

Modelowanie procesów biznesowych

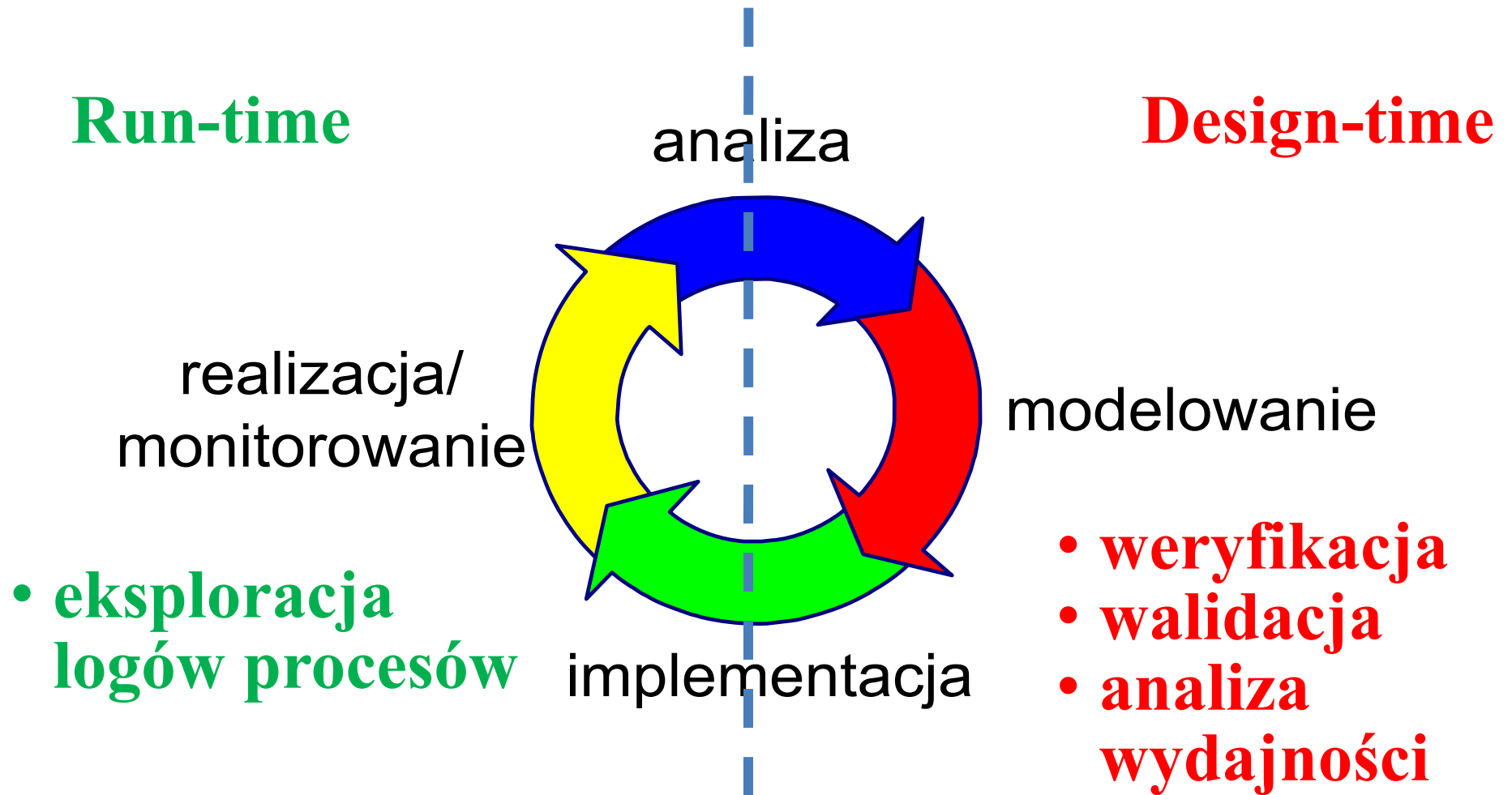
Standard BPMN

- Specyfikacja BPMN 1.2 3-01-2009 (OMG, BPMI)
- Specyfikacja BPMN 2.0 3-01-2011 (OMG)
- *Notacja modelowania procesów biznesowych – podstawy*, Marek Piotrowski, Wydawnictwo BTC

Standard UML – Activity Diagram

Notacja Event-Driven Process Chain

Cykl życia modelu procesów



Cele modelowania procesów

- **Walidacja** służy do ustalenia zgodności modelu z rzeczywistością
- **Weryfikacja** służy do ustalenia poprawności modelu
 - Czy w przepływie pracy mogą wystąpić zakleszczenia?
 - Czy działa poprawnie w konkretnych przypadkach?
 - Czy dla wszystkich przypadków wystąpienia procesów zostaną zakończone?
 - Czy jest możliwa zmiana kolejności wykonywania zadań?
- **Analiza wydajności** służy do ilościowego ustalenia jakości modelu
 - Jak wiele wystąpień procesu można zrealizować w jednostce czasu?
 - Jaki jest średni czas wykonania wystąpienia procesu?
 - Ile zasobów jest potrzebnych do wykonania wystąpienia procesu?
- **Implementacja** - budowa systemu informatycznego, który będzie realizował, zarządzał i monitorował przebieg procesów

Cele implementacji systemu zarządzania procesami

- **Monitorowanie procesów** dla umożliwienia ingerencji w przebieg wystąpień procesów np. w wypadku ich zakleszczenia, opóźnienia, itp.
- **Wymuszanie prawidłowego przebiegu procesów**
- **Rejestracja wystąpień procesów** w celu umożliwienia wykonywania analiz faktycznego przebiegu procesów
- Umożliwienie **kooperacji wykonawców** w ramach procesów typu ad-hoc
- Częściowa lub całkowita **automatyzacja procesów**

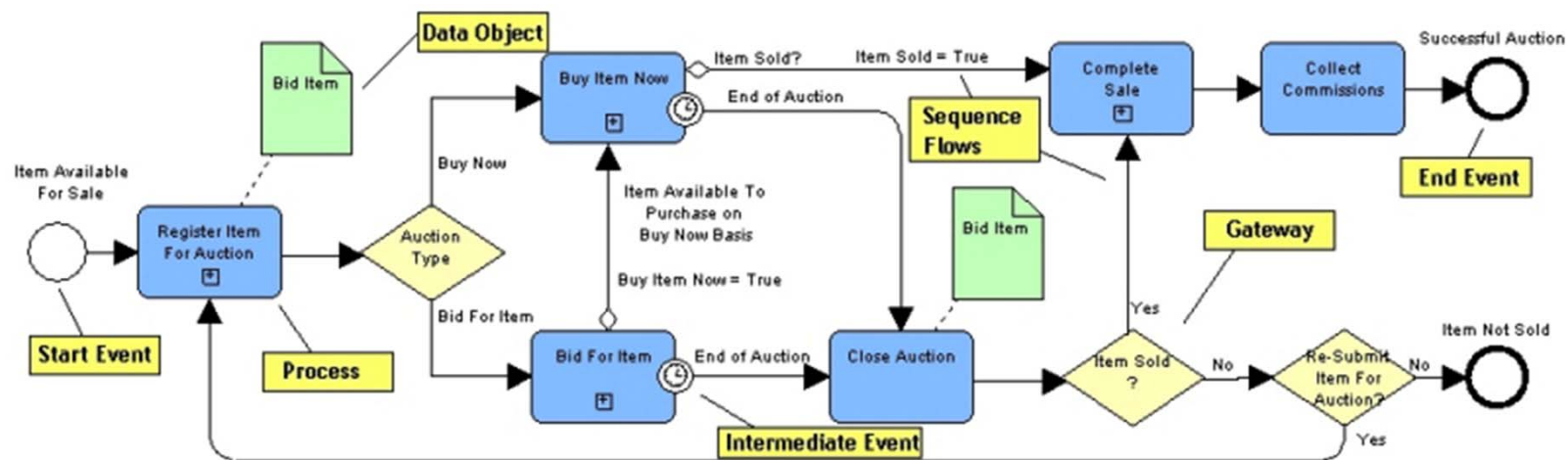
Narzędzia do modelowania, analizy i zarządzania wykonaniem procesów

Nazwa	Walidacja	Weryfikacja	Analiza ilościowa	Wykonywanie
UML	✓			
EPC Event-Driven Process Chain	✓			
BPMN - Business Process Modeling Notation	✓	nieformalna		✓
BPEL4WS - Business Process Execution Language for Web Services	?			✓
Sieci Petriego	✓	✓	✓	
YAWL - Yet Another Workflow Language	✓	✓	✓	

Business Process Modeling Notation

BPMN jest graficznym narzędziem dedykowanym do modelowania procesów biznesowych. Standard BPMN został opracowany przez: *Business Process Management Initiative - BPMI*, która włączyła się z *Object Management Group* OMG.

Aktualna wersja standardu to BPMN 2.0 ze stycznia 2011



Podstawy BPMN

Cztery podstawowe typy konstruktorów modelu

Elementy aktywne procesów

- Zdarzenia
- Działania

Co

Wykonawcy zadań

- Jednostki organizacyjne (baseny)
- Jednostki składowe (tory)

Kto

Przepływy

- Bramki
- Przepływ pracy
- Przepływ komunikatu
- Powiązanie

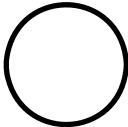

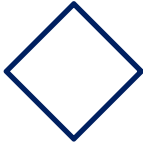
Jak

Artefakty

- Dane
- Grupy
- Adnotacje

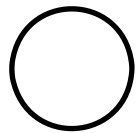
**Objaśnienia
pomocnicze**

Elementy aktywne

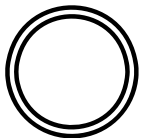
Konstruktor	Opis	Notacja
Zdarzenie	Zdarzenia występujące w trakcie trwania procesów, spowodowane lub mające wpływ na procesy. Wyróżnia się trzy typy zdarzeń: początkowe, końcowe i pośrednie.	
Działanie	Jednostka pracy składającej się na proces biznesowy. Może to być podproces lub elementarne zadanie. Dodatkowe elementy graficzne pozwalają opisać dodatkowe cechy działań; np. złożone, iteracyjne, transakcyjne, itp.	
Bramka	Element rozgałęziania i łączenia przepływów. Symbole wewnątrz elementu graficznego służą do rozróżnienia typu bramki.	

Typy zdarzeń

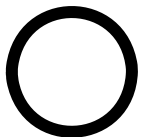
Wyróżnia się trzy podstawowe typy zdarzeń



- Zdarzenie **początkowe** rozpoczyna proces



- Zdarzenie **pośrednie** występuje w trakcie trwania procesu





























- Zdarzenie **końcowe** kończy dany proces

Typy zdarzeń 1.2

Przyczyny występowania zdarzeń określają dziesięć predefiniowanych typów zdarzeń. Dodatkowo rozróżnia się zdarzenia, które są przyczynami albo skutkami działań:

- catching
- throwing

	"Catching"		"Throwing"	
Message				
Timer				
Error				
Cancel				
Compensation				
Conditional				
Link				
Signal				
Terminate				
Multiple				

Typy zdarzeń 2.0

Przyczyny występowania zdarzeń określają dziesięć predefiniowanych typów zdarzeń.











Dodatkowo rozróżnia się zdarzenia, które są przyczynami albo skutkami działań:

- throwing
- catching

oraz przerywających i nie przerywających bieżących wątków procesów.

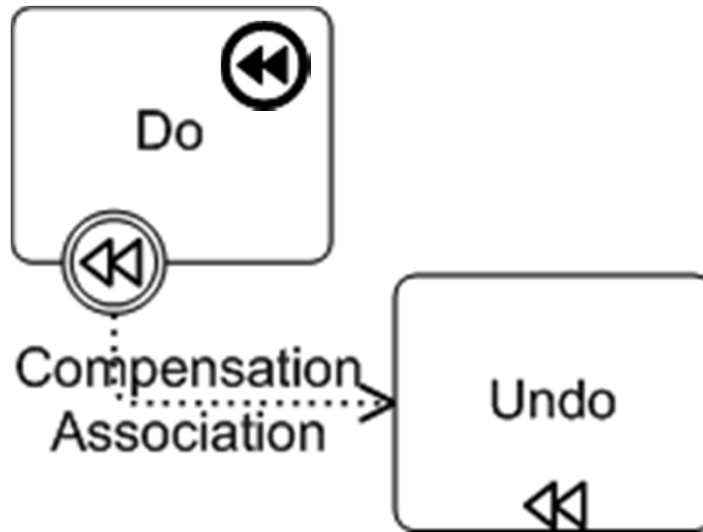
	"Catching"		"Throwing"		Non-Interrupting	
Message						
Timer						
Error						
Escalation						
Cancel						
Compensation						
Conditional						
Link						
Signal						
Terminate						
Multiple						
Parallel Multiple						

Typy zdarzeń

Message	Odebranie/wysyłanie wiadomości	
Timer	Wystąpienie określonej daty/godziny lub upływ czasu	
Error	Odebranie/zgłoszenie błędu	
Cancel	Wykonanie/zgłoszenie żądania wycofania wyników działania procesów transakcyjnych	
Compensation	Wykonanie/zgłoszenie żądania wykonania kompensacji	
Conditional	Spełnienie zdefiniowanego warunku	
Link	Połączenie dwóch części procesu	
Signal	Odebranie/wysyłanie sygnału	
Multiple	Zdarzenia, z którym jest skojarzonych wiele przyczyn	
Terminate	Koniec wykonywania procesu, bez kompensacji i obsługi błędów	

Zdarzenia na krawędzi procesów

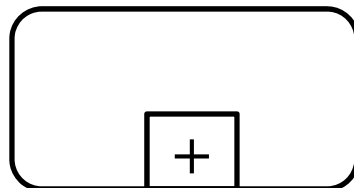
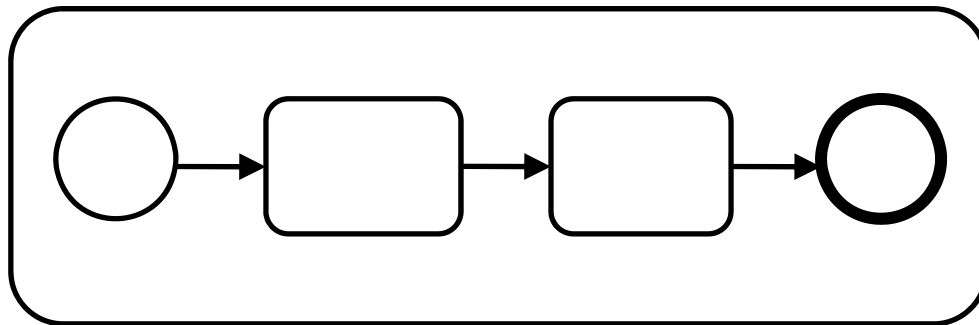
Zdarzenie pośrednie są często przypisywane do krawędzi procesów jako zdarzenia typu *catch*.



Działanie

Wyróżnia się dwa podstawowe typy działań:

- podprocesy
- zadania



- Zadanie
- Podproces
- Podproces zwinięty do pojedynczego działania

Typy podprocesów

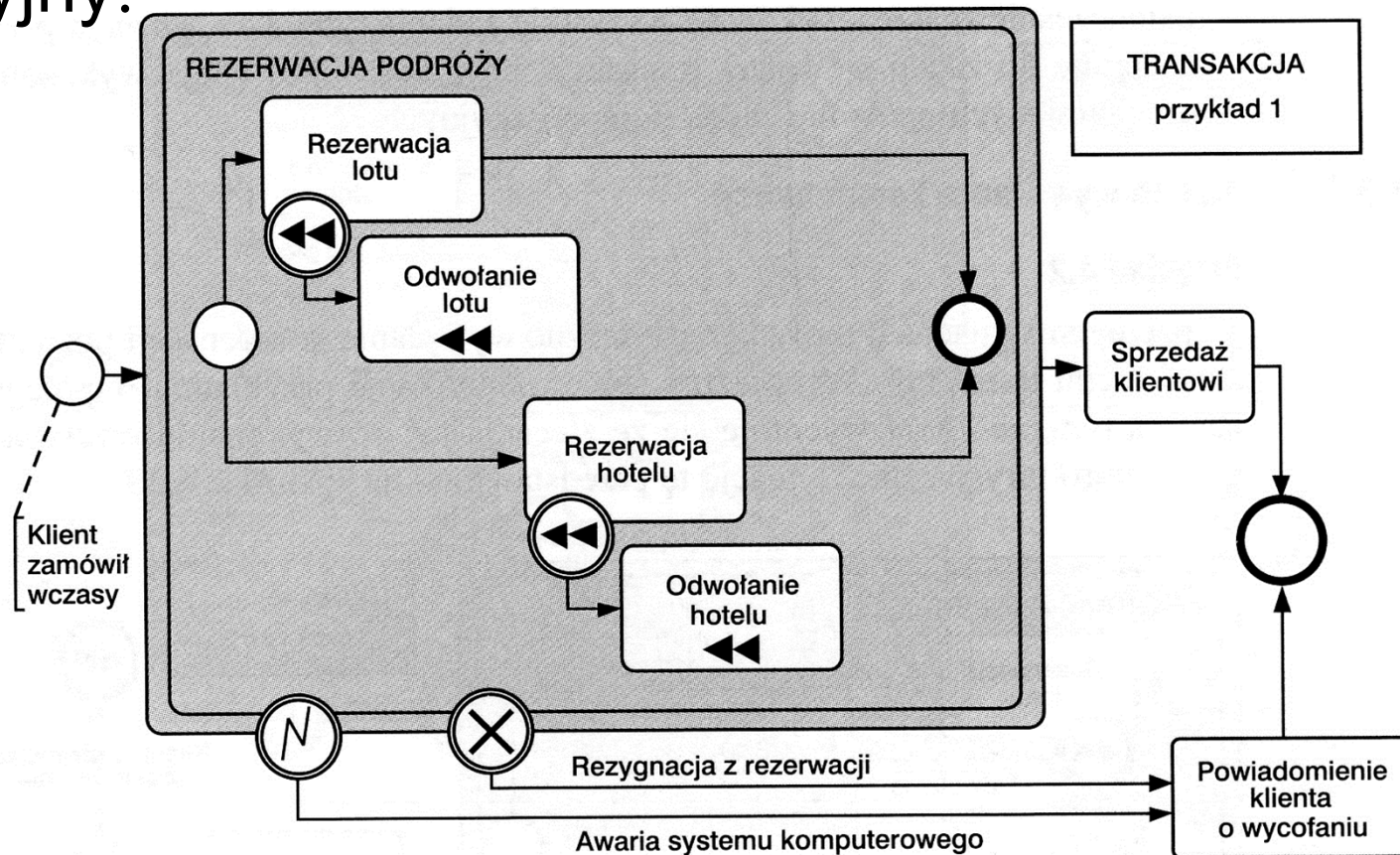
Zwinięte złożone podprocesy mogą mieć dodatkowo przypisane własności:

- podproces z wewnętrzną pętlą,
- wiele-wystąpień podprocesu,
- podproces typu ad-hoc,
- podproces kompensacyjny.



Podprocesy transakcyjne

BPMN pozwala wyróżniać zbiory zadań, które są przetwarzane w ramach podprocesu w sposób transakcyjny:



Standardowe typy zadań



- **service** – usługa realizowana jako usługa sieciowa lub zautomatyzowana aplikacja



- **receive** – oczekiwanie na komunikat



- **send** – wysłanie komunikatu



- **user** – zadanie wykonywane przez człowieka wspieranego przez aplikację komputerową



- **manual** – zadanie wykonywane bez pomocy programów komputerowych



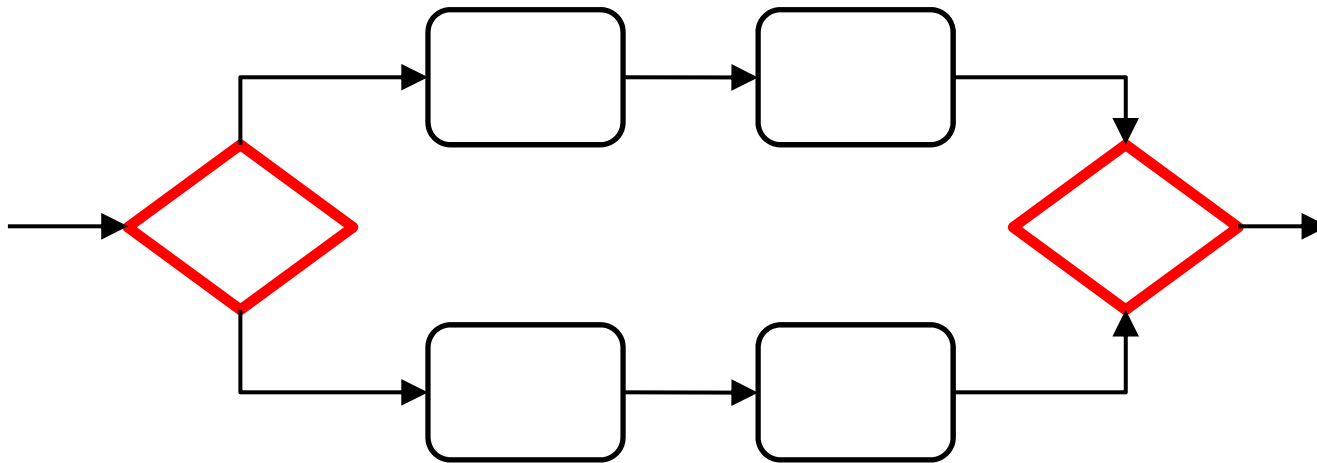
- **script** – zadanie wykonywane przez system zarządzania przepływami pracy



- **reference** – odwołanie do już zdefiniowanych zadań

Bramki

Przejścia są elementem modelu używanym do rozwidlania i łączenia sekwencyjnych fragmentów przepływów.



Typy bramek



Event-Based



Inclusive (OR)



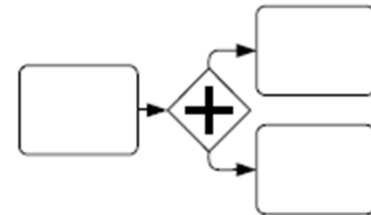
Complex



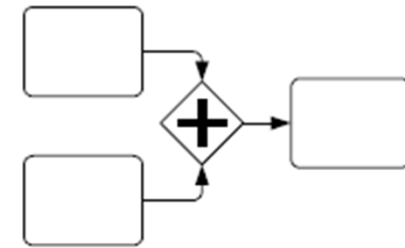
Parallel (AND)



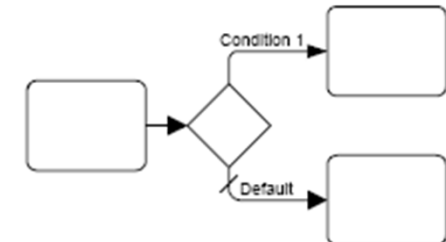
Fork (and split)



Join (and join)

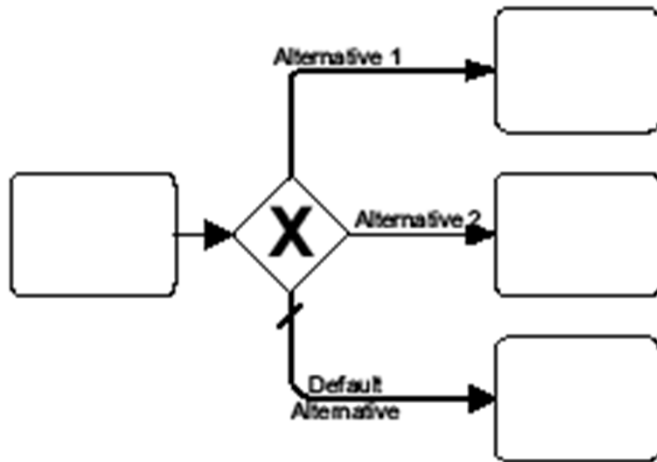


Alternatywa

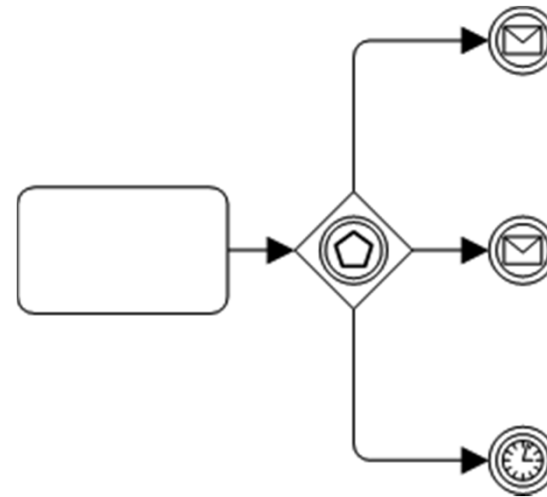


Rozwidlenie typu XOR

Bramka bazująca na danych.
Tylko jedna z alternatyw
może być spełniona. Praca
przepływnie dokładnie jedną z
alternatywnych ścieżek.

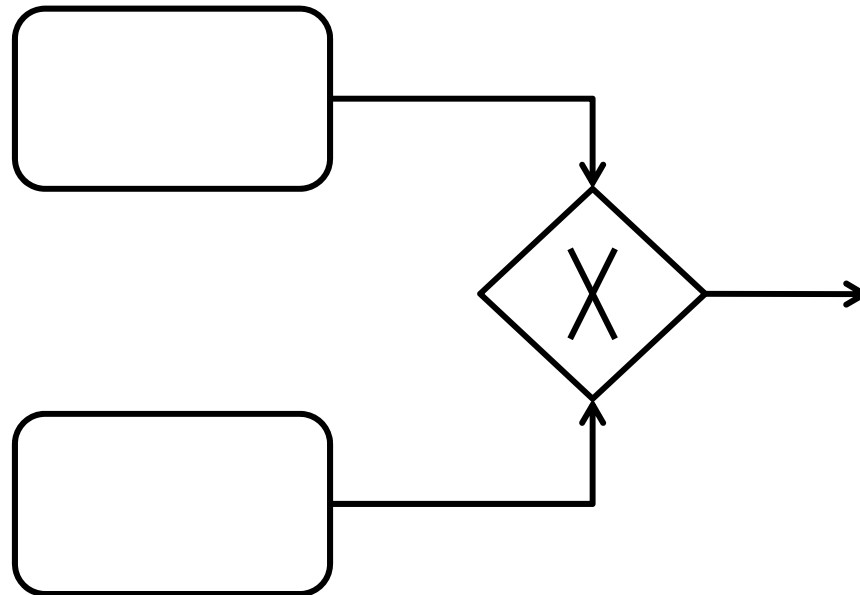


Bramka bazujące na
zdarzeniach:



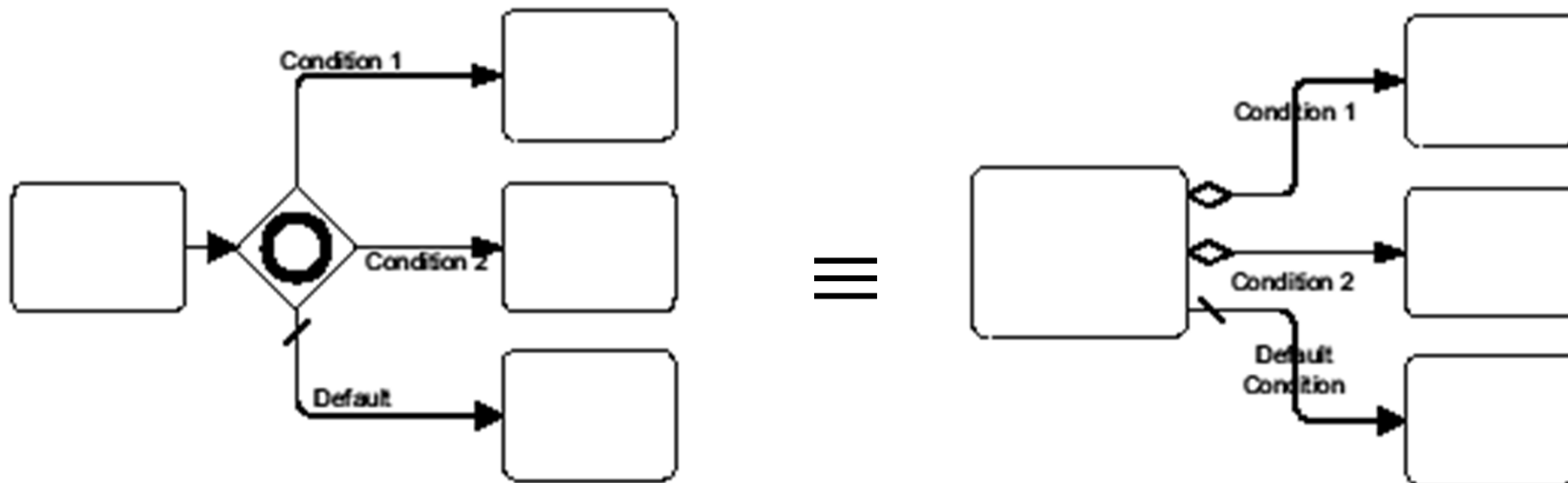
Złączenie typu XOR

Bramka odblokuje przepływ bezpośrednio po dopłynięciu do niej pierwszej z alternatywnych gałęzi przepływów.



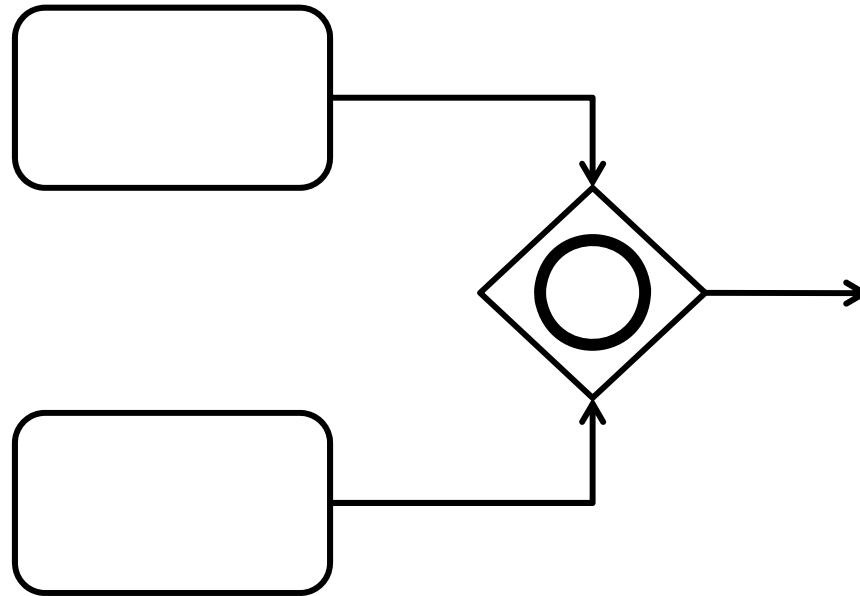
Rozwidlenie typu OR

W wypadku zastosowania bramki typu OR, co najmniej jedna z alternatyw musi być spełniona. Praca przepływnie co najmniej jedną z alternatywnych ścieżek.



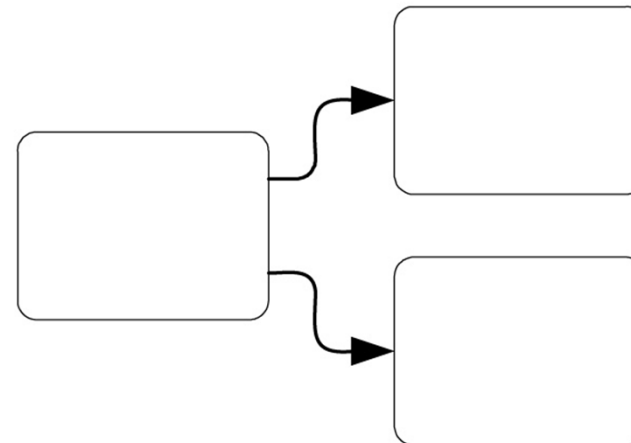
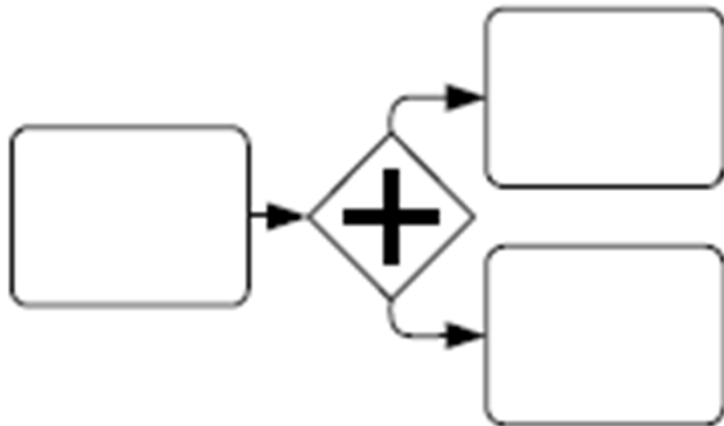
Połączenie typu OR

W wypadku połączenia, bramka typu OR synchronizuje kontynuację przepływu prace przez oczekiwanie na zakończenie wszystkich aktywnych gałęzi przepływu.



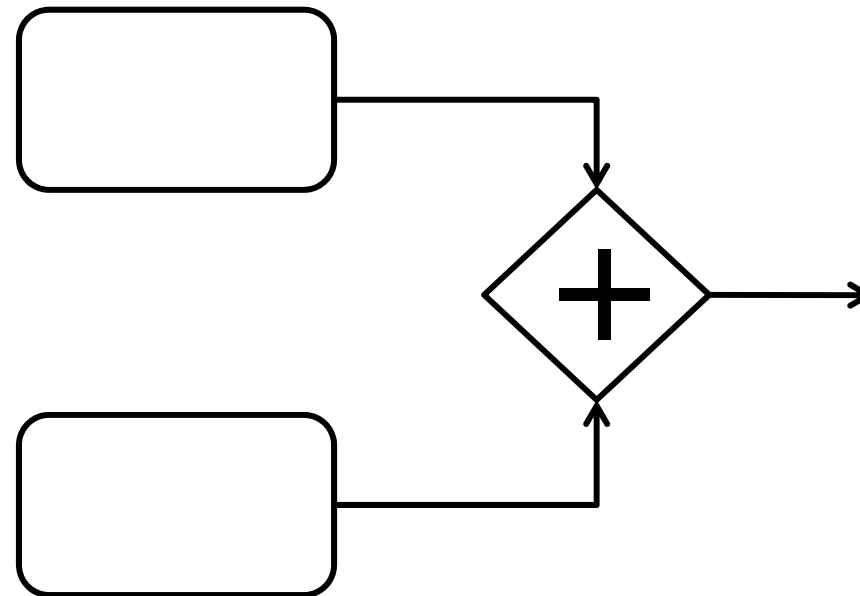
Rozwidlenia typu AND

W wypadku zastosowania bramki typu AND to rozwidlania procesów, przepływ pracy równocześnie przepływnie wszystkimi równoległymi ścieżkami.






Połączenia typu AND

W wypadku zastosowania bramki typu AND do łączenia równoległych gałęzi procesów, bramka przepuści przepływ dopiero po zakończeniu działania wszystkich równoległych gałęzi wykonywania procesu.



Połączenia

Konstruktor	Opis	Notacja
Przepływ pracy	Określa kierunek przepływu pracy między poszczególnymi działaniami.	
Przepływ komunikatu	Pokazuje przepływ komunikatów między uczestnikami procesów (jednostkami organizacyjnymi).	
Powiązanie	Umożliwia kojarzenie z elementami przepływu informacji tekstowych i graficznych.	

Przykłady połączeń

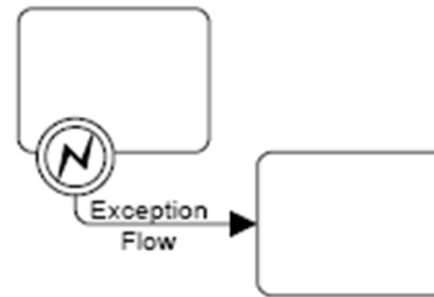
Przeptyw warunkowy



Przeptyw domyślny



Przeptyw będący wynikiem wystąpienia wyjątku





Powiązanie kompensacyjne



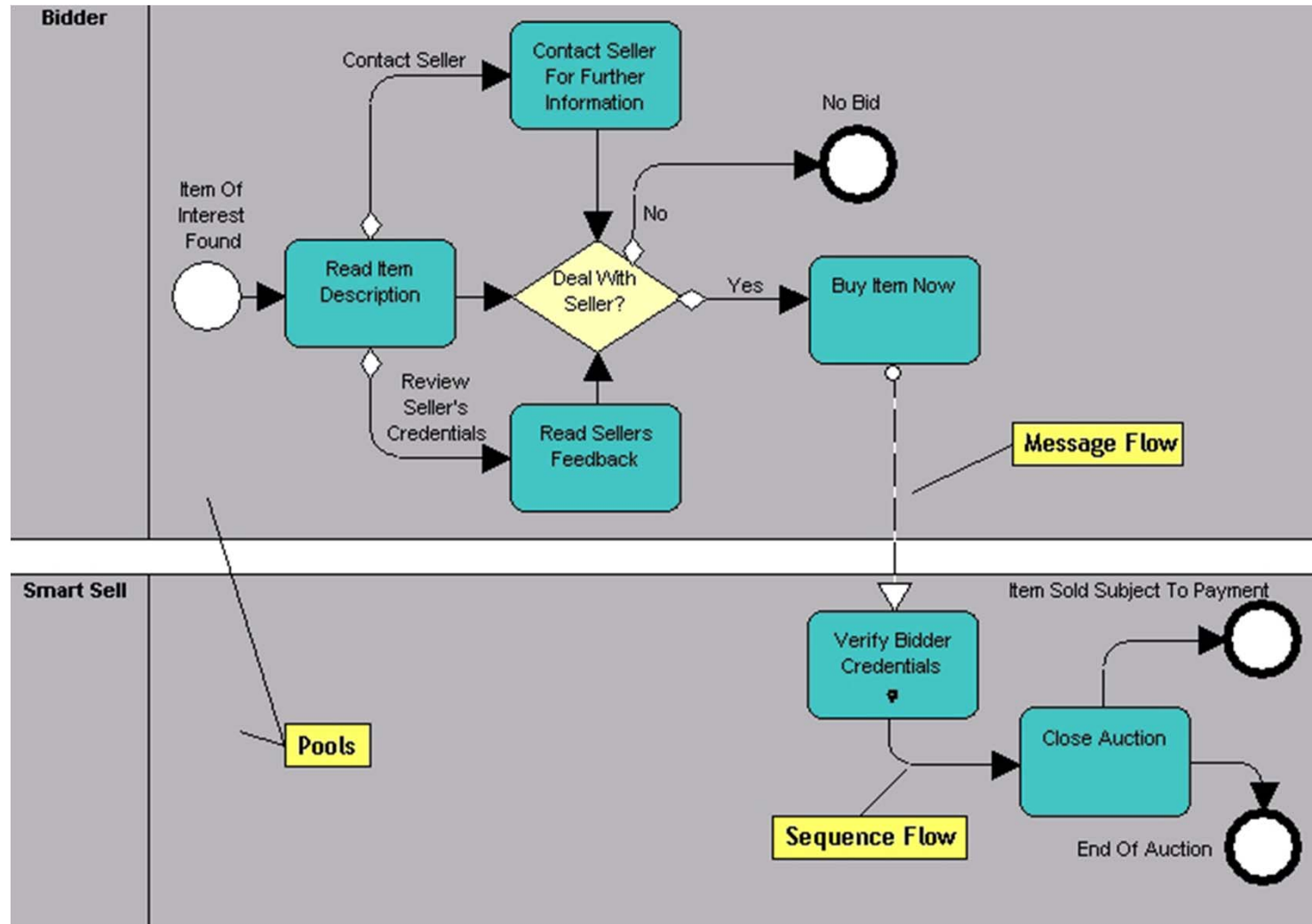
Wykonawcy

Działania (składowe procesów) są wykonywane przez konkretnych wykonawców. Wykonawcami mogą być jednostki organizacyjne i ich jednostki składowe, ale również funkcje pracownicze lub lokalizacje.

Konstruktor	Opis	Notacja
Jednostka organizacyjna ang. pool	Reprezentuje wykonawców procesu i grupuje aktywne elementy procesów. Zazwyczaj jest to firma.	
Jednostka składowa ang. lane	Reprezentuje wykonawców procesu i grupuje aktywne elementy procesów wewnątrz jednostki organizacyjnej. Zazwyczaj jest to wydział firmy	

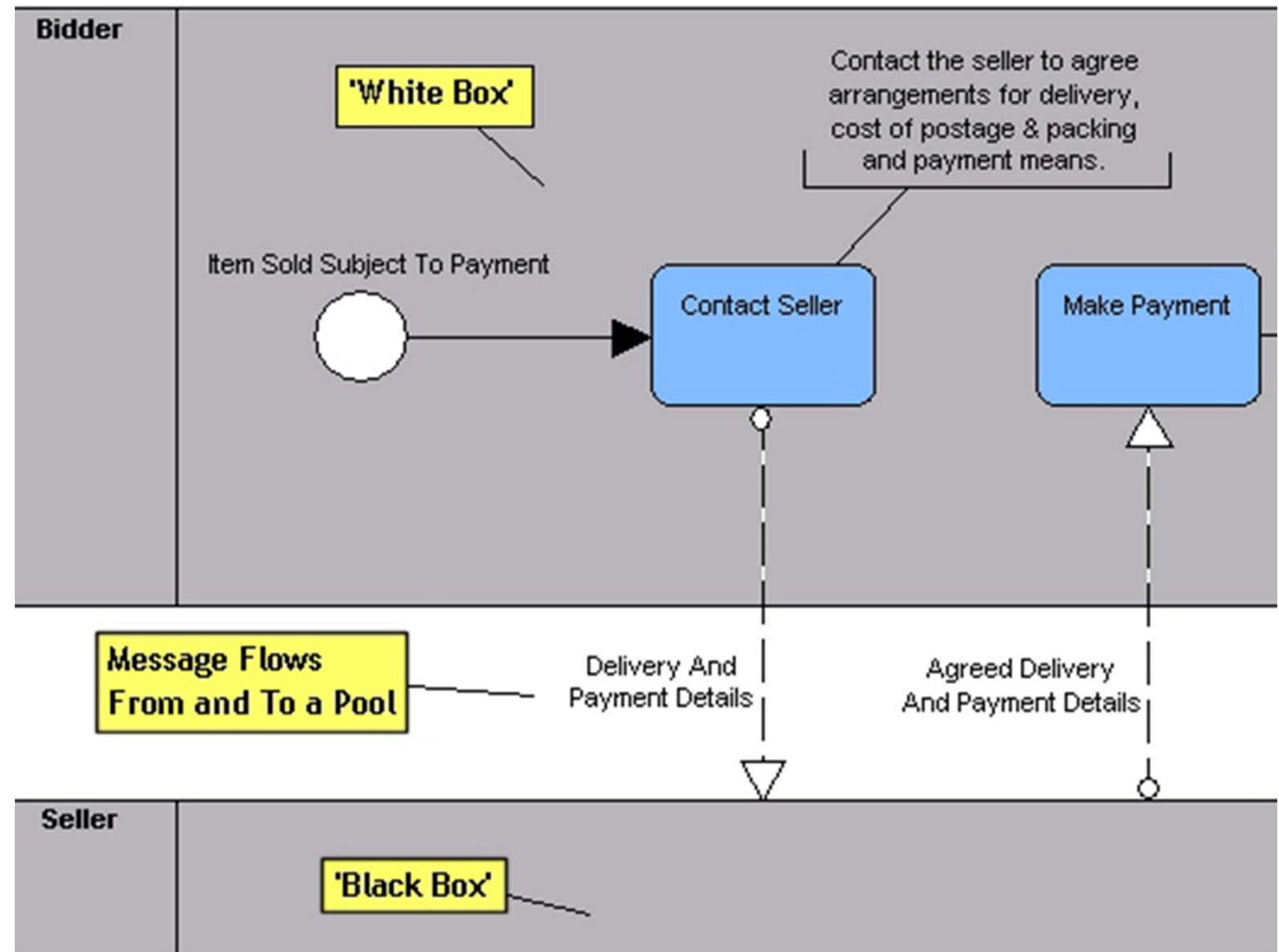
Wykonawcy

Przykład:






Wykonawcy

Możliwe jest modelowanie wykonawcy jako czarnej skrzynki; tj. bez opisu wewnętrznych procesów.

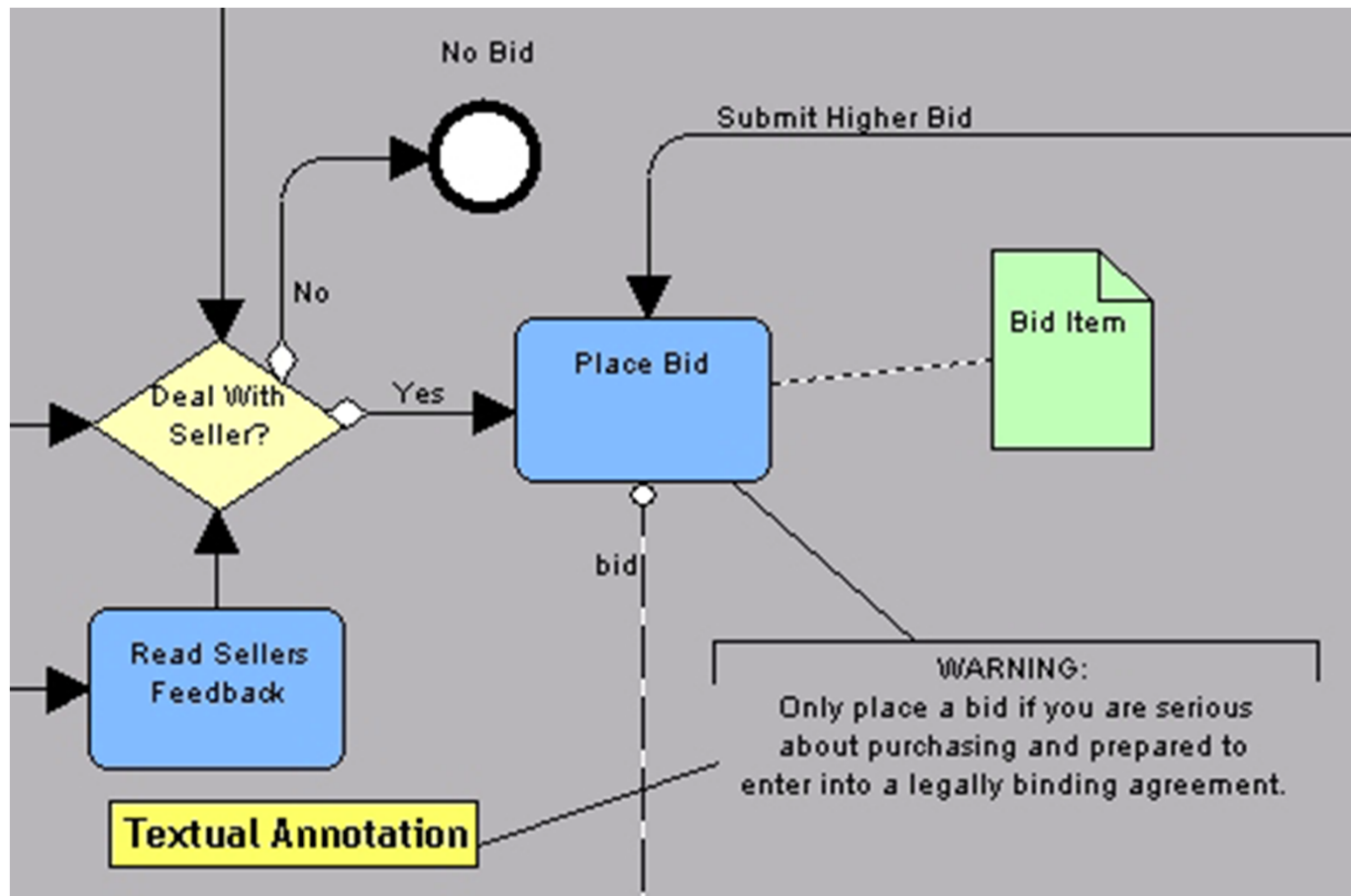


Artefakty













Konstruktor	Opis	Notacja
Dane	Modelują dane wejściowe niezbędne do wykonania określonego działania lub dane wyjściowe będące wynikiem działania.	 nazwa
Grupa	Grupuje działania w celu dokumentowania lub analizy. Grupa może przekraczać granice zbiorników i torów.	
Adnotacja	Umożliwia przypisywanie poszczególnym elementom procesów dodatkowych informacji tekstowych.	 tekst

Artefakty

Przykład zastosowania artefaktów.

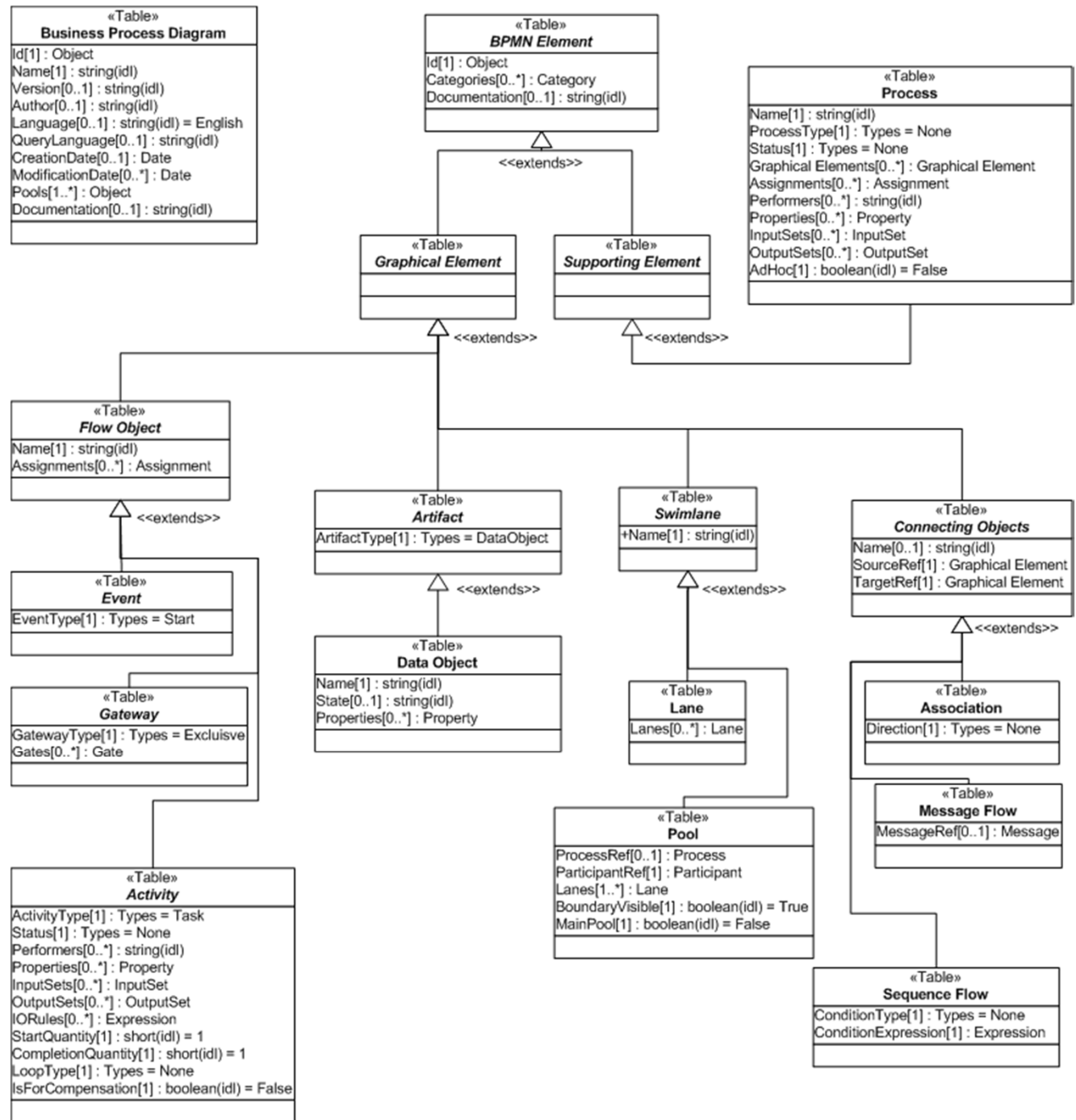


Reguły topologii przepływów

From\To						
		↗	↗	↗	↗	↗
		↗	↗	↗	↗	↗
		↗	↗	↗	↗	↗
		↗	↗	↗	↗	↗
		↗	↗	↗	↗	↗
						

Atrybuty elementów notacji BPMN

BPMN zakłada stosowanie dodatkowych informacji opisujących elementy graficzne diagramów. Atrybuty opisują zarówno diagram jako całość, jak i wszystkie jego elementy składowe.



Atrybuty elementów notacji BPMN

Na przykład działania są opisane następującymi atrybutami

Nazwa	Opis
ActivityType (Task Sub-process)	Typ działania - dozwolone dwie wartości: zadanie lub podproces.
Status (None Ready Active Canceled ...)	Określa stan danego działania.
Performers (0-n)	Opis zasobów niezbędnych do wykonania działania. Wartościami mogą być nazwiska osób, nazwy etatów.
Properties (0-n)	Cechy przypisane do zadania, np. nazwa działania: „Rejestrowanie zamówienia”.
InputSet (0..n)	Definiuje dane wejściowe wymagane dla uruchomienia danego działania
OutputSet(0..n)	Definiuje dane wyjściowe wymagane na wyjściu danego działania
IORules(0..n)	Wyrażenia opisujące sposób wyznaczania danych wyjściowych na podstawie danych wejściowych
...	...

Odzworowanie diagramów BPMN do języka BPEL

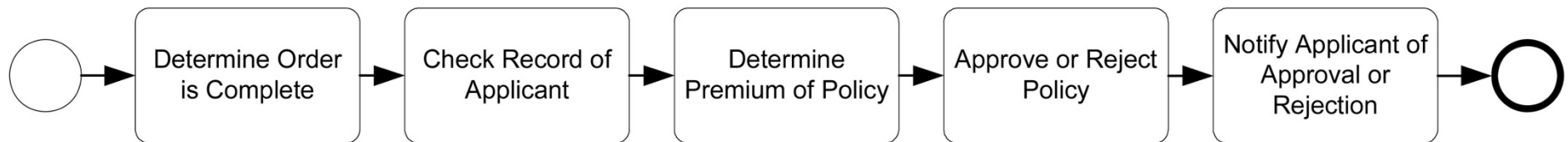
- Znaczą część specyfikacji standardu BPMN 1.2 (100 stron z niewiele ponad 300) zawiera opis przykładów odzworowania elementów graficznych diagramów razem z ich atrybutami do języka BPEL.
- Opis ten jest podstawą do konstrukcji oprogramowania automatyzującego taką transformację

Odwzorowanie diagramów BPMN do BPEL

Looping	Mapping to BPEL4WS
LoopCounter	Atrybut będzie odwzorowany do zmiennej <i>LoopCounter</i> języka BPEL: <pre><variable name="[activity.Name]_loopCounter" messageType="loopCounterMessage" /></pre>
Initialization of the LoopCounter	Zmienna <i>LoopCounter</i> będzie inicjalizowana przed implementacją właściwej pętli: <pre><assign name="[activity.Name]_initialize_loopCounter"> <copy> <from expression="0"/> <to variable="[activity.Name]_loopCounter" part="loopCounter" /> </copy> </assign></pre>
Incrementing the LoopCounter	Zmienna <i>LoopCounter</i> będzie uaktualniana na końcu pętli while <pre><assign name="[activity.Name]_increment_loopCounter"> <copy> <from expression=" bpws:getVariableData([activity.Name]_loopCounter, loopCount) + 1"/> <to variable="[activity.Name]_loopCounter" part="loopCounter" /> </copy> </assign></pre>

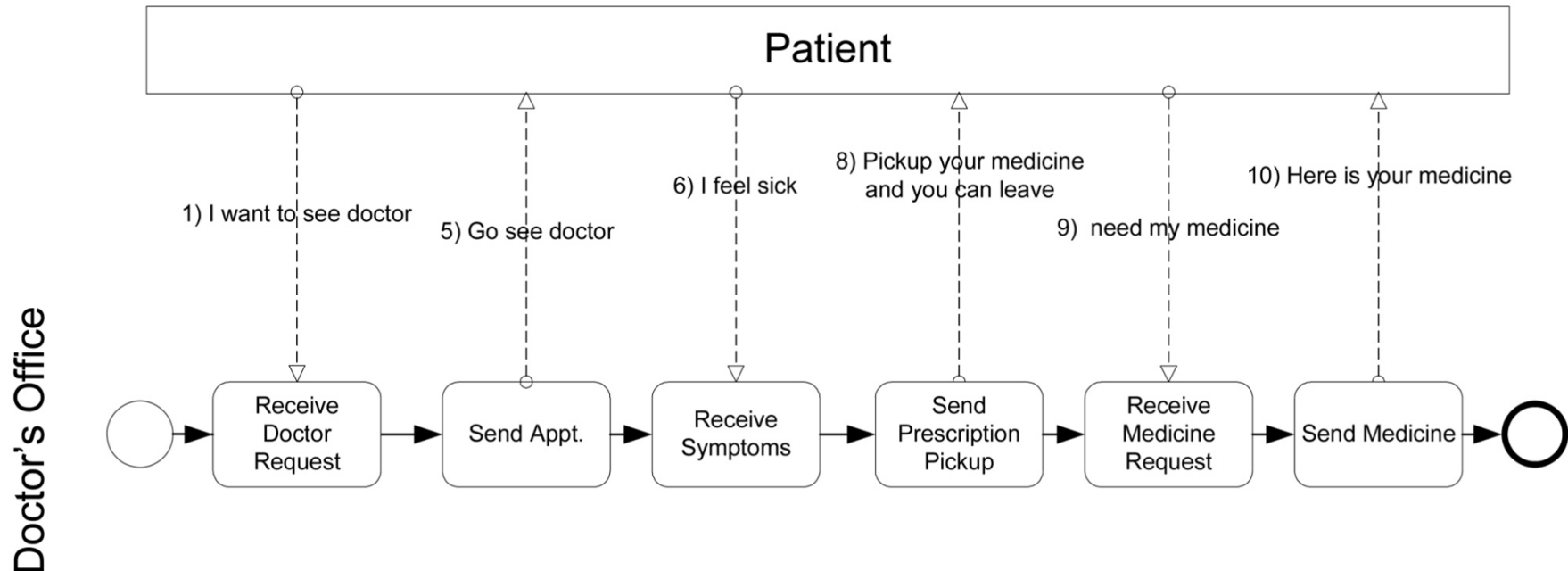
Podstawowe typy diagramów BPMN

- **Prywatny proces biznesowy** – jest to proces wykonywany wewnątrz danej jednostki organizacyjnej. Dla tej klasy procesów typowe jest pomijanie notacji wykonawców zadań.



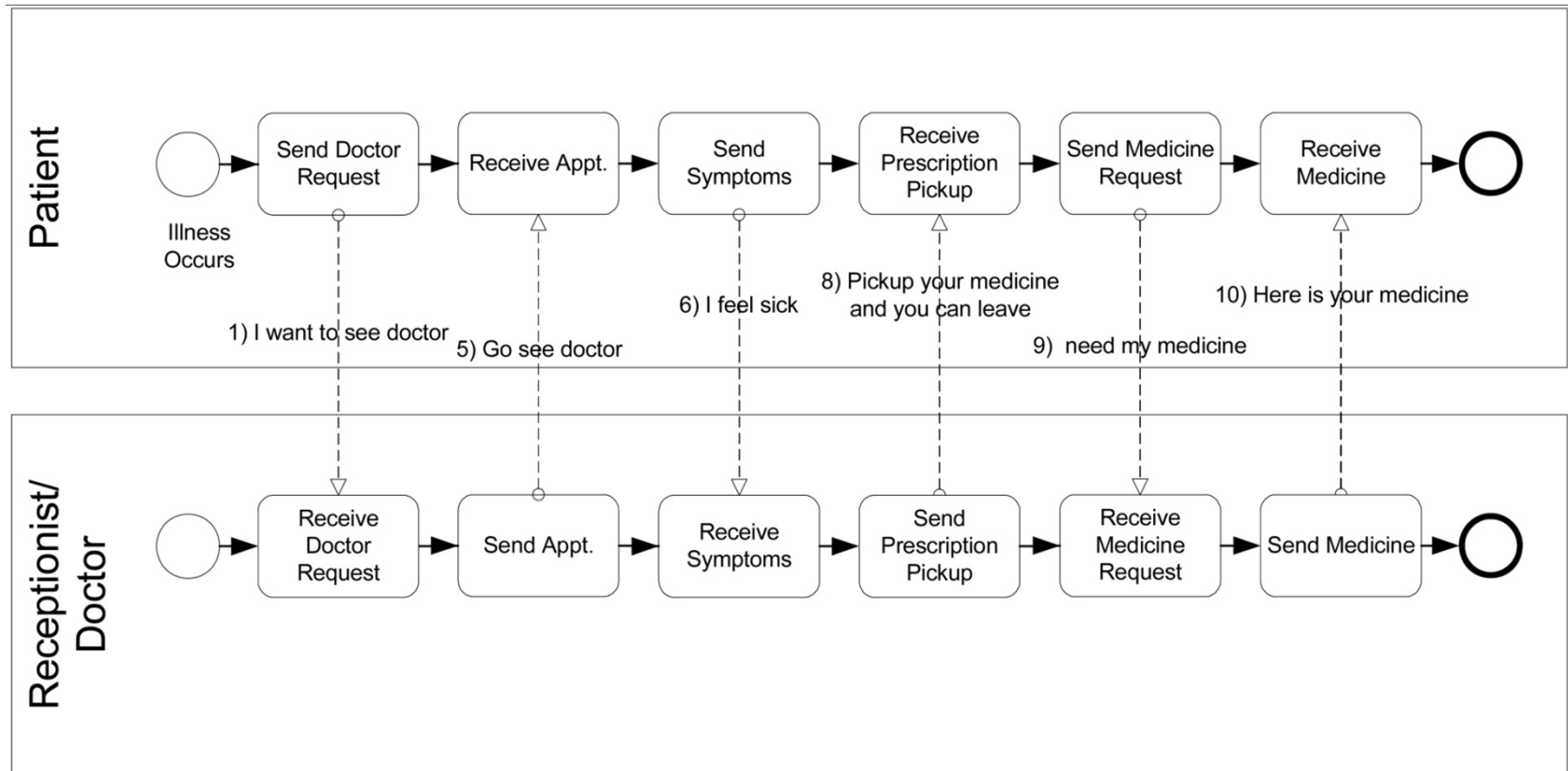
Podstawowe typy diagramów BPMN

- **Publiczny (abstrakcyjny) proces biznesowy** – służy do modelowania interakcji procesu prywatnego z innym procesem lub wykonawcą. W procesie prywatnym uwidocznione są jedynie działania związane z komunikacją. Drugi proces lub wykonawca są modelowani w postaci czarnej skrzynki.



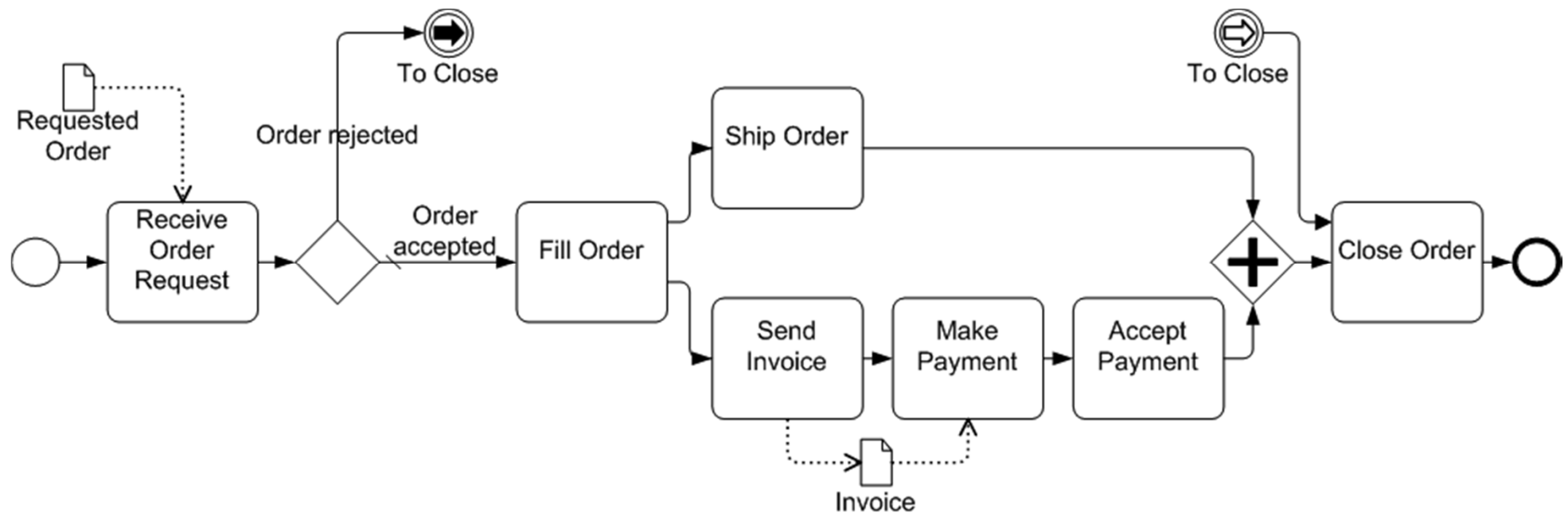
Podstawowe typy diagramów BPMN

- **Globalny (współpracy) proces biznesowy** – pokazuje komunikację między dwoma procesami prywatnymi.

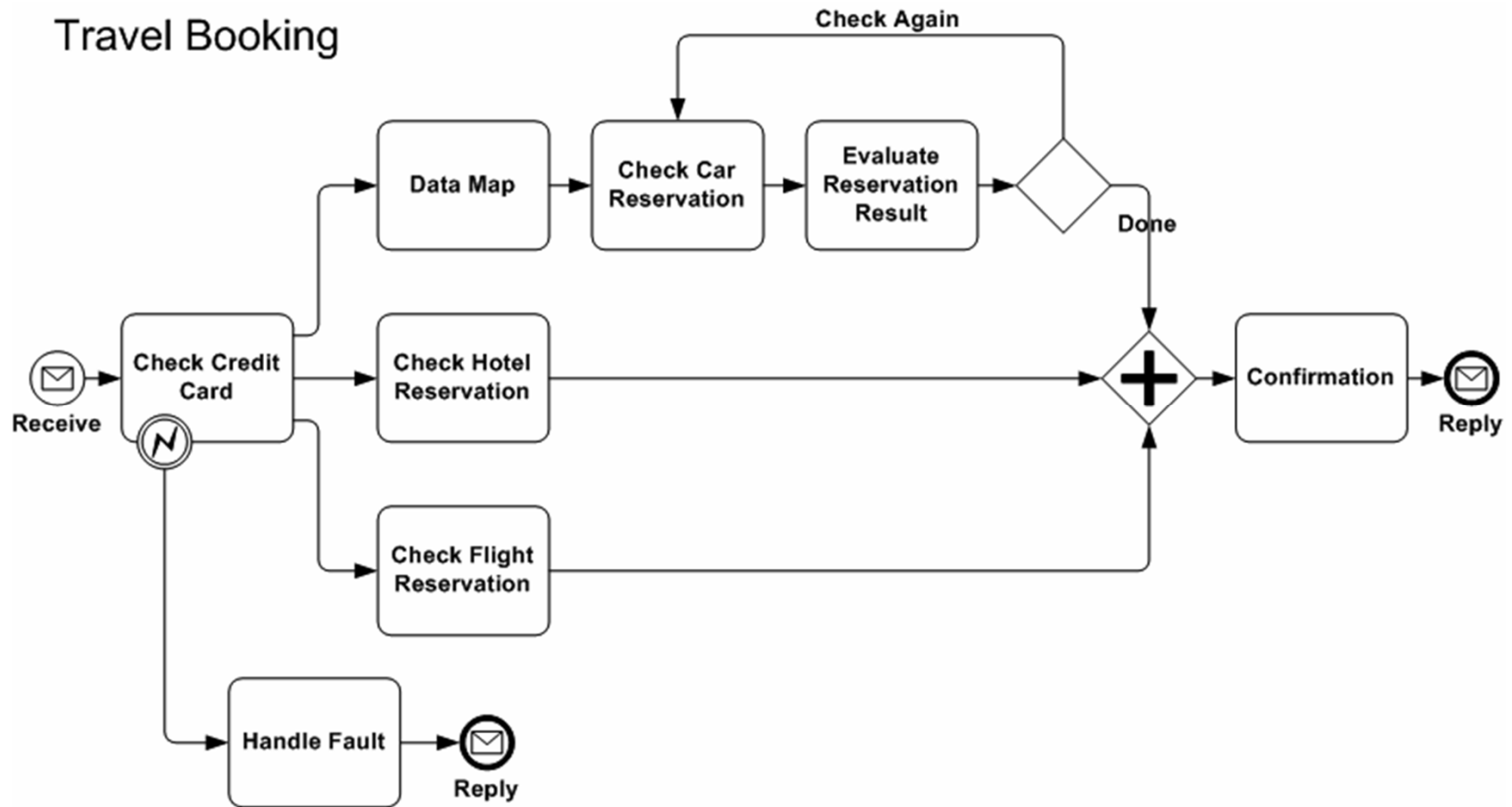


Przykłady diagramów BPMN

Składanie zamówień: przykład zdarzenia Link

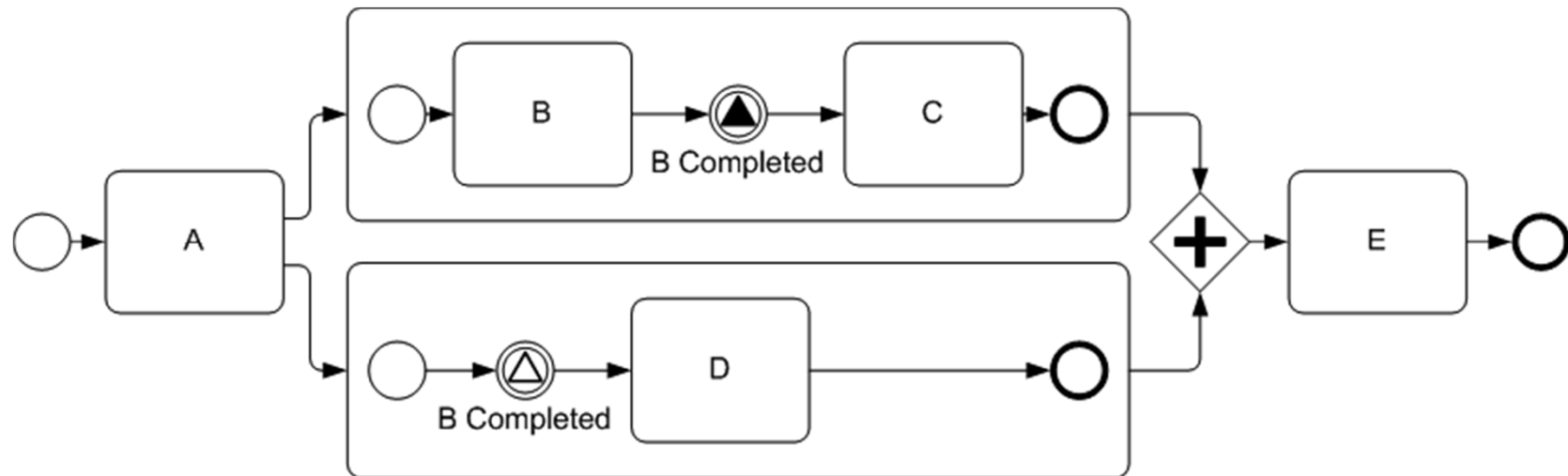


Przykłady diagramów BPMN



Przykłady diagramów BPMN

Przykład synchronizacji pod-procesów

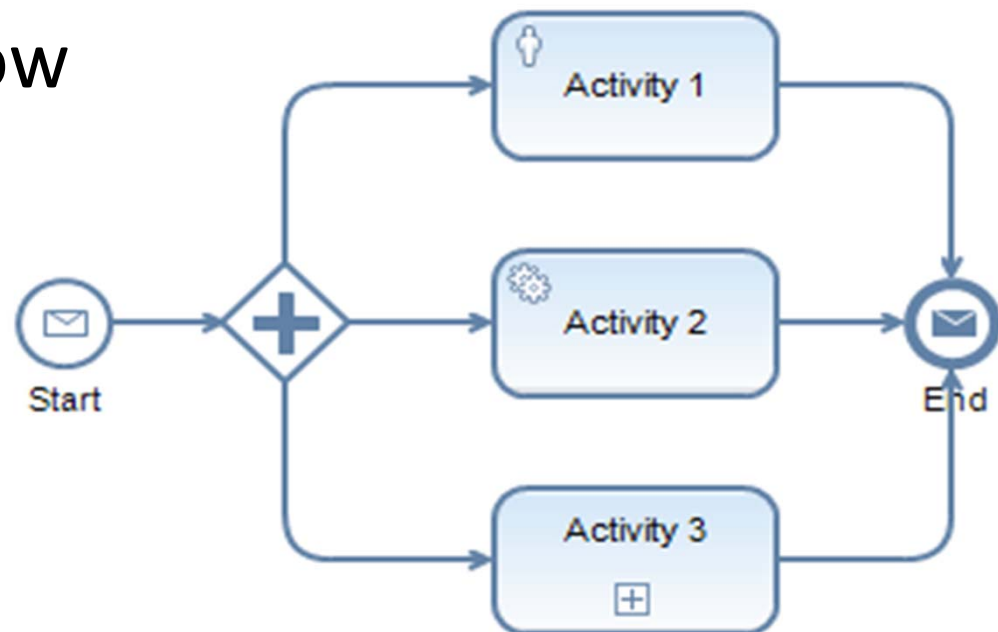


Wzorce procesowe w BPMN

- **Sekwencja działań**

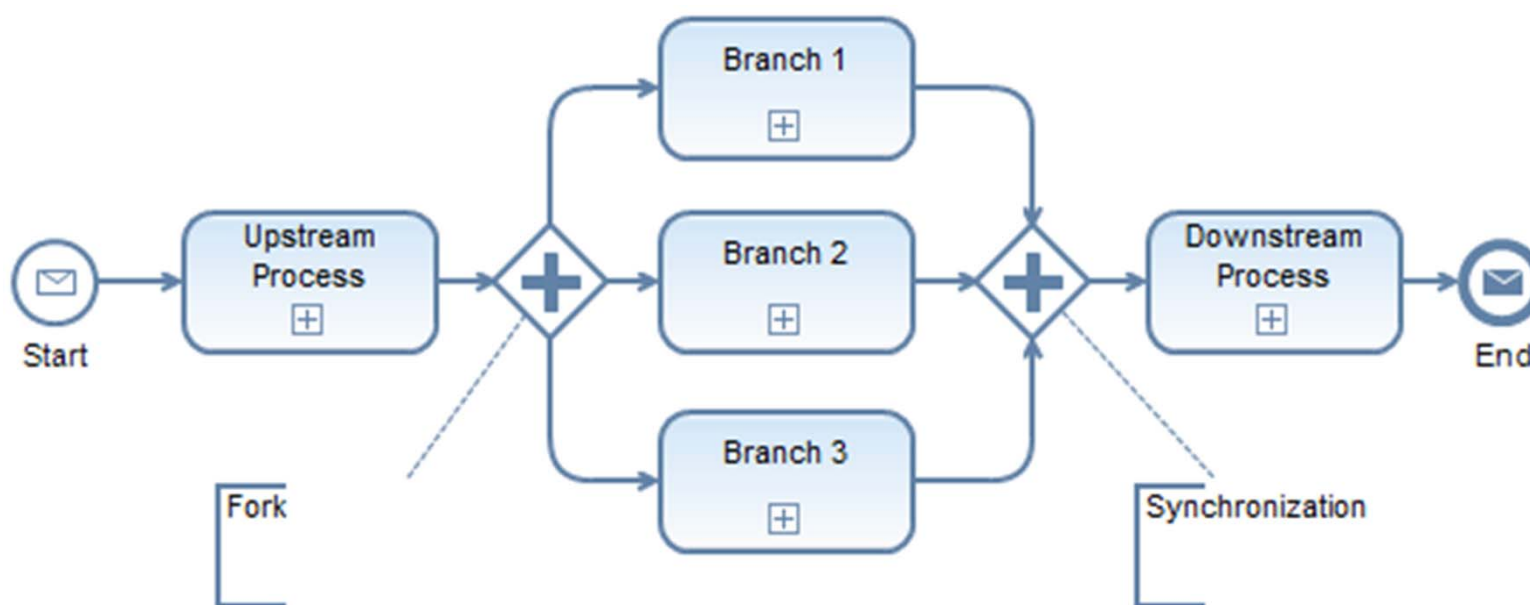


- **Współbieżność wątków wewnątrz procesu –**
ang. *parallel split*



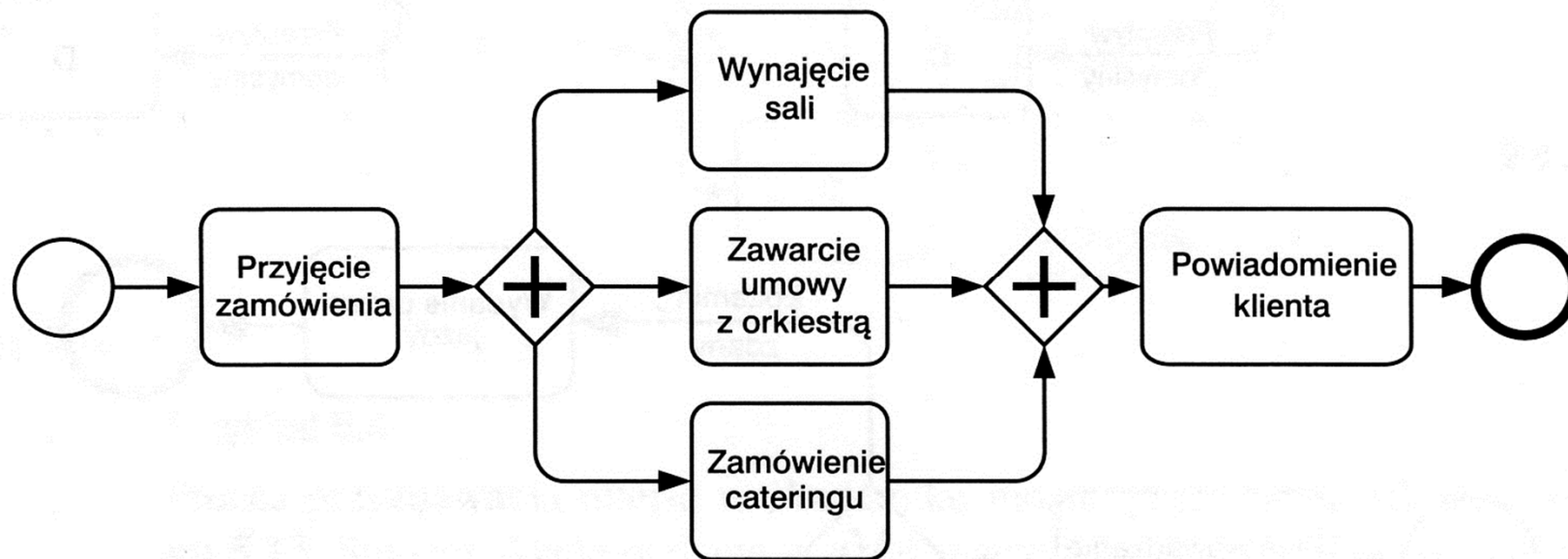
Wzorce procesowe w BPMN

- **Synchronizacja** - kilku współbieżnych wątków tego samego procesu. Dalszy przebieg procesu wymaga zakończenia wszystkich współbieżnych wątków.



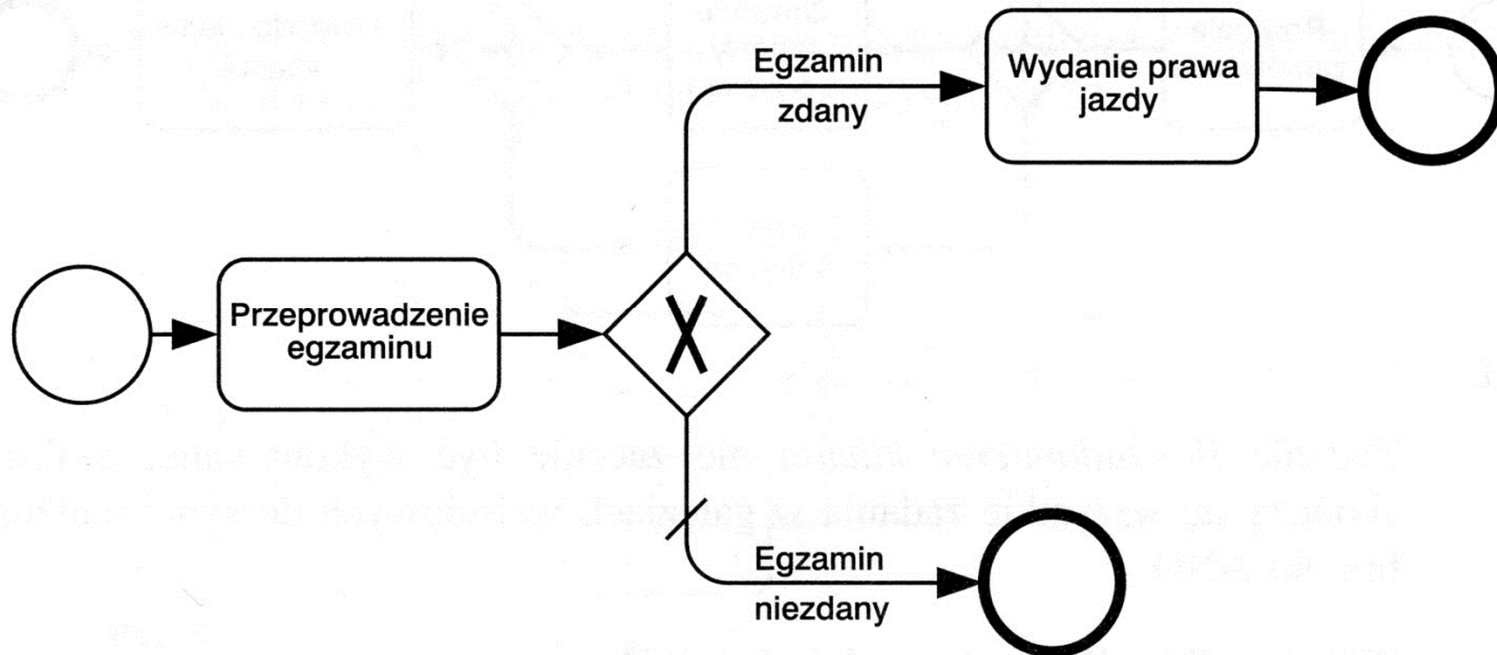
Wzorce procesowe w BPMN

- Przykład synchronizacji wątków procesu



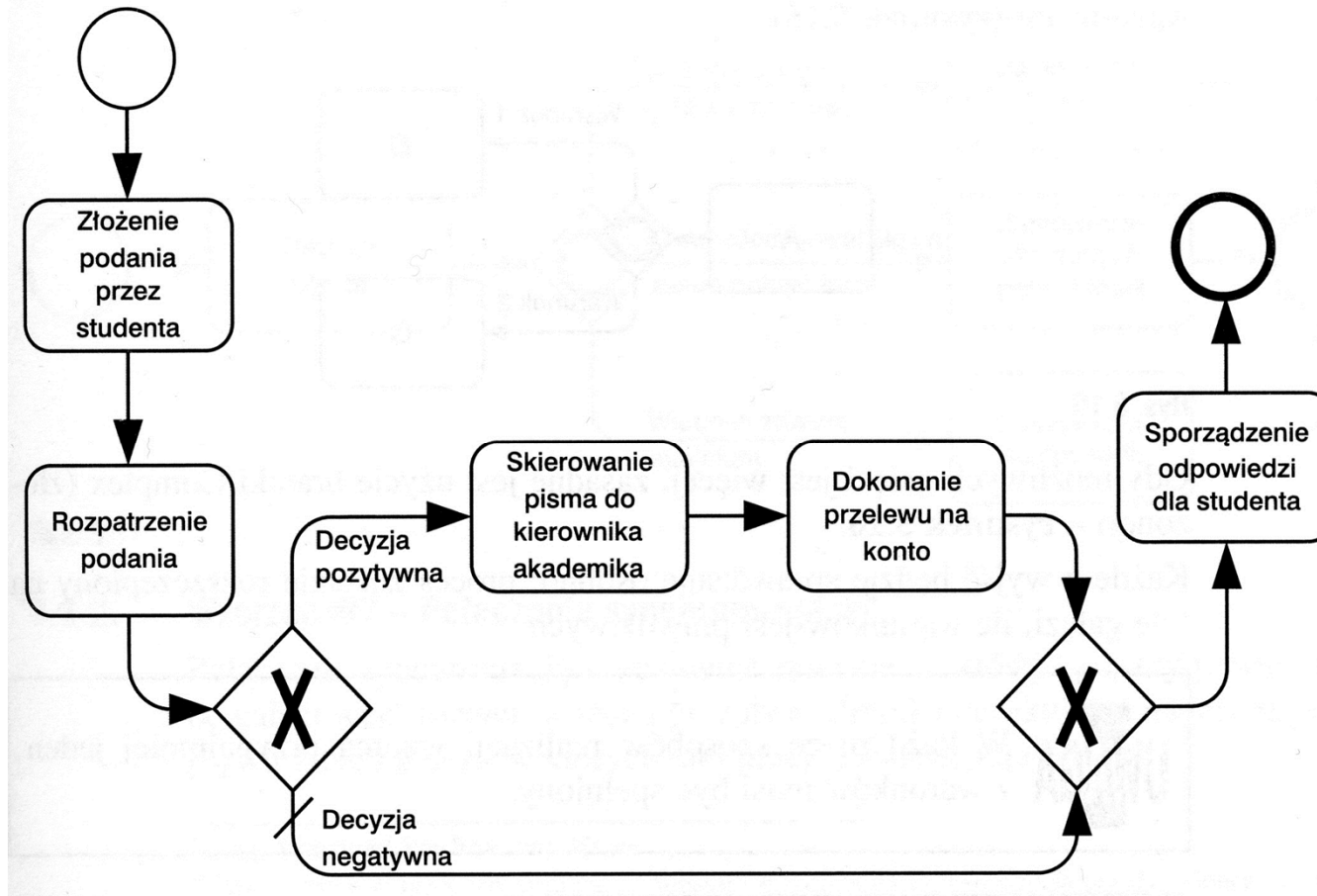
Wzorce procesowe w BPMN

- **Wyłączny wybór ścieżki** – wątek procesu będzie przebiegał dokładnie jedną z alternatywnych dróg



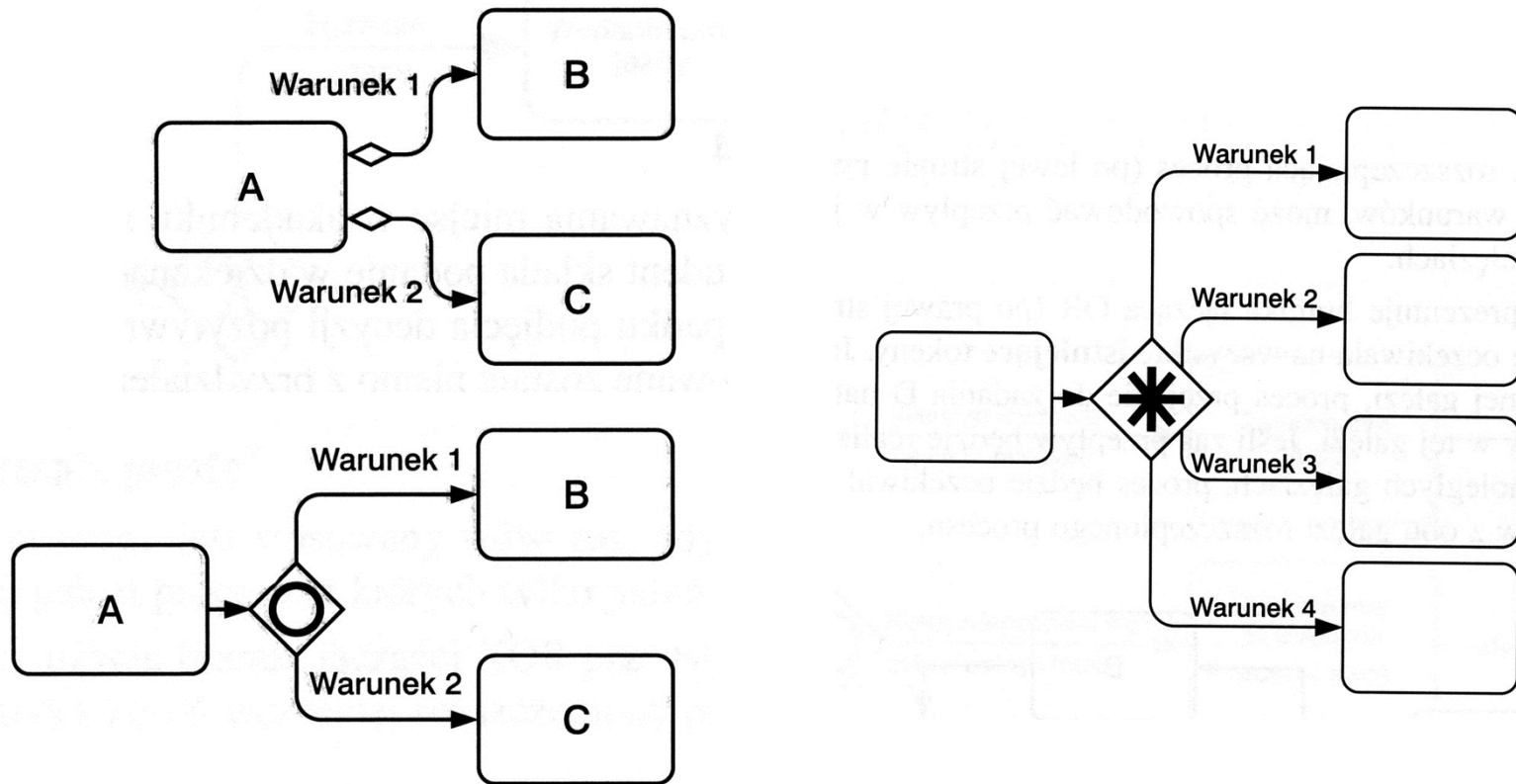
Wzorce procesowe w BPMN

- **Proste połączenie** – proces będzie kontynuowany po pojawieniu się w węźle dokładnie jednego wątku



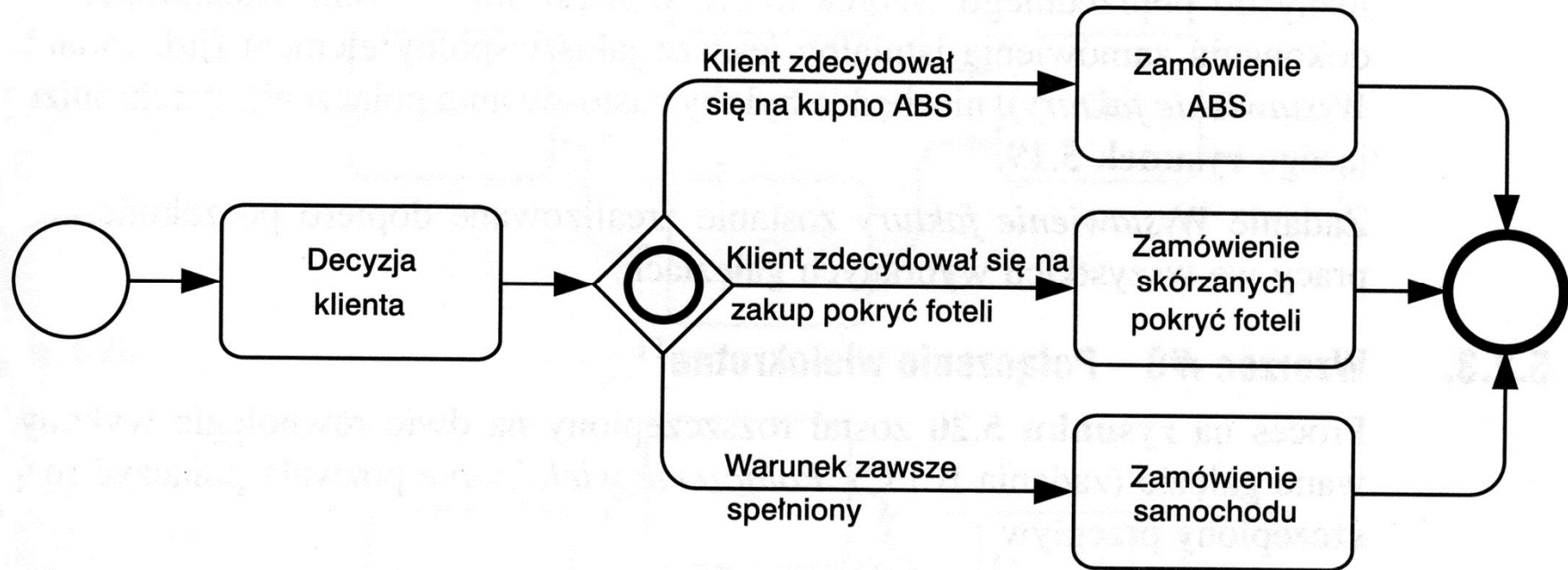
Wzorce procesowe w BPMN

- **Wybór wielokrotny** – proces jest kontynuowany przez jeden lub więcej współbieżnych wątków



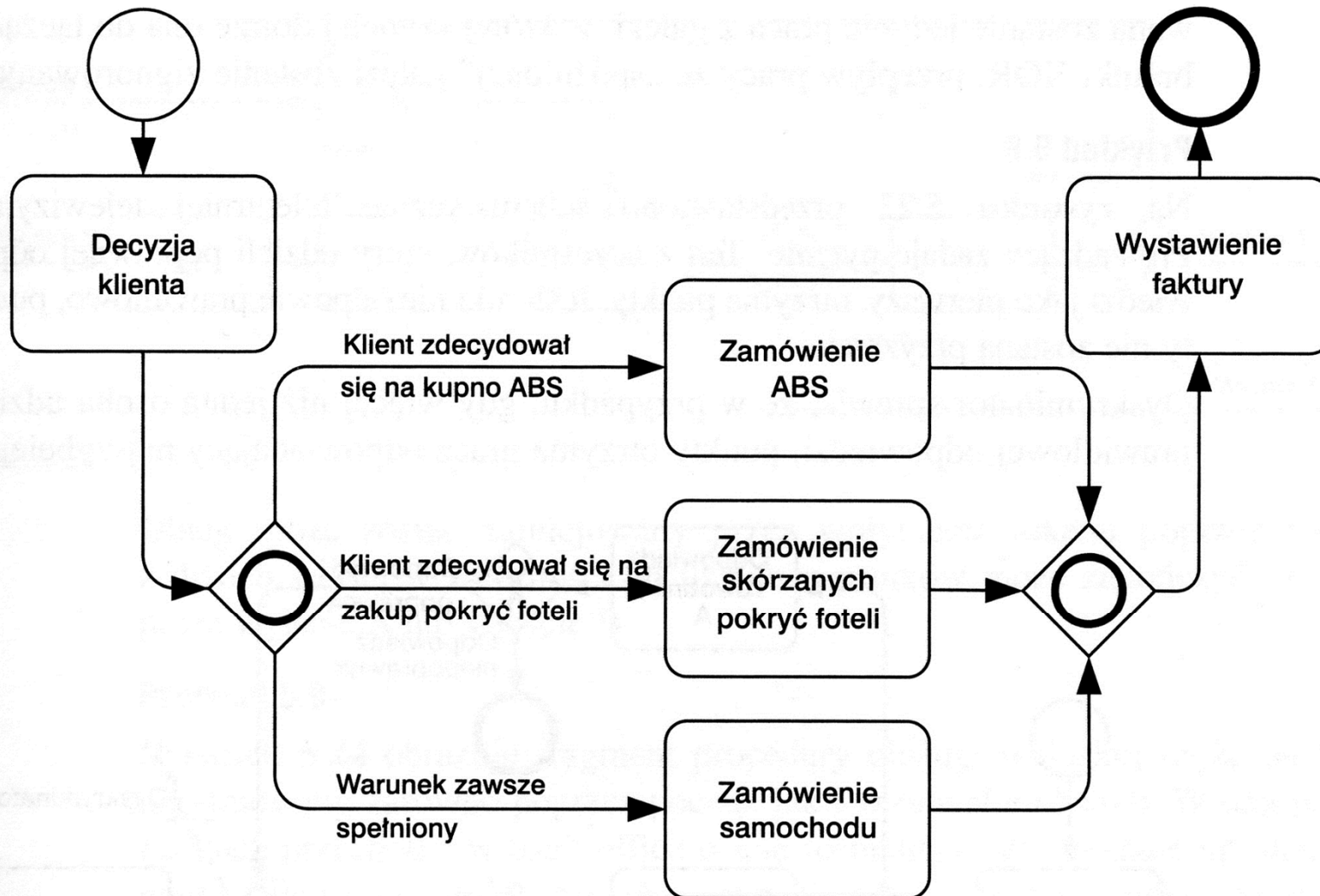
Wzorce procesowe w BPMN

- **Wybór wielokrotny – przykład**



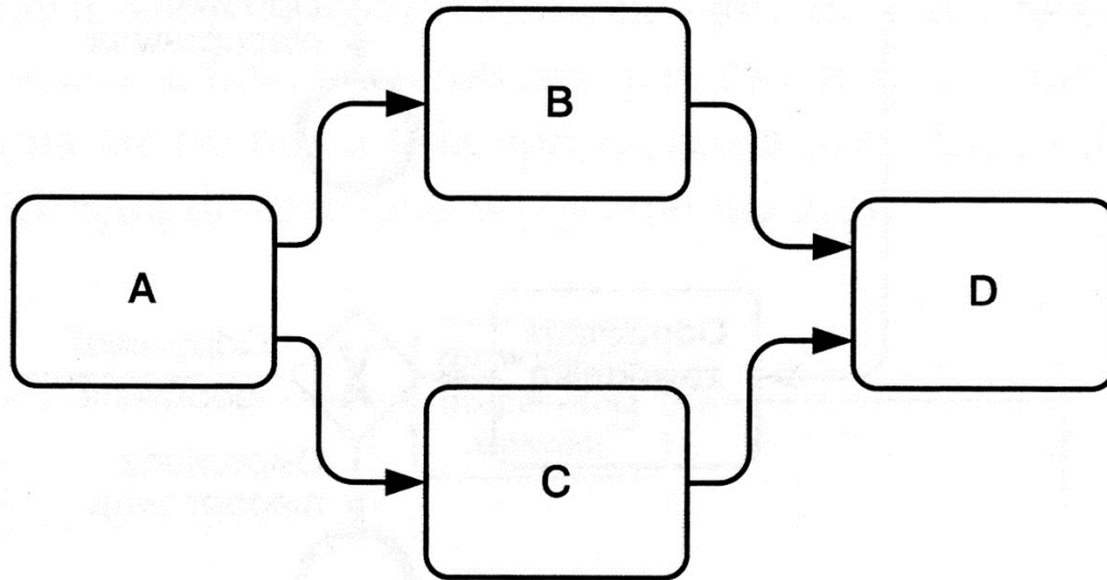
Wzorce procesowe w BPMN

- **Połączenie synchronizujące**



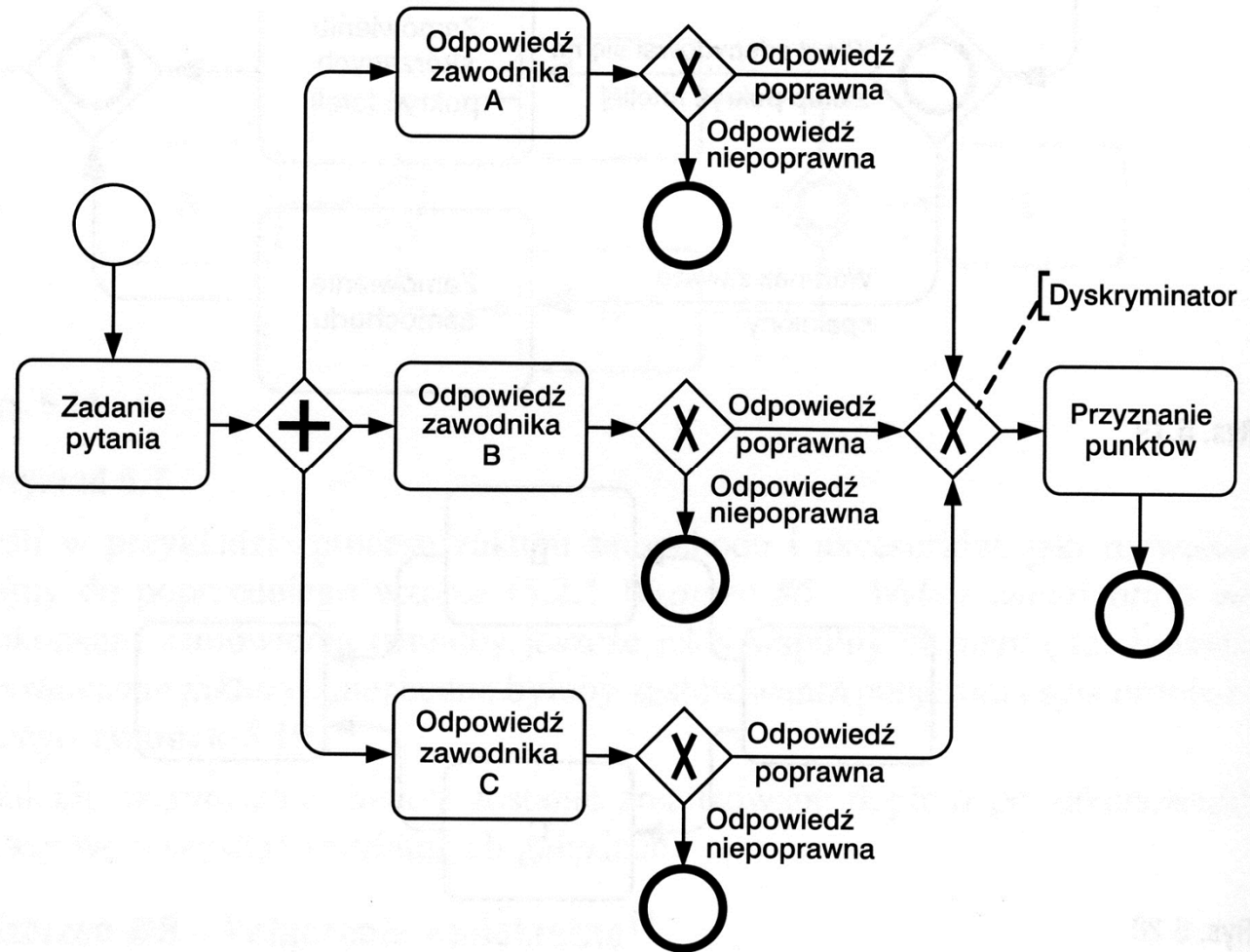
Wzorce procesowe w BPMN

- **Połączenie wielokrotne** – równoległe wątki nie są synchronizowane, lecz kontynuowane w tych samych zadaniach



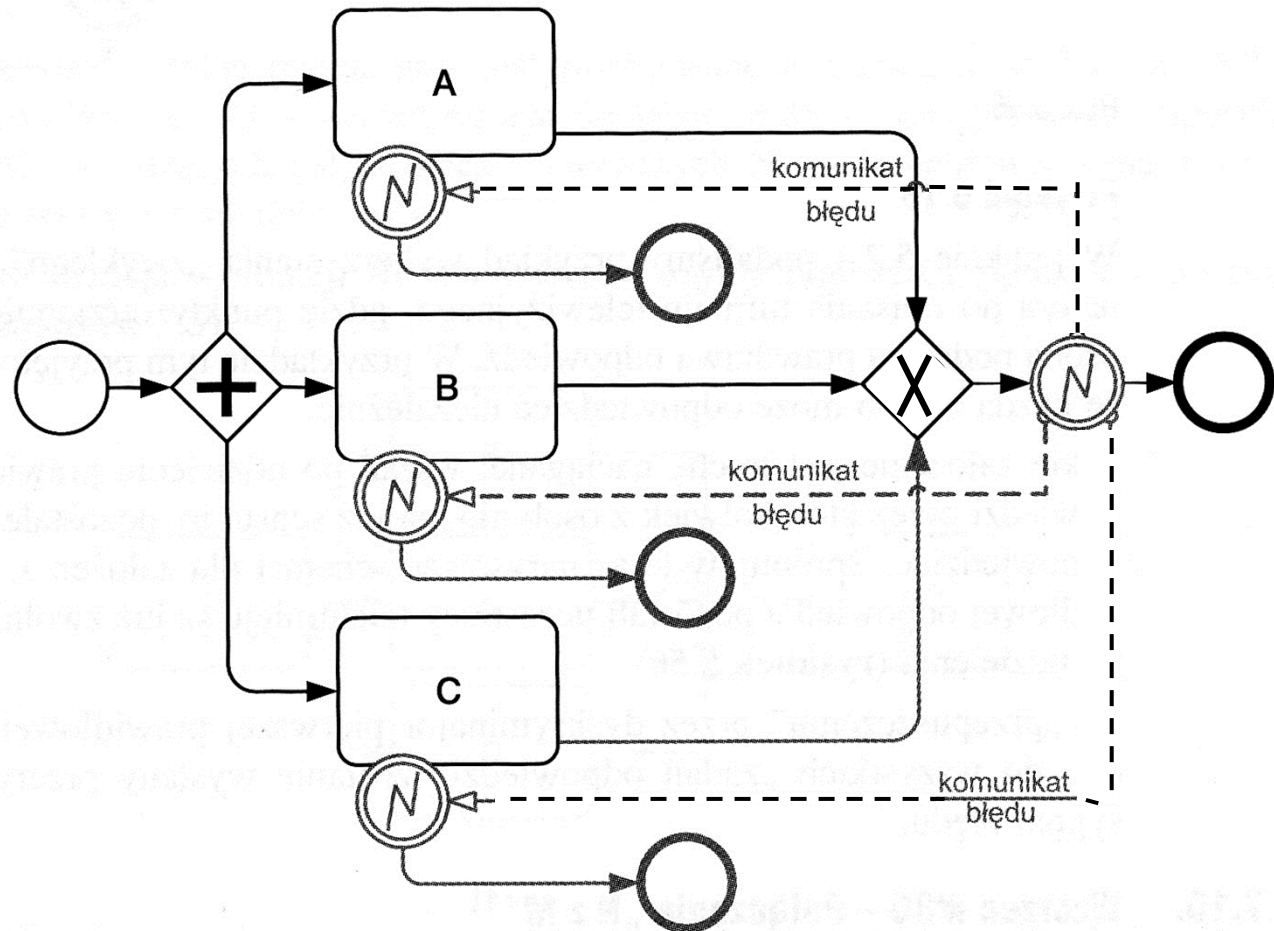
Wzorce procesowe w BPMN

- **Dyskryminator** – kontynuowany jest tylko jeden ze współbieżnych wątków procesu, który pojawi się jako pierwszy na przejściu; pozostałe wątki są **ignorowane**



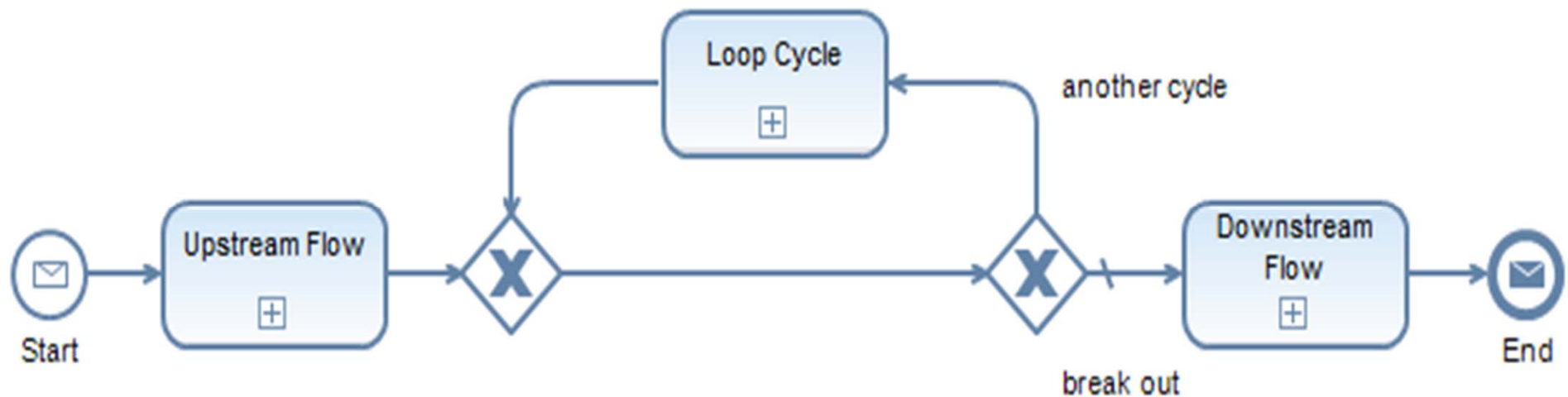
Wzorce procesowe w BPMN

- **Dyskryminator odwołujący** – kontynuowany jest tylko jeden ze współbieżnych wątków procesu, który pojawi się jako pierwszy na przejściu; pozostałe wątki są **odwoływane**



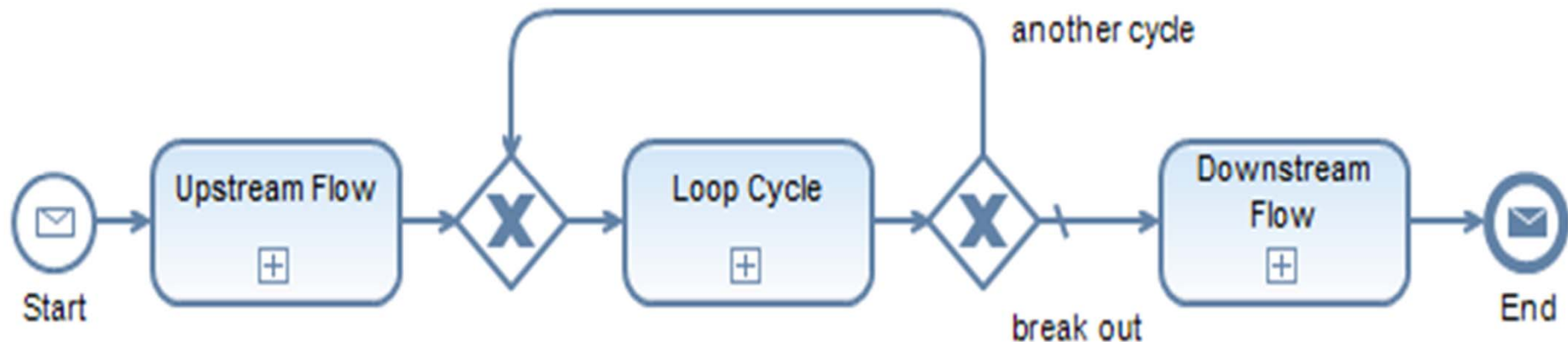
Wzorce procesowe w BPMN

- **Pętla** – iteracyjne powtarzanie zbioru czynności – warunek sprawdzany przed wykonaniem czynności



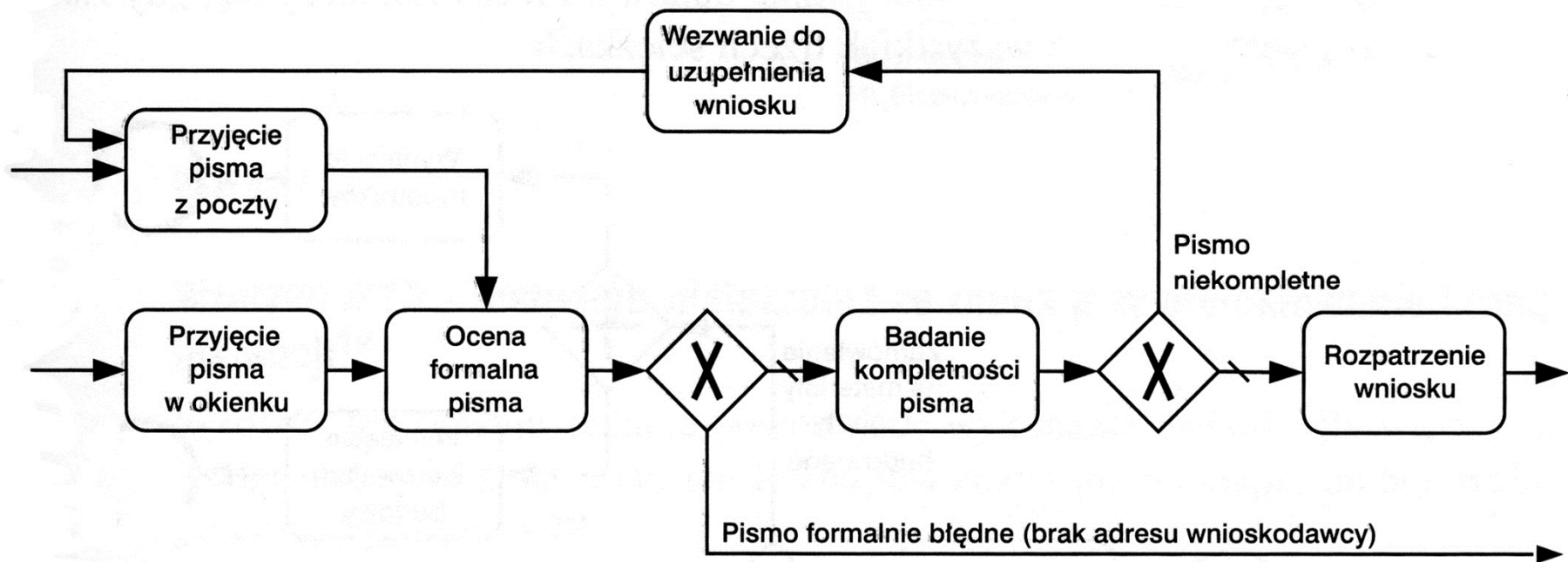
Wzorce procesowe w BPMN

- **Pętla** – iteracyjne powtarzanie zbioru czynności – warunek sprawdzany po wykonaniu czynności



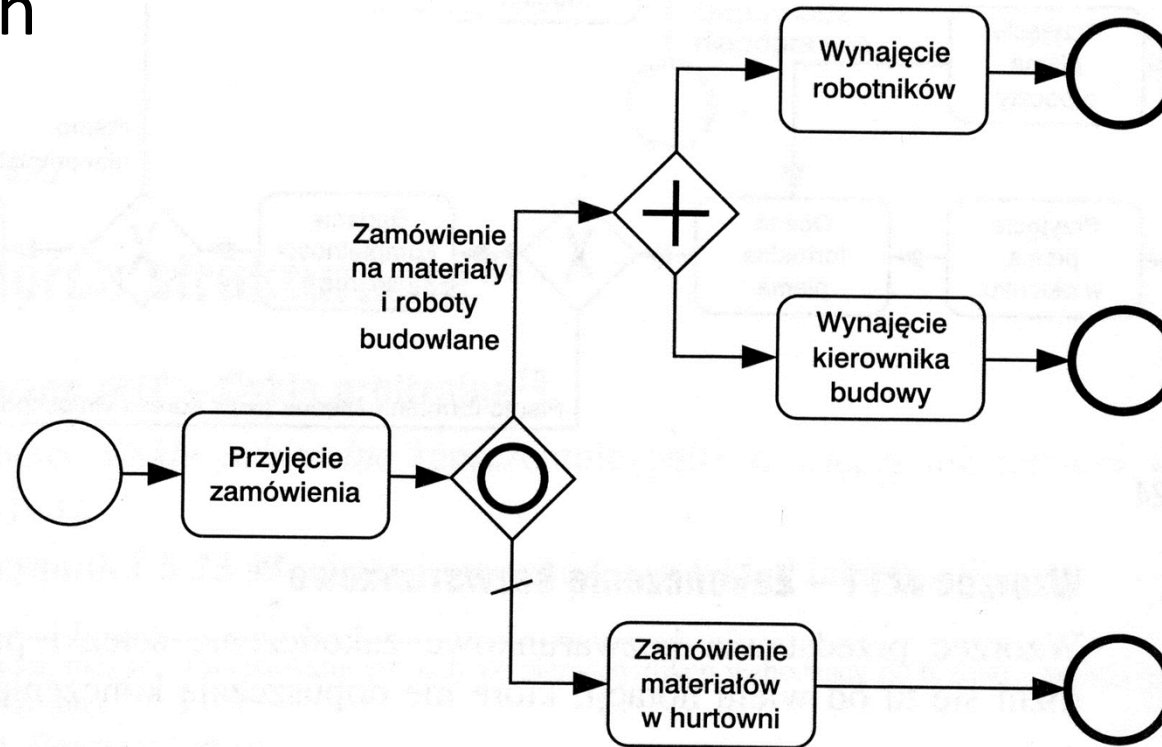
Wzorce procesowe w BPMN

- **Cykle arbitralne** – proces/podproces o więcej niż jednym wejściu i wyjściu



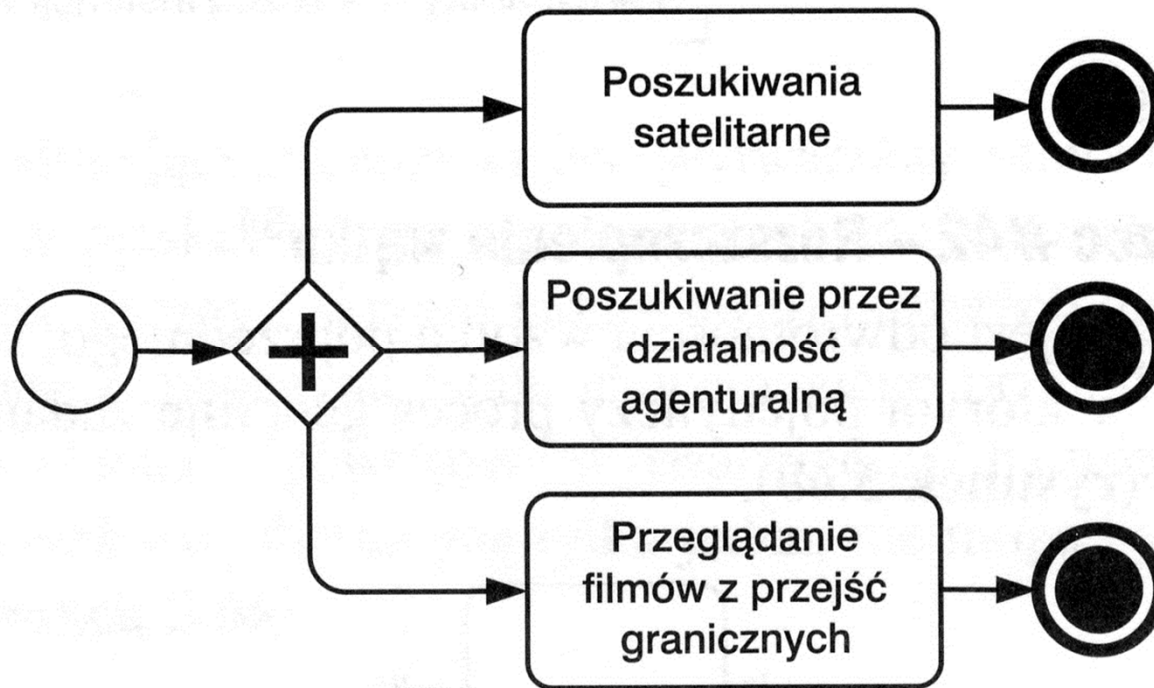
Wzorce procesowe w BPMN

- **Domyślne zakończenie procesu** – proces kończy się po osiągnięciu przez wszystkie wątki stanów końcowych



Wzorce procesowe w BPMN

- **Zakończenie kategoryczne** – proces kończy się po zakończeniu jednego z wątków



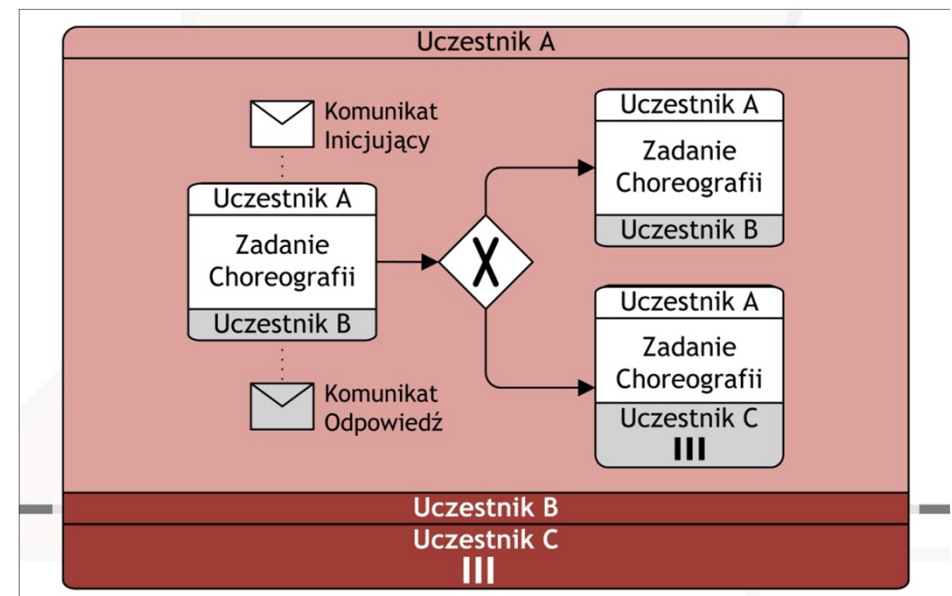
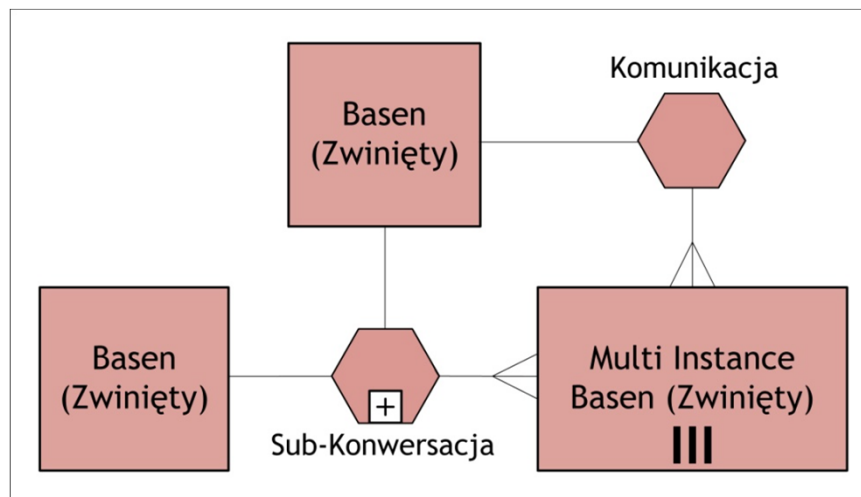
Wzorce procesowe w BPMN

Wzorce trudne/niemożliwe do modelowania w BPMN

- **Rekurencja** – definicja procesu zawiera odwołanie się do niego
- **Sekcja krytyczna** – w podprocesie znajduje się w danym momencie tylko jeden wątek

Nowa wersja standardu BPMN 2.0

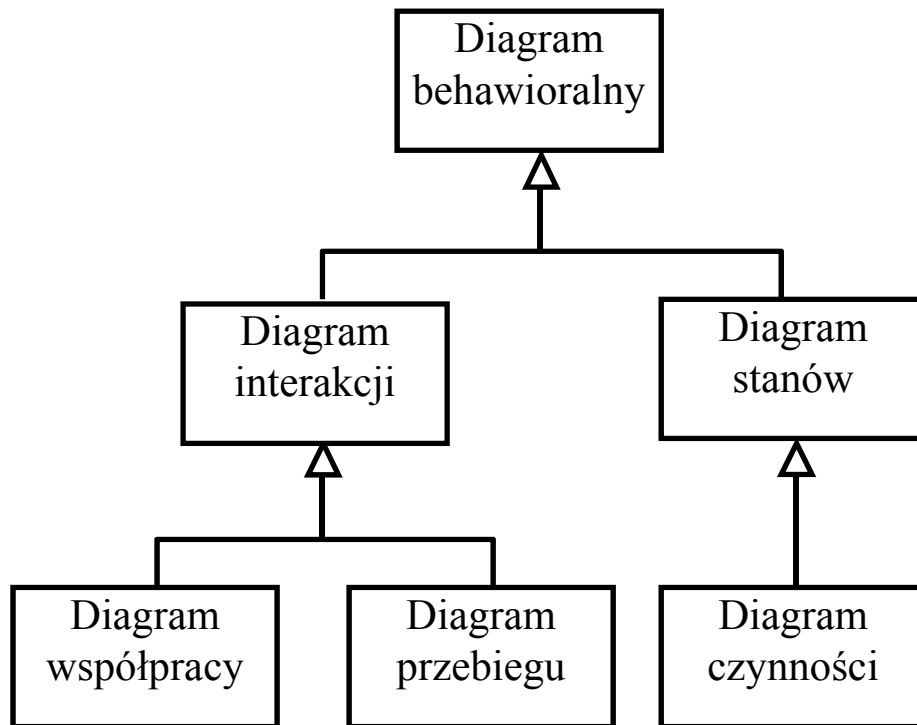
- Komitet standaryzacyjny pracuje na nową wersją standardu BPMN. W czerwcu 2010 ogłoszono roboczą wersję standardu BPMN 2.0.
- Główne rozszerzenia dotyczą modelowania procesów w architekturze choreografii oraz konwersacji.



Alternatywne notacje dla modelowania procesów

- Diagram czynności (Activity Diagram) UML – jeden z diagramów behawioralnych standardu UML (wersje 1.0 – 2.3, lata 1996 – 2010)
- Event-Driven Process Chain EPC – notacja oferowana przez narzędzie IDS/Scheer ARIS, opracowana przez Scheera, Kellera i Nuttgensa w 1991 roku w ramach metodyki Architecture of Integrated Information System (ARIS)

Modele behawioralne w UML*



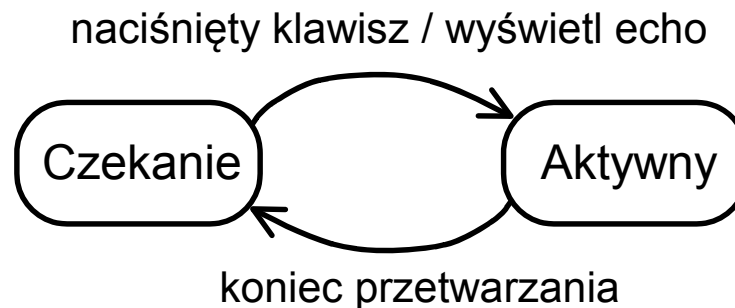
*) za wyjątkiem diagramu UseCase

- **Diagram interakcji** przedstawia interakcję między obiektami obejmującą przesyłanie komunikatów i przekazywanie wartości zwrotnych.
- **Diagram współpracy** obrazuje interakcje opartą o organizację strukturalną obiektów.
- **Diagram przebiegu** obrazuje kolejność przesyłania komunikatów w czasie.
- **Diagram stanów** obrazuje dynamikę obiektów, klas lub przypadków użycia jako maszyny stanów.
- **Diagram czynności** obrazuje przepływ sterowania. Jest szczególnym przypadkiem diagramu stanów, w którym stany odpowiadają wykonywanym operacjom, a zdarzenia reprezentują zakończenie operacji.

Diagram stanów

Diagramy stanów opisują:

- zbiory dopuszczalnych stanów w jakich mogą się znaleźć elementy diagramów statycznych;
- dopuszczalne przejścia między tymi stanami;
- zdarzenia, które powodują przejścia między stanami;
- akcje związane z poszczególnymi przejściami.



Zdarzenia

- *Zdarzenia* wyzwalają przejścia między stanami obiektu lub wykonywanie pewnych działań przez obiekt. Zdarzenia są specyfikowane w diagramie jako etykiety krawędzi modelujących przejścia między stanami.

Typ zdarzenia	Opis	Notacja
sygnał	Odebranie przez obiekt nazwanego, asynchronicznego sygnału <i>sname</i> , wysłanego przez inny lub ten sam obiekt.	<i>sname (a: T)</i>
wywołanie	Odebranie przez obiekt nazwanego, synchronicznego żądania wykonania operacji <i>op</i> . Obiekt wywołujący czeka na zakończenie wykonania operacji.	<i>op (a: T)</i>
zmiana	Zdarzenie polegające na wystąpieniu stanu opisanego przez wyrażenie logiczne <i>exp</i> .	when (<i>exp</i>)
zdarzenia czasowe	Wystąpienie określonego punktu czasu lub względny upływ czasu.	after (<i>time</i>)

Stany

Stany modelują pewne sytuacje występujące w trakcie życia obiektów związane:

- z oczekiwaniem obiektu na pewne zdarzenia,
- wykonywaniem przez obiekt pewnych operacji,
- spełnianiem pewnych warunków.


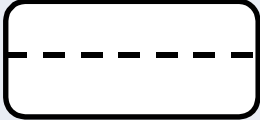
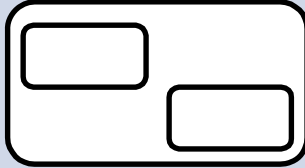


Specyfikacja *stanu* obejmuje:

- Akcje wejściowe - wykonywane przy wejściu w dany stan
- Akcje wyjściowe – wykonywane podczas opuszczania danego stanu
- Przejścia wewnętrzne – akcje wykonywane po wystąpieniu określonych zdarzeń, nie powodujące zmiany stanu
- Operacje wewnętrzne, działania wykonywane w czasie przebywania w danym stanie

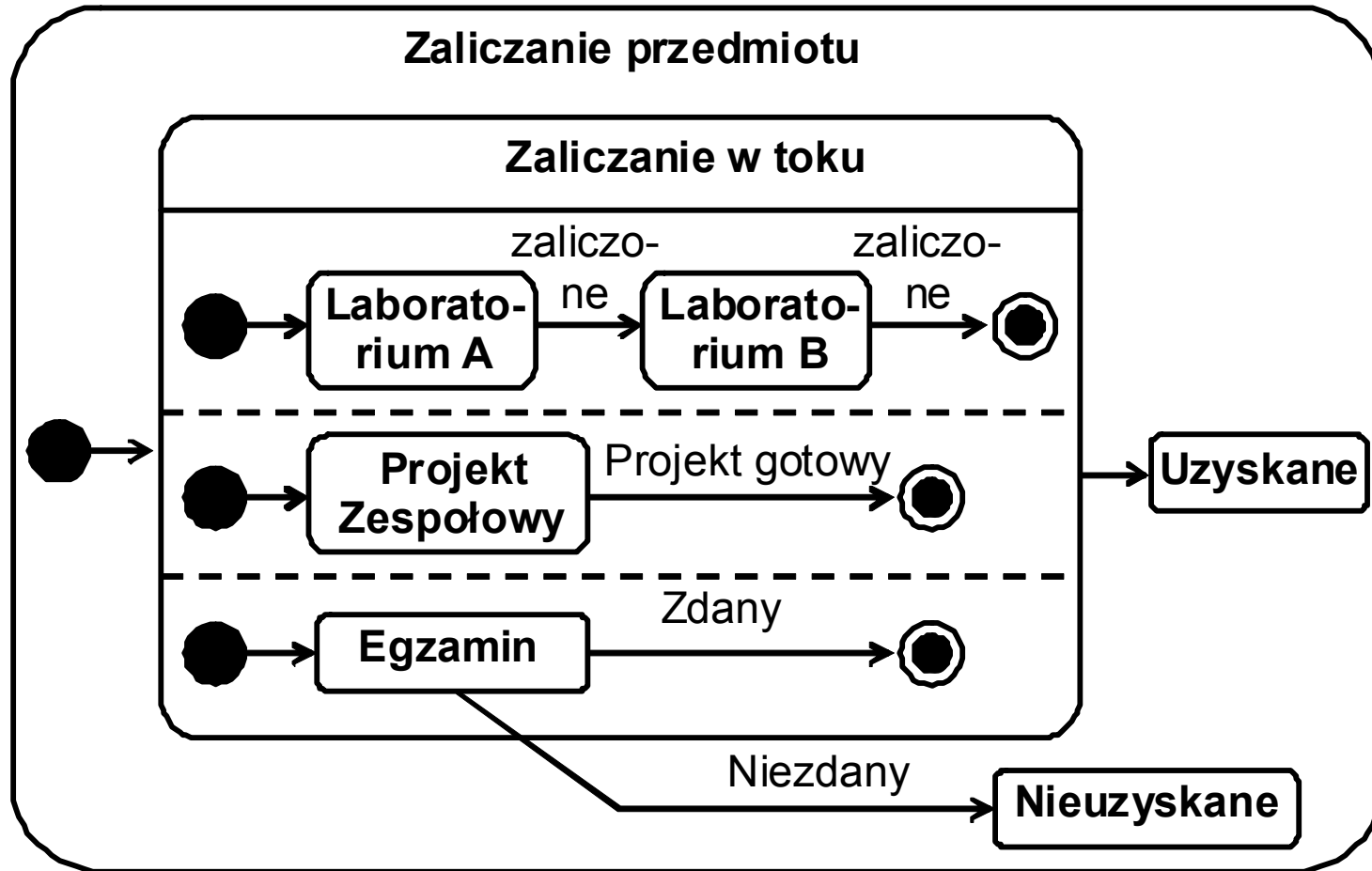
Wprowadzanie hasła

entry / hasło.zeruj()
exit / hasło.sprawdź()
cyfra **or** litera / wprowadź znak
zeruj / hasło.zeruj()
pomoc / wyświetl pomoc
wyświetl / **defer**
do / zablokuj echo

Rodzaje stanów

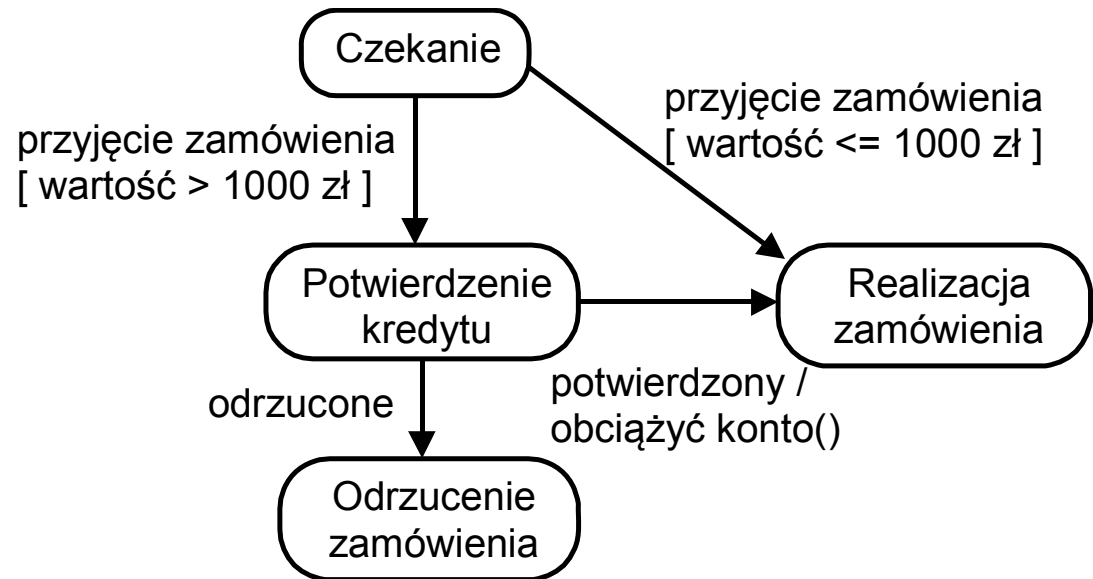
Typ stanu	Opis	Notacja
stan prosty	Stan bez struktury wewnętrznej.	
współbieżny stan złożony	Stan złożony znajdujący się równocześnie w kilku stanach składowych.	
sekwencyjny stan złożony	Stan złożony znajdujący zawsze w dokładnie jednym stanie składowym.	
stan początkowy	Pseudo-stan reprezentujący początkowy stan obiektu.	
stan końcowy	Wyróżniony stan reprezentujący końcowy stan obiektu.	

Przykład stanów złożonych



Przejścia

Przejścia modelują reakcje obiektu znajdującego się w danym stanie na wystąpienie określonych zdarzeń. Z przejściami mogą być związane dodatkowe akcje.



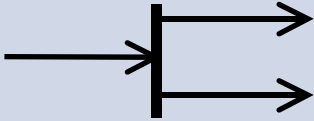
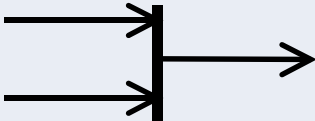
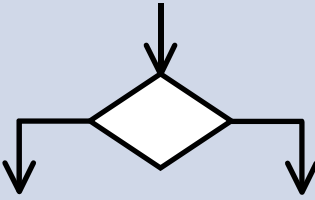
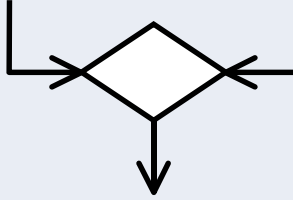
Elementy specyfikacji przejść

- **Stan źródłowy** – stan, w którym znajduje się obiekt przed operacją przejścia.
- **Zdarzenie wyzwajające** – zdarzenie, które wyzwala przejście.
- **Warunek** – wyrażenie logiczne, które musi być spełnione aby nastąpiło przejście.
- **Akcja** – niepodzielne przetwarzanie wykonywane opcjonalnie na stanie danego obiektu i pośrednio na innych dostępnych obiektach.
- **Stan docelowy** – stan osiągany w wyniku przejścia.

Rodzaje przejść i akcji

Typ przejścia	Opis	Składnia
Akcja wejściowa	Akcja związana z przechodzeniem do danego stanu.	entry/ <i>akcja</i>
Akcja wyjściowa	Akcja związana z wychodzeniem z danego stanu.	exit/ <i>akcja</i>
Przejście zewnętrzne	Przejście, które zmienia aktywny stan. Może być związane z realizacją dodatkowej akcji: <i>akcja</i> . Dodatkowo przejście może wyzwolić akcję wejściową nowego stanu lub wyjściową opuszczanego stanu. Przejście jest wynikiem wystąpienia zdarzenia $e(a:T)$ i spełnienia opcjonalnego warunku logicznego <i>exp</i> .	$e(a:T)[exp]/$ <i>akcja</i>
Przejście wewnętrzne	Przejście, które powoduje uaktywnienie związanej z nim akcji <i>akcja</i> , ale nie zmienia stanu obiektu.	$e(a:T)[exp]/$ <i>akcja</i>

Złożona topologia przejść

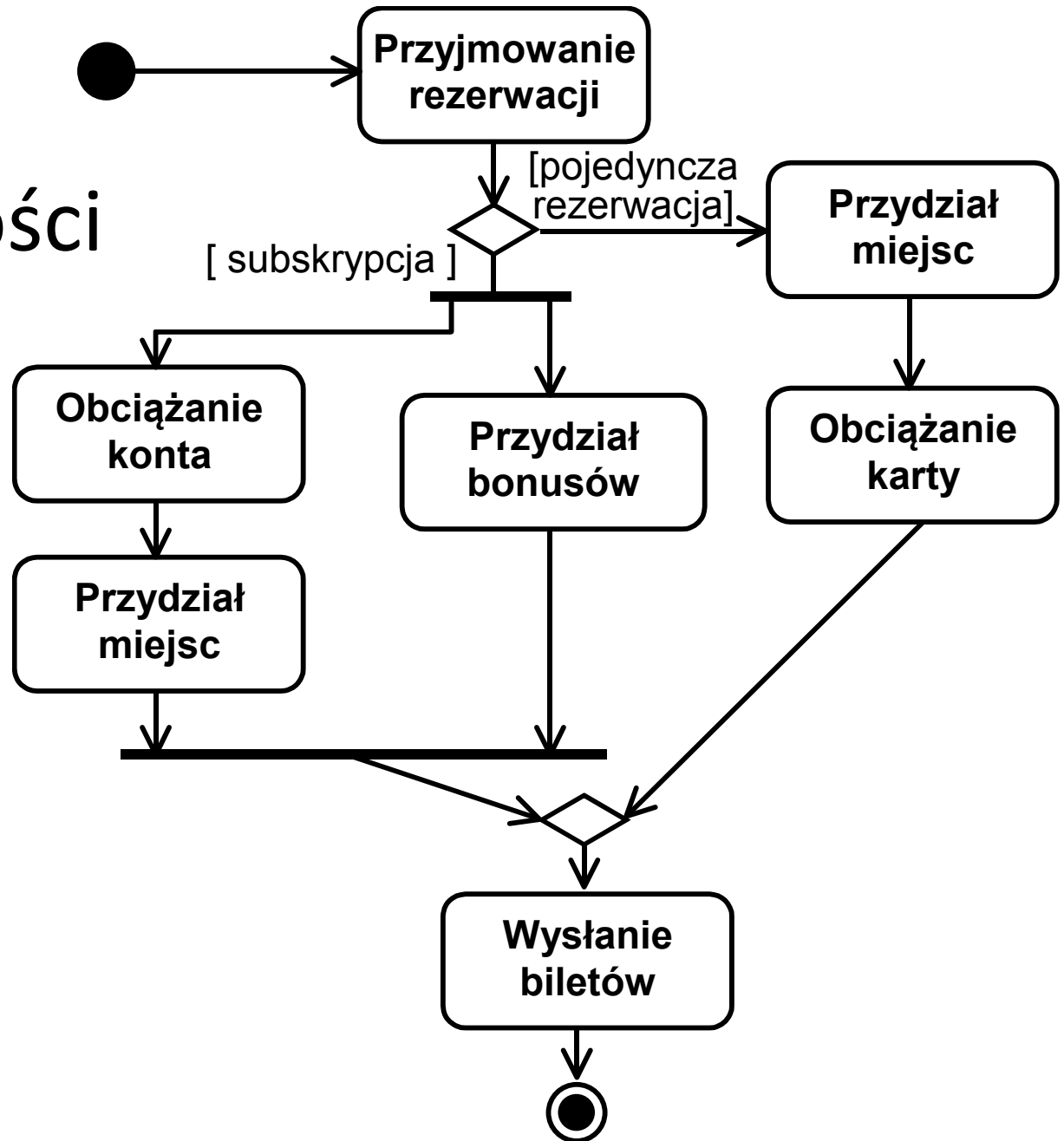
Złożony typ przejścia	Symbol
Zrównoleglenie procesów (fork)	
Połączenie procesów (join)	
Rozgałęzienie alternatywnych przepływów sterowania	
Połączenie alternatywnych przepływów sterowania	

Diagramy czynności

Diagramy czynności są szczególnym przypadkiem diagramów stanu, służącym do modelowania procesów lub przepływu sterowania:

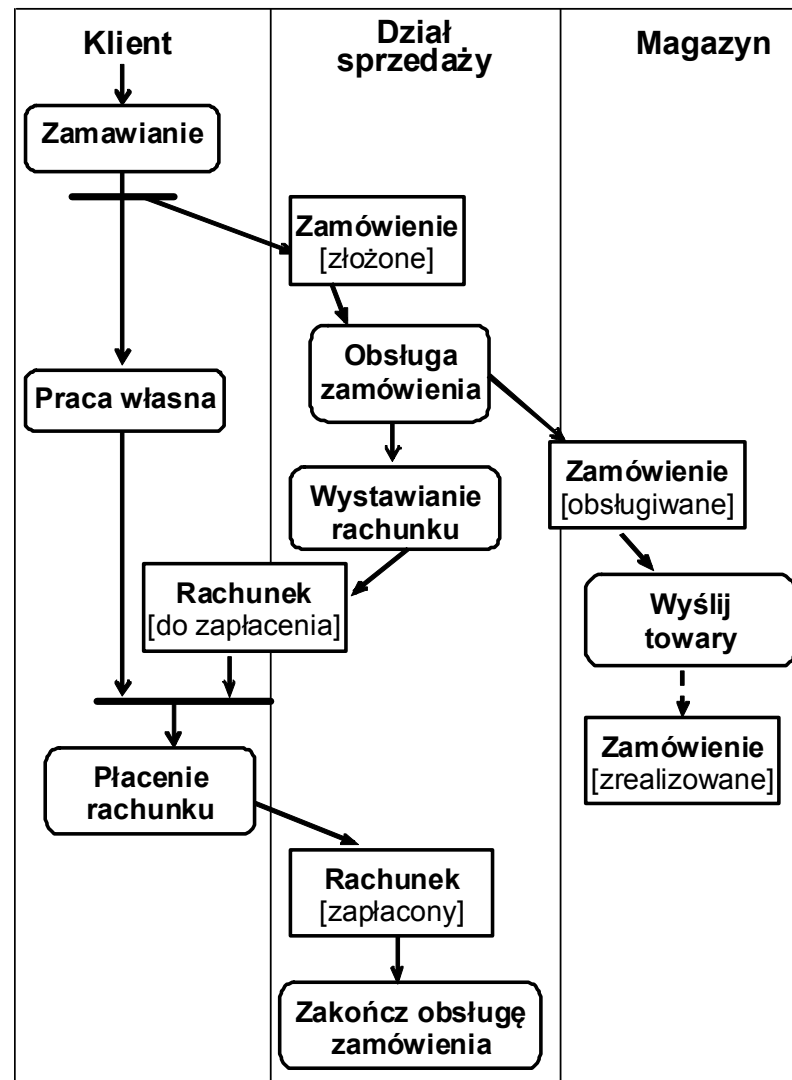
- stany ograniczone są do czasu wykonywania akcji
- przejścia są związane z zakończeniem akcji
- dla poszczególnych akcji istotne jest miejsce ich wykonania lub ich wykonawca
- Krawędzie przejść mogą mieć przypisane dane w postaci obiektów, przepływające między poszczególnymi akcjami stanów

Przykład diagramu czynności



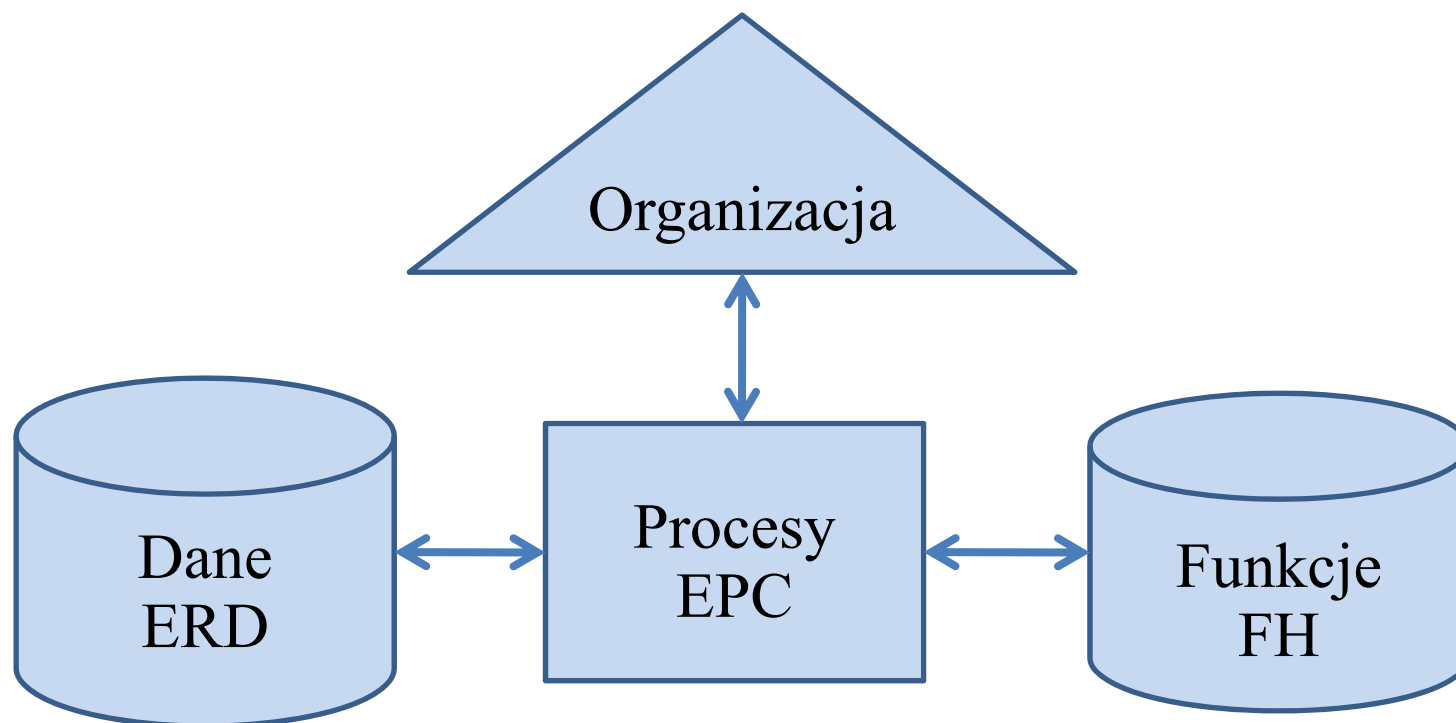
Wykonawcy czynności

Diagramy czynności oferują specjalną notację do opisu wykonawcy czynności. Stany/czynności diagramy są umieszczane w *torach basenów*.

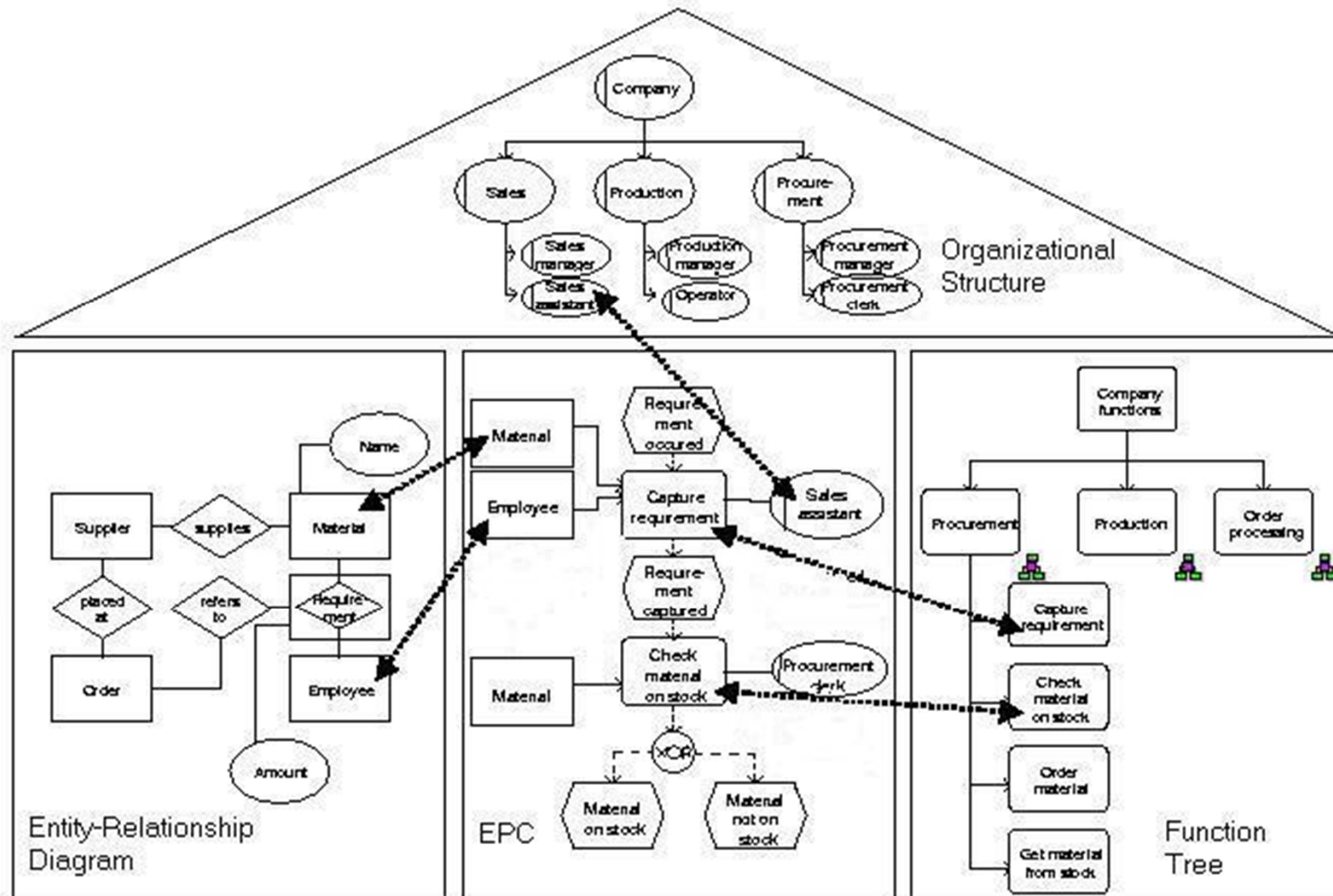


Event-Driven Process Chain

- Model procesów jest elementem perspektywy procesowej (Control View), która integruje przewidziane w systemie Aris perspektywy: organizacyjną, funkcyjną i danych.



ARIS - Integracja perspektyw



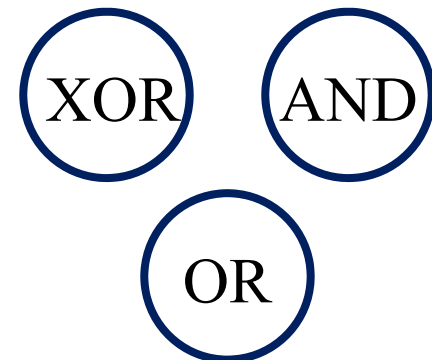
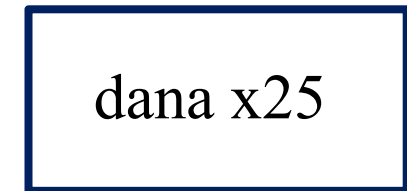
Elementy notacji EPC

- **Funkcje** – modelują zadania i podprocesy wykonywane w danej firmie.
- **Zdarzenia** – opisują warunki wejściowe wymagane do uaktywnienia funkcji lub warunki końcowe wykonania funkcji
- **Jednostki organizacyjne** – osoby, stanowiska lub działy ze struktury organizacyjnej firmy, odpowiedzialne za wykonanie określonych funkcji



Elementy notacji EPC

- **Dane** – modelują dane wejściowe lub wyjściowe funkcji odpowiadające encjom z ERD
- **Bramki logiczne** – umożliwiają modelowanie procesów o złożonym przepływie sterowania
- **Ścieżki procesów** – sekwencyjne powiązania między procesami; zakończenie jednego procesu powoduje automatyczne rozpoczęcie następnego procesu



Elementy notacji EPC

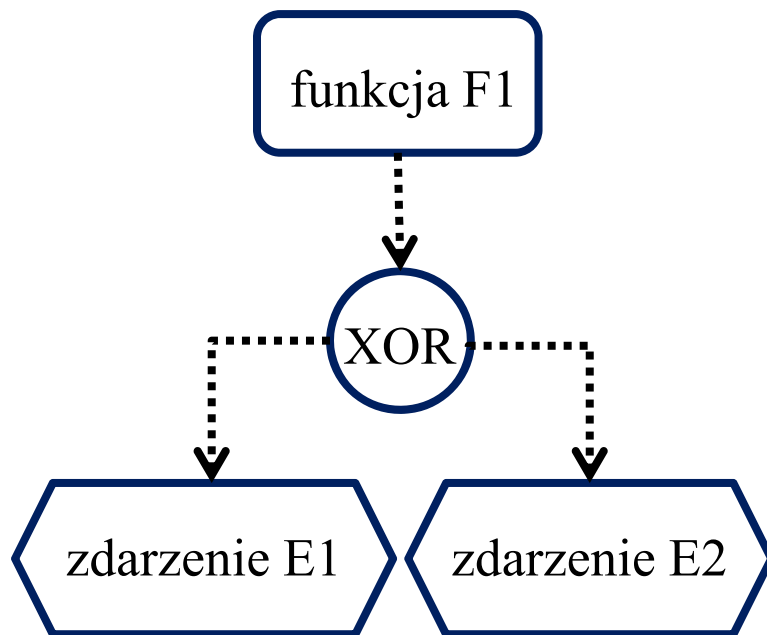
- **Przepływ sterowania** – łączy zdarzenia z funkcjami (ścieżkami procesów) i bramkami logicznymi łącząc je w chronologiczne sekwencje
- **Przepływ danych** – reprezentuje powiązania między funkcjami, a danymi wejściowymi lub wyjściowymi
- **Przypisanie wykonawcy** – reprezentuje związek między funkcją, a odpowiedzialną za nią jednostką organizacyjną



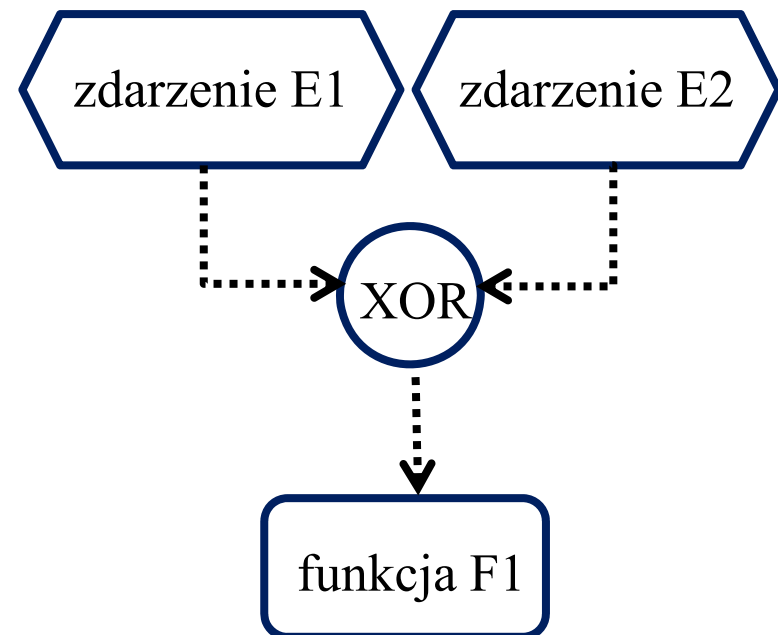
Złożona topologia przepływów

Rozgałęzianie procesu

- Wykonanie funkcji F1 powoduje wystąpienie stanu E1 albo stanu E2

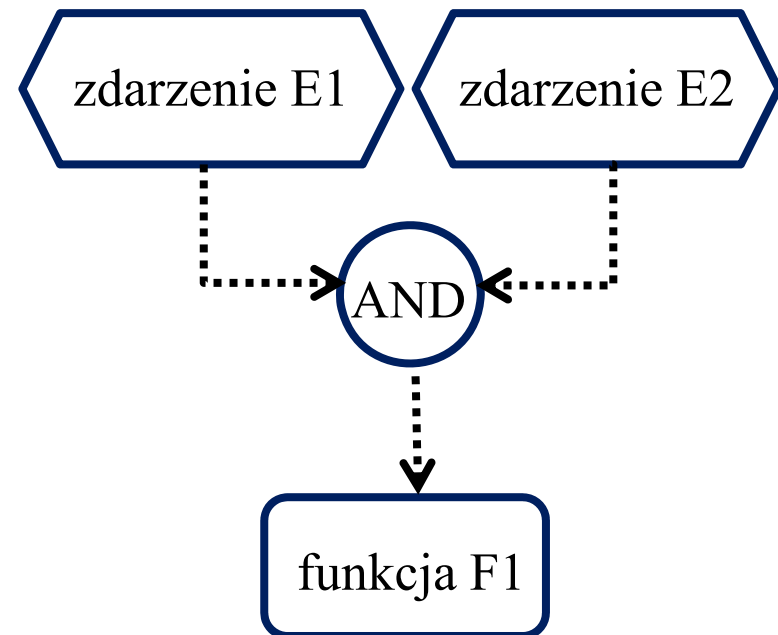
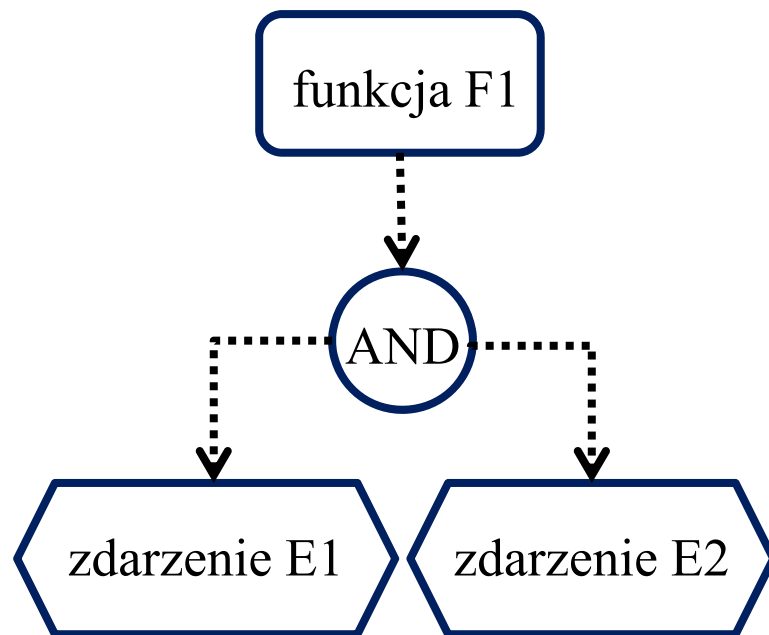


- Wystąpienie stanu E1 albo stanu E2 powoduje wykonanie funkcji F1



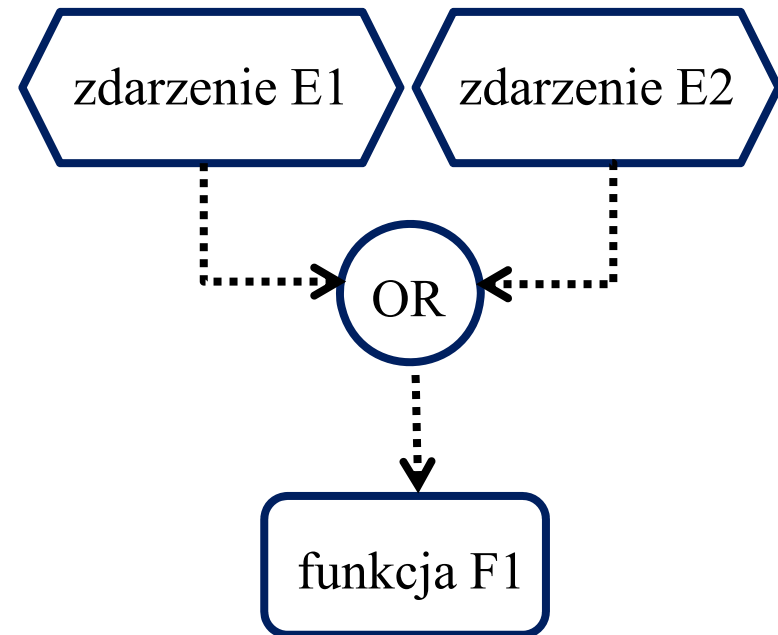
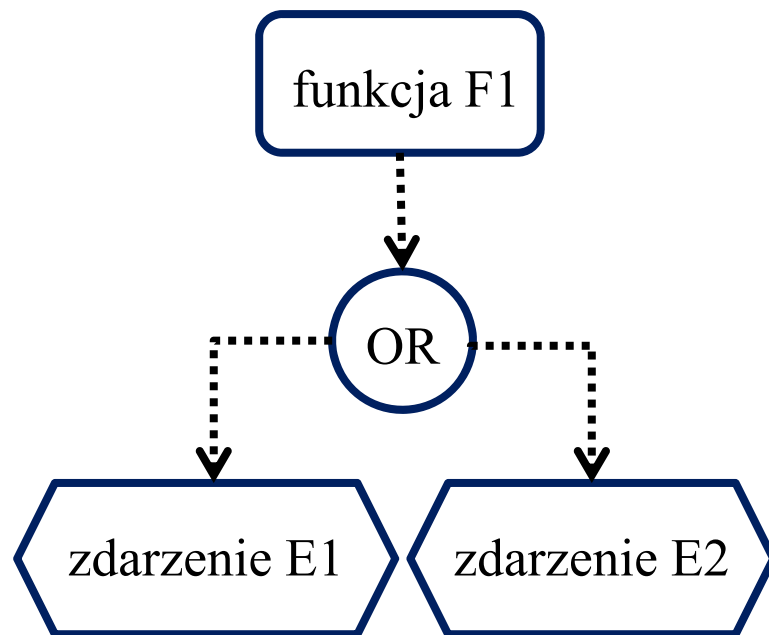
Zrównoleglenie procesu

- Zakończenie wykonania funkcji F1 powoduje równoczesne wystąpienie stanu E1 i stanu E2
- Jeżeli równocześnie wystąpią stany E1 i stanu E2, spowoduje to rozpoczęcie wykonywanie funkcji F1



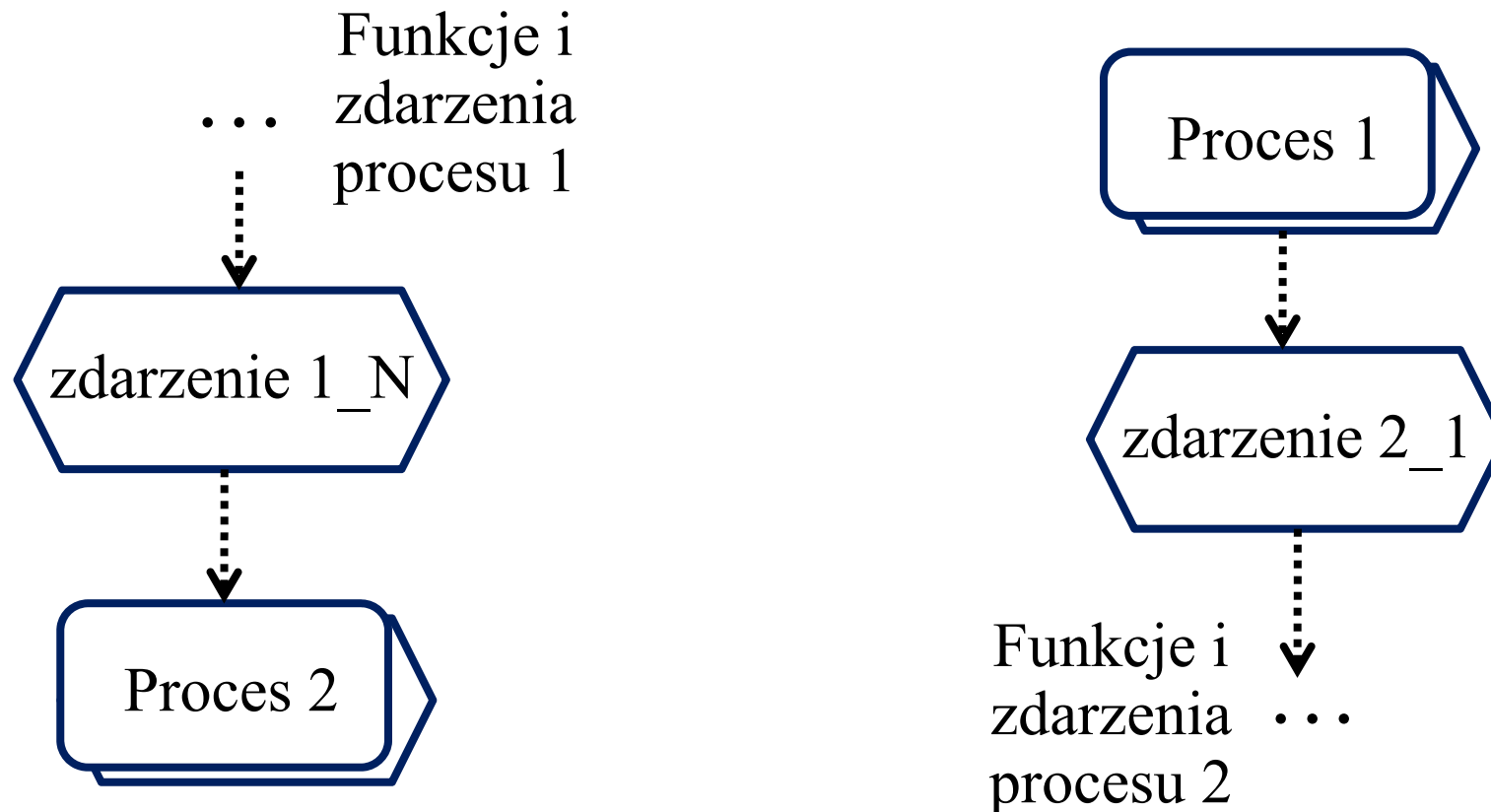
Opcjonalne zrównoleglenie procesu

- Zakończenie wykonania funkcji F1 powoduje wystąpienie stanu E1 albo stanu E2 albo obydwu stanów naraz
- Rozpoczęcie wykonania funkcji F1 jest spowodowane wystąpieniem stanu E1 albo stanu E2 albo obydwu stanów naraz



Ścieżki procesów

- Zakończenie procesu 1 powoduje rozpoczęcie procesu 2



Formalne reguły poprawności EPC

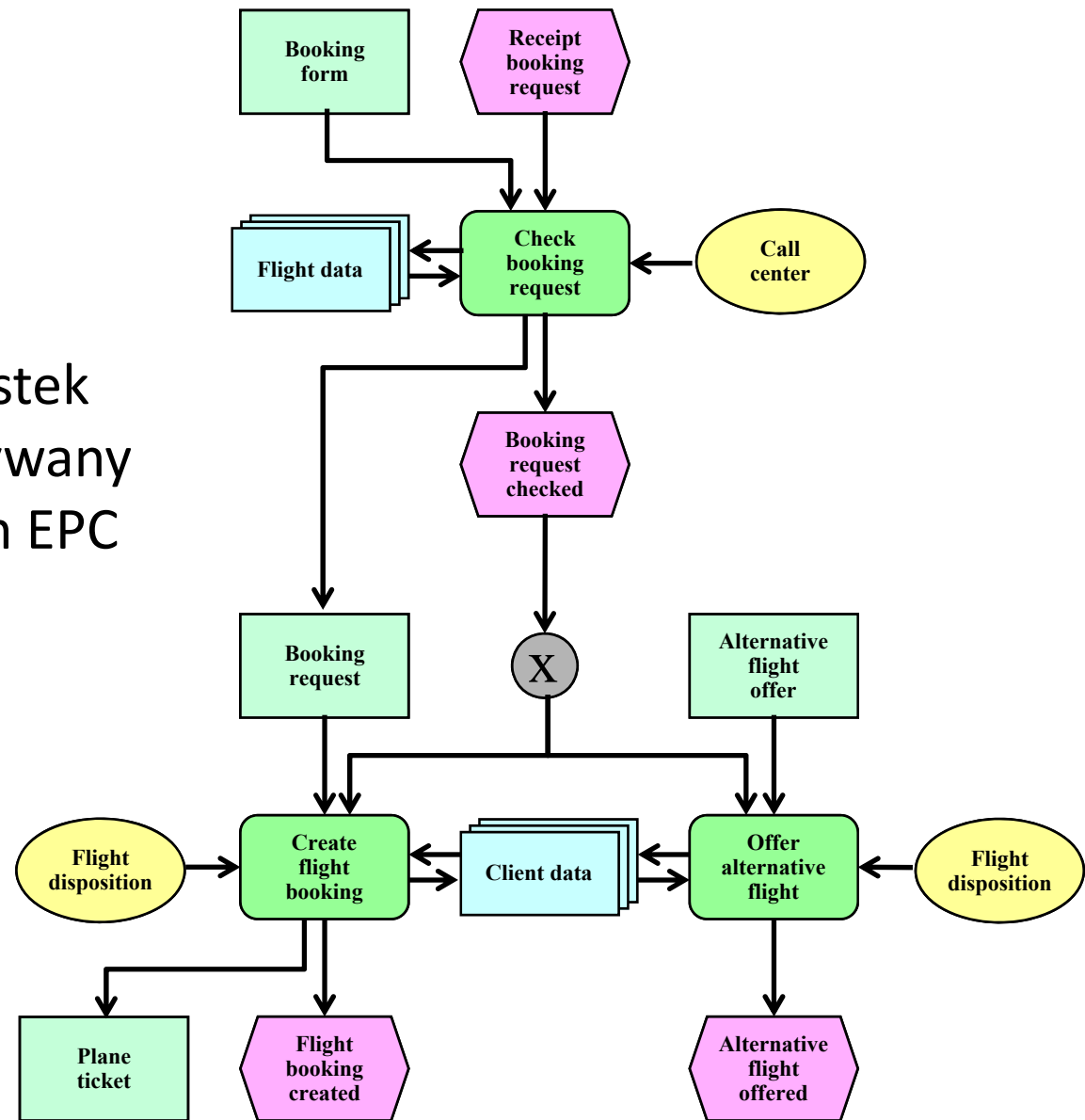
- W grafie, w którym krawędziami są przepływy sterowania, węzłami mogą być jedynie: funkcje (w tym również ścieżki procesów), zdarzenia i bramki.
- Zdarzenia nie mogą być łączone bezpośrednio z innymi zdarzeniami. Dla prostej sekwencji funkcje dozwolone jest pominięcie zdarzeń trywialnych i w konsekwencji bezpośrednie połączenie jednej funkcji z drugą.
- Zdarzenia są elementami pasywnymi i nie mogą pojawiać się na wejściach bramek decyzyjnych XOR lub OR.
- Początkiem każdego procesu jest co najmniej jedno zdarzenie początkowe, a końcem co najmniej jedno zdarzenie końcowe.

Formalne reguły poprawności EPC

	Zdarzenie na wejściu bramki		Funkcja na wejściu bramki	
	Pojedyncza	Wielokrotna	Pojedyncza	Wielokrotna
AND	<p>Diagram showing a single Event (pink hexagon) connected to an AND gate symbol (circle with ^). Two dashed arrows lead from the gate to two Function (green rounded rectangles) outputs.</p>	<p>Diagram showing two Event (pink hexagons) connected to an AND gate symbol (circle with ^). A dashed arrow leads from the gate to one Function (green rounded rectangle) output.</p>	<p>Diagram showing a single Function (green rounded rectangle) connected to an AND gate symbol (circle with ^). Two dashed arrows lead from the gate to two Event (pink hexagons) outputs.</p>	<p>Diagram showing two Function (green rounded rectangles) connected to an AND gate symbol (circle with ^). A dashed arrow leads from the gate to one Event (pink hexagon) output.</p>
OR	Nie- dozwolone	<p>Diagram showing two Event (pink hexagons) connected to an OR gate symbol (circle with v). A dashed arrow leads from the gate to one Function (green rounded rectangle) output.</p>	<p>Diagram showing a single Function (green rounded rectangle) connected to an OR gate symbol (circle with v). Two dashed arrows lead from the gate to two Event (pink hexagons) outputs.</p>	<p>Diagram showing two Function (green rounded rectangles) connected to an OR gate symbol (circle with v). A dashed arrow leads from the gate to one Event (pink hexagon) output.</p>
XOR	Nie- dozwolone	<p>Diagram showing two Event (pink hexagons) connected to an XOR gate symbol (circle with x). A dashed arrow leads from the gate to one Function (green rounded rectangle) output.</p>	<p>Diagram showing a single Function (green rounded rectangle) connected to an XOR gate symbol (circle with x). Two dashed arrows lead from the gate to two Event (pink hexagons) outputs.</p>	<p>Diagram showing two Function (green rounded rectangles) connected to an XOR gate symbol (circle with x). A dashed arrow leads from the gate to one Event (pink hexagon) output.</p>

Diagram eEPC

- Diagram rozszerzony o elementy danych i jednostek organizacyjnych jest nazywany rozszerzonym diagramem EPC



Porównanie notacji

- **Kontekst** – notacje EPC i UML AD są częścią bardziej ogólnych notacji, w których modelowanie procesów jest tylko fragmentem (AD drobnym fragmentem) innych aspektów modelowanej rzeczywistości. Notacja BPMN jest dedykowana tylko i wyłącznie do modelowania procesów. Pojęcie procesu jest podstawowym elementem tej notacji.

Porównanie notacji

Semantyka – notacja BPMN bogatsza od notacji EPC i AD UML o:

- predefiniowane typy zdarzeń
- więcej typów bramek,
- predefiniowane typy zadań,
- obsługę wyjątków,
- mniejsze ograniczenia, co do topologii przepływów pracy w procesach,
- dodatkowe niegraficzne atrybuty wszystkich elementów notacji,
- specyfikację transformacji do języka BPEL.

Porównanie notacji

- **Precyzja/jednoznaczność** – żadna z trzech prezentowanych notacji nie gwarantuje pełnej precyzji i jednoznaczności modelowania procesów dla wielu mniej typowych wzorców przepływu sterowania w procesach. Nie umożliwiają one jednoznacznego modelowania interakcji między wystąpieniami tego samego procesu i formalnej analizy wielu wymaganych cech procesów (żywoćność, ograniczenia, zakleszczenia, itp.) oraz analizy wydajnościowej. Analiza taka jest możliwa dopiero po transformacji modeli do sieci Petriego.

Porównanie notacji

- **Rozwój notacji** – notacje EPC i UML AD nie są od dawna rozwijane (EPC nie zmieniła się od momentu powstania). Notacja BPMN jest najnowsza i ciągle rozwijana. Dostępna jest już kolejnej wersja notacji 2.0 od 3-I-2011.

Metodyki modelowania procesów

- Top down
- Bottom up
- *Postaci normalne* modelu procesów