

Marek Piotrowski

one
press



Procesy
biznesowe
w polskich
warunkach

PROCESY BIZNESOWE W PRAKTYCE

Projektowanie, testowanie i optymalizacja

Hellon 

Wszelkie prawa zastrzeżone. Nieautoryzowane rozpowszechnianie całości lub fragmentu niniejszej publikacji w jakiegokolwiek postaci jest zabronione. Wykonywanie kopii metodą kserograficzną, fotograficzną, a także kopiowanie książki na nośniku filmowym, magnetycznym lub innym powoduje naruszenie praw autorskich niniejszej publikacji.

Wszystkie znaki występujące w tekście są zastrzeżonymi znakami firmowymi bądź towarowymi ich właścicieli.

Autor oraz Wydawnictwo HELION dołożyli wszelkich starań, by zawarte w tej książce informacje były kompletne i rzetelne. Nie biorą jednak żadnej odpowiedzialności ani za ich wykorzystanie, ani za związane z tym ewentualne naruszenie praw patentowych lub autorskich. Autor oraz Wydawnictwo HELION nie ponoszą również żadnej odpowiedzialności za ewentualne szkody wynikłe z wykorzystania informacji zawartych w książce.

Redaktor prowadzący: Magdalena Dragon-Philipczyk
Projekt okładki: Jan Paluch

Fotografia na okładce została wykorzystana za zgodą Shutterstock.

Wydawnictwo HELION
ul. Kościuszki 1c, 44-100 GLIWICE
tel. 32 231 22 19, 32 230 98 63
e-mail: onepress@onepress.pl
WWW: <http://onepress.pl> (księgarnia internetowa, katalog książek)

Drogi Czytelniku!

Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres

<http://onepress.pl/user/opinie/probiz>

Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

ISBN: 978-83-246-7120-5

Copyright © Helion 2014

Printed in Poland.

- [Kup książkę](#)
- [Poleć książkę](#)
- [Oceń książkę](#)

- [Księgarnia internetowa](#)
- [Lubię to! » Nasza społeczność](#)

Spis treści

Wstęp	11
Rozdział 1. Proces	13
1.1. Składowe opisu procesu	15
1.2. Elementy określające proces	16
Rozdział 2. Notacja BPMN	19
2.1. Rodzaje zadań	22
2.2. Rozgałęzianie procesu	23
2.2.1. Bramka ALBO (XOR)	23
2.2.2. Bramka LUB (OR)	25
2.2.3. Bramka I (AND)	25
2.2.4. Bramka złożona (COMPLEX)	27
2.3. Symbole zdarzeń	27
2.3.1. Grubość i rodzaj obramowania okręgu, czyli symbole zdarzeń	27
2.3.2. Styl linii obramowania okręgu, czyli zdarzenia przerywające i nieprzerywające	37
2.3.3. Ikona wewnątrz symbolu, czyli oznaczenie czynności	40
2.3.4. Rodzaj ikony (zdarzenie przyjęcia/zdarzenie wysłania)	40
2.4. Łączenie gałęzi procesu	47
2.4.1. Bramka łącząca ALBO (XOR)	47
2.4.2. Bramka łącząca I (AND)	47
2.4.3. Bramka łącząca LUB (OR)	49
2.4.4. Bramka łącząca złożona (COMPLEX)	49
2.4.5. Łączenie bez bramek	53
2.5. Określanie uczestników procesu	54
2.5.1. Przykład opisu	54
2.5.2. Dlaczego warto opisywać proces za pomocą ról, a nie nazwisk?	55

2.6.	Reprezentacja interakcji z podmiotami zewnętrznymi	59
2.6.1.	<i>Baseny</i>	59
2.6.2.	<i>Procesy prywatne i publiczne</i>	59
2.6.3.	<i>Kolaboracja i konwersacja</i>	61
2.6.4.	<i>Podmioty wieloinstancyjne</i>	66
2.7.	Podprocesy	68
2.7.1.	<i>Podprocesy osadzone</i>	68
2.7.2.	<i>Podprocesy zdarzeniowe</i>	73
2.7.3.	<i>Podprocesy niesekwencyjne (dorażne)</i>	75
2.7.4.	<i>Podproces Pętla</i>	77
2.7.5.	<i>Podprocesy wieloinstancyjne</i>	77
2.8.	Pomocnicze elementy notacji	86
2.8.1.	<i>Obiekty obrazujące dane</i>	86
2.8.2.	<i>Grupy</i>	91
2.8.3.	<i>Symbole prywatne</i>	91
2.8.4.	<i>Adnotacje</i>	92
2.9.	Choreografie	94
2.9.1.	<i>Podstawowe pojęcia</i>	94
2.9.2.	<i>Choreografia złożona</i>	96
2.9.3.	<i>Użycie symboli zwielokrotniających</i>	97
2.9.4.	<i>Przykład zastosowania</i>	97
2.9.5.	<i>Sekwencje poprawne i niepoprawne</i>	98
2.9.6.	<i>Użycie bramek w diagramach choreografii</i>	101
2.9.7.	<i>Zdarzenia w choreografiach</i>	109
2.9.8.	<i>Wykorzystanie choreografii w diagramie kolaboracji (współpracy)</i>	110
2.9.9.	<i>Posługiwanie się globalną choreografią lub globalnym zadaniem choreografii</i>	110
2.10.	Dla porządku — podsumowanie	110
Rozdział 3. Pozostałe notacje		115
3.1.	UML	116
3.1.1.	<i>Diagram czynności</i>	116
3.1.2.	<i>Diagram stanów</i>	122
3.1.3.	<i>Interakcje po raz pierwszy, czyli diagram sekwencji (przebiegu)</i>	124
3.1.4.	<i>Interakcje po raz drugi, czyli diagram komunikacji</i>	129
3.1.5.	<i>Interakcje po raz trzeci, czyli diagram czasowy</i>	130
3.1.6.	<i>Interakcje po raz czwarty, czyli diagram przeglądu interakcji</i>	131
3.1.7.	<i>Diagram przypadków użycia</i>	132
3.2.	RAD	138
3.3.	Przypadki użycia	139
3.4.	Flowchart	144

3.5. Diagram przepływu danych (DFD)	147
3.6. Diagram stanów	149
3.7. Diagramy księgi jakości	151
3.8. Service blueprint	152
3.9. Nierysunkowe metody zapisu	153
3.9.1. <i>Macierz RACI</i>	155
3.9.2. <i>Macierz SIPOC</i>	155
Rozdział 4. Tworzenie opisu procesu	157
4.1. Identyfikacja procesów	157
4.1.1. <i>Analiza dokumentacji biznesowej</i>	157
4.1.2. <i>Kwestionariusze</i>	157
4.1.3. <i>Wywiad</i>	159
4.1.4. <i>User stories</i>	159
4.1.5. <i>Warsztaty</i>	160
4.1.6. <i>Obserwacja (autorejestracja bądź śledzenie)</i>	161
4.1.7. <i>Praktyka (terminowanie)</i>	162
4.1.8. <i>Metoda trawnika — dekretacja</i>	162
4.1.9. <i>Przedstawiciel klienta</i>	162
4.1.10. <i>Prototypowanie</i>	163
4.1.11. <i>Podsumowanie</i>	164
4.2. Sporządzenie opisu procesu	166
4.2.1. <i>Metryka dokumentu</i>	166
4.2.2. <i>Wstęp</i>	166
4.2.3. <i>Opis procesu „as is”</i>	167
4.2.4. <i>Opis procesu „to be”</i>	177
4.2.5. <i>Zdefiniowanie procesu to dopiero początek</i>	181
4.3. Sporządzanie schematu procesu na podstawie diagramu struktury produktów	182
Rozdział 5. Rodzaje obiegów	191
5.1. Podział ze względu na sposób definiowania ścieżki obiegu	191
5.1.1. <i>Dekretacja</i>	191
5.1.2. <i>Sekwencje zdarzeń — już nie dekretacja, jeszcze nie workflow</i>	193
5.1.3. <i>Obieg typu workflow</i>	199
5.1.4. <i>Obieg stanowy</i>	202
5.1.5. <i>Obieg definiowany za pomocą silnika reguł</i>	204
5.2. Podział procesów ze względu na medium obiegu	207
Rozdział 6. Najczęstsze błędy	209
6.1. Błąd typu „Przetwarzanie gniazdowe”	209
6.2. Błąd typu „Szybka pętla”	210

6.3.	Błąd typu „Bezkresna pętla”	213
6.3.1.	Schemat z wykorzystaniem zdarzenia pośredniego	214
6.3.2.	Schemat z rozszerzoną pętlą	215
6.4.	Gdy „tak” spotyka się z „nie”, czyli błąd zbędnej decyzji	216
6.5.	Drobne błędy notacyjne	219
6.5.1.	Cancel zamiast znaku uniwersalnego	219
6.5.2.	Oczekiwanie na zdarzenie Cancel	219
6.5.3.	Nieprawidłowe warunki bramki ALBO (XOR)	220
6.5.4.	Nieszkodliwe błędy	222
6.5.5.	Błąd czy nie błąd?	222
Rozdział 7. Typowe zagadnienia		225
7.1.	Kanały komunikacji systemu kancelaryjnego z otoczeniem	225
7.2.	Wpływ dokumentu papierowego	225
7.2.1.	Przypadek trywialny	225
7.2.2.	Przypadek trywialny z połączeniem kancelarii z archiwum	229
7.2.3.	Rozwiązanie ze skanerem masowym i rozproszonym opisem dokumentu	230
7.2.4.	Rozwiązanie z kodem kreskowym	234
7.2.5.	Rozwiązanie z kodem kreskowym i archiwizacją w kartonach	235
7.2.6.	Rozwiązanie z rejestracją przesyłek	236
7.2.7.	Przesyłki rejestrowane	239
7.3.	Wpływ dokumentu uzupełniającego sprawę	240
7.3.1.	Kojarzenie na podstawie numeru sprawy	242
7.3.2.	Kojarzenie ręczne	242
7.4.	Wpływ dokumentu papierowego uzupełniającego przepływ elektroniczny	243
7.4.1.	Wersja procesu bez automatyzacji	244
7.4.2.	Wersja z wykorzystaniem kodu kreskowego	245
7.4.3.	Wersja z kodem dwuwymiarowym	248
7.5.	Wysyłka dokumentu papierowego	252
7.5.1.	Wysyłka jednego dokumentu w jednej kopercie	252
7.5.2.	Kilka dokumentów w jednej przesyłce	254
7.6.	Jeszcze o przyjmowaniu dokumentów	261
Rozdział 8. Niektóre zagadnienia optymalizacyjne		263
8.1.	Kształtowanie przebiegu procesu w zależności od priorytetów	263
8.1.1.	Topologia minimalizująca czas trwania procesu	263
8.1.2.	Topologia minimalizująca korespondencję	265
8.1.3.	Topologia minimalizująca nakład pracy	265

8.2. Metoda ścieżki krytycznej w wydaniu BPM	266
8.2.1. Czym jest ścieżka krytyczna i dlaczego jej wyznaczenie jest tak ważne?	267
8.2.2. Wyznaczanie ścieżki krytycznej	267
8.3. Pobieranie zadań przez wykonawców ze wspólnej puli	272
8.4. Przydział wykonawców do zadań przy przetwarzaniu masowym	274
8.5. Priorytetyzacja	279
8.5.1. [Ważność sprawy]	281
8.5.2. [Liczba dni zapasu]	281
8.6. Zjawisko wąskiego gardła	282
8.7. Doskonalenie procesów	284
Rozdział 9. Testowanie procesów	287
9.1. Najprościej	287
9.1.1. Opis systemu przy użyciu diagramu stanów	287
9.1.2. Co to wszystko ma wspólnego z testami?	289
9.1.3. Dla ciekawskich, czyli dlaczego zamieszczone diagramy nazwałem stanami uproszczonymi?	290
9.2. Nieco trudniejszy przykład	292
9.3. Definiowanie kompletnego testu za pomocą schematu BPMN	296
9.3.1. Opis rozważanego procesu	296
9.3.2. Etap 1. Porządkowanie schematu	297
9.3.3. Etap 2. Jawne wprowadzenie działań wyzwalanych zdarzeniami ...	300
9.3.4. Etap 3. Wprowadzenie na schemat wyjść niejawnych	303
9.3.5. Etap 4. Zaznaczenie literami kroków, w których następuje rozgałęzienie	306
9.3.6. Etap 5. Zaznaczenie działań wewnętrznych w krokach	308
9.3.7. Etap 6. Opisanie kryteriów powodzenia dla zdarzeń wewnętrznych ...	310
9.3.8. Etap 7. Specyfikacja rozgałęzień	311
9.3.9. Etap 8. Sporządzenie szkieletu tablicy kontrolnej	311
9.3.10. Etap 9. Definiowanie przebiegów testowych	314
9.3.11. Etap 10. Sporządzenie formularza testów	328
9.4. Przeprowadzanie testów	333
Rozdział 10. Pomiary	335
10.1. Po co definiować wskaźniki?	335
10.2. Cechy dobrego wskaźnika	336
10.2.1. Cechy techniczne	336
10.2.2. Merytoryczne cechy wskaźników	337
10.3. Wpływ pomiarów na proces	340

10.4. Proces definiowania wskaźników	340
10.5. Rodzaje mierników	340
10.6. Mierniki niezmiennie (z założenia)	343
10.7. Aspekt ludzki	343
Rozdział 11. Na koniec: trochę ideologii, czyli o procesach w ogóle	345
11.1. Procesowa struktura przedsiębiorstwa i jej skutki, czyli po co w ogóle wdrażać procesy biznesowe	346
11.1.1. <i>Podejście departamentowe a podejście procesowe</i>	347
11.1.2. <i>Firma zorientowana procesowo</i>	352
11.1.3. <i>IT — integracja istniejących w firmie systemów informatycznych</i> ..	356
11.1.4. <i>Wymuszanie stosowania procedur jakości</i>	358
11.1.5. <i>Korzyści na poziomie stanowiska pracy</i>	359
11.2. Model dojrzałości procesowej organizacji CMMI	359
11.3. Zmiany, zmiany, zmiany...	360
Dodatek A	363
A.1. Rozdzielczości skanerów potrzebne do skanowania lub skutecznego fotografowania dokumentów	363
A.2. Zestaw symboli BPMN	364
A.3. Wybrane symbole UML	365
A.4. Wybrane symbole UML	366
A.5. Wybrane symbole RAD	366
A.6. Wybrane symbole flowchart	367
A.7. Wybrane symbole stosowane w księgach jakości	368
A.8. Inne symbole spotykane na diagramach procesów	369
Dodatek B Zalecana lektura	371

Rozdział 3.

Pozostałe notacje

Rozdział ma na celu przegląd najpopularniejszych (poza BPMN) notacji, które są stosowane do opisu procesów biznesowych. Ich znajomość, choćby pobieżna, jest dla analityka niezbędna — nie tylko dlatego, że daje szerszą perspektywę, ale przede wszystkim dlatego, że nasi klienci korzystają z wielu różnych notacji (a nie byłoby dobrze, gdyby analityk procesów nie potrafił zrozumieć dostarczonych materiałów...).

Tak robić



TYLKO DLA PRYMUSÓW, CZYLI TEORETYCZNIE O NOTACJACH

Aby notacja była kompletna, powinna posiadać swoją semantykę, syntaktykę i pragmatykę (razem nazywane semiotyką).

Pojęcia te powstały w odniesieniu do języka. Semantyka służy do określenia znaków (wyrażeń) występujących w języku, syntaktyka pokazuje, jak je łączyć. A czym zajmuje się pragmatyka? Określa, „jak stosować” język w kontekście pojęć, przyzwyczajeni i sposobu myślenia jego użytkowników.

Profesor Jacek Malinowski podaje przykład prostego pytania, które na gruncie semantyki i syntaktyki (bez pragmatyki) byłoby niezrozumiałe:

Czy byłby pan tak miły i powiedział mi, gdzie jest najbliższy postój taksówek?

„Biegła w problematyce logicznej osoba rozpozna to pytanie, jako pytanie rozstrzygnięcia. Ma ona pod ręką odpowiednią teorię pytań tego rodzaju, i wie, że pytający spodziewa się od niej odpowiedzi: »tak« bądź »nie«. Rozpozna również od razu fakt, że pytanie to ma presupozycję, iż pytany zna odpowiedź — wie, gdzie jest najbliższy postój taksówek. Jednak logiczna analiza tego pytania, nawet jeśli byłaby poprawna z punktu widzenia logiki, nie daje nam nawet cienia szansy na poprawną odpowiedź na to pytanie. Wszyscy wiemy bowiem, że pytający nie spodziewa się odpowiedzi o formie »tak« lub »nie«. Powyższy problem, pojawiający się w opisanej sytuacji, zawiera różnicę pomiędzy dosłownym znaczeniem oraz znaczeniem sugerowanym przez mówiącego. Zamierzone znaczenie jest znane wszystkim kompetentnym użytkownikom języka polskiego: »Gdzie jest najbliższy postój taksówek«¹.

Jak rozumieć określenia „semantyka”, „syntaktyka” i „pragmatyka” w odniesieniu do notacji?

Semantyka notacji (w uproszczeniu) mówi o symbolach, jakie są stosowane w notacji, i ich znaczeniu. **Syntaktyka** opisuje sposób, w jaki można tych symboli używać, jak je łączyć i budować z nich konstrukcje. Stosunkowo najmniej rozumiane jest pojęcie „pragmatyka notacji”. **Pragmatyka** opisuje, w jaki sposób stosować symbole i konstrukcje do opisu rzeczywistości.

¹ J. Malinowski, *Pragmatyczne interpretacje wypowiedzi*, Instytut Filozofii i Socjologii PAN, cytata za: <http://www.home.umk.pl/~jacekm/pragpol.pdf> [dostęp: 19 maja 2013].

Oczywiście, semantyka czy nawet reguły syntaktyczne są znacznie prostsze do przekazania. Można się ich po prostu nauczyć i — po nabyciu pewnej wprawy — poprawnie z nich korzystać. Pragmatyka jest trudniejsza do przekazania (a tym samym do przyswojenia). Umiejętność jej stosowania nabywa się w trakcie długoletniej praktyki. Można zaryzykować twierdzenie, że o ile semantyka i syntaktyka tworzą rzemiosło, to stosowanie pragmatyki jest sztuką.

3.1. UML

Notacja UML (ang. *Unified Modeling Language* — zunifikowany język modelowania), choć stworzona do modelowania systemów informatycznych, zawiera kilka diagramów przydatnych przy modelowaniu procesów². Ponieważ nie zamierzam przeprowadzać tutaj kursu UML, ograniczę się jedynie do omówienia tych „procesowych” diagramów (w terminologii UML nazywanych „diagramami zachowań”) oraz diagramu przypadków użycia.



Diagramy „procesowe” UML przedstawiam, ponieważ opis procesów za pomocą UML jest wymagany przez niektórych klientów. Dodam, że robię to nieco wbrew sobie — UML nie nadaje się do tego celu, gdyż został stworzony do opisu systemów w świecie obiektów, a nie do opisywania procesów biznesowych.

3.1.1. Diagram czynności

Diagram czynności służy do przedstawienia dynamiki systemu, a dokładniej do zaprezentowania następstwa czynności wykonywanych w celu realizacji zadań zleconych przez aktorów. Jest on podobny do definicji procesu w BPMN³ — opisuje odpowiedzialność za poszczególne fazy procesu oraz ukazuje następstwo czynności.

Podstawowe symbole pokazano w tabeli 3.1.

Tych kilka znaków pozwala już na narysowanie prostego, nierozgałęzionego procesu (rysunek 3.1).

Następnym etapem jest zastosowanie bramek rozdzielających gałęzie procesu. Symbol (romb) i znaczenie symbolu bramki odpowiadają symbolowi i znaczeniu bramki ALBO (XOR) w BPMN (rysunek 3.2A).






Łączenie procesu przed *Czynnością 4* można także poprawnie narysować z zastosowaniem bramki łączącej (patrz rysunek 3.2B).

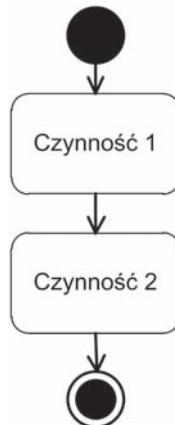
Bramce równoległej I (AND) odpowiada w UML *pasek synchronizujący*. Jego podstawowymi zadaniami są rozdzielanie procesu na gałęzie wykonywane równoległe i synchronizacja ich ponownego połączenia (rysunek 3.3).

² Raport *Dojrzałość procesowa polskich organizacji* przygotowany przez portal procesowcy.pl ujawnił, że UML jest wykorzystywany do modelowania procesów ponad trzy razy rzadziej niż BPMN.

³ Właściwie należałoby napisać, że to BPMN wykazuje podobieństwo do diagramu czynności — notacja UML jest znacznie starsza i niewątpliwie to z niej pochodzą pewne formalne aspekty obecne podczas projektowania notacji BPMN.

TABELA 3.1. Podstawowe symbole diagramu czynności

	Początek procesu
	Czynność — reprezentuje funkcjonalność realizowaną w kolejnym kroku przez system
	Akcja — „atomowa” (niepodzielna) czynność. Tego symbolu używa się raczej rzadko, zwykle wówczas, gdy zachodzi potrzeba dokładnej dekompozycji skomplikowanej czynności
	Przejście — przekazanie sterowania pomiędzy czynnościami
	Koniec procesu



RYSUNEK 3.1. Przykład prostego procesu

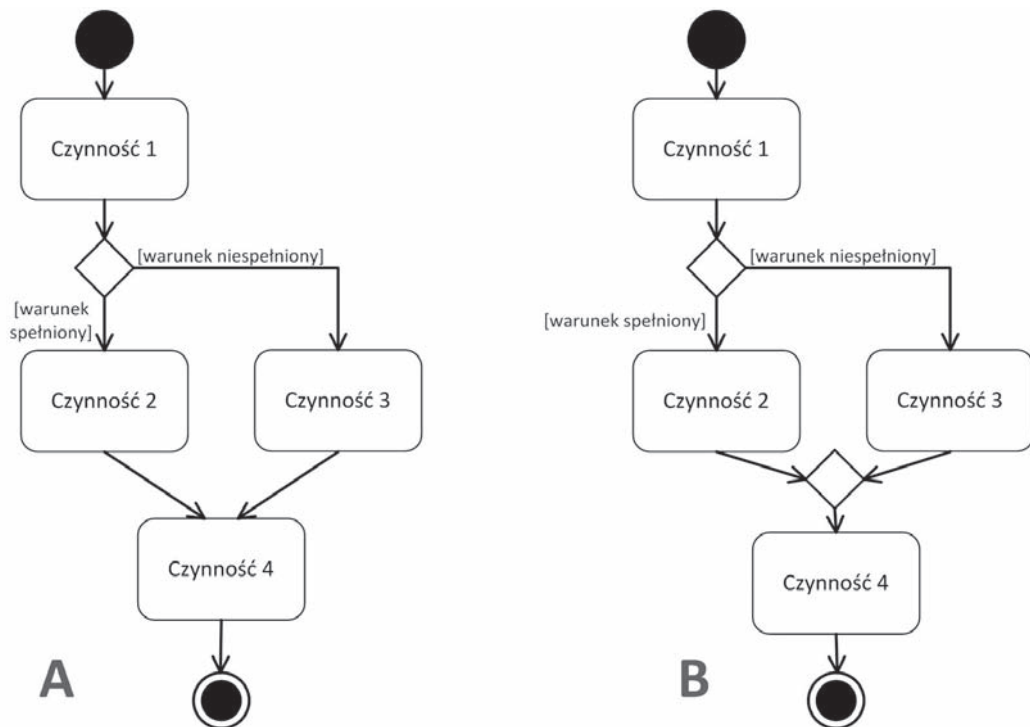
Pasek synchronizujący, choć funkcjonalnie odpowiada BPMN-owskiej bramce równoległej, w UML bywa stosowany w nieco szerszym zakresie. Na przykład rysunek 3.4 przedstawia konstrukcję oczekiwania na zdarzenie (tu zdarzenie zegarowe, symbolizowane przez stylizowaną klepsydrę). To oczekiwanie jest realizowane jako synchronizacja procesu ze zdarzeniem.

W UML dostępnych jest kilka rodzajów zdarzeń (tabela 3.2).

Na rysunku 3.5 zaprezentowano kilka przykładów użycia zdarzeń.

Gałęzie równoległe, utworzone w wyniku użycia *paska synchronizującego*, nie muszą się schodzić ponownie — mogą być kończone niezależnie. Służy do tego symbol zakończenia gałęzi — okrąg z wpisanym krzyżykiem (rysunek 3.6).

Formalnie schemat **A** z rysunku 3.6 jest równoważny schematowi **B** — zakończenie procesu następuje po zakończeniu wszystkich gałęzi. Nie zawsze jednak równoważność jest tak oczywista, co pokazuje rysunek 3.7.



RYСУNEK 3.2. Diagram czynności — zastosowanie bramki rozdzielającej

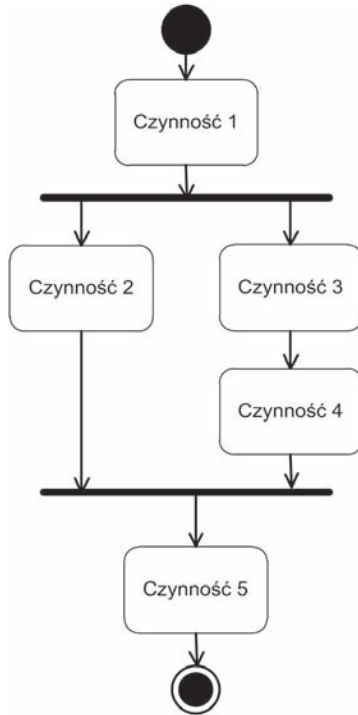
Na schemacie występuje zarówno symbol zakończenia procesu, jak i symbol zakończenia gałęzi. Jeżeli gałąź **c** zostanie zakończona przed zakończeniem gałęzi **a** i **b**, system zachowa się w sposób identyczny jak konstrukcje **a** i **b** pokazane na rysunku 3.7. Jeśli jednak czynności w gałęziach **a** i **b** zostaną zakończone przed zakończeniem gałęzi **c**, proces zostanie zakończony (bez dalszego kompletowania czynności gałęzi **c**).

Niekiedy symbol zakończenia gałęzi może być stosowany w sposób faktycznie zastępujący zakończenie procesu. Przykładem takiego zastosowania jest schemat pokazany na rysunku 3.8. System sprawdza uprawnienia osoby próbującej otworzyć sejf elektroniczny — w przypadku braku uprawnień czynność *Otwórz sejf* nigdy nie zostanie wykonana (a więc praca⁴ „nie dotrze” do symbolu zakończenia procesu). Proces zostanie zakończony bezpośrednio po stwierdzeniu braku uprawnień osoby próbującej otworzyć sejf.

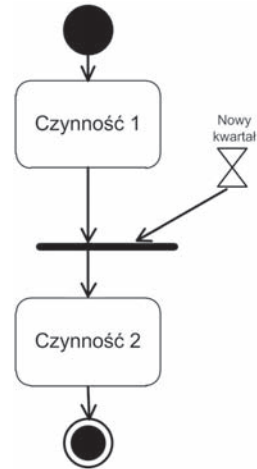
Następnym symbolem ciekawym dla nas z punktu widzenia procesowego jest *punkt synchronizacji*. Wskazuje on miejsce, w którym dwa równoległe strumienie czynności zostają zsynchronizowane w czasie (rysunek 3.9).

Jeżeli w którejkolwiek gałęzi proces dotrze do miejsca związanego z punktem synchronizacji, poczeka na drugą gałąź.

⁴ W tym miejscu użyteczne byłoby pojęcie „token” omówione w podrozdziale „Obieg typu workflow”.



RYSUNEK 3.3. Rozdzielenie i późniejsza synchronizacja gałęzi procesu przy użyciu paska synchronizującego



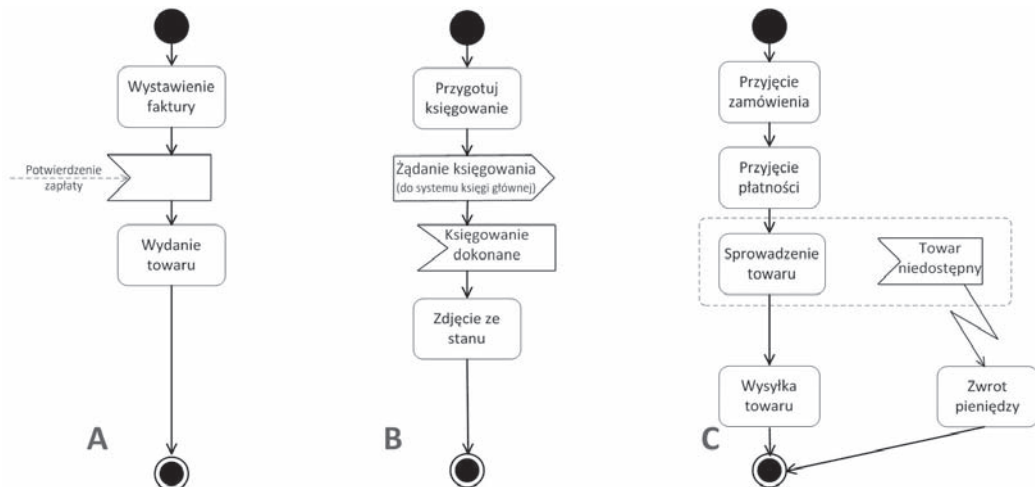
RYSUNEK 3.4. Proces oczekujący na zdarzenie (czasowe)

TABELA 3.2. Symbole zdarzeń w diagramie czynności

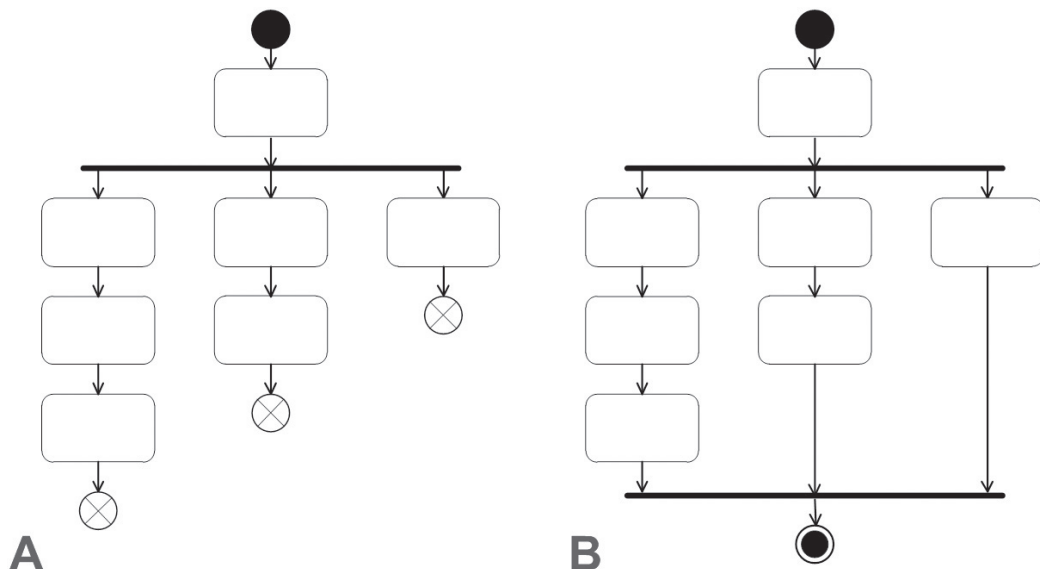
	Zdarzenie czasowe (termin, zdarzenie cykliczne itd.)
	Zdarzenie mające charakter błędu (wyjątku)
	Dowolne zdarzenie (może oznaczać zarówno przyjęcie komunikatu, jak i wiadomość o zdarzeniu)
	Wysyłka komunikatu lub żądania

W przytoczonych wyżej diagramach UML nie definiowano wykonawców poszczególnych czynności, jednak UML (podobnie jak BPMN) pozwala na zastosowanie *pasów pływających* dla poszczególnych aktorów procesu (rysunek 3.10).

Niekiedy schemat procesu staje się zbyt skomplikowany; na jego czytelność najgorzej wpływają przecięcia linii. Aby temu zaradzić, można skorzystać z symbolu łączenia — jest to okrąg z wpisaną literą. Miejsca z takim samym okręgiem można traktować tak, jakby zostały połączone linią (rysunek 3.11).



RYСУNEK 3.5. Przykłady zdarzeń: A — odbiór komunikatu, B — wysyłka komunikatu i odebranie odpowiedzi, C — zdarzenie zewnętrzne awaryjnie przerywające proces

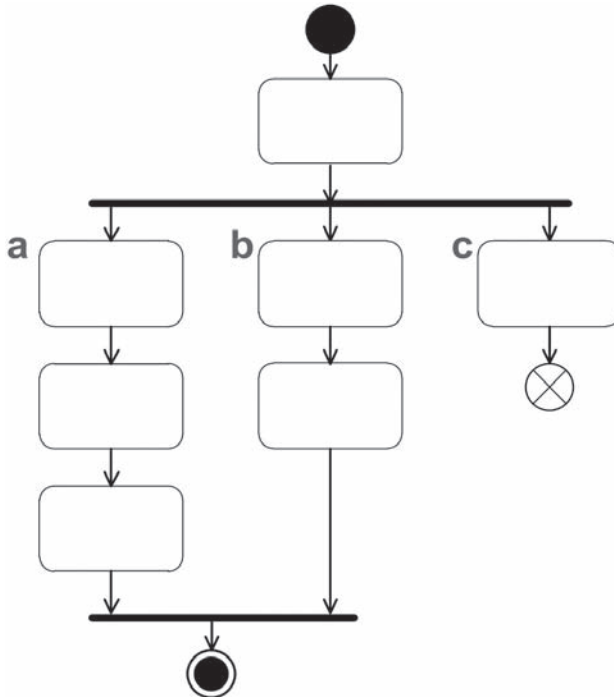


RYСУNEK 3.6. Zastosowanie znaku zakończenia gałęzi i schemat równoważny

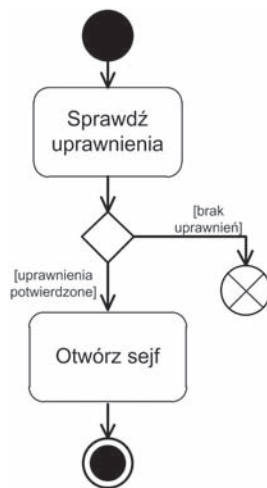
Przy okazji wprowadzono nowy symbol — kartkę z zagiętym rogiem — oznaczający komentarz.

W przeciwieństwie do BPMN, diagram czynności dopuszcza wprowadzanie symboli obiektów wprost na diagramie obiegu⁵.

⁵ BPMN dopuszcza wykorzystanie takich obiektów przy użyciu symboli dodatkowych, które są połączone z obiektami linią przerywaną.

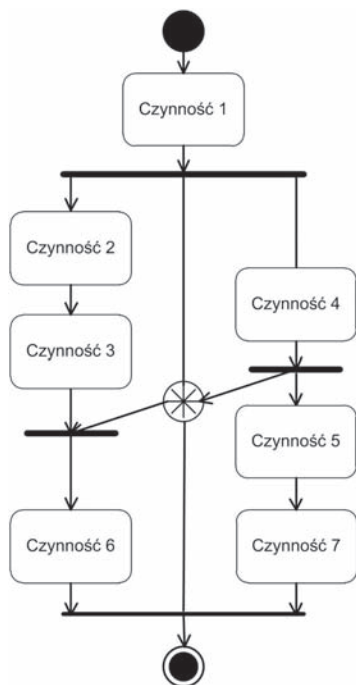


RYSUNEK 3.7. Schemat z symbolem zakończenia procesu i zakończenia gałęzi

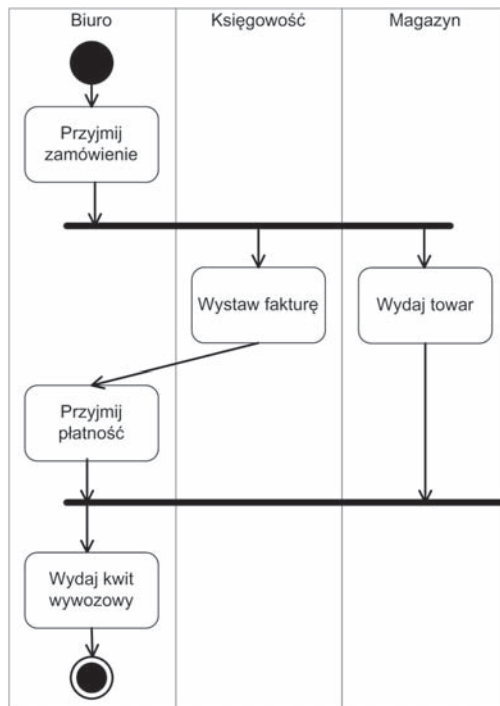


RYSUNEK 3.8. Proces obsługi sejfów elektronicznych

Schematy pokazane na rysunku 3.12 są równoważne; przepływ obiektu można prawidłowo zapisać w każdym z przedstawionych sposobów.



RYСУNEK 3.9. Punkt synchronizacji
— przykład użycia



RYСУNEK 3.10. Proces narysowany przy użyciu pasów pływackich dla określenia aktorów poszczególnych czynności

3.1.2. Diagram stanów

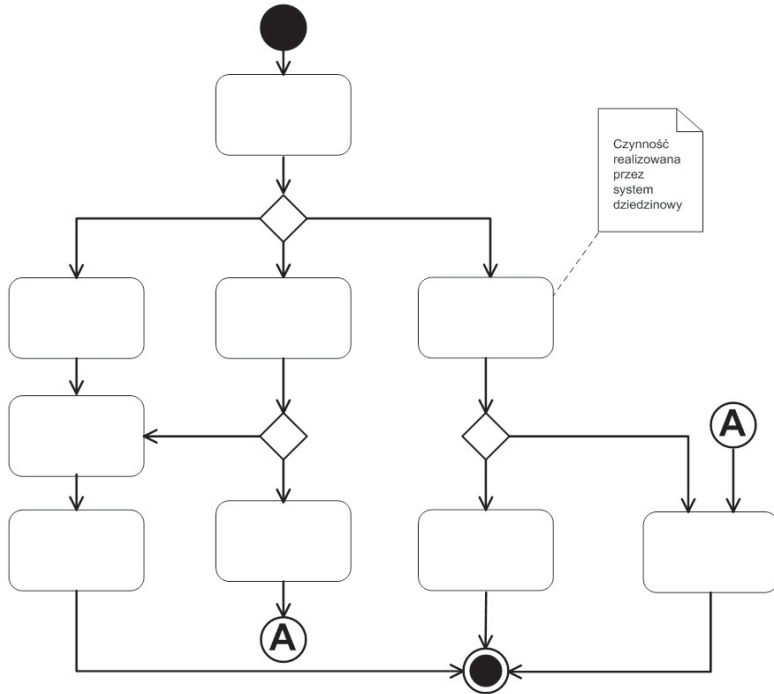
Kolejny diagram traktuje proces jako *automat skończony* — tj. układ, który może przyjmować pewną skończoną (choć niekiedy bardzo dużą) liczbę różnych stanów. Przejścia pomiędzy niektórymi stanami są dozwolone, pomiędzy innymi nie.

Diagram stanów jest w pewnym sensie rewersem diagramu czynności; tak jak tamten skupiał się na operacjach, które muszą zostać wykonane, tak diagram stanów skupia się na stanach, jakie osiągnie system w wyniku ich wywołania.

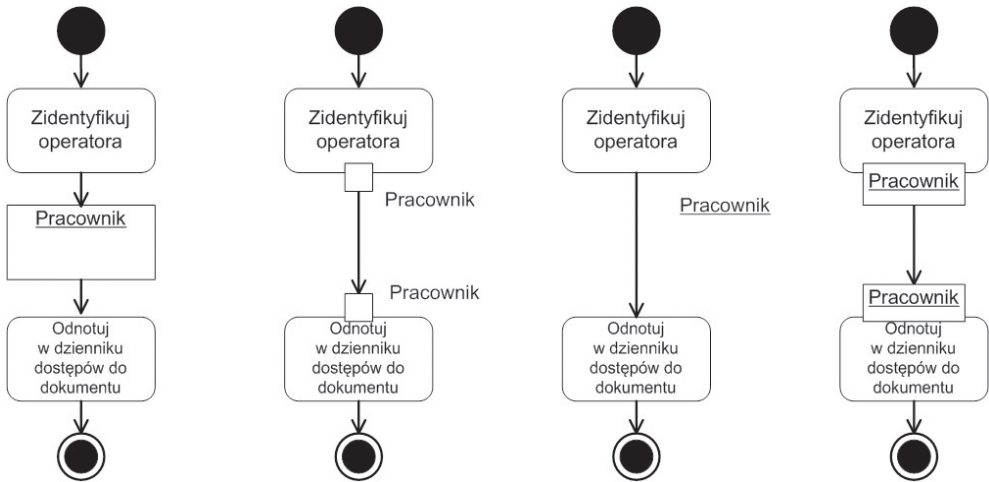
Weźmy bardzo prosty schemat procesu handlowego, przedstawiony w postaci diagramu czynności na rysunku 3.13A. Czynności oznaczone są prostokątami o zaokrąglonych rogach. Celem każdej wykonanej czynności jest zmiana stanu procesu. Poszczególne stany są oznaczone na diagramie czynności tekstem w nawiasach kwadratowych, umieszczonym przy strzałkach oznaczających przejścia pomiędzy operacjami.

Po prawej stronie rysunku 3.13 (schemat B) narysowano ten sam proces w postaci diagramu stanów. Jak widać, jest tu odwrotnie niż na diagramie czynności — prostokąty symbolizują osiągnięte stany, zaś strzałki — czynności powodujące przejście pomiędzy stanami⁶.

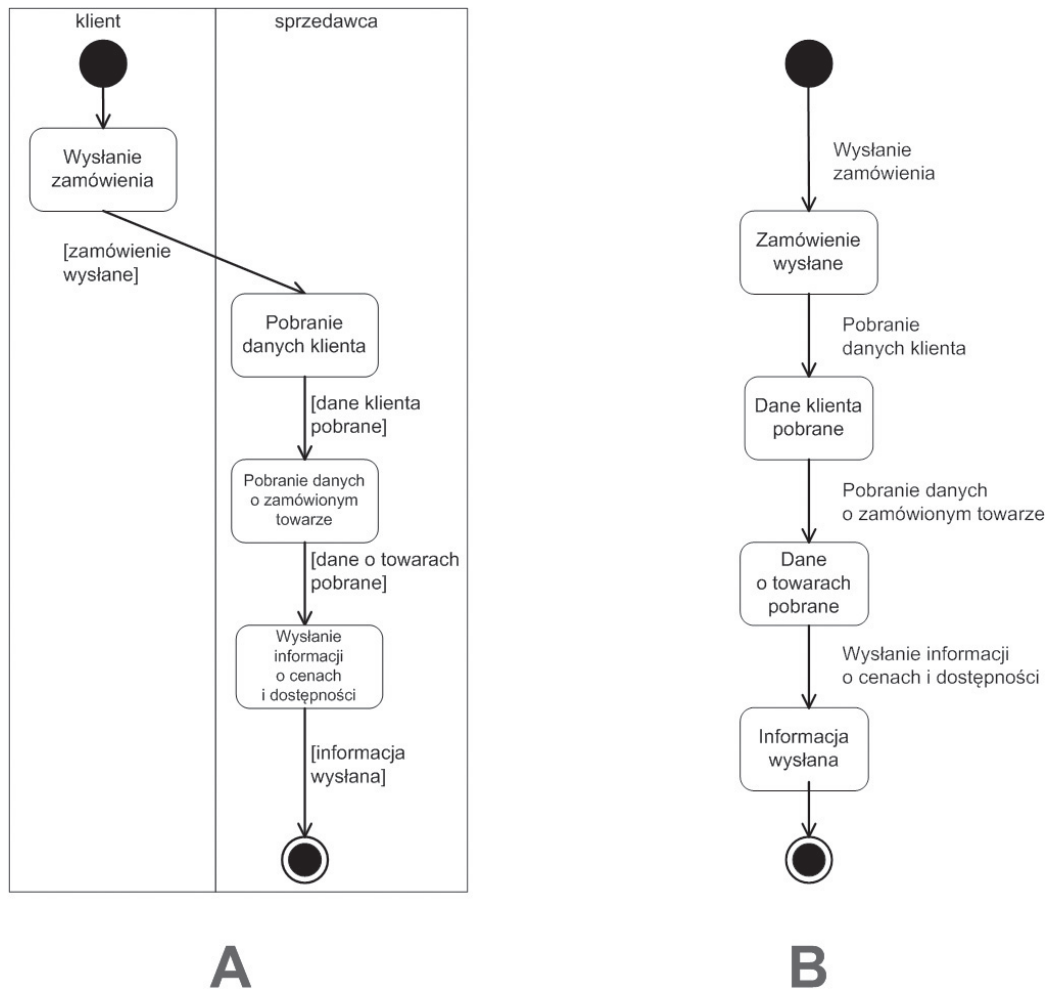
⁶ Do tematu diagramów stanu powrócimy jeszcze przy omawianiu testowania procesów.



RYSUNEK 3.11. Upraszczanie procesu z wykorzystaniem symbolu łączenia



RYSUNEK 3.12. Przepływ obiektów



RYSUNEK 3.13. Prosty proces handlowy: A — diagram czynności, B — diagram stanów

3.1.3. Interakcje po raz pierwszy, czyli diagram sekwencji (przebiegu)

Od razu na wstępie chciałbym zaznaczyć, że opisuję ten diagram w kontekście procesów wbrew mojemu najgłębszemu przekonaniu, iż nie nadaje się on do opisu procesów biznesowych. Robię to tylko dlatego, że diagram ten bywa wymieniany w specyfikacjach wymagań pochodzących od klientów.

3.1.3.1. Klasyfikatory

Diagram sekwencji skupia się na przedstawieniu komunikatów pomiędzy instancjami klasyfikatorów. Czym są klasyfikatory? Mogą to być:

- aktorzy,
- przypadki użycia,
- klasy (instancją jest obiekt),
- interfejsy,
- pakiety,
- komponenty.

Stosowane są cztery rodzaje symboli⁷. Wymieniono je w tabeli 3.3.

TABELA 3.3. Symbole klasyfikatorów

	<p>Aktor Obiekt zewnętrzny wobec systemu. Jeżeli aktor reprezentuje większą grupę (np. w każdym procesie jest jeden klient, ale niemal za każdym razem inny), wówczas można to zaznaczyć, wpisując cięciwę w okrąg oznaczający głowę.</p>	
	<p>Klasa graniczna Reprezentuje interfejs pomiędzy systemem a bytami poza nim.</p>	
	<p>Obiekt klasy danych Reprezentuje dane w systemie.</p>	
	<p>Obiekt klasy sterującej Steruje działaniem klas.</p>	

Klasyfikatory są wyposażone w pionowe, cienkie „linie życia”. Prostokąty na tych liniach odwzorowują czas życia i działania poszczególnych klasyfikatorów.

Oś czasu biegnie z góry na dół; pomiędzy klasyfikatorami przesyłane są komunikaty (rysunek 3.14).

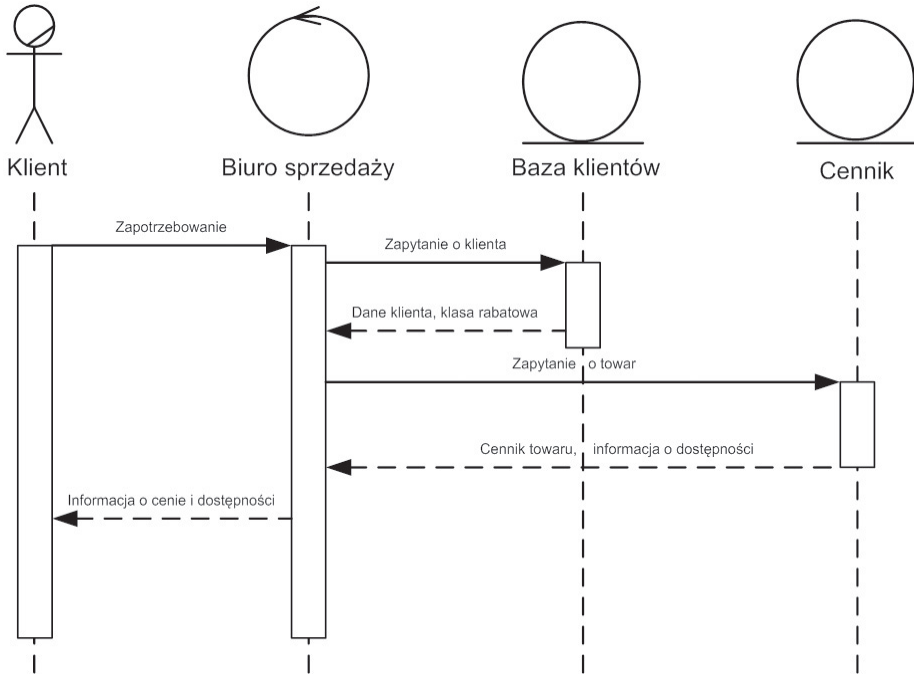
W praktyce zwykle rezygnuje się ze stosowania różnorodnych symboli, rozróżniając jedynie symbol aktora, a pozostałe symbole zastępując prostokątem. W takiej konwencji diagram z rysunku 3.14 wyglądałby jak na rysunku 3.15.

Ogólny schemat opisu obiektów jest następujący:

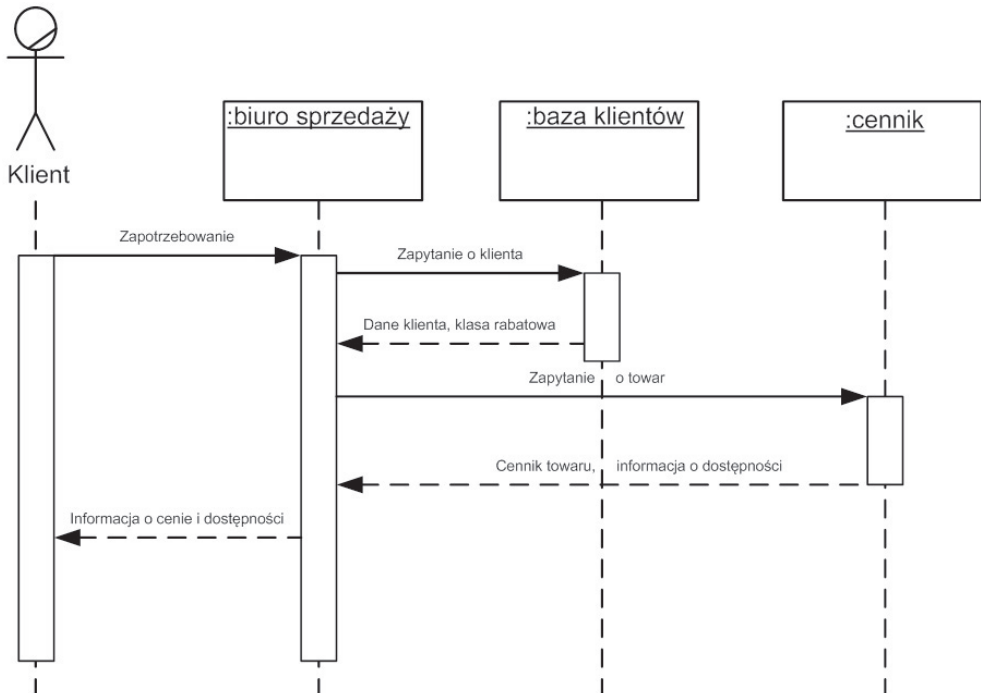
[Nazwa] [:nazwa klasy]

Dozwolone jest zastosowanie części powyższej sekwencji — tak jak zrobiono to na rysunku 3.15.

⁷ Używanie przedstawionych symboli nie jest zbyt powszechne; duża część analityków-praktyków porzeka na symbolu aktora oraz uniwersalnym symbolu obiektu w postaci prostokąta z nazwą w środku (tak jak to pokazano na diagramie komunikacji na rysunku 3.19).



RYСУNEK 3.14. Przykładowy diagram sekwencji — obsługa zamówienia



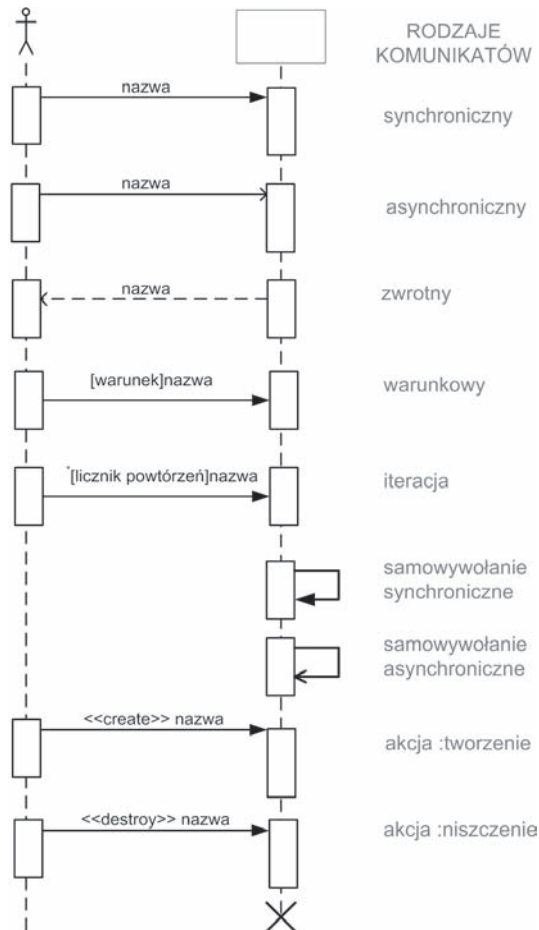
RYСУNEK 3.15. Diagram sekwencji z zastosowaniem uproszczonego opisu obiektów

3.1.3.2. Komunikaty

Wyróżniamy następujące rodzaje komunikatów:

- **asynchroniczny** — klasyfikator wysyłający, który nie czeka z ewentualnym dalszym działaniem na reakcję odbierającego;
- **synchroniczny** — wywołanie procedury;
- **zwrotny** — odpowiedź na wywołanie procedury;
- **akcja** — tworzenie lub niszczenie obiektu;
- **samowywołanie** — asynchroniczne lub synchroniczne.

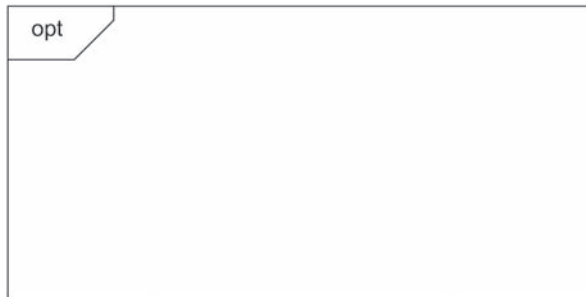
Ponadto komunikaty mogą mieć charakter wielokrotnie powtarzanych interakcji. Mogą też być warunkowe. Rodzaje komunikatów i ich symbole zostały zaprezentowane na rysunku 3.16.



RYСУNEK 3.16. Rodzaje komunikatów i ich symbole

3.1.3.3. Obszary wyodrębnione

Obszar wyodrębniony to fragment procesu, który — w zależności od rodzaju obszaru — powinien zostać potraktowany w specjalny sposób. Symbolem jest prostokąt z oznaczeniem rodzaju obszaru w lewym górnym rogu (rysunek 3.17).



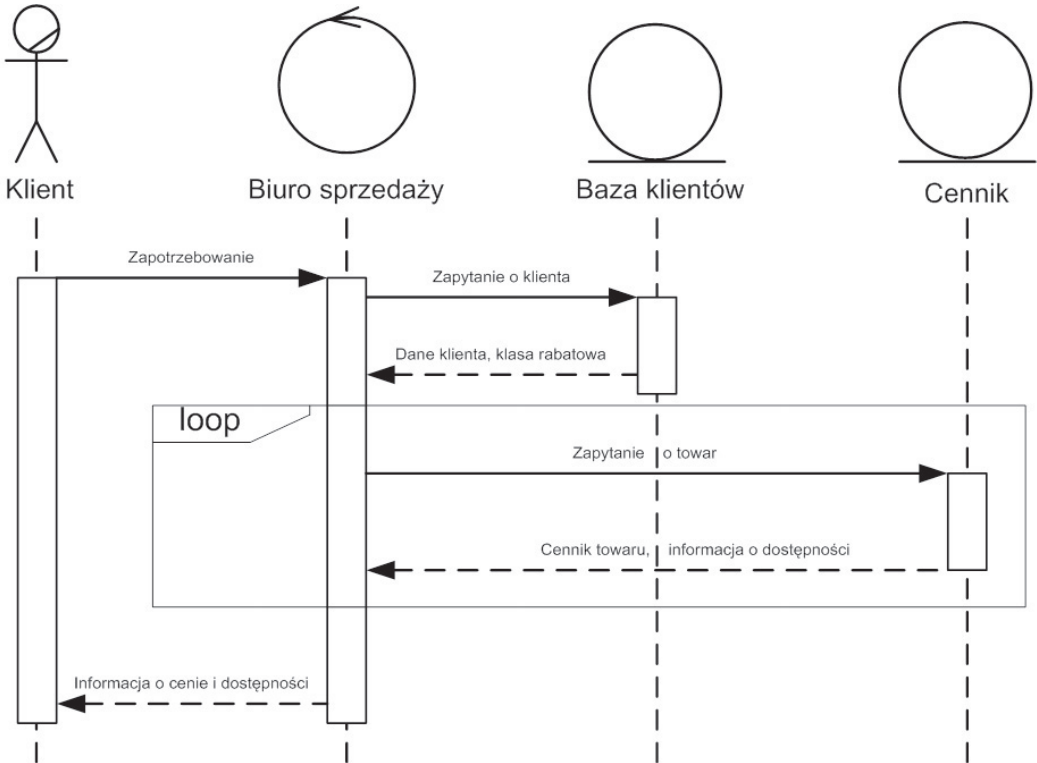
RYСУNEK 3.17. Symbol obszaru wyodrębnionego

Wyróżniamy następujące rodzaje obszarów wyodrębnionych:

- alt* (alternatywa)** — możliwy wybór (na podstawie warunku) tylko jednego operandu fragmentu wyodrębnionego;
- assert* (formuła)** — wykonanie algorytmu;
- break* (przerwanie)** — wykonanie obszaru wyodrębnionego przerywa wykonanie innych interakcji;
- consider* (istotność)** — komunikaty obszaru wyodrębnionego muszą zostać wykonane;
- critical* (krytyczny)** — obszar z najwyższym priorytetem wykonania; klasyfikatory uczestniczące w obszarze krytycznym nie mogą obsługiwać innych operacji (przeciwieństwo *ignore*);
- ignore* (ignorowanie)** — komunikaty w obszarze mają niewielki wpływ na całość interakcji (przeciwieństwo *critical*);
- loop* (iteracja)** — przetwarzanie wielokrotne;
- neg* (błąd)** — obsługa wyjątków;
- opt* (opcja)** — opcjonalne wykonanie operandu w obszarze wyodrębnionym;
- par* (współbieżność)** — jednoczesne wykonanie wszystkich obszarów;
- seq* (następstwo)** — operandy w obszarze muszą być wykonywane dokładnie we wskazanej kolejności (przeciwieństwo *strict*);
- strict* (ściśle)** — komunikaty dla różnych klasyfikatorów mogą wystąpić w dowolnej kolejności (przeciwieństwo *seq*).

Przykład: diagram przedstawiony na rysunku 3.14 opisuje zamówienie jednopozycyjne; zapytanie jest kierowane w sprawie jednego towaru. Jeśli jednak chcemy przedstawić proces

obsługi zamówienia wielopozycyjnego, przydatne będzie zastosowanie obszaru wyodrębnionego *loop* (iteracja). Każda iteracja operandów wewnątrz obszaru będzie dotyczyła jednej pozycji zamówienia (rysunek 3.18).



RYСУNEK 3.18. Diagram z użyciem obszaru wyodrębnionego

Od autora



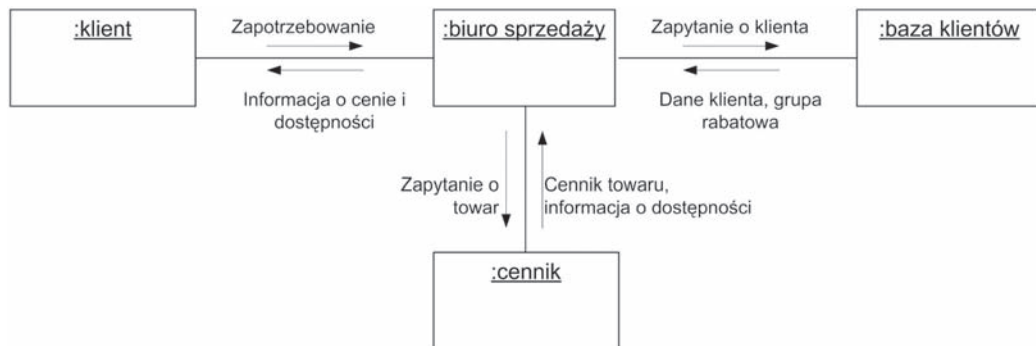
TAK SOBIE MYŚLĘ...

Prawdę mówiąc, nie pojmuję, dlaczego diagram sekwencji uważa się za mechanizm przydatny do opisu procesów biznesowych. Ze swojej natury nadaje się on bardziej do określania konstrukcji oprogramowania niż działań o charakterze biznesowym i tylko przy naprawdę dobrej woli można go uznać za przejrzysty sposób dokumentowania tych drugich. Oczywiście, podany na rysunku 3.18 przykład dowodzi, że można to uczynić, ale... nie polecam.

3.1.4. Interakcje po raz drugi, czyli diagram komunikacji

Diagram komunikacji⁸ obrazuje wzajemne oddziaływanie na siebie obiektów oraz wymianę pomiędzy nimi komunikatów. Jest bardzo prosty; prostokąty oznaczają obiekty, linie — związki pomiędzy obiektami, zaś strzałki — kierunek przepływu komunikatów (rysunek 3.19).

⁸ W poprzednich wersjach UML schemat ten nazywano *diagramem współpracy*.

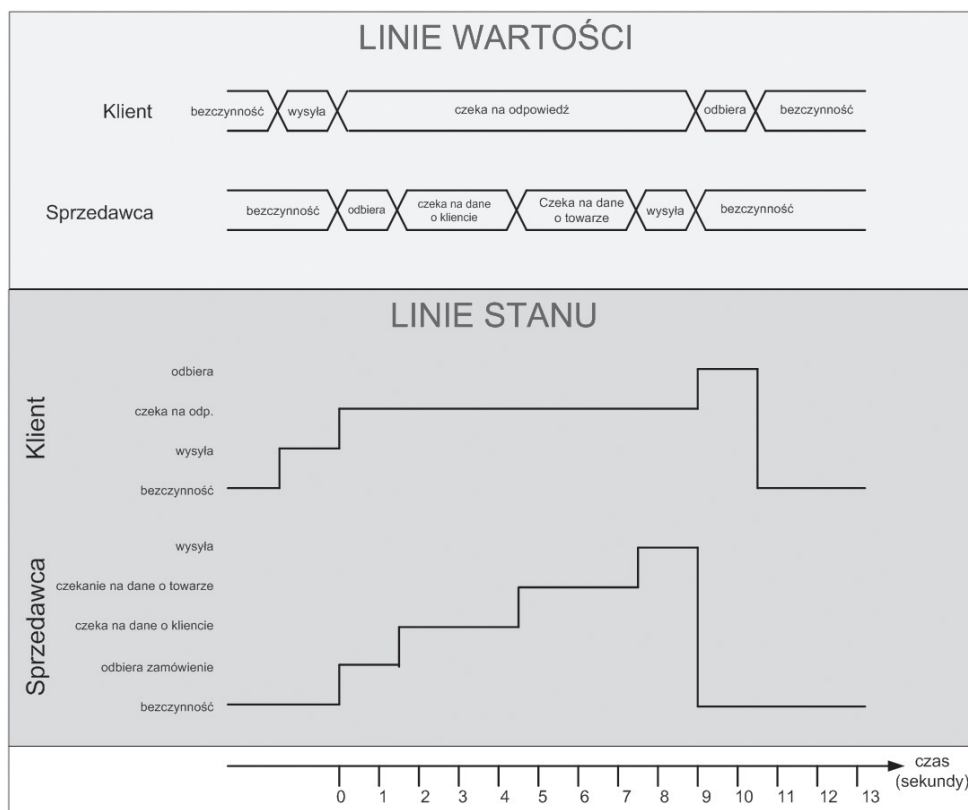


RYСУNEK 3.19. Diagram komunikacji dla przykładu z obsługą zamówienia

3.1.5. Interakcje po raz trzeci, czyli diagram czasowy

Diagram czasowy ma dwie postacie: *linię wartości* (wartości są na niej określone wyłącznie tekstowo) oraz *linię stanów* (prezentuje wykres z wartościami dyskretnymi).

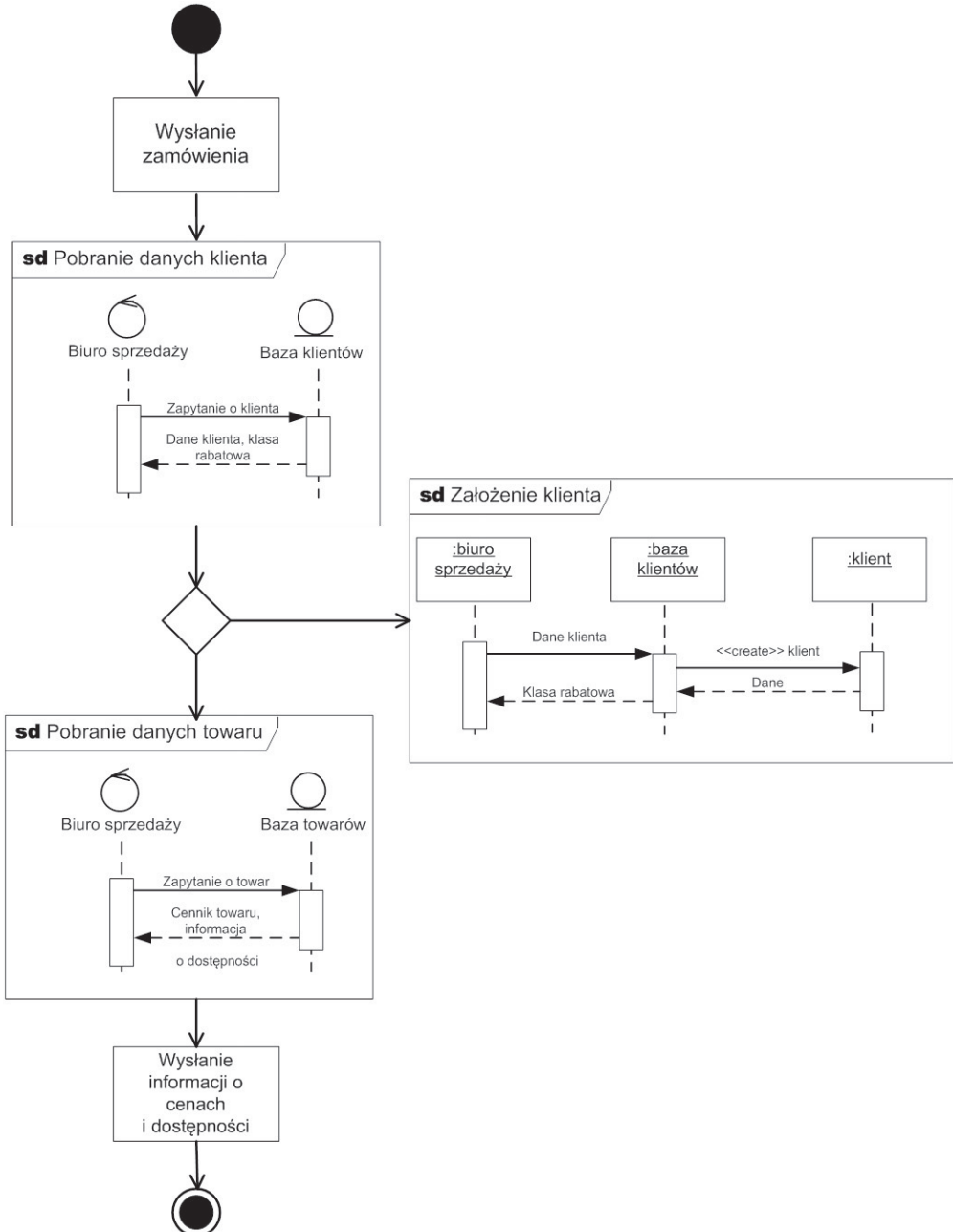
W obu przypadkach linia czasu biegnie z lewej strony do prawej (rysunek 3.20).



RYСУNEK 3.20. Diagram czasowy

3.1.6. Interakcje po raz czwarty, czyli diagram przeglądu interakcji

Diagram przeglądu interakcji to diagram czynności, w którym czynności i/lub akcje zastąpiono diagramami sekwencji (przykład na rysunku 3.21).



RYSUNEK 3.21. Diagram przeglądu interakcji

Diagram ten wprowadzono w wersji 2.0 UML.

Od autora



TAK SOBIE MYŚLĘ

Osobiście uważam — a nie jestem w tym względzie odosobniony — że wprowadzenie diagramu przeglądu interakcji do notacji UML było zbędne. Nie tylko nie łączy on zalet diagramu czynności i diagramu sekwencji, ale ma też wady obu tych schematów.

3.1.7. Diagram przypadków użycia

Diagram przedstawia uczestników (aktorów), usługi świadczone im przez system (przypadki użycia) oraz związki pomiędzy nimi. Nie należy go mylić z **metodą** przypadków użycia przedstawioną w dalszej części rozdziału (choć w tej ostatniej używa się diagramu UML jako jednego z elementów opisu). Diagram ten nie jest notacją procesową; przedstawiam go ze względu na przydatność w trakcie tworzenia opisu procesu.

3.1.7.1. Aktorzy

Symbole

Diagram nadaje się do zobrazowania funkcjonalności systemu. Jego twórcy wyszli z założenia, że system ma taką wartość, jaką mają świadczone przez niego usługi.

A zatem aktorami w podstawowej postaci przypadków użycia są usługobiorcy — użytkownicy systemu. Na diagramie są oni oznaczeni stylizowaną postacią człowieka (rysunek 3.22).

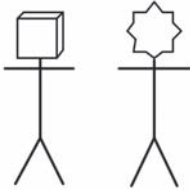


RYСУNEK 3.22. Symbol aktora osobowego

Zdarza się, że system świadczy usługi na rzecz innych systemów informatycznych. Dla obsłużenia takich sytuacji zdefiniowano dwa symbole „maszynowe” — dla zewnętrznych systemów informatycznych i dla urządzeń (rysunek 3.23).

Ostatni rodzaj aktora to wyzwalacz czasowy. Stosuje się go wówczas, gdy któraś z usług systemu jest uruchamiana cyklicznie lub zostaje uruchomiona w określonym terminie. Symbol takiego wyzwalacza pokazano na rysunku 3.24.

W praktyce symbole z rysunków 3.23 i 3.24 są rzadko stosowane; analitycy pozostają przy symbolu aktora osobowego (rysunek 3.22), traktując go jako symbol uniwersalny.



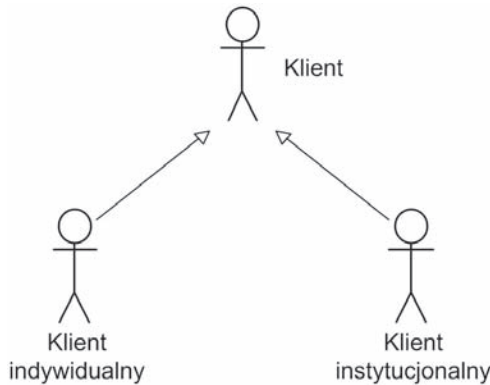
RYSUNEK 3.23. Aktorzy „maszynowi” — system zewnętrzny (po lewej) oraz urządzenie (po prawej)



RYSUNEK 3.24. Aktor — wyzwalacz czasowy

Generalizacja

Jak niemal wszystko w UML, aktorów można generalizować. Na przykład ogólne pojęcie „klient” może oznaczać zarówno klienta indywidualnego, jak i instytucjonalnego. Generalizację zapisujemy jak na rysunku 3.25.



RYSUNEK 3.25. Generalizacja aktorów

3.1.7.2. Przypadki użycia

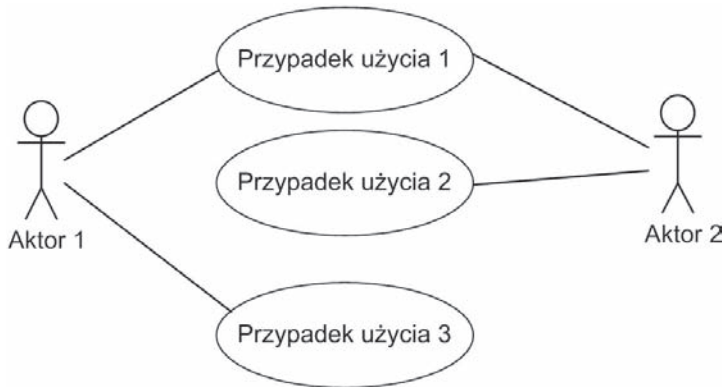
Czym jest przypadek użycia? Jest to zbiór (niekiedy jednoelementowy) akcji⁹, które zaspokajają któreś z potrzeb aktora.

Stosowane symbole

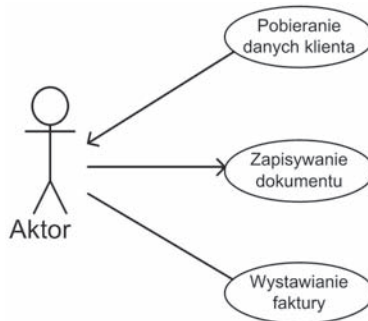
Przypadki użycia to usługi świadczone przez system na rzecz aktorów. Mogą się one składać z pojedynczej czynności lub z całej ich serii. Aby przypisać przypadek użycia do aktora, łączymy ich symbole linią ciągłą. Na rysunku 3.26 z *Przypadku użycia 1* korzystają obaj aktorzy, *Przypadek użycia 2* wykorzystuje Aktor 2, a *Przypadek użycia 3* — Aktor 1.

Niekiedy zaznacza się kierunek przepływu danych pomiędzy aktorem a przypadkiem użycia (rysunek 3.27).

⁹ Akcja jest czynnością „atomową” (niepodzielną i nieprzerywalną) — jest to najniższy poziom dekompozycji operacji systemu.



RYСУNEK 3.26. Przykład diagramu przypadków użycia



RYСУNEK 3.27. Zaznaczanie kierunku przepływu danych

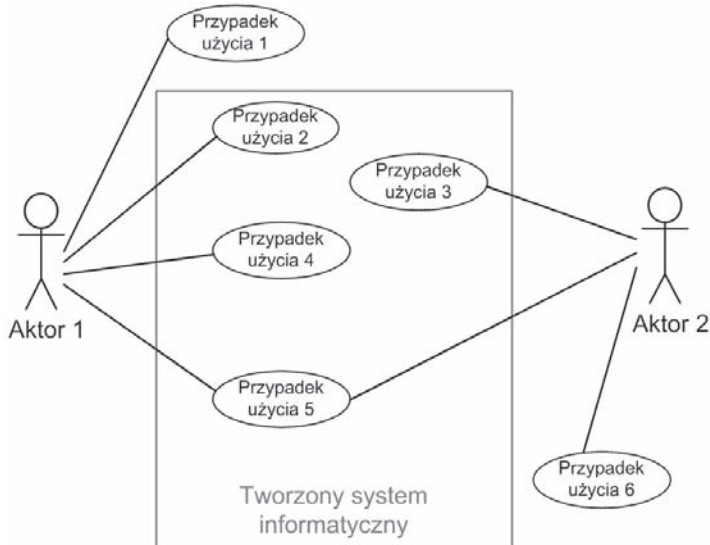
W pierwszym przypadku dane klienta są pobierane z przypadku użycia, w drugim — przypadek użycia pobiera dane, z kolei w trzecim — dane przepływają w obie strony; najpierw do przypadku użycia wysyłane są parametry faktury, a następnie wygenerowana faktura jest przesyłana do aktora. W takim przypadku aktora i przypadek użycia łączy się linią ciągłą (narzucający się symbol strzałki z grotami z obu stron nie jest stosowany).

Określanie granic systemu

Niekiedy może być tak, że opisujemy usługi w systemie biznesowym, a tworzony system informatyczny nie obejmuje ich wszystkich. Innymi słowy, system informatyczny nie zaspokaja wszystkich potrzeb biznesowych. W takim przypadku diagram przypadków użycia staje się bardzo użytecznym narzędziem do określania granic systemu informatycznego.

System jest wówczas symbolizowany przez prostokąt, w którym umieszczone są realizowane przez niego przypadki użycia (rysunek 3.28).

Wymagania reprezentowane na rysunku 3.28 przez *Przypadek użycia 1* i *Przypadek użycia 6* nie będą realizowane w ramach tworzonego systemu informatycznego.



RYСУNEK 3.28. Określanie granic systemu informatycznego

Określanie podziału funkcjonalności systemu między moduły

Warto wspomnieć o jeszcze jednym, nieco kontrowersyjnym zastosowaniu diagramu przypadków użycia, a mianowicie o **dokumentowaniu podziału funkcjonalności na moduły**. Niektórzy robią to, rysując granice modułów tak jak poprzednio granice systemu informatycznego¹⁰. Przykład pokazano na rysunku 3.29.

Wydaje się, że takie użycie diagramu, choć nie do końca zgodne z notacją, jest dość przejrzyste i nie powinno wywoływać nieporozumień. Dla porządku jednak dobrze jest zastosować symbole pakietów (rysunek 3.30).

Zawieranie

Często zdarza się, że dwa przypadki użycia (lub większa ich liczba) mają część wspólną.

W ramach przykładu załóżmy, że mamy prosty system kancelaryjny wspomagający użytkownika w:

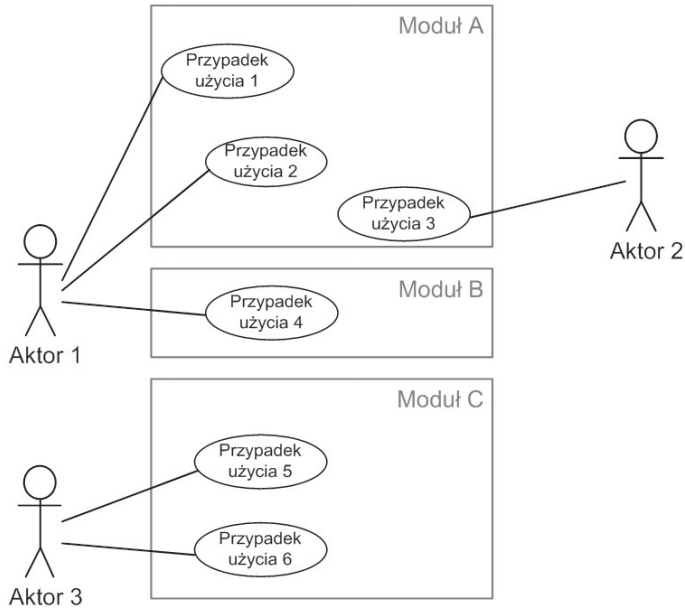
- wysyłce umów,
- wysyłce faktur.

Przykładowy diagram może wyglądać na przykład tak jak na rysunku 3.31.

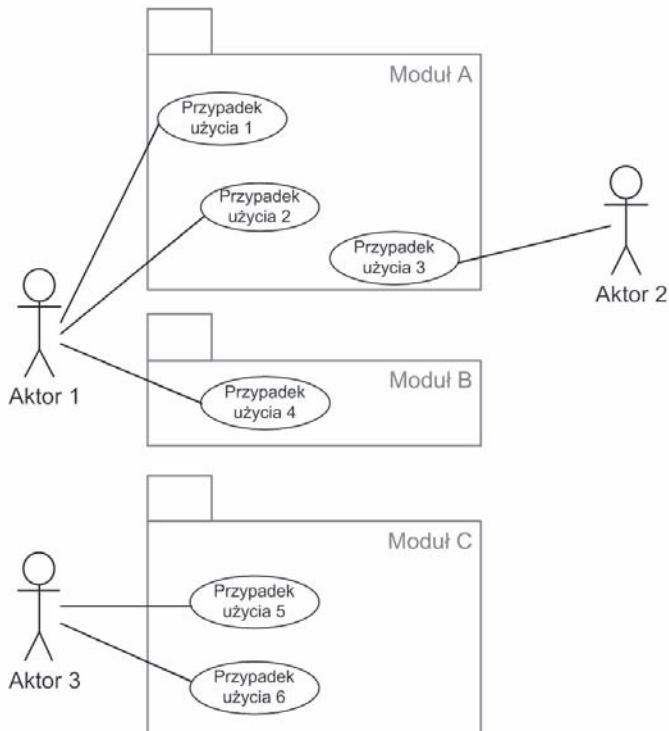
Każdy z tych przypadków składa się z pewnej sekwencji czynności:

- **wysyłanie umów** — pobranie umowy z działu klienckiego, wydrukowanie umowy, zaadresowanie przesyłki;

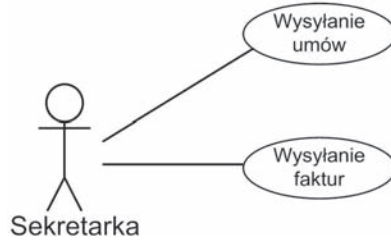
¹⁰ Analitycy nie są zgodni, czy takie zastosowanie diagramu jest zgodne z UML.



RYСУNEK 3.29. Zastosowanie diagramu przypadków użycia do zobrazowania podziału wymagań pomiędzy moduły systemu



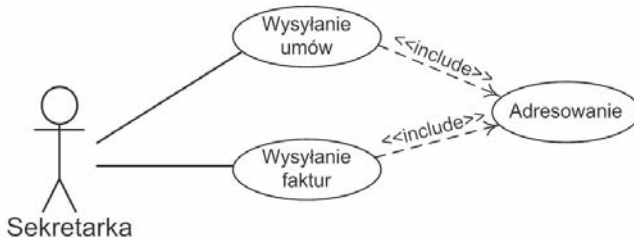
RYСУNEK 3.30. Diagram z rysunku 3.29 z zastosowaniem symboli pakietów



RYSUNEK 3.31. Przykładowy diagram (system kancelaryjny)

- **wysyłanie faktur** — pobranie faktury z rachuby, opieczątowanie, zaadresowanie przesyłki.

Łatwo zauważyć, że wspólną częścią w obu przypadkach jest adresowanie przesyłki. Można to pokazać na diagramie (rysunek 3.32), stosując symbol zawierania (przerywaną linię z pojedynczym grotem i słowem kluczowym `<<include>>`)¹¹.



RYSUNEK 3.32. Diagram systemu kancelaryjnego z wykorzystaniem zawierania

Nazwa „zawieranie” wzięła się stąd, że główny przypadek użycia zawiera (jako fragment swojej listy czynności) listę czynności przypadku łączonego za pomocą relacji zawierania.

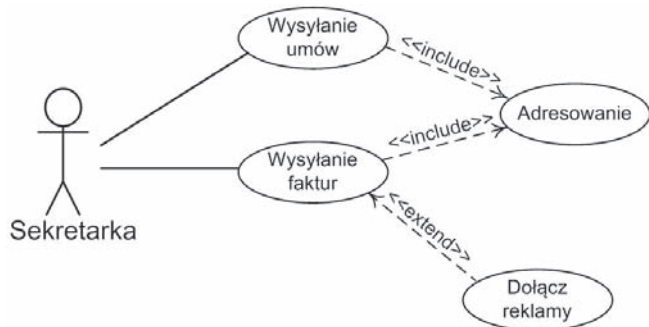
Rozszerzanie

O rozszerzaniu przypadku użycia mówimy wówczas, gdy lista jego czynności zostaje uzupełniona o dodatkowe pozycje. Na przykład przypadek użycia *Wysyłanie faktur* można rozbudować o dodatkową czynność dołączania do faktury druków reklamowych (rysunek 3.33).

Rozszerzenie oznaczamy przerywaną linią z pojedynczym grotem i słowem kluczowym `<<extend>>`¹².

¹¹ Przyjęło się stosować to słowo także na diagramach w języku polskim.

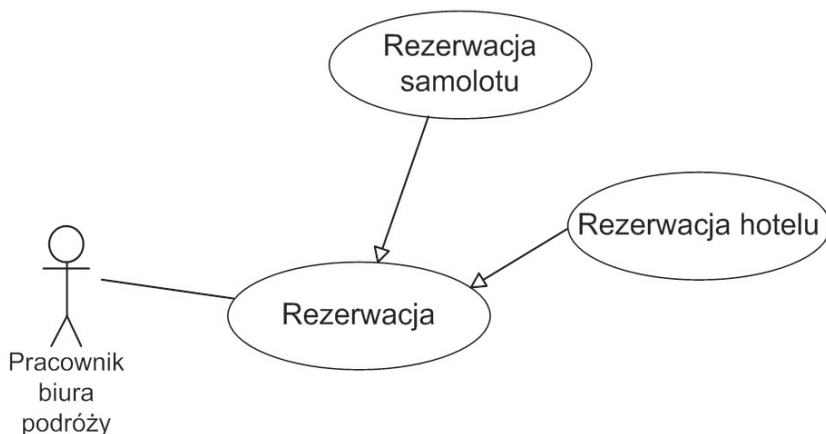
¹² Podobnie jak *include*, stosuje się to słowo w oryginalnym brzmieniu nawet na polskojęzycznych diagramach.



RYSUNEK 3.33. Rozszerzanie przypadków użycia

Generalizacja

Generalizacja przypadków użycia jest co do zasady identyczna z opisaną wcześniej generalizacją aktorów. Przypadek uniwersalny stanowi uogólnienie przypadków szczegółowych (rysunek 3.34).



RYSUNEK 3.34. Generalizacja przypadków użycia

3.2. RAD

Diagram RAD (ang. *Role Activity Diagram*) jest częścią notacji Riva — mało u nas znanej i nieczęsto stosowanej. Może szkoda, ponieważ RAD charakteryzuje się niezwykle prostotą i nadaje się do rozmów z klientami (no, powiedzmy, po 10-minutowym szkoleniu — w każdym razie ta notacja jest prostsza nawet od „starego” BPMN, czyli wersji 1.0). Niestety, na jej podstawie nie da się automatycznie tworzyć definicji procesów w systemach BPM.

Mimo podobieństwa angielskich nazw nie należy tej notacji mylić z diagramem aktywności UML.

Diagram RAD dzięki symbolowi stanu łączy pewne cechy diagramów skupiających się na czynnościach i tych opartych na zapisie stanu procesu.

Muszę przyznać, że mi najbardziej podoba się symbol zygzaka, oznaczający elementy, które nie są dla nas ważne (albo, mówiąc otwarcie, takie, których nie chce nam się opisywać).

Symbole RAD zostały przedstawione w tabeli 3.4.

3.3. PRZYPADKI UŻYCIA

Jak wcześniej wspomniano, opisu z wykorzystaniem przypadków użycia nie należy mylić z *diagramem przypadków użycia* znanym z notacji UML. W gruncie rzeczy opis jest tekstowy, choć bywa wspomagany diagramami BPMN lub diagramem czynności UML. Nierzadko też *diagram przypadków użycia* stanowi wstęp do *opisu przypadków użycia*.

Ideą notacji jest opisanie systemu z punktu widzenia usług, jakie świadczy na rzecz użytkowników¹³.

Wersji opisu za pomocą przypadków użycia jest mnóstwo — niemal tyle co używających go analityków. Poniżej przedstawię wersję, której sam używam. Zawiera ona następujące elementy¹⁴:

1. Numer przypadku użycia i jego nazwa.

Nie należy lekceważyć tej — pozornie nieistotnej — rubryki. Porządne ponumerowanie przypadków użycia według określonego klucza — jeśli trzeba, to nawet w sposób hierarchiczny — stanowi pierwszy krok do zachowania porządku w projekcie.

Układy, sekwencje („maski”) takiego numeru mogą być przeróżne — ja używam następującej:

PU-[3-literowy skrót systemu]-[2-literowy skrót modułu]-[3-cyfrowy numer przypadku]-[nazwa], na przykład dla systemu przyjmowania korespondencji i modułu skanowania pierwszy przypadek użycia wygląda następująco:





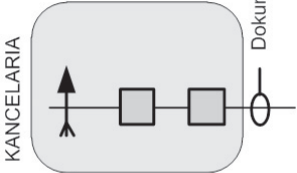
PU-SPK-SK-001-Wprowadzanie dokumentu

2. Diagram przypadków użycia (UML, jak w podrozdziale „Diagram przypadków użycia”) — stosowany tylko wówczas, gdy przypadek użycia jest złożony i można w nim wyróżnić części.

¹³ Słowo „użytkownik” zostało tu użyte w bardzo szerokim sensie; może to być zarówno osoba (rola), jak i system informatyczny.

¹⁴ Wymienione dane często formatuje się w postaci tabeli.

TABELA 3.4. Symbole notacji RAD

Symbol	Opis	Przykład
OPIS 	Symbol roli (notacja podobna jak w BPMN)	
	Zdarzenie rozpoczynające proces	
	Czynność	
	Miejsce, w którym proces przyjmuje określony stan — także zakończenie procesu	<p>KANCELARIA</p>  <p>Wpływ dokumentu</p> <p>Rejestracja</p> <p>Dekretacja</p> <p>Dokument przyjęty</p>

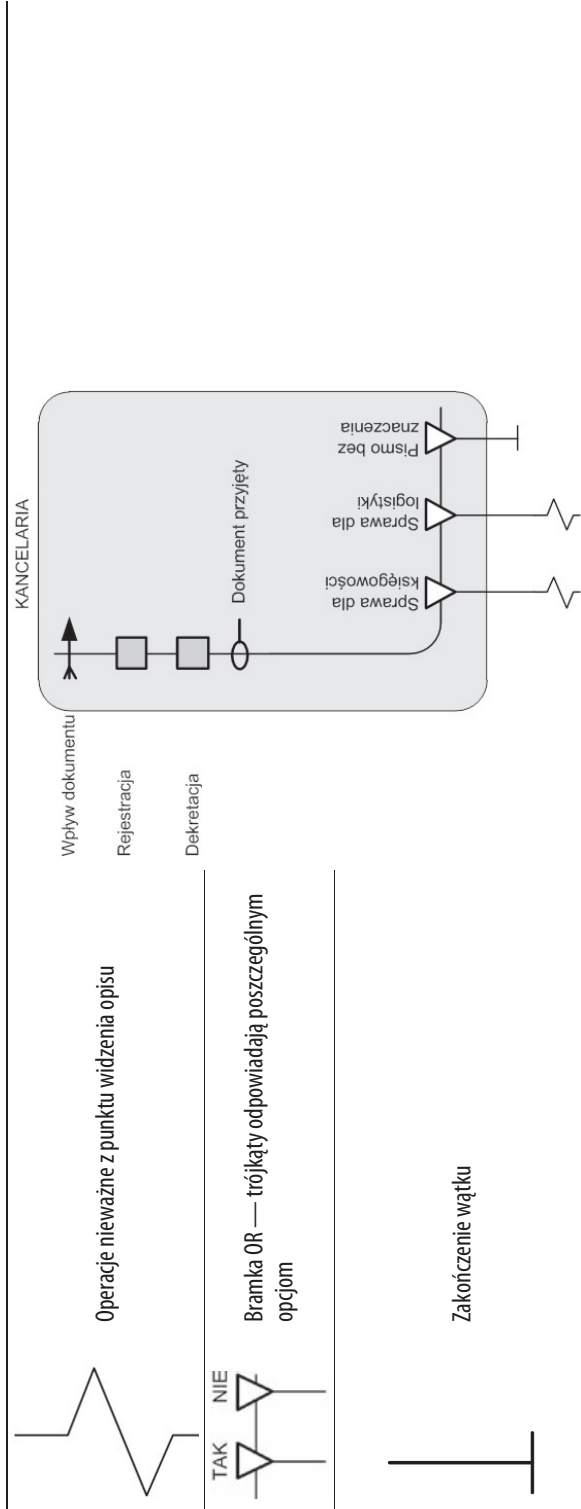

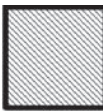

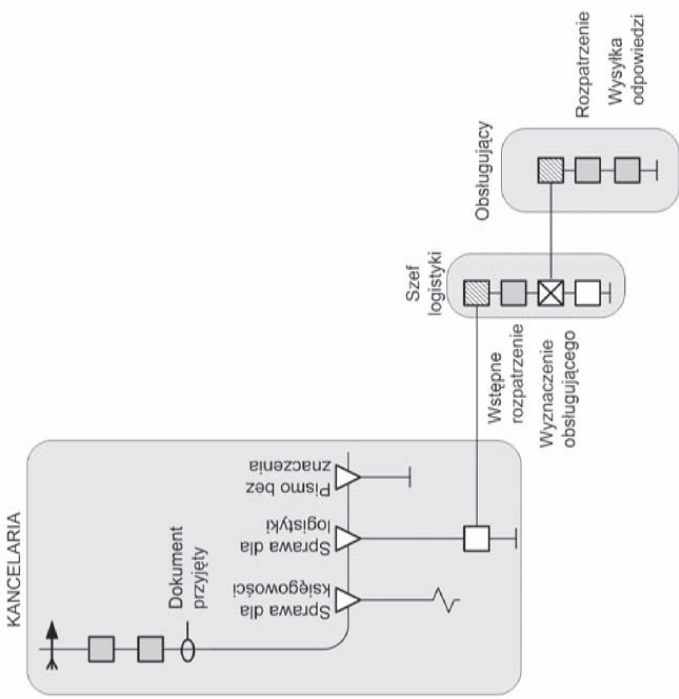
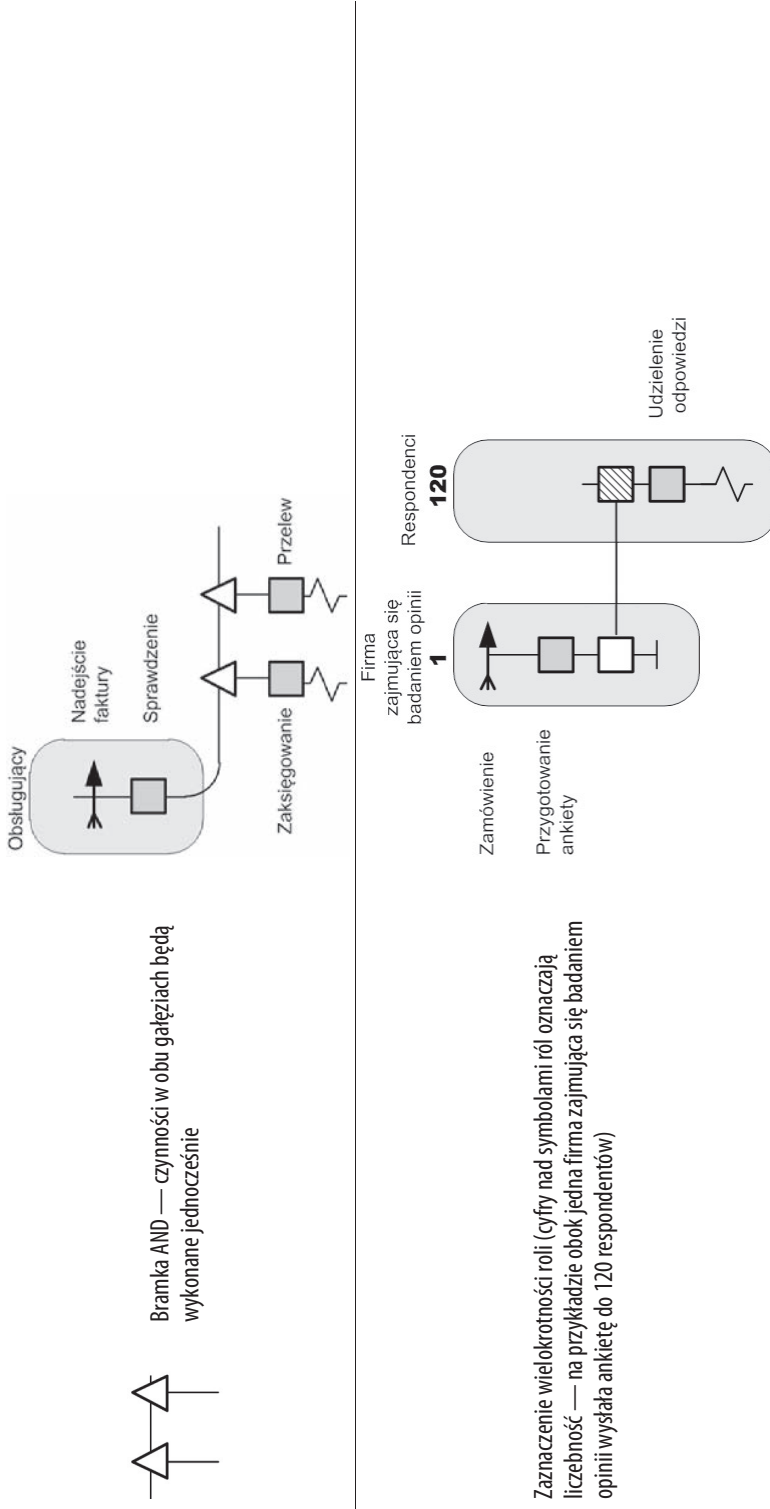


TABELA 3.4. Symbole notacji RAD — ciąg dalszy

Symbol	Opis	Przykład
	Zadanie wysłania komunikatu	Wpływ dokumentu Rejestracja
	Zadanie odebrania komunikatu	Dekretacja
	Utworzenie nowej roli	 <p>The diagram illustrates a process within a 'KANCELARIA' (Office). It shows a flow of documents: 'Sprawa dla księgości' (for the accounting department), 'Sprawa dla logistyki' (for logistics), and 'Pismo bez znaczenia' (without significance). A 'Dokument przyjęty' (received document) is also shown. The process involves several roles: 'Szef logistyki' (Logistics Manager), 'Wstępne rozpatrzenie Wyznaczenie obsługującego' (Preliminary review and assignment of handler), and 'Obsługujący' (Handler). The final steps are 'Rozpatrzenie' (Review) and 'Wysyłka odpowiedzi' (Sending response).</p>



3. Diagram ukazujący wymianę danych z otoczeniem (np. taki jak w rozdziale „Tworzenie opisu procesu”, podrozdział „Proces w otoczeniu (wejścia/wyjścia”).

4. Opis formalny.

Składa się on z następujących elementów:

- a) Streszczenie przypadku użycia (do 5 zdań).
- b) Aktorzy — z wyróżnieniem głównego aktora oraz aktora inicjującego (jeśli jest różny od głównego).
- c) Zdarzenie inicjujące, dane oraz warunki początkowe.
- d) Główny scenariusz powodzenia (wraz z opisem wyniku końcowego)¹⁵.
- e) Scenariusze alternatywne (wraz z opisem wyników).
- f) Wykaz encji biorących udział w scenariuszu¹⁶ [opcjonalnie].
- g) Projekt interfejsu¹⁷ [opcjonalnie].

5. Dodatkowe informacje — diagramy procesów wewnętrznych, bliższe opisy kroków, szkice interfejsu, opisy danych [opcjonalnie].

W tabeli 3.5 pokazano przykład definicji zrealizowanej w formie przypadku użycia.

3.4. FLOWCHART

Flowchart jest notacją opartą na diagramie czynności. Funkcjonalnie ma wiele wspólnego z pierwszą (lepszą!) wersją BPMN. Na rysunku 3.36 przedstawiono schemat procesu sprzedaży towaru: klient przysyła zamówienie (zostaje ono przetworzone przez proces *Przyjęcie zamówienia*, numer *A001*). Następnie zamówienie jest opiniowane i podejmowana jest decyzja. W przypadku gdy jest ona negatywna, sekretariat wysyła do klienta odmowę. W przeciwnym wypadku proces oczekuje na sygnał z banku o wpłacie od klienta; gdy nadejdzie, przelew jest księgowany i magazyn wydaje klientowi towar.

Zaletą notacji jest dokładny, formalny opis zadania (rysunek 3.37), uwzględniający także możliwość wyzwalania zadań przez zdarzenia (w powyższym przykładzie takim wyzwalaczem jest fakt wpływu pieniędzy na konto bankowe). Notacja wskazuje też wykonawcę zadania (z pewnością określenie wykonawcy nie jest tak przejrzyste jak w notacjach z pasami pływackimi, ale za to — ze względu na to, że nie ma potrzeby rozrzucania zadań po pasach — można narysować bardziej zwarte schematy).

¹⁵ Przy bardziej skomplikowanych scenariuszach można umieścić np. ich diagram BPMN (choć niektórzy analitycy uważają to za błąd w notacji).

¹⁶ Element ten właściwie nie należy do notacji; prawdę mówiąc, na etapie sporządzania opisu przypadków użycia encje mogą być jeszcze nieznanne. Mimo to warto uzupełnić ich opis na późniejszych etapach projektowania.

¹⁷ O ile nie przyjęto opcji „projektowania przez interfejs”, część tę uzupełnia się na dalszych etapach tworzenia projektu.

TABELA 3.5. Przykład definicji przypadku użycia

Identyfikator PU	PU-SYD-KA-001-Przyjęcie poczty
Diagram przypadków użycia	
System w otoczeniu	
OPIS FORMALNY	
Streszczenie	Odbiór worków z korespondencją od listonosza, przeliczenie i potwierdzenie zawartości, rejestracja przesyłek w systemie informatycznym, przydzielenie ich konkretnym osobom w firmie.
Aktorzy	Pracownicy biura: 1. Osoba do spraw kontaktu z pocztą (dalej <i>AKTOR1</i>). 2. Osoba (lub osoby) rejestrujące, segregujące i roznoszące przesyłki w firmie (dalej <i>AKTOR2</i>).
Zdarzenie inicjujące	Nadejście korespondencji.

OPIS FORMALNY — ciąg dalszy

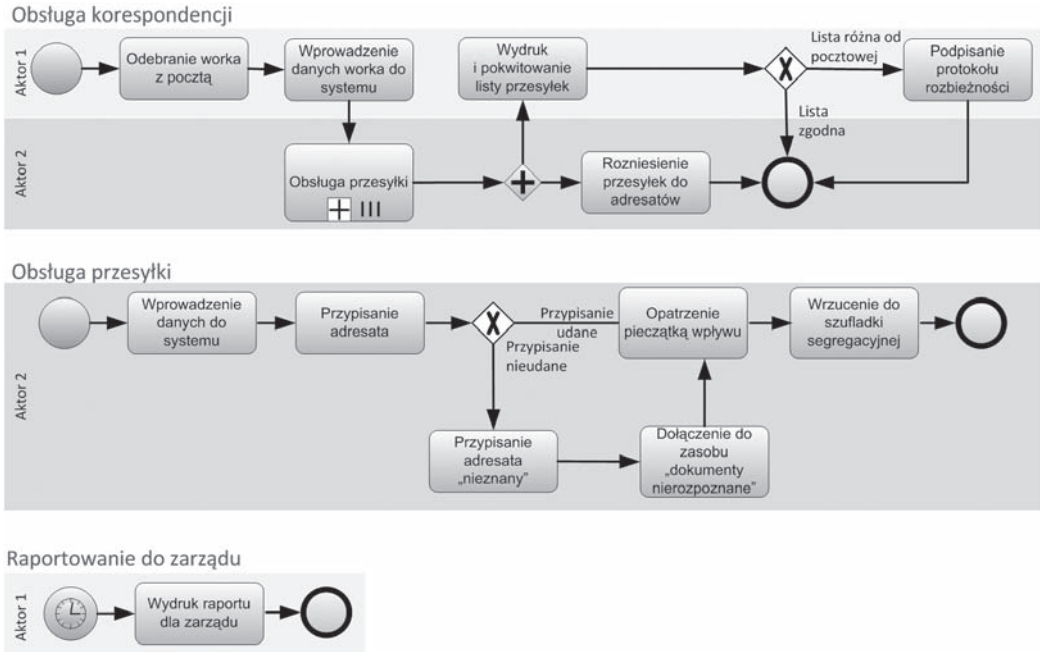
Główny scenariusz powodzenia	<ol style="list-style-type: none"> 1. Odbiór worka z pocztą (<i>AKTOR 1</i>). 2. Wprowadzenie danych dostawy poczty do systemu (data, godzina) (<i>AKTOR 1</i>). 3. Obsługa przesyłek (<i>AKTOR 2</i>). Dla każdej przesyłki: <ol style="list-style-type: none"> a) wprowadzenie danych przesyłki do systemu, b) przypisanie adresata (w systemie), c) opatrzenie pieczęcią wpływu, d) wrzucenie do odpowiedniej szufladki segregacyjnej. 4. Wydruk listy przyjętych przesyłek i pokwitowanie jej (dla listonosza) (<i>AKTOR 1</i>). 5. Rozniesienie przesyłek do adresatów w firmie (<i>AKTOR 2</i>). 6. Wydruk raportu dla zarządu (<i>AKTOR 1</i>, na koniec dnia). <p>Schematy BPMN dla scenariuszy pokazano na rysunku 3.35.</p>
Scenariusze alternatywne	<p>SCENARIUSZ ALTERNATYWNY A</p> <p>W punkcie 4. okazuje się, że lista przyjętych przesyłek nie zgadza się z listą pocztową. W takim przypadku należy dołączyć dodatkową czynność: „4a. Podpisanie protokołu rozbieżności”.</p> <p>SCENARIUSZ ALTERNATYWNY B</p> <p>W trakcie operacji przypisywania adresata (3b) nie udaje się tego zrobić. W takim przypadku należy po czynności 3b dodać dwie inne:</p> <ul style="list-style-type: none"> • Przypisanie przesyłce adresata „nieznany”. • Dołączenie w systemie danych przesyłki do zasobu „dokumenty nierozpoznane”.
Wykorzystywane encje systemu	<ul style="list-style-type: none"> • <i>Klient</i>; • <i>Przesyłka</i>; • <i>Worek korespondencji</i>.

Zwykle schemat zostaje uzupełniony tabelą podobną do tabeli 3.6.

Wyzwalacz nie jest elementem koniecznym — jeśli go brak, czynność zostanie wykonana po skończeniu poprzedniej czynności na schemacie obiegu. Jeśli jednak operacja zostanie wyposażona w wyzwalacz, to — mimo zakończenia poprzedniej operacji na schemacie — nie zostanie wykonana, dopóki nie zajdzie zdarzenie symbolizowane przez ten wyzwalacz.

Wyzwalaczami mogą być:

- zdarzenia zewnętrzne (np. wpływ dokumentu, telefon klienta itd.),
- zdarzenia związane z czasem (np. data upływu terminu zapłaty faktury, ostatni dzień miesiąca albo godzina 18:00),
- zdarzenia pochodzące z innych procesów (np. komunikat o wpisie do ewidencji, komunikat o kompletacji dokumentów itd.),
- zdarzenia systemowe (np. w systemie płacowo-kadrowym — przekroczenie 30 dni zwolnienia w ciągu roku, w systemie bankowym — przekroczenie limitu kwoty naliczonych odsetek za opóźnienie spłaty kredytu).



RYСУNEK 3.35. Procesy z przykładu opisu przypadku użycia przyjęcia korespondencji

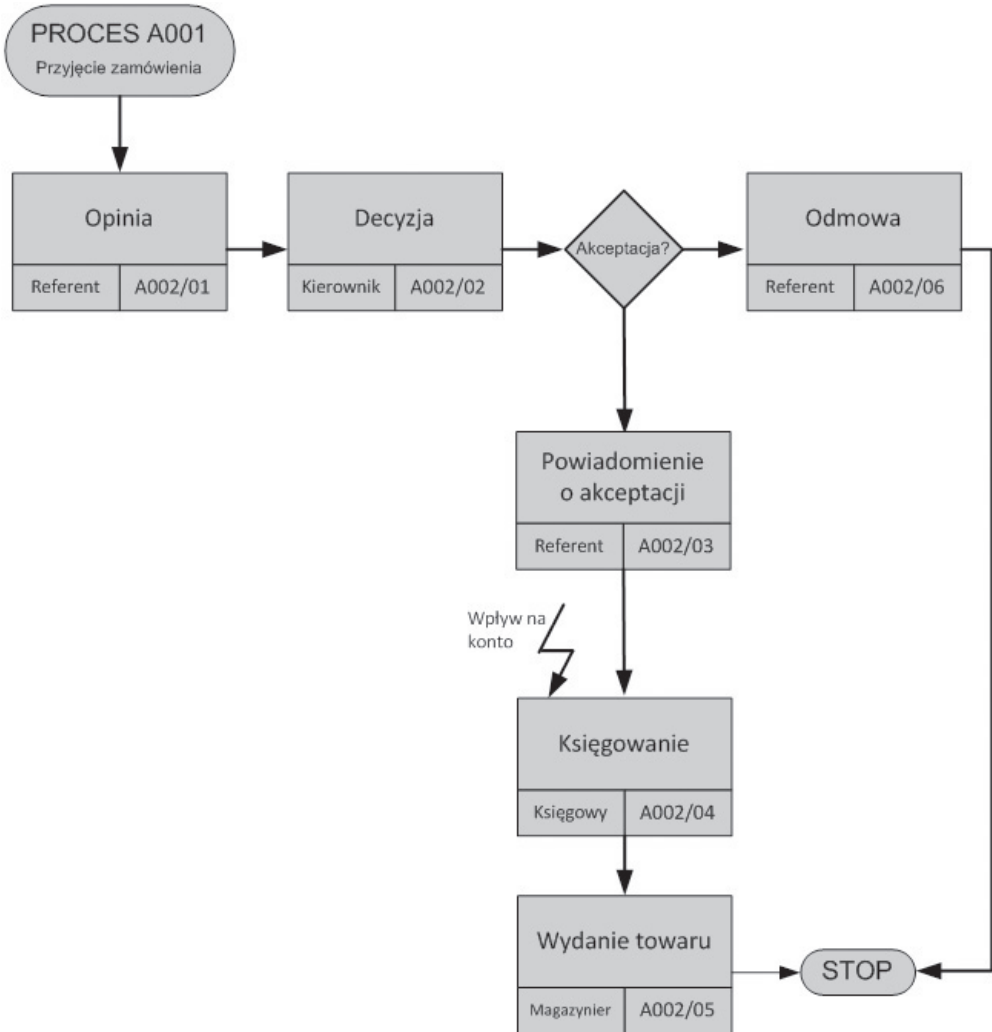
Flowchart nie jest notacją zbyt dobrze zdefiniowaną. Nazwą tą określa się bardzo różne zestawy symboli — od (moim zdaniem najlepszej) przedstawionej na rysunkach 3.36 i 3.37 po zwykle schematy blokowe (jeden z alternatywnych zestawów znaków pokazano na rysunku 3.38).

3.5. DIAGRAM PRZEPŁYWU DANYCH (DFD)

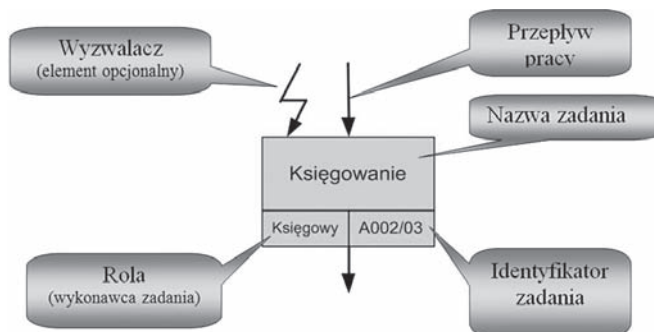
Diagramy DFD (ang. *Data Flow Diagram*) powstały do opisu tego, w jaki sposób dane są przemieszczane i przetwarzane w dowolnych systemach informatycznych, jednak z powodzeniem można je zastosować do opisu procesów.

W diagramie każda pojedyncza operacja jest traktowana jako proces. Dane pochodzą z zewnętrznego źródła (rysunek 3.39), skąd wpływają do systemu. Tam są przekazywane pomiędzy poszczególnymi operacjami, które je przekształcają. Przekształcone dane, do czasu kiedy będą potrzebne innemu procesowi, są przechowywane w magazynach danych.

Na rysunku 3.40 pokazano prosty przykład schematu DFD opisującego firmę handlową. Zbiera ona zamówienia i przekazuje informacje do magazynu dystrybucyjnego, który rozwozi towar do siedzib klientów.



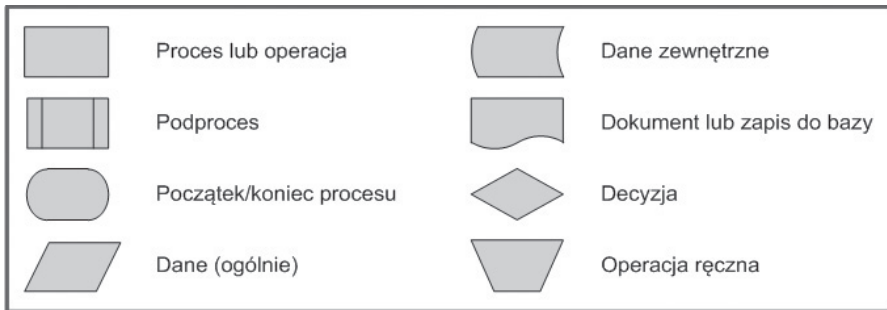
RYSUNEK 3.36. Proces w notacji Flowchart



RYSUNEK 3.37. Elementy opisu zadania

TABELA 3.6. Przykładowa tabela uzupełniająca diagram

	Zadania	Umiejętności/cechy	Wykształcenie	Zasoby
Referent	A002/01, A002/03, A002/06	<ul style="list-style-type: none"> dokładność obsługa urządzeń biurowych 	średnie techniczne	<ul style="list-style-type: none"> komputer MS Excel
Kierownik	A002/02	<ul style="list-style-type: none"> szeroka wiedza na temat branży umiejętność szybkiego podejmowania decyzji 	wyższe menedżerskie	<ul style="list-style-type: none"> komputer
Księgowy	A002/04	<ul style="list-style-type: none"> sumienność znajomość finansów 	wyższe ekonomiczne	<ul style="list-style-type: none"> komputer program księgowy
Magazynier	A002/05	<ul style="list-style-type: none"> uczciwość dokładność 	średnie	



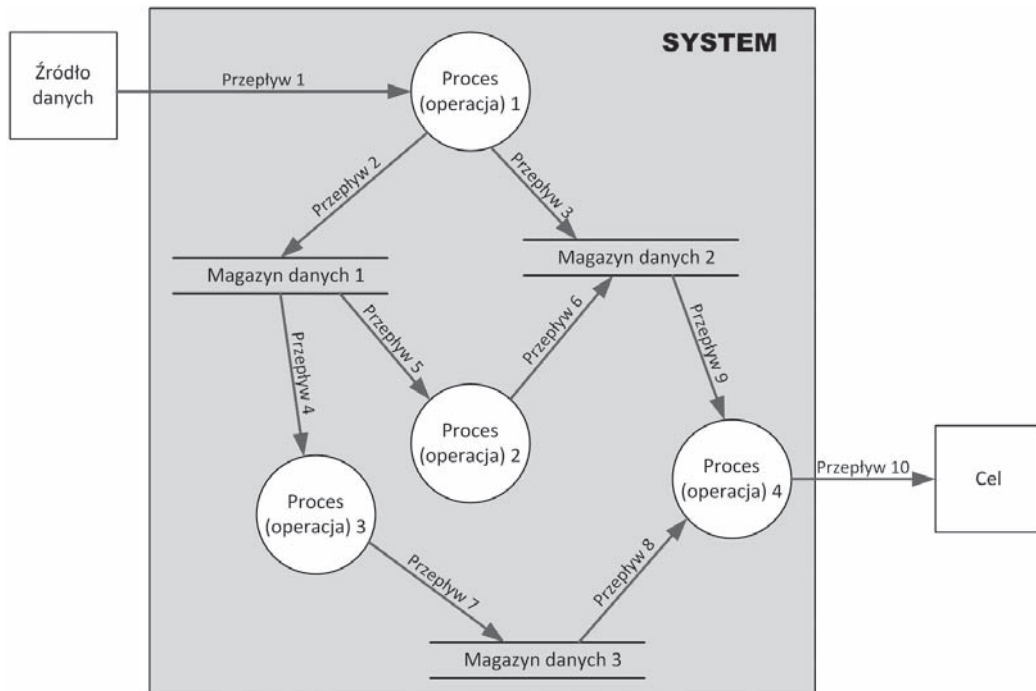
RYSUNEK 3.38. Jeden z wielu alternatywnych zestawów symboli dla notacji Flowchart

Źródłem danych w tym przypadku jest klient. Przekazuje on swoje *Życzenia* (wymagania) do działu kontaktów, który przekłada je na język formalny, tworząc *Zamówienia*. Jednocześnie aktualizowany jest inny magazyn danych — *Dane klientów*. Na podstawie *Zamówień* oraz *Danych klientów* dział zbytu tworzy *Listy dostaw* wysyłane do *Magazynu dystrybucyjnego*.

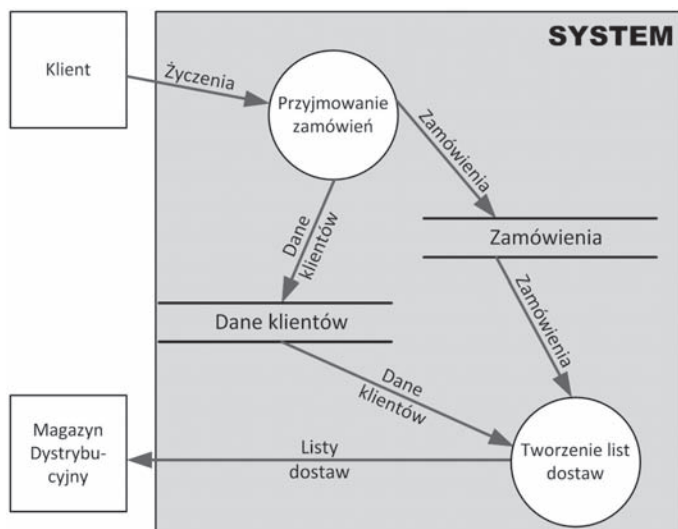
Warto zauważyć, że diagram DFD dostarcza wprawdzie informacji o zadaniach, jakie należy wykonać, lecz w żaden sposób nie określa, kto ma być ich wykonawcą. A zatem fakt, że jedną operację wykonuje dział kontaktów, a drugą — dział zbytu, nie zostaje w żaden sposób zanotowany.

3.6. DIAGRAM STANÓW

Diagram stanów jest stosowany dla procesów, dla których osiągnięte stany są ważniejsze niż sposoby ich osiągnięcia (operacje). Zostanie on bliżej omówiony przy okazji testowania procesów.



RYSUNEK 3.39. Schemat DFD



RYSUNEK 3.40. Praktyczny przykład opisu DFD

3.7. DIAGRAMY KSIĘGI JAKOŚCI

Diagramy księgi jakości są bardzo intuicyjną metodą zapisu procesu. Większość osób (nawet takich, które nie miały wcześniej styczności z procesami biznesowymi i ich analizą) rozumie ten rodzaj diagramów na pierwszy rzut oka.

Stosuje się w nich zaledwie kilka prostych znaków, które zostały przedstawione w tabeli 3.7.

TABELA 3.7. Znaki diagramu księgi jakości














	Uczestnik procesu (rola) — symbol reprezentuje pojedynczego wykonawcę o wyspecyfikowanej nazwie.
[Nazwa]	
	Zbiorowy uczestnik procesu (rola) — symbol reprezentuje grupę wykonawców (np. dział firmy) o wyspecyfikowanej nazwie.
[Nazwa]	
	Automatyczny uczestnik procesu (system informatyczny) o wyspecyfikowanej nazwie.
[Nazwa]	
	Przejście do innej procedury o wyspecyfikowanej nazwie.
[Nazwa]	
	Zdarzenie — w rozumieniu notacji zdarzeniem jest zarówno wpływ dokumentu, jak i jedna z opcji wyboru (przykładowo na rysunku 3.41 po <i>Ocenie formalnej</i> następuje jedno ze zdarzeń: <i>Pismo formalnie poprawne</i> lub <i>Pismo formalnie niepoprawne</i>).
[Nazwa zdarzenia]	
	Bramka LUB (OR) — aktywna co najmniej jedna ze ścieżek wyjściowych (zgodnie z warunkiem w opisie).
[warunek]	
	Bramka ALBO (XOR) — aktywna jedna i tylko jedna ze ścieżek wyjściowych (zgodnie z warunkiem w opisie).
[warunek]	

TABELA 3.7. Znaki diagramu księgi jakości — *ciąg dalszy*

	Bramka I (AND) — aktywne wszystkie ścieżki wyjściowe (rozgałęzienie procesu).
	Artefakt — wytworzony lub przetwarzany dokument o wyspecyfikowanej nazwie.
[Nazwa]	
	Artefakt — wytworzony lub przetwarzany e-mail.
[opis]	
	Artefakt — wytworzony lub przetwarzany dokument w postaci sformalizowanej (formularz) o wyspecyfikowanej nazwie.
[Nazwa]	
	Artefakt — wytworzony lub przetwarzany dokument raportu o wyspecyfikowanej nazwie.
[Nazwa]	
	Rejestr o wyspecyfikowanej nazwie.
[Nazwa]	

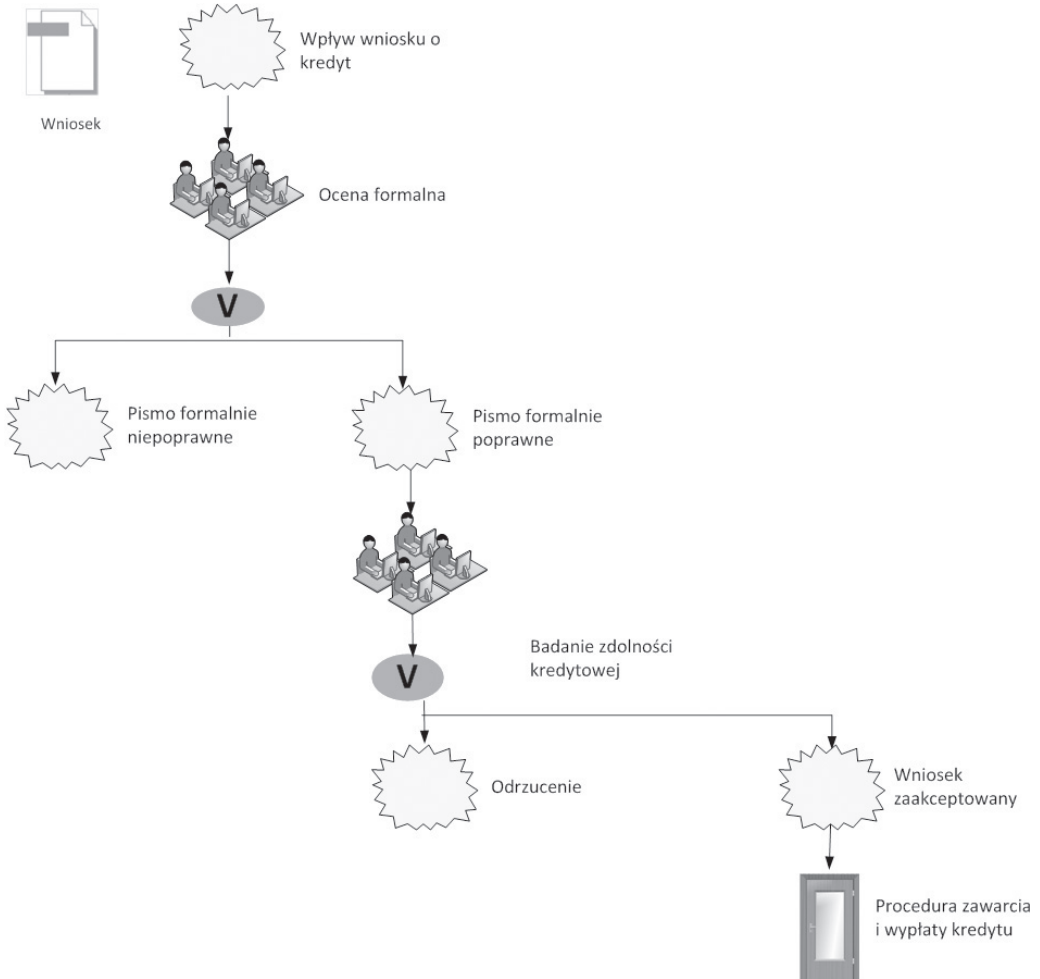
Na rysunku 3.41 pokazano prosty przykład procesu przyznawania kredytu.

3.8. SERVICE BLUEPRINT

Service blueprint nie jest w zasadzie odrębną notacją, jej elementy — linia interakcji i linia widoczności — mogą być po prostu wprowadzane do innych notacji. Notacja skupia się na określeniu działań z silnym podziałem na działania podejmowane w bezpośredniej relacji z klientem i te wykonywane w tle, które oddziela tzw. *linia widoczności* (rysunek 3.42).

Z lewej strony opisani są wykonawcy (można to utożsamiać z BPMN-owymi rolami). Opisy nad rysunkiem prezentują niezbędne elementy wyposażenia lub artefakty. Czarne okręgi z literą B w środku to oznaczenie miejsc, gdzie następuje najczęściej błędów.

Metodę często stosuje się do opisu usług, w przypadku których dochodzi do bezpośredniego kontaktu z klientem.

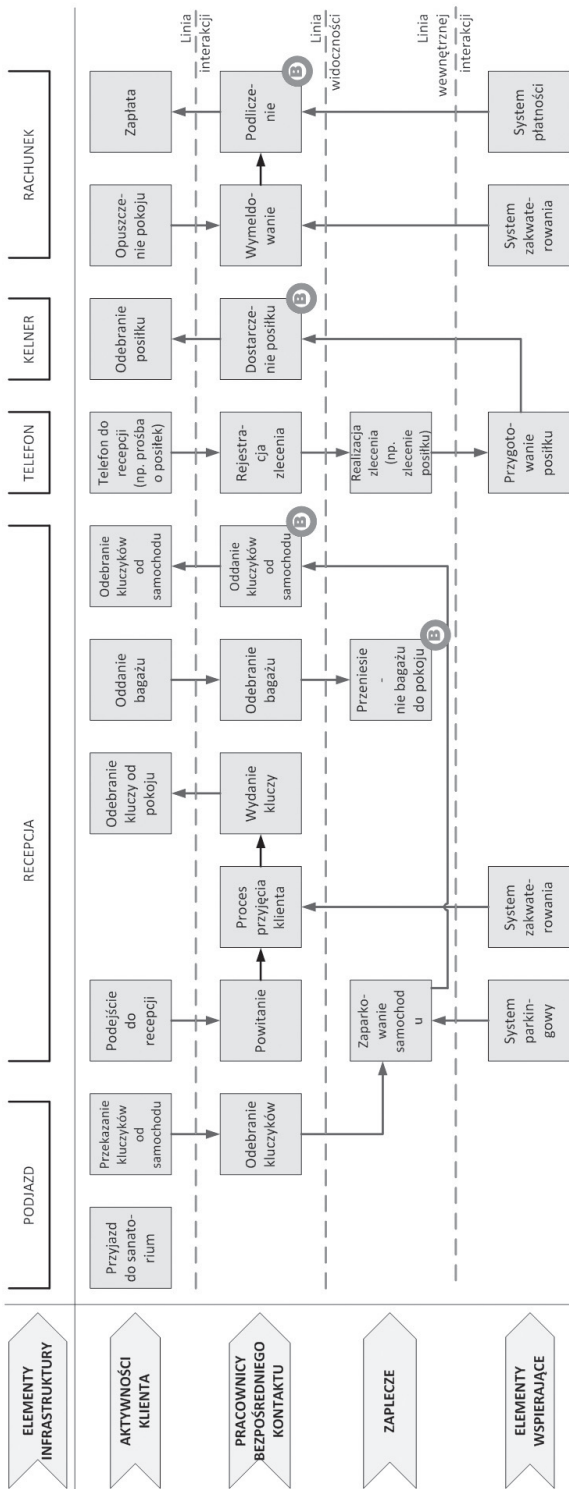


RYСУNEK 3.41. Przykładowy proces przyznawania kredytu

3.9. NIERYSUNKOWE METODY ZAPISU

Oprócz klasycznych, obrazkowych notacji wypracowano — z różnych zresztą powodów — szereg sposobów zapisu procesu opartych na tabelach. Trudno je właściwie nazwać notacjami (notacja powinna bowiem posiadać swoją semantykę i syntaktykę)¹⁸, nie wydaje się też, by były one wystarczające do samodzielnego opisu procesu — mogą jednak stanowić cenne uzupełnienie. Poniżej zostaną przedstawione dwie takie metody.

¹⁸ Patrz ramka na początku rozdziału.



RYСУNEK 3.42. Przykład wykorzystania usługi Service blueprint

3.9.1. Macierz RACI

Metoda zakłada istnienie czterech ról dla każdej czynności:

Responsible — wykonawca zadania,

Accountable — decydent,

Consult — konsultant,

Inform — osoba informowana o rezultatach wykonania zadania.

Wyobraźmy sobie prosty proces sprzedaży ubezpieczenia; agent odwiedza klienta i przekonuje go do sprzedaży; podpisany wniosek zostaje oceniony pod względem ryzyka przez dział underwritingu¹⁹, wreszcie, zawarte ubezpieczenie jest oceniane przez dział analiz. Macierz RACI będzie wyglądała tak, jak to przedstawiono w tabeli 3.8.

TABELA 3.8. Macierz RACI

Zadanie → rola	Sprzedaż	Ocena ryzyka	Ocena wyników
Agent	R	IC	I
Szef zespołu	AI	I	
Dział underwritingu		R	
Dyrekcja		A	IC
Dział analiz	C		AR

3.9.2. Macierz SIPOC

Metoda macierzy SIPOC została zaczerpnięta z narzędziowni Six Sigma. Założeniem początkowym jest, że każdy proces można opisać, specyfikując następujące elementy:

Supplier — dostawca zasobów wejściowych (*input*) do procesu,

Input — zasoby wejściowe,

Process — sposób, w jaki zasoby wejściowe (*input*) są przekształcane w wynik procesu (*output*),

Output — wynik procesu,

Customer — podmiot, dla którego jest przeznaczony wynik procesu.

SIPOC ma tę zaletę, że opisuje proces wraz z jego otoczeniem biznesowym. Macierz bywa niekiedy rozszerzana o dwa dodatkowe elementy: *Resources* (zasoby niezbędne do wykonania procesu) oraz *Rules* (regulacje prawne).

Tabela 3.9 przedstawia przykładową macierz SIPOC, skonstruowaną dla procesu sprzedaży i realizacji usługi budowy marmurowych schodów w domach jednorodzinnych.

¹⁹ Dział w firmach ubezpieczeniowych zajmujący się oceną ryzyka i definiujący warunki sprzedawanych polis.

TABELA 3.9. Przykładowa macierz SIPOC

Supplier	Input	Process	Output	Customer
<ul style="list-style-type: none"> • właściciel domu • deweloper • sprzedawca domokrężny • sprzedawca telefoniczny 	<ul style="list-style-type: none"> • zamówienie • materiały budowlane • materiały wykończeniowe • betoniarka • piła do marmuru • architekt • kierownik budowy z uprawnieniami 	<pre> graph TD A[Pozyskanie zamówienia] --> B[Projekt] B --> C[Uzyskanie zezwolenia] C --> D[Budowa] D --> E[Podpisanie odbioru, wydanie gwarancji] E --> F[Przyjęcie płatności] </pre>	<ul style="list-style-type: none"> • kontrakt • zezwolenie budowlane • schody • pisemna gwarancja • płatność 	<ul style="list-style-type: none"> • właściciel domu

PROGRAM PARTNERSKI

GRUPY WYDAWNICZEJ HELION



- 1. ZAREJESTRUJ SIĘ**
- 2. PREZENTUJ KSIĄZKI**
- 3. ZBIERAJ PROWIZJĘ**

Zmień swoją stronę WWW
w działający bankomat!

Dowiedz się więcej i dołącz już dzisiaj!

<http://program-partnerski.helion.pl>

GRUPA WYDAWNICZA

 **Helion SA**

Biznes procesowo uporządkowany

W tej książce, napisanej przez specjalistę informatyka od piętnastu lat zajmującego się procesami biznesowymi, poznasz zagadnienia dotyczące tych procesów z perspektywy praktycznej. Może ona okazać się tym cenniejsza, że autor opisuje, jak to wszystko działa w polskich warunkach, a ponadto omawia zagadnienia dotyczące zarówno przedsiębiorstw komercyjnych, jak i urzędów publicznych. Dlatego właśnie należy spodziewać się raczej porad praktycznych niż wykładu o charakterze akademickim. Miejsce naukowych definicji zajmuje praktyka oparta na wieloletnim doświadczeniu i... popełnionych przez autora błędach, na których teraz możesz się uczyć.

Marek Piotrowski skupia się na czterech blokach zagadnień. Rozdziały od 2 do 5 traktują o formalnych aspektach projektowania procesów, o procesie w ogóle, o używanych notacjach i rodzajach obiegów. Następna część zawiera omówienie praktycznych zagadnień związanych z procesami biznesowymi, najczęściej popełnianymi błędami, rozwiązaniami typowych problemów spotykanych w warunkach polskich przedsiębiorstw i optymalizacją. Część trzecia mówi o testowaniu i pomiarach procesów. Ostatni, rozdział mówi o tym, jakie zmiany w organizacji implikuje wprowadzenie struktury zorientowanej na procesy.



Marek Piotrowski — absolwent Politechniki Gdańskiej, od dwudziestu trzech lat zajmujący się informatyką, od piętnastu — procesami biznesowymi w ramach informatyki. W 2007 roku ukazała się jego pierwsza książka: *BPMN — notacja modelowania procesów biznesowych*.

książki **klasy** business

Nr katalogowy: 13574



Księgarnia internetowa:

<http://onepress.pl>



Zamówienia telefoniczne:

0 801 339900



0 601 339900

o n e
p r e s s

Sprawdź najnowsze promocje:

• <http://onepress.pl/promocje>

Książki najchętniej czytane:

• <http://onepress.pl/handbellery>

Zamów informacje o nowościach:

• <http://onepress.pl/nowości>

Helion SA

ul. Kościuszki 1c, 44-100 Gliwice

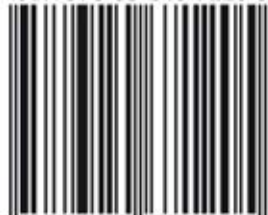
tel.: 32 230 98 63

e-mail: onepress@onepress.pl

<http://onepress.pl>

Cena 59,00 zł

ISBN 978-83-246-7120-5



9 788324 671205