

Poziom analityczny

**BPMN**

# W kierunku poziomu analitycznego



- Poziom opisowy – adaptacja diagramów przepływu do notacji BPMN
- Poziom analityczny wymaga precyzji semantycznej
  - Stosowanie się do ścisłych reguł w specyfikacji elementów BPMN
  - Ze specyfikacji BPMN trudno jest te reguły wydobyć.

# BPMN

semantyka

# Aspekty semantyczne



- Uruchamianie i kończenie
- Wysyłanie i odbieranie
- Zadania automatyczne
- Decyzje i reguły biznesowe

# Uruchamianie



- Interpretacja oparta na abstrakcji „silnika procesu”
  - Uruchamianie i kończenie zamiast „schematu blokowego”
  - Poziom opisowy:
    - Czynność A następuje po czynności B
  - Poziom analityczny
    - Po zakończeniu czynności A, natychmiast uruchamiana jest czynność B.

# Uruchamianie a rodzaj zadania



- Usługa (ang. Service task)
  - Wywołuje usługę (wysyła żądanie) w momencie, gdy przepływ procesu osiągnie ten punkt.
  - Kiedy usługa zostanie zrealizowana, kończy się zadanie i przepływ kontynuuje do kolejnego kroku.
- Zadanie użytkownika (ang. User task)
  - Zadanie jest prezentowane przypisanemu wykonawcy w postaci tzw. listy zadań (ang. worklist) – programowej reprezentacji
    - Na liście pojawia się nowe zadanie (ang. work item)
    - Wykonawca może zostać dodatkowo poinformowany
    - Może mieć przyłączone zdarzenie (ang. boundary event) typu czas
  - Technicznie zadanie zostało uruchomione (tak jak w przypadku Usługi), ale czeka na wykonanie.
- Podproces uruchamiany jest natychmiast po osiągnięciu przez przepływ jego symbolu
  - Zdarzenie początkowe bez typu

# Kto jest wykonawcą?



- Czynności są wykonywane przez zasoby procesu
- Zdarzenia i bramki nie są wykonywane przez zasoby procesu
  - Logika sterowania, którą posiada silnik procesu
  - Nawet jeżeli nie ma silnika ich logika jest wykonywana natychmiast po osiągnięciu w przepływie (i jest punktowa w czasie)

# Kończenie



- Przepływ procesu wyzwalany jest przez zakończenie czynności
- Zakończony (pod)proces
  - wszystkie równoległe ścieżki osiągnęły zdarzenie zakończenia – zakończenie prawidłowe
  - Nastąpiło przerwanie ze względu na wystąpienie sytuacji wyjątkowej – zakończenie nieprawidłowe



# Wysyłanie wiadomości (1)



- Wiadomość
  - Każda komunikacja pomiędzy procesem a elementem poza nim.
- Wysyłający na poziomie opisowym:
  - Zadanie użytkownika, usługa, podproces, zdarzenie zakończenia typu wiadomość
- Na poziomie analitycznym dodatkowo:
  - Zadanie typu wysłanie (ang. Send task), pośrednie zdarzenie rzucenia typu wiadomość (ang. Intermediate throw event)
  - Symboli nie można wykorzystać do wysyłania wiadomości pomiędzy torami!



Intermediate Throw Event

# Wysyłanie wiadomości (2)



Intermediate Throw Event

- Zadanie wysłanie wiadomości nie ma wykonawcy i jest równoważne zdarzeniu rzucenia typu wiadomość (nie wykonuje pracy i jest natychmiastowe).
  - Dobrze modeluje komunikację pomiędzy systemami (zgodnie z podejściem SOA)
  - Nie nadaje się do modelowania komunikacji „międzyludzkiej” (trzeba wykonać pracę – przeprowadzić rozmowę telefoniczną, napisać email)

# Wysyłanie wiadomości (3)



- Przepływ wiadomości reprezentuje „ludzką” formę komunikacji (np. fax, email)
  - Modelowany przez zadanie użytkownika (ang. User task) – uruchomienie oznacza dodanie zadania do listy zadań
- Przepływ wiadomości reprezentuje komunikację pomiędzy systemami
  - Modelowany przez zadanie lub zdarzenie rzucenia typu wiadomość

# Modelowanie komunikacji w granicach puli (procesu) (1)



- Jak przesłać wiadomość do wykonawcy zadania w kolejnych krokach procesu?
  - Wiadomość przesyłana jest niejawnie w ramach przepływu procesu
    - Dla zadań użytkownika:
      - Jednostka pracy dodawana jest do listy zadań
      - Użytkownik jest informowany o pojawieniu się zadania
      - Dostarczone zostają dokumenty procesu (wraz z załącznikami). Zakłada się, że dokumenty takie zostały utworzone w poprzednich krokach lub są dostarczone z poza procesu.

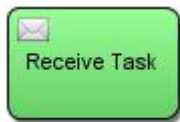
# Modelowanie komunikacji w granicach puli (procesu) (2)



- Informowanie uczestnika w ramach puli bez tworzenia jednostki pracy?
  - Nie ma dobrego rozwiązania w BPMN
    - Można utworzyć zadanie użytkownika z etykietą „Powiadomienie X” (gdzie X jest nazwą uczestnika procesu) np. „Powiadomienie kierownika”

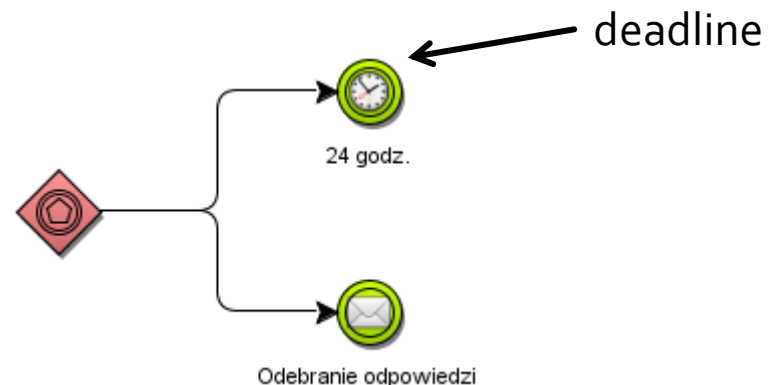
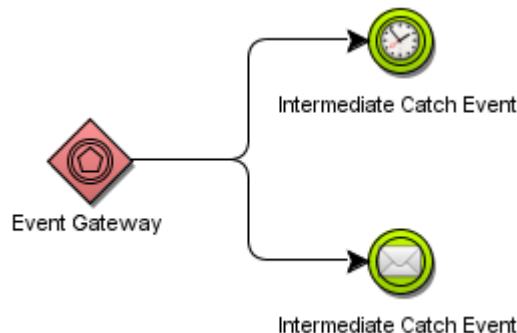
# Odbieranie wiadomości (1)

- Odbierający na poziomie opisowym:
  - Zadanie użytkownika, usługa, podproces, zdarzenie początkowe typu wiadomość
- Na poziomie analitycznym dodatkowo:
  - Zadanie typu odebranie Wiadomości, Przechwytyjące zdarzenie pośrednie typu Wiadomość



# Odbieranie wiadomości (2)

- Co jeżeli wiadomość nie nadejdzie?
  - Bramka sterowana zdarzeniami (ang. event based gateway)
    - Zawsze następują po niej zdarzenia (lub zadania) przechwytyjące lub zadania



# Zadania automatyczne



- Na poziomie opisowym
  - Zadanie typu usługa
- Na poziomie analitycznym
  - Zadanie typu
    - Skrypt
    - Usługa
    - Wywołanie asynchroniczne



# Skrypt (ang. Script task)

- Zadanie reprezentujące kod (skrypt) wykonywany przez silnik procesu.
  - Proste funkcje użytkowe
  - Na poziomie wykonywalnym (BPMSuite) możliwość rozszerzania funkcjonalności silnika



# Usługa (ang. Service task)



- BPMN zakłada, że zadanie wywołuje usługę
  - Zadanie wykonywane przez coś innego niż silnik procesu
  - Funkcja biznesowa wykonywana na poczet żądania, zazwyczaj
  - Zaleca się stosowanie dla usług synchronicznych
    - Zadanie się kończy w momencie wykonania usługi



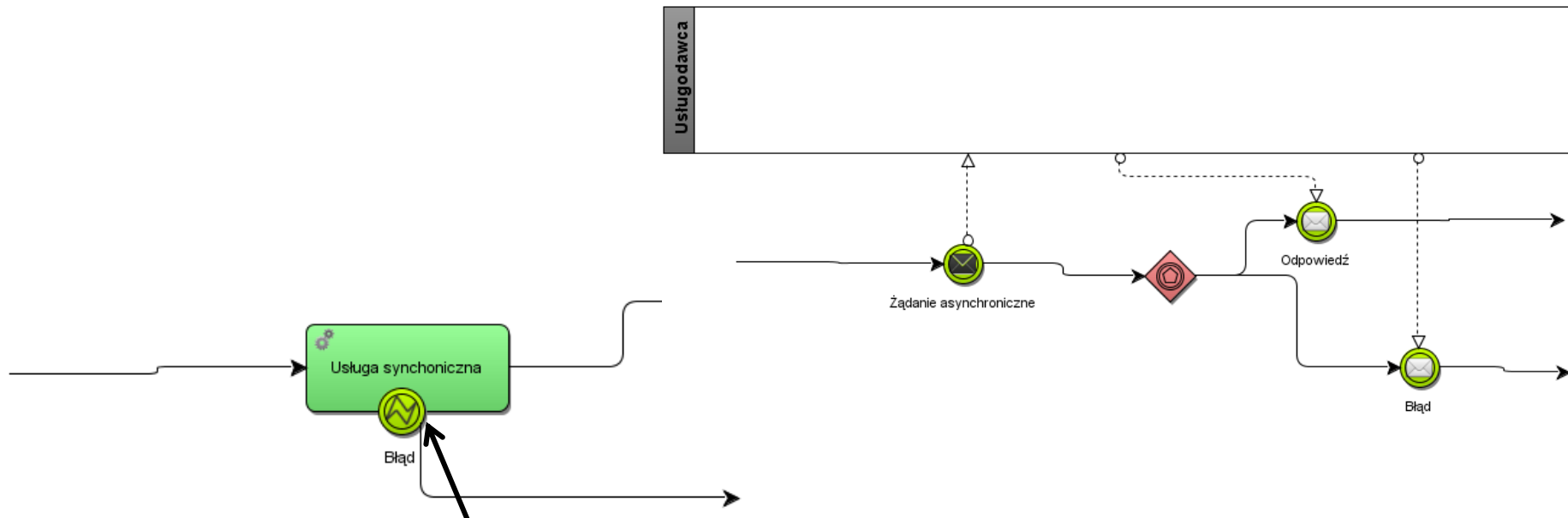
# Wywołanie asynchroniczne (1)



- Proces wysyła żądanie ale nie zatrzymuje się w oczekiwaniu na odpowiedź
  - W przypadku długotrwałych usług
  - Pobranie odpowiedzi może odbyć się później (lub w równoległej ścieżce)
- Jak modelować?
  - Usługodawca jako uczestnik zewnętrzny (ang. black-box pool) + zadanie wysłania/zdarzenie rzucenia typu wiadomość
  - Odpowiedź może być przechwycona w dalszej części procesu za pomocą symbolu zdarzenia przechwytyjacego

# Wywołanie asynchroniczne (1)

- Jeżeli proces nie ma nic do roboty poza czekaniem na odpowiedź długotrwałej usługi
  - Można modelować jako usługę synchroniczną (logicznie nie ma różnicy)



Przyłączone/krawędziowe zdarzenie pośrednie  
(ang. attached/boundary intermediate event - error)

# Decyzje i reguły biznesowe (1)



- Reguła biznesowa
  - Nazwany zbiór powiązanych warunków i rezultatów
  - Niezależna od procesu
  - Ogólnie rezultat reguły może być akcją, ale w BPMN częściej jest ustawieniem wartości (obiektu)
    - Akcja wymagana przez regułę biznesową modelowana jest jako czynność procesu realizowana pod warunkiem oczekiwanej wartości obiektu będącej rezultatem reguły.

# Decyzje i reguły biznesowe (2)



- W BPMN wyrażenia warunkowe na przepływach wychodzących z bramek są regułami „trasowania”
  - „Zakodowana” w ramach procesu
  - Bramka może odnosić się tylko do danych znanych procesowi

# Reguła biznesowa w BPMN (1)

- Zadanie typu zasada/reguła biznesowa (ang. business rule task)
  - Nowe w BPMN 2.0
  - Wywołanie zewnętrznego w stosunku do silnika procesu „silnika reguł biznesowych” czy też „usługi decyzyjnej”
- Zdarzenie typu warunek
  - Umożliwia modelowanie „silnika reguł biznesowych” w postaci zewnętrznego systemu
  - Imituje sytuację w której „silnik reguł biznesowych” monitoruje stan logiczny reguł i generuje zdarzenie w momencie zmiany



# BPMN

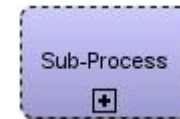
Podproces



# Zalety wykorzystania podprocesów



- Unifikacja całości procesu
  - Hierarchiczność
- Biznesowe usługi wielokrotnego użycia
  - Wykorzystanie zwiniętego podprocesu w wielu procesach.
- Segmentacja procesu ze względu na zarządzanie
  - Różne organizacje
- Definiowanie zakresu zdarzenia
  - Obszar w ramach którego zdarzenie jest wychwytywane
  - Tzw. Podproces zdarzeniowy (Podproces wyzwalany zdarzeniem) (ang. Event subprocess)



# Typy podprocesów (1)

- Podproces (ang. subprocess) i czynność wywoływana (ang. call activity)

- Podproces

- Podproces zagnieżdżony w procesie



- Czynność wywoływana

- Podproces zdefiniowany poza definicją procesu



# Typy podprocesów (2)

- Osadzony i rozwijany hierarchicznie
- Podproces ad-hoc
  - Może nie posiadać przepływu procesu (lista czynności)



# Typy podprocesów (3)

- Podproces transakcyjny
  - Transakcja biznesowa

