

PROCESY
BIZNESOWE
W POLSKICH
WARUNKACH

MAREK PIOTROWSKI

PROCESY BIZNESOWE W PRAKTYCE

> PROJEKTOWANIE, TESTOWANIE I OPTYMALIZACJA

WYDANIE II

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/probi2>
Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

ISBN: 978-83-283-2521-0

Copyright © Helion 2016

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 (doraźne)</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 ramek 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>Postępowanie 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 „Bezkraina 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 przesylek	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. Pomiar	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. Podejście departamentowe a podejście procesowe	347
11.1.2. Firma zorientowana procesowo	352
11.1.3. IT — integracja istniejących w firmie systemów informatycznych	356
11.1.4. Wymuszanie stosowania procedur jakości	358
11.1.5. Korzyści na poziomie stanowiska pracy	359
11.2. Model dojrzałości procesowej organizacji CMMI	359
11.3. Zmiany, zmiany, zmiany... ..	360
Rozdział 12. Wspomaganie projektowania procesów przy zastosowaniu Enterprise Architect	363
12.1. Interfejs Enterprise Architect	363
12.2. Przykład I — diagramy procesów w Enterprise Architect	364
12.2.1. Tworzenie modelu	365
12.2.2. Tworzenie widoku	368
12.2.3. Tworzenie pakietu	369
12.2.4. Tworzenie diagramu procesu	371
12.2.5. Podłączanie procesu podrzędnego pod aktywność w procesie głównym	384
12.2.6. Proces podrzędny — diagram Decyzja	385
12.2.7. Formatowanie artefaktów	389
12.3. Przykład II — rejestracja przypadków użycia w Enterprise Architect	392
12.3.1. Tworzenie diagramu przypadków użycia	393
12.3.2. Rejestrowanie atrybutów przypadku użycia	398
12.3.3. Rejestrowanie scenariusza przypadku użycia	401
12.4. Zarządzanie stopniem zaprojektowania przypadków użycia	411
12.4.1. Statusy przypadków użycia	411
12.4.2. Definiowanie własnych statusów	411
12.4.3. Śledzenie stopnia zaprojektowania przypadków użycia	415
12.5. Gdzie szukać pomocy	420
Dodatek A Informacje pomocnicze	421
Dodatek B Zalecana lektura	429

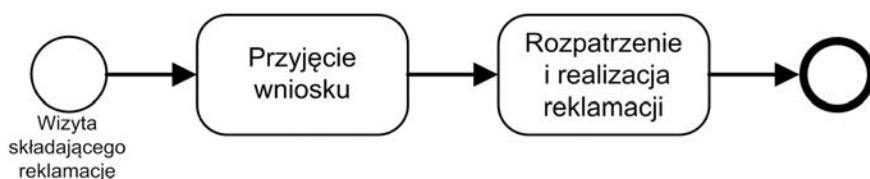
Rozdział 2.

Notacja BPMN

Notacja BPMN (ang. *Business Process Model and Notation*) została stworzona specjalnie do opisywania procesów biznesowych i stanowi dziś podstawowy sposób ich zapisu — dlatego został jej poświęcony osobny rozdział. Pierwotnie składała się tylko z jednego diagramu (ang. *Business Process Diagram*). W wersji 2.0 zestaw diagramów wzbogacono o diagramy choreografii i kolaboracji (współpracy).

Rozpoczniemy od zdefiniowania prostego procesu, który — w miarę wprowadzania nowych symboli — będziemy komplikować.

Przedstawiony na rysunku 2.1 przebieg obrazuje proces rozpatrywania reklamacji w maksymalnie uproszczonej wersji.



RYСУNEK 2.1. Proces reklamacyjny

Proces składa się z dwóch działań (zwanych *krokami* lub *zadaniami*)¹: przyjęcia wniosku reklamacyjnego oraz rozpatrzenia reklamacji.

Rozpoczyna się wizytą osoby składającej reklamację — mówimy, że jest to dla procesu *zdarzenie inicjujące* lub *zdarzenie początkowe*. Zdarzenie początkowe oznaczamy okręgiem narysowanym cienką linią.

Zakończenie procesu nazywamy umownie *zdarzeniem końcowym* i rysujemy jako okrąg narysowany grubą linią.

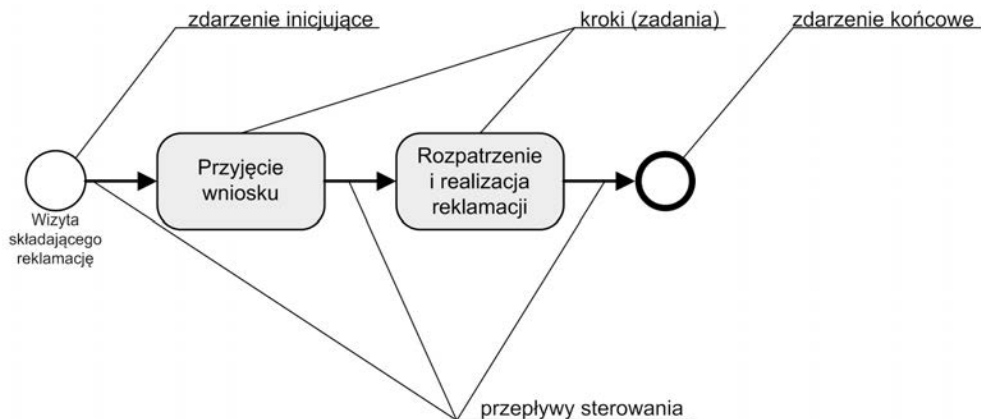
Symbole zdarzeń mogą być uściślane przez dodatkowe znaki (patrz rysunek 2.2).

¹ W notacji BPMN konsekwentnie stosowane jest określenie „zadanie”. W dalszej części książki będziemy się posługiwać następującymi terminami:

Operacja — pojedyncza, samodzielna aktywność w procesie.

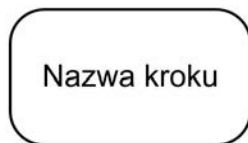
Krok — zespół operacji wykonywanych bezpośrednio po sobie (w tej samej sesji) przez tego samego aktora (w szczególnych przypadkach krok może się składać z jednej operacji).

Zadanie — ogólne pojęcie mogące oznaczać zarówno operację, jak i krok.



RYСУNEK 2.2. Najważniejsze elementy opisu procesu

Krok oznaczamy w BPMN prostokątem o zaokrąglonych rogach z wpisaną pośrodku nazwą (rysunek 2.3).



RYСУNEK 2.3. Symbol kroku (zadania)

Zdarzenie początkowe, kroki i zdarzenie końcowe są połączone strzałkami wskazującymi *przepływy sterowania* (czyli, mówiąc po ludzku, pokazują kolejność, w jakiej są wykonywane kroki).



Wyobraźmy sobie akcję piłkarską składającą się z następujących elementów:

- wybiecie piłki przez bramkarza (po gwizdku sędziego);
- prowadzenie piłki przez pomocnika i podanie skrzydłowemu;
- prowadzenie piłki przez skrzydłowego i podanie do napastnika;
- strzał w stronę bramki i gol.

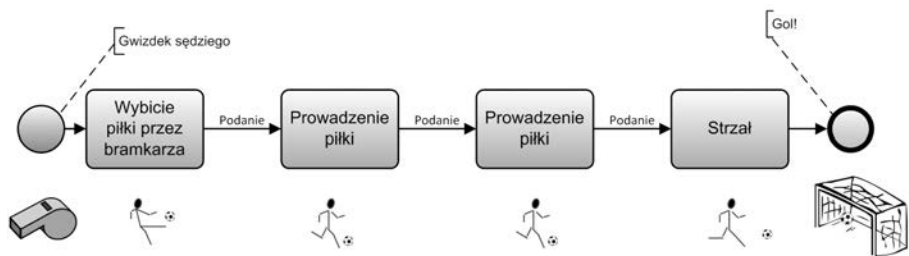
Gdyby narysować tę akcję w postaci procesu w notacji BPMN, diagram wyglądałby jak na rysunku 2.4.

Rysunek pozwala prześledzić podstawowe elementy procesu: **zdarzenia** oznaczają wystąpienie pewnych stanów (lub właśnie zdarzeń, takich jak gwizdek sędziego czy strzelenie gola), **przepływy** są odpowiednikiem podawania piłki przez zawodników, a wykonywane przez nich **czynności** (w tym przypadku np. bieg z piłką) odpowiadają zadaniom procesu.

Opis elementów diagramu przedstawia tabela 2.1.

Przepływy w naszym piłkarskim przykładzie są bardzo wyraźne — podania nie da się nie zauważyć, jest ono wynikiem świadomego działania piłkarza. Jednak wspomniane przejścia nie w każdym procesie są jawne — niekiedy mogą oznaczać jedynie kolejność (następstwo) zadań.

Jak widać, przepływy nie są jakimś fizycznym przekazaniem tokena — pokazują po prostu, którą czynność należy podjąć po skończeniu aktualnie wykonywanej.

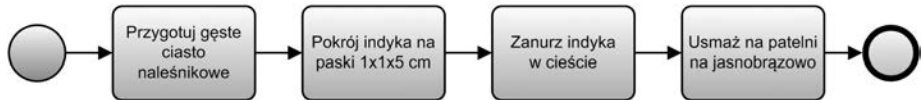


RYSUNEK 2.4. Proces „akcja piłkarska”

TABELA 2.1. Opis elementów diagramu z rysunku 2.4

Nazwa zadania	Zadanie (czynność wykonywana w trakcie procesu) oznaczamy w postaci prostokąta z zaokrąglonymi rogami: w jego środku będziemy wpisywać nazwę zadania. Zadaniami w przypadku akcji piłkarskiej są: - wybiecie piłki przez bramkarza; - bieg z piłką; - strzał na bramkę przeciwnika.	
	Zdarzenie inicjujące oznaczamy kółkiem otoczonym cienką linią. Zdarzeniem inicjującym do rozpoczęcia akcji piłkarskiej jest gwizdek sędziego.	
	Zdarzenie końcowe (osiągnięcie wyniku procesu) oznaczamy kółkiem narysowanym grubą linią. W naszym procesie celem będzie gol.	
	Przepływ sterowania (inaczej: przekazanie tokena)² oznaczamy strzałką. W naszym procesie odpowiada on podaniu piłki do kolejnego gracza.	
	Jeśli chcemy dodać jakiś komentarz do któregoś z elementów podpisu, możemy wykorzystać symbol adnotacji. W obszarze tekstowym wpisujemy treść notatki, a koniec przerywanej linii powinien się łączyć z elementem, którego opis dotyczy.	

Rysunek 2.5 definiuje przebieg procesu smażenia indyka w cieście.



RYSUNEK 2.5. Proces przygotowywania indyka w cieście

² Pojęcie „token” można wyjaśnić, odwołując się do harcerskiej zabawy, w której może mówić tylko ten, kto akurat ma w rękę totem (odpowiadający naszemu tokenowi). W procesie jest wykonywana ta czynność, w której jest umieszczony umowny token — czyli innymi słowy, aktualny krok procesu (pojęcie „token” zostało dokładniej omówione w podrozdziale „Obieg typu workflow” w rozdziale 5.).

W przykładowym piłkarskim procesie tokenem jest po prostu piłka — czynność wykonuje zawodnik, który jest w jej posiadaniu.

2.1. RODZAJE ZADAŃ

Symbolce zadań mogą zostać uzupełnione ikonami określającymi ich rodzaj (tabela 2.2).

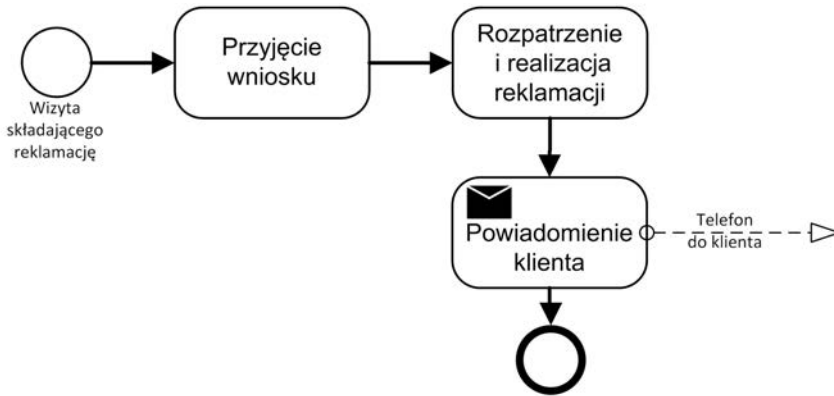
TABELA 2.2. Rodzaje zadań

	Zadanie „Usługa” Zadanie wykonywane przez system w sposób automatyczny. Może (ale nie musi) być związane z usługą sieciową. Usługę może wykonywać np. system dziedziny.
	Zadanie wysyłające Zadanie wysyłające komunikat do podmiotu spoza procesu ³ . UWAGA: wysyłka odbywa się w sposób automatyczny! Jeżeli na przykład zadanie jest realizowane przez użytkownika wysyłającego e-mail, należy posłużyć się opisanym niżej symbolem osoby ().
	Zadanie odbierające Zadanie odbierające komunikat od podmiotu spoza procesu. UWAGA: podobnie jak wcześniej wysyłka, odbiór odbywa się w sposób automatyczny.
	Zadanie odbierające inicjujące (rozpoczynające proces) Zadanie odbierające komunikat inicjujący od podmiotu spoza procesu. UWAGA: odbiór odbywa się w sposób automatyczny.
	Zadanie użytkownika Zadanie realizowane przez użytkownika przy użyciu jakiegoś elementu systemu komputerowego.
	Zadanie ręczne Zadanie realizowane przez użytkownika poza systemem komputerowym (np. spinanie kartek, stemplowanie, wkładanie pisma do segregatorów itp.).
	Zadanie skryptowe Zadanie realizowane na podstawie skryptu przez silnik procesowy.
	Zadanie „Reguła biznesowa” Zadanie wykonywane przez silnik reguł biznesowych „na zlecenie” silnika procesowego (proces dostarcza i odbiera dane, silnik reguł biznesowych je przetwarza). Może służyć np. do sprawdzania limitów, agregacji itd. (patrz podrozdział „Obieg definiowany za pomocą silnika reguł” w rozdziale 5.).

³ Do „innego basenu” — patrz podrozdział „Reprezentacja interakcji z podmiotami zewnętrznymi”.

W opisie rodzajów zadań wspomniano o *komunikatach*. Komunikaty są informacją wpływającą z procesu biznesowego lub docierającą do niego (często wpływającą z jednej jego części i docierającą do innej). Najistotniejszym zastosowaniem komunikatów jest łączność pomiędzy procesami biznesowymi przebiegającymi w różnych organizacjach (czyli „basenach” — będzie o nich mowa w podrozdziale „Reprezentacja interakcji z podmiotami zewnętrznymi”).

Komunikat oznaczamy przerywaną linią zakończoną niewypełnionym grotem strzałki o trójkątnym kształcie. Na rysunku 2.6 oznaczono nią komunikat powiadomienia klienta o rozpatrzeniu reklamacji.



RYСУNEK 2.6. Proces reklamacji uzupełniony o komunikat o jej rozpatrzeniu

2.2. ROZGAŁĘZIANIE PROCESU

Niestety, bardzo rzadko procesy są zwykłymi listami czynności do wykonania. Rozgałęzienia stosujemy w dwóch podstawowych sytuacjach:

- gdy przebieg procesu zależy od spełnienia jakichś warunków,
- gdy dwie lub więcej czynności mogą być wykonywane jednocześnie.

Najczęściej stosowanym mechanizmem rozgałęziania są *bramki*.

2.2.1. Bramka ALBO (XOR)

Najbardziej „naturalnym” rodzajem jest **bramka ALBO** (określana też jako **XOR**). Stosowana jest wówczas, gdy stoimy przed wyborem mającym tylko jedno poprawne rozwiązanie. W najprostszym przypadku jest to wybór: Tak/Nie (rysunek 2.7).

Gałęzi wyjściowych może być więcej. Istotne jest to, że zawsze tylko jedna z gałęzi spełnia warunek bramki (patrz rysunek 2.8).

Rozdział 5.

Rodzaje obiegu

W tym rozdziale zostaną scharakteryzowane rodzaje obiegu; będzie to wstęp do omówienia najbardziej typowych przebiegów w dalszej części książki.

5.1. PODZIAŁ ZE WZGLĘDU NA SPOSÓB DEFINIOWANIA ŚCIEŻKI OBIEGU

5.1.1. Dekretacja

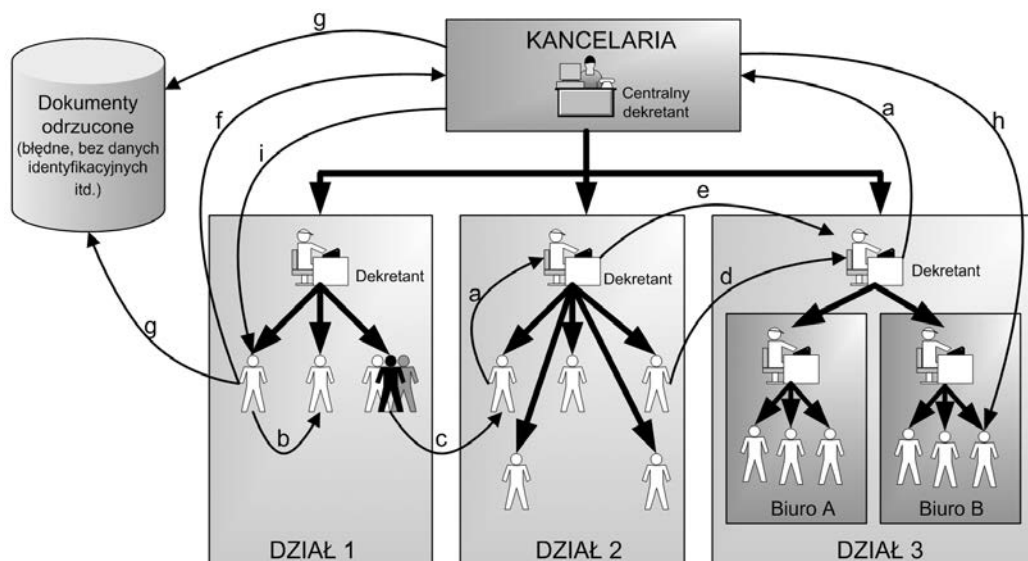
Najprostszym rodzajem obiegu jest dekretacja. Obieg dekretacyjny charakteryzuje się¹:

- brakiem stałej trajektorii procesu,
- uzależnieniem wyboru wykonawcy następnego kroku od jawnej decyzji wykonawcy kroku poprzedniego,
- brakiem pewności zakończenia procesu.

Obieg dekretacyjny opiera się przeważnie na systemie dekretantów. Wszystkie przesyłki trafiają do centralnego dekretanta, który rozdziela je na działy. W praktyce korespondencja trafia do dekretantów w działach, którzy z kolei dekretują pisma przekazywane poszczególnym osobom (lub grupom obsługowym). Jeśli istnieje taka potrzeba, opisany mechanizm może mieć więcej poziomów — z dekretantem na każdym z nich. Tego rodzaju przesłania zaznaczono na rysunku 5.1 pogrubionymi strzałkami.

Oprócz przesłań hierarchicznych może występować także cały szereg innych przesłań uzupełniających. Ich dostępność dla poszczególnych użytkowników systemu zależy od przyznanych uprawnień. Na rysunku 5.1 pokazano parę przykładów:

¹ Jeżeli za stan dokumentu przyjąć przypisanie go w danej chwili do użytkownika (końcowego lub dekretanta), system dekretacyjny można traktować jako automat skończony, mogący się znajdować w jednym z wielu stanów.



RYСУNEK 5.1. Różne rodzaje przesłań w dekretacji

- a** — odesłanie sprawy do dekreteranta wyższego rzędu (najczęściej przyznawane uprawnienie).
- b** — odesłanie sprawy do innego użytkownika w tym samym dziale. Uprawnienie często stosowane, ale niebezpieczne — istnieje ryzyko, że użytkownicy będą sobie nawzajem podsyłali trudniejsze zadania (zwłaszcza jeśli są premiovani za liczbę załatwionych spraw). Może dojść nawet do swoistej żonglerki sprawą pomiędzy dwoma lub trzema użytkownikami (tzw. efekt ping-ponga).
- c** — odesłanie sprawy do użytkownika w innym dziale (rzadko stosowane, raczej w niewielkich organizacjach, zagrożenia podobne jak w przypadku „b”).
- d** — przesłanie do dekreteranta innego działu (rzadko stosowane — klienci wybierają raczej opcję „e”).
- e** — przesłanie pomiędzy równorzędnymi dekreterantami (alternatywne do przesłania do dekreteranta nadrzędnego, który z kolei przekieruje sprawę do właściwego dekreteranta niższego poziomu).
- f** — zwrot do dekreteranta najwyższego poziomu (stosuje się najczęściej wówczas, jeśli korzysta się z opcji „h”).
- g** — uznanie dokumentu za wybrakowany.
- h** — bezpośrednia dekreteracja dokumentu przez głównego dekreteranta do użytkownika końcowego — stosuje się najczęściej wówczas, kiedy dla każdego rodzaju wpływających dokumentów przydzielono obsługujące je osoby.

Mimo wielu wad (np. braku pewności zakończenia sprawy — może ona być w nieskończoność przesyłana wewnątrz organizacji) dekreteacja posiada także pewne zalety. Przede wszystkim jest **łatwiejsza do wdrożenia**. Zainstalowanie systemu daje **natychmiastowe, odczuwalne przez użytkowników korzyści**. Ponadto dekreteacja pozwala na stopniowe osvajanie użytkowników z systemami obiegu dokumentów i nie wymaga natychmiastowej identyfikacji wszystkich przebiegów.

W istocie niekiedy dekreteacja bywa stosowana jako metoda istniejących w organizacji, a dotąd nie opisanych procesów (patrz rozdział 4., „Tworzenie opisu procesu”, punkt „Metoda trawnika — dekreteacja”). Analityk procesowy rejestruje rodzaj wpływających dokumentów i wszystkie przesłania, jakim te dokumenty podlegają, aby wysledzić istniejące w organizacji „nieformalne” procesy. Oczywiście tak zidentyfikowane procedury należy jeszcze potem poddać optymalizacji.

5.1.2. Sekwencje zdarzeń — już nie dekreteacja, jeszcze nie workflow

Sekwencje zdarzeń są trywialnym przypadkiem procesów biznesowych — w zasadzie (przynajmniej w swej najprostszej postaci) bez bramek decyzyjnych. Ich realizacja sprowadza się do kolejnego wykonywania operacji z listy.

5.1.2.1. Najprostsza realizacja sekwencji w praktyce

Najprostszą realizację zaprezentowano na rysunku 5.2.

Definiowanie sekwencji działań

Klient: Wytwórnia kruszyw "Cyklop"

Nazwa sekwencji: Oferta wiosenna dla Cyklopa

Właściciel sekwencji: Marcin Wojdecki

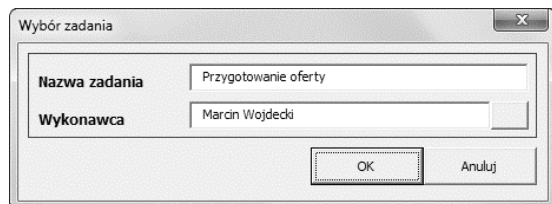
Zadanie	Wykonawca
Opis <i>opportunity</i>	Marcin Wojdecki
Przygotowanie oferty produktowej	Marcin Wojdecki
Przygotowanie oferty finansowej	Wacław Ziomecki
Nadanie ostatecznej postaci ofercie	Marcin Wojdecki
Akceptacja przełożonego	Tomasz Twardowski
Zatwierdzenie w underwritingu	Dział underwritingu
Wysłanie oferty przed wizytą	Asystenci Marcina Wojdeckiego
Wizyta	Marcin Wojdecki
Raport z kontaktu i ustalenie dalszych zadań	Marcin Wojdecki

Dodaj nowe zadanie Usuń zadanie Uruchom sekwencję Zapisz Anuluj

RYСУNEK 5.2. Najprostsza realizacja przepływu sekwencyjnego — ekran definiowania sekwencji

Aby zdefiniować sekwencję zadań, jej właściciel (w powyższym przykładzie *Marcin Wojdecki*) uruchamia formatkę przedstawioną na rysunku 5.2. Kolejno dodaje zadania, definiując

ich nazwy i wykonawców (w celu zdefiniowania zadania należy wcisnąć przycisk *Dodaj nowe zadanie* i wpisać jego nazwę w okienku, które się wyświetli — rysunek 5.3).

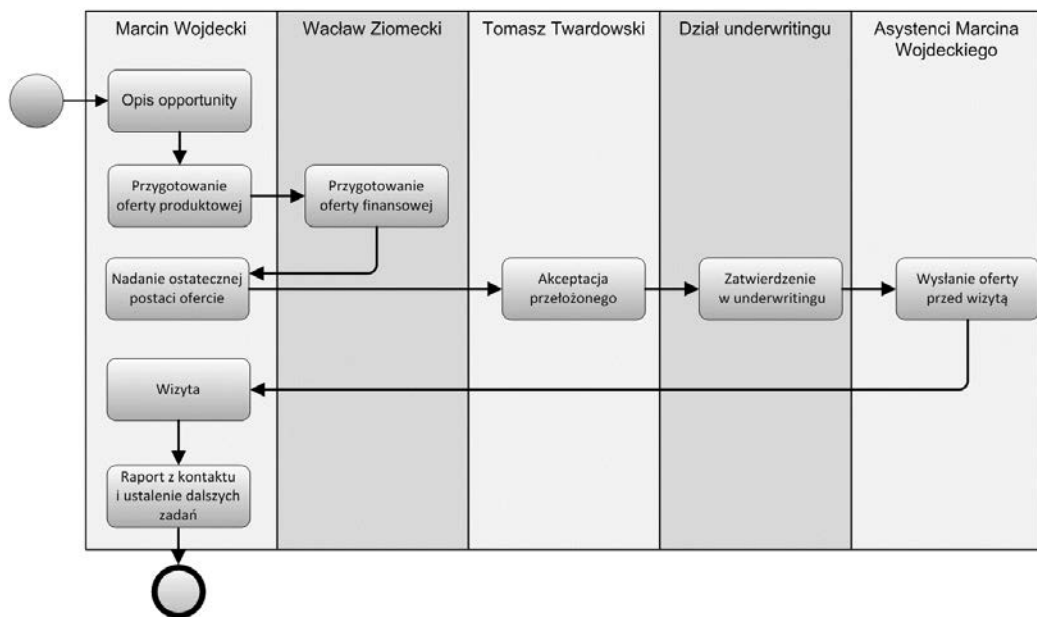


RYСУNEK 5.3. Definiowanie zadania

Po zdefiniowaniu całej sekwencji właściciel przechodzi do jej realizacji, wciskając przycisk *Uruchom sekwencję*.

W tym momencie w skrzynce zadań osoby wykonującej pierwsze zadanie pojawi się ono ze statusem „do wykonania”. Po jego realizacji kolejne zadanie sekwencji pojawi się u swego wykonawcy — i tak dalej, aż do wyczerpania listy zadań.

Tak zdefiniowana sekwencja odpowiada prostemu procesowi (bez rozgałęzień), jak na rysunku 5.4.



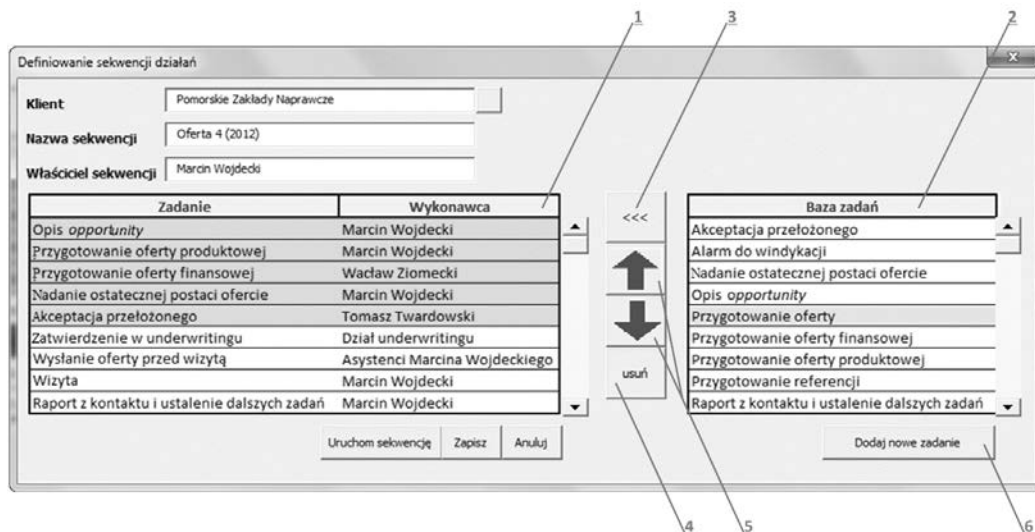
RYСУNEK 5.4. Proces odpowiadający sekwencji z rysunku 5.2

5.1.2.2. Modyfikacje zmierzające do ułatwienia pracy właścicielowi sekwencji

Na rysunku 5.5 przedstawiono modyfikacje, które mogą ułatwić definiowanie sekwencji.

1. Stworzenie bazy zadań.

Raz stworzone zadania mogą być gromadzone w bazie zadań. Dzięki temu w przyszłości będą mogły być dodawane do sekwencji poprzez wybór z listy (2 na rysunku 5.5) i wciśnięcie przycisku kopiowania (3). Podczas przenoszenia niezbędne będzie jedynie zdefiniowanie wykonawcy zadania.



RYSUNEK 5.5. Modyfikacje ułatwiające obsługę

2. Umożliwienie zmiany kolejności zadań.

Przyciski ze strzałkami (5) umożliwiają zmianę kolejności zadań w sekwencji — wystarczy zaznaczyć zadanie i wcisnąć jeden z nich.

3. Sygnalizowanie na liście zadań, które z nich zostały już wykonane.

Dzięki kolorowaniu zadań już wykonanych (na liście 1) właściciel sekwencji (osoba definiująca) może sprawdzać jej stan realizacji.

Rysunek 5.5 pokazuje także garść modyfikacji ułatwiających pracę z sekwencjami:

- Tablica z bazą zadań (2) zawiera gotowe zadania przeznaczone do umieszczenia w sekwencji. Przycisk <<< (3) przenosi zaznaczone w bazie (2) pozycje do tablicy sekwencji (1).
- Przycisk *usuń* (4) usuwa pozycję z tablicy sekwencji (1).
- Przyciski ze strzałkami (5) umożliwiają zmianę kolejności zaznaczonych pozycji w tablicy sekwencji (1).

5.1.2.3. Modyfikacje rozszerzające funkcjonalność mechanizmu sekwencji

Rysunek 5.6 ukazuje formatkę definiowania sekwencji, która jest wyposażona w dodatkowe mechanizmy rozszerzające możliwości funkcjonalne.

Rozdział 7.

Typowe zagadnienia

7.1. KANAŁY KOMUNIKACJI SYSTEMU KANCELARYJNEGO Z OTOCZENIEM

Istotnym elementem systemu jest sposób, w jaki komunikuje się on z otoczeniem. Prawdę mówiąc, jest to kluczowe zagadnienie — żaden system nie działa bowiem na swoje własne potrzeby, lecz wypełnia zadania dla otoczenia.

Rysunek 7.1 przedstawia typowe otoczenie systemu i standardowe kanały komunikacji na osi otoczenie – system.

Oczywiście wachlarz kanałów w realnym systemie jest zwykle pewnym podzbiorem tych zaproponowanych powyżej — trudno też wykluczyć istnienie innych kanałów (np. IVR)¹.

Oprócz innych atrybutów związanych z działaniem systemu lub potrzebami biznesowymi obiekty wpływające poprzez poszczególne kanały komunikacji posiadają charakterystyczne dla tych kanałów atrybuty, co ilustruje tabela 7.1.

Dobłą praktyką jest unifikacja wszystkich kanałów wejściowych do możliwie podobnej postaci (np. pliku graficznego i pliku z atrybutami).

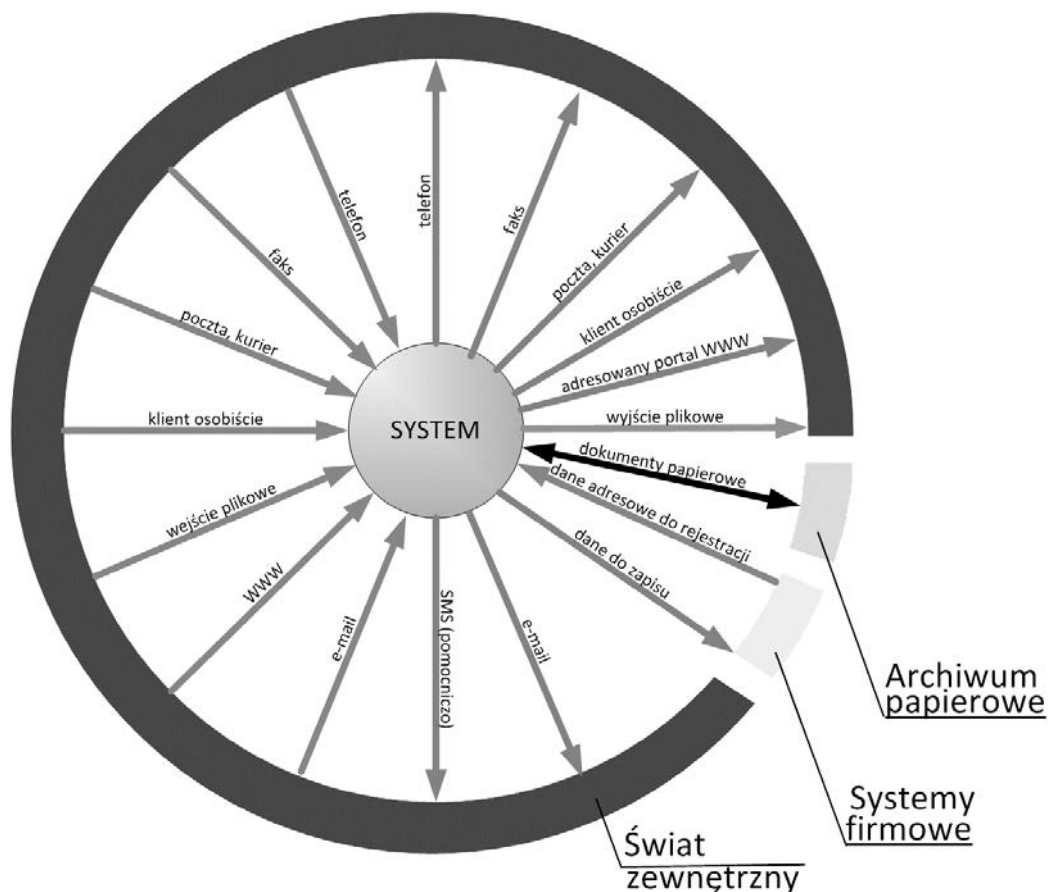
Ciekawym rozwiązaniem jest również wyjście zunifikowane: zawartość pliku wyjściowego zostaje zapisana w bazie zleceń ze znacznikiem kanału, jakim ma być wysłana; wbudowany trigger zajmuje się wysyłką poprzez zadeklarowane kanały (rysunek 7.2).

7.2. WPŁYW DOKUMENTU PAPIEROWEGO

7.2.1. Przypadek trywialny

Najprostsze rozwiązanie systemu przyjmowania korespondencji pokazano na rysunku 7.3 (schemat BPMN — rysunek 7.4). Oto przebieg przetwarzania:

¹ IVR (ang. *Interactive Voice Response*) — systemy automatycznej obsługi telefonicznej, sterowanej przez naciskanie odpowiednich klawiszy telefonu.

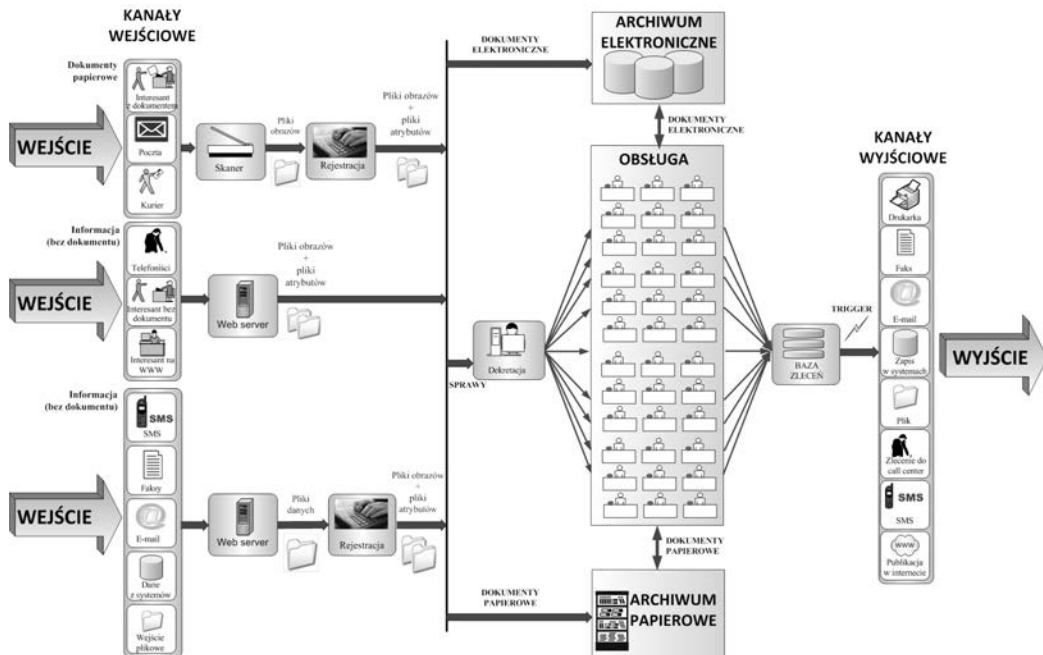


RYSUNEK 7.1. Typowe kanały komunikacji z otoczeniem

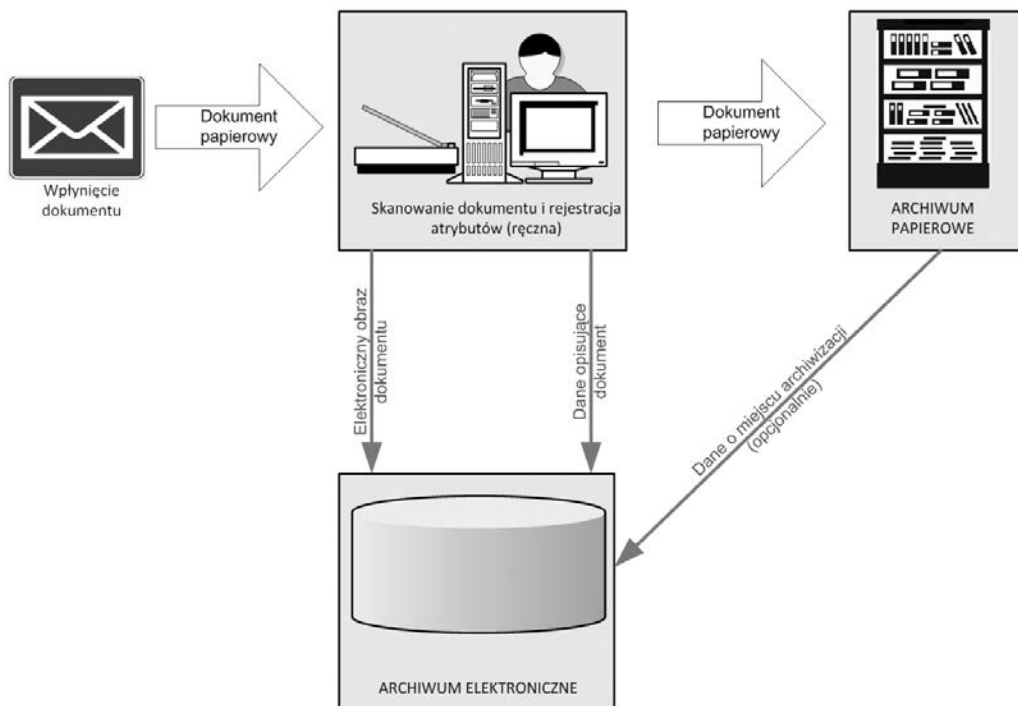
1. Dokument wpływa do stanowiska skanowania/opisu dokumentu.
2. Pracownik skanuje napływające dokumenty, a następnie opisuje je atrybutami w systemie. Obrazy dokumentów zostają zapisane wraz z atrybutami w archiwum elektronicznym.
3. Ostatnim etapem jest przekazanie dokumentu do archiwum. Tam dokument zostaje zarchiwizowany.
4. Opcjonalnie archiwum może przekazać do bazy dane o miejscu przechowywania dokumentu — w tym celu musi jednak odnaleźć dokument w systemie i wprowadzić miejsce przechowywania jako jeden z jego atrybutów opisu.

Tabela 7.1. Przykład określenia atrybutów aktywnych dla poszczególnych kanałów wejścia/wyjścia

Kanały → Kierunek → Atrybuty ↓	Papierowy (poczta)		Papierowy (dokumenty składane osobiście)		Faks		E-mail		Formularz WWW	SMS		IVR	Plikowy zwykły		Pakietów wymiany		Interfejsowe (wymiana z innymi systemami)	
	wejście	wyjście	wejście	wyjście	wejście	wyjście	wejście	wyjście	wejście	wejście	wyjście	wejście	wejście	wyjście	wejście	wyjście	wejście	wyjście
Data utworzenia	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Data/czas wpływu	X		X		X		X		X	X		X						
Data/czas wpływu do kancelarii	X		X		X		X		X	X		X						
Data nadania	X	X	X	X	X	X	X	X	X	X	X	X						
Data przekazania na pocztę/do kuriera		X																
Data wydania				X														
Typ przesyłki	X	X																
Waga przesyłki		X																
Cena netto	X	X																
Wartość polecenia	X	X																
Wartość potwierdzenia odbioru	X	X																
Wartość brutto	X	X																
Nadawca	X		X		X		X		X	X		X	X		X		X	
Typ doręczenia	X	X																
Numer identyfikacyjny (tzw. numer „erki”) i przesyłek kurierskich (numer listu przewozowego)	X	X																
Data i czas rejestracji	X		X		X													
Kanał	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Sygnatura	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X		X	X
Odbiorca		X		X		X		X			X			X		X		X



RYSUNEK 7.2. Kanały wejściowe i wyjściowe



RYSUNEK 7.3. Trywialny system przyjmowania korespondencji (rysunek poglądowy)

Rozdział 12.

Wspomaganie projektowania procesów przy zastosowaniu Enterprise Architect

Enterprise Architect to jedno z najbardziej popularnych narzędzi CASE (ang. *Computer-Aided Software Engineering*), będące wygodnym środowiskiem pracy analityka. Z wersji na wersję obsługuje coraz więcej standardów i notacji. W chwili obecnej wspiera UML, BPMN, Archimate, SysML, ArcGIS, TOGAF, Zachman, UPDM; zawiera też diagramy i narzędzia wspierające zarządzanie projektem, wymaganiami i testami oraz generowanie dokumentacji. Oprogramowanie pozwala ponadto na automatyczne generowanie kodu w kilkunastu odmiannach¹, a także na symulacje biznesowe. Uzupełnieniem są diagramy pomocnicze (np. wspierające *mind mapping*)².

Wersję demonstracyjną Enterprise Architect można pobrać ze strony producenta <http://sparxsystems.com>.

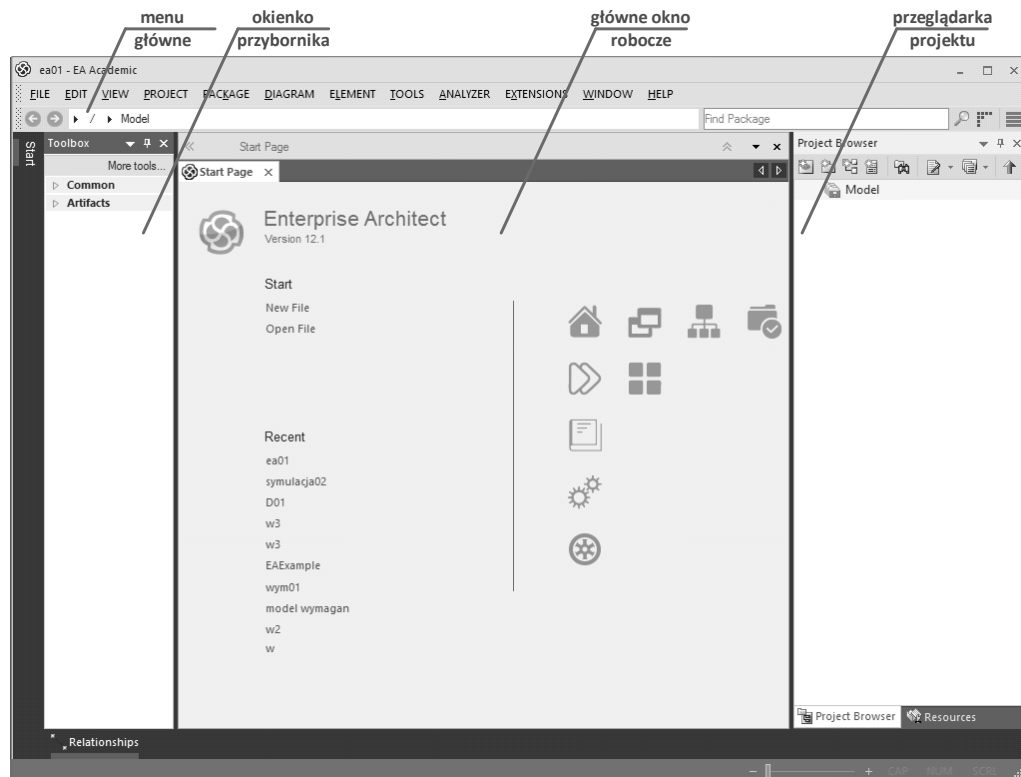
Dziękuję firmie Sparx Systems za udostępnienie licencji Enterprise Architect na potrzeby pisania niniejszego rozdziału oraz za zezwolenie na publikację zrzutów ekranowych.

12.1. INTERFEJS ENTERPRISE ARCHITECT

W Enterprise Architect wszystkie elementy projektu są umieszczane w hierarchicznym drzewie wyświetlanym w oknie przeglądarki projektu po prawej stronie ekranu aplikacji (rysunek 12.1). Standardowo po lewej stronie aplikacja wyświetla przybornik z dostępnymi elementami. Miejsce pośrodku zajmuje główne okno robocze.

¹ Obecnie dostępne są następujące szablony: ActionScript, C, C#, C++, Delphi, Java, PHP, Python, Visual Basic (w wersjach 6 i .NET).

² Należy się spodziewać dalszego rozwoju Enterprise Architect w kierunku standardów definiowanych przez BABOK®, gdyż firma Sparx Systems podpisała w 2015 roku strategiczne partnerstwo z International Institute of Business Analysis (IIBA).

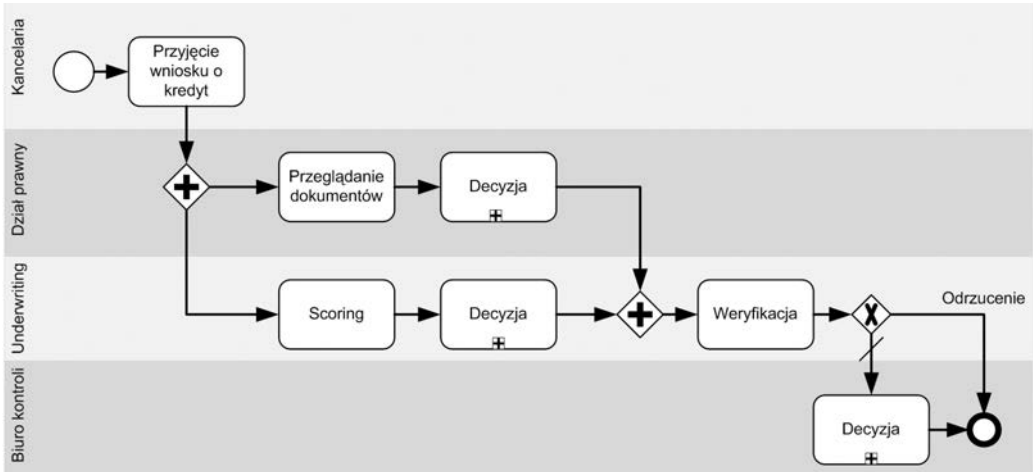


RYSUNEK 12.1. Układ okien Enterprise Architect

12.2. PRZYKŁAD I — DIAGRAMY PROCESÓW W ENTERPRISE ARCHITECT

Dla przykładu (rysunek 12.2, będący kopią rysunku 2.82) zostanie zamodelowany proces rozpatrywania kredytów, opisany w rozdziale 2.7.1.2.

Znając z doświadczenia problemy osób rozpoczynających swoją przygodę z Enterprise Architect z orientacją w interfejsie programu, przy omawianiu jakiejś czynności (np. tworzenia widoku) po raz pierwszy dokładnie ją opiszę i dołączę odpowiednie zrzuty ekranu. Przy ponownym wykonywaniu podobnej operacji zrezygnuję już jednak z detalicznego opisu i ilustrowania składających się na nią czynności, podając tylko istotne parametry.



RYSUNEK 12.2. Modelowany diagram (kopia rysunku 2.82)

Diagramy w Enterprise Architect są zorganizowane w sposób hierarchiczny (drzewiasty). Najwyższym elementem w hierarchii jest **model**. Cały model dzieli się na widoki (*Views*), w których umieszczone są pakiety (*Packages*) — to w nich dopiero znajduje się użyteczna część modelu. Często pakiety mają swoje pakiety podrzędne, w których dopiero mieszczą się diagramy (*Diagrams*).

12.2.1. Tworzenie modelu

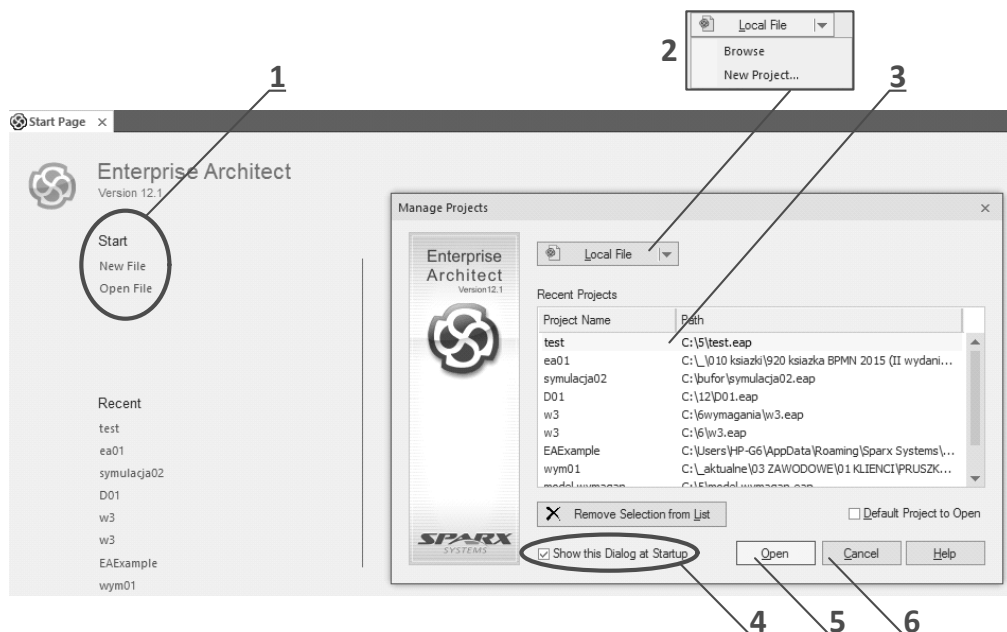
Pierwszą czynnością jest stworzenie modelu. Po uruchomieniu Enterprise Architect system wita nas, wyświetlając ekran startowy.

Projekt można utworzyć na dwa sposoby: albo klikając link *New File* po lewej stronie ekranu (1 na rysunku 12.3), albo wybierając z pola rozwijanego *Local File* (2) pozycję *New Project*³.

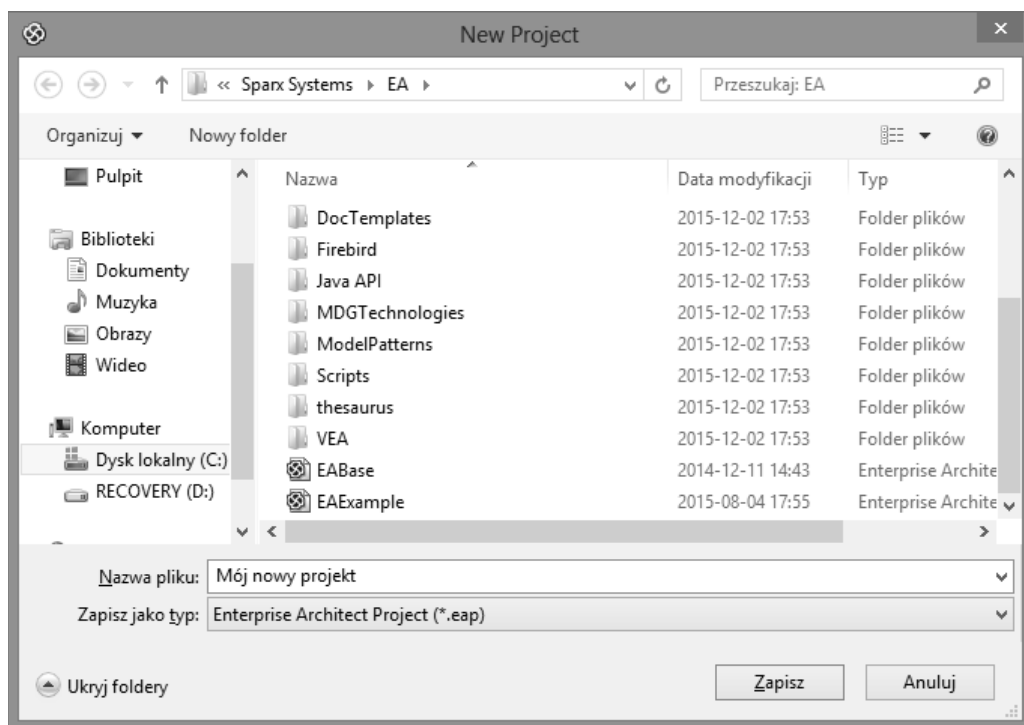
Można też wcisnąć przycisk *Cancel* (6) i wybrać z menu polecenie *File/New Project*.

Każda z tych akcji spowoduje wyświetlenie okienka, w którym użytkownik będzie mógł wybrać katalog i wprowadzić nazwę pliku dla nowego projektu (rysunek 12.4).

³ Polecenia *Open File* (1) i *Browse* (2) pozwalają na wybór z dysku pliku z zapisanym wcześniej projektem. Lista (3) pozwala na szybki wybór spośród niedawno zapisanych plików. Jeśli nie chcemy, by ekran startowy wyświetlał się przy kolejnych uruchomieniach programu, należy usunąć zaznaczenie opcji *Show this Dialog at Startup* (4).

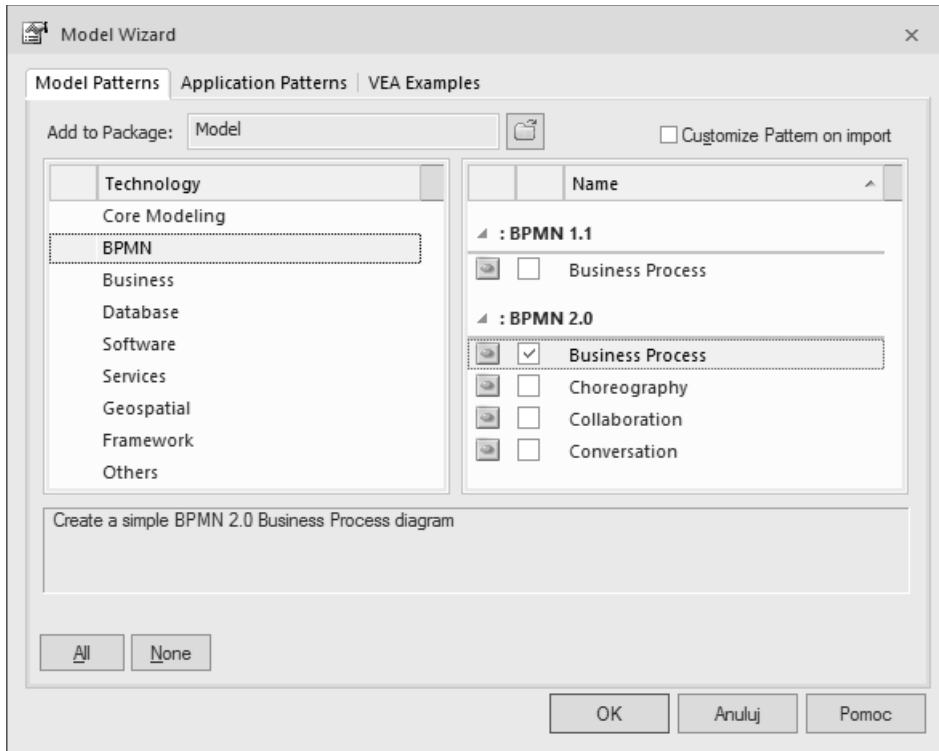


RYSUNEK 12.3. Ekran startowy Enterprise Architect



RYSUNEK 12.4. Zapisywanie pliku nowego projektu

Po zatwierdzeniu przyciskiem *Zapisz* zostanie wyświetlone okno *Model Wizard* (rysunek 12.5).



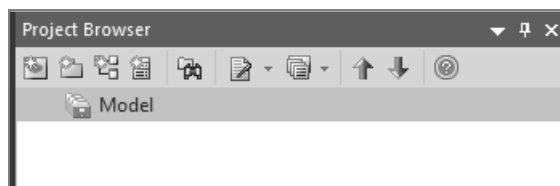
RYSUNEK 12.5. Okno Model Wizard

Można skorzystać z kreatora⁴, jednak ja rekomenduję ręczne stworzenie struktury modelu i przystosowanie go do konkretnego projektu (a także do przyzwyczajenia użytkownika). Ponadto przejście przez całą procedurę w naszym przykładzie pozwoli na zademonstrowanie sposobu tworzenia wszystkich elementów projektu (widoków, pakietów i diagramów).

Aby zrezygnować z kreatora, opuszczamy okienko, wciskając przycisk *Anuluj*. W okienku przeglądarki projektu po prawej stronie widnieje tylko jeden element — *Model* (rysunek 12.6).

Nadszedł czas na utworzenie widoku.

⁴ Gdybyśmy chcieli skorzystać z kreatora, należałoby w naszym przypadku (tworzenia modelu procesów biznesowych w notacji BPMN) wybrać — jak na rysunku 12.5 — z listy po lewej stronie opcję *BPMN*, zaś po prawej pozycję *Business Process*.

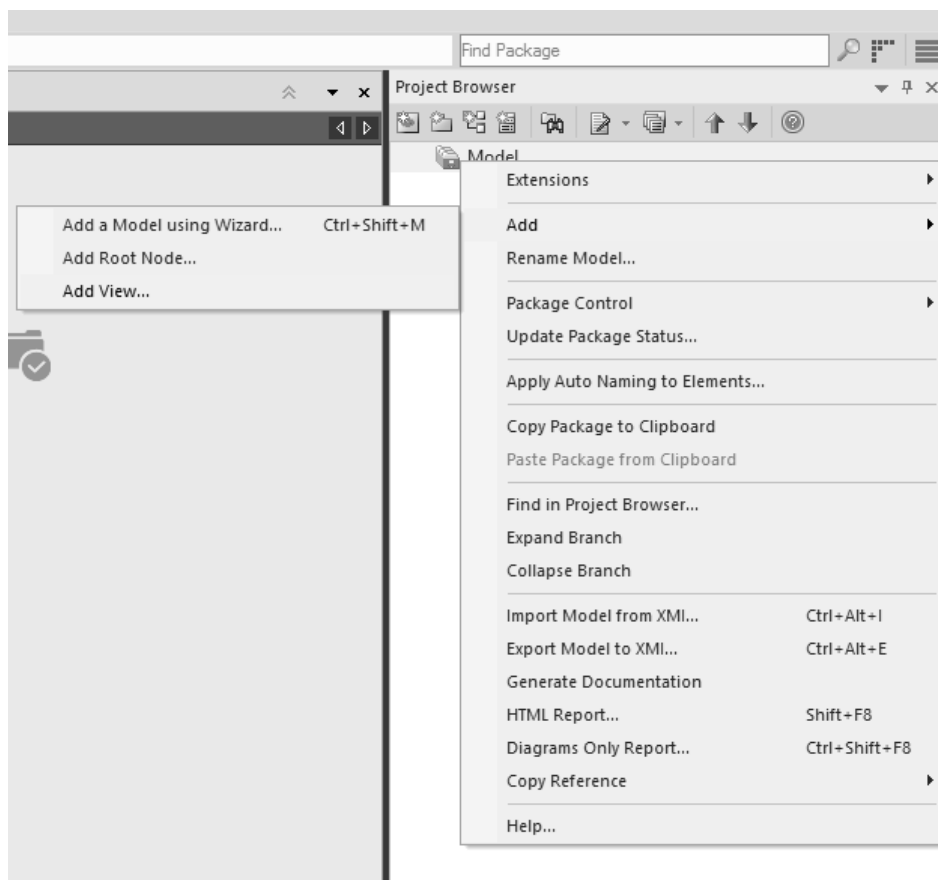


RYSUNEK 12.6. Model utworzony w przeglądarce Enterprise Architect

12.2.2. Tworzenie widoku

Aby utworzyć widok, należy postępować zgodnie z poniższymi krokami.

1. Klikamy prawym przyciskiem myszy ikonę *Model*. Z menu podręcznego wybieramy kolejno opcje *Add/Add View* (rysunek 12.7).



RYSUNEK 12.7. Tworzenie widoku — krok 1.

PROGRAM PARTNERSKI

GRUPY WYDAWNICZEJ HELION



1. ZAREJESTRUJ SIĘ
2. PREZENTUJ KSIĄŻKI
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**

Jedna z najlepszych pozycji z zakresu zarządzania procesami na rynku!

(z recenzji pierwszego wydania książki)

Proces biznesowy to pojęcie z obszaru zarządzania przedsiębiorstwem. To seria powiązanych ze sobą działań czy zadań, która ma prowadzić do osiągnięcia oczekiwanego efektu bądź rozwiązania postawionego problemu. Tajniki zarządzania firmą lub organizacją poprzez procesy nie są łatwe i nie da się ich w mig opanować, jednak ich wdrożenie okazuje się zwykle wysoce opłacalne, a często wręcz konieczne tam, gdzie przykładą się dużą wagę do sprawnego kierowania. Czyli praktycznie w każdym przedsiębiorstwie, którego menedżerowie myślą o podnoszeniu wydajności i efektywności pracy.

Projektowanie procesów wymaga przygotowania merytorycznego oraz zdolności do powiązania praktyki biznesowej ze znajomością odpowiednich narzędzi. Dobrze, jeśli praktykę z tego obszaru zdobywa się na konkretnym rynku, na którym funkcjonuje przedsiębiorstwo. To ważne, bo nasza rodzima rzeczywistość jest całkiem inna niż choćby amerykańska, o czym dobrze wie Marek Piotrowski, od kilkunastu lat specjalizujący się w projektowaniu procesów biznesowych z wykorzystaniem narzędzi informatycznych. Za pomocą niniejszego podręcznika autor pragnie przeprowadzić czytelnika przez projektowanie, testowanie i optymalizację procesów ze szczególnym uwzględnieniem problemów, z jakimi można się spotkać na polskim gruncie.

Wydanie drugie wzbogacono o dodatkowy rozdział, omawiający modelowanie procesów przy użyciu Enterprise Architect, czyli najpopularniejszego narzędzia przeznaczonego do tego celu.



MAREK PIOTROWSKI — absolwent Politechniki Gdańskiej, od dwudziestu siedmiu lat zajmujący się informatyką, a od dziewiętnastu — procesami biznesowymi w ramach informatyki. W 2007 roku ukazała się jego pierwsza książka: *BPMN — notacja modelowania procesów biznesowych*.

książkiklasybusiness

Nr katalogowy: **44494**



Księgarnia internetowa:

<http://onepress.pl>



Zamówienia telefoniczne:

0 801 339900



0 601 339900

one
p r e s s

Sprawdź najnowsze promocje:

🔗 <http://onepress.pl/promocje>

Książki najchętniej czytane:

🔗 <http://onepress.pl/bestsellery>

Zamów informacje o nowościach:

🔗 <http://onepress.pl/novosci>

Helion SA

ul. Kościuszki 1c, 44-100 Gliwice

tel.: 32 230 98 63

e-mail: onepress@onepress.pl

<http://onepress.pl>

Cena 69,00 zł

ISBN 978-83-283-2521-0



9 788328 325210