

Temat przykładu: proces sprzedaży książki on-line przez księgarnię wysyłkową

Zadanie 1 - Diagram kolaboracji.

Przeciągnij element Pool – ustaw go jako Black Box (z menu kontekstowego) i nazwij go Klient.

Przeciągnij drugi element Pool – z menu kontekstowego dla Pool wybierz „Insert Lane After”. Diagram zmodyfikuj w ten sposób, aby mieć w basenie 3 tory. Nazwij je kolejno od góry Dział sprzedaży, Dział finansowy, Dział dostawy. Dla basenu (pool) ustaw nazwę „Księgarnia – proces sprzedaży książki on-line”.

W toolbox’ie na elemencie zdarzenie początkowe rozwiń czarny znaczek trójkąta i ustaw typ zdarzenia początkowego na zdarzenie odbierające komunikat (Message Start Event). Następnie kliknij w torze Dział sprzedaży a zostanie umieszczone w nim odpowiednie zdarzenie początkowe. Wykonaj podwójny klik na wstawionym zdarzeniu i nazwij je „Zamówienie otrzymano”.

Wybierz z toolboxa element przepływu komunikatów (message flow) i połącz pool (basen) klienta ze zdarzeniem początkowym. Nadaj linii nazwę „Zamówienie”.

Wybierz czynność typu user task, umieść ją w tym samym torze za zdarzeniem początkowym i nazwij „Weryfikacja poprawności zamówienia”.

Połącz elementy linią przepływu sterowania (sequence flow) od zdarzenia początkowego do umieszczonego tasku. Kierunek przepływu będzie poziomy od zdarzenia początkowego do zdarzenia końcowego. Całość modelu będzie przysłać od lewej strony okna do prawej strony (przesunięcie elementu rozszerzy obszar toru w poziomie). Staraj się zachować czytelność modelu.

Wybierz z toolboxa artefakt Data Input i umieść go nad czynnością „Weryfikacja ...”, nazwij go „Zamówienie”. Podepnij artefakt do czynności linią Data Association –kierunek od artefaktu do czynności, ponieważ czynność ta potrzebuje te dane na wejściu (konsumuje je w trakcie wykonywania czynności). Wykonaj na artefakcie Data Input klik prawym przyciskiem myszy i wybierz z menu kolejno State->Create, wpisz nazwę stanu „Utworzone”.

Wybierz kolejny artefakt, ale tym razem Data Object, nazwij „Zamówienie”, podepnij jak poprzednio, ale w kierunku od czynności do artefaktu (co w modelu oznacza daną wyjściową). Wykonaj na artefakcie klik prawym przyciskiem myszy i wybierz z menu kolejno State->Configure-> Add; dodaj kolejno stany „Zweryfikowane”, „W realizacji” i „Wycofane”. Ustaw stan dokumentu na „Zweryfikowane”.

Następnie umieść bramkę xor sterowaną danymi za ostatnio wstawioną czynnością w tym samym torze. Bramkę nazwij „Zamówienie poprawne?”.

Uzupełnij linię przepływu sterowania od czynności do bramki.

Wybierz z toolboxa zadanie (task) o typie Service Task i umieść go w torze „Dz. sprzedaży” za bramką xor, task nazwij „Zapisz zamówienie”.

Uzupełnij linię przepływu sterowania od bramki do czynności „Zapisz zamówienie”, nazwij ją „TAK”.

Umieść w tym samym torze zdarzenie końcowe wysyłające komunikat (Message End Event), nazwij go „Zamówienie odrzucono” i połącz linią przepływu komunikatów z basenem (pool) klienta (kierunek do klienta). Linię komunikatu nazwij „Zamówienie niepoprawne”.

Uzupełnij linię przepływu sterowania od bramki do zdarzenia końcowego „Zamówienie odrzucono”, linię nazwij „NIE”.

Umieść czynność User task o nazwie „Sprawdzenie dostępności pozycji”, podepnij przepływ sterowania między czynnościami. Wstaw artefakt danych o nazwie „Zamówienie” i ustaw jego stan na „W realizacji”. Potwierdź pytanie o referencję do już istniejącego artefaktu danych o nazwie „Zamówienie”. Artefakt zamówienie [W realizacji] podepnij linią asocjacji (Association) do przepływu sterowania między tymi czynnościami (artefakt ten jest wynikowy z „Zapisz zamówienie” i wejściowy do „Sprawdzenie dostępności pozycji”).

Wybierz z toolbox’a bramkę xor sterowaną danymi i umieść za czynnością „Sprawdzenie dostępności”, bramkę nazwij „Pozycje dostępne?”. Połącz elementy linią przepływu sterowania.

Wybierz z toolboxa element Sub-process (podproces) i umieść w lane (torze) „Dz. dostawy”. Pozostaw standardową nazwę „Process” (będzie modyfikowane w zad. 3). Dodaj basen wydawnictwo typu BlackBox. Sub-process zepnij w obie strony z basenem wydawnictwa liniami wymiany komunikatów.

Wybierz bramkę xor typu merge sterowaną danymi i umieść w torze „Dz. finansowy”. Usuń nazwę standardową.

Aby uzupełnić przepływ sterowania od bramki „Pozycje dostępne” wykorzystaj ikonę połączenia dostępną obok tej bramki (po kliknięciu bramki – patrz rys.1)



rys.1 – ikona połączeń – górny prawy róg ponad rombem

Otworzy się okno z typami elementów – wybierz typ podprocesu i wpisz „Pro” – wśród elementów umieszczonych w basenie wyszukane zostaną elementy typu podproces o nazwach pasujących do wprowadzonego ciągu. Wybierz element o nazwie „Proces”. Nazwij linię „Jeden lub więcej pozycji niedostępne”.

Powtórz czynność dla typu bramka (gateway) i znajdź elementy o pustej nazwie (będzie to bramka w torze „Dział finansowy”) – linie przepływu sterowania nazwij „Wszystkie elementy dostępne”.

W torze „Dz.finansowy” umieść bramkę zrównoleglenia (Parallel Fork) i uzupełnij przepływ sterowania.

Wstaw w ten sam tor czynność Service task o nazwie „Wyliczenie opłaty” a następnie czynność typu Send task o nazwie „Poinformowanie o kwocie do zapłaty”. Uzupełnij przepływ sterowania.

Za ostatnio umieszczonym zadaniem w torze „Dz. finansowy” wstaw zdarzenie pośrednie o typie Message Intermediate Event – nazwij „Opłatę odebrano”. Uzupełnij przepływ sterowania oraz przepływ komunikatu od klienta (linię komunikatu nazwij Opłata).

Za zdarzeniem pośrednim wstaw zadanie typu User task i nazwij „Wystawienie faktury”. Linią artefaktów dopnij do tej czynności artefakt danych (Data Object) nazwany „Faktura”.

W torze „Dz. dostawy” umieść zadanie typu Manual task i nazwij je „Skompletowanie pozycji zamówienia” – jest ono wywoływane równolegle do zadania „Wyliczenie opłaty”. Uzupełnij przepływ sterowania.

Umieść bramkę spinającą Parallel Join (nie nazywając jej) łączącą przepływ od czynności „Skompletowanie pozycji zamówienia” i od czynności „Wystawienie faktury”.

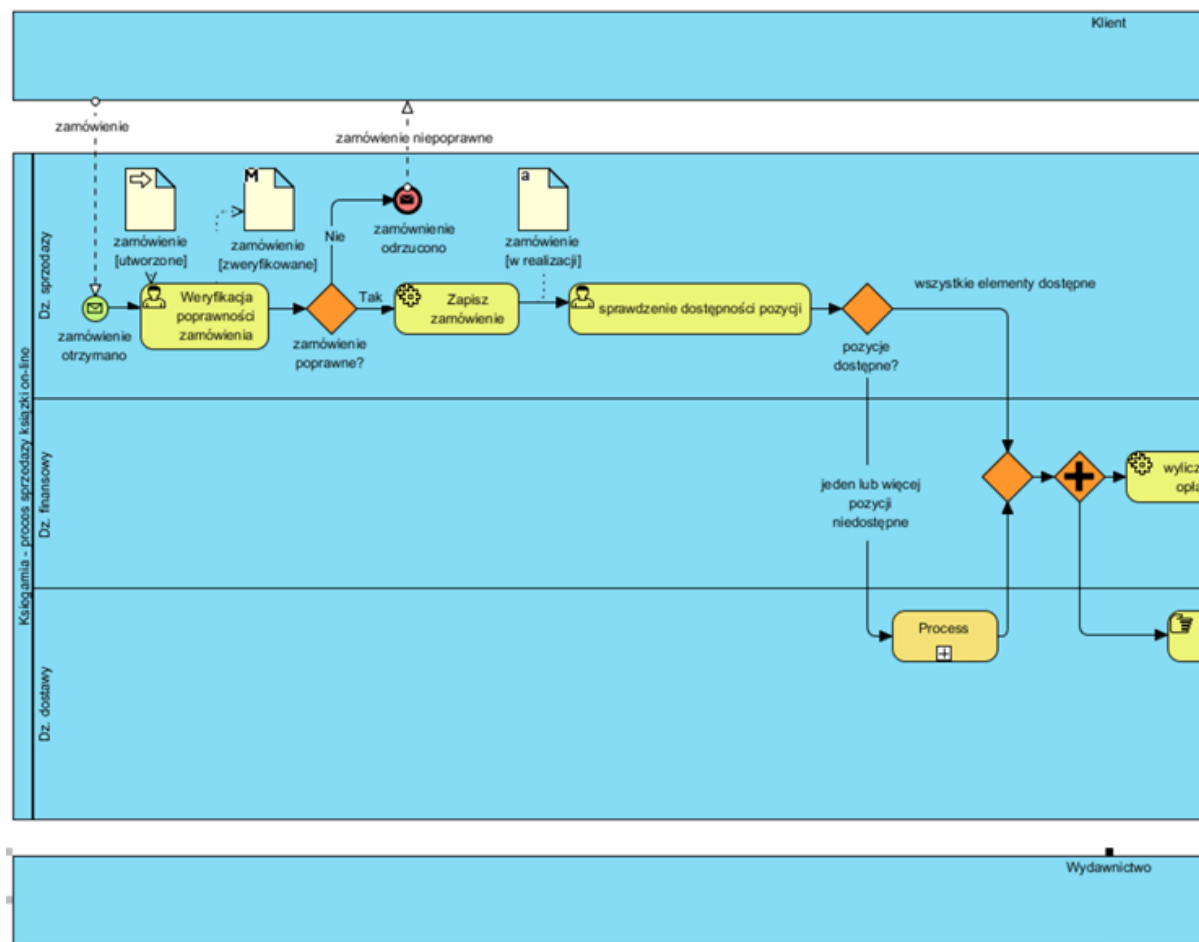
W torze „Dział dostawy” za bramką umieść czynność manualną (manual task) i nazwij ją „Dostarczenie pozycji zamówienia”. Uzupełnij przepływ sterowania.

Kolejnym elementem w tym torze jest zdarzenie pośrednie odbioru komunikatu – nazwij je „Odebranie potwierdzenia dostawy”, połącz przepływ sterowania i umieść linię przesyłu komunikatów (message flow) od klienta do zdarzenia (linię nazwij „Potwierdzenie dostawy”).

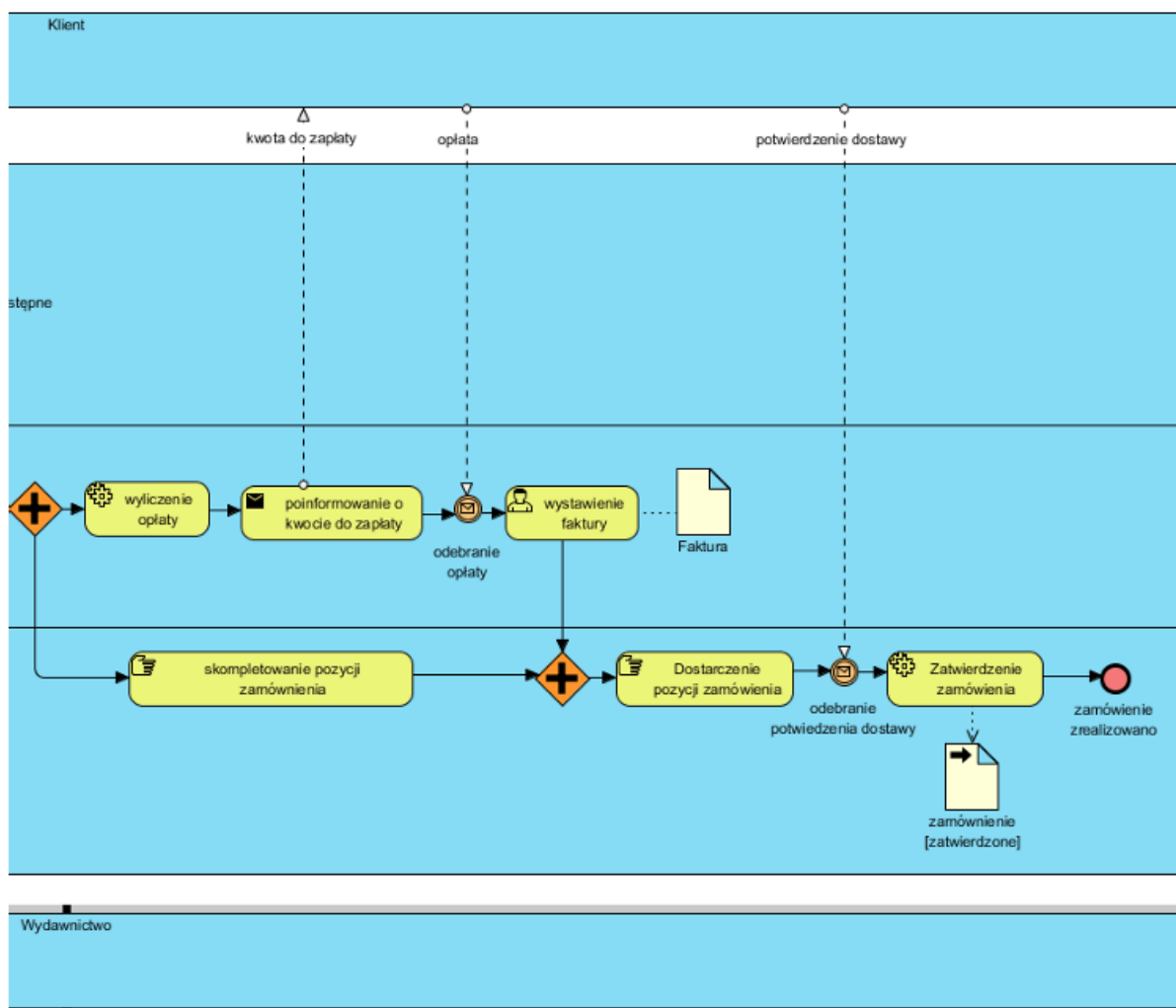
Następnikiem poprzedniego zdarzenia odbioru komunikatu jest czynność typu service task – nazwij ją „Zatwierdzenie zamówienia” i połącz przepływem sterowania. Czynność ta zmienia stan artefaktu (danej o nazwie „Zamówienie”) więc należy umieścić w torze „Dział dostawy” obok czynności „Zatwierdzenie zamówienia” element Data Output Object. Dla Data Output Object ustawić status na

„Zatwierdzone” oraz podpiąć linią połączenia artefaktu (Data Association) od czynności do artefaktu danych.

Ostatnim elementem tego procesu jest zdarzenie końcowe. Umieść je w „Dz. dostawy”, nazwij „Zamówienie zrealizowano” i uzupełnij przepływ sterowania. Postać bieżąca modelowanego procesu powinna wyglądać jak na rys.2a + 2b.



Rys 2a. Początek procesu sprzedaży książki on-line.



Rys 2b. Ciąg dalszy procesu sprzedaży książki on-line.

Zadanie 2 – obsługa sytuacji wyjątkowych (eskalacja + kompensacja)

Wybierz z Toolboxa czynność typu Task (bez określonego typu), umieść w torze Dz. Sprzedaży i nazwij go „Anulowanie zamówienia”. Na tym zadaniu kliknij prawy przycisk myszy i wybierz Open Specification. W zakładce General w dole okna dostępny jest checkbox Compensation – zaznacz go i zamknij okno zatwierdzając przyciskiem OK.

Na czynności typu service task o nazwie „Zapisz zamówienie” umieść na granicy tej czynności zdarzenie pośrednie typu Compensation Intermediate Event, nazwij je „Zamówienie odwołano”. Podepnij linią asocjacji czynność Anulowanie zamówienia do zdarzenia granicznego kompensacji. Kliknij na zdarzeniu granicznym kompensacji prawym przyciskiem myszy i wybierz z menu pozycję Trigger –w rozwiniętym menu zauważ, że zdarzenie to może być jedynie zdarzeniem łapiącym (opcja generujące jest niedostępna) i przerywającym wykonywanie czynności, do której zostało dopięte (nie ma opcji ustawienia na zdarzenie nieprzerywające).

W torze Dział dostawy dla podprocesu, który pierwotnie nazywał się „Process” (zobacz rysunek 2a) ustaw zdarzenie graniczne eskalacji Escalation Intermediate Event.

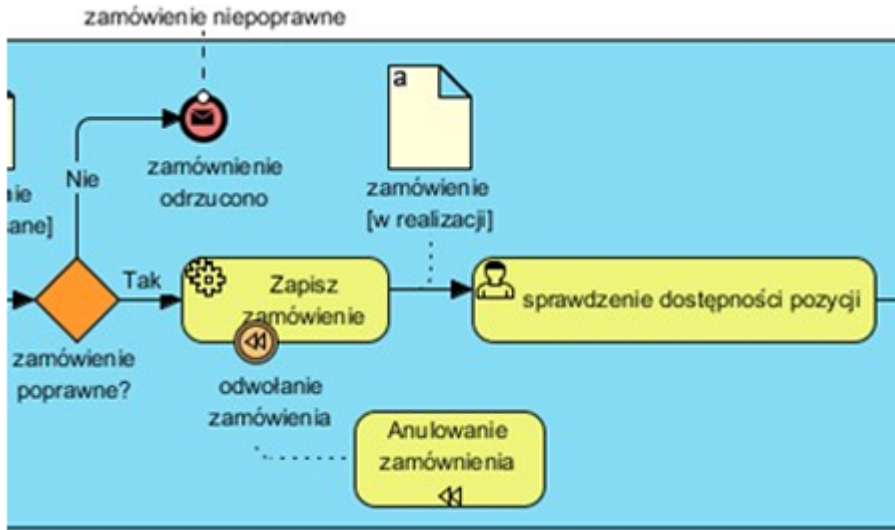
Dodaj czynność wysyłającą komunikat o nazwie „Poinformowanie o braku możliwości realizacji zamówienia”, połącz je do klienta linią komunikatów o nazwie „Zamówienie niemożliwe do realizacji”. Zdarzenie graniczne połącz do czynności wysyłającej komunikat linią przepływu sterowania.

Za czynnością wysyłającą wstaw zdarzenie pośrednie generujące wywołanie kompensacji – nazwij je „odwołanie zamówienia”. Kliknij prawym przyciskiem myszy na zdarzeniu i wybierz Trigger oraz w podmenu ustaw typ zdarzenia na Throwing. Zdarzenie otrzyma wypełnienie wewnątrz znaku kompensacji. Połącz poprzednie zadanie i zdarzenie pośrednie linią przepływu sterowania zgodnie z kierunkiem przepływu sterowania w procesie.

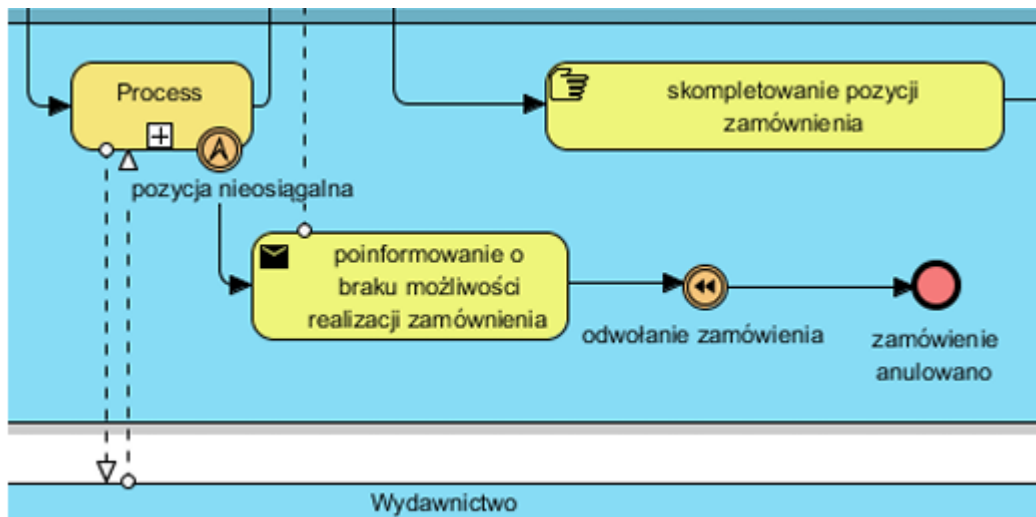
Wstaw zdarzenie końcowe (End Event) o nazwie „Zamówienie anulowano”. Uzupełnij przepływ sterowania.

Zauważ, że można wstawić zamiast zdarzenia pośredniego kompensacji i końcowego jeden element – Compensation End Event – poszukaj go w toolboxie.

Postać końcowa przyjmuje postać odpowiednio z rysunków 3a i 3b.



Rys. 3a Podpięcie obsługi kompensacji.



Rys. 3b Przechwycenie i obsługa eskalacji z wywołaniem kompensacji.

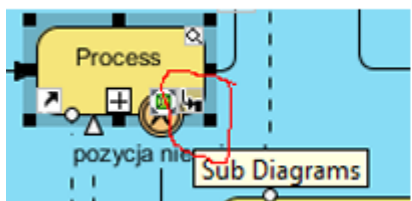
Zadanie 3 – Modelowanie podprocesu pozyskania brakującej /brakujących pozycji

Zadanie przeznaczone jest do samodzielnej realizacji. Zadanie polega na uzupełnieniu podprocesu „Process” z rys. 3b podprocesem „pozyskiwania brakujących pozycji zamówienia” lub „pozyskania brakującej pozycji zamówienia”.

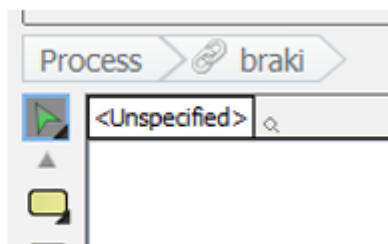
Należy rozważyć czy podproces w kontekście dotychczas wykonanego modelu powinien być zamodelowany jako proces wieloinstancyjny (per pozycja zamówienia), powtarzalny (type: loop) czy jako proces, który w swoim wnętrzu ma pętlę po ilości zamówień. Należy zamodelować go stosownie do przyjętego wyboru.

Podproces otrzymuje na wejściu listę pozycji niedostępnych (dodajmy do bieżącego modelu taki artefakt danych jako element wyjściowy z zadania „Sprawdzanie dostępności pozycji” w torze Dz. Sprzedaży i jednocześnie wejściowy do podprocesu). Jeśli wszystkie brakujące pozycje są uzupełnione to proces główny kontynuuje realizację zamówienia. Jeśli pozycja jest niedostępna należy wyrzucić w podprocesie eskalację, która jest łapana na poziomie procesu głównego jako zdarzenie graniczne „Pozycja niedostępna” (obsłużone w zad.2)

Aby przejść do edycji podprocesu należy kliknąć myszą zadanie typu podproces o nazwie roboczej „Process” oraz kliknąć element Sub Diagrams - wyróżniony na rys. 4. Wybieramy opcję New Diagram i typ procesu jako Business Process Diagram, zatwierdzając okno przyciskiem Next. Nazwijmy roboczo diagram „braki” i zatwierdźmy przyciskiem OK. Zostanie otwarte nowe puste okno przestrzeni do modelowania diagramu podprocesu. Podproces „braki” jest dowiązany do procesu głównego co przedstawiono na wycinku na rys 5. – kliknięcie na „proces” spowoduje powrót na poziom główny proszę wykonać tę operację.



Rys 4 – Ikona linkująca do podprocesu.



Rys 5 – powiązanie procesu i podprocesu

Aby wejść na poziom podrzędy należy kliknąć na ikonie plusa umieszczonej w zadaniu typu podproces i wybrać „Open Business Proces Diagram”. Otwiera się okno, w którym mamy od razu znacznik początkowy spięty z właściwym wejściem z poziomu wyższego oraz znacznik końca procesu podobnie skonfigurowany.

W implementacji podprocesu uwzględnij następujące aspekty:

- Posiadamy listę dostawców, którzy mogą teoretycznie dostarczyć daną pozycję – wybieramy zawsze najpierw najtańszego z nich, jeśli ten nie może dostarczyć to kolejnego z listy;
- Do dostawcy wysyłane jest zamówienie
- Reagujemy odpowiednio w zależności od sytuacji tj.:
 - Jeśli otrzymamy od dostawcy informację o dostarczeniu, to podproces kończy się prawidłowym wyjściem (do procesu nadrzędnego);
 - Jeśli minie czas – 6 godzin od przekazania zamówienia i nie mamy odpowiedzi, to wysyłamy ponaglenie udzielenia odpowiedzi;
 - Jeśli odbierzemy informację o odmowie dostawy przez tego dostawcę, to próbujemy u innego dostawcy;
 - Jeśli wszyscy dostawcy odmówią dostawy, to oznaczamy produkt/produkty w katalogu jako chwilowo niedostępne (zakładamy że inny proces nasłuchuje na sygnał o wznowieniu możliwości dostawy i zmienia w katalogu status) i spinamy z właściwym znacznikiem końca;

Zadanie 4 – symulacja procesu sprzedaży książek on-line

Wybierz z menu głównego aplikacji Visual Paradigm sekcję Modeling. Następnie z górnego paska narzędzi wybierz Simulation. W dodatkowych opcjach, które otworzyły się pod modelem ustaw kolejno:

- Czasy i koszty dla elementów modelu – klikając na poszczególne elementy modelu:

element modelu	średni czas	koszt/usd
----------------	-------------	-----------

zamówienie otrzymano	2 sekundy	0,5
weryfikacja poprawności	4 min	2
zapisz zamówienie	10 s	0,2
anulowanie zamówienia	12 s	0,2
sprawdzenie dostępności pozycji	8 min	4
podproces	5 godzin	15
Poinformowanie o braku możliwości realizacji...	1 min	3
odwołanie zamówienia	1 s	0,1
wyliczenie opłaty	10 s	0,5
poinformowanie o kwocie do zapłaty	1 min	3
odebranie opłaty	2 s	0,1
wystawienie faktury	4 min	2,5
skompletowanie pozycji zamówienia	10 min	3
dostarczenie pozycji	1,5 godz	6
odebranie potwierdzenia dostawy	3 s	0,2
zatwierdzenie zamówienia	4 s	1

- Ilość instancji pracujących w poszczególnych torach- ustaw klikając tor – Najpierw podaj nastawy dla sytuacji 1 (nastawy dla sytuacji 2 wykorzystasz dopiero przy drugim uruchomieniu symulacji):

lane:	instance /sytuacja 1	instance /sytuacja 2
dział sprzedaży	1	2
dział finansowy	1	1
dział dostawy	1	3

- Zasoby i ich ilość-
W oknie Available resources zdefiniuj dostępne zasoby zgodnie z tabelą poniżej (zasób + ilość).

Odszukaj na modelu czynności podane w kolumnie “wykorzystywany przez czynność” i dla każdej z nich (po jej kliknięciu) przypisz w oknie Required resources odpowiedni zasób w ilości zgodnie z tabelą.

zasób	ilość	wykorzystywany przez czynność:
Telefon Dział Finansowy	1	poinformowanie o kwocie do zapłaty
Telefon Dział Dostawy	1	poinformowanie o braku możliwości realizacji zamówienia
Komputer Dział Finansowy	1	wystawienie faktury
Drukarka Dział Finansowy	1	wystawienie faktury
Komputer Dział Sprzedaży	1	Weryfikacja poprawności zamówienia, sprawdzenie dostępności pozycji

- Ścieżki przejść modelu w symulacji

Kliknij przycisk „Show Scenarios” i dodaj kolejno ścieżki z odpowiadającym im założonym procentem występowania. Pojedynczą ścieżkę definiuj w oknie Path. Za całkowitą ilość symulowanych instancji przyjmij wartość 100 – wpisz ją w pole „Total”.

Uwaga: Oznaczenie dla ścieżki checkbox’a `editable` wymusi dostępność tej ścieżki do edycji.

nazwa	pokonywana ścieżka:	%
Ścieżka 1	start->odrzucone zamówienie	1
Ścieżka 2	start->zamówienie poprawne-> pozycje dostępne-> zamówienie zrealizowane	70
Ścieżka 3	start->zamówienie poprawne->brak pozycji->zamówienie zrealizowane	20
Ścieżka 4	start->zamówienie poprawne ->brak pozycji->zamówienie anulowano	9

Symulacja modelu ASIS:

Ustaw ilość instancji (pracowników) pracujących w poszczególnych torach klikając wybrany tor (lane) i podając wartość w polu instance zgodnie z poniższą tabelą:

Lane:	Instance (sytuacja ASIS)
Dział sprzedaży	1
Dział finansowy	1
Dział dostawy	1

Do poniższych symulacji dobierz właściwą skalę obserwacji.

Symulacja 1: Łączny czas i koszt dla wykonania 100 instancji zapisz w tabeli wyników dla symulacji ASIS. Zweryfikuj obciążenie zasobów (ikona wykresów obok kosztu otwiera okna z wykresami).

Symulacja 2: Zmodyfikuj ilość dostępnych zasobów w Available resources ustawiając w dziale sprzedaży 2 dostępne komputery, Zweryfikuj czas, koszt i obciążenie zasobów -zapisz wyniki.

Tabela wyników symulacja ASIS:

	dzień godziny: minuty: sekundy	koszt/USD
Sytuacja 1	12D14:27:29	2595
Sytuacja 2	12D14:27:29	2595

Symulacje modelu TOBE:

Zmień ilość pracowników pracujących w poszczególnych działach zgodnie z poniższą tabelą:

Lane:	Instance (sytuacja TOBE)
Dział sprzedaży	2
Dział finansowy	1
Dział dostawy	3

Symulacja 1: zmodyfikuj ilość dostępnych zasobów w Available resources ustawiając w dziale sprzedaży 1 dostępny komputer. Łączny czas i koszt dla wykonania 100 instancji zapisz w tabeli wyników dla symulacji TOBE. Zweryfikuj obciążenie zasobów (ikona wykresów obok kosztu otwiera okna z wykresami).

Symulacja 2: Zmodyfikuj ilość dostępnych zasobów w Available resources ustawiając w dziale sprzedaży 2 dostępne komputery, Zweryfikuj czas, koszt i obciążenie zasobów -zapisz wyniki.

Tabela wyników symulacja TOBE:

	dzień godziny: minuty: sekundy	koszt/USD
Sytuacja 1	4D10:12:08	2595

Sytuacja 2	4D06:43:21	2595
------------	------------	------

Modyfikacje

Przy obecnych wartościach zaproponowanych w symulacjach ASIS oraz TOBE, ciężko wywnioskować coś ponad to, że więcej=szybciej. Na wykresach obciążenia wszystkie wykresy zasobów utrzymują się na pułapie 100%, zarówno w modelu ASIS oraz TOBE. Dodatkowo obecne symulacje nie pokazują żadnych korzyści z automatyzacji procesów, jako że różnią się tylko ilością pracowników.

1. Propozycja ASIS:

Dla modelu ASIS ustawię wartość każdego z zasobów (pracowników pracujących w działach oraz dostępnych im sprzętów) na 100 - wartość graniczna w tym przypadku, zwiększenie bardziej któregośkolwiek zasobu nie przyspieszy już bardziej symulacji.

	dzień godziny: minuty: sekundy	koszt/USD
ASIS - graniczny	00D06:52:19	2595

Oczywiście to raczej bez sensu żeby zatrudniać dodatkowego pracownika dla sekundy zysku ale teraz na podstawie wykresów zużycia można sprawdzić realne obciążenie zasobów w trakcie wykonywania symulacji i spróbować dostosować wartości w modelu TOBE tak aby odzwierciedlały jakąś optymalizację.

2. Propozycja TOBE:

Dla modelu TOBE zaproponuję dwie sytuacje, najpierw spróbuję samą zmianą zasobów nieco zoptymalizować propozycje z modelu ASIS, a następnie dodam automatyzację niektórych procesów.

Z wykresu zużycia zasobów w modelu ASIS widzimy że:

- Komputery i pracownicy w dziale sprzedaży są wykorzystywani praktycznie w 100% ale tylko przez pierwsze kilka minut z prawie 7h symulacji. Zmniejszę liczbę pracowników i komputerów do 2.
- Pracownicy i sprzęty z działu finansów, wykorzystywane są w najgorszym momencie w 70%, natomiast przez znaczną większość czasu również nie są potrzebni. Zmniejszenie tych zasobów do 70 w ogóle nie wydłużyłoby czasu trwania symulacji ale co ważniejsze zmniejszenie do 2 również, jako że praca

w tym wydziale wykonywana jest równoległe z pracą wydziału dostaw. Gdyby zredukować liczbę zasobów do 1 czas wykonywania symulacji wzrósłby o godzinę (kilka procent), toteż na razie zostawię liczbę pracowników i sprzętów w wydziale finansów na 2.

- Pracownicy w dziale dostaw w szczytowym momencie wykorzystywani są w 50% natomiast telefon tylko w 1%, od razu można więc zredukować liczbę pracowników do 50, a telefonów do 1. Problemem jest to że dalsza redukcja pracowników zaczyna praktycznie liniowo (kolejne 50% mniej pracowników = 50% dłuższa symulacja) wydłużać symulację. Automatyzacja tego działu dałaby bardzo dużo ale dział ten opiera się na manualnych pracach trudnych do zautomatyzowania (zbieranie przedmiotów do zamówienia oraz ich dostawa). Zmniejszenie liczby pracowników do 2 w tym dziale tak aby suma pracowników odpowiadała sumie z oryginalnego modelu TOBE (6) znacząco zwiększyłoby czas symulacji co też pokazuje że pracownicy w tym dziale są najbardziej potrzebni. Na Razie ustawię liczbę pracowników tego działu na 3.

Tak oto prezentuje się pierwsza sytuacja modelu TOBE, względem oryginału, zatrudniłem jednego pracownika więcej w dziale finansów, razem z dodatkowymi sprzętami dla niego, a wyniki prezentują się tak:

	dzień godziny: minuty: sekundy	koszt/USD
TOBE - oryginalny	4D06:43:21	2595
TOBE - nowy 1	4D06:43:21	2595

Jak widać mimo zatrudnienia dodatkowego pracownika w dziale, czas pozostał taki sam - zupełnie nieopłacalne rozwiązanie, toteż teraz w sytuacji drugiej spróbuję zautomatyzować dział finansów oraz sprzedaży

W tym celu:

- Wszystkie taski typu user w działach finansów i sprzedaży zamienię na taski typu service:
 - Order validation, teraz dzieje się automatycznie, zajmuje 5 sekund i kosztuje 0,2 USD
 - Item availability check, również automatyczne przeszukiwanie bazy, zajmuje 20 sekund i kosztuje 0,3 USD
 - Invoice issue, też dzieje się cyfrowo, żadnego drukowania faktury, wszystko załatwione przez serwis, zajmuje 20 sekund i kosztuje 0,3USD

- Task typu message "Inform about the amount to pay" również zmienimy z rozmowy telefonicznej na maila/smsa, wysyłanego automatycznie, zajmuje 5 sekund i kosztuje 0,2 USD
- Do tego sprzęty takie jak drukarka telefon czy komputer przestały być tu potrzebne, a instancje 2 pracowników na dział zamieniliśmy pojedynczą instancją serwisu.

Tym samym po tych zmianach realna liczba pracowników to 3 (w dziale dostaw), a wyniki prezentują się następująco.

	dzień godziny: minuty: sekundy	koszt/USD
Automatyzacja	4D03:31:41	1617

Jak widać czas nie różni się znacząco ale koszty zmalały o około 40%, teraz jeszcze w miejsce 4 zredukowanych stanowisk w działach sprzedaży i finansów (zastąpionych instancją serwisu tak żeby symulacja dalej działała), zatrudnimy 3 nowych pracowników do działu dostaw (aby suma pracowników wynosiła tyle co w oryginalnym TOBE) i porównamy wyniki.

	dzień godziny: minuty: sekundy	koszt/USD
TOBE - oryginalny	4D06:43:21	2595
TOBE - nowy 1	4D06:43:21	2595
TOBE - nowy 2	2D02:21:00	1617

Tym samym w uproszczeniu (mam nadzieję że nie za dużym) widzimy że automatyzacja działu finansów i sprzedaży, oraz zatrudnienie większej ilości pracowników w dziale dostaw kosztem zredukowanych stanowisk w zautomatyzowanych działach, przyspieszyły czas wykonywania symulacji o ponad 50% i zmniejszyły koszty o prawie 40%.