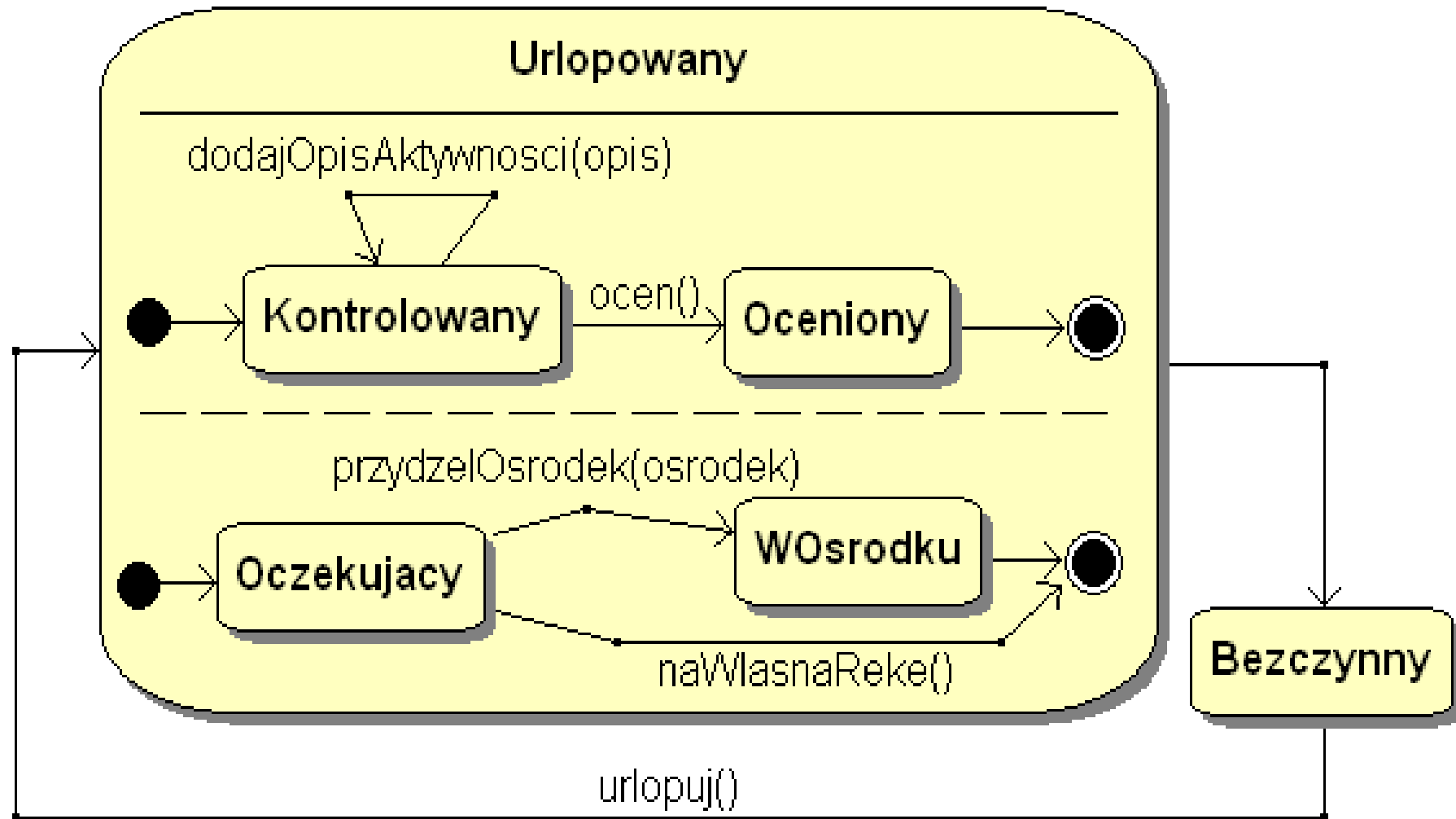


Punkty wejścia i wyjścia



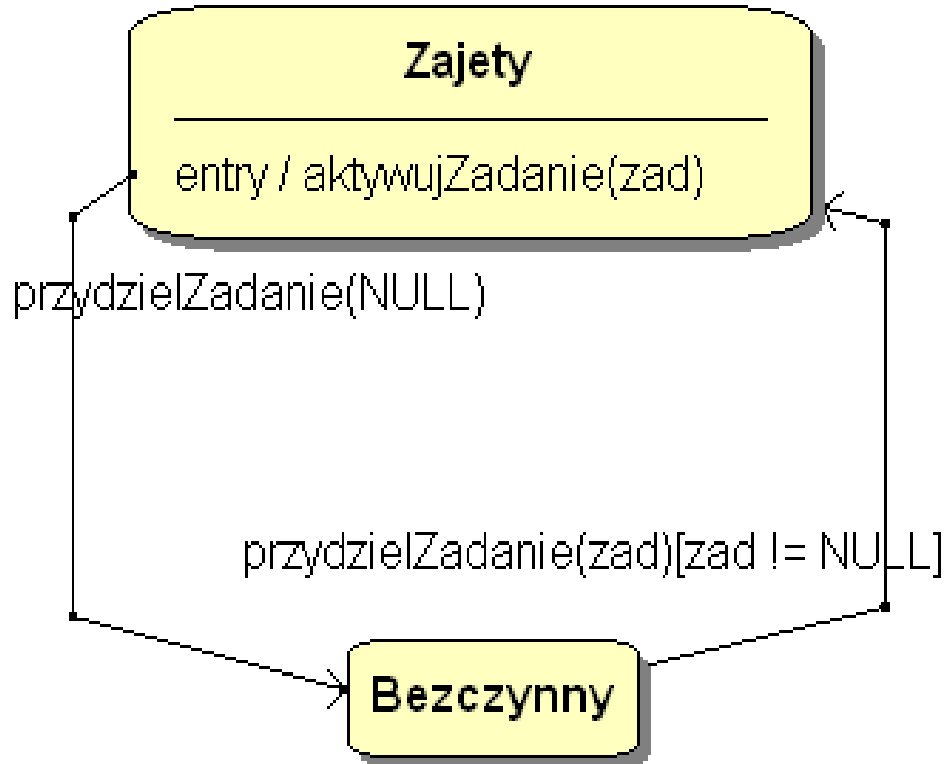


Obszary współbieżne





Generowanie kodu 1/2



```
...  
void przydzielZadanie (Zadanie* zad)  
{  
    switch(stan) {  
        case Zajety:  
            if(zad == NULL)  
                stan = Bezczynny;  
            break;  
        case Bezczynny:  
            if(zad != NULL)  
            {  
                aktywujZadanie(zad);  
                stan = Zajety;  
            }  
            break;  
    }  
}  
...
```



Generowanie kodu 2/2 - podejście zaawansowane [4]

- Stany zamodelowane pod postacią klas dziedziczących po klasie *State*
- Każdy stan - klasa posiada definicje:
 - Operacji *entry()* i *exit()*
 - Operacji reprezentujących przejścia do innych stanów (np. *przydzielZadanie()*)
- Modelowane byty (klasy) dysponują listą obiektów - stanów



Podsumowanie

- Diagramy stanów stanowią graficzną reprezentację maszyn stanowych
- Reprezentują jeden z popularniejszych sposobów modelowania dynamiki
- Wykazują dużą przydatność w inżynierii do przodu



Literatura

1. Wrycza S, Marcinkowski B, Wyrzykowski K. Język UML 2.0 w modelowaniu systemów informatycznych. Helion, 2005.
2. Booch G, Rumbaugh J, Jacobson I. UML - przewodnik użytkownika. WNT, 2001.
3. BOUML, www.bouml.free.fr.
4. Niaz I A, Tanaka J. Mapping UML statecharts to Java code. 2004