

# **Podstawy Inżynierii Oprogramowania**

## *Diagramy Stanów*

Wersja 1.2

opracował dr inż. Radosław Adamus

## Historia zmian

<i>Data</i>	<i>Wersja</i>	<i>Autor</i>	<i>Opis zmian</i>
30.11.2010	0.1	Radosław Adamus	Utworzenie dokumentu
3.12.2010	1.0	Radosław Adamus	Pierwsza oficjalna wersja dokumentu
3.12.2011	1.1	Radosław Adamus	Rozbudowanie opisu, poprawki edycyjne dla edycji 2011
6.12.2011	1.2	Radosław Adamus	Poprawka dotycząca semantyki stanu końcowego (wg UML 2.4 nie jest pseudo-stanem tylko stanem)

# Wprowadzenie

Diagramy stanów (State Machine Diagrams) należą do grupy diagramów dynamicznych, pozwalających na modelowanie zachowania systemu. Diagramy stanów pozwalają modelowanie zmian w zachowaniu wybranych elementów systemu (np. obiektów) w odpowiedzi na zmianę ich stanu. Przyjmiemy, że punktem odniesienia do wyjaśnienia pojęcia stanu będzie obiekt i rozważać będziemy stan obiektu (przyjmując, że w szczególnym przypadku cały system również jest obiektem).

## Stan obiektu i zmiana stanu.

Stan obiektu możemy rozumieć na kilka sposobów (wykorzystywanych w zależności od potrzeb) jako:

- Zestaw wartości jego atrybutów oraz powiązań z innymi obiektami w danej chwili czasowej. Z tego punktu widzenia zmianą stanu obiektu będzie każda modyfikacja wartości dowolnego atrybutu lub powiązania. Jednak nie każda zmiana wartości może spowodować zmianę stanu obiektu. Na przykład obiekt reprezentujący zbiornik, posiadający atrybut *poziom*, którego wartość oznacza poziom cieczy w zbiorniku, może zmienić swój stan gdy zmiana wartości atrybutu spowoduje przekroczenie ustalonego poziomu krytycznego. W takiej sytuacji zachowanie obiektu ulegnie zmianie i nie będzie możliwe dalsze zwiększanie wartości atrybutu *poziom*. Natomiast zmiana wartości tego atrybutu w zakresie poniżej wartości krytycznej nie wpływa na zachowanie obiektu (pomimo tego iż zmienia jego stan). Brak zmiany zachowania obiektu w odpowiedzi na zmianę jego stanu może być interpretowany jako tzw. zmiana wewnętrzna (obiekt zmienił swój stan, jednak nie ma to wpływu na jego zewnętrzne zachowanie).
- Okres czasu, w którym obiekt oczekuje na zdarzenie. W tym kontekście zmiana stanu jest równoważna pojawieniu się zdarzenia. Takim zdarzeniem może być zmiana wartości atrybutu/powiązania obiektu.
- Okres czasu, w którym obiekt przetwarza. W takiej sytuacji, obiekt wchodzi do stanu, w chwili rozpoczęcia przetwarzania a opuszcza go w chwili zakończenia.

## Maszyna stanów

Maszyna stanów opisuje możliwe stany obiektu wraz z dopuszczalnymi przejściami pomiędzy stanami. Reprezentacją maszyny stanów jest graf skierowany, którego węzłami są stany a krawędziami przejścia pomiędzy stanami. Węzeł grafu, reprezentujący stan, może być wyposażony w atrybuty przechowujące informacje o stanie, takie jak np. akcje wykonywane w momencie wejścia czy wyjścia ze stanu, aktywność wykonywaną w trakcie przebywania w danym stanie, czy też zestaw warunków, które są zawsze spełnione w danym stanie (tzw. niezmienniki).

Krawędzie grafu (przejścia pomiędzy stanami), są również być wyposażone w atrybuty informujące m.in. o zdarzeniu, które powoduje przejście,warunku, który musi zostać spełniony, aby przejście mogło nastąpić, czy też akcji która jest wykonywana w trakcie przejścia.

Rodzaje przejść:

- Przejście zewnętrzne - w którym następuje zmiana stanu.
- Samo przejście – szczególny przypadek przejścia zewnętrznego, w którym o węzeł docelowy jest równy węzłowi źródłowemu),
- Przejście wewnętrzne (w którym nie następuje zmiana stanu).

# Diagram Stanów

Elementem notacji języka UML służącego przedstawianiu maszyny stanów jest diagram stanów (ang. State machine diagram). Diagram ten należy do grupy diagramów dynamicznych, pozwalających na modelowanie zachowania systemu. Diagramy stanów umożliwiają modelowanie zmian w zachowaniu wybranych elementów systemu (np. obiektów) w odpowiedzi na zmianę ich stanu.

Do podstawowych elementów notacji wykorzystywanych na diagramie stanów należą:

## 1. Region

Podstawowym elementem podziału maszyny stanów w języku UML jest tzw. region. Stanowi on swego rodzaju kontener na elementy opisujące maszynę stanów (stany, pseudostany, przejścia). Pojedynczy diagram stanów reprezentuje domyślnie pojedynczy region. Regiony mogą również służyć do reprezentowania pod-maszyn stanów i mogą obejmować konkretny stan na diagramie (tzw. stan złożony). Podzielenie stanu (złożonego) na regiony oznacza, że w maszynie stanów istnieją wykluczające się wzajemnie „przestrzenie” stanów.

## 2. Stan

Podstawowy węzeł w grafie reprezentującym maszynę stanów. Modeluje sytuację, podczas której spełnione są pewne (zazwyczaj niejawne) warunki – niezmienniki stanu. Niezmienniki te mogą reprezentować sytuację statyczną – obiekt czeka na zaistnienie zdarzenia zewnętrznego. Możliwa jest również sytuacja, gdy niezmienniki te modelują sytuację dynamiczną – obiekt przetwarza, wykonuje operacje – po zakończeniu której następuje zmiana stanu. UML definiuje następujące rodzaje stanów:

- prosty
- złożony
- pod-maszyny stanów

## 3. Pseudo-stan

Abstrakcyjny węzeł maszyny stanów, który zazwyczaj wykorzystywany jest do wielu przejść w bardziej złożoną ścieżkę przejścia maszyny stanów. Typami pseudo-stanu są:

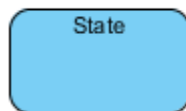
- Stan początkowy
- Decyzja
- Synchronizacja rozdzielenie (fork)
- Synchronizacja połączenie (join)
- Historia
- Łącznik
- Punkt wejścia
- Punkt wyjścia

## 4. Przejście

Skierowana relacja pomiędzy węzłami maszyny stanów (źródłowym i docelowym). Może być elementem złożonego przejścia modelującego pełną odpowiedź maszyny stanów na zaistnienie określonego typu zdarzenia. Przejście może posiadać opcjonalną listę wyzwalaczy specyfikujących zdarzenia indukujące przejście. Ponieważ to samo zdarzenie może powodować więcej niż jedno przejście, możliwe jest również definiowanie opcjonalnych warunków jakie muszą zaistnieć aby określony typ zdarzenia wyzwolił przejście.

**Notacja:**

### Stan (prosty):



Podstawowy węzeł w grafie reprezentującym maszynę stanów. Reprezentuje stan obiektu. Może on reprezentować sytuację statyczną (oczekiwanie na zdarzenie) lub dynamiczną (przetwarzanie).

Najważniejsze atrybuty:

**nazwa** - nazwa stanu

**entry** – opcjonalna akcja wykonywana w momencie wchodzenia do stanu (przed wykonaniem jakiegokolwiek operacji w obrębie danego stanu).

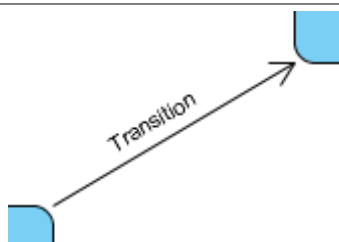
**exit** – opcjonalna akcja wykonywana w momencie wychodzenia ze stanu (po wykonaniu wszystkich operacji w obrębie danego stanu)

**do** – opcjonalna aktywność wykonywana w trakcie przebywania w bieżącym stanie.

Wykonywanie aktywności rozpoczyna się po wejściu do stanu a kończy samoczynnie (aktywność została ukończona) lub w momencie wyjścia z bieżącego stanu.

**State invariants** – niezmienniki stanu. Definiuje warunki, które są spełnione przez cały czas przebywania w bieżącym stanie.

### Przejście:



Skierowane powiązanie pomiędzy węzłem źródłowym a węzłem docelowym. Jeżeli węzłami są stany (lub pseudo-stany), obrazuje przejście maszyny stanów z jednego stanu do drugiego. Przejście jest odpowiedzią maszyny stanów na zaistnienie określonego typu zdarzenia.

Najważniejsze atrybuty:

**nazwa** - nazwa przejścia

**source** – stan źródłowy przejścia.

**target** – stan docelowy przejścia

**effect** – opcjonalna akcja wykonywana w momencie zmiany stanu

**guard** – warunek, którego spełnienie umożliwia dokonanie przejścia przy zaistnieniu określonego zdarzenia.

**Triggers** – wyzwalacze/zdarzenia, które mogą spowodować przejście.

Rodzaje zdarzeń: call (wywołanie operacji), time event (czas),

### (Pseudo-) stan początkowy:



Domyślny węzeł diagramu, stanowiący źródło dla przejścia do stanu domyślnego w bieżącym maszynie. Na określonym poziomie (regionie) może istnieć co najwyżej jeden taki element.

### Stan końcowy:



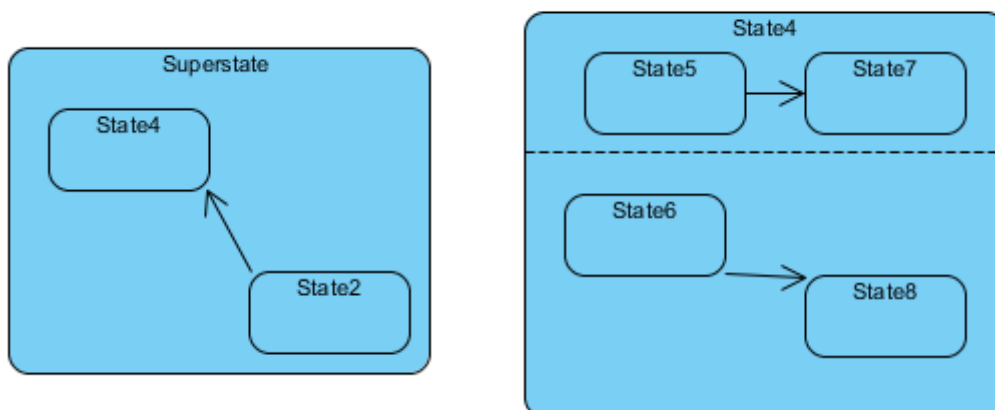
Specjalny stan, którego celem jest zasygnalizowanie, że dany region maszyny stanów został ukończony.

Najważniejsze atrybuty:

**nazwa** - nazwa stanu

**State invariants** – niezmienniki stanu. Definiuje warunki, które są spełnione przez cały czas przebywania w bieżącym stanie.

### Stan złożony:

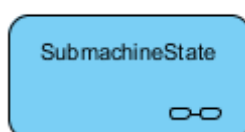


Stan złożony, którego opis składa się z pod-maszyny stanów. Każdy z pod-stanów dziedziczy przejścia nad-stanu. Tylko jeden z pod-stanów może być aktywny w danym momencie. Stan złożony może być podzielony na regiony. Każdy region zawiera zestaw wzajemnie się wykluczających rozłącznych węzłów. Z punktu widzenia semantyki stan złożony może w danym momencie być reprezentowany tylko przez jeden region.

Najważniejsze atrybuty:

patrz: **Stan**.

### Pod-maszyna stanów:



Element umożliwiający wstawienie specyfikacji pod-maszyny stanów opisanej zewnętrznie w stosunku do bieżącej maszyny. Semantycznie równoważny stanowi złożonemu.

Najważniejsze atrybuty:

**submachine** – nazwa pod-maszyny stanów

pozostałe: patrz **Stan**.

#### Historia:



Informuje, że maszyna stanu zapamiętuje swój stan. W razie powrotu do stanu nadrzędnego (stan złożony) pierwszym pod-stanem będzie stan historyczny (zamiast domyślnego stanu początkowego). W danym regionie może istnieć tylko jeden taki węzeł. Występuje w wersji „płytkiej” (ang. shallow) oraz „głębokiej” (ang. deep)

#### Decyzja:



Reprezentuję dynamiczną ewaluację warunku, której wynik określa ścieżkę przejścia.

#### Rozdzielenie (fork) /połączenie (join) procesów:



Wersja fork - Rozdzielenie przejścia na dwa lub większą liczbę niezależnych przejść, których stany docelowe znajdują się w różnych regionach. Wyjściowe przejścia nie mogą posiadać ustawionych atrybutów: zdarzenie oraz warunek.

Wersja join – Połączenie niezależnych przejść z różnych regionów do jednego przejścia.