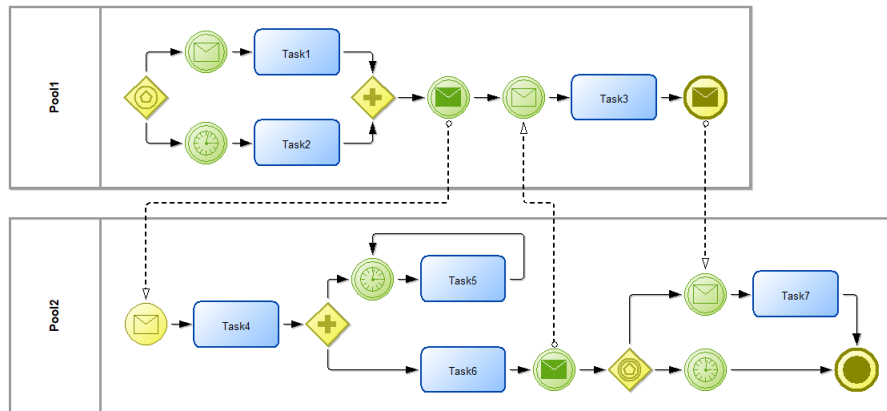


Modelowanie procesów biznesowych

Zdarzenia początkowe, końcowe i pośrednie Bramki oparte na zdarzeniach Podprocesy zdarzeniowe



Zdarzenie (*event*) reprezentuje oddziaływanie zewnętrzne lub wewnętrzne, które jest istotne dla przebiegu procesu i musi być uwzględnione w modelu w celu zapewnienia jego czytelności. Wystąpienie zdarzenia może być niezwiązane ze zwykłym przebiegiem procesu, a chwila w którym się pojawia może być nieprzewidywalna. Zdarzenia nie generują dodatkowych kosztów wykonania procesu. Typowym zastosowaniem zdarzeń jest zapewnienie komunikacji z innymi procesami lub częściami procesu oraz obsługa sytuacji nadzwyczajnych, które wymagają odejścia od zwykłego trybu działania.

Klasyfikacja zdarzeń ze względu na miejsce wystąpienia

- zdarzenia początkowe – określają sposób inicjalizowania nowego procesu (okrąg z pojedynczą, niepogrubioną krawędzią),
- zdarzenia końcowe – określają miejsce zakończenia procesu (okrąg z pojedynczą pogrubioną krawędzią),
- zdarzenia pośrednie – określają sytuacje występujące w czasie trwania procesu (okrąg z podwójną niepogrubioną krawędzią).

Klasyfikacja zdarzeń ze względu na sposób działania

- rzucające/nadawcze (*throw*) – zdarzenia, które generują informację o wystąpieniu określonego bodźca (generują pewien rezultat),
- chwytające/odbiorcze (*catch*) – zdarzenia, które oczekują na wystąpienie określonego bodźca (wstrzymują przepływ procesu do chwili jego wystąpienia).



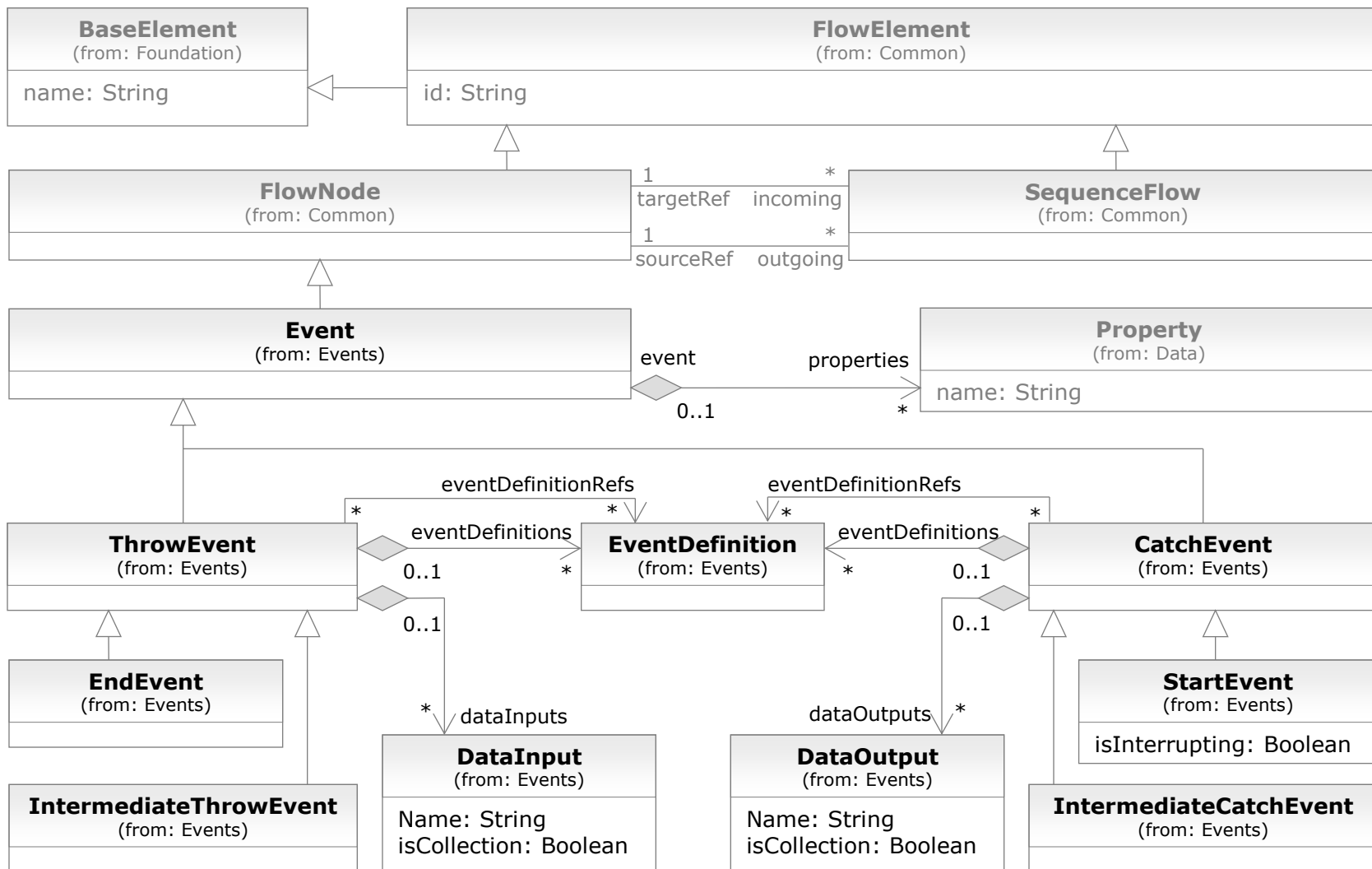
Wyzwalacz (*trigger*) określa rodzaj bodźca, który wywołał zdarzenie oraz jego kategorię.

Kategorie zdarzeń w notacji BPMN

- nieokreślone* (*none*) – nie ma określonego wyzwalacza,
- anulowanie (*cancel*) – przerwanie transakcji,
- błąd (*error*) – błąd wykonania czynności lub procesu,
- czas* (*timer*) – określona chwila czasowa,
- eskalacja (*escalation*) – wykonanie dodatkowych operacji wykraczających poza typowy przebieg procesu,
- kompensacja (*compensation*) – wycofanie skutków zakończonych operacji,
- komunikat* (*message*) – wymiana informacji z innym uczestnikiem,
- łącze* (*link*) – przerwanie przepływu sekwencyjnego i wznowienie w innym miejscu,
- sygnał (*signal*) – rozgłoszenie zawiadomienia w ramach całego modelu,
- warunek (*conditional*) – spełnienie wyrażenia warunkowego,
- zakończenie* (*terminate*) – zakończenie wszystkich wątków procesu,
- wielozdarzenie (*multiple*) – wystąpienie kilku bodźców,
- wielozdarzenie równoległe (*parallel mult.*) – jednoczesne wystąpienie kilku bodźców.

* Wyzwalacze dopuszczalne w modelu poglądowym (wykład01, s.14)

Zdarzenia – diagram klas



BPMN v.2.0.2, 10.5.Events, s.233

Klasa **Event**

- **properties**: Property [0..*] – lista własności zdarzenia (definiowana przez projektanta).

Klasy **ThrowEvent** i **CatchEvent**

- **eventDefinitionRefs**: EventDefinition [0..*] – odwołanie do współużywalnych definicji zdarzeń (zdefiniowanych jako elementy wyższego poziomu), określających wyzwalacze rzucone/chwyte przez zdarzenie; brak odwołania oznacza zdarzenie nieokreślone, wiele odwołań oznacza wielozdarzenie,
- **eventDefinitions**: EventDefinition [0..*] – odwołanie do definicji zdarzenia określającej wyzwalacz rzucony/chwyty przez zdarzenie; definiuje indywidualny typ zdarzenia używany wyłącznie dla konkretnego węzła,
- **dataInputs**: DataInput [0..*] – zestaw danych przekazywanych przez zdarzenie generujące (*throw*), dotyczy wybranych kategorii zdarzeń,
- **dataOutputs**: DataOutput [0..*] – zestaw danych odbieranych przez zdarzenie chwytające (*catch*), dotyczy wybranych kategorii zdarzeń.

Klasa **StartEvent**








- **isInterrupting**: boolean – określa czy zdarzenie początkowe jest przerywające (dotyczy zdarzeń początkowych podprocesów).

Zdarzenie początkowe (*start event*) określa sposób inicjowania procesu. Może przechwytywać wyzwalacze, nigdy ich nie generuje. Każde wystąpienie zdarzenia początkowego powoduje powstanie nowej instancji (egzemplarza) procesu i pociąga za sobą wygenerowanie nowego tokenu z kolejnym *TokenID*.

Cechy zdarzeń początkowych

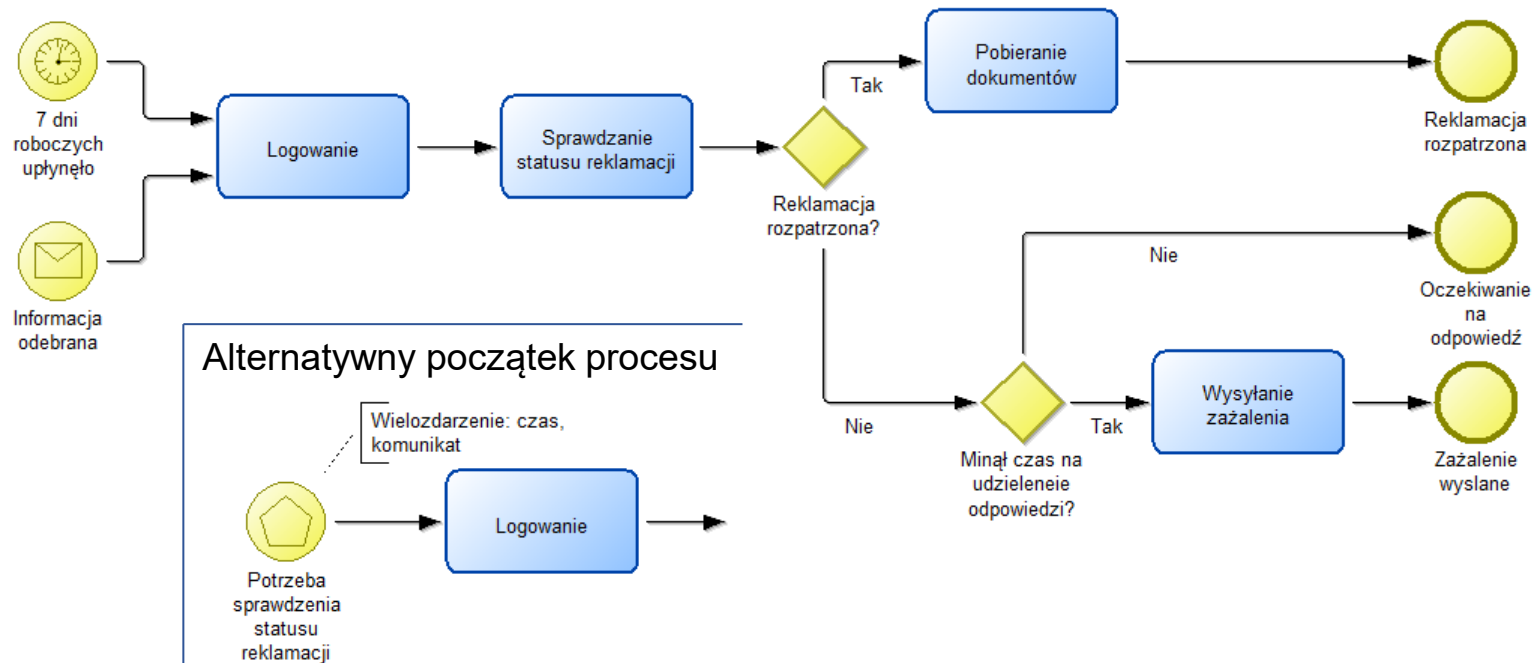
- okrąg z niepogrubioną krawędzią, ikona wyzwalacza biała (niewypełniona);
- brak wejściowych przepływów sekwencyjnych (dozwolony przepływ komunikatu);
- opcjonalne jeżeli warunki uruchomienia procesu są oczywiste;
- obowiązkowe jeżeli w modelu jawnie występuje zdarzenie końcowe;
- dozwolone kilka zdarzeń początkowych.

Kategorie zdarzeń początkowych

-  nieokreślone* – nie ma określonego wyzwalacza uruchamiającego proces;
-  czas* – proces uruchamiany w określonej chwili lub cyklicznie;
-  komunikat* – proces uruchamiany po przechwyceniu komunikatu (*message*);
-  sygnał – proces uruchamiany po przechwyceniu sygnału (*signal*);
-  warunek – proces uruchamiany po spełnieniu określonego warunku;
-  wielozdarzenie – proces uruchamiany przez jedno ze zbioru zdarzeń;
-  wielozdarzenie równoległe – proces uruchamiany po przechwyceniu wszystkich zdarzeń z określonego zbioru.



Sprawdzenie statusu reklamacji



Klient przystępuje do sprawdzenie statusu złożonej reklamacji gdy:

- Upłynął określony czas (wyzwalacz *timer*) – w tym przypadku 7 dni roboczych,
- Otrzymał informację o zakończeniu rozpatrywania wniosku (wyzwalacz *message*).










Zależnie od stwierdzonego statusu oraz czasu, który upłynął od chwili złożenia reklamacji pobiera dokumenty, wysyła zażalenie lub oczekuje na odpowiedź.

Zdarzenie końcowe (*end event*) określa sposób zakończenia procesu. Może generować wyzwalacze (rezultaty), nigdy ich nie przechwytuje.

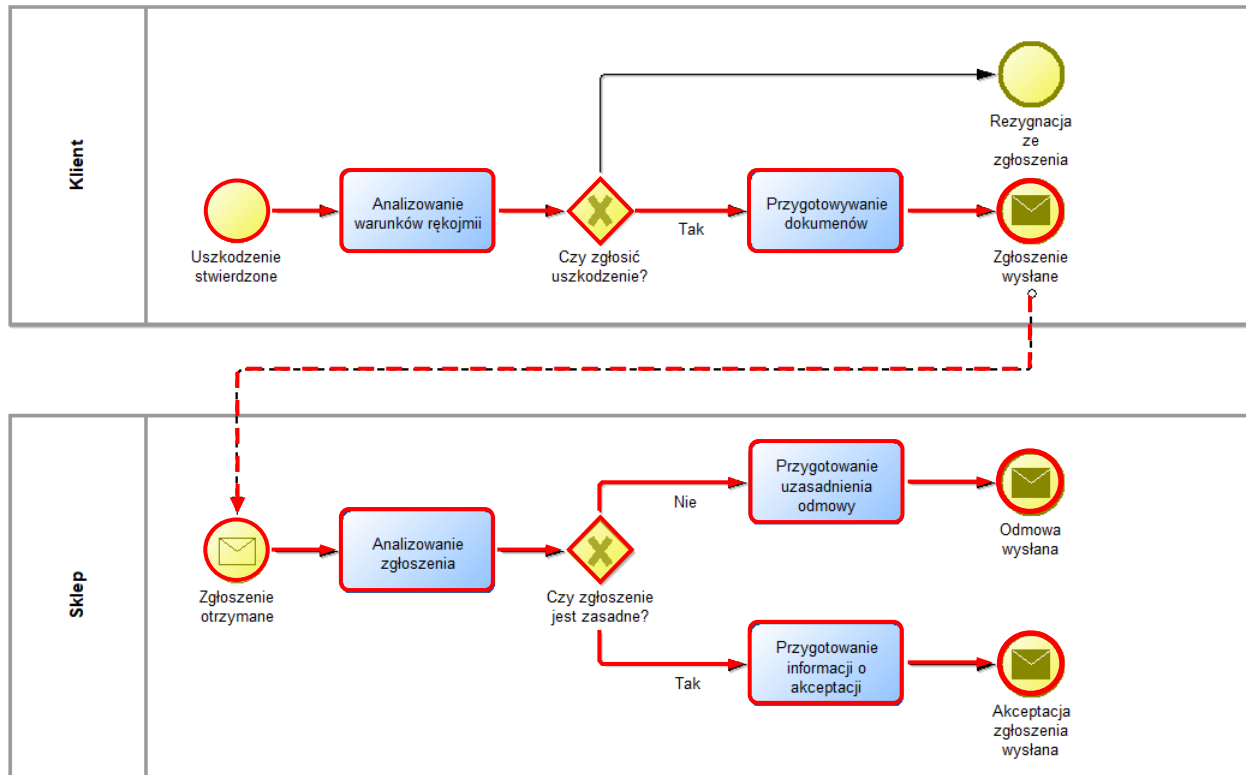
Cechy zdarzeń końcowych

- okrąg z pogrubioną krawędzią, ikona wyzwalacza czarna (wypełniona);
- brak wyjściowych przepływów sekwencyjnych (dozwolony przepływ komunikatu);
- opcjonalne jeżeli proces nie generuje wyzwalacza;
- obowiązkowe gdy występuje zdarzenie początkowe lub jest generowany wyzwalacz;
- dozwolone kilka zdarzeń końcowych.

Kategorie zdarzeń końcowych

-  nieokreślone* – proces nie generuje wyzwalacza;
-  anulowanie – przerwanie transakcji z wycofaniem wykonanych operacji;
-  błąd – proces zakończony z błędem, wymaga obsługi;
-  eskalacja – wymagane wykonanie dodatkowych operacji po zakończeniu;
-  kompensacja – usunięcie skutków czynności, wymaga dodatkowych operacji;
-  komunikat* – proces wysyła komunikat (*message*) do określonego uczestnika;
-  sygnał – rozgłoszenie sygnału (*signal*) do wszystkich uczestników;
-  zakończenie* – natychmiastowe przerwanie wszystkich wątków procesu;
-  wielozdarzenie – proces generuje kilka wyzwalaczy.



Zgłoszenie reklamacji z tytułu rękojmi



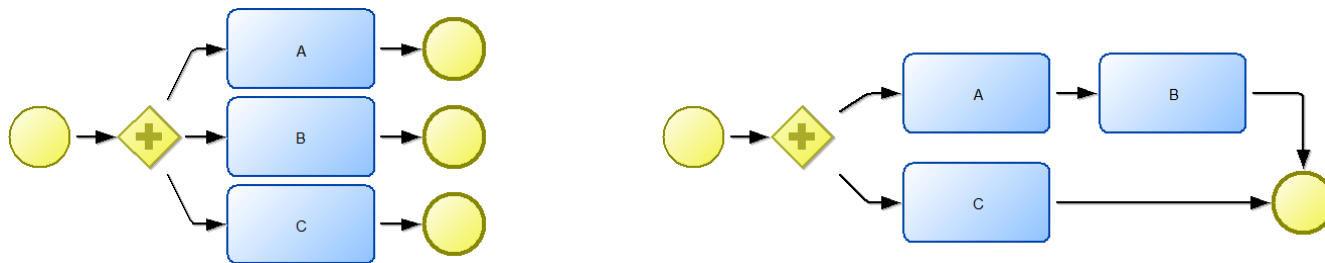
- *Klient* stwierdza uszkodzenie zakupionego produktu,
- Analizuje warunki rękojmi i podejmuje decyzję i wysyła zgłoszenie,
- Odebranie zgłoszenia uruchamia proces w *Sklepie*,
- Pracownik analizuje informacje i akceptuje lub odmawia przyjęcia zgłoszenia.

Zdarzenia końcowe – zakończenie procesu

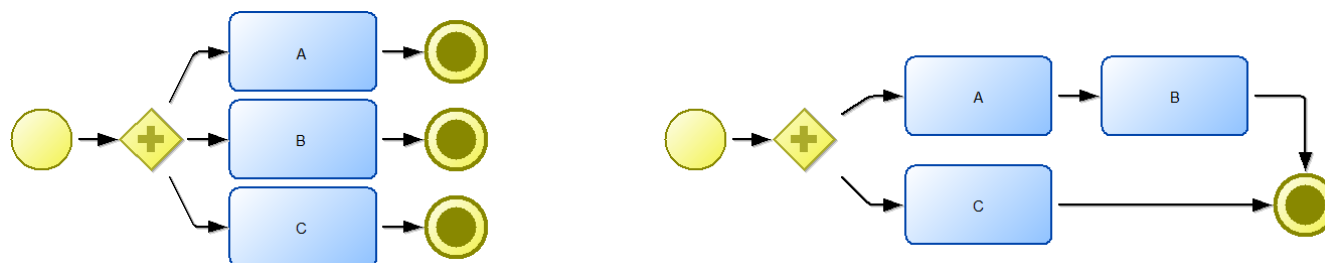
Kategorie zdarzeń końcowych w BPMN 2.0

-  zwykłe (nieokreślone) zdarzenie końcowe – kończy pojedynczy wątek procesu;
-  przerywające zdarzenie końcowe – kończy wszystkie wątki procesu.

Przykład 1. Procesy zakończą się, gdy tokeny wszystkich wątków dotrą do zdarzenia końcowego, stąd czynności A, B, C zawsze zostaną wykonane.



Przykład 2. Procesy zakończą się, gdy pierwszy subtoken dotrze do zdarzenia końcowego (zostanie zakończony jeden wątek), pozostałe czynności nie będą wykonane, aktualnie realizowane operacje zostaną przerwane.



Zdarzenie pośrednie (*intermediate event*) występuje pomiędzy zdarzeniem początkowym i końcowym. Może chwytać lub rzucać (generować) wyzwalacze. Wpływa na przebieg procesu, ale nie może uruchamiać lub kończyć instancji procesu.

Typy zdarzeń pośrednich

- zdarzenia pośrednie rzucające (*throwing*) – generują wyzwalacz (*throw trigger*) gdy token procesu dociera do zdarzenia.
- zdarzenia pośrednie chwytające (*catching*) – wstrzymują proces gdy token dociera do zdarzenia, wykonanie jest wznawiane po przechwyceniu wyzwalacza (*catch trigger*).
Uwaga: Wyzwalacze wygenerowane zanim token osiągnie zdarzenie są ignorowane.















Cechy zdarzeń pośrednich

- okrąg z podwójną niepogrubioną krawędzią;
- w przypadku zdarzeń rzucających ikona wyzwalacza czarna (wypełniona);
- w przypadku zdarzeń chwytających ikona wyzwalacza biała (niewypełniona);
- wymagany przepływ wejściowy i wyjściowy, dozwolony przepływ komunikatu.

Zastosowanie

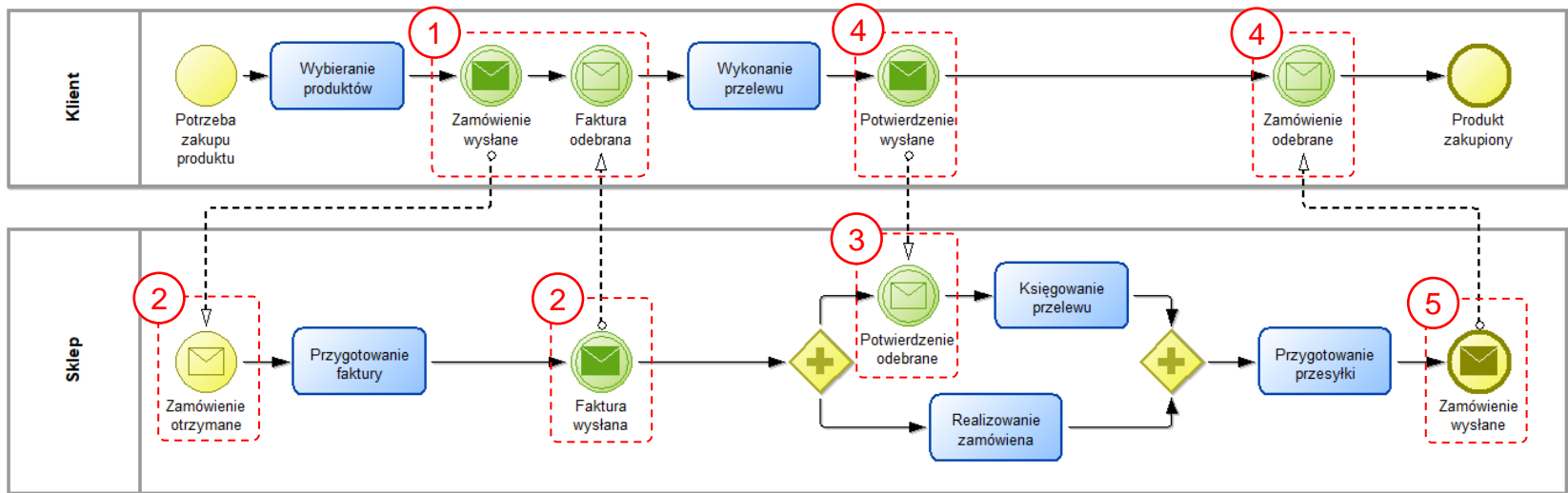
- prezentacja komunikatów i sygnałów, które muszą być odebrane/wysłane;
- określenie opóźnień czasowych procesu;
- obsługa sytuacji wyjątkowych, kompensacji i eskalacji podczas wykonania procesu.

Kategorie zdarzeń pośrednich

-  nieokreślone* – nie ma zdefiniowanego wyzwalacza, występuje tylko jako zdarzenie rzucające, używane do monitorowania procesu (kamienie milowe);
-  czas* – wstrzymuje proces na określony czas lub do pewnej chwili;
-  eskalacja – generuje wyzwalacz uruchamiający dodatkowe operacje;
-   komunikat* – generuje komunikat skierowany do określonego użytkownika (z.rzucające) lub oczekuje na nadejście komunikatu (z.chwytające);
-  kompensacja – generuje wyzwalacz uruchamiający operacje, których zadaniem jest usunięcie skutków zakończonej czynności;
-   łącze* – połączenie dwóch części procesu w jeden ciąg, eliminuje długie linie przepływów sekwencyjnych, ułatwia realizację pętli i skoków;
-   sygnał – generuje i rozgłasza sygnał (z.rzucające) lub oczekuje na pojawienie się sygnału (z.chwytające);
-  warunek – wstrzymuje proces do chwili spełnienia warunku (proces nie zatrzyma się jeżeli warunek będzie prawdziwy gdy token dotrze do zdarzenia);
-   wielozdarzenie – generuje kilka wyzwalaczy (z.rzucające) lub oczekuje na wystąpienie jednego z kilku wyzwalaczy (z.chwytające);
-  wielozdarzenie równoległe – oczekuje na wystąpienie wszystkich zdefiniowanych wyzwalaczy.

Zdarzenia pośrednie – komunikat

Realizacja zamówienia (modyfikacja modelu z W07, s.9 – wykorzystanie zdarzeń)

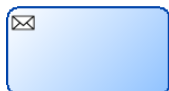


1. Klient przygotowuje zamówienie, wysyła je do Sklepu i oczekuje na fakturę.
2. Przepływ komunikatu uruchamia proces Sklepu, który przyjmuje zamówienie, przygotowuje fakturę i przesyła ją do Klienta.
3. Po przesłaniu faktury Sklep rozpoczyna realizację zamówienia i oczekuje na wpłatę.
4. Po odebraniu faktury Klient wykonuje przelew i wysyła potwierdzenie, a następnie oczekuje na zamówione produkty.
5. Po zakończeniu realizacji, odebraniu przelewu i zaksięgowaniu wpłaty Sklep wysyła zamówione produkty, co kończy jego proces
6. Odbiór zamówienia kończy proces Klienta.

Zdarzenie komunikat vs. zadania wysyłki i odbioru



Zadanie wysyłki (*send task*) wysyła komunikat do innego uczestnika, jest zakończone po wysłaniu komunikatu.



Zadanie odbioru (*receive task*) oczekuje i odbiera komunikat; jest zakończone po odebraniu komunikatu.



Rzucające zdarzenie komunikat (*throwing*) generuje komunikat skierowany do określonego użytkownika.



Chwytające zdarzenie komunikat (*catching*) oczekuje na nadejście komunikatu, proces jest kontynuowany po odebraniu komunikatu.

Norma BPMN 2.0 nie określa wprost różnic pomiędzy zadaniami wysyłki/odbioru a zdarzeniem komunikat. Zadanie wysyłki jest odpowiednikiem zdarzenia rzucającego, a zadanie odbioru zdarzenia chwytającego wyzwalacz. Różnica wynika z charakteru obiektu: zadania są szczególnym przypadkiem aktywności (reprezentują pracę) i ich wykonanie generuje określony koszt, wykonanie zdarzenia nie generuje dodatkowych kosztów realizacji danego etapu procesu. Dodatkowo: aktywności pozwalają na obsługę sytuacji wyjątkowych (błędów), które mogą wystąpić podczas ich wykonania (*patrz zdarzenia krawędziowe, wykład 08*).



Bramki oparte na zdarzeniach reprezentują punkt rozgałęzienia procesu, w którym wykonanie alternatywnej ścieżki przepływu zależy od wystąpienia pewnego zdarzenia, a nie warunków związanych z danymi procesowymi. Wybór ścieżki może zależeć np. od komunikatu odebranego od innego uczestnika, więc decyzja może być podejmowana w oparciu o dane, które są dla modelowanego procesu niewidoczne.

Przepływy wyjściowe bramek opartych na zdarzeniach nie mogą mieć związanych warunków, muszą prowadzić do zdarzeń pośrednich lub zadania odbioru. Element diagramu umieszczony na przepływie wyjściowym jest częścią konfiguracji bramki i stanowi wyzwalacz, który uruchamia dany przepływ.

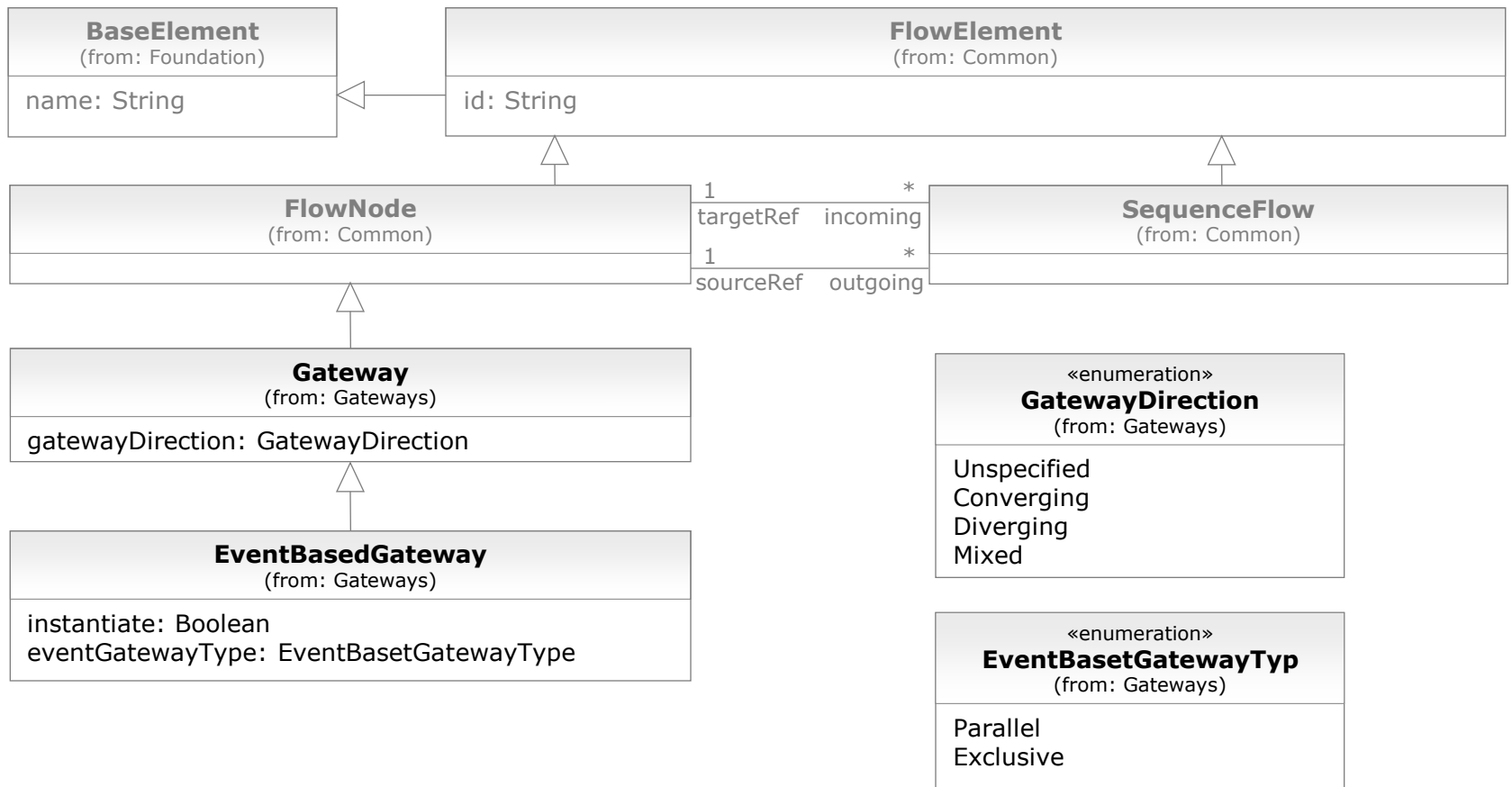
Typy bramek opartych na zdarzeniach

- oparte na zdarzeniach początkowych – rozpoczynają proces (bramka nie ma przepływu wejściowego) który zależy od kilku zdarzeń początkowych;
- oparte na zdarzeniach pośrednich – wstrzymują wykonanie procesu do chwili wystąpienia jednego lub kilku zdarzeń.

Elementy dozwolone na przepływach wyjściowych

- zdarzenia pośrednie: komunikat, sygnał, timer, zdarzenie warunkowe, wielozdarzenie złożone z wymienionych zdarzeń;
- zadanie odbioru.

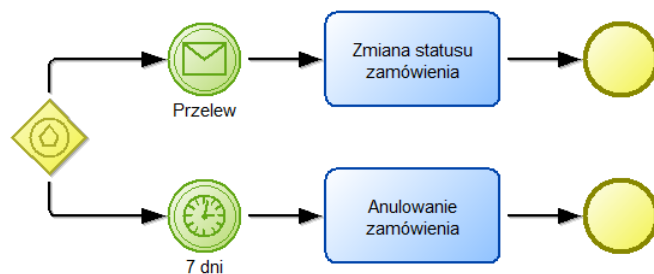
Bramki oparte na zdarzeniach – diagram klas



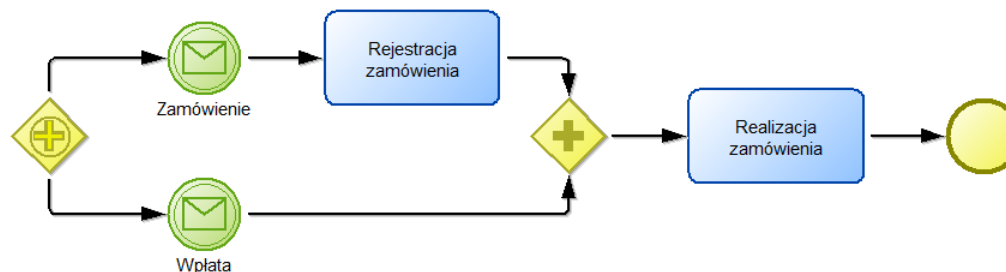
Bramki oparte na zdarzeniach początkowych



Wykluczająca bramka oparta na zdarzeniach początkowych rozpoczyna proces gdy wystąpi dowolne spośród wymienionych zdarzeń lub zostanie wykonane zadanie odbioru. Pozostałe zdarzenia występujące po rozpoczęciu procesu będą zignorowane.



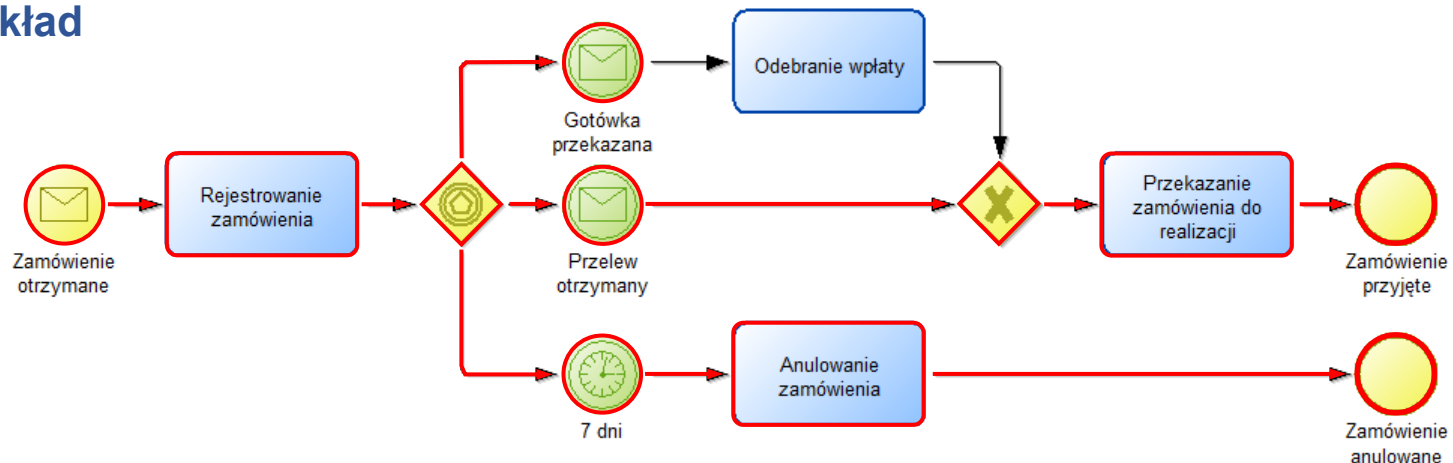
Równoległa bramka oparta na zdarzeniach początkowych rozpoczyna proces gdy wystąpi dowolne spośród wymienionych zdarzeń lub zostanie wykonane zadanie odbioru. Do poprawnego zakończenia rozpoczętego procesu wymagane jest wystąpienie wszystkich pozostałych zdarzeń.



Bramka oparta na zdarzeniach pośrednich

Wykluczająca bramka oparta na zdarzeniach pośrednich występuje zawsze przed zdarzeniami, których dotyczy. Gdy token dociera do takiej bramki aktywowane są wszystkie przepływy wychodzące i proces zostaje wstrzymany w oczekiwaniu na wystąpienie zdarzenia związanego z jednym z przepływów. Token wybiera ścieżkę prowadzącą do zdarzenia, które wystąpiło jako pierwsze, pozostałe są dezaktywowane. Jest to jedyny typ bramki opartej na zdarzeniach pośrednich w BPMN 2.0.

Przykład



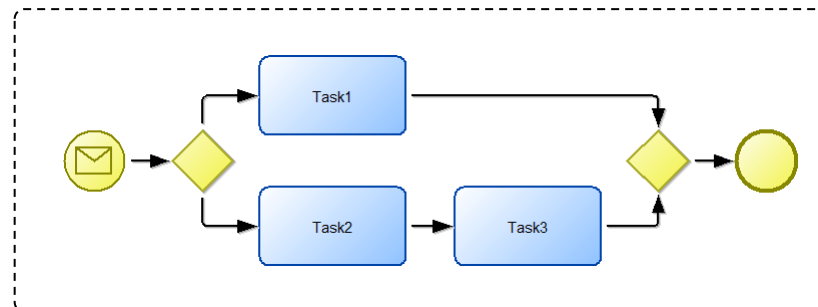
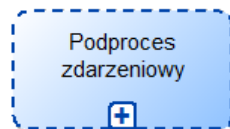
Po otrzymaniu zamówienia sklep oczekuje na wpłatę zanim rozpocznie jego realizację (możliwe dwie formy płatności: gotówka lub przelew). Jeżeli wpłata nastąpi przed upływem 7 dni jest księgowana i zamówienie przekazywane jest do realizacji. Jeżeli zamówienie nie będzie opłacone w terminie 7 dni zostaje anulowane.



Podproces zdarzeniowy jest szczególnym przypadkiem podprocesu, który nie jest uruchamiany przez proces nadrzędny tylko przez określony wyzwalacz zdarzenia. Jest używany wewnątrz pewnego procesu nadrzędnego (procesu głównego lub innego podprocesu), jednak nie występuje w jego sekwencji przepływu.

Cechy podprocesów zdarzeniowych

- symbol jak podproces osadzony z przerywaną krawędzią (zarówno w wersji zwiniętej jak i rozwiniętej);
- ma dokładnie jedno zdarzenie początkowe z określonym wyzwalaczem, dozwolone: błąd, czas, eskalacja, kompensacja, komunikat, sygnał, warunek, wielozdarzenie, wielozdarzenie równoległe;
- nie ma przepływu wejściowego i wyjściowego (znajduje się poza sekwencją przepływu procesu nadrzędnego);
- może być uruchomiony tylko gdy proces nadrzędny jest aktywny, może nie być wykonany ani razu lub wielokrotnie (po każdym przechwyceniu wyzwalacza).














Typy podprocesów zdarzeniowych w BPMN 2.0

- Przerwywające (*interrupting*) – po przechwyceniu wyzwalacza przerywają proces nadrzędny i wykonują własny przepływ sekwencyjny;
- Nieprzerwywające (*non-interrupting*) – po przechwyceniu wyzwalacza wykonują własny przepływ sekwencyjny równoległe z procesem nadrzędnym, ich zakończenie nie kończy procesu nadrzędnego.

Typ procesu określa zdarzenie początkowe: okrąg z ciągłą krawędzią dla procesów przerywających lub okrąg z przerywaną krawędzią dla procesów nieprzerwywających.

Kategorie zdarzeń początkowych dozwolone dla podprocesów zdarzeniowych

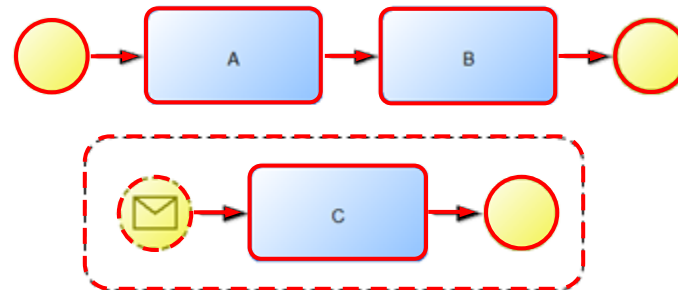
-  błąd (przerwywający) – obsługa błędu zgłoszonego przez proces nadrzędny;
-  czas – podproces uruchamiany cyklicznie lub w określonej chwili;
-  eskalacja – wykonanie dodatkowych aktywności (nietypowych/wyjątkowych);
-  kompensacja (przerwywający) – wycofanie efektów procesu nadrzędnego;
-  komunikat – podproces uruchamiany po przechwyceniu komunikatu;
-  sygnał – podproces uruchamiany po przechwyceniu sygnału;
-  warunek – podproces uruchamiany po spełnieniu określonego warunku;
-  wielozdarzenie – podproces uruchamiany przez jedno ze zdarzeń;
-  wielozdarzenie równoległe – podproces uruchamiany po wyst. kilku zdarzeń.

Nieprzerywające podprocesy zdarzeniowe



Nieprzerywające podprocesy zdarzeniowe są aktywowane po wystąpieniu odpowiedniego wyzwalacza. Ich uruchomienie nie przerywa aktualnie wykonywanych czynności procesu głównego, więc mogą być uruchomione kilka razy w ramach danej instancji. Instancja procesu jest zakończona po zakończeniu wszystkich wątków procesu głównego i podprocesów zdarzeniowych

Przykład



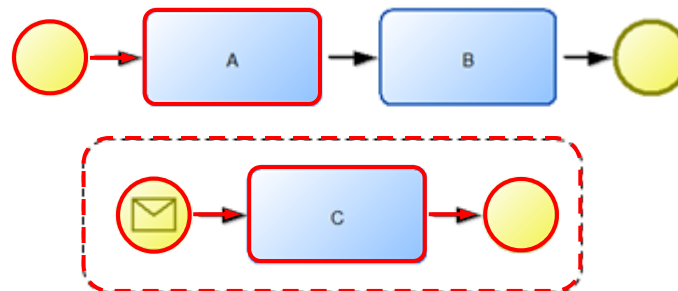
W ramach procesu głównego są realizowane aktywności A i B, nieprzerywający podproces zdarzeniowy oczekuje na wyzwalacz typu komunikat. Każdorazowe przechwycenie wyzwalacza powoduje wykonuje aktywność C, ale nie przerywa procesu głównego. Podproces może być uruchomiony kilka razy.

Przerywające podprocesy zdarzeniowe



Przerywające podprocesy zdarzeniowe są aktywowane po wystąpieniu odpowiedniego wyzwalacza. Ich uruchomienie przerywa aktualnie wykonywane czynności procesu głównego, więc mogą być uruchomione tylko raz w ramach danej instancji. Instancja procesu jest zakończona po zakończeniu podprocesu zdarzeniowego.

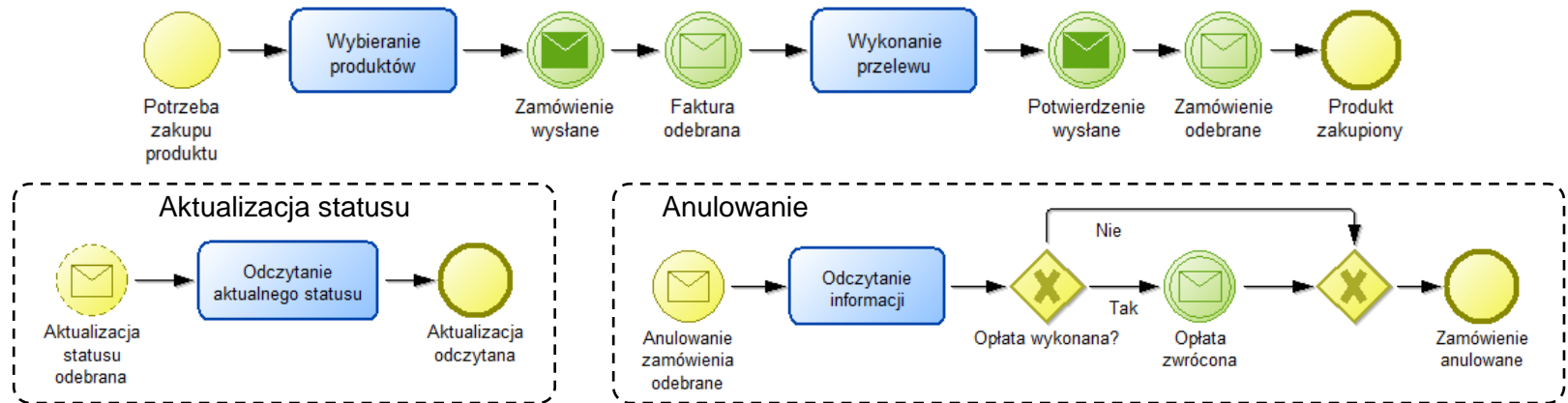
Przykład



W ramach procesu głównego są realizowane aktywności A i B, przerywający podproces zdarzeniowy oczekuje na wyzwalacz typu komunikat. Przechwycenie wyzwalacza przerywa proces główny i powoduje wykonanie aktywności C.



Realizacja zamówienia (proces klienta, s.13)



Podczas przebiegu procesu w dowolnej chwili mogą wystąpić dwa typy komunikatów:

- Zmiana statusu zamówienia – może wystąpić wielokrotnie (księgowanie opłaty, rozpoczęcie realizacji, wysyłka, itp.), nie wpływa na przebieg procesu głównego.
- Anulowanie zamówienia – sytuacja wyjątkowa (np. brak zamówionego produktu), jej wystąpienie przerywa działanie procesu głównego.

Zalety rozwiązania z wykorzystaniem procesów zdarzeniowych

- Nie komplikuje struktury zasadniczego procesu (brak dodatkowych bramek),
- Pozwala na obsługę zdarzenia niezależnie od chwili jego wystąpienia (zdarzenia będzie przechwycone niezależnie od stanu procesu głównego).

Wzorce procesowe wykorzystujące zdarzenia

Wzorce procesowe omówione na kolejnych slajdach

- ❑ WP11. Niejawne zakończenie (*implicit termination*)
- ❑ WP43. Jawne zakończenie (*explicit termination*)
- ❑ WP19. Anulowanie czynności (*cancel activity*)
- ❑ WP16. Odroczoney wybór (*differed choice*)
- ❑ WP23. Przemijający wyzwalacz (*transient trigger*)
- ❑ WP24. Trwały wyzwalacz (*persistent trigger*)

Uwaga: część wzorców ma kilka realizacji w BPMN, przykłady pokazane na kolejnych stronach ograniczają się jedynie do zastosowania zdarzeń, bramek opartych na zdarzeniach oraz procesów zdarzeniowych.

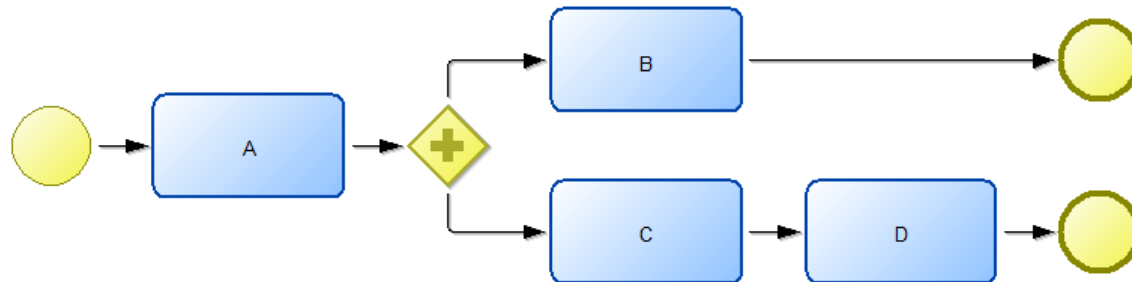
WP11. Niejawne zakończenie (*implicit termination*)

Opis: Proces zostaje zakończony gdy zostały zrealizowane wszystkie zadania.

Komentarz: Wzorzec opisuje przypadek, w którym wszystkie wątki procesu są wykonywane do chwili, gdy zrealizowały wszystkie przewidziane zadania. Każdy z nich może zakończyć się w innym terminie, cały proces zostaje zakończony po zakończeniu wszystkich wątków.

Przykłady: po wpłynięciu zlecenia na wykonanie kilku produktów dział produkcji rozpoczyna jego realizację, zlecenie zostanie zakończone gdy wszystkie produkty będą wykonane.

Element BPMN: zdarzenie końcowe bez wyzwalacza.



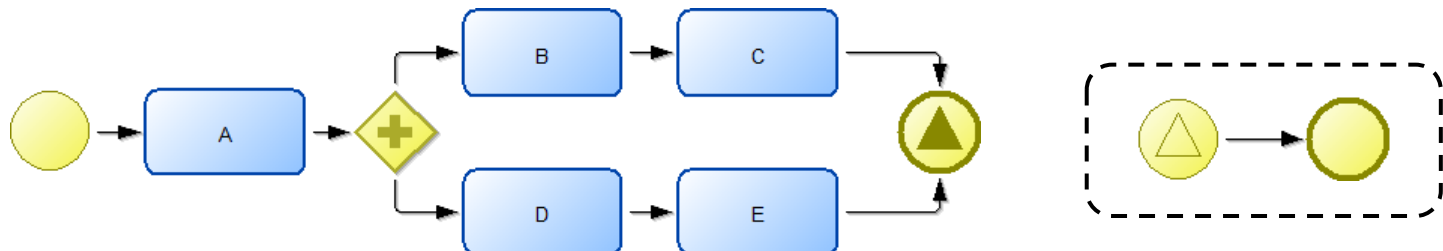
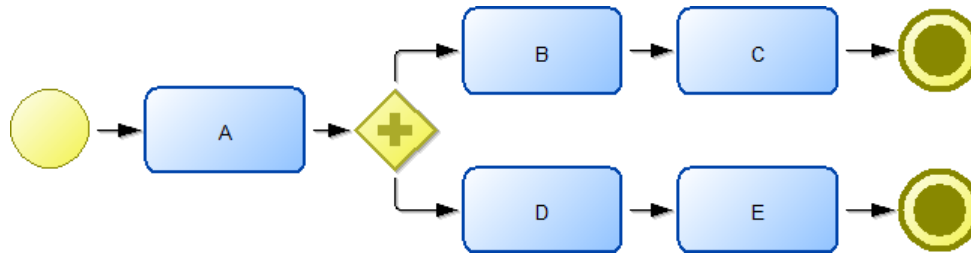
WP43. Jawne zakończenie (*explicit termination*)

Opis: Proces zostaje zakończony gdy osiągnie określony stan.

Komentarz: Wzorzec opisuje przypadek, w którym instancja procesu powinna zostać przerwana gdy został osiągnięty określony węzeł przepływu. Wszystkie pozostałe do wykonania zadania zostają przerwane, proces jest oznaczony jako zakończony prawidłowo.

Przykłady: dwa zespoły pracują nad wykonaniem pewnego zadania, cel zostaje osiągnięty (proces jest zakończony) gdy jeden z zespołów ukończy zadanie.

Element BPMN: zdarzenie końcowe typu zakończenie (*terminate*), przerywający podproces zdarzeniowy.



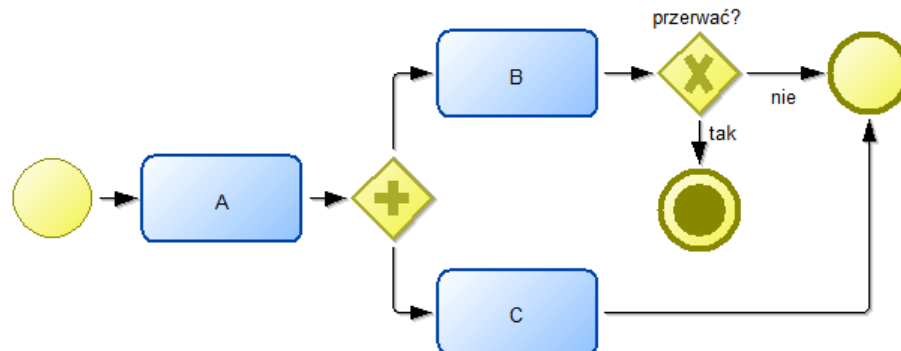
WP19. Anulowanie czynności (*cancel activity*)

Opis: Zadanie, jeśli to możliwe, jest wycofywane (zatrzymywane i usuwane).

Komentarz: Wzorzec opisuje przypadek, w którym instancja procesu powinna zostać przerwana gdy został osiągnięty określony węzeł przepływu. Wszystkie pozostałe do wykonania zadania zostają przerwane, proces jest oznaczony jako zakończony prawidłowo.

Przykłady: dwóch pracowników realizuje pewne zadania, które są częścią większego projektu, w pewnych przypadkach projekt jest akceptowany po zakończeniu zadania pierwszego pracownika i zadanie drugiego jest anulowane.

Element BPMN: zdarzenie końcowe typu zakończenie (*terminate*).



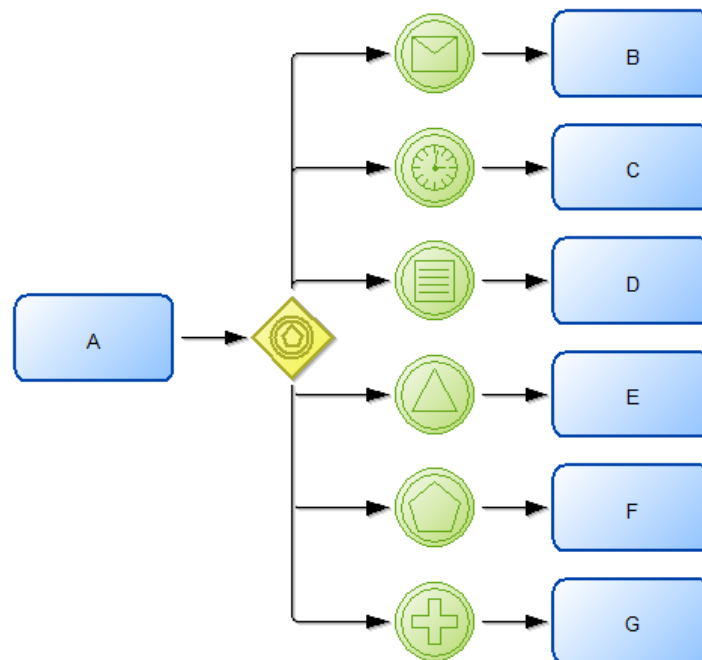
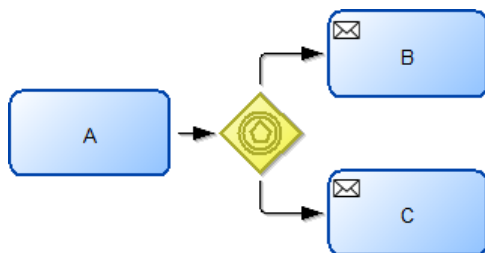
WP16. Odroczoney wybór (*differed choice*)

Opis: Wybór jednego z przepływów alternatywnych na podstawie interakcji ze środowiskiem operacyjnym.

Komentarz: Wybór przepływu odbywa się na podstawie stanu procesu, a nie danych, które są przetwarzane. Aktywowany jest przepływ związany z wyzwalaczem, który pojawił się jako pierwszy, pozostałe przepływy są wycofane.

Przykłady: sklep oczekuje na potwierdzenie wpłaty, jeżeli informacja dotrze w ciągu pięciu dni zamówienie jest wysyłane, gdy minie ustalony czas zostanie wycofane.

Element BPMN: bramka oparta na zdarzeniach pośrednich.



WP23. Przemijający wyzwalacz (*transient trigger*)

Opis: Wykonanie zadania po odebraniu sygnału zewnętrznego pochodzącego z innej części procesu lub spoza niego.

Komentarz: Wyzwalacz ma charakter przemijający co oznacza, że jest tracony jeśli nie czeka na niego zdarzenie/zadanie chwytające (wyzwalacze wygenerowane przed dotarciem tokenu do zdarzenia/zadania są ignorowane).

Przykłady: podczas logowania, po wprowadzeniu nazwy użytkownika i hasła użytkownik czeka na kod jednorazowy przesyłany na jego komórkę/e-mail, po odebraniu kontynuuje proces logowania.

Element BPMN: chwytające zdarzenia pośrednie sygnał i komunikat.

Wykonanie zadania po odebraniu sygnału z innej części procesu



Wykonanie zadania po odebraniu sygnału zewnętrznego spoza procesu



WP24. Trwały wyzwalacz (*persistent trigger*)

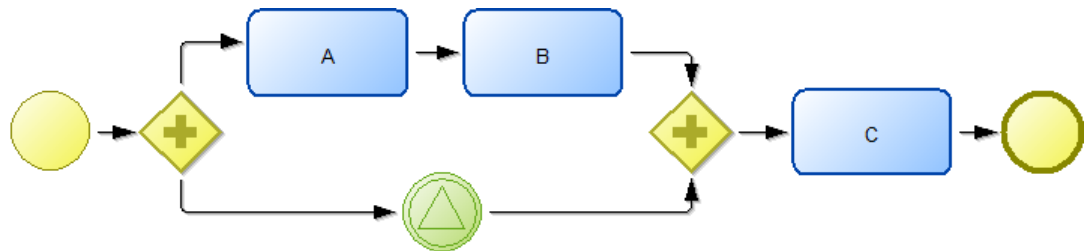
Opis: Wykonanie zadania po odebraniu sygnału zewnętrznego pochodzącego z innej części procesu lub spoza niego.

Komentarz: Wyzwalacz ma charakter trwały co oznacza, że proces może go zarejestrować i przechować informację o jego wystąpieniu. Wzorec pozwala na sprawdzenie czy wyzwalacz pojawił się podczas realizacji pewnego etapu procesu.

Przykłady: realizowane zamówienie nie może być wysłane dopóki nie zostanie opłacone, informacja o opłacie może pojawić się podczas realizacji innych operacji (nieprzewidywalna chwila wystąpienia) i powinna zostać odebrana oraz zarejestrowana.

Element BPMN: chwytające zdarzenia pośrednie sygnał i komunikat.

Wykonanie zadania po odebraniu sygnału z innej części procesu



Wykonanie zadania po odebraniu sygnału zewnętrznego spoza procesu

