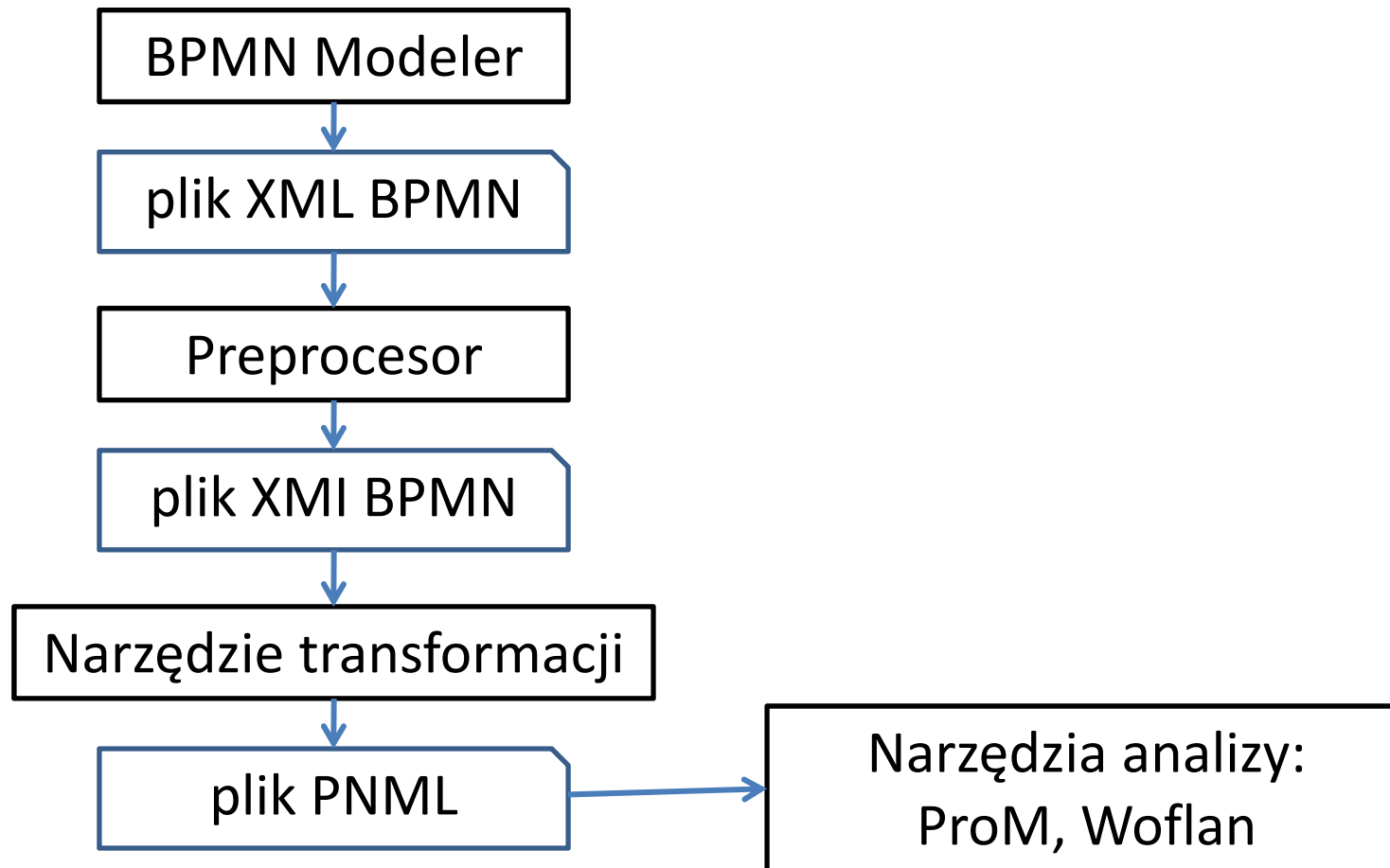


Odwzorowanie BPMN w sieć Petriego

Proces odwzorowania

- Scenariusz procesu odwzorowania BPMN2PN



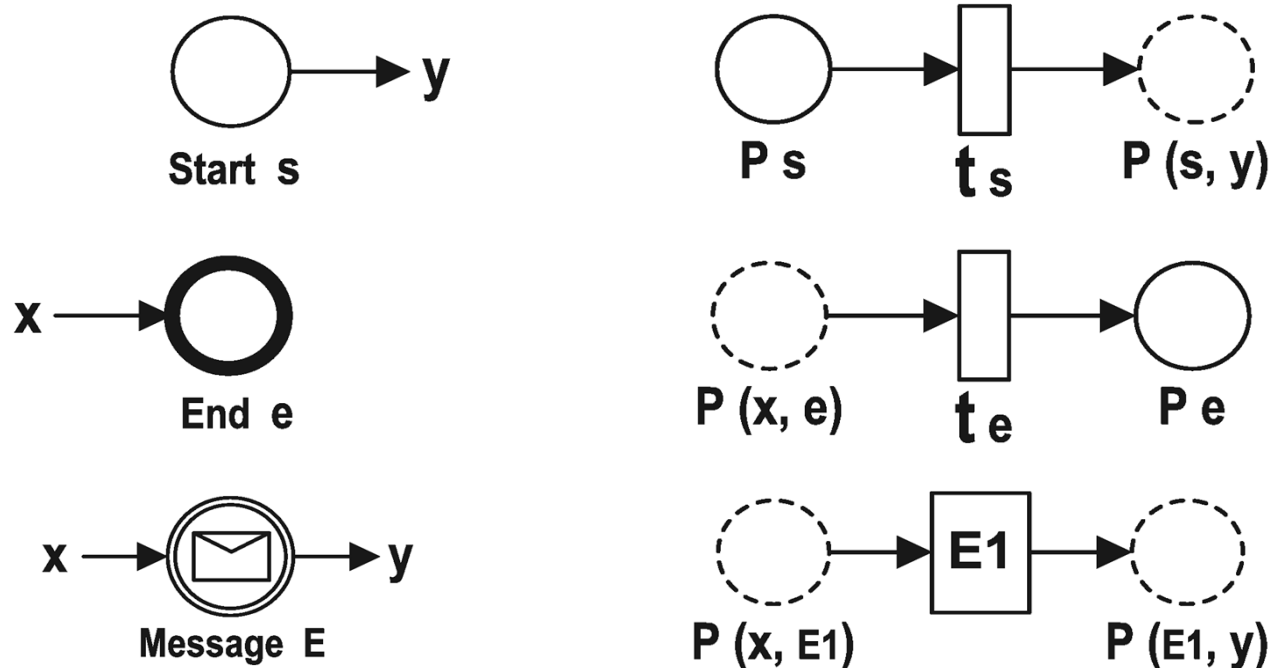
Warunki poprawnego odwzorowania

Odwzorowanie diagramów BPMN w sieć Petriego zakłada spełnienie określonych warunków:

- Zdarzenia początkowe i zdarzenia wyjątków muszą mieć dokładnie jeden łuk wyjściowy
- Zdarzenia końcowe muszą mieć dokładnie jeden łuk wejściowy
- Działania i zdarzenia pośrednie (poza wyjątkami) muszą mieć dokładnie jeden łuk wejściowy i jeden wyjściowy
- Bramki rozwidlające muszą mieć dokładnie jeden łuk wejściowy i więcej niż jeden wyjściowy
- Bramki synchronizujące muszą mieć dokładnie jeden łuk wyjściowy i więcej niż jeden wejściowy
- Łuki wyjściowe bramek sterowanych zdarzeniami muszą być dołączone do zdarzeń pośrednich typu: message lub timer, lub do zadania typu receive.
- Każdy element diagramu znajduje się na ścieżce między zdarzeniem początkowym lub zdarzeniem wyjątku, a zdarzeniem końcowym

Zdarzenia

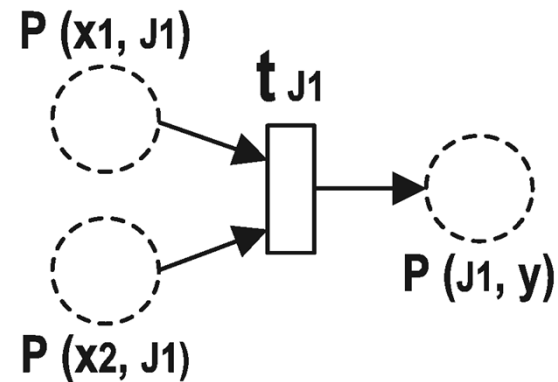
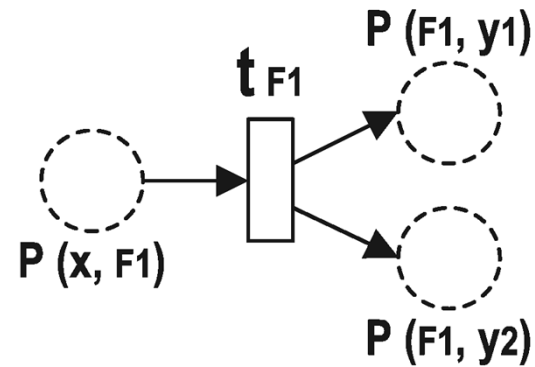
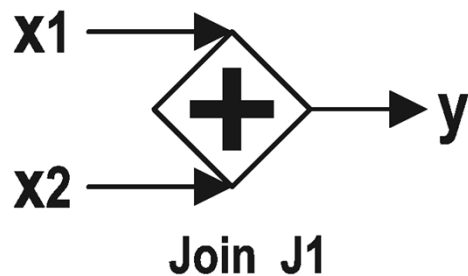
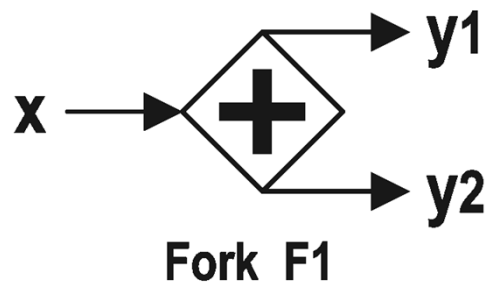
- Zdarzenie należy odwzorowywać jako odpowiednie kombinacje przejść i miejsc:



- Symbole x , x_1 , x_2 reprezentują obiekty wejściowe, a y , y_1 i y_2 obiekty wyjściowe. Miejsca narysowane linią przerywaną nie będą unikalne w pojedynczym module odwzorowania. Będą służyły do łączenia modułów reprezentujących poszczególne obiekty BPMN w spójną sieć Petriego.

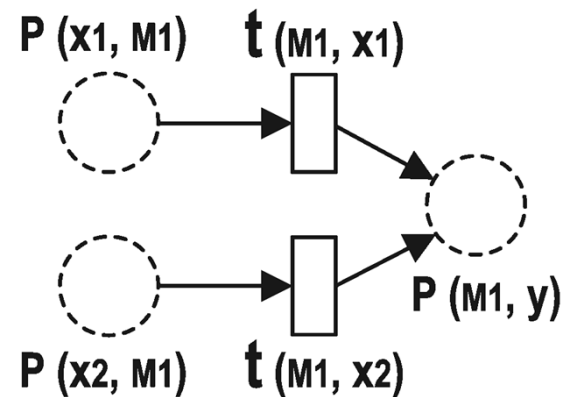
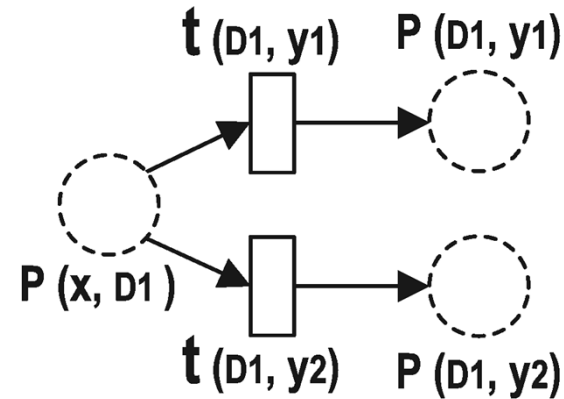
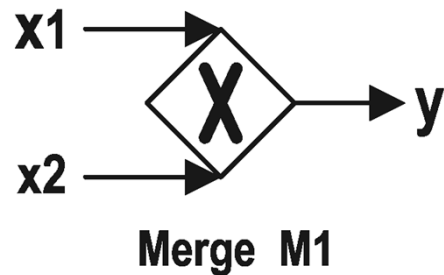
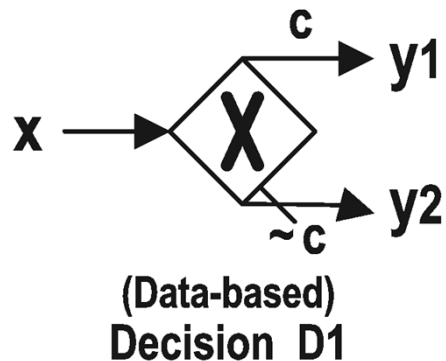
Bramki AND

- Bramki AND rozszczepiające i synchronizujące są odwzorowywane w pojedyncze przejścia zbiory miejsc, które będą łącznikami do innych modułów.



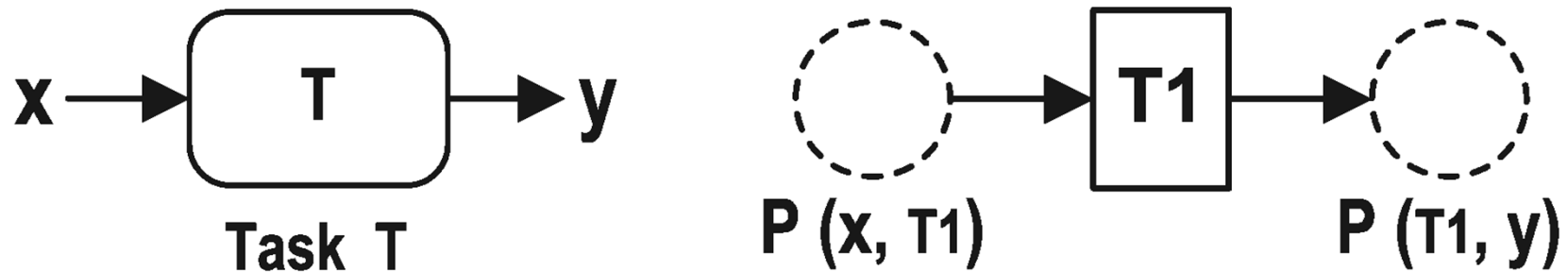
Bramki XOR

- Bramki XOR są odwzorowywane w zbiory przejść oraz miejsca, które będą służyły jako łączniki modułów



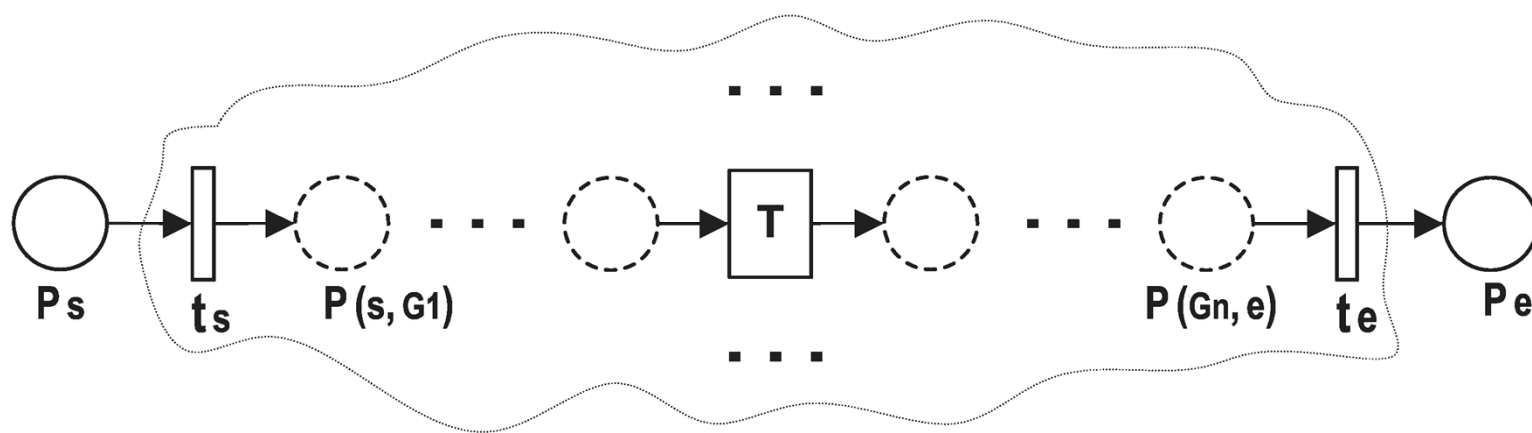
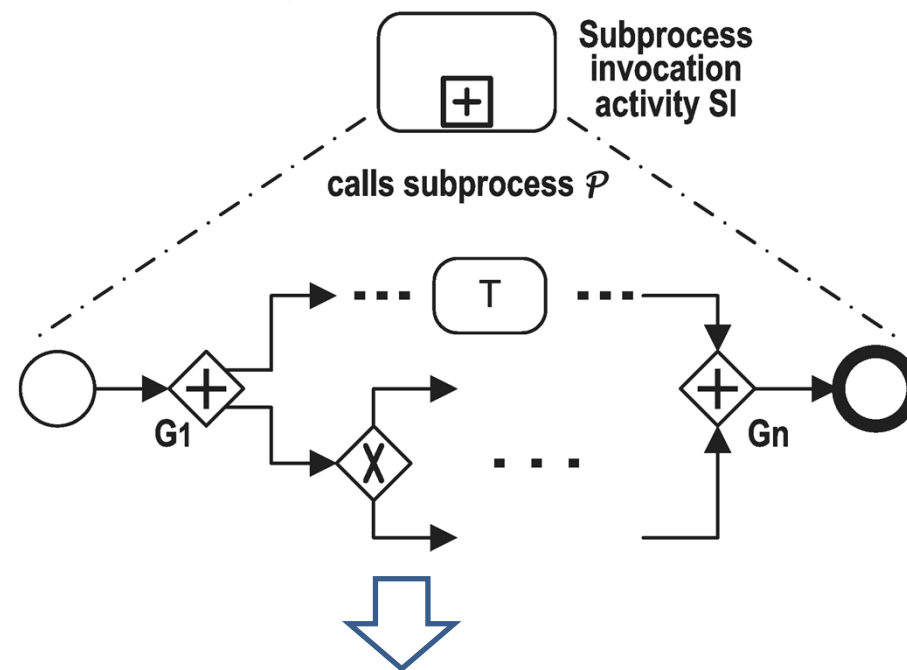
Zadania

- Zadania są odwzorowywane w pojedyncze przejścia oraz miejsca wejściowe i wyjściowe, które będą łącznikami do innych modułów.



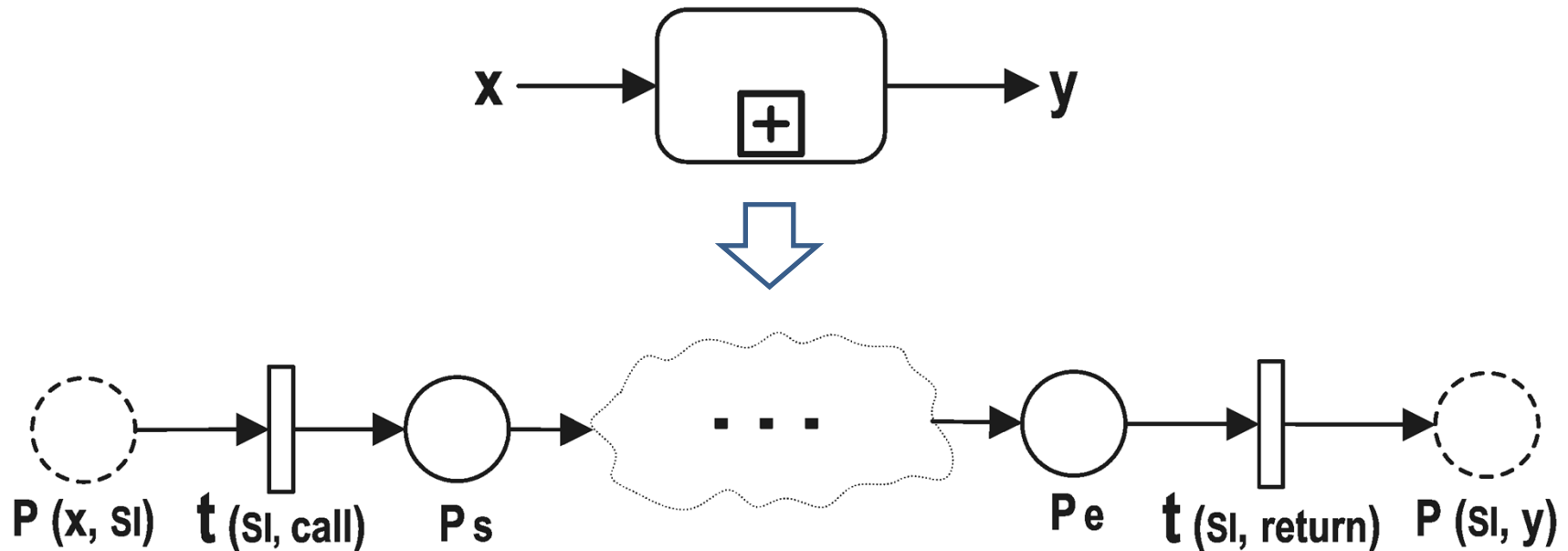
Podprocesy

- Podprocesy są odwzorowywane analogicznie jak niezależne procesy, za wyjątkiem obsługi błędów



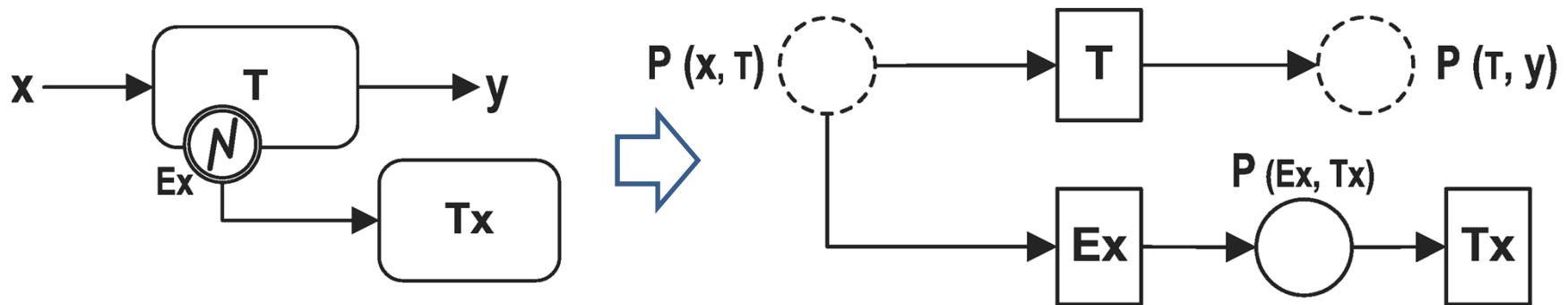
Podprocesy

- Wywołanie podprocesu, wymaga dodania dwóch przejść, z których $t(SI, call)$ reprezentuje wywołanie podprocesu przez proces zewnętrzny, a $t(SI, return)$, powrót pracy do procesu głównego.



Obsługa błędów

- Ponieważ zadanie T jest atomowe, odwzorowanie będzie generowało moduł sieci, w której przejścia reprezentujące wykonanie zadania i zdarzenia wyłapania wyjątku będą rywalizowały o ten sam żeton w miejscu wejściowym $P(x, T)$.

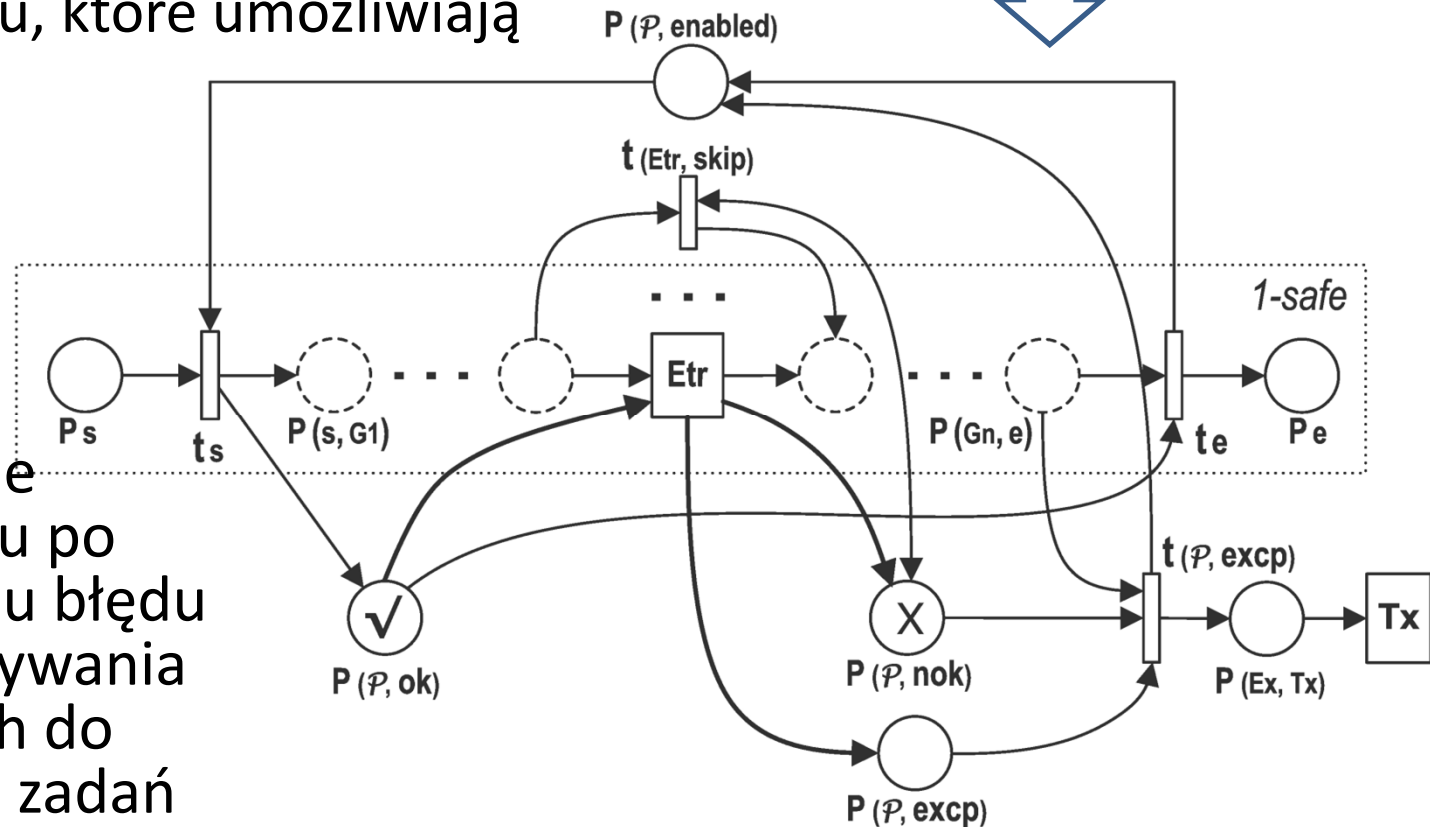
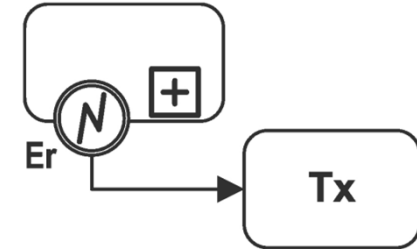


Obsługa błędów podprocesów

Potrzeba zamodelowania zakończenia wykonywania podprocesu:

- odkurzacz żetonów
- dodatkowe miejsca reprezentujące status podprocesu, które umożliwiają

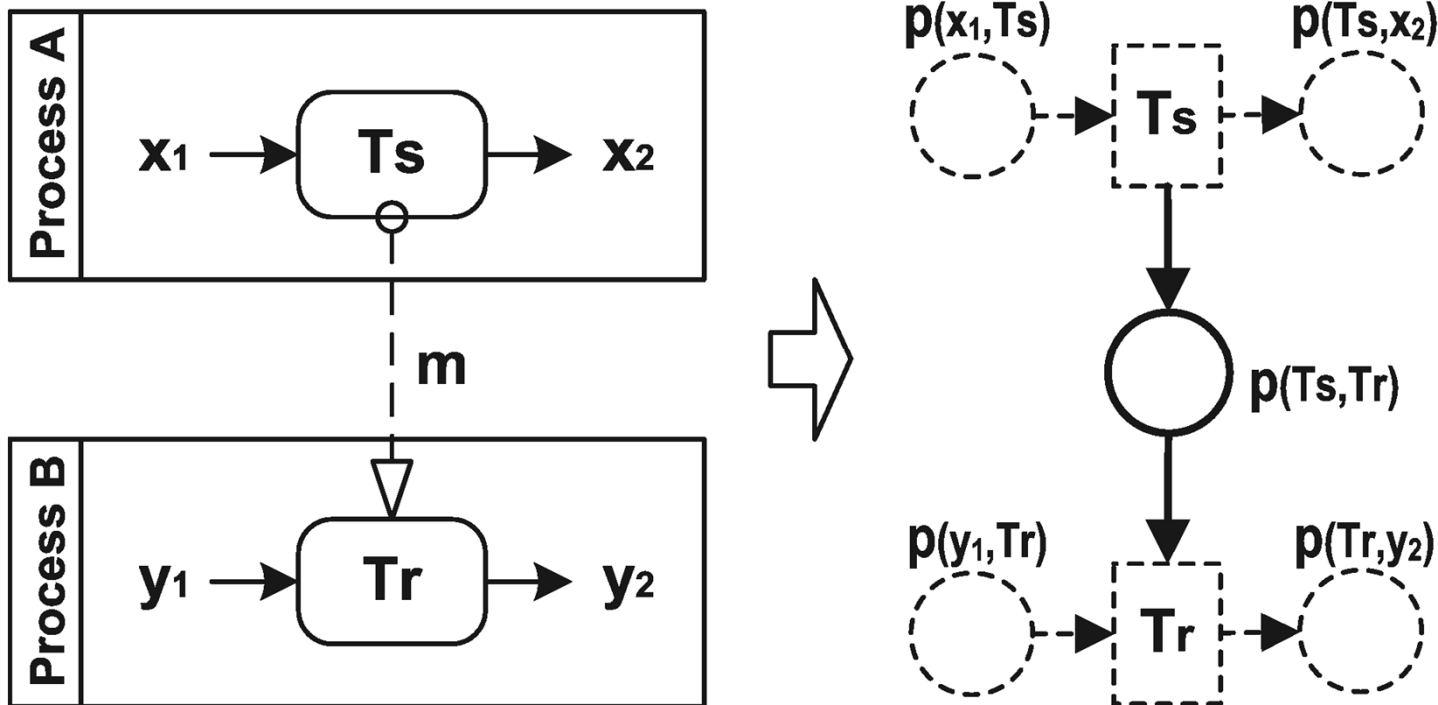
Call subprocess \mathcal{P}



zakończenie podprocesu po wystąpieniu błędu bez wykonywania pozostałych do wykonania zadań

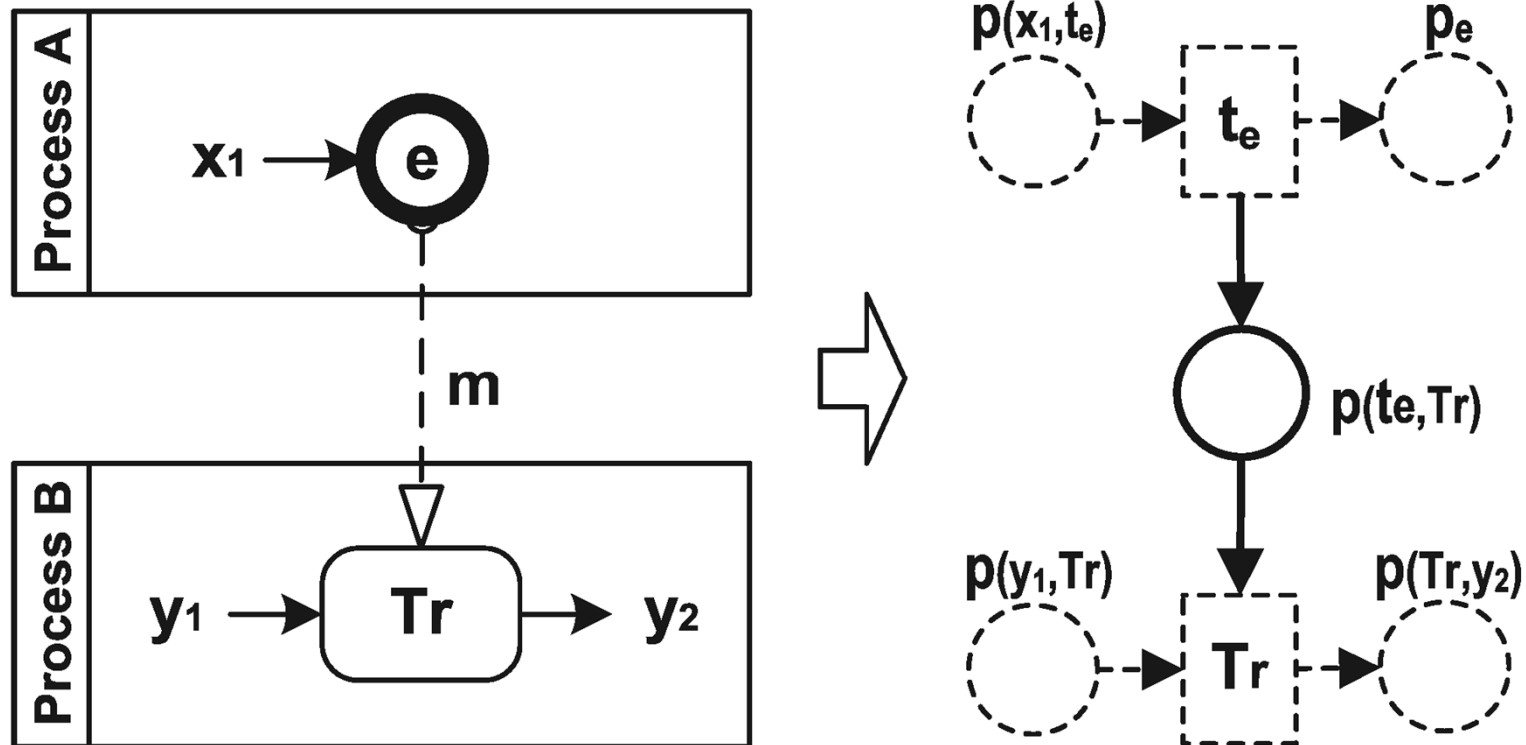
Sygnały

- Odwzorowanie przekazywania sygnałów z zadania jednego procesu do zadania innego procesu. Pojedyncze zadanie może tylko odbierać albo tylko wysyłać sygnały (odpalanie przejść jest atomowe).



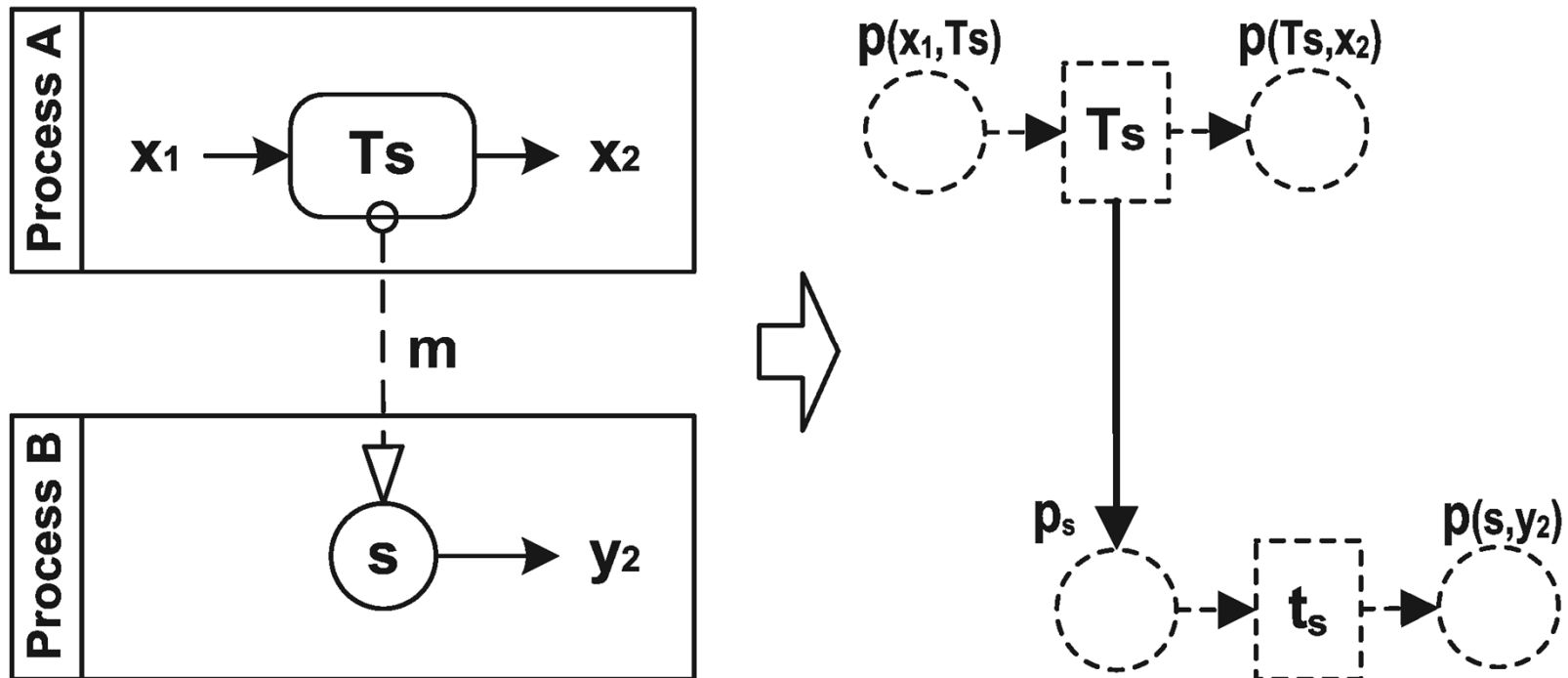
Sygnaty

- Odwzorowanie przekazywania sygnałów ze zdarzenia końcowego jednego procesu do zadania innego procesu:



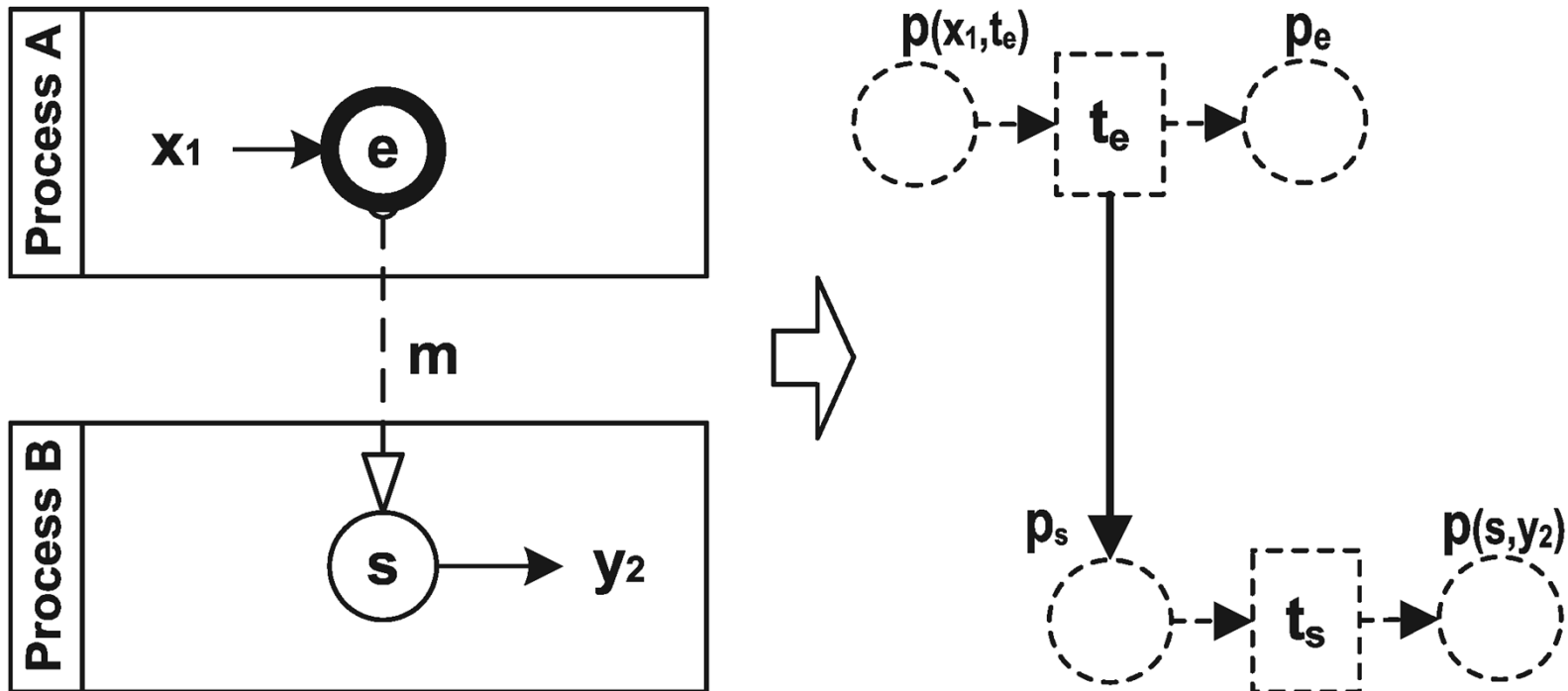
Sygnały

- Odwzorowanie przekazywania sygnałów z zadania jednego procesu do zdarzenia początkowego innego procesu:



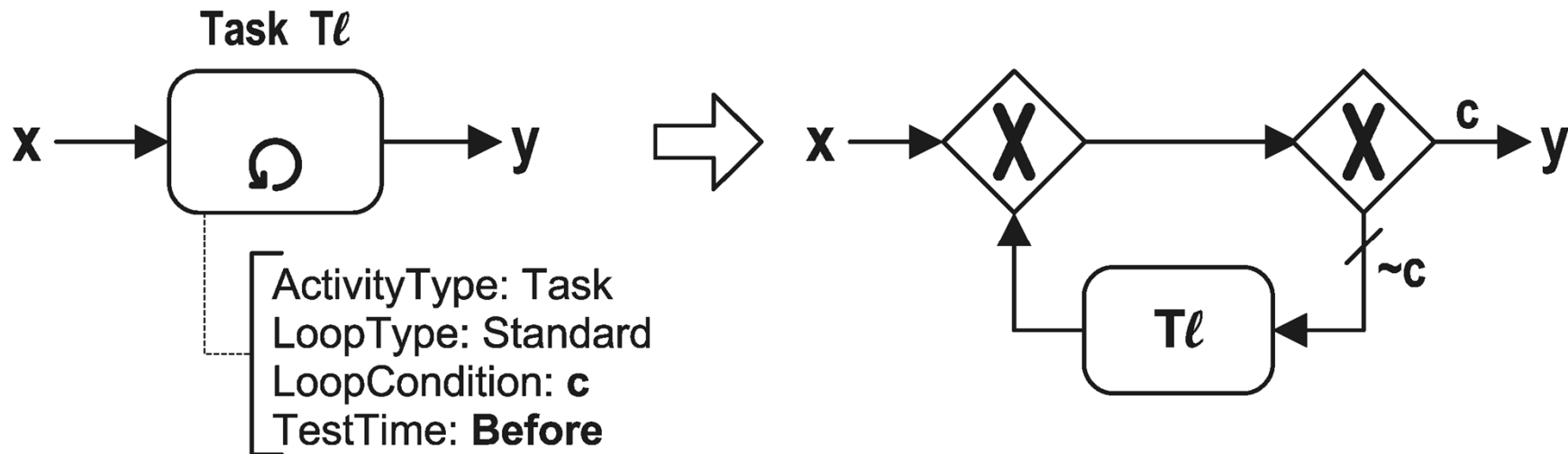
Sygnaly

- Odwzorowanie przekazywania sygnałów ze zdarzenia końcowego jednego procesu do zdarzenia początkowego innego procesu:



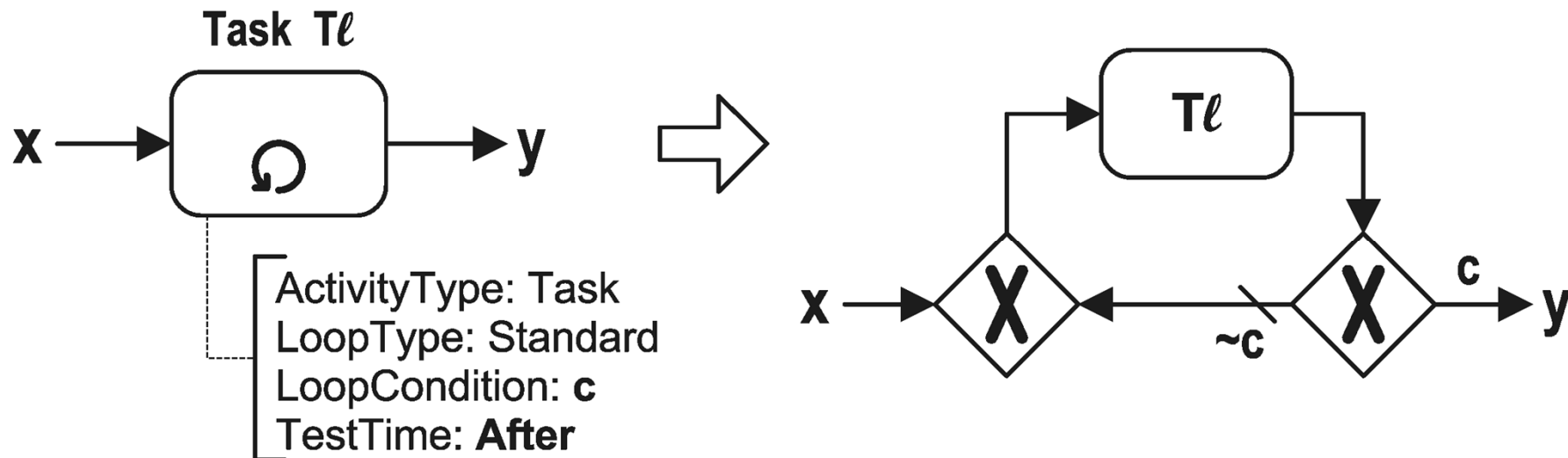
Odwzorowanie złożonych typów zadań

- Odwzorowanie zadań z wewnętrzną pętlą typu *while-do* (TestTime: **Before**) za pomocą wstępnego przetworzenia diagramu BPMN



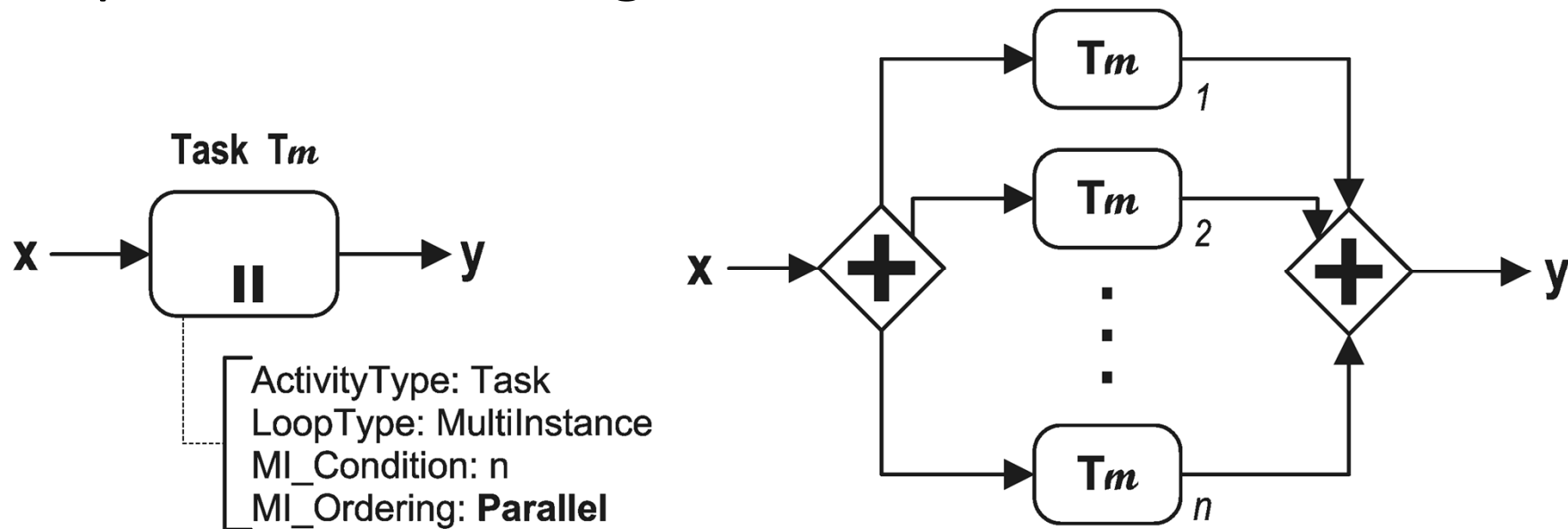
Odwzorowanie złożonych typów zadań

- Odwzorowanie zadań z wewnętrzną pętlą typu *Do-until* (TestTime: **After**) za pomocą wstępnego przetworzenia diagramu BPMN



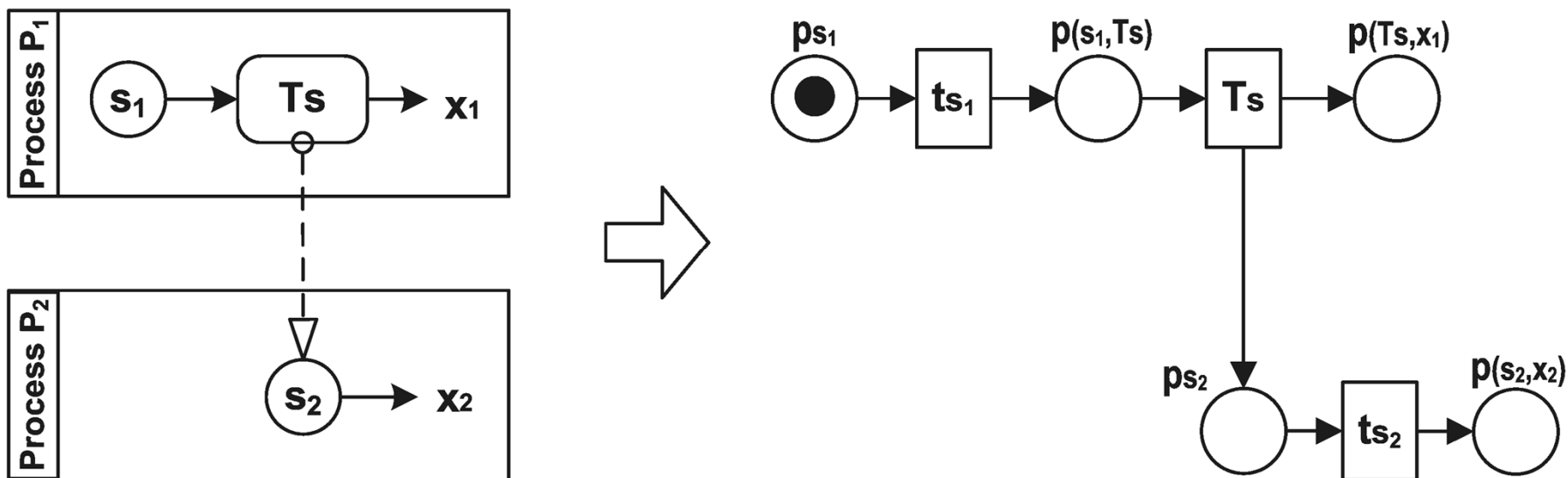
Odwzorowanie złożonych typów zadań

- Odwzorowanie zadań współbieżnych (**Multinstance, Parallel**) za pomocą wstępnego przetworzenia diagramu BPMN



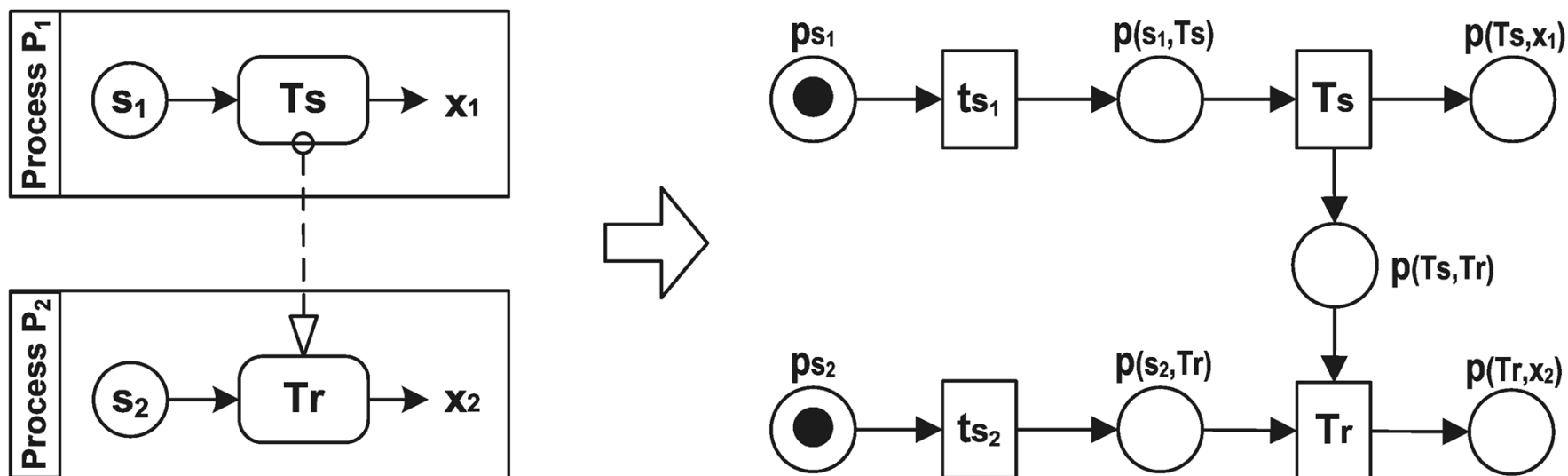
Znakowania początkowe sieci

- Znakowanie wszystkich miejsc reprezentujących zdarzenia początkowe procesów najwyższego poziomu, to znaczy nie wywoływanych przez inne procesy, i które nie są zależne od sygnałów wysyłanych przez inne procesy.



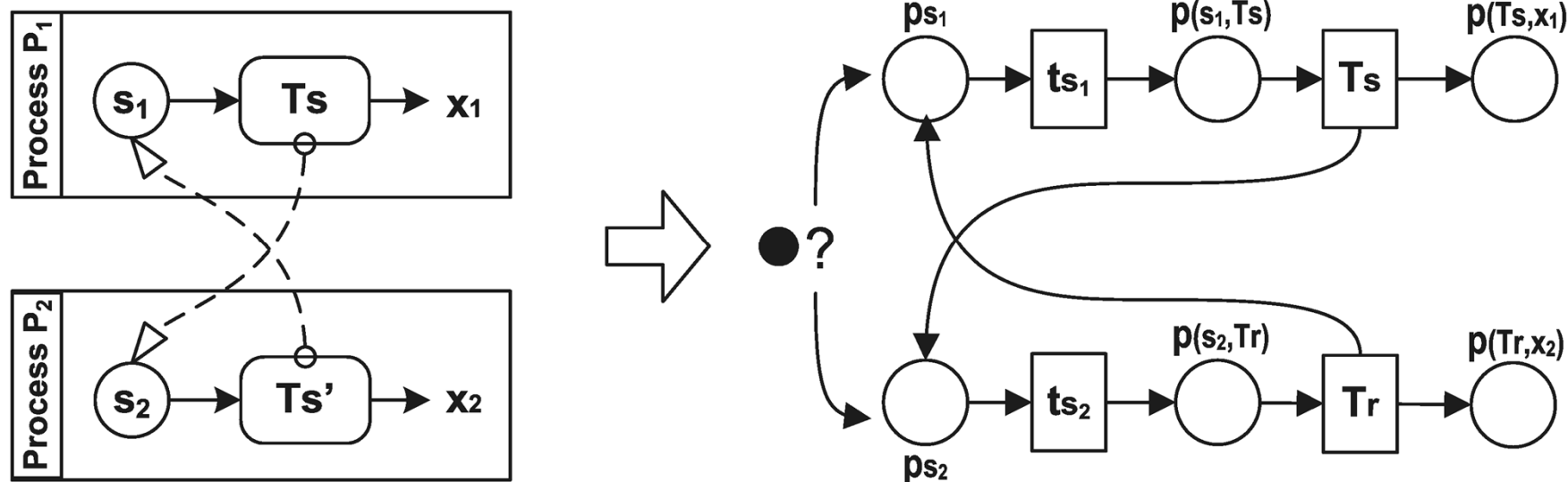
Znakowanie początkowe

- Niezależnie inicjowane procesy



Znakowanie początkowe

- Wzajemna zależność inicjacji procesów, wymaga wyboru jednego z procesów jako początkowego.

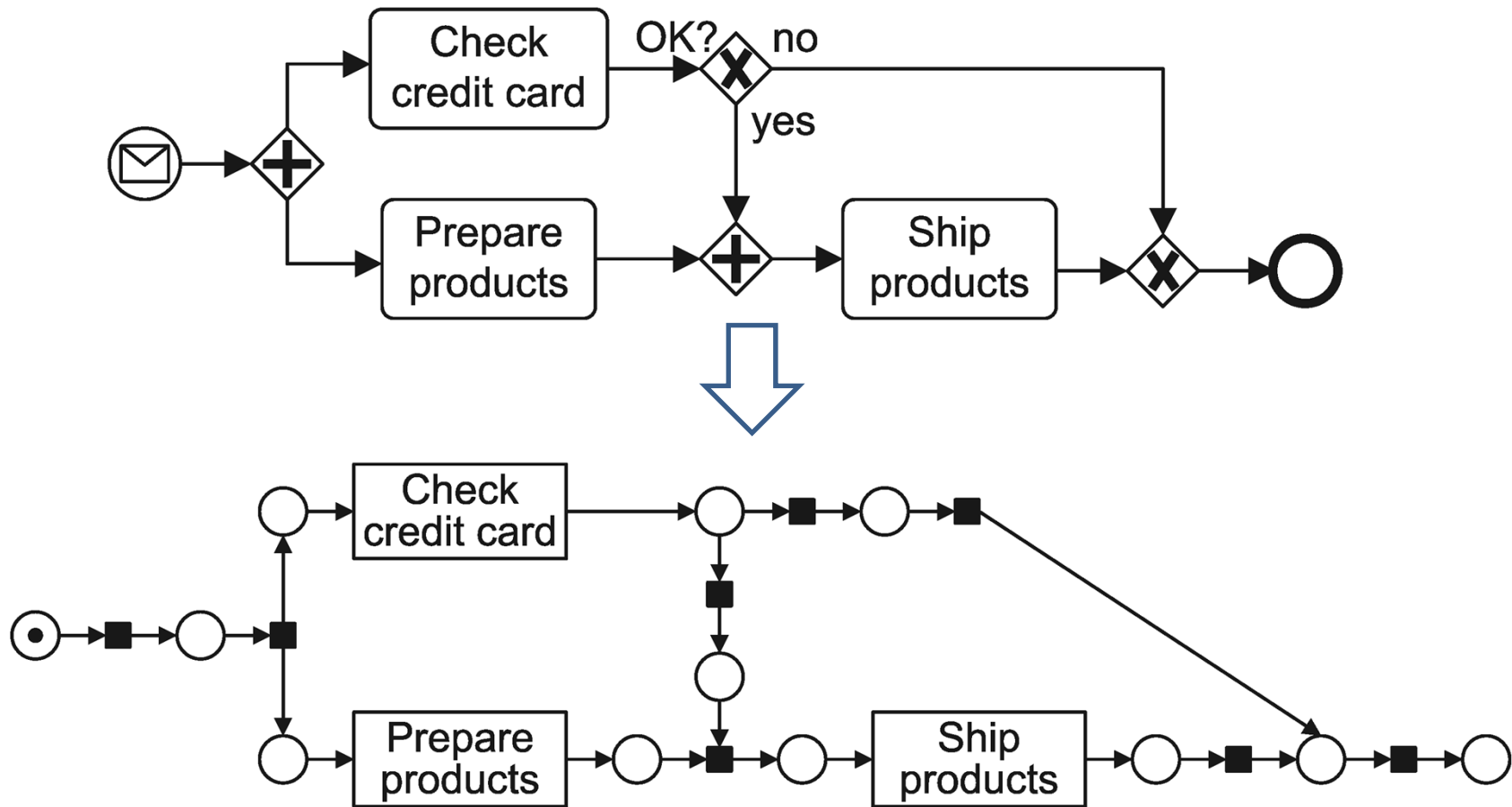


Potencjalne niejednoznaczność specyfikacji diagramów BPMN

- Modele procesów z wieloma zdarzeniami początkowymi
- Zakończenie wystąpienia procesu
- Obsługa błędów współbieżnych wystąpień pod-procesów
- Bramki łączące typu OR

Przykładowe odwzorowania

- Proces realizacji zamówienia



Złożoność modelu PN

Przykłady rozmiarów diagramów BPMN i odpowiadających im modeli sieci Petriego

Model No.	BPMN Model							Petri Net Model	
	tasks	events	XOR	AND	subprocesses	messages	exceptions	places	transitions
1	11	2	9	2				31	34
2	7	4	4	4				23	21
3	9	8	3		2		2	35	39
4	4	2	2					10	10
5	3	2	2	2				12	11
6	4	8	4			4		24	20
7	5	12	4			5		31	25