

**zip**

ZARZĄDZANIE  
I INŻYNIERIA  
PRODUKCJI

Waldemar Bojar  
Katarzyna Rostek  
Leszek Knopik

# Systemy wspomagania decyzji



Polskie Wydawnictwo Ekonomiczne

Biblioteka Politechniki Krakowskiej



100000336374



# Systemy wspomagania decyzji

# Cykl podręczników akademickich dla kierunku „Zarządzanie i inżynieria produkcji”

## Redaktor naukowy cyklu

Ryszard Knosala

### Podręczniki dla I stopnia studiów

#### Zarządzanie dla inżynierów

Ewa Masłyk-Musiał, Anna Rakowska, Elżbieta Krajewska-Bińczyk

#### Finanse i rachunkowość

Alina Dyduch, Maria Sierpińska, Zofia Wilimowska

#### Rachunek kosztów dla inżynierów

Józef Matuszek, Mariusz Kołosowski, Zofia Krokosz-Krynke

#### Zarządzanie produkcją i usługami

Edward Pająk, Marek Klimkiewicz, Anna Kosieradzka

#### Zarządzanie jakością i bezpieczeństwem

Zofia Zymonik, Adam Hamrol, Piotr Grudowski

#### Logistyka w przedsiębiorstwie

Iwona Pisz, Tadeusz Sęk, Władysław Zielecki

#### Procesy produkcyjne

Józef Gawlik, Antoni Świć, Jarosław Plichta

#### Automatyzacja i robotyzacja procesów produkcyjnych

Gabriel Kost, Piotr Łebkowski, Łukasz Węsierski

#### Informatyka i komputerowe wspomaganie prac inżynierskich

Cezary Orłowski, Jerzy Lipski, Andrzej Loska

#### Projektowanie inżynierskie i grafika inżynierska

Piotr Gendarz, Szymon Salamon, Piotr Chwastyk

#### Marketing dla inżynierów

Agnieszka Izabela Baruk, Katarzyna Hys, Adam Dzikowski

#### Ekologia

Andrzej Misiólek, Edward Kowal, Aneta Kucińska-Landwójtowicz

#### Zarządzanie środowiskowe

Edward Kowal, Aneta Kucińska-Landwójtowicz, Andrzej Misiólek

#### Nauka o materiałach

Danuta Kotnarowska, Andrzej Wasiak, Barbara Kucharska, Tomasz Lipiński

#### Metrologia

Władysław Jakubiec, Sławomir Zator, Paweł Majda

### Podręczniki dla II stopnia studiów

#### Zarządzanie strategiczne dla inżynierów

Grażyna Gierszewska, Barbara Olszewska, Jan Skonieczny

#### Organizacja systemów produkcyjnych

Jerzy Lewandowski, Bożena Skołod, Dariusz Plinta

#### Zintegrowane systemy zarządzania

Zbigniew Banaszak, Sławomir Kłos, Janusz Mleczek

#### Prognozowanie i symulacja w przedsiębiorstwie

Artur Maciąg, Roman Pietroń, Sławomir Kukla

#### Zarządzanie projektem

Marek Wirkus, Henryk Roszkowski, Ewa Dostatni, Wacław Gierulski

#### Zarządzanie innowacjami

Ryszard Knosala, Anna Boratyńska-Sala, Magdalena Jurczyk-Bunkowska, Aleksander Moczala

#### Systemy wspomagania decyzji

Waldemar Bojar, Katarzyna Rostek, Leszek Knopik

#### Zarządzanie wiedzą

Jędrzej Trajer, Alfred Paszek, Stanisław Iwan



ZARZĄDZANIE  
I INŻYNIERIA  
PRODUKCJI

Waldemar Bojar  
Katarzyna Rostek  
Leszek Knopik

# Systemy wspomagania decyzji



---

Polskie Wydawnictwo Ekonomiczne

Recenzent

**Prof. dr hab. Celina Olszak**

Projekt okładki

**Paweł Rosolek**

Redaktor

**Beata Skwierczyńska-Mizerska**

*Cyf. 67.*

11-362132

© Copyright by Polskie Wydawnictwo Ekonomiczne S.A.

Warszawa 2014

ISBN 978-83-208-2076-8

Polskie Wydawnictwo Ekonomiczne S.A.  
ul. Canaletta 4, 00-099 Warszawa  
tel. (centrala): 22 827 80 01, faks: 22 827 55 67  
e-mail: [pwe@pwe.com.pl](mailto:pwe@pwe.com.pl)  
[www.pwe.com.pl](http://www.pwe.com.pl)

Skład i łamanie: Master, Łódź

# Spis treści

<b>Wstęp</b> . . . . .	11
<b>Rozdział 1. Istota systemów wspomaganie decyzji</b> . . . . .	13
1.1. Geneza i znaczenie systemów wspomaganie decyzji (SWD) . . . . .	13
1.1.1. Definicja SWD i historia powstania . . . . .	13
1.1.2. Pierwsze przykłady — funkcje i struktura SWD . . . . .	15
1.1.3. Główne koncepcje, obszary zastosowań i kierunki rozwoju SWD . . . . .	19
1.2. Fazy procesu decyzyjnego . . . . .	21
1.2.1. Problem decyzyjny . . . . .	21
1.2.2. Decyzja . . . . .	22
1.2.3. Proces decyzyjny . . . . .	23
1.3. Poziomy procesów decyzyjnych . . . . .	24
1.3.1. Poziomy decyzji w procesie zarządzania . . . . .	25
1.3.2. Poziomy decyzji ze względu na stopień ich strukturalizowania . . . . .	25
1.3.3. Poziomy decyzji ze względu na stopień ryzyka decyzyjnego . . . . .	26
1.3.4. Wspomaganie decyzji ze względu na poziomy procesu decyzyjnego . . . . .	27
<b>Rozdział 2. Symulacje i techniki komputerowe w SWD</b> . . . . .	28
2.1. Przewidywanie wyników za pomocą eksperymentów symulacyjnych . . . . .	28
2.1.1. Klasyfikacja modeli symulacyjnych . . . . .	29
2.1.2. Istota procesu decyzyjnego – modele i symulacje . . . . .	34
2.1.3. Modele optymalizacyjne w procesie decyzyjnym . . . . .	36
2.1.4. Rodzaje i przykłady eksperymentów symulacyjnych . . . . .	42
2.2. Zastosowania metod optymalizacyjnych . . . . .	45
2.2.1. Logiczne metody optymalizacyjne . . . . .	46
2.2.2. Statystyczne metody optymalizacyjne . . . . .	51

2.2.3. Metody heurystyczne . . . . .	55
2.3. Techniki kalkulacyjne w eksperymentach symulacyjnych . . . . .	58
2.3.1. Obliczenia statystyczne — przykłady . . . . .	58
2.3.2. Techniki iteracyjne — przykłady . . . . .	59
2.3.3. Reguły decyzyjne i algorytmy wnioskowania logicznego — przykłady . . . . .	60
2.4. Przygotowanie BD na potrzeby SWD . . . . .	62
2.4.1. Struktura i istota działania BD . . . . .	63
2.4.2. Modelowanie logiczne i fizyczne BD . . . . .	64
2.4.3. Przetwarzanie transakcyjne w BD . . . . .	68
2.4.4. Funkcjonalne i eksploatacyjne aspekty przygotowania BD dla celów SWD . . . . .	69
2.4.5. Przykład przetwarzania faktów i reguł decyzyjnych w BD . . . . .	70
2.5. Pytania i zadania kontrolne . . . . .	72
2.5.1. Pytania kontrolne . . . . .	72
2.5.2. Przykładowe zadania . . . . .	72
<b>Rozdział 3. Interfejs użytkownika SWD . . . . .</b>	<b>74</b>
3.1. Wymagania funkcjonalne . . . . .	75
3.1.1. Modelowanie wymagań funkcjonalnych . . . . .	76
3.1.2. Język specyfikacji wymagań funkcjonalnych . . . . .	77
3.1.3. Przykład specyfikacji wymagań funkcjonalnych dla potrzeb logistycznych . . . . .	79
3.2. Wymagania eksploatacyjne . . . . .	80
3.2.1. Założenia techniczno-organizacyjne . . . . .	81
3.2.2. Założenia ekonomiczne . . . . .	82
3.2.3. Przykład specyfikacji wymagań eksploatacyjnych . . . . .	82
3.3. Projektowanie interfejsu użytkownika . . . . .	83
3.3.1. Ergonomiczne cechy interfejsu . . . . .	85
3.3.2. Bezpieczeństwo interfejsu . . . . .	85
3.3.3. Przykłady interfejsów . . . . .	86
3.4. Pytania i zadania kontrolne . . . . .	88
3.4.1. Pytania kontrolne . . . . .	88
3.4.2. Przykładowe zadania . . . . .	88
<b>Rozdział 4. Modele abstrakcyjne w projektowaniu SWD . . . . .</b>	<b>90</b>
4.1. Płaskie i hierarchiczne problemy decyzyjne . . . . .	90
4.1.1. Graficzny model płaskiego problemu decyzyjnego . . . . .	90
4.1.2. Wartościowanie decyzji oraz procedura dekompozycji drzewa decyzyjnego . . . . .	96
4.1.3. Procedura dekompozycji drzewa . . . . .	97
4.1.4. Grafowy model hierarchicznego problemu decyzyjnego . . . . .	97



4.2. Zbiory rozmyte w modelowaniu problemów decyzyjnych . . . . .	99
4.2.1. Koncepcja zbiorów rozmytych L. Zadeha . . . . .	99
4.2.2. Operacje na zbiorach rozmytych . . . . .	103
4.2.3. Relacje rozmyte i ich własności . . . . .	106
4.2.4. Operacje wnioskowania na zbiorach rozmytych . . . . .	108
4.2.5. Przykłady procesów decyzyjnych na zbiorach rozmytych . . . . .	110
4.3. Zbiory przybliżone Z. Pawlaka w modelowaniu . . . . .	113
4.3.1. Koncepcja zbiorów przybliżonych Z. Pawlaka . . . . .	113
4.3.2. Operacje wnioskowania na zbiorach przybliżonych . . . . .	118
4.3.3. Przykłady procesów decyzyjnych na zbiorach przybliżonych . . . . .	124
4.4. Zasada charakteryzacji w modelowaniu problemów decyzyjnych . . . . .	128
4.4.1. Koncepcja zasady charakteryzacji . . . . .	128
4.4.2. Problem decyzyjny w ujęciu zasady charakteryzacji . . . . .	131
4.4.3. Przykład algorytmu konstruowania zbioru decyzji alternatywnych . . . . .	134
4.5. Pytania i zadania kontrolne . . . . .	141
4.5.1. Pytania kontrolne . . . . .	141
4.5.2. Przykładowe zadania . . . . .	142
<b>Rozdział 5. Konkretyzacja rozwiązań projektowych SWD . . . . .</b>	<b>144</b>
5.1. Strategie projektowania SWD . . . . .	144
5.1.1. Przyrostowa i kaskadowa strategia projektowania . . . . .	148
5.1.2. Spiralna strategia projektowania . . . . .	149
5.1.3. Architektoniczne uwarunkowania rozwoju SWD . . . . .	149
5.2. Elementy składowe architektury SWD . . . . .	150
5.2.1. Bazy i hurtownie danych . . . . .	151
5.2.2. Baza wiedzy i moduł wnioskujący . . . . .	155
5.2.3. Magistrała danych i oprogramowanie integrujące . . . . .	165
5.2.4. Konsole i interfejsy użytkownika . . . . .	165
5.3. Hurtownia danych (HD) w SWD . . . . .	168
5.3.1. Struktura i istota działania HD . . . . .	168
5.3.2. Modelowanie logiczne i fizyczne HD . . . . .	170
5.3.3. Przetwarzanie analityczne w HD . . . . .	172
5.3.4. Funkcjonalne i eksploatacyjne aspekty projektowania HD na potrzeby SWD . . . . .	174
5.3.5. Przykład przetwarzania analitycznego w HD . . . . .	178
5.4. Baza wiedzy (BW) i moduł wnioskujący w SWD . . . . .	182
5.4.1. Definicja i istota działania BW . . . . .	182
5.4.2. Charakterystyka i projektowanie regułowych baz wiedzy . . . . .	186
5.4.3. Przykład wykorzystania BW i modułu wnioskującego w SWD . . . . .	189
5.5. Pytania i zadania kontrolne . . . . .	190

5.5.1. Pytania kontrolne . . . . .	190
5.5.2. Przykładowe zadania . . . . .	190
<b>Rozdział 6. Metody i narzędzia projektowania SWD . . . . .</b>	<b>192</b>
6.1. Metody projektowania SWD . . . . .	192
6.1.1. Specyfikacja funkcjonalności SWD . . . . .	192
6.1.2. Modelowanie architektury SWD . . . . .	193
6.1.3. Weryfikacja niesprzeczności modeli elementów architektury SWD . . . . .	197
6.2. Narzędzia projektowania SWD . . . . .	198
6.2.1. Narzędzia inżynierii wiedzy . . . . .	198
6.2.2. Prototypowanie SWD . . . . .	202
6.2.3. Weryfikacja prototypów SWD . . . . .	203
6.3. Przykłady prostych narzędzi projektowania SWD . . . . .	208
6.3.1. Arkusz kalkulacyjny MS Excel . . . . .	208
6.3.2. Tablice decyzyjne . . . . .	210
6.3.3. Scenariusze i animacja zdarzeń . . . . .	216
6.4. Zastosowanie systemów zarządzania BD w projektowaniu SWD . . . . .	218
6.4.1. Elementy języka SQL . . . . .	219
6.4.2. Przykłady zapytań SQL . . . . .	224
6.4.3. Przykład definiowania BD w SWD . . . . .	227
6.5. Języki programowania wysokiego poziomu w projektowaniu SWD . . . . .	229
6.5.1. Języki i pakiety inżynierii wiedzy . . . . .	229
6.5.2. Przykład formułowania decyzji w języku programowania wysokiego poziomu . . . . .	232
6.5.3. Przykład zastosowania pakietów inżynierii wiedzy w realizacji SWD . . . . .	235
6.6. Pytania i zadania kontrolne . . . . .	235
6.6.1. Pytania kontrolne . . . . .	235
6.6.2. Przykładowe zadania . . . . .	236
<b>Rozdział 7. Weryfikacja i wdrożenie SWD . . . . .</b>	<b>238</b>
7.1. Metodyka wdrożenia systemu informatycznego klasy SWD . . . . .	238
7.1.1. Opracowanie docelowego modelu systemu informacyjnego przedsiębiorstwa . . . . .	239
7.1.2. Opracowanie strategii i planu wdrożenia SWD . . . . .	241
7.1.3. Przygotowanie finansowe i logistyczne przedsięwzięcia . . . . .	242
7.1.4. Zarządzanie projektem wdrożenia SWD . . . . .	244
7.2. Metodyka wielokryterialnego wyboru systemu informatycznego klasy SWD . . . . .	246
7.2.1. Istota modelu wielokryterialnego . . . . .	247
7.2.2. Budowa modelu przedsięwzięcia . . . . .	250
7.2.3. Przestrzeń decyzyjna . . . . .	252
7.2.4. Wielokryterialna procedura wyboru . . . . .	256

7.3. Pytania i zadania kontrolne . . . . .	257
7.3.1. Pytania kontrolne . . . . .	257
7.3.2. Przykładowe zadania . . . . .	258
<b>Rozdział 8. Wpływ SWD na gospodarczą działalność organizacji . . . . .</b>	<b>259</b>
8.1. Wpływ SWD na funkcjonowanie organizacji . . . . .	259
8.1.1. Współczesne kierunki rozwoju i obszary zastosowań SWD . . . . .	260
8.1.2. Wielokryterialne podejmowanie decyzji . . . . .	263
8.1.3. Psychologiczne aspekty podejmowania decyzji . . . . .	266
8.1.4. Prawne aspekty zastosowań SWD . . . . .	268
8.2. Rola wiedzy eksperckiej w SWD . . . . .	271
8.2.1. Wiedza eksperta zapisana w bazie przypadków . . . . .	271
8.2.2. Rola przestrzeni tolerancji w przetwarzaniu wiedzy eksperta . . . . .	272
8.2.3. Formułowanie zapytań — prezentacja wyводу eksperckiego . . . . .	273
8.3. Pytania i zadania kontrolne . . . . .	276
8.3.1. Pytania kontrolne . . . . .	276
8.3.2. Przykładowe zadania . . . . .	277
<b>Rozdział 9. Zastosowanie metod sztucznej inteligencji w SWD . . . . .</b>	<b>278</b>
9.1. Metody sztucznej inteligencji . . . . .	278
9.1.1. Klasyfikacja metod sztucznej inteligencji . . . . .	279
9.1.2. Zastosowanie aparatu logiki matematycznej . . . . .	280
9.1.3. Zastosowanie analogii naturalnych . . . . .	282
9.2. Sieci neuronowe . . . . .	290
9.2.1. Idea i struktura sieci neuronowych . . . . .	291
9.2.2. Strojenie (uczenie) sieci neuronowej . . . . .	294
9.2.3. Model decyzyjny sieci neuronowej . . . . .	296
9.2.4. Przykłady zastosowań sieci neuronowych . . . . .	297
9.3. Algorytmy genetyczne . . . . .	298
9.3.1. Idea budowy i schemat funkcjonowania algorytmów genetycznych . . . . .	298
9.3.2. Model decyzyjny algorytmu genetycznego . . . . .	301
9.3.3. Przykłady praktycznych zastosowań algorytmów genetycznych . . . . .	303
9.4. Algorytmy mrówkowe . . . . .	303
9.4.1. Idea budowy i schemat funkcjonowania algorytmów mrówkowych . . . . .	303
9.4.2. Model decyzyjny algorytmu mrówkowego . . . . .	305
9.4.3. Przykłady zastosowań algorytmów mrówkowych . . . . .	307
9.5. Hybrydowe modele decyzyjne . . . . .	308
9.5.1. Idea i struktura hybrydowych modeli decyzyjnych . . . . .	308
9.5.2. Przykłady rozwiązań hybrydowych modeli decyzyjnych . . . . .	310
9.6. Pytania i zadania kontrolne . . . . .	311
9.6.1. Pytania kontrolne . . . . .	311
9.6.2. Przykładowe zadania . . . . .	312

<b>Rozdział 10. Integracja w systemach SWD</b> . . . . .	313
10.1. Business Intelligence (BI) . . . . .	313
10.1.1. Koncepcja i istota działania BI . . . . .	314
10.1.2. Struktura i metody BI . . . . .	315
10.1.3. Projektowanie rozwiązań BI . . . . .	317
10.2. System ekspertowy . . . . .	323
10.2.1. Struktura i istota działania SE . . . . .	323
10.2.2. Projektowanie SE . . . . .	327
10.2.3. Przykłady zastosowań SE . . . . .	329
10.3. Hybrydowy SWD . . . . .	334
10.3.1. Architektura hybrydowego SWD . . . . .	335
10.3.2. Problemy projektowania i eksploatacji HSWD . . . . .	336
10.3.3. Przykłady zastosowań HSWD . . . . .	336
10.4. Pytania i zadania kontrolne . . . . .	338
10.4.1. Pytania kontrolne . . . . .	338
10.4.2. Przykładowe zadania . . . . .	339
<b>Rozdział 11. Przykłady praktycznych zastosowań SWD</b> . . . . .	340
11.1. System Business Intelligence w zarządzaniu ryzykiem ubezpieczeniowym . . . . .	340
11.1.1. Cechy funkcjonalne systemu BI . . . . .	341
11.1.2. Architektura systemu BI . . . . .	343
11.1.3. Eksperymenty symulacyjne . . . . .	344
11.1.4. Charakterystyki eksploatacyjne systemu BI . . . . .	346
11.2. System ekspertowy w zarządzaniu produkcją rolniczą . . . . .	347
11.2.1. Cechy funkcjonalne SE . . . . .	348
11.2.2. Architektura SE . . . . .	350
11.2.3. Eksperymenty symulacyjne . . . . .	352
11.2.4. Wyniki zastosowania SE w planowaniu wyposażenia przedsiębiorstw rolnych . . . . .	352
11.3. Pytania i problemy kontrolne . . . . .	353
11.3.1. Pytania kontrolne . . . . .	353
11.3.2. Problemy zastosowań SWD . . . . .	354
<b>Podsumowanie</b> . . . . .	355
<b>Bibliografia</b> . . . . .	357
<b>Indeks</b> . . . . .	357

# Wstęp

Prezentowana publikacja stanowi jeden z elementów serii podręczników dedykowanych kierunkom Zarządzanie i Inżynieria Produkcji studiów I i II stopnia na uczelniach wyższych (technicznych, ekonomicznych i rolniczych). Jej układ i zawartość jest zgodna z ministerialnymi wytycznymi odnoszącymi się do przedmiotu *Systemy wspomagania decyzji*. Należy dodać, że w innych podręcznikach serii (np. *Zarządzanie wiedzą*, *Zintegrowane systemy zarządzania*, *Informatyka i komputerowe wspomaganie prac inżynierskich*) można znaleźć uzupełnienia i rozszerzenia pewnych zagadnień, których autorzy nie chcieli w związku z tym powtórnie przytaczać.

Mimo że książka powstała na potrzeby konkretnego kierunku studiów, to z powodzeniem może być wykorzystywana przez studentów innych kierunków, takich jak: zarządzanie, logistyka, administracja oraz kierunki pokrewne. Celem podręcznika jest wprowadzenie Czytelnika w zakres problematyki systemów wspomagania decyzji (SWD) zarówno w zakresie teoretycznych podstaw konstrukcji i budowy, jak i praktycznych aspektów metod ich wdrażania oraz użytkowania. Utrwaleniu przygotowanego materiału służą pytania i zadania kontrolne umieszczone na końcu każdego rozdziału. Szeroki zakres źródeł literaturowych umożliwia samodzielne pogłębienie zawartej w nich wiedzy.

Podręcznik został tak napisany, aby można go było studiować w zależności od potrzeb bieżących i poziomu zaawansowania wiedzy teoretycznej i praktycznej w zakresie informatyki oraz organizacji i zarządzania przedsiębiorstwem. Stąd też w analogicznych sytuacjach te same zagadnienia merytoryczne są prezentowane w różnych ujęciach, z uwzględnieniem ich bieżącego kontekstu i spodziewanych potrzeb Czytelnika w układzie następujących jedenastu rozdziałów:

- rozdział 1 — obejmuje wprowadzenie do problematyki zagadnień SWD w zakresie genezy, znaczenia podstawowych pojęć oraz klasyfikacji systemów wspomagania decyzji;

- rozdział 2 — prezentuje symulacje i techniki komputerowe obejmujące logiczne, statystyczne oraz heurystyczne modele i metody optymalizacyjne oraz eksperymenty symulacyjne stosowane do opracowywania decyzji i przewidywania ich skutków w SWD;
- rozdział 3 — przedstawia interfejs użytkownika SWD w zakresie definiowania cech użytkowych SWD, od których zależy efektywność i poprawność jego zastosowań w procesie komunikacji: SWD–użytkownik (ekspert);
- rozdział 4 — omawia modele abstrakcyjne w projektowaniu SWD w zakresie wybranych modeli teoretycznych, na podstawie których w SWD są formułowane problemy decyzyjne i ich rozwiązania;
- rozdział 5 — dotyczy konkretyzacji rozwiązań projektowych SWD na poziomie projektowania podstawowych elementów składowych architektury systemu;
- rozdział 6 — prezentuje weryfikację i wdrożenie SWD w zakresie dwóch metod, kluczowych dla efektywności wyboru i stosowania SWD: metodyki wielokryterialnego wyboru systemu informatycznego klasy SWD oraz metodyki wdrożenia SWD jako systemu informatycznego;
- rozdział 7 — opisuje wybrane metody i narzędzia inżynierii wiedzy, stosowane w projektowaniu i budowaniu prototypów SWD;
- rozdział 8 — przedstawia wpływ i metody oceny skuteczności działania SWD na gospodarczą działalność organizacji;
- rozdział 9 — obejmuje zastosowanie w SWD wybranych metod sztucznej inteligencji, takich jak: sieci neuronowe, algorytmy genetyczne oraz algorytmy mrówkowe;
- rozdział 10 — prezentuje klasę systemów ekspertowych (SE), której przedstawiciele należą do grupy zintegrowanych hybrydowych SWD, najbardziej zaawansowanych pod względem wykorzystywanych technologii inżynierii wiedzy;
- rozdział 11 — zawiera przykłady rzeczywistych zastosowań SWD w organizacjach polskiej gospodarki.

Autorzy mają nadzieję, że lektura podręcznika będzie inspiracją dla Czytelnika do samodzielnych poszukiwań metod i sposobów efektywnego rozwiązywania problemów decyzyjnych w organizacjach gospodarczych.

# Istota systemów wspomagania decyzji

## 1.1. Geneza i znaczenie systemów wspomagania decyzji (SWD)

Rozwój technologii informacyjnych umożliwił efektywne wspieranie i częściową automatyzację procesu podejmowania decyzji. Metody i techniki sztucznej inteligencji, systemy ekspertowe, hurtownie i składnice danych oraz bazy wiedzy są przykładami rozwiązań, które mogą być wykorzystywane w tym obszarze. Przybliżenie znaczenia i istoty rozwiązywania problemów decyzyjnych nie jest łatwe, jednak autorzy tego podręcznika podjęli to wyzwanie, starając się je ukazać w możliwie najbardziej przystępnym i szerokim kontekście.

W tym rozdziale przedstawiono genezę, a także główny cel, jakiemu służą SWD. Przykłady ilustrują historię rozwoju i stan współczesnych SWD. Podkreślono znaczenie SWD we wspomaganiu procesów decyzyjnych na różnych poziomach zarządzania. Znaczną część rozważań skupiono na tendencjach i kierunkach rozwoju obserwowanych we współczesnych SWD.

### 1.1.1. Definicja SWD i historia powstania

Jedną z pierwszych definicji systemów wspomagania decyzji podaje, że są to systemy wspierające biznesowe i organizacyjne czynności decyzyjne. Przez systemy SWD (ang. *Decision Support System* — DSS) rozumie się najczęściej rozwiązania informatyczne, które sprawdzają się w takich sytuacjach decyzyjnych, dla których nie istnieją jednoznaczne procedury prowadzące do optymalnego rozwiązania.

Podstawowym zadaniem SWD jest dostarczanie dokładnych, przetworzonych informacji dla menedżerów, handlowców i analityków. Informacje te są przez nich wykorzystywane przy podejmowaniu decyzji operacyjnych, taktycznych i strategicznych — aby informacje zostały wykorzystane w systemie decyzyjnym organizacji, SWD powinien również realizować zadanie przygotowania wariantów decyzji oraz zadanie wspomaganie wyboru optymalnego wariantu decyzji. To ostatnie zadanie dotyczy przede wszystkim podejmowania decyzji taktycznych i strategicznych,

których podjęciu towarzyszy zwiększone ryzyko popełnienia błędu i poniesienia kosztów znacznie wyższych niż w przypadku decyzji operacyjnych.

Według pionierów komputerowego wspomagania decyzji P. Keena i Ch. Stabella [Keen, Scott-Morton 1978] rozwój SWD zapoczątkowały prowadzone pod koniec lat 50. ubiegłego stulecia teoretyczne studia nad zagadnieniami podejmowania decyzji w organizacjach oraz rozwój informatycznych systemów interaktywnych na początku lat 60.

Pierwsze koncepcje SWD pojawiły się w połowie lat 60., kiedy na szeroką skalę zaczęły powstawać systemy informacyjne zarządzania (SIZ, ang. MIS), nazywane również systemami informatycznymi zarządzania. Głównym zadaniem SIZ było cykliczne dostarczanie raportów i zestawień statystycznych, przygotowywanych w oparciu o dane transakcyjne oraz dane finansowe przedsiębiorstwa.

Intensywne badania prowadzone nad SIZ doprowadziły do powstania nowego ich typu, nazwanego systemem wspomagania decyzji zorientowanym na model (ang. *model-oriented*) lub systemem wspomagania decyzji zarządczych (ang. MDS).

W roku 1971 została opublikowana praca doktorska M.S. Scotta-Mortona *Systemy wspomagania decyzji zarządczych. Komputerowe wspomaganie podejmowania decyzji* [Scott-Morton 1971], która uważana jest za pierwsze opracowanie naukowe dotyczące SWD. W części badawczej M.S. Scott-Morton obserwował efektywność modeli analitycznych we wspieraniu podejmowania decyzji zarządczych. W tym celu przeprowadził wiele eksperymentów badawczych, polegających na wykorzystaniu do koordynowania planowania zaopatrzenia pralni przygotowanego przez siebie systemu MDS. Jego praca zawierała definicję, implementację oraz testy jakościowe systemu wspomagania decyzji zorientowanego na model.

Również w 1971 r. T.P. Gerrity Jr. przedstawił artykuł, dokumentujący jego projekt systemu wspomagającego decyzje inwestycyjne [Gerrity 1971, s. 59–75]. System był dedykowany wspomaganianiu zarządzania portfelem klientów przez maklerów giełdowych i zapoczątkował rozwój SWD w obszarze wspomaganiania finansów i inwestycji.

Rozwój relacyjnych baz danych najpierw umożliwił, a potem przyspieszył dalsze badania nad rozwojem SWD. Zmierzają one teraz w kierunku wprowadzenia jednolitej klasyfikacji oraz precyzyjnego zdefiniowania specjalizacji SWD. W roku 1979 J. Rockart opublikował artykuł [Rockart 1979, s. 81–93], w których wyróżnił wśród systemów wspomagających podejmowanie decyzji:

- systemy informowania kierownictwa (SIK, ang. EIS),
- systemy wspomaganiania kierownictwa (SWK, ang. ESS).

W roku 1981 R.H. Bonczek, C.W. Holsapple i A.B. Whinston w swojej książce *Podstawy systemów wspomagania decyzji* [Bonczek, Holsapple, Whinston 1981] wprowadzili teoretyczne podstawy SWD, ukierunkowanych na wiedzę. Zauważyli, że takie dyscypliny naukowe, jak sztuczna inteligencja oraz systemy ekspertowe mogą znacząco wpłynąć na rozwój SWD.



Lata 80. ubiegłego wieku przyniosły rozwój badań nad SWD wspierającymi grupowe i organizacyjne podejmowanie decyzji zarządczych. Pierwsze rozwiązania w tym zakresie opisali w swojej pracy G. DeSanctis i R.B. Gallupe [DeSanctis, 1987, s. 589–609]. Przez kolejne dziesięciolecie temat ten był przedmiotem wielu prac, eksperymentów i badań, a systemy tej kategorii zyskały nazwę grupowych systemów wspomagania decyzji (ang. GDSS).

Lata 90. to rozwój technologii hurtowni danych oraz analitycznego przetwarzania danych (ang. OLAP). Spowodowały one odejście od pierwotnych systemów SWD ukierunkowanych na model do systemów SWD ukierunkowanych na dane. Obecnie ta właśnie kategoria systemów SWD rozwija się najszybciej. Dodatkowo upowszechnienie technologii internetowych wpłynęło na mobilność i ogólną dostępność rozwiązań SWD w zastosowaniach organizacyjnych.

Obecnie badania nad SWD [Power 2002, s. 5] w obszarze baz i hurtowni danych koncentrują się na efektywnym zarządzaniu danymi oraz dokumentami w skali całej organizacji. W zakresie nauk o zarządzaniu oraz badań operacyjnych doskonałone są rozwiązania wykorzystywane w systemach SWD ukierunkowanych na model. Nauki poznawcze, a szczególnie behawioralne, przyczyniają się do uwzględniania w systemach SWD także informacji opisowych i empirycznych.

Prowadzone są również prace nad upowszechnieniem rozwiązań SWD poprzez ich sieciowe udostępnienie i interakcyjną komunikację z użytkownikiem [Olszak 2000], [Olszak 2007]. Systemy SWD stają się zatem obszarem integrującym wiele dyscyplin wiedzy, do których można zaliczyć: sztuczną inteligencję, inżynierię programowania, inżynierię wiedzy, rozpoznawanie obrazów i uczenie maszyn czy telekomunikację.

### 1.1.2. Pierwsze przykłady — funkcje i struktura SWD

Pierwsze wykorzystanie systemu SWD odnotowano 4 listopada 1952 r. — w dniu wyborów prezydenckich w USA. Wtedy właśnie pięć komputerów UNIVAC zostało zaangażowanych do przetworzenia częściowych wyników wyborów prezydenckich i prognozowania na tej podstawie wygranej. Jak relacjonował potem J.A.N. Lee w swoim artykule, opublikowanym przez IEEE Computer Society Press w 1996 r. [Lee 1996], już kilka minut po zamknięciu ostatnich urn wyborczych komputer UNIVAC był w stanie przewidzieć zwycięstwo D. Eisenhowera nad A. Stevensonem II. Jednakże stacja telewizyjna CBS, która była w posiadaniu tych wyników, nie odważyła się podać owych przewidywań do publicznej wiadomości. Kiedy jednak tuż po północy prognozowany przez system wynik został potwierdzony, prowadzący program złożył publiczne oświadczenie w sprawie pierwszego spektakularnego sukcesu systemu SWD.

W roku 1958 rozpoczęto prace nad pierwszym systemem SWD, ukierunkowanym na dane, o bardzo szerokim spektrum działania. Główną funkcjonalnością systemu SAGE była koordynacja działań urządzeń radiolokacyjnych oraz samolotów

patrolujących granice. Głównym jego zadaniem była wczesna informacja o nieprzyjacielskich samolotach, naruszających patrolowany obszar. System SAGE został w pełni uruchomiony w 1963 r., a pewne jego elementy były wykorzystywane do 1983 r. Według R.R. Everetta [Everett, Zraket, Bennington 1957, s. 148–155] SAGE był systemem czasu rzeczywistego w zakresie: kontroli, łączności oraz informowania kierownictwa.

Również w 1963 r. pojawił się system JOSS, będący pierwszym systemem SWD ukierunkowanym na model, którego zadanie polegało na wspieraniu wytwarzania oprogramowania. JOSS został uznany za pionierskie rozwiązanie w obszarze integracji inżynierii programowania, sztucznej inteligencji oraz technik interaktywnych.

W roku 1965 grono naukowców ze Stanford University opracowało pierwszy system ekspertowy DENDRAL [Harmon, King 1985]. Podstawowym jego zadaniem było ustalanie struktury molekularnej nieznanymi chemicznie związków organicznych na podstawie analizy ich widm spektroskopowych. System opracowano z wykorzystaniem języka INTERLISP oraz algorytmu J. Lederberga, który generował wszystkie możliwe struktury cząsteczkowe. System osiągnął sprawność porównywalną, a w niektórych przypadkach nawet przewyższającą ludzkie możliwości ekspertów. Kontynuacje systemu DENDRAL stały się typowymi narzędziami laboratoriów chemicznych.

W roku 1967 M.S. Scott-Morton w swojej pracy doktorskiej przedstawił system MDS (ang. *Management Decision System*) wspierający podejmowanie powtarzalnych decyzji planistycznych [Scott-Morton 1971]. Narzędzie przygotowane przez M.S. Scotta-Mortona jest uważane za pionierskie rozwiązanie w obszarze SWD ukierunkowanych na model.

Innym spektakularnym przykładem jest przeprowadzenie 9 grudnia 1968 r. publicznej demonstracji możliwości systemu NLS [Power 2006]. Był to pierwszy pokaz działania myszki komputerowej jako narzędzia wskazywania wariantu decyzyjnego oraz wielu pionierskich funkcjonalności interaktywno-multimedialnych, takich jak: tworzenie hipertekstu, dynamiczne adresowanie plików sieciowych czy audio-wizualna komunikacja dwóch osób za pośrednictwem sieci.

Szybki rozwój systemów SWD spowodował potrzebę ich klasyfikacji. W roku 1980 S. Alter wyróżnił trzy podstawowe wymagania dla każdego systemu SWD [Alter 1980]:

- 1) są projektowane w celu wspomagania procesu podejmowania decyzji,
- 2) powinny wspomagać, a nie automatyzować podejmowanie decyzji,
- 3) powinny elastycznie dostosowywać się do ciągle zmieniających się wymagań osób podejmujących decyzje.

Kolejne definicje systemu SWD podkreślają, że jest to interaktywny system informatyczny, umożliwiający osobom podejmującym decyzje wykorzystywanie danych i modeli do rozwiązywania problemów o różnym poziomie strukturyzacji [Sprague, Carlson 1982]. Ten główny cel daje się osiągać za pomocą różnych metod

i w obszarze bardzo odmiennych problemów decyzyjnych. To zatem, co odróżnia systemy SWD między sobą, to: typ i liczba użytkowników końcowych systemu, rodzaj wykonywanych analiz, ukierunkowanie systemu (na model, na dane, na komunikację lub na proces). Współczesna typologia SWD uwzględnia pięć czynników klasyfikacyjnych, decydujących o przynależności do odpowiedniej grupy (tablica 1.1).

**Tablica 1.1. Kryteria klasyfikacyjne systemów SWD**

Nazwa klasyfikatora	Wartości klasyfikatora
Dominujący element architektury systemu	<ul style="list-style-type: none"> <li>• Komunikacja</li> <li>• Baza danych</li> <li>• Baza dokumentów</li> <li>• Baza wiedzy</li> <li>• Modele wspomagania decyzji</li> </ul>
Użytkownicy systemu	<ul style="list-style-type: none"> <li>• Wewnętrzni</li> <li>• Zewnętrzni</li> </ul>
Cel funkcjonowania	<ul style="list-style-type: none"> <li>• Wąski</li> <li>• Szeroki</li> </ul>
Wykorzystana technologia	<ul style="list-style-type: none"> <li>• Mainframe</li> <li>• Sieć komputerowa</li> <li>• Klient/serwer</li> <li>• Komputer osobisty</li> </ul>

Źródło: Opracowanie własne na podstawie: [Power 2002, s. 16].

Najważniejszym kryterium klasyfikacyjnym jest dominujący element architektury systemu i to on różnicuje systemy SWD na 5 podstawowych typów:

- 1) SWD ukierunkowane na komunikację (w tym SWD grupowe),
- 2) SWD ukierunkowane na dane,
- 3) SWD ukierunkowane na dokumenty,
- 4) SWD ukierunkowane na wiedzę,
- 5) SWD ukierunkowane na model wspomagania decyzji.

Pozostałe kryteria (użytkownicy systemu, cel funkcjonowania oraz wykorzystana technologia) pozwalają wydzielić jeszcze trzy dodatkowe podkategorie SWD:

- 1) międzyorganizacyjne i wewnątrzorganizacyjne,
- 2) wąskiego i szerokiego przeznaczenia,
- 3) sieciowe oraz procesowe.

W SWD ukierunkowanym na komunikację oraz w SWD grupowym dominującym elementem architektury systemu jest swoboda wzajemnego porozumiewania się i umożliwienia współpracy grupowej. SWD ukierunkowany na dane to obecnie naj-

prężniej rozwijająca się grupa systemów SWD. Obejmuje ona takie rozwiązania, jak: systemy informowania kierownictwa (ang. *Executive Information System* — EIS), systemy informacji zarządczej (ang. *Business Intelligence* — BI) oraz systemy hurtowni danych (ang. *data warehouse*).

SWD ukierunkowane na dokumenty pozwalają na pozyskiwanie i zarządzanie informacjami, przechowywanymi w dokumentach nieustrukturalizowanych lub na stronach internetowych. Zadaniem SWD ukierunkowanych na wiedzę jest sugerowanie optymalnego rozwiązania decyzyjnego na podstawie przeprowadzonej przez system ekspertyzy problemu. Do grupy wykorzystywanych w tej kategorii rozwiązań należą systemy ekspertowe (ang. *Expert System* — ES) oraz systemy przetwarzania eksploracyjnego danych (ang. *data mining*).

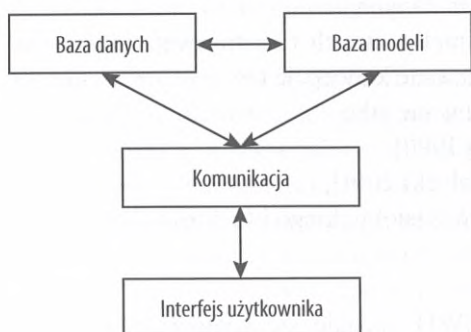
SWD ukierunkowane na model obsługują analizy: matematyczne, finansowe i optymalizacyjne. Charakteryzują się niewielką liczbą danych oraz parametrów, które często na bieżąco są wprowadzane (lub generowane) przez użytkownika systemu w celu uzyskania wyników prowadzonej analizy. Przykładem hybrydowego SWD ukierunkowanego na model są systemy przetwarzania analitycznego na bieżąco (ang. *Online Analytical Processing* — OLAP), które łączą w sobie dużej wielkości zbiory danych (liczone w giga- lub terabajtach i pochodzące zazwyczaj z hurtowni danych, będącej elementem dominującym SWD ukierunkowanego na dane) z modelami statystycznymi (będącymi cechą charakterystyczną systemów SWD ukierunkowanych na model).

W SWD ukierunkowanych na proces decyzja jest podejmowana w wyniku realizacji procedury algorytmicznej, w której różnym końcowym operacjom odpowiadają zróżnicowane decyzje (zdarzenia) ze zbioru wcześniej zaplanowanych alternatyw. W tym celu alternatywy decyzji i ich przesłanki (zdarzenia oraz towarzyszące im stany) gromadzi się w bazie wiedzy SWD w oparciu o nabyte pozytywne doświadczenia operatorów-ekspertów. Decyzje i przesłanki procesu są modelowane w postaci sieci zdarzeń i odpowiadających im formuł zdarzeń [Krupa 2006b], [Krupa 2009].

W systemach do bezpośredniego i ciągłego nadzoru funkcjonowania obiektów (np. bloków energetycznych w elektrowniach konwencjonalnych i atomowych ([Maj, Krupa 2008], [Maj, Krupa 2010])) stosuje się zaawansowane hybrydowe rozwiązania SWD.

SWD wewnątrz- i międzyorganizacyjne charakteryzują się wykorzystaniem technologii sieciowych. Różnica polega na zasięgu wykorzystania systemu, który w przypadku wariantu wewnątrzorganizacyjnego jest ograniczony jedynie do obszaru wewnętrznego organizacji, natomiast w wersji międzyorganizacyjnej wykracza poza jej granice. SWD wąskiego i szerokiego przeznaczenia odróżnia zakres funkcjonalności biznesowej systemu. SWD szerokiego przeznaczenia mają ogólnie zdefiniowany cel funkcjonowania, np. realizacja zapytań analitycznych użytkownika. SWD wąskiego przeznaczenia szczegółowo definiuje zakres funkcjonalności systemu, np. optymalizacja harmonogramu dostaw.

Rysunek 1.1. Struktura systemu wspomaganie decyzji



Źródło: [Power 2002, s. 17], [Czermiński 2002, s. 80].

Charakteryzując SWD, należy wskazać wartości każdego z czterech podstawowych klasyfikatorów różnicujących te systemy (tablica 1.1), np. określić SWD jako: sieciowy, intraorganizacyjny, wąsko specjalizowany (dedykowany optymalizacji obsługi klienta w firmie) i ukierunkowany na dane. Kompozycja wartości tych czterech czynników ma decydujący wpływ na strukturę SWD, która musi zawierać cztery podstawowe komponenty: bazę danych, bazę modeli, warstwę komunikacyjną oraz interfejs użytkownika (rysunek 1.1) [Czermiński 2002, s. 80].

Interfejs użytkownika jest warstwą, która odpowiada za komunikację użytkownika z SWD. Warstwa komunikacyjna, obejmująca architekturę techniczną SWD, realizuje polecenia użytkownika wydane z poziomu interfejsu. Realizacja polecenia wymaga każdorazowo zaangażowania zasobów danych, pobieranych z bazy danych oraz odpowiednich modeli analitycznych, dostępnych w bazie modeli. Ta podstawowa koncepcja komputerowego wspomaganie decyzji może być różnie realizowana w różnych typach SWD, co zostanie zaprezentowane w kolejnych rozdziałach książki.

### 1.1.3. Główne koncepcje, obszary zastosowań i kierunki rozwoju SWD

Kolejne okresy historii SWD:

- lata 60. XX w. — rozwój *Management Information Systems*, rozwój teorii SWD, badania nad systemami interaktywnymi;
- lata 70. XX w. — pierwsze biznesowe wdrożenia SWD, rozwój systemów zarządzania bazami danych;
- lata 80. XX w. — różnicowanie SWD ze względu na przeznaczenie biznesowe oraz zakres wsparcia decyzji zarządczych, rozwój systemów informowania kierownictwa (EIS) oraz systemów ekspertowych (SE);

- lata 90. XX w. — rozwój internetowych SWD, rozwój technologii hurtowni danych oraz analitycznego przetwarzania danych;
- współczesność — rozwój technologii inteligentnych i zastosowań nauk behawioralnych w SWD wpłynęły na współczesne koncepcje teoretyczne w zakresie systemów wspomagania decyzji, definiowane jako:
  - następcy DIS Z. Hellwiga [Hellwig 1990],
  - SWD według W.T. Bieleckiego [Bielecki 2001],
  - SWD i bazy wiedzy (BW) według J. Kisielnickiego i H. Sroki [Kisielnicki, Sroka 2005].

Typowa architektura współczesnych SWD cechuje się wielowymiarowością modelu logicznego, wielowymiarowością modelu fizycznego (tablice wymiarów, tablice relacji, tablice faktów oraz tablice transformacji) oraz ujęciem procesowym. Dominująca funkcjonalność obserwowana we współczesnych SWD to: intuicyjność interfejsu, krótki czas reakcji systemu na żądanie wygenerowania raportu, zdolność poruszania się w strukturze danych hierarchicznych z poziomu raportu końcowego (zwana rozwijaniem danych), możliwość zagnieżdżania wymiarów w raporcie tabelarycznym, rotacja, filtrowanie i sortowanie danych, przejścia z raportu tabelarycznego na graficzny, dostęp do danych z innych aplikacji (np. MS Excel, Lotus), projektowanie wykonywalnych (powtarzalnych) raportów.

Rozwój współczesnych SWD następuje w wyniku postępu nauki, technologii oraz w rezultacie kształtowania się nowych zastosowań naukowo-technicznych, biznesowych i administracyjno-społecznych. Współczesne SWD mogą być wykorzystywane do rozwiązywania problemów decyzyjnych:

- związanych z gromadzeniem, porządkowaniem i udostępnianiem informacji zarządczych;
- dotyczących gromadzenia, porządkowania i udostępniania wiedzy;
- dotyczących zapewnienia jakości przechowywanej i udostępnianej informacji;
- ochrony bezpieczeństwa dostępu do informacji i wiedzy;
- zapewniających szybkość i efektywność podejmowania decyzji.

Postęp nauki (zwłaszcza osiągnięcia w zakresie budowy modeli sztucznej inteligencji, powstające licznie nowe metody modelowania procesów i prowadzenia zaawansowanych analiz ekonometrycznych, metody holistyczne modelowania sytuacyjnego i refleksyjnego systemów organizacyjno-technicznych oraz hybrydowe systemy identyfikacji i wydobywania informacji, a także gromadzenia wiedzy z danych operacyjnych) powoduje, że procesy i systemy podejmowania decyzji w coraz większym stopniu stają się domeną technologii informacyjnych realizowanych za pomocą współczesnych środków techniki komputerowej. Pod względem zastosowań SWD mogą być dostarczane jako rozwiązania:

- branżowe — wspomagające procesy produkcyjne, wykorzystywane w służbie zdrowia lub w zaawansowanym rozpoznawaniu chorób lub zagrożeń chorobowo-

wych, w transporcie lądowym, morskim lub powietrznym, w zastosowaniach rolniczych, we wspomaganiu działalności bankowej i ubezpieczeniowej instytucji finansowych;

- funkcjonalne — wyspecjalizowane w określonych zakresach działalności wspólnej dla różnych branż, np. jako systemy wykrywania zagrożeń płynności finansowej, a także jako systemy wykrywania zagrożeń powodziowych, pożarowych, chemicznych, terrorystycznych, stosowane w procesach kierowania personelem, zarządzania projektami, w procesach logistycznych itp.;
- o szczególnym znaczeniu dla bezpieczeństwa państwa — systemy obronności cywilnej i wojskowej, systemy antyterrorystyczne, systemy zarządzania sytuacją kryzysową.

## 1.2. Fazy procesu decyzyjnego

Konieczność podejmowania decyzji, często w warunkach niepewności i bez możliwości dostępu do potrzebnych informacji, jest nieodłącznie związana z procesem zarządzania przedsiębiorstwem. Wiąże się z nią zawsze ryzyko podjęcia niewłaściwej decyzji, której konsekwencje będą odczuwalne przez dłuższy czas. W sytuacjach szczególnych niewłaściwa decyzja (lub raczej ciąg niewłaściwie podjętych decyzji) może doprowadzić przedsiębiorstwo do stanu upadłości, w pozostałych przypadkach spowoduje większe lub mniejsze problemy związane z jego funkcjonowaniem. Zawsze jednak jest sytuacją niepożądaną. Celem wdrożenia i głównym zadaniem SWD jest zatem wspomaganie realizacji procesu podejmowania decyzji na wszystkich jego etapach, prowadzące do zmniejszenia stopnia ryzyka towarzyszącego realizacji tego procesu.

Zdaniem W.T. Bieleckiego [Bielecki 2001, s. 17], podejmowanie decyzji oznacza dokonanie ostatecznego wyboru wariantu decyzji po rozważeniu alternatywnych możliwości działania. Wynika z tego, że jest to proces wymagający dokonania: identyfikacji i strukturyzacji problemu decyzyjnego, zebrania wiedzy eksperckiej w celu opracowania możliwych wariantów rozwiązań tego problemu, dokonania metody ich oceny oraz selekcji, podjęcia decyzji i jej wdrożenia w działanie. Szczegółowe definicje związane z problematyką decyzji oraz procesu decyzyjnego zostaną przedstawione w poniższym punkcie.

### 1.2.1. Problem decyzyjny

Decyzja jest podejmowana w momencie, kiedy pojawi się sytuacja decyzyjna, czyli zaistnieją co najmniej dwa możliwe warianty działania, różniące się między sobą stopniem korzyści. Wtedy też mówi się o pojawieniu się **problemu decyzyjnego**, czyli konieczności porównania stanu oczekiwanego ze stanem rzeczywistym, pomiaru występujących między nimi odchyżeń i na tej podstawie dokonania wyboru jednego z istniejących wariantów działania.

Nawiązując do przedstawionych faz procesu decyzyjnego, należy zauważyć, że rozwiązanie problemu decyzyjnego wymaga utworzenia co najmniej jednego modelu decyzyjnego, którego zadaniem jest wybranie takiej decyzji, dla której stopień korzyści jest możliwie największy. Matematycznie model problemu decyzyjnego można przedstawić jako zbiór zmiennych decyzyjnych  $S_x$ , wyjść modelu decyzyjnego  $S_y$  oraz dopuszczalnych decyzji  $X$ , takich że (formuła [1.1] i rysunek 1.2) [Granat 2009]:

$$f: S_x \rightarrow S_y \text{ oraz } Y = f(X) \subset S_y, \quad [1.1]$$

gdzie:

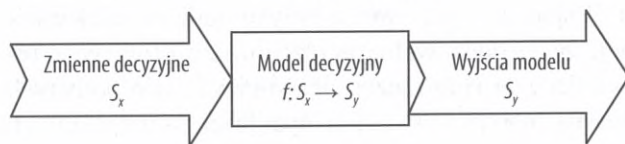
$S_x$  — zbiór zmiennych decyzyjnych,

$X$  — zbiór decyzji dopuszczalnych,

$S_y$  — zbiór wyjść modelu decyzyjnego,

$Y$  — zbiór dopuszczalnych wartości wyjść modelu decyzyjnego.

**Rysunek 1.2. Definicja modelu decyzyjnego**



Źródło: Opracowanie własne.

Szczególną kategorię problemów decyzyjnych stanowią problemy o bardzo dużym stopniu złożoności. Są to rzeczywiste problemy występujące w jakiejś dziedzinie życia lub ich komputerowe symulacje (np. program symulujący zarządzanie firmą). Im wyższy jest stopień złożoności problemu, tym jego rozwiązanie wymaga większej wiedzy, nakładów finansowych, osobowych oraz finansowych, co uzasadnia potrzebę wsparcia go przez SWD.

## 1.2.2. Decyzja

**Decyzja** jest świadomym wyborem jednego z rozpoznanych i uznanych za możliwe wariantów działania [Kozłowski, Piotrowski 1996, s. 102]. Elementami charakterystyki i jednocześnie klasyfikacji decyzji są [Szarfenberg 2007]:

- przedmiot decyzji — dziedzina, w której znajduje się zbiór dopuszczalnych i możliwych wariantów decyzji;
- zakres decyzji — zbiór dopuszczalnych i możliwych wariantów decyzji (co najmniej dwóch);
- podmiot decyzji — decydent dokonujący świadomego i nielosowego wyboru wariantu decyzji;
- metoda decyzji — sposób, w jaki dokonywana jest redukcja wariantów decyzyjnych;



- treść decyzji — zbiór wyselekcjonowanych wariantów decyzji;
- wynik decyzji — pojedynczy wyselekcjonowany wariant decyzji.

Podstawą decydowania jest świadome i nielosowe porównywanie wariantów decyzji, zidentyfikowanych jako dopuszczalne i możliwe. Selekcja tychże wariantów za pomocą wybranej metody oraz podjęcie decyzji realizowane jest w procesie decyzyjnym.

### 1.2.3. Proces decyzyjny

**Proces decyzyjny** jest to ogół logicznie ze sobą powiązanych działań, które są podejmowane w celu rozwiązania problemu decyzyjnego poprzez wybór jednego z możliwych wariantów decyzji. Składają się na niego następujące fazy ([Witkowski 2010, s. 548–549], rysunek 1.3):

- analizy — obejmuje etapy identyfikacji problemu, zbierania danych, gromadzenia wiedzy pochodzącej od ekspertów, a także precyzowania i uszczegóławiania problemu decyzyjnego oraz jego struktury;
- projektowania — dokonuje się w niej poszukiwania lub tworzenia modeli problemu, ich analizy, gromadzenia wiedzy o możliwościach rozwiązań oraz dokonuje się podziału rozwiązania problemu na etapy możliwe do realizacji;
- wyboru — dokonuje się w tej fazie selekcji optymalnego rozwiązania ze zbioru dostępnych alternatyw;
- implementacji — realizuje się przyjęte w poprzednim etapie rozwiązanie;
- oceny — dokonuje się oceny uzyskanych wyników.

Matematycznie proces decyzyjny można przedstawić w postaci formuły, uwzględniającej warunki oraz skutki podejmowanej decyzji [Kisielnicki 2008b, s. 61]:

$$PD = \{P, S, W, H\}, \quad [1.2]$$

gdzie:

$PD$  — podmiot decyzji, czyli decydent;

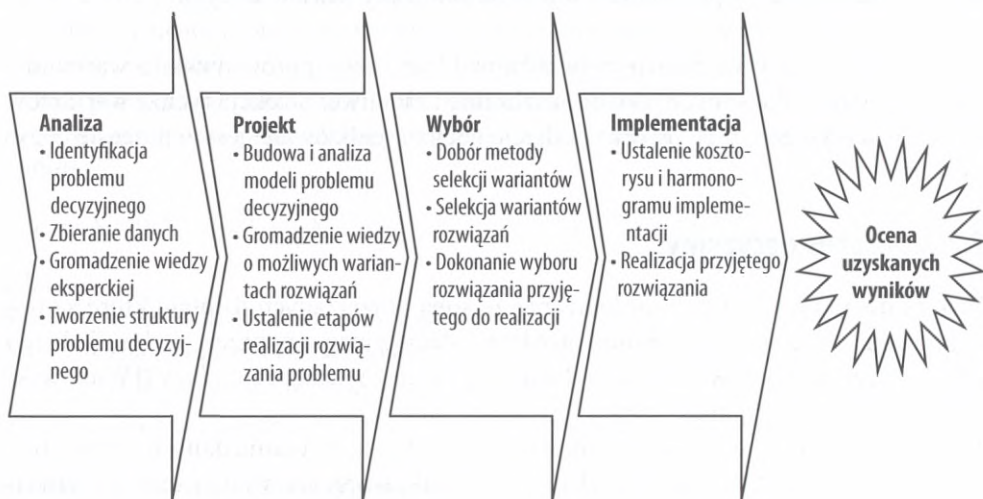
$S$  — zbiór sytuacji decyzyjnych (warunków decyzyjnych), mających bezpośredni wpływ na podejmowaną decyzję;

$W$  — zbiór przewidywanych wyników (skutków decyzji), uwzględniających wpływ poszczególnych warunków decyzyjnych;

$H$  — zbiór hipotez dotyczących prognozowania przyszłych sytuacji (warunków) kształtujących decyzje.

Prawidłowe zidentyfikowanie zbiorów  $S$  i  $W$  z formuły [1.2] wymaga określenia zakresu działania podejmowanej decyzji. Ułatwieniem w tym zakresie staje się dokonanie klasyfikacji decyzji względem charakteryzujących ją kryteriów.

Rysunek 1.3. Etapy realizacji procesu decyzyjnego



Źródło: Opracowanie własne.

### 1.3. Poziomy procesów decyzyjnych

Poznanie **typologii decyzji** stanowi istotny element identyfikacji i analizy problemów decyzyjnych. Klasyfikację decyzji można prowadzić ze względu na wiele czynników kategoryzujących, prezentowanych w tabelicy 1.2.

Tabela 1.2. Typologia decyzji

Czynnik klasyfikujący	Typy decyzji
Rola w procesie zarządzania	Strategiczne, taktyczne lub operacyjne
Struktura problemu decyzyjnego	Ustrukturalizowane, słabo ustrukturalizowane, nieustrukturalizowane
Stopień ryzyka	Podjęwane w warunkach pewności, ryzyka lub niepewności
Zakres rozwiązania	Zamknięte lub otwarte
Cel implementacji rozwiązania	Regulacyjne, sterujące lub innowacyjne
Współuczestnictwo	Indywidualne, indywidualne ze zbiorowym rozpoznaniem lub grupowe
Źródło problemu	Inicjowane przez jednostkę organizacyjną, kierownictwo lub pracowników szeregowych
Rola w funkcji zarządzania	Planistyczne, organizatorskie, koordynacyjne, rozkazodawcze oraz kontrolne

Źródło: Opracowanie własne.

Ze względu na potrzebę zastosowania SWD najistotniejsza jest klasyfikacja decyzji względem roli w procesie zarządzania, struktury problemu decyzyjnego oraz stopnia ryzyka.

### 1.3.1. Poziomy decyzji w procesie zarządzania

Według A.K. Koźmińskiego i M. Zawiślaka [Koźmiński, Zawiślak 1982] treścią procesu zarządzania jest stałe podejmowanie decyzji strategicznych, w wyniku których ustalane są cele dla całej organizacji, a następnie decyzji taktycznych i operacyjnych zapewniających możliwość osiągnięcia tychże celów. Porównanie trzech poziomów decyzji w procesie zarządzania ze względu na kryteria: czasu, stopnia ryzyka, struktury, powtarzalność decyzji oraz źródła informacji prezentuje tablica 1.3.

Tablica 1.3. Porównanie decyzji w procesie zarządzania

Kryteria	Decyzje		
	strategiczne	taktyczne	operacyjne
Horyzont czasu	Lata	Miesiące	Dni
Stopień ryzyka	Wysoki	Malejący	Niski
Struktura problemu decyzyjnego	Znana w niewielkim stopniu	Znana częściowo	Znana całkowicie
Powtarzalność decyzji	Trudno programowalne	Mieszane	Programowalne
Źródła informacji	Zewnętrzne	Mieszane	Wewnętrzne

Źródło: Opracowanie własne na podstawie: [Bielski 1997, s. 248].

### 1.3.2. Poziomy decyzji ze względu na stopień ich strukturalizowania

Stopień ustrukturalizowania decyzji ściśle wiąże się z rolą decyzji w procesie zarządzania, czyli stanowi uszczegółowienie charakterystyki poziomów decyzji z poprzedniego punktu.

Na poziomie operacyjnym decyzje są dobrze ustrukturalizowane. Są powtarzalne, rutynowe, wykonywane zgodnie z instrukcjami i procedurami. Metody poszukiwania rozwiązania problemu decyzyjnego są relatywnie proste, a przede wszystkim powszechnie znane i stosowane.

Na poziomie taktycznym pojawiają się decyzje słabo ustrukturalizowane. Są to decyzje, w których znaczącą rolę odgrywają doświadczenie, preferencje i intuicja decydenta. Dodatkowo wpływają na nie ograniczenia decyzji oraz subiektywne podejście osoby podejmującej decyzję. Na tym poziomie wszelkiego rodzaju metody

poszukiwania rozwiązania wymagają indywidualnego wypracowania w oparciu o sprawdzone wzorce i modele.

Na szczeblu strategicznym decyzje są nieustrukturalizowane, a długi horyzont czasowy podejmowania decyzji czyni je najtrudniejszymi w tej grupie. Ogromne znaczenie ma w tym przypadku kreatywność i zdolność do twórczego myślenia decydenta, ponieważ każda sytuacja decyzyjna wymaga indywidualnego podejścia.

### **1.3.3. Poziomy decyzji ze względu na stopień ryzyka decyzyjnego**

Stopień ryzyka związanego z podejmowaniem decyzji nieodłącznie wiąże się z zasobem potrzebnych informacji, którymi dysponuje decydent w chwili podejmowania decyzji. Rozróżnia się w tej kategorii trzy zasadnicze poziomy procesu decyzyjnego — podejmowanie decyzji w warunkach pewności, ryzyka oraz niepewności.

Możliwość podejmowania decyzji w warunkach pewności jest sytuacją komfortową dla decydenta i oznacza, że decydent ma pełną informację dotyczącą podejmowanej decyzji, w związku z czym potrafi przewidzieć jej skutki. Przykładowo, zepsuła się drukarka na stanowisku pracy, należy zebrać informacje dotyczące możliwych miejsc naprawy urządzenia i podjąć decyzję dotyczącą dalszego postępowania. Ponieważ przedsiębiorstwa zazwyczaj posiadają sprawdzone procedury działania w takich wypadkach, więc prawdopodobieństwo pomyłki jest niewielkie.

Odmianą sytuacją jest podejmowanie decyzji w warunkach ryzyka, kiedy warunki kształtujące decyzję oraz jej wyniki są szacowane z pewnym prawdopodobieństwem. Przykładowo, podjęcie decyzji o wprowadzeniu nowego produktu na rynek jest zawsze poprzedzone przeprowadzeniem badań rynkowych oraz kampanii marketingowych, które mają za zadanie zdiagnozować gotowość rynku do przyjęcia nowego produktu oraz zachęcić potencjalnych klientów do zakupu. Niestety, nawet najdokładniejsze badania nie są w stanie zagwarantować sukcesu produktu, dopóki jego sprzedaż nie stanie się faktem.

W ostatniej kategorii — podejmowania decyzji w warunkach niepewności mieści się większość decyzji strategicznych, które dotyczą długiego horyzontu czasowego, trudnego do przewidzenia i właściwego oszacowania. Przykładowo, trudno jest zaplanować optymalną strategię inwestycji, jeżeli dotyczy długiego (np. kilkuno-stoletniego) horyzontu czasowego. Dlatego ta kategoria charakteryzuje się brakiem możliwości przewidzenia skutków decyzji, nawet z uwzględnieniem stopnia prawdopodobieństwa.

### 1.3.4. Wspomaganie decyzji ze względu na poziomy procesu decyzyjnego

Jak wynika z porównania prezentowanego w tabelicy 1.4, im wyższy poziom zarządzania oraz niższy poziom strukturalizacji problemu decyzyjnego, tym większa potrzeba wspierania go przez zaawansowane rozwiązania SWD.

Tabela 1.4. Wspomaganie procesu decyzyjnego

Struktura problemu decyzyjnego	Poziomy zarządzania		
	operacyjny	taktyczny	strategiczny
<b>Ustrukturalizowane</b>	<ul style="list-style-type: none"> <li>• Modele zarządzania</li> <li>• Systemy transakcyjne</li> <li>• SIK</li> </ul>	<ul style="list-style-type: none"> <li>• Modele zarządzania</li> <li>• Systemy transakcyjne</li> <li>• SIK</li> <li>• SWD</li> </ul>	<ul style="list-style-type: none"> <li>• Modele zarządzania</li> <li>• Systemy transakcyjne</li> <li>• SIK</li> <li>• SE</li> </ul>
<b>Słabo ustrukturalizowane</b>	<ul style="list-style-type: none"> <li>• Modele zarządzania</li> <li>• SIK</li> <li>• SWD</li> </ul>	<ul style="list-style-type: none"> <li>• Modele zarządzania</li> <li>• SWD</li> <li>• SE</li> </ul>	<ul style="list-style-type: none"> <li>• SWD</li> <li>• SE</li> </ul>
<b>Nieustrukturalizowane</b>	<ul style="list-style-type: none"> <li>• Modele zarządzania</li> <li>• SIK</li> <li>• SWD</li> <li>• SE</li> </ul>	<ul style="list-style-type: none"> <li>• Modele zarządzania</li> <li>• SWD</li> <li>• SE</li> </ul>	<ul style="list-style-type: none"> <li>• SWD</li> <li>• SE</li> </ul>

Źródło: Opracowanie własne.

W zarządzaniu operacyjnym, przy dobrze ustrukturalizowanym problemie decyzyjnym, gdzie występuje niski stopień ryzyka decyzyjnego, w procesie podejmowania decyzji wystarcza znajomość modeli i metod zarządzania, przy ewentualnym wsparciu informacyjnym systemów obsługujących bieżące transakcje oraz przetwarzających statystycznie dane transakcyjne. Wraz z podwyższaniem poziomu i wydłużaniem horyzontu zarządzania oraz malejącą możliwością ustalenia struktury problemu decyzyjnego wzrasta stopień ryzyka decyzyjnego i pojawia się potrzeba zastosowania rozwiązań, które wspomagałyby proces podejmowania decyzji. Rozwiązania takie oparte na przetwarzaniu analitycznym danych o strukturze innej niż transakcyjne, posiadających zdolność gromadzenia i przetwarzania wiedzy, naśladowania pracy ekspertów czy znajdowania rozwiązania optymalnego w zbiorze rozwiązań dopuszczalnych, stanowią treść kolejnych rozdziałów niniejszego podręcznika.

# Symulacje i techniki komputerowe w SWD

W rozdziale przedstawiono logiczne, statystyczne oraz heurystyczne modele i metody optymalizacyjne oraz eksperymenty symulacyjne stosowane do opracowywania decyzji i przewidywania ich skutków. Istotną rolę w procesie przygotowania i podejmowania decyzji odgrywa eksperyment symulacyjny, który umożliwia sprawdzenie różnych wariantów decyzji w krótkim czasie i bez konieczności ich faktycznej realizacji, co eliminuje ryzyko błędu i obniża koszty procesów decyzyjnych.

Uzupełnieniem prezentowanych modeli i metod optymalizacyjnych są techniki kalkulacyjne, a zwłaszcza obliczenia statystyczne, techniki iteracyjne oraz reguły decyzyjne i algorytmy wnioskowania logicznego realizowane w modułach wnioskujących stosowanych w systemach ekspertowych.

Rozdział kończy się przedstawieniem bazy danych (BD) jako podstawowego sposobu gromadzenia i transakcyjnego przetwarzania danych operacyjnych (faktów) towarzyszących procesom podejmowania decyzji. Baza danych stanowi zasadniczą część każdego systemu informatycznego. Proces ich przygotowania na potrzeby SWD rozpoczyna się w momencie projektowania modelu logicznego BD, od którego będą zależeć jej funkcjonalne i eksploatacyjne cechy w procesie przetwarzania faktów i reguł decyzyjnych.

## 2.1. Przewidywanie wyników za pomocą eksperymentów symulacyjnych

Każde zjawisko, proces lub stan występujący w rzeczywistości da się przedstawić w postaci pewnego zbioru wzajemnie ze sobą powiązanych komponentów, których łączne funkcjonowanie zmierza do ściśle określonego celu. Taki zbiór komponentów może być nazwany modelem systemu.

W badaniach zjawisk i procesów rzeczywistych często posługujemy się modelem systemu do odwzorowania analizowanych obszarów rzeczywistości w postaci zaawansowanych modeli matematycznych (np. logicznych), aby, po pierwsze, uchwycić najistotniejsze czynniki charakteryzujące badany obszar i rozwiązać związane z tym problemy, a po drugie — zmniejszyć koszty analiz. Częstość prowadzenia eksperymentów jest z przyczyn prawnych, etycznych, technologicznych lub organizacyjnych w ogóle niemożliwe. Wtedy buduje się model, starając się wyodrębnić elementy najważniejsze z punktu widzenia badanego problemu.

Cel badań determinuje szczegółowość odzwierciedlenia w modelu funkcjonowania obiektu rzeczywistego. Model złożony z bardzo wielu czynników ma strukturę zanadto (w stosunku do celu) rozbudowaną, co utrudnia (czasem nawet uniemożliwia) znalezienie pożądanego rozwiązania. W przypadku stosowania analitycznych metod rozwiązywania problemów stanowi to poważne ograniczenie. Z kolei, tworząc model bardzo uproszczony, można uzyskać rezultaty o wiele odbiegające od rzeczywistych.

Poprzez skojarzenie badań przeprowadzanych na systemie rzeczywistym (eksperymentalnym) z metodą analityczną można metodą symulacyjną badać zachowania systemu. W celu uzyskania interesujących nas charakterystyk działania systemu postępuje się analogicznie, jak w przypadku badań na systemie rzeczywistym, z tym że model matematyczny w postaci programu komputerowego i rozszerzony o funkcję obserwacji procesów odtwarzanych podczas eksperymentów zastępuje system.

Należy zatem mieć na uwadze, że obserwuje się nie procesy rzeczywiste, lecz ich reprezentację symboliczną w komputerze, co może mieć istotny wpływ na zgodność uzyskanych ocen i wyznaczanych charakterystyk z ich wartościami rzeczywistymi.

### 2.1.1. Klasyfikacja modeli symulacyjnych

Klasyfikacja modeli i kryteria stosowane do ich podziału zależą od natury opisywanego zjawiska lub procesu, a także od sposobu opisu uzależnionego od języka i aparatu pojęciowego i symbolicznego, którym posługuje się konstruktor modeli.

Przynależność do danej klasy zależy od środków wykorzystanych do budowy modelu, przy uwzględnieniu sposobu odzwierciedlenia wybranych własności, procesów i związków zachodzących w modelowanym systemie oraz celu badań, któremu jest podporządkowany charakter poszukiwanych informacji. Według takich kryteriów modele można podzielić na cztery klasy.

1. **Modele materialne** (działające, rzeczywiste) mogą być utworzone, specjalnie w celu wykonania badań, z istniejących obiektów o określonym przeznaczeniu użytkowym, przy zachowaniu ich fizycznej tożsamości z oryginałem. Podczas funkcjonowania w wybranym segmencie własności procesów i związków generują one informacje poszukiwane przez badacza, a po zakończeniu badań mogą być nadal wykorzystywane zgodnie z ich przeznaczeniem.

2. **Modele idealne** nie mają tej samej, co badany system, natury fizycznej i nie są do niego podobne ani w sensie fizycznym, ani geometrycznym. Nazwa tych modeli nie wyraża ściśle ich charakteru i wynika z istniejącej tradycji. Jako szczególny rodzaj takiego modelu idealnego można wyróżnić model cybernetyczny. Jednak model cybernetyczny jest zbyt skomplikowany, aby był przedmiotem bezpośredniego poznania; może jednak stanowić podstawę do utworzenia innego, bardziej uproszczonego modelu idealnego.
3. **Modele sformalizowane** są reprezentacją modeli fizycznych na jeszcze wyższym poziomie abstrakcji. Taką reprezentację można utworzyć wtedy, gdy pojęcia występujące w modelu fizycznym dadzą się wyrazić za pomocą znaków i relacji matematycznych lub logicznych. Cechą modelu sformalizowanego jest zatem kompletny brak podobieństwa między elementami i relacjami, z których go zbudowano, a składem i strukturą modelowanego systemu. Model jest umowny, a nie poglądowy, i nie ma nic wspólnego z charakterem elementów i relacji tworzących modelowany system. Rozwój matematyki i fizyki przyczynił się do tego, że w naukach ścisłych i technicznych modele sformalizowane, zwane po prostu modelami matematycznymi, stanowią najbardziej reprezentatywną grupę modeli abstrakcyjnych. Są one zapisywane w postaci równań różniczkowych, całkowych, deterministycznych lub probabilistycznych. Modelowanie matematyczne pozwala wnikać w istotę badanych systemów i udostępnia szczegółowemu badaniu wiele własności, procesów i związków, które dotychczas wymykały się analizie [Wilk, Müller 1997].

W innym ujęciu model fizyczny jest repliką (wiernym odbiciem) rzeczywistości, natomiast model abstrakcyjny jest syntetycznym ujęciem najbardziej istotnych elementów rzeczywistości.

**Model statyczny** przedstawia rzeczywistość w jakiejś konkretnej chwili w oderwaniu od stosunków następstwa czasowego, natomiast model dynamiczny wykorzystuje czas jako podstawowy element opisu zjawiska w jego powiązaniu ze zdarzeniami występującymi wcześniej i później.

**Model deterministyczny** przedstawia wartość każdej zmiennej decyzyjnej za pomocą jednej wielkości, natomiast model probabilistyczny wskazuje na możliwe zakresy zmiennych w postaci rozkładów prawdopodobieństwa.

**Model normatywny** służy ocenie dostępnych kierunków działania i wskazaniu tego, który z nich w danej sytuacji jest najlepszy, czyli optymalny. Model opisowy (opisowy model werbalny) z kolei ogranicza się do przedstawienia pewnego fragmentu rzeczywistości bez próby jej oceny.

Wśród modeli analogowych można wyróżnić analogowe modele werbalne oraz analogowe modele fizyczne.

**Model symboliczny** wykorzystuje symbole matematyczne. Wśród matematycznych metod prognozowania i symulacji można wyróżnić:



- w zależności od sposobu uwzględnienia czynnika zmienności czasu — modele statyczne i dynamiczne;
- w zależności od metody zapisu badanych zmiennych — modele jedno- i wielorównaniowe;
- w zależności od charakteru funkcji odwzorowującej związek między zmienną zależną i zmiennymi niezależnymi — modele liniowe, liniowe linearyzowane (uboczne efekty linearyzacji) i nieliniowe.

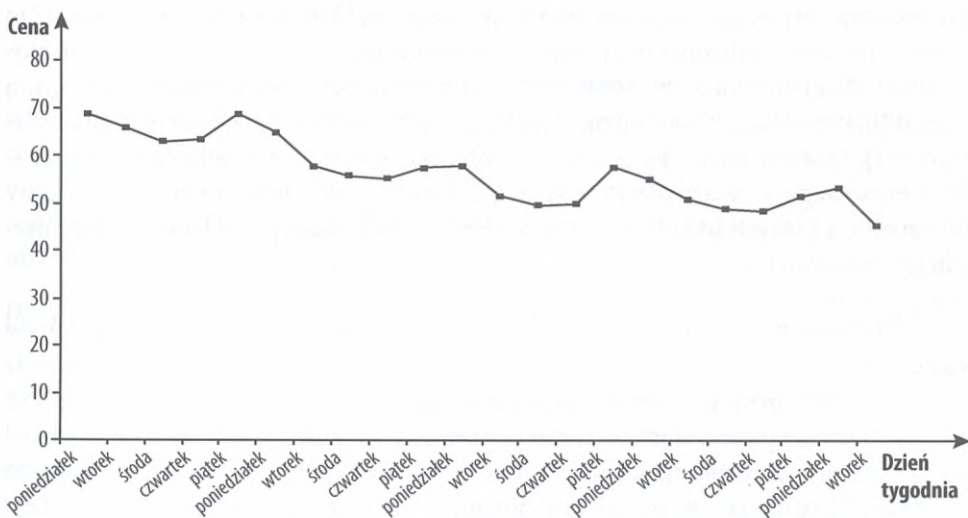
Przykładowo, model opisujący sezonowość zmian ceny pewnego dobra z trendem liniowym, która na giełdzie rolnej kształtowała się jak przedstawia to rysunek 2.1, może przyjąć postać:

$$y_t = \alpha_0 + \alpha_1 t + \alpha_2 x_{1t} + \alpha_3 x_{2t} + \alpha_4 x_{3t} + \alpha_5 x_{4t} + \varepsilon_t, \quad [2.1]$$

gdzie:

- $t$  — zmienna czasowa — numer obserwacji (dnia),
- $x_{1t}$  — zmienna zero-jedynkowa równa 1 w poniedziałki,
- $x_{2t}$  — zmienna zero-jedynkowa równa 1 we wtorki,
- $x_{3t}$  — zmienna zero-jedynkowa równa 1 w środy,
- $x_{4t}$  — zmienna zero-jedynkowa równa 1 w czwartki.

Rysunek 2.1. Model opisujący sezonowość cen z trendem czasowym



Źródło: Opracowanie własne na podstawie [http://oizet.p.lodz.pl/ania/ekonometria/W1\\_Pojecia\\_ekonometrii.doc#dochód\\_n](http://oizet.p.lodz.pl/ania/ekonometria/W1_Pojecia_ekonometrii.doc#dochód_n).

Otrzymany model jest w postaci:

$$\hat{y}_t = 70,31 - 0,94t - 0,28x_{1t} - 5,06 - 6,41x_{3t} - 5,67x_{4t}$$

Współczynniki sezonowe interpretuje się w odniesieniu do poziomu właściwego dla piątku. W prognozowaniu szeregów czasowych wykorzystuje się modele adaptacyjne. Sporządza się punktowe i przedziałowe prognozy na podstawie różnych modeli analitycznych, np. modeli przedziałowych. Prognozy są także opracowywane na podstawie modeli dynamicznych (autoregresji i średniej ruchomej), a metodami stacjonaryzacji szeregu czasowego (transformacje stacjonaryzujące) — modele: ARMA, ARMAX i ARiMAX.

Dla szeregów, gdzie nieregularności przeszkadzają w uchwyceniu składowej systematycznej, stosuje się uśrednianie. W ten sposób eliminowane są przypadkowe wahania niemające większego wpływu na ogólny trend w szeregu. Uśrednianie przeprowadza się względem wszystkich wcześniejszych okresów — elementów szeregu czasowego (prognoza wyznaczana jest na jeden okres w przyszłość).

$$y_{t+1}^* = (1/t) \cdot (y_1 + y_2 + \dots + y_t), \quad [2.2]$$

gdzie:

- $y_{t+1}^*$  — prognozowana wartość na okres  $t + 1$ ,  
 $y_t$  — obserwowana wartość w momencie  $t$ .

Model średniej ruchomej ważonej i jego innowacyjność polega na rozróżnieniu stopnia wpływu wartości szeregu na prognozę ze względu na czasową lokalizację danych. Chodzi o to, że im dłuższy czas mija od wydarzenia do momentu, na który sporządzana jest prognoza, tym mniejszy ma on wpływ na jej wynik. Siłę tego wpływu oblicza się dwoma metodami: za pomocą wag liniowych oraz wag harmonicznych. Wagi harmoniczne kładą większy nacisk na najnowsze wartości niż czynią to wagi liniowe. Wagami w tym przypadku są liczby rzeczywiste z przedziału otwartego  $(0, 1)$ , których suma daje 1. Ich wielkość maleje wraz ze zwiększaniem się rozstępu czasowego w analizowanym szeregu. Liczba wag uzależniona jest od liczby momentów, z których układana jest prognoza na okres następny. Ogólna postać modelu jest następująca:

$$y_{t+1} = w_1 \cdot y_{t+1-k} + w_2 \cdot y_{t+2-k} + \dots + w_k \cdot y_t, \quad [2.3]$$

gdzie:

- $y_{t+1}$  — prognozowana wartość na okres  $t + 1$ ,  
 $y_i$  — obserwowana wartość w momencie  $i$ ,  
 $w_i$  — waga uzależniona od  $i$ ,  
 $k$  — liczba wyrazów średniej ruchomej,

wagi liniowe ustalane są jako:

wagi harmoniczne ustalane są jako:

$$w_j = (2 \cdot j) / [k \cdot (k + 1)],$$

$$w_j = w_{j-1} + 1 / [k \cdot (k + 1 - j)].$$

W modelach autoregresji wartość obecna  $y_t$  zmiennej prognozowanej określana jest funkcją od wartości tej samej zmiennej z okresów poprzednich oraz czynnika losowego.

$$y_t = f(y_{t-1}, y_{t-2}, \dots, \varepsilon). \quad [2.4]$$

Tego typu modele często są używane do prognozowania zjawisk gospodarczych. Stosując modele autoregresyjne, napotyka się dwojakiemu typu trudności:

- 1) określenie rzędu autoregresji (liczby zmiennych opóźnionych) wchodzących do modelu;
- 2) szacowanie parametrów modelu autoregresyjnego; zmienne opóźnione są zazwyczaj silnie skorelowane ze zmienną objaśnianą i stosowanie metody najmniejszych kwadratów często nie może być brane pod uwagę.

Analiza faktów w ujęciu dynamicznym związana jest z tworzeniem zbioru obserwacji następujących po sobie w czasie, co nazywamy szeregiem czasowym. Analizując go, rozpatrujemy szereg jako realizację procesu stochastycznego.

Rozróżniamy szereg czasowy ciągły, wtedy gdy zbiór ten jest ciągły, bądź dyskretny, gdy zbiór jest dyskretny. Ten drugi zaś pojawia się przy pobieraniu próbki ciągłego szeregu czasowego lub przy kumulowaniu zmiennej w jakimś okresie. Szereg czasowy jest deterministyczny, gdy pewna funkcja dokładnie określa przyszłe wartości. Szereg czasowy jest statystyczny lub niedeterministyczny, gdy przyszłe wartości da się zapisać jedynie w terminach rozkładu prawdopodobieństwa. Obserwacja szeregu czasowego dla prognozy jego przyszłej wartości może wyznaczać bazę celów: planowania w ekonomii i handlu, planowania produkcji, kontroli inwentaryzacyjnej i kontroli produkcji, sterowania i optymalizacji procesów przemysłowych.

Prognozy przygotowywane są z wyprzedzeniem na pewien okres, który zależy od rozpatrywanego zagadnienia. Przykładowo, w zagadnieniu prognozowania zbytu towarów na kolejne trzy miesiące można wykorzystać dane ze sprzedaży tych samych towarów z miesięcy poprzednich. Na ich podstawie możliwe jest przewidywanie wielkości zbytu w kolejnych miesiącach (oczywiście z określonym prawdopodobieństwem potwierdzenia się prognozy).

Funkcję pozwalającą ustalić prognozy w określonym momencie dla wszystkich wyprzedzeń nazywamy funkcją prognozy w momencie. Naszym zadaniem jest uzyskanie takiej funkcji prognozy, dla której wartości średnie kwadratu odchyłeń między wartością faktyczną a wartością prognozowaną są możliwie najmniejsze dla każdego wyprzedzenia. Musimy również określić dokładność prognozy. Można ją przedstawić za pomocą przedziałowej prognozy przez ustalenie wartości krytycznych prawdopodobieństwa realizacji prognoz. Potrzebujemy jej m.in. do obliczania ryzyka związanego z decyzjami opartymi na danych prognozach. Szerzej o tych zagadnieniach wypowiadają się m.in. tacy autorzy, jak: M. Cieślak [2004], J.M. Gajda [2001], R. Tadeusiewicz [1993].

## 2.1.2. Istota procesu decyzyjnego — modele i symulacje

**Symulacja komputerowa** jest działaniem, w trakcie którego inicjowany jest wirtualny model analizowanego lub projektowanego systemu. Eksperymenty na tym modelu symulują operacje i ich wyniki, które częstokroć są niewykonalne na rzeczywistych realizacjach systemu. W języku modelowania przez model symulacyjny rozumie się model matematyczny, którym można manipulować w celu oceny skutków możliwych działań bez potrzeby uciekania się do prób rzeczywistych. Posługując się językiem prakseologicznej teorii organizacji, można powiedzieć, że model symulacyjny pozwala na imanentyzację [Kotarbiński 2003]. Model symulacyjny można również nazywać modelem proceduralnym, ponieważ manipulacja modelem polega na stosowaniu procedury opisującej relacje między zmiennymi decyzyjnymi. Symulacyjna metoda badania systemu polega na skojarzeniu metod analitycznych z badaniami eksperymentalnymi. Postępowanie prowadzące do uzyskania interesujących nas charakterystyk działania systemu jest analogiczne jak dla badań eksperymentalnych, z tym że model matematyczny poszerzony o funkcje obserwacji procesów odtwarzanych podczas eksperymentów symulacyjnych zastępuje system.

Należy uwzględnić fakt, że obserwujemy nie procesy rzeczywiste, lecz ich symboliczną reprezentację w komputerze. Może to mieć merytoryczne znaczenie w zakresie zgodności uzyskanych ocen wyznaczanych charakterystyk z ich wartościami rzeczywistymi.

Przewidywalność zdarzeń wyznacza podział na symulacje stochastyczne, korzystające z generatora liczb pseudolosowych lub (bardzo rzadko) losowych (szczególnie popularna jest metoda Monte Carlo), oraz na deterministyczne, w których wynik jest powtarzalny i zależy tylko od danych wejściowych i ewentualnych interakcji ze światem zewnętrznym.

Kryterium sposobu upływu czasu pozwala wyodrębnić:

- symulacje z czasem ciągłym, w których czas przyrasta o stałe okresy;
- symulacje z czasem dyskretnym, lecz wartości próbek sygnałów są interpolowane dla chwil pośrednich między momentami odczytu;
- symulacje z czasem dyskretnym, w których czas przyrasta o stałe okresy, a krok czasowy dobiera się optymalnie ze względu na bardzo wysokie zapotrzebowanie na zasoby systemu, jego wydajność i charakter symulowanego obiektu lub zjawiska (mikrosekundy w obwodach elektrycznych i miliony lat przy symulacji ewolucji gwiazd);
- symulacje zdarzeń dyskretnych, w których czas zwiększa się skokowo, ale jego przyrosty są zmienne (ważniejsza jest sekwencja zdarzeń niż rzeczywisty lub wirtualny upływ czasu).

Rzeczywisty czas ciągły możliwy jest do uzyskania jedynie w symulatorach analogowych, np. w komputerach analogowych, ale nie stosuje się ich w SWD.

Symulacje można też sklasyfikować ze względu na formę danych wyjściowych jako: statyczne, kiedy wynikiem jest zbiór danych (statyczny obraz), dynamiczne, kiedy wynikiem jest proces przebiegający w czasie (np. animacja), oraz interaktywne, reagujące na sygnały ze świata zewnętrznego (np. od operatora).

Liczba użytych komputerów jest wyznacznikiem podziału na symulacje lokalne (przetwarzanie odbywa się na pojedynczym komputerze) oraz rozproszone (przetwarzanie odbywa się na wielu komputerach połączonych w sieć LAN lub WAN).

Wśród narzędzi stosowanych do symulacji można wyróżnić m.in.: język programowania GPSS, pakiet oprogramowania Crystal Ball oraz Simulink.

Symulację komputerową wykorzystuje się w biznesie do analizy systemów kolejkowych w wielu dziedzinach, np. do rozwiązywania zagadnień związanych z kolejkami w supermarketach, stacjach obsługi samochodów, na lotniskach. Mogą być one także przydatnym narzędziem rozwiązywania problemów logistycznych w systemach produkcyjnych, gdy trzeba na czas dostarczyć, np. surowce lub półfabrykaty do procesu wytwórczego albo gotowe wyroby do klienta. Inni autorzy wykorzystują symulacje jako skuteczne wsparcie serwisów aukcyjnych, jak proponują to m.in. P. Wołoszyn i J. Wołoszyn [2011] w opisie metody doboru par sprzedawca–kupujący w celu zrównoważenia ofert w symulacyjnym modelu serwisu aukcyjnego.

Ciekawym polem współczesnych zastosowań metod symulacji komputerowych do podejmowania decyzji są eksperymenty z wykorzystaniem systemów multiagentowych do modelowania cech dynamicznych społeczności ludzkich, co oczywiście ma zastosowanie chociażby w marketingu [Wołoszyn 2010].

Szerzej zagadnienia zastosowań metod statystycznych i ekonometrycznych w zarządzaniu, np. ekonomicznych aspektów dynamicznego wyboru konsumenta, modelowania zapasów dla potrzeb wspomagania decyzji, są podejmowane w pracach W. Sadowskiego i zespołu [1985] czy M. Lipiec-Zajchowskiej i współautorów [2003]. B. Guzik [2005] stosuje statystyczne i ekonometryczne metody szacowania atrakcyjności lokalizacji mieszkań m.in. do szacowania atrakcyjności lokalizacji nieruchomości i ich cen.

Komputerowe gry symulacyjne mają za zadanie naśladować wrażenia różnych doznań, jakich doświadczamy w codziennej rzeczywistości, a także takich, które są wytworem naszej wyobraźni. Profesjonalne symulatory kabinowe są wykorzystywane do szkoleń jednostek specjalnych i pilotów.

W naukach przyrodniczych wykorzystuje się symulacje komputerowe do generowania prognoz pogody — przewidywania czasowych i przestrzennych zmian stanu atmosfery. Związane z nimi takie wielkości, jak: zachmurzenie, zamglenie, stan morza, zanieczyszczenia atmosferyczne są także elementami prognozy pogody i często zależą od prognozy podstawowych elementów. Prognoza pogody zależy od liczby i jakości danych początkowych.

W pracach inżynierskich symulowane są obciążenia konstrukcji budowlanych, np. mostów, wiaduktów, i ich odwzorowanie w sposób graficzny. Niekiedy w proces symulacji komputerowej włącza się rzeczywiste zjawiska fizyczne, np. prowadzi się

próby wytrzymałości konstrukcji mostu, monitorując za pomocą komputerów obciążenia dynamiczne i statyczne wywołane przejazdem czołgów.

### 2.1.3. Modele optymalizacyjne w procesie decyzyjnym

**Modele optymalizacyjne** w procesie decyzyjnym opisują problem decyzyjny w danej dziedzinie i są projektowane do maksymalizacji funkcji użyteczności przy spełnieniu określonych ograniczeń.

Celem użycia modeli optymalizacyjnych jest poszukiwanie rozwiązań optymalnych. Możliwe są w tym postępowaniu dwa podejścia:

- 1) charakterystyka systemu jest przybliżana za pomocą modelu regresji, wykreowanego na podstawie badań symulacyjnych; rozwiązanie optymalne generowane jest na podstawie zdefiniowanego w ten sposób modelu stanowiącego funkcję kryterium zadania optymalizacyjnego; ekstremum funkcji przy uwzględnieniu ograniczeń zadania optymalizacyjnego;
- 2) na podstawie wyników badań symulacyjnych bezpośrednio wyznaczana jest wartość rozwiązania optymalnego — bez określania modelu regresji; priorytetem jest ustalenie adekwatności modelu symulacyjnego do apriorycznego funkcjonowania systemu oraz przeprowadzenie obserwacji (potwierdzenie hipotez), które są podstawą do wypowiedzania się o istocie oraz jakości funkcjonowania badanego systemu niezależnie od przyjętego celu badania symulacyjnego (wyznaczenie charakterystyk działania systemu czy też rozwiązanie sformułowanego zadania optymalizacyjnego).

Adekwatność modelu symulacyjnego jest pojęciem oznaczającym zgodność dwóch procesów — symulowanego i badanego. Aby ocenić tę adekwatność, należy wyznaczyć regułę decyzyjną, która na podstawie wyników badań eksperymentalnych przeprowadzonych na danym systemie i rozwiązań modelu symulacyjnego pozwala stwierdzić, że:

- nie ma podstaw do odrzucenia hipotezy o adekwatności modelu symulacyjnego do procesu zachodzącego w badanym systemie;
- uzyskane wyniki zaprzeczają prawdziwości hipotezy o adekwatności modelu symulacyjnego.

Rozumienie istoty metody symulacyjnej oraz znajomość mechanizmów funkcjonowania systemów na niej opartych, a także umiejętności modelowania matematycznego, programowania, głębokiej wiedzy z zakresu statystyki matematycznej, a w szczególności wyznaczania modeli regresji i planowania eksperymentów decyduje o powodzeniu badaczy w praktycznym stosowaniu tej metody do badania funkcjonowania systemów. Ich kompleksowa znajomość pozwala uzyskiwać dokładniejsze oceny wyznaczanych charakterystyk oraz osiągać pożądane wyniki, minimalizując jednocześnie nakłady na ich uzyskanie. W. Chmielarz [2005]

m.in. zastosował analizę możliwości migracji systemu opartego na bazie danych w kierunku systemu optymalizującego sterowanie produkcją, a następnie systemu zarządzania wiedzą. Na podstawie założeń budowy systemu bazodanowego, budowy modelu conceptualnego oraz rzeczywistego autor wykreował model optymalizacji sterowania produkcją oraz wskazał jego miejsce i rolę w systemie wraz z mechanizmami integracji. Zaprezentowano w pracy także koncepcję mechanizmu transformacji informacji zawartych w modelu na wiedzę przechowywaną w systemie oraz mechanizm porównań rzeczywistych wariantów rozwoju z przechowywanymi wzorcami.

Jednym z przydatnych narzędzi stosowania metod optymalizacji w procesie decyzyjnym jest program nieliniowej optymalizacji GRG2, który jest algorytmem opartym na metodzie uogólnionego gradientu zredukowanego. Optymalizację przeprowadza się w dwóch fazach. Jeżeli pierwszy punkt wprowadzony przez użytkownika nie spełnia wszystkich  $g_i$  ograniczeń, następuje uruchomienie optymalizacji I fazy. Funkcja celu fazy I jest sumą zniekształceń (naruszeń) ograniczeń oraz opcjonalnie uwzględnia część celu. Ta optymalizacja kończy się w przypadku, kiedy jest wysyłany komunikat, iż rozwiązanie problemu jest niemożliwe (nie dopuszczalne), lub też wysyłany jest komunikat o znalezieniu możliwych (dopuszczalnych) rozwiązań. Należy zwrócić uwagę na to, że jeśli wysyłany jest komunikat o niedopuszczalnych rozwiązaniach, procedura poszukiwania rozwiązania programu może zostać zatrzymana w lokalnym minimum celu fazy I (lub została uwzględniona zbyt duża „część realnego celu”), a problem może faktycznie mieć możliwe (dopuszczalne) rozwiązania. Jeśli podejrzewamy, że tak jest, to należy wybrać inne punkty wyjścia (lub zmniejszyć udział realnych celów) i spróbować uruchomić procedury poszukiwania rozwiązań ponownie. Faza II rozpoczyna się od uwzględnienia rozwiązania dopuszczalnego, znalezionego w fazie I lub przez użytkownika i podjęcia próby optymalizacji funkcji celu określonego przez użytkownika. W fazie II powstaje sekwencja dopuszczalnych punktów (wartości), których wartość nigdy nie ulega pogorszeniu. Istnieją dwa sposoby wykorzystywania GRG2 — przez interfejs pliku lub przez oddzielny podprogram [Lasdon, Warren, Jain, Ratner 1978].

Rozważmy przykład optymalizacji wyposażenia laboratorium komputerowego, prowadzonej za pomocą programu nieliniowej optymalizacji GRG2, przedstawionego w dodatku MS Excel Solver. Zakładamy, że uzyskaliśmy z projektu kwotę na sfinansowanie wyposażenia laboratorium komputerowego w wysokości 500 000 zł. Należy wydatkować wszystkie środki finansowe. Minimalna cena zakupu jednostki centralnej nie może być niższa niż 2000 zł, drukarki 800 zł i monitora 1000 zł. Wartości wyrażające liczbę zakupionych komputerów, drukarek i monitorów należą do zbioru liczb całkowitych i muszą spełniać warunek  $\geq 1$ ; jedna drukarka sieciowa może obsłużyć nie więcej niż 20 komputerów. Wyniki modelowania za pomocą programu GRG2 przedstawiono w tablicy 2.1 oraz na rysunku 2.2.

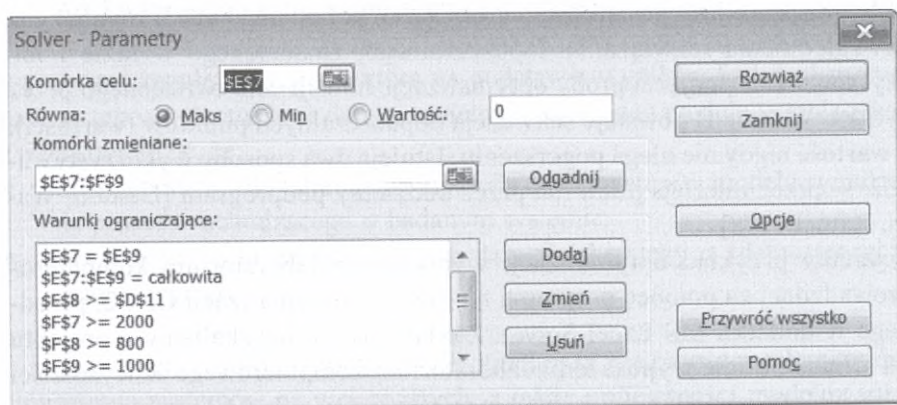
**Tablica. 2.1. Założenia i wyniki modelu optymalizującego wyposażenie laboratorium komputerowego**

Towary	Liczba	Cena (zł)	Wartość (zł)
Komputery	163	2 001,83	326 298,52
Drukarki	13	800,14	10 401,82
Monitory	163	1 001,84	163 299,66
		<b>Kwota</b>	500 000,00
		<b>Projekt</b>	500 000,00
		<b>Reszta</b>	0,00

Źródło: Opracowanie własne.

Przy uwzględnieniu wszystkich wyspecyfikowanych założeń wyposażenia laboratorium komputerowego i wykorzystaniu wszystkich środków z projektu (pp. wartość = 0) w oknie dialogowym Solvera uzyskaliśmy informację, że należy zakupić 163 komputery i monitory oraz 13 drukarek sieciowych za łączną kwotę 500 000 zł, przy cenie 2001,83 zł za jeden komputer, 800,14 zł za jedną drukarkę i 1001,84 zł za jeden monitor (tablica 2.1 oraz rysunek 2.2).

**Rysunek 2.2. Okno dialogowe dodatku MS Excel Solver — modelu optymalizującego wyposażenie laboratorium komputerowego, wykorzystującego program nieliniowej optymalizacji Generalized Reduced Gradient**



Źródło: Opracowanie własne.

Innym z ważnych narzędzi optymalizacji jest metoda simpleks z ograniczeniami na zmienne o nazwie B & B. W rozpatrywanej metodzie przyjęto następujący tok postępowania: najpierw porównuje się bieżący stan najlepszego rozwiązania z obecnym stanem rozwiązania danego problemu w określonym węzle drzewa roz-



wiązań i na tej podstawie podejmuje się decyzję, czy dalsze poszukiwanie rozwiązań na danej gałęzi ma sens.

Przykład zastosowania metod optymalizacyjnych do rozwiązania problemu komiwojażera dobrze ilustruje istotę charakteryzowanych metod. Rozważamy trzy punkty  $A$ ,  $B$ ,  $C$ . Komiwojażer musi objechać te 3 punkty. Jaką kolejność powinien obrać, aby straty wynikające z objazdu były minimalne? Dana jest macierz kosztów w postaci:

$$\mathbf{M} = \begin{matrix} & \begin{matrix} A & B & C \end{matrix} \\ \begin{matrix} A \\ B \\ C \end{matrix} & \begin{bmatrix} 0 & 11 & 10 \\ 13 & 0 & 12 \\ 20 & 7 & 0 \end{bmatrix} \end{matrix}.$$

Na przykład, liczba 13 oznacza, że koszt przejazdu z  $C$  do  $A$  wynosi 13 jednostek. Macierz  $\mathbf{M}$  nie jest macierzą symetryczną, tzn. np. koszt z  $A$  do  $B$  wynosi 9, natomiast z  $B$  do  $A$  wynosi 11 jednostek. Zadanie komiwojażera z macierzą symetryczną jest szczególnym przypadkiem ogólnego zadania komiwojażera. W analizowanym przykładzie możliwych jest 6 rozwiązań dopuszczalnych. Rozwiązanie optymalne jest jednym z rozwiązań dopuszczalnych. Wyznamy teraz rozwiązania dopuszczalne:

$$AB, BC, CA \text{ koszt} = 9 + 12 + 13 = 34,$$

$$AC, CB, BA \text{ koszt} = 10 + 7 + 11 = 28,$$

$$BC, CA, AB \text{ koszt} = 12 + 13 + 9 = 34,$$

$$BA, AC, CB \text{ koszt} = 11 + 10 + 7 = 28,$$

$$CA, AB, BC \text{ koszt} = 13 + 9 + 12 = 34,$$

$$CB, BA, AC \text{ koszt} = 7 + 11 + 10 = 28.$$

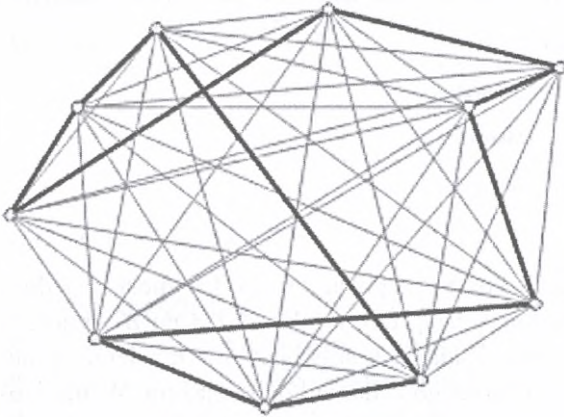
Można wnioskować z przykładu, iż minimalny koszt generują rozwiązania 2, 4 i 6. Wartość minimalnej straty wynosi 28 jednostek.

Na rysunku 2.2 przedstawiono grafowy model sieci miast rozważany w typowym problemie komiwojażera. Wagi krawędzi grafu mogą odpowiadać odległościom między miastami (czasowi podróży lub kosztom przejazdu – zależy, co chcemy w podróży komiwojażera zminimalizować). Trasa komiwojażera jest cyklem Hamiltona przechodzącym przez każdy wierzchołek grafu dokładnie jeden raz.

Znalezienie właściwego cyklu Hamiltona jest zadaniem trudnym obliczeniowo. Wyobraźmy sobie graf pełny o 10 wierzchołkach (graf, w którym każdy wierzchołek jest połączony z każdym). Ile różnych cykli Hamiltona zawiera taki graf? Pierwszą krawędź cyklu można wybrać na 9 różnych sposobów, ponieważ każdy wierzchołek grafu jest połączony krawędzią z pozostałymi dziewięcioma. Po wyborze pierwszej krawędzi pozostaje nam 8 możliwych wyborów, później 7, 6, 5, ..., 1. W efekcie otrzymujemy liczbę cykli Hamiltona  $L_H = 9 \cdot 8 \cdot \dots \cdot 1 = 362\,880$ , a dla  $n$  wierzchołków

$L_H = (n - 1)!$ . Wynik ten jest bardzo niekorzystny, ponieważ prowadzi do wykładniczej klasy złożoności obliczeniowej  $O(n!)$ .

**Rysunek 2.3. Przykład grafu pełnego z zaznaczonym cyklem Hamiltona**



Źródło: Opracowanie własne na podstawie: [West 2001].

Dla każdego znalezionej cyklu Hamiltona obliczamy sumę wag krawędzi i zapamiętujemy cykl o najmniejszej sumie wag. Na przykład, w naszym grafie (rysunek 2.3) może to być cykl zaznaczony pogrubionymi krawędziami (nie sugerujemy się długością krawędzi na rysunku – ważne są wagi).

Rozważmy np. realistyczny graf posiadający 100 wierzchołków. Załóżmy, iż każdy wierzchołek łączy się z czterema innymi wierzchołkami grafu (stopień grafu wynosi 4). Liczba cykli Hamiltona do przebadania w rozważanym przypadku będzie wynosić:

$$L_H = 4_{1 \text{ wyb.}} \cdot 3_{2 \text{ wyb.}} \cdot 3_{3 \text{ wyb.}} \cdot \dots \cdot 3_{97 \text{ wyb.}} \cdot 2_{98 \text{ wyb.}} \cdot 1_{99 \text{ wyb.}} \cdot 1_{100 \text{ wyb.}} = 5,09 \cdot 10^{46}$$

lub w przybliżeniu  $L_H \approx (s - 1)^n$ ,

gdzie:

$s$  — stopień grafu,

$n$  — liczba wierzchołków.

Liczba cykli do przebadania rośnie wykładniczo w stosunku do liczby wierzchołków. Nawet na bardzo szybkim komputerze wyznaczenie rozwiązania może przekroczyć porównanie z wiekiem wszechświata. Problemy algorytmiczne o złożoności wykładniczej są traktowane jako nierozwiązywalne dla dużych  $n$ .

Oto opis algorytmu postępowania według metodyki PCAM na przykładzie działania obliczeniowych algorytmów równoległych wykorzystywanych w procesorach

rach komputerów. Rozpatrywana jest funkcjonalna dekompozycja zadań. W metodzie B & B zakładamy drobnoziarnisty podział zadań, przyjmując jako element stanowiący pojedyncze zadanie węzeł drzewa. Na zadanie składa się wtedy eksploracja wszystkich rozgałęzień wychodzących z danego węzła. Nowe zadania są tworzone z listy węzłów, która powstaje podczas przeszukiwania drzewa wszerz. Kolejnym problemem jest także zapewnienie posiadania przez dane zadanie aktualnej wartości najlepszego rozwiązania. Może to zostać rozwiązane poprzez stworzenie pojedynczego zadania, które będzie przechowywać najlepszą znaną do tej pory wartość i na żądanie zadań obliczeniowych udostępniać ją.

Lista zagadnień do sprawdzenia po etapie podziału:

1. Czy podział definiuje przynajmniej o rząd wielkości więcej zadań niż liczba procesorów w komputerze docelowym?  
Liczba zadań zależy od przyjętego poziomu zagłębienia poprzez przeszukiwania drzewa wszerz. W zależności od liczby przedmiotów, które mamy do wyboru, liczba wszystkich węzłów w drzewie wynosi  $n!$ , co nawet przy drugim warunku co do zagłębienia (suma mas) daje dużą możliwość wyboru liczby zadań.
2. Czy podział unika zbędnych obliczeń i zajmowanego miejsca?  
W tym przypadku problemem może być posiadanie aktualnej informacji o najlepszym rozwiązaniu, w wyniku czego może się okazać, że część utworzonych zadań już na starcie zostaje zakończona, ponieważ ich wynik jest gorszy od najlepszego znanego do tej pory.
3. Czy zadania są porównywalnej wielkości?  
Wielkość danego zadania obliczeniowego zależy od tego, czy przyjmiemy strategię rozdzielania zadań do pewnej głębokości, a potem obliczeń będziemy dokonywać sekwencyjnie (np. algorytmem przeszukiwania drzewa w głąb), czy też istnienie pojedynczego zadania w każdym węźle drzewa. W drugim przypadku zadania są porównywalne, jednak narzut związany z rozpoczęciem zadania i komunikacją może się okazać niekorzystny w stosunku do samego obliczenia wewnątrz zadania.
4. Czy liczba zadań skaluje się z rozmiarem problemu?  
Tak, liczba zadań może maksymalnie przyjąć wartość silni od rozmiaru problemu (jeśli za rozmiar problemu przyjmiemy liczbę możliwych przedmiotów do załadowania).
5. Czy były rozważane inne przypadki podziału?  
Jak wspomniano w odpowiedzi na pytanie nr 3, można rozważyć co najmniej dwie metody podziału.

W metodzie B & B spotykamy się z dwoma rodzajami komunikatów: informacja o obliczonej do tej pory wartości w danym węźle, przenoszona do węzłów potomnych, oraz informacja o najlepszym znanym do tej pory rozwiązaniu (zarówno komunikat z informacją o wartości obecnej, jak i komunikat służący do zgłoszenia najlepszej odpowiedzi).

Jak wspomniano wcześniej, informacja o tej wartości mogłaby być przechowywana w specjalnym zadaniu. Natomiast kwestią wartą rozpatrzenia pozostaje częstotliwość odświeżania tej informacji przez węzły obliczeniowe. Taka komunikacja może się odbywać przed każdą instrukcją, która wykorzystuje tę wartość do podjęcia decyzji o kontynuacji obliczeń. Innym podejściem jest periodyczne rozsyłanie komunikatu o obecnej najlepszej wartości rozwiązania. Zmniejsza to liczbę komunikatów i powoduje wykonanie wielu zbędnych obliczeń, ponieważ mogą one zakładać, że wartość najlepszego rozwiązania jest gorsza od wyniku obecnego.

Lista zagadnień do sprawdzenia po etapie komunikacji:

1. Czy wszystkie zadania przeprowadzają podobną liczbę operacji komunikacji? Biorąc pod uwagę, że każde zadanie musi przekazać informację w przypadku odnalezienia rozwiązania lepszego od najlepszego obecnie oraz pytać o najlepsze istniejące rozwiązanie, liczba komunikatów w ramach pojedynczego zadania jest równa.
2. Czy każde zadanie komunikuje się tylko z małą liczbą swoich sąsiadów? Zadania komunikują się z jednym centralnym zadaniem.
3. Czy operacje komunikacji mogą przebiegać współbieżnie? Komunikacje między poszczególnymi zadaniami a zadaniem centralnym mogą przebiegać niezależnie.
4. Czy obliczenia związane z każdym zadaniem mogą przebiegać współbieżnie? Obliczenia w ramach każdego z węzłów mają naturę sekwencyjną. Jednak przyjmując, że od pewnej głębokości zadania sekwencyjne składają się nie z pojedynczego węzła, a wszystkich rozgałęzień wychodzących z danego węzła, zadanie obliczeniowe można wtedy próbować wykonać współbieżnie.

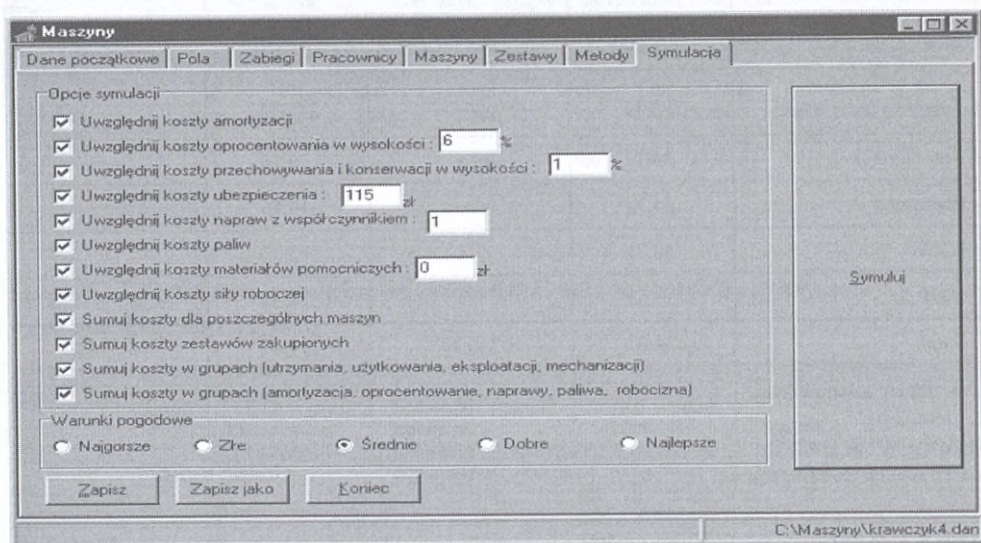
## 2.1.4. Rodzaje i przykłady eksperymentów symulacyjnych

Na rysunku 2.4 został przedstawiony interaktywny model interfejsu symulacji efektywności ekonomicznej operacji technologicznych produkcji roślinnej z możliwościami wyboru opcji ustalania kosztów.

Moduł raportowania wyników zawiera wiele alternatywnych opcji kosztowych w modelu przedsiębiorstwa, w którym użytkownik-przedsiębiorca rolny może dodatkowo prześledzić wpływ różnych parametrów decyzyjnych na uzyskane warianty rozwiązań w zależności, np. od prawdopodobieństwa wystąpienia skrajnie złych, średnio dobrych lub bardzo złych warunków pogodowych, wpływających bezpośrednio na dyspozycyjny czas wykonania określonych operacji technologicznych. W ten sposób rozwiązania modelowe wspomagają producentów w podejmowaniu właściwych decyzji logistyczno-transportowych w planowaniu operacyjnym zabiegów polowych produkcji roślinnej na podstawie generowanych kosztów oraz czasów dyspozycyjnych maszyn i urządzeń zmechanizowanego sprzętu rolniczego (np. gęste błoto uniemożliwia zbiór korzeni buraków cukrowych w warunkach opadów i potem trzeba wykonać

zbiór znacznie szybciej, bo dni do pierwszych przymrozków ubywa wraz z opóźnieniem zbioru — w rezultacie ze 120 godzin pozostaje 100).

**Rysunek 2.4. Interaktywny model symulacji efektywności ekonomicznej operacji technologicznych produkcji roślinnej z możliwościami wyboru opcji ustalania kosztów**



Źródło: Opracowanie własne.

Przedstawiony przykład działania interaktywnego modelu analizy ekonomiczno-finansowej dotyczy symulacji wpływu czynnika niezależnego od producenta, czyli stopy oprocentowania kapitału jako kosztu alternatywnego, na kosztocłonność zbioru korzeni buraka cukrowego w wybranym przedsiębiorstwie rolnym (tablica 2.2).

W przypadku nieuwzględnienia kosztów utraconych korzyści wynikających z tytułu niezainwestowania pieniędzy przez przedsiębiorcę rolnego w produkty bankowe zamiast w sprzęt do zbioru buraków, koszty zbioru korzeni buraków ulegają zmniejszeniu łącznie aż o 16,43%. Rozwiązanie takie daje lepszy ogłęd sytuacji zarządzającemu przedsiębiorstwem rolnym, jakie są skutki alternatywnego zainwestowania kapitału w działalność pozarolniczą w warunkach gospodarki rynkowej i wolności wyboru.

Symulację zmian wartości sprzedaży towarów pod wpływem zmian wolumenu sprzedaży oraz zmian przelicznika walutowego euro przedstawia rysunek 2.5. Wartość płac brutto zmienia się również adekwatnie do liczby i wartości sprzedanych towarów.

Przedstawione wyniki symulacji (rysunki 2.5– 2.7) wskazują, że w procesie planowania menedżerowie mogą przewidywać różne scenariusze sytuacji w zależności

**Tablica 2.2. Wpływ kosztów oprocentowania kapitału na koszty eksploatacji zestawu do zbioru buraków cukrowych**

Maszyna	Koszty eksploatacji			
	z oprocentowaniem kapitału		bez oprocentowania kapitału	
	jednostkowe (zł·h-1)	jednostkowe zestawu (zł·ha-1)	jednostkowe (zł·h-1)	jednostkowe zestawu (zł·ha-1)
Kombajn buraczany	293,33		245,33	
Przyczepa_1	24,24		19,16	
Przyczepa_2	24,24		19,16	
C360	50,71		43,21	
C330	31,55		27,54	
Suma	424,07	4 240,70	354,40	3 544,00
% udział kosztów eksploatacji	Suma = 100	Suma = 100	Suma = 83,57	Suma = 83,57

Źródło: Opracowanie własne.

od zmian niezależnych od nich czynników, np. cena towaru, lub częściowo niezależnych od inwencji sprzedawców, np. wolumen sprzedanego towaru.

Poprzez wskaźniki oceny jakościowej sprzedawców i stosowną premię za jakość obsługi, również generowaną losowo, można kształtować płace brutto pracownika za miesiąc, działając w ten sposób promocyjnie i zwiększając jakość obsługi klientów.

**Rysunek 2.5. Zmiany wartości plac brutto w zależności od liczby i wartości sprzedanych towarów**

	G	H	I	J	K	U	P	AA
		Cena rynkowa	Sprzedaz losowa za jeden dzień (kg/szt. - ilość)		Sprzedaz za jeden dzień	Wartość sprzedaży za okres (PLN)	Wartość sprzedaży za okres (EUR)	Placa brutto za miesiąc
1 Towary		Cena zł (kg/s symulowana (kg/szt.)						
2 Wieprzowina	14	=LOS()*(H2+0,2)-(H2-0=LOS()*(K2+K2*0,1)-(I 1000				=I2*J2*U2	=O2/+ziemiaki!Y\$9	=Y2+Z2
3 Sery	21	=LOS()*(H3+0,2)-(H3-0=LOS()*(K3+K3*0,1)-(I 250				=I3*J3*U3	=O3/+ziemiaki!Y\$9	=Y3+Z3
4 Wołowina	18	=LOS()*(H4+0,2)-(H4-0=LOS()*(K4+K4*0,1)-(I 150				=I4*J4*U4	=O4/+ziemiaki!Y\$9	=Y4+Z4
5 Masło	4	=LOS()*(H5+0,2)-(H5-0=LOS()*(K5+K5*0,1)-(I 3000				=I5*J5*U5	=O5/+ziemiaki!Y\$9	=Y5+Z5
6 Ogórki	2	=LOS()*(H6+0,2)-(H6-0=LOS()*(K6+K6*0,1)-(I 300				=I6*J6*U6	=O6/+ziemiaki!Y\$9	=Y6+Z6
7 Pomidory	5	=LOS()*(H7+0,2)-(H7-0=LOS()*(K7+K7*0,1)-(I 500				=I7*J7*U7	=O7/+ziemiaki!Y\$9	=Y7+Z7

Źródło: Opracowanie własne.

**Rysunek 2.6. Wyniki zastosowania losowej zmiany liczby i wartości sprzedaży towarów**  
— rezultat pierwszy (los() = 0,8133)

	G	H	I	J	K	O	P	AA
			Cena rynkowa symulowana	Sprzedaż losowa za jeden dzień	Sprzedaż za jeden dzień	Wartość sprzedaży za okres (PLN)	Wartość sprzedaży za okres (EUR)	Placa brutto za miesiąc
1 Towary	Cena zł	(kg/szt.)	(kg/szt.)	(kg/szt.-ilość)				
2 Wieprzowina	14		14,13 zł	1 063	1000	20 474 202,72 zł	€4 600 944,43	22 738,25 zł
3 Sery	21		21,13 zł	266	250	7 413 793,00 zł	€1 666 020,90	8 501,63 zł
4 Wołowina	18		18,13 zł	159	150	3 790 575,14 zł	€851 814,64	5 194,64 zł
5 Masło	4		4,13 zł	3 188	3000	18 175 286,84 zł	€4 084 334,12	25 508,85 zł
6 Ogórki	2		2,13 zł	319	300	882 844,74 zł	€198 392,08	897,20 zł
7 Pomidory	5		5,13 zł	531	500	3 905 120,82 zł	€877 555,24	3 855,37 zł

Źródło: Opracowanie własne.

**Rysunek 2.7. Wyniki zastosowania losowej zmiany ilości i wartości sprzedaży towarów**  
— rezultat drugi (los() = 0,5430)

	G	H	I	J	K	O	P	AA
			Cena rynkowa symulowana	Sprzedaż losowa za jeden dzień	Sprzedaż za jeden dzień	Wartość sprzedaży za okres (PLN)	Wartość sprzedaży za okres (EUR)	Placa brutto za miesiąc
1 Towary	Cena zł	(kg/szt.)	(kg/szt.)	(kg/szt.-ilość)				
2 Wieprzowina	14		14,02 zł	1 009	1000	19 283 888,16 zł	€4 333 458,01	25 419,35 zł
3 Sery	21		20,80 zł	225	250	6 182 280,00 zł	€1 389 276,40	9 077,10 zł
4 Wołowina	18		17,80 zł	135	150	3 152 736,00 zł	€708 480,00	4 320,54 zł
5 Masło	4		3,80 zł	2 700	3000	14 179 320,00 zł	€3 186 364,04	19 900,61 zł
6 Ogórki	2		1,80 zł	270	300	633 258,00 zł	€142 305,17	102,29 zł
7 Pomidory	5		4,80 zł	450	500	3 097 440,00 zł	€696 053,93	3 057,98 zł

Źródło: Opracowanie własne.

## 2.2. Zastosowania metod optymalizacyjnych

**Reguły i algorytmy wnioskowania** należą do teoretycznego kanonu SWD. W podrozdziale przedstawiono podstawowe rozwiązania z tego zakresu, a w szczególności: aksjomatykę wnioskowania, heurystyki oraz mechanizm modelowania doświadczenia.

W procesach wspomagania decyzji mamy do czynienia, z reguły, z problemami złożonymi i źle ustrukturyzowanymi. Dlatego reguły oraz algorytmy wnioskowania oparte na logicznych i statystycznych metodach optymalizacyjnych oraz na metodach heurystycznych, które mogą objąć jednocześnie wiele zmiennych i złożonych relacji między nimi oraz implementować doświadczenie ekspertów, mogą skutecznie wspomagać menedżerów w podejmowaniu właściwych decyzji, szczególnie strategicznych, obarczonych dużym ryzykiem związanym z trudnością w precyzyjnym przewidywaniu przyszłych stanów i ich wpływ na badane procesy, zjawiska oraz fakty.

### 2.2.1. Logiczne metody optymalizacyjne

**Logiczne metody optymalizacyjne** stanowią część podejścia nazywanego syntezą logiczną i tworzą proces wyszukiwania ekwiwalentnej reprezentacji specyficznego cyklu logicznego (obejmującego wiele wejść i jedno wyjście — wartość wynikową) przy spełnieniu jednego lub wielu wyspecyfikowanych ograniczeń. Generalnie cykl logiczny (obieg) jest ograniczony do minimalnego obszaru spełniającego specyfikowane z wyprzedzeniem opóźnienie.

W języku Prolog podstawę reprezentacji wiedzy stanowią klauzule — w takim przypadku jednak jest to zawsze tzw. klauzula Horna, czyli klauzula z co najwyżej jednym pozytywnym (niezanegowanym) literałem. Są one oczywistym wyborem stanowiącym podstawę dowodzenia twierdzeń i praktycznego programowania, ponieważ wnioskowanie na podstawie klauzul Horna jest relatywnie proste.

Stwierdzenie bez negacji to literał pozytywny:  $l = p$ , stwierdzenie zanegowane to literał negatywny:  $l = \neg p$ .

Klauzula to wyrażenie logiczne, będące alternatywą literałów:

$$k = l_1 \text{ or } l_2 \text{ or } \dots l_n. \quad [2.5]$$

Klauzulę nazywamy prawdziwą, jeżeli alternatywa jej literałów jest zawsze prawdziwa. Klauzula Horna to klauzula, w której co najwyżej jeden element jest niezanegowany. Przykładami klauzul Horna są następujące zapisy:

$$\{p, \neg r, \neg q\} \text{ i } \{\neg r, \neg q\}.$$

Klauzule Horna zapisuje się również w postaci implikacyjnej:  $p \leftarrow r \wedge q$ . Są one dogodną formą zapisu formuł do stosowania podanej reguły wnioskowania *modus ponens*. Do klauzul dowolnej postaci może być również wykorzystana zasada rezolucji [Robinson 1965].

Niestety, nie każdą formułę można przedstawić w postaci klauzuli. Najprostszym przykładem formuły, dla której nie jest to możliwe, jest koniunkcja dwóch literałów. Reguła *modus ponens* opisuje możliwą realizację pojedynczego kroku wnioskowania. Zwyczajowo regułę wnioskowania zapisuje się w następującej postaci:

$$\frac{\alpha_1, \alpha_2, \dots, \alpha_n}{\beta}, \quad [2.6]$$

gdzie:

$\alpha_1, \alpha_2, \dots, \alpha_n$  — wzorce formuł wejściowych reguły,  
 $\beta$  — wzorec formuły wynikowej.

Zastosowanie reguły wnioskowania polega na odnalezieniu w BW (połączonej ze zbiorem aksjomatów) formuł pasujących do wzorców formuł wejściowych i wygenerowaniu odpowiedniej formuły wynikowej. Oto przykłady prostych reguł wnio-



skowania, których sposób stosowania został opisany słownie, aby wyjaśnić istotę ich znaczenia.

W regule [2.7] wzorce formuł wejściowych pasują do dowolnych dwóch formuł, z których jedna jest implikacją, a druga — poprzednikiem tej implikacji. Reguła mówi, że mając dwie takie formuły, możemy wygenerować formułę będącą następnikiem tej implikacji:

$$\frac{\alpha \mid \rightarrow \beta, \alpha}{\beta} \quad [2.7]$$

Reguła *modus tollens* [2.8] może być zastosowana do dowolnych dwóch formuł, z których jedna jest implikacją, a druga — zanegowanym następnikiem tej implikacji. Wówczas reguła generuje formułę, będącą zanegowanym poprzednikiem implikacji:

$$\frac{\alpha \rightarrow \beta, \neg \beta}{\beta} \quad [2.8]$$

Rozumowanie to proces ukierunkowany na cel myślenia realistycznego, w którym ze zbioru faktów wyciąga się wnioski. Ta definicja została wykorzystana w metodach sztucznej inteligencji, których celem jest przynajmniej częściowe zastąpienie człowieka w tym procesie. Na rozumowanie składają się dwa zasadnicze procesy: proces indukcji (od szczegółu do ogółu) oraz proces dedukcji (od ogółu do szczegółu).

Arystoteles zdefiniował sylogizmy kategoryczne jako szczególną formę rozumowania dedukcyjnego. Składają się one z dwóch przesłanek i konkluzji. W skład sylogizmu kategorycznego wchodzi trzy terminy:

- 1) termin mniejszy (subiekt wniosku — *S*),
- 2) termin większy (predykat wniosku — *P*),
- 3) termin środkowy (relacja *R* między subiektem a predykatem).

Rozpatrzmy przykład sylogizmu: „każdy przychód jest wpływem — każdy wpływ jest powiększeniem aktywów — zatem każdy przychód jest powiększeniem aktywów”. W przedstawionym przykładzie *przychód* jest pojęciem podrzędnym, właściwość *bycia wpływem* — terminem środkowym, a właściwość *bycia powiększeniem aktywów* — predykatem [Lewicka 2000, s. 275–316]. Dlatego tak ważną rolę w metodach sztucznej inteligencji wspomagających decyzje menedżerów odgrywają sylogizmy, które ułatwiają na podstawie bardzo dużej liczby faktów i sprecyzowanych prawidłowości wysuwanie logicznych wniosków. W praktyce wiele relacji logicznych można wywieść wyłącznie ze związków pośrednich, stąd kluczową funkcję pełni sylogizm kategoryczny należący do wnioskowań pośrednich.

Sylogizm jest poprawny, gdy spełnia określone warunki.

1. Występują w nim tylko trzy terminy (nazwy).
2. Termin średni przynajmniej w jednej z przesłanek musi być rozłożony (w podanym przykładzie „wpływ” jest terminem średnim).

3. Termin rozłożony we wniosku musi być rozłożony w przesłance.
4. Następnie obie przesłanki nie mogą być zdaniami przeczącymi.
5. Obie przesłanki nie mogą być zdaniami szczegółowymi.
6. Jeżeli jedna z przesłanek jest zdaniem przeczącym, to wniosek nie może być zdaniem twierdzącym.
7. Gdy jedna z przesłanek jest zdaniem szczegółowym, wówczas wniosek nie może być zdaniem ogólnym.
8. Wniosek nie może być zdaniem przeczącym, jeżeli żadna z przesłanek nie jest zdaniem przeczącym.

Oto przykłady sylogizmów.

**Prawo sylogizmu hipotetycznego koniunkcyjnego** można zapisać jako:

$$[(p \rightarrow q) \wedge (q \rightarrow r)] \rightarrow (p \rightarrow r). \quad [2.9]$$

Rozważana teza jest formułą implikacyjną, w której poprzedniku występuje koniunkcja dwóch implikacji; następnik pierwszej implikacji przechodzi w poprzednik drugiej, której następnikiem jest trzecia zmienna zdaniowa  $r$ .

#### Przykład 2.1

Jeśli z faktu, że  $X(p)$  jest brygadzystą wynika, że nadzoruje  $Y(q)$  i jednocześnie nadzorowanie  $Y(q)$  wymaga znajomości nauk o zarządzaniu  $Z(r)$ , to z tego należy wyciągnąć wniosek, że  $X(p)$  jest brygadzystą, który zna się na zarządzaniu  $Z(r)$ .

**Prawo sylogizmu hipotetycznego bezkoniunkcyjnego**, nazywane również sylogizmem hipotetycznym warunkowym, ma postać:

$$(p \rightarrow q) \rightarrow [(q \rightarrow r) \rightarrow (p \rightarrow r)]. \quad [2.10]$$

#### Przykład 2.2

Z tego, że  $X(p)$  jest brygadzystą, który kieruje  $Y(q)$  wynika, że jeśli kierowanie  $Y(p)$  wymaga znajomości nauk o zarządzaniu  $Z(r)$ , to  $X(p)$  jako brygadzysta zna się na zarządzaniu  $Z(r)$ .

Podobnie ważną funkcję, jak pojęcie sylogizmu, w „maszynowym rozumowaniu” pełni pojęcie kontrapozycji, której rolę należy wyjaśnić w świetle znaczenia zestawienia najważniejszych właściwości związków logicznych, takich jak: przemienność, łączność, przechodniość. **Zasada kontrapunkcji** wyraża kluczową dla logicznego wynikania ideę, iż jedyną cechą zdań, istotną przy wnioskowaniu, jest ich struktura logiczna.

Jeśli ze zbioru przesłanek  $X$  wynika zdanie  $P$  i w miejsce pewnych, dowolnie wybranych zmiennych zdaniowych w zbiorze  $X \cup P$  wstawimy inne zmienne (podstawienie  $e$ ), to powstałe w ten sposób zdanie  $e(P)$  będzie logiczne, a konsekwencją otrzymanego zbioru zdań będzie kontrapozycja  $e(X)$ . Innymi słowy, kontrapozycja jest rodzajem wnioskowania bezpośredniego, które powstaje przez kombinację konwersji i obwersji.

**Obwersja** polega na zmianie jakości przy zachowaniu ilości i zanegowaniu orzecznika. Obwersji podlegają wszystkie jej zdania kategoryczne:

$$S a P \leftrightarrow S e \neg P.$$

„Każde  $S$  jest  $P$ ”, np. każda maszyna jest wytworem materialnym jest równoważne z „Każde  $S$  nie jest nie  $P$ ”, czyli każda maszyna nie jest nie wytworem materialnym. Można tego typu obwersję zastosować do kwalifikacji w przedsiębiorstwie środków materialnych i niematerialnych, co w gospodarce opartej na wiedzy jest ważne przy wycenie wartości przedsiębiorstwa coraz bardziej nasyconego dobrami niematerialnymi, prawnymi (patentami, licencjami) itp.

$$S i P \leftrightarrow S o \neg P,$$

$$S e P \leftrightarrow S a \neg P,$$

$$S o P \leftrightarrow S i \neg P.$$

### Przykład 2.3

Niektórzy pracownicy są inteligentni  $\leftrightarrow$  niektórzy pracownicy nie są nieinteligentni.  
 Żaden menedżer nie jest ubogi  $\leftrightarrow$  każdy menedżer jest zamożny.  
 Niektórzy dyrektorzy nie są otwarci  $\leftrightarrow$  niektórzy dyrektorzy są asertywni.

**Kontrapozycja całkowita** polega na zamianie miejscami terminów i zanegowanie obu:

$$S a P \leftrightarrow \neg P a \neg S,$$

$$S o P \leftrightarrow \neg P o \neg S,$$

$$S e P \rightarrow \neg P o \neg S.$$

### Przykład 2.4

Każdy dyrektor jest pracownikiem  $\leftrightarrow$  każdy nie pracownik nie jest dyrektorem.  
 Niektórzy pracownicy nie są uporządkowani  $\leftrightarrow$  niektórzy uporządkowani pracownicy nie są pracownikami.  
 Żadna maszyna nie jest zasobem wyłącznie materialnym  $\rightarrow$  niektóre nie maszyny nie są zasobem wyłącznie materialnym.

**Kontrapozycja częściowa** oznacza zanegowanie tylko orzecznika:

$$S \text{ a } P \leftrightarrow \neg P \text{ e } S,$$

$$S \text{ o } P \leftrightarrow \neg P \text{ i } S,$$

$$S \text{ e } P \rightarrow \neg P \text{ i } S.$$

### Przykład 2.5

Niektórzy kierownicy nie są dyrektorami naczelnymi  $\leftrightarrow$  niektórzy nie dyrektorzy naczelnicy są kierownikami.

Kontrapozycja częściowa jest konwersją wyniku obwersji. **Kontrapozycja zupełna** jest obwersją kontrapozycji częściowej lub obwersją konwersji wyniku obwersji.

Kontrapozycji zupełnej dokonuje się w następujący sposób: najpierw dokonujemy obwersji, potem konwersji, a następnie obwersji.

$$A \rightarrow o A' \rightarrow k A'' \rightarrow o A''',$$

$$S \text{ a } P \rightarrow o S \text{ e } (\neg P) \rightarrow k (\neg P) \text{ e } S \rightarrow o (\neg P) \text{ a } (\neg S),$$

$$S \text{ e } P \rightarrow o S \text{ a } (\neg P) \rightarrow k (\neg P) \text{ i } S \rightarrow o (\neg P) \text{ o } (\neg S).$$

$S \text{ i } P$  nie podlega kontrapozycji zupełnej.

$$S \text{ o } P \rightarrow o S \text{ i } (\neg P) \rightarrow k (\neg P) \text{ i } S \rightarrow o (\neg P) \text{ o } (\neg S).$$

Przykład kontrapozycji częściowej lub zupełnej przedstawia rysunek 2.8.

**Rysunek 2.8. Przykład ilustrujący kontrapozycję:**

**żadna mała firma nie jest korporacją**

— zdanie ogólnoprzeczające

$$S \text{ e } P \rightarrow o S \text{ a } (\neg P) \rightarrow k (\neg P) \text{ i } S \rightarrow o (\neg P) \text{ o } (\neg S)$$



Źródło: Opracowanie własne.

### Przykład 2.6

Niektórzy mężczyźni nie są kierownikami (kontrapozycja częściowa).

$$S \text{ o } P \rightarrow o S \text{ i } (\neg P) \rightarrow k (\neg P) \text{ i } S.$$

Niektórzy mężczyźni są nie kierownikami.

Niektórzy nie kierownicy są mężczyznami.

## Przykład 2.7

Tylko pracownicy są dyrektorami.

Tylko  $S$  są  $P \rightarrow P$  a  $S$  (każde  $P$  są  $S$ ).

Tylko  $S$  nie są  $P \rightarrow$  każde  $(\neg S)$  jest  $P$ , czyli  $(\neg S \wedge P) \leftrightarrow$  każdy nie pracownik jest nie dyrektorem.

Każdy dyrektor jest pracownikiem.

$S \wedge P \rightarrow$  o  $S \wedge (\neg P) \rightarrow$  k  $(\neg P) \wedge S \rightarrow$  o  $(\neg P) \wedge (\neg S)$ .

Żaden dyrektor nie jest nie pracownikiem.

Żaden nie pracownik nie jest dyrektorem  $(\neg P \wedge S)$ .

Każdy nie pracownik jest nie dyrektorem  $(\neg P) \wedge (\neg S)$ .

Opisane przykłady logicznych metod optymalizacyjnych, jak sylogizm i kontrapozycja, wskazują, że metody te są bardzo przydatne w procesie wspomagania decyzji na etapie eliminacji logicznie niespójnych wariantów rozwiązań, co pozwala zawęzić obszar poszukiwań realnych i wykonalnych rozwiązań, a tym samym zwiększyć efektywność i skuteczność procesu podejmowania decyzji, np. w zarządzaniu kadrami.

## 2.2.2. Statystyczne metody optymalizacyjne

Wśród statystycznych metod optymalizacyjnych poczesne miejsce zajmuje **teoria gier**, która jest działem matematyki zajmującej się badaniem optymalnego zachowania jednostek, organizacji lub różnego rodzaju grup społecznych w przypadku konfliktu interesów. Wywodzi się z badań gier hazardowych. Nowoczesna teoria gier powstała w 1944 r., a jej twórcami byli: J. Von Neumann i O. Morgenstern [Von Neumann 1944].

Stosując teorię gier, próbuje się przewidzieć sukces jednostki w procesie podejmowania decyzji, w zależności od wyboru innych uczestników w tym procesie. Jeśli gracz uczestniczy w grze-biznesie, przy wyborze metod działania powinien wziąć pod uwagę możliwości innych graczy-biznesmenów. Jednak myśląc o swoich wyborach, należy uznać, że inni myślą o moich wyborach także, próbując je przewidzieć i dostosować się do nich. Zatem w strategii podejmowania decyzji należy wziąć pod uwagę swój tok myślenia, tok rozumowania innych, którzy z kolei starają się wziąć pod uwagę swoje refleksyjne myślenie o myśleniu innych graczy (biznesmenów). Stosowanie tej metody wpisuje się więc dobrze w koncepcję zarządzania i sterowania refleksyjnego [Kim, Nazaretov, Krupa 1991], [Krupa 2004].

Centralnym punktem wszelkich procesów decyzyjnych jest konflikt interesów jawnie lub niejawnie (natura) biorących udział w tym procesie decydentów (graczy). Sytuacja konfliktowa jest wynikiem faktu, iż wszystkie działania (decyzje) wpływają w określony sposób na wynik procesu, który z kolei jest różnie oceniany przez decydentów. Przyczyną konfliktu może być nie tylko kolizja interesów

poszczególnych graczy, ale ograniczenia informacyjne istniejące w procesie, które uniemożliwiają porozumienie się lub przynajmniej sygnalizowanie zachowań. W przypadku problemów o sumie zerowej sytuacja konfliktowa jest najbardziej klarowna, wygrana jednego z graczy jest przegraną drugiego, a porozumienie jest całkowicie niemożliwe. Ścisłej z grą o sumie zerowej mamy do czynienia wówczas, gdy określony jest tylko jeden wskaźnik kosztów (funkcja celu), przy czym jeden z graczy ma na celu minimalizację, a drugi maksymalizację tego wskaźnika. Jeżeli gracze podejmują decyzje w tym samym momencie lub, ogólniej biorąc, porządek podejmowania decyzji nie jest istotny dla przebiegu procesu, gra jest statyczna. Jeżeli zbiór możliwych do podjęcia decyzji każdego z graczy jest skończony, mówimy o grze skończonej. Ponieważ jest on wówczas zbiorem dyskretnym, mamy do czynienia z dyskretnym procesem decyzyjnym. W grze o sumie zerowej może brać udział tylko dwóch graczy i w związku z tym wystarczy podać tylko wypłaty jednego z nich (tablica 2.3).

**Tablica 2.3. Przykład 1**

<i>WK</i>	<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>
<i>A</i>	12	-1	1	0
<i>B</i>	5	1	7	-20
<i>C</i>	3	2	4	3
<i>D</i>	-16	0	0	16

Źródło: Opracowanie własne.

Strategia *C* dla gracza *K* jest zdominowana przez strategię *B*. Oznacza to, że każdy wynik dawany przez *B* jest co najmniej równie korzystny, co odpowiedni wynik dawany przez *C*, a przynajmniej jeden wynik dawany przez *B* jest lepszy niż odpowiedni wynik dawany przez *C*. Strategia *C* dla gracza *W* i *B* dla gracza *K* jest punktem siodłowym. Grając *C*, *W* zapewnia sobie wygraną co najmniej 2. Grając *B*, *K* zapewnia sobie, że *W* wygra co najwyżej 2.

**Tablica 2.4. Przykład 2**

<i>WK</i>	<i>A</i>	<i>B</i>
<i>A</i>	2	-3
<i>B</i>	0	3

Źródło: Opracowanie własne.

Obaj gracze powinni wybrać zawierające go strategie, jeżeli gra ma punkt siodłowy. Wyznaczenie punktu siodłowego może się odbywać w sposób następujący: jeżeli maks-min (największa z najmniejszych wartości) wierszy i min-maks (najmniejsza z największych wartości) kolumn jest taki sam, oznacza to, że leży on w punkcie siodłowym. W przeciwnym przypadku gra nie posiada punktu siodłowego. Optymalne rozwiązania w tej sytuacji mają zwykle postać strategii mieszanych.

Czy istnieje strategia mieszana dla  $K$ , taka że nawet jeżeli  $W$  będzie ją znał, to nie będzie mógł tego wykorzystać? Niech  $K$  gra  $A$  z prawdopodobieństwem  $p$  i  $B$  z prawdopodobieństwem  $(1 - p)$ . Jeżeli  $W$  gra  $A$ , to wygrywa średnio:  $2p - 3(1 - p) = 5p - 3$ . Jeżeli natomiast  $W$  gra  $B$ , to wygrywa  $3(1 - p)$ . Przyrównując, otrzymujemy  $p = 3/4$ . Grając  $(3/4A, 1/4B)$ ,  $K$  jest pewien, że  $W$  średnio nie wygra więcej niż  $3/4$ . Analogiczne rozumowanie przeprowadzone dla  $W$  daje strategię  $(3/8A, 5/8B)$  oraz pewność, że wygra co najmniej  $3/4$ . Przedstawione obliczenia prowadzą do ustalenia wartości gry.

Gra o sumie niezerowej może być wykorzystywana do tworzenia podstaw podejmowania decyzji przez firmy, które chcą ze sobą kooperować lub kooperować w niektórych obszarach, a w innych konkurować (wchodzić w relacje kooperacyjne), co we współczesnej rzeczywistości gospodarczej ma coraz częściej miejsce w świetle modeli biznesu opartych na outsourcingu czy też modeli zadaniowych realizowanych przez konsorcja wirtualne.

Przykładem gry dwuosobowej o sumie niezerowej mogą być interesy graczy, które nie są już dokładnie przeciwstawne — możliwa jest zatem kooperacja. Wymaga ona jednak komunikacji między graczami [Straffin, 2004]. Istnieją trzy założenia dotyczące możliwości komunikacji: brak komunikacji, komunikowanie się przed wyborem strategii, negocjacje co do wspólnego wyboru rozwiązania satysfakcjonującego dla obu stron.

Do pojęć, które mają zastosowanie zarówno w grach o sumie zerowej, jak i o sumie niezerowej, należą: kryterium dominacji (strategia zdominowana, dominująca), diagram przesunięć i równowaga (punkt równowagi). Z definicji wynika, iż równowaga (punkt równowagi) to taka para strategii (czystych lub mieszanych), że żadnemu z graczy nie opłaca się zmiana swojej strategii. Równowagi w grach o sumie niezerowej nazywamy równowagami Nasha. **Równowagę Nasha** w strategiach mieszanych stanowią strategie wyrównujące graczy.

Równowaga Nasha może mieć niepożądane własności. W szczególnym przypadku może ona być niekorzystna dla obu stron. Wynika to stąd, że znajdując równowagę Nasha (czyli swoje strategie wyrównujące), gracze ignorują swoje własne wypłaty; interesuje ich tylko to, aby przeciwnik nie mógł zyskać więcej.

Wynik gry jest **nieoptymalny w sensie Pareto**, jeśli gra ma inny wynik, dający obu graczom wyższe wypłaty lub tylko jednemu z graczy wyższą wypłatę. Wynik gry jest **Pareto-optymalny**, jeżeli taki inny wynik nie istnieje. Wnioskiem ogólnym może być stwierdzenie, że równowagi Nasha mogą być nieoptymalne w sensie Pareto, a to oznacza, że nie można ich traktować jako dobrego (racjonalnego) rozwiąza-

nia gry. Dlatego w przypadku gier o sumie niezerowej poszukuje się innych metod ich rozwiązywania. Generalnie nie istnieje jedna uniwersalna metoda rozwiązywania gier o sumie niezerowej [Straffin 2004].

Pojęcia teorio-decyzyjne stanowią obecnie powszechnie akceptowany język statystyki matematycznej [Magiera 2007]. Statystyczne zagadnienie decyzyjne pozwala zrealizować ciąg zadań, które zapewnią największą wartość oczekiwaną ostatecznego wyniku, np. wartość sprzedaży przedsiębiorstwa. Decyzje rozumiane są w tym obszarze jako elementy iloczynu kartezjańskiego w przestrzeni decyzyjnej  $D = X_1 \times X_2 \times X_3, \dots$ , a zmienne  $x_1, x_2, x_3, \dots$  oznaczają wszystkie cechy opisujące obiekty będące przedmiotem procesu decyzyjnego, istotne ze względu na ten proces,  $X_1, X_2, X_3, \dots$  zaś odpowiednio oznaczają dziedziny zmiennych, np. nakłady na promocję produktu, innowacyjność produktu, stopień niezawodności działania danego wytworu itp. Funkcja ryzyka związana jest z terminem „ryzyko” i oznacza przede wszystkim przedsięwzięcie, którego wynik jest nieznany albo niepewny, lub możliwość, że coś się uda albo nie uda, czy też inaczej jako stan, w którym rezultat osiągnięty w przyszłości jest nieznany, ale można zidentyfikować jego przyszłe alternatywy przy założeniu, że szanse wystąpienia tych alternatyw są znane. Ryzyko jest zatem niepewnością związaną z przyszłymi wydarzeniami lub wynikami decyzji. Wyniki decyzji prowadzą do wystąpienia niespodziewanej wysokości straty lub zysku. Oszacowanie ryzyka pomaga podejmującym decyzję ocenić prawdopodobieństwo odniesienia sukcesu lub porażki z jej wdrożenia do praktyki gospodarczej, a zatem lepiej zarządzać organizacją.

Ważnym metodycznie działaniem w procesach wspomagania decyzji jest **randomizacja**, która polega na losowym doborze osób do badania, przy czym losowość należy rozumieć tak, że każdy przypadek z badanej populacji ma taką samą szansę bycia uczestnikiem badania. Stosując odpowiednie metody losowania, badacze przyznają szansę każdej osobie z populacji na bycie osobą badaną. Powody stosowania randomizacji w badaniach są następujące:

- zminimalizowanie wpływu zmiennych niekontrolowanych; stosując losowość doboru osób do badania, minimalizowany jest wpływ niekontrolowanych czynników;
- odzwierciedlenie charakterystyki populacji (np. właściciele przedsiębiorstw) w badanej próbie;
- uprawdopodobnienie otrzymanych rezultatów badań; losowe dobieranie osób do badania sprawia, że istnieje nieznaczące prawdopodobieństwo, że badane grupy będą się różnić między sobą innymi zmiennymi niż tymi zaplanowanymi przez badacza, np. badacz nie mierzy poziomu inteligencji osób badanych, ale stosując losowy dobór osób do badania, sprawia, że poziom inteligencji w grupach badawczych powinien być podobny.

Klasyczne zagadnienia statystyczne rozpatrywane w ujęciu teorio-decyzyjnym mają dzisiaj szczególnie ważne znaczenie w SWD ze względu na dynamiczny rozwój modeli, metod i narzędzi opartych na regułach decyzyjnych.



**Reguły decyzyjne** stanowią łatwy do interpretacji opis regularności zawartych w danych, dzięki czemu, oprócz zadań automatycznej klasyfikacji, nadają się też do tworzenia zrozumiałych dla człowieka raportów syntetyzujących i zestawiających w odpowiednio zagregowany sposób wyniki finansowe, ekonomiczne czy też produkcyjne potrzebne menedżerom w skutecznym zarządzaniu opartym na wiedzy. Ich zaletą jest łatwość przełożenia na język naturalny zrozumiały przez kadre zarządzającą.

Zakres informacji dostępnej przy podejmowaniu decyzji opisany jest za pomocą trzech stanów:

- 1) **pewności** — w momencie podejmowania decyzji znamy stan świata zewnętrznego, zadanie wówczas polega na wyborze takiej decyzji, której jest przyporządkowana największa korzyść;
- 2) **ryzyka** — na skutek decyzji może wystąpić dwa lub więcej wyników, przy czym wszystkie wyniki, jak też ich prawdopodobieństwo wystąpienia jest znane podejmującym decyzje;
- 3) **niepewności** — gdy w rezultacie działania może zaistnieć dwa lub więcej wyników, przy czym nie znamy wyników i prawdopodobieństwa ich wystąpienia.

Za pomocą reguł decyzyjnych działających w oparciu o kryterium minimum nie ocenia się skutków błędnych decyzji, lecz skutki (straty) wynikające z niepodjęcia decyzji, która przy danym stanie byłaby najlepsza.

### 2.2.3. Metody heurystyczne

Koncepcja metod i strategii heurystycznych wynika z przesłanki, że dla większości problemów przestrzeni stanów zawiera pewne dodatkowe informacje. Koszt wyznaczenia tych informacji jest niewielki, a pozwalają one na dodatkowe klasyfikowanie stanów i prostsze wybieranie optymalnych kierunków przeszukiwań. Za pomocą stanów i operatorów można opisać rozwiązywany problem. Stanowi to uproszczenie. Operatory wykorzystane do stanów generują nowe stany. Stany początkowe, występujące na początku rozwiązywania problemu, i operatory tworzą graf przestrzeni stanów. Heurystyczne przeszukiwanie tego grafu jest procesem poszukiwaniażądanego podgrafu spełniającego zdefiniowane warunki.

**Heurystyka** jest metodą postępowania, która pozwala wyszukać dobre rozwiązania przy akceptowalnych nakładach obliczeniowych, ale bez gwarancji osiągalności optymalności celu, a nawet — w wielu przypadkach — bez określenia, w jakiej odległości od optymalnego jest otrzymane rozwiązanie. Wybierając najlepsze operatory, przeszukiwanie heurystyczne posługuje się różnymi środkami, takimi jak analogie czy uproszczenia. Celem jest ograniczenie zbioru przeszukiwanych stanów. Heurystyka nie gwarantuje jednak znalezienia satysfakcjonującego rozwiązania. Właściwie dobrana pozwala osiągać zadowalające wyniki w zadanym czasie.

Wśród argumentów przemawiających za stosowaniem strategii heurystycznej jest jej akceptowalna złożoność obliczeniowa oraz realistyczne założenia (dokładne rozwiązanie przybliżonego modelu czy też przybliżone rozwiązanie dokładnego modelu).

Biorąc pod uwagę efektywność przeszukiwania, istotne jest uwzględnienie takich czynników, jak niepewność wyniku czy też fragmentaryczność dostępnej wiedzy. Heurystyka może zwiększać niepewność uzyskania wyniku. Niepewność otrzymania wyniku wzrasta, jeśli wykorzystujemy wiedzę w postaci różnych praw, reguł, intuicji, których użyteczność nie jest do końca określona.

Metody heurystyczne są stosowane tam, gdzie brakuje algorytmu poszukiwania rozwiązań lub też znane algorytmy identyfikują niezadowolające rozwiązanie albo też nie dają gwarancji znalezienia rozwiązania zadania. Kluczowe znaczenie mają one przy rozwiązywaniu problemów o dużej złożoności obliczeniowej, zwłaszcza przy rozwiązywaniu problemów NP-zupełnych.

Posługując się metodami heurystycznymi, można stwierdzić, że wyszukiwanie żądanego stanu odbywa się najczęściej w sposób subiektywny. Zależy od reguł wypracowanych empirycznie, opierających się na opiniach profesjonalistów. Mianem „heurystyczne” określa się wszelkie prawa, kryteria, zasady i intuicje (także o niepewnej skuteczności) stosowane w zadaniach przeszukiwania. Pozwalają one na wybranie najbardziej efektywnych kierunków działania, z punktu widzenia osiągnięcia danego celu.

W metodach przeszukiwania heurystycznego stosuje się rozmaite strategie mające określone wady i zalety. Poniżej scharakteryzowano następujące strategie: w głąb, wszersch, najpierw najlepszy oraz strategię zachłanną.

**Strategia w głąb** może być nieskuteczna, kiedy spróbujemy ją zastosować do grafów o dużej głębokości. W takim przypadku wybieranie węzłów o coraz większej głębokości mogłoby być nieskuteczne nawet wtedy, gdyby węzeł celu miał głębokość skończoną. Dlatego też omawiana strategia jest zwykle udoskonalana poprzez uzupełnienie jej o mechanizm kontroli ograniczenia głębokości. Kiedy głębokość węzła przekroczy ograniczenie lub węzeł spełnia kryterium końcowe, następuje powrót. Istotą strategii w głąb jest badanie kolejnych węzłów wzdłuż jednej ścieżki. Dlatego też ta strategia jest najwłaściwsza dla tych problemów i grafów, w których ocena właściwości węzłów zależy ściśle od oceny właściwości ich poprzedników na ścieżce.

**Strategia wszersch** gwarantuje, że dla lokalnie skończonych grafów (tj. takich, w których każdy węzeł ma skończoną liczbę potomków) osiągnie się węzeł celu, jeżeli taki istnieje. Dlatego też strategia wszersch ma bardzo pożądaną cechę zbieżności. Ponadto jako pierwsze wyznaczane jest rozwiązanie optymalne pod względem długości ścieżki rozwiązania. Strategia ta ma także poważne wady. Analizowane są wszystkie węzły o głębokości mniejszej od głębokości wyznaczonego węzła celu. Zamiast jednej ścieżki w pamięci są przechowywane wszystkie węzły o danej głębokości przed wygenerowaniem jakiegokolwiek węzła o głębokości o jeden większej. Stąd też strategia ta charakteryzuje się dużymi wymaganiami dotyczącymi pamięci.

W każdej fazie strategii następują powroty do węzłów wygenerowanych i przechowywanych w pamięci.

**Strategia najpierw najlepszy** wykorzystuje pewną informację heurystyczną związaną z rozwiązywaniem problemu do zminimalizowania kosztów przeszukiwania. W tym celu stosuje się pewną funkcję heurystyczną, która wyraża ocenę węzła ze względu na następujące kryteria:

- zbieżność, czyli możliwość osiągnięcia celu;
- najmniejszy koszt drogi wyznaczonej od węzła początkowego, przez węzeł  $w$ , do węzła końcowego;
- najmniejszej złożoności obliczeniowej procesu przeszukiwania.

Do dalszego rozszerzania przestrzeni rozwiązań wybieramy „najlepszy” węzeł spośród wszystkich węzłów rozpatrywanych do tej pory, niezależnie od ich położenia w grafie. Przyjmujemy, że węzeł najbardziej obiecujący ma najmniejszą wartość funkcji heurystycznej.

Działanie strategii najpierw najlepszy można przedstawić na podstawie przykładu tzw. pięciu hetmanów. Należy wyobrazić sobie ustawienie pięciu hetmanów na szachownicy o wymiarach  $5 \times 5$ , przy czym żaden z hetmanów nie może atakować innego. Graf w przestrzeni stanów jest generowany, zaczynając od stanu początkowego, który odpowiada pustej szachownicy. Hetmany są umieszczane w kolejnych wierszach o numerach od 1 do 5, a węzły na każdym poziomie drzewa reprezentują możliwe pozycje hetmanów w następnych wierszach. Do oceny bieżącej pozycji jest wykorzystywana funkcja heurystyczna  $f$ . Jej wartość może być wyznaczona w różny sposób, np. jest to minimalna liczba nieatakowanych pól w wolnych wierszach (pozycja jest bardziej obiecująca, gdy liczba ta jest większa).

Uporządkowanie listy (zawierającej węzły grafu przestrzeni stanów z niewykorzystanymi krawędziami) według wartości funkcji heurystycznej jest najważniejszą operacją, odróżniającą strategię najpierw najlepszy od wcześniej omawianych algorytmów. Rozszerzenie węzłów jest dokonywane, podobnie jak w strategii w głąb, przez ekspansję, czyli za pomocą generowania wszystkich potomków.

Podstawową operacją **strategii zachłannej** jest ekspansja węzłów. Po jej wykonaniu są badane nowe węzły i najbardziej obiecujący z nich jest wybierany do dalszej ekspansji. Strategia zachłanna wykorzystuje lokalną optymalizację i nie są w niej możliwe powroty do żadnego przodka aktualnie badanego węzła. Dlatego strategia ta jest nieodwracalna i oczywiście nie jest symetryczna. Strategia zachłanna odznacza się prostym algorytmem obliczeniowym. Jej słabą stroną jest brak możliwości powrotu do kierunków przeszukiwania, które na pewnym etapie były lokalnie gorsze. Może to prowadzić do badania drogi wiodącej do węzła końcowego, która nie spełnia kryterium celu lub do przeszukiwania drogi nieskończonej [Cormen 1997].

## 2.3. Techniki kalkulacyjne w eksperymentach symulacyjnych

Techniki kalkulacyjne w SWD mogą być wykorzystywane do obliczeń dokonywanych w oparciu o dane z przeszłości w celu modelowania doświadczenia i planowania oraz przewidywania przyszłych zachowań organizacji i jej elementów w zmienionych warunkach ich otoczenia. Doświadczenie aktywnych użytkowników SWD może posłużyć do tworzenia, weryfikacji i uaktualniania normatywnych parametrów BD oraz BW, co przyczyni się do uwiarygodnienia wyników uzyskiwanych z zestawień wyjściowych SWD.

### 2.3.1. Obliczenia statystyczne — przykłady

Do obliczeń statystycznych prowadzonych dla celów selekcji i pogrupowania odpowiednich danych mogą być wykorzystywane **operacje analizy wielowymiarowej** w systemach OLAP: obracania, rankingowania, selekcji, projekcji, wycinania i kawałkowania oraz drążenia danych, tabele przestawne jako klienci OLAP czy też tabele przestawne jako narzędzia do tworzenia i zarządzania HD. Przydatne w podejmowaniu decyzji mogą być również raporty tworzone z analizy rozkładu częstotliwości oraz krzyżowe zestawienia rozmaitych danych w różnych wariantach, a także grupowanie elementów, np. według podsumowań: efektywność sprzedaży według pracowników, produktów, firm, rejonów geograficznych itp.

**Tablica 2.5.** Tabela przestawna do wygenerowania raportu zbiorczego wartości sprzedaży kawy i herbaty w poszczególnych kwartałach roku

	G	H	I	J	K	U	P	AA
		Cena rynkowa	Sprzedaż losowa za jeden dzień (kg/szt. - ilość)		Sprzedaż za jeden dzień	Wartość sprzedaży za okres (PLN)	Wartość sprzedaży za okres (EUR)	Placa brutto za miesiąc
1 Towary		Cena zł (kg/sz. symulowana (kg/szt.))						
2 Wieprzowina	14	=LOS(I*(H2+0,2)-H2-0=LOS(I*(K2+K2*0,1)-I) 1000				=I2*J2*N2	=O2/+ziemiaki!Y\$9	=Y2+Z2
3 Sery	21	=LOS(I*(H3+0,2)-H3-0=LOS(I*(K3+K3*0,1)-I) 250				=I3*J3*N3	=O3/+ziemiaki!Y\$9	=Y3+Z3
4 Wolowina	18	=LOS(I*(H4+0,2)-H4-0=LOS(I*(K4+K4*0,1)-I) 150				=I4*J4*N4	=O4/+ziemiaki!Y\$9	=Y4+Z4
5 Masło	4	=LOS(I*(H5+0,2)-H5-0=LOS(I*(K5+K5*0,1)-I) 3000				=I5*J5*N5	=O5/+ziemiaki!Y\$9	=Y5+Z5
6 Ogórki	2	=LOS(I*(H6+0,2)-H6-0=LOS(I*(K6+K6*0,1)-I) 300				=I6*J6*N6	=O6/+ziemiaki!Y\$9	=Y6+Z6
7 Pomidory	5	=LOS(I*(H7+0,2)-H7-0=LOS(I*(K7+K7*0,1)-I) 500				=I7*J7*N7	=O7/+ziemiaki!Y\$9	=Y7+Z7

Źródło: Opracowanie własne.

Przedstawione przykłady zastosowań tabel przestawnych ilustrują możliwości wytwarzania zestawień wynikowych przydatnych w podejmowaniu decyzji dotyczących dystrybucji produktów spożywczych (kawy i herbaty) z punktu widzenia optymalizacji sprzedaży według różnych kryteriów czy też oceny skuteczności sprzedawców (tablice 2.5–2.6).

**Tablica 2.6. Tabela przestawna do wygenerowania raportu zbiorczego wartości sprzedaży kawy i herbaty dla poszczególnych pracowników analizowanej firmy**

	G	H	I	J	K	O	P	AA
		Cena rynkowa symulowana Cena zł ((kg/szt.)	Sprzedaż losowa za jeden dzień (kg/szt.-ilość)	Sprzedaż za jeden dzień	Wartość sprzedaży za okres (PLN)	Wartość sprzedaży za okres (EUR)	Placa brutto za miesiąc	
1 Towary								
2 Wieprzowina	14	14,13 zł	1 063	1000	20 474 202,72 zł	€4 600 944,43	22 738,25 zł	
3 Sery	21	21,13 zł	266	250	7 413 793,00 zł	€1 666 020,90	8 501,63 zł	
4 Wołowina	18	18,13 zł	159	150	3 790 575,14 zł	€851 814,64	5 194,64 zł	
5 Masło	4	4,13 zł	3 188	3000	18 175 286,84 zł	€4 084 334,12	25 508,85 zł	
6 Ogórki	2	2,13 zł	319	300	882 844,74 zł	€198 392,08	897,20 zł	
7 Pomidory	5	5,13 zł	531	500	3 905 120,82 zł	€877 555,24	3 855,37 zł	

Źródło: Opracowanie własne.

### 2.3.2. Techniki iteracyjne — przykłady

**Iteracja** to technika umożliwiająca osiągnięcie oczekiwanego rezultatu drogą wielokrotnych powtórzeń tych samych operacji. W eksperymentach symulacyjnych odgrywa znaczącą rolę, ponieważ ułatwia znalezienie optymalnego rozwiązania wśród zbioru rozwiązań dopuszczalnych.

Założmy, że problem decyzyjny polega na ustaleniu wielkości zamówienia części zamiennych dla urządzeń pracujących na linii produkcyjnej. Liczba części, które ulegają comiesięcznemu zużyciu, jest zmienna i zależna m.in. od takich czynników, jak: obciążenie linii produkcyjnej, jakość i trwałość wykorzystywanych w urządzeniach części, stan techniczny i wiek urządzeń oraz doświadczenie osób obsługujących te urządzenia. Wielkość zamówienia powinna być taka, aby, z jednej strony, zapewnić realizację zaplanowanej wielkości produkcji, z drugiej zaś strony nie generować nadmiernych kosztów zakupów oraz zbędnego przepełnienia magazynów. Problem ten wymaga zatem rozpatrzenia różnych wariantów planowanej wielkości zamówienia w zależności od zaplanowanej wielkości produkcji i prawdopodobnego zużycia części urządzeń (tablica 2.7).

Zastosowanie techniki iteracyjnej pozwala na przeprowadzenie szczegółowej analizy przedstawionego problemu i dobrania takiego wariantu rozwiązania, które, przy założonych ograniczeniach (wielkość produkcji i przewidywana liczba zużytych części), pozwala na osiągnięcie maksymalnego przewidywanego zysku.

Wyniki modeli symulacyjnych mogą stanowić również zbiór danych uczących modele analityczne w systemach SWD. Na ich podstawie mogą być generowane uogólnione reguły decyzyjne, które następnie są wykorzystywane we wspomaganiu podejmowania decyzji podczas rozwiązywania problemów decyzyjnych. Reguły decyzyjne i algorytmy wnioskowania logicznego stanowią treść kolejnego rozdziału.

Tablica 2.7. Przykład zastosowania iteracji w eksperymencie symulacyjnym

Nr iteracji	Planowana wielkość produkcji	Przewidywana liczba zużytych części	Planowana wielkość zamówienia	Koszt zamówienia	Koszt magazynowania	Przewidywany zysk
1	1 000	150	120	2 940	1 500	25 110
2	1 000	150	135	3 308	1 688	24 780
3	1 000	150	150	3 675	1 875	24 450
4	1 300	195	156	3 822	1 950	32 643
5	1 300	195	176	4 300	2 194	32 214
6	1 300	195	195	4 778	2 438	31 785
7	1 600	240	192	4 704	2 400	40 176
8	1 600	240	216	5 292	2 700	39 648
9	1 600	240	240	5 880	3 000	39 120

Źródło: Opracowanie własne.

### 2.3.3. Reguły decyzyjne i algorytmy wnioskowania logicznego — przykłady

Naturalne reguły decyzyjne są **regułami stosowanymi** spontanicznie przez osoby dokonujące ocen i podejmujące decyzje. Pozwalają one integrować oceny cząstkowe dokonane w układzie: kryteria x warianty. Stosowanie reguł decyzyjnych jest podobne do stosowania reguł gramatycznych w języku. Problemem jest często wybór odpowiedniej reguły (reguła wyboru reguł).

Formuły FOL są wyrażone w logice pierwszego rzędu. Nie można w logice pierwszego rzędu dowieść fałszywego twierdzenia. Wszystkie prawdziwe twierdzenia mają dowód.

Logika pierwszego rzędu i rachunek predykatów pierwszego rzędu mają ograniczone możliwości ekspresji, np. w FOL nie można zapisać stwierdzenia: „wszystkie predykaty mają jeden argument”.

**Reguła koniunkcji** polega na wyborze takiego wariantu, w którym wszystkie ocenione cechy osiągną lub przekraczają założone progi krytyczne. Ułatwia ona podejmowanie ostrożnych decyzji. Decydent stosujący tę regułę najprawdopodobniej zdiagnozował decyzję jako bardzo ryzykowną, jako wielowymiarowy problem wymagający myślenia analitycznego. Reguła ta stawia duże wymagania decydentowi i prowadzi do bardzo ostrej selekcji wariantów działania.

**Reguła dysjunkcji** polega na wyborze takiego wariantu, w którym przynajmniej jedna spośród ocenianych cech przekracza lub osiąga założony próg kry-

tyczny. Przeciwnością reguły dysjunkcyjnej jest sytuacja, w której uwzględniamy jedną, co najwyżej dwie najbardziej wyróżniające się cechy, a potem wybieramy najczęściej ten wariant, który jest najbardziej skontrastowany albo który jako pierwszy nasunął się nam na myśl. Reguła ta może być przydatna w sytuacjach powtarzalnych, ale w sytuacjach nietypowych prowadzi do błędów i chaotycznych decyzji.

Wśród zalet reprezentacji logicznej można wymienić:

- spójność: wszystkie fakty to stwierdzenia logiczne;
- zupełność: można wywnioskować wszystkie prawdziwe stwierdzenia dające się wyrazić w ramach FOL;
- oddzielenie części epistemologicznej od dedukcyjnej.

Oto wady FOL:

- problemy z eksplozją kombinatoryczną, niewielka efektywność wnioskowania,
- niektóre formy wiedzy są trudne do wyrażenia za pomocą reprezentacji logicznej.

W niektórych przypadkach zmienne zależne mogą być tylko pośrednio związane z rozwiązywanym zagadnieniem, a wtedy możemy próbować znaleźć rozwiązanie dla pewnej miary efektu (z taką sytuacją mamy np. do czynienia przy dopasowywaniu krzywej metodą najmniejszych kwadratów). Ten typ obliczeń występuje również w zagadnieniach związanych z analizą typu „co, jeśli” oraz w zadaniach optymalizacji decyzji.

W rozważanych przypadkach często zachodzi potrzeba stosowania tych samych technik iteracyjnych lub rekurencyjnych, jakich używa się do rozwiązywania równań nieliniowych lub jakie stosuje się w programowaniu dynamicznym [Bellman 1967].

Wśród algorytmów wnioskowania logicznego metody oparte na twierdzeniu sformułowanym przez XVIII-wiecznego matematyka T. Bayesa odgrywają znaczną i ostatnio rosnącą rolę w dziedzinie sztucznej inteligencji. Wzór Bayesa stał się podstawą do rozwoju teorii i algorytmów różnych form wnioskowania probabilistycznego.

Opisany poniżej przykład obrazuje sens zastosowań wzoru Bayesa we wspomaganie decyzji dotyczących jakości procesów wytwarzania.

Rozważmy przykład związany z produkcją pewnego detalu. Detal ten wytwarzają fabryki *A*, *B* i *C*, mające udziały w rynku odpowiednio: 60%, 30% i 10% całej produkcji. Wadliwość dla poszczególnych fabryk wynosi odpowiednio: 3%, 2% i 1%.

Odpowiemy na dwa pytania nurtujące menedżerów:

- 1) Jakie jest prawdopodobieństwo, że losowo wybrany detal z magazynu jest wadliwy?
- 2) Wylosowano detal wadliwy. Jakie jest prawdopodobieństwo, że detal pochodzi z fabryki *B*?

Wprowadźmy oznaczenia następujących zdarzeń:

$A$  — wylosowanie detalu z fabryki  $A$ ,

$B$  — wylosowanie detalu z fabryki  $B$ ,

$C$  — wylosowanie detalu z fabryki  $C$ .

Wiadomo, że  $P(A) = 0,6$ ,  $P(B) = 0,3$ ,  $P(C) = 0,1$ . Przez  $P(W|A)$  oznacza się prawdopodobieństwo warunkowe, że wylosowany w fabryce  $A$  detal jest wadliwy. Na podstawie danych otrzymujemy:

$$P(W|A) = 0,03, P(W|B) = 0,02, P(W|C) = 0,01.$$

Zadanie (a) polega na zastosowaniu wzoru do obliczania prawdopodobieństwa całkowitego. W naszym przypadku:

$$P(W) = P(A)P(W|A) + P(B)P(W|B) + P(C)P(W|C),$$

gdzie:

$P(W)$  — wylosowanie detalu wadliwego.

$$P(W) = 0,6 \cdot 0,03 + 0,3 \cdot 0,02 + 0,1 \cdot 0,01 = 0,018 + 0,006 + 0,001 = 0,025.$$

Zadanie (b) polega na zastosowaniu wzoru Bayesa:

$$P(B|W) = \frac{P(B)P(W|B)}{P(W)} = \frac{0,006}{0,025} = \frac{6}{25}.$$

Na podstawie wykonanych obliczeń można wnioskować, że prawdopodobieństwo, iż wybrany z magazynu detal jest wadliwy wynosi 0,025 oraz prawdopodobieństwo, że detal pochodzi z fabryki  $B$  wynosi  $6/25$  (0,24).

Przykłady ciekawych zastosowań reguł decyzyjnych można znaleźć w pracach zespołu R. Budzińskiego [Budziński, Szafranek 2006], [Budziński, Wawrzyniak 2006].

## 2.4. Przygotowanie BD na potrzeby SWD

**Bazy danych** (BD), w których dane są gromadzone, przechowywane i udostępniane, stanowią istotną część każdego systemu informatycznego. W SWD szczególnie znaczenie mają: typ, zakres, ilość, jakość oraz czas przechowywania danych. Na wybranym przykładzie bazy relacyjnej zostanie przedstawiona podstawowa koncepcja BD, przygotowywanej dla systemu SWD.

Omawiany system SWD ma powstać w celu usprawnienia funkcjonowania sieci wodociągowej. Na podstawie gromadzonych w bazie danych, dotyczących stanu tej



sieci na przestrzeni rozpatrywanego okresu, można zbudować modele analityczno-prognostyczne, które będą wspierały podejmowanie decyzji w zakresie utrzymania właściwego poziomu obciążenia i przeciwdziałania awariom sieci wodociągowej. Baza danych w tym przypadku stanowi główne źródło zasilające system SWD.

### 2.4.1. Struktura i istota działania BD

Według podstawowej definicji J.D. Ullman [Ullman, Widom 2002] baza danych jest to uporządkowany zbiór danych, opisujących wybrany fragment rzeczywistości, które są na trwałe przechowywane w pamięci komputera i do których może mieć dostęp wielu użytkowników w dowolnym momencie czasu. Podobną definicję proponują: [Delobel, Adiba 1989], [Ostrowska 2002], [Silberschatz, Korth, Sudarshan 2006], prezentując BD jako zbiór danych o określonej strukturze, zapisanych na zewnętrznym nośniku pamięci komputera, mogących zaspokoić potrzeby wielu użytkowników, korzystających z niego w sposób selektywny, w dogodnym dla siebie czasie. Podsumowując, należałoby stwierdzić, że BD to komputerowe źródło danych, uporządkowanych i dobranych tematycznie, z którego może korzystać jednocześnie wielu użytkowników. BD tworzymy wtedy, kiedy potrzebujemy przechowywać oraz selekcjonować duże zbiory danych, które można pogrupować i podzielić na mniejsze jednostki informacyjne.

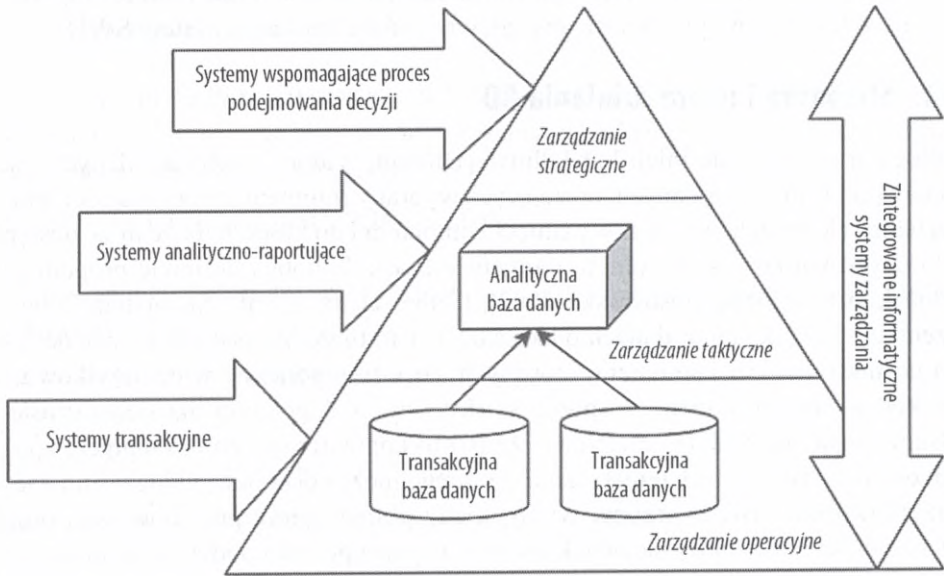
BD wykorzystywane w przedsiębiorstwach dzielimy na dwie zasadnicze grupy:

- 1) transakcyjne BD, obsługujące bieżącą działalność przedsiębiorstwa,
- 2) analityczne BD, obsługujące analityczne potrzeby przedsiębiorstwa.

Jak wynika z rysunku 2.9, transakcyjne bazy danych są zasilane przez systemy transakcyjne w trakcie ich codziennego funkcjonowania. Są magazynami danych, wspomagającymi zarządzanie operacyjne w przedsiębiorstwie. Wyselekcjonowane dane z poziomu transakcyjnego trafiają do analitycznych baz danych, gdzie stanowią wsparcie dla systemów analitycznych na taktycznym poziomie zarządzania firmą. SWD na poziomie strategicznym mogą wykorzystywać wszelkie niezbędne zasoby danych — z poziomu transakcyjnego lub/i analitycznego. Tylko zintegrowane informatyczne systemy zarządzania (MRP, MRP II, ERP), które w swojej strukturze posiadają zarówno transakcyjny, jak i analityczny magazyn danych, obejmują swoją funkcjonalnością całość zarządzania przedsiębiorstwem.

Analityczne bazy danych zostaną opisane w kolejnych rozdziałach podręcznika, a teraz skupimy się na charakterystyce transakcyjnych BD. Wśród tego typu baz największą popularnością oraz najszerszym zakresem zastosowań charakteryzuje się relacyjny model BD. Dlatego na nim skoncentrują się dalsze rozważania dotyczące projektowania transakcyjnych BD.

Rysunek 2.9. Miejsce baz danych i systemów wspomagających zarządzanie w hierarchii zarządzania



Źródło: Opracowanie własne.

## 2.4.2. Modelowanie logiczne i fizyczne BD

Każdy produkt informatyczny funkcjonuje w określonym cyklu życia, począwszy od koncepcji jego powstania, a skończywszy na wycofaniu go z eksploatacji. Sposób realizacji poszczególnych etapów, ich wyniki i forma prezentacji zależą w znaczącym stopniu od końcowego produktu: jego złożoności oraz przyjętej metody projektowania. W przypadku projektowania relacyjnych baz danych projekt obejmuje dwie zasadnicze fazy: modelowania konceptualnego i fizycznego [Rostek 2010c]. **Model konceptualny** BD definiuje jej strukturę logiczną, obejmującą encje oraz związki między encjami.

Encja (rysunek 2.10), jako element struktury relacyjnej BD, składa się z nazwy encji (*StanS* na rysunku 2.10) oraz przynależących do niej atrybutów (*idStanu*, *DataS*, *GodzinaS*, *PrzepływW*, *CisnienieW*, *StanAlarm*). Każdy atrybut charakteryzowany jest przez nazwę atrybutu oraz typ danych, które przechowuje (np. *DataS* — dane typu data, *PrzepływW* — dane typu liczba rzeczywista, *Kontroler* — dane typu tekstowego). Dodatkowo na liście atrybutów zostaje wyróżniony ten, który jest wartością jednoznacznie identyfikującą każdy rekord (wiersz) w fizycznej tabeli danych, a który nosi nazwę klucza podstawowego (znak # przy atrybucie *idStanu* na rysunku 2.10). Atrybuty encji mogą być wymagane (ich wartości muszą zostać obowiązkowo wypełnione przy wprowadzaniu wartości do fizycznej tabeli danych — znak \* przy

atrybutach) lub opcjonalne (ich wartości mogą zostać wprowadzone opcjonalnie — znak ° przy atrybutach na rysunku 2.10).

Rysunek 2.10. Przykład encji

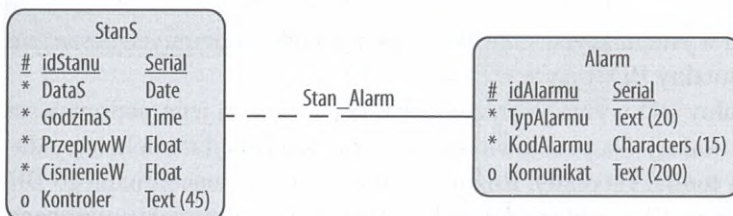


Źródło: Opracowanie własne.

Związki między encjami modelu conceptualnego mogą być typu 1–1 (rysunek 2.11), 1– $n$  (rysunek 2.12) lub  $n$ – $m$  (rysunek 2.13).

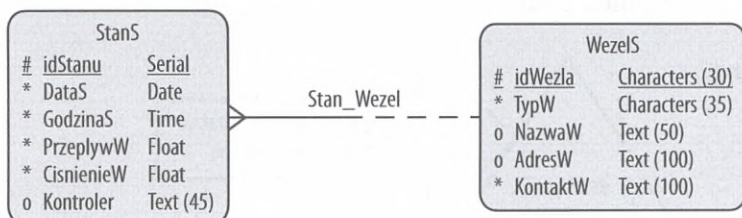
Związek typu 1–1 oznacza, że każdy element encji pierwszej (*StanS*) może się łączyć z co najwyżej jednym elementem encji drugiej (*Alarm*) oraz że każdy element encji drugiej może się łączyć z co najwyżej jednym elementem encji pierwszej. Występowanie w związku może być obligatoryjne (obowiązkowe — linia ciągła) lub opcjonalne (nieobowiązkowe — linia przerywana). Przykładowo, na rysunku 2.6 *StanS* może, ale nie musi, mieć przypisany alarm w BD (połączenie linią przerywaną). Jeżeli natomiast *Alarm* został już zarejestrowany, to koniecznie musi być związany ze stanem, którego dotyczy (połączenie linią ciągłą).

Rysunek 2.11. Związek typu 1–1



Źródło: Opracowanie własne.

Związek typu 1– $n$  oznacza, że każdy element encji pierwszej (*WezelS* na rysunku 2.12) może się łączyć z dowolną liczbą elementów encji drugiej (*StanS* na rysunku 2.12), ale każdy element encji drugiej może się łączyć co najwyżej z jednym elementem encji pierwszej. Występowanie encji *StanS* w związku jest obligatoryjne — jeżeli rejestrowany jest stan sieci, należy podać, którego węzła sieci dotyczy. Natomiast występowanie encji *WezelS* w związku jest opcjonalne — można wprowadzić do BD dane o węzle sieci, którego żaden stan nie był jeszcze zarejestrowany (rysunek 2.12).

Rysunek 2.12. Związek typu 1– $n$ 

Źródło: Opracowanie własne.

Ostatni typ związku  $n$ – $m$  oznacza, że każdy element encji pierwszej (*StanS* na rysunku 2.13) może się łączyć z dowolną liczbą elementów encji drugiej (*Decyzja* na rysunku 2.13) i w drugą stronę zachodzi ta sama sytuacja. Występowanie obydwu encji *StanS* w związku jest opcjonalne, natomiast występowanie encji *Decyzja* jest obligatoryjne.

Rysunek 2.13. Związek typu  $n$ – $m$ 

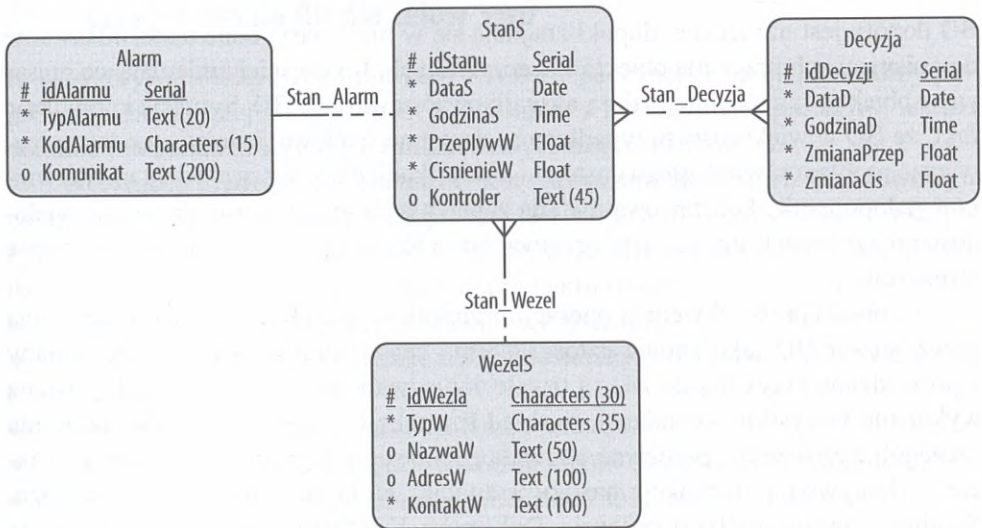
Źródło: Opracowanie własne.

Suma wszystkich występujących w strukturze encji i ich wzajemnych powiązań tworzy model konceptualny BD (rysunek 2.14).

Model konceptualny BD wymaga przetestowania pod kątem jego poprawności semantycznej i strukturalnej. Na podstawie sprawdzonej koncepcji modelu bazy danych generowany jest **model fizyczny**. Różni się on od modelu konceptualnego BD tym, że uwzględnia specyfikę środowiska technicznej realizacji projektu (wymaga zdefiniowania platformy bazodanowej, na której zostanie zrealizowany projekt) oraz wyróżnia (oprócz kluczy podstawowych) także klucze obce, generujące fizyczne połączenia między tabelami BD (rysunek 2.15).

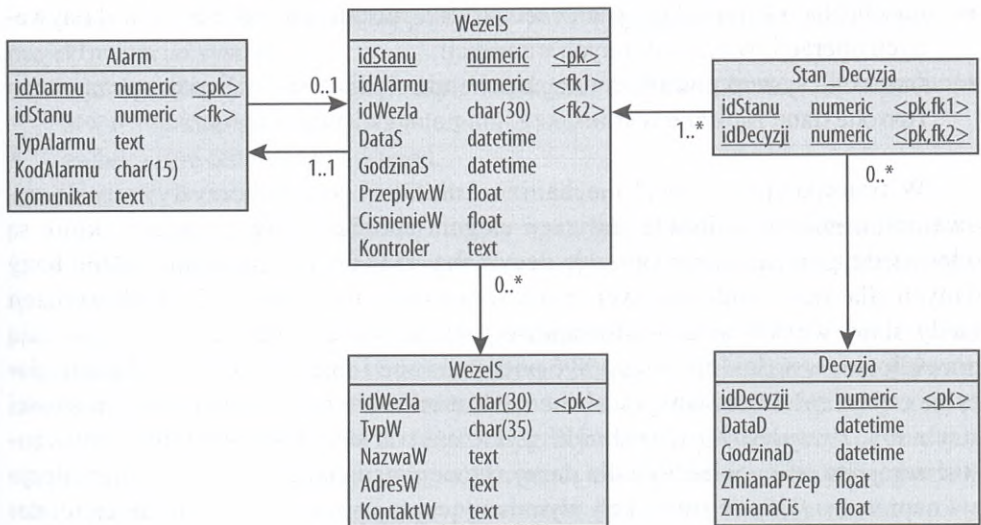
Prezentowany na rysunku 2.15 model fizyczny BD został wygenerowany dla środowiska MS SQL Server 2008. Należy zwrócić uwagę, że związek typu  $n$ – $m$  z modelu konceptualnego (rysunek 2.13) został zamieniony na dwa związki typu 1– $n$  oraz dodatkową tabelę *Stan–Decyzja*, w której obowiązkowo na liście atrybutów pojawiają się klucze obce, będące kluczami podstawowymi w tabelach, pozostającymi w związku typu  $n$ – $m$  (*idStanu*–klucz podstawowy tabeli *StanS* i *idDecyzji*–klucz podstawowy tabeli *Decyzja*).

Rysunek 2.14. Model konceptualny BD



Źródło: Opracowanie własne.

Rysunek 2.15. Model fizyczny BD



Źródło: Opracowanie własne.

Poprawny model fizyczny jest podstawą do wygenerowania gotowej struktury bazy danych, która może być użytkowana na odpowiednio przygotowanej platformie.

### 2.4.3. Przetwarzanie transakcyjne w BD

BD dopóty jest użyteczna, dopóki znajdują się w niej wiarygodne dane, adekwatne do opisywanych przez nią obiektów rzeczywistych. Im częściej zmieniają się opisywane obiekty, tym częściej należy aktualizować zawartość BD. Sytuację komplikuje fakt, że BD w większości przypadków są dostępne sieciowo, należy więc także rozwiązać problem równoległego dostępu i modyfikacji danych przez wielu użytkowników jednocześnie. Podstawową metodą zapewniania integralności danych w wielodostępnych bazach danych jest przetwarzanie ich w operacjach noszących nazwę transakcji.

**Transakcja** to sekwencja operacji przeprowadzanych na danych, traktowana przez serwer BD jako spójna całość. Istotną cechą transakcji jest to, że zmiany wprowadzane przez nią do BD są trwale zapisywane tylko wówczas, gdy zostaną wykonane wszystkie wchodzące w skład transakcji operacje. Działanie takie ma szczególne znaczenie, ponieważ pozwala zachować integralność BD, gdy w trakcie wykonywania transakcji nastąpi sytuacja, która tej integralności zagraża. Zgodnie z zasadą ACID [Coulouris, Dollimore, Kindberg 1998], każda transakcja musi być:

- niepodzielna — transakcja może być wykonana tylko w całości albo wcale;
- spójna — po wykonaniu transakcji system zawsze będzie spójny, czyli nie zostaną naruszone żadne zasady integralności danych;
- niezależna — transakcja jest przetwarzana niezależnie od innych wykonywanych operacji, w tym od innych transakcji;
- trwała — system potrafi zawsze uruchomić się i udostępnić spójne oraz nienaruszone dane zapisane w transakcji, np. po nagłej awarii zasilania.

W transakcyjnych SWD mechanizm transakcji jest wykorzystywany do zagwarantowania prawidłowej realizacji ciągom automatycznych operacji, które są odpowiedzią na zaistniałą sytuację decyzyjną. Odwołajmy się do przykładu bazy danych dla sieci wodociągowej, przedstawionego na rysunku 2.15. W sytuacji kiedy stany węzłów sieci wodociągowej, rejestrowane w tabeli *StanS*, wykazują prawidłowe wartości, nie muszą być podejmowane i rejestrowane w BD żadne decyzje czy opisy aktywowanych alarmów. Natomiast w razie awarii, kiedy wartości atrybutów *PrzepływW* i *CisnienieW* wykazują wartości nieprawidłowe, musi zostać zarejestrowany właściwy dla danej sytuacji stan alarmowy oraz podjęta decyzja naprawcza. Użycie transakcji wymuszającej rejestrację i alarmu, i podjętej decyzji w momencie powstania awarii, zabezpieczy SWD przed generowaniem niekompletnych danych, utrudniających lub wręcz uniemożliwiających jego prawidłowe działanie.

## 2.4.4. Funkcjonalne i eksploatacyjne aspekty przygotowania BD dla celów SWD

Proces przygotowania BD na potrzeby SWD rozpoczyna się na etapie analizy projektowej obejmującej takie elementy, jak: analiza i reinżynieria procesów, analiza dokumentów, analiza wymagań funkcjonalnych użytkowników.

W sytuacji kiedy przedsiębiorstwo jest zarządzane procesowo, podstawowym elementem poprzedzającym każdy projekt jest analiza (a jeżeli zachodzi taka potrzeba — reinżynieria) realizowanych procesów. Analiza procesu pozwala odpowiedzieć na wiele pytań, bardzo istotnych dla projektu BD oraz całego systemu SWD:

- Jaki powinien być zakres funkcjonalności systemu?
- Jakie dane, komu, kiedy i gdzie należy dostarczyć, aby proces mógł być zrealizowany?
- Jakie dane powstają w wyniku realizacji procesu?
- Jakie dokumenty krążą między poszczególnymi etapami realizacji procesu?
- Co można zrobić w celu podniesienia efektywności realizacji procesu?

Odpowiedzi na te pytania są elementem wyjścia dla projektu modelu logicznego BD. Należy je jeszcze uzupełnić analizą dokumentów (np. raportów, zestawień), które koniecznie muszą być uwzględnione w projektowanym SWD oraz analizą wymagań funkcjonalnych, zgłoszonych przez przyszłych użytkowników systemu.

Na poziomie modelu fizycznego BD należy uwzględnić aspekty eksploatacyjne projektowanego systemu:

- wymagania eksploatacyjne, czyli wydajność, skalowalność, niezawodność, otwartość i bezpieczeństwo systemu;
- architekturę techniczną systemu.

Na podstawie przeprowadzonej analizy wymagań eksploatacyjnych można podjąć decyzję dotyczącą doboru optymalnego rozwiązania technicznego projektowanego systemu.

Inne problemy należy rozwiązać w przypadku integracji SWD z istniejącymi w przedsiębiorstwie systemami transakcyjnymi. Wtedy SWD będzie wykorzystywał istniejące już bazy danych, a analiza polega na ocenie zgodności funkcjonujących rozwiązań z wymaganiami SWD. Jeżeli integracja dotyczy pojedynczej bazy transakcyjnej, na poziomie projektu wprowadzane są niezbędne modyfikacje jej struktury oraz budowana jest warstwa przetwarzania danych i interfejsy użytkownika. Projekt komplikuje się w sytuacji, gdy mamy do czynienia z kilkoma scentralizowanymi BD — transakcyjnymi lub rozproszonymi.

**System rozproszony** jest zbiorem powiązanych ze sobą niezależnych komputerów (przeważnie niejednorodnych), które dla użytkownika są widoczne jako jeden [Silberschatz, Korth, Sudarshan 2000]. Znamienne dla systemów rozproszonych jest, że użytkownik widzi je tak samo jak system scentralizowany, mimo że przetwarza-

nie informacji w przypadku systemu rozproszonego odbywa się na kilku lub kilkadziesiąciu powiązanych ze sobą sprzętowo i programowo jednostkach komputerowych. Stanowi to kontrast dla sieci, w której użytkownik jest świadomy obecności komputerów o różnych lokalizacjach oraz udostępnianych zasobach.

Do zalet systemów rozproszonych należy zaliczyć [Silberschatz, Korth, Sudarshan 2000]: relatywnie niskie koszty utrzymania (porównując moc obliczeniową do ceny jej uzyskania), szybkość przetwarzania danych, niezawodność (awaria jednego z urządzeń nie wpływa na możliwość pracy pozostałych jednostek), dostępność (zapewnienie dostępu do systemu użytkownikom o różnej lokalizacji geograficznej), możliwość rozbudowy (zwiększanie mocy systemu poprzez jego rozbudowę), skalowalność (możliwość elastycznego dostosowywania się do zmiennych wymagań użytkowników). Za wady należy uznać [Silberschatz, Korth, Sudarshan 2000]: złożoność oprogramowania, zależność wydajności systemu od jakości połączeń sieciowych, mniejsze bezpieczeństwo zasobów (więcej potencjalnych luk programowych w systemie jako rezultat dzielenia zasobów). Przy obecnych możliwościach technologicznych zalety systemów rozproszonych znacznie rekompensują ich wady.

Systemy rozproszone mogą być konfigurowane i optymalizowane w zależności od potrzeb, jakie mają realizować. Komputery pracujące w systemie mogą dzielić między sobą: urządzenia, dane i zasoby. Szczególnym rodzajem systemu ze współdzielonymi danymi jest rozproszona BD. Według definicji [Connolly, Begg 2004] jest to baza danych, w której informacje nie są przechowywane w jednym miejscu, ale są rozmieszczone na komputerach połączonych ze sobą w sieć [Connolly, Begg 2004]. Zgodnie z regułą systemu rozproszonego baza jest widoczna dla użytkownika jako jeden obiekt wirtualny. Fizycznie składa się jednak z odseparowanych BD, zarządzanych przez serwery pracujące w różnorodnych środowiskach sprzętowych lub/i programowych [Connolly, Begg 2004]. W każdym przypadku naczelnym kryterium jest bezpieczeństwo danych — przede wszystkim ochrona przed utratą danych w wyniku błędów projektowych lub celowego działania.

Wybór ostatecznego rozwiązania będzie zawsze zależny od wyników przeprowadzonej analizy, której celem jest wyspecyfikowanie wszystkich wymagań funkcjonalnych i eksploatacyjnych tworzonego SWD, a następnie dobranie optymalnego rozwiązania bazodanowego.

## **2.4.5. Przykład przetwarzania faktów i reguł decyzyjnych w BD**

Model SWD (rysunek 2.16), przetwarzający fakty i reguły decyzyjne w oparciu o dane zgromadzone w BD, może wspierać nadzorowanie pracy sieci wodociągowej. Jak wynika z rysunku 2.16, BD gromadzi dane dostarczane na bieżąco przez system monitoringu stanów sieci wodociągowej. Dane te są następnie pobierane i analizowane przez moduł przetwarzania faktów i generowania reguł decyzyjnych. Analiza



polega na wielowymiarowym przetwarzaniu zgromadzonych w bazie danych, obejmujących: stany sieci wodociągowej, diagnostykę awarii oraz podjęte decyzje naprawcze, a także tworzące fakty SWD.

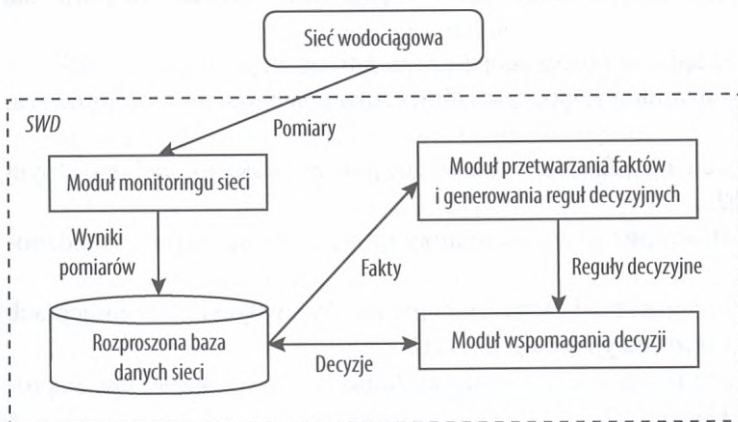
Techniki analityczne, takie jak np. sieci neuronowe czy drzewa decyzyjne, na podstawie uzyskanych faktów generują reguły decyzyjne, które zawierają w sobie wskazanie konkretnych decyzji w przypadku zaistnienia określonych wartości faktów:

IF 2000  $\leq$  *PrzeptywW* < 3000 AND 500  $\leq$  *CisnienieW* < 700 THEN *Decyzja1* {*Włączyć moc dodatkową bieżącej pompy*}.

IF 1000  $\leq$  *PrzeptywW* < 2000 AND 250  $\leq$  *CisnienieW* < 500 THEN *Decyzja2* {*Włączyć pompę dodatkową*}.

IF *PrzeptywW* < 1000 AND *CisnienieW* < 250 THEN *Decyzja3* {*Aktywować stan alarmowy. Rozpocząć działania naprawcze*}.

Rysunek 2.16. SWD z transakcyjną bazą danych



Źródło: Opracowanie własne na podstawie: [Rojek 2007, s. 188].

Wygenerowane reguły są wykorzystywane przez moduł wspomagania decyzji do przygotowywania scenariuszy decyzyjnych, optymalizujących pracę sieci wodociągowej w zakresie utrzymania stałych wartości przepływu i ciśnienia wody w sieci.

Przykład, wykorzystany w bieżącym rozdziale, jest oparty na charakterystyce inteligentnego SWD sieci wodociągowych (ISWDSC) [Rojek 2007], w którym BD sieci wodociągowej jest częścią systemu rozproszonych branżowych BD przedsiębiorstwa sieci wodociągowej, zasilanych przez system monitoringu sieci wodociągowej. Dostarczane do BD dane charakteryzują stan wybranych punktów sieci wodociągowej. Na podstawie analizy tych danych modele analityczne są w stanie prognozować obciążenie oraz awaryjność sieci wodociągowej. Następnie SWD, na podstawie opracowanych modeli prognostycznych, tworzy zoptymalizowane scenariusze sterowania siecią wodociągową. Wspomagają one użytkowników systemu

w szybkim reagowaniu na zmiany parametrów, które pojawiają się w sieci wodociągowej oraz zapobieganiu nieprawidłowościom pracy sieci.

## 2.5. Pytania i zadania kontrolne

### 2.5.1. Pytania kontrolne

1. Jakie rodzaje modeli matematycznych można wyróżnić z punktu widzenia ich zastosowania w prognozowaniu i symulacji?
2. Jakie znaczenie mają metody heurystyczne w efektywnym wyszukiwaniu dobrych rozwiązań decyzyjnych?
3. Dlaczego techniki iteracyjne odgrywają znaczącą rolę w eksperymentach symulacyjnych?
4. Jaką funkcję pełnią transakcyjne BD w SWD?
5. Jak należy przygotować projekt budowy i wdrożenia transakcyjnej BD dla SWD?
6. Z jakich etapów składa się proces projektowania transakcyjnej BD dla SWD?
7. Jakiego typu powiązania występują między encjami w modelu konceptualnym bazy danych?
8. Co to jest redundancja danych i dlaczego jest zjawiskiem niekorzystnym w bazach danych?
9. Jakie metody są stosowane na etapie analizy projektowej systemów bazodanowych?
10. Jakie problemy funkcjonalne i eksploatacyjne należy uwzględnić w projektach systemów SWD z transakcyjną bazą danych?
11. W jakich sytuacjach konieczne jest wykorzystanie scentralizowanej lub rozproszonej BD w SWD?

### 2.5.2. Przykładowe zadania

1. Zaprojektuj wyposażenie laboratorium komputerowego za pomocą zbudowanego modelu optymalizacyjnego w dodatku MS Excel Solver, ustalając ceny zakupu oraz liczbę komputerów, drukarek, monitorów oraz pakietów oprogramowania, wiedząc, że dysponujemy środkami finansowymi na wyposażenie laboratorium komputerowego w kwocie 1 mln zł, a reszta z wydatkowanych kwot powinna wynosić 0 zł.
2. Zastosuj metodę optymalizacji do rozwiązania problemu komiwojażera, wiedząc, że dostawca pizzy musi porozwozić towar do trzech klientów zlokalizowanych w punktach  $A$ ,  $B$  i  $C$ . Komiwojażer musi objechać te 3 punkty możliwie

najtańszą trasą. Jaką kolejność powinien obrać, aby straty wynikające z objazdu były minimalne? Dana jest macierz kosztów w postaci:

$$\mathbf{M} = \begin{matrix} & \begin{matrix} A & B & C \end{matrix} \\ \begin{matrix} A \\ B \\ C \end{matrix} & \begin{bmatrix} 0 & 11 & 10 \\ 13 & 0 & 12 \\ 20 & 7 & 0 \end{bmatrix} \end{matrix}.$$

Na przykład, liczba 20 oznacza, że koszt przejazdu z  $C$  do  $A$  wynosi 20 jednostek.

Macierz  $\mathbf{M}$  nie jest macierzą symetryczną, tzn. np. koszt z  $A$  do  $B$  wynosi 11, natomiast z  $B$  do  $A$  — 13 jednostek. Zadanie komiwojażera z macierzą symetryczną jest szczególnym przypadkiem ogólnego zadania komiwojażera. W analizowanym przykładzie możliwych jest 6 rozwiązań dopuszczalnych. Rozwiązanie optymalne jest jednym z rozwiązań dopuszczalnych. Wyznacz teraz rozwiązania dopuszczalne i optymalne.

3. Zastosuj kontrapozycję całkowitą do precyzyjnego uporządkowania obiektów w procesie magazynowania surowców, odpowiednio je grupując według dostawców, czasów dostaw i czasów magazynowania.
4. Ustal prawdopodobieństwo, że losowo wybrany detal z magazynu jest wadliwy, zakładając, że detal ten wytwarzają fabryki  $A$ ,  $B$  i  $C$  mające udziały w rynku odpowiednio: 40%, 25% i 15% całej produkcji, a wadliwość dla poszczególnych fabryk wynosi odpowiednio 4%, 3% i 1,5% (zob. punkt 2.3.3).
5. Zaproponuj zbiór reguł decyzyjnych dla SWD, wspierającego optymalizację planowania dostaw do magazynu w taki sposób, aby minimalizować stan magazynu i zagwarantować ciągłość sprzedaży.
6. Zaprojektuj model konceptualny BD dla SWD, wspierającego optymalizację planowania dostaw do magazynu w taki sposób, aby minimalizować stan magazynu oraz zagwarantować ciągłość sprzedaży. Zbiór reguł dla tego systemu został zaprojektowany w zadaniu 6.
7. Zaproponuj zbiór reguł decyzyjnych dla SWD, wspierającego optymalizację czasu naprawy poprzez lepsze harmonogramowanie pracy personelu serwisu.
8. Zaprojektuj model fizyczny BD dla SWD, wspierającego optymalizację czasu naprawy poprzez lepsze harmonogramowanie pracy personelu serwisu. Zbiór reguł dla tego systemu został zaprojektowany w zadaniu 7.

# Interfejs użytkownika SWD

Rozdział wprowadza w zagadnienia definiowania cech użytkowych SWD, od których zależy efektywność i poprawność jego zastosowań w procesie komunikacji z użytkownikiem. Dlatego też dla decydenta funkcjonowanie interfejsu użytkownika jest podstawowym weryfikatorem pracy całego SWD [Olszak 2000]. Cechy użytkowe SWD są prezentowane w dwóch ujęciach:

- 1) jako wymagania funkcjonalne (dotyczące zakresu i sposobu użytkowania),
- 2) jako wymagania eksploatacyjne (parametry techniczno-ekonomiczne), które będą decydować o technologicznej i ekonomicznej efektywności SWD oraz możliwościach jego rozwoju.

Złożoność omawianego problemu wynika z faktu, że SWD musi w określonych proporcjach spełniać wiele różnorodnych, niekiedy wzajemnie sprzecznych, wymagań funkcjonalnych, eksploatacyjnych oraz prawnych. Ze względu na rozległą skalę złożoności i różnorodności SWD projekt interfejsu dla skomplikowanego (np. hybrydowego) SWD może przybrać postać unikatowego rozwiązania lub staje się produktem handlowym o szerokich możliwościach zastosowania.

Pojęcie interfejsu użytkownika jest różnie definiowane. Interfejs użytkownika (ang. *user interface*) jest to w technice część urządzenia odpowiedzialna za interakcję z użytkownikiem. Człowiek nie jest zdolny do bezpośredniej komunikacji z maszynami. Aby było to możliwe, urządzenia są wyposażone w odpowiednie urządzenia wejścia-wyjścia tworzące interfejs użytkownika:

- interfejs tekstowy — urządzenie wejściowe to klawiatura, a wyjściowe to drukarka znakowa lub wyświetlacz w trybie znakowym;
- interfejs graficzny — wejście to urządzenie wskazujące (np. myszka), a wyjściowe to wyświetlacz graficzny;
- interfejs strony internetowej — wejście i wyjście jest realizowane poprzez stronę internetową wyświetlaną w przeglądarce internetowej.

W informatyce najczęściej jako interfejs użytkownika rozpatruje się część oprogramowania zajmującą się obsługą urządzeń wejścia/wyjścia przeznaczonych do interakcji z użytkownikiem. W komputerach na ogół za obsługę większości funkcji interfejsu użytkownika odpowiada system operacyjny, który narzuca standaryzację wyglądu różnych aplikacji. Zwykli użytkownicy postrzegają oprogramowanie wyłącznie poprzez interfejs użytkownika<sup>1</sup>. Według L. Barfielda [1993] interfejs użytkownika jest to zespół narzędzi programowych i sprzętowych, który umożliwia użytkownikowi realizację jego zadań. Definicja ta nie obejmuje jednak całej złożoności procesów zachodzących między człowiekiem a komputerem, zwłaszcza zaś w dobie przyrastających gwałtownie multimedialnych możliwości technologii informacyjnych, dzięki którym nawet komunikacja ludzi niepełnosprawnych z komputerem staje się realna.

Ciekawe i nowatorskie spojrzenie na interfejs użytkownika SWD jako specyficzny pomost między decydem a inżynierią oprogramowania proponuje C.M. Olszak [2000], wprowadzając pojęcie inteligentnego, multimedialnego interfejsu, którego funkcjonowanie powinno ściśle korespondować z rolą, jaką odgrywa mentalny model decydenta w interakcji z systemem komputerowym.

### 3.1. Wymagania funkcjonalne

Celem projektowania SWD jest budowa narzędzi wspomagania decyzji w postaci otwartych prototypów lub w formie profesjonalnych rozwiązań handlowych przeznaczonych do szerokiego zastosowania. Od SWD wymaga się spełnienia trzech warunków użytkowych; powinny być:

- 1) adekwatne (zawarte w nich dane i reguły wnioskowania powinny odpowiadać potrzebom operacyjnym, taktycznym i strategicznym);
- 2) dostępne (komunikacja z SWD powinna być prowadzona w sposób gwarantujący bezpieczną i jednoznaczną interpretację komunikatów — poprawny proces semiozy [Oleński 2003], działań wykonywanych przez SWD oraz
- 3) elastyczne (w zakresie poziomu kompetencji i uprawnień użytkownika — operatora SWD).

Funkcjonalność interfejsu na wysokim poziomie adekwatnym do potrzeb użytkownika może być osiągnięta m.in. przez uwzględnienie w jego cechach mentalnego modelu decydenta, a nie projektanta. Wynika to z faktu, iż decydentów interesują zadania i cele organizacji, które mają być wspomagane przez system, natomiast uwaga projektanta skupia się na ogół na obiektach, ich wizualnej prezentacji na ekranie czy metodach interakcji w systemie. W interfejsie powinny być szeroko stosowane analogie do obiektów fizycznych z wykorzystaniem takich, które naśladują opisywane cechy i ich działanie, a także metafory ze świata zewnętrznego, które są

<sup>1</sup> [http://it.pwshz-ns.edu.pl/~ajanisiz/ftp/materialy\\_do\\_projektowania/p-09-interfejs\\_uzytkownika.pdf](http://it.pwshz-ns.edu.pl/~ajanisiz/ftp/materialy_do_projektowania/p-09-interfejs_uzytkownika.pdf).

proste i relatywnie łatwe do zapamiętania. Należy uwzględniać przyzwyczajenia i oczekiwania decydenta, co potwierdzają w praktyce udane próby zastosowań systemu ekspertowego dla rolników Ma'ayan [McCown 2002]. Ważne jest także replikowanie języka decydenta, stosowanie jego stylów interakcji czy ikon zgodnych z jego wyobrażeniami. Należy także stosować właściwe sprzężenia zwrotne w zakresie dostarczania informacji o stanie i zmianach systemu, wizualnego przedstawiania wyników działania systemu oraz dostarczaniu klarownych i konkretnych informacji o błędach popełnianych przez decydenta [Olszak 2000, s. 102–103]. Rozwinięcia analizy podjętej problematyki należy poszukiwać m.in. w pracach E. Niedzielskiej, H. Dudycz, M. Dyczkowskiego [2005] czy też H. Dudycz [2007], M. Sikorskiego, L. Garnika i K. Redlarskiego [2011] oraz M. Sikorskiego [2010a].

### 3.1.1. Modelowanie wymagań funkcjonalnych

Dla uzyskania efektywnych interfejsów użytkowników, spełniających ich oczekiwania biznesowe, jest niezbędne stosowanie podejścia ewolucyjnego, w którym poprzez wykorzystanie powszechnie uznanych standardów, np. Agile Modelling (AM), następuje ulepszanie rozwiązań w wyniku ciągłej interakcji projektantów z użytkownikami aplikacji B. Baxleya [Constantine, Lockwood 2001] o nazwie **Uniwersalny Model Interfejsu Użytkownika** aż do momentu, kiedy ocenia się, że kolejne zgłaszane pomysły nie wnoszą nic konstruktywnego do obsługi programowych interfejsów SWD.

Jedną z zalecanych obecnie metod doskonalenia procesu budowy interfejsu graficznego jest prototypowanie jako sposób implementowania — wspólnie z użytkownikami aplikacji — zgłaszanych idei, pozwalający jednocześnie na ocenę jakości rozwiązań we wczesnych stadiach procesu tworzenia koncepcji interfejsu.

Wśród podstawowych zasad, które powinny być uwzględnione w projektowaniu interfejsów ze względu na wymagania funkcjonalne stawiane SWD, można wymienić:

- adekwatność SWD do rozwiązywanych problemów decyzyjnych;
- elastyczność SWD rozumianą jako łatwość dostosowania SWD do potrzeb użytkownika;
- architekturę otwartą na nowe metadane wprowadzane przez użytkownika;
- prostotę obsługi zapewniającą szybki dostęp do najczęściej wykorzystywanych funkcji;
- zachowanie funkcjonalności we wszystkich platformach sprzętowo-programowych dostępnych dla użytkownika;
- zagwarantowanie potrzeb użytkowników niedowidzących (możliwość wyboru szablonu o zwiększonym kontraście i rozmiarze czcionek);
- umożliwienie modyfikacji sposobu wyświetlania poszczególnych komponentów w taki sposób, aby użytkownik mógł zmieniać aranżację ekranu głównego i innych obszarów roboczych, co zapewni mu szybszy dostęp do wykorzystywanych przez niego funkcji [Ambler 2002], [Constantine, Lockwood 1999–2002].

C.M. Olszak [2000] potwierdza, że o efektywności pracy interfejsu z punktu widzenia użytkownika powinny decydować takie czynniki, jak: czas uczenia się obsługi przez użytkownika, wspomaganie go w różnorodnych zadaniach decyzyjnych, prostota obsługi eliminująca liczbę popełnianych błędów do minimum, pomoc w razie potrzeby na życzenie, adaptacyjność w sensie uwzględniania różnego poziomu użytkowników czy też nieduża liczba szczegółów technicznych, na których użytkownik musi koncentrować uwagę.

### 3.1.2. Język specyfikacji wymagań funkcjonalnych

Język specyfikacji wymagań funkcjonalnych SWD powinien być dobrany szczególnie starannie, ponieważ SWD jest zazwyczaj traktowany jako systematycznie rozbudowywana i modyfikowana technologiczna platforma podejmowania decyzji. Z punktu widzenia poziomu szczegółowości wymagania stawiane SWD można podzielić na:

- wymagania użytkownika (ogólna definicja wymagań) co do funkcjonowania systemu, wyrażone głównie za pomocą języka dziedziny zastosowań;
- wymagania systemowe (specyfikacja wymagań), przedstawiające szczegółowy opis funkcji, usług i ograniczeń SWD, wyrażone za pomocą terminologii wykorzystywanej w projektowaniu systemów informatycznych.

W metodach formalnych tworzenia oprogramowania interfejsów dąży się do takiego poziomu szczegółowości określenia wymagań, który pozwoliłby na ustalenie wszelkich aspektów funkcjonowania programu i docelowo — na automatyczne tworzenie kodu programowego interfejsu.

**Wymagania użytkownika** często rozumie się jako ogólne wskazania (np. „system powinien być odporny na ingerencję z zewnątrz” czy też „sposób naliczania wynagrodzeń pracowników powinien być zgodny z obowiązującymi przepisami”). Transformacja takich ogólnych wymagań w specyfikację dotyczące oprogramowania, w tym interfejsu użytkownika, jest pracą wymagającą dużego doświadczenia, inwencji oraz dobrej znajomości dziedziny zastosowań SWD.

Istotne jest używanie — przy określaniu specyfikacji wymagań na SWD — konkretnych, precyzyjnych, jednoznacznych i podlegających weryfikacji pojęć i działań [Glushko, Lindsay 2009]. Stworzenie modelu SWD w ramach rozważanego obszaru zastosowań może być przydatne w uściśleniu kwantyfikowalnych miar dla różnych kategorii wymagań, np.:

- dla szybkości działania: liczba transakcji w jednostce czasu;
- dla czasu reakcji na zdarzenie: czas odświeżania strony;
- dla oceny sposobu użycia: czas konieczny na przeszkolenie obsługi, liczba i struktura stron (okien) pomocy;
- dla niezawodności działania: przeciętny czas od awarii do awarii, prawdopodobieństwo wystąpienia niedostępności;

- dla odporności systemu: odsetek zdarzeń stanowiących przyczynę awarii, prawdopodobieństwo zniszczenia danych w trakcie awarii;
- dla przenośności danych: procent instrukcji zależnych od systemu operacyjnego lub bazy danych, liczba uwzględnionych systemów operacyjnych (baz danych).

W celu usunięcia wieloznaczności i ułatwienia konkretyzacji wymagań stosuje się ich specyficzne sposoby formalizacji oraz standaryzacji, przy czym bardziej formalizuje się zasady zapisu specyfikacji systemowych niż użytkownika. Wymagania użytkownika, nawet takie, które są wyspecyfikowane w języku potocznym, także są uwzględniane i artykułowane, jeśli stosowane są w ich wyrażaniu pewne zasady formalne, np.: ustalony format zapisu, rozróżnienie wymagań funkcjonalnych od niezwiązanych z funkcjonalnością, stosowanie spójnej i precyzyjnej terminologii, wyraźne rozgraniczenie wymagań od ich uzasadnień. Sformalizowana treść wymagań staje się częstokroć bardzo ważna po upływie pewnego czasu eksploatacji, kiedy zachodzi potrzeba dokonania znacznej modyfikacji wymagań [Leffingwell, Widrig 2003], [Murphy 2009], [Sommerville 2003].

Zapisy wymagań w postaci formularzy, interfejsów lub języków formalnych w sposób oczywisty zakładają dość szczegółowy poziom wiedzy na temat projektowanego systemu oraz uprzednie podjęcie wielu istotnych decyzji projektowych (wybór modelu oprogramowania, konkretnych funkcji lub struktur danych). Z tego powodu w modelach budowy oprogramowania fazy określania wymagań oraz projektowania i implementacji tworzą powtarzające się cykle dopóty, dopóki użytkownik interfejsu nie zostanie usatysfakcjonowany.

**Narzędzia CASE** [Biddle, Noble 2003], które wspomagają proces określania wymagań, są najczęściej związane z konkretną, ustaloną notacją lub metodyką postępowania. Istniejące narzędzia CASE pomagają tworzyć diagramy i dokonywać transformacji między nimi. Typowe dla fazy określania wymagań jest tworzenie słownika stosowanej terminologii w postaci BD związanych z realizowanym projektem. BD o odpowiedniej strukturze pozwala np. na śledzenie konsekwencji dokonywanych zmian w wymaganiach. Możliwe bywa także automatyczne tworzenie raportów na podstawie diagramów i słowników. Ostateczny dokument określający wymagania powinien powstawać zgodnie z precyzyjnymi zasadami mającymi postać standardów (np. US DoD, IEEE). W opinii I. Sommerville'a od opisu wymagań oczekuje się, że będzie zawierał następujące elementy [Sommerville 2003]:

- przedmowa (docelowi adresaci opisu, historia wersji);
- wstęp (motywacja powstania systemu, ogólny opis systemu i środowiska, w którym ma on funkcjonować);
- słownik (słownik pojęć użytych w opisie wymagań);
- definicja wymagań użytkownika (opis usług i wymagań niefunkcjonalnych zrozumiały dla klientów, współpraca z innymi systemami).



W ostatecznym opisie wymagań powinna być uwzględniona ogólna architektura systemu (podsystemy, ponownie wykorzystywane komponenty) oraz specyfikacja wymagań systemowych (wymagania funkcjonalne i нефункционалне scharakteryzowane szczegółowo).

Dobrą ilustracją języka specyfikacji wymagań funkcjonalnych jest sposób przygotowania specyfikacji wymagań systemu według protokołu IBM Rational Requisite Pro<sup>2</sup>. Protokół ten jest narzędziem ułatwiającym zespołom projektowym terminowe tworzenie i analizowanie wymagań aplikacji, projektów działania oraz zarządzanie nimi. System specyfikacji jest łatwy w obsłudze w procesach zarządzania wymaganiami i projektami działania. Pozwala usprawnić komunikację i współpracę w ramach zespołów oraz ograniczyć ryzyko związane z projektami. Zapewnia połączenie obsługi dokumentów programu Microsoft Word z rozbudowanymi funkcjami obsługi baz danych w celu zwiększenia skuteczności analizy i badania wymagań. Zawiera mechanizmy umożliwiające zespołom rozpoznanie skutków zmian oraz informowanie wszystkich uczestników prowadzonego projektu o najbardziej aktualnych wymaganiach w celu zapewnienia spójności [Sommerville 2003], [Lefingwell 2003].

### 3.1.3. Przykład specyfikacji wymagań funkcjonalnych dla potrzeb logistycznych

Klasyczne modele analityczne, obecnie wykorzystywane we wspomaganiu planowania przepływu dóbr, nie są już wystarczająco szczegółowe dla potrzeb koordynacji przepływu produktów między ogniwami łańcucha dostaw. Coraz częściej do analizy procesów zachodzących w łańcuchach dostaw jest wykorzystywana **symulacja**. Nowym, często stosowanym podejściem w symulacyjnej analizie przepływu produktów w łańcuchu dostaw jest podejście interaktywne, co pozwala szczegółowo śledzić wszystkie wielkości charakteryzujące analizowaną konfigurację łańcucha oraz wielkości przepływów między ogniwami. Ponadto, dzięki specjalnym interfejsom użytkownika, umożliwia ono uczestniczenie w symulacji rozproszonym zespołom użytkowników.

Celem zaprezentowania pierwszego przykładu było znalezienie rozwiązania dla potrzeb zaplanowania tras gwarantujących realizację zamówień, których łączna długość będzie najmniejsza. Wyniki pierwszych czterech okresów I etapu symulacji przedstawiono w tablicy 3.1. Wiersze 1–6 obrazują wyniki zespołów biorących udział w symulacji. Wiersz 7. zawiera średnią z wyników wszystkich zespołów. Ostatni, 8 wiersz, przedstawia rozwiązanie wzorcowe, uzyskane za pomocą metody „podział–trasa”.

<sup>2</sup> [www-142.ibm.com/software/products/pl/pl/reqpro/](http://www-142.ibm.com/software/products/pl/pl/reqpro/).

Tablica 3.1. Wyniki I etapu symulacji: planowanie dystrybucji na podstawie zamówień

Lp.	Ze-spół	okres 1		okres 2		okres 3		okres 4		Łączna długość tras	Suma dostarczonych produktów	Wskaźnik (km/ /szt.)
		dłu-gość tras	licz-ba tras	dłu-gość tras	licz-ba tras	dłu-gość tras	licz-ba tras	dłu-gość tras	licz-ba tras			
1.	01	92,88	4	44,68	2	73,16	3	89,01	4	299,73	41 520	0,0072
2.	02	97,06	4	44,68	2	70,50	3	85,28	4	297,52	41 520	0,0072
3.	03	101,36	4	53,59	2	88,68	4	89,05	4	332,68	43 200	0,0077
4.	04	97,81	4	44,68	2	75,35	3	85,28	4	303,12	41 520	0,0073
5.	05	107,88	4	54,23	2	74,59	3	80,58	4	317,28	41 520	0,0076
6.	06	98,78	4	52,88	2	84,17	4	80,58	4	316,42	41 520	0,0076
7.	AVG	<b>99,295</b>		<b>49,123</b>		<b>77,741</b>		<b>84,963</b>		<b>311,125</b>		<b>0,00743</b>
8.	TR	82,61	4	44,68	2	72,50	3	73,97	4	273,77	41 520	0,0066

Ź r ó d ł o: [Hanczar 2007].

## 3.2. Wymagania eksploatacyjne

Jednym z najważniejszych zadań w obszarze spełniania wymagań eksploatacyjnych jest ściśle przestrzeganie standardów, ponieważ odejście od nich, np. poprzez naśladownictwo innych projektów, może prowadzić do przekreślenia korzyści uzyskanych dzięki oryginalnym rozwiązaniom w zakresie wymagań biznesowych. W szczególności dotyczy to bezpieczeństwa i poufności danych, których wysoki poziom jest koniecznym warunkiem upowszechnienia produktu na rynku.

Niezawodność działania w okresie silnych zmian otoczenia gospodarczego i ostrej rywalizacji między konkurentami jest także czynnikiem kluczowym dla pozytywnej oceny wypracowanych rozwiązań, a otwarta architektura pozwalająca na komunikację danego systemu z wieloma innymi platformami sprzętowo-programowymi jest także wymogiem ważnym w sensie pozytywnego odbioru użytkowników biznesowych działających często z przyczyn organizacyjnych w różnych środowiskach technologicznych.

Do zasad, których respektowanie jest warunkiem koniecznym w procesie prawidłowego projektowania interfejsu użytkownika, można zaliczyć:

- zagwarantowanie importu danych z możliwie dużej liczby formatów i platform;
- dużą wydajność;
- istnienie modelu szkoleń e-learningowych;
- system zarządzania bazą danych musi uwzględniać:

- materiały tekstowo-graficzne (książkowe),
- materiały prezentacyjne (narracyjne, wykładowe),
- materiały dźwiękowe,
- materiały filmowe,
- interaktywne (quizy, testy, sprawdziany);
- zgodność ze standardami;
- niezawodność;
- zapewnienie poprawności formalnej;
- dostępność;
- bezpieczeństwo rozumiane jako odporność na:
  - niezgodne z przeznaczeniem wykorzystanie udostępnionych materiałów,
  - nieuprawniony dostęp do niezakupionych materiałów lub materiałów, do których czasowy dostęp wygasł,
  - spowodowanie czasowego braku dostępu do usługi przez innych użytkowników,
  - nieuprawnione zmiany w bazie danych materiałów, użytkowników lub finansów jak też inne cechy bezpieczeństwa systemów informatycznych;
- konserwację serwerów i ich oprogramowania, która nie będzie wymagać bieżącej administracji; w przypadku problemów technicznych, gdzie automatyczna kontrola pracy systemu jest niemożliwa, system musi sygnalizować problem skutecznie, informując administratora (przez Internet lub w razie zaniku połączenia internetowego — przez bramkę SMS);
- przenośność infrastruktury SWD na bardziej nowoczesne i wydajne platformy technologiczne.

### 3.2.1. Założenia techniczno-organizacyjne

Interfejs użytkownika składa się z modułu systemu, który jest widoczny dla użytkownika i którym może on zarządzać, oraz z modeli myślowych, nagromadzonych doświadczeń i działań intuicyjnych powstających podczas pracy z tą częścią systemu.

W celu opracowania interfejsu użytkownika należy najpierw opracować model potrzeb informacyjnych użytkownika, który powinien umożliwiać poprawne wnioskowanie dotyczące przewidywanego zachowania systemu. W procesie projektowania interfejsu należy przeanalizować zadania i cele, kontekst użycia, określenie klas użytkowników i na tej podstawie zdefiniować możliwie pełną koncepcję rozwiązania.

Etapy projektowania interfejsu użytkownika powinny objąć takie elementy, jak:

- poznanie wymagań użytkownika,
- dobór rodzajów okien i opracowanie systemu menu,
- opracowanie elementów sterujących związanych z urządzeniami,
- rozplanowanie położenia okien,
- dobór kolorów,
- utworzenie ikon,

- przygotowanie komunikatów i tłumaczeń,
- testowanie.

Zalecane jest w projektowanym interfejsie konstruowanie widoków użytkownika w postaci np. diagramów obiektowych z udziałem architektury Models Views Controls (MVC) [Kuźniarz, Piasecki 1998]. Bardzo ważne jest również stosowanie reguł, np. trzech części lub silnego środka gwarantujących utrzymanie równowagi na ekranie, co nie musi oznaczać jednak symetrii [Jerram, Gosney 1993].

### 3.2.2. Założenia ekonomiczne

Wśród założeń ekonomicznych należy przeanalizować stosowane metody i standardy prototypowania interfejsu użytkownika w celu uniknięcia niepotrzebnych błędów na dalszych etapach procesu generowania interfejsu, co może prowadzić do powstawania nieuzasadnionych kosztów. Do powszechnie uznawanych standardów projektowania interfejsu użytkownika zalicza się modelowanie zwinne (ang. *Agile Modelling* — AM) oraz Uniwersalny Model Interfejsu Użytkownika Boba Baxleya (ang. *Bob Baxley's Universal Model of a User Interface*).

Cykliczność dostarczania kolejnych fragmentów SWD i wynikające z tego założenia korzyści zwiększają się wraz ze wzrostem kosztu pozyskania kapitału oraz zróżnicowania wartości kolejnych wersji. Istotne jest także zagwarantowanie projektantowi możliwości opóźnienia decyzji o kontynuacji prac nad SWD do momentu uzyskania pełniejszych informacji o warunkach zewnętrznych, jak również przyjęcie rygorystycznych założeń upraszczających i tym samym potaniecie procesu tworzenia systemu.

### 3.2.3. Przykład specyfikacji wymagań eksploatacyjnych

Cechy pakietu oprogramowania RealFlex 6 stanowią dobry wzorzec specyfikacji wymagań funkcjonalnych. Obejmują one takie elementy, jak:

- pakiet oprogramowania dla serwera pracującego w czasie rzeczywistym do SCADA i aplikacji kontroli procesu;
- konfiguracja pełnej gotowości (ang. *Hot Standby*);
- projekt wysokiej wydajności, odpowiedni dla dużych BD;
- wsparcie dla milisekundowych znaczników czasu;
- przetwarzanie danych w czasie rzeczywistym;
- archiwa baz danych;
- alarmy i systemy przetwarzania zdarzeń;
- procesor kalkulacji;
- zdolność kontroli;
- CSL;
- FlexView HMI;

- wsparcie dla opcjonalnych klientów;
- szeroka gama standardowych wspieranych protokołów, np. DNP 3, IEC870 i Modbus;
- duży konfigurowalny interfejs alarmów i buforów zdarzeń poprzez konsole FlexView;
- nieograniczona liczba klientów FlexView;
- automatyczna migracja z systemu RealFlex 4;
- zestaw migracyjny dla sterowników RealFlex 4;
- serwer API dla aplikacji QNX, które wchodzi w interakcję z konsolami FlexView;
- zestaw do prac ze sterownikiem i aplikacjami.

### 3.3. Projektowanie interfejsu użytkownika

Współczesne systemy integracji danych oraz inne duże SWD oparte na BD cechuje modułowość budowy. Struktura ta jest wynikiem dekompozycji dużego zadania na wiele mniejszych. Takie podejście pozwala na łatwiejsze tworzenie, zarządzanie i utrzymanie oprogramowania. Dodatkową korzyścią jest możliwość dostarczenia klientowi tylko tej funkcjonalności, której potrzebuje, a jednocześnie umożliwia łatwiejsze skalowanie systemu, dzięki czemu potencjalne grono odbiorców jest większe.

We współczesnych tendencjach rozwoju interfejsów użytkownika dominują kierunki podkreślające konieczność uwzględniania zasady komunikacji i interaktywnej współpracy z użytkownikiem i stałego ulepszania interfejsu tak długo, aż satysfakcja użytkownika nie będzie pełna. Takie podejście proponuje m.in. B. Baxley w swoim Uniwersalnym Modelu Interfejsu Użytkownika [Ambler 2002].

Według S. Wryczy [1999] projektowanie i użytkowanie interfejsu użytkownika należy rozpatrywać w układzie czterech warstw:

- 1) metafory stanowiącej pewne imitacje sytuacji rzeczywistej;
- 2) metod reprezentujących sposoby kontaktowania się użytkownika z komputerem;
- 3) warstwy urządzeń eksploatowanych przez użytkownika;
- 4) warstwy fizycznej reprezentującej czynności fizyczne wykonywane w związku z wykorzystywaniem metod i urządzeń.

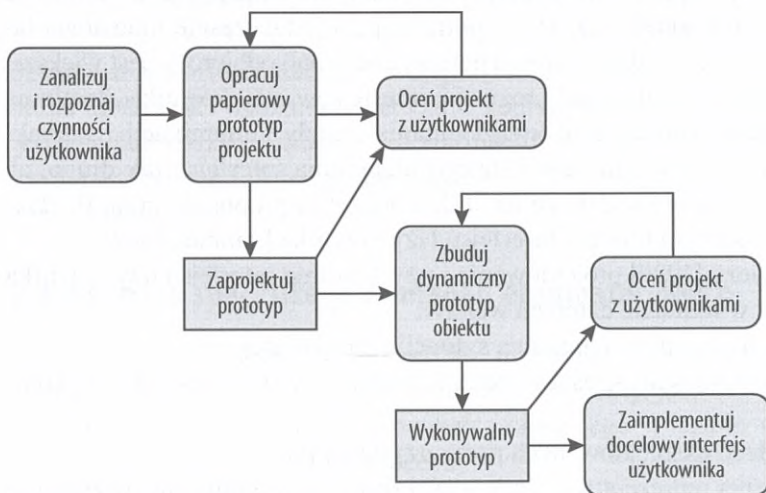
C.M. Olszak [2000] zaleca wyróżnienie kilku poziomów w budowie interfejsu użytkownika, obejmujących: analizę zadań i celów systemu, analizę języka oraz stosowanych metafor (poziom leksykalny), analizę spostrzegania systemu przez decydena, (poziom semantyczny) i wybór standardu interfejsu oraz środowiska jego implementacji, czyli poziom semantyczny. Autorka w ramach pierwszego poziomu projektowania łączy komunikacji z decydem proponuje wprowadzić proces adaptacyjny, polegający na obserwacji użytkownika przez system i wychwytywanie regularności w jego

zachowaniu. W ramach poziomu leksykalnego projektowania celowe jest skoncentrowanie uwagi na opisie języka oraz pojęć stosowanych przez decydentów, co w znacznym stopniu polepszy percepcję informacji zgromadzonych w systemie. W ramach poziomu semantycznego projektowania można wspomagać ten proces metodami zorientowanymi na dane (np. SADT) i procesy (np. SASS).

Bardzo dobrą metodą projektowania interfejsu użytkownika jest prototypowanie, które pozwala generować nowe rozwiązania w zakresie projektowania i oceniać ich jakość we wczesnej fazie tego procesu. Dobrze ten interaktywny proces prototypowania ilustruje rysunek 3.1. Po wstępnym zdefiniowaniu potrzeb następuje faza tworzenia prototypu i jej ocena, która jest powtarzana cyklicznie dopóty, dopóki potrzeby użytkownika nie zostaną zaspokojone.

Tworząc prototyp, należy: uwzględnić potrzeby klienta, przeprowadzić analizę testów użyteczności oraz konsultacje z zespołem wdrożeniowym, określić relacje między szerokością a głębokością (stopień wgłębiania się w szczegóły), dokonać wyboru alternatywy wireframe kontra projekt wizualny oraz przeprowadzić analizę ROI, czyli uściślić dokładnie, czego oczekujemy od prototypu.

Rysunek 3.1. Ilustracja procesu prototypowania interfejsu użytkownika



Źródło: Opracowanie własne.

Scenariusz tworzenia prototypu powinien objąć takie działania, jak:

- Krok 1. Wyodrębnij najistotniejsze ekrany, co pozwoli skoncentrować się na najważniejszych ustalonych celach budowy prototypu interfejsu użytkownika. Ekrany te posłużą następnie jako szablony dla pozostałych ekranów interfejsu.
- Krok 2. Określ na ekranie najważniejsze obszary (schemat blokowy). Za pomocą schematu blokowego wyodrębnij, gdzie mają znajdować się: menu nawigacyjne, informacje o statusie, treść dokumentu, formularza itp.

- Krok 3. Określ strukturę najważniejszego ekranu. Uzupełnij schemat blokowy tego ekranu o niezbędne szczegóły wytypowane w kroku 2.
- Krok 4. Określ zestaw pozostałych ekranów. Wykorzystaj najważniejszy ekran (krok 3) jako szablon do zaprojektowania ich struktur. Takie postępowanie pomoże w utrzymaniu spójności prototypu.
- Krok 5. Opracuj dokumentację prototypu. Dzięki dokumentacji prototypu będzie możliwa ewaluacja postępu prac rozwojowych i wdrożeniowych w zakresie możliwości uzyskania celów biznesowych, funkcjonalnych i technicznych prototypu decydujących o jego użyteczności.

Szerzej na temat projektowania interfejsu użytkownika wypowiadają się inni autorzy, np. M. Sikorski [2010b] czy M. Sikorski i M. Wichary [2004].

### 3.3.1. Ergonomiczne cechy interfejsu

Ze względu na konieczność wypracowania ergonomicznych cech interfejsu ważne jest zgromadzenie wiedzy o potencjalnych użytkownikach SWD (klasyfikacja użytkowników, przewidywana częstość wykorzystywania SWD), zaproponowanie budowy modułowej SWD zakładającej zróżnicowany poziom zaawansowania użytkowników systemu oraz wzorowanie się na sprawdzonych wzorcach w tym zakresie, np. na modelu Collinsa, obejmującego iterację, testowanie i ocenę. Ważne jest także bezpieczeństwo interfejsu związane z zapewnieniem poufności i zabezpieczenia danych przed utratą. Szerzej na temat ergonomicznych cech interfejsu wypowiada się M. Sikorski [2010a].

### 3.3.2. Bezpieczeństwo interfejsu

Na poziomie projektu interfejsu użytkownika zależność między bezpieczną pracą a stosowanymi zabezpieczeniami jest podstawowym czynnikiem zapewniającym optymalizację wydajności SWD jako całości. Interfejsy operatora realizowane z użyciem GE Fanuc Automation Cimplicity<sup>3</sup> zapewniają autoryzowanym użytkownikom dostęp do danych sterujących z dowolnego miejsca w przedsiębiorstwie — jednakże zmiany parametrów produkcji mogą być dokonywane wyłącznie z wydzielonych do tego celu lokalizacji. W przypadku interfejsów użytkownika różnica między bezpieczeństwem a zabezpieczeniami jest określona w czytelny i prosty sposób. Bezpieczeństwem określa się takie działanie programu sterującego PLC<sup>4</sup>, które wraz z dodatkowym oprzyrządowaniem stanowią o tym, czy aplikacja spełnia funkcje bezpiecznego działania, czy też nie. Funkcje bezpieczeństwa są zwykle implementowane w programie PLC.

<sup>3</sup> [www.koamtech.com/sub\\_fa/MMI/HMI\\_Catalog\\_English/HMI\\_Catalog\\_English.pdf](http://www.koamtech.com/sub_fa/MMI/HMI_Catalog_English/HMI_Catalog_English.pdf).

<sup>4</sup> [www.astor.com.pl/nasi-dostawcy/wonderware.html](http://www.astor.com.pl/nasi-dostawcy/wonderware.html).

Zabezpieczenia systemu służą przeciwdziałaniu próbom włamań do systemu sterowania w celu kradzieży danych lub uszkodzenia systemu. A zatem bezpieczeństwo i zabezpieczenia zwykle funkcjonują osobno. Jednakże w przypadku interfejsów HMI oba te pojęcia (a tym samym obszary zastosowań) uzupełniają się wzajemnie.

Z punktu widzenia interfejsów HMI wyróżnia się trzy główne scenariusze:

- 1) ktoś z zewnątrz włamuje się poprzez firmowy firewall i poprzez sieć przedsiębiorstwa dociera do interfejsu użytkownika, a w rezultacie powoduje w nim zmiany niepożądane z punktu widzenia cech funkcjonalnych lub eksploatacyjnych;
- 2) pracownik firmy z nieznanых powodów przeprowadza atak na HMI;
- 3) trzeci scenariusz mówi o kimś, kto będąc pracownikiem firmy, powoduje zadziałanie systemów zabezpieczeń albo wywołuje problem w działającej prawidłowo aplikacji SWD.

Zdaniem zarządzających, przedsiębiorstwa wpadają w poważne kłopoty, kiedy powierzają zabezpieczanie systemów sterowania zespołom zajmującym się technologiami klasy IT. Specjaliści z obszaru IT na ogół poszukują rozwiązań zabezpieczeń poprzez izolowanie maszyn od siebie, aby osoby pracujące w sieci, np. nie rozprzestrzeniły wirusów w sieci przedsiębiorstwa. Metody takie sprawdzają się w zastosowaniach technologii informatycznych w zarządzaniu, jednak podejście takie wyraźnie ogranicza komunikację między maszynami, nie pozwalając osiągnąć pożądanej efektywności ich działania.

W procesie projektowania systemów sterowania maszyny są projektowane w taki sposób, aby bezkolizyjnie wymieniać między sobą dane, informacje oraz polecenia. Większość maszyn w systemach sterowania, w świetle dostępnych aktualnie rozwiązań informatycznych, jest jednocześnie serwerami i klientami. Tym samym wykorzystanie powszechnego modelu klient-serwer wydaje się niemożliwe. Sposobem zabezpieczenia systemu sterowania może być zamknięcie go za „murem ochronnym”, określanym mianem firewall z zapewnieniem kontroli wymiany danych między strzeżoną przestrzenią wewnętrzną a zewnętrzną. Wtedy na serwerze przechowywane są wszystkie dane związane z zatrzymaniem lub innymi zdarzeniami procesowymi. Dostęp do nich ma każdy pracownik, który ich potrzebuje, ale musi być do tego uprawniony. Informacje są bowiem przechowywane w specjalnie wydzielonej części sieci odizolowanej od sieci przedsiębiorstwa.

### 3.3.3. Przykłady interfejsów

Obecnie spotykane są rozmaite standardowe typy interfejsów, pozwalających zwiększać funkcjonalność aplikacji pod względem zapewnienia optymalnych kosztów, bardziej komfortowego i efektywnego ich użytkowania czy lepszego zabezpieczania danych oraz automatyzowania tych procesów, a także konstruowania w sposób efektywny aplikacji. Do takich typów należą m.in. konsole, systemy za-



bezpieczeń czy też generatory aplikacji, co zostało bardziej szczegółowo omówione w punkcie 5.1.4.

Architektura systemu ESS II jest dobrym przykładem nowoczesnych rozwiązań w zakresie modułowości i funkcjonalności metod komunikacji człowieka z komputerem.

Struktura systemu ESS II składa się z dwóch głównych warstw: serwerowej i klienckiej. Warstwa serwerowa zawiera podwarstwy odpowiedzialne za komunikację, mające postać zestawu interfejsów oraz podwarstwy przetwarzające w postaci procesów serwerów.

W warstwie klienckiej można wyróżnić podwarstwę, w skład której wchodzi narzędzia konfiguracyjne oraz podwarstwę użytkową. System ESS II jest otwarty i pozwala na dołączanie nowych aplikacji korzystających z funkcji API. W zależności od charakteru stworzonej aplikacji mogą one pozwalać na konfigurację lub udostępnić któryś z trybów pracy.

Innym przykładem jest otwarta platforma Eclipse umożliwiająca korzystanie z wygodnych interfejsów (tzw. punktów rozszerzeń) do przejrzystej integracji narzędzi. Platforma taka gwarantuje, że całe instrumentarium pakietu programistycznego zawarte jest w jednolitym graficznie stylu. Takie rozwiązanie eliminuje konieczność uczenia się za każdym razem interfejsu danego narzędzia czy też każdej z używanych platform.

Przykładem zastosowania najbardziej innowacyjnych interfejsów użytkownika jest nowy model Jaguara XJ, który został zaprojektowany tak, aby znajdująca się na jego pokładzie technologia była maksymalnie przyjazna użytkownikowi. W samochodzie zamiast „fizycznego” zestawu wskaźników jest zainstalowany 12,5-calowy ekran o wysokiej rozdzielczości wyświetlający wyrafinowane — i bardzo szczegółowe — wskaźniki wirtualne, które obejmują wszystkie funkcje tradycyjnych wskaźników. Wykorzystując elastyczność w projektowaniu funkcjonalności wirtualnych wskaźników, inżynierowie firmy nadali najwyższy priorytet istotnym informacjom oraz zapewnili kierowcy wrażenia estetyczne w kokpicie. Duży, 8-calowy ekran dotykowy w konsoli środkowej stanowi dopełnienie wirtualnego zestawu wskaźników, zapewniając intuicyjny sposób sterowania wieloma funkcjami modelu XJ — np. klimatyzacją, systemem audio, funkcjami komunikacyjnymi i nawigacją. Interfejs dotykowy pozwala na ograniczenie liczby fizycznych przycisków i przełączników do minimum, a pod ekranem znajduje się tylko kilka głównych elementów sterujących. Ekran typu Touchscreen ma nie tylko charakter intuicyjny, ale wykorzystuje również technologię Dual-View o znaczącym poziomie zaawansowania, która pozwala kierowcy i pasażerowi z przodu na oglądanie zupełnie różnych obrazów na tym samym ekranie. Pasażer może np. oglądać telewizję lub film DVD, podczas gdy kierowca widzi mapę z trasą systemu nawigacyjnego. Stosowany w tym modelu Jaguara interfejs połączeniowy Media Hub oferuje możliwość elastycznego podłączenia urządzeń za pośrednictwem dwóch portów USB, strumieniowe przesyłanie audio przez łącze Bluetooth oraz posiada dodatkowe wejście, do którego można podłączyć np. iPod,

iPhone'a, laptopa lub telefon Blackberry. Taki typ samochodu z zaprezentowanymi możliwościami interfejsu użytkownika może stanowić i zastępować sprawnie działające biuro często przemieszczającego się menedżera.

## 3.4. Pytania i zadania kontrolne

### 3.4.1. Pytania kontrolne

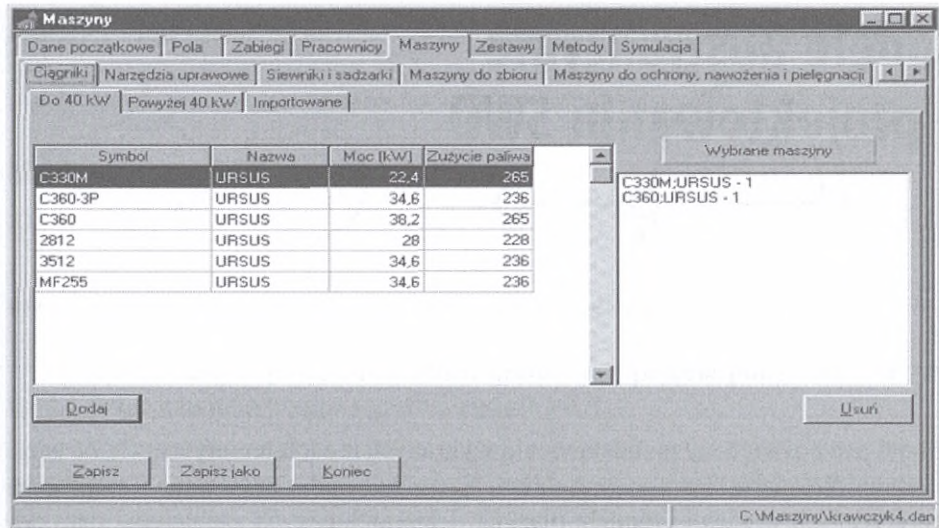
1. Jakie specyfikacje funkcjonalne interfejsu użytkownika powinny być spełnione, aby usatysfakcjonować użytkownika aplikacji?
2. Jakie wymagania eksploatacyjne interfejsu użytkownika są konieczne do prawidłowego funkcjonowania aplikacji?
3. Omów wymagania bezpieczeństwa zawarte w normie ISO/IEC 27001 : 2007.
4. W jaki sposób stosowanie podejścia ewolucyjnego poprzez wykorzystanie Uniwersalnego Modelu Interfejsu Użytkownika B. Baxleya i standardu modelowania zwinnego przyczynia się od podniesienia jakości rozwiązań interfejsu użytkownika w zakresie ergonomii?

### 3.4.2. Przykładowe zadania

1. Zaprojektuj prototyp interfejsu użytkownika uwzględniający wejścia, wyjścia oraz drogi przepływu danych umożliwiające, na podstawie ocen końcowych w semestrze, obliczenie średniej ocen jako podstawy przyznania stypendium naukowego.
2. Na podstawie rysunku 3.2 (aplikacja programu symulacyjnego na potrzeby minimalizacji kosztów mechanizacji produkcji roślinnej; program służy do tworzenia zestawów maszynowo-ciągnikowych w modelu przedsiębiorstwa) zaproponuj, co zmieniłbyś w interfejsie użytkownika (kolorystyka, rozmieszczenie obiektów graficznych, opis słowny, symbole pomocnicze, inne), aby dane z baz słownikowych, widniejące z lewej strony ekranu i ukryte w menu górnej części ekranu, mogłyby być efektywnie i intuicyjnie wykorzystane do tworzenia zestawów produkcyjnych przedsiębiorstwa rolnego (z prawej strony ekranu pod nagłówkiem: „wybrane maszyny”).

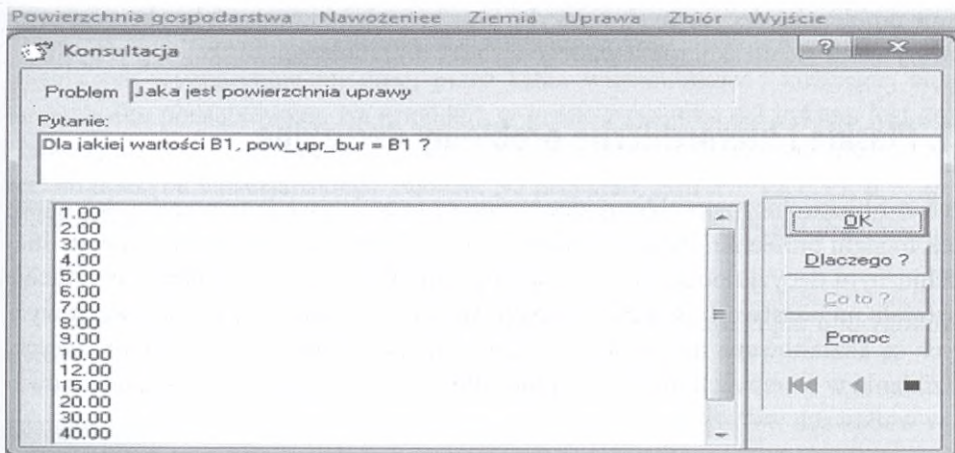
Biorąc pod uwagę rysunek 3.3, zweryfikuj poprawność zaprojektowanego okienka aplikacji, która uwzględnia omówione kryteria funkcjonalności użytkownika związane z indywidualizacją dostosowania interfejsu do specyficznych potrzeb użytkownika oraz adekwatność tekstów i liczb pojawiających się w okienku, a związanych z koniecznością wyboru określonej powierzchni uprawy buraków cukrowych.

**Rysunek 3.2. Ekran aplikacji programu symulacyjnego dla potrzeb minimalizacji kosztów mechanizacji produkcji roślinnej, ilustrujący działanie interfejsu użytkownika**



Źródło: Opracowanie własne.

**Rysunek 3.3. Ekran aplikacji systemu ekspertowego dla potrzeb oceny organizacyjno-technicznej procesu uprawy buraków cukrowych**



Źródło: Opracowanie własne.

# Modele abstrakcyjne w projektowaniu SWD

Rozdział jest poświęcony przedstawieniu wybranych modeli teoretycznych, na bazie których w SWD formułowane są problemy decyzyjne i ich rozwiązania. Listę modeli teoretycznych otwierają modele płaskich i hierarchicznych problemów decyzyjnych, wywodzące się z analizy morfologicznej powiązanych obszarów decyzyjnych, zaproponowanej w 1948 r. przez F. Zwicky'ego. Teoria zbiorów rozmytych, podobnie jak i teoria zbiorów przybliżonych, umożliwia stosowanie pojęć nieostrych w formułowaniu problemów decyzyjnych i w znajdowaniu ich rozwiązań. Teoretycznej nieostrości pojęć lub nierozróżnialności przypadków odpowiada wiele rzeczywistych sytuacji decyzyjnych. Zasada charakteryzacji pozwala na formułowanie i rozwiązywanie problemów decyzyjnych w języku prostych zdań logicznych bez konieczności odwoływania się do skomplikowanego aparatu minimalizacji funkcji logicznych.

## 4.1. Płaskie i hierarchiczne problemy decyzyjne

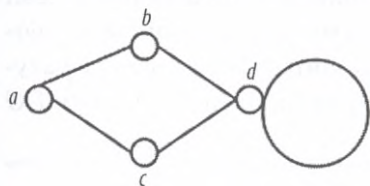
Płaskie i hierarchiczne problemy decyzyjne są rozwiązywane w wyniku dekompozycji modelu problemu decyzyjnego do postaci drzewa rozwiązań. W systemie hierarchicznym decyzje podejmowane są na poziomie bieżącym w oparciu o sytuację zaistniałą na warstwach poprzedzających. W ten sposób decyzje w warstwach wyższych są kształtowane na podstawie stanu realizacji zadań w warstwach niższych, a działania w warstwach niższych są formułowane w oparciu o decyzje podejmowane w warstwach wyższych.

### 4.1.1. Graficzny model płaskiego problemu decyzyjnego

Problemy związane z podejmowaniem decyzji często przedstawia się za pomocą grafów. Potocznie **grafem** nazywa się zbiór punktów połączonych łukami. Możliwe jest połączenie punktu z samym sobą. Niektóre pary punktów mogą być połączone wie-

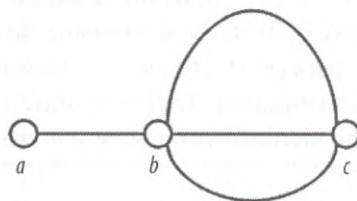
lokrotnie (łuki równoległe). Na rysunkach 4.1–4.4 przedstawiono niektóre szczególne postaci grafów.

Rysunek 4.1. Graf z pętlą



Źródło: Opracowanie własne.

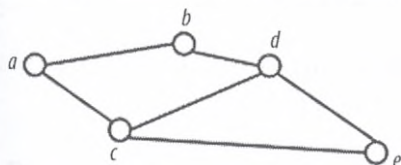
Rysunek 4.2. Graf z łukami wielokrotnymi



Źródło: Opracowanie własne.

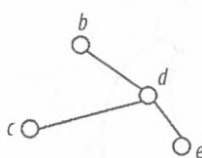
Bardzo ważnym pojęciem w teorii grafów jest pojęcie **podgrafu**. Na rysunku 4.4 przedstawiono podgraf grafu z rysunku 4.3.

Rysunek 4.3. Graf bez łuków wielokrotnych i pętli



Źródło: Opracowanie własne.

Rysunek 4.4. Podgraf grafu z rysunku 4.3



Źródło: Opracowanie własne.

Graf przedstawiony na rysunku 4.4 nie posiada dróg, zwanych cyklami, umożliwiających wyznaczenie tej drogi przez kilka wierzchołków i kończącej się na wierzchołku początkowym. Na przykład, w grafie z rysunku 4.3 cyklem jest droga prowadząca przez wierzchołki  $a, b, d, c$ . Ścisła definicja grafu  $G$  związana jest z pojęciem iloczynu kartezjańskiego zbiorów. Na przykład, jeśli  $A = \{a, b, c\}$ ,  $B = \{d, e\}$ , to iloczynem kartezjańskim  $A \times B$  zbiorów  $A$  i  $B$  nazywamy zbiór par postaci:

$$A \times B = \{(a, d), (a, e), (b, d), (b, e), (c, d), (c, e)\}.$$

Zatem iloczyn kartezjański  $A \times B$  zbiorów  $A$  i  $B$  jest zbiorem par uporządkowanych.

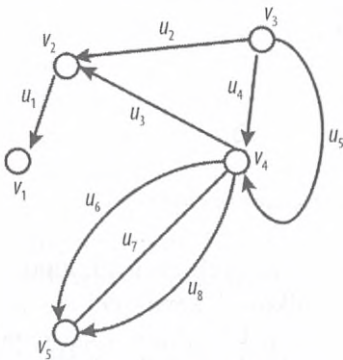
#### Definicja 4.1

**Grafem** nazywamy trójkę  $(V, U, \varphi)$ , gdzie  $V$  jest zbiorem wierzchołków grafu,  $U$  — zbiorem łuków grafu,  $\varphi$  — funkcją incydencji przyporządkowującą każdemu łukowi  $u \in U$  parę punktów  $(v_1, v_2)$ . Można teraz zapisać  $\Phi: U \rightarrow V \times V$ . Punkty  $v_1$  i  $v_2$  nazywają się początkiem i końcem łuku.

Jeśli  $\varphi(U) = (v_1, v_2)$ , to oznacza, że łuk  $u$  jest incydenty ze swymi końcami. Jeśli zbiór  $U \cup V$  jest skończony, to graf  $G$  nazywa się grafem skończonym. Wierzchołek, który nie jest incydenty z żadnym łukiem, nazywamy izolowanym. Wierzchołek incydenty z dokładnie jednym łukiem nazywamy wiszącym. Jeśli para  $(v_1, v_2)$  jest uporządkowana i  $\varphi(U) = (v_1, v_2)$ , to łuk  $u$  nazywamy skierowanym (zorientowanym). Jeśli wszystkie łuki grafu  $G$  są skierowane, to graf  $G$  nazywa się skierowanym. Analogicznie, jeśli para  $\varphi(U) = (v_1, v_2)$  jest nieskierowana (niezorientowana), to łuk  $u$  nazywa się nieskierowanym. Jeśli wszystkie łuki grafu  $G$  są nieskierowane, to graf  $G$  nazywamy nieskierowanym (niezorientowanym).

Istnieją różne sposoby zadania grafów. Związane są one z różnym sposobem określenia funkcji  $\varphi$ . W celu przedstawienia sposobu zadania grafu rozważmy przykład grafu zadanego na rysunku 4.5.

**Rysunek 4.5. Graf, w którym  $V = \{v_1, v_2, v_3, v_4, v_5\}$ ,  
 $U = \{u_1, u_2, u_3, u_4, u_5, u_6, u_7, u_8\}$**



Źródło: Opracowanie własne.

A. Spis łuków wraz z końcami:

$$u_1(v_2, v_1); u_2(v_2, v_3); u_3(v_4, v_2); u_4(v_4, v_3); u_5(v_4, v_5); u_6(v_4, v_5); u_7(v_4, v_5); u_8(v_5, v_4).$$

B. Macierz incydencji grafu

Macierz incydencji grafu jest to macierz  $\mathbf{A} = [a_{ij}]$  o wymiarach  $n \times m$ ,  $n$  — liczba wierszy — wierzchołków,  $m$  — liczba łuków — kolumn macierzy  $\mathbf{A}$ . Macierz  $\mathbf{A}$  definiuje się następująco:

$$a_{ij} = \begin{cases} 0, & \text{jeśli wierzchołek } v_i \text{ nie jest incydenty z łukiem } v_j \\ 1, & \text{jeśli wierzchołek jest końcem niezorientowanego łuku } u_j \\ -1, & \text{jeśli } v_i \text{ jest początkiem łuku } u_j. \end{cases}$$

Dla łuku niezorientowanego w obu przypadkach wierzchołkom przyporządkowuje się 1. W każdej kolumnie macierzy  $\mathbf{A}$  są przynajmniej dwa elementy niezerowe (ale nie dla pętli):

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 1 & -1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}.$$

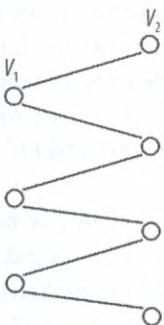
C. Macierz sąsiedztwa

Macierz  $\mathbf{B}$  sąsiedztwa jest macierzą kwadratową o wymiarze  $N \times N$ , gdzie:  $n$  — liczba wierzchołków,  $\mathbf{B} = [b_{ij}]$ ,  $i, j = 1, 2, \dots, n$ ,  $b_{ij}$  jest równe liczbie łuków łączących początek łuku  $v_i$  z końcem  $v_j$ . Równość  $b_{ij} = b_{ji}$  nie jest na ogół prawdziwa. Równość powyższa jest prawdziwa dla grafów niezorientowanych.

Macierz  $\mathbf{A}$  incydencji zadaje graf jednoznacznie, jednak macierz  $\mathbf{B}$  sąsiedztwa wierzchołków określa graf z dokładnością do zamiany dowolnego łuku niezorientowanego przez parę przeciwnie skierowanych łuków między tymi samymi wierzchołkami. Dla grafu bez łuków wielokrotnych macierz  $\mathbf{B}$  sąsiedztwa zadaje graf jednoznacznie.

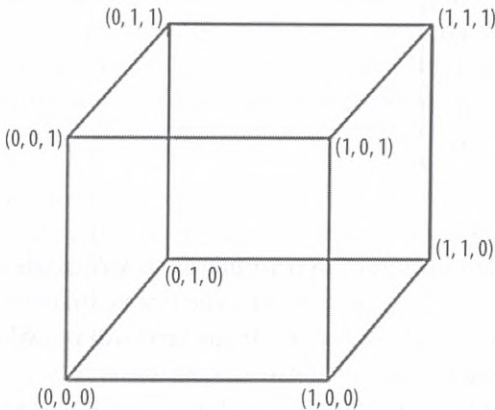
- D. Grafem pełnym nazywa się graf bez wielokrotnych łuków i bez pętli, w którym dwa dowolne wierzchołki są połączone łukiem zorientowanym lub niezorientowanym.
- E. Grafem dwudzielnym nazywa się graf, którego wierzchołki można rozbić na dwa rozłączne podzbiory,  $V = V_1 \cup V_2$ , a łuki wiążą wierzchołki z różnych klas. Przykład grafu dwudzielnego przedstawiono na rysunku 4.6.

Rysunek 4.6. Graf dwudzielnny



- F. Kostką  $n$ -wymiarową nazywamy graf, którego wierzchołkami są  $n$ -elementowe ciągi złożone z 0 i 1. Łuki łączą te wierzchołki, które różnią się na dokładnie jednej pozycji. Przypadek kostki 3-wymiarowej pokazano na rysunku 4.7.

Rysunek 4.7. Kostka 3-wymiarowa



Źródło: Opracowanie własne.

W tym podpunkcie podaje się niezbędne wiadomości związane z pojęciem podgrafu danego grafu.

Niech  $G = (V, U, \varphi)$  będzie dowolnym grafem. Graf  $H = (V_1, U_1, \varphi_1)$  nazywa się podgrafem grafu  $G$ , jeśli  $V_1 \cup V_2$ ,  $U_1 \cup U_2$  oraz  $\varphi_1$  jest obcięciem funkcji  $\varphi$  na zbiór  $U$ . Fakt ten zapisuje się następująco:  $H \in G$ . Przy czym każdy łuk z  $U_1$  wchodzi do podgrafu  $H$  wraz z końcami.

- G. Spójność, cykle, łańcuchy

Ciąg wierzchołków i łuków grafu  $G$  w postaci:

$$v_{i_0} u_1 v_{i_1} u_2 v_{i_2} u_3 \dots v_{i_{n-1}} u_n v_{i_n} \quad [4.1]$$

nazywamy drogą  $\langle v_{i_0}, v_{i_n} \rangle$  z wierzchołka  $v_{i_0}$  do wierzchołka  $v_{i_n}$  dla  $k = 1, 2, \dots, n$ . Wierzchołek  $v_{i_0}$  nazywa się początkiem drogi,  $v_{i_n}$  — końcem drogi, liczbę  $n$  — długością drogi (droga o długości zero jest złożona z jednego wierzchołka). W ogólnym przypadku wśród wierzchołków ciągu [4.1] mogą się powtarzać wierzchołki lub łuki. Każdy łuk może być zorientowany (z  $v_{i_{k-1}}$  do  $v_{i_k}$ ) lub niezorientowany. Mówi się, że droga  $\langle v_{i_0}, v_{i_n} \rangle$  przechodzi przez wierzchołki  $v_{i_0}, v_{i_1}, v_{i_2}, \dots, v_{i_{n-1}}, v_{i_n}$ , po łukach  $u_1, u_2, \dots, u_n$ .

Łańcuchem nazywamy ciąg wierzchołków i łuków tworzących drogę w odpowiadającym grafie niezorientowanym. Wiadomo teraz, że każda droga jest łańcuchem, odwrotne twierdzenie jest prawdziwe tylko dla grafów niekierowanych, dla których oba pojęcia się pokrywają.



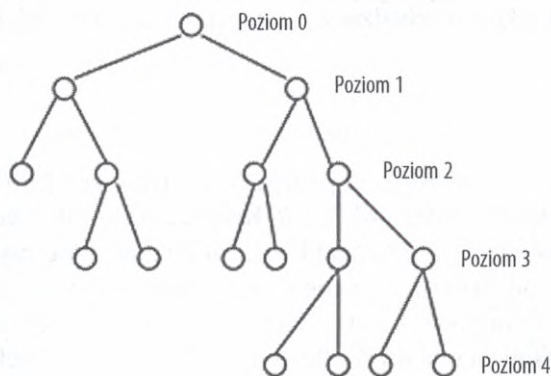
Jeśli w ciągu [4.1]  $v_{i_0} = v_{i_n}$  ( $n > 0$ ), to taką drogę (odpowiednio łańcuch) nazywamy konturem. Drogę, łańcuch i kontur nazywamy prostym (elementarnym), jeśli każdy łuk (każdy wierzchołek i każdy łuk) wchodzi do ciągu [4.1] dokładnie jeden raz (nie uwzględniając ostatniego wierzchołka w cyklu).

#### H. Drzewa

Jeśli dla dowolnych  $v_1, v_2 \in V$  istnieje droga  $\langle v_1, v_2 \rangle$  łącząca te wierzchołki, to graf nazywamy spójnym. Łuk  $u$  dowolnego grafu  $G$  nazywa się cyklicznym, jeśli należy on do przynajmniej jednego cyklu. Graf spójny, którego wszystkie łuki są acykliczne, nazywamy drzewem.

Następująca własność charakteryzuje drzewo: graf jest drzewem wtedy i tylko wtedy, gdy każde dwa wierzchołki drzewa są powiązane przez dokładnie jeden łańcuch elementarny. Wybieramy w drzewie dowolny wierzchołek  $\alpha_0 \in V$  i nazywamy go pierwiastkiem lub wierzchołkiem zerowego poziomu. Sąsiednie wierzchołki nazywamy wierzchołkami 1. poziomu. Każdy wierzchołek drzewa należy do dokładnie jednego poziomu. Numer poziomu jest równy odległości między jego wierzchołkami a pierwiastkiem drzewa. Takie przedstawienie drzewa jest bardzo wygodne. Pokazuje ono, że w każdym skończonym drzewie zawsze istnieją końcowe wierzchołki (rysunek 4.8).

Rysunek 4.8. Przykład drzewa



Źródło: Opracowanie własne.

Wyznaczanie pierwiastka w drzewie  $D$  określa na zbiorze wierzchołków drzewa częściowy porządek. Jeśli  $\alpha < \beta$ , przy  $\alpha \neq \beta$ , to elementarny łańcuch z pierwiastka drzewa do wierzchołka  $\beta$  zawiera wierzchołek  $\alpha$ .

#### I. Drzewa decyzyjne

Drzewa decyzyjne są graficzną metodą wspomaganą procesu decyzyjnego. Metoda drzewa decyzyjnego należy do najczęściej wykorzystywanych technik

analizy danych. Rozwiązywanie zadań decyzyjnych w warunkach niepewności polega na wyznaczeniu optymalnej decyzji (można przedstawić za pomocą drzewa). Wierzchołki, z których wychodzi co najmniej jedna krawędź, są nazywane węzłami, a pozostałe wierzchołki liśćmi. W każdym węźle sprawdzany jest warunek dotyczący obserwacji. Na podstawie wyniku tego sprawdzenia wybierana jest jedna z kilku gałęzi prowadzących do kolejnego wierzchołka na następnym poziomie.

### 4.1.2. Wartościowanie decyzji oraz procedura dekompozycji drzewa decyzyjnego

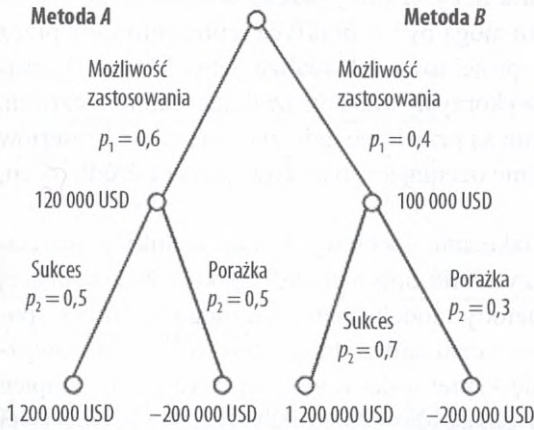
Każdy poziom **drzewa decyzyjnego** posiada tylko jedną zmienną decyzyjną. Zmienna decyzyjna opisująca dany poziom może przyjmować kilka wartości. Liczba jest w praktyce równa liczbie łuków drzewa wychodzących z wierzchołków danego poziomu. Kryterium wyboru testu dla zmiennej decyzyjnej rozstrzyga o złożoności drzewa decyzyjnego. Przy wyborze testu należy się kierować zasadą, aby zbudować możliwie jak najprostsze drzewo. Do najbardziej popularnych metod należy zaliczyć wykorzystanie pojęcia entropii do określania nierównomierności rozkładu kategorii.

Często problemy decyzyjne rozwiązuje się za pomocą probabilistycznych losowych drzew decyzyjnych. W każdym węźle określa się zmienną losową skokową, przyjmującą tyle wartości, ile jest łuków wychodzących z danego węzła (rysunek 4.9).

#### Przykład 4.1

Firma wydobywająca gaz musi podjąć decyzję, czy rozpocząć wiercenie i jaką metodą. Do wyboru istnieją dwie metody wiercenia: *A* i *B*. Koszt zastosowania metody *A* wynosi 120 000 USD, a metody *B* — 100 000 USD. Możliwość wykorzystania jednej z dwóch metod zależy od struktury geologicznej i głębokości wiercenia. Analiza statystyczna pokazuje, że metoda *A* jest użyteczna w 60% wierceń. Stąd wiadomo, że metoda *B* jest stosowana w 40% wierceń. Wykorzystanie metody *A* daje sukces w 50% wierceń, natomiast zastosowanie metody *B* odnosi pozytywny skutek w 70% wierceń. Zysk w przypadku sukcesu odwiertu wynosi średnio (bez względu na wykorzystaną metodę) 1 200 000 USD, natomiast strata to średnio 200 000 USD.

Rysunek 4.9. Drzewo decyzyjne dla przykładu 4.1



Źródło: Opracowanie własne.

### 4.1.3. Procedura dekompozycji drzewa

**Dekompozycja drzewa decyzyjnego** polega na wyznaczeniu wszystkich dróg w drzewie decyzyjnym. Po naniesieniu wszystkich danych dotyczących drzewa decyzyjnego należy przystąpić do wyznaczenia optymalnej decyzji. Drzewo decyzyjne analizuje się od wierzchołka do ostatniego poziomu.

Oczekiwane zyski dla dostępnych metod odwiertu w drzewie z przykładu 4.1 są następujące:

Metoda A:  $1\,200\,000\text{ USD} - 120\,000\text{ USD} = 1\,080\,000\text{ USD}$ ,

Metoda B:  $1\,200\,000\text{ USD} - 100\,000\text{ USD} = 1\,100\,000\text{ USD}$ .

Możliwe straty w przypadku obydwu metod to:

Metoda A:  $-120\,000\text{ USD} - 200\,000\text{ USD} = -320\,000\text{ USD}$ .

Metoda B:  $-100\,000\text{ USD} - 200\,000\text{ USD} = -300\,000\text{ USD}$ .

Widać, że w przypadku obydwu metod możliwy zysk i strata są podobnej wielkości. Decydujące zatem będzie prawdopodobieństwo odniesienia sukcesu lub porażki (pod warunkiem możliwości zastosowania określonej metody), które wynosi odpowiednio:

Metoda A:  $0,6 \cdot 0,5 = 0,3$  (sukces),  $0,6 \cdot 0,5 = 0,3$  (porażka),

Metoda B:  $0,4 \cdot 0,7 = 0,28$  (sukces),  $0,4 \cdot 0,3 = 0,12$  (porażka).

Jak widać z otrzymanych wyników, metoda B jest obciążona znacznie niższym ryzykiem porażki niż metoda A, co przy zbliżonych kosztach oraz zyskach zastosowania może mieć kluczowe znaczenie przy ostatecznym wyborze metody.

### 4.1.4. Grafowy model hierarchicznego problemu decyzyjnego

**Metoda hierarchicznego wspomaganie decyzji** służy przede wszystkim do wyboru wariantów decyzyjnych. Warianty te mogą być w praktyce reprezentowane przez produkty, obiekty fizyczne, warianty projektowe lub realizacyjne. Metody hierarchicznego wspomaganie decyzji są wykorzystywane w naukach ekonomicznych, technicznych i społecznych. Szczególnie są przydatne, gdy znaczna część kryteriów ma charakter jakościowy i doświadczenie oceniającego stanowi główne źródło ocen, mających charakter strukturalny.

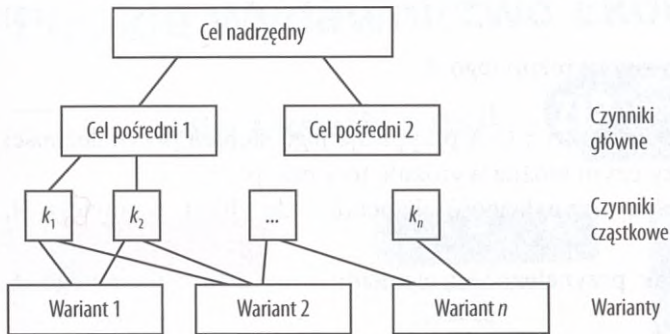
Cel nadrzędny zdefiniowany jest jako stan docelowy. Jest on wynikiem realizacji celu decyzyjnego. Na przykład, uzyskania optymalnego zysku i zadowalającej jakości. Pierwsza faza opracowania metody modelu hierarchicznego zadania wspomaganie decyzji polega na zdefiniowaniu celu nadrzędnego. Jest to poziom najwyższy. Na poziomie niższym rozważa się kryteria decyzyjne wpływające na stopień osiągnięcia celu nadrzędnego. Liczba poziomów pośrednich zależy od złożoności problemu i przyjętego modelu problemu decyzyjnego. Na poziomie najniższym rozważa się warianty decyzyjne. Metodę wspomaganie procesu decyzyjnego w modelu hierarchicznym realizuje się w następujących etapach.

1. Budowa modelu hierarchicznego. Dekompozycja problemu decyzyjnego i ustalenie hierarchii czynników (kryteriów) wpływających na rozwiązanie problemu.
2. Porównanie parami kryteriów i wariantów decyzyjnych.
3. Wyznaczenie preferencji, określenie wzajemnych priorytetów w odniesieniu do kryteriów i wariantów decyzyjnych.
4. Klasyfikacja wariantów decyzyjnych. Identyfikacja uporządkowania wariantów decyzyjnych ze względu na ich udział w osiągnięciu celu nadrzędnego.

Podstawą analizy hierarchicznego problemu decyzyjnego jest modelowanie graficznie hierarchii celów. Prowadzi to do przedstawienia problemu w postaci drzewa hierarchicznego, co pozwala opisać strukturę decyzyjną problemu. Wiadomo teraz, że osiągnięcie celu głównego przez każdy z wariantów wynika ze spełnienia celów pośrednich, wyrażonych przez odpowiednie kryteria. Hierarchiczna struktura decyzyjna jest przedstawiona na rysunku 4.10.

Występujące w modelu czynniki główne i cząstkowe stanowią kryteria oceny wariantów pod względem zysków, dochodów, jakości badanego systemu. Wariantami decyzyjnymi są rozwiązania. W modelu hierarchicznym zapewniona jest ilościowa i jakościowa porównywalność kryteriów z tego samego poziomu. Występujące w modelu czynniki główne i cząstkowe oceny wariantów stanowią kryteria oceny jakości decyzji pod względem ewentualnych zysków. W modelu hierarchicznym zysk jest celem nadrzędnym.

Rysunek 4.10. Schemat struktury hierarchicznej



Źródło: Opracowanie własne.

## 4.2. Zbiory rozmyte w modelowaniu problemów decyzyjnych

Teoria zbiorów rozmytych L. Zadeha opisuje zjawisko przynależności elementu do zbioru „w pewnym stopniu” wyrażanym wartością liczbową z przedziału zamkniętego  $[0, 1]$ .

Nieostre określenie przynależności elementu do zbioru zapoczątkowało szeroki strumień badań nad praktycznym wykorzystaniem pojęć nieostrych w procesach rozpoznawania obrazów i w konsekwencji w podejmowaniu decyzji w sytuacjach opisanych w sposób niejednoznaczny [Zadeh, 1965].

### 4.2.1. Koncepcja zbiorów rozmytych L. Zadeha

Za pomocą **zbiorów rozmytych** można w sposób formalny wyrazić pojęcia nieprecyzyjne [Szmuc 2002], np. szybki wzrost gospodarczy, gwałtowny spadek indeksów giełdowych lub duża aglomeracja. Zanim zdefiniujemy zbiór rozmyty, niezbędne jest ustalenie tzw. obszaru rozważań (ang. *the universe of discourse*). W przypadku pojęcia wieloznacznego, np. duże gospodarstwo rolne: inny areał użytków rolnych będzie uważany za duży, jeśli przyjmiemy obszar rozważań w przedziale  $[1, 50]$ , a inny zaś — jeśli przyjmiemy przedział  $[20, 1000]$ . Obszar rozważań będziemy nazywać w dalszym ciągu przestrzenią lub zbiorem i oznaczać literą  $X$ , gdzie  $X$  jest zbiorem nierozmytym.

#### Definicja 4.2

Według L. Rutkowskiego [Rutkowska, Pliński, Rutkowski 1999] **zbiorem rozmytym**  $A$  w pewnej (niepustej) przestrzeni  $X$ , co zapisujemy jako  $A \subseteq X$ , nazywamy zbiór par:

$$A = \{(x, \mu_A(x)); x \in X\}, \quad [4.2]$$

w którym:

$$\mu_A: X \rightarrow [0,1] \quad [4.3]$$

jest funkcją przynależności zbioru rozmytego  $A$ .

Funkcja ta każdemu elementowi  $x \in X$  przypisuje jego stopień przynależności do zbioru rozmytego  $A$ , przy czym można wyróżnić trzy przypadki:

- 1)  $\mu_A(x) = 1$  oznacza pełną przynależność elementu  $x$  do zbioru rozmytego  $A$ , tzn.  $x \in A$ ;
- 2)  $\mu_A(x) = 0$  oznacza brak przynależności elementu  $x$  do zbioru rozmytego  $A$ , tzn.  $x \notin A$ ;
- 3)  $0 < \mu_A(x) < 1$  oznacza częściową przynależność elementu  $x$  do zbioru rozmytego  $A$ .

Przykłady 4.2–4.3 są ilustracją różnych zbiorów rozmytych.

#### Przykład 4.2

Niech  $N$  jest zbiorem liczb naturalnych. Określimy pojęcie zbioru liczb naturalnych „bliskich liczby 10”. W tym celu można zdefiniować zbiór:

$$A = \{(7, 0,2), (8, 0,5), (9, 0,8), (11, 0,8), (12, 0,5), (13, 0,2)\}.$$

#### Przykład 4.3

Niech  $X = R$  i  $R$  jest zbiorem liczb rzeczywistych. Zbiór liczb rzeczywistych „bliskich liczby 10” można zdefiniować za pomocą funkcji przynależności w postaci:

$$\mu_A(x) = \frac{1}{1+(x-10)^2}. \quad [4.4]$$

Funkcje przynależności można definiować na wiele różnych sposobów.

W przykładzie 4.4 podaje się kilka wzorów funkcji przynależności.

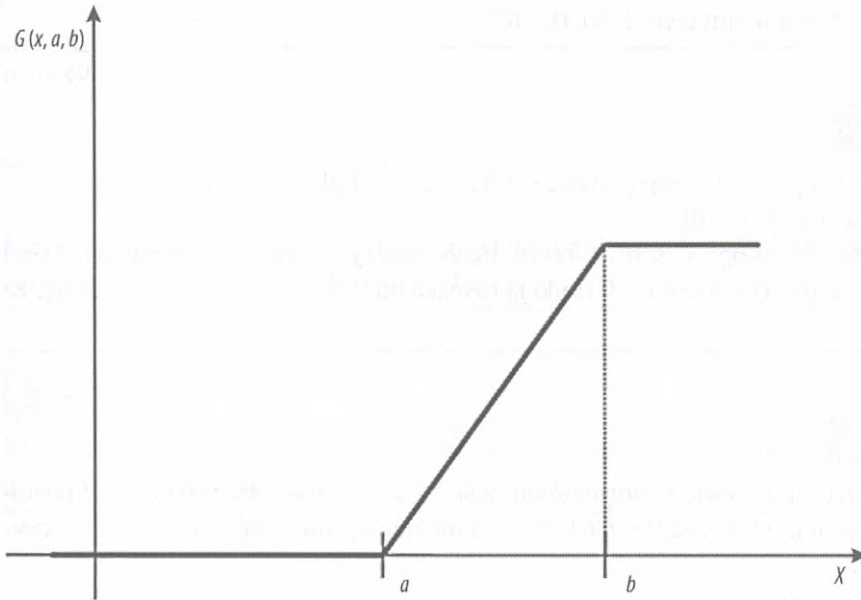
#### Przykład 4.4

**Funkcję przynależności** można zdefiniować jako:

$$g(x, a, b) = \begin{cases} 0 & \text{dla } x \leq a, \\ x-a & \text{dla } a < x \leq b, \\ b-a & \\ 1 & \text{dla } x > b. \end{cases} \quad [4.5]$$

Wykres funkcji przynależności  $g(x, a, b)$  pokazano na rysunku 4.11. Można zauważyć, że jest to dystrybuanta rozkładu jednostajnego w przedziale  $(a, b)$ .

Rysunek 4.11. Wykres funkcji przynależności  $g(x, a, b)$  z przykładu 4.3



Źródło: [Rutkowska, Pliński, Rutkowski 1999].

Oto definicje określające podstawowe własności zbiorów rozmytych.

#### Definicja 4.3

Zbiór elementów przestrzeni  $X$ , dla których  $\mu_A(x) > 0$ , nazywamy **nośnikiem zbioru rozmytego**  $A$  i oznaczamy przez  $\text{supp } A$  (ang. *support*). Można to zapisać następująco:

$$\text{supp } A = \{x \in X; \mu_A(x) > 0\}. \quad [4.6]$$

#### Definicja 4.4

**Wysokość zbioru rozmytego**  $A$  oznacza się przez  $h(A)$  i definiuje jako:

$$h(A) = \sup_{x \in X} \mu_A(x), \quad [4.7]$$

gdzie:  $\sup$  — kres górny zbioru, w przypadku zbioru skończonego jest to element maksymalny tego zbioru.

W przykładach 4.5–4.8 wyznacza się nośniki zbiorów rozmytych i ich wysokości.

#### Przykład 4.5

Jeżeli  $X = \{1, 2, 3, 4, 5, 6, 7\}$  oraz  $A = \{(1, 0,2), (2, 0,4), (3, 0,7)\}$ , to  $\text{supp } A = \{1, 2, 4\}$ . Wysokość zbioru rozmytego  $A$ ,  $h(A) = 0,7$ .

#### Przykład 4.6

Jeżeli  $X = R$ ,  $\mu_A(x) = 1 - \exp(-x)$  dla  $x \geq 0$  i  $\mu_A(x) = 0$  dla  $x \leq 0$ , to:  $\text{supp } X = \{x; x \notin R, x > 0\}$ .

Oznacza to, że  $\text{supp } X$  jest zbiorem liczb rzeczywistych nieujemnych. Jeżeli  $\mu_A(x) = 1 - \exp(-x)$  jest dla  $x \geq 0$  funkcją rosnącą od 0 do 1, to stąd wnioskuje się, że  $h(A) = 1$ .

#### Definicja 4.5

Zbiór rozmyty  $A$  nazywamy normalnym, jeśli  $h(A) = 1$ . Jeśli zbiór rozmyty  $A$  posiada wysokość  $h(A)$  taką, że  $0 < h(A) < \infty$ , to można wprowadzić nową funkcję przynależności:

$$\mu_{AN}(x) = \frac{\mu_{AN}(x)}{h(A)}, \quad [4.8]$$

gdzie:  $\sup_{x \in X} \mu_A(x) = 1$ .

#### Przykład 4.7

Dany jest zbiór rozmyty  $A = \{(2, 0,1), (4, 0,5), (6, 0,3)\}$ . Po znormalizowaniu mamy zbiór rozmyty w postaci:  $A_N = \{(2, 0,2), (4, 1), (6, 0,6)\}$ .

#### Definicja 4.6

Zbiór rozmyty  $A$  jest pusty, co zapisujemy jako  $A = \emptyset$ , jeśli  $\mu_A(x) = 0$  dla każdego  $x \in X$ .

#### Definicja 4.7

Zbiór rozmyty  $A$  zawiera się w zbiorze rozmytym  $B$ , co zapisujemy jako  $A \subset B$ , jeśli  $\mu_A(x) \leq \mu_B(x)$  dla każdego  $x \in X$ .



## Przykład 4.8

Rozważamy zbiory rozmyte  $A$  i  $B$  z funkcjami przynależności zadanymi wzorami:

$$\mu_A(x) = \begin{cases} 0 & \text{dla } x \leq 0, \\ x & \text{dla } 0 < x \leq 1, \\ 1 & \text{dla } x > 1, \end{cases} \quad [4.9]$$

natomiast:

$$\mu_B(x) = \begin{cases} 0 & \text{dla } x \leq 2, \\ x-2 & \text{dla } 2 < x \leq 4, \\ 3-x & \text{dla } x > 4. \end{cases} \quad [4.10]$$

Łatwo można sprawdzić, że  $\mu_A(x) \geq \mu_B(x)$  dla każdego  $x \in X$ . Definicja ta jest podstawową definicją w teorii zbiorów rozmytych.

## Definicja 4.8

Zbiór rozmyty  $A$  jest równy zbiorowi rozmytemu  $B$ , co zapisujemy jako  $A = B$ , jeśli  $\mu_A(x) = \mu_B(x)$  dla każdego  $x \in X$ .

## 4.2.2. Operacje na zbiorach rozmytych

W tym podpunkcie podamy podstawowe operacje na zbiorach rozmytych.

## Definicja 4.9

**Przekrojem (przecięciem) zbiorów rozmytych**  $A, B \subset X$  jest zbiór rozmyty o funkcji przynależności postaci:

$$\mu_{A \cap B}(x) = \min \{ \mu_A(x), \mu_B(x) \}, \quad [4.11]$$

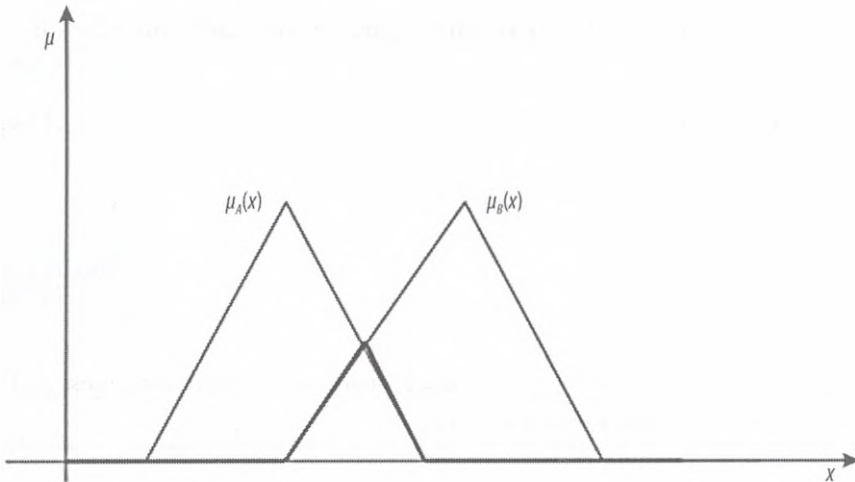
dla każdego  $x \in X$ .

Prawą stroną równości [4.11] zapisuje się następująco:

$$\min \{ \mu_A(x), \mu_B(x) \} = \mu_A(x) \wedge \mu_B(x).$$

Graficznie operację przekroju zbiorów rozmytych przedstawiono na rysunku 4.12.

Rysunek 4.12. Działanie przecięcia zbiorów rozmytych



Źródło: Jak rysunku 4.11.

Operację przecięcia zbiorów rozmytych bardzo łatwo można uogólnić na  $n$  argumentów. Przecięciem zbiorów rozmytych  $A_1, A_2, \dots, A_n$  określa się jako:

$$\mu_{A_1 \cap A_2 \cap \dots \cap A_n}(x) = \min \{ \mu_{A_1}(x), \mu_{A_2}(x), \dots, \mu_{A_n}(x) \}, \quad [4.12]$$

dla każdego  $x \in X$ .

Podobnie definiuje się sumę zbiorów rozmytych.

#### Definicja 4.10

**Sumą zbiorów rozmytych**  $A$  i  $B$  (rysunek 4.13) jest zbiór rozmyty  $A \cup B$  z funkcją przynależności w postaci:

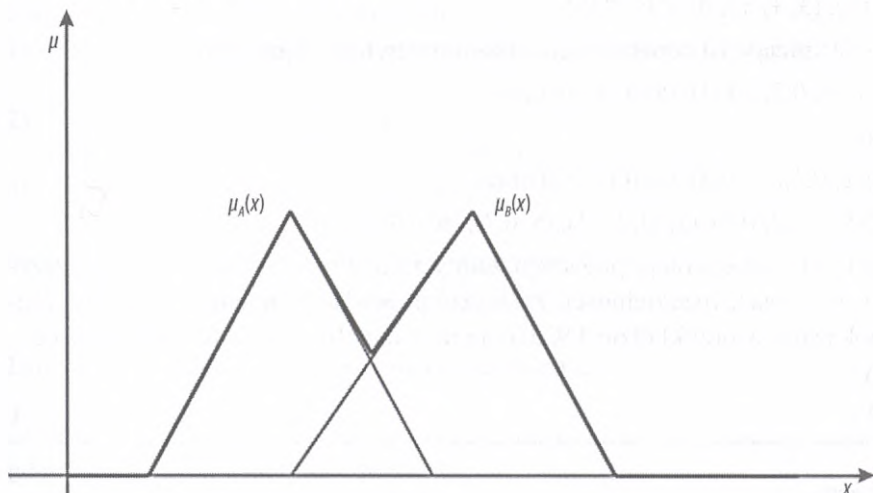
$$\mu_{A \cup B}(x) = \max \{ \mu_A(x), \mu_B(x) \}, \quad [4.13]$$

dla każdego  $x \in X$ .

Prawą stronę równości [4.13] można zapisać jako:

$$\max \{ \mu_A(x), \mu_B(x) \} = \mu_A(x) \vee \mu_B(x).$$

Rysunek 4.13. Suma zbiorów rozmytych



Źródło: Jak rysunku 4.11.

#### Przykład 4.9

Zakładamy, że  $X = \{1, 2, 3, 4, 5, 6, 7\}$  oraz

$$A = \{(3, 0,9), (4, 1), (6, 0,6)\},$$

$$B = \{(3, 0,7), (5, 1), (6, 0,4)\}.$$

Na mocy definicji 4.8 można zapisać, że:

$$A \cap B = \{(3, 0,7), (6, 0,4)\}.$$

Na mocy definicji 4.9 można zapisać, że:

$$A \cup B = \{(3, 0,9), (4, 1), (5, 1), (6, 0,6)\}.$$

#### Definicja 4.11

**Dopełnieniem zbioru rozmytego**  $A \subset X$  jest zbiór rozmyty  $\hat{A}$  o funkcji przynależności  $\mu_{\hat{A}}(x) = 1 - \mu_A(x)$ ,  
dla każdego  $x \in X$ . [4.14]

#### Przykład 4.10

W przykładzie pokażemy, że operacja dopełnienia zbioru rozmytego prowadzi do zbioru, który nie posiada własności klasycznej operacji dopełnienia zbioru. Zakładamy, że:

$X = \{1, 2, 3, 4, 5, 6\}$  oraz

$A = \{(2, 0,3), (3, 1), (5, 0,7), (6, 0,9)\}$ .

Zgodnie z definicją 4.10 dopełnieniem zbioru rozmytego  $A$  jest zbiór:

$\hat{A} = \{(1, 1), (2, 0,7), (4, 1), (5, 0,3), (6, 0,1)\}$ .

Zauważmy, że:

$A \cap \hat{A} = \{(2, 0,3), (5, 0,3), (6, 0,1)\} \neq 0$  oraz

$A \cup \hat{A} = \{(1, 1), (2, 0,7), (3, 1), (4, 1), (5, 0,7), (6, 0,9)\} \neq X$ .

Można wykazać, że operacje przekroju, sumy i dopełnienia mają własności przemienności, łączności, rozdzielności, zachodzą prawa de Morgana i absorpcji. Jednak, jak pokazano w przykładzie 4.9, dla operacji dopełnienia zachodzą równości:

$A \cap \hat{A} \neq 0$ ,

$A \cup \hat{A} \neq X$ .

#### Definicja 4.12

**Iloczyn kartezjański zbiorów rozmytych**  $A \subset X$  i  $B \subset Y$  oznaczamy  $A \times B$  i definiujemy jako:

$$\mu_{A \times B}(x, y) = \mu_A(x) \wedge \mu_B(y) = \min \{\mu_A(x), \mu_B(y)\}, \quad [4.15]$$

lub  $\mu_{A \times B}(x, y) = \mu_A(x) \mu_B(y)$ .

#### Przykład 4.11

Zakładamy, że  $X = \{1, 2\}$ ,  $Y = \{1, 2, 3\}$  oraz

$A = \{(1, 0,5), (2, 0,9)\}$ ,

$B = \{(2, 0,3), (2, 0,7), (3, 0,1)\}$ .

Stosując definicję 4.11 iloczynu kartezjańskiego zbiorów rozmytych  $A$  i  $B$ , otrzymujemy:

$A \times B = \{((1, 1), 0,3), ((1, 2), 0,5), ((1, 3), 0,1), ((2, 1), 0,3), ((2, 2), 0,7), ((2, 3), 0,1)\}$ .

### 4.2.3. Relacje rozmyte i ich własności

Punkt rozpoczyna definicja pojęcia norm trójkątnych, która jest niezbędna do określenia operacji złożenia relacji rozmytych.

#### Definicja 4.13

Funkcję  $T$  dwóch zmiennych rzeczywistych nazywamy  **$T$ -normą**, jeśli:

$$T: [0, 1] \times [0, 1] \rightarrow [0, 1]$$

oraz spełnione są następujące postulaty:

- 1) funkcja  $T$  jest nierosnąca względem obu argumentów; oznacza to, że: jeśli  $a \leq b$  i  $c \leq d$ , to  $T(a, c) \leq T(b, d)$ ,
- 2) funkcja  $T$  ma własność symetrii: dla dowolnych  $a$  i  $b$  zachodzi  $T(a, b) = T(b, a)$ ,
- 3) funkcja  $T$  spełnia warunek łączności:  $T(T(a, b), c) = T(a, T(b, c))$ ,
- 4) funkcja  $T$  spełnia warunki brzegowe postaci:  $T(a, 0) = 0$ ,  $T(a, 1) = a$ , gdzie:  $a, b, c, d \in [0, 1]$ .

Dowolna  $T$ -norma spełnia następujące nierówności:

$$T_w(a, b) \leq T(a, b) \leq \min(a, b),$$

gdzie:  $T_w(a, b)$  jest  $T$ -normą w postaci:

$$T_w(a, b) = \begin{cases} a & \text{dla } b = 1, \\ b & \text{dla } a = 1, \\ 0 & \text{dla } a \neq 1, b \neq 1. \end{cases} \quad [4.16]$$

Zdania typu „ $x$  jest prawie równe” lub „ $x$  jest znacznie większe od  $y$ ” mogą być sformalizowane za pomocą relacji rozmytych. Pojęcie relacji rozmytej jest jednym z podstawowych pojęć teorii zbiorów rozmytych. Oto definicja relacji rozmytej oraz złożenia relacji rozmytej.

#### Definicja 4.14

**Relacją rozmytą**  $R$  między dwoma niepustymi i nierozmytymi zbiorami  $X$  i  $Y$  nazywamy zbiór rozmyty określony na iloczynie kartezjańskim, tzn.:

$$R \subset X \times Y = \{(x, y); x \in X, y \in Y\}. \quad [4.17]$$

Oznacza to, że relacja rozmyta jest zbiorem par:

$$R = \{((x, y), \mu_R(x, y))\}, \quad [4.18]$$

gdzie:

$$\mu_R: X \times Y \rightarrow [0, 1] \text{ jest funkcją przynależności.}$$

Funkcja ta każdej parze  $(x, y)$ ,  $x \in X$ ,  $y \in Y$  przypisuje stopień przynależności  $\mu_R(x, y)$ , który interpretujemy jako stopień powiązania między elementami  $x \in X$  i  $y \in Y$ .

## Przykład 4.12

Niech  $X = \{3, 4, 5\}$ ,  $Y = \{4, 5, 6\}$ . Zastosujemy definicję 4.11 do sformalizowania nieprecyzyjnego sformułowania „ $y$  jest mniej więcej równe  $x$ ”. Wiadomo, że:

$$X \times Y = \{(3, 4), (3, 5), (3, 6), (4, 4), (4, 5), (4, 6), (5, 4), (5, 5), (5, 6)\}.$$

Relację rozmytą można w tym przypadku zdefiniować następująco:

$$R = \{((3, 4), 0,8), ((3, 5), 0,6), ((3, 6), 0,4), ((4, 4), 1), ((4, 5), 0,8), ((4, 6), 0,6), ((5, 4), 0,8), ((5, 5), 1), ((5, 6), 0,8)\}.$$

Analiza relacji  $R$  prowadzi do wniosku, że funkcję przynależności  $\mu_R(x, y)$  relacji  $R$  można zapisać w postaci:

$$\mu_R(x, y) = \begin{cases} 1 & \text{dla } x = y, \\ 0,8 & \text{dla } |x - y| = 1, \\ 0,6 & \text{dla } |x - y| = 2, \\ 0,2 & \text{dla } |x - y| = 3. \end{cases} \quad [4.18]$$

## Przykład 4.13

Niech  $X = [0, 120]$ ,  $Y = [0, 100]$  oznaczają odpowiednio prędkości samochodu osobowego i ciężarowego. Relacja  $R$  o funkcji przynależności:

$$\mu_R(x, y) = \begin{cases} 0 & \text{dla } x - y \leq 1, \\ x - y & \text{dla } 0 < x - y \leq 40, \\ 40 & \\ 1 & \text{dla } x - y > 2 \end{cases} \quad [4.19]$$

reprezentuje nieprecyzyjne stwierdzenie „samochód osobowy  $x$  jedzie dużo szybciej od samochodu ciężarowego  $y$ ”. Z definicji relacji rozmytej widać, że wprowadzone wcześniej pojęcia przecięcia, sumy i dopełnienia można zastosować do relacji. W szczególności:

$$\mu_{R \cap S}(x, y) = \min(\mu_R(x, y), \mu_S(x, y)), \quad [4.20]$$

$$\mu_{R \cup S}(x, y) = \max(\mu_R(x, y), \mu_S(x, y)), \quad [4.21]$$

$$\mu_{\bar{R}}(x, y) = 1 - (\mu_R(x, y)). \quad [4.22]$$

#### 4.2.4. Operacje wnioskowania na zbiorach rozmytych

**Reguły wnioskowania** w logice rozmytej (reguły rozmytej implikacji) stanowią podstawę ich zastosowań w praktyce.

**Modele wnioskowania** dzielą się na dwie zasadnicze kategorie:

- 1) **modele lingwistyczne**, których podstawą jest zbiór reguł „jeśli-to”, stanowiących jakościowy opis systemu najbardziej bliski językowi naturalnemu;
- 2) **modele oparte na wnioskowaniu Takagi–Sugeno**; są one tworzone przez reguły logiczne, które mają rozmytą część poprzedników i funkcyjny następnik, a w istocie są kombinacją modeli rozmytych i klasycznych.

**Modele rozmyte** z wnioskowaniem typu Mamdani oparte są na procesie przyczynowo-skutkowym typu Mamdaniego, przyjmującym za podstawę bazę reguł i stosowanie operatorów lingwistycznych. Jest to podejście najbardziej naturalne z punktu widzenia logiki rozmytej, i tym samym jest najszerzej upowszechnione w zastosowaniach.

Zakładamy, że prognozowany jest dynamiczny rozwój sektora  $x$  gospodarki. Zużycie wody przez sektor  $x$  jest bardzo niskie, prognoza zużycia oparta na wzroście produkcji i wymaganych do produkcji zasobach prowadzi do wniosku, że rozwój gospodarczy w tym sektorze nie będzie miał wpływu na zużycie wody. Na podstawie tablic przepływów międzygałęziowych można wyznaczyć sektory, z których produkcji korzysta sektor  $x$ . Okazuje się, że sektor ten korzysta z produkcji bardzo wodochłonnych sektorów. Prognoza wzrostu zapotrzebowania na produkty sektora  $x$  jest przekładana na związany z tym wzrost produkcji w pozostałych sektorach, a następnie wyznaczany jest wzrost zużycia wody w tych sektorach. Ostatecznie określa się, że rozwój gospodarczy w sektorze  $x$  będzie miał wpływ na zużycie wody.

Modelowanie zużycia zasobów wodnych dotyczy dwóch najistotniejszych źródeł poborów — wód powierzchniowych i wód podziemnych. Stąd macierz  $\mathbf{B}$  będzie dwuwierszowa i będzie zawierać współczynniki zużycia wód powierzchniowych oraz współczynniki zużycia wód podziemnych przez poszczególne sektory gospodarki [Godyń, Chmielowski 2008].

Podstawowym elementem wnioskowania rozmytego jest pojęcie zmiennej lingwistycznej (np. „wodochłonność produkcji”), która przyjmuje wartości lingwistyczne, takie jak: „niska”, „bardzo niska”, „średnia”, „wysoka”, „bardzo wysoka” itp. Wartościom lingwistycznym są przypisywane odpowiednie zbiory rozmyte, a zależności między zmiennymi lingwistycznymi — rozmytymi zdaniami warunkowymi. Na przykład, mamy dwie zmienne lingwistyczne  $L$  i  $K$  takie, że wartość zmiennej  $L$  jest zbiorem rozmytym  $A$  określonym w  $X$  oraz wartość zmiennej  $K$  jest zbiorem rozmytym  $B$  określonym w  $Y$ , to zależność między  $L$  i  $K$ , a właściwie między wartościami  $A$  i  $B$ , można zapisać jako:

JEŻELI  $L = A$  TO  $K = B$ .

Wnioskowanie rozmyte oparte na logice zbiorów rozmytych polega na wyciąganiu wniosków na podstawie reguł opartych na wartościach lingwistycznych. Najczęściej stosowana jest architektura Mamdaniego, w której na podstawie wiedzy eks-

perta tworzy się bazę reguł postaci, np.: dla dwóch zmiennych wejściowych  $x_1$  i  $x_2$ , zmiennej wyjściowej  $y$ , przyjmujących po trzy wartości lingwistyczne (ujemny  $U$ , zero  $Z$ , dodatni  $D$ ) możliwe jest utworzenie 9 reguł odpowiadających wszystkim kombinacjom zmiennych wejściowych. Reguły te mogą mieć postać: Jeżeli  $x_1$  przyjmuje wartość *ujemną* i  $x_2$  przyjmuje wartość *ujemną*, to wielkość wyjściowa  $y$  przyjmuje wartość *ujemną*, co można zapisać jako [Godyń, Chmielowski 2008]:

1) JEŻELI  $x_1 = U$  ORAZ  $x_2 = U$  TO  $y = U$ ,

i kolejne reguły:

2) JEŻELI  $x_1 = U$  ORAZ  $x_2 = Z$  TO  $y = U$ ,

3) JEŻELI  $x_1 = U$  ORAZ  $x_2 = D$  TO  $y = Z$ ,

4) JEŻELI  $x_1 = Z$  ORAZ  $x_2 = U$  TO  $y = U$ ,

5) JEŻELI  $x_1 = Z$  ORAZ  $x_2 = Z$  TO  $y = Z$ ,

6) JEŻELI  $x_1 = Z$  ORAZ  $x_2 = D$  TO  $y = D$ ,

7) JEŻELI  $x_1 = D$  ORAZ  $x_2 = U$  TO  $y = Z$ ,

8) JEŻELI  $x_1 = D$  ORAZ  $x_2 = Z$  TO  $y = D$ ,

9) JEŻELI  $x_1 = D$  ORAZ  $x_2 = D$  TO  $y = D$ .

#### 4.2.5. Przykłady procesów decyzyjnych na zbiorach rozmytych

Praca modelu Mamdaniego przebiega według następującej procedury. Do modelu wprowadzane są ostre wielkości zmiennych wejściowych, które są zamieniane na odpowiednie zbiory rozmyte (dana wartość ostra może odpowiadać jednemu lub dwóm zbiorom rozmytym, np. wartość  $-0,001$  będzie w pewnym stopniu przynależała do zbioru „ujemny” oraz do zbioru „zero”) i z tego powodu ostre wartości dwóch zmiennych mogą uruchomić od 1 do kilku reguł (w zależności od podziału uniwersum). Każda z tych reguł jest spełniona w pewnym stopniu, ponieważ wejścia miały pewne określone stopnie przynależności do odpowiadających im zbiorów rozmytych. Jeżeli przesłanka reguły składa się z dwóch przesłanek dotyczących dwóch wejść połączonych spójnikiem koniunkcyjnym „oraz”, to stopień przynależności do całej reguły liczy się najczęściej jako stopień przynależności do relacji będącej iloczynem dwóch zbiorów rozmytych; najczęściej stosowanym do obliczeń operatorem są operatory  $t$ -normy: minimum MIN oraz iloczyn algebraiczny PROD. W wyniku uruchomienia 4 reguł otrzymujemy 4 konkluzje i odpowiadające im wielkości wyjścia (o różnych stopniach przynależności). Końcowy wynikowy zbiór rozmyty jest otrzymywany jako suma konkluzji poszczególnych reguł, czyli suma zbiorów rozmytych będących wyjściami z poszczególnych reguł. Przynależność do takiej relacji obliczana jest najczęściej jako maksimum MAX lub inny operator typu  $s$ -normy, np. suma logiczna. W efekcie końcowym w bloku wnioskowania otrzymywana jest wartość zmiennej wyjściowej w postaci zbioru rozmytego. Ostatnim blokiem modelu jest blok wyostrażania — defuzyfikacji, który pozwala na



przekształcenie wyjściowego zbioru rozmytego do wyjścia w postaci wielkości ostrej. Opracowano wiele metod defuzyfikacji, a najpopularniejsze to metody: środka maksimum, środka ciężkości, środka sum — szeroko opisane w literaturze [Kasprzyk 2001], [Piegat 1999].

Modelowanie zmienności współczynnika zużycia wody należy przeprowadzić dla każdego z sektorów gospodarki. Z uwagi na fakt, iż niezbędne dane statystyczne są dostępne wyłącznie na poziomie głównych sektorów gospodarki w przykładzie dokonano uproszczenia struktury gospodarki do sektorów przedstawionych w tabelicy 4.1.

**Tabela 4.1. Schemat (podział) gospodarki przyjęty do modelowania**

Lp.	Sektor		Udział w poborach w 2004 r.		
			wody ogółem (%)	wody powierzchniowe (%)	wody podziemne (%)
1.	Rolnictwo	A	0,8	0,9	—
2.	Rybaństwo	B	9,0	10,6	—
3.	Górnictwo	C	0,8	0,1	0,5
4.	Przetwórstwo przemysłowe	D	5,9	5,2	8,9
5.	Energetyka	E-cn.	63,7	75,3	1,5
6.	Pobór wód	E-pob.	19,2	7,6	86,6
7.	Pozostałe	F-O	0,6	0,0	2,4

Źródło: [Godyń, Chmielowski 2008].

Na podstawie analizy danych historycznych przyjęto, że do modeli dynamiki wodochłonności (osobno wód podziemnych i powierzchniowych) poszczególnych sektorów gospodarki będą używane dwie zmienne — dynamika produkcji globalnej oraz dynamika nakładów inwestycyjnych w tych sektorach. Wyznaczono i zestawiono w tabelicy 4.2 historyczne wartości dynamiki (względnej zmiany w stosunku do roku poprzedniego) produkcji globalnej (PG), nakładów inwestycyjnych (NI), użycia wód powierzchniowych ( $W_{pw}$ ) i podziemnych ( $W_{pd}$ ).

Do modelowania zmienności współczynników zużycia wód wybrano modele wykorzystujące wnioskowanie rozmyte w architekturze Mamdaniego. Każdą ze zmiennych opisano przez 5 wartości lingwistycznych, którym przypisano odpowiednie zbiory rozmyte o funkcjach przynależności typu gaussowskiego. Zastosowano mechanizm inferencji MAX-MIN i metodę defuzyfikacji — metodę środka ciężkości.

**Tablica 4.2. Dynamika zmienności produkcji globalnej, nakładów inwestycyjnych i wodochłonności wód w sektorach gospodarki w 1993–2004**

Sektor	Zmiana	1993	1994	1995	1996	1997	1998	1999	2000	2001	2002	2003	2004
A	PG	0,015	-0,041	0,157	0,012	-0,105	-0,032	-0,093	-0,023	0,021	-0,066	-0,014	0,056
	NI	-0,017	0,058	0,237	0,311	-0,051	-0,188	0,009	-0,037	-0,092	-0,016	-0,025	0,161
	W <sub>pw</sub>	-0,108	-0,181	-0,178	-0,357	-0,072	0,062	-0,102	0,231	-0,261	0,126	-0,001	-0,054
B	PG	-0,297	-0,014	-0,171	0,041	0,011	-0,104	0,047	-0,176	-0,253	-0,344	0,283	-0,203
	NI	0,802	-0,544	-0,688	1,632	-0,218	0,242	0,305	-0,087	-0,280	-0,233	-0,042	0,562
	W <sub>pw</sub>	0,460	-0,050	0,147	-0,087	0,040	0,017	0,027	0,210	0,339	0,451	-0,120	0,331
C	PG	0,037	0,143	-0,029	-0,019	0,042	-0,150	-0,043	-0,006	-0,041	-0,014	-0,018	0,183
	NI	-0,062	0,121	-0,053	-0,060	-0,017	0,022	-0,004	-0,215	0,183	-0,046	-0,014	0,059
	W <sub>pw</sub>	-0,053	-0,123	0,066	-0,130	-0,127	-0,489	-0,032	-0,031	-0,152	-0,132	0,094	-0,420
	W <sub>pd</sub>	-0,306	-0,129	0,228	-0,031	-0,171	-0,347	0,053	-0,009	-0,049	0,388	-0,387	-0,254
D	PG	0,031	0,105	0,055	0,049	0,093	0,025	0,011	0,063	-0,034	-0,005	0,104	0,152
	NI	-0,125	0,355	0,142	0,254	0,179	0,152	-0,077	-0,113	-0,134	-0,015	0,117	0,134
	W <sub>pw</sub>	-0,137	-0,118	-0,050	-0,104	-0,129	-0,112	-0,142	-0,056	-0,152	0,002	-0,416	0,412
	W <sub>pd</sub>	-0,110	-0,136	-0,119	-0,070	-0,079	-0,154	-0,126	-0,171	-0,092	-0,039	-0,112	-0,138
E-en.	PG	-0,111	0,007	0,000	-0,012	0,023	-0,007	0,115	0,045	0,182	0,054	0,044	-0,025
	NI	0,141	0,097	0,223	0,135	-0,001	-0,039	-0,009	-0,203	0,149	-0,005	-0,127	-0,061
	W <sub>pw</sub>	0,116	-0,004	0,047	0,046	-0,032	-0,003	-0,128	-0,066	-0,159	-0,026	-0,015	0,029
	W <sub>pd</sub>	0,065	-0,084	-0,021	-0,071	0,013	-0,054	0,046	-0,016	-0,193	-0,090	0,006	0,038
E-pb.	PG	0,144	0,046	-0,189	0,028	0,068	0,037	0,105	0,222	0,041	0,136	0,012	-0,035
	NI	0,170	0,083	-0,066	0,026	0,027	-0,145	-0,077	-0,081	-0,051	-0,126	-0,024	0,220
	W <sub>pw</sub>	-0,161	-0,128	0,111	-0,077	-0,108	-0,094	-0,127	-0,222	-0,120	-0,178	-0,021	-0,019
	W <sub>pd</sub>	-0,149	-0,060	0,213	-0,003	-0,123	-0,069	0,078	-0,178	-0,078	-0,117	0,005	0,006
F-O	PG	0,025	0,009	0,127	0,122	0,121	0,111	0,090	0,096	0,032	0,023	0,011	0,025
	NI	0,004	0,012	0,148	0,189	0,309	0,197	0,125	0,076	-0,116	-0,137	-0,026	0,043
	W <sub>pw</sub>	0,853	0,356	1,731	-0,443	-0,340	-0,513	0,422	0,001	-0,145	0,043	0,054	-0,010
	W <sub>pd</sub>	-0,035	0,007	-0,150	-0,184	-0,094	-0,175	-0,073	-0,202	-0,123	-0,111	0,012	0,001

Źródło: Jak tablicy 4.1.

Parametry z tablicy 4.2 pozwalają na prowadzenie odpowiedniej gospodarki wodnej w kraju poprzez umożliwienie wyznaczania trendów przydatnych w prognozowaniu dynamiki zmian i współzależności produkcji globalnej, nakładów inwestycyjnych oraz wodochłonności wód powierzchniowych i podziemnych.

### 4.3. Zbiory przybliżone Z. Pawlaka w modelowaniu problemów decyzyjnych

**Teoria zbiorów przybliżonych** Z. Pawlaka oparta jest na logice wnioskowania indukcyjnego, w której rozstrzyganie prawdziwości hipotez odbywa się w sposób eksperymentalny. W tym kontekście w teorii zbiorów przybliżonych proces analizy danych polega na budowaniu prostych tablic decyzyjnych, których elementami są przypadki opisywane zbiorem atrybutów.

Podstawowym pojęciem koncepcji zbiorów jest pojęcie nierozróżnialności. Nierozróżnialność przypadków ma z kolei istotny wpływ na proces klasyfikacji, i w konsekwencji podjęcie decyzji o wyborze właściwych przypadków.

#### 4.3.1. Koncepcja zbiorów przybliżonych Z. Pawlaka

Teoria zbiorów przybliżonych, zaproponowana przez Z. Pawlaka [Pawlak 1982, s. 341–356], stanowi rozwinięcie klasycznej teorii zbiorów. Dostarcza ona narzędzi matematycznych, umożliwiających formalny opis wiedzy niepełnej i niedokładnej. Dlatego znalazła zastosowanie w procesie pozyskiwania wiedzy z ogromnych zbiorów danych i jest wykorzystywana również w SWD.

Zbiory przybliżone są zbiorami, których elementy są między sobą nierozróżnialne, a metodą ich rozróżnienia jest wyznaczenie wartości funkcji przynależności do zbioru przybliżonego. Funkcja ta przyjmuje wartości odpowiadające numerowi grupy, do której dany element został przypisany: 1 — gdy dany element należy do grupy 1, 2 — gdy dany element należy do grupy 2 itd. Dodatkowo jest wykorzystywana wartość 0, która zostaje przyporządkowana do elementów niesklasyfikowanych, tzn. takich, o których nie jesteśmy w stanie powiedzieć, do jakiej grupy należą.

Podstawowe działania na zbiorach przybliżonych są takie same, jak działania wykonywane na zbiorach klasycznych, ale dodatkowo wprowadzone są pojęcia charakterystyczne tylko dla zbiorów przybliżonych. Przede wszystkim w celu zdefiniowania zbioru przybliżonego stosuje się relację nierozróżnialności.

Niech  $SI = (U, A, V, f)$  będzie systemem informacyjnym, gdzie:  $U$  jest zbiorem obiektów,  $A$  — zbiorem atrybutów,  $V$  — dziedziną atrybutów, zaś  $f$  — funkcją informacji. W tablicy 4.3 zawarto informacje o wybranych silnikach, montowanych w samochodach osobowych marki Opel, gdzie:

$U = \{O1, O2, O3, O4, O5\}$  są przykładowymi silnikami,

$A = \{At1, At2, At3, At4, At5\}$  jest listą atrybutów charakteryzujących silniki,

$V = V_{At1} \cup V_{At2} \cup V_{At3} \cup V_{At4}$  jest dziedziną poszczególnych atrybutów,

$f: U \times A \Rightarrow V$  (np.  $f(O1, At2) = 996$ ,  $f(O3, At5) = „1,2 Twinport”$ ) są przykładami funkcji informacyjnych.

Tablica 4.3. Obiekty systemu informacyjnego SI

$U$	$A$				
	Liczba cylindrów ( $At1$ )	Pojemność skokowa ( $At2$ )	Zużycie paliwa ( $At3$ )	Emisja CO <sub>2</sub> ( $At4$ )	Model ( $At5$ )
O1	3	996	4,7	120	1,2 Twinport
O2	3	1242	4,4	120	1,2 Twinport
O3	4	1242	4,4	120	1,2 Twinport
O4	3	1242	4,4	131	1,3 CDTI
O5	4	1242	4,4	120	1,3 CDTI

Źródło: Opracowanie własne.

**Relacją nierozróżnialności** na zbiorze obiektów  $U$ , generowaną przez zbiór atrybutów  $B \subseteq A$  nazywa się relację [Komorowski, Polkowski, Skowron 1999, s. 3]:

$$IND_S(B) = \{(x, y) \in U \times U: \forall a \in B, a(x) = a(y)\}. \quad [4.23]$$

Relacja nierozróżnialności jako relacja równoważności wyznacza rodzinę klas abstrakcji. Poszczególne klasy abstrakcji nazywa się zbiorami elementarnymi ( $B$ -elementarnymi) dla formuły [4.23]. Wszystkie elementy zbioru  $B$ -elementarnego mają te same wartości wszystkich atrybutów należących do zbioru  $B$  (są nierozróżnialne względem tych atrybutów). Zbiór nazywamy dokładnym ( $B$ -dokładnym), jeśli jest sumą zbiorów elementarnych ( $B$  zbiorów elementarnych). Każdy zbiór, który nie jest sumą zbiorów  $B$ -elementarnych, jest zbiorem  $B$ -przybliżonym.

Dla systemu SI prezentowanego w tablicy 4.3 można wyznaczyć relacje nierozróżnialności, generowane przez różne zbiory atrybutów systemu. Jeżeli dane są następujące zbiory atrybutów:  $A_1 = (At1, At2, At3)$ ,  $A_2 = (At4)$ ,  $A_3 = (At3, At5)$ , to:

$$IND_{SI}(A_1) = \{(O1, O1), (O2, O2), (O3, O3), (O4, O4), (O5, O5), (O2, O4), (O4, O2), (O3, O5), (O5, O3)\},$$

$$IND_{SI}(A_2) = \{(O1, O1), (O2, O2), (O3, O3), (O4, O4), (O5, O5), (O1, O2), (O2, O1), (O1, O3), (O3, O1), (O1, O5), (O5, O1), (O2, O3), (O3, O2), (O2, O5), (O5, O2), (O3, O5), (O5, O3)\},$$

$IND_{SI}(A_3) = \{(O1, O1), (O2, O2), (O3, O3), (O4, O4), (O5, O5), (O2, O3), (O3, O2), (O4, O5), (O5, O4)\}$ .

Przedstawione relacje dzielą zbiór obiektów systemu SI na następujące klasy abstrakcji (zbiory elementarne):

$U/IND_{SI}(A_1) = \{\{O1\}, \{O2, O4\}, \{O3, O5\}\}$ ,

$U/IND_{SI}(A_2) = \{\{O1, O2, O3, O5\}, \{O4\}\}$ ,

$U/IND_{SI}(A_3) = \{\{O1\}, \{O2, O3\}, \{O4, O5\}\}$ .

Jeżeli dane są następujące zbiory obiektów:  $X_1 = \{O1, O2, O4\}$  i  $X_2 = \{O2, O3, O4, O5\}$ , to:

- zbiór  $X_1$  jest zbiorem  $A_1$ -dokładnym, ponieważ jest sumą zbiorów  $A_1$ -elementarnych:

$$X_1 = \{\{O1\} \cup \{O2, O4\}\};$$

- zbiór  $X_2$  jest zbiorem  $A_1$ -dokładnym, ponieważ jest sumą zbiorów  $A_1$ -elementarnych:

$$X_2 = \{\{O2, O4\} \cup \{O3, O5\}\};$$

- zbiór  $X_1$  jest zbiorem  $A_2$ -przybliżonym, ponieważ nie jest sumą zbiorów  $A_2$ -elementarnych: zawiera obiekty O1 i O2, ale nie zawiera obiektów O3 i O5, należących do tego samego zbioru elementarnego;
- zbiór  $X_2$  jest zbiorem  $A_2$ -przybliżonym, ponieważ nie jest sumą zbiorów  $A_2$ -elementarnych: zawiera obiekty O2, O3 i O5, ale nie zawiera obiektu O1, należącego do tego samego zbioru elementarnego;
- zbiór  $X_1$  jest zbiorem  $A_3$ -przybliżonym, ponieważ nie jest sumą zbiorów  $A_3$ -elementarnych: zawiera obiekty O2 i O4, ale nie zawiera obiektów O3 i O5, należących do tego samego zbioru elementarnego;
- zbiór  $X_2$  jest zbiorem  $A_3$ -dokładnym, ponieważ jest sumą zbiorów  $A_3$ -elementarnych:

$$X_2 = \{\{O2, O3\} \cup \{O4, O5\}\}.$$

Jeśli  $SI = (U, A, V, f)$  jest systemem informacyjnym [Komorowski, Polkowski, Skowron 1999, s. 4] takim, że  $B \subseteq A$  oraz  $X \subseteq U$ , to:

- $B$ -dolnym przybliżeniem (aproksymacją) zbioru  $X$  w systemie informacyjnym SI nazywamy zbiór:

$$B X = \{x \in U: I_{SI, B}(x) \subseteq X\}; \quad [4.24]$$

- $B$ -górnym przybliżeniem (aproksymacją) zbioru  $X$  w systemie informacyjnym SI nazywamy zbiór:

$$\bar{B}X = \{x \in U: I_{SI,B}(x) \cap X \neq \emptyset\}; \quad [4.25]$$

- $B$ -pozytywnym obszarem zbioru  $X$  w systemie informacyjnym SI nazywamy zbiór:

$$POS_B(x) = \underline{B}X; \quad [4.26]$$

- $B$ -brzegiem zbioru  $X$  w systemie informacyjnym SI nazywamy zbiór:

$$BN_B(x) = \bar{B}X - \underline{B}X; \quad [4.27]$$

- $B$ -negatywnym obszarem zbioru  $X$  w systemie informacyjnym SI nazywamy zbiór:

$$NEG_B(X) = U - \bar{B}X. \quad [4.28]$$

Z przedstawionych definicji można wyciągnąć następujące wnioski [Komorowski, Polkowski, Skowron 1999, s. 5]:

- $\rightarrow \underline{B}X \subseteq X \subseteq \bar{B}X$ ,
- $\rightarrow$  zbiór  $X$  jest  $B$ -dokładny, gdy  $\underline{B}X = \bar{B}X \Leftrightarrow BN_B(X) = \emptyset$ ,
- $\rightarrow$  zbiór  $X$  jest  $B$ -przybliżony, gdy  $\underline{B}X \neq \bar{B}X \Leftrightarrow BN_B(X) \neq \emptyset$ .

Dla przykładu prezentowanego w tablicy 4.3 można wyznaczyć aproksymacje dla zbiorów  $X_1$  i  $X_2$  w obszarze atrybutów  $A_3$ :

$$\underline{A_3}X_1 = \{O1\},$$

$$\bar{A_3}X_1 = \{O1, O2, O3, O4, O5\},$$

$$POS_{A_3} = \underline{A_3}X_1 = \{O1\},$$

$$BN_{A_3}(X_1) = \bar{A_3}X_1 - \underline{A_3}X_1 = \{O2, O3, O4, O5\} \text{ — potwierdzenie faktu,}$$

że zbiór  $X_1$  jest zbiorem  $A_3$ -przybliżonym,

$$\bar{A_3}X_2 = \{O2, O3, O4, O5\},$$

$$POS_{A_3}(X_2) = \underline{A_3}X_2 = \{O2, O3, O4, O5\},$$

$$BN_{A_3}(X_2) = \bar{A_3}X_2 - \underline{A_3}X_2 = \{\emptyset\} \text{ — potwierdzenie faktu,}$$

że zbiór  $X_2$  jest zbiorem  $A_2$ -dokładnym,

$$NEG_{A_3}(X_2) = U - \bar{A_3}X_2 = \{O1\}.$$

Każdy zbiór (przybliżony lub dokładny) można scharakteryzować ilościowo za pomocą współczynnika dokładności przybliżenia (aproksymacji). **Współczynnik aproksymacji zbioru**  $X$  w systemie informacyjnym SI względem zbioru atrybutów  $B$  wyraża się wzorem:

$$\alpha_B(X) = \frac{\text{card}(\text{POS}_B(X))}{\text{card}(\overline{BX})} = \frac{\text{card}(\underline{BX})}{\text{card}(\overline{BX})}, \quad [4.29]$$

gdzie:

$\text{card}(X)$  — liczność zbioru  $X$ .

Można zatem zauważyć, że:

- jeżeli  $X$  jest zbiorem dokładnym, to:  $\alpha_B(X) = 1$ ,
- jeżeli  $X$  jest zbiorem przybliżonym, to:  $0 \leq \alpha_B(X) < 1$ .

Współczynnik aproksymacji zbiorów  $X_1$  i  $X_2$  w obszarze atrybutów  $A_3$  wynosi:

$$\bullet \quad \alpha_{A_1}(X_1) = \frac{\text{card}(\text{POS}_{A_3}(X_1))}{\text{card}(\overline{A_3X_1})} = \frac{\text{card}(\underline{A_3X_1})}{\text{card}(\overline{A_3X_1})} = \frac{1}{5} = 0,2 \quad \text{— potwierdzenie faktu,}$$

że zbiór  $X_1$  jest zbiorem  $A_3$ -przybliżonym;

$$\bullet \quad \alpha_{A_1}(X_2) = \frac{\text{card}(\text{POS}_{A_3}(X_2))}{\text{card}(\overline{A_3X_2})} = \frac{\text{card}(\underline{A_3X_2})}{\text{card}(\overline{A_3X_2})} = \frac{4}{4} = 1 \quad \text{— potwierdzenie faktu,}$$

że zbiór  $X_2$  jest zbiorem  $A_3$ -dokładnym.

Atrybut  $a$  nazywamy zbędnym w  $B$ , gdy:  $\text{IND}_{\text{SI}}(B) = \text{IND}_{\text{SI}}(B - \{a\})$ ; w przeciwnym przypadku — niezbędnym w  $B$ . Zbiór atrybutów  $B$  nazywamy niezależnym w systemie informacyjnym SI, gdy każdy atrybut należący do  $B$  jest niezbędny w  $B$ . W przeciwnym przypadku zbiór  $B$  nazywamy zależnym. Zbiór atrybutów  $Q$  ( $Q \subseteq B$ ) nazywamy **reduktem**, przy założeniu że zbiór atrybutów  $Q$  jest niezależny i  $\text{IND}_{\text{SI}}(B) = \text{IND}_{\text{SI}}(Q)$ . Redukt  $R_{\text{SI}}(B)$  zbioru atrybutów  $B$  w systemie informacyjnym SI jest to najmniejszy zbiór atrybutów, przy którym zostaje zachowana dotychczasowa klasyfikacja (rozróżnialność) obiektów.

Zbiór wszystkich reduktów zbioru atrybutów  $B$  w systemie informacyjnym SI oznaczany jest przez  $\text{RED}_{\text{SI}}(B)$ . **Jądro (rdzeń) systemu informacyjnego**  $\text{CORE}_{\text{SI}}(B)$  jest częścią wspólną wszystkich reduktów tego systemu  $\text{RED}_{\text{SI}}(B)$ :

$$\text{CORE}_{\text{SI}}(B) = \bigcap_{R \in \text{RED}_{\text{SI}}(B)} R. \quad [4.30]$$

Jądro zbioru reduktów  $\text{CORE}_{\text{SI}}(B)$  zawiera wszystkie atrybuty niezbędne w zbiorze  $B$ .

W celu wyznaczenia zbioru wszystkich reduktów  $\text{RED}_{\text{SI}}$  zbioru prezentowanego w tabelicy 4.3 należy wyznaczyć relację nierozróżnialności dla całego zbioru atrybutów:

$$A = (At1, At2, At3, At4): \text{IND}_{\text{SI}}(A) = \{(O1), (O2), (O3), (O4), (O5)\}.$$

Następnie sprawdzane są zmiany w obrębie klasyfikacji obiektów, gdy ze zbioru usuniemy pojedynczy atrybut:

$IND_{SI}(A-At1) = \{(O1), (O2), (O3), (O4), (O5)\} \neq IND_{SI}(A)$  — atrybut  $At1$  jest niezbędny w systemie, ponieważ jego usunięcie powoduje utratę informacji o rozróżnialności obiektów  $O2$  i  $O3$ ;

$IND_{SI}(A-At2) = \{(O1), (O2), (O3), (O4), (O5)\} = IND_{SI}(A)$  — atrybut  $At2$  jest zbędny w systemie, ponieważ jego usunięcie nie powoduje zmiany informacji o rozróżnialności obiektów systemu;

$IND_{SI}(A-At3) = \{(O1), (O2), (O3), (O4), (O5)\} = IND_{SI}(A)$  — atrybut  $At3$  jest zbędny w systemie, ponieważ jego usunięcie nie powoduje zmian informacji o rozróżnialności obiektów systemu;

$IND_{SI}(A-At4) = \{(O1, O3), (O2), (O4), (O5)\} = IND_{SI}(A)$  — atrybut  $At4$  jest zbędny w systemie, ponieważ jego usunięcie nie powoduje zmian informacji o rozróżnialności obiektów systemu;

$IND_{SI}(A-At5) = \{(O1), (O2), (O3, O5), (O4)\} \neq IND_{SI}(A)$  — atrybut  $At5$  jest niezbędny w systemie, jego usunięcie spowoduje utratę informacji o rozróżnialności obiektów  $O3$  i  $O5$ .

Przeprowadzona analiza wskazuje, że atrybuty  $At1$  i  $At5$  to wszystkie atrybuty niezbędne w systemie SI. Zatem  $CORE_{SI}(A) = \{At1, At5\}$ .

Redukt  $RED_{SI}(A)$  zawiera w sobie jądro  $CORE_{SI}(A)$  oraz pozwala na zachowanie klasyfikacji obiektów  $IND_{SI}(A)$  przy najmniejszej możliwej liczbie atrybutów. Można wyznaczyć dwa takie redukty:

$$IND_{SI}(At1, At2, At5) = IND_{SI}(A),$$

$$IND_{SI}(At1, At3, At5) = IND_{SI}(A).$$

### 4.3.2. Operacje wnioskowania na zbiorach przybliżonych

Jeżeli w tablicy 4.3 obiekty zastąpimy oznaczeniem reguły, atrybuty  $At1-At4$  — atrybutami warunkowymi ( $C$ ), a atrybut  $At5$  — decyzją ( $D$ ), to otrzymamy tablicę decyzyjną, wykorzystywaną w SWD (tablica 4.4). Definicja tablicy decyzyjnej przyjmuje postać:

$$TD = \langle U, C, D, V, f \rangle, \quad [4.31]$$

gdzie:

- $U$  — zbiór obiektów,
- $C$  — atrybuty warunkowe,
- $D$  — atrybuty decyzyjne,
- $V$  — dziedzina atrybutów,
- $f$  — funkcja decyzyjna.



Jeżeli  $v \in V_d$ , to klasą decyzyjną TD, odpowiadającą wartości  $v$  atrybutu decyzyjnego, nazywany jest zbiór  $X_v = \{u \in U: d(u) = v\}$ . Dla TD z tablicy 4.4 można wyróżnić dwie klasy decyzyjne:  $X_1 = \{1,2 \text{ Twinport}\}$  oraz  $X_2 = \{1,3 \text{ CDTI}\}$ .

**Tablica 4.4. Tablica decyzyjna TD**

$U$	$C$				$D$
	Liczba cylindrów ( $c1$ )	Pojemność skokowa ( $c2$ )	Zużycie paliwa ( $c3$ )	Emisja CO <sub>2</sub> ( $c4$ )	Model ( $d1$ )
R1	3	996	4,7	120	1,2 Twinport
R2	3	1242	4,4	120	1,2 Twinport
R3	4	1242	4,4	120	1,2 Twinport
R4	3	1242	4,4	131	1,3 CDTI
R5	4	1242	4,4	120	1,3 CDTI

Źródło: Opracowanie własne.

Treść zawartą w tablicy 4.4 można przedstawić w postaci reguł:

**R1:** Jeżeli  $c1 = 3$  i  $c2 = 996$  i  $c3 = 4,7$  i  $c4 = 120$  To  $d1 = 1,2 \text{ Twinport}$ .

**R2:** Jeżeli  $c1 = 3$  i  $c2 = 1242$  i  $c3 = 4,4$  i  $c4 = 120$  To  $d1 = 1,2 \text{ Twinport}$ .

**R3:** Jeżeli  $c1 = 4$  i  $c2 = 1242$  i  $c3 = 4,4$  i  $c4 = 120$  To  $d1 = 1,2 \text{ Twinport}$ .

**R4:** Jeżeli  $c1 = 3$  i  $c2 = 1242$  i  $c3 = 4,4$  i  $c4 = 131$  To  $d1 = 1,3 \text{ CDTI}$ .

**R5:** Jeżeli  $c1 = 4$  i  $c2 = 1242$  i  $c3 = 4,4$  i  $c4 = 120$  To  $d1 = 1,3 \text{ CDTI}$ .

Jeżeli określimy reguły należące do dolnych i górnych przybliżeń każdej z wyznaczonych w tablicy 4.4 klas decyzyjnych, to otrzymamy:

- reguły pewne — należące do obszaru dolnego przybliżenia klasy decyzyjnej,
- reguły możliwe — należące do obszaru górnego przybliżenia klasy decyzyjnej,
- reguły przybliżone — należące do obszaru brzegowego klasy decyzyjnej, które nie mają jednoznacznie określonej wartości atrybutu decyzyjnego.

Dla tablicy decyzyjnej prezentowanej w tablicy 4.4, w obszarze zidentyfikowanych klas decyzyjnych  $X_1 = (R1, R2, R3)$  i  $X_2 = (R4, R5)$ , można wyznaczyć aproksymacje dla zbioru atrybutów warunkowych  $\bar{C} = (C1, C2, C3, C4)$ :

$$\text{IND}_{\text{SI}}(\bar{C}) = \{(R1, R1), (R2, R2), (R3, R3), (R4, R4), (R5, R5), (R3, R5), (R5, R3)\},$$

$$U/\text{IND}_{\text{SI}}(\bar{C}) = \{\{O1\}, \{O2\}, \{O3, O5\}, \{O4\}\},$$

$$\underline{C}X_1 = \{R1, R2\} \text{ — zbiór reguł pewnych,}$$

$$\bar{C}X_1 = \{R1, R2, R3, R5\} \text{ — zbiór reguł możliwych,}$$

$BN_C(X_1) = \underline{C}X_1 - \overline{C}X_1 = \{R3, R5\}$  — zbiór reguł przybliżonych,

$\underline{C}X_2 = \{R4\}$  — zbiór reguł pewnych,

$\overline{C}X_2 = \{R3, R4, R5\}$  — zbiór reguł możliwych,

$BN_C(X_1) = \overline{C}X_1 - \underline{C}X_1 = \{R3, R5\}$  — zbiór reguł przybliżonych.

Z przedstawionych rozważań wynika, że w tablicy decyzyjnej (tablica 4.4) występują trzy reguły pewne  $\{R1, R2, R4\}$  oraz dwie reguły przybliżone  $\{R3, R5\}$ . Teoria zbiorów przybliżonych dopuszcza przetwarzanie obydwu typów reguł, przykładowo określając dla reguł przybliżonych stopień ufności, z jakim można ją przyjąć. Jest to bardzo korzystne w sytuacji przetwarzania wiedzy niepełnej lub niepewnej.

Można również spróbować doprowadzić strukturę tabeli decyzyjnej do takiej postaci, w której będą występowały jedynie reguły pewne. Do sprawdzenia poprawności budowy tablicy decyzyjnej wykorzystuje się definicję **reguły deterministycznej i niedeterministycznej**.

Niech  $TD = (U, C, D, V, f)$  będzie tablicą decyzyjną. **Reguła w tablicy decyzyjnej TD jest deterministyczna**, gdy takim samym wartościom atrybutów warunkowych odpowiada taka sama wartość atrybutów decyzyjnych. Fakt ten możemy wyrazić za pomocą następującej zależności dla obiektów tablicy decyzyjnej:

$$\forall x, y \in U (\forall_{c \in C} f(x, c) = f(y, c)) \Rightarrow (\forall_{d \in D} (f(x, d) = f(y, d))). \quad [4.32]$$

$x \neq y$

**Reguła w tablicy decyzyjnej TD jest niedeterministyczna**, gdy równość atrybutów warunkowych nie implikuje równości atrybutów decyzyjnych, co można wyrazić następującą zależnością dla obiektów tablicy decyzyjnej:

$$\exists x, y \in U (\forall_{c \in C} f(x, c) = f(y, c)) \wedge (\exists_{d \in D} (f(x, d) \neq f(y, d))). \quad [4.33]$$

$x \neq y$

W obszarze atrybutów warunkowych  $C = \{C1, C2, C3, C4\}$  przykładowej tablicy decyzyjnej (tablica 4.4) reguły R3 i R5 są niedeterministyczne, bo takim samym wartościom poszczególnych atrybutów warunkowych nie odpowiada taka sama wartość decyzyjnej:

**R3:** Jeżeli  $c1 = 4$  i  $c2 = 1242$  i  $c3 = 4,4$  i  $c4 = 120$  To  $d1 = 1,2$  Twinport.

**R5:** Jeżeli  $c1 = 4$  i  $c2 = 1242$  i  $c3 = 4,4$  i  $c4 = 120$  To  $d1 = 1,3$  CDTI.

Tablica, która zawiera reguły niedeterministyczne, jest tablicą niespójną i źle określoną. Dla odróżnienia tablicę, która zawiera jedynie reguły deterministyczne, nazywamy spójną i dobrze określoną. W praktyce oznacza to, że na podstawie reguł prezentowanych w tablicy 4.4 nie można w sposób jednoznaczny wnioskować o typie modelu silnika samochodu. Eliminacja niespójności w tablicy decyzyjnej jest re-

alizowana dwoma sposobami [Rutkowski 2005, s. 40–41]: poprzez usunięcie reguł niedeterministycznych (tablica 4.5) lub poprzez rozszerzenie liczby atrybutów warunkowych (tablica 4.6).

**Tablica 4.5. Usuwanie niespójności TD poprzez usuwanie reguł niedeterministycznych**

<i>U</i>	<i>C</i>				<i>D</i>
	Liczba cylindrów ( <i>c1</i> )	Pojemność skokowa ( <i>c2</i> )	Zużycie paliwa ( <i>c3</i> )	Emisja CO <sub>2</sub> ( <i>c4</i> )	Model ( <i>d1</i> )
R1	3	996	4,7	120	1,2 Twinport
R2	3	1242	4,4	120	1,2 Twinport
R4	3	1242	4,4	131	1,3 CDTI

Źródło: Opracowanie własne.

**Tablica 4.6. Usuwanie niespójności TD poprzez rozszerzenie liczby atrybutów warunkowych**

<i>U</i>	<i>C</i>					<i>D</i>
	Liczba cylindrów ( <i>c1</i> )	Pojemność skokowa ( <i>c2</i> )	Zużycie paliwa ( <i>c3</i> )	Emisja CO <sub>2</sub> ( <i>c4</i> )	Moc maksymalna ( <i>c5</i> )	Model ( <i>d1</i> )
R1	3	996	4,7	120	90	1,2 Twinport
R2	3	1242	4,4	120	60	1,2 Twinport
R3	4	1242	4,4	120	90	1,2 Twinport
R4	3	1242	4,4	131	90	1,3 CDTI
R5	4	1242	4,4	120	60	1,3 CDTI

Źródło: Opracowanie własne.

Łatwo sprawdzić, że wszystkie reguły prezentowane w tablicach 4.5–4.6 są deterministyczne, zatem tablice decyzyjne są spójne i dobrze określone.

Zidentyfikowanie atrybutów zbędnych w obszarze atrybutów warunkowych tablicy decyzyjnej umożliwia redukcję tej tablicy tylko do atrybutów niezbędnych. Posługując się tablicą decyzyjną z tablicy 4.6, można wyznaczyć relację nierozróżnialności dla całego zbioru atrybutów  $C = (C1, C2, C3, C4, C5)$ :

$$IND_{TD}(C) = \{(R1), (R2), (R3), (R4), (R5)\}.$$

Następnie należy sprawdzić zmiany, które nastąpią w klasyfikacji obiektów, kiedy ze zbioru usuniemy pojedynczy atrybut warunkowy:

$IND_{TD} \{(C-C1) = \{(R1), (R2), (R5), (R3), (R4)\} \neq IND_{TD}(C)$  — atrybut warunkowy  $C1$  jest niezbędny w systemie, ponieważ jego usunięcie powoduje utratę informacji o rozróżnialności obiektów  $R2$  i  $R5$ ;

$IND_{TD}(C-C2) = \{(R1), (R2), (R3), (R4), (R5)\} = IND_{TD}(C)$  — atrybut  $C2$  jest zbędny w systemie, ponieważ jego usunięcie nie powoduje zmiany informacji o rozróżnialności obiektów systemu;

$IND_{TD}(C-C3) = \{(R1), (R2), (R3), (R4), (R5)\} = IND_{TD}(C)$  — atrybut  $C3$  jest zbędny w systemie, ponieważ jego usunięcie nie powoduje zmian informacji o rozróżnialności obiektów systemu;

$IND_{TD}(C-C4) = \{(R1), (R2), (R3), (R4), (R5)\} = IND_{TD}(C)$  — atrybut  $C4$  jest zbędny w systemie, ponieważ jego usunięcie nie powoduje zmian informacji o rozróżnialności obiektów systemu;

$IND_{TD}(C-C5) = \{(R1), (R2), (R3), (R5), (R4)\} \neq IND_{TD}(C)$  — atrybut  $C5$  jest niezbędny w systemie, jego usunięcie spowoduje utratę informacji o rozróżnialności obiektów  $R3$  i  $R5$ .

Przeprowadzona analiza wskazuje, że atrybuty warunkowe  $C1$  i  $C5$  to wszystkie atrybuty niezbędne w tablicy decyzyjnej TD. Zatem jądro tablicy decyzyjnej  $CORE_{TD}(C) = \{C1, C5\}$ .

Tablica 4.7. Tablica zbudowana na podstawie jądra  $CORE_{TD}(C) = \{C1, C5\}$

$U$	$C$		$D$
	Liczba cylindrów ( $c1$ )	Moc maksymalna ( $c5$ )	Model ( $d1$ )
R1	3	90	1,2 Twinport
R2	3	60	1,2 Twinport
R3	4	90	1,2 Twinport
R4	3	90	1,3 CDTI
R5	4	60	1,3 CDTI

Źródło: Opracowanie własne.

Gdyby zawęzić definicję tablicy decyzyjnej do atrybutów  $C1$  i  $C5$  (tablica 4.7), wówczas uzyskalibyśmy pięć reguł, a w tym:

- trzy reguły pewne:

R2: Jeżeli  $c1 = 3$  i  $c5 = 60$  To  $d1 = 1,2$  Twinport,

R3: Jeżeli  $c1 = 4$  i  $c5 = 90$  To  $d1 = 1,2$  Twinport,

R5: Jeżeli  $c1 = 4$  i  $c5 = 60$  To  $d1 = 1,3$  CDTI;

- dwie reguły przybliżone:

R1: Jeżeli  $c1 = 3$  i  $c5 = 90$  To  $d1 = 1,2$  Twinport,

R4: Jeżeli  $c1 = 3$  i  $c5 = 90$  To  $d1 = 1,3$  CDTI.

Jak wynika z przedstawionego przykładu, zestawienie identycznych wartości atrybutów warunkowych  $C1$  i  $C5$  nie pozwala na jednoznaczne przypisanie wartości atrybutu decyzyjnego. Należy zatem rozszerzyć atrybutów warunkowe do takiego minimalnego podzbioru, który zapewni wygenerowanie reguł pewnych w obszarze całej tablicy. W tym celu należy posłużyć się definicją reduktu tablicy decyzyjnej.

Redukt  $RED_{TD}(C)$  zawiera w sobie jądro  $CORE_{TD}(C)$  oraz pozwala na zachowanie klasyfikacji obiektów  $IND_{TD}(C)$  przy najmniejszej możliwej liczbie atrybutów. Można wyznaczyć trzy takie redukty:  $IND_{TD}(C1, C2, C5) = IND_{TD}(C)$  oraz  $IND_{TD}(C1, C3, C5) = IND_{TD}(C)$  oraz  $IND_{TD}(C1, C4, C5) = IND_{TD}(C)$ . Zatem z tablicy 4.7 można przejść do postaci tablicy zredukowanej, zawierającej jeden z wyznaczonych wariantów reduktu:

$IND_{TD}(C1, C2, C5) \rightarrow$  (tablica 4.8),

$IND_{TD}(C1, C3, C5) \rightarrow$  (tablica 4.9),

lub  $IND_{TD}(C1, C3, C5) \rightarrow$  (tablica 4.10).

**Tablica 4.8. Zredukowana tablica wariant reduktu  $IND_{TD}(C1, C2, C5)$**

$U$	$C$			$D$
	Liczba cylindrów ( $c1$ )	Pojemność skokowa ( $c2$ )	Moc maksymalna ( $c5$ )	Model ( $d1$ )
R1	3	996	90	1,2 Twinport
R2	3	1242	60	1,2 Twinport
R3	4	1242	90	1,2 Twinport
R4	3	1242	90	1,3 CDTI
R5	4	1242	60	1,3 CDTI

Źródło: Opracowanie własne.

Tablica 4.9. Zredukowana tablica wariant reduktu  $IND_{TD}$  (C1, C3, C5)

U	C			D
	Liczba cylindrów (c1)	Zużycie paliwa (c3)	Moc maksymalna (c5)	Model (d1)
R1	3	4,7	90	1,2 Twinport
R2	3	4,4	60	1,2 Twinport
R3	4	4,4	90	1,2 Twinport
R4	3	4,4	90	1,3 CDTI
R5	4	4,4	60	1,3 CDTI

Źródło: Opracowanie własne.

Tablica 4.10. Zredukowana tablica wariant reduktu  $IND_{TD}$  (C1, C4, C5)

U	C			D
	Liczba cylindrów (c1)	Emisja CO <sub>2</sub> (c4)	Moc maksymalna (c5)	Model (d1)
R1	3	120	90	1,2 Twinport
R2	3	120	60	1,2 Twinport
R3	4	120	90	1,2 Twinport
R4	3	131	90	1,3 CDTI
R5	4	120	60	1,3 CDTI

Źródło: Opracowanie własne.

### 4.3.3. Przykłady procesów decyzyjnych na zbiorach przybliżonych

Teoria zbiorów przybliżonych znajduje współcześnie wiele zastosowań, wśród których należy wyróżnić: prognozowanie finansowe i ekonomiczne, rozpoznawanie obrazów, sygnałów i osób, uczenie się maszyn, systemy wspomaganie decyzji, systemy ekspertowe.

Zastosowanie zbiorów przybliżonych stwarza znacznie szersze możliwości wykorzystania posiadanych zbiorów danych w procesie pozyskiwania oraz przetwarzania wiedzy niepewnej i niepełnej. Oto dwa przykłady wykorzystania zbiorów przybliżonych w procesach decyzyjnych:

- zbiory przybliżone w analizie satysfakcji klienta serwisu pojazdów,
- zbiory przybliżone w diagnostyce aparatury paliwowej silników o zapłonie samoczynnym.

Przykład zastosowania zbiorów przybliżonych do analizy satysfakcji klienta serwisu pojazdów został opracowany na podstawie materiałów [Klimkiewicz, Moczulska 2008, s. 165–172]. Dane do analizy dotyczące klientów serwisu pozyskano z bazy danych serwisu zajmującego się naprawą pojazdów samochodowych. Badanie pomiaru satysfakcji klienta zostało przeprowadzone za pomocą programu komputerowego ROSE 2, opracowanego w Instytucie Informatyki Politechniki Poznańskiej [Prędko, Słowinski, Stefanowski, 1998].

Do oceny satysfakcji klienta zastosowano metodę indukcji zbioru reguł ze zbioru przypadków  $U$ , który zawierał 70 obserwacji. Model regułowy w postaci fragmentu tablicy decyzyjnej dla 5 wybranych przypadków przedstawiono w tablicy 4.11.

**Tablica 4.11. Fragment tablicy decyzyjnej oceny satysfakcji klienta**

Przypadki	Atrybuty warunkowe									Atrybut decyzyjny
	$C1$	$C2$	$C3$	$C4$	$C5$	$C6$	$C7$	$C8$	$C9$	$D1$
1	3	1	1	0	1	Dobra	1	1	Pochwała	Bardzo zadowolony
2	1	1	0	0	0	Średnia	1	1	Reklamacja	Mało zadowolony
3	1	0	1	0	1	Zła	0	1	Krytyka	Średnio zadowolony
4	1	1	1	1	1	Dobra	1	1	Pochwała	Bardzo zadowolony
5	2	0	1	0	1	Dobra	1	0	Krytyka	Zadowolony

Źródło: [Prędko, Słowinski, Stefanowski 1998, s. 167].

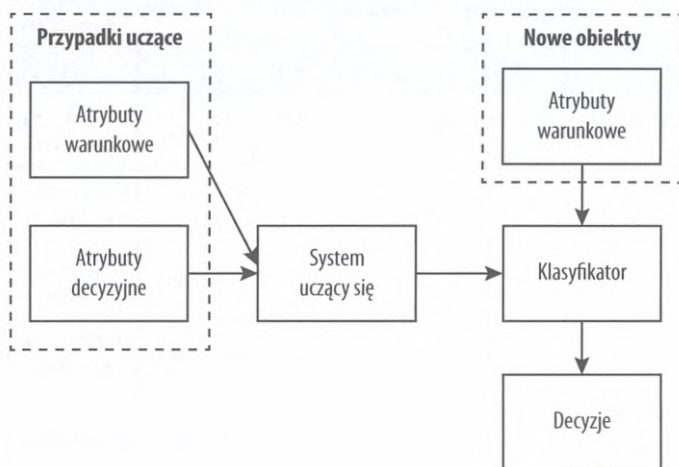
Poszczególne atrybuty warunkowe oznaczają [Prędko, Słowinski, Stefanowski, s. 167]:

- $C1$  — proponowany termin wykonania usługi (liczba dni),
- $C2$  — dotrzymanie terminu wykonania usługi (tak/nie),
- $C3$  — sprawdzenie stanu samochodu przy kliencie (tak/nie),
- $C4$  — pomoc doradcy w dotarciu do domu (tak/nie),
- $C5$  — usunięcie zgłoszonych niesprawności (tak/nie),
- $C6$  — ocena jakości obsługi klienta (tak/nie),
- $C7$  — dotrzymanie ceny wykonania usługi (tak/nie),
- $C8$  — wyjaśnienie wysokości rachunku (tak/nie),
- $C9$  — relacja klienta (tak/nie).

Natomiast atrybutem warunkowym D1 jest poziom zadowolenia klienta.

Sposób przeprowadzania klasyfikacji przypadków przez model regułowy przedstawiono na rysunku 4.14. Korzystając z programu ROSE 2, wygenerowano 25 reguł pewnych i 3 reguły przybliżone oraz zredukowano liczbę atrybutów warunkowych. Pozwoliło to na stwierdzenie, które atrybuty w najwyższym stopniu decydują o zadowoleniu klienta (klasa klientów bardzo zadowolonych), a które o niezadowoleniu klienta (klasa klientów mało zadowolonych). Okazało się, że krótki termin wykonania usługi, dotrzymanie terminu wykonania usługi, wyjaśnienie wysokości wystawionego rachunku oraz pochwała klienta dotycząca pracy serwisu warunkują wysoki poziom zadowolenia klienta. Natomiast brak usunięcia zgłoszonej niesprawności, niedotrzymanie terminu wykonania usługi oraz zła jakość obsługi klienta wpływają na jego niezadowolenie.

Rysunek 4.14. Klasyfikacja przypadków przez model decyzyjny



Źródło: [Prędko, Słowinski, Stefanowski 1998, s. 166].

Przykład zastosowania zbiorów przybliżonych w diagnostyce aparatury paliwowej silników o zapłonie samoczynnym opracowano na podstawie [Klimkiewicz 2005, s. 177–183].

W badaniach wykrywania usterek w silnikach spalinowych o zapłonie samoczynnym zastosowano model regułowy, w którym atrybutami warunkowymi są symptomy wskazujące na niewłaściwą pracę silnika oraz kontrolne pomiary wykonane przez mechanika. Atrybutem decyzyjnym jest usterka, którą należy usunąć. Prawie 1077 przypadków usterek i odpowiadających im symptomów oraz wartości pomiarowych było rejestrowanych w serwisie specjalizującym się w naprawach silników paliwowych. W tabelicy 4.12 przedstawiono listę atrybutów warunkowych modelu, a w tabelicy 4.13 zaprezentowano listę wartości atrybutu decyzyjnego USTERKA.



Tablica 4.12. Lista atrybutów warunkowych modelu diagnostyki aparatury paliwowej silników

Atrybut warunkowy	Opis atrybutu warunkowego
C1	Czy silnik uruchamia się po wyjęciu rdzenia z zaworka elektromagnetycznego?
C2	Czy brak pęcherzyków powietrza w przezroczystym przewodzie doprowadzającym paliwo do pompy?
C3	Czy ręczna pompka podaje paliwo?
C4	Czy brak zapachu benzyny w paliwie?
C5	Czy akumulator jest w odpowiednim stanie?
C6	Czy świece żarowe są sprawne?
C7	Czy kąt początku tłoczenia jest ustawiony prawidłowo?
C8	Czy podczas napędu silnika rozrusznikiem z króćca przelewowego wypływa paliwo?
C9	Jakie jest ciśnienie sprężania?
C10	Czy drożny jest układ dolotowy powietrza?
C11	Czy drożny jest układ wydechowy?
C12	Czy wtryskiwacze są w dobrym stanie?
C13	Czy kolejność połączenia przewodów wysokiego ciśnienia jest właściwa?
C14	Czy regulacja zderzaka śruby zapobiegającej zgaśnięciu silnika jest prawidłowa?
C15	Czy prawidłowo wyregulowana jest maksymalna prędkość obrotowa silnika?

Źródło: [Klimkiewicz 2005, s. 179].

Korzystając z programu ROSE 2 [Prędko, Słowinski, Stefanowski 1998], wygenerowano 38 reguł, w tym 36 reguł pewnych i 2 reguły przybliżone. Zbudowany model regułowy jest wykorzystany do diagnostyki usterek, ale również do szkolenia pracowników zajmujących się naprawą silników o zapłonie samoczynnym.

Tablica 4.13. Lista wartości atrybutu decyzyjnego USTERKA

Lista wartości atrybutu decyzyjnego usterka
Usterka silnika
Uszkodzona pompa wtryskowa
Niewłaściwa regulacja maksymalnej prędkości obrotowej
Niewłaściwa regulacja śruby zderzaka i wolnych obrotów
Niewłaściwa kolejność połączenia przewodów wysokiego ciśnienia
Zużyte rozpylacze
Niedrożność układu paliwowego
Niedrożność układu dolotowego powietrza
Niskie ciśnienie sprężania
Niewłaściwie ustawiony kąt wyprzedzenia wtrysku paliwa
Niesprawne świece żarowe
Niesprawny akumulator
Olej napędowy zawiera benzynę
Zapowietrzony układ doprowadzający paliwo do pompy
Uszkodzony zawór elektromagnetyczny
Zablokowany układ wydechowy

Źródło: Jak tablicy 4.12.

## 4.4. Zasada charakteryzacji w modelowaniu problemów decyzyjnych

Cechą charakterystyczną wspólnie rozwiązywanych problemów jest ich duża złożoność obliczeniowa. Dotyczy to zadań związanych z optymalizacją kombinatoryczną, dla których podczas rozwiązywania powstaje duża liczba wariantów rozwiązań dopuszczalnych. Matematyka dyskretna dostarcza efektywne pod względem szybkości obliczeń i wymaganej pamięci komputera algorytmy kombinatoryczne. Aplikacyjne zadania z tego zakresu można podzielić na zadania analizy i syntezy systemów dyskretnych.

### 4.4.1. Koncepcja zasady charakteryzacji

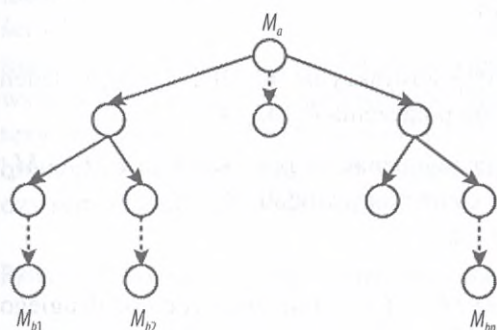
Za rozwiązanie zadania analizy uznaje się problem:

Czy model  $M_a$  systemu dyskretnego posiada żądane własności?

Rozwiązanie zadania syntezy polega na przekształceniu modelu  $M_a$  na model  $M_b$ , w którym jest osiągnięte ekstremum badanego funkcjonału jakości  $F(b)$ . W obu

przypadkach można mówić o równoważności modeli. W zadaniach analizy buduje się na podstawie modelu  $M_a$  jego równoważnik  $M_b$ , zachowujący własności modelu  $M_a$ . W zadaniach syntezy modelu  $M_a$  utworzony model  $M_b$  jest równoważny. Analizę i syntezę przeprowadza się za pomocą algorytmów kombinatorycznych. Bardzo proste w realizacji są algorytmy kombinatoryczne prowadzące pełny przegląd możliwych wariantów. Przykład drzewa rozwiązań dla równoważności syntaktycznej pokazano na rysunku 4.15.

**Rysunek 4.15. Przykład drzewa rozwiązań dla równoważności syntaktycznej**



Źródło: [Gorbatov 1978].

W tym przypadku nie korzysta się z wyników teoretycznych, na podstawie których można by zaproponować ulepszony algorytm rozwiązania. W algorytmach tej klasy realizuje się równoważność syntaktyczną. Równoważność ta, zwana przekształceniem równoważnym, odpowiada poziomowi wiedzy, przy którym znany jest pełny zbiór aksjomatów i polega na budowie kolejnego wariantu dla zadania syntezy i analizy. Zadanie syntezy oznacza zamianę modelu  $M_a$  na model  $M_b$  na podstawie tego lub innego zbioru aksjomatów i twierdzeń wyprowadzonych z tych aksjomatów. Przykład drzewa rozwiązań dla równoważności syntaktycznej pokazano na rysunku 4.15. Każdy wierzchołek wiszący odpowiada rozwiązaniu, którego nie można ulepszyć. Można sformułować następujące własności drzewa:

- przy liniowym wzroście wymiaru zadania przynajmniej wykładniczo rośnie liczba wierzchołków wiszących;
- istnieje konieczność powrotu z  $i$ -tego poziomu drzewa na  $(i - 1)$ -ty poziom drzewa.

W wyniku tej własności powstaje potrzeba przeglądu całego drzewa przy wyborze rozwiązania minimalnego. Szczególnie ważne jest opracowanie efektywnych algorytmów i programów komputerowych do rozwiązywania problemów automatyzacji projektowania systemów. Do zwiększenia szybkości algorytmów często wykorzystuje się heurystyki opracowane na podstawie doświadczeń i różnych analogii.

Algorytmy heurystyczne realizują równoważenie heurystyczne. Stosowanie algorytmów heurystycznych zwiększa szybkość algorytmów, ale najczęściej ocena jakości otrzymanego rozwiązania jest niemożliwa. Nierealne jest także stwierdzenie, czy rozwiązania otrzymane przy wykorzystaniu algorytmu heurystycznego jest niepoprawialne. Na osiągnięcie maksymalnej szybkości algorytmów bez generowania wszystkich niepoprawialnych rozwiązań pozwalają algorytmy trzeciej klasy, realizujące równoważenie semantyczne. Przy tym równoważeniu znany jest nie tylko pełny zbiór aksjomatów, lecz także konstruktywne kryterium pozwalające rozstrzygać prawdziwość predykatu  $P_0(M_a)$ :

$$P_0(M_a) = \begin{cases} 1, & \text{jeśli model } M_a \text{ posiada zadaną własność,} \\ 0, & \text{w przeciwnym przypadku.} \end{cases}$$

W zadaniu syntezy wiąże się dwa różne abstrakcyjne modele  $M_a$  i  $M_b$  w jeden model za pomocą predykatu funkcjonalnego połączenia  $P_0(M_a, M_b)$ :

$$P_0(M_a, M_b) = \begin{cases} 1, & \text{jeśli istnieje wzajemnie jednoznaczne przekształcenie } M_a \rightarrow M_b, \\ & \text{odpowiedniości między elementami modeli } M_a \text{ i } M_b, \\ 0, & \text{w przeciwnym przypadku.} \end{cases}$$

Ogólnie, to, co charakteryzuje modele  $M_a$  i  $M_b$  i co odróżnia jeden od drugiego to sens przekształcenia  $M_a \rightarrow M_b$ . Inaczej mówiąc, przekształcenie  $M_a \rightarrow M_b$ , opiera się na własnościach językowych wyrażań, niezmienniczych względem przedstawienia modeli. Badaniem sensu przekształceń zajmuje się semantyka logiki, która jest działem metalogiki. W tych rozważaniach przez semantykę rozumie się interpretację jednego formalnego języka w kategoriach drugiego. Oba języki formalizują te same systemy pojęć. Taką semantykę będziemy nazywali rzutową. W szczególnym przypadku, gdy oba języki się pokrywają, bada się istnienie określonych własności modelu. W tej sytuacji semantykę rzutową nazywa się refleksywną. Semantyka ta pozwala rozwiązywać zadania analizy modelu. Semantyka rzutowa w odniesieniu do przekształcenia  $M_a \rightarrow M_b$  zapewnia możliwość wyznaczenia ekstremalnej wartości funkcjonału  $\varphi(M_a)$ , budując odpowiadający model  $M_b$  bez tworzenia wszystkich równoważnych modeli  $M_b$ . Pozwala to na znaczne zmniejszenie złożoności algorytmów optymalizacji. Do wyznaczenia rzutowej semantyki przekształcenia  $M_a \rightarrow M_b$  należy:

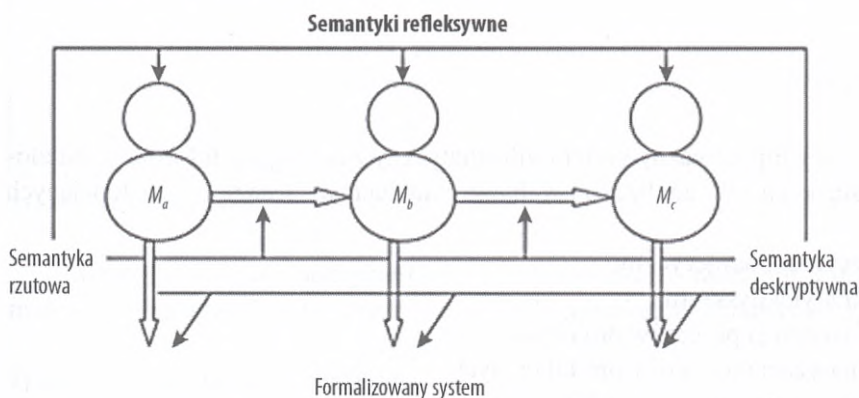
1. Znaleźć liczbowe charakterystyki  $v_i$  i  $\in N$  modelu  $M_b$ , jednoznacznie określające wartości  $\varphi(M_b)$ .
2. Ustalić własności  $S_a$  modelu  $M_a$ , za pomocą których można wyznaczyć jednoznacznie określające  $v_i$  i  $\in N$ .
3. Wyznaczyć własności  $S_a$  modelu  $M_a$ , jednoznacznie określające własności  $S_b$  modelu  $M_b$ .
4. Znaleźć charakterystyki liczbowe modelu  $M_a$ , spełniające własności  $S_a$  i jednoznacznie określające  $\varphi(M_a)$ .

Z przedstawionych rozważań można wnioskować, że własności  $S_a$  pozwalają jednoznacznie obliczać  $\varphi(M_b)$  bez budowania modelu  $M_b$ . Do wyznaczenia refleksywnej semantyki analizy modelu  $M_a$  trzeba:

- wyznaczyć własności  $S_a$  modelu  $M_a$  jednoznacznie określające predykat  $P_0(M_a)$ ;
- znaleźć charakterystyki liczbowe modelu  $M_a$  określające obecność własności  $S_a$ .

W obu przypadkach podstawowym zadaniem jest ustalenie własności  $S_a$  modelu  $M_a$ , które określają prawdziwość predykatów  $P_0(M_a)$  i  $P_0(M_a, M_b)$ . Dalej bada się własności modelu  $M_a$  pod kątem istnienia figur zabronionych. Własność istnienia tych figur będzie podstawą tworzenia kryterium zachodzenia własności  $S_b$ . Znajomość figur zabronionych umożliwi budowanie kryterium sprawdzenia prawdziwości własności  $S_b$ . Wyznaczone figury pozwalają na efektywne rozwiązywanie zadania analizy i konstruktywne obliczanie wartości funkcjonału  $\varphi(M_b)$ , bez generowania wszystkich modeli równoważnych modelom  $M_b$ , przy rozwiązywaniu zadania syntezy. Wybór semantyki refleksywnej i rzutowej na podstawie znajomości figur zabronionych będziemy nazywali semantyką konstruktywną. Wzajemne relacje między semantykami ilustruje rysunek 4.16.

Rysunek 4.16. Wzajemne relacje między semantykami



Źródło: Jak rysunku 4.15.

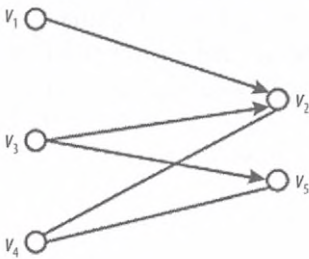
#### 4.4.2. Problem decyzyjny w ujęciu zasady charakteryzacji

W tym punkcie rozważymy przykład charakteryzacji **grafu dwudzielnego**. Zadanie określenia dwudzielności grafu lub przekształcenia grafu w dwudzielny ma wiele zastosowań: od projektowania niezawodnych urządzeń do budowania efektywnych systemów informatycznych z pracą równoległą. Zakłada się, że kilka baz danych jest przechowywanych na dwóch komputerach. W celu osiągnięcia maksymalnej wydajności systemu wyszukiwania plików z baz danych należy dekomponować tak bazy danych na dwie części, aby otrzymać minimalny czas dostępu do obu komputerów.

Plikom bazy danych przyporządkowuje się odpowiednie wierzchołki grafu. Dwa wierzchołki są incydentne, jeśli oba pliki są niezbędne do znalezienia odpowiedzi na pytanie pewnego rodzaju. Optymalny rozkład plików na komputerach określa się przez podział zbioru wierzchołków grafu na dwa podzbiory, wewnątrz których jest możliwa jak najmniejsza liczba łuków (odpowiada to uzyskaniu maksymalnej równoległości).

Zadanie to sprowadza się do wskazania rzutowej semantyki przekształcania grafów w grafy dwudzielne przy minimalnym odrzuceniu łuków grafu. Semantykę określa rozwiązanie problemu zadania charakteryzacji dwudzielności grafów. Przykładowy graf dwudzielny pokazano na rysunku 4.17.

Rysunek 4.17. Graf dwudzielny



Źródło: Jak rysunku 4.15.

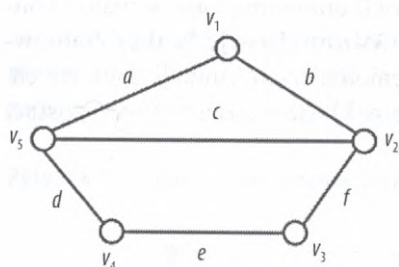
Rozważmy hipotetyczny system informatyczny zawierający informacje katalogowe o książkach. W analizowanej bazie znajdują się książki z następujących dziedzin:

- $v_1$  — eksperyment numeryczny,
- $v_2$  — matematyka dyskretna,
- $v_3$  — automatyzacja procesów projektowania,
- $v_4$  — automatyzacja procesów produkcyjnych,
- $v_5$  — metody matematyczne, modele.

Wymienione dziedziny nie są rozłączne. Oznacza to, że w katalogu są książki należące do więcej niż jednej dziedziny. Powiązania między dziedzinami pokazuje graf na rysunku 4.18. Dwie dziedziny  $v_k$  i  $v_l$  ( $k \neq l$ ) są połączone łukiem, jeśli  $v_k \cap v_l \neq \emptyset$ .

Rozważmy przekształcenia grafu z rysunku 4.18 w graf dwudzielny po usunięciu odpowiednich łuków. Za funkcjonalność jakości przyjmujemy liczbę usuniętych łuków. W tym zadaniu figurami zakazanymi są cykle o nieparzystej liczbie łuków. Można wyróżnić dwa zbiory zakazane:  $A_{31} = \{a, b, c\}$  i  $A_{32} = \{a, b, d, e, f\}$ . W tablicy 4.14 przedstawiono tablicę semantyczną dla tego zadania.

**Rysunek 4.18.** Graf zawierający powiązania między działami książek w katalogu bibliotecznym



Źródło: Opracowanie własne.

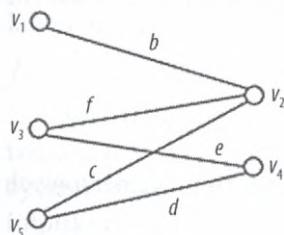
**Tablica 4.14.** Tablica dla grafu z rysunku 4.18

$A_{31}$	$A_{32}$	
1	1	$a$
1	1	$b$
1		$c$
	1	$d$
	1	$e$
	1	$f$

Źródło: [Gorbatov 1978].

Minimalne jest np. pokrycie  $\Pi = \{a\}$ . Odpowiada to usunięciu łuku  $a$ . Otrzymuje się wtedy graf dwudzielny (rysunek 4.19), a funkcjonal jakości  $\varphi(A_j) = 1$ .

**Rysunek 4.19.** Graf dwudzielny otrzymany w wyniku usunięcia łuku  $a$



Źródło: [Gorbatov 1978].

### 4.4.3. Przykład algorytmu konstruowania zbioru decyzji alternatywnych

W tym punkcie na przykładzie funkcji boolowskich omówimy zastosowanie koncepcji budowania zbioru decyzji alternatywnych. **Minimalizację funkcji boolowskich** można prowadzić w różnych bazach elementarnych funkcji logicznych. Zwykle minimalizację tych funkcji rozpatruje się w klasie alternatywnych postaci normalnych.

#### Definicja 4.15

**Funkcją boolowską** nazywamy funkcję w postaci:

$$f: \{0, 1\}^n \rightarrow \{0, 1\},$$

gdzie:  $\{0, 1\}^n$  jest zbiorem  $n$ -elementowych ciągów złożonych z zer i jedynek. Najwygodniej zadaje się funkcje boolowskie za pomocą tablic (tablica 4.15).

#### Przykład 4.14

Tablica 4.15. Przykład funkcji boolowskiej zależnej od 3 zmiennych

$x_1 x_2 x_3$	$f(x_1, x_2, x_3)$	Równoważnik dziesiętny
0 0 0	0	0
0 0 1	0	1
0 1 0	0	2
1 0 0	0	4
0 1 1	1	3
1 0 1	1	5
1 1 0	1	6
1 1 1	1	7

Źródło: Jak tablicy 4.14.

Funkcje boolowskie realizuje się za pomocą formuł logicznych. W pierwszym etapie buduje się pełną alternatywną postać normalną:

$$f(x_1, x_2, \dots, x_n) = \bigvee_{\bar{x} \in \{0,1\}^n} x_1^{\sigma_1} x_2^{\sigma_2} \dots x_n^{\sigma_n} f(\sigma_1, \sigma_2, \dots, \sigma_n),$$

gdzie:

sumowanie logiczne jest prowadzone po wszystkich  $2^n$  ciągach  $n$ -elementowych zbioru  $\{0,1\}^n$ ,

$$x^\sigma = \begin{cases} x, & \text{gdy } \sigma = 1, \\ \bar{x}, & \text{gdy } \sigma = 0. \end{cases}$$

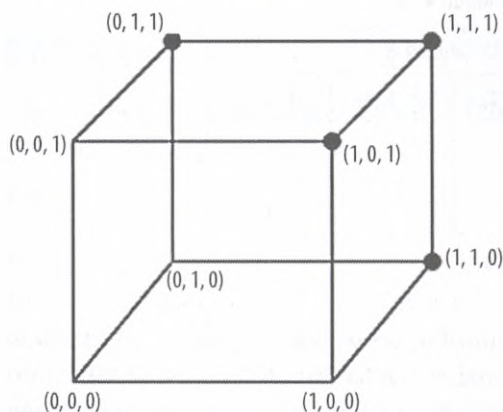


Dla funkcji  $f(x_1, x_2, x_3)$  z przykładu 4.14 pełna alternatywna postać normalna przyjmuje następującą postać:

$$f(x_1, x_2, x_3) = \bar{x}_1 x_2 x_3 \vee x_1 \bar{x}_2 x_3 \vee x_1 x_2 \bar{x}_3 \vee x_1 x_2 x_3.$$

Funkcja boolowska rozważana w przykładzie 4.14 ma prostą interpretację geometryczną. Zbiór  $\{0, 1\}^n = \{0, 1\} \times \{0, 1\} \times \dots \times \{0, 1\}$  można geometrycznie interpretować jako  $n$ -wymiarową kostkę jednostkową.

Rysunek 4.20. Interpretacja geometryczna funkcji z przykładu 4.14



Źródło: Jak rysunku 4.19.

Aby zadać funkcję  $f(x_1, x_2, x_3)$ , wystarczy każdemu z 8 wierzchołków sześcianu  $\{0, 1\}^3$  przyporządkować 0 lub 1. Realizacja geometryczna funkcji z przykładu 4.14 jest podana na rysunku 4.20. Każdy wierzchołek sześcianu  $\{0, 1\}^n$  można interpretować, a binarne rozwinięcie liczby jest podane w systemie dziesiętnym. W związku z tym funkcję boolowską można przedstawić jako ciąg liczb w systemie dziesiętnym. W rozważanym przykładzie można zapisać symbolicznie jako:

$$f(x_1, x_2, x_3) = \sum (3, 5, 6, 7).$$

Analogicznie do pełnej alternatywnej postaci normalnej można rozważać pełne koniunkcyjne postacie normalne. Dla funkcji zadanej tablicą 4.16 można zapisać:

$$f(x_1, x_2, x_3) = (x_1 \vee x_2 \vee x_3)(x_1 \vee x_2 \vee \bar{x}_3)(x_1 \vee \bar{x}_2 \vee x_3)(\bar{x}_1 \vee x_2 \vee x_3).$$

Jeśli zbiór  $E_f = \{(x_1, x_2, \dots, x_n) : f(x_1, x_2, \dots, x_n) = 1\}$  zawiera **kostkę**  $I_k$  wymiaru  $k$ , tzn.  $I_k \subset E_f$  i żadna **podkostka**  $I_{k+1}$  nie zawiera się w zbiorze  $E_f$ , to podkostkę  $I_k$  nazywamy **maksymalną**. Koniunkcję odpowiadającą podkostce  $I_k$  określamy mianem **implikantu prostego** funkcji  $f(x_1, x_2, \dots, x_n)$ . Na podstawie rysunku 4.20 wnioskujemy, że zbiór  $E_f$  zawiera następujące podkostki maksymalne:

$$K_1 = \{(1, 1, 0), (1, 1, 1)\},$$

$$K_2 = \{(1, 0, 1), (1, 1, 1)\},$$

$$K_3 = \{(0, 1, 1), (1, 1, 1)\}.$$

Kostkę  $K_1$  oznacza się skrótowo (11–),  $K_2$  jako (1–1) i  $K_3$  jako (–11). Kostkom maksymalnym odpowiadają kolejno następujące implikanty proste  $I_1 = x_1x_2$ ,  $I_2 = x_1x_3$  i  $I_3 = x_2x_3$ . W celu wyznaczenia minimalnej postaci alternatywnej buduje się tablicę implikantów prostych (tablica 4.16).

**Tablica 4.16. Tablica implikantów prostych dla przykładu 4.14**

Implikanty proste	Punkty zbioru $E_f$			
	(0, 1, 1)	(1, 0, 1)	(1, 1, 0)	(1, 1, 1)
$I_1$			$x$	$x$
$I_2$		$x$		$x$
$I_3$	$x$			$x$

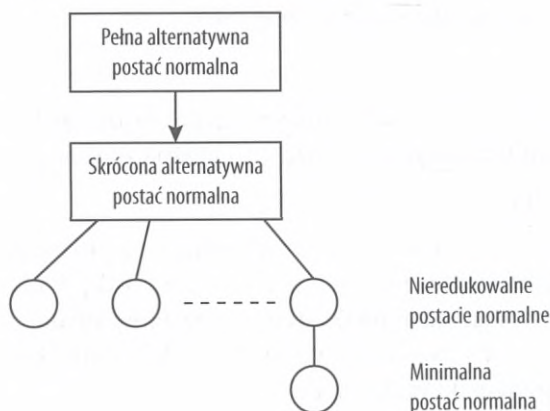
Źródło: Jak tablicy 4.14.

Analiza tablicy 4.16 prowadzi do wniosku, że w celu wybrania implikantów prostych, tak aby w każdej kolumnie był znak  $x$ , trzeba uwzględnić wszystkie implikanty. W kolumnach 1, 2 i 3 jest dokładnie jeden znak  $x$ . Oznacza to, że do pokrycia kolumn tablicy muszą wchodzić implikanty  $I_1$ ,  $I_2$ ,  $I_3$ . Implikanty te tworzą jądro funkcji  $f(x_1, x_2, x_3)$ . Istnieje jedna alternatywna postać minimalna:

$$f(x_1, x_2, x_3) = x_1x_2 \vee x_1x_3 \vee x_2x_3.$$

Ogólny schemat minimalizacji funkcji  $f(x_1, x_2, x_3)$  pokazano na rysunku 4.21.

**Rysunek 4.21. Ogólny schemat minimalizacji funkcji**



Źródło: Jak rysunku 4.19.

Dla danej skróconej postaci normalnej zawierającej wszystkie implikanty proste można zbudować nieredukowalne postaci normalne złożone z niektórych implikantów prostych. Nieredukowalna postać nie może być uproszczona przez usunięcie implikantu prostego. Zbiór nieredukowalnych postaci alternatywnych zawiera minimalne postaci alternatywne. Praktyka minimalizacji funkcji pokazuje, że zasadniczą trudnością w realizacji algorytmu minimalizacji jest duża liczba nieredukowalnych postaci alternatywnych.

Najbardziej pracochłonny okazuje się etap wyboru postaci minimalnej ze zbioru postaci nieredukowalnych.

#### Przykład 4.15

Zadanie polega na zbadaniu złożoności minimalnego strukturalnego grafu realizującego funkcję boolowską zadaną za pomocą liczb dziesiętnych.

$$f(x_1, x_2, x_3, x_4) = \sum (0, 1, 2, 4, 9, 11, 13).$$

W pierwszym etapie wyznacza się implikanty proste metodą Quine'a-McCluskeya. Liczby dziesiętne podane w definicji funkcji  $f(x_1, x_2, x_3, x_4)$  zapisuje się w postaci binarnej, oddzielając rozwinięcia o różnej liczbie jedynek. Wyniki sklejanie ciągów umieszczono w drugiej kolumnie. Skleja się ciągi z sąsiednich klas różniące się dokładnie na jednej pozycji. W drugim etapie nie można wykonać żadnych sklejeń. Proces wyznaczania implikantów kończy się.

Tablica 4.17. Wyznaczanie implikantów prostych

<u>0 0 0 0</u>	0 0 0
0 0 0 1	0 0 - 0
0 0 1 0	<u>0 - 0 0</u>
<u>0 1 0 0</u>	- <u>0 0 1</u>
<u>1 0 0 1</u>	1 0 - 1
1 0 1 1	1 - 0 1
1 1 0 1	

Źródło: Jak tablicy 4.14.

Skrócona alternatywna postać normalna jest następująca:

$$f(x_1, x_2, x_3, x_4) = \bar{x}_1 \bar{x}_2 x_3 \vee \bar{x}_1 \bar{x}_2 x_4 \vee \bar{x}_1 \bar{x}_3 \bar{x}_4 \vee \bar{x}_2 \bar{x}_3 \bar{x}_4 \vee x_1 \bar{x}_2 x_4 \vee x_1 \bar{x}_3 x_4.$$

W następnym etapie tworzy się tablicę Quine'a.

Tablica 4.18. Tablica Quine'a dla przykładu 4.15

Implikanty proste		Punkty kostki $E_f$						
		0000	0001	0010	0100	1001	1011	1101
1	000-	x	x					
2	00-0	x		⊗				
3	0-00	x			⊗			
4	-001		x			x		
5	10-1					x	⊗	
6	1-01					x		⊗

Źródło: Jak tablicy 4.14.

Symbol  $\otimes$  oznacza, że odpowiadający wiersz wchodzi do jądra pokrycia, które tworzą następujące implikanty proste:

$$I = \{\bar{x}_1\bar{x}_2\bar{x}_4, \bar{x}_1\bar{x}_3\bar{x}_4, x_1\bar{x}_2x_4, x_1\bar{x}_3x_4\}.$$

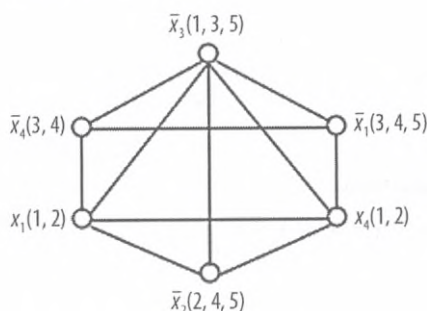
Koniunkcje należące do jądra wchodzi do każdej postaci normalnej. Usuwając z tablicy 4.18 implikanty proste należące do jądra  $J$ , można zauważyć, że analizowana w przykładzie 4.15 funkcja boolowska posiada dwie postaci nieredukowalne. Są to następujące funkcje:

$$f'(x_1, x_2, x_3, x_4) = x_1\bar{x}_3x_4 \vee x_1\bar{x}_2x_4 \vee \bar{x}_1\bar{x}_2\bar{x}_3 \vee \bar{x}_1\bar{x}_2\bar{x}_4 \vee \bar{x}_1\bar{x}_2\bar{x}_3,$$

$$f''(x_1, x_2, x_3, x_4) = x_1\bar{x}_3x_4 \vee x_1\bar{x}_2x_4 \vee \bar{x}_1\bar{x}_2\bar{x}_3 \vee \bar{x}_1\bar{x}_2\bar{x}_4 \vee \bar{x}_2\bar{x}_3x_4.$$

Oczywiście w tym przypadku każda z nieredukowalnych postaci alternatywnych jest minimalną alternatywną postacią normalną. Pierwszej nieredukowalnej postaci normalnej odpowiada graf przedstawiony na rysunku 4.22. W tym celu koniunkcje występujące w funkcji  $f'(x_1, x_2, x_3, x_4)$  numeruje się liczbami naturalnymi od 1 do 5.

Rysunek 4.22. Graf dla pierwszej nieredukowalnej postaci alternatywnej



Wierzchołek  $\bar{x}_1(3, 5)$  oznacza, że wchodzi do koniunkcji o numerach 1, 3, 5. Między wierzchołkami  $\bar{x}_1(3, 5)$  i  $x_1(3, 5)$  został poprowadzony łuk, ponieważ w obu wierzchołkach jest wspólna liczba równa 5. W grafie na rysunku 4.22 można wyróżnić trzy figury zakazane; są to cykle o długości 3, z cyklicznym przedstawieniem wag w postaci:

$$Q_{A1} = \{\bar{x}_2(4, 5), \bar{x}_3(5, 3), \bar{x}_4(3, 4)\},$$

$$Q_{A2} = \{x_1(1, 2), \bar{x}_2(2, 5), \bar{x}_3(5, 1)\},$$

$$Q_{A3} = \{x_4(1, 2), \bar{x}_2(2, 5), \bar{x}_3(5, 1)\}.$$

Tablica semantyczna odpowiadająca pierwszej nieredukowalnej jest pokazana w tablicy 4.19.

**Tablica 4.19. Tablica semantyczna dla zbiorów  $Q_{A1}$ ,  $Q_{A2}$  i  $Q_{A3}$**

Numer wężła	Węzeł	$Q_{A1}$	$Q_{A2}$	$Q_{A3}$
1	$x_1(1, 2)$	0	1	0
2	$\bar{x}_2(4, 5)$	1	0	0
3	$\bar{x}_2(2, 5)$	0	1	1
4	$\bar{x}_3(3, 5)$	1	0	0
5	$\bar{x}_3(1, 5)$	0	1	1
6	$\bar{x}_4(3, 4)$	1	0	0
7	$x_4(1, 2)$	0	0	1

Źródło: Jak tablicy 4.14.

Łatwo zauważyć, że pierwszy wiersz jest pochłaniany przez trzeci wiersz, szósty przez czwarty, siódmy przez piąty. W wyniku tej redukcji (ściśnięcia) otrzymujemy tablicę 4.20. Druga kolumna jest pochłaniana przez trzecią. Ostateczne wyniki tej redukcji umieszczono w tablicy 4.21.

**Tablica 4.20. Zredukowana tablica semantyczna**

Numer wężła	Węzeł	$Q_{A1}$	$Q_{A2}$	$Q_{A3}$
1	$\bar{x}_2(4, 5)$	1	0	0
2	$\bar{x}_2(2, 5)$	0	1	
3	$\bar{x}_3(3, 5)$	1	0	0
4	$\bar{x}_3(1, 5)$	0	1	1

Źródło: Jak tablicy 4.14.

Tablica 4.21. Tablica utworzona z połączenia tablic 4.19 i 4.20

$Q$	$\bar{x}_2(4,5)$	$\bar{x}_2(2,5)$	$\bar{x}_3(3,5)$	$\bar{x}_3(1,5)$
$Q_{A1}$	1	0	1	0
$Q_{A3}$	0	1	0	1
1	0	0	0	1
2	0	1	0	0
3	0	0	1	0
4	1	0	0	0
5	1	1	1	1

Źródło: Jak tablicy 4.14.

Mamy cztery pokrycia pierwszej podtablicy. Rozważmy pokrycie pierwszej podtablicy  $\{\bar{x}_2(4,5), \bar{x}_2(2,5)\}$ . Odpowiadającą temu pokryciu podtablicę podtablicy z tablicy 4.21 umieszczamy w tablicy 4.22.

Tablica 4.22. Podtablica tablicy 4.21

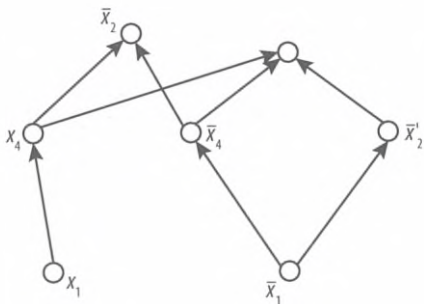
Identyfikatory słowa	$\bar{x}_2(4,5)$	$\bar{x}_2(2,5)$
2	0	1
4	1	0
5	1	1

Źródło: Jak tablicy 4.14.

Minimalnemu pokryciu tablicy 4.22 odpowiada wiersz z numerem 5. Wynika stąd, że pod wyrażeniem litery  $\bar{x}_2$  w piątym słowie otrzymujemy alternatywną postać normalną:

$$\tilde{f}'(x_1, x_2, x_3, x_4, x_2') = x_1 \bar{x}_3 x_4 \vee x_1 \bar{x}_2 x_4 \vee \bar{x}_1 \bar{x}_3 \bar{x}_4 \vee \bar{x}_1 \bar{x}_2 \bar{x}_4 \vee \bar{x}_1 \bar{x}_2' \bar{x}_3.$$

Można zatem pokazać, że graf rysunku 4.23 ma postać minimalną.

Rysunek 4.23. Graf realizujący funkcję  $f(x_1, x_2, x_3, x_4)$ 

Źródło: Jak rysunku 4.19.

Analogicznie można rozpatrywać pozostałe pokrycia i analizować figury zakazane dla drugiej postaci nieredukowalnej. Otrzymuje się w wyniku podobną postać normalną, co pozwala stwierdzić, że graf z rysunku jest grafem minimalnym realizującym funkcję  $f(x_1, x_2, x_3)$ .

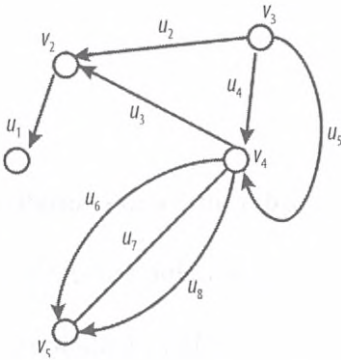
## 4.5. Pytania i zadania kontrolne

### 4.5.1. Pytania kontrolne

1. Podaj na przykładach praktycznych różnice między drzewami, dendrytami i grafami.
2. Przekształć ustalone decyzyjne drzewo logiczne na dowolne zmodyfikowane drzewo logiczne.
3. Czy mogą istnieć pustostany informacyjne w tablicy kombinacji dla wariantów teoretycznych danego problemu organizatorskiego?
4. Czy mogą istnieć wielokrotne optymalne dekompozycje drzew decyzyjnych?
5. Dlaczego wygodnie jest rysować graf zależności z kolejnymi wierzchołkami na okręgu?
6. Jak wytłumaczyć wzajemną konkurencyjność grup rozkładu grafu zależności?
7. Co to jest informacja granularna?
8. Czym się różni zbiór przybliżony od dokładnego?
9. Jaki związek występuje między reduktem i rdzeniem zbioru przybliżonego?
10. Przedstaw algorytm redukcji tablicy decyzyjnej dla zbiorów przybliżonych.
11. O czym informuje ocena jakości klasyfikacji zbioru przybliżonego?
12. Dlaczego fakt, że za pomocą zbiorów rozmytych można w sposób formalny wyrazić pojęcie nieprecyzyjne, przesądził o ich przydatności w procesie wspomagania decyzji?
13. Wymień przypadki określające stopień przynależności elementów do zbioru rozmytego.

### 4.5.2. Przykładowe zadania

1. Dla grafu przedstawionego na poniższym rysunku znajdź dowolny podgraf bez cykli.



2. Narysuj drzewo binarne o 13 łukach i 3 poziomach (podpowiedź — w drzewie binarnym z każdego wierzchołka wychodzą dokładnie dwie gałęzie).
3. Dla funkcji boolowskiej  $f(x_1, x_2, x_3) = \sum (0, 1, 7)$  utwórz pełną alternatywną postać normalną, wyznacz implikanty proste i utwórz tablicę Quine'a.
4. Utwórz tabelę kombinacji dla dowolnie ustalonego zbioru zmiennych decyzyjnych o różnej wielowartościowości.
5. Znajdź podrozwiązania prawdziwe ze zbioru wszystkich wariantów teoretycznych danego problemu organizatorskiego.
6. Podaj przykłady drzew prawdopodobieństwa dla danego przykładu praktycznego z uwzględnieniem zasady, że informacja jest tym bardziej szczegółowa, im mniejsze jest prawdopodobieństwo jej przybycia.
7. Podaj przykłady praktyczne płaskich i hierarchicznych problemów decyzyjnych.
8. Zaproponuj strukturę tablicy decyzyjnej dla problemu diagnozowania awarii układu hamulcowego na podstawie takich parametrów, jak: jałowy skok pedału hamulca, jałowy skok dźwigni hamulca awaryjnego, luz w mechanizmach hamulcowych, skok rezerwowego pedału hamulca, opóźnienie hamowania, całkowita siła hamowania.
9. Na przykładzie tablicy decyzyjnej (tablica 4.4) zaprezentuj górną i dolną granicę przybliżenia zbioru niedokładnego.
10. Na przykładzie tablicy decyzyjnej (tablica 4.4) zaprezentuj pozytywny i negatywny obszar zbioru przybliżonego.
11. Na przykładzie tablicy decyzyjnej (tablica 4.4) zaprezentuj proces redukcji zbioru przybliżonego.



12. Przy założeniu, że  $X = R$  i  $R$  jest zbiorem liczb rzeczywistych, określ, jak za pomocą funkcji przynależności można zdefiniować zbiór liczb rzeczywistych „bliskich liczby 100”.
13. Skonkretyzuj przykład w określonym scenariuszu decyzyjnym w przedsiębiorstwie przy założeniu, że mamy dwie zmienne lingwistyczne  $L$  i  $K$  takie, że jeżeli wartość zmiennej  $L$  jest zbiorem rozmytym  $A$  określonym w  $X$  oraz wartość zmiennej  $K$  jest zbiorem rozmytym  $B$  określonym w  $Y$ , to zależność między  $L$  i  $K$ , a właściwie między wartościami  $A$  i  $B$ , można zapisać jako:

JEŻELI  $L = A$  TO  $K = B$ .

14. Dla funkcji boolowskiej  $f(x_1, x_2, x_3) = \sum (0, 1, 7)$  utwórz pełną alternatywną postać normalną.
15. Dla funkcji z poprzedniego zadania wyznacz implikanty proste i utwórz tablicę Quine'a.

# Konkretyzacja rozwiązań projektowych SWD

SWD jest opisywany jako zbiór metod, narzędzi i modeli integrujących i przetwarzających zdarzenia zapisane w bazach i hurtowniach danych. W rozdziale przedstawiono strategię projektowania i rozwoju SWD. Ilustracją rozwiązań projektowych SWD są funkcjonalne i eksploatacyjne aspekty projektowania HD, BW oraz modułu wnioskującego. Komponenty architektury SWD są odpowiedzialne za procesy gromadzenia i przetwarzania danych (bazy i hurtownie danych) oraz procesy gromadzenia i przetwarzania reguł wiedzy (baza wiedzy, moduł wnioskujący).

Warunkiem skutecznego formułowania problemów i zapytań w SWD jest integracja odpowiednio zaprojektowanego interfejsu użytkownika ze wspomnianymi komponentami SWD.

## 5.1. Strategie projektowania SWD

W wielu opracowaniach podkreśla się, że w procesie projektowania i implementowania systemów SWD nie należy koncentrować uwagi na efektywności przetwarzania, ale raczej na dostarczaniu właściwych informacji podejmującym decyzje. Zdaniem C.M. Olszak [2000], ze względu na rozległą teorię podejmowania decyzji, dynamicznie zmieniający się charakter potrzeb decydentów, tempo budowy systemów i łatwość ich obsługi tradycyjne podejście do analizy i projektowania systemów informacyjnych okazuje się niewłaściwe do opracowania nowych strategii budowy SWD. Dlatego Autorka proponuje trzy zasadnicze kierunki w tym obszarze: metodologię proceduralną, podejście zorientowane na reprezentację oraz podejście zorientowane na decyzje i organizację.

Ideą metodologii proceduralnej jest iteracyjne projektowanie czy oparcie się na strukturalizacji procesów, analizie danych, projektowaniu i implementacji oraz przyjęciu założenia, że na początku każdego projektu dysponujemy pewnym stabilnym

zestawem potrzeb, struktura problemu jest znana, a preferencje kierownicze, procedury podejmowania decyzji i otoczenie decydenta są niezmiennie w trakcie życia systemu [Olszak 2000, s. 45]. Szerzej temat ten został rozwinięty w opracowaniach m.in. Z. Szyjewskiego [1993] i S. Wryczy [1999]. Z uwagi na charakter iteracyjnego projektowania w układzie projektant–użytkownik w literaturze definiuje się opisywaną metodologię jako ewolucyjne podejście czy prototypowanie. Projektowanie jednak w takim podejściu zawsze rozpoczyna się od poziomu problemu, a następnie uruchamiany jest proces uogólniania lub uszczegóławiania [Sprague 1994]. Podejście takie stosowane jest przez większość kadry menedżerskiej, ale wymusza ono nierzadko wprowadzanie radykalnych zmian organizacyjnych jako efekt konieczności dostosowania komórek organizacyjnych przedsiębiorstwa do wymagań determinujących osiągnięcie wysokiej sprawności i skuteczności przepływów informacyjnych. Wadą analizowanego podejścia jest trudność określenia ostatecznego terminu wygenerowania systemu oraz jego kosztów, a także konieczność stosowania wysokiej klasy narzędzi, np. obiektowej analizy projektowania i programowania, języków czwartej generacji. Zdaniem wielu autorów podejście to może wspomagać z sukcesem prace projektowe lub kończyć się fiaskiem w przypadku wykorzystywania tych narzędzi jako generatorów kodów aplikacji. Wymagają one także dużej dyscypliny projektowej oraz gruntownej znajomości metod i technik analizy oraz projektowania [Szyjewski 1993, Billewicz, Ziemia 1999]. Wśród wybranych narzędzi do tworzenia SWD można wymienić generatory SWD, np. finansowe języki modelowania, arkusze kalkulacyjne, narzędzia SWD (np. generatory raportów, bazy danych, pakiety analiz statystycznych, narzędzia RAD, narzędzia do zarządzania informacją, narzędzia do grafiki biznesowej, narzędzia do zarządzania informacją czy narzędzia do komunikacji itp.).

Adaptacyjny mechanizm procesu projektowania SWD opisywany przez Simona jako rozwiązanie, które dopasowuje się do różnych rodzajów zmian w trzech głównych horyzontach czasowych, charakteryzuje H. Sroka [1986].

W podejściu ukierunkowanym na reprezentację eksponuje się struktury danych, metody reprezentacji wiedzy oraz modele i systemy komunikacji [Power 1998]. Priorytetowym zagadnieniem przy budowie SWD jest model, przy czym albo można decydentowi dostarczyć gotowe modele, albo narzędzia umożliwiające tworzenie własnych modeli. Zawsze jednak najistotniejsze jest wierne odwzorowanie rzeczywistości, co często się nie udaje ze względu na złożoność i wieloaspektowość sytuacji rzeczywistych, o czym szerzej pisze J. Gołuchowski [1997]. Należy zatem w projektowaniu przewidzieć możliwość poszerzania zakresu modeli, ich rozwinięć, integracji z innymi, a także objęcie nimi różnych poziomów zarządzania obiektami — od strategicznego (np. wyznaczającego długofalową politykę rozwoju firmy), poprzez poziom taktyczny (np. dotyczący intensyfikacji sprzedaży), aż do operacyjnego (np. z zakresu kontroli zapasów i jakości produkcji). Wykorzystanie SZBD w SWD pozwala użytkownikom na łatwe i elastyczne manipulowanie danymi heterogenicznymi, integrację danych z modelami, stawianie zapytań do bazy danych czy

generowanie zestawień wynikowych w różnych przekrojach i formatach. Transakcyjne, specjalistyczne bazy danych czy pochodzące z otoczenia instytucji tworzą takie możliwości wraz z technologiami analizy danych opartymi na: hurtowniach danych, OLAP, eksploracji danych, technologii agentów czy też technologii obiektowej i multimedialnej. Kluczową, a często niedocenianą, rolę w ocenie SWD przez użytkownika odgrywa dialog między nim a systemem, o czym szerzej pisze H. Sroka [1994]. Ten aspekt SWD przybliżono w rozdziale 3. Należy podkreślić, że dotychczas niedoceniano takich aspektów SWD, jak: działanie języka, sprawność języka komunikacji czy baza wiedzy, jaką decydent musi opanować, aby efektywnie porozumiewać się z systemem. Obecnie zastosowanie obrazu, dźwięku, metafory czy dialogu w języku naturalnym jako kanałów komunikacji między nadawcą (systemem) a odbiorcą (decydem) tworzy znacznie szersze możliwości indywidualizacji interfejsu użytkownika w sensie dostosowania go do jego cech osobowościowych [Olszak 1999]. Przełom w projektowaniu nastąpił w wyniku opracowania systemów ekspertowych (bliżej scharakteryzowanych w rozdziałach 10 i 11), w których, w odróżnieniu od podejść poprzednich, zastosowano nowe sposoby reprezentowania wiedzy, heurystyczne metody rozwiązywania problemów oraz oddzielono system wiedzy od wnioskowania i kontroli z obszaru SWD, generując nową klasę systemów SWD z bazą wiedzy [Sroka 1994]. Taka nowa architektura SWD umożliwiła włączenie specjalistycznej wiedzy i ekspertyz do systemu, a także generowanie konkluzji i wyjaśnień w ramach specyficznych problemów decyzyjnych. Szerzej na ten temat wypowiadają się m.in. tacy autorzy, jak: R.H. Sprague, E. Carslon [1982] i C.M. Olszak [1988b].

W koncepcji projektowania SWD zorientowanej na decyzje i organizację uwaga skoncentrowana jest nie na dostarczaniu informacji, ale na ustalaniu i wspomaganiu kluczowych decyzji. Problemy decyzyjne postrzegane są jako bardziej rozmyte i różnorodne z powodu presji turbulentnego otoczenia opartego na coraz bardziej komplikujących się związkach kooperacyjnych [Bojar 2010]. Ze względu na różnorodność sytuacji decyzyjnych zakres działań SWD oraz stopień złożoności sytuacji wspomaganych przez systemy C.M. Olszak [2000] proponuje bardziej szczegółową ich typologię, a poszerza ją w kontekście wzbogacenia SWD o bazę wiedzy i mechanizm wnioskowania, co w efekcie poszerza obszary ich zastosowań, np. w sferze porad eksperckich, wyjaśniania wniosków eksperta, inteligentnego wspomagania decydenta w obszarze modelowania, wyboru właściwej metody rozwiązania problemu czy pomocy przy formułowaniu pytań. Jednak koncepcja taka adresowana jest głównie do pojedynczego decydenta, co w warunkach często grupowego, opartego na konsensusie sposobu wypracowywania decyzji determinuje konieczność tworzenia grupowych SWD–GSWD. Ich głównym celem jest nadzorowanie procedur zbiorowego podejmowania decyzji poprzez m.in. zapewnianie wsparcia równoległego przetwarzania informacji, rozwiązywania konfliktów, generowania idei, szerszej wymiany poglądów, symultanicznych interakcji użytkowników itp. [Olszak, Sroka 1999]. GSWD zwiększają efektywność podejmowania decyzji poprzez stymulowa-

nie członków grupy do sprawniejszej pracy, lepsze diagnozowanie błędów przez grupę niż przez pojedyncze osoby, dostęp do poszerzonych informacji, głębsze rozumienie problemów itp. Wśród wad GSWD dostrzega się m.in. generowanie rozwiązań kompromisowych kosztem jakości, eliminowanie oryginalnych pomysłów, tłumienie dynamicznej wymiany zdań, skrępowanie uczestników itp. Zarówno SWD, jak i GSWD koncentrują się na projektowaniu specyficznych lub wąskich problemów decyzyjnych, co powoduje, że ich więzi z organizacją pojmowaną holistycznie są niedostateczne. Nowe zatem generacje SWD powinny obejmować zadania, decyzje i procesy biznesowe, także w kontekście rosnącego znaczenia sieci i łańcuchów dostaw, zapewniać wychodzenie poza granice organizacji, czyli stosowanie podejścia „inside-out”, szerzej charakteryzowanego przez H. Srokę i S. Stanka [1999], a polegającego na identyfikowaniu szans i skali korzyści w bliższym i dalszym otoczeniu organizacji, np. poprzez wirtualizację. Architektura systemów powinna uwzględniać kategorie ogólnodostępnych baz danych, efektywnych narzędzi komunikacji oraz systemów multimedialnych. Zaawansowane w tych obszarach rozwiązania IT powinny wytyczać kierunki doskonalenia SWD tak, aby identyfikować kryteria projektowania narzędzi podwyższających jakość podejmowania decyzji, opracowywać skuteczniejsze metody włączania menedżerów do współpracy z systemem, pogłębiać zrozumienie interakcji człowiek–komputer oraz na podstawie oddziaływania SWD na menedżerów i organizacje doskonalić strategie ich projektowania, wdrażania oraz eksploatacji. Należy także oceniać szanse i zagrożenia we wspomaganium kadry zarządczej i na tej podstawie wytyczać nowe metody podejmowania decyzji oraz tworzyć skuteczne narzędzia informatyczne.

W dalszej części (podrozdział 5.2) przybliżono Czytelnikowi szczegółowe rozwiązania z obszaru projektowania SWD oraz najnowsze trendy w rozwoju architektury SWD.

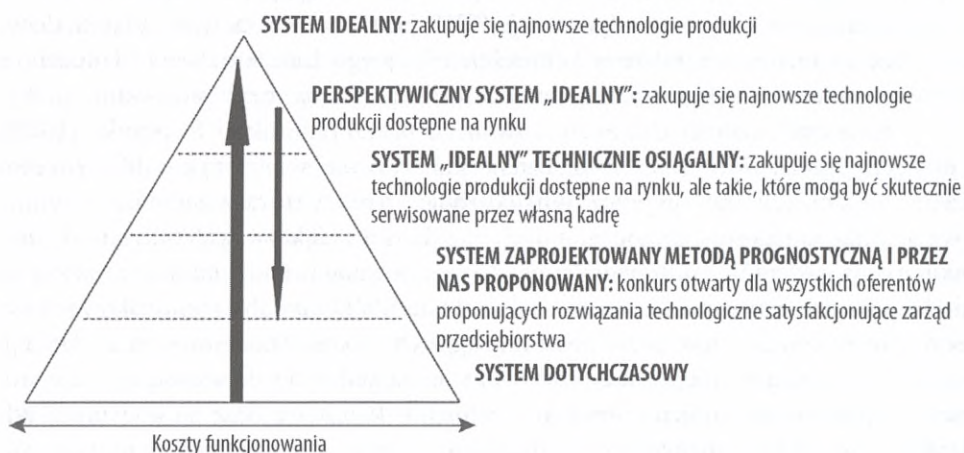
Metoda systemu idealnego — IDEALS jest to koncepcja projektowania i usprawniania systemów działania, a przede wszystkim organizacji, opracowana przez G. Nadlera. Koncepcja ta polega na tym, że zamiast usprawnień elementów badanego systemu poszukuje się właściwej koncepcji efektywnego zorganizowania tego systemu, przechodząc od koncepcji idealnej i zbliżając się stopniowo do koncepcji systemu dającego się zrealizować w praktyce, tj. uwzględniającego warunki ograniczające [Martyniak 1986].

Koncepcję systemu idealnego G. Nadler wyjaśnia graficznie za pomocą trójkąta. Postępowanie według metody trójkąta Nadlera polega na systematycznym i jednoczesnym badaniu wyobrażonych systemów idealnych i systemów istniejących w celu przejścia od ideału do najlepszego rozwiązania dającego się zastosować w praktyce. G. Nadler definiuje ogół czynników fizycznych, ekonomicznych i socjologicznych, które wspólnie tworzą środowisko systemu organizacyjnego.

Projektowanie współczesnych SWD, wyposażonych w zaawansowane narzędzia inżynierii wiedzy (których podstawowym zadaniem jest zbieranie i formalizacja wiedzy ekspertów do postaci gotowych do zastosowania reguł decyzyjnych) po-

winno się odbywać z uwzględnieniem wszechstronnie zweryfikowanych rozwiązań koncepcyjnych — warto w takim przypadku wzorować się na koncepcji trójkąta Nadlera (rysunek 5.1; [Martyniak 1986]) i wspierających ją technikach projektowania organizacyjnego.

Rysunek 5.1. Trójkąt Nadlera



Źródło: Opracowanie własne na podstawie: [Martyniak 1986].

Najbardziej zaawansowaną postacią SWD są systemy ekspertowe wspomagające i uzasadniające (wyjaśniające) procesy podejmowania decyzji, zwłaszcza w zastosowaniach projektowych i w zarządzaniu projektami [Power 2000]. Podejście stosowane do projektowania w zależności od uwzględnienia parametru czasu może być:

- diagnostyczne (jak jest dzisiaj),
- pragmatyczne (trzeba się zaszczepić na grypę — jak będzie jutro),
- w zależności od metody działania: przyrostowa, kaskadowa, spiralna, ewolucyjna.

### 5.1.1. Przyrostowa i kaskadowa strategia projektowania

**Przyrostowa strategia projektowania** jest oparta na modelu przyrostowym. Dla całego systemu są łącznie realizowane:

- etapy wstępne (wymagania i analiza),
- etapy końcowe (testy, instalacja i wdrażanie),
- etapy środkowe (są realizowane oddzielnie, rozłącznie w różnych okresach).

Procedura przyrostowa stosowana jest wtedy, kiedy istnieją ograniczenia kadrowe i finansowe. Przy dobrej organizacji pracy zespół projektowy jest stale zajęty realizacją projektu. Jest to klasyczny przykład zastosowania podejścia strukturalnego.

Model kaskadowy związany z **kaskadową strategią projektowania** jest oparty na założeniu, że wszystkie wątpliwości są rozwiązywane na czas. W praktyce chęć jednoznacznego określenia wszystkich elementów prowadzi do zwiększenia ryzyka i wzrostu kosztów. Procedura kaskadowa polega na tym, że proces projektowy odbywa się stopniowo, stosowana jest zasada sprzężenia zwrotnego, ponieważ nie znamy wszystkich elementów projektu, często powracamy do etapu poprzedniego.

Model ewolucyjny realizuje strategię, w której cele stale ulegają zmianie i stąd konieczne jest permanentne monitorowanie oraz korekta realizacji projektu. Procedura ewolucyjna polega na tym, że projekt jest wielokrotnie modyfikowany i dostosowany do zmieniających się wymagań użytkowników.

### 5.1.2. Spiralna strategia projektowania

**Spiralna strategia projektowania** oparta jest na trójwymiarowym modelu spiralnym. Model jest w całości realizowany oraz przeszedł fazę weryfikacji i testów. Czas realizacji jest w tym przypadku najdłuższy, najbardziej kosztowny i stosowany do realizacji sprzętów.

Procedura spiralna polega na opracowaniu tzw. prototypów. Każdy prototyp jest sprawdzany w wyniku stosowania metod symulacyjnych. Procedura składa się z: planowania, analizy ryzyka, konstrukcji prototypów oraz weryfikacji.

### 5.1.3. Architektoniczne uwarunkowania rozwoju SWD

Rozwój możliwości sieciowych zastosowań SWD jest związany z rozwojem techniki mikroprocesorowej, sieci i telekomunikacyjnych technologii sieciowych, internetu oraz architektury klient-serwer, co stworzyło warunki do udostępnienia instrumentów decyzyjnych znacznie szerszej grupie interesariuszy, niż było to możliwe wcześniej.

Do niedawna SWD przeznaczone dla wybranych organizacji i wąskiej grupy decydentów mają szansę stać się obecnie rozwiązaniami powszechnie dostępnymi. Dla podkreślenia tego faktu wprowadzono określenie sieciowe SWD lub SWD na żądanie [Power 2000], [Olszak 2007]. **Sieciowy model SWD** ma na celu generowanie nowych relacji między przedsiębiorstwami, klientami, dostawcami i akcjonariuszami oraz wykształcenie na kanwie współpracy tych interesariuszy kolektywnej inteligencji. Proponowane rozwiązanie modelowe odwołuje się do pracy w multidyscyplinarnych zespołach, które wykraczają poza tradycyjne granice organizacji i wychodzą naprzeciw nowej koncepcji zarządzania, określanej mianem kooperacji.

Tworzenie sieciowych SWD okazuje się jednak zadaniem trudnym i skomplikowanym. Związane jest to z koniecznością udostępnienia efektów tych rozwiązań użytkownikom o bardzo zróżnicowanych potrzebach przez całą dobę i 7 dni w tygodniu (na żądanie). Wysuwa się sugestie, aby ich struktura nie była narzucona, ale raczej wyłaniała się w naturalny sposób i była budowana interaktywnie w odpowie-

dzi na szybko zmieniające się potrzeby organizacji. Jak wynika z opisanych uwarunkowań, architektura Web-SWD wymaga dostępu do obszernych baz danych i powinna być wyposażona w mechanizmy zapewniające szybką i sprawną reakcję systemu na zadawane przez użytkownika pytania. W założeniu system powinien dostarczać informacji i narzędzi decyzyjnych z wykorzystaniem przeglądarki internetowej, np. Netscape Navigator lub Internet Explorer. Serwer, który jest hostem aplikacji SWD, łączy się z komputerem użytkownika przez sieć i umożliwia zadawanie pytań *ad hoc*, raportowanie oraz analizę informacji zawartych w bazach danych, hurtowniach danych i zintegrowanych systemach typu MRP II [Power 2000]. Do wyświetlania wyników przetwarzania oraz przeprowadzania interaktywnych analiz danych i modeli służą m.in. takie narzędzia informatyczne, jak np.: Java, JavaScript, Perl. Mimo barier w budowie sieciowych SWD pojawia się coraz więcej ofert w internecie, które realizują tę ideę. Są to np.: InterNetivity, Microsoft Carpoint, drkoop.com, Netscape's Decision Guides, Intrinsic Value Calculator. Użytkownicy mają sposobność korzystania z wielu narzędzi służących do przeprowadzania analiz typu „co będzie, gdy”, z równoczesnym dostępem do obszernych baz danych, przechowujących informacje na temat polityki finansowej, fiskalnej, zagranicznej, prawnej, ekologicznej itd. [Martyniak 1986], [Dziuba 2005], [Olszak 2007].

## 5.2. Elementy składowe architektury SWD

SWD jest klasą technologii komputerowej integrującą informacje z baz danych i analityczne metody modelowania (sztuczną inteligencję, analizę decyzyjną, modele optymalizacyjne) według L. Adelmana [1991] i H. Herbsta [1996]. Cechą SWD powinna być łatwość tworzenia zapytań i realizowania raportów w dowolnej formie, bez konieczności długotrwałego studiowania struktur danych i uczenia się nowych narzędzi. Ponadto SWD umożliwia wykonanie analiz biznesowych zróżnicowanych pod względem sposobu doboru niezbędnych danych, jak również pod względem wymaganej szczegółowości. Dlatego systemy SWD są budowane w oparciu o potencjał informacyjny zawarty w hurtowniach danych (HD). Podstawą konstruowania tak zaawansowanych rozwiązań są równie skomplikowane narzędzia wykorzystywane w pracy HD.

Przez pojęcie **SWD** należy rozumieć zintegrowany system, który pozwala uzyskać dowolny, pożądaný stopień szczegółowości danych o procesach realizowanych w przedsiębiorstwach. Podejście takie różni się od dawniej spotykanych rozwiązań, polegających na tworzeniu odrębnych architektur dla klas użytkowników, charakteryzujących się różnymi wymaganiami w zakresie szczegółowości analiz, a także zakresu tematycznego wykorzystywanych danych.

Poprzez swój uniwersalny charakter hurtownie danych wykorzystywane we współczesnych SWD pozwalają zapobiegać problemom technicznym, np. wynikającym z konieczności utrzymania spójności danych. Są one także rozwiązaniem umożliwiającym minimalizację kosztów zastosowania rozwiązań SWD. Zawarte w hur-



towniach dane historyczne są podstawą trafnego planowania i podejmowania właściwych decyzji ze względu na dłuższy horyzont czasowy, który na ogół obejmują. Problemy właściwego zastosowania HD w organizacjach naświetla L. Drelichowski i zespół [2012], diagnozując czynniki sukcesu ich efektywnego wykorzystania w procesie wspomagania decyzji.

W dalszej części rozdziału przedstawiono koncepcje teoretyczne dominujące w SWD. Kluczową rolę odgrywa architektura programowo-techniczna SWD, od której zależą cechy użytkowe tych systemów.

### 5.2.1. Bazy i hurtownie danych

Główną różnicą między budową BD w standardowym systemie OLTP zorientowanym na przetwarzanie (systemy ERP, CRM, MIS) a HD jest to, że relacyjny model systemu źródłowego w HD jest zdenormalizowany do tabel faktów i wymiarów, które są typowymi komponentami architektury HD. Różnice te wynikają przede wszystkim z odmienności ich przeznaczenia.

W typowym systemie OLTP wydajność BD jest ważnym parametrem jej funkcjonowania, ponieważ interfejs użytkownika musi szybko odpowiadać i reagować na to, co dzieje się w systemie. Jakość i użyteczność takiego systemu jest w znacznym stopniu zależna od tego, jak szybko dane mogą być zapisane do BD. Aby to osiągnąć, BD systemu OLTP jest z reguły zoptymalizowana pod kątem szybkości operacji INSERT, UPDATE, DELETE i nastawiona na przechowywanie jak najmniejszej liczby rekordów. W przeciwieństwie do systemów OLTP HD to system, który powinien być w stanie odpowiedzieć niemalże natychmiast na praktycznie każde pytanie związane z działalnością przedsiębiorstwa.

W większości przypadków informacja dostarczana z HD jest używana przez menedżerów, którzy podejmują decyzje biznesowe. Dlatego informacja ta powinna być dostępna szybko i w prosty sposób, ale niekoniecznie musi ona być najbardziej aktualna i bardzo szczegółowa (menedżer będzie raczej szukał w HD odpowiedzi na pytanie o wartość sprzedaży w podziale na regiony na dzień wczorajszy niż dzisiejszych faktur z ostatniej godziny). HD są najczęściej odświeżane w godzinach nocnych lub w trybie miesięcznym — gdy systemy źródłowe i sama HD są najmniej obciążone i mogą zostać szybko i sprawnie zaktualizowane. Główną cechą HD jest przechowywanie zagregowanych danych historycznych. Z punktu widzenia analizy zapytań SQL do HD ich znaczna część zaczyna się od instrukcji SELECT. Typowe modele organizacji danych w HD to schematy: gwiazdy, płotka śniegu oraz konstelacji faktów.

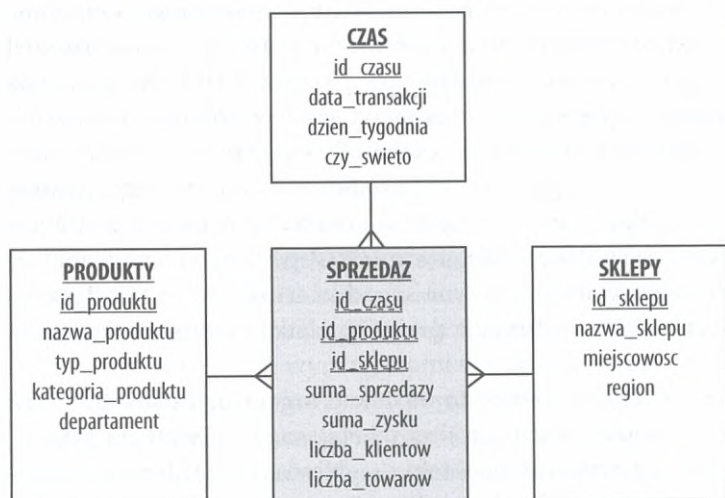
**Architektura gwiazdy** jest najprostszym modelem organizacji danych w HD. Główną cechą schematu gwiazdy jest centralna tabela, nazywana tabelą faktów, z którą połączone są tabele wymiarów. Model taki umożliwia przeglądanie poszczególnych kategorii oraz przetwarzanie faktów według różnych kryteriów określonych przez użytkownika, pozwalających m.in. na selekcionowanie danych i ich odpowied-

nie zestawianie oraz opracowywanie. Tabela faktów w HD modelu gwiazdy jest w trzeciej postaci normalnej, podczas gdy tabele wymiarów są w HD zdenormalizowane, czyli reprezentują drugą postać normalną. Tabela faktów jest zaprojektowana zupełnie inaczej niż typowa tabela relacyjnej BD. Jej denormalizacja jest celowa, aby zwiększyć wydajność i responsywność zapytań kierowanych do BD.

**Tabela faktów** zawiera rekordy gotowe do eksploracji, zwykle łatwo dostępne poprzez zapytania pisane *ad hoc*. Rekordy w tabeli faktów mogą być postrzegane jako wydarzenia, co jest spowodowane naturą hurtowni danych. Klucz główny tabeli faktów jest zwykle złożony ze wszystkich kolumn, z wyjątkiem wartości numerycznych (np. ILOŚĆ TOWARU, WAGA TOWARU, WARTOŚĆ SPRZEDAŻY). Prawie wszystkie informacje w typowej tabeli faktów są reprezentowane również w jednej lub w wielu tabelach wymiarów. Głównym celem utrzymywania tabeli wymiarów jest umożliwienie przeglądania kategorii przez użytkownika w sposób przystępny i oszczędny w czasie. Klucz główny każdej z tabeli wymiarów jest związany z tabelą faktów i jest to składowa złożonego klucza głównego tabeli faktów.

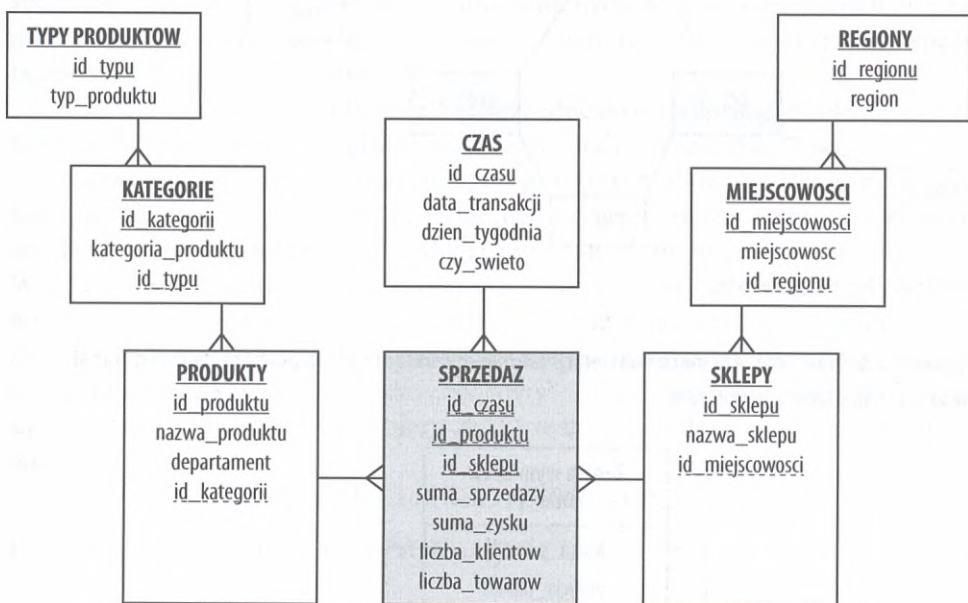
W schemacie gwiazdy występuje tylko jedna zdenormalizowana tabela dla każdego z wymiarów. Przykład architektury HD w schemacie gwiazdy został przedstawiony na rysunku 5.2. **Schemat płatka śniegu** można definiować jako model reprezentujący bardziej złożoną wersję schematu gwiazdy. Główna różnica między nimi wynika z faktu, że w schemacie płatka śniegu tabele wymiarów są zaprojektowane zgodnie z modelem relacyjnej BD, czyli są znormalizowane. Schematu takiego używa się przede wszystkim wtedy, kiedy tabela wymiarów osiąga duże rozmiary i w schemacie gwiazdy jest trudno przedstawić kompleksowość takiej struktury danych.

Rysunek 5.2. Przykład architektury hurtowni danych w schemacie gwiazdy



Na przykład, jeżeli tabela wymiarów z danymi o produktach (PRODUKT) zawiera miliony wierszy, to użycie schematu płątka śniegu powinno znacznie zwiększyć wydajność zapytań — głównie w wyniku przeniesienia części danych do innej tabeli (np. kategorie produktów do tabeli KATEGORIA\_PROD). Problem polega na tym, że im bardziej znormalizowana jest tabela wymiarów, tym bardziej złożone będą zapytania SQL, aby pobrać odpowiednie dane z BD. Przyczyną jest fakt, że zapytanie SQL musi sięgać do kilku tabel, a dane te są zagregowane. Dlatego też zapytanie takie będzie miało rozbudowaną część po poleceniu WHERE oraz GROUP BY. Przykład architektury HD w schemacie płątka śniegu został przedstawiony na rysunku 5.3.

Rysunek 5.3. Przykład architektury hurtowni danych w schemacie płątka śniegu



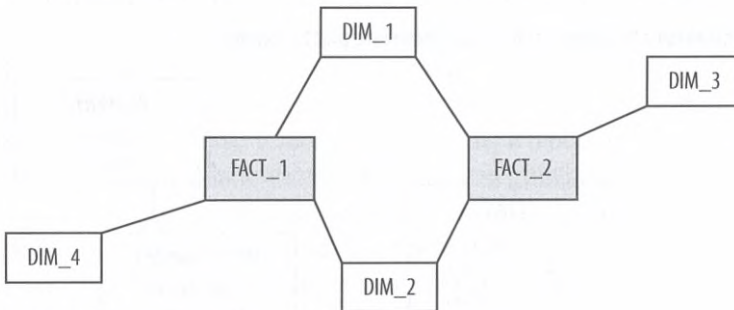
Źródło: Opracowanie własne.

**Architekturę konstelacji faktów** można zbudować na podstawie dowolnego schematu gwiazdy lub schematu płątka śniegu. Architektura ta jest bardziej skomplikowana, ponieważ może zawierać wiele tabel faktów, które mogą się odwoływać do różnych poziomów danego wymiaru. Rozwiązanie to jest bardzo elastyczne i daje wiele możliwości, jednak niekorzystny jest wzrost kosztów utrzymania na skutek wyższego stopnia trudności. Słabą stroną tego rozwiązania jest wysoki stopień złożoności, ponieważ podczas analiz należy rozważyć wiele różnych możliwości agregacji. W schemacie konstelacji fakty są przyporządkowane tym wymiarom, które są istotne dla danej tabeli faktów. Architektura ta jest często wykorzystywana w sytuacji, gdy fakty są przyporządkowane wymiarom na różnym poziomie szczegółowości, np. kiedy tabela faktów sprzedaży (o poziomie szczegółowości obejmującym dokładne daty

sprzedaży i faktury) ma być połączona z tabelą prognoz, które są naliczane miesięcznie w układzie klient–produkt.

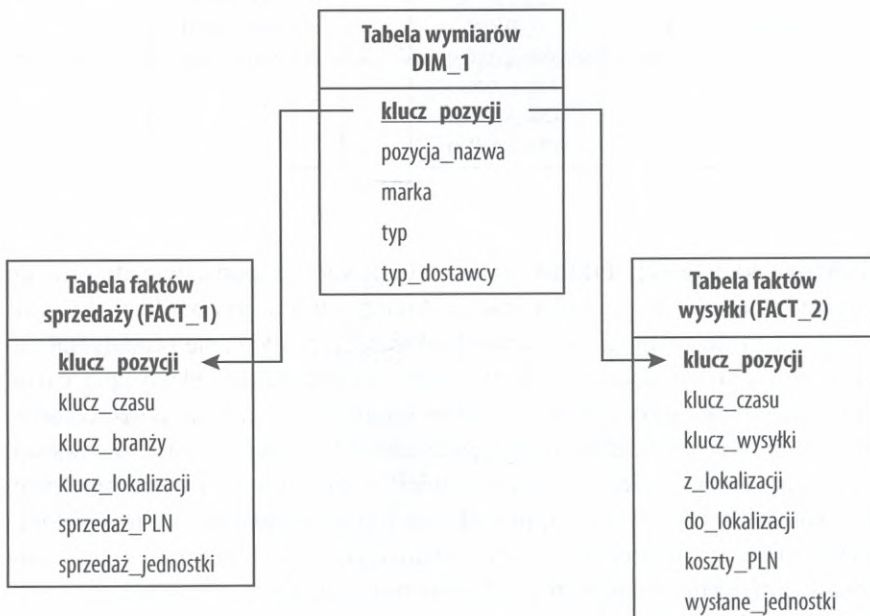
Przykład architektury hurtowni danych w schemacie konstelacji faktów został przedstawiony na rysunku 5.4, na którym pokazano powiązania poszczególnych tabel faktów z tabelami wymiarów, oraz na rysunku 5.5, na którym przedstawiono szczegółowo relacje dwóch tabel faktów z jedną tabelą wymiarów.

**Rysunek 5.4. Przykład uproszczonego schematu konstelacji faktów:**  
różne tabele faktów mogą się odwoływać do różnych poziomów danego wymiaru



Źródło: Opracowanie własne.

**Rysunek 5.5. Fragment schematu konstelacji faktów obrazujący sposób powiązania dwóch tabel faktów z jedną tabelą wymiarów**



Źródło: Opracowanie własne.

## 5.2.2. Baza wiedzy i moduł wnioskujący

**Bazę wiedzy** (BW) stanowią wszystkie udowodnione i zapisane fakty o opisywanym obiekcie lub systemie (aksjomaty). Aksjomaty to formuły, których w systemie wnioskowania używa się wraz z formułami z początkowej bazy wiedzy jako „punktów początkowych” procesu wnioskowania, wyprowadzając z nich nowe formuły. Ściśle ujmując, aksjomaty są nie tyle formułami, ile raczej wzorcami formuł, które można konkretyzować odpowiednio do potrzeb. Z punktu widzenia teorii jako sformalizowanego opisu świata rzeczywistego zbiór aksjomatów logicznych (tautologii) oraz aksjomatów specyficznych, a także język, czyli zbiór formuł i reguły wnioskowania, są jej niezbędnymi składowymi opisującymi. Jednak aksjomaty logiczne jako zbiory ogólnych prawd o świecie, nieudowodnionych, lecz przyjmowanych za pewniki, oraz reguły wnioskowania są ustalone. Zatem teorię tworzą aksjomaty specyficzne i one budują bazę wiedzy.

BW zawiera fakty i reguły, które są niezbędne do rozwiązania problemu w określonej dziedzinie, np. w zarządzaniu zasobami wiedzy przedsiębiorstwa.

**Fakty** są zdaniem oznajmującymi, np. kombajn zbożowy Bizon ma wydajność techniczną 1 ha/h. Fakt ten może być reprezentowany w postaci związku między pewnymi obiektami i charakteryzować się różnymi cechami (atrybutami). W podanym przykładzie obiekt (kombajn zbożowy) ma dwie cechy: nazwę oraz wydajność techniczną. Między obiektami w bazie wiedzy mogą istnieć różne związki. Na przykład, kombajn ma wszystkie charakterystyki maszyny przeznaczonej do zbioru i oprócz tego jako „zbożowy” — specjalne charakterystyki, które są specyficzne dla maszyn do zbioru zbóż i roślin technologicznie podobnych, np. do zbioru rzepaku.

Oprócz faktów baza wiedzy zawiera **reguły** o postaci:

IF warunek THEN wniosek AND/OR akcja. [5.1]

Relacje odzwierciedlają zależności i asocjacje (skojarzenia) między faktami w bazie wiedzy.

**Reguła decyzyjna** jest to wyrażenie opisujące zbiór warunków, które muszą być spełnione, aby został wykonany ciąg czynności. Zatem reguła decyzyjna jest swoistą relacją między warunkiem i tezą według postaci:

$r$ : (warunek)  $\rightarrow$  (teza). [5.2]

Uznanie jakiejś reguły nazywa się jej uaktywnieniem (ang. *firing*), co oznacza, iż następuje uznanie prawdziwości wszystkich faktów w części warunkowej i reguła zostaje uaktywniona, a hipoteza uznana za prawdziwą kończy wnioskowanie. Dzięki uaktywnieniu reguł nowe fakty (wnioski) są dodawane do BW i/lub wykonywane są odpowiednie akcje. Systemy ekspertowe oparte na regułach nazywa się systemami regułowymi, ponieważ reguły są jedną z najczęściej używanych form reprezentacji wiedzy w tych systemach [Valenta, Śnieżyński, Zygmunt 2000].

Podczas tworzenia bazy wiedzy inżynier wiedzy musi odpowiedzieć na następujące pytania:

- Jakie obiekty należy zdefiniować?
- Jakie relacje występują między obiektami?
- Jak należy formułować i przetwarzać reguły?
- Czy pod względem rozwiązywania specyficznego problemu baza wiedzy jest kompletna i spójna?

Możemy wyróżnić następujące rodzaje baz wiedzy:

- baza tekstów (ang. *text base*),
- baza danych (ang. *data base*),
- baza reguł (ang. *rule base*),
- baza modeli (ang. *model base*),
- baza wiedzy nieformalnej (zdroworozsądkowej, niejawnej; ang. *common sense knowledge base*).

**Baza tekstów** przechowuje specyficzne, szczegółowe informacje, zapisane w sposób uporządkowany, w wyniku którego jest możliwe potem jej przeszukiwanie według kryteriów interesujących użytkowników tej **bazy danych**.

**Baza reguł** zawiera wiedzę o obowiązujących w wybranej dziedzinie zależnościach oraz prawidłowościach, a także prawach i jest najważniejszą częścią bazy wiedzy.

**Baza modeli** zawiera modele matematyczne (logiczne) danej dziedziny. Modele te są logicznym przedstawieniem pojęcia, systemu lub działań. Wyróżniamy trzy typy modeli:

- 1) modele deterministyczne — są analitycznym przedstawieniem pojęcia, systemu lub działań, w którym dla danych wielkości wejściowych wyniki są określone jednoznacznie;
- 2) modele niedeterministyczne (lub stochastyczne) — modele, w których powiązania funkcyjne zależą od wielkości losowych; dla danych wielkości wejściowych wyniki mogą być jedynie przewidziane zgodnie z zasadami probabilistyki;
- 3) modele wartości oczekiwanych — modele, w których wielkościom losowym zostały nadane ich wartości oczekiwane.

**Baza wiedzy nieformalnej** (zdroworozsądkowej, niejawnej) odzwierciedla potencjalne, racjonalne zachowania się człowieka. Zawiera ona reguły definiujące sposoby podejmowania decyzji, reprezentujące metawiedzę systemu ekspertowego. Metawiedza stanowi wiedzę o wiedzy, czyli o sposobach przetwarzania wiedzy z danej dziedziny.

Wyróżniamy dwa podstawowe typy symbolicznej reprezentacji wiedzy:

- 1) reprezentację proceduralną, określającą zbiór procedur, których działanie reprezentuje wiedzę o dziedzinie;

- 2) reprezentację deklaratywną, określającą zbiór specyficznych dla rozpatrywanej dziedziny faktów, stwierdzeń i reguł.

Do najczęściej stosowanych **metod reprezentowania wiedzy** (technik organizowania baz wiedzy) należy zaliczyć:

- metody bazujące na bezpośrednim zastosowaniu logiki: rachunek zdań, rachunek predykatów;
- metody wykorzystujące zapis stwierdzeń;
- metody wykorzystujące systemy regułowe (wektory wiedzy);
- metody z wykorzystaniem sieci semantycznych;
- metody oparte na ramach;
- metody używające modeli matematycznych.

Inną metodą reprezentacji wiedzy są reprezentacje niesymboliczne [Krawiec 2003]. Metody te odwołują się do doświadczeń i obserwacji otaczającego nas świata materii żywej — organizmów żywych. Tzw. sztuczne sieci neuronowe symulują własności reprezentacji wiedzy i jej sposób przetwarzania na wzór komórek neuronowych zwierząt i ludzi. Przetwarzanie wiedzy odbywa się tam w sposób dynamiczny. Wiedza jest kumulowana w pewnym systemie połączeń między poszczególnymi neuronami oraz za pomocą pewnych wartości wag określających siłę tych połączeń.

**Algorytmy genetyczne** stanowią inną technikę reprezentacji wiedzy, również wzorowaną na metodzie przekazywania informacji z pokolenia na pokolenie przez istoty żywe, co pozwala udostępniać następnym generacjom wiedzę o całym gatunku. Wiedza ta jest zapisana w genach.

Głównym zadaniem **logiki rachunku zdań** jest upraszczanie wyrażeń logicznych w celu ich bardziej efektywnego przetwarzania lub dowodzenia ich poprawności. Wykorzystuje się tu równoważność wyrażeń. Oprócz zdań często używane są funkcje zdaniowe, które stają się zdaniami po ustaleniu wartości zmiennych. Bazy wiedzy mające takie cechy, jak: modularność, deklaratywność i nieproceduralność, a oparte na logice, są trudne do przetwarzania, ponieważ błyskawicznie następuje w nich eksplozja kombinatoryczna, czyli lawinowy i niekontrolowany rozrost bazy wiedzy. Powielane są fakty będące kopią istniejących już informacji lub powstają niepożądane struktury. Procesowi temu można zapobiegać przez korzystanie z wiedzy uzyskiwanej w wyniku zastosowania metod i reguł poszukiwania heurystycznego.

Czasami zapobiega to kombinatorycznej eksplozji drzewa możliwości. „Heurystyczny” oznacza proces mogący doprowadzić do rozwiązania, ale nie gwarantuje tego. Proces heurystyczny jest więc przeciwstawieniem ślepego przeszukiwania.

**Stwierdzenia** są jednym z głównych elementów baz wiedzy. Dotyczą one takich zagadnień, jak: zdarzenia, zjawiska, objawy, czynności. Stwierdzenia najczęściej są zapisywane w postaci uporządkowanej trójki:

(<OBIEKT> , <ATRYBUT> , <WARTOŚĆ>). [5.3]

Dla uproszczenia zapisów stwierdzeń stosuje się słowniki nazw obiektów i atrybutów oraz ich wartości. Umożliwia to identyfikowanie zapisów za pomocą etykiet — bez wielokrotnego powtarzania nazw. Na ogół stosuje się słowniki otwarte. Sieci semantyczne lub ramy są zwykle używane do wyrażenia relacji między obiektami. Należy zauważyć, że w reprezentacji wiedzy za pomocą stwierdzeń występują trudności negowania stwierdzeń w przypadku atrybutów wielowartościowych.

W niektórych systemach stwierdzenia są zapisywane w postaci uporządkowanej czwórki:

(<OBIEKT> , <ATRYBUT> , <WARTOŚĆ> , <CF>), [5.4]

gdzie:

CF — stopień pewności.

CF prowadzi do tzw. stwierdzeń przybliżonych, ponieważ każdemu stwierdzeniu jest przypisywany pewien współczynnik określający stopień pewności tego stwierdzenia.

Rozwinięto także statystyczne metody wnioskowania wykorzystujące tzw. metody Bayesa, teorię podejmowania decyzji i metody weryfikacji hipotez statystycznych. Baza wiedzy takich systemów jest zbiorem stwierdzeń i relacji o pewnym wy-cinku świata rzeczywistego, której dotyczy.

**Regułowa reprezentacja wiedzy** stanowi, obok reprezentacji w postaci ram, najpowszechniej stosowaną metodę formalnego zapisu wiedzy. Łatwo zauważyć, że zbiór stwierdzeń nie jest wystarczający do opisania jakiejś dziedziny wiedzy. Są jeszcze niezbędne reguły, których ogólna postać może być wyrażona jako:

IF <przesłanka> THEN <konkluzja> OR <działanie>. [5.5]

**Przesłanka** może zawierać pewną liczbę stwierdzeń połączonych funktorami logicznymi.

Baza wiedzy zawiera w tym przypadku zbiór reguł oraz zbiór faktów. Reprezentacja regułowa jest często wykorzystywana w systemach dedukcyjnych, gdzie zbiór faktów początkowych jest przekształcany w pewien zbiór faktów końcowych. W zależności od przeznaczenia system może spełniać różne funkcje, np. klasyfikowanie, diagnozowanie, dowodzenie, ustalanie przyczyn, najlepszy dobór, planowanie, prognozowanie, monitorowanie itp.

Niekiedy stosowany jest bardziej formalny zapis reguł, w którym jest opuszczane określenie IF, a zamiast słowa THEN używa się symbolu implikacji. Na przykład zamiast:

IF „A przyjmuje wartość  $u$ ” AND „F przyjmuje wartość  $w$ ” THEN „G przyjmuje wartość  $y$ ”,

możemy zapisać:



$$(A, u) \Rightarrow (F, w) \Rightarrow (G, v). \quad [5.6]$$

Praktycznie działające systemy ekspertowe oparte na regułach mogą zawierać reguły charakteryzowane stopniami pewności. Zbiór reguł można rozpatrywać jako szczególny sposób zapisu pewnej sieci stwierdzeń, ponieważ z prawdziwości jednego stwierdzenia mogą wynikać inne. W niektórych systemach jest dopuszczalna rozwinięta (tzw. pełna) postać reguł, która zawiera dodatkowe stwierdzenie (zob. Konkluzja 2) uznawane za prawdziwe w razie niespełnienia przesłanki. Ogólna postać reguły rozwiniętej jest następująca:

$$\text{IF } \langle \text{przesłanka} \rangle \text{ THEN } \langle \text{konkluzja} \rangle \text{ ELSE } \langle \text{konkluzja2} \rangle. \quad [5.7]$$

Taka postać reguł może jednak czasami prowadzić do uznania nieoczekiwanych konkluzji. Ze względu na założoną nieraz kompletność bazy wiedzy, brak przesłanki jest utożsamiany z uznaniem tej przesłanki za przesłankę fałszywą.

Ze względu na sposób uzyskiwania ostatecznych konkluzji w procesie wnioskowania rozróżniamy dwa rodzaje reguł:

- 1) reguły proste — mają postać wniosków pośrednich,
- 2) reguły złożone — umożliwiają bezpośrednie wyznaczenie wniosków przez system.

IF <fakty potwierdziły zdarzenie (spadek zdatności maszyny poniżej minimum)> THEN zastosować odpowiednią procedurę naprawy z Systemu Utrzymania Niezawodności Parku Maszyn i Urządzeń.

Ponieważ każda z reguł złożonych w konkluzji zawiera jakiś wniosek końcowy, nie wymagają one maszyn wnioskujących o skomplikowanym sposobie działania. Korzystne jest, że uaktywnienie jednej reguły pozwala osiągnąć wynik. Wadą takiego podejścia jest trudność formułowania odpowiedniego zbioru reguł oraz złożony sposób jego weryfikacji i uzupełniania.

Łatwość weryfikacji zbioru reguł i ograniczenie ich redundancji należy do zalet reguł prostych. Ich wadą jest konieczność uaktywnienia wielu reguł podczas realizacji złożonych działań przez maszynę wnioskującą. Łańcuch wnioskowania tworzą kolejne udowodnione wnioski pośrednie.

W systemach ekspertowych w części warunkowej znajdują się przesłanki, dzięki którym są zadawane pytania o związki między cechami obiektów. Badanie przesłanek jest najczęściej związane z licznymi efektami ubocznymi, np. analizowanie innych reguł, zadawanie pytań użytkownikowi itp. Znaczenie ma kolejność elementów przesłanki. Odpowiednia kolejność reguł jest ważna w procesie ograniczania liczby operacji wykonywanych przez system podczas sprawdzania przesłanek.

W systemach ekspertowych są stosowane reguły, które odnoszą się do konkretnych obiektów, jak też reguły obejmujące pewien zbiór obiektów — tzw. reguły ogólne. Mogą one być spełnione przez różne obiekty zbioru. Dzięki temu reguły ogólne umożliwiają znaczne zwiększenie stopnia ogólności utworzonej bazy wiedzy. Uzyskuje się to poprzez przyjęcie założenia, że elementy stwierdzeń nie są wartościami stałymi, lecz

zmiennymi. W procesie wnioskowania zmienne są zastępowane odpowiednimi stałymi, określonymi w wyniku dopasowania reguły do istniejącego zbioru stwierdzeń. Nazywamy to procesem unifikacji. Tym sposobem na podstawie jednej reguły zawierającej zmienne można generować wiele reguł dopasowanych do danej bazy faktów.

**Wektory wiedzy** zapisane w postaci wektorowej są pewnego rodzaju uogólnieniem reguł. W podejściu tym najpierw zapisujemy daną bazę reguł w tradycyjny sposób, przy czym poszczególne reguły powinny zawierać jednakową liczbę warunków i wniosków. Następnie dokonujemy kodowania poszczególnych członów reguł z wykorzystaniem symboli (TAK, NIE, nie występuje). W rezultacie zamiast zapisu pełnej reprezentacji poszczególnych reguł otrzymujemy bardzo zwarty opis w postaci wektorów, zawierających trzy wymienione symbole. Po uzyskaniu postaci wektorowej łatwo można przejść do opisu zawierającego pełną treść reguł. Wektory wiedzy są wygodne do weryfikacji poprawności bazy wiedzy i mogą obejmować zestaw wszystkich pytań (warunków), podobnie jak w przypadku reguły złożonej.

Rozważmy przykład bazy reguł zbudowanej z trzech reguł, np.:

- R1:** IF jest dzień THEN linia produkcyjna (l.p.) jest uruchomiona,  
**R2:** IF jest noc THEN linia produkcyjna (l.p.) nie jest uruchomiona,  
**R3:** IF jest awaria linii energetycznej THEN wezwę serwis.

W przedstawionych regułach występuje pięć stwierdzeń, przy czym niektóre w postaci zanegowanej (warunki: jest dzień, jest noc, jest awaria linii energetycznej; wnioski: (nie) jest uruchomiona, wezwę serwis). Do oznaczenia pomijalności lub prawdziwości stwierdzeń występujących w warunkach i wnioskach zastosujemy oznaczenia:

- \* — warunek lub wniosek nie występuje w regule,  
**T** — warunek lub wniosek jest prawdziwy,  
**N** — warunek lub wniosek jest fałszywy.

W rezultacie zakodowania reguł za pomocą symboli otrzymujemy zwarty i przejrzysty zapis reguł w postaci 5-elementowych wektorów, przedstawiony w tabelicy 5.1.

Ważnym narzędziem w metodach reprezentacji wiedzy jest rachunek predykatów, stanowiący podstawę programowania w logice. **Rachunek predykatów** jest rozszerzeniem rachunku zdań przez wprowadzenie kwantyfikatorów:

- $\forall$  — „dla każdego” oraz  
 $\exists$  — „istnieje takie ..., że”.

**Predykat** składa się z nazwy i dowolnej liczby argumentów, które są nazywane termami. Mogą nimi być stałe (symbole alfanumeryczne, jak też numeryczne), zmienne oraz wyrażenia (funkcje). Termy są argumentami predykatów. W wyniku podstawienia stałych za zmienne otrzymujemy zdania prawdziwe lub fałszywe.

Tablica 5.1. Ilustracja konstrukcji i działania reguł decyzyjnych

Dla reguły R1	Dla reguły R2	Dla reguły R3	Znaczenie symboli w warunku/wniosku
T	*	*	Jest dzień
*	T	*	Jest noc
*	*	T	Wezwę serwis
T	N	*	L.p. jest uruchomiona
*	*	T	Wezwę serwis

Źródło: Opracowanie własne.

Definiowanie semantyki języka predykatów następuje poprzez podporządkowanie termom symboli, a nazwie — relacji między obiektami. Zaletą predykatów są proste i zrozumiałe interpretacje wyrażania zdań.

Można zauważyć, że nie wszystkie pojęcia opisujące otaczającą nas rzeczywistość dają się reprezentować w logice. Rachunek predykatów wystarcza do opisu formalizmów większości współczesnych dyscyplin naukowych. Formalność teorii logicznych jest ich istotną zaletą, ponieważ zapewnia ona ścisły i jednoznaczny opis ich konstrukcji oraz mechanizmów, co przy opisie wielu problemów odgrywa kluczową rolę.

Kolejnym modelem wykorzystywanym przy reprezentowaniu wiedzy są **sieci semantyczne**. Zostały one opracowane przez R.M. Quilliana [Quillian 1968]. Pierwotnym założeniem R.M. Quilliana był zamiar stworzenia modelu pamięci ludzkiej. U podstaw modelu sieci semantycznych znalazło się założenie, że baza wiedzy stanowi zbiór stwierdzeń i relacji między nimi. Posługując się tymi pojęciami, możemy utworzyć tzw. sieć stwierdzeń — jako pewien graf, którego węzłami są stwierdzenia, a gałęziami relacje. Węzłom (podobnie jak gałęziom) mogą być przypisane wagi określające np. stopnie przekonania o słuszności tych stwierdzeń. Sieci semantyczne lub asocjacyjne są uogólnieniem koncepcji sieci stwierdzeń. Uogólnienie to polega na przyjęciu założenia, że węzły odpowiadają kompletnym opisom pojęć lub obiektów i nie są wyłącznie stwierdzeniami. R.M. Quillian przyjął założenie, że pamięć ludzką najlepiej opisuje model asocjacyjny. Oznacza to, że jedne terminy są wyjaśniane przez inne terminy. Każdy element jest zdefiniowany przez inny element. Powstaje zatem wtedy pewna struktura powiązań, która może być zamknięta, a grafy mogą być skierowane.

Sieć semantyczna jest pewnego rodzaju logiką, gdzie relacje między obiektami są przedstawione w postaci grafu. Mamy więc do czynienia z grafowym odwzorowaniem mechanizmu dedukcji. Wnioskowanie odpowiada „poruszaniu się” po grafie. Na podstawie analizy sieci semantycznej wyprowadza się różne konkluzje.

Model reprezentacji wiedzy za pomocą sieci semantycznych ma również swoje wady. Problemem jest np. precyzyjne określenie, czy węzły sieci oznaczają jeden

obiekt, czy też klasę obiektów. Dlatego też sieci semantyczne odzwierciedlają zwykle strukturę złożoną z ram lub z reguł. Ramy w takim podejściu odpowiadają obiektom i opisują ich strukturę wewnętrzną, sieć semantyczna natomiast odpowiada relacjom między ramami.

Istotą sieci semantycznych jest graficzne ujęcie dwuargumentowych relacji między obiektami nazywanymi węzłami. Każdy węzeł odpowiada typowi obiektu i może być związany z innymi obiektami za pomocą łuków skierowanych, nazywanych też połączeniami. Węzły i łuki mają swoje nazwy.

Węzły reprezentują:

- obiekty fizyczne,
- obiekty konceptualne (czynności, wydarzenia),
- deskryptory — dodatkowe opisy charakterystycznych cech obiektu.

Łuki reprezentują relacje zachodzące między obiektami (deskryptory związków zachodzących między obiektami). Wyróżnia się następujące typy relacji:

- relacja typu *I* reprezentuje fakt zawierania się obiektu w obiekcie nadrzędnym;
- relacja typu *J* reprezentuje wartość pojedynczej cechy obiektu (stopa procentowa wynosi 20%) lub szerzej — przynależność do klasy obiektów (złoty jest jednostką monetarną);
- relacja typu *M* identyfikuje obiekty będące częścią innych obiektów (szyba jest częścią okna).

Rysunek 5.6. Przykład sieci semantycznej



Na rysunku 5.6 przedstawiono przykładową sieć semantyczną opisującą kombajn zbożowy Malinowskiego. Sieć ta tworzy graf skierowany. Między poszczególnymi węzłami grafu zachodzą relacje:  $J$  — jest,  $M$  — ma,  $I$  — jest w. Skierowanie grafu oznacza, że relacje zachodzą w jedną stronę. Na przykład, między węzłami *kombajn zbożowy Malinowskiego* i *maszyna samobieźna* zachodzi relacja „jest”. Zatem na podstawie grafu można wyprowadzić stwierdzenie: *kombajn zbożowy Malinowskiego jest maszyną samobieźną*. Nie można jednak wyprowadzić twierdzenia: *maszyna samobieźna jest kombajnem zbożowym Malinowskiego*, co oznaczałoby, że *maszyny samobieźne należą do klasy o nazwie kombajn zbożowy Malinowskiego*.

Model reprezentacji wiedzy za pomocą sieci semantycznych ma również swoje wady. Problemem jest np. określenie, czy węzły sieci oznaczają jeden obiekt czy klasę obiektów. Dlatego też sieci semantyczne wiąże się zazwyczaj z ramami lub z regułami.

W sieci semantycznej mogą także wystąpić pewne pułapki dedukcyjne. W przykładzie pokazanym na rysunku 5.6 możemy wydedukować, że obiekt techniczny jest kupowany przez producentów rolnych, a ten warunek nie zawsze musi być spełniony. Zaletą sieci semantycznych jest ich elastyczność, wynikająca z braku ograniczeń liczby węzłów, łuków i możliwych interpretacji zachodzących między nimi.

Metoda sieci semantycznych jest często stosowana w systemach analizy i interpretacji języka naturalnego. Wynika to z jasności, z jaką możemy przedstawiać zawile struktury składniowe języka za pomocą tej reprezentacji. Sieci semantyczne są także przydatne do tłumaczenia z jednego języka na inny oraz wspomagania uczenia.

Podstawowe koncepcje mechanizmów wnioskowania większości znanych systemów ekspertowych wywodzą się z klasycznej logiki dwuwartościowej. Opis cech otaczającego nas świata jest formułowany w postaci zdań. O zdaniu prawdziwym mówimy, że ma wartość logiczną 1, o zdaniu fałszywym, że ma wartość logiczną 0. Zdania oznaczamy symbolami, np.  $A, B, C, D, \dots$ . Mogą one być łączone za pomocą funktorów zdaniotwórczych (spójników logicznych), przedstawionych w tablicy 5.2.

Tablica 5.2. Ilustracja konstrukcji i działania reguł decyzyjnych

$x$	$y$	$\neg x$ negacja $x$	$x \wedge y$ koniunkcja	$x \vee y$ alternatywa	$x \Rightarrow y$ implikacja	$x \Leftrightarrow y$ równoważność
0	0	1	0	0	1	1
0	1	1	0	1	1	0
1	0	0	0	1	0	0
1	1	0	1	1	1	1

Źródło: Opracowanie własne.

Łącząc zdania funktorami, można tworzyć wyrażenia logiczne, tzw. **formuły rachunku zdań**.

Formuły, które zawsze dają zdanie prawdziwe niezależnie od wartości logicznych zmiennych zdaniowych, nazywamy **tautologiami**. Istnieją dwie metody weryfikacji, czy dana formuła jest tautologią:

- 1) metoda zero-jedynkowa,
- 2) metoda dedukcji (wnioskowania).

W metodzie zero-jedynkowej wartość logiczna formuły złożonej jest wyznaczana przez wartości logiczne jej składników. Aby orzec, czy formuła jest tautologią, należy rozważyć wszystkie możliwe kombinacje wartości logicznych zmiennych w niej występujących. Jeżeli w każdym przypadku wartość formuły wynosi 1, to ta formuła jest tautologią.

W metodzie dedukcji wnioskowanie odbywa się za pomocą reguł wnioskowania, które umożliwiają, na podstawie prawdziwości pewnych zdań zwanych przesłankami, wnioskowanie o prawdziwości innego zdania, zwanego wnioskiem.

Maszyna wnioskująca składa się z procedur przeszukiwania i wnioskowania, co służy znalezieniu rozwiązania i jego uzasadnienia.

Wnioskowanie można rozumieć jako umiejętność podejmowania decyzji na podstawie posiadanej wiedzy. Należy zaznaczyć, że SE dostarcza tylko wiedzy do prawidłowego podjęcia decyzji przez człowieka. W tym znaczeniu twierdzenie, że wnioskowanie to umiejętność podejmowania decyzji jest daleko idącym uproszczeniem, ponieważ żadna aplikacja programowa nie może zastąpić człowieka w tym procesie, a tylko może go wspomóc. W przypadku systemów ekspertowych jest to wyznaczanie nowych faktów na podstawie bazy wiedzy i zbioru faktów początkowych, zadeklarowanych przez użytkownika.

**Wnioskowanie „do przodu”** to postępowanie prowadzące od warunków do wniosku. Na podstawie dostępnych reguł i faktów należy generować nowe fakty tak długo, aż wśród wygenerowanych faktów znajdzie się postawiona hipoteza (np. jeżeli źródło napięcia jest sprawne i obwód jest zamknięty, to silnik elektryczny pracuje).

**Wnioskowanie „wstecz”** to postępowanie prowadzące od hipotezy do warunków. Polega ono na wykazaniu prawdziwości hipotezy głównej na podstawie prawdziwości przesłanek (np. gdy silnik nie pracuje, wówczas obwód jest przerwany lub źródło napięcia jest niesprawne).

Ważnym elementem procesu wnioskowania jest pamięć robocza, która przechowuje wyniki pośrednie toku rozumowania, czyli informacje o tym, jak dotychczas przebiegało wnioskowanie i jakie są jego rezultaty oraz jakie fakty i reguły są obecnie wykorzystywane.

Do zalet rozdzielenia bazy wiedzy i systemu wnioskującego należy:

- łatwość dostępu do wiedzy dziedzinowej, danej w postaci jawnej pliku tekstowego;
- prostota modyfikacji bazy wiedzy bez potrzeby naruszania integralności właściwego programu — systemu wnioskującego;

- możliwość tworzenia uniwersalnych systemów ekspertowych, tzn. systemów, które użytkownicy mogą zapełniać użytkową przez nich wiedzą dziedzinową.

### 5.2.3. Magistrala danych i oprogramowanie integrujące

Współczesne systemy integracji danych oraz inne systemy klasy *middleware* oparte na BD cechuje modułowość. Struktura ta jest wynikiem dekompozycji dużego zadania na wiele mniejszych. Takie podejście pozwala na łatwiejsze tworzenie, zarządzanie i utrzymanie oprogramowania. Dodatkową korzyścią jest możliwość dostarczenia do klienta tylko tej funkcjonalności, której potrzebuje, a jednocześnie umożliwienia łatwiejsze skalowanie systemu, dzięki czemu potencjalne grono odbiorców jest większe. Przykłady programowych rozwiązań integrujących to system ESS II i platforma Eclipse.

Architektura systemu ESS II jest nowoczesnym rozwiązaniem w zakresie systemów integrujących. Struktura tego systemu składa się z dwóch głównych warstw: serwerowej i klienckiej. Warstwa serwerowa zawiera podwarstwę odpowiedzialną za komunikację (mającą postać zestawu interfejsów) oraz podwarstwę przetwarzającą złożoną z procesów i serwerów. W warstwie klienckiej można wyróżnić podwarstwę, w skład której wchodzi narzędzia konfiguracyjne, oraz podwarstwę użytkową. System ESS II jest otwarty i pozwala na podłączanie aplikacji korzystających z funkcji API. W zależności od charakteru stworzonej aplikacji mogą one pozwalać na konfigurację lub udostępniać któryś z trybów pracy.

Otwarta platforma Eclipse posiada ponadto rozszerzalną architekturę dołączanych modułów (wtyczek), która pozwala na pracę z praktycznie każdym rodzajem danych. Pakiet programistyczny zawiera np. wiele dołączanych modułów do nawigacji po źródłowych kodach C/C++, analizy dołączanych i ładowanych obrazów, analizy wydajności systemów pracujących w trybie czasu rzeczywistego. Stwarza także możliwość samodzielnego podłączania lub użytkowania gotowych wtyczek innych producentów. W każdym z opisywanych przypadków integracja podłączanych modułów z istniejącymi narzędziami jest w pełni realizowana.

### 5.2.4. Konsole i interfejsy użytkownika

Współczesne przedsiębiorstwo napotyka problem konieczności analizy przyrastającej w dużym tempie liczby danych, co jest związane ze wzrostem stopnia komplikacji i dynamiki zmian zarówno procesów wewnętrznych firm, jak i zjawisk w ich otoczeniu. Sprawność działania przedsiębiorstwa oraz jego konkurencyjność na rynku zależy bezpośrednio od wdrożonych mechanizmów zarządzania cyklem życia informacji.

Z punktu widzenia częstotliwości wykorzystania oraz dostępności dane mogą zostać podzielone na dwie grupy: dane operacyjne oraz dane referencyjne.

Dane operacyjne są wykorzystywane bardzo często i wymagany jest do nich bardzo szybki dostęp. Dostęp do danych referencyjnych nie jest krytyczny. Bardzo często danymi referencyjnymi stają się po pewnym okresie dane produkcyjne, które nie są już wykorzystywane jako dane operacyjne, ponieważ zmieniły się technologie wytwarzania i park maszynowy.

Analizowany podział nasuwa potrzebę wdrożenia odpowiedniej struktury systemu przetwarzania danych w przedsiębiorstwie. Dane produkcyjne powinny być przechowywane na szybkich, łatwo dostępnych i jednocześnie kosztownych urządzeniach pamięci masowej. Utrzymywanie danych referencyjnych, ze względu na charakter dostępu, na tych samych urządzeniach pamięci masowej oznaczałoby ponoszenie niepotrzebnych kosztów. Takie informacje, bez negatywnego wpływu na jakość działania przedsiębiorstwa, mogą być przechowywane na znacznie tańszych (mniej wydajnych) nośnikach danych. To podejście gwarantuje optymalizację kosztów infrastruktury informatycznej. Uwarunkowania te powodują, że niezwykle ważne staje się zarządzanie mechanizmami składowania danych. Proces ten powinien być zautomatyzowany i w pełni transparentny dla użytkowników, a mogą temu służyć m.in. konsole oraz generatory aplikacji. Przykłady konsoli i generatorów aplikacji to: konsola HP OpenView Storage Data Protector, menedżer zabezpieczeń Oracle Enterprise, system SCADA RealFlex 6.4 SCADA oraz QNX Photon microGUI.

**Konsola** jest intuicyjnym interfejsem graficznym użytkownika oraz zestawem programów umożliwiającym użytkownikowi interakcję, np. z Data Protector. Zapewnia też komfortowe i efektywne użytkowanie aplikacji. Wyróżniamy dwa typy agentów: dyskowy i urządzeń. Funkcją agenta dyskowego jest odczytywanie (*backup*) lub zapisywanie (*restore*) danych w systemie. Agent urządzenia odpowiada za zapisywanie (*backup*) lub odczytywanie (*restore*) danych na urządzeniu zachowującym dane. Zarządzanie procesami wykonywania backupów w całym środowisku (komórce) i ich nadzorowanie realizuje z centralnego miejsca zarządca komórki. Centralnym jego elementem jest baza danych zawierająca informacje o konfiguracji wszystkich elementów środowiska tworzenia kopii zapasowych. Krytycznym elementem całej instalacji Data Protector jest menedżer komórki. Od dostępności menedżera komórki Data Protector zależy możliwość wykonania operacji zachowania/odtworzenia. Oprogramowanie Data Protector jest systemem otwartym w całym tego słowa znaczeniu, co oznacza, że jest ono przeznaczone do zarządzania środowiskiem składającym się z różnych systemów operacyjnych.

HP OpenView Storage Data Protector charakteryzuje się m.in.:

- architekturą klient/serwer, zapewniającą centralną kontrolę wykonywania kopii zapasowych rozproszonego, heterogenicznego systemu komputerowego;
- graficznym interfejsem użytkownika zintegrowanym z platformą zarządzania HP OpenView (dostępna jest także linia komend CLI);
- automatyzacją procesu tworzenia kopii zapasowych ściśle według harmonogramu;



- możliwością wykonywania różnych typów kopii (całościową, przyrostową, różnicową);
- efektywną wewnętrzną bazą danych z funkcją wykonywania kopii zapasowej;
- wysoką wydajnością osiąganą poprzez mechanizmy równoległego zapisu/odczytu na wiele urządzeń backupowych.

Inną formą konsoli, umożliwiającą wzrost efektywności zabezpieczenia informacji (problemu jakże ważnego dla przedsiębiorstw w związku z ostrą konkurencją), może być menedżer zabezpieczeń Oracle Enterprise. W interfejsie użytkownika konsoli do uzyskiwania dostępu do różnych stron tego narzędzia służą karty, czyli wyspecjalizowane mikrokomputery do wyświetlania. Problem bezpieczeństwa danych dla dużych firm urasta do rangi zagadnień kluczowych. Firmy, które muszą monitorować i kontrolować duże systemy, potrzebują stabilnego, bezpiecznego i potężnego systemu. Przykładem efektywnego wykorzystywania konsoli może być System SCADA RealFlex 6.4 SCADA (*Supervisory Control and Data Acquisition*). Systemy SCADA muszą pracować 24 godziny na dobę, 365 dni w roku, a to wymaga pełnej gotowości. W systemie SCADA najważniejsze jest bezpieczeństwo. Serwer RealFlex 6.4 działa na systemie operacyjnym QNX Neutrino i korzysta z dynamicznie kodowanego protokołu między Klientami i Serwerem, co zapewnia bezpieczną komunikację.

Za pomocą **konstruktora aplikacji** — graficznego narzędzia projektowania dla systemu z okienkami QNX Photon microGUI — można w prosty sposób tworzyć pełnowartościowe interfejsy użytkownika, wykorzystując tylko funkcję „wskaż i kliknij”. Nie trzeba pisać żadnego kodu — konstruktor aplikacji może automatycznie wygenerować w pełni funkcjonalny prototyp graficznego interfejsu użytkownika.

Jest on całkowicie zintegrowany ze środowiskiem programowania, dzięki czemu można wykorzystać istniejące kreatory do stworzenia dowolnego projektu aplikacji.

Za pomocą konstruktora aplikacji można:

- szybko przystąpić do pracy, używając istniejących szablonów lub tworzyć własny interfejs użytkownika, wykorzystując szeroką paletę widgetów;
- łączyć okna dialogowe i menu bezpośrednio do widgetów lub dodawać wywołania kodu źródłowego;
- edytować, kompilować, testować i debugować aplikacje graficzne za pomocą wielu zróżnicowanych narzędzi środowiska programowania;
- dynamicznie generować kopie utworzonych wcześniej widgetów w dowolnej liczbie wymaganej przez daną aplikację;
- tłumaczyć interfejs użytkownika na wiele języków bez konieczności ponownej kompilacji i komponowania kodu, używając tego samego modułu.

**Kreator aplikacji** oferuje możliwość korzystania z komponentów interfejsu użytkownika dla szerokiego spektrum aplikacji: od urządzeń kieszonkowych do sta-

cji roboczych. Komponenty bazowe platformy Eclipse to: przyciski, kontenery, wskaźniki, grafika wektorowa, obrazy, listy, drzewa, menu, skróty i tekst. Komponenty dodatkowe platformy Eclipse to: HTML, terminal, wybór czcionki, wybór drukarki, Flash, kontener wideo.

## 5.3. Hurtownia danych (HD) w SWD

Hurtownia danych (HD) to baza danych dedykowana analitycznym procesom przetwarzania danych w czasie rzeczywistym OLAP. Jej rolą jest gromadzenie danych pochodzących z długiego horyzontu czasowego, a następnie organizacja ich i udostępnianie w takiej formie i postaci, która gwarantuje szybką realizację złożonych zapytań użytkownika. Przygotowanie HD do pracy w warunkach SWD jest pod względem metodycznym i technicznym podobne do sytuacji, jaka zachodzi w przypadku łączenia SWD z istniejącą już transakcyjną BD. W sytuacji kiedy zachodzi potrzeba zastosowania różnych rozwiązań bazodanowych i przede wszystkim należy zadbać o technologiczną jednorodność środowiska, istnieje możliwość budowy platformy typu *middleware*.

### 5.3.1. Struktura i istota działania HD

Bazy danych stanowią centrum gromadzenia i przetwarzania danych, które są wytwarzane lub pozyskiwane w przedsiębiorstwie. Podstawowy podział systemów bazodanowych wyróżnia bazy transakcyjne i analityczne. **Bazy transakcyjne** wspierają bieżącą działalność przedsiębiorstwa. **Bazy analityczne** umożliwiają przetwarzanie i analizę danych, ekstrahowanych z poziomu transakcyjnego, w celu pozyskania z nich nowej, biznesowo użytecznej informacji, wspierającej proces podejmowania decyzji. Zgodnie z opinią SAS Institute, lidera na rynku oprogramowania Business Intelligence (BI), transakcyjne systemy bazodanowe „napędzają” działalność przedsiębiorstwa, a analityczne systemy bazodanowe „przyglądają się”, w jaki sposób ta działalność jest napędzana.

Po raz pierwszy pojęcie HD zostało wprowadzone w 1988 r. przez pracowników firmy IBM. B. Davlin i P. Murphy [Hayes 2002] przedstawili HD jako: **środowisko do wspierania użytkowników informacji biznesowej, zapewniające informatykom możliwość dbania o jakość danych**. Od tego czasu powstało wiele różnych definicji HD, m.in. dwóch największych ekspertów z tej dziedziny — R. Kimballa [Kimball 2004] oraz W.H. Inmona [Inmon 2005].

Hurtownia danych według R. Kimballa jest **systemem, który wydobywa, czyści, dostosowuje i dostarcza dane źródłowe do wymiarowych miejsc przechowywania danych, a następnie wspiera i implementuje zapytania i analizy w celach podejmowania decyzji**. Zdaniem W.H. Inmona **HD to tematyczna, zintegrowana, zmienna w czasie składnica nieulotnych danych, przeznaczona do wspierania**

**procesów podejmowania decyzji.** Ta ostatnia definicja nawiązuje do najważniejszych cech, którymi powinna się wyróżniać hurtownia danych.

Wiele projektów korporacyjnych HD, mimo zaangażowania dużych środków finansowych, nie udało się lub nie spełniło oczekiwań użytkowników. W roku 1994 wprowadzono pojęcie mini HD tematycznej lub składnicy danych (SD), czyli HD przeznaczonej do realizacji potrzeb analitycznych jednego działu przedsiębiorstwa lub jednego profilu jego działalności [Inmon 2005].

Podstawową różnicą między HD a SD jest zakres i rozmiar przechowywanych danych. HD obejmuje wiele zakresów tematycznych, zarządzanych z jednego centralnego miejsca organizacji. Zawiera dane pochodzące z różnych systemów źródłowych, które są do niej ekstrahowane oraz w niej integrowane. Nie można natomiast w prosty sposób zdefiniować wielkości lub złożoności SD, ponieważ wynika ona z jej indywidualnej funkcjonalności. SD są mniejsze i mniej złożone niż korporacyjna HD, ponieważ są ukierunkowane na określone zastosowanie tematyczne (nie obejmują całej działalności przedsiębiorstwa). Często budowa SD stanowi etap poprzedzający utworzenie korporacyjnej HD, która integruje działania poszczególnych składnic. Taka przyrostowa budowa jest bardzo korzystna dla przedsiębiorstwa, ponieważ:

- realizacja każdego nowego przyrostu jest relatywnie krótka (trwa od kilku miesięcy do jednego roku);
- użytkownik może szybko korzystać oraz ocenić efekt wdrożenia;
- użytkownik i wykonawca uczą się przy każdym realizowanym przyroście;
- często kolejne działy przedsiębiorstwa zaczynają się interesować projektem, który został wdrożony i jest efektywnie wykorzystywany w innym.

Według R. Kimballa wśród SD można wyróżnić dwa typy [Kimball 2004]:

- 1) szczegółowe SD — przechowują dane w strukturach wielowymiarowych na najwyższym poziomie szczegółowości (dane elementarne) lub dane lekko zagregowane, co ma na celu przyspieszenie odpowiedzi na zapytania użytkowników;
- 2) zagregowane SD — przechowują dane zagregowane według rdzennych procesów biznesowych.

Bez względu jednak na rozmiar wdrożonej HD głównym celem ich zastosowania jest możliwość realizacji analiz z obszaru: sprzedaży, finansów, logistyki, produkcji czy zarządzania zasobami ludzkimi. Dzięki szybkiej analizie bazującej na pełnej i aktualnej informacji o stanie firmy kadra zarządzająca może podejmować trafniejsze decyzje o strategicznym znaczeniu dla rozwoju danego przedsiębiorstwa. HD zapewniają zatem wiele korzyści, takich jak:

- oddzielenie danych analitycznych od operacyjnych, czyli warstwy decyzyjnej od wykonawczej;
- stworzenie podstawy systemów wspomagania podejmowania decyzji;

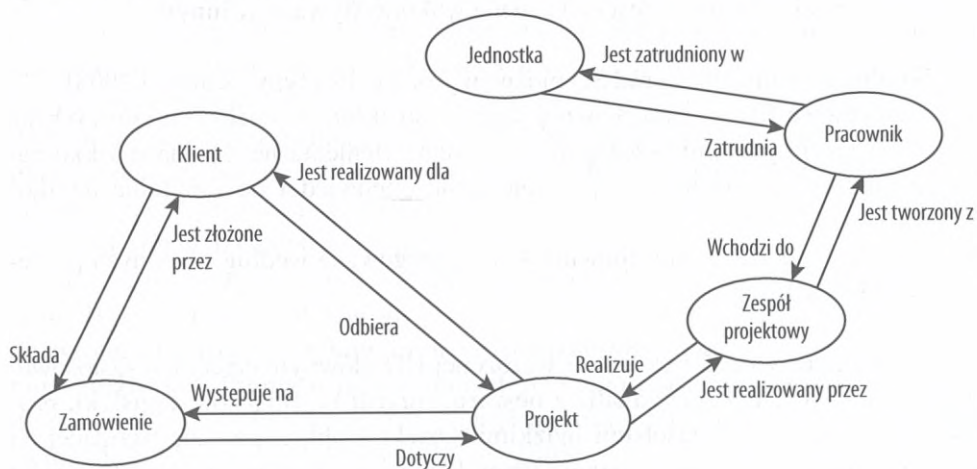
- uporządkowanie danych w przejrzysty, hierarchiczny sposób;
- udostępnienie danych na różnych poziomach agregacji, co nie zmusza do wykonywania pracochłonnych zapytań;
- umożliwienie pełnego wykorzystania danych.

### 5.3.2. Modelowanie logiczne i fizyczne HD

Projekty baz analitycznych, podobnie jak projekty baz transakcyjnych, powstają w oparciu o trzyetapową metodykę tworzenia pojęciowego, logicznego i fizycznego modelu danych.

Pierwszym etapem projektowania HD jest utworzenie **modelu pojęciowego** [Rostek 2010b]. Zadaniem modelu pojęciowego jest pokazanie istotnych zależności między realizowanymi procesami biznesowymi i obiektami biznesowymi, które następnie muszą zostać uwzględnione podczas tworzenia modelu logicznego. Model pojęciowy prezentuje zakres tematyczny HD w ujęciu biznesowym, nie technologicznym, i stanowi element łączący perspektywę biznesu i perspektywę technologii w projekcie HD. Model pojęciowy może powstać wyłącznie przy aktywnym współuczestnictwie użytkownika projektowanego systemu. Elementami modelu pojęciowego są obszary lub obiekty, na których temat gromadzone są informacje. Mają one zazwyczaj postać rzeczowników, np. klient, produkt, pracownik, departament, transakcja, zamówienie (rysunek 5.7).

Rysunek 5.7. Model pojęciowy hurtowni danych



Źródło: Opracowanie własne.

Następnym etapem projektowania jest uszczegółowienie każdego elementu modelu pojęciowego w modelu logicznym. Elementem modelu pojęciowego uszczegółowianym na rysunku 5.8 będzie Projekt.

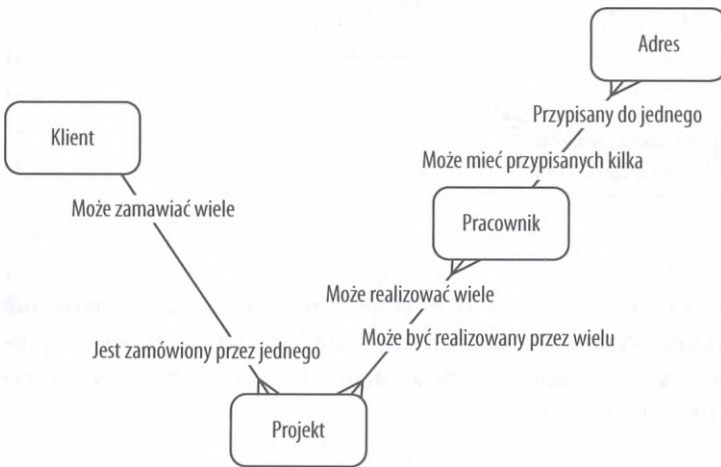
**Model logiczny** jest tworzony na wyższym poziomie szczegółowości jak model pojęciowy i korzysta z notacji, która operuje następującymi pojęciami:

- encje — fragmenty rzeczywistości, które powinien uwzględniać model logiczny;
- atrybuty — cechy szczegółowe, opisujące poszczególne encje;
- zależności — związki występujące między poszczególnymi encjami w modelu logicznym.

Budowa modelu logicznego przebiega w dwóch etapach.

- 1) tworzenie modelu wyższego poziomu, obejmującego:
  - zdefiniowanie encji w oparciu o listę elementów modelu pojęciowego,
  - odnalezienie i zdefiniowanie zależności między encjami;
- 2) tworzenie modelu niższego poziomu, obejmującego:
  - rozłożenie relacji typu  $n-m$ ,
  - zdefiniowanie atrybutów poszczególnych encji,
  - ustalenie kluczy głównych (podstawowych) dla każdej encji,
  - normalizację danych.

**Rysunek 5.8. Model logiczny wyższego poziomu dla elementu modelu pojęciowego — Projektu**



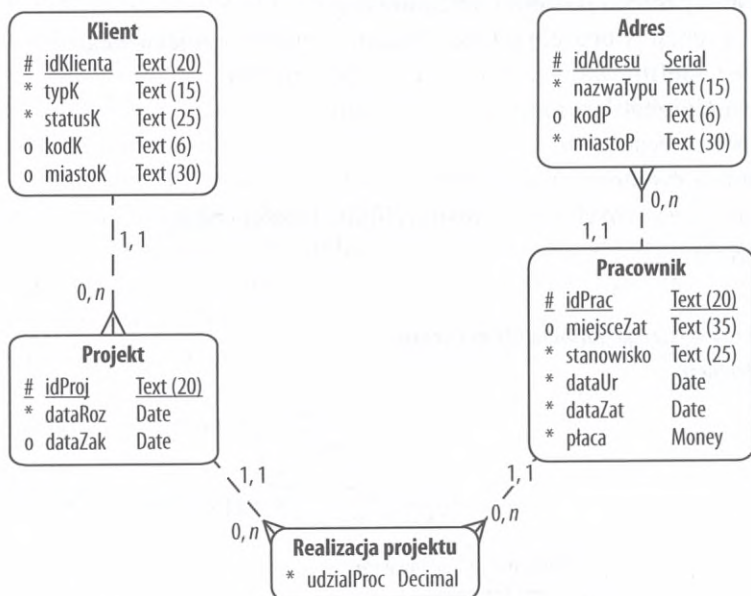
Źródło: Opracowanie własne.

Zadaniem modelu wyższego poziomu jest identyfikacja encji oraz zależności występujących między nimi i przedstawienie ich w formie zrozumiałej dla kadry zarządzającej (rysunek 5.8). Umożliwia to wychwycenie i korektę błędów zanim zostanie zdefiniowana ostateczna postać modelu danych HD.

Akceptacja modelu wyższego poziomu pozwala na uszczegółowienie go do modelu niższego poziomu (rysunek 5.9)

**Model fizyczny** jest tworzony na podstawie modelu logicznego niższego poziomu, przy rozszerzeniu go jedynie o dodatkowe klucze i fizyczne właściwości modelu [Inmon 2005, s. 98]. Wyróżniamy kilka rodzajów struktur fizycznych, które mogą przechowywać dane w hurtowni: modele znormalizowane, struktury ROLAP, bazy wielowymiarowe MOLAP, tabele zagregowane, połączone widoki i tabele.

**Rysunek 5.9. Model logiczny niższego poziomu dla elementu modelu pojęciowego — Projektu**



Źródło: Opracowanie własne.

W modelu fizycznym można skorzystać z jednej z wymienionych struktur lub stworzyć kompilacje hybrydowe. Każdy z modeli ma swoje wady i zalety, które prowadzą się do sprostania wymaganiom raportowym użytkowników, kosztom utrzymania i uaktualniania danych.

### 5.3.3. Przetwarzanie analityczne w HD

**Hurtownie danych**, jako element SWD, zwiększają efektywność realizacji procesu podejmowania decyzji, dzięki ograniczeniu trzech typów opóźnień wynikających z konieczności [Hackathorn 2004]: przygotowania danych do analizy, przygotowania analizy, dostarczenia wyników analizy. Co prawda, samodzielnie HD nie posiadają aż tak szerokiego zakresu funkcjonalności, ale mogą ją dostarczyć za pośrednictwem systemu analitycznego przetwarzania danych, opartego na technologii OLAP lub/i eksploracji danych.

Podstawowym typem analizy prowadzonej w hurtowniach danych jest **przetwarzanie analityczne na bieżąco**. Termin ten został wprowadzony w 1993 r. przez E.F. Codd [Codd, Codd, Smalley 1993]. Stwierdził on, że dane pochodzące z systemów transakcyjnych nie są wystarczające do udzielania kompletnych odpowiedzi na pytania osób zarządzających. Podstawową ideą OLAP [Codd, Codd, Smalley 1993] jest umożliwienie użytkownikom manipulowania modelami danych poprzez wiele wymiarów w celu rozumienia zmian, które w nich występują.

**OLAP** stanowi kategorię oprogramowania, która umożliwia analitykom i menedżerom swobodny dostęp do informacji poprzez szybki, interaktywny i szeroki wybór widoków, zawierających odpowiednio transformowane dane oraz wyniki analiz. Widoki te odzwierciedlają wielowymiarowość przedsiębiorstwa w taki sposób, w jaki postrzegają i rozumieją ją użytkownicy. Mają oni możliwość zdefiniowania własnej analizy w oparciu o dostępne dane, utworzenia lub modyfikacji wymiarów analizy, a także zbudowania agregatów przy wykorzystaniu funkcji analizy statystycznej.

**Eksploracja danych** umożliwia analizę danych w odniesieniu do tych problemów, które ze względu na swój rozmiar są trudne do rozwiązania przez człowieka lub w odniesieniu do których nie dysponujemy pełną wiedzą — chcemy ją dopiero pozyskać z danych. Metody eksploracji danych polegają na efektywnym znajdowaniu nieznanych dotychczas zależności i związków między danymi. Automatyczna eksploracja danych otwiera nowe szanse w zakresie interakcji użytkownika z systemem hurtowni danych, umożliwiając formułowanie zapytań na znacznie wyższym poziomie abstrakcji, niż pozwalała na to technologia OLAP.

Według D.J. Handa [Hand 2001] eksploracja danych to „nauka zajmująca się wydobywaniem informacji z dużych zbiorów danych lub BD”. M.J. Berry i G.S. Linoff twierdzą, że za pomocą tego terminu można zdefiniować [Berry, Linoff 2004] „przeszukiwanie oraz analizę dużych ilości danych w celu znalezienia znaczących zależności, schematów, wzorców i zasad”. Ważną cechą eksploracji danych jest możliwość jej realizacji w dwojaki sposób:

- 1) od początku znany jest główny cel badań eksploracyjnych, np. poszukiwany jest zbiór czynników zwiększających ryzyko odejścia pracownika do konkurencji;
- 2) główny cel badań nie jest określony na początku procesu, np. poszukiwane są wszystkie nowe, nietrywialne i użyteczne zależności oraz wzorce ukryte w zbiorze wytypowanym do badania.

Duże znaczenie ma również fakt, że eksploracja danych, w odróżnieniu od zwykłych analiz statystycznych (zazwyczaj jednowymiarowych), uwzględnia jednoczesny wpływ wielu czynników na badane zjawisko. Dzięki temu umożliwia wyodrębnienie sekwencji cech, często połączonych wzajemnie hierarchiami, których wystąpienie w populacji podwyższa lub obniża prawdopodobieństwo zaistnienia konkretnego zjawiska.

Eksploracja danych jest naturalnym rozwinięciem i dopełnieniem tworzenia hurtowni, czyli organizowania dużych, wielowymiarowych zbiorów danych w sposób ułatwiający pozyskiwanie informacji dla celów analitycznych. Techniki eksploracji danych mogą operować na dowolnym typie nieprzetworzonej informacji, mogą być również stosowane do przeglądania i zestawiania danych generowanych za pomocą zapytań OLAP i w tym przypadku dostarczają bardziej dogłębnej, często wieloaspektowej wiedzy. Eksploracja danych jest podejściem analitycznym, będącym rozszerzeniem technik OLAP.

Eksploracja danych z założenia korzysta ze wszelkich metod pozwalających z danych tworzyć nową i potencjalnie użyteczną wiedzę, i w związku z tym nie ma ustalonego kanonu technik. Istnieje jednak dosyć duży zbiór algorytmów, które szczególnie często są wykorzystywane w badaniach eksploracyjnych. Należą do nich: analiza asocjacji, regresja, drzewa decyzyjne, sieci neuronowe, analiza szeregów czasowych czy sieci Bayesa. Dobór właściwej techniki do drażenia danych zależy oczywiście od typu rozpatrywanego problemu analitycznego.

### **5.3.4. Funkcjonalne i eksploatacyjne aspekty projektowania HD na potrzeby SWD**

Według statystyk wdrożenie HD w okresie pierwszych trzech lat użytkowania przynosi średnio zwrot inwestycji przekraczający 400%. Jednak skrajne wartości zwrotu inwestycji w badanych projektach osiągają wartości +16 000% w najbardziej udanych oraz -2000% w projektach zakończonych niepowodzeniem. Tak wielki rozrzut świadczy o tym, że systemy HD łączą w sobie ogromny potencjał korzyści z niemałym ryzykiem. Zatem istotne są elementy, które pozwalają obniżyć ryzyko inwestycyjne projektu, a zaliczane są do nich: strategiczny kontekst analiz, ścisła współpraca z użytkownikiem podczas projektowania i budowy systemu, iteracyjna budowa systemu, szczegółowa definicja architektury oraz efektywna metodyka wykorzystana na etapie projektowania, budowy i wdrożenia.

Systemy SWD oparte na zasobach HD są dedykowane kadrze menedżerskiej oraz analitykom biznesowym, a głównym kryterium ich użyteczności jest wspomaganie procesu biznesowego. Strategiczny kontekst analiz zabezpiecza te potrzeby, dostarczając w szybki sposób informacje, które są krytyczne dla podjęcia właściwych decyzji gospodarczych i niemożliwe do łatwego uzyskania bez HD. Istotne jest, aby każdy projekt HD rozpocząć od zidentyfikowania zastosowań i analiz, którym będzie dedykowany.

Żaden projekt informatyczny, a projekt HD w szczególności, nie może powstawać bez ścisłej współpracy z użytkownikiem. Pozwala ona uzyskać informacje związane z biznesową architekturą projektowanego systemu, takie jak: specyfikacja zakresu wspieranych strategii i celów biznesowych czy miary i wskaźniki ekonomiczne, które powinien śledzić. Umożliwia ona także ocenę przygotowywanego rozwiązania w fazie prototypowej i ogranicza niechęć pracowników do wykorzystywa-



nia nowego i nieznanego narzędzia, która przy dużej eskalacji problemu może doprowadzić do całkowitego fiaska wdrożenia.

Iteracyjna budowa systemu umożliwia ograniczenie bardzo istotnego czynnika ryzyka wdrażania projektu — złożoności budowy HD. Wprowadza także element uczenia się, zarówno zespołu projektowo-wykonawczego, jak i przede wszystkim jego przyszłych i potencjalnie nowych użytkowników, zwiększając świadomość ich potrzeb i oczekiwań w stosunku do przygotowywanego rozwiązania.

Architektura rozwiązania opartego na HD jest kluczowym elementem projektu, ponieważ musi zapewnić realizację podstawowych procesów hurtowni:

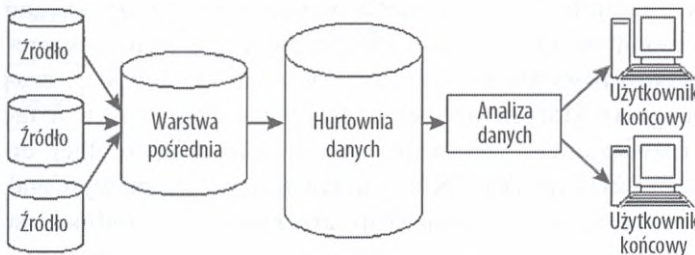
- ekstrakcji danych źródłowych;
- czyszczenia, integracji, ekstrakcji i transformacji danych w celu przygotowania ich do umieszczenia w strukturach HD;
- magazynowania i zarządzania danymi w bazie HD;
- udostępniania danych użytkownikom za pośrednictwem zapytań analitycznych i raportów — zarówno gotowych, jak i przygotowywanych *ad hoc*.

Rozwiązanie, które zostanie przyjęte, może mieć jednak różną architekturę, zgodną z potrzebami funkcjonalnymi i eksploatacyjnymi jego użytkowników. Cztery najczęściej spotykane rozwiązania architektoniczne to [Inmon 2005]:

- 1) monolityczna korporacyjna HD (ang. *Monolithic Enterprise Data Warehouse*),
- 2) niezależne składnice danych,
- 3) Bus Architecture (podejście propagowane przez R. Kimballa),
- 4) Corporate Information Factory (rozwiązanie proponowane przez B. Inmona).

Podstawową architekturę systemu opartego na monolitycznej, korporacyjnej HD prezentuje rysunek 5.10.

**Rysunek 5.10. Architektura monolitycznej korporacyjnej hurtowni danych**



Źródło: Opracowanie własne.

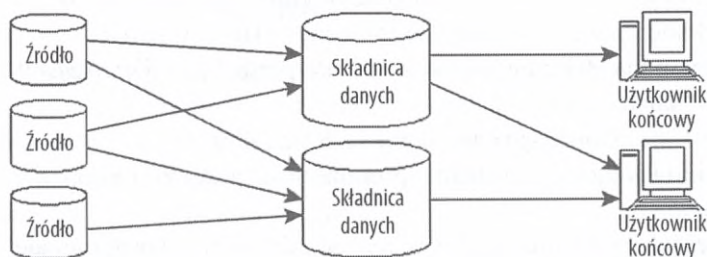
Warstwa pośrednia w hurtowni pełni funkcję „przedsionka” BD, do którego dane są pozyskiwane w wyniku ekstrakcji ze źródeł operacyjnych, a następnie przygotowywane do wprowadzenia do struktur hurtowni — czyszczone, integrowane

i transformowane. Dopiero sprawdzone i przygotowane dane są ładowane z warstwy pośredniej do bazy HD. Dane z hurtowni są udostępniane użytkownikom za pomocą przygotowanych dla nich interfejsów.

Hurtownia korporacyjna z założenia powinna obsługiwać wszystkie aspekty działalności przedsiębiorstwa. Musi być zatem projektem czasochłonnym i kosztownym, obciążonym wysokim ryzykiem niepowodzenia. Alternatywą takiego podejścia jest zastosowanie iteracyjnej budowy systemu (podział projektu na mniejsze etapy) lub wykorzystanie innej koncepcji architektury systemu, np. niezależnych SD.

Architekturę systemu opartego na niezależnych SD prezentuje rysunek 5.11. Każda SD jest niezależnym projektem, który trwa od kilku miesięcy do jednego roku. Doświadczenia nabyte podczas wdrożenia jednej składnicy można wykorzystać przy tworzeniu kolejnych. Skracą się również czas oraz maleją koszty kolejnych projektów, a rośnie świadomość korzyści oraz naturalnie kreuje się potrzeba wśród użytkowników.

Rysunek 5.11. Architektura niezależnych składnic danych



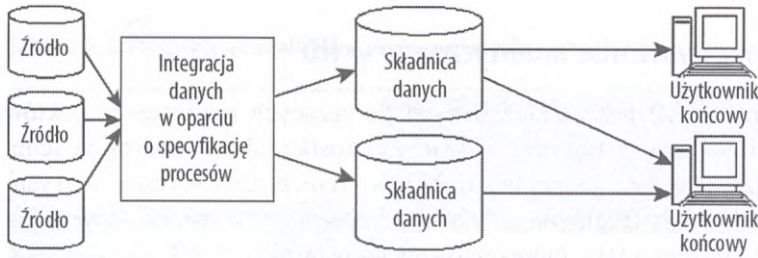
Źródło: Opracowanie własne.

Rozwiązanie to jednak nie jest pozbawione wad. Przede wszystkim nie przewiduje ono wspólnej warstwy centralnej — integrującej wszystkie dane pozyskiwane w wyniku ekstrakcji z poziomu operacyjnego. Integracja dotyczy jedynie poszczególnych SD. Poza tym użytkownicy zaczynający korzystać z wdrożonych SD, z czasem formułują nowe wymagania, których nie spełnia dotychczasowy projekt. W takiej sytuacji jedynym rozwiązaniem jest wdrożenie kolejnej SD, zdolnej do realizacji nowych wymagań użytkowników. Kiedy pojawią się następne wymagania, proces się powtórzy, i w ten sposób powstaje struktura, którą coraz trudniej jest zarządzać.

Architektura niezależnych SD jest przystosowana do obsługi kolejnych jednostek administracyjnych przedsiębiorstwa. R. Kimball [Kimball 2004] proponuje alternatywne rozwiązanie oparte na zapewnieniu obsługi procesów biznesowych przedsiębiorstwa (rysunek 5.12). Rozwiązanie to również zapewnia możliwość realizacji projektu krótkimi, zamkniętymi etapami, a ponieważ kolejne składnice budo-

wane są dla procesów, a nie dla jednostek organizacyjnych, więc precyzyjniej można zdefiniować niezbędną funkcjonalność każdej ze składnic.

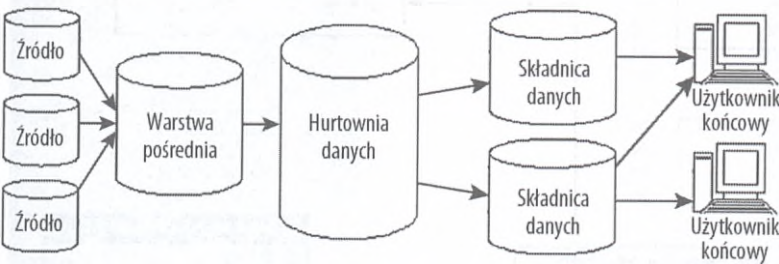
**Rysunek 5.12. Architektura Kimball's BUS Architecture**



Źródło: Opracowanie własne na podstawie: [Kimball 2004].

W propozycji R. Kimballa nie ma korporacyjnej hurtowni danych, która integrowałaby i centralnie udostępniała wszystkie dane analityczne. Natomiast centralną hurtownię danych wprowadził do swojej koncepcji B. Inmon (rysunek 5.13). W architekturze CIF hurtownia danych stanowi centralną warstwę integrującą dane. HD jest w tym rozwiązaniu znormalizowaną, relacyjną BD. SD przechowują natomiast dane zagregowane w strukturach wielowymiarowych, przystosowanych do prowadzenia na nich bezpośredniej analizy danych.

**Rysunek 5.13. Architektura CIF B. Inmona**



Źródło: Opracowanie własne na podstawie: [Inmon 2005].

**Dobranie właściwej metodyki wdrożenia** jest możliwe po ustaleniu architektury i technologii projektowanego rozwiązania. Każdy z producentów narzędzi informatycznych, dedykowanych tworzeniu HD, proponuje autorską metodykę realizacji wdrożenia. Są również określone zalecenia metodyczne związane z danym typem architektury. Ważne jest, żeby przyjąć metodykę, która sprawdziła się w innych wdrożeniach, najlepiej o podobnym zakresie funkcjonalnym i eksploatacyjnym oraz

żeby w miarę możliwości korzystać ze wsparcia organizacji posiadających doświadczenie w tym zakresie.

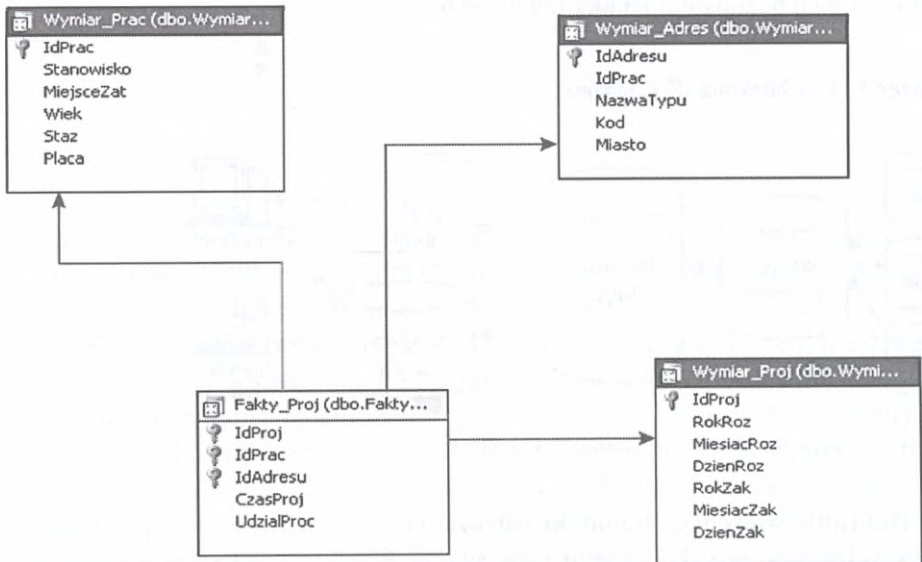
Perspektywa realnych korzyści, uzyskiwanych przez firmę efektywnie eksploatującą HD, znacząco przerasta ewentualne ryzyko, które musi podjąć w fazie jego wdrożenia.

### 5.3.5. Przykład przetwarzania analitycznego w HD

Głównym zadaniem OLAP jest analiza danych dotyczących realizacji projektów w pewnej firmie projektowo-usługowej. Utworzono kostkę OLAP, opartą na schemacie gwiazdy, składającą się z jednej tabeli faktów i trzech tabel wymiarów (rysunek 5.14). Schemat ten jest fragmentem modelu fizycznego, wygenerowanego na podstawie modelu logicznego HD, zaprezentowanego w punkcie 5.3.2.

Tabela faktów zawiera dane (CzasProj, UdzialProc), na których podstawie zostaną obliczone miary analityczne (LiczbaProjektow, CzasProjektow, UdzialPracownikow). Tabele wymiarów zawierają dane (np. MiejsceZat, Stanowisko, RokZak, MiesiacZak), które będą stanowiły przekroje analityczne. Tak zdefiniowana struktura pozwoli stwierdzić, od jakich czynników zależy liczba zrealizowanych projektów, czas ich wykonania oraz poziom zaangażowania pracowników.

Rysunek 5.14. Definicja wielowymiarowej kostki OLAP



Źródło: [Rostek 2010a].

Utworzona kostka OLAP pozwala na sporządzanie perspektyw analitycznych (tablica 5.3), gdzie przekrojami analitycznymi są wymiary kostki (np. RokZak, Mi-

siacZak, DzieńZak, MiejsceZat, Stanowisko, Staz), a wielkościami analizowanymi jej miary (np. LiczbaProjektow). Perspektywa prezentowana w tabelicy 5.3 umożliwia zbadanie zależności między liczbą zrealizowanych projektów a datą zakończenia projektu (w podziale na rok, miesiąc i dzień zakończenia) oraz realizatorami projektu (w podziale na jednostkę zatrudnienia, zajmowane stanowisko i staż pracy).

**Tablica 5.3. Perspektywa analityczna wielowymiarowej kostki OLAP**

			Rok Zak ▾	Miesiąc Zak ▾	Dzień Zak ▾	
			☒ 1996	☒ 1997	☒ 1998	Suma końcowa
Miejsce Zat ▾	Stanowisko ▾	Staz ▾	Liczba Projektow	Liczba Projektow	Liczba Projektow	Liczba Projektow
☒ 100			170	7	457	634
☒ 110			30		328	358
☒ 120			191	4	467	662
Suma końcowa			391	11	1252	1654

Źródło: [Rostek 2010a].

Przeprowadzoną analizę można uszczegółowić, rozwijając wymiary: MiejsceZat i Stanowisko (tablica 5.4).

**Tablica 5.4. Rozwijanie wymiarów danych**

			Rok Zak ▾	Miesiąc Zak ▾	Dzień Zak ▾	
			☒ 1996	☒ 1997	☒ 1998	Suma końcowa
Miejsce Zat ▾	Stanowisko ▾	Staz ▾	Liczba Projektow	Liczba Projektow	Liczba Projektow	Liczba Projektow
☒ 100	☒ analityk		60		181	241
	☒ kierownik		18	1	31	50
	☒ projektant		68	6	144	218
	☒ sekretarka		24		101	125
	Suma		170	7	457	634
☒ 110	☒ analityk				69	69
	☒ kierownik				64	64
	☒ programista				101	101
	☒ projektant				84	84
	☒ sekretarka		30		10	40
Suma		30		328	358	
☒ 120	☒ analityk	5			141	141
		11	18	1	9	28
		16	19		37	56
		Suma	37	1	187	225
	☒ grafik				52	52
	☒ kierownik		8		4	12
	☒ programista		130	3	137	270
	☒ projektant		16		8	24
	☒ sekretarka				79	79
	Suma		191	4	467	662
Suma końcowa			391	11	1252	1654

Źródło: Jak tablicy 5.3.

Rozwinięcie wymiaru MiejsceZat powoduje wyświetlenie liczby projektów realizowanych przez osoby zajmujące poszczególne stanowiska w danym miejscu zatrudnienia (tablica 5.4). Rozwinięcie wymiaru Stanowisko pozwala na pokazanie liczby projektów wykonywanych przez osoby o określonym stażu zatrudnienia na danym stanowisku (tablica 5.4). Uzyskane wyniki analizy OLAP mogą być wykorzystywane bezpośrednio w procesie zarządzania przedsiębiorstwem. Jeżeli natomiast dane, poddane analizie OLAP, okazałyby się interesujące i trudne do jednoznacznej interpretacji, analiza ta może być uzupełniona metodami eksploracyjnymi.

Wśród ponad 450 osób zatrudnionych w firmie tylko niespełna 60 bierze udział w realizacji projektów. Zarząd firmy chce aktywować swoich pracowników (ewentualnie doprowadzić do zmian w strukturze zatrudnienia), ale chce też wiedzieć, na jakie cechy swoich obecnych lub przyszłych pracowników powinien zwracać szczególną uwagę. W tym celu wszystkim pracownikom, których dane przechowywane są w hurtowni, przydzielono wartość dodatkowej zmiennej o nazwie Efektywność. Dla pracowników, którzy w trakcie pracy w firmie wzięli udział w realizacji co najmniej 10 projektów i przynajmniej w jeden zaangażowali się na poziomie udziału 30%, zmienna ta przyjmuje wartość 1. Dla wszystkich pozostałych pracowników wartość zmiennej Efektywność wynosi 0. Tak utworzona zmienna objaśniana pozwala na przeprowadzenie analizy klasyfikacyjnej, której celem będzie opisanie pracownika efektywnego i zaangażowanego w projektową działalność firmy.

Rysunek 5.15. Definicja modeli eksploracyjnych

Structure	Drzewo Decyzyjne	Regresja	Sieć Neuronowa
	Microsoft_Decision_Trees	Microsoft_Logistic_Regression	Microsoft_Neural_Network
BHP	Input	Input	Input
Efektywnosc	PredictOnly	PredictOnly	PredictOnly
Id Prac	Key	Key	Key
Liczba Dzieci	Input	Input	Input
Miejsce Zat	Input	Input	Input
Placa	Input	Input	Input
Pozyczka	Input	Input	Input
Stanowisko	Input	Input	Input
Staz	Input	Input	Input
Wiek	Input	Input	Input

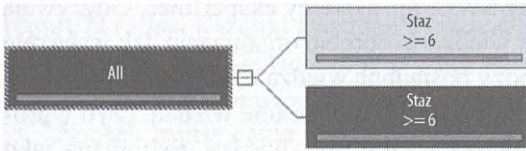
Źródło: Jak rysunku 5.14.

Do przeprowadzenia analizy zostały wytypowane trzy techniki eksploracyjne: drzewa decyzyjne, model regresji logistycznej oraz sieci neuronowe. Przed rozpoczęciem analizy trzeba wyspecyfikować zmienne, które będą wprowadzone do każ-

dego modelu (rysunek 5.15). Efektywność, oznaczona w modelach jako PredictOnly, to zmienna objaśniana. Zmienne oznaczone w modelu jako Input mają charakter zmiennych objaśniających. Analiza eksploracyjna polega na tym, aby wykazać te wartości cech zmiennych objaśniających, które mają największy wpływ na wartość (1 lub 0) zmiennej objaśnianej.

Uruchomienie modelu spowoduje wyświetlenie jego wyników, które mogą mieć postać graficzną (wyniki drzewa decyzyjnego, rysunek 5.16) lub opisową (wyniki regresji, rysunek 5.17).

Rysunek 5.16. Wyniki modelu drzew decyzyjnych



Źródło: Jak rysunku 5.14.

Rysunek 5.17. Wyniki modelu regresji logistycznej

Variables:			
Attribute	Value	Favors 0	Favors 1
Liczba Dzieci	4		
Placa	5000		
Placa	10000		
Pozyczka	20000		
Staz	6,315 - 10,257		
Placa	7000		
Placa	7500		
Liczba Dzieci	3		
Wiek	34,008 - 45,780		

Źródło: Jak rysunku 5.14.

Analizując wyniki modeli, można opracować następujące profile:

- pracownika efektywnego — jest to osoba w średnim wieku, mająca liczną rodzinę, z doświadczeniem zawodowym, dobrze zarabiająca i związana z firmą przez zaciągnięcie wysokiej pożyczki;
- pracownika nieefektywnego — jest to osoba młoda, średnio zarabiająca, krótko pracująca w firmie (szczególnie nieefektywne jest stanowisko grafika).

Interpretacja tych wyników może spowodować konieczność przeprowadzenia kolejnej analizy, która np. odpowie na pytanie, dlaczego graficy niechętnie angażują się w realizację projektów firmowych. Może charakter tych zleceń nie pozwala im

na właściwą aktywność, a może zbyt niska pensja zmusza ich do pracy również poza firmą, co w konsekwencji wiąże się z ograniczeniami czasowymi i niską efektywnością tej grupy pracowników. W ten właśnie sposób przetwarzanie eksploracyjne danych odkrywa nieznane dotąd informacje, które właściwie zinterpretowane powiększają wiedzę o przedsiębiorstwie, jego działalności, klientach, pracownikach, zagrożeniach ze strony konkurencji itd.

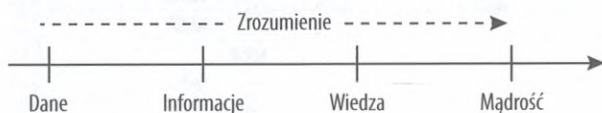
## 5.4. Baza wiedzy (BW) i moduł wnioskujący w SWD

**Bazy wiedzy** (ang. *knowledge base*) są magazynami zasobów wiedzy gromadzonymi i udostępnianymi przez niektóre typy SWD, np. systemy ekspertowe. Odgrywają one istotną rolę w procesie zarządzania wiedzą w obrębie organizacji. Jak wskazuje praktyka, istnieje duża rozbieżność między posiadaną wiedzą a umiejętnością jej zastosowania [Suttonm 2002]. Dlatego konieczne jest zarządzanie wiedzą, czyli wprowadzenie takich działań wewnątrz organizacji, aby luka wiedzy, rozumiana jako różnica między posiadanymi a potrzebnymi w procesie podejmowania decyzji zasobami wiedzy, była jak najmniejsza [Kisielnicki 2004]. Jedną z metod osiągnięcia tego celu jest wdrożenie systemu z bazą wiedzy i modułem wnioskującym.

### 5.4.1. Definicja i istota działania BW

Termin „wiedza” jest w literaturze interpretowany w różnorodny sposób i stosowany zamiennie z pojęciem informacji [Nonaka,Takeuchi 2000]. Tymczasem są to pojęcia rozdzielne, chociaż wzajemnie zależne, co prezentuje rysunek 5.18.

Rysunek 5.18. Dane–informacje–wiedza–mądrość



Źródło: [Davenport, Prusak 1998].

**Dane** to znaki i symbole, które mogą być przetwarzane umysłowo lub komputerowo [Gadomski 1999]. Stanowią też źródło **informacji**, która powstaje w wyniku ich przetworzenia. Warto jednak zwrócić uwagę, że w zależności od indywidualnego systemu i metod przetwarzania te same dane będą źródłem różnych informacji. Na przykład, wartość przebiegu kilometrów samochodu, należącego do taboru przedsiębiorstwa, dla jego kierowcy jest podstawą do obliczenia należnej mu wypłaty, ale dla mechanika z działu technicznego jest sygnałem konieczności wykonania przeglądu technicznego. Z kolei **wiedza** to ogół wiarygodnych informacji o rzeczywistości wraz z umiejętnością ich wykorzystywania [Encyklopedia PWN 2012]. Sama wiedza nie wystarczy, żeby podejmować na jej podstawie właściwe decyzje. Niezbędna



jest **mądrość**, która stanowi zdolność praktycznego wykorzystania posiadanej wiedzy i podejmowania uzasadnionych decyzji [Krzywka 2005]. Z przedstawionych rozważań wynika znaczenie wiedzy w procesie podejmowania decyzji. Dlatego ważny jest problem gromadzenia tej wiedzy w repozytorium, zwanym bazą wiedzy. Baza wiedzy to zbiór powiązanych logicznie faktów, dotyczących określonej dyscypliny naukowej, technicznej, społecznej czy organizacyjnej, przechowywanych z regułami logicznymi sformułowanymi przez ekspertów, umożliwiającymi rozwiązywanie problemów z zakresu, którego dotyczą.

**Fakty** są to prawdziwe stwierdzenia, określające pewną sytuację rzeczywistości [Kwiatkowska 2007, s. 43]. Najczęściej przyjmują formę zdań oznajmujących, opisujących właściwość podmiotu zdania. Przykładowo: *Maszyna jest niesprawna*. Z kolei reguły określają warunki zaistnienia poszczególnych faktów. Są to zdania podrzędnie złożone, które dają się zapisać w postaci warunku. Przykładowo: *Jeżeli maszyna została włączona, jest napięcie w sieci i silnik nie pracuje, to maszyna jest niesprawna*.

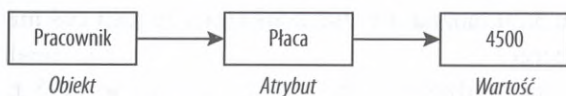
Sposób zapisywania faktów i reguł w bazie wiedzy nazywamy metodą **reprezentacji wiedzy**. Ma ona za zadanie [Krzywka 2005, s. 43]:

- nadać wiedzy formę, która będzie możliwa do zarejestrowania w bazie wiedzy,
- zachować poprawność wiedzy pozyskanej od eksperta,
- ustalić strukturę bazy wiedzy w taki sposób, aby dodawanie, aktualizacja i usuwanie z niej reguł i faktów było możliwie proste.

Do najczęściej wykorzystywanych metod reprezentacji wiedzy można zaliczyć [Krzywka 2005, s. 43–47]: stwierdzenia, sieci semantyczne, ramki i reguły.

**Stwierdzenia**, czyli trójka OAW: *Objekt–Atrybut–Wartość* (ang. *Object–Attribute–Value* — OAV), są metodą reprezentacji faktów (rysunek 5.19). Interpretacja trójki OAW oznacza stwierdzenie, że obiekt *O* posiada atrybut *A* o wartości *V*; obiektem jest konkretny (rzecz) lub abstrakt (pojęcie), a atrybut to cecha obiektu wyróżniająca go spośród innych obiektów.

Rysunek 5.19. Przykład trójki *Objekt–Atrybut–Wartość*



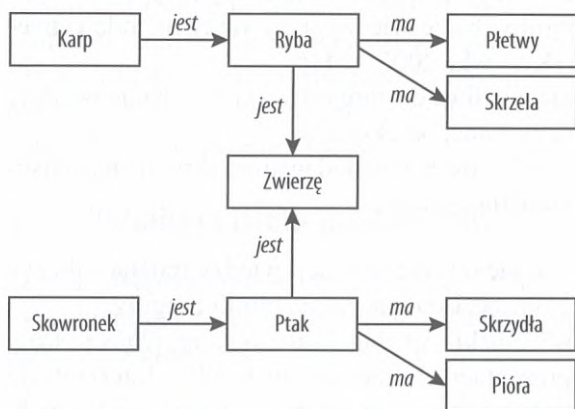
Źródło: Opracowanie własne.

W celu umożliwienia budowania hipotez i przypuszczeń stwierdzeniu jest przypisany współczynnik określający stopień pewności (ang. *Certainty Factor* — CF). Zazwyczaj jest to liczba z przedziału  $(-1, 1)$ . Współczynnik pewności równy 1 oznacza, że stwierdzenie jest w pełni prawdziwe. Gdy współczynnik pewności wynosi  $-1$ , oznacza to, że stwierdzenie jest w pełni fałszywe. Natomiast dla współczynnika pewności równego 0 nie można potwierdzić ani prawdziwości, ani fałszywości stwierdzenia.

**Sieci semantyczne** (ang. *semantic nets*) są najstarszym i najbardziej ogólnym typem reprezentacji wiedzy. Punktem wyjścia do powstania sieci semantycznej stała się w 1966 r. praca doktorska R.M. Quilliana [Quillian 1968], której celem było stworzenie modelu pamięci ludzkiej. Okazało się, że model w postaci sieci semantycznej może być wykorzystywany również do reprezentacji wiedzy.

Posługując się pojęciem bazy wiedzy jako zbioru stwierdzeń i relacji między nimi, można utworzyć graf, którego węzłami są stwierdzenia, a gałęziami relacje. Węzłom i gałęziom mogą być przypisane współczynniki pewności. Istotą sieci semantycznych jest zatem graficzna prezentacja relacji między obiektami, nazywanymi węzłami (rysunek 5.20). Każdy węzeł odpowiada typowi obiektu i może być związany z innymi obiektami za pomocą łuków, nazywanych również połączeniami.

Rysunek 5.20. Przykład sieci semantycznej



Źródło: Opracowanie własne.

Mechanizm wnioskowania w sieci semantycznej jest wynikiem poruszania się po grafie. Na podstawie inspekcji sieci semantycznej wyprowadzane są konkluzje. Przykładowo, na podstawie rysunku 5.20 można wysnuć konkluzję, że jeśli coś ma skrzydła i jest ptakiem, to jest to zwierzę.

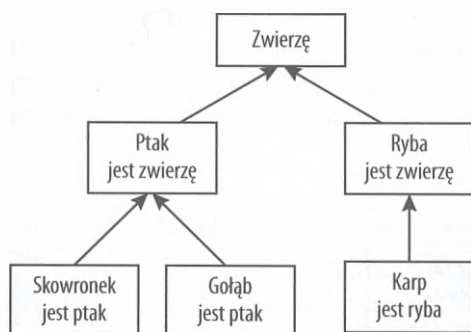
**Ramki** (ang. *frames*) zostały wprowadzone przez M. Minsky'ego w 1975 r. [Minsky 1975]. W metodzie tej każdy obiekt jest reprezentowany w postaci ramki. Można wyróżnić kilka rodzajów ramek [Goczyła, Zawadzka 2006]:

- klasy,
- metaklasy,
- obiekty.

**Klasy**, czyli **pojęcia (koncepty)**, tworzą hierarchię ze względu na występujące między nimi zależności podrzędności i nadrzędności (rysunek 5.21). Każda klasa zawiera klatki, reprezentujące właściwości danej klasy. Klatki też są ramkami, do

których dodatkowo mogą być przypisane tzw. fasety, czyli właściwości klatek nakładające na nie ograniczenia (np. liczność, maksimum, minimum, wartości dopuszczalne). Klasy mogą dziedziczyć od siebie klatki.

Rysunek 5.21. Przykład hierarchii klas



Źródło: Opracowanie własne.

**Klatki** dowolnej ramki mogą być własne lub szablonowe. Klatki własne są własnością charakterystyczną dla danej ramki. Klatki szablonowe są dziedziczone przez kolejne ramki, będące wystąpieniami tych pierwszych. Klatki szablonowe danej klasy stają się klatkami własnymi wszystkich jej wystąpień. Jeśli wystąpienie klasy ma swoje klatki szablonowe, to ta pierwsza klasa staje się metaklasą, czyli „klasą klas”. Ramka, która nie ma klatek szablonowych, nie może mieć swoich wystąpień i wówczas jest obiektem. Przykład hierarchii zbioru ramek prezentuje rysunek 5.22.

Najpopularniejszą formą reprezentacji wiedzy są reguły. **Reguła** składa się z przesłanki i konkluzji [Kwiatkowska 2007, s. 45]. Można ją zapisać w następującej postaci ogólnej:

IF *przesłanka* THEN *konkluzja* lub IF *przesłanka* THEN *działanie*.

Zarówno przesłanka, jak i konkluzja mają postać wyrażen, zwanych klauzulami, np.:

IF *wiek\_czlowieka* < 1 rok THEN *czlowiek jest niemowlęciem*.

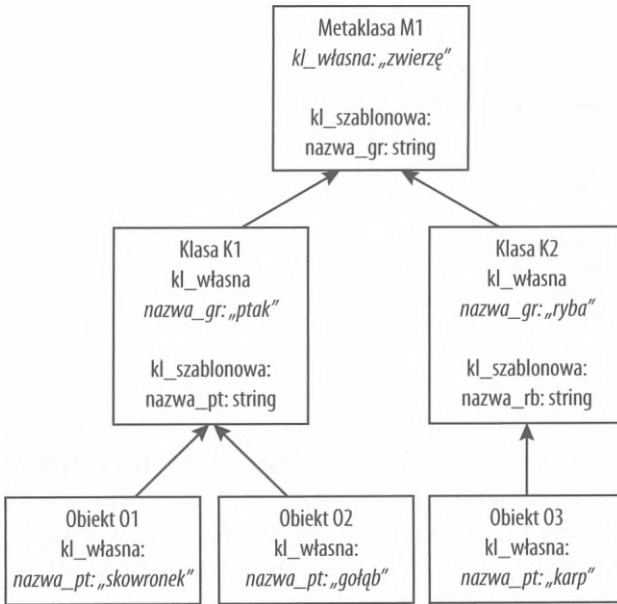
Przesłanki mogą się składać z kilku stwierdzeń połączonych faktoremami logicznymi AND/OR, np.:

IF *skowronek jest ptakiem* AND *ptak jest zwierzęciem* THEN *skowronek jest zwierzęciem*.

Zbiór reguł tworzy w bazie sieć stwierdzeń, gdzie z jednych stwierdzeń wynikają inne i budują zbiór wzajemnych zależności. Ze względu na sposób uzyskiwania ostatecznych konkluzji w procesie wnioskowania rozróżniamy dwa rodzaje reguł:

- 1) reguły proste — mają postać wniosków pośrednich,
- 2) reguły złożone — umożliwiają bezpośrednie wyznaczenie wniosków przez system.

Rysunek 5.22. Przykład hierarchii zbioru ramek



Źródło: Opracowanie własne.

W regułach prostych kolejne wnioski pośrednie tworzą łańcuch wnioskowania. W regułach złożonych każda z reguł w konkluzji zawiera jakiś wniosek końcowy.

## 5.4.2. Charakterystyka i projektowanie regułowych baz wiedzy

Pierwszym krokiem, rozpoczynającym opracowanie **regułowej bazy wiedzy**, jest zgromadzenie odpowiedniej wiedzy. Metody pozyskiwania wiedzy można pogrupować w następujące klasy:

- bezpośrednie pozyskiwanie wiedzy od eksperta,
- pozyskiwanie wiedzy na podstawie doświadczeń i przykładów,
- pozyskiwanie wiedzy na podstawie obserwacji zjawisk i zdarzeń,
- pozyskiwanie wiedzy na podstawie wyszukiwania sytuacji analogicznych,
- pozyskiwanie wiedzy na podstawie instrukcji,
- pozyskiwanie wiedzy na podstawie eksploracji danych w systemie informatycznym.

Kolejny krok to formalizacja zapisu wiedzy w taki sposób, w jaki będzie ona reprezentowana w bazie wiedzy.

Niech rozpatrywanym problemem będzie możliwość zakwalifikowania klienta, zainteresowanego kredytem bankowym, do jednej z trzech klas:  $A$  — wysokiego ryzyka,  $B$  — średniego ryzyka,  $C$  — niskiego ryzyka.

Jeżeli  $C$  oznacza zbiór obiektów  $C = \{C_1, C_2, C_3, \dots, C_n\}$  i każdy z elementów  $C$  jest określony przez zbiór atrybutów  $A = \{A_1, A_2, A_3, \dots, A_m\}$ , a każdy z obiektów  $C$  może należeć do klasy:  $A$ ,  $B$  lub  $C$ , to zbiór danych w rozpatrywanym przykładzie może mieć postać zaprezentowaną w tabelicy 5.5.

Tablica 5.5. Dane źródłowe dla regułowej bazy wiedzy

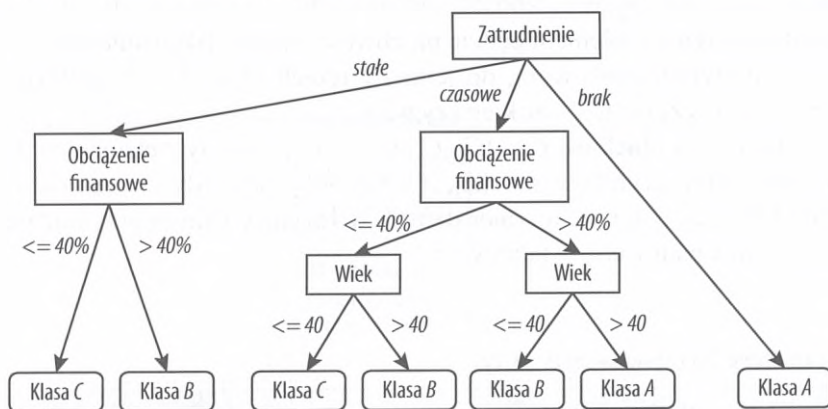
Zatrudnienie	Obciążenia finansowe (jako % wynagrodzenia)	Wiek	Klasa ryzyka
Stałe	$\leq 40\%$	$\leq 40$	$C$
Stałe	$\leq 40\%$	$> 40$	$C$
Stałe	$> 40\%$	$\leq 40$	$B$
Stałe	$> 40\%$	$> 40$	$B$
Czasowe	$\leq 40\%$	$\leq 40$	$C$
Czasowe	$\leq 40\%$	$> 40$	$B$
Czasowe	$40\%$	$\leq 40$	$B$
Czasowe	$> 40\%$	$> 40$	$A$
Brak	$\leq 40\%$	$\leq 40$	$A$
Brak	$\leq 40\%$	$> 40$	$A$
Brak	$> 40\%$	$\leq 40$	$A$
Brak	$> 40\%$	$> 40$	$A$

Źródło: Opracowanie własne.

Obiekt {Zatrudnienie, Obciążenie finansowe, Wiek} to w analizowanym przypadku zbiór obiektów. Wśród atrybutów opisujących dany obiekt można wymienić: obiekt Zatrudnienie = {Stałe, Czasowe, Brak}, obiekt Obciążenie finansowe = { $\leq 40\%$ ,  $> 40\%$ } i obiekt Wiek = { $\leq 40$ ,  $> 40$ }. Każdy obiekt może należeć do klasy  $A$  (wysokie ryzyko kredytowe),  $B$  (średnie ryzyko kredytowe) lub  $C$  (niskie ryzyko kredytowe).

Na podstawie wyników z tabelicy 5.5 można zbudować **drzewo decyzyjne** (rysunek 5.23), które ułatwi tworzenie reguł decyzyjnych.

Rysunek 5.23. Drzewo decyzyjne dla danych regułowej bazy wiedzy



Źródło: Opracowanie własne.

Drzewa decyzyjne mają zdolność do szybkiego rozrastania się, co może wpłynąć niekorzystnie na efektywność funkcjonowania bazy wiedzy. Należy zatem zoptymalizować strukturę drzewa do tych reguł, które faktycznie powinny stanowić strukturę bazy wiedzy. Wykorzystuje się do tego celu **algorytmy optymalizacyjne** (np. algorytm Quinlana [Kwiatkowska 2007, s. 61–67]). W przypadku tablicy 5.5 należy zauważyć, że:

- brak zatrudnienia automatycznie powoduje przydzielenie klienta do klasy wysokiego ryzyka — *A*;
- w przypadku stałego zatrudnienia o przynależności do klasy *B* lub *C* decyduje jedynie stan obciążenia finansowego.

Na podstawie drzewa decyzyjnego (rysunek 5.23) są tworzone następujące reguły:

**R1:** *IF* Zatrudnienie *IS* Brak *THEN* Klasa = *A*,

**R2:** *IF* Zatrudnienie *IS* Czasowe *AND* Obciążenie finansowe  $\leq 40\%$  *AND* Wiek  $\leq 40$  *THEN* Klasa = *C*.

Ponieważ w trakcie eksploatacji baza wiedzy podlega modyfikacjom — jedno reguły są usuwane, inne wprowadzane, łatwo jest utracić kontrolę nad zawartością, spójnością i poprawnością bazy wiedzy. Żeby uniknąć takich sytuacji, należy grupować reguły oraz tworzyć sieci zależności między nimi [Kwiatkowska 2007, s. 48–49]. Reguły mogą być pogrupowane według konkluzji lub przesłanki. Następnie należy uporządkować reguły w ramach każdej grupy, np. według stopnia ich ważności. Przykładowo, umieszczenie obok siebie reguł o tych samych konkluzjach pozwala na prześledzenie przesłanek, które do nich prowadzą, a co się z tym wiąże — wyeliminowanie wszelkich powtórzeń oraz niespójności. Sieci zależności są z kolei graficzną metodą sprawdzenia kompletności bazy reguł. Dzięki tej metodzie można

zidentyfikować reguły, które nie są powiązane z innymi regułami, nie prowadzą do żadnej konkluzji końcowej, zatem nie są wykorzystywane w procesie wnioskowania.

### 5.4.3. Przykład wykorzystania BW i modułu wnioskującego w SWD

Wykorzystywana w służbie więziennej aplikacja NOE spełnia wymagania **regulowego systemu ekspertowego** [Strzykowski 2005, s. 173], wspomagającego proces rozmieszczania osób osadzonych w zakładach karnych i aresztach śledczych. Posiada wyodrębnioną bazę wiedzy, mechanizm wnioskowania oraz moduł komunikacji z użytkownikiem.

Baza wiedzy zapisana jest w postaci: reguł prostych oraz reguł złożonych za pomocą koniunkcji przesłanek, alternatywy przesłanek lub koniunkcji wniosków. Oto przykład reguły złożonej z koniunkcją przesłanek [Strzykowski 2005, s. 173]:

JEŻELI przyczyna\_zdjecia\_z\_ewidencji\_jednostki = zgon i ilosc\_orzeczen\_zakonczonych\_zgon > 0 TO komunikat ('w polu miejsce\_pobytu wprowadź miejsce pochowania zwłok') W PRZECIWNYM RAZIE komunikat ('NOE wykrył, że żadne z orzeczeń nie zostało zakończone z powodu zgonu...')

Algorytm Avalon, wbudowany w mechanizm wnioskowania, zapewnia automatyczne i suboptymalne rozmieszczanie osób osadzonych w miejscach zakwaterowania [Strzykowski 2005, s. 175]. Do formalnego opisu kryteriów i ograniczeń rozmieszczenia osadzonych zastosowano rachunek predykatów. Oto przykład ograniczenia dotyczącego rozmieszczania osób osadzonych zaliczonych do kategorii „niebezpiecznych” (NB), zapisanego w rachunku predykatów:

$$\exists x, y, c \ x \neq y: \text{NB}(x, c) \cap \neg \text{NB}(y, c) \Rightarrow \text{PRO}(c) = F,$$

$$\exists x, y, c \ x \neq y: \text{NB}(x, c) \cap \text{NB}(y, c) \Rightarrow \text{PRO}(c) = T.$$

Komunikacja z użytkownikiem odbywa się poprzez interfejs użytkownika i pozwala na interakcyjną współpracę i porozumiewanie się z systemem.

System pełni funkcje aplikacji:

- doradczej, np.: „W wybranej przez Ciebie celi brak miejsc. Czy umieścić dyspozycję osadzenia w poczekalni?” [Strzykowski 2005, s. 174];
- kontrolnej, np.: „Dane przyjętego do jednostki osadzonego są niepełne. Wymagane uzupełnienie danych.” [Strzykowski 2005, s. 174];
- automatyzującej podejmowanie decyzji, np.: w czasie rzeczywistym podejmowane są decyzje o konieczności transferowania faktów z bazy wiedzy do bazy centralnej systemu transakcyjnego [Strzykowski 2005, s. 174].

Z systemu NOE korzysta ponad 2,5 tys. użytkowników we wszystkich zakładach karnych i aresztach śledczych.

## 5.5. Pytania i zadania kontrolne

### 5.5.1. Pytania kontrolne

1. Co to jest i do czego jest wykorzystywana HD w SWD?
2. Czym się różni zastosowanie HD i składnicy danych w SWD?
3. W jaki sposób jest tworzony i z jakich elementów składa się model pojęciowy HD?
4. Jaki jest związek między modelem logicznym i fizycznym HD?
5. Porównaj analizę OLAP i eksplorację danych.
6. Wymień kilka praktycznych zastosowań HD w SWD.
7. Co to jest BW i jakie ma zastosowanie w SWD?
8. W jakim celu pozyskiwana jest wiedza w przedsiębiorstwie? Jakie są metody pozyskiwania wiedzy?
9. Jakie są zadania i jakie różnią się typy reprezentacji wiedzy w BW?
10. Jak jest zastosowanie algorytmów optymalizacyjnych w BW?

### 5.5.2. Przykładowe zadania

1. Zdefiniuj tabelę faktów oraz tabele wymiarów dla modelu danych hurtowni wspomagającej pracę działu planowania sprzedaży. Hurtownia danych ma powstać dla ogólnopolskiej sieci sklepów spożywczych w celu lepszej organizacji planu dostaw oraz skoordynowania go z planami sprzedaży.
2. Przedsiębiorstwo korzysta z monolitycznej korporacyjnej hurtowni danych. Jaką zmianę w architekturze rozwiązania należy zalecić w przypadku, kiedy użytkownicy skarżą się na wysoki poziom skomplikowania obsługi systemu opartego na takiej koncepcji hurtowni? Odpowiedź uzasadnij.
3. Jakie analizy (OLAP, *data mining*) można zaproponować w dziale obsługi klienta firmy ubezpieczeniowej, które mogłyby wpłynąć na podniesienie jakości usługi i obniżenie liczby zerwanych przez klientów polis ubezpieczeniowych? Odpowiedź uzasadnij.
4. Zaproponuj przykłady: reguły prostej, reguły złożonej z koniunkcją przesłanek, reguły złożonej z alternatywą przesłanek oraz reguły złożonej z koniunkcją wniosków.
5. Zaprojektuj dowolnie złożoną regułę decyzyjną warunkującą zdatność maszyny do pracy przy uwzględnieniu minimum trzech kryteriów, korzystając z ogólnej postaci:

IF *warunek* THEN *wniosek* AND/OR *akcja*.



6. Zaprojektuj własny przykład bazy reguł zbudowanej z minimum trzech reguł dotyczących zadowolenia klienta z powodu dostarczenia towaru wadliwego lub towaru o wymaganej jakości w terminie lub z opóźnieniem.
7. Zaprojektuj dowolną sieć semantyczną przyporządkowującą dowolny obiekt techniczny do szerszej klasy obiektów.
8. Na podstawie sieci semantycznej z rysunku 5.20 zbuduj zbiór reguł z przypisanymi współczynnikami pewności CF.
9. Na podstawie danych z tablicy 5.5 zbuduj sieć semantyczną z przypisanymi współczynnikami pewności CF.

## Rozdział 6

# Metody i narzędzia projektowania SWD

W rozdziale przedstawiono podstawowe metody oraz narzędzia inżynierii wiedzy stosowane w projektowaniu i budowaniu prototypów SWD. W szczególności pokazano zastosowanie arkusza kalkulacyjnego do budowy tablic decyzyjnych oraz opracowywania scenariuszy i animacji zdarzeń towarzyszących obliczeniom symulacyjnym w trakcie opracowywania lub prognozowania skutków decyzji. W rozdziale przedstawiono również język SQL oraz przykłady języków programowania wysokiego poziomu i pakiety narzędziowe inżynierii wiedzy, służące do konstruowania prototypów SWD. Wspomaganie procesu podejmowania decyzji częstokroć następuje w rezultacie eksperymentów testujących lub symulacyjnych, jakie towarzyszą procesowi prototypowania SWD. Najbardziej zaawansowaną postacią projektowanych SWD są systemy ekspertowe wspomagające, uzasadniające (wyjaśniające) i monitorujące procesy podejmowania decyzji.

## 6.1. Metody projektowania SWD

SWD jest zdywersyfikowaną klasą technologii komputerowej integrującą informacje z baz danych i analityczne metody modelowania (sztuczną inteligencję, analizę decyzyjną, modele optymalizacyjne) [Adelman 1991], [Herbst 1996].

### 6.1.1. Specyfikacja funkcjonalności SWD

Funkcjonalność współczesnych systemów SWD jest uwarunkowana wieloma ograniczeniami. Do najważniejszych należą oczywiście koszty oraz czas konieczny na opracowanie i wdrożenie, a także dostępność technologii niezbędnych do zaprojektowania i wdrożenia SWD. Bariera jest również konieczność uaktualniania BD i baz wiedzy (BW) — jeśli wdrożenie ma być pożyteczne, co wiąże się z koniecznością komercjalizacji SWD i interaktywnym udziałem użytkowników SWD gotowych także płacić za udostępnianą przez SWD informację.

Wśród współczesnych trendów w tym obszarze rozpowszechnione są tzw. metodyki zwinnego projektowania, obejmujące najważniejsze kierunki innowacji w następujących obszarach:

- integracja systemów danych i procesów;
- unifikacja funkcji cząstkowych systemów;
- zwiększanie dostępności do BD dla wszystkich komórek organizacyjnych;
- upowszechnienie nowoczesnych sposobów prezentacji danych (wizualizacji) ze względu na cele wspomaganie ich analizy;
- doskonalenie procesów podejmowania decyzji i ich przekazywania;
- zmierzanie do budowy modułowej i otwartości całego systemu.

Z bardziej szczegółowych zmian w tym zakresie należy wyróżnić m.in. zapewnienie kompleksowego charakteru funkcjonowania SWD (dostępność, skuteczność podjętych decyzji rozumiana kompleksowo), permanentne podnoszenie poziomu technologicznego (metod zarządzania i samego systemu), wzrost elastyczności funkcjonalnej i strukturalnej, zapewnienie stałej spójności ze zmieniającymi się elementami otoczenia, a zwłaszcza ze stanem prawnym ewoluującym zgodnie z przyjętymi procedurami legislacyjnymi, a także z wymaganiami bezpieczeństwa, poufności oraz integralności.

## 6.1.2. Modelowanie architektury SWD

Twórcy **metod sztucznej inteligencji** (AI) starają się tak zaprojektować architekturę systemów, aby niezależnie zdeterminowany technologią i narzędziami programistycznymi interfejs użytkownika i BD od logicznych i pojęciowych zagadnień, odnoszących się bezpośrednio do rozwiązywanego problemu. Ma temu sprzyjać architektura trójwarstwowa, w której ramach proponuje się podział systemu na trzy poziomy: schemat wewnętrzny — jego fizyczną implementację; schemat pojęciowy, czyli abstrakcyjny model wycinka rzeczywistości (biznesu, firmy) odzwierciedlanej przez system oraz pytania w cyklu życia systemu, gdzie wyodrębnia się takie fazy, jak:

- właściwa struktura obrazująca podział na części i relacje występujące między rozważanym systemem a jego otoczeniem;
- określony sposób funkcjonowania;
- proces rozwoju, tzn. każdy system ma swój początek, rozwija się i wreszcie jest wycofywany z użycia.

Rozwój metod wspomaganie decyzji wszedł obecnie w fazę, w której pojawiły się dwa podstawowe wyzwania: po pierwsze, tworzenie teorii i metod użytecznych dla projektantów systemów oraz po drugie — weryfikacja badań oraz doskonalenie projektów w kierunku coraz sprawniejszej i skuteczniejszej eksploatacji [Power 2001–2003].

Do tradycyjnego zestawu komponentów uczestniczących w procesie projektowania SWD (dane, dialog, modelowanie), a określonych w paradygmacie Sprague'a—

–Carlsona, należy obecnie dodać komponent kreatywności oraz składnik komunikacji [Stanek, Sroka 1999–2001]. Dlatego też niezbędne jest ukierunkowanie architektury SWD na następujące nowe możliwości:

- dominacja komponentu kreatywności doskonałego spójnie z intensywnie poszerzaną wiedzą o technicznych podstawach komputerowego wspomaganie kreatywności;
- uwzględnianie wysiłków pracowników wiedzy podejmujących wyzwanie diagnozy zagrożeń, możliwości oraz rozwiązań wpływających istotnie na funkcjonowanie ich macierzystych organizacji;
- wspomaganie holistycznych, wizualnych, intuicyjnych procesów — co tradycyjnie uznaje się za fakt w „mózgu”;
- umożliwienie, poprzez zastosowanie technologii hybrydowego systemu wieloagentowego, integracji komponentu kreatywności z pozostałymi ww. komponentami.

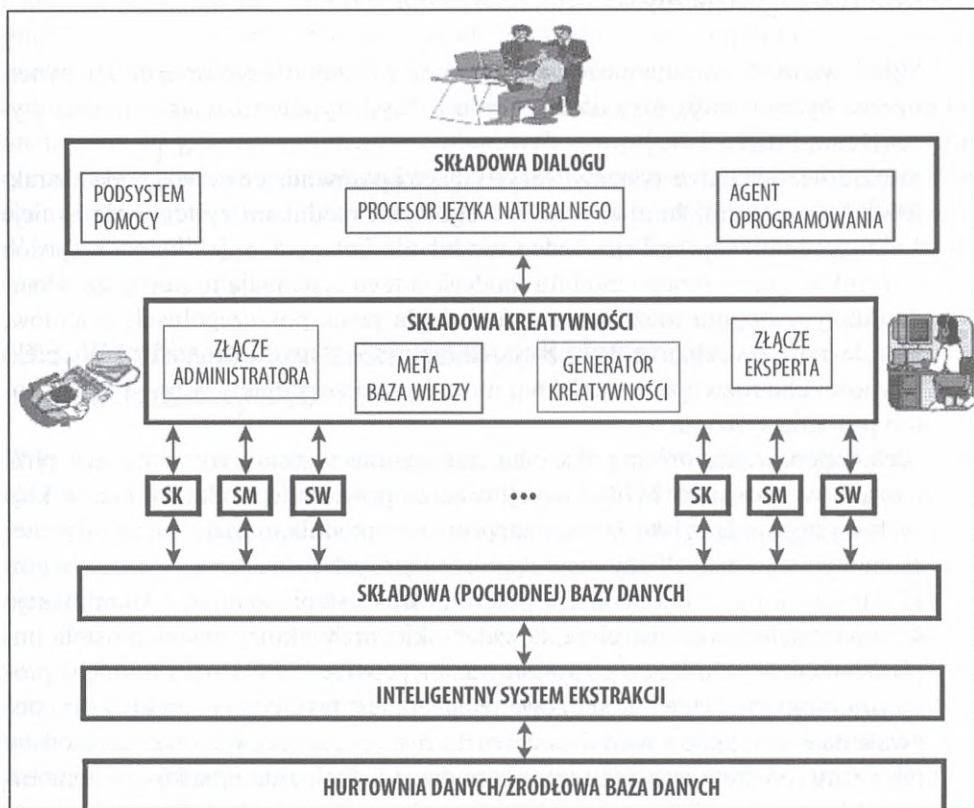
**Architektura kreatywnego SWD** (rysunek 6.1) zawiera następujące elementy [Stanek 1999]:

- składowa dialogu: podsystem pomocy, procesor języka naturalnego, agent oprogramowania;
- składowa kreatywności: złącze administratora, metabaza wiedzy, generator kreatywności, złącze eksperta;
- składowa komunikacji;
- składowa modelowania;
- składowa wiedzy;
- składowa BD;
- inteligentny system ekstrakcji;
- hurtownia danych (HD), źródłowe BD.

W odniesieniu do składowej komunikacji należy zwrócić uwagę na fakt, że nie można przewidzieć wszystkich przyszłych oczekiwań użytkowników. Zatem założono, że **hybrydowy system sztucznej inteligencji** powinien mieć jak najbardziej otwartą architekturę złożoną z możliwie najbardziej niezależnych elementów [Kostrubala, Stolecki, Twardowski 2004–2005]. W celu sprostania temu wyzwaniu zastosowano podejście agentowe, gdzie poszczególne części składowe systemu hybrydowego są agentami wykonującymi samodzielnie zadania.

Koncepcja Blackboard Systems jest najbardziej zbieżnym rozwiązaniem z ideą zespołowego realizowania zadań, w których poszczególni agenci (członkowie zespołu) zapisują na tablicy kolejne kroki działań (na podstawie już zapisanych na tablicy informacji) tak długo, aż znajdą rozwiązanie końcowe. Rozproszony charakter omawianego rozwiązania sprawia, że systemy tablicowe byłyby trudne do zrealizowania ze względu na współdzielenie zasobów — dlatego zdecydowano się na protokół Contract Net, którego efektywność w realizacji systemów rozproszonych jest

Rysunek 6.1. Architektura kreatywnego systemu wspomagania decyzji



Źródło: [Stanek 1999].

wysoka, a zarazem dobrze oddaje zespołową pracę agentów, kooperujących ze sobą w celu realizacji zadań pośrednich, składających się na cel główny [d'Inverno 2006]. Agent zarządzający, który inicjuje proces, a następnie poszukuje potencjalnych dostawców rozwiązania, odpowiada w omawianym protokole za współpracę agentów między sobą.

Istnieją zróżnicowane źródła danych wejściowych w zależności od stopnia generalizacji obserwacji i horyzontów podejmowania decyzji. Dane wejściowe, w zależności od zadanego stopnia generalizacji obserwacji (poziom operacyjny bądź strategiczny), pochodzą z:

- systemów transakcyjnych bezpośrednio z BD i/lub jako sygnały alarmowe generowane poprzez zaimplementowane procedury kontrolujące bieżące zmiany w bazach danych [Bassiliades, Vlahayas 2000];
- HD, gdzie odpowiednio predefiniowane agregacje danych ilościowych przechowywane w kostkach OLAP oraz dane jakościowe pozwalają na generowanie

sygnałów ostrzegawczych dla potrzeb długookresowego monitorowania osiągniętych celów strategicznych.

Składowa modelowania może być oparta na założeniu uzyskania efektu synergii poprzez hybrydyzację rozwiązań. Wśród różnych typów rozwiązań można wymienić [Duan, Burreli 1995]:

- niezależne, oddzielne systemy weryfikujące i porównujące rozwiązania charakteryzujące się tym, że między poszczególnymi modułami systemu nie istnieje żadna komunikacja ani też żaden moduł nie korzysta w jakikolwiek sposób z wyników pracy innego modułu; podejścia tego typu mają tę zaletę, że w bardzo dużym stopniu możliwa jest równoległa praca poszczególnych modułów; niezależność dwóch pracujących systemów może służyć również badaniu efektywności obu rozwiązań; model ten może być także przyjęty w przypadku prototypowania systemu;
- zdekomponowane, luźno połączone inteligentne systemy wymieniające pliki danych; w systemach hybrydowych o luźno powiązanej architekturze, w których występuje zjawisko komunikacji między modułami; najczęściej jako medium wymiany danych używane są pliki, w których jeden proces zapisuje wyniki danych, inny je odczytuje, a potem może nastąpić zamiana; komunikacja w tym modelu jest dwustronna; do zalet takiej architektury należy prostota implementacji, w wyniku czego systemy na niej oparte są bardzo przydatne w prototypowaniu rozwiązań; dodatkową jej zaletą jest fakt, że taka architektura pozwala na korzystanie z modułów zewnętrznych dostarczonych przez inne osoby lub firmy (prostota tego rozwiązania podnosi jednak znacząco koszty komunikacji obejmujące odczyt, zapis na dysku oraz parsowanie danych);
- systemy zwarte przekazujące sobie informacje poprzez pamięciowe struktury danych;
- systemy ewoluujące z jednego typu systemu w inny;
- systemy pełnej integracji dzielące struktury danych i reprezentację wiedzy, których zaletą jest usunięcie zbytecznej redundancji, a rozwiązania są ściśle dedykowane do danego typu problemu i trudno je zaadaptować do innych projektów; trudne może być zapewnienie równoległości pracy modułów ze względu na dużą ich liczbę i skomplikowane zależności między nimi [Medsker 1994].

Składowa dialogu może zostać uwzględniona poprzez zastosowanie pulpitu analitycznego. W proponowanym podejściu pulpit analityczny jest agentem interfejsu użytkownika działającym w środowisku intranetu lub Internetu, co umożliwia dostęp do wyników analiz i wygenerowanych raportów poprzez samodzielne poszukiwanie adekwatnej do problemu informacji przez każdego z agentów (obok dodatkowych źródeł danych, takich jak HD).

Składowa kreatywności może zostać uwzględniona w podejściu opartym na sygnalizatorach ryzyka. Podejście to polega na generowaniu, a następnie na we-

ryfikacji określonego zbioru hipotez (scenariuszy). Sygnalizatory ryzyka są kluczowymi wskaźnikami efektywności (KPI) identyfikowanymi dla każdego diagnozowanego obszaru. Na podstawie interaktywnych zapytań analityka do wielowymiarowych BD oraz BW systemu tworzone są w SE hipotezy w postaci scenariuszy decyzyjnych [Stanek, Sroka 2001]. Wynikiem użycia procedur analitycznych jest identyfikacja ewentualnych symptomów zagrożeń w poszczególnych diagnozowanych obszarach aktywności biznesowej organizacji gospodarczej (przedsiębiorstwa, sieci biznesu). Stwierdzone odchylenia sygnalizatorów w istotny sposób kształtują poziom ryzyka w badanych obszarach osiągnięcia celów organizacji gospodarczych.

### 6.1.3. Weryfikacja niesprzeczności modeli elementów architektury SWD

W procesie weryfikacji niesprzeczności modeli elementów architektury SWD można zastosować **teorię zbiorów przybliżonych** (TZP), której podstawowym celem jest opanowanie niepewności i nieprecyzyjności informacji. Zastosowanie zbiorów przybliżonych pozwala na indukcję reguł decyzyjnych w sytuacji, gdy zbiór danych opisujących zagadnienie jest wewnętrznym sprzeczny.

Dane pochodzące z rynków finansowych, zwłaszcza dane wysokiej częstotliwości, powodują zwykle bardzo wyraźną niespójność informacyjną. Sytuacje opisane tymi samymi wartościami atrybutów warunkowych, lecz prowadzące do różnych zachowań ceny, występują bardzo często. Statystyczna przewaga na korzyść jednego kierunku zmiany ceny bywa minimalna. Wówczas ujawnienie deterministycznych reguł okazuje się niezwykle trudne lub wręcz niemożliwe.

Wśród najważniejszych zalet teorii zbiorów przybliżonych, predestynujących ją do zastosowania w automatycznych systemach transakcyjnych, warto wymienić następujące fakty [Pawlak 2007]:

- generowane reguły są jasne, zrozumiałe dla człowieka i łatwe do zaimplementowania w systemie automatyzacji procesów biznesowych;
- istnieje możliwość uwzględnienia dowolnych atrybutów — ciągłych, dyskretnych oraz kwalifikatywnych;
- TZP pozwala na efektywną budowę reguł dla sprzecznych wewnątrz danych.

W największym stopniu informatyzacji poddaje się etap wyboru rozwiązań (określenia wartości zmiennych decyzyjnych), w czasie którego następuje poszukiwanie rozwiązania modelu (poprzez optymalizację, zastosowanie technik sztucznej inteligencji), testowanie i ocena rozwiązań (np. symulacyjnych przez analizę wrażliwości) oraz integrowanie (likwidowanie niespójności) częściowych rozwiązań z założeniami decydenta (korzystanie z systemów ekspertowych, zwłaszcza o architekturze tablicowej). Jako ostatni w kolejności jest wykonywany etap adaptacji rozwiązań do zastosowań w systemie rzeczywistym. Oczywiście w przypadku

braku powodzenia następuje iteracja do odpowiednio wcześniejszego etapu podejmowania decyzji [Vanthienen, Wets 1995].

Weryfikację niesprzeczności modeli elementów architektury SWD uwzględniono m.in. w pakiecie CLIPS, który zawiera wiele elementów wsparcia weryfikacji i walidacji systemów ekspertowych, np.: projektowanie modularne i podział bazy wiedzy, statyczną i dynamiczną kontrolę atrybutów pól i argumentów funkcji oraz analizę semantyczną wzorcowych reguł w celu ustalenia, czy niespójności mogłyby przeszkodzić w odpaleniu reguły lub wygenerować błąd.

## 6.2. Narzędzia projektowania SWD

SWD są wyposażone w narzędzia inżynierii wiedzy, których podstawowym zadaniem jest zbieranie i formalizacja wiedzy ekspertów do postaci gotowych do zastosowania reguł decyzyjnych. Najbardziej zaawansowaną postacią SWD są systemy ekspertowe wspomagające i uzasadniające (wyjaśniające) procesy podejmowania decyzji.

W kolejnych punktach przedstawiono narzędzia inżynierii wiedzy stosowane w projektowaniu SWD i budowaniu ich prototypów. Wspomaganie procesu podejmowania decyzji następuje w rezultacie eksperymentów testujących lub symulacyjnych, jakie towarzyszą procesowi prototypowania. Na podstawie zweryfikowanych prototypów projektuje się i wdraża najbardziej skuteczne rozwiązania techniczno-metodyczne.

### 6.2.1. Narzędzia inżynierii wiedzy

Charakteryzując narzędzia inżynierii wiedzy, omówiono techniki konsultacji z ekspertem, wybrane manualne metody pozyskiwania wiedzy, metody odkrywania wiedzy z danych (SQL, OLAP, OLTP, *Data Mining*) oraz techniki badań kwestionariuszowych.

W zakresie technik konsultacji z ekspertem można wyróżnić następujące sposoby pozyskiwania wiedzy:

- obserwacja eksperta w miejscu pracy, gdzie inżynier wiedzy jest biernym obserwatorem, dążącym jednocześnie do pełnego przeanalizowania złożoności problemu;
- dyskusja problemu, która ma na celu określenie metody organizacji wiedzy eksperta o problemie;
- opisywanie problemu — na tym etapie ekspert opisuje problem dla każdej prawdopodobnej kategorii danych wejściowych, którą może podać użytkownik, co pomaga sformułować prototypowy problem; a później następuje wybór strategii rozwiązania problemu spośród proponowanych wariantów;
- analizowanie problemu — inżynier wiedzy proponuje ekspertowi do rozwiązania wiele problemów, analizując sposób rozumowania eksperta w trakcie ich rozwiązywania, a właściwa analiza przyczynia się do poprawnego zrozumienia



problemu oraz opracowania skutecznej metody jego rozwiązywania; nawet zbyt szczegółowa analiza problemu jest bardziej wskazana niż jej dokonanie w sposób niezadowolający;

- doskonalenie systemu — ekspert wskazuje inżynierowi wiedzy problemy do rozwiązania, począwszy od bardzo łatwych, a skończywszy na dość trudnych; w ten sposób ekspert przekonuje się, czy przekazał swoją wiedzę we właściwy sposób; następuje wtedy sprzężenie zwrotne między inżynierem wiedzy a ekspertem, gwarantujące wzrost jakości opracowywanego systemu;
- testowanie systemu — w tej fazie pracy ekspert testuje i wydaje opinię o każdej regule wprowadzonej do prototypu systemu ekspertowego oraz dokonuje także oceny strategii stosowanych do wyboru reguł, a w efekcie następuje weryfikacja koncepcji pod kątem prawidłowości ich wyrażenia w sposób abstrakcyjny oraz zaimplementowania w systemie;
- legalizacja systemu — inni eksperci oceniają poprawność merytoryczną działania prototypu systemu ekspertowego i dokonują jego krytycznej analizy, a wniesione korekty mają na celu udoskonalenie systemu.

**Manualne metody pozyskiwania wiedzy** są wykorzystywane najczęściej podczas prototypowania SE, tworzenia jego pierwotnej wersji. Polegają one na pozyskaniu wiedzy niezbędnej do stworzenia prototypu SE od ekspertów z danej dziedziny wiedzy. Na tym etapie tworzy się bazę słownikową, czyli bazę zawierającą terminy specjalistyczne i fachowe z danej dziedziny, które później będą wykorzystywane podczas komunikacji z użytkownikiem. Metody manualne powinny także umożliwić stworzenie bazy faktów, danych i reguł, bez których działanie prototypu systemu ekspertowego byłoby niemożliwe. Metody manualne, ze względu na swoją specyfikę, wymagają ścisłej współpracy między ekspertem a inżynierem wiedzy, od którego zależy, w jaki sposób jego SE będzie odzwierciedlał rzeczywistość i jak precyzyjne dane pozyska od eksperta. Do metod manualnych zaliczamy takie metody, jak: wywiad, analiza protokołów, analizowanie problemu, kwestionariusze, obserwacja ekspertów w miejscu pracy, burza mózgów, wykorzystanie symulacji komputerowej, inne metody (np. gromadzenie materiałów źródłowych, przegląd literatury, kreowanie eksperymentów).

Istnieją również **półautomatyczne metody pozyskiwania wiedzy**, a wśród nich można wyróżnić, takie jak:

- metoda analogii — metoda ta polega na takiej transformacji istniejącej bazy wiedzy, aby mogła być ona użyteczna do opisów faktów podobnych, choć nie tych samych, do tych, które już wcześniej zostały zawarte w bazie wiedzy systemu;
- bezpośrednie zapisanie wiedzy — tzw. uczenie na pamięć (ang. *rote learning*) nie wymaga od systemu doradczego, który podlega uczeniu, żadnego wnioskowania czy też transformacji danych; przykładem zastosowania takiego algorytmu uczenia może być bezpośrednio zaprogramowanie bazy wiedzy;

- analizowanie przykładów — jest to szczególny sposób uczenia o charakterze indukcyjnym;
- tworzenie instrukcji — tzw. uczenie poprzez przekazanie informacji; pozyskiwanie wiedzy polega na wykorzystaniu źródeł wiedzy wskazanych przez eksperta, np. odpowiedniej bibliografii, a następnie na ich transformacji na język zrozumiały przez system;
- prowadzenie obserwacji — uczenie, zwane uczeniem bez nauczyciela, stanowiące uogólnioną formę uczenia indukcyjnego, kiedy w schematach pozyskiwania wiedzy nie korzysta się z pomocy nauczyciela; w metodzie tej znaczny udział uczącego się; tego rodzaju metoda pozyskiwania wiedzy jest wykorzystywana w technikach grupowania czy też technikach rozpoznawania obrazów.

**Odkrywanie wiedzy z danych** (narzędzia: SQL, OLAP, OLTP, *Data Mining*) [Morzy 1999], na skutek rosnących lawinowo zasobów danych, jak również potrzeb konkurujących przedsiębiorstw o dostęp do szybko dostarczonej i trafnej informacji, stało się ważnym powodem powszechnego pojawiania się coraz to bardziej zaawansowanych narzędzi inżynierii wiedzy.

Postęp technologiczny w zakresie cyfrowego generowania i gromadzenia informacji doprowadził do przekształcenia się BD wielu przedsiębiorstw, urzędów i placówek badawczych w rezerwuary ogromnych zasobów danych. Na bezprecedensowy, wielki rozrost systemów bazodanowych złożyło się przede wszystkim upowszechnienie kodów kreskowych i kart płatniczych oraz pojawienie się szybszych, pojemniejszych i tańszych pamięci masowych. Nasze możliwości analizowania i rozumienia tak dużych wolumenów danych są znacznie mniejsze od możliwości ich zbierania i przechowywania. Zebrane w BD zapisy, np. o dotychczasowej działalności przedsiębiorstwa, poziomie i strukturze sprzedaży oraz cechach klientów, mogą być wykorzystane do wspomagania podejmowania decyzji o dalszym kształtowaniu sprzedaży i kierunkach marketingu przedsiębiorstwa. Powstające SWD<sup>1</sup> bazują na wiedzy ekspertów, po części pochodzącej z analizy zawartości BD i HD. Na prostą i szybką analizę pozwalają w szczególności środowiska typu OLAP, które umożliwiają wielowymiarową obserwację zagregowanych wartości danych pobranych z HD.

Odkrywanie wiedzy nie wymaga przygotowywania hipotez przez ekspertów, ponieważ są one automatycznie generowane i weryfikowane. Zadania ekspertów sprowadzają się do oceny i akceptacji odkrytej wiedzy, najczęściej poprzez kontrolowanie jej wskaźników statystycznych. Na przykład, w procesie odkrywania wiedzy ekspert wskazuje zbiór danych o sprzedaży określonego typu maszyn i ustala, że zależności interesujące to te, które są spełnione przez co najmniej 60% transakcji sprzedaży. W odpowiedzi ekspert uzyskuje zbiór wszystkich zależności, jakie zachodzą, w co najmniej 60% danych o sprzedaży. Znalezione zależności są zwykle

<sup>1</sup> [www.informationbuilders.com/decision-support-systems-dss.html](http://www.informationbuilders.com/decision-support-systems-dss.html).

bardziej precyzyjne (OLAP np. informuje uzupełniająco o sprzedaży maszyn w układzie geograficznym w I kwartale danego roku).

Można wyróżnić takie etapy odkrywania wiedzy, jak:

- selekcja danych — wybór relacji i krotek, które będą eksplorowane, definiowanie sposobu łączenia relacji;
- transformacja danych — konwersja typów atrybutów, definiowanie atrybutów wywiedzionych, dyskretyzacja wartości ciągłych;
- eksploracja — ekstrakcja wiedzy z danych: generowanie reguł, drzew decyzyjnych, sieci neuronowych itp.;
- interpretacja wyników — wybór najbardziej interesującej wiedzy, logiczna i graficzna wizualizacja wyników.

W **eksploracji danych** używa się różnych technik, które budują specyficzne rodzaje wiedzy. Odkrywana wiedza może odwzorowywać klasyfikacje, regresje, klastrowanie, charakterystyki, dyskryminacje, asocjacje i inne w zależności od jej zastosowania.

Według ocen analityków wolumen danych biznesowych przechowywanych w firmach rośnie co roku dwukrotnie. Nowoczesna gospodarka natomiast stawia coraz wyższe wymagania wobec jakości i szybkości analizy danych wykorzystywanych w procesach gospodarczych. Systemy informatyczne stosowane w firmach służą do obsługi działalności danego przedsiębiorstwa, m.in. do przetwarzania transakcji w czasie rzeczywistym (w trybie online<sup>2</sup>), np. systemy OLTP.

Ponieważ OLTP nie wspierają bezpośrednio procesów analizy danych na potrzeby SWD, opracowano nowe narzędzia przetwarzania analitycznego OLAP, które dla wspomagania procesów decyzyjnych wykorzystuje odpowiednio przygotowane dane gromadzone w HD.

Przykładem takiej analizy jest zapytanie o sprzedaż produktów w supermarkecie w kolejnych kwartałach, miesiącach, tygodniach itp., zapytanie o sprzedaż produktów z podziałem na rodzaje produktów (AGD, produkty spożywcze, kosmetyki itp.) czy wreszcie zapytanie o sprzedaż produktów z podziałem na oddziały supermarketu. Odpowiedzi na powyższe zapytania umożliwiają decydom określenie wąskich gardeł sprzedaży, produktów przynoszących deficyt oraz podjęcie odpowiednich działań poprawiających sytuację.

Chcąc posłużyć się kwestionariuszem jako metodą pozyskiwania wiedzy, inżynier wiedzy najpierw tworzy kwestionariusz, który daje do wypełnienia ekspertowi lub grupie ekspertów w celu weryfikacji wyników. Kwestionariusz powinien zawierać pytania, które umożliwią inżynierowi wiedzy zdobycie interesujących go informacji oraz poprzez odpowiednio sformułowane pytania wymuszają ekstrakcję wiedzy.

<sup>2</sup> <http://databases.about.com/cs/development/g/OLTP.htm>.

## 6.2.2. Prototypowanie SWD

**Prototypowanie SWD** jest procesem selekcionowania koncepcji poprzez proces budowy coraz doskonalszych modeli rozważanego systemu w trójkącie kompromisów: jakość, koszty i czas. Wysoką rangę prototypowania podkreślono już w rozdziale 3, poświęconym m.in. projektowaniu interfejsu użytkownika SWD.

Ponieważ w sensie technologicznym SD jest przede wszystkim systemem informatycznym (SI), jego projektowanie przebiega według typowego schematu (rysunek 6.2) [Fuglewicz, Stąpor, Trojan 1996], który ilustruje ciąg kolejnych etapów cyklu życia SI. Ciąg ten złożony jest z wyodrębnionych, spójnych etapów, pozwalających na pełne i skuteczne zaprojektowanie, oprogramowanie, a następnie użytkowanie SI.

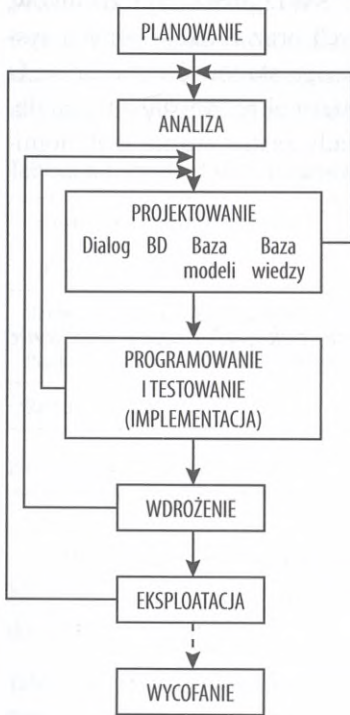
**Modele cyklu życia systemu** SDLC można przedstawiać w następujących klasyfikacjach: tradycyjny (liniowy, kaskadowy), ewolucyjny, przyrostowy, systemów BD Fry'ego, z prototypem, spiralny. Model z prototypem obejmuje kolejne etapy: ogólne określenie wymagań, budowa prototypu, weryfikacja prototypu przez użytkownika, modyfikacja prototypu, realizacja zgodnie z modelem kaskadowym, eksploatacja i modyfikacja systemu. Do metod wykorzystywanych do tworzenia prototypów należy zaliczyć, takie jak: niepełna realizacja, języki wysokiego poziomu, wykorzystywanie gotowych procedur i bibliotek, generatory interfejsu użytkownika, szybkie programowanie, biała tablica, karteczki — rozrysowanie interfejsu systemu. Wady prototypowego projektowania SWD to przede wszystkim: wysoki koszt tworzenia systemu, wymaganie posiadania odpowiednich narzędzi, długie oczekiwanie użytkownika na gotowy system, a prototyp jest opracowywany szybko. Natomiast do zalet prototypowego projektowania SWD należy zaliczyć: wszystkie nieporozumienia są na bieżąco rozstrzygane, szybkie wykrywanie błędów i braków, zaangażowanie użytkownika, krótki czas opracowania prototypu.

Analiza ryzyka w modelu spiralnym w układzie: planowanie, prototypowanie, konstrukcja, weryfikacja przebiega w następujących fazach:

- a) rozpoczyna się od ustalenia wstępnych wymagań i analizy ryzyka ich realizacji;
- b) na tej podstawie buduje się pierwszy prototyp i tworzy konceptualny plan całości;
- c) po kolejnej analizie ryzyka buduje się następny prototyp i tworzy się wymagania dotyczące oprogramowania;
- d) powstaje plan i w jego wyniku projekt oprogramowania;
- e) kolejny cykl (fazy: a–e) przynosi projekt szczegółowy, oprogramowanie, testy i wdrożenie.

Tworzenie serii prototypów, z których każdy jest rozwiniętą wersją poprzedniego prototypu, jest ważnym aspektem procesu projektowania docelowego SWD.

Rysunek 6.2. Etapy cyklu życia typowego systemu informatycznego



Źródło: Opracowanie własne.

### 6.2.3. Weryfikacja prototypów SWD

Modele cyklu rozwoju SWD można podzielić na następujące rodzaje [Geoffrion 1991]: kaskadowy, pragmatyczny, prototypowania prowizorycznego, prototypowania ewolucyjnego, wielokrotnego użycia, automatycznej syntezy. Weryfikacja obejmuje odniesienie modeli konceptualnych systemu do rzeczywistości oraz określenie i przeprowadzenie pożądanych zmian. Z kolei formalizacja i implementacja polega na odwzorowaniu modelu konceptualnego na model fizyczny ze wskazaniem na: interakcyjne techniki prototypowania, pakiety wspomagane analizą ryzyka, systemy probabilistyczne, analizy wielokryterialne i systemy sztucznej inteligencji.

W rozdziale przedstawiono skale szacunkowe oraz metodę korelacji rangowej do szacowania zgodności ocen ekspertów, jak również statystyczną istotność współczynnika zgodności.

Skale szacunkowe służą do oszacowania wielkości cechy lub właściwości badanego obiektu. Oceny tej może dokonać każdy, kto zna bliżej obiekty będące jej przedmiotem. Wartości na wszystkich skalach można zapisywać za pomocą liczb

— wartości na skalach: dychotomicznej, nominalnej i porządkowej, a także za pomocą innych symboli, np. tekstów.

Wymienione skale mogą być stosowane do oceny SWD przez użytkowników, zarówno interfejsów, jak i innych cech eksploatacyjnych oraz funkcjonalnych systemu. Oceniającymi mogą być także eksperci. Skale mogą stanowić m.in. kryteria klasyfikacji oraz oceny i weryfikacji, jak również pomiaru określonych sytuacji decyzyjnych oraz parametrów je opisujących. Oto przykłady zastosowania skal: nominalnej, porządkowej, przedziałowej oraz ilorazowej.

#### Przykład 6.1

Skalę nominalną obrazują odpowiedzi na pytanie:

*W którym z wymienionych miejsc robi Pan(i) najczęściej zakupy? Proszę wymienić tylko jedno, najczęściej odwiedzane miejsce:*

- 1) Kaufland,
- 2) Piotr i Paweł,
- 3) Carrefour,
- 4) Lidl,
- 5) Leclerc,
- 6) Tesco,
- 7) Auchan,
- 8) Real,
- 9) M1,
- 10) Netto,
- 11) Biedronka.

Skalę porządkową obrazują odpowiedzi na pytanie:

*Jak często robi Pan(i) zakupy w tym Centrum Handlowym?*

- 1) Średnio dwa razy w tygodniu,
- 2) Jeden raz w tygodniu,
- 3) Raz na dwa tygodnie,
- 4) Raz na trzy tygodnie,
- 5) Rzadziej niż raz na trzy tygodnie.

Skalę przedziałową obrazują odpowiedzi na pytanie:

*Ile razy odwiedził(a) Pan(i) Centrum Handlowe?*

- 1) Nie był(e)m w nim ani razu,
- 2) Jeden raz,
- 3) Dwa razy,
- 4) Trzy razy,
- 5) Cztery razy,
- 6) Pięć razy i więcej.

Skalę stosunkową (ilorazową) ilustruje przykład 6.2.

## Przykład 6.2

Proszę się zastanowić nad Centrum Handlowym, w którym się teraz znajdujemy, i ocenić, na ile są istotne dla Pana/i jego elementy. Istotność proszę oceniać według skali punktowej. Proszę przyporządkować punkty poszczególnym elementom tak, aby suma wszystkich ocenionych elementów wynosiła 100 (tablica 6.1).

Tablica 6.1. Przykład skali ilorazowej

Główna hala hipermarketu	30
Lokale gastronomiczne	30
Sklepy w Centrum Handlowym	20
Punkty usługowe w Centrum Handlowym	20
Razem	100

Źródło: Opracowanie własne.

Przykłady poziomów pomiaru klasyfikacji S. Stevensa zamieszczono w tablicy 6.2 [Aczel 2000], a dyferencjału semantycznego oraz skali Likerta w przykładzie 6.3.

Tablica 6.2. Poziomy pomiaru w klasyfikacji S. Stevensa

Poziomy pomiaru	Dopuszczalne operacje matematyczne	Statystyki	Skale podstaw
Nominalny	Różnicowanie, relacja równości, algebra zbiorów	Liczebności, proporcje, test $\chi^2$	Dychotomiczna, wielokategorialna, pozycji
Porządkowy	+ Szeregowanie, relacja większości i mniejszości, algebra relacji porządkujących	Mediana, kwartyle, korelacja rang, testy nieparametryczne	Rang, porównań parami, semantyczna, Likerta, Guttmana
Przedziałowy	+ Dodawanie i odejmowanie, równość przedziałów i różnic	Średnia arytmetyczna, wariancja, stosunki korelacyjne, testy parametryczne	Thurstone'a (dzieli dużą liczbę obiektów tak, by odległości między grupami były jednakowe)
Stosunkowy (interwałowy)	+ Dzielenie i mnożenie, równość stosunków	Średnia geometryczna, współczynnik zmienności	Stałej sumy

Źródło: Opracowanie własne.

Skala dyferencjału semantycznego jest to skala pomiarowa stosowana powszechnie w ocenie postaw, produktów, usług itp. Skala składa się z kilku kategorii (najczęściej pięciu lub siedmiu) oznaczonych liczbami. Bieguny skali (czyli jej po-

czątek i koniec) opisane są antonimami, którymi można określić oceniany obiekt: nudny–ciekawy, tradycyjny–nowoczesny, tani–drogi itp. Dyferencjał semantyczny nazywany jest również skalą semantyczną lub skalą Osgooda.

### Przykład 6.3

Przykładowa skala dyferencjału semantycznego może wyglądać następująco:  
*Czy Pana(i) zdaniem produkt X jest:*

Tani	1	2	3	4	5	6	7	Drogi
------	---	---	---	---	---	---	---	-------

Respondent może wskazać na kategorie skrajne lub wybraną kategorię pośrednią, która najbardziej odpowiada jego ocenie.

Skala Likerta jest wykorzystywana głównie do pomiaru postaw. Jest to jedna z najbardziej znanych skal pomiarowych, stosowana powszechnie w badaniach marketingowych, rynkowych i społecznych. Nazwa skali pochodzi od nazwiska jej twórcy, amerykańskiego socjologa Likerta. Początkowo skala Likerta była znacznie bardziej złożona; obecnie stosuje się jej uproszczoną wersję. Respondentowi przedstawiana jest lista twierdzeń dotyczących ocenianego produktu, usługi lub innego zjawiska, a jego odpowiedzi mierzone są na dwubiegunowej, zazwyczaj pięciopunktowej skali: 1) *zdecydowanie się nie zgadzam*; 2) *raczej się nie zgadzam*; 3) *ani się nie zgadzam, ani się zgadzam*; 4) *raczej się zgadzam*; 5) *zdecydowanie się zgadzam*.

Poszczególnym odpowiedziom mogą być przypisywane wartości numeryczne od 1 do 5, rzadziej od  $-2$  do  $+2$  (zero oznacza odpowiedź neutralną).

Korelacja rangowa oparta jest na badaniu zależności dotyczących zmienności dyskretnej w celu wskazania związków między badanymi cechami [Górniak, Wawnicki 2003]. Ranga to kolejny numer obserwacji, uzyskany po uporządkowaniu obserwacji według ich wartości. Rangi odzwierciedlają porządkowe relacje między poszczególnymi obserwacjami w próbie. W zależności od kolejności porządkowania wartości (rosnącej lub malejącej) większa ranga przypisywana jest większej lub mniejszej wartości (przy sortowaniu rosnącym ranga 1 przypisywana jest najmniejszej wartości w próbie, a największa ranga największej (ostatniej) obserwacji).

**Współczynniki korelacji rangowej**, w odróżnieniu od klasycznego współczynnika korelacji, mierzą monotoniczną zależność między zmiennymi, a nie tylko liniową, np. jeśli jedna ze zmiennych jest rosnącą funkcją drugiej, to korelacja rangowa przyjmuje wartość maksymalną.

Współczynniki korelacji rangowej są bardziej odporne na obserwacje odstające. Najczęściej używanymi współczynnikami korelacji rangowej dla dwóch zmiennych są: współczynnik korelacji rangowej rho Spearmana, tau Kendalla, gamma Krukskala, d Sommersa.



**Metody rangowe** to zbiór metod statystycznych, w których próba jest na wstępie rangowana, tzn. każda wartość każdej cechy jest zastępowana jej pozycją (rangą) na uporządkowanej rosnąco liście wszystkich wartości tej cechy lub przynajmniej niezmienniczych ze względu na operację rangowania danych wejściowych. Operacja taka sprawia, że zachowują się informacje o wzajemnych relacjach między wartościami danej cechy, jednak niezależnie od rozkładów przed rangowaniem rozkład każdej cechy po rangowaniu jest identyczny (jednostajny). Niezależnie od rozkładu, z którego losowana jest próba, każda permutacja rang jest jednakowo prawdopodobna, co jest wykorzystywane w wielu testach statystycznych.

Opisane właściwości sprawiają, że metody rangowe — w przeciwieństwie do metod parametrycznych:

- nie wymagają żadnych założeń dotyczących rozkładu w populacji,
- są odporne na obserwacje odstające,
- mogą przetwarzać zmienne na skali porządkowej.

Metody rangowe można stosować w sytuacjach, gdy metody parametryczne zawodzą. Jeśli populacja ma dobrze określony rozkład (np. normalny) i nie ma obserwacji odstających, metody parametryczne, jako bardziej wyspecjalizowane, dają zwykle lepsze wyniki. Metody rangowe są w tej interpretacji szczególnym przypadkiem metod statystyki nieparametrycznej.

Korelacja rang Spearmana to w statystyce jedna z nieparametrycznych miar monotonicznej zależności statystycznej między zmiennymi losowymi. Ch. Spearman zdefiniował swój współczynnik jako zwykły współczynnik korelacji Pearsona ustalany dla rang zmiennych (stąd nazwa współczynnik korelacji rang). Obecnie stosowanych jest kilka jego wersji, nieznacznie różniących się od siebie. Ich wartości są identyczne w przypadku, gdy obserwacje każdej zmiennej w próbie nie powtarzają się.

Korelacja  $W$  Kendalla [Kowalczyk 2008; Kendall 1948] (znana także jako współczynnik zgodności Kendalla) jest statystyką nieparametryczną, unormowaną wersją statystyki testu Friedmana. Statystyka ta może być używana do sprawdzania zgodności między rankingami pochodzącymi z wielu źródeł, np. ocenami tej samej rzeczy, pochodzącymi od różnych osób. Jej wartości mieszczą się w przedziale od 0 (brak zgodności) do 1 (pełna zgodność).

Mechanizm działania ww. metod ilustruje przykład 6.4.

#### Przykład 6.4

Załóżmy, że grupa klientów była poproszona o uszeregowanie preferencji zakupowych od najbardziej do najmniej lubianej marki produktu; jeśli obliczona dla tego zbioru statystyka  $W$  Kendalla będzie równa 1, wszyscy respondenci zgodnie podali ten sam ranking; jeśli będzie równa 0, prawdopodobnie nie istnieje żadna prawidłowość w odpowiedziach respondentów, a wartości pośrednie odpowiadają mniejszej lub większej zgodności ocen.

Korelacja  $W$  Kendalla zakłada jedynie, że porównywane oceny są zmiennymi porządkowymi lub rangowymi. Nie ma ograniczeń co do maksymalnej liczby obserwacji lub zmiennych. Statystyka  $W$  Kendalla jest często używana w psychometrii do szacowania zgodności sędziów (ekspertów) kompetentnych.

W bazie modeli SWD, w zależności od charakteru związków występujących między zmiennymi opisującymi analizowaną sytuację decyzyjną, mogą być stosowane różne metody rozwiązań opartych, np. na różnego typu statystykach, czyli na takich, które najwierniej odzwierciedlają rzeczywiste fakty, procesy, zjawiska. W tym znaczeniu opisywane metody pomiaru i estymacji mają znaczenie w prototypowaniu SWD. Celem jest jak najlepsze dopasowanie procedury przetwarzania do charakteru rozwiązywanego problemu decyzyjnego.

**Inżynier wiedzy** uczestniczący w procesie pozyskiwania wiedzy próbuje ją wydobyć od eksperta z danej dziedziny za pomocą różnych metod. Biorąc odpowiedzialność za budowę SE, staje przed nim wyzwanie zdobycia wiedzy z określonej dziedziny. Problemy te nie są obce ekspertowi dziedzinowemu, który spotykając się z danymi zadaniami w codziennej pracy, potrafi je skutecznie rozwiązywać. Dlatego tak ważny jest udział eksperta w procesie tworzenia BW SWD. Jednak nawet dysponując pomocą doskonałego eksperta, inżynier wiedzy stoi przed trudnym zadaniem systematycznego gromadzenia tej wiedzy.

Dlatego też rozwijane są formalne metody pozyskiwania wiedzy fachowej. Rozwój tych metod uwarunkowany jest m.in. tym, że:

- ekspert może być osobą niezainteresowaną dzieleniem się wiedzą, pozyskiwaną często w wyniku własnych, wieloletnich i unikalnych doświadczeń;
- ekspert może się czuć zagrożony prawdopodobieństwem odebrania mu przez SE źródła zarobkowania;
- przekazywanie wiedzy jest procesem czasochłonnym i w istotny sposób może być ograniczone brakiem dziedzinowego przygotowania inżyniera wiedzy do współpracy z ekspertem, którego czas pracy jest szczególnie cenny.

## 6.3. Przykłady prostych narzędzi projektowania SWD

W podrozdziale przedstawiono możliwości wykorzystania arkusza kalkulacyjnego Excel do modelowania problemów decyzyjnych i podejmowania decyzji w oparciu o przeprowadzone eksperymenty symulacyjne.

### 6.3.1. Arkusz kalkulacyjny MS Excel

Wykorzystanie symulacji komputerowej w procesie podejmowania decyzji jest korzystniejsze w stosunku do rozwiązań bazujących wyłącznie na metodach analitycznych, a szczególnie w odniesieniu do problemów i systemów o dużej złożoności.

W opinii A.M. Lawa i M.G. McComasa [Law, McComas 2001] model symulacyjny może zastąpić system rzeczywisty w tym sensie, że jego użytkownik może rozważać wiele wariantów działania w systemie rzeczywistym (oceniać zjawiska *ex ante* i wyznaczać najbardziej korzystne sposoby działania w niepewnej przyszłości) bez potrzeby przeprowadzania eksperymentów w systemie rzeczywistym, co zwykle jest nieopłacalne lub nawet niewykonalne (np. ze względu na konieczność zakłócania, w wyniku ingerencji, funkcjonowania systemu rzeczywistego oraz nieporównywalności warunków w kolejnych eksperymentach). Bardzo dobra adaptacja komputerowych modeli symulacyjnych do prowadzenia analizy wrażliwości rozwiązań (reakcji modelu) na zmieniane parametry decyzyjne stanowi kolejny argument przemawiający za stosowaniem symulacji komputerowej. Procedury symulacyjne można przeprowadzić, np. stosując teorię eksperymentu, w której zakłada się, że w celu poznania zależności między wybranymi wielkościami charakteryzującymi obiekt badań (matematyczny model badanej rzeczywistości) wykorzystuje się określony zbiór metod dostępny również w specjalizowanych bibliotekach zawartych w pakiecie MS Excel, a także w oparciu o rejestrację wskaźników przedstawiających stopień ich optymalności (np. poprzez ich porównanie z już przeanalizowanymi wariantami lub z przyjętą wartością progową, powyżej której rozwiązanie kwalifikowane jest do dalszych badań).

Przeprowadzone eksperymenty pozwalają wnioskować o relacjach między zmiennymi modelu, a także dokonać wstępnej weryfikacji wielu alternatywnych strategii (w krótkim czasie) oraz wyselekcjonować spośród nich najodpowiedniejszą w określonych warunkach. Limitowana w określonym czasie liczba wykonanych eksperymentów — uzyskanych efektów konkretnego wariantu decyzyjnego nie daje jednak (w przeciwieństwie do metod analitycznych) gwarancji ustalenia rozwiązania optymalnego. Dlatego też czynione są próby opracowania metod mierzenia i badania nieuchwytnych dotychczas wymiarów działalności przedsiębiorstw, takich jak:

- personel — jego umiejętności oraz doświadczenie;
- relacje z klientami — BD klientów, marka i zaufanie do niej;
- procesy — przebieg działalności wewnątrz organizacji oraz sprzężenia zwrotne z otoczeniem;
- odnawianie i rozwój — czy przedsiębiorstwo wykazuje wzrost, czy rokuje nadzieje na rozwój oraz czy docenia przygotowanie się na przyszłość i prowadzenie badań.

Podjęcie decyzji (służących realizacji zadań, które mają stanowić rozwiązanie problemów występujących w działalności organizacji) musi być poprzedzone wykonaniem analizy decyzyjnej. Potwierdzają to Shim i in. [Shim, Warkentin, Courtney, Power, Sharda, Carlsson 2002], którzy wyodrębnili trzy kategorie działań badawczych prowadzonych podczas podejmowania decyzji:

- 1) postrzeżenie, uświadomienie, systematyzacja i sformułowanie istniejących w przedsiębiorstwie problemów (możliwe do wykonania jedynie dzięki inteligencji, wyobraźni, doświadczeniu i wiedzy analityka-przedsiębiorcy);

- 2) stosowanie opracowanych metod rozwiązywania problemów, zapis wariantów decyzyjnych w języku modeli, tworzenie nowych nieszablonowych rozwiązań w wyniku koncepcyjnej pracy projektanta;
- 3) analizowanie wariantów, symulacja i dokonywanie wyboru spośród postawionych alternatyw, a także analiza wykonalności, optymalizacja, kontrolowanie realizacji (praca o charakterze operacyjnym w znacznym stopniu programowalna).

Wykonywanie analizy decyzyjnej polega na opracowaniu oceny poszczególnych działań i ich wariantów, które powinny prowadzić do osiągnięcia celów formułowanych przez zarząd przedsiębiorstwa. Oceny powinny być oparte na jasno określonych kryteriach (właściwych dla polityki kierownictwa), co pozwoli ustalić, w jakim stopniu poszczególne działania i ich następstwa będą sprzyjały osiągnięciu celów władz firmy.

Celem analizy decyzyjnej jest strukturalizowanie metod poszukiwania rozwiązań optymalnych. Warto podkreślić, że w analizie decyzyjnej mamy do czynienia z dwoma typami zadań: zawierających czynnik losowy (np. ubezpieczenie działalności, jeśli nie wiemy, z jakich odszkodowań będziemy korzystać) i niezawierających czynnika losowego (np. wybór dostawcy na podstawie lokalizacji jego siedziby).

W przypadku problemów decyzyjnych Markowa (procesu stochastycznego, w którym przejście ze stanu poprzedniego do stanu bieżącego zależy wyłącznie od stanu poprzedniego) stan procesu nie zależy zatem od trajektorii stanów, gdyż całkowity niezbędny zasób informacji znajduje się w stanie poprzednim. Tak często dzieje się w systemach sterowania układami techniczno-biologicznymi, w których np. średnia wielkość udoju dziennego mleka od danej krowy w dniu  $n$  decyduje o obliczonej przez SWD i zadawanej przez automat dawce paszy treściwej w dniu  $n + 1$  [Cassandras 1993]. Jednym z zalecanych sposobów rozwiązań jest symulacja systemu rzeczywistego, tak jak np. w automatycznym systemie wykrywania zapalenia wymienia krów mlecznych, w którym na podstawie ustalonych statystycznie związków między wystąpieniem choroby a zmianami temperatury zwierząt (populacji) przewiduje się wystąpienie choroby z odpowiednim wyprzedzeniem u konkretnego zwierzęcia, zapobiegając w ten sposób skażeniu mleka pochodzącego od całego stada krów. W tym znaczeniu w projektowaniu SWD niezbędne jest wykorzystanie w modelach tych systemów konkretnych prawdowości (oszacowywanych statystycznie) w celu uzyskania rozwiązania problemu decyzyjnego.

### 6.3.2. Tablice decyzyjne

**Tablice decyzyjne** (TD) są formą reprezentacji wiedzy, która jest konkurencyjna względem reguł wnioskowania. Najważniejsza różnica dotyczy formatu zapisu. Wiedza w TD jest zakodowana w postaci funkcji, której argumentem jest produkt kartezyjański wartości (stanów) warunków służących do wyznaczenia decyzji, a wartością (przeciwdziedzina) — produkt kartezyjański wariantów decyzyjnych. Istnieją

metody wydobywania reguł decyzyjnych z TD za pomocą zastosowań teorii zbiorów przybliżonych [Boryczka 1988].

Z innego punktu widzenia tablica decyzyjna jest szczególnym rodzajem systemu informacyjnego. W tablicy decyzyjnej zbiór atrybutów podzielony jest na dwa niepuste i rozłączne podzbiory, z których jeden nazywany jest zbiorem atrybutów warunkowych, a drugi — zbiorem atrybutów decyzyjnych. W zastosowaniach praktycznych przyjmuje się, że zbiór atrybutów decyzyjnych jest jednoelementowy.

Niech  $A = \langle U, A \rangle$  będzie systemem informacyjnym oraz:

$$C, D \subseteq A \text{ i } C \neq \emptyset, D \neq \emptyset, C \cap D = \emptyset, A = C \cup D.$$

Czwórkę  $TD = \langle U, A, C, D \rangle$  nazywamy tablicą decyzyjną.

Elementy zbioru  $C$  nazywamy atrybutami warunkowymi, elementy zbioru  $D$  — atrybutami decyzyjnymi. W praktyce dla TD przyjmujemy oznaczenie  $\langle U, A \cup \{d\} \rangle$ , zakres wartości atrybutu  $a$  będziemy oznaczać  $V_a$ .

Przykładem zastosowania TD wykorzystującej logikę wielowartościową jest moduł reprezentacji wiedzy w systemie ekspertowym FINEVA [Matsatsinis, Doumpos, Zopounidis 1997]. Definiując TD w sposób formalny, można ją przedstawić jako uporządkowaną piątkę:

$$TD = \langle U, C, D, V, f \rangle, \quad [6.1]$$

gdzie:

$U$  — niepusty, skończony zbiór, którego elementy stanowią obiekty (uniwersum),

$C$  — zbiór atrybutów warunkowych,

$D$  — zbiór atrybutów decyzyjnych,

$$C, D \subseteq A \text{ i } C \neq \emptyset, D \neq \emptyset, C \cap D = \emptyset, A = C \cup D,$$

$$V = \bigcup_{a \in A} V_a,$$

gdzie:

$V_a$  — dziedzina atrybutu  $a \in A$ ,

$v$  — dziedziny atrybutów  $D$  ( $v \in V_0$ ), czyli klasy decyzyjne,

$f$  — funkcja decyzyjna.

TD służą przede wszystkim do graficznej, bardzo przejrzystej prezentacji decyzji, jaką należy podjąć w zaistniałych warunkach. Abstrahują one od pozostałych elementów procesu decyzyjnego (nie określają sposobu ani adresata decyzji, procesów wprowadzania i wyprowadzania danych). Podstawą budowy TD są związki przyczynowo-skutkowe (jeżeli  $\parallel$ , to  $\parallel$ ). TD (tablica 6.3) jest pewną strukturą opisu zbioru związanych ze sobą reguł decyzyjnych.

Struktura TD (tablica 6.3) składa się z czterech rodzajów pól:

- opisu (wykazu) warunków: zawiera opisy (nazwy) poszczególnych warunków;

- stanu (zapisu) warunków: zawiera wszystkie kombinacje wartości, jakie mogą przyjąć poszczególne warunki;
- opisu (wykazu) czynności (działań): zawiera zestaw wszystkich możliwych wariantów decyzji;
- stanu (zapisu) czynności (działań): zawiera oznaczone warianty decyzji, jakie należy podjąć przy określonej kombinacji wartości warunków.

Tablica 6.3. Struktura TD

	Reguła decyzyjna 1	Reguła decyzyjna 2	Reguła decyzyjna 3	Reguła decyzyjna 4	Reguła decyzyjna 5	Reguła decyzyjna 6
ODCINKI						
WARUNKÓW			POZYCJE			
			warunków			
ODCINKI			POZYCJE			
CZYNNOŚCI			czynności			

Źródło: Opracowanie własne.

TD możemy podzielić na rodzaje:

- ze względu na poziom kompletności: pełne (kompletne), zredukowane (uproszczone), pośrednie;
- ze względu na zapis warunków: proste, rozszerzone, mieszane.

Zasady redukcji TD obejmują następujące działania: eliminowanie reguł logicznie sprzecznych wewnątrznie, łączenie reguł podobnych i grupowe łączenie reguł przez użycie reguły ELSE.

W polu warunków możemy przyjąć, że:

- „T” — wskazuje konieczność spełnienia warunku sformułowanego w odcinku;
- „N” — wskazuje, że warunek odcinka nie może być spełniony;
- „-” — wskazuje, że spełnienie lub niespełnienie warunku z odcinka nie jest istotne;
- „\*” — warunek z odcinka nie jest spełniony, jeżeli jest sprawdzony pewien inny warunek jawny;
- „\$” — warunek z odcinka jest spełniony, jeżeli jest sprawdzony pewien inny warunek jawny.

TD przedstawiona w tabelicy 6.4 zawiera: osiem obiektów będących przedsiębiorstwami ( $P_1 - P_8$ ), trzy atrybuty opisujące sytuację obiektów: wzrost sprzedaży, wzrost liczby reklamacji i wskaźnik fluktuacji kadr oraz decyzję banku, czy udzielić badanemu przedsiębiorstwu kredytu, czy też nie.

**Tabela 6.4. TD dla ustalania możliwości podjęcia decyzji o udzieleniu kredytu**

Atrybuty Obiekt	Wzrost sprzedaży	Wzrost liczby reklamacji	Wskaźnik fluktuacji kadr	Czy udzielić kredytu?
$P_1$	T	N	Niski	T
$P_2$	T	N	Przeciętny	T
$P_3$	T	N	Wysoki	N
$P_4$	N	N	Przeciętny	T
$P_5$	N	N	Niski	T
$P_6$	N	T	Przeciętny	N
$P_7$	N	T	Niski	N
$P_8$	N	N	Wysoki	N

Źródło: Opracowanie własne.

W TD poszczególne opcje są identyfikowane w kolumnach za pomocą grupy odpowiednich wierszy kryterialnych; następnie mają przypisane unikatowe decyzje w grupie wierszy działań. Zapis w języku strukturalnym odwołuje się do naturalnego opisu działań, używając formalnych warunków logicznych (np. IF–THEN–ELSE, SELECT–CASE–WHEN) do zapisu reguł wyboru opcji.

W przykładzie 6.5. przedstawiono formalizację procedury decyzyjnej dotyczącej procesu podejmowania decyzji producenta rolnego.

### Przykład 6.5

Producent rolny posiada plantacje różnych roślin uprawnych ( $P_1 - P_{12}$ ), które potencjalnie mogą zostać zaatakowane przez różne agrofagi, czyli organizmy szkodliwe w produkcji roślinnej (i leśnej). Należą do nich patogeny roślin (grzyby, bakterie, wirusy), szkodniki (zwierzęta) i chwasty (rośliny). Zwalczanie chemiczne jest możliwe tylko podczas bezwietrznej pogody, przy czym wystąpienie deszczu krótko po oprysku znacząco zmniejsza skuteczność zabiegu. Rodzaj zastosowanego środka chemicznego zależy od rodzaju zagrożenia. Proces podejmowania decyzji o podjęciu konkretnego działania można sformalizować za pomocą TD (tabela 6.5).

Tablica 6.5. TD dla ustalania możliwych działań przy zwalczaniu agrofagów

Atrybuty Obiekt	Wystąpienie agrofagów	Możliwość opadów deszczu	Aktualne warunki	Atrybut decyzyjny
$P_1$	Chwasty	Małe	Wietrznie	Oprysk niemożliwy
$P_2$	Patogeny	Małe	Wietrznie	Oprysk niemożliwy
$P_3$	Szkodniki	Małe	Wietrznie	Oprysk niemożliwy
$P_4$	Chwasty	Duże	Wietrznie	Oprysk niemożliwy
$P_5$	Patogeny	Duże	Wietrznie	Oprysk niemożliwy
$P_6$	Szkodniki	Duże	Wietrznie	Oprysk niemożliwy
$P_7$	Chwasty	Małe	Bezwietrznie	Oprysk herbicydami
$P_8$	Patogeny	Małe	Bezwietrznie	Oprysk fungicydami
$P_9$	Szkodniki	Małe	Bezwietrznie	Oprysk insektycydami
$P_{10}$	Chwasty	Duże	Bezwietrznie	Oprysk ryzykowny
$P_{11}$	Patogeny	Duże	Bezwietrznie	Oprysk ryzykowny
$P_{12}$	Szkodniki	Duże	Bezwietrznie	Oprysk ryzykowny

Źródło: Opracowanie własne.

### Przykład 6.6

Przykład dotyczy sklepu oferującego sprzedawcom system prowizji zależnej od ceny, wielkości sprzedaży produktów i pensji pracownika. Sprzedaż produktów droższych niż 500 zł przekraczająca 400 szt. jest premiovana prowizją 15% dla pracownika z pensją do 2000 zł, a 12% przy pensji powyżej 2000 zł. Sprzedaż nie większa niż 400 szt. artykułów droższych niż 500 zł jest premiovana prowizjami odpowiednio 8% i 6%, w zależności, czy pensja nie przekracza 2000 zł, czy jest wyższa. Artykuły w cenie do 500 zł sprzedane w ilościach powyżej 800 szt. są premiovane prowizją 10%, gdy pensja pracownika wynosi do 3000 zł, a 8% przy pensji powyżej 3000 zł. Jeśli sprzedaż produktów w cenie do 500 zł nie przekracza 800 szt., to prowizje wynoszą odpowiednio 8% i 5%, zależnie, czy pensja jest do 3000 zł, czy wyższa.

Proces ustalania prowizji dla sprzedawców można sformalizować za pomocą TD przedstawionej w tablicy 6.6.



Tablica 6.6. TD dla ustalania prowizji należnej sprzedawcom po spełnieniu określonych kryteriów

Cena (zł)	> 500				≤ 500			
Sprzedaż (szt.)	> 400		≤ 400		> 800		≤ 800	
Prowizje/Pensja	≤ 2000	> 2000	≤ 2000	> 2000	≤ 3000	> 3000	≤ 3000	> 3000
5%								T
6%				T				
8%			T			T	T	
10%					T			
12%		T						
15%	T							

Źródło: Opracowanie własne.

W tablicy 6.7 przedstawiono reguły umożliwiające podjęcie decyzji na podstawie przynależności pracownika do jednej z dwóch grup (*E* — pracownik etatowy, *A* — pracownik pracujący na akord). Decyzja dotycząca wysokości płacy podejmowana jest stopniowo (hierarchicznie): najpierw rodzaj pracownika, później (jeśli pierwszy atrybut nie wyczerpał warunków) liczba przepracowanych godzin. Reguły umożliwiają podjęcie działań opisanych w tablicy 6.7 (wyplata pensji, obliczenie płacy według określonych stawek lub stworzenie raportu nieobecności pracownika).

Tablica 6.7. Przykład hierarchicznej TD

Warunki/Akcje	Reguły					
	1	2	3	4	5	6
Rodzaj pracownika	<i>E</i>	<i>A</i>	<i>E</i>	<i>A</i>	<i>E</i>	<i>A</i>
Liczba przepracowanych godzin	< 40	< 40	40	40	> 40	> 40
Wyplacić pensję podstawową	X		X		X	
Obliczyć płacę godzinową		X		X		X
Obliczyć nadgodziny						X
Stworzyć raport nieobecności		X				

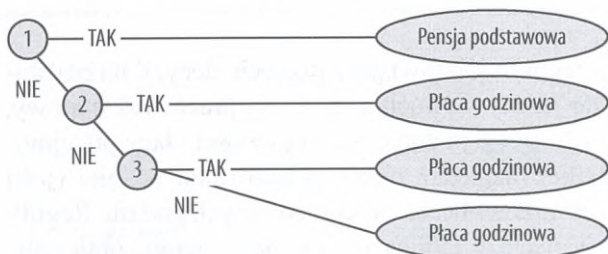
Źródło: Opracowanie własne.

Reguły w TD opisują związki i relacje między faktami, zazwyczaj w prostej implikacji, np.: jeżeli kalkulowany zysk z inwestycji jest korzystniejszy niż zysk z lokaty w banku, to klient inwestuje; jeżeli pracownik przepracował mniej niż 40 godzin, to był kiedyś nieobecny w pracy.

W przypadkach bardziej złożonych warunków opisana reguła staje się mało czytelna, dlatego też wtedy wykorzystuje się drzewa decyzyjne. Jest to graf, którego jeden z wierzchołów został wyróżniony jako korzeń reprezentujący wybrany (najczęściej najbardziej istotny) atrybut, a poszczególne gałęzie reprezentują wartości tego atrybutu. Kolejne węzły drzewa, położone na następnych poziomach, mają przyporządkowane dalsze atrybuty występujące w zadaniu klasyfikacji. Na rysunku 6.3 przedstawiono drzewo decyzyjne ilustrujące zależności opisane we wcześniejszej TD (tablica 6.7).

Drzewa decyzyjne znajdują bardzo szerokie zastosowanie przy rozwiązywaniu wielowarstwowych problemów decyzyjnych. W takich przypadkach wybór danej opcji może spowodować okoliczności, które będą wymagały podjęcia następnej decyzji itd.

Rysunek 6.3. Przykład drzewa decyzyjnego



- 1 – pracownik etatowy  
 2 – przepracowane mniej niż 40 godzin  
 3 – przepracowane równo 40 godzin

Źródło: Opracowanie własne.

### 6.3.3. Scenariusze i animacja zdarzeń

W tym punkcie zaprezentowano przydatność scenariuszy arkusza MS Excel do prowadzenia analizy typu „co, jeśli” (zmiany wartości w komórkach w celu sprawdzenia, jak te zmiany wpłyną na wyniki formuł w arkuszu) oraz tworzenia scenariuszy dla potrzeb porównania wariantów decyzyjnych w zakresie prognozowanych wartości sprzedaży jabłek w różnych krajach i ugrupowaniach krajów w zależności od zmian ceny i przelicznika walutowego euro.

Na rysunkach 6.4–6.6 zobrazowano zastosowanie scenariuszy i zmiennej losowej przy planowaniu wartości produkcji jabłek w walucie euro w różnych krajach i ugrupowaniach krajów w zależności od zmiennej losowo ceny jabłek w zł oraz przelicznika walutowego euro, do którego symulacji zastosowano funkcję los(). Ponadto stworzono 4 scenariusze produkcji jabłek, które w zależności od czynników przyrodniczych kształtują się różnie (normalnie, pesymistycznie, optymistycznie, niespodziewanie).

Rysunek 6.4. Sposób działania scenariuszy w zależności od wartości funkcji los() i wybranego scenariusza („normalny”)

Jabłka		Zmienny losowo przelicznik euro	Zmienna losowo średnia cena jabłek za kg
Production (Mt)	0	3,513297145	1,00 zł
		Zmienna wartość produkcji jabłek w zależności od zmiennej ceny i przelicznika	
	produkcja jabłek - wolumen ilościowy	ilościowy	EURO
Kraj			
Africa	3213710	474 609 390,40 EUR	
European Union (15)	10562697	2 530 320 334,96 EUR	
Netherlands	416	100 303,89 EUR	
Poland	4337427	614 633 033,81 EUR	
	18235019		

Menedżer scenariuszy	
Scenariusze:	<input type="radio"/> normalny <input type="radio"/> pesymistyczny <input type="radio"/> optymistyczny <input type="radio"/> niespodziankowy
	<input type="button" value="Pojaz"/> <input type="button" value="Zamknij"/> <input type="button" value="Dodaj..."/> <input type="button" value="Usuń"/> <input type="button" value="Edytuj..."/> <input type="button" value="Skaluj..."/> <input type="button" value="Podsumowanie..."/>
Komórki zmieniane:	\$D\$5:\$D\$9

Źródło: Opracowanie własne.

Rysunek 6.5. Sposób działania scenariuszy w zależności od wartości funkcji los() i wybranego scenariusza („optymistyczny”)

Jabłka		Zmienny losowo przelicznik euro	Zmienna losowo średnia cena jabłek za kg
Production (Mt)	0	3,811690659	1,05 zł
		Zmienna wartość produkcji jabłek w zależności od zmiennej ceny i przelicznika	
	produkcja jabłek - wolumen ilościowy	ilościowy	EURO
Kraj			
Africa	3535081	461 196 332,96 EUR	
European Union (15)	11607856	2 458 810 304,46 EUR	
Netherlands	457	97 469,18 EUR	
Poland	4771169	597 165 558,48 EUR	
	20058520		

Menedżer scenariuszy	
Scenariusze:	<input type="radio"/> normalny <input type="radio"/> pesymistyczny <input checked="" type="radio"/> optymistyczny <input type="radio"/> niespodziankowy
	<input type="button" value="Pojaz"/> <input type="button" value="Zamknij"/> <input type="button" value="Dodaj..."/> <input type="button" value="Usuń"/> <input type="button" value="Edytuj..."/> <input type="button" value="Skaluj..."/> <input type="button" value="Podsumowanie..."/>
Komórki zmieniane:	\$D\$5:\$D\$9

Źródło: Opracowanie własne.

W ten sposób arkusz MS Excel pozwala symulować różne scenariusze decyzyjne z uwzględnieniem czynników produkcyjnych i wskaźników rynkowych. Oczywiście wolumen produkcji wcale nie musi przesądzać o uzyskiwanych wartościach sprzedaży, bo one są zależne od wskaźników ekonomicznych, takich jak ceny czy przelicznik walutowy [Moore, Weatherford 2001], [Lobel, Brust, Forte 2010].

Rysunek 6.6. Sposób działania scenariuszy w zależności od wartości funkcji los() i wybranego scenariusza („pesymistyczny“)

Jabłka	Zmienny losowo przelicznik euro	Zmienna losowo średnia cena jabłek za kg
Production (Mt)	3,666711106	0,95 zł
0		
Kraj	produkcja jabłek - wolumen ilościowy	Zmienna wartość produkcji jabłek w zależności od zmiennej ceny i przelicznika EURO
Africa	2892339	435 118 830,28 EUR
European Union (15)	9497337	2 319 781 375,27 EUR
Netherlands	375	91 957,96 EUR
Poland	3903685	563 399 924,76 EUR
	16411518	

**Menedżer scenariuszy**

Scenariusze:

- normalny
- pesymistyczny**
- optymistyczny
- niepodzielkowy

Komórki zmieniane:  
\$D\$5:\$D\$9

Pokaż  
Zamknij  
Dodaj...  
Usuń  
Edytuj...  
Skopiuj...  
Podsumowanie...

Źródło: Opracowanie własne.

## 6.4. Zastosowanie systemów zarządzania BD w projektowaniu SWD

W latach 70. XX w. w firmie IBM został opracowany język komunikacji relacyjnych baz danych, który nazwano SQL (ang. *Structured Query Language*). Pierwszą firmą, która włączyła SQL do swojego systemu zarządzania bazą danych, był Oracle. Wprowadzanie języka SQL w produktach innych firm wiązało się z modyfikacjami jego pierwotnej postaci. Utrzymanie jednolitości języka SQL wymagało wprowadzenia standardu.

W roku 1986 SQL stał się oficjalnym standardem, wspieranym przez Międzynarodową Organizację Normalizacyjną ISO (ang. *International Organization for Standardization*) i jej członka — Amerykański Narodowy Instytut Normalizacji ANSI (ang. *American National Standards Institute*). Wczesne wersje specyfikacji (SQL86 i SQL89) były przede wszystkim określeniem wspólnej płaszczyzny, łączącej różne istniejące wówczas produkty i pozostawiały wiele swobody twórcom implementacji. Z czasem jednak systemy komputerowe uległy integracji i rynek zaczął domagać się aplikacji oraz ich funkcji faktycznie współpracujących z wieloma różnymi bazami danych. Pojawiła się potrzeba określenia ściślejszego standardu. Tak powstał standard SQL92, obowiązujący w produktach komercyjnych do dziś, chociaż stopniowo rozwijany. Ostatnie zmiany wprowadzono w 2008 r., kiedy przedstawiono kolejną, szóstą już wersję SQL : 2008.

Głównym celem stosowania języka SQL jest umożliwienie komunikacji z bazą danych. Do jego podstawowych zadań należy zaliczyć: tworzenie, zapełnianie danymi, a następnie użytkowanie i administrowanie bazą danych. Ze względu na sposób wykorzystania wyróżnia się trzy formy języka SQL:

- 1) interakcyjny SQL (autonomiczny), wykorzystywany w celu bezpośredniego pobierania lub wprowadzania danych do bazy;
- 2) statyczny SQL, który nie ulega zmianom i pisany oraz uruchamiany jest wraz z całą aplikacją; statyczny SQL występuje w dwóch odmianach: osadzony — oznacza włączenie kodu SQL do kodu źródłowego innego języka oraz modułowy — kiedy moduły SQL łączone są z modułami kodu w innym języku;
- 3) dynamiczny SQL generowany jest w trakcie pracy aplikacji jako odpowiedź na działania użytkownika.

Wymagania trzech opisanych form SQL różnią się i znajduje to odbicie w wykorzystywanych przez nie konstrukcjach językowych. Zasadnicza część języka pozostaje jednak dla wszystkich form identyczna.

### 6.4.1. Elementy języka SQL

Użycie SQL, zgodnie z jego nazwą, polega na zadawaniu zapytań do bazy danych. Zapytania można zaliczyć do jednego z trzech głównych podzbiorów:

- 1) SQL DML (ang. *Data Manipulation Language*) — instrukcje manipulowania danymi;
- 2) SQL DDL (ang. *Data Definition Language*) — instrukcje definiowania danych;
- 3) SQL DCL (ang. *Data Control Language*) — instrukcje kontroli danych.

Tablica 6.8. Przykładowa tabela PRACOWNICY

Imię	Nazwisko	Pensja	Staż
MARCIN	OLEŃKO	9 000,00 zł	6
PIOTR	OSTAŁOWSKI	4 000,00 zł	6
PAWEŁ	OSTAPIUK	4 000,00 zł	6
JOANNA	OSTROWSKA	1 500,00 zł	5
MARIUSZ	OSTROWSKI	9 000,00 zł	5
ADAM	OWSIANIK	4 000,00 zł	5
MALGORZATA	PACHOLCZYK	1 500,00 zł	5
SYLWIA	PACULA	7 000,00 zł	5
MARIUSZ	PALUCH	6 500,00 zł	6
BŁAŻEJ	PAŁUCHA	7 300,00 zł	6
JOANNA	PARTYCKA	10 000,00 zł	10
AGNIESZKA	PARZONKO	1 500,00 zł	5
KATARZYNA	PASICKA	6 000,00 zł	12
ŁUKASZ	PASKUDZKI	7 500,00 zł	6
PAWEŁ	PAŚNIKOWSKI	4 000,00 zł	5
KATARZYNA	PASZKOWSKA	1 500,00 zł	6

Rekord: 1 z 456 Bez filtru Wyszukaj

Do zilustrowania wybranych instrukcji języka SQL zostanie wykorzystana tabela 6.8 (PRACOWNICY), która składa się z czterech pól (Imie, Nazwisko, Pensja i Staz) opisujących pracownika i zawiera 456 rekordów z danymi.

Instrukcje grupy DML służą do wykonywania operacji na danych, takich jak: wprowadzanie, usuwanie, aktualizacja i usuwanie danych z bazy danych. Najważniejsze polecenia z tego zbioru to:

- SELECT — pobranie danych z bazy,
- INSERT — wprowadzanie danych do bazy,
- UPDATE — aktualizacja danych,
- DELETE — usunięcie danych z bazy.

Przykładem użycia instrukcji SELECT może być wybranie z tabeli 6.8 listy utworzonej ze wszystkich kolumn (\*) oraz tych wierszy, dla których wartość w kolumnie Pensja > 2000 (WHERE). Otrzymany wynik jest sortowany (ORDER BY) malejąco (DESC) według wartości kolumny Staz:

```
SELECT * FROM Pracownicy WHERE Pensja > 2000 ORDER BY Staz DESC.
```

Wynik działania przedstawionej instrukcji prezentuje tabela 6.9, w której są 303 rekordy spełniające wyspecyfikowane w poleceniu SELECT warunki. Tabela 6.9 pokazuje jedynie kilkanaście pierwszych rekordów z listy wyników.

Tablica 6.9. Wynik działania instrukcji SELECT

Imie	Nazwisko	Pensja	Staz
ŚLAWOMIR	ZUMECKI	7 000,00 zł	17
MARIA	NOWAK	10 000,00 zł	17
ROBERT	RYMANOWSKI	6 000,00 zł	17
KRZYSZTOF	LERNACKI	7 000,00 zł	16
ANNA	NARCYZ	5 000,00 zł	15
DANIEL	WALICKI	7 000,00 zł	15
ROBERT	KOWALCZYK	5 000,00 zł	13
ZBIGNIEW	WIŚNIEWSKI	5 000,00 zł	13
JANUSZ	KOWALCZYK	10 000,00 zł	13
WIESŁAW	ZABRZYCKI	5 000,00 zł	12
HELENA	KACPRZYK	7 000,00 zł	12
KATARZYNA	PASICKA	6 000,00 zł	12
ANNA	ZYNDRO	5 000,00 zł	12
JANINA	WYCIER	7 000,00 zł	12
JAN	WYCIER	7 000,00 zł	11
PIOTR	ZABŁOCKI	5 000,00 zł	11

Rekord: 1 z 303 Bez filtru Wyszukaj

Instrukcja INSERT INTO dodaje do tabeli PRACOWNICY wiersz zawierający dane pojedynczego pracownika (VALUES):

```
INSERT INTO Pracownicy (Imie, Nazwisko, Pensja, Staz),
VALUES (,Jan', ,Nowacki', 5500, 1).
```

Nowy rekord zostaje dopisany w tablicy 6.10 na końcu listy jako 457. rekord w tablicy 6.8 (tabela PRACOWNICY).

Tablica 6.10. Wynik działania instrukcji INSERT INTO

Imie	Nazwisko	Pensja	Staz
ŁUKASZ	ŻUKOWSKI	4 000,00 zł	6
SŁAWOMIR	ZUMECKI	7 000,00 zł	17
DOROTA	ŻUROWSKA	1 500,00 zł	6
ILONA	ZWIERZ	1 500,00 zł	6
MARIUSZ	ZYCHOWICZ	9 000,00 zł	6
MARIA	ŻYLIŃSKA	1 500,00 zł	6
ANNA	ZYNDRO	5 000,00 zł	12
JAN	NOWACKI	5 500,00 zł	1

Rekord: 457 z 457 Bez filtru Wyszukaj

Źródło: Opracowanie własne.

Instrukcja UPDATE wszystkim pracownikom w tabeli PRACOWNICY, których staż pracy jest większy niż 6 lat (WHERE), aktualizuje kolumnę Pensja, zwiększając jej wartość o 10%:

```
UPDATE Pracownicy SET Pensja = Pensja * 1.1 WHERE Staz > 6.
```

W tablicy 6.8 (tabela PRACOWNICY) znaleziono 29 rekordów spełniających powyższe kryteria. W tablicy 6.11 zaprezentowano fragment listy wyników, gdzie po lewej stronie znajduje się widok bazy danych przed wykonaniem aktualizacji, a po prawej stronie widać efekt działania instrukcji UPDATE.

Instrukcja DELETE usuwa z tablicy 6.8 (tabela PRACOWNICY — FROM) wszystkie wiersze, dla których w kolumnie Imie znajduje się wartość Jan i w kolumnie Nazwisko występuje wartość Kowalski:

```
DELETE FROM Pracownicy WHERE Imie = ,Jan' AND Nazwisko = ,Kowalski'.
```

Tablica 6.11. Wynik działania instrukcji UPDATE

Imie	Nazwisko	Pensja	Staz
HELENA	KACPRZYK	7 000,00 zł	12
ELŻBIETA	KARTONOWSKA	5 000,00 zł	7
MICHAŁ	KARTOWSKI	7 000,00 zł	7
JANUSZ	KOWALCZYK	10 000,00 zł	13
ROBERT	KOWALCZYK	5 000,00 zł	13
KATARZYNA	KUMELA	1 500,00 zł	7
KRZYSZTOF	LERNACKI	7 000,00 zł	16
AGNIESZKA	MACIĄG	7 000,00 zł	10

Imie	Nazwisko	Pensja	Staz
HELENA	KACPRZYK	7 700,00 zł	12
ELŻBIETA	KARTONOWSK	5 500,00 zł	7
MICHAŁ	KARTOWSKI	7 700,00 zł	7
JANUSZ	KOWALCZYK	11 000,00 zł	13
ROBERT	KOWALCZYK	5 500,00 zł	13
KATARZYNA	KUMELA	1 650,00 zł	7
KRZYSZTOF	LERNACKI	7 700,00 zł	16
AGNIESZKA	MACIĄG	7 700,00 zł	10

Źródło: Opracowanie własne.

Dzięki instrukcjom DDL można operować na strukturach, w których dane są przechowywane, czyli dodawać, aktualizować i usuwać tabele lub bazy. Najważniejsze polecenia tej grupy to:

- CREATE (np. CREATE TABLE, CREATE DATABASE, ...) — utworzenie struktury (np. bazy, tabeli, indeksu);
- DROP (np. DROP TABLE, DROP DATABASE, ...) — usunięcie struktury;
- ALTER (np. ALTER TABLE ADD COLUMN, ALTER TABLE ALTER COLUMN AS) — aktualizacja struktury (np. dodanie kolumny do tabeli, zmiana typu danych w kolumnie tabeli).

Poniższa instrukcja CREATE TABLE tworzy tabelę PRACOWNICY (tablica 6.8), zawierającą kolumny tekstowe Imie i Nazwisko, kolumnę Pensja typu liczba rzeczywista oraz Staz typu całkowitego:

```
CREATE TABLE Pracownicy (Imie varchar(255), Nazwisko varchar(255), Pensja float, Staz int).
```

Instrukcja DROP TABLE usuwa z bazy tabelę o nazwie PRACOWNICY:

```
DROP TABLE Pracownicy.
```

Instrukcja ALTER TABLE dodaje do tabeli PRACOWNICY kolumnę Dzial, jako pole tekstowe o długości 255 znaków:

```
ALTER TABLE Pracownicy ADD Dzial varchar (255).
```

Nowo dodana kolumna jest pusta (tablica 6.12). Chcąc ją zapełnić danymi, należy skorzystać z instrukcji INSERT INTO.



Tablica 6.12. Wynik działania instrukcji ALTER TABLE

Imie	Nazwisko	Pensja	Staz	Dzial
BEATA	ADAMCZYK	1 500,00 zł	6	
MARCIN	ADAMCZYK	9 000,00 zł	6	
AGNIESZKA	ADAMUS	1 500,00 zł	6	
WOJCIECH	ADYNOWSKI	9 000,00 zł	6	
PIOTR	AMBROŻY	6 000,00 zł	5	
MONIKA	ANDRUSZUK	1 500,00 zł	6	
MAREK	ANDRZEJEWSK	9 000,00 zł	6	
ANNA	AUGUSTYNIAK	1 500,00 zł	6	
AGNIESZKA	BAGIŃSKA	1 500,00 zł	6	
KAROLINA	BĄK	1 500,00 zł	6	

Źródło: Opracowanie własne.

DCL można zastosować do nadawania uprawnień do obiektów bazodanych. Najważniejsze polecenia w tej grupie to:

- GRANT — przyznawanie uprawnień;
- REVOKE — cofanie uprawnień, które zostały wcześniej przyznane poleceniem GRANT;
- DENY — bezpośrednie cofanie uprawnień, również tych, które wynikają z przynależności do grupy uprawnień.

Instrukcja GRANT umożliwia wszystkim użytkownikom (TO) czytanie danych (ON) z tabeli Pracownicy:

```
GRANT SELECT ON Pracownicy TO PUBLIC.
```

Jeżeli użytkownik otrzymał dodatkowe uprawnienia, które nie wynikają z jego przynależności do grupy czy roli, to mogą one zostać odebrane instrukcją REVOKE. Poniższa instrukcja prezentuje odebranie użytkownikowi User1 (FROM) prawa do wstawiania, usuwania i modyfikacji danych w tabeli PRACOWNICY (ON):

```
REVOKE INSERT, DELETE, UPDATE ON PRACOWNICY FROM User1.
```

Podobną konstrukcję ma instrukcja DENY, ale działa również w stosunku do uprawnień wynikających z przynależności użytkownika do grupy czy roli. Oto przykład odebrania uprawnień czytania danych w tabeli PRACOWNICY (ON) użytkownikowi User2 (FROM):

```
DENY SELECT ON Pracownicy FROM User2.
```

## 6.4.2. Przykłady zapytań SQL

**Procedury składowane** są zbiorami instrukcji języka SQL zapisanymi pod wspólną nazwą i wywoływanymi jak pojedyncza instrukcja. Procedury składowane umożliwiają:

- wykonywanie dowolnych instrukcji języka SQL,
- przekazywanie parametrów wywołania,
- wywoływanie innych procedur składowanych,
- zwracanie wyników do programu, który dokonał wywołania procedury,
- zwracanie informacji o wykonanej poprawnie lub niewykonanej procedurze.

Procedury składowane są powszechnie używane w celu:

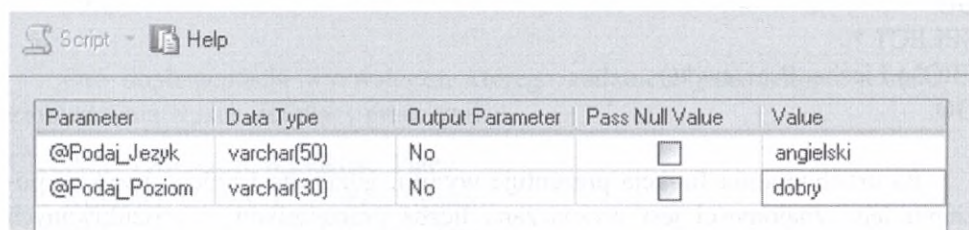
- implementacji reguł logiki biznesowej;
- zabezpieczenia obiektów BD przed nieumiejętnym lub nieuprawnionym dostępem użytkowników;
- ochrony BD przed atakami dokonywanymi na kodzie SQL;
- poprawy wydajności najczęściej realizowanych instrukcji;
- optymalizacji obciążenia sieci dzięki temu, że przesyłaniu podlega wyłącznie wywołanie procedury, a nie cały jej kod SQL'owy.

Zadaniem przykładowej procedury składowanej będzie wyszukiwanie informacji o pracownikach, którzy posługują się konkretnym językiem na określonym poziomie. Język oraz jego poziom będą parametrami tej procedury, czyli wynik działania procedury będzie uzależniony od wartości parametrów, które wprowadzi użytkownik. Oto implementacja tej procedury.

```
CREATE PROCEDURE Język_Poziom
    @Podaj_Język varchar(50),
    @Podaj_Poziom varchar(30)
AS
BEGIN
    SELECT NazwiskoP, ImieP, Stanowisko, Język, Poziom
    FROM [Pracownicy] INNER JOIN [PracJęzyki]
        ON [Pracownicy].[idPrac]=[PracJęzyki].[idPrac]
    WHERE [Język] LIKE @Podaj_Język and [Poziom] LIKE @Podaj_Poziom
    ORDER BY NazwiskoP
END
GO.
```

Na rysunku 6.7 przedstawiono widok parametrów procedury. Wybrano wyszukiwanie pracowników, którzy posługują się językiem angielskim na poziomie do-  
brym.

Rysunek 6.7. Ustalanie wartości parametrów procedury

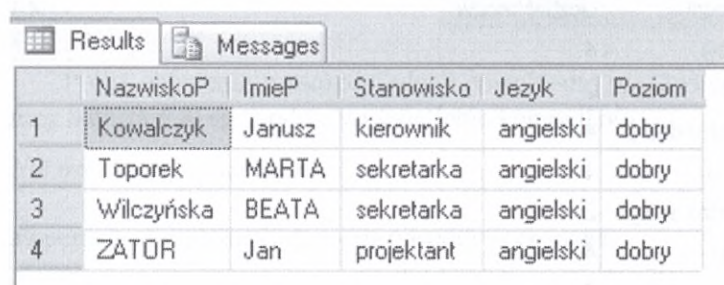


Parameter	Data Type	Output Parameter	Pass Null Value	Value
@Podaj_Jezyk	varchar(50)	No	<input type="checkbox"/>	angielski
@Podaj_Poziom	varchar(30)	No	<input type="checkbox"/>	dobry

Źródło: Opracowanie własne.

Rysunek 6.8 prezentuje wynik uruchomienia procedury przy ustalonych wartościach parametrów.

Rysunek 6.8. Wynik uruchomienia procedury składowanej



	NazwiskoP	ImieP	Stanowisko	Jezyk	Poziom
1	Kowalczyk	Janusz	kierownik	angielski	dobry
2	Toporek	MARTA	sekretarka	angielski	dobry
3	Wilczyńska	BEATA	sekretarka	angielski	dobry
4	ZATOR	Jan	projektant	angielski	dobry

Źródło: Opracowanie własne.

**Funkcjom**, podobnie jak procedurom, można przekazać pewną liczbę parametrów; funkcja ta nie tylko wykonuje operacje, ale także zwraca wynik obliczony na podstawie przekazanych parametrów.

Niech zadaniem będzie przygotowanie **funkcji bazodanowej**, która wyznaczy liczbę pracowników posługujących się różnymi językami na różnych poziomach ich znajomości. Oto implementacja tej funkcji.

```
CREATE FUNCTION Liczba_PracJezyk ()
RETURNS @LiczbaPrac TABLE (Jezyk varchar(50), Poziom varchar(30), LiczbaPracow int)
AS
BEGIN
    INSERT @LiczbaPrac
    SELECT Jezyk, Poziom, COUNT(idPrac)
    FROM PracJezyki
    GROUP BY Poziom, Jezyk
    RETURN
```

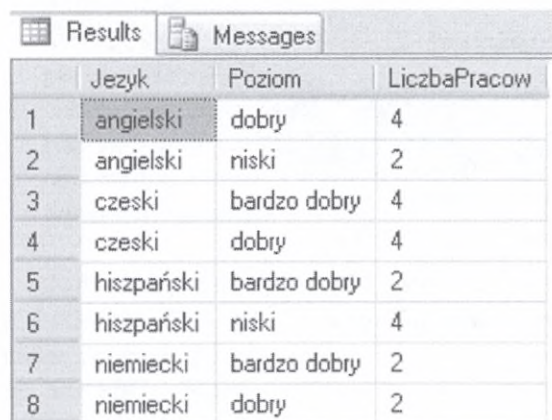
```

END
GO
SELECT *
FROM Liczba_PracJęzyk()
GO.

```

Po uruchomieniu funkcja prezentuje wyniki, gdzie dla każdego języka i poziomu jego znajomości jest wyznaczana liczba pracowników o poszukiwanych umiejętnościach (rysunek 6.9).

Rysunek 6.9. Wynik uruchomienia funkcji bazodanowej



	Język	Poziom	LiczbaPracow
1	angielski	dobry	4
2	angielski	niski	2
3	czeski	bardzo dobry	4
4	czeski	dobry	4
5	hiszpański	bardzo dobry	2
6	hiszpański	niski	4
7	niemiecki	bardzo dobry	2
8	niemiecki	dobry	2

Źródło: Opracowanie własne.

**Wyzwalacze** są specjalnym typem procedur składowanych, które są wykonywane (inaczej: wyzwalane) na skutek zajścia określonego zdarzenia w bazie danych, determinowanego działaniem instrukcji INSERT, DELETE lub UPDATE. Podstawowym zadaniem wyzwalaczy jest zapewnienie integralności danych oraz ich zgodności z regułami logiki biznesowej.

Wyzwalacze umożliwiają m.in.:

- kaskadowe aktualizowanie danych w strukturach tabel powiązanych;
- sprawdzanie poprawności danych przechowywanych w różnych tabelach;
- równoległe sprawdzanie danych zmodyfikowanych w dowolnej liczbie wierszy tabeli;
- wyświetlanie zdefiniowanych przez użytkownika komunikatów o błędach;
- monitorowanie aktywności oraz działań użytkowników w BD;

- modyfikację struktury bazy danych ze względu na wymaganie osiągnięcia 3PN<sup>3</sup>.

Oto implementacja wyzwalacza, którego zadaniem jest niedopuszczenie do zmniejszenia wysokości płacy pracownika.

```
CREATE TRIGGER Wysokosc_Placy
  BEFORE UPDATE ON Pracownicy
  REFERENCING NEW AS Po_Zmianie
  OLD AS Przed_Zmiana
  FOR EACH ROW
  BEGIN
  IF Po_Zmianie.Placa < Przed_Zmiana.Placa THEN
  RAISERROR ,Nie wolno zmniejszac placy!!!;
  END IF;
END.
```

Próba obniżenia wysokości płacy dowolnemu pracownikowi w tabeli Pracownicy (tablica 6.8) spowoduje wyświetlenie komunikatu:

*Nie wolno zmniejszac placy!!!,*

a operacja aktualizacji płacy nie zostanie wykonana.

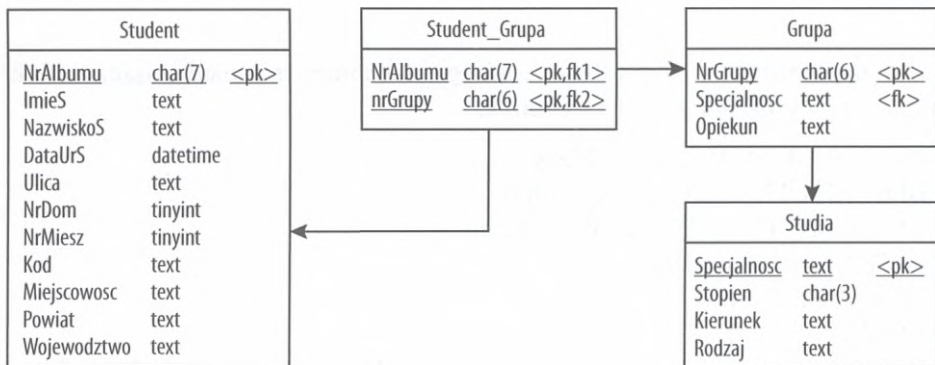
### 6.4.3. Przykład definiowania BD w SWD

Język SQL umożliwi również tworzenie kodu, który uruchomiony w środowisku określonego systemu bazodanowego, utworzy kompletną strukturę BD. Kod taki można wygenerować również zewnętrznie (np. w narzędziu do projektowania i modelowania baz danych), a następnie przenosić go i wykorzystywać w dowolnych sytuacjach. Jest to bardzo wygodna możliwość szczególnie wtedy, kiedy projekt BD jest częścią większego projektu systemu informatycznego (np. SWD) i wymaga realizacji złożonego procesu projektowania, modelowania czy integracji. Na rysunku 6.10 został przedstawiony przykład modelu bazy relacyjnej, składającej się z czterech tabel powiązanych związkami typu 1 : n. Na rysunku znajduje się kod SQL, który uruchomiony w dowolnym środowisku bazodanowym, wygeneruje strukturę bazy danych.

---

<sup>3</sup> 3PN (III postać normalna) — pojęcie znane z procesu normalizacji bazy danych, a oznaczające wyeliminowanie z bazy danych zjawiska nadmiarowości (redundancji) danych. Proces normalizacji obejmuje pięć postaci normalnych, ale przyjmuje się, że minimalnym wymogiem poprawności relacyjnej struktury bazy danych jest właśnie III postać normalna.

Rysunek 6.10. Przykładowy model relacyjnej bazy danych



Źródło: Opracowanie własne.

```

CREATE TABLE Grupa (
  NrGrupy      char(6)    not null,
  Specjalnosc text      not null,
  Opiekun     text       null,
  CONSTRAINT PK_GRUPA PRIMARY KEY nonclustered (NrGrupy)
)
GO
CREATE TABLE Student (
  NrAlbumu    char(7)    not null,
  ImieS       text       null,
  NazwiskoS   text       not null,
  DataUrS     datetime   not null,
  Ulica       text       null,
  NrDom       tinyint    null,
  NrMiesz     tinyint    null,
  Kod         text       not null,
  Miejscowosc text      not null,
  Powiat      text       null,
  Wojewodztwo text      not null,
  CONSTRAINT PK_STUDENT PRIMARY KEY nonclustered (NrAlbumu)
)
GO
CREATE TABLE Student_Grupa (
  NrAlbumu    char(7)    not null,
  NrGrupy     char(6)    not null,
  CONSTRAINT PK_STUDENT_GRUPA PRIMARY KEY (NrAlbumu, NrGrupy)
)
  
```

```
GO
CREATE TABLE Studia (
  Specjalnosc text not null,
  Stopien char(3) null,
  Kierunek text null,
  Rodzaj text null,
  CONSTRAINT PK_STUDIA PRIMARY KEY nonclustered (Specjalnosc)
)
GO.
```

Ponieważ język SQL jest zrozumiały dla wszystkich systemów zarządzania BD i większości współczesnych środowisk programowania, zatem przedstawiony kod jest przenośny i może być wykorzystany w takim narzędziu, w którym będzie implementowana baza danych dla SWD.

## 6.5. Języki programowania wysokiego poziomu w projektowaniu SWD

### 6.5.1. Języki i pakiety inżynierii wiedzy

W tym punkcie omówiono: zintegrowany pakiet sztucznej inteligencji SPHINX, język służący do tworzenia systemów ekspertowych CLIPS, JESS jako w pełni zintegrowane środowisko do tworzenia SE oraz Mandarax jako przykład formułowania decyzji w języku programowania wysokiego poziomu.

SPHINX jest zintegrowanym pakietem sztucznej inteligencji rozwijanym od 1990 r. przez firmę Aitech Artificial Intelligence Laboratory. W skład pakietu wchodzi następujące elementy:

- system PC-Shell — szkieletowy system ekspertowy,
- system Neuronix — symulator sieci neuronowej,
- system CAKE — system komputerowego wspomaganie inżynierii wiedzy,
- system HybRex — system do budowy inteligentnych aplikacji,
- system Predyktor — system prognostyczny,
- system deTreex — indukcyjny system pozyskiwania wiedzy,
- system demoViewer — system do prezentacji aplikacji pakietu AitechSPHINX.

Pakiet wyposażony jest we własny język reprezentacji wiedzy oraz translator. Ma charakter narzędziowy, zatem nie jest zorientowany dziedzinowo. System PC-Shell jest systemem hybrydowym, zawierającym elementy architektury tablicowej. Obecnie jest wykorzystywany w dydaktyce i pracach badawczych na polskich uczelniach [Bojar 2005] oraz w firmach komercyjnych, m.in. w zakładach energetycznych

oraz bankach. Język opisu bazy wiedzy systemu PC-Shell służy do formalnego opisu wiedzy eksperckiej z określonej dziedziny. Językami tego typu posługują się przede wszystkim inżynierowie wiedzy. Ich podstawowym zadaniem jest pozyskanie wiedzy od specjalisty, stosowanie różnych technik wypracowanych przez teorię i praktykę dziedziny SE i zakodowanie jej za pomocą jednego z języków formalnych. Język opisu BW jest dostatecznie przejrzysty i może być wykorzystywany również przez niespecjalistów. Jednak, jak wynika z doświadczeń [Bojar 2005], wymagana jest znajomość informatyki na poziomie języków programowania niższego poziomu, jak np. Delphi, C++ oraz podstawowa znajomość SQL.

Pakiet PC-Shell jest dziedzinowo-niezależnym narzędziem służącym do budowy SE. Typowe obszary zastosowań systemu PC-Shell to:

- systemy doradcze i wspomaganie decyzji,
- dydaktyka (wyższe uczelnie i szkoły średnie).

System może być m.in. wykorzystywany w takich dziedzinach, jak:

- analizy finansowe (ekonomiczne),
- analizy wniosków kredytowych w bankach,
- doradztwo podatkowe,
- wspomaganie planowania i rozliczania procesów produkcyjnych.

Dzięki otwartej architekturze może być łatwo zintegrowany z Systemami Informowania Kierownictwa (SIK), służąc np. do automatycznej analizy wskaźników ekonomicznych, a w technice np. do analizy danych pomiarowych. System nadaje się do budowy zarówno małych, średnich, jak i dużych aplikacji.

System PC-Shell jest systemem o architekturze hybrydowej, tj. łączącej w sobie różne metody rozwiązywania problemów i reprezentacji wiedzy. Posiada m.in. wbudowany, w pełni zintegrowany symulator sieci neuronowej. Struktura tablicowa PC-Shell jest istotną zaletą systemu, gdyż umożliwia podzielenie dużej bazy wiedzy na mniejsze moduły — zorientowane tematycznie, tzw. źródła wiedzy.

Wraz z wersją 2.1 systemu PC-Shell został wprowadzony mechanizm dostępu do konwencjonalnych baz danych. Zastosowane w systemie rozwiązanie wykorzystuje mechanizm ODBC. Mechanizm ten pozwala na dostęp do dowolnego systemu zarządzania bazą danych pod warunkiem posiadania odpowiednich interfejsów dostarczanych przez producentów DBSM. W systemie PC-Shell dostęp do baz danych został zaimplementowany poprzez dodanie wielu nowych instrukcji. Można obecnie realizować następujące operacje:

- inicjalizacja dostępu do BD;
- przesłanie dowolnego zapytania SQL w tzw. trybie bezpośrednim wraz z możliwością pozyskania wyniku działania zapytania;
- sterowanie transakcjami za pomocą odpowiednich instrukcji programowania.



Innym językiem służącym do projektowania SE jest język CLIPS, który został opracowany przez NASA Johnson Space Center. Jego zastosowania są znane m.in. w rozpoznawaniu obrazów, rozumieniu scen itp. CLIPS jest znacznie prostszy w porównaniu z językiem PROLOG czy LISP. Mechanizmy wewnętrzne tego języka realizują wnioskowanie wprzód. Program w języku CLIPS stanowi bazę wiedzy złożoną z faktów i reguł. Składnia tego języka jest podobna do składni języka LISP — wszystkie wyrażenia symboliczne są objęte nawiasami. Dalsze rozszerzenia i ulepszenia, a w szczególności dodanie możliwości pisania programów proceduralnych i obiektowych, przekształciły CLIPS w wersji 6.1 z narzędzia treningowego w pełnowartościowe narzędzie do projektowania i oprogramowania SE. Fakt że CLIPS służy do reprezentacji wiedzy i stanowi spójne narzędzie jej przetwarzania, wspierające trzy różne paradygmaty: regułowy, obiektowy i proceduralny, zadecydowały o sukcesie i rozpowszechnieniu tego języka.

Programowanie oparte na regułach pozwala reprezentować wiedzę jako heurystyki lub ogólne zasady (reguły) oparte na doświadczeniu, które podają zbiór działań, jakie należy wykonać w danej sytuacji. Programowanie obiektowe pozwala modelować złożone systemy jako modularne komponenty (które można łatwo użyć ponownie do modelowania innych systemów lub do tworzenia nowych komponentów). Możliwości programowania proceduralnego są podobne, jak w językach C, Pascal, Ada i LISP.

CLIPS jest łatwo przenaszalny (gwarantuje osiągnięcie przenośności i szybkości), ponieważ został napisany w języku C, co pozwoliło na jego instalację na wielu różnych komputerach bez zmian kodu (IBM PC, Macintosh, VAX11/780, Sun 3/260). Może być przenoszony na dowolny system posiadający kompilator języka ANSI C. Przenośność zapewnia dostarczany z systemem CLIPS kod źródłowy, który może być modyfikowany stosownie do potrzeb użytkownika. Kod w języku CLIPS można wbudować wewnątrz kodu proceduralnego, wywoływanego jako podprogram i zintegrowanego z takimi językami, jak: C, C++, FORTRAN i ADA, co zapewnia integrację i rozszerzalność.

Istnieją również ściśle zdefiniowane zasady tworzenia rozszerzeń języka CLIPS. Standardowa wersja języka CLIPS zawiera interakcyjne środowisko opracowywania programu, obejmujące zintegrowany edytor, narzędzia do uruchamiania oraz natychmiastową pomoc, co tworzy warunki do interakcji z użytkownikiem. CLIPS zawiera wiele cech wspierających weryfikację i walidację SE: modularne projektowanie i podział BW, statyczną i dynamiczną kontrolę atrybutów pól i argumentów funkcji. Funkcjonalnością CLIPS jest także analiza semantyczna wzorcowych reguł w celu ustalenia, czy niespójności mogłyby przeszkodzić w odpaleniu reguły lub wygenerować błąd.

Regułowy system ekspertowy zakodowany w języku CLIPS oparty jest na koncepcji wnioskowania wstępującego. Oznacza to, że zadaje się pewne dane, które po dopasowaniu do przesłanek pozwalają na sformułowanie wniosków. SE napisany w języku CLIPS jest zatem programem sterowanym danymi, ponieważ fakty

i obiekty są tymi danymi, które stymulują wykonanie poprzez maszynę wnioskującą. Dokumentacja, kody źródłowe i binarne języka CLIPS dostępne są bez opłat na wielu serwerach sieci Internet.

Celem narzędzia (metody) Mandarax jest zintegrowanie procesów przyczynowo-skutkowych za pomocą języka JAVA. Mandarax jest kompilatorem reguł (w miejsce reguł wnioskowania) do standardu kodu JAVA. Wykorzystuje on także specyficzny język DSL dla potrzeb definiowania reguł. Język DSL wspomaga proces integracji definicji reguł z językiem JAVA. DSL jest wykorzystywany m.in. do budowy języków, z których mają korzystać eksperci w danych dziedzinach. Należy wyjaśnić, iż programiści, projektanci (ze względu na inne kompetencje) nie zagłębiają się w dane zagadnienie problemowe (np. z obszaru przemysłu chemicznego, metalurgicznego, elektromechanicznego czy marketingu, sprzedaży wyrobów), a tylko po konsultacji z ekspertami konkretnej dziedziny budują język, który ci eksperci wykorzystają do rozwiązywania problemów decyzyjnych. DSL może być także wykorzystywany do tworzenia: testów oprogramowania, zapytań, aby wyciągnąć dane z różnego typu zbiorów zarządzanych różnymi metodami. Najważniejsze decyzje, jakie należy podjąć, to wybór składni języka, z którego będziemy korzystać. Składnia ta powinna wiernie wyrażać problemy oraz obiekty, które chcemy opisać w danym języku. Od tego zależy spójność tworzonych w DSL czy w środowisku Mandarax aplikacji z potrzebami decyzyjnymi zarządzających.

Mandarax jest infrastrukturą przeznaczoną do definiowania, zarządzania oraz tworzenia reguł BW, opartą na wnioskowaniu wstecznym. Mandarax pozwala tworzyć SWD wykorzystujące zaawansowane i inteligentne systemy przechwytywania danych ze zróżnicowanej software'owo i hardware'owo przestrzeni wirtualnej, udostępniając ważne w procesie wspomagania decyzji parametry, takie jak np.: informacje giełdowe, przeliczniki walutowe, normy produkcyjne, notowania sprzedaży i inne.

System Mandarax jest bezpłatny i typu open source. Jest on objęty powszechną licencją publiczną GNU — licencją wolnego i otwartego oprogramowania. Może być wykorzystywany w projektach zarówno typu open source, jak i komercyjnych [Die-trich 2003].

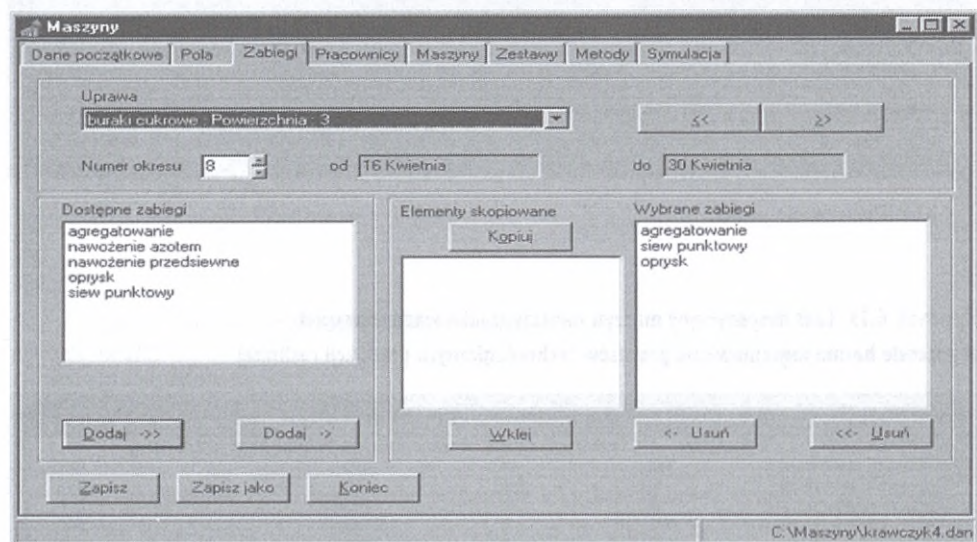
### **6.5.2. Przykład formułowania decyzji w języku programowania wysokiego poziomu**

Język Delphi jest stosowany do formułowania i wyboru wariantów decyzyjnych w modelu przedziałowym odzwierciedlającym dynamikę procesu wykonywania operacji — zabiegów technologicznych na polach uprawnych. Dzięki dostępności obiektów projektantom aplikacji systemu symulacyjnego udało się zbudować prosty w obsłudze interfejs oraz procedury przetwarzania i raportowania wyników, umożli-

wiające użytkownikowi projektowanie własnego modelu przedsiębiorstwa i procesów w nim zachodzących za pomocą danych zagregowanych, opcjonalny wybór wielu parametrów decyzyjnych oraz uzyskiwane rozwiązania w postaci, np. informacji o nadwyżkach lub niedoborach czasu dyspozycyjnego poszczególnych maszyn, co decyduje o prawidłowym harmonogramowaniu zabiegów agrotechnicznych oraz właściwej organizacji pracy.

Rysunek 6.11 obrazuje sposób wykorzystania zagregowanych danych do definiowania procesu wykonywania zabiegów agrotechnicznych pod określoną uprawę, w określonym okresie roku i za pomocą określonych zestawów maszynowych.

**Rysunek 6.11. Sposób definiowania procesów technologicznych produkcji roślinnej za pomocą programu napisanego przy wykorzystaniu języka Delphi**

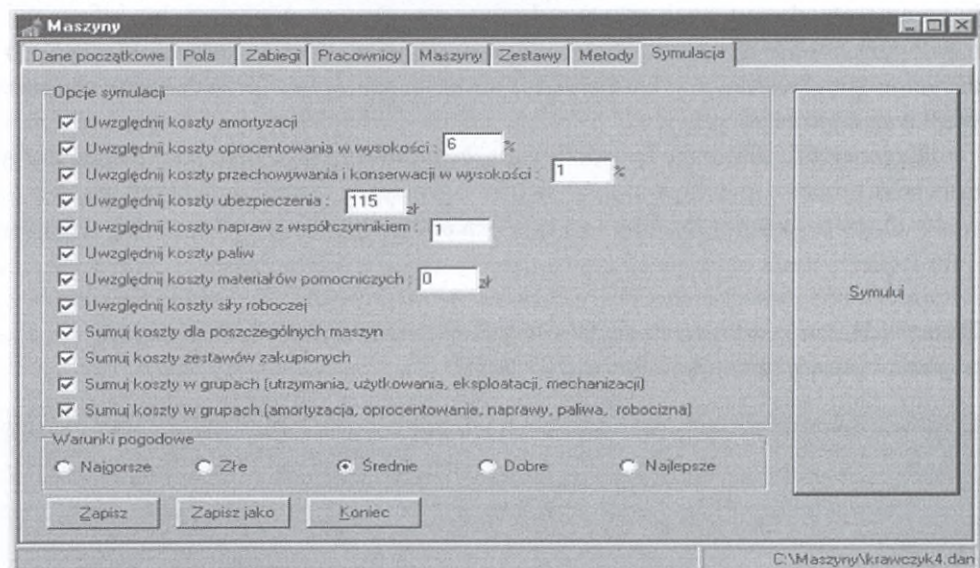


Źródło: Opracowanie własne.

Rysunek 6.12 przedstawia z kolei sposób raportowania wyników w zależności od przyjętych przez użytkownika SWD parametrów decyzyjnych, np. warunki pogodowe „średnie” i inne.

Rysunek 6.13 ilustruje uzyskany rezultat ważny w procesie harmonogramowania procesów ze względu na długość tzw. czasów dyspozycyjnych różnego typu sprzętu rolniczego.

Rysunek 6.12. Opcjonalny sposób raportowania wyników w zależności od przyjętych przez użytkownika SWD parametrów decyzyjnych



Źródło: Opracowanie własne.

Rysunek 6.13. Czas dyspozycyjny maszyn rolniczych jako ważny parametr w procesie harmonogramowania procesów technologicznych produkcji roślinnej

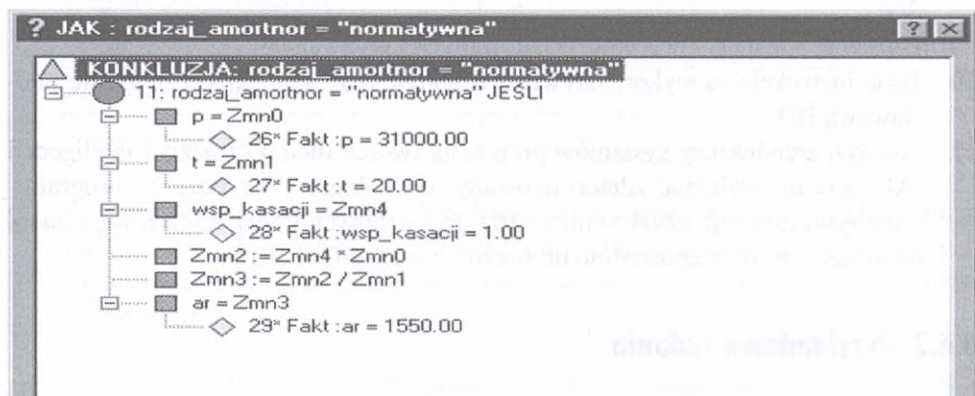
	A	B	C	D	E
1	Maszyny i p	Okresy	Czasy_nak	Czasy dysp	Różnica czasów
2	opryskiwac	10	4	95	91
3	rozsiwacz	7	19	90	71
4	rozsiwacz	9	3	90	87
5	rozsiwacz	18	6	100	94
6	rozsiwacz	16	1	95	94
7	rozrzutnik	20	23	90	67
8	przyczepa	15	53	110	57
9	przyczepa	17	37	115	78
10	przyczepa	22	2	90	88
11	ladowacz C	19	23	95	72
12	dmuchawa	19	19	95	76
13	plug_U023	19	19	95	76
14					
15					
16					
17					
18					
19					

Źródło: Opracowanie własne.

### 6.5.3. Przykład zastosowania pakietów inżynierii wiedzy w realizacji SWD

Rysunek 6.14 pokazuje, w jaki sposób, przy wykorzystaniu pakietu inżynierii wiedzy SPHINX modułu PC-Shell, przebiega proces podejmowania decyzji o wyborze metody obliczania amortyzacji i sposobie kalkulacji jej wartości w odniesieniu do ciągnika rolniczego analizowanego przedsiębiorstwa rolnego.

Rysunek 6.14. System wspomagania decyzji pokazuje, za pomocą jakich parametrów została zweryfikowana hipoteza



Fakt: ar — fakt bazy wiedzy (amortyzacja ustalona metodą normatywną),  
 p — cena maszyny,  
 t — liczba lat użytkowania maszyny.

Źródło: Opracowanie własne.

Moduł objaśniający dokładnie pokazuje użytkownikowi systemu, jak powstają parametry wyjściowe decydujące o uzyskanym rezultacie końcowym, czyli o kosztach mechanizacji procesów technologicznych produkcji roślinnej. Przedstawione na rysunku 6.13 atrybuty i fakty bazy wiedzy oraz reguły typu „if then” pozwalają właściwie skalkulować koszty użycia ciągnika rolniczego w specyficznych warunkach badanego przedsiębiorstwa.

## 6.6. Pytania i zadania kontrolne

### 6.6.1. Pytania kontrolne

1. Jaką istotną funkcję dla użytkownika systemu SWD pełni moduł wyjaśniający i dlaczego w odróżnieniu od wcześniejszych narzędzi wspomagających decyzje jest ona tak ważna?

2. Wymień wady i zalety prototypowego projektowania SWD.
3. Jak można podzielić modele cyklu rozwoju SWD?
4. Jak można wykorzystać funkcję scenariuszy MS Excel w planowaniu przedsięwzięć inwestycyjnych w przedsiębiorstwie przy jednoczesnym zastosowaniu funkcji z grupy finansowych?
5. Do jakich zastosowań służy język SQL?
6. Wymień najważniejsze rozszerzenia podstawowej struktury języka SQL.
7. Na jakie trzy podstawowe grupy dzielone są instrukcje języka SQL? W każdej z grup wymień ich przedstawicieli.
8. Porównaj własności procedur składowanych i funkcji bazodanowych języka SQL.
9. Porównaj własności procedur składowanych i wyzwalaczy języka SQL.
10. Jakie instrukcje są wykorzystywane w kodzie SQL tworzącym strukturę podstawową BD?
11. Jaki typ architektury systemów proponują twórcy metod sztucznej inteligencji (AI), aby uniezależnić zdeterminowany technologią i narzędziami programistycznymi interfejs użytkownika i BD od logicznych i pojęciowych zagadnień, odnoszących się bezpośrednio do rozwiązywanego problemu?

## 6.6.2. Przykładowe zadania

1. Dla tabeli Książka napisz instrukcję SQL, która wybiera kolumny Tytuł, Autor i RokW dla każdej pozycji należącej do Dziedzina typu 'historia' lub 'dramat'.

Autor	Tytuł	RokW	Dziedzina
ANTONY BEEVOR	Berlin 1945	2009	dramat
KAROL BUNSCH	Odnowiciel	2003	historia
KAROL BUNSCH	Zdobycie Kolobrzegu	2003	historia
CZESLAW MILOSZ	Wiersze ostatnie	2006	poezja
JAN KOCHANOWSKI	Fraszki. Piesni. Trens	2004	poezja

2. Napisz instrukcję SQL, która zmienia w tabeli Książka wartość kolumny Dziedzina z wartości 'dramat' na 'powieść'.
3. Napisz instrukcję SQL, która usuwa z tabeli Książka wszystkie rekordy, dla których w polu Dziedzina występuje wartość 'poezja'.
4. Jaki jest wynik działania następującej instrukcji SQL:

```
SELECT Dziedzina, Count(Tytuł)
FROM Książka
GROUP BY Dziedzina.
```

5. Jakie warunki/akcje, reguły dodałbyś do poniższej tablicy decyzyjnej, aby przy kwalifikacji pracowników do jednej z dwóch grup (*E* — pracownik etatowy, *A* — pracownik pracujący na akord) ocenić trafniej jakość pracy pracowników?

Warunki/Akcje	Reguły					
	1	2	3	4	5	6
Rodzaj pracownika	<i>E</i>	<i>A</i>	<i>E</i>	<i>A</i>	<i>E</i>	<i>A</i>
Liczba przepracowanych godzin	< 40	< 40	40	40	> 40	> 40
Wypłacić pensję podstawową	X		X		X	
Obliczyć płacę godzinową		X		X		X
Obliczyć nadgodziny						X
Stworzyć raport nieobecności		X				

6. Zastosuj funkcję scenariuszy MS Excel do wygenerowania różnych prognoz wartości sprzedaży kawy w zależności od kształtowania się jej cen w różnych kwartałach na różnym poziomie. Utwórz w ten sposób przynajmniej 4 różne scenariusze.

Kwartaly	Data zamówienia	Pracownik	Kod firmy	Towar	Kategoria	Cena	Ilość	Wartość
II_kw	1-05-2010	Górecki	KONIG	Jamaican Blu Kawa		97,2	50	4860
I_kw	10-02-2010	Halski	RATTL	Jamaican Blu Kawa		97,2	50	4860
I_kw	10-02-2010	Urbaniak	FOLIE	Jamaican Blu Kawa		97,2	50	4860
I_kw	18-02-2010	Grabowski	FRANK	Jamaican Blu Kawa		97,2	50	4860
IV_kw	22-11-2010	Zieliński	HILAR	Jamaican Blu Kawa		97,2	50	4860
I_kw	17-03-2010	Halski	NORTH	Jamaican Blu Kawa		97,2	50	4860
I_kw	31-03-2010	Zieliński	OCEAN	Jamaican Blu Kawa		97,2	50	4860
II_kw	5-04-2010	Lipiński	ANTON	Jamaican Blu Kawa		97,2	50	4860
II_kw	5-05-2010	Dąbrowski	HUNGR	Jamaican Blu Kawa		97,2	50	4860

# Weryfikacja i wdrożenie SWD

Efektywne wdrożenie SWD w przedsiębiorstwie wymaga właściwego przygotowania i przeprowadzenia tego procesu, w skład którego wchodzi następujące zagadnienia [Gołoś 2010, s. 236]: wybranie odpowiedniej metodyki wdrożenia, określenie budżetu, zaplanowanie poszczególnych czynności procesu wdrożenia, zdefiniowanie harmonogramu wdrożenia, przydział zasobów materialnych i niematerialnych niezbędnych do realizacji wdrożenia oraz oszacowanie czynników i ograniczeń mogących mieć wpływ na przebieg procesu wdrożenia.

Jednym z krytycznych elementów warunkujących sukces wdrożenia jest wybór sprawdzonej metodyki. W przypadku SWD powinna być to metodyka stosowana przy implementacji średniej wielkości systemów informatycznych w warunkach przedsiębiorstwa dysponującego wiedzą i doświadczeniem zgromadzonym przy wdrażaniu innych systemów dziedzinowych (np. ERP, CRM, SCM), dostosowaną do integracyjnych cech modelu systemu informacyjnego przedsiębiorstwa. W rozdziale przedstawiono dwie metodyki kluczowe dla efektywności wyboru i stosowania SWD: metodykę wdrożenia SWD jako systemu informatycznego oraz metodykę wielokryterialnego wyboru systemu informatycznego klasy SWD spośród dostępnych rozwiązań.

## 7.1. Metodyka wdrożenia systemu informatycznego klasy SWD

Od początków wdrażania pierwszych systemów i złożonych rozwiązań informatycznych zaobserwowano pojawianie się problemów z realizacją projektów zgodnie z zakładanym zakresem, budżetem i harmonogramem. Zjawisko to występuje do dnia dzisiejszego i jest na tyle powszechne, że powstało wiele publikacji i analiz poświęconych temu zagadnieniu. Przykładem może być firma Standish Group [Standish Group 1994–2009], która od 1994 r. systematycznie publikuje statystyki sukcesów i niepowo-



dzeń wdrożeń informatycznych w cyklu corocznych raportów. Statystyki te są punktem wyjścia analizy ryzyka wdrożeń produktów informatycznych. **Ryzyko w projekcie informatycznym** to prawdopodobieństwo wystąpienia zdarzenia, które niekorzystnie wpłynie na proces jego realizacji. Przyczyn niepowodzenia może być wiele:

- nieprawidłowo przeprowadzona analiza projektowa,
- nieprecyzyjne sformułowanie wymagań projektowych,
- niepełne i niespójne wykonanie projektu,
- „utrata” pracowników zaangażowanych w projekt,
- problemy z utrzymaniem harmonogramu,
- problemy z utrzymaniem budżetu.

Jednym z ważniejszych elementów ograniczających ryzyko przedsięwzięcia jest sprawdzona metodyka wdrożenia, rozumiana raczej nie jako jednolity algorytm, ale zespół dobrych praktyk, wspierających zespół projektowy i wdrożeniowy przez cały okres jego działalności, aż do efektywnego wdrożenia systemu informatycznego. W przypadku SWD metodyka ta powinna być dostosowana do integracyjnych cech modelu systemu informacyjnego przedsiębiorstwa.

### 7.1.1. Opracowanie docelowego modelu systemu informacyjnego przedsiębiorstwa

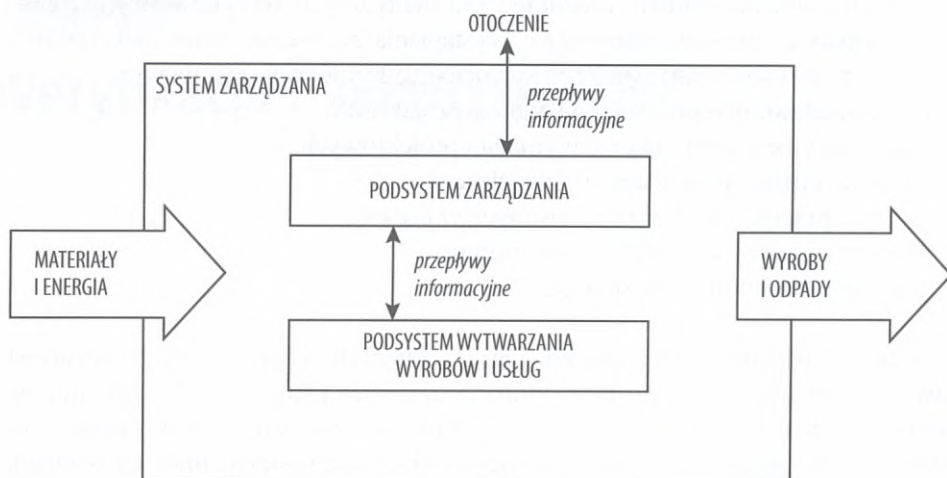
**Model systemowy przedsiębiorstwa** można przedstawić jako jednostkę, która posiada wejścia i wyjścia [Senczyzna 2000; Klonowski 2004] (rysunek 7.1).

Działalność wytwórcza, przetwórcza i usługowa przedsiębiorstwa powoduje, że wejściowy strumień materiałów i energii jest przekształcany w strumień wyjściowy wyrobów oraz odpadów. Nie byłoby to możliwe bez sprawnej komunikacji podsystemu zarządzania z podsystemem wytwarzania oraz systemu przedsiębiorstwa z jego otoczeniem. Obsługę tych kontaktów zapewniają przepływy informacyjne, które z kolei składają się na strukturę systemu informacyjnego przedsiębiorstwa.

W dziedzinie organizacji i zarządzania system informacyjny przedsiębiorstwa stanowi obraz komunikacji zachodzącej między jednostkami organizacyjnymi oraz wejściami i wyjściami przedsiębiorstwa. Typ działalności prowadzonej przez przedsiębiorstwo, określone rozwiązania organizacyjne i wymagania otoczenia (kontra-hentów) przedsiębiorstwa warunkują strukturę systemu informacyjnego. Natomiast doświadczenia zebrane w procesie analizy systemów informacyjnych w wielu przedsiębiorstwach służą do opracowywania wzorcowych rozwiązań takich systemów.

**Funkcje systemu informacyjnego** w przedsiębiorstwach i ogólnie rozumianych jednostkach organizacyjnych są realizowane za pomocą systemów informatycznych. W tym celu wykonuje się analizy oraz modelowanie systemu informacyjnego, gdzie znajdują zastosowanie narzędzia informatyki. W efekcie otrzymujemy projekt systemu informatycznego, który zawiera specyfikację oprogramowania oraz architekturę systemu komputerowego.

Rysunek 7.1. Model systemowy przedsiębiorstwa



Źródło: Opracowanie własne na podstawie: [Klonowski 2004].

**Proces projektowania** może być realizowany dzięki różnym podejściom metodycznym, dobieranym ze względu na dwa podstawowe kryteria [Kisielnicki 2008, s. 150]:

1) czas:

- projektowanie diagnostyczne — punktem wyjścia jest obecny, zdiagnozowany stan przedsiębiorstwa,
- projektowanie prognostyczne — punktem wyjścia jest wizja przedsiębiorstwa w przyszłości, a stan obecny nie jest diagnozowany;

2) sekwencja wykonywania działań:

- projektowanie kaskadowe — wszystkie działania są wykonywane sekwencyjnie i chronologicznie, bez zakładania możliwości powrotu do etapów już zakończonych, co znacząco ogranicza elastyczność tej metodyki,
- projektowanie ewolucyjne — podejście nadążne, reagujące na zmieniające się w czasie wyznaczone zadania działania w celu zapewnienia wysokiej aktualności produktu końcowego,
- projektowanie przyrostowe — zakłada, że projekt będzie realizowany etapami, często przez różne zespoły projektantów, co ogranicza liczbę jednocześnie zaangażowanych w projekt osób; poszczególne etapy są spinane w spójną całość za pośrednictwem: przygotowania wymagań i analizy (realizowanych na początku projektu) oraz testowania i wdrożenia (wykonywanych na końcu projektu),
- projektowanie spiralne — projekt jest realizowany cyklicznie, a w każdym cyklu powstaje prototyp systemu; założeniem jest, że każdy kolejny cykl generuje jego lepszą wersję, aż do powstania produktu końcowego.

Praktyka wskazuje, że zazwyczaj nie wykorzystuje się jednej wybranej procedury projektowania, ale ich mieszankę uzależnioną od typu produktu końcowego i ogólnej sytuacji projektowej [Kisielnicki 2008, s. 156]. Przykładowo, kiedy powstaje projekt typowy, w którym krytyczny jest czas i koszt przedsięwzięcia, zazwyczaj wybiera się podejście diagnostyczno-kaskadowe. W przypadku projektów złożonych, nowatorskich i trudnych w realizacji rekomendowane jest podejście prognostyczno-spiralne. Wieloletni rozwój projektowania, wytwarzania i zastosowań systemów informatycznych spowodował, że metody analityczne informatyki stanowią podstawowe narzędzia modelowania systemów informacyjnych.

## 7.1.2. Opracowanie strategii i planu wdrożenia SWD

**Wdrożenie systemu SWD** może przebiegać w trzech różnych wariantach jako wdrożenie [Kisielnicki, Sroka 2005, s. 163–193]: całościowe, cząstkowe lub równoległe.

**Wdrażanie całościowe** to implementacja nowego systemu przy równoczesnej rezygnacji z eksploatacji dotychczas stosowanego. Takie podejście jest najwygodniejsze i zazwyczaj najtańsze, ale obarczone jest też sporym ryzykiem, polegającym przede wszystkim na braku możliwości korzystania ze starego systemu, kiedy nowy nie jest jeszcze w pełni przetestowany i sprawdzony.

**Wdrażanie cząstkowe** niesie ze sobą mniejsze ryzyko, ale może być dla organizacji bardziej kosztowne i sprawiać większe problemy organizacyjne. Niemniej w przypadku złożonych i obszernych rozwiązań informatycznych to właśnie strategia wdrażania cząstkowego jest zalecana. Istotne jest wtedy podzielenie całego projektu na części, które mogą być niezależnie wdrażane. Korzystne jest, jeżeli każda część stanowi integralną całość, która może być testowana i oceniana przez użytkownika, zanim dojdzie do wdrożenia kolejnych części systemu. Wtedy taki proces wdrażania ma również walory uczenia się i lepszego zrozumienia potrzeb użytkowników, a jednocześnie stwarza szansę na przyspieszenie realizacji kolejnych etapów wdrożenia.

W celu skorzystania z metody konieczne jest dokonanie podziału projektu:

- w ujęciu przestrzennym — wdrażanie w każdym oddziale firmy będzie przebiegało niezależnie;
- w ujęciu funkcjonalnym — każdy moduł funkcjonalny systemu będzie wdrażany oddzielnie.

Bezpieczne, ale najbardziej kosztowne jest **wdrażanie równoległe**. Zgodnie z zaleceniami tej strategii stary system funkcjonuje dopóty, dopóki nowy nie zostanie w pełni wdrożony do eksploatacji. Pracują więc dwa równoległe systemy, a zatem i dwa zespoły pracowników albo ci sami pracownicy wykonują swoje zadania w dwóch wariantach — w starym i nowym systemie. Jest to czasem organizacyjnie zbyt trudne do realizacji.

Bez względu na wybrany typ strategii wdrożenia kierownictwo zespołu wdrożeniowego musi zapewnić [Kisielnicki 2008, s. 211]:

- dopływ zasobów umożliwiających realizację wdrożenia;
- dobór metod, procesów i procedur realizacji wdrożenia;
- określenie zasad i sekwencji ich wykonania;
- przydział ról, zadań i zakresu odpowiedzialności członkom zespołu wdrożeniowego;
- wskazanie miejsca realizacji poszczególnych zadań;
- przygotowanie harmonogramu wdrożenia;
- kontrolę wykonania harmonogramu wdrożenia;
- zdefiniowanie kryteriów oceny wdrożenia oraz weryfikacji osiągniętych celów globalnych i częściowych;
- przygotowanie taktycznego i operacyjnego planu wdrożenia.

Realizacja wdrożenia zgodnie z przedstwowym planem teoretycznie powinna zapewnić sukces tego działania. Jednak wdrożenie nie jest procesem odizolowanym, ma na niego wpływ wiele czynników negatywnych i pozytywnych. Dla powodzenia wdrożenia najważniejsze jest prawidłowe zidentyfikowanie czynników negatywnych. Czynniki ryzyka w procesie wdrożenia można podzielić na dwie podstawowe grupy [Kisielnicki 2008, s. 219]:

- 1) wynikające z błędów użytkownika, np.: zbyt wielu użytkowników, sprzeczne cele wdrożenia, brak środków na realizację wdrożenia, zmienność polityki kierownictwa;
- 2) wynikające z błędów wdrażających, np.: rozpad zespołu wdrożeniowego, brak doświadczenia zespołu wdrożeniowego, błędy w opracowaniu kosztorysu i harmonogramu wdrożenia, trudności techniczne.

Efektywne zarządzanie procesem wdrożenia wymaga zidentyfikowania **czynników ryzyka wdrożenia** już we wstępnej fazie projektowania. Następnie dla każdego z nich należy zaproponować skuteczny plan przeciwdziałania. Jeżeli takiej możliwości nie ma lub nie można jej zastosować w określonym przypadku, to dalszą realizację projektu należy wstrzymać lub nawet jej zaniechać [Kisielnicki 2008, s. 219].

### 7.1.3. Przygotowanie finansowe i logistyczne przedsięwzięcia

„Logistyka” jest terminem opisującym proces planowania, realizowania i kontrolowania sprawnego i efektywnego ekonomicznie przepływu surowców, materiałów, wyrobów gotowych oraz odpowiedniej informacji z punktu pochodzenia do punktu konsumpcji w celu zaspokojenia wymagań klienta [Coyle, Bardi, Langley 2002, s. 51–52]. W odniesieniu do realizacji projektu informatycznego ma zapewnić dopływ odpowiednich zasobów, gwarantujących możliwość jego wykonania. Aby możliwe było zarządzanie procesem logistycznym w projektowaniu i wdrażaniu SWD, musi zostać opracowany szczegółowy harmonogram przedsięwzięcia, a na jego podstawie wyspecyfikowane niezbędne elementy, takie jak:

- koszty realizacji etapu,
- osoby odpowiedzialne za realizację etapu,
- narzędzia wspomagające realizację etapu,
- wiedza ekspertów niezbędna do realizacji etapu.

Przygotowanie finansowe projektu polega na ustaleniu zasobów potrzebnych do jego realizacji. Należy również uwzględnić późniejszy koszt użytkowania wytworzonego produktu. Zarządzanie kosztami obejmuje:

- planowanie zasobów ludzkich, materiałowych i maszynowych.
- szacowanie kosztów zasobów,
- budżetowanie i kontrolę kosztów w budżecie.

Planowanie zasobów polega na wskazaniu, jakie fizyczne zasoby oraz w jakich ilościach zostaną wykorzystane do wykonania projektu.

Szacowanie kosztów zasobów ma na celu określenie łącznych oraz jednostkowych kosztów wykorzystywania zasobów. Jest to zadanie indywidualnie złożone. W trakcie jego realizacji należy więc wspomagać się takimi środkami, jak:

- modelowanie algorytmiczne,
- ekspertyzy specjalistów,
- porównania z już zrealizowanymi projektami,
- ograniczenia (np. warunkami zamówienia, terminem realizacji czy stanem finansowym firmy).

Przykładem algorytmicznego szacowania kosztu realizacji przedsięwzięcia informatycznego jest model COCOMO (ang. *COst CONstruction MOdel*) lub COCOMO II [Boehm, Abts, Brown 2000]. W modelu tym koszt wyraża się jako funkcję miary kodu (miarą może być liczba linii kodu lub liczba tzw. punktów funkcyjnych<sup>1</sup>). Model COCOMO pozwala także szacować nakład pracy (w osobomiesiącach) oraz czas realizacji projektu.

Budżetowanie polega na przydzieleniu ogólnych szacunków kosztów do poszczególnych elementów pracy, co umożliwi pomiar postępów wykonania projektu.

Kontrola kosztów w budżecie jest ściśle powiązana z ogólnym nadzorem zmian. Celem kontroli jest monitorowanie wykonania kosztów, zapobieganie niewłaściwym zmianom oraz prawidłowe wprowadzanie koniecznych zmian.

Prawidłowe przygotowanie logistyczne i finansowe przedsięwzięcia informatycznego jest jedną z ważniejszych determinant jego powodzenia.

---

<sup>1</sup> Punkty funkcyjne oparte są na połączeniu takich elementów, jak: wejścia-wyjścia systemu, interakcje z użytkownikiem, interfejsy zewnętrzne, pliki wykorzystywane przez system; z każdym z elementów jest związana waga o wartości 3–15 pkt; liczba punktów funkcyjnych systemu jest wyznaczana na podstawie sumy wag elementów systemu.

## 7.1.4. Zarządzanie projektem wdrożenia SWD

Sukces projektu wdrożeniowego w dużej mierze zależy od zdolności i kwalifikacji socjotechnicznych kierownika projektu. Aby wdrożenie SWD zakończyło się sukcesem, należy [Frączkowski 2003, s. 119]:

- uzyskać poparcie i zaangażowanie kierownictwa firmy w projekt wdrożenia;
- zapewnić niezbędne zasoby ludzkie, konieczne do efektywnej realizacji wdrożenia;
- przyjąć metodykę wdrożenia lub ustalić indywidualne zasady wdrożenia SWD;
- zapewnić elastyczność w doborze parametrów projektu wdrożenia SWD;
- uwzględnić aspekty socjotechniczne związane z przyzwyczajeniami i oczekiwaniami przyszłych użytkowników SWD;
- wybrać lidera procesu wdrożeniowego.

Zadania zespołu wdrożeniowego różnią się w zależności od tego, czy wdrażany system został zakupiony w postaci gotowej, czy też był przygotowywany na zamówienie użytkownika. Jednakże w obydwu przypadkach głównym zadaniem kierownictwa projektu jest stworzenie takich warunków, żeby system został przyjęty, zaakceptowany, a następnie był efektywnie wykorzystywany w organizacji. Kierownictwo wdrożenia tworzy plany taktyczne i operacyjne, uszczegóławiające cały proces wdrożenia do jego określonych etapów, do których należy zaliczyć:

- zainicjowanie projektu wdrożenia,
- określenie i weryfikację wymagań wdrożeniowych,
- wykonanie projektu wdrożenia,
- przygotowanie kosztorysu wdrożenia,
- zaplanowanie realizacji i testów wdrożenia,
- techniczne wdrożenie i testy systemu,
- organizacyjne wdrożenie i szkolenia użytkowników,
- zamknięcie projektu i eksploatację wdrożonego systemu.

Zarządzając projektem wdrożenia, nie można zapomnieć o ryzyku wdrożenia systemów informatycznych. Ryzyko w projekcie informatycznym to prawdopodobieństwo wystąpienia zdarzenia, które niekorzystnie wpłynie na proces jego realizacji. Przyczyn niepowodzenia może być wiele:

- nadmierna złożoność projektu;
- nieprawidłowo przeprowadzona analiza projektowa;
- nieprecyzyjne sformułowanie wymagań;
- niepełna i niespójna specyfikacja projektu;
- utrata pracowników zaangażowanych w projekt;
- problemy wynikające z przekroczenia czasu wyznaczonego harmonogramem wdrożenia;
- problemy wynikające z przekroczenia ustalonego budżetu wdrożenia.

Nadmierna złożoność projektu jest realnym zagrożeniem w przypadku SWD, ponieważ z definicji jest on dedykowany rozwiązywaniu złożonych problemów decyzyjnych. Dodatkowym problemem staje się fakt, że użyteczność takiego systemu jest ściśle związana z jak najszerszym zakresem jego wykorzystywania w procesie podejmowania decyzji zarządczych. Dlatego efektywność użytkowania systemu SWD powinna być monitorowana przez cały okres jego funkcjonowania w organizacji.

Etap analizy, który poprzedza sformułowanie szczegółowych wymagań oraz tworzenie projektu, jest jednym z najważniejszych elementów decydujących o powodzeniu i efektywności całego przedsięwzięcia. Źle lub nieprecyzyjnie przeprowadzona analiza projektowa uniemożliwi prawidłowe sformułowanie wymagań, a następnie realizację całego projektu.

Wdrożenie projektu informatycznego na każdym etapie wymaga zaangażowania i poparcia pracowników. Utrata pracowników merytorycznych w okresie wdrożenia może doprowadzić do braku możliwości realizacji przedsięwzięcia.

Problemy z przestrzeganiem harmonogramu i budżetu występują w ok. 50% wdrożeń projektów informatycznych. Wiąże się to, niestety, z niedogodnościami i utrudnieniami podczas wdrożenia, ale prawdziwym problemem jest w tym przypadku zagrożenie pozytywnego zakończenia całego przedsięwzięcia.

W aspekcie wymienionych zagrożeń istotnym zagadnieniem staje się możliwość ograniczenia ryzyka podczas wdrażania złożonych projektów informatycznych. Według Standish Group głównymi czynnikami sukcesu projektów informatycznych są [Stokalski 2010, s. 355–356]:

- zaangażowanie użytkowników — jasne sprecyzowanie: ról, zadań, zakresu odpowiedzialności i uprawnień, ale przede wszystkim zrozumienie idei projektu przez jego przyszłych użytkowników;
- wsparcie decydentów — szczególnie tych, którzy komunikują się z osobami odpowiedzialnymi za realizację przedsięwzięcia lub/i uczestniczącymi w procesie jego wdrożenia w celu osiągnięcia jak najlepszej realizacji planowanych założeń;
- jasno określone cele biznesowe — zdefiniowanie mierzalnych kryteriów osiągnięcia celu, pozwalających kontrolować, oceniać, i w razie potrzeby modyfikować decyzje dotyczące zakresu i funkcjonalności projektu jeszcze w czasie jego realizacji;
- racjonalizacja zakresu i wymagań — elastyczność projektu umożliwiająca weryfikację rozwiązań przez użytkowników oraz wprowadzanie zmian będących wynikiem zaistniałych w trakcie wdrażania problemów;
- doświadczony kierownik przedsięwzięcia — ocena doświadczenia kierownika z punktu widzenia specyfiki przedsięwzięcia jest ważnym elementem analizy ryzyka wdrożenia systemu informatycznego;
- podział przedsięwzięcia na iteracje — zgodnie z metodyką ARAD (ang. *Architected Rapid Application Development*) realizacja systemu informatycznego nie powinna przekroczyć 120–200 dni roboczych; projekty przekraczające ten czas

- powinny być podzielone na zamknięte etapy (iteracje), z których każdy dostarczy użytkownikowi określony, zamknięty zakres funkcjonalności systemu;
- zarządzanie budżetem i aspektami finansowymi projektu — podstawowe ramy opłacalności przedsięwzięcia wyznacza jego model korzyści (ang. *business case*); przewidywany wzrost kosztów lub opóźnienie wdrożenia może na tyle wpłynąć na model korzyści, że projekt stanie się nieopłacalny; wtedy rezygnacja z części wymagań, uproszczenie produktu końcowego i szybsze niż planowano wdrożenie może radykalnie wpłynąć na poprawę tej opłacalności;
  - zespół z odpowiednimi kompetencjami — analiza kompetencji zespołu jest ważnym elementem analizy ryzyka, zwłaszcza gdy wykorzystywane są lekkie metodyki wytwórcze, wymagające od członków zespołu dyscypliny i wielu umiejętności;
  - formalna (zdefiniowana) metodyka realizacji przedsięwzięcia — czyli efektywna adaptacja dobrych praktyk, dostosowanych do specyfiki projektu, wymagającego tempa pracy i oczekiwań jakościowych odbiorcy systemu;
  - jednolite narzędzia i infrastruktura projektu — niezbędne we współczesnych projektach, pozwalają na wydajną pracę, zwiększając jednocześnie dyscyplinę związaną z zapewnieniem jakości.

Uwzględnienie opisanych wskazówek oraz wykorzystanie sprawdzonych metod wdrożenia może w znaczny sposób wpłynąć na sukces przedsięwzięcia, a także na efektywność jego eksploatacji podczas funkcjonowania systemu w przedsiębiorstwie.

## 7.2. Metodyka wielokryterialnego wyboru systemu informatycznego klasy SWD

Jednym z podstawowych zastosowań technologii systemów informacyjnych jest ich wykorzystanie w organizacji procesów produkcyjnych. Pierwszym obszarem, w których użyto komputerów, była rejestracja danych osobowych (administracyjnych) oraz zarządzanie finansami (księgowość). Współcześnie zakres ten znacznie się rozszerzył. Komputery wykorzystuje się do wspomagania systemów produkcji, obsługi sprzedaży i wielu innych zastosowań. Dużą grupę systemów informacyjnych stanowią systemy przetwarzania transakcji. Wchodzą one w skład systemów informacyjnych zarządzania. Zadanie tych systemów polega na tworzeniu informacji zbiorczych na podstawie masowych danych transakcyjnych. Późniejsze koncepcje rozwinęły się w kierunku systemów wspomagania decyzji (SWD). Systemy te dostarczają zarządzającym informacji, które mogą być przydatne w podejmowaniu decyzji strategicznych.

Zastosowanie w praktyce SWD jest coraz skuteczniejsze, ponieważ częściej wykorzystuje się systemy interakcyjne. Systemy interakcyjne umożliwiają użytkownikom wprowadzanie danych bezpośrednio do systemu informacyjnego w miejscu ich powstawania. Umożliwiają również w pewnym stopniu wyszukiwanie danych w tym systemie.



## 7.2.1. Istota modelu wielokryterialnego

W zagadnieniach technicznych, technologicznych i ekonomicznych pojawia się często zadanie optymalizacji budowy pewnego złożonego modelu systemu lub procesu działania tego systemu. **Metody optymalizacyjne** stosuje się w celu osiągnięcia pewnego maksymalnego efektu. Sformułowanie celu optymalizacji wymaga zdefiniowania pewnej liczby zmiennych  $x_1, x_2, \dots, x_n$ . Następnie formułuje się różne ograniczenia, które są odzwierciedleniem wymagań stawianych modelowi. Podstawowym zadaniem jest sformułowanie funkcji kryterialnej, opisującej jakość badanego modelu. Jeśli wystarczy jedna cecha opisująca jakość działania modelu, to zadanie sprowadza się do wyznaczenia maksimum lub minimum funkcji kryterialnej. Jednak w zadaniach z zakresu wspomaganego decyzji jakość działania modelu opisywana jest przez kilka kryteriów:

$$f_i = f_i(x_1, x_2, \dots, x_n),$$

gdzie:

$$i = 1, 2, \dots, k.$$

Dla takiego modelu można formułować zadania osiągnięcia maksymalnego zysku, przy ograniczonych nakładach, maksymalnej niezawodności i gotowości badanego modelu systemu. Wiadomo, że przy sformułowaniu kilku funkcji kryterialnych występują sprzeczności polegające na tym, że funkcje kryterialne nie mogą jednocześnie być zoptymalizowane. Na przykład, jeśli funkcja  $f_1$  osiąga maksimum, to funkcja  $f_2$  może przyjmować wartości dalekie od swych wartości ekstremalnych. W związku z tym powstaje prawie zawsze problem znalezienia kompromisu. Dla prostych zadań pomocna jest intuicja i doświadczenie. Jednak dla większej liczby zmiennych decyzyjnych i większej liczby kryteriów pilne staje się opracowanie środków matematycznych, ułatwiających znajdowanie możliwie obiektywnych rozwiązań kompromisowych. W najprostszym przypadku podejmuje się próby, jak wybrać jedno kryterium jako najbardziej istotne i ograniczyć się do optymalizacji tego kryterium. W wielu przypadkach tworzy się pewne zastępcze kryterium jakości z różnych, częściowo przeciwstawnych kryteriów. Optymalizacja tej zastępczej funkcji kryterialnej zapewnia rozwiązanie kompromisowe problemu. Jedną z najbardziej znanych metod rozwiązywania zadań programowania wielokryterialnego jest zastąpienie wektora funkcji kryterialnych  $(f_1, f_2, \dots, f_k)$  nową funkcją  $F(f_1, f_2, \dots, f_k)$ . O kryterium  $F$  mówi się, że jest zagregowanym kryterium w stosunku do  $f_1, f_2, \dots, f_k$ . Najbardziej rozpowszechniony sposób agregacji funkcji  $f_1, f_2, \dots, f_k$  to wykorzystanie funkcji liniowej w postaci:

$$F = \sum_{i=1}^k a_i f_i,$$

gdzie:

$a_i$  — liczby dodatnie, charakteryzujące ważność kryteriów.

## Relacja pierwszeństwa, funkcje użyteczności i wyboru

Dostatecznie dobrze opisano sposób określenia pierwszeństwa na przykładzie relacji między wektorami binarnymi. Relacje takie są wykorzystywane nie tylko w przypadku wektorów binarnych, ale także obiektów różnej natury.

**Relacją binarną** na zbiorze  $A$  nazywamy dowolny podzbiór iloczynu kartezjańskiego  $A \times A$ :

$$R \subset A \times A.$$

Oznacza to, że relacja jest zbiorem par uporządkowanych  $(a, b)$ ,  $a, b \in A$ . Jeśli  $(a, b) \in R$ , to mówimy, że elementy  $a$  i  $b$  są w relacji  $R$  i fakt ten zapisujemy jako  $a R b$ . W zbiorze relacji binarnych wprowadza się operacje mnogościowe, takie jak: iloczyn  $\cap$ , dodawanie  $\cup$  i inne. Wprowadza się także działania specyficzne tylko dla relacji. Relacją odwrotną do relacji  $R \subset A \times A$  nazywa się relację  $R^{-1}$  zdefiniowaną następująco:

$$R^{-1} = \{(a, b) \subset A \times A; (b, a) \in R\},$$

ozn. że para  $(a, b)$  należy do relacji  $R^{-1}$  wtedy i tylko wtedy, gdy para  $(b, a)$  należy do  $R$ .

Kiedy  $R \subset A$ , wtedy relacja  $R_B = \{(a, b) \in R; a, b \in B\}$  nazywa się obcięciem relacji  $R$  do zbioru  $B$ . Relacja  $R$  jest zwrotna, jeśli  $(a, a) \in R$ , dla każdego  $a \in A$ . Relacja  $R$  jest przeciwzwrotna, jeśli  $(a, a) \notin R$ , dla każdego  $a \in A$ . Relacja  $R$  nazywa się symetryczną, jeśli z tego, że  $(a, b) \in R$  wynika, iż  $(b, a) \in R$ . Relacja  $R$  nazywa się asymetryczną, jeśli z tego, że  $(a, b) \in R$  wynika, iż  $(b, a) \notin R$ . Relacja  $R$  nazywa się antysymetryczną, jeśli z tego, że  $(a, b) \in R$  i  $(b, a) \in R$  wynika, iż  $a = b$ . Relacja asymetryczna jest przeciwzwrotna. Relacja  $R$  nazywa się przechodnią, jeśli z tego, że  $a R b$  i  $b R c$  wynika, że  $a R c$ . Relacja  $R$  jest zupełna, jeśli z tego, że  $(a, b) \notin R$  wynika, iż  $(b, a) \in R$  dla dowolnych  $a, b \in A$ . Elementy zbioru  $a, b \in A$  są porównywalne, jeśli  $a R b$  lub  $b R a$  i nieporównywalne względem relacji  $R$ , jeśli nie zachodzi  $a R b$  i  $b R a$ . Relacja  $R$  nazywa się spójną, jeśli dla dowolnych  $a, b \in A$  zachodzi  $a R b$  lub  $b R a$ .

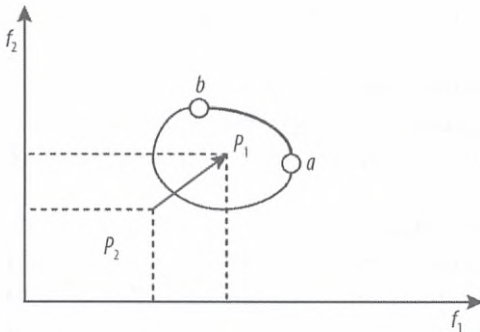
Relację zwrotną, symetryczną i przechodnią nazywa się **relacją równoważności**. Przykładem relacji równoważności jest relacja równości wektorów w  $\mathbf{E}^n = \{0, 1\}^n$ . Relacje równoważności odgrywają bardzo dużą rolę w zastosowaniach w matematyce. Jest to związane z tym, że relacja równoważności  $R \subset A \times A$  ściśle odnosi się do rozbicia zbioru  $A$  na podzbiory rozłączne. Podzbiory  $A_i$ ,  $i = 1, 2, \dots, n$  tworzą rozbicie zbioru  $A$ , jeśli  $A_1 \cup A_2 \cup \dots \cup A_n$  i dla każdego  $i \neq j$  zachodzi  $A_i \cap A_j = \emptyset$ .

Związek między relacją równoważności  $R$  i rozbiciem zbioru można ująć następująco: każdej relacji równoważności  $R$  odpowiada jednoznacznie pewien podział zbioru  $A$ , i odwrotnie dla każdego podziału zbioru  $A$  można określić jednoznacznie relację równoważności. Relacja przeciwzwrotna przechodnia (asymetryczna) nazywa się silnym (częściowym) porządkiem.

Jeśli  $a_i \geq b_i$  dla  $i = 1, 2, \dots, n$ , to:  $a \geq b$ ,  $a = (a_1, a_2, \dots, a_n)$ ,  $b = (b_1, b_2, \dots, b_n)$  i relacja  $\geq$  jest relacją częściowego porządku w zbiorze  $\mathbf{E}^n = \{0, 1\}^n$ .

Jeśli dla  $a \in B$  nie istnieje w  $B$  element  $b$  taki, że  $b \geq a$ , to  $b$  jest punktem zbioru Pareto. Zbiór wszystkich punktów Pareto tworzy zbiór Pareto. Aby zilustrować powyższe rozważania, przyjmuje się, że  $m = 2$  oraz że obie funkcje kryterialne podlegają maksymalizacji. Na rysunku 7.2 pokazano obszar dopuszczalnych rozwiązań dla zadania maksymalizacji funkcji  $f_1(x_1, x_2), f_2(x_1, x_2)$ .

Rysunek 7.2. Ilustracja zbioru Pareto



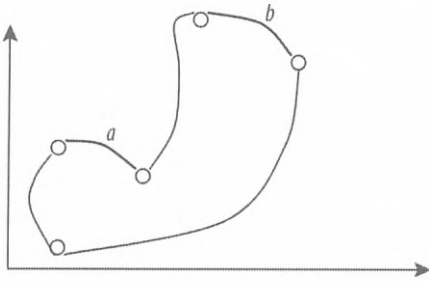
Źródło: Opracowanie własne.

W przypadku jednoczesnej maksymalizacji funkcji  $f_1(x_1, x_2)$  i  $f_2(x_1, x_2)$  punkt  $P_1$  reprezentuje lepsze rozwiązanie dla obu funkcji kryterialnych niż punkt  $P_2$ . Za pomocą takiego rozumowania można teraz wyjaśnić, że wszystkie rozwiązania, których współrzędne leżą wewnątrz obszaru, mogą być jeszcze poprawione. Rozumując w ten sposób, otrzymuje się zbiór rozwiązań kompromisowych tworzących zbiór Pareto, do którego należą punkty brzegowe. Na rysunku 7.2 zbiór Pareto jest brzegiem figury między punktami  $a$  i  $b$ . Jest oczywiste, że do zbioru Pareto należą tylko punkty brzegowe obszaru, ponieważ każdy punkt leżący wewnątrz można polepszyć przez przesunięcie do brzegu tego obszaru. Fakt, że oba kryteria nie przyjmują swych wartości ekstremalnych w jednym punkcie wskazuje na istnienie pewnej sprzeczności w sformułowaniu zadania. Jeżeli poruszamy się po linii z punktu  $a$  do  $b$  tworzącej zbiór Pareto, to powiększaniu wartości funkcji  $f_2$  towarzyszy jednocześnie zmniejszanie wartości  $f_1$ . Zbiór Pareto opisuje wszystkie możliwe rozwiązania kompromisowe. Jeśli jedną z funkcji, na przykład  $f_1$ , należy zminimalizować, to wystarczy rozważyć maksymalizację funkcji  $-f_1$ .

Warunkiem koniecznym przynależności jakiegoś punktu brzegowego do zbioru Pareto przy maksymalizacji funkcji  $f_1$  i  $f_2$  jest ujemna wartość współczynnika kierunkowego stycznej do krzywej brzegowej w każdym punkcie tego zbioru. Za warunek dostateczny trzeba ponadto uznać taki, który zapewni, że analizowane zbiory brzegowe będą zawierały punkty niegwarantujące poprawy w funkcji celu. Na rysunku 7.3 pokazano dwa odcinki brzegowe, jednak brzeg  $a$  nie należy do zbioru Pa-

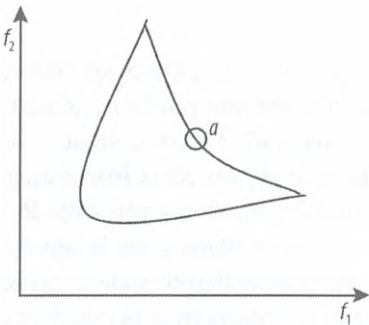
reto, gdyż jest on poprawialny przez brzeg  $b$ . Przedstawiony na rysunku 7.4 zbiór redukuje się do jednego punktu.

**Rysunek 7.3. Zbiór Pareto w przypadku zbioru niewypukłego**



Źródło: Opracowanie własne.

**Rysunek 7.4. Zbiór Pareto jednoelementowy**



Źródło: Opracowanie własne.

Wyznaczanie zbioru Pareto jest pierwszym etapem zadania optymalizacji wielokryterialnej. Trzeba również ustalić dodatkowe kryteria wyboru jednego lub większej liczby punktów ze zbioru Pareto.

## 7.2.2. Budowa modelu przedsięwzięcia

Każde przedsięwzięcie dotyczące SWD ma pewne cechy szczególne, ale można je porównywać z ogólnym modelem przedsięwzięcia. W SWD bardzo ważna jest budowa modelu procesu działania systemu. Model procesu wymaga określenia zbioru cech, które:

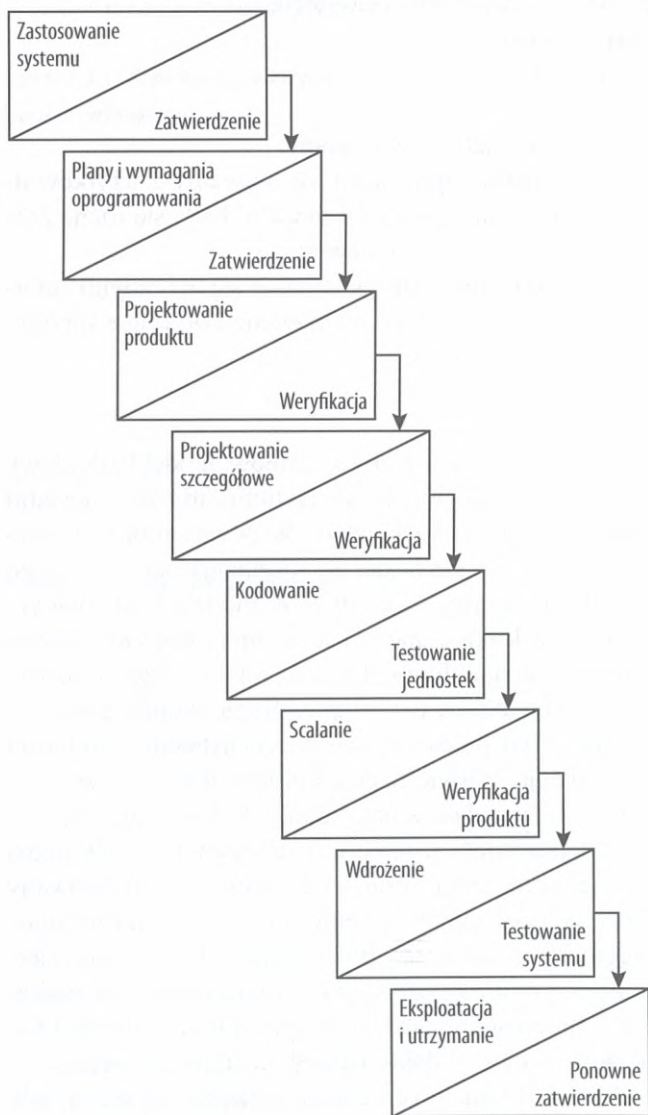
- dadzą się dopasować do dużej grupy zastosowań,
- zapewniają poprawną definicję każdego przedsięwzięcia,
- są łatwe do interpretacji i zrozumienia,
- mogą być aktualizowane w czasie.

Tworzenie modelu SWD można podzielić na trzy etapy:

- 1) analiza wymagań — na tym etapie przeprowadza się wywiady z użytkownikami, aby poznać opinię na temat niedoskonałości modelu; bada się różne źródła informacji dotyczące systemu i nowych wymagań;
- 2) specyfikacja wymagań — na tym etapie analityk zajmuje się zebranymi informacjami; tworzony jest dokument zawierający wymagania dotyczące sprzętu, oprogramowania, wymagań funkcjonalnych;
- 3) tworzenie projektu wysokiego poziomu.

Istnieją dwa podstawowe modele cyklu tworzenia systemów: model kaskadowy i model spiralny. Wszystkie inne modele są zwykle wariantami lub rozwinięciami tych modeli. Jako pierwszy opiszemy model kaskadowy. W pewnym okresie rozwoju przedsięwzięć informatycznych stało się jasne, że wymagają one większego stopnia sformalizowania niż miało to miejsce wcześniej. W modelu kaskadowym tworzenie systemu jest podzielone na kilka etapów. Każdy musi być zakończony zanim zaczną się prace na następnym etapie. Etapy dzieli się na dwie części: pierwsza dotyczy rzeczywistej pracy wykonanej na tym etapie, druga obejmuje weryfikację i zatwierdzenie tej pracy. Weryfikacji dokonuje się w celu ustalenia zgodności między produktem a jego specyfikacją. Zatwierdzenie dotyczy tego, czy w przypadku danego produktu są realizowane zaplanowane zadania. Należy zauważyć, że w ramach etapu istnieje pewna powtarzalność wykonywanych czynności, natomiast między etapami powtarzalność jest niewielka. Model kaskadowy jest traktowany jako model sekwencyjny. Identyfikacja i nazwy etapów nie są sztywno ustalone i mogą być modyfikowane zgodnie ze specyfiką poszczególnych przedsięwzięć. Analizowany model ma wiele zalet, ponieważ oprócz zasadniczego celu porządkowania działań zawiera elementy zarządzania jakością. Etapowa natura modelu kaskadowego oraz zamknięcie każdego etapu to dobre narzędzia do planowania, kontroli i zarządzania ryzykiem. Modele kaskadowe działają najlepiej wówczas, gdy utrzymywany jest minimalny poziom powtórzeń prac nad produktami, a same produkty pozostają niezmiennione po zakończeniu etapu. Na rysunku 7.5 pokazano schemat modelu kaskadowego dla systemu informatycznego.

Rysunek 7.5. Kaskadowy model tworzenia systemu



Źródło: [Cadle, Yeates 2004].

### 7.2.3. Przestrzeń decyzyjna

Budowa **przestrzeni decyzyjnej** związana jest ściśle z planowanym przedsięwzięciem. Tworzenie SWD jest przedsięwzięciem złożonym, obejmującym zwykle syntezę różnych elementów, takich jak: sprzęt komputerowy, oprogramowanie, zbieranie danych, opracowanie odpowiednich algorytmów, szkolenie użytkowników

systemu. Takie przedsięwzięcie ma szansę powodzenia, jeśli jest z góry dokładnie zaplanowane. Plan przedsięwzięcia jest tworzony w celu przekazania niezbędnych informacji wszystkim zainteresowanym tworzeniem SWD. Znajomość planu przez kierownika przedsięwzięcia umożliwia rozpoznanie, czy prace nad systemem przebiegają zgodnie z harmonogramem i czy konieczne są działania korekcyjne. Brak planu uniemożliwia kierującemu projektem podejmowanie niezbędnych działań. Zasadniczym punktem wyjścia do budowy planu jest właściwe postawienie wymagań, które muszą odpowiadać na pytanie: „Co chcemy osiągnąć, budując SWD?”. Tworzy się specyfikacje wymagań o wysokim stopniu szczegółowości i wystarczającym zakresie. Jednak jeśli zaczynamy badanie strategii opracowania planu systemu wspomagania decyzji, wymagania mogą być szerokie. Na dalszym etapie planowania SWD potrzeba bardziej precyzyjnych definicji wymagań. Po sformułowaniu celów przedsięwzięcia należy rozważyć, jak analizować te cele. Aby to osiągnąć, konieczny jest podział pracy, co sprowadza się do opracowania listy działań potrzebnych do realizacji danego przedsięwzięcia. Bardzo często przedsięwzięcie dzieli się na dwie zasadnicze części: prowadzenie badania i sporządzenie raportu (rysunek 7.6).

**Rysunek 7.6. Struktura podziału pracy: poziom najwyższy**



Źródło: Jak rysunku 7.5.

Pierwszą ze składowych podziału, tzn. prowadzenie badania, można dzielić dalej, tak jak na rysunku 7.7. Przedstawiono na nim pięć pakietów roboczych. Wiadomo, że można dzielić dalej. Idea podziału pracy polega na tym, że każde działanie dzieli się

**Rysunek 7.7. Struktura podziału prac: poziom drugi**



Źródło: Jak rysunku 7.5.

do momentu, gdy otrzyma się działanie na tyle elementarne, iż nie da się go przydzielić więcej niż jednej osobie oraz na tyle małe, aby ocenić czas trwania działania. W planowaniu SWD ważne jest uwzględnianie planowania jakości. Plan jakości może być osobnym dokumentem lub może stanowić część planu przedsięwzięcia. Zawiera działania, takie jak przeglądy jakościowe czy badania dodatkowe uwzględniające specyfikę danego przedsięwzięcia. Należy zauważyć, że w planowaniu SWD niektóre kryteria formułuje się w postaci dopuszczalnych przedziałów, zwanych tolerancjami. Podczas budowy modelu nie powinno się przekraczać pewnych ograniczeń dla parametrów. W praktyce okazało się, że wiele systemów informatycznych znacznie przekroczyło planowany czas i budżet realizacji. W związku z tym przedsięwzięcia w zakresie planowania SWD wymagają dobrych metod szacowania parametrów tego przedsięwzięcia. Istnieje wiele różnych metod szacowania, a każda z nich jest podatna na różne błędy i zależy od subiektywnego odczucia rozmiaru realizowanych zadań. Jedną z metod szacowania jest metoda analogii. Metoda ta polega na porównywaniu z podobnymi, wcześniej realizowanymi przedsięwzięciami. Główną zaletą metody analogii jest to, że pozwala ona na szybkie i pobieżne oszacowanie parametrów przedsięwzięcia, np. przygotowanie wymagań do przetargu. Nowsze przedsięwzięcia mogą być bardziej złożone, wymagające np. niezrealizowanych technik; wtedy może powstać pewne niedoszacowanie parametrów. Na etapie wstępnych oszacowań dla przedsięwzięcia stosuje się metodę analizy nakładów. Ogólna zasada tej metody polega na oszacowaniu nakładów potrzebnych do wykonywania analizy dla założonych funkcji przedsięwzięcia i wprowadzeniu kolejnych etapów tego przedsięwzięcia przy wykorzystaniu wskaźników w stosunku do nakładów na analizę. Najbardziej szczegółowa metoda szacowania jest oparta na podziale przedsięwzięcia. Wyróżnia się dwie podstawowe metody podziału: strukturę podziału lub strukturę podziału produktu. Na początkowym etapie wdrożenia metody podziału sporządza się listę zadań przedsięwzięcia, następnie eksperci oceniają nakłady na wykonanie każdego zadania. Jeśli ich wiedza i doświadczenie są odpowiednie, to metoda ta zapewni bardzo wiarygodne oszacowania. Jednak nie zawsze możliwe jest zastosowanie tej metody ponieważ:

- na początku przedsięwzięcia na ogół nie ma dostatecznej ilości informacji do określenia pełnego zestawu zadań;
- jest ona bardzo czaso- i pracochłonna;
- koszty bezpośredniego oszacowania mogą nie być uzasadnione, jeśli wiarygodny wynik uzyskano innymi metodami, np. przez analogię.

Inna metoda, czyli metoda delficka, polega na uzyskaniu ocen od odpowiednio wykwalifikowanych osób, a następnie na połączeniu tych ocen w celu stworzenia oszacowania końcowego.

Poziom doświadczenia ekspertów może być różny, dlatego podejście do metody delfickiej składa się z kilku etapów:



- każdy ekspert otrzymuje specyfikację pracy zawierającej działania i zadania,
- ekspert dostarcza oszacowanie dotyczące specyfikacji,
- oszacowania wszystkich ekspertów są sumowane i dostarczane każdemu z nich.

Każdy ekspert może porównać swoje oszacowanie z oszacowaniem podsumującym i ewentualnie je skorygować. U podstawy tej metody leży zasada anonimowości. Harmonogram przedsięwzięcia obejmuje dwa elementy:

- 1) kolejność wykonywania pracy,
- 2) daty rozpoczęcia i zakończenia planowanych działań.

Opracowanie harmonogramów jest zwykle procesem iteracyjnym. Po sformułowaniu założeń wstępnych wykonuje się pierwszy szkic harmonogramu i porównuje wyniki z oczekiwanymi. Przy opracowywaniu harmonogramu dzieli się przedsięwzięcia na poszczególne działania, bada zależności między nimi i oszacowuje czas trwania tych działań. Aby harmonogram spełnił należycie swoją funkcję, musi być udokumentowany w sposób jasny i przystępny dla wszystkich, którzy w tym przedsięwzięciu uczestniczą. Jedną z kilku decyzji, które trzeba podjąć wcześniej, dotyczy odpowiedzi na pytania: „Czy plan przedsięwzięcia powinien obejmować plany jakości i zarządzania ryzykiem?”, „Czy też powinny być to niezależne dokumenty?”. Całościowy plan przedsięwzięcia powinien co najmniej zawierać następujące elementy [Cadle, Yeates 2004]:

- **Wstęp.** Opis dokumentu lub jego części.
- **Wykaz autoryzacji i uzupełnień.** Opis statusu dokumentu, informacje o jego autorach i osobach zatwierdzających oraz historia uzupełnień.
- **Rozdzielnik.** Lista osób, które powinny otrzymać dokument.
- **Dokumenty powiązane.** Odnośniki do różnych dokumentów, np. do dokumentu zawierającego plan zarządzania ryzykiem i planu jakości, jeśli stanowią odrębne dokumenty, lub do takich, jak standardy organizacyjne, z którymi dane przedsięwzięcie musi być zgodne.
- **Opis przedsięwzięcia.** Krótki opis przedsięwzięcia i jego celów. Jeśli plan dotyczy tylko jednego etapu, to punkt powinien zawierać informację, gdzie i jak dany etap łączy się z ogólnym programem prac.
- **Produkty pośrednie i wynikowe.** Wszystkie produkty powinny być wymienione łącznie z odwołaniem do miejsc, w których są dokładnie zdefiniowane, najczęściej w załączniku do planu przedsięwzięcia, w oddzielnym dokumencie opisującym produkt lub w zewnętrznym dokumencie.
- **Kamienie milowe.** Określenie głównych kamieni milowych w przedsięwzięciu, wyjaśnienie, co one oznaczają i kiedy mogą wystąpić.
- **Organizacja i uprawnienia.** Opis głównych ról i uprawnień w przedsięwzięciu oraz ich łączenie, jeśli potrzeba, ze schematem organizacyjnym pokazującym związki między rolami.

- **Monitorowanie i kontrola.** Opis i wyjaśnienia metod, które będą zastosowane do monitorowania postępów przedsięwzięcia, łącznie z harmonogramem sposobów monitorowania i kontroli, np. zebrań.
- **Kontrola jakości.** Zdefiniowanie miejsc i form kontroli. Obejmuje również opis metody porządkowania uwag z przeglądu oraz możliwości ich uwzględniania. Kontrola jakości może być opisana w oddzielnym planie jakości.
- **Sprawozdawczość.** Opis mechanizmów sprawozdawczych zawierających informacje o tym, kto będzie tworzył raporty dotyczące zarządzania, co one będą obejmowały, jak często będą sporządzane i kto będzie je otrzymywał.
- **Przegląd i zatwierdzenia.** Dotyczy to kontroli jakości i polega na pisemnym przydzieleniu osób odpowiedzialnych za tworzenie, przeglądanie i zatwierdzanie różnych produktów przedsięwzięcia.
- **Ryzyko w przedsięwzięciu.** Może być opisane w oddzielnym planie zarządzania tym ryzykiem. Punkt ten zawiera opis głównych elementów ryzyka, ocenę jego prawdopodobieństwa i ewentualnych następstw oraz szkic przewidzianych działań w celu jego uniknięcia.
- **Harmonogram przedsięwzięcia.** Zawiera diagram sieciowy i wykres słupkowy danego przedsięwzięcia.

Do każdego zadania z diagramu sieciowego i wykresu słupkowego powinien być sporządzony opis obejmujący:

- cel zadania,
- pracę do wykonania,
- metody realizacji zadania,
- standardy, które należy zastosować,
- przydzielone nakłady,
- produkty wynikowe i pośrednie,
- kryteria zakończenia; określenie momentu zakończenia zadania.

## 7.2.4. Wielokryterialna procedura wyboru

Celem tego punktu jest opisanie możliwości metod wielokryterialnej analizy decyzyjnej. Wiadomo, że z natury problem decyzyjny jest wielokryterialny. Wielokryterialną procedurę wyboru można scharakteryzować następująco:

- została zdefiniowana skończona liczba mierzalnych lub stopniowalnych funkcji celu opisujących obiekt decyzyjny;
- cele są osiąmane przez wybór wspólnych decyzji dopuszczalnych;
- podstawą podjęcia decyzji jest zbiór celów sformułowanych dla funkcji kryterialnych;
- w zbiorze funkcji kryterialnych występuje zjawisko konkurencyjności, które powoduje, że stopień osiągnięcia jednego z celów oznacza zmniejszenie stopnia osiągnięcia innych celów.

Problemem zasadniczym w procedurze wielokrotnego wyboru jest określenie sposobu postępowania prowadzącego do uzyskania rozwiązania „lepszego” od pozostałych. W zadaniach optymalizacji z jedną funkcją kryterialną teoria programowania matematycznego rozstrzyga problem jednoznacznie, o ile zbiór rozwiązań zadania optymalizacji jest zbiorem niepustym. Zadanie optymalizacji wielokryterialnej polega na znalezieniu takiej decyzji (rozwiązania dopuszczalnego), która zapewni zadowalające osiągnięcie konkurencyjnych celów. Rozwiązywanie zadania wielokrotnego wyboru powinno być dostosowane do specyfiki zadania. W wielokryterialnej analizie decyzyjnej rozważa się dwa podejścia. Pierwsze podejście wykorzystywane jest w odniesieniu do modelu funkcjonalnego, drugie — do modelu relacyjnego. Podejście funkcjonalne było już wcześniej opisane. Polega ono na agregowaniu różnych funkcji celu w jedną funkcję. Zagregowaną funkcję optymalizuje się na podstawie teorii optymalizacji. Model funkcjonalny wyklucza sytuację nieporównywalności, a także spełniona w nim jest własność przechodniości preferencji. Model relacyjny oparty jest na wykorzystaniu relacji z najbardziej znaną relacją przewyższania. Relacja ta, na podstawie posiadanych informacji, daje ściśle określone preferencje podejmującemu decyzje. Powstała w ten sposób relacja nie jest ani zupełna, ani przechodnia.

## 7.3. Pytania i zadania kontrolne

### 7.3.1. Pytania kontrolne

1. Co to jest metodyka? Jakie są różnice między metodyką i metodą?
2. Jakie metody analizy wielokryterialnej mogą wspomagać dobór gotowego SWD w zależności do potrzeb użytkownika?
3. Co to jest system informacyjny przedsiębiorstwa?
4. Podaj podstawowy podział metodyki projektowania systemów informatycznych.
5. Porównaj różne warianty strategii wdrożenia systemu SWD.
6. Co to jest ryzyko wdrożenia projektu informatycznego?
7. Podaj klasyfikację czynników ryzyka wdrożeniowego.
8. Do jakich celów wykorzystywana jest metoda COCOMO?
9. Jakie są najważniejsze czynniki sukcesu wdrożenia SWD?

### 7.3.2. Przykładowe zadania

1. Dla zadania maksymalizacji funkcji  $f_1(x, y)$  i  $f_2(x, y)$ , dla których zbiór rozwiązań kompromisowych jest kołem o równaniu  $x^2 + y^2 - 2x - 2y + 1 \leq 0$ , należy wyznaczyć zbiór Pareto.

2. Zaproponuj wielokryterialną metodę doboru systemu zarządzania magazynem w średniej wielkości przedsiębiorstwie produkującym części zamienne do maszyn rolniczych.
3. Dokonaj wyboru metodyki wdrożenia (całościowa, cząstkowa, równoległa) systemu wspomagającego podejmowanie decyzji planistycznych (planowanie i harmonogramowanie wielkości produkcji w poszczególnych cyklach produkcyjnych) w przedsiębiorstwie produkcyjnym. Od jakich czynników będzie zależał wybór metodyki w określonym przypadku?
4. Wskaż najważniejsze czynniki ryzyka wdrożenia systemu wspomagającego podejmowanie decyzji planistycznych (planowanie i harmonogramowanie wielkości produkcji w poszczególnych cyklach produkcyjnych) w przedsiębiorstwie produkcyjnym.
5. Dla wyznaczonych w zadaniu 3 czynników ryzyka wdrożenia systemu wspomagającego podejmowanie decyzji zaproponuj sposoby przeciwdziałania tym problemom i zwiększenia szansy powodzenia wdrożenia.

# Wpływ SWD na gospodarczą działalność organizacji

SWD musi spełniać wiele różnorodnych wymagań funkcjonalnych i eksploatacyjnych w zakresie technologii, bezpieczeństwa informacji oraz ograniczeń kosztowych, jakie pojawiają się w fazie wdrożenia, a następnie w fazie eksploatacji i rozwoju systemu. Nie bez znaczenia są również wymagania prawne co do poprawności licencyjnej i użytkowania zgodnego z przepisami. Celem rozdziału jest zaprezentowanie możliwego wpływu SWD na różne aspekty gospodarczej działalności organizacji. W szczególności rozważane są takie charakterystyczne zagadnienia, jak: wielokryterialne metody podejmowania decyzji, psychologiczne i prawne aspekty procesów decyzyjnych, a także rola wiedzy eksperckiej w SWD.

## 8.1. Wpływ SWD na funkcjonowanie organizacji

Systemy wspomagania decyzji, z definicji, służą głębokiemu penetrowaniu procesów decyzyjnych w celu [Kisielnicki, Sroka 2005, s. 244]:

- poszukiwania wariantów możliwych rozwiązań problemu decyzyjnego,
- wzbogacania umiejętności podejmowania decyzji,
- planowania i wdrożenia decyzji,
- oceny i przewidywania skutków decyzji,
- adaptacji decyzji do rzeczywistych warunków.

Wszystkie te funkcje powodują, niewątpliwie, zwiększenie efektywności podejmowanych decyzji, ale jednocześnie mają wpływ na korzystających z nich użytkowników, przyczyniając się do upowszechniania wiedzy w organizacji. Ogromne znaczenie mają systemy wspomagania decyzji z bazą wiedzy oraz systemy ekspertowe. Nigdy wcześniej pozyskiwanie, gromadzenie i udostępnianie wiedzy w organizacji nie było możliwe na tak szeroką skalę.

Kolejną cechą SWD jest zwiększenie skuteczności działania dzięki ułatwieniu komunikacji między osobami podejmującymi decyzje w organizacji. Może mieć na to wpływ swoboda dostępu i publikacji informacji w SWD, a także jednolite standardy danych i pojęć, którymi posługuje się system, a w konsekwencji także jego użytkownicy.

SWD przyczyniają się do poprawienia kontroli działalności i funkcjonowania przedsiębiorstwa. Budowa i praca z różnorodnymi modelami podejmowania decyzji pozwala na bieżące monitorowanie osiąganych wyników, przewidywanie możliwych problemów i zagrożeń oraz wykorzystywanie pojawiających się szans.

SWD może w charakterze jednego ze swych interfejsów informacyjnych zasilać odpowiednio oprogramowaną tablicę (arkusz) metody SWOT<sup>1</sup>, pozwalając w ten sposób na bieżące śledzenie dynamiki zmian w zakresie wewnętrznych silnych i słabych stron przedsiębiorstwa oraz zewnętrznych szans i zagrożeń dla jego działalności i rozwoju.

Uznając wszystkie wymienione zalety SWD, nie można pomijać wyzwań oraz niebezpieczeństw, które systemy tego typu niosą ze sobą. SWD są jedynie narzędziami, które mogą być wykorzystywane przez odpowiednio przygotowanych i uprawnionych użytkowników. W pracy z SWD konieczne jest również uwzględnienie ergonomicznych i psychologicznych warunków procesów podejmowania decyzji. Krytycznym pod tym względem przykładem są systemy kierowania ruchem lotniczym w przestrzeni powietrznej, a w szczególności w bezpośrednim otoczeniu portów lotniczych. Przetwarzanie i udostępnianie wiedzy za pośrednictwem SWD musi się odbywać z zachowaniem wymaganych prawem procedur bezpieczeństwa fizycznego i ochrony danych osobowych.

Przedstawione we wprowadzeniu elementy szans i zagrożeń, które niesie ze sobą wykorzystanie SWD w przedsiębiorstwie, zostaną bliżej przedstawione w kolejnych podrozdziałach.

### 8.1.1. Współczesne kierunki rozwoju i obszary zastosowań SWD

Współczesne systemy SWD powinny się charakteryzować [Power 2007, s. 2–4]:

- szerokim zakresem zastosowań i możliwością swobodnego dopasowania funkcjonalności SWD do indywidualnych potrzeb użytkownika — każdy przypadek decyzyjny powinien być traktowany przez system indywidualnie i dla każdego powinien być przygotowywany dedykowany model analityczny;
- szybkim i swobodnym dostępem do danych — bez względu na rozmiar zasobów oraz miejsce ich fizycznego składowania, a systemy SWD ukierunkowane

---

<sup>1</sup> SWOT (ang. *Strengths – Weaknesses – Opportunities – Threats*) to technika analityczna planowania strategicznego, która kategoryzuje posiadane informacje w czterech grupach: słabe i mocne strony oraz szanse i zagrożenia.

na dane muszą także umożliwiać tworzenie składnic danych historycznych, dedykowanych przetwarzaniu analitycznemu;

- możliwie krótkim czasem wdrożenia systemu — skrócenie czasu przygotowania projektu i budowy SWD jest częściowo zasługą wykorzystania nowoczesnych technologii informatycznych, ale ważną rolę odgrywa także możliwość skorzystania z technik szybkiego prototypowania, gotowych szablonów struktury systemu oraz aplikacji branżowych, dostarczających gotowe rozwiązania dedykowane określonej docelowej grupie zastosowań;
- błyskawiczną odpowiedzią systemu — obecnie opóźnienia transmisji obrazu i dźwięku, wyszukiwania danych czy przekazywania wyników modeli analitycznych powinny być dla użytkownika niezauważalne;
- integracją z systemami transakcyjnymi — SWD nie może stanowić odizolowanej aplikacji, ponieważ decyzje podejmowane w przedsiębiorstwie wymuszają konieczność objęcia swoim zakresem coraz szerszego obszaru zarządzania w czasie rzeczywistym; dlatego system SWD musi również uczestniczyć w tworzeniu jednolitego i znormalizowanego środowiska przetwarzania transakcji;
- niskim kosztem użytkowania systemu — koszty projektowania, wdrożenia oraz eksploatacji systemów SWD nie maleją mimo pojawiających się na rynku rozwiązań typu open source w obszarze narzędzi Business Intelligence oraz systemów CRM;
- wielodostępem — SWD musi być platformą łączącą wszystkie osoby podejmujące decyzje, ułatwiającą ich wzajemny kontakt oraz integrującą działania w procesie podejmowania decyzji;
- wszechstronnością — systemy SWD powinny być tak przygotowane, aby umożliwiać wspieranie wszystkich decyzji w przedsiębiorstwie i dostarczać rozwiązań wszystkich pojawiających się problemów decyzyjnych;
- działaniem w czasie rzeczywistym — coraz więcej użytkowników korzysta z rozwiązań SWD na bieżąco, w czasie podejmowania działań i coraz więcej aplikacji daje taką możliwość;
- przyjazną obsługą — system SWD powinien być narzędziem intuicyjnym, łatwym w obsłudze i zrozumiałym dla użytkownika; np. wyniki analiz powinny być sortowane w celu ułatwienia użytkownikowi dotarcia do tych najbardziej interesujących;
- wizualizacją wyników — narzędzia graficzne udostępniane za pośrednictwem interfejsu użytkownika znacząco ułatwiają interpretację otrzymanych z systemu wyników, zwłaszcza gdy użytkownik ma możliwość dostosowywania wyglądu pulpitu czy raportu do swoich indywidualnych wymagań.

Szerokie spektrum wymagań kierowanych do rozwiązań klasy SWD wymaga korzystania z najnowszych osiągnięć techniki i nauki, wśród których należy wyróżnić metody sztucznej inteligencji. Systemy dążące do osiągnięcia synergii między wspomaganiem decyzji a sztuczną inteligencją noszą nazwę inteligentnych SWD.

Rozszerzają one możliwości klasycznego modelu SWD o następujące elementy [Wolny, Zadora 2008, s. 4]:

- możliwość korzystania z porady eksperckiej,
- wyjaśnienie wyników wnioskowania,
- inteligentne wspomaganie decyzji,
- pomoc przy formułowaniu zapytań,
- inteligentne wspomaganie w procesie budowy modelu.

Jednym z kierunków rozwoju inteligentnych SWD jest łączenie ze sobą różnych metod przetwarzania, wnioskowania i ekstrahowania wiedzy w jednym hybrydowym systemie doradczym (HSWD). Znane są dwa podstawowe podejścia do tworzenia systemów hybrydowych [Rudas 2002]: inteligencja obliczeniowa (ang. *Computational Intelligence* — CI) oraz obliczenia miękkie (ang. *Soft Computing* — SC).

Według [Duch, Mandziuk 2004] inteligencja obliczeniowa to dziedzina nauki, zajmująca się rozwiązywaniem problemów, które nie dają się przedstawić w postaci algorytmu obliczeniowego. Częścią inteligencji obliczeniowej jest sztuczna inteligencja, zajmująca się modelowaniem wiedzy i rozwiązywaniem niealgorytmizowanych problemów w oparciu o symboliczną reprezentację wiedzy.

**SWD jest obliczeniowo inteligentny**, jeżeli zawiera elementy rozpoznawania wzorców, jest odporny na błędy obliczeniowe, wykazuje zdolność dostosowywania się do zaistniałych warunków oraz posiada szybkość reakcji zbliżoną do reagowania człowieka [Wolny, Zadora 2008, s. 6]. W SWD obliczeniowo inteligentnych wykorzystywane są takie techniki, jak: sieci neuronowe, algorytmy genetyczne czy programowanie ewolucyjne.

**Obliczenia miękkie** różnią się od tradycyjnych **obliczeń twardych** (ang. *hard computing*) tym, że umożliwiają uwzględnienie niepewności, braku precyzji, przybliżeń i częściowej prawdy [Zadeh 1994]. Podstawowym wzorcem dla obliczeń miękkich jest ludzki umysł.

Podejście do obliczeń miękkich koncentruje się na tworzeniu systemów przetwarzających wiedzę i poza metodami wykorzystywanymi przez obliczenia inteligentne stosuje także elementy teorii: uczenia maszynowego<sup>2</sup>, chaosu<sup>3</sup> i wnioskowania probabilistycznego<sup>4</sup> [Wolny, Zadora 2008, s.7].

Chcąc wskazać obszary zastosowań SWD, należałoby stwierdzić ogólnie — są one wszędzie tam, gdzie podejmowane są decyzje o wysokim stopniu złożoności

---

<sup>2</sup> Uczenie maszynowe — metoda doskonalenia się systemu na podstawie zgromadzonego doświadczenia (czyli danych) i nabywania na tej podstawie nowej wiedzy.

<sup>3</sup> Teoria chaosu — nauka opierająca się na założeniu, iż możliwe jest dokonywanie pomiarów, kontrolowanie lub odtwarzanie matematycznie nieprzewidywalnego zachowania się układów lub przebiegu zjawisk.

<sup>4</sup> Wnioskowanie probabilistyczne — wnioskowanie stosowane w modelach sztucznej inteligencji, gdzie na podstawie niekompletnej i często niepewnej wiedzy, z zadaniem poziomem prawdopodobieństwa, określa się prawdziwość faktów.



i niepewności, a to oznacza wykorzystanie rozwiązań SWD we wszystkich dziedzinach gospodarki, na każdym poziomie zarządzania. Dzięki SWD przedsiębiorstwa mogą na bieżąco monitorować i korygować realizację procesów biznesowych, usprawniając ich przebieg i ograniczając liczbę powstających błędów i usterek. SWD znajdują zastosowanie także przy optymalizacji planowania i harmonogramowania produkcji oraz dostaw, co wiąże się bezpośrednio z lepszym wykorzystaniem posiadanych zasobów.

Analiza finansowa, realizowana za pomocą SWD, pozwala na bieżąco śledzić wyniki przedsiębiorstwa oraz podejmować optymalne decyzje dotyczące jego działalności i rozwoju. SWD są wykorzystywane do oceny poziomu ryzyka finansowego, wykrywania fałszerstw i oszustw czy prognozowania sytuacji finansowej. Badanie zyskowności produktów i usług przez SWD pomaga podejmować właściwe decyzje dotyczące kształtowania ich portfela; przygotowania efektywnej strategii marketingowej oraz strategii sprzedaży. Systemy SWD mogą współuczestniczyć w znajdowaniu segmentów klientów, prognozowaniu poziomu ich zadowolenia i lojalności wobec firmy oraz przeciwdziałaniu odchodzeniu do konkurencji. Dobre poznanie posiadanych klientów pomaga obniżyć koszty marketingu, przy jednoczesnym wzroście stopnia jego skuteczności.

SWD są szeroko stosowane w diagnostyce medycznej, technicznej czy finansowej. Pomagają odnaleźć rozwiązania, wzorce, opisy problemów, sytuacji i zjawisk, które do tej pory nie miały wytłumaczenia. W sposób najbliższy ludzkiemu umysłowi wspomagają procesy wnioskowania tam, gdzie inne narzędzia i metody nie są w stanie wspomóc działań człowieka.

## 8.1.2. Wielokryterialne podejmowanie decyzji

**Podejmowanie decyzji** jest to proces wyboru decyzji w celu rozstrzygnięcia określonego problemu (por. podrozdział 1.2). Wielokryterialny problem decyzyjny dotyczy sytuacji, w której przez podjęcie decyzji decydent chce osiągnąć cel, wyznaczony poprzez wiele kryteriów. Warianty decyzyjne, przed ich ostatecznym wyborem, są oceniane ze względu na co najmniej dwa kryteria. Przy czym kryteria te mogą być wzajemnie sprzeczne — stąd poprawienie oceny jednego z nich może powodować pogorszenie innego lub innych [Stachowiak 2002]. Wtedy wybór decyzji jest poszukiwaniem wariantu, który byłby akceptowalny ze względu na każde kryterium oceny. W celu znalezienia takiej decyzji należy zastosować **metody optymalizacji wielokryterialnej**.

**Problem decyzyjny**, rozpatrywany jako **zadanie optymalizacji wielokryterialnej**<sup>5</sup>, można przedstawić w postaci wzoru [Łodziński 2006]:

$$\max_x \{f_1(x), \dots, f_m(x) : x \in X_0\}, \quad [8.1]$$

<sup>5</sup> Optymalizacja wielokryterialna — dział badań operacyjnych zajmujących się wyznaczaniem optymalnej decyzji w przypadku, kiedy istnieje więcej niż jedno kryterium wyboru.

gdzie:

- $x \in X_0$  — wektor zmiennych decyzyjnych,
- $\{f_1, \dots, f_m\}$  — funkcja wektorowa, która przyporządkowuje każdemu wektorowi zmiennych decyzyjnych  $x \in X_0$  jego wektor ocen  $y = f(x)$ ,
- $X_0$  — zbiór decyzji dopuszczalnych.

Uwzględniając dziedzinę problemu decyzyjnego, przyjętą przez decydenta strategię rozwiązania problemu oraz sformułowane cele decyzyjne, problematyka wielokryterialnej analizy decyzyjnej skupia się na zagadnieniach [Trzaskalik 2006]:

- wyboru, czyli wyznaczenia możliwie najmniej licznego podzbioru wariantów decyzyjnych, które ze względu na rozpatrywane kryteria zostały uznane za najlepsze;
- sortowania, czyli przedstawienia zbioru wariantów decyzyjnych należących do zdefiniowanych przez decydenta kategorii decyzyjnych;
- porządkowania, czyli podziału wariantów decyzyjnych na klasy decyzji porównywalne ze względu na ich ocenę.

W toku badań nad metodami wielokryterialnego podejmowania decyzji powstały dwa podejścia do tego problemu. Są to modele funkcjonalny i relacyjny.

**Model funkcjonalny** wywodzi się z amerykańskiej szkoły wielokryterialnej analizy decyzyjnej [Triantaphyllou 2000]. Podejście reprezentowane przez model funkcjonalny oparte jest na wielokryterialnej teorii użyteczności<sup>6</sup> i zakłada porównywalność wariantów decyzyjnych. Przykładowymi metodami, które wykorzystują model funkcjonalny, są [Ziemba, Piwowski 2008, s. 142]:

- MAUT (ang. *Multiattribute Utility Theory*) — wieloatrybutowa teoria użyteczności zajmująca się agregacją kryteriów w jednej funkcji zbiorczej, a następnie poszukiwaniem maksimum tej funkcji, czyli takiego zbioru wartości kryteriów, dla których wielkość zagregowanej funkcji użyteczności będzie najwyższa z możliwych;
- SMART (ang. *Simple Multiattribute Rating Technique*) — prosta wieloatrybutowa technika rankingowa jest uproszczoną wersją metody MAUT, gdzie funkcja agregacji jest obliczana jako średnia arytmetyczna wag przypisanych poszczególnym kryteriom;
- AHP (ang. *Analytic Hierarchy Process*) — proces analitycznej hierarchizacji, którego celem jest znalezienie rankingu wariantów, uszeregowanych według ich przydatności z punktu widzenia różnych kryteriów; metoda AHP nie wymaga

<sup>6</sup> Teoria użyteczności — zakłada, że osoba podejmująca decyzję dąży do maksymalizacji użyteczności wyniku. Według tej teorii decyzje podejmowane są ze względu na możliwość osiągnięcia zamierzonego celu, a wybór optymalnego rozwiązania opiera się na porównaniu przewidywanego skutku działania z zamierzonym celem.

przypisania wag poszczególnym kryteriom i wariantom, ale operuje wyłącznie na ocenach względnych porównywanych elementów;

- UTA (ang. *Utility Theory Additive*) — addytywna teoria użyteczności, która bazując na metodzie MAUT, rezygnuje z funkcji agregacji kryteriów na rzecz wprowadzenia elementów addytywnego programowania liniowego do poszukiwania optimum wartości funkcji użyteczności.

Model funkcjonalny zakłada, że każde dwa warianty decyzyjne są ze sobą porównywalne, tzn. mając te dwa warianty, zawsze można jednoznacznie stwierdzić, który z nich jest lepszy. Tymczasem są sytuację, kiedy wynik takiego porównania nie jest jednoznaczny. Wtedy rozwiązaniem staje się zdefiniowanie relacji występujących między porównywanymi wariantami decyzyjnymi, które mogą oznaczać:

- nierozróżnialność — warianty są nierozróżnialne, czyli równoważne sobie;
- silną preferencję — pierwszy wariant jest silnie preferowany w stosunku do wariantu drugiego;
- słabą preferencję — pierwszy wariant jest słabo preferowany w stosunku do wariantu drugiego;
- nieporównywalność — warianty są nieporównywalne ze sobą.

**Model relacyjny**, wywodzący się z europejskiej szkoły podejmowania decyzji [Roy 1996], skupia się na trzech pierwszych typach preferencji (nierozróżnialności, słabej lub silnej preferencji), budując w oparciu o nie relację przewyższania. Relacja ta reprezentuje preferencje decydenta w odniesieniu do porównywanych wariantów decyzji i jest wykorzystywana w następujących metodach [Ziembra, Piwowarski 2008, s. 143]:

- Electre (fr. *Elimination Et Choix Traduisant la Réalité*) — opiera się na porównywaniu alternatyw decyzyjnych parami i tworzeniu na tej podstawie zbiorów zgodności i niezgodności alternatyw, co z kolei prowadzi do zbudowania grafu przedstawiającego uporządkowanie alternatyw ze względu na preferencje decydenta;
- Promethee (ang. *Preference Ranking Organization Method for Enrichment Evaluations*) — metoda wielokryterialnej analizy preferencji, która uwzględnia wartości poszczególnych kryteriów preferencji oraz określa intensywność tych preferencji ze strony decydenta (wagi); na tej podstawie warianty decyzyjne są rankingowane i może zostać wytypowany najlepszy z nich.

Każda z prezentowanych metod doczekała się kolejnych generacji (np. Electre I, Electre II, Electre III, Promethee I, Promethee II itd.), tworząc obecnie grupę metod o wspólnej bazie teoretycznej, ale różniących się rozwiązaniami szczegółowymi.

### 8.1.3. Psychologiczne aspekty podejmowania decyzji

Wśród modeli podejmowania decyzji wyróżniamy modele [Kahneman, Slovic, Tversky 1982]:

- formalne — matematyczne, normatywne,
- behawioralne — psychologiczne, deskryptywne.

Wykorzystanie modeli formalnych zakłada, że decydent działa w sposób racjonalny, czyli jego postępowanie jest zgodne z przyjętą teorią i skutecznie prowadzi go do wyznaczonego celu. Tymczasem decydent kieruje się często kryterium chwilowej lub dostatecznej korzystności, a ocena tego kryterium może się zmieniać w czasie. Podejście psychologiczne w procesie podejmowania decyzji stanowi próbę wyjaśnienia i uporządkowania teorii nieracjonalności zachowań decydentów, wynikających z cech osobowościowych lub sytuacji, w której znajduje się decydent.

Sytuacją, w rozumieniu psychologicznym, jest uporządkowany ciąg par, składający się z interakcji umysłu i otoczenia [Tyszka, Zaleśkiewicz 2002]. Na sytuację składają się bodźce, sygnały i informacje pochodzące z otoczenia oraz schematy pojęciowe i profile doświadczenia istniejące w osobie. Każdą sytuację charakteryzuje złożoność i zmienność, które są wymiarami otoczenia, ale również niepewność, będąca wymiarem umysłu osoby podejmującej decyzję. Sytuacja decyzyjna sprowadza się do rozwiązania problemu niepewności.

Niepewność to stan umysłu poznającego (kategoryzującego) otoczenie i działającego w nim. Oznacza sytuację, w której określone decyzje mogą spowodować różne skutki, przy czym nie są znane prawdopodobieństwa wystąpienia poszczególnych z nich. Bezpośredni wpływ na stopień niepewności ma złożoność i zmienność sytuacji decyzyjnej. Niepewność może być wewnętrzna, wynikająca ze świadomości własnych ograniczeń typu [Kahneman, Slovic, Tversky 1982]: poznawczego, umysłowego, kompetencyjnego lub braku doświadczenia. Niepewność może być także zewnętrzna [Kahneman, Slovic, Tversky 1982]:

- wynikająca z braku lub niekompletności wiedzy o obiektywnie istniejących cechach i mechanizmach powstawania zdarzeń;
- reprezentowana w umyśle jako rozkład zdarzeń lub jako niepowtarzalne (wyjątkowe) zdarzenie jednostkowe.

Ze względu na **stopień niepewności podejmowania decyzji** sytuacje decyzyjne dzielimy na: pewne, ryzykowne i niepewne. Sytuacje są pewne wtedy, gdy wynik decyzji całkowicie zależy od działań decydenta. Sytuacja ryzykowna występuje wtedy, kiedy wynik decyzji jest uzależniony nie tylko od działań decydenta, ale również od pewnych zdarzeń zewnętrznych, na które decydent nie ma wpływu, ale które potrafi ocenić. Sytuacje niepewne są wtedy, kiedy wiedza decydenta o zdarzeniach warunkujących następstwa decyzji jest tak mała, że nie jest w stanie dokonać żadnych dokładniejszych oszacowań tych zdarzeń. Większość rzeczywistych sytuacji decyzyjnych ma charakter ryzykowny.

Sytuacji decyzyjnej zawsze towarzyszy determinanta w postaci **stopnia swobody wyboru**, która dzieli sytuacje decyzyjne na:

- otwarte (szerokie) — kiedy decydent ma poczucie dużej swobody wyboru rozwiązania, sam określa zbiór wariantów działania i kryteria ich oceny;
- zamknięte (wąskie) — kiedy swoboda wyboru decyzji jest ograniczona przez zbiór wariantów działania lub kryteria ich oceny.

Podejście psychologiczne uwzględnia również zachowania decydentów w trakcie podejmowania decyzji, zakładając że:

- ich działania mają często charakter spontaniczny, nie do końca przemyślany, a nawet chaotyczny;
- występuje ograniczenie zachowania — postawa pasywna jest również rodzajem wyboru działania;
- zachowania decydentów zawsze mają sens, choć może on być ukryty, a nawet nieuświadomiony przez nich;
- zachowania decydentów mają dwoisty charakter: ofensywny — dążący do zmiany i poprawy własnej sytuacji oraz defensywny — mający na celu utrzymanie *status quo*.

Biorąc pod uwagę wymienione stwierdzenia, proces podejmowania decyzji w podejściu psychologicznym przestaje być poszukiwaniem absolutnego optimum, a staje się narzędziem służącym do osiągnięcia wyznaczonego, bieżącego celu z uwzględnieniem zdefiniowanych warunków i kryteriów oceny działania. Należy również pamiętać, że jest to proces ciągły i zmienny w czasie, podczas którego decydent podejmuje optymalną jego zdaniem decyzję.

Wybór właściwej decyzji może być wspierany przez różne strategie. Według J. Kozielskiego [Kozielski 1977] najbardziej optymalna jest strategia SEU (ang. *Subjective Expected Utility*), czyli maksymalizacja subiektywnie oczekiwanej użyteczności, która zaleca wybór działania, pozwalającego najlepiej połączyć to, co prawdopodobne z tym, co subiektywnie wartościowe dla decydenta. Strategia SEU bierze pod uwagę dwie zmienne: prawdopodobieństwo subiektywnie antycypowanych wyników oraz ich użyteczność. Z kolei T. Tyszka [Tyszka 1986] proponuje zbiór reguł decyzyjnych, wspomagających decydenta w procesie wyboru decyzji:

- reguła dominacji — wybór wariantu korzystniejszego od pozostałych pod względem przynajmniej jednej cechy oraz osiągającego możliwie najlepsze wyniki oceny dla pozostałych cech;
- reguła koniunkcji — wybór wariantu, dla którego wszystkie ocenione cechy osiągają lub przekraczają przyjęte wartości progowe;
- reguła dysjunkcji — wybór wariantu, w którym przynajmniej jedna spośród ocenianych cech przekracza lub osiąga przyjętą wartość progową;
- reguła leksykograficzna — wybór wariantu najkorzystniejszego pod względem najważniejszej cechy, a następnie możliwie najkorzystniejszego dla każdej kolejnej cechy w sposób uwzględniający hierarchię ważności cech;

- reguła eliminacji — odrzucenie wszystkich tych wariantów, które dla określonych cech nie osiągnęły założonych wartości progowych, wykonywane zgodnie z przyjętą hierarchią ważności cech;
- reguła maksymalizacji — wybór jednego z dwóch porównywanych wariantów ze względu na maksymalną liczbę posiadanych korzystnych cech, którymi górują nad konkurentem;
- reguła maksymalizacji subiektywnej użyteczności — wybór wariantu osiągającego najwyższą wartość sumy ważonej użyteczności dla wszystkich ocenianych cech.

Psychologiczne aspekty podejmowania decyzji, skupiające się na zachowaniach jednostki w procesie podejmowania decyzji, należałoby uzupełnić także o aspekty socjologiczne, uwzględniające uwarunkowania społeczne i grupowe, takie jak: miejsce decydenta w organizacji, wpływ stylu kierowania organizacją, podejmowanie decyzji w grupie oraz związane z tym konflikty ([Watkins 2005], [Surma 2010] i inne).

#### 8.1.4. Prawne aspekty zastosowań SWD

Zastosowanie systemu SWD w przedsiębiorstwie nieodłącznie wiąże się z problematyką zarządzania własnością intelektualną. **Zarządzanie własnością intelektualną** obejmuje zarówno wytworzenie przedmiotu ochrony (np. wiedzy powstającej w skutek wykorzystywania SWD), jak i zapewnienie jego ochrony. Podstawowymi przepisami regulującymi prawo własności intelektualnej w Polsce są [Łazewski, Gołębiowski 2006, s. 8]:

- Ustawa z dnia 30 czerwca 2000 r. — Prawo własności przemysłowej (Dz.U. z 2003 r. nr 119, poz. 1117 z późn. zm.);
- Ustawa z dnia 4 lutego 1994 r. o prawie autorskim i prawach pokrewnych (Dz.U. z 2000 r., nr 80, poz. 904; Dz. U. z 2005 r., nr 3, poz. 12).

W zakresie praw własności intelektualnej funkcjonują także przepisy o charakterze międzynarodowym — w szczególności w formie umów międzynarodowych. W ramach Unii Europejskiej bardzo istotnym źródłem prawa są rozporządzenia i dyrektywy. Wszystkie źródła prawa, dotyczące własności intelektualnej, można podzielić na dwie grupy [Łazewski, Gołębiowski 2006, s. 8]:

- 1) przepisy ustanawiające minimalny poziom ochrony między poszczególnymi krajami:
  - Konwencja Paryska o ochronie własności przemysłowej (Dz.U. z 1975 r., nr 9, poz. 51),
  - Porozumienie w sprawie handlowych aspektów praw własności intelektualnej (TRIPS),

- załącznik do umowy ustanawiającej Światową Organizację Handlu (Dz.U. z 1996 r., nr 32, poz.143),
  - dyrektywy Unii Europejskiej dotyczące harmonizacji przepisów np. w zakresie znaków towarowych, prawa patentowego czy praw autorskich;
- 2) przepisy zmierzające do stworzenia ponadnarodowych instytucji umożliwiających uzyskanie w jednym postępowaniu prawa wyłącznego w kilku krajach:
- Porozumienie Madryckie o międzynarodowej rejestracji znaków (Dz.U. z 1993 r., nr 116, poz. 514) i protokół do tego porozumienia (Dz.U. z 2003 r., nr 13, poz. 129),
  - Układ o współpracy patentowej (Układ PCT) (Dz.U. z 1991 r., nr 70, poz. 303),
  - Konwencja o udzielaniu patentów europejskich (Konwencja o patencie europejskim, Dz.U. z 2004 r., nr 79, poz. 737),
  - Rozporządzenie o wspólnotowym znaku towarowym (Rozporządzenie Rady WE nr 207/2009 z dnia 26 lutego 2009 r. w sprawie wspólnotowego znaku towarowego).

Najprostszą formą ochrony dóbr niematerialnych jest utrzymanie ich w tajemnicy. Istnieją jednak pewne minimalne warunki, jakie należy spełnić, aby ochrona tajemnicy przedsiębiorstwa mogła mieć zastosowanie. Są nimi możliwość określenia przedmiotu ochrony oraz kontynuowania podjęcia należytych starań w celu utrzymania stanu jego poufności.

Najlepszą formą ochrony jest sytuacja, kiedy informacje o przedmiocie ochrony nie są nikomu udostępniane. Z praktycznego punktu widzenia jest to zazwyczaj niemożliwe do spełnienia. Nawet w pierwszych fazach cyklu życia projektu, kiedy ochrona tajemnicy jest najintensywniej stosowana i przestrzegana, pojawia się jednak konieczność odtajnienia projektu w celu jego prezentacji podwykonawcom czy inwestorom. Wtedy koniecznym zabezpieczeniem staje się podpisanie zobowiązań o poufności. Zobowiązanie takie może być również elementem umowy związanej ze współpracą w ramach projektu.

Innym rozwiązaniem jest skorzystanie z prawa patentowego. W przeciwieństwie do ochrony rozwiązań, postrzeganej jako tajemnica przedsiębiorstwa, podstawą ochrony w przypadku prawa patentowego jest pełne przedstawienie chronionego rozwiązania w dokumencie zwanym opisem patentowym. W krajach Unii Europejskiej funkcjonuje także wymóg technicznego charakteru wynalazku, który odnosi się np. do wszystkich projektów informatycznych. **Patent** jest prawem udzielanym na ograniczony okres (do 20 lat), jeżeli zostaną spełnione następujące warunki ochrony patentowej [Łazewski, Gołębiowski 2006, s. 10–11]:

- nowość — rozwiązanie nie może być przed datą zgłoszenia podane do wiadomości publicznej;
- poziom wynalazczy (nieoczywistość) — wynalazek nie może być jedynie kombinacją lub rozszerzeniem rozwiązań już wcześniej znanych;

- stosowalność przemysłowa — wymóg odnosi się do możliwości powtarzalnego zrealizowania wynalazku.

Ochronie na podstawie prawa autorskiego podlegają wszelkiego rodzaju utwory: literackie, filmowe, programy komputerowe, zdjęcia, rzeźby, mapy, utwory sztuki przemysłowej lub utwory architektoniczne. W Polsce ochroną własności intelektualnej zajmują się: Urząd Patentowy Rzeczypospolitej Polskiej, Polska Izba Rzeczników Patentowych, Urząd Harmonizacji Rynku Wewnętrznego. Prawo autorskie pozostaje wciąż podstawowym narzędziem ochrony w zakresie oprogramowania komputerowego. Ochrona dotyczy w tym wypadku kodu programowego, który chroniony jest w sposób analogiczny do tekstu literackiego. Na podstawie prawa autorskiego uprawniony może zakazać kopiowania i rozpowszechniania wytworu, a także ma kontrolę nad powstawaniem utworów zależnych. Jeżeli jednak właściciel kodu chce w pewnym zakresie, na pewnych warunkach i na określony czas udostępnić swój wytwór, to może to zrobić na podstawie umowy licencyjnej.

**Licencja** jest umową dwustronną, w której jedna ze stron (licencjodawca) upoważnia drugą (licencjobiorcę) do wkroczenia w swój własny zakres wyłączności. Zakres licencjonowanych uprawnień może się rozciągać od wszystkich praw przysługujących licencjodawcy do bardzo dokładnie określonego prawa, np. sprzedaży danego towaru na wąsko zakreślonym obszarze geograficznym. Pozwala to na zbudowanie sieci licencjobiorców z odpowiednio ukształtowanymi uprawnieniami.

Problem bezpieczeństwa teleinformatycznego, dotyczący również aspektu projektowania, budowy i eksploatacji systemów SWD, na świecie jest regulowany przez standardy, takie jak: OECD (ang. *Organisation for Economic Cooperation and Development*) oraz ustawodawstwo krajowe typu *The Computer Security Act* oraz *FISMA* (ang. *The Federal Information Security Management Act*) przyjęte w USA. Od 23 listopada 2001 r. w państwach Unii Europejskiej obowiązuje Konwencja Rady Europy o cyberprzestępczości. Konwencja definiuje przestępstwo komputerowe oraz jego szczegółowe rodzaje, a także środki organizacyjno-prawne dedykowane do przeciwdziałania i zwalczania cyberprzestępstw w państwach członkowskich. W Polsce podstawą do walki z cyberprzestępczością jest kodeks karny z 1997 r. oraz regulacje pozakodeksowe (np. ustawa o ochronie danych osobowych).

Poza regulacjami prawnymi istnieją także **kodeksy etyki komputerowej**, których zadaniem jest wypracowanie w społeczeństwie informacyjnym etycznych zasad współpracy i współdzielenia cyberprzestrzeni (np. 10 przykazań etyki komputerowej, które powstały w *The Computer Ethics Institute* w Waszyngtonie w USA [Fitzpatrick, Bronstein 2006, s. 116]. D. Gotterbarn [Gotterbarn 2001] zaproponował wprowadzenie identyfikacji ewentualnych negatywnych skutków społecznych i organizacyjnych tworzonego projektu informatycznego i wyznaczanie działań niwelujących te problemy w formie *SoDIS* (ang. *Software Development Impact Statement*). Projektowanie według *SoDIS* zachęca wykonawców do troski o ludzi, grupy i organizacje (tzw. udziałowców projektu) na każdym etapie jego powstawania. Celem tych



wszystkich działań jest wprowadzenie kanonów etycznych oraz przyjęcie zasad etyki komputerowej nie tylko na poziomie projektowania systemów, ale także na poziomie kształcenia inżynierów-projektantów i inżynierów-programistów [Kozielecki 1977].

## 8.2. Rola wiedzy eksperckiej w SWD

Wraz z rozwojem technologii informacyjnej oraz wzrostem możliwości gromadzenia i udostępniania danych pojawiła się koncepcja upowszechnienia wiedzy eksperckiej. Jest ona zasobem rzadkim i drogim, powstałym w wyniku zmagazynowania zasobów wiedzy rzeczoznawców w pamięci komputera, aby mogło z niej korzystać znacznie więcej osób i instytucji niż miało to miejsce przy rozpowszechnianiu wiedzy metodami tradycyjnymi.

Stąd efektywną metodą upowszechniania wiedzy eksperckiej stały się właśnie systemy ekspertowe, nazwane tak, gdyż w systemach tych zapisana jest wiedza rzeczoznawców w postaci reguł decyzyjnych. Proces translacji wiedzy ekspertów z określonych dziedzin na jej reprezentację w komputerze dokonywany jest przez inżynierów wiedzy. Według prekursora tej koncepcji deWitta systemy ekspertowe powinny być, z założenia, bardziej kompleksowe niż modele symulacyjne, ponieważ „cegiełki wiedzy” nie muszą być w nich poukładane jedna na drugiej w sposób uporządkowany, jak w modelach symulacyjnych, ale naśladują niekompletną i nieściłą wiedzę charakterystyczną dla umysłów ludzkich. Zdaniem tego autora, niedobór ekspertów-rzeczoznawców z poszczególnych dziedzin powoduje, że rozwój metod komputerowych zastępujących ekspertów jawi się jako imperatyw [McCown 2002].

Dla potrzeb aktualizacji i uwiarygodnienia wiedzy eksperckiej należy ją w sposób ciągły i usystematyzowany weryfikować w zakresie poprawności i skuteczności. W związku z tym powstaje potrzeba oceny jej jakości i społecznej użyteczności.

Wiedza ekspertów wykorzystywana w SWD jest niewymiernym zasobem, którego wartość staje się realna w sytuacji, kiedy ekspert jest niedostępny. Dlatego też pojawiła się konieczność kodyfikacji wiedzy rzeczoznawców w specjalnej bazie wiedzy SWD, nazywanej bazą przypadków.

### 8.2.1. Wiedza eksperta zapisana w bazie przypadków

**Metoda CBR** (ang. *Case Based Reasoning*), czyli metoda oparta na bazie przypadków, jest wykorzystywana w procesie prognozowania dokonywanego na podstawie doświadczeń z przeszłości zapisywanych w bazie przypadków. SWD oparte na CBR mogą być stosowane do diagnozowania, przewidywania, oceny, planowania, projektowania lub tworzenia harmonogramów. W ramach tej metody wnioskowanie jest realizowane na podstawie przypadków wykorzystujących specyficzną wiedzę ekspercką nabytą w sytuacjach decyzyjnych z przeszłości, zwanych przypadkami (ang. *cases*) [Zadora 1999]. CBR są także określane jako systemy przydatne w rozwiązywaniu nowych problemów poprzez adaptację rezultatów, które były wykorzy-

stane podczas rozwiązywania starych kwestii [Riesbeck 1989]. Nowy problem jest rozstrzygany poprzez odnalezienie podobnego do niego przypadku w zbiorze i zastosowanie rozwiązania skojarzonego z tym odnalezionym przypadkiem. Cenną właściwością CBR jest zdolność do uczenia się systemu opartego na tej metodzie, co odbywa się poprzez gromadzenie rozwiązań problemów z przeszłości i udostępnianie ich do rozwiązywania nowych problemów. Przypadki będące faktami bazy wiedzy systemów CBR obejmują zazwyczaj wiedzę specyficzną dla danego problemu. Przypadki zgromadzone w pamięci nie muszą zawierać kompletnej wiedzy dla danej dziedziny [Wrona, Zadora 1997]. Na cykl działania systemu realizującego algorytm CBR składają się cztery procesy:

- 1) odzyskanie (ang. *retrieve*) z bazy podobnych do opisanego problemu przypadków;
- 2) ponowne użycie (ang. *reuse*) sugerowanego rozwiązania podobnego przypadku;
- 3) skorygowanie i dopasowanie (ang. *revise*) rozwiązania do nowego problemu, jeśli to konieczne;
- 4) zachowanie (ang. *retain*) nowego rozwiązania, jeśli było ono zatwierdzone i uzasadnione [Aamodt, Plaza 1994], [Pokojski 2003].

Problem znajdujący się na wejściu systemu jest porównywany z przypadkami zgromadzonymi w ich bazie. Wyodrębniany jest z bazy jeden lub więcej przypadków spełniających kryterium podobieństwa do przypadku wejściowego. Jeżeli podobieństwo wyodrębnionych przypadków nie jest zbyt duże, może się okazać konieczna odpowiednia adaptacja. Po jej wprowadzeniu gotowe rozwiązanie jest zapisywane w bazie.

W klasycznych systemach CBR wiedza zapisana w bazie przypadków ma postać uporządkowanych, wydzielonych porcji informacji. Obiekty znajdujące się na wyższym stopniu uogólnienia, obejmujące bazy przypadków wraz z metodami dostępu i aktualizacji, zwane są modelami. Wyróżnia się m.in. dwa takie modele: pamięci dynamicznej oraz kategorii przypadków [Zadora 1999]. Poszczególne atrybuty opisują specyficzne wartości przypadku. Przypadek jest definiowany za pomocą cech i repertuarów ich wartości, a traktowany jest jako para: problem–rozwiązanie. Opis cech dotyczy zarówno problemu, jak i rozwiązania. Operator bazy przypadków SWD przeprowadza eksperyment i ustala miary podobieństwa dla każdej pary przypadków.

### 8.2.2. Rola przestrzeni tolerancji w przetwarzaniu wiedzy eksperta

W metodzie CBR konstruowana jest **przestrzeń tolerancji** (przestrzeń klas podobnych przypadków) w procesie porównywania wartości wybranych cech przypadków charakterystycznych dla określonej sytuacji problemowej. Kolejny rozpatrywany

przypadek albo przynależy do jednej z istniejących już klas, albo też — ze względu na przekroczenie granic tolerancji podobieństwa — tworzy nową klasę. Precyzyjnie określona i eksperymentalnie potwierdzona reakcja podejmującego decyzję na zaistniałą sytuację odpowiada każdej klasie przypadków.

Jednym z pozytywnie zweryfikowanych zastosowań metody CBR jest jej wykorzystanie do szacowania kosztów wytwarzania w fazie projektowania. Przeprowadzone badania pozwalają wnioskować, iż metoda CBR daje dobre wyniki oszacowań, których błędy kształtują się poniżej poziomu 20%, co jest górną — ogólnie przyjętą — granicą dla wartości szacowanych. Metodę CBR charakteryzuje duża elastyczność, gdyż umożliwia szacowanie kosztów wytwarzania na podstawie od dwóch do siedmiu parametrów technicznych opisujących projekt, w zależności od tego, jakimi informacjami dysponujemy. Metoda CBR, ze względu na dynamiczną funkcję dopasowania, która uwzględnia nowe przypadki w bazie, daje za każdym kolejnym użyciem lepsze oszacowania. Dokładność oszacowanych kosztów zwiększa się wraz ze wzrostem danych o nowym projekcie<sup>7</sup>.

### 8.2.3. Formułowanie zapytań — prezentacja wyводу eksperckiego

**Formułowanie zapytań** w systemie ekspertowym opiera się na weryfikacji prawdziwości przesłanek w procesie wnioskowania (JEŚLI przesłanka, TO konkluzja). Zatem w przesłankach zawartych w formie reguł decyzyjnych musi być odzwierciedlona w miarę możliwości pełna i aktualna dziedzinowa wiedza eksperta lub ekspertów. Proces translacji opinii ekspertów na temat rozwiązania określonego problemu leży w gestii inżynierii wiedzy, która określa ogólne zasady takiego procesu. Wiedza systemowa powstaje w wyniku procesu kodyfikacji wiedzy eksperckiej, którego celem jest zebranie w systemie ekspertowym wiedzy pochodzącej nierzadko z wielu rozproszonych źródeł w taki sposób, aby tworzyła pełny i spójny obraz danego zagadnienia. W ten sposób zostaje opracowany system ekspertowy wspomagający np. procesy przygotowania produkcji, w którym wykorzystuje się wiedzę o strukturze konstrukcyjnej wyrobu, możliwościach technologicznych systemu produkcyjnego oraz normatywy czasów produkcyjnych.

**Wiedza ekspercka** na ogół jest przekazywana w sposób werbalny lub za pomocą notatek i zapisków. **Wiedza systemowa** powinna być systematycznie aktualizowana, w miarę jak zmienia się np. wyposażenie przedsiębiorstwa lub pojawia się nowy asortyment produkcyjny. Jakość systemu ekspertowego zależy wprost od jakości zgromadzonej w nim wiedzy. W procesie tworzenia bazy wiedzy obejmującej dane i informacje wykorzystuje się różne materiały źródłowe oraz konsultacje z ekspertem. Można wyróżnić wiele technik pozyskiwania wiedzy eksperta,

<sup>7</sup> Zob. [www.procax.org.pl/pliki/Referat\\_2009\\_Knosala.pdf](http://www.procax.org.pl/pliki/Referat_2009_Knosala.pdf) (pobranie danych: 16.02.2012).

np. obserwację prowadzoną przez eksperta w miejscu pracy w celu wyrobienia sobie poglądu na złożoność i strukturę problemu. Inną metodą jest dyskusja problemu, obejmująca zestawy pytań zadawanych ekspertowi przez inżyniera wiedzy. Oto przykłady pytań.

- W jaki sposób zdefiniowany problem różni się od prototypowych problemów danego obszaru?
- Jakich danych (w sensie ilościowym i jakościowym) wymaga rozwiązanie określonego problemu?
- Jakie klasy rozwiązań są adekwatne dla danego problemu?
- Czy i w jaki sposób problem może być poddany rozkładowi na nieoddziałujące na siebie podproblemy?
- Jakie obszary (dyscypliny) wiedzy są niezbędne do rozwiązania danego problemu?
- W jaki sposób następuje wyjaśnienie algorytmu rozwiązania problemu?

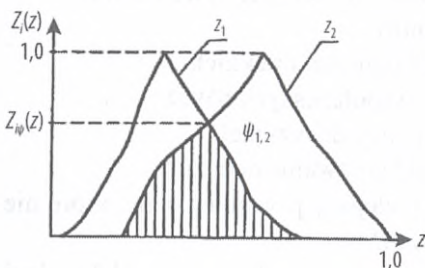
W kolejnej metodzie pozyskiwania wiedzy od rzeczoznawcy ekspert opisuje problem dla każdej realnie możliwej kategorii danych wejściowych podawanych przez użytkownika SWD, co pozwala sformułować prototypowy problem. Na tej podstawie dokonywana jest selekcja strategii rozwiązania problemu. W ramach analizowania problemu inżynier wiedzy prosi eksperta o rozwiązanie wielu problemów, badając tok rozumowania eksperta w tym procesie. Z kolei w ramach doskonalenia systemu ekspert zadaje inżynierowi wiedzy całą gamę problemów do rozwiązania — od bardzo łatwych do bardzo trudnych, weryfikując w ten sposób poprawność przekazu swojej wiedzy. Przy pozyskiwaniu wiedzy metodą instrukcji wykorzystywane są źródła wiedzy wskazane przez eksperta. Udziela on również instrukcji o sposobie ich stosowania w poszukiwaniu rozwiązań problemów decyzyjnych. W wyniku tego wiedza zostaje przekształcona do postaci akceptowanej przez system. W obszarze przygotowania produkcji w przedsiębiorstwie na podstawie tej metody zostaje pozyskana wiedza z różnych norm zakładowych i normatywów branżowych, np. z tabel warunków obróbki.

Jeśli ekspert przedstawia nowe rozwiązanie i wskazuje fakty analogiczne do już rozwiązanych problemów, dokonuje akwizycji wiedzy metodą analogii. Pozyskiwana wiedza ulega takim transformacjom, aby umożliwić opis faktów podobnych do faktów uprzednio zawartych w bazie wiedzy. Przykładem zastosowania tej metody są analogie w projektowaniu procesów technologicznych elementów maszyn. Jedną z nich jest wielkość produkcji, która wpływa na określony typ produkcji (np. jednostkowy lub seryjny). Fakty pozyskane w ten sposób dotyczą doboru materiału wejściowego oraz stanowisk technologicznych do produkcji. Pozyskiwanie wiedzy na podstawie przykładów jest szczególnym przypadkiem uczenia indukcyjnego. Polega ono na generowaniu ogólnego opisu pojęć (klas) na podstawie zbioru przykładów obiektów reprezentujących te pojęcia. Przykłady są dobierane przez eksperta, np. na podstawie wcześniej utworzonych baz danych, a w przedsiębiorstwie są gromadzone w dokumentacji produkcyjnej. Analiza tej dokumentacji pozwala na uzyskanie

wiedzy z danego obszaru, np. wiedzy o projektowaniu struktury procesu produkcyjnego, zapisanej na kartach technologicznych i instrukcyjnych.

Preferowanie stosowania którejś z ww. metod zależy od charakteru rozwiązywanych problemów oraz od wielkości baz wiedzy. Niejednokrotnie też konstruktor systemu musi połączyć metody akwizycji albo też sam opracować własną metodę, co pozwoli mu efektywnie wprowadzać wiedzę do systemu. Dla tworzenia mniejszych i prostych baz wiedzy najczęściej stosowana jest metoda bezpośredniego zapisu wiedzy lub na podstawie obserwacji. Gdy klasa rozwiązywanych problemów decyzyjnych wymaga większych interakcji z ekspertem, wówczas najbardziej odpowiednimi metodami są: pozyskiwanie wiedzy na podstawie instrukcji, analogii i przykładów. W punkcie 6.2.1 dokładniej opisano etapy testowania systemu oraz legalizacji systemu prowadzących do jego udoskonalenia [Mulawka 1997]. Proces uzgadniania opinii ekspertów jest odrębnym zagadnieniem, którego szerzej nie opisano w książce, ale jakość otrzymanych wyników istotnie zależy od jakości ocen ekspertów [Jaźwiński 2000]. W metodzie R. Knosala wykorzystuje się optymalizację wielokryterialną, która pozwala na zastosowanie różnorodnych kryteriów ocenianych na podstawie opinii ekspertów (rysunek 8.1).

**Rysunek 8.1. Przykłady funkcji przynależności otrzymanych w wyniku łączenia ocen dla dwóch wariantów**



Źródło: [Knosala 1989].

Uwzględniane są przy tym kryteria o charakterze deterministycznym, probabilistycznym i rozmytym. Wszystkim zdefiniowanym kryteriom przyporządkowuje się oceny ich ważności w procesie podejmowania decyzji. Najczęściej polega to na porównywaniu kryteriów ze sobą. Na podstawie ocen wystawionych przez każdego z ekspertów powstaje macierz ważności kryteriów według metody opisanej przez Saaty'ego. Następnie opracowywana jest lista wariantów dopuszczalnych rozwiązań problemu decyzyjnego. Eksperti dokonują niezależnej oceny każdego wariantu poprzez przypisanie odpowiedniej wartości punktowej z przedziału założonego przez osobę nadzorującą przebieg całego procesu, określaną mianem eksperta wiodącego.

Na podstawie uzyskanych wartości tworzone są funkcje przynależności z ocenami: górną, dolną i średnią. Kolejny etap polega na łączeniu ocen poszczególnych wariantów i uwzględnieniu wagi kryteriów. Wykorzystuje się do tego funkcję agregującą, która opisuje preferencje danego wariantu ze względu na wszystkie kryteria. Po połączeniu ocen otrzymuje się kolejną funkcję przynależności, która prezentuje stopień preferencji danego wariantu.

Analiza tej funkcji pozwala ocenić, czy dany wariant jest bardziej preferowany (wartość funkcji  $Z_j(z) = 1$  leży bliżej  $z = 1$ ) czy też mniej (wartość funkcji  $Z_j(z) = 1$  leży bliżej  $z = 0$ ). W wyniku nałożenia na siebie wykresów funkcji przynależności dla poszczególnych wariantów (rysunek 8.1) staje się możliwe dostrzeżenie różnic między wariantami. Pozwala to obiektywnie ocenić wybór najlepszego rozwiązania problemu decyzyjnego.

Metoda R. Knosali została zastosowana w implementacji komputerowej, która dzięki prostemu i zrozumiałemu interfejsowi pozwala każdemu z ekspertów przejść przez cały proces podejmowania decyzji.

## 8.3. Pytania i zadania kontrolne

### 8.3.1. Pytania kontrolne

1. Jakimi cechami powinien się charakteryzować nowoczesny system SWD?
2. Scharakteryzuj SWD obliczeniowo inteligentny.
3. Scharakteryzuj SWD wykorzystujący model obliczeń miękkich.
4. Wymień najważniejsze obszary zastosowań współczesnych SWD.
5. Zdefiniuj problematykę wielokryterialnej analizy decyzyjnej.
6. Co to jest działanie racjonalne w procesie podejmowania decyzji?
7. Jakie zachowania decydentów uwzględnia podejście psychologiczne, które nie były brane pod uwagę w modelach formalnych?
8. Wymień podstawowe akty prawne dotyczące ochrony własności intelektualnej w Polsce.
9. Wymień jednostki zajmujące się ochroną własności intelektualnej w Polsce i Unii Europejskiej.
10. Jakie procesy składają się na cykl działania systemu realizującego algorytm CBR?
11. Wymień podstawowe metody pozyskiwania wiedzy eksperta.
12. Scharakteryzuj bliżej wiedzę pozyskiwaną od ekspertów metodą analogii.
13. Jakie są zalety metody CBR w konstruowaniu przestrzeni tolerancji?

### 8.3.2. Przykładowe zadania

1. Zaproponuj zakres zastosowań dla SWD obliczeniowo inteligentnego.
2. Zaproponuj zakres zastosowań dla SWD wykorzystującego model obliczeń miękkich.
3. Dla przykładów opracowanych w zadaniu 1 określ problemy organizacyjne, prawne, psychologiczno-społeczne, które należy rozwiązać, przystępując do wdrożenia takiego rozwiązania.
4. Dla przykładów opracowanych w zadaniu 2 określ problemy organizacyjne, prawne, psychologiczno-społeczne, które należy rozwiązać, przystępując do wdrożenia takiego rozwiązania.
5. Opracuj przykład ilustrujący zastosowanie optymalizacji wielokryterialnej w metodzie R. Knosali.

# Zastosowanie metod sztucznej inteligencji w SWD

W rozdziale przedstawiono najbardziej reprezentatywne dla procesu podejmowania decyzji modele sztucznej inteligencji: sieci neuronowe, algorytmy genetyczne oraz algorytmy mrówkowe. Sieci neuronowe i algorytmy genetyczne wykorzystuje się do rozwiązywania złożonych problemów decyzyjnych, takich jak np.: wybór strategii inwestycyjnej, znajdowanie optymalnej drogi w sieciach logistycznych, wybór optymalnego wariantu wdrożenia systemu informatycznego. Istotą algorytmów genetycznych jest naśladowanie zjawisk zachodzących w przyrodzie ożywionej, w której kolejne pokolenia dziedziczą najkorzystniejsze cechy swych gatunków, przekazując je sobie w postaci kodu genetycznego. Algorytmy mrówkowe naśladowują mechanizm rozpoznawania śladów feromonowych (zapachowych) pozostawianych przez mrówki, które odniosły „sukces” w chaotycznym poszukiwaniu drogi do celu. Algorytmy mrówkowe stosuje się do rozwiązywania problemów decyzyjnych powstających np. w komunikacyjnych lub logistycznych strukturach sieciowych. Algorytmy genetyczne częstokroć stosuje się w procesie wspomaganiania uczenia sieci neuronowej; podobnie algorytmy mrówkowe mogą być łączone z heurystycznymi modelami obliczeń kombinatorycznych — w ten sposób powstają hybrydowe modele decyzyjne (HMD).

## 9.1. Metody sztucznej inteligencji

W ewolucji rozwoju **metod sztucznej inteligencji** [Rutkowski 2012; Smith 1999], z jednej strony, możemy się doszukiwać przesłanek wynikających z rozwoju szeroko pojętej technologii informacyjnej zarówno w warstwie sprzętowej, jak i programowej, a z drugiej — inspiracją w tym zakresie były rozwijające się dynamicznie zastosowania IT w wielu dyscyplinach nauki, a także praktyki gospodarczej. Dlatego też analogicznie do ww. sformułowanych kryteriów, z jednej strony, można zaobserwować dynamiczny rozwój **teorii sztucznej inteligencji** (ang. *Artificial Intelligence* — AI), której celem jest wytwarzanie, w zakresie oprogramowania i sprzętu, urządzeń



(np. elektronicznych, biologicznych, bioelektronicznych, mechatronicznych) zdolnych do rozwiązywania zagadnień na poziomie wyższych czynności umysłowych (niekiedy wymaga to też naśladowania procesów biologicznych). Z drugiej strony, metody budowane na bazie TSE<sup>1</sup> w części aplikacyjnej (wytwarzanie urządzeń) należą do informatyki. W części podstawowej zajmują się one zagadnieniami wspólnymi z naukami, w których obszarze zainteresowań znajdują się czynności umysłowe i reakcje organizmu. Należą do nich niektóre dyscypliny nauk opisowych, np. biologia, psychologia, a także pewne nauki normatywne, np. logika, teoria gier, teoria decyzji itp. TSE należy zatem do teorii kreowanej na bazie części wspólnych ww. dyscyplin.

Lepsze rozeznanie w paradygmatach leżących u podstaw rozwoju metod AI daje też spojrzenie filozoficzne na te zagadnienia. Zostały w tym obszarze wyodrębnione dwa odległe cele:

- 1) komputer pozwala formułować i sprawdzać hipotezy dotyczące mózgu — wersja słaba AI;
- 2) komputer odpowiednio zaprogramowany jest w istotny sposób równoważny mózgowi, można mu przypisać stany poznawcze — wersja silna AI.

Niektóre z metod AI są z powodzeniem wykorzystywane w rozwiązywaniu problemów decyzyjnych. Rozwiązaniem jest w takim przypadku grupa alternatywnych decyzji wybrana w wyniku zastosowania określonej metody sztucznej inteligencji. Szczególnym przypadkiem w tym postępowaniu jest wykorzystanie aparatu logiki matematycznej jako swoistego abstraktu uniwersalnego rozumowania [Michalewicz 2003].

### 9.1.1. Klasyfikacja metod sztucznej inteligencji

Ogólna klasyfikacja metod AI może być prowadzona według następujących kryteriów:

- kryterium charakterystyki i metod przeprowadzania analiz;
- kryterium funkcji użytecznych, mechanizmów biologicznych i sposobu modelowania inteligencji człowieka;
- kryterium modeli przetwarzania.

Z punktu widzenia charakterystyki i metod przeprowadzania analiz można różnicę podział metod AI na dwie gałęzie:

- 1) gałąź analizy logicznej, symbolicznej i koncepcyjnego rozumienia inteligencji,
- 2) gałąź subsymboliczna oparta na próbach modelowania funkcji mózgu oraz symulacjach komputerowych zamiast analiz logicznych.

---

<sup>1</sup> TSE (ang. *Technology, Science and Engineering*) — technologie i systemy informatyczne.

Kryterium podziału metod AI — uwzględniające funkcje użytkowe, mechanizmy biologiczne oraz sposób wykorzystania inteligencji człowieka w systemach AI — odnosi się do praktycznych zastosowań w takich obszarach, jak:

- systemy ekspertowe (SE),
- robotyka,
- przetwarzanie mowy i języka naturalnego,
- heurystyka i strategia poszukiwań,
- kognitywistyka,
- badania inteligencji mrówek, pszczół i innych organizmów,
- modelowanie sztucznego życia,
- inteligentne programy, roboty, boty itp.

Kryterium modeli przetwarzania stosowanych w AI pozwala rozróżnić metody wykorzystujące:

- reprezentację wiedzy za pomocą zbiorów przybliżonych,
- reprezentację wiedzy za pomocą zbiorów rozmytych,
- sieci neuronowe,
- algorytmy ewolucyjne,
- metody grupowania danych,
- systemy neuronowo-rozmyte typu Mamdaniego, logicznego i Takagi–Sugeno,
- elastyczne systemy neuronowo-rozmyte.

## 9.1.2. Zastosowanie aparatu logiki matematycznej

Powszechnie uważa się, że w budowie baz wiedzy (BW) krytycznym czynnikiem jest pozyskiwanie wiedzy od eksperta. Ważnym aspektem rozwiązania tego problemu jest przyjęcie odpowiedniej reprezentacji wiedzy w SWD, umożliwiającej przekształcanie trudnych, złożonych zadań w zadania prostsze. W celu zapewnienia efektywnej pracy SWD niezbędne jest określenie techniki zapisu wiedzy, które powinno być poprzedzone ustaleniem przedmiotu zapisu; mogą to być np. obiekty indywidualne, własności obiektów, relacje między obiektami, klasy obiektów, rodzaje zdarzeń, konkretne zdarzenia, motywy i przyczyny zdarzeń, skutki zdarzeń, twierdzenia itp. Należy również uwzględnić problemy związane z zapisem informacji warunkowych, sprzecznych, wyjątków, informacji niepewnych, zmieniających się w czasie. Jest to istotne zwłaszcza w tych systemach, gdzie wiedza jest pozyskiwana od kilku niezależnych ekspertów lub grupy ekspertów, w wyniku czego pewne informacje mogą być wprost sprzeczne. Pomocne w rozwiązywaniu wspomnianych problemów reprezentacji wiedzy staje się zastosowanie aparatu logiki matematycznej, a w szczególności predykatów zorganizowanych w postaci języka rachunku predykatów, który stanowi podstawę programowania w logice.

**Predykat** jest najbliższym odpowiednikiem tego, co tradycyjna gramatyka języka naturalnego nazywa orzeczeniem. Korzystając z analogii, można stwierdzić, że

każdy predykat określa pewien zbiór przedmiotów — tych, o których da się go prawdziwie orzec. To samo odnosi się do orzeczenia (na co wskazuje użyty przed chwilą termin „orzec”), stąd wspomniana analogia. I tak, predykat „okrąża Słońce” określa zbiór ciał okrążających Słońce, czyli planet, ponieważ orzeka się go prawdziwie o planetach.

W dawniejszym ujęciu **rachunek predykatów** definiowano równoznacznie z rachunkiem funkcyjnym, ponieważ predykat jest rodzajem funktora. Nazwa ta nie jest jednak adekwatna do obecnego stanu logiki [Marciszewski 2002]. Istnieje też terminologia opisująca hierarchię teorii logicznych — teorii zdań przypisuje się rząd zerowy, rachunek predykatów w wersji rozważanej nazywa się rachunkiem predykatów pierwszego rzędu lub logiką pierwszego rzędu, następnie mamy do czynienia z logiką drugiego rzędu itd.

Rachunek predykatów dostarcza praw wnioskowania odwołujących się do wewnętrznej budowy zdań, w której wyróżnia się predykaty (odpowiednik orzeczenia w gramatyce), argumenty predykatów (odpowiednik podmiotu w gramatyce) oraz wyrażenia, zwane kwantyfikatorami (wskazujące, czy predykat odnosi się do wszystkich, czy do niektórych argumentów). W związku z tym rachunek predykatów często nazywany jest **logiką kwantyfikatorów** [Marciszewski 2002].

Predykat składa się z nazwy i dowolnej liczby argumentów, które są nazywane termami. Termami mogą być stałe (symbole alfanumeryczne i numeryczne) i zmienne oraz wyrażenia. W wyniku podstawienia stałych za zmienne otrzymujemy zdania prawdziwe lub fałszywe. Zaletą predykatów jest możliwość budowania zdań logicznych o prostych i zrozumiałych interpretacjach.

Rachunek predykatów jest językiem formalnym. Na język rachunku predykatów składają się:

- alfabet rachunku predykatów,
- zbiór termów ( $T_m$ ),
- formuły atomowe ( $A_{Form}$ ),
- formuły rachunku predykatów ( $Form$ ).

**Formuły rachunku predykatów** są budowane ze zmiennych i formuł z użyciem podstawowych operacji logicznych i kwantyfikatorów [Marciszewski 2002].

Każda formuła rachunku predykatów powstaje z formuł atomowych przez zastosowanie skończoną liczbę razy operatorów negacji, koniunkcji, alternatywy, implikacji i równoważności oraz kwantyfikatorów ogólnych i egzystencjalnych.

Ważną rolę w reprezentacji wiedzy odgrywają literały i klauzule. Przykładem klauzuli jest formuła rachunku predykatów:

$$\{x_1 \& x_2, \neg x_1 \& x_2, x_1 \& \neg x_2\}.$$

**Tautologią logiki predykatów** nazywamy każdą taką formułę języka rachunku predykatów, która jest prawdziwa przy dowolnych wartościach występujących w niej

stałych, zmiennych i predykatów. Przykładem klauzuli będącej tautologią jest poniższa formuła rachunku predykatów:

$$\{x_1 \& x_2, \neg x_1 \& x_2, x_1 \& \neg x_2, \neg x_1 \& \neg x_2\},$$

gdzie:

$x_i$  — zmienna logiczna o wartościach ze zbioru  $\{0,1\}$ .

### 9.1.3. Zastosowanie analogii naturalnych

**Analogie naturalne** występują w następujących przykładach konstrukcji teoretycznych wykorzystywanych w zastosowaniach sztucznej inteligencji:

- sieć neuronowa,
- algorytm genetyczny,
- algorytm mrówkowy.

Wszystkie metody sztucznej inteligencji, bazujące na analogiach naturalnych, wykorzystują mechanizmy zaobserwowane w organizmach żywych, w których w ciągu milionów lat ewolucji powstały bardzo efektywne systemy zapamiętywania, składowania oraz przetwarzania informacji, a które w praktyce zaczęły przynosić bardziej zadowalające wyniki niż metody matematyczne. Jest to szczególnie zauważalne w rozwiązywaniu bardzo złożonych i źle strukturalizowanych problemów, które z założenia miały być rozstrzygane przy wykorzystaniu metod sztucznej inteligencji [Goldberg 1995], [Rutkowska, Pliński, Rutkowski 1999].

Pozyskiwanie wiedzy na podstawie instrukcji wymaga wzajemnej współpracy między ekspertem a systemem. Polega ona na wskazaniu źródeł wiedzy, sposobu korzystania z niej i jej zastosowania. Istniejące gotowe, pełne zestawy instrukcji dotyczących badanego tematu mogą znacznie uprościć proces pozyskiwania wiedzy.

Metoda oparta na analogii polega na transformacji istniejącej wiedzy tak, aby mogła ona być użyteczna do opisu obiektów podobnych. Dokonuje się tego poprzez odszukanie wiedzy odpowiadającej opisowi obiektu analogicznego do opisanego w BW, a następnie jej odpowiednią transformację do postaci dobrze opisującej nowy obiekt. Uzyskana w ten sposób nowa wiedza jest dołączona do BW. Bezpośrednie zapisywanie wiedzy nie wymaga od systemu żadnego działania i sprowadza się do prostego zapamiętywania. Wiedza wprowadzana jest w sposób arbitralny, bez użycia mechanizmów wnioskowania.

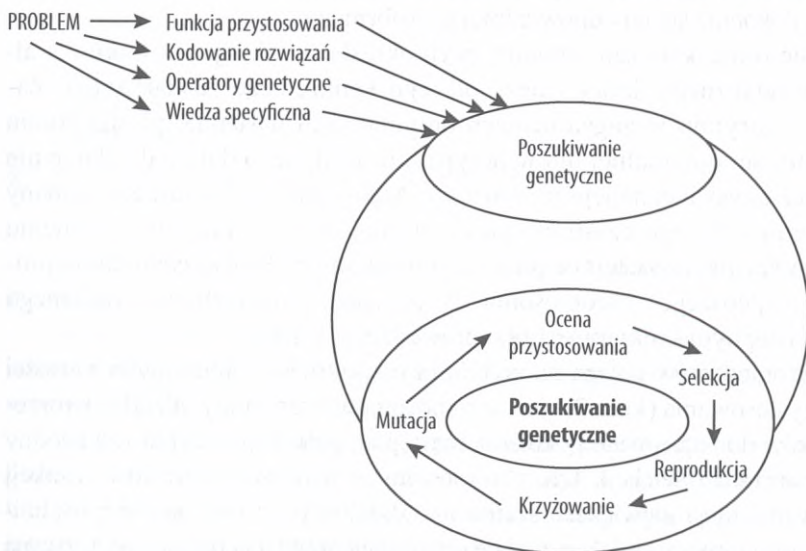
**Sztuczne sieci neuronowe** określa się jako interdyscyplinarną dziedzinę wiedzy zajmującą się konstrukcją, trenowaniem i badaniem możliwości tego rodzaju sieci. Istnieje obecnie przekonanie, że sztuczne sieci neuronowe nie stanowią dobrego modelu mózgu, choć różne ich postaci wykazują cechy charakterystyczne dla biologicznych układów neuronowych, takich jak: zdolność do uogólniania wiedzy i uaktualniania kosztem wcześniej poznanych wzorców oraz udzielanie mylnych odpowiedzi po przepełnieniu.

Sztuczne sieci neuronowe znajdują zastosowanie: w rozpoznawaniu i klasyfikacji wzorców (przydzielaniu wzorcom kategorii), w predykcji szeregów czasowych, w analizie danych statystycznych, w odszumianiu i kompresji obrazu oraz dźwięku, a także w innych zagadnieniach. Ponadto wykorzystywane są w procesach podejmowania decyzji, szczególnie w systemach sterowania i automatyzacji procesów obliczeniowych, technologicznych i wytwórczych.

U podstaw koncepcji konstrukcji i wykorzystania algorytmów genetycznych leży wzorowanie się na mechanizmach naturalnych procesów ewolucji. Okazuje się, że rozwiązania uzyskane metodami ewolucyjnymi są z reguły lepsze od algorytmów proponowanych przez człowieka. Obliczenia ewolucyjne są dziś ważną częścią AI i występują w wielu odmianach [Kwaśnicka 1999; Korczak, Lipinski 2001].

**Metody ewolucyjne** powstały w celu przybliżonego rozwiązania problemów optymalizacyjnych w taki sposób, aby można było generować rozwiązanie, unikając pułapek minimów lokalnych. Obliczenia ewolucyjne są kombinacją: współczesnych technik obliczeniowych, znaczącego postępu wiedzy w zakresie ewolucji biologicznej oraz możliwości, jakie w zastosowaniach uzyskuje się, na przykład, dzięki biotechnologiom i nanotechnologiom. Metodę postępowania w trakcie rozwiązywania problemu optymalizacyjnego za pomocą algorytmów ewolucyjnych [Duzinkiewicz 2009] przedstawia rysunek 9.1.

**Rysunek 9.1. Struktura procesu poszukiwania rozwiązania metodą algorytmu ewolucyjnego**



Źródło: [Duzinkiewicz 2009].

Najpopularniejsze obecnie **algorytmy genetyczne** (AG) odzwierciedlają kolejność czynności w trakcie symulowanej ewolucji. Sposób działania algorytmu genetycznego można przedstawić następująco:

- określenie sposobu kodowania rzeczywistych parametrów problemu w postaci chromosomu;
- przyjęcie postaci funkcji przystosowania oceniającej analizowany zestaw parametrów pod względem jakości poszukiwanego rozwiązania;
- losowy dobór punktu startowego zestawu parametrów;
- selekcja najlepiej przystosowanych chromosomów do nowej populacji;
- zastosowanie na nowej populacji operatorów genetycznych w postaci krzyżowania i mutacji;
- sprawdzenie wartości funkcji przystosowania.

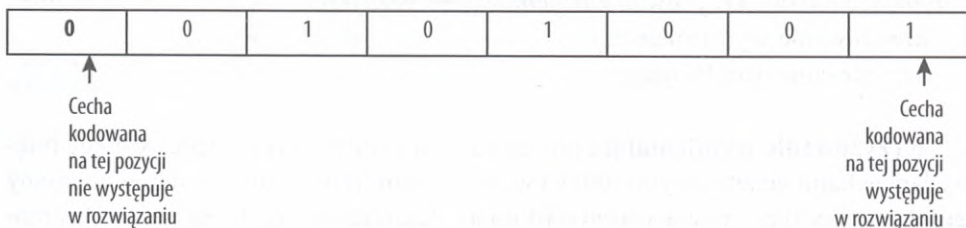
W podstawowym algorytmie genetycznym występują następujące operacje:

1. Inicjalizacja, czyli utworzenie populacji początkowej; polega to na losowym wyborze żądanej liczby chromosomów (osobników) reprezentowanych przez ciągi binarne określonej długości.
2. Ocena przystosowania chromosomów w populacji, polegająca na obliczeniu wartości funkcji przystosowania dla każdego chromosomu z tej populacji. Wraz ze wzrostem wartości tej funkcji jakość chromosomów ulega polepszeniu. Postać funkcji przystosowania zależy od rodzaju rozwiązywanego problemu. W zależności funkcja przystosowania przyjmuje zawsze wartości nieujemne, a rozwiązywany problem optymalizacji jest problemem poszukiwania maksimum tej funkcji. Pierwotna postać funkcji musi spełniać sformułowane założenia, w przeciwnym przypadku dokonuje się jej odpowiedniej transformacji.
3. Sprawdzenie warunków zatrzymania, czyli określenie takiego warunku dla algorytmu genetycznego, który zależy od jego konkretnego zastosowania. Zatrzymanie algorytmu w zagadnieniach optymalizacji następuje po uzyskaniu żądanej wartości optymalnej lub w przypadku, kiedy jego dalsze działanie nie poprawia już uzyskanej najlepszej wartości. Algorytm może zostać zatrzymany po upływie określonego czasu lub po określonej liczbie iteracji. Po spełnieniu warunku zatrzymania następuje przejście do ostatniego kroku, czyli do wyprowadzenia „najlepszego” chromosomu. W przypadku niespełnienia opisanego warunku następnym krokiem jest przeprowadzenie selekcji.
4. Selekcja chromosomów polega na wybraniu na podstawie obliczonych wartości funkcji przystosowania (krok 2) tych chromosomów, które brały udział w tworzeniu potomków do następnego pokolenia (następnej generacji). Wybór jest zgodny z zasadą naturalnej selekcji, tzn. chromosomy o największej wartości funkcji przystosowania mają największe szanse na udział w tworzeniu nowych osobników. W wyniku procesu selekcji zostaje utworzona populacja rodzicielska, zwana pulą rodzicielską o liczebności równej liczebności bieżącej populacji.
5. Zastosowanie algorytmów genetycznych do chromosomów wybranych metodą selekcji pozwala utworzyć nową populację. Stanowi ona populację potomków otrzymanych z wybranej populacji rodziców (metodą selekcji). W AG wyróżnia się dwa podstawowe operatory genetyczne: operator krzyżowania i operator mutacji.

6. Utworzenie nowej populacji polega na tym, iż chromosomy otrzymane w wyniku działania operatorów genetycznych na chromosomy w tymczasowej populacji rodzicielskiej wchodzi w skład nowej populacji, która staje się populacją bieżącą dla danej iteracji algorytmu genetycznego. W każdej kolejnej iteracji oblicza się wartość funkcji przystosowania każdego z chromosomów tej populacji. Następnie sprawdzany jest warunek zatrzymania algorytmu, w efekcie czego generuje się wynik w postaci chromosomu o największej wartości funkcji przystosowania, w przeciwnym razie przechodzi się do selekcji.
7. Wyprowadzenie „najlepszego” chromosomu następuje wtedy, kiedy jest spełniony warunek zatrzymania algorytmu genetycznego; wówczas generowany jest wynik. Najlepszym rozwiązaniem jest chromosom o największej wartości funkcji przystosowania.

Punktem wyjścia do zbudowania algorytmu genetycznego jest przedstawienie rozważanego zadania w kategoriach chromosomów (łańcuchów genetycznych) zapisywanych najczęściej w postaci wektorów binarnych (rysunek 9.2) [Back 1993].

**Rysunek 9.2. Budowa chromosomu jako wektora binarnego w AG**



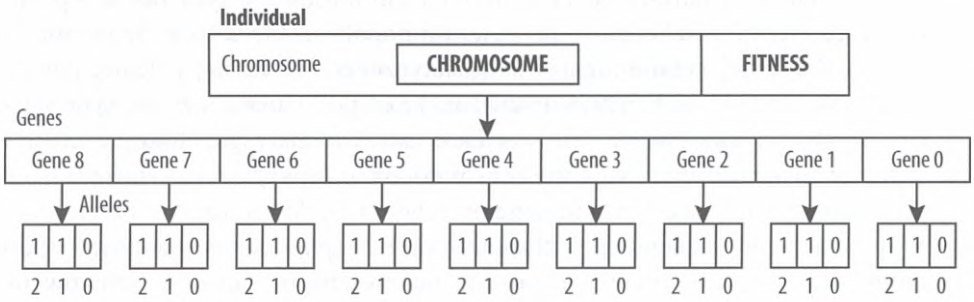
Źródło: Opracowanie własne na podstawie: [Back 1993], [Arabas 2001].

Ważne jest, aby chromosom dobrze opisywał „osobnika”, tak jak wskazano na rysunku 9.3.

Jeśli danego zadania nie udaje się efektywnie przedstawić w postaci chromosomu i funkcji przystosowania, to w celu rozwiązania go można podjąć próbę zastosowania innych metod ewolucyjnych.

Ogólny schemat działania algorytmu genetycznego opiera się na założeniu, że populacja rozwiązań rozwija się poprzez mutacje, krzyżowania oraz pomijanie słabiej przystosowanych osobników. W algorytmach genetycznych rozróżnia się dwie interpretacje populacji, reprezentowane przez ujęcie Michigan lub Pittsburg. W ujęciu Michigan wszystkie osobniki są traktowane jednostkowo (oceniane są poszczególne osobniki). Osobniki w populacji rywalizują ze sobą, chcąc przetrwać. Natomiast w ujęciu Pittsburg całą populację traktuje się jako jednostkę, która podlega działaniu operatorów genetycznych (oceniana jest cała populacja). W tym przypadku można zaobserwować współpracę osobników w celu wykształcenia jak najlepszej

Rysunek 9.3. Struktura chromosomu jako funkcja przekazywania wartości cech



*Chromosome* — chromosom — ciąg kodowy; *Individual* — indywidualny; *Gene* — gen — cecha, znak, detektor; *Alleles* — allele — wartość danego genu, nazywana też wartością cechy lub wariantem cechy na określonej pozycji; *Fitness* — dopasowanie „osobnika” do środowiska.

Źródło: Jak rysunku 9.2.

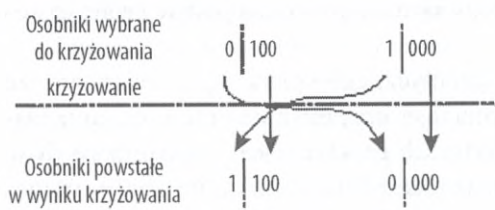
społeczności. Oba ujęcia mają swoje uzasadnienie i można wskazać problemy, w odniesieniu do których warto zastosować albo jedno, albo drugie ujęcie.

W funkcjonowaniu i zastosowaniach algorytmów genetycznych rozróżnia się następujące rodzaje krzyżowań łańcuchów genetycznych:

- krzyżowanie wymieniające,
- krzyżowanie uśredniające.

**Krzyżowanie wymieniające** polega na wymianie materiału genetycznego między łańcuchami genetycznymi rodziców, przy czym liczba punktów takiej wymiany genów może przyjmować wartości od 1 do  $n - 1$ , gdzie:  $n$  to liczba genów w chromosomie. Przykład krzyżowania jednopunktowego został przedstawiony na rysunku 9.4 [Arabas 2001].

Rysunek 9.4. Schemat jednopunktowego krzyżowania wymieniającego



Źródło: Jak rysunku 9.2.

**Krzyżowanie uśredniające** jest specyficzne dla kodowania za pomocą liczb rzeczywistych i powoduje modyfikację wartości genów w chromosomach. Wartości



genów w chromosomach potomnych są zwarte między największą i najmniejszą wartością chromosomów rodzicielskich. Krzyżowanie to jest wykonywane w dwóch etapach. Pierwszy etap polega na określeniu miejsca krzyżowania się chromosomu i wymiany informacji. Rysunek 9.4 przedstawia przykład, w którym miejsce przecięcia jest określone po pierwszym genie. Następnie dokonywana jest wymiana informacji genetycznej, w wyniku której powstają dwa nowe osobniki (dołączone do nowej populacji będą kontynuować proces ewolucji). Zgodnie z rysunkiem 9.4 dwa osobniki o wartości funkcji przystosowania 4 i 8 tworzą dwa nowe osobniki o wartości funkcji przystosowania 11 i 0. Na rysunku 9.5 przedstawiono, w jaki sposób stosunkowo słabe osobniki (wartości funkcji przystosowania równe odpowiednio: 2 i 0) dają materiał genetyczny do utworzenia osobnika o bardzo dużej wartości funkcji przystosowania.

**Rysunek 9.5. Schemat jednopunktowego krzyżowania wymieniającego**



Źródło: Jak rysunku 9.2.

Krzyżowanie wymieniające daje z równym prawdopodobieństwem możliwość utworzenia  $2(g - 1)$  genów, gdzie  $g$  oznacza ich liczbę w chromosomie.

Krzyżowanie uśredniające, w przeciwieństwie do krzyżowania wymieniającego, wpływa na wartości poszczególnych genów. Krzyżowanie to jest specyficzne dla kodowania całkowitoliczbowego i zmiennoprzecinkowego. Cechą wspólną różnych wariantów krzyżowania uśredniającego jest to, że wartość każdego genu chromosomów potomków jest liczbą zawierającą się między największą a najmniejszą wartością genu chromosomów rodziców.

W jednym z wariantów krzyżowania uśredniającego nowy osobnik tworzony jest według następującego schematu: losowana jest pewna liczba z przedziału  $\langle 0, 1 \rangle$ , każdy z genów modyfikowany jest zgodnie ze schematem przedstawionym na rysunku 9.6. Wartość każdego genu zmieniona jest w równych proporcjach w stosunku do różnicy wartości genów rodziców.

Inny wariant krzyżowania uśredniającego przewiduje (rysunek 9.7), że proporcje zmiany genu są ustalane indywidualnie na podstawie losowanej liczby z przedziału  $\langle 0, 1 \rangle$  osobno dla każdego genu.

**Rysunek 9.6. Schemat krzyżowania uśredniającego — wariant I**



Źródło: Jak rysunku 9.2.

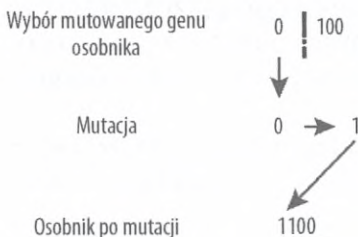
**Rysunek 9.7. Schemat krzyżowania uśredniającego — wariant II**



Źródło: Jak rysunku 9.2.

**Mutacja kodu binarnego** polega na tym, że dla każdego genu osobnika podejmuje się decyzje o jego modyfikacji z określonym prawdopodobieństwem. Modyfikacja polega na zanegowaniu danego genu zgodnie ze schematem przedstawionym na rysunku 9.8.

**Rysunek 9.8. Schemat mutacji kodu binarnego**



Źródło: Jak rysunku 9.2.

Jeśli geny przyjmują wartości ze zbioru liczb rzeczywistych, to mutacja polega na perturbacji wartości zmiennych niezależnych znajdujących się w chromosomie (rysunek 9.8). Perturbacja ta jest wykonywana przez dodanie wektora, będącego realizacją  $n$ -wymiarowej zmiennej losowej  $\mathcal{L}_r$  o założonym rozkładzie:

$$Y = X + \mathcal{L}_r. \quad [9.1]$$

Jako rozkład  $r$  najczęściej jest wykorzystywany rozkład normalny lub coraz bardziej popularny rozkład Cauchy'ego.

W najbardziej rozpowszechnionych schematach mutacji korzysta się z rozkładów nieskorelowanych, co pozwala zastąpić wzór ogólny wzorem obowiązującym dla każdego genu:

$$Y_i = X_i + \mathcal{L}_{r,i} \quad [9.2]$$

gdzie:

- $i$  — numer genu,
- $\mathcal{L}_{r,i}$  — realizacja zmiennej losowej o rozkładzie  $r$ , definiowanej indywidualnie dla każdego genu.

Sukcesja z całkowitym zastępowaniem (trywialna) jest podstawowym i najczęściej wykorzystywanym schematem. Polega ona na tym, że nową populacją bazową staje się populacja potomna, czyli  $P_i + I = O_i$ . Sukcesja trywialna nie wprowadza nacisku selektywnego.

**Algorytm mrówkowy** został opracowany przez M. Dorigo [Dorigo, Stützle 2004], [Michalewicz 2003] jako technika stosowana głównie w przypadku wyszukiwania najlepszych ścieżek w grafie. Inspiracja pochodziła z systemu organizacji społeczności mrówek, które potrafią znaleźć najkrótszą drogę między mrowiskiem a pożywieniem. Na początku, wędrując w stronę pożywienia, mrówki wybierają trasę losowo, ale wracając do mrowiska, pozostawiają na swojej drodze ślad feromonowy.

Oto podstawowe zasady działania algorytmu mrówkowego:

- feromony stopniowo parują;
- na krótszej trasie parowanie jest stosunkowo wolniejsze niż na innych;
- po odnalezieniu przez mrówki najkrótszej ścieżki, następne wybierają ją chętniej, wzmacniając ślad feromonowy; jest to zjawisko dodatniego sprzężenia zwrotnego — atrakcyjna ścieżka staje się coraz bardziej atrakcyjna, a mniej atrakcyjna trasa traci na znaczeniu.

Model algorytmu mrówkowego oparty jest na pośredniej interakcji między mrówkami oraz pewnej formie gromadzenia doświadczeń i ich wykorzystywania w dalszych poszukiwaniach. Inteligencja zbiorowa mrówek objawia się we wspólnym wypracowywaniu zbioru najkrótszych ścieżek wiodących je do wyznaczonych celów. Rozwiązania te doskonalone są w miarę upływu czasu.

Działanie poszczególnych agentów w **systemie wieloagentowym** odpowiada działaniu pojedynczych mrówek poszukujących odpowiednich dla nich zasobów. Każdy z agentów (mrówka) realizuje identyczną strategię poszukiwania najkrótszej drogi do celu.

Algorytm mrówkowy został z powodzeniem użyty do rozwiązania problemu komiwojażera. Istota tego postępowania jest następująca:

1. Rozmiar utworzonej populacji mrówek jest jednym z parametrów algorytmu.
2. Pojedyncza mrówka generuje swoją ścieżkę niezależnie od swoich towarzyszek.
3. Dla każdej mrówki generowane jest losowo miasto, z którego ma rozpocząć wędrówkę.
4. Mrówka porusza się po grafie, szukając sekwencji wierzchołków grafu tworzącej najkrótszą drogę od wierzchołka startowego do końcowego. Mrówka jest wyposażona w pamięć, w której przechowuje listę takich wierzchołków, aby wykluczyć powrót do już odwiedzonych. Na starcie lista jest pusta, a po dojściu do celu zawiera wierzchołki w kolejności ich odwiedzenia.
5. Mrówka rusza z miasta początkowego i, dochodząc do każdego kolejnego miasta, pozostawia na krawędzi grafu feromon. Przy wyborze dalszej drogi mrówka nie bierze pod uwagę wierzchołków z listy miejsc już odwiedzonych; kieruje się natomiast feromonem pozostawionym przez te mrówki, które szły przed nią.
6. Wybór kolejnego odcinka drogi jest losowy, ale zgodny z zasadą: im więcej feromonu na danej krawędzi grafu, tym większe prawdopodobieństwo wyboru tej drogi przez mrówkę.

Zaprezentowany algorytm ma iteracyjny charakter, ponieważ po zakończeniu bieżącej iteracji każda mrówka czeka na wyznaczenie nowego miasta początkowego, skąd w następnej iteracji ponownie wyruszy, aby przemierzać swoją trasę. Gromadzenie się feromonu na krawędziach grafu w miarę upływu czasu w kolejnych iteracjach jest kluczowym efektem działania algorytmu. W miarę upływu czasu pozostawiony feromon paruje, przez co zmniejsza się jego ilość na ścieżkach.

Kluczowy w reprezentacji danych jest fakt, że żadna informacja, która mogłaby być trwale dostępna w kolejnych iteracjach, nie jest gromadzona w mrówkach. Rola mrówek sprowadza się do generowania trasy na podstawie ilości feromonu złożonego na poszczególnych krawędziach grafu, a następnie do złożenia swojej porcji feromonu na kolejno przebytych krawędziach.

Najważniejszym elementem algorytmu, przechowującym informacje o rozwiązaniu, jest więc struktura zawierająca wartości poziomu feromonu na poszczególnych krawędziach.

## 9.2. Sieci neuronowe

**Sztuczne sieci neuronowe** wykorzystuje się do rozwiązywania złożonych problemów decyzyjnych. Sieci neuronowe doskonale nadają się do zastosowania w anali-

zie oceny ryzyka kredytowego czy ubezpieczeniowego, segmentacji rynku, wyboru najlepszej strategii inwestycyjnej, znalezienia optymalnej drogi w procesach logistycznych, analizy bezpieczeństwa transakcji z użyciem kart kredytowych czy wykrywania nadużyć ubezpieczeniowych. Zastosowanie sztucznych sieci neuronowych pozwala na skuteczne zarządzanie ryzykiem, kontrolę jakości czy ocenę konkurencyjności rynku w stopniu, jaki nie byłby osiągalny innymi znanymi metodami.

### 9.2.1. Idea i struktura sieci neuronowych

**Sieci neuronowe** są systemem wzajemnie połączonych ze sobą neuronów [Tadeusiewicz 1993], [Roszkowski 2004, s. 1]. Inspiracją do stworzenia sieci neuronowej była budowa mózgu ludzkiego, a istota działania sieci neuronowej opiera się na zasadzie funkcjonowania systemu nerwowego człowieka. Oczywiście jest, że matematyczny model neuronu stosuje wiele uproszczeń, które umożliwiają wygodną implementację i eksploatację sieci neuronowej [Roszkowski 2004, s. 2]. Taki matematyczny model nosi nazwę neuronu formalnego i stanowi podstawę budowy sztucznej sieci neuronowej [Kosiński 2007, s. 33].

Sztuczna sieć neuronową najogólniej można zdefiniować jako równoległą strukturę przetwarzania danych o topologii grafu skierowanego, gdzie wierzchołki grafu nazywane są jednostkami obliczeniowymi lub sztucznymi neuronami, a krawędzie grafu są ich połączeniami [Hecht-Nielsen 1990]. Każdy neuron może mieć dowolną liczbę wejść i wyjść, a każde wyjście neuronu przesyła ten sam sygnał w kierunku zgodnym z kierunkiem połączeń [Bezulski 2009, s. 11]. Połączeniom neuronu przyporządkowane są wartości, zwane wagami. Każda jednostka (neuron) posiada tzw. funkcję aktywacji, która decyduje o wartości sygnału wyjściowego. Pojedynczy neuron realizuje funkcję aktywacji przedstawioną w postaci wzoru [9.3] [Bezulski 2009, s.12]:

$$y = f(x_1, \dots, x_n, w_1, \dots, w_n), \tag{9.3}$$

gdzie:

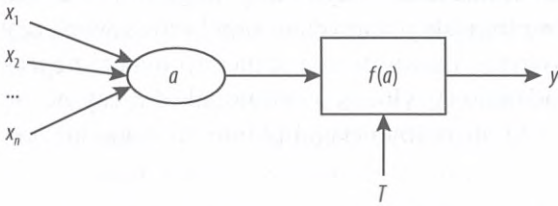
- $x_i$  — wartości sygnałów wejściowych neuronu,
- $w_i$  — wartości wag połączeń (wejść) neuronów.

Sygnały wejściowe neuronu są w modelu sieci agregowane. Zagregowana wartość sygnału wejściowego, zwana pobudzeniem neuronu [Bezulski 2009, s. 12], została przedstawiona za pomocą wzoru [9.4]:

$$a = w_0 + \sum_{i=1}^n x_i w_i, \tag{9.4}$$

Jeden z pierwszych modeli neuronów, sformułowany w 1943 r. przez W.S. McCullocha i W.H. Pittsa [McCulloch, Pitts 1943], jest oparty właśnie na metodzie agregacji sygnałów wejściowych sieci (rysunek 9.9).

Rysunek 9.9. Model sztucznego neuronu McCullocha–Pittsa



Źródło: Opracowanie własne na podstawie: [Roszkowski 2004].

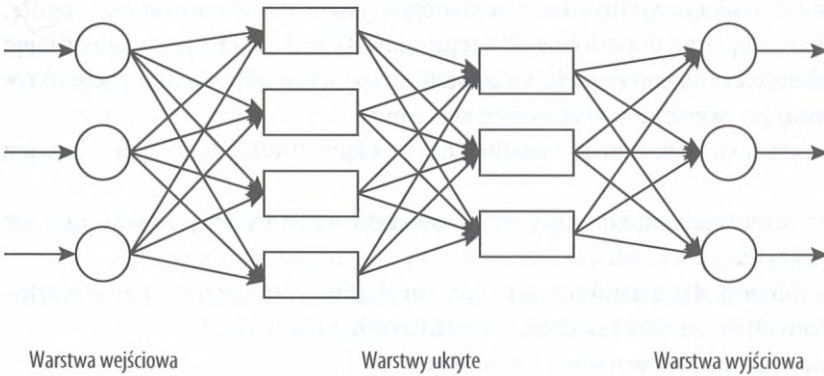
Do każdego neuronu (rysunek 9.9) docierają wartości wejściowe o określonej ważności (wadze). Na podstawie sumy iloczynów wartości sygnałów wejściowych ( $x_i$ ) i wagi każdego sygnału ( $w_i$ ) obliczany jest sygnał łącznego pobudzenia ( $a$ ). Sygnał wyjściowy neuronu powstaje w wyniku podstawienia sygnału łącznego pobudzenia neuronu ( $a$ ) jako argumentu funkcji aktywacji  $f(a)$  przy określonym progu aktywacji  $T$ . Funkcja aktywacji neuronu warunkuje sposób jego działania. Do najbardziej popularnych **funkcji aktywacji**, wykorzystywanych w sztucznych sieciach neuronowych, należą [Bezulski 2009, s. 12–13]:

- funkcja tożsamościowa:  $f(a) = a$ ,
- funkcja Heaviside'a:  $f(a) = \begin{cases} 1, & \text{dla } a \geq 0, \\ 0, & \text{dla } a < 0, \end{cases}$
- funkcja sygnum:  $f(a) = \begin{cases} 1, & \text{dla } a > 0, \\ 0, & \text{dla } a = 0, \\ 0, & \text{dla } a < 0, \end{cases}$
- funkcja logistyczna:  $f(a) = \frac{1}{1 + e^{-\beta_a}}$ ,
- tangens hiperboliczny:  $f(a) = \operatorname{tg} h(\beta_a) = \frac{e^{\beta_a} - e^{-\beta_a}}{e^{\beta_a} + e^{-\beta_a}}$ ,
- funkcja Gaussa:  $f(a) = e^{-\frac{a^2}{2}}$ .

Ponieważ możliwości pojedynczego neuronu w zakresie przetwarzania informacji są niewielkie, dlatego stosowane są grupy sztucznych neuronów połączone ze sobą w tzw. sztucznej sieci neuronowej, naśladującej strukturę systemu nerwowego organizmu żywego i pozwalającej na przeprowadzanie znacznie bardziej złożonych obliczeń.

W sieci pojedyncze neurony ułożone są w **warstwach**. Wartości wyjściowe wyznaczone dla neuronów jednej warstwy wprowadzane są na wejścia neuronów warstwy następnej. Wyjątkiem jest warstwa pierwsza, zwana warstwą wejściową, składająca się z neuronów, do których doprowadzane są wartości zmiennych zewnętrznych (pochodzących spoza sieci), oraz warstwa ostatnia (tzw. wyjściowa), składająca się z neuronów wyznaczających końcowy wynik obliczeń, traktowanych następnie jako wartość wynikowa działania sieci. Warstwy znajdujące się między warstwą wejściową i wyjściową, które wyznaczają wartości danych pośrednich, będących podstawą do wyznaczenia ostatecznego rozwiązania, nazywane są warstwami ukrytymi ([Kasperski 2003, s.173], rysunek 9.10).

Rysunek 9.10. Struktura sztucznej sieci neuronowej



Źródło: [Kasperski 2003, s. 173].

Specyficzny dla danej sieci sposób pracy jest uzależniony od wielu czynników, do których należy zaliczyć przede wszystkim: przyjęte modele neuronów, typ sieci neuronowej, wartości współczynników wagowych neuronów, liczbę warstw sieci, liczbę neuronów w poszczególnych warstwach sieci, przyjęty typ struktury sieci neuronowej. Ze względu na wymienione elementy wśród rodzajów sztucznych sieci neuronowych rozróżnia się:

- sieci jednokierunkowe jedno- i wielowarstwowe — nie występuje w nich sprzężenie zwrotne, czyli pojedynczy wzorzec lub sygnał przechodzi przez każdy neuron dokładnie raz;
- sieci rekurencyjne jedno- i wielowarstwowe — połączenia między neuronami stanowią graf z cyklami;
- samoorganizujące się mapy SOM (sieci Kohonena) — przetwarzają wielowymiarowe dane wejściowe i przedstawiają je na dwuwymiarowej płaszczyźnie, czyli mapie;
- maszyny wektorów wspierających SVM — oparte na radialnej funkcji bazowej, której wartość zależy od odległości zmiennej od określonego punktu zwanego centrum (lub punktem aproksymacyjnym);

- sieci typu propagacji wstecznej — proces uczenia jest oparty na algorytmie propagacji wstecznej, polegającym na obliczaniu błędu sieci najpierw dla warstwy wyjściowej, następnie dla kolejnych warstw poprzedzających; celem jest uzyskanie takiej struktury wartości wag sieci, aby błąd końcowy był mniejszy niż ten, który uzyskano podczas pierwszego szacowania.

## 9.2.2. Strojenie (uczenie) sieci neuronowej

**Strojenie (uczenie) sieci neuronowej** polega na wyznaczeniu wartości wag w oparciu o zestaw posiadanych danych wejściowych oraz wyjściowych w taki sposób, aby zminimalizować przyjęty wskaźnik jakości działania sieci [Frydrychowicz 2008, s. 64]. Na podstawie zbioru przypadków, obejmującego cechy wprowadzanych danych korzystnych i niekorzystnych dla badanego zjawiska, system buduje regułę, która jest najbardziej prawdopodobna dla tego zjawiska. Reguła ta, uwzględniając parametry nieliniowej funkcji o wielu wejściach, w sposób możliwie najlepszy określa zależność między wejściem a wyjściem systemu.

Proces uczenia sieci przebiega według następujących etapów [Frydrychowicz 2008, s. 64]:

- przesłanie informacji od warstwy wejściowej do warstwy wyjściowej poprzez warstwy ukryte;
- obliczenie błędów dla neuronów warstwy wyjściowej przez porównanie wartości obliczonych przez sieć i wartości zakładanych na wyjściach;
- modyfikacja neuronów warstwy wyjściowej;
- przesłanie informacji o błędzie do neuronów warstwy poprzedniej (ukrytej); obliczona dla neuronów wyjściowych informacja o błędzie przesyłana jest przez te same połączenia, co informacja służąca do obliczenia wartości wyjściowych, odwrotny jest tylko kierunek jej przesyłania; informacja o błędzie jest pomnożona przez współczynniki wagowe;
- uczenie neuronów warstw ukrytych.

Uczenie ma charakter wieloetapowy i ma w efekcie doprowadzić do minimalizacji błędu sieci, będącego zagregowaną miarą różnic między rzeczywistymi wartościami wyjściowymi a wartościami obliczonymi za pomocą sieci. Najczęściej stosowaną formułą służącą do obliczania błędu jest suma kwadratów różnic między wspomnianymi wartościami, czyli [Frydrychowicz 2008, s. 64]:

$$E = \sum_{i=1}^N (d_i - y_i)^2, \quad [9.5]$$

gdzie:

- $d_i$  — rzeczywista wartość wyjściowa,
- $y_i$  — odpowiadająca jej wartość wyznaczona przez sieć,
- $N$  — liczebność zbioru uczącego.



**Sygnal błędu** jest wykorzystywany przez neuron do korygowania swoich współczynników wagowych. Osiągnięcie celu uczenia jest możliwe poprzez modyfikację wartości parametrów (wag) sieci. Sygnały wyjściowe, w odpowiedzi na sygnały wejściowe, określają rozwiązanie zadań stawianych sieci. Neuron sieci ma zdolność adaptacji, a jego wagi podlegają modyfikacji podczas uczenia. Ogólna zasada nauki przyjęta dla sieci neuronowych mówi, że wektor wag:  $\mathbf{W}_i = [w_{i1}, w_{i2}, \dots, w_{in_i}]$  rośnie proporcjonalnie do iloczynu sygnałów wejściowego  $x$  i uczącego  $r$ . Sygnał uczący  $r$  jest w ogólnej postaci funkcją  $w_{i,x}$  i sygnału nauczyciela  $d_i$ . Wśród reguł uczenia wyróżnia się [Wabnic 2007]:

- regułę Hebba — odnosi się do uczenia bez nauczyciela i przyjmuje, że sygnałem uczącym jest po prostu sygnał wyjściowy neuronu;
- regułę perceptronową — dotyczy nauki z nauczycielem i przyjmuje, że sygnał uczący jest różnicą między odpowiedzią pożądaną a rzeczywistą; odnosi się ona do sieci z neuronami dyskretnymi;
- regułę delta — obowiązuje dla neuronów z ciągłymi funkcjami aktywacji i nadzorowania trybu uczenia; stanowi odpowiednik reguły perceptronowej dla neuronów o ciągłej funkcji aktywacji; bywa zwana również ciągłą regułą perceptronową;
- regułę korelacyjną — w regule korelacyjnej poprawka każdej składowej wektora wag jest proporcjonalna do iloczynu odpowiedniej składowej obrazu wejściowego i pożądanego przy tym wzorca wyjścia; taka zasada uczenia czyni regułę korelacyjną szczególnie przydatną do zapamiętywania informacji w sieciach z neuronami bipolarnymi; można też uważać tę regułę za odpowiednik reguły Hebba dla nauki z nauczycielem;
- regułę „wygrywający bierze wszystko” — jest przykładem nauki z rywalizacją, stosowanej zazwyczaj do poznawania własności statystycznych sygnałów wejściowych w trybie nauki bez nauczyciela; działa na zasadzie, że poprawieniu podlega tylko waga neuronu o maksymalnej wartości funkcji aktywacji.

Realizowane kroki wyznaczające kolejne przybliżenia optymalnych wartości parametrów nazywane są epokami uczenia [Frydrychowicz 2008, s. 65]. Epoka obejmuje jednorazową prezentację wszystkich przypadków uczących i przeprowadzoną na tej podstawie modyfikację parametrów sieci (wag). Z punktu widzenia systemu uczącego sieci neuronowej można wyróżnić trzy podstawowe **typy uczenia się**:

- 1) pod nadzorem (z nauczycielem) — system jest wykorzystywany, gdy znane są zarówno dane wejściowe, jak i pożądanego wartości wyjściowe;
- 2) z krytykiem — system jest wykorzystywany, kiedy brak jest wartości poświadczonych na wyjściu systemu, a znana jest informacja o tym, czy podjęta przez system akcja (zmiana wartości wag) daje wyniki pozytywne czy negatywne w sensie oczekiwanego zachowania systemu;
- 3) samoorganizujące się sieci (bez nadzoru) — system jest wykorzystywany, kiedy pożądana odpowiedź sieci nie jest znana i sieć musi się uczyć poprzez analizę

reakcji na pobudzenia, których natury nie zna; w trakcie analizy parametry sieci podlegają zmianom, co nazywane jest samoorganizacją sieci.

### 9.2.3. Model decyzyjny sieci neuronowej

Sieci neuronowe są wykorzystywane najczęściej jako mechanizm sterujący procesem lub jako mechanizm decyzyjny. Współczesne zastosowania, jakie są możliwe do zaimplementowania za pomocą sieci neuronowych, to m.in. [Roszkowski 2004, s. 4–5]: predykcja, analiza danych, filtracja sygnałów, optymalizacja i klasyfikacja.

W dziedzinie predykcji zadanie sieci polega na tym, aby na podstawie pewnych danych wejściowych wyznaczyć określone dane wyjściowe. Przykładami takich problemów mogą być: prognozowanie zapotrzebowania kadrowego, prognozowanie obciążeń maszyn i urządzeń, ocena zdolności kredytowej, prognozowanie wskaźników ekonomicznych, przewidywanie zbiorów, prognozowanie upadłości przedsiębiorstw. Mimo nieznamości mechanizmów charakterystycznych dla danego problemu badawczego sieć neuronowa może nabyć zdolności przewidywania sygnałów wyjściowych na podstawie jedynie obserwacji ciągu uczącego. Ta właściwość jest wykorzystywana przede wszystkim w odniesieniu do problemów, których nie sposób opisać w postaci reguł lub algorytmów.

Sieci neuronowe potrafią także przeszukiwać dane pod względem związków przyczynowych czy też incydentalnych. Taka analiza danych umożliwia: ustalenie przyczyn niepowodzeń określonych przedsięwzięć (wyciągnięcie wniosków), analizę kondycji finansowej przedsiębiorstw lub analizę rynku kapitałowego.

Filtracja sygnału jest najczęściej stosowana w telekomunikacji oraz automatycznej diagnostyce medycznej. Umożliwia ona wyeliminowanie w danych wejściowych błędów celowych i systematycznych oraz przekłamań, a także uzupełnienie danych niekompletnych.

Klasyfikacja jest jedną z najbardziej typowych form przetwarzania neuronowego. Polega na przewidywaniu identyfikatora klasy, do której można zaliczyć dane wyjściowe. Dzięki temu sieć może znajdować istotne dane w takich zastosowaniach, jak: rozpoznawanie obrazu, pisma lub twarzy, identyfikacja obszarów poddanych wpływowi czynników negatywnych, rozpoznawanie i klasyfikacja wzorców oraz wzajemnych zależności między badanymi obiektami.

Zidentyfikowany problem decyzyjny jest rozwiązywany przez sieć neuronową na podstawie danych treningowych, które sieć przetwarza w fazie uczenia. Otrzymany wynik jest sprawdzany na danych walidacyjnych lub/i danych testowych. Bardzo istotnym elementem jest obecność zbioru walidacyjnego, którego zadaniem jest zapobieganie tzw. przetrenowaniu sieci, polegającego na doskonałych wynikach sieci w odniesieniu do zbioru treningowego oraz niezadowolających wynikach w odniesieniu do wszystkich innych zbiorów danych. Model, który został sprawdzony na danych walidacyjnych, może zostać przyjęty do użytkowania, a jego wyniki uznane za wiarygodne i bezpieczne do wykorzystania. Dodatkowym potwierdze-

niem jakości modelu może być zbiór danych testowych, które po wprowadzeniu do modelu powinny potwierdzić jakość wyników uzyskanych na podstawie zbioru treningowego i walidacyjnego.

## 9.2.4. Przykłady zastosowań sieci neuronowych

Najczęściej spotykanym obszarem zastosowań technicznych sieci neuronowych są zagadnienia rozpoznawania, klasyfikacji, analizy oraz przetwarzania obrazów, w którym można wyróżnić: kompresję, segmentację, odtwarzanie oraz interpretację obrazów według wzorca. Aby sklasyfikować i rozpoznawać wzorce, sieć uczy się podstawowych cech wzorca, takich jak: odwzorowanie geometryczne, pikselowy układ wzorca, rozkład składników głównych wzorca, składniki transformacji lub inne właściwości. W trakcie procesu uczenia się wyznaczane są różnice występujące między wzorcami.

Kolejną dziedziną zastosowań sztucznych sieci neuronowych jest ich wykorzystanie do zadań przetwarzania sygnałów, takich jak: konwersja, filtracja lub aproksymacja. Inne, często spotykane zastosowania sieci neuronowych dotyczą robotyki, automatyki, a także teorii sterowania i zagadnień optymalizacji, percepcji ruchu i jego planowania. W zagadnieniach identyfikacji i sterowania procesami dynamicznymi sieć neuronowa pełni kilka funkcji. Stanowi model nieliniowy tego procesu, pozwalający na wypracowanie odpowiedniego sygnału sterującego. Pełni również funkcje układu śledzącego i nadążnego, adaptując się do zmiennych warunków środowiskowych. Ważną rolę, zwłaszcza w sterowaniu robotów, odgrywa także funkcja klasyfikatora dalszego przebiegu procesu, wykorzystywana w podejmowaniu decyzji.

Sieci neuronowe są stosowane do budowania pamięci asocjacyjnej. W zadaniach asocjacji sieć neuronowa pełni funkcję pamięci skojarzeniowej. W przypadku pamięci asocjacyjnej skojarzenie dotyczy tylko poszczególnych składników wektora wejściowego. W pamięci heteroasocjacyjnej zadaniem sieci jest skojarzenie ze sobą dwóch wektorów. Gdy na wejściu będzie podany wektor odkształcony, sieć neuronowa jest w stanie odtworzyć wektor oryginalny, pozbawiony szumów.

Sieci neuronowe są wykorzystywane do przewidywania określonych rozwiązań na podstawie danych początkowych. Przykładami mogą być wszelkiego rodzaju prognozy ekonomiczne, np. przewidywania wahań giełdowych czy prognozowanie kondycji finansowej przedsiębiorstwa. Analiza danych prowadzona z wykorzystaniem sieci pozwala np. ustalić przyczyny niepowodzeń określonych przedsięwzięć podejmowanych w przeszłości, dzięki czemu łatwiej będzie unikać błędów w przyszłości. Sieci mogą klasyfikować na podstawie otrzymanych danych bilansowych obecny poziom rozwoju i kondycję przedsiębiorstwa. Mogą być również wykorzystywane w zagadnieniach optymalizacyjnych, kiedy poszukują rozwiązań wspomagających podejmowanie optymalnych decyzji gospodarczych.

## 9.3. Algorytmy genetyczne

Istotą algorytmów genetycznych jest naśladowanie zjawisk zachodzących w przyrodzie ożywionej, w której kolejne pokolenia roślin i zwierząt dziedziczą cechy swoich gatunków, przekazując je w postaci kodu genetycznego. W trakcie realizacji tego przekazu następuje selekcja, reprodukcja, krzyżowanie się oraz mutacja łańcuchów genetycznych poszczególnych osobników. Istotą tego zjawiska jest tworzenie się pamięci genetycznej kolejnych pokoleń, która utrwała cechy najsilniejszych i najliczniejszych swych przedstawicieli.

### 9.3.1. Idea budowy i schemat funkcjonowania algorytmów genetycznych

**Algorytmy genetyczne** (AG) należą do szerszej grupy algorytmów ewolucyjnych (AE), wykorzystywanych przede wszystkim w **zadaniach optymalizacji**<sup>2</sup>. Nazwa AG wywodzi się ze sposobu, w jaki wyszukiwane jest rozwiązanie przypominające mieszanie się i dobór genów w trakcie ewolucji. W działaniu AG podczas rozwiązywania problemów rzeczywistych można wyróżnić dwie fazy [Chodak, Kwaśnicki 2002, s. 2–3]:

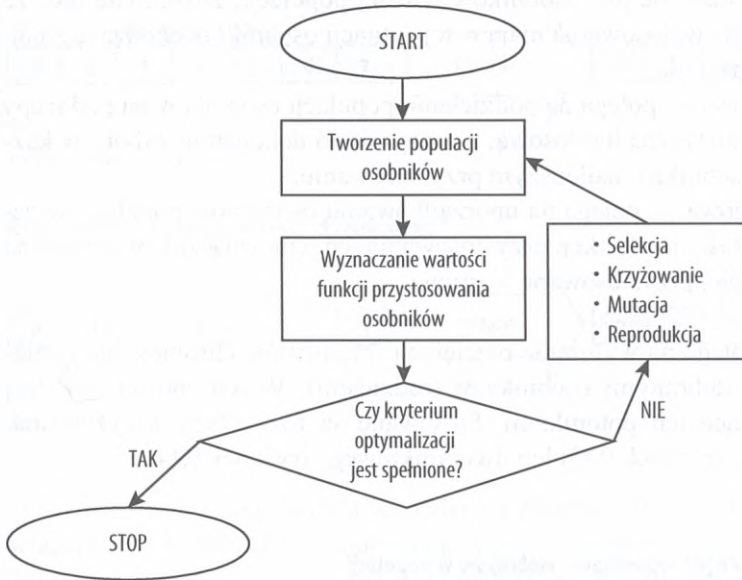
- 1) fazę wstępną — generowania wstępnej populacji dopuszczalnych rozwiązań zadania;
- 2) fazę poszukiwania rozwiązań — generowania i poszukiwania rozwiązań, które najlepiej spełnią zdefiniowane cele, warunki i kryteria ograniczające zadania.

Podstawowa wersja AG, zaproponowana przez J. Hollanda [Holland 1975], nazywana jest prostym lub elementarnym algorytmem genetycznym (ang. *Simple Genetic Algorithm* — SGA). Jego działanie prezentuje rysunek 9.11 i pokazuje, w jaki sposób dąży on do wygenerowania jak najlepiej przystosowanej populacji osobników. SGA rozpoczyna swoje działanie od stałej populacji losowo lub deterministycznie wygenerowanych osobników, gdzie dla każdego obiektu obliczana jest wartość jego funkcji przystosowania do środowiska. W celu utworzenia nowej, lepszej (czyli lepiej przystosowanej) populacji są wykonywane operacje genetyczne: selekcji, krzyżowania, mutacji i reprodukcji. Po ich zakończeniu stara populacja zastępowana jest przez nową. Ta pętla jest powtarzana do momentu, kiedy zostanie spełniony zdefiniowany warunek optymalizacyjny (lub warunek zatrzymania) dla algorytmu.

Dane osobników, na jakich działa algorytm, kodowane są binarnie. Struktura osobnika, zwana **chromosomem**, stanowi  $n$ -elementowy ciąg kodowy o stałej długości, którego pojedynczy znak nazywany jest genem i oznacza określoną cechę osobnika (rysunek 9.12).

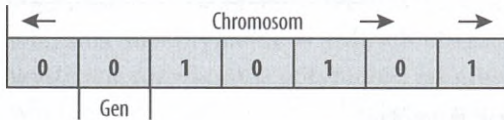
<sup>2</sup> Zadanie optymalizacji — poszukiwanie rozwiązania najlepszego ze względu na przyjęte kryteria w zbiorze rozwiązań dopuszczalnych.

Rysunek 9.11. Definicja osobnika w populacji SGA



Źródło: Opracowanie własne.

Rysunek 9.12. Struktura osobnika w populacji algorytmu SGA



Źródło: Opracowanie własne na podstawie: [Klosow, Jasiński 2007].

Geny przyjmują dwie wartości: 1 — cecha występuje u osobnika populacji, 0 — cecha nie występuje u osobnika populacji. Każdy binarny ciąg kodowy reprezentuje argument funkcji przystosowania osobnika do środowiska. Funkcja przystosowania stanowi miernik zysku, wydajności, jakości, użyteczności lub innej wielkości, która ma być optymalizowana poprzez AG. Optymalizacja polega na sprawdzaniu wartości funkcji przystosowania dla coraz to nowych generacji populacji osobników w poszukiwaniu wartości optymalnej. Tworzenie kolejnych zbiorów populacji wymaga zastosowania metod selekcji, krzyżowania, mutacji i reprodukcji.

**Selekcja osobników** populacji, mająca na celu zwielokrotnienie kodów genetycznych tych najlepiej przystosowanych, odbywa się metodami typu:

- ruletka — polega na przyporządkowaniu każdemu osobnikowi populacji sektora koła o rozmiarze proporcjonalnym do wartości jego funkcji przystosowania;

następnie realizowane jest losowanie fragmentów koła (oznaczonych liczbami na ruletce) tyle razy, ile jest osobników w całej populacji; zrozumiałe jest, że największe szanse wylosowania mają w tej sytuacji osobniki pochodzące z większego obszaru koła;

- selekcja turniejowa — polega na podzieleniu populacji osobników na podgrupy metodą deterministyczną lub losową, a następnie na dokonaniu wyboru w każdej podgrupie osobnika o najlepszym przystosowaniu;
- selekcja rankingowa — polega na uporządkowaniu osobników populacji w zależności od wartości ich funkcji przystosowania, co jednoznacznie wskazuje na przypadki najlepiej przystosowane w grupie.

Krzyżowanie polega na wymianie rozciętych fragmentów chromosomów między dwoma losowo dobranymi osobnikami (rodzicami). W ten sposób powstają nowe osobniki, będące ich potomkami. Stosowane są różne typy krzyżowania, np. jednopunktowego (rysunek 9.13) lub dwupunktowego (rysunek 9.14).

**Rysunek 9.13. Krzyżowanie jednopunktowe osobników w populacji**



Źródło: Jak rysunku 9.12.

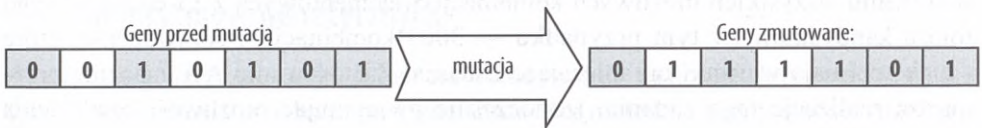
**Rysunek 9.14. Krzyżowanie dwupunktowe osobników w populacji**



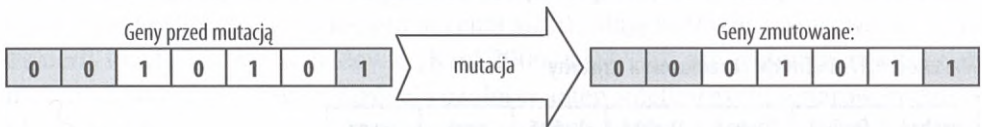
Źródło: Jak rysunku 9.12.

**Mutacja** powoduje zmianę wartości binarnej losowo dobranych genów na przeciwną (rysunek 9.15). Mutacja może polegać także na inwersji, czyli wzajemnej zamianie wartości losowo dobranych genów (rysunek 9.16). Możliwe jest również całkowite usunięcie genu z chromosomu.

Uzyskane w ten sposób nowe osobniki populacji, dziedziczące dobre cechy swoich rodziców i ograniczających cechy niekorzystne, na etapie reprodukcji zostają wielokrotnie powielone, generując w ten sposób nową populację osobników-potomków.

**Rysunek 9.15. Mutacja dwóch niesąsiadujących genów chromosomu**

Źródło: Jak rysunku 9.12.

**Rysunek 9.16. Inwersja odcinka chromosomu**

Źródło: Jak rysunku 9.12.

Opisane działanie AG jest wypadkową zdolności do eksploracji przestrzeni rozwiązań oraz możliwości znajdowania w niej rozwiązań optymalnych [Wierchoń 2001]. Wzrost nasilenia krzyżowania, mutacji bądź obniżenie naporu selekcyjnego<sup>3</sup> osobników zwiększa różnorodność populacji, wpływając pozytywnie na zdolności eksploracyjne AG. Jednocześnie zmniejszeniu ulegają jego zdolności eksploatacyjne, związane z poszukiwaniem rozwiązań optymalnych. Ponadto zwiększenie naporu selekcyjnego poprawia zdolności eksploatacyjne algorytmu kosztem zmniejszenia jego zdolności eksploracyjnych. W efekcie algorytm może szybciej znajdować rozwiązania suboptymalne w jednych obszarach, gubiąc jednak te obszary, w których występują rozwiązania optymalne. Istotnym problemem staje się utrzymanie równowagi między zdolnościami eksploracyjnymi a eksploatacyjnymi AG. Wsparciem dla tego celu ma być zatem stosowanie właściwego doboru metod selekcji, krzyżowania, mutacji i reprodukcji w trakcie jego realizacji.

### 9.3.2. Model decyzyjny algorytmu genetycznego

**Zadanie optymalizacyjne** polega na wybraniu za pomocą AG możliwie najlepszej pięcioosobowej drużyny spośród 15 kandydatów do teleturnieju, w którym 30% pytań jest z matematyki, 30% z informatyki, 20% z literatury, 15% z filmu i 5% z historii [Skorupska 2004]. Wybór członków drużyny jest dodatkowo ograniczony przez konieczność spełnienia następujących warunków: suma wieku wszystkich zawodników nie może przekroczyć 150 lat oraz w składzie muszą się zawsze znaleźć dwie kobiety i dwóch mężczyzn.

<sup>3</sup> Napór selekcyjny — stosunek funkcji przystosowania najlepszego osobnika w populacji do średniego przystosowania całej populacji.

Poszukiwanie dopuszczalnych rozwiązań metodą tradycyjną polegałoby na utworzeniu wszystkich możliwych kombinacji 5-elementowych z 15-elementowego zbioru kandydatów (w tym przypadku — 3003 kombinacji) i sprawdzeniu, które z nich spełniają warunki ograniczające zadania. Zastosowanie AG znacznie przyspiesza realizację tego zadania, jednocześnie gwarantując możliwość znalezienia kombinacji optymalnej dla zdefiniowanych ograniczeń.

Kodowanie chromosomu drużyny polega na zdefiniowaniu zestawu osób wchodzących w jej skład, przy uwzględnieniu narzuconych warunków ograniczających oraz oceny szansy tak utworzonego zespołu na sukces w teleturnieju (rysunek 9.17).

**Rysunek 9.17. Definicja chromosomu drużyny**

Osoba1	Osoba2	Osoba3	Osoba4	Osoba5	wiek	ocena
--------	--------	--------	--------	--------	------	-------

Źródło: Opracowanie własne na podstawie: [Skorupska 2004, s. 157].

Gen  $Osoba_N$  w chromosomie ma strukturę złożoną, obejmującą biegłość osoby w obszarach poszczególnych dziedzin teleturnieju, jej wiek oraz płeć (rysunek 9.18).

**Rysunek 9.18. Definicja pola osoba**

Nazwa	matematyka	informatyka	literatura	film	historia	wiek	płeć
-------	------------	-------------	------------	------	----------	------	------

Źródło: Jak rysunku 9.17.

Przykładowe geny osoby płci męskiej oraz płci żeńskiej mogą przyjmować następującą postać: {'Jan Kowalski', 1, 1, 0, 1, 0, 45, 1}, {'Anna Nowak', 0, 0, 1, 1, 1, 26, 0}. Populacja rodzicielska została wygenerowana losowo z posiadanego zbioru kandydatów. Wartość funkcji przystosowania była określona na podstawie stopnia spełnienia zbioru warunków ograniczeń oraz oceny szansy drużyny na sukces w teleturnieju. Do generowania kolejnych pokoleń osobników zostały wykorzystane metody krzyżowania po trzecim genie oraz selekcja losowa. AG kończył swoje działanie w chwili, kiedy były wygenerowane wszystkie pokolenia populacji.

W wyniku działania AG uzyskano osobniki o bardzo wysokiej wartości funkcji przystosowania (nawet do 100%) oraz spełniające wszystkie warunki ograniczeń, ale w zdecydowanie krótszym czasie niż przy użyciu metod tradycyjnych [Skorupska 2004, s. 160]. Ten prosty przykład obrazuje sposób, w jaki AG mogą być wykorzystane w SWD jako narzędzie wspomagające ocenę i wybór decyzji optymalnej. Dobór kryteriów oceny początkowej populacji wariantów decyzyjnych oraz warunków ograniczających może być oczywiście różny i uzależniony od rozwiązywanego problemu. W każdej jednak sytuacji, korzystając z metod generowania kolejnych pokoleń decyzji o coraz wyższej funkcji przystosowania, zbliżamy się do odnalezienia decyzji optymalnej w zbiorze rozwiązań dopuszczalnych.



### 9.3.3. Przykłady praktycznych zastosowań algorytmów genetycznych

Algorytmy genetyczne znajdują zastosowanie wszędzie tam, gdzie poszukiwany jest sposób rozwiązania problemu przy zdefiniowanej metodzie oceny jakości rozwiązania. Stosowane są zatem w matematyce, np. do poszukiwania przybliżeń ekstremów funkcji lub znajdowania rozwiązania nieliniowych równań różniczkowych. Wykorzystuje je również inżynieria komputerowa w systemach rozpoznawania wzorców i obrazów [Goldberg 2003] albo jako narzędzia wspierające działanie sieci neuronowych (np. do sterowania procesem uczenia sieci). Mają szerokie zastosowanie w inżynierii i technice, np. w budowie [Król 2006, s. 14–15]: samolotów niewykrywalnych przez radary, wysokiej jakości i czułości anten satelitarnych, sonarów, radarów i kamer wideo oraz robotów i automatów mobilnych. Najszerszym obszarem zastosowań AG jest optymalizacja, np. w obszarach: planowania i harmonogramowania zadań produkcji [Setlak 2004], poszukiwania optymalnego rozmieszczenia obiektów lub połączeń między elementami struktury albo sieci [Makarewicz, Zawieska 2003] czy optymalizacji rozwiązań problemów transportowych i logistycznych [Zduńczuk, Przystupa 2008]. Algorytmy genetyczne wspomagają badania i analizy ekonomiczno-rynkowe, np. w zakresie prognozowania wielkości popytu [Chodak 2002] lub zachowań klientów [Goldberg 2003], doboru partnerów inwestycyjnych [Rybarski, Tadeusiewicz 2006] czy wspomagania decyzji giełdowych [Klimek 2006]. Wykorzystywane są również szeroko w nauce: w fizyce — do modelowania zjawisk, w których występuje duża liczba zmiennych dynamicznych, w biologii — do modelowania zachowania populacji świata zwierząt, w chemii — do generowania nowych lub modyfikacji istniejących związków chemicznych (np. leczniczych).

## 9.4. Algorytmy mrówkowe

Algorytmy mrówkowe naśladują mechanizm rozpoznawania śladów feromonowych (zapachowych) pozostawianych przez mrówki, które odniosły „sukces” w chaotycznym poszukiwaniu drogi do celu. Wzmocnienie śladu feromonowego następuje w rezultacie potwierdzenia go przez inne mrówki, podążające drogą oznaczoną feromonem.

### 9.4.1. Idea budowy i schemat funkcjonowania algorytmów mrówkowych

**Algorytm mrówkowy** (ang. *Ant System* — AM) naśladuje zachowania społeczne mrówek i ich sposób poszukiwania najkrótszej drogi od mrowiska do pożywienia. Mrówki, wędrując, pozostawiają na przebytej trasie substancję, zwaną feromonem. Osobniki decydują o wyborze drogi na podstawie ilości feromonu na niej się znajdu-

jącego, co stanowi metodę komunikacji pośredniej między nimi. AM stanowią zatem metodę uczenia grupowego (inaczej: inteligencji grupowej).

AM wykorzystują kolonię osobników (agentów) do budowy rozwiązania problemu na podstawie pamięci grupowej, zwanej feromonem lub śladem feromonowym. W pierwszym kroku algorytmu agenci są ustawiani losowo w wybranych węzłach grafu. W kolejnych krokach każdy osobnik wybiera swoją drogę poprzez iteracyjne dodawanie kolejnych jej etapów. W trakcie wykonywania iteracji podejmowana jest decyzja, który z elementów dozwolonego zbioru tras należy wybrać. Kryterium wyboru trasy są wartości funkcji odległości między kolejnymi węzłami oraz śladu feromonowego odłożonego na krawędzi. Prawdopodobieństwo przejścia osobnika z punktu  $i$  do punktu  $j$  jest określone wzorem [Zduńczuk, Przystupa 2008, s. 240]:

$$p_{ij}^k(t) = \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{l \in N_i^k} ([\tau_{il}(t)]^\alpha \cdot [\eta_{il}]^\beta)} \quad \text{dla } j \in N_i^k, \quad [9.6]$$

gdzie:

- $\tau_{ij}(t)$  — natężenie feromonu na krawędzi łączącej punkty  $i$  oraz  $j$  w bieżącej iteracji  $t$ ;
- $\eta_{ij} = \frac{1}{d_{ij}}$  — odwrotność jakości krawędzi  $d_{ij}$  (długości lub czasu);
- $\alpha$  i  $\beta$  — wagi określające wpływ pozostawionego feromonu na obliczone prawdopodobieństwo;
- $N_i^k$  — dopuszczalne sąsiedztwo dla określonego położenia osobnika  $k$ , tj. zbiór jeszcze nieodwiedzonych punktów, do których istnieje połączenie z punktu  $i$ .

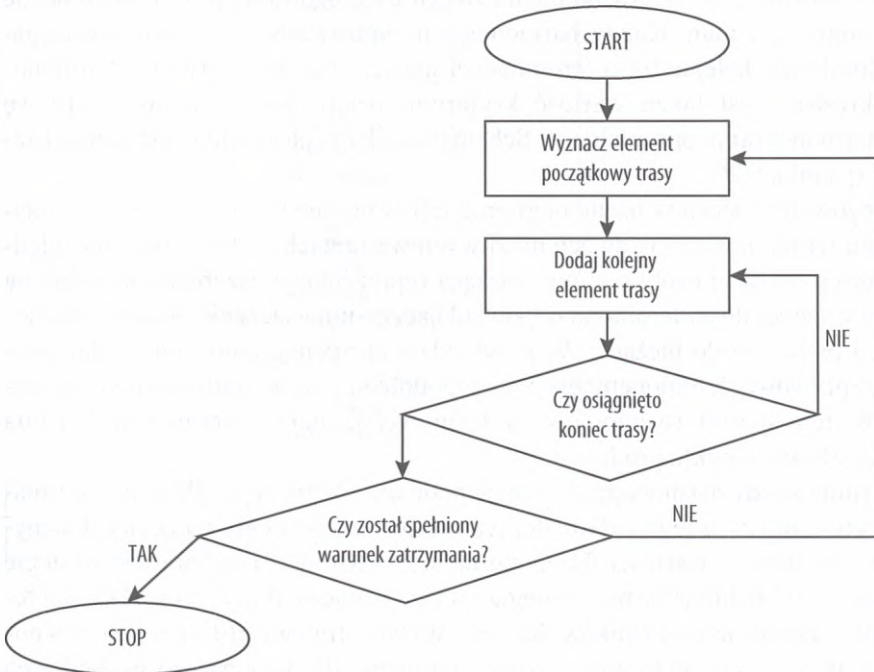
Wartości prawdopodobieństwa są przechowywane w tablicy i uaktualniane po każdorazowym naniesieniu nowych wartości feromonu. Przejście agenta do węzła odwiedzonego już wcześniej jest niedozwolone, a gwarantuje to tzw. lista tabu, czyli rejestr węzłów odwiedzonych przez poszczególnych agentów.

Po zbudowaniu kompletnego rozwiązania zostaje ono poddane ocenie, zgodnie z warunkami ograniczającymi zadania. Następnie aktualizowana jest wartość feromonu na ścieżkach, w wielkości proporcjonalnej do jakości otrzymanych rozwiązań. Istnieją dwie podstawowe strategie aktualizacji wartości feromonu:

- 1) wszystkie osobniki pozostawiają feromon na odwiedzonych ścieżkach — wartość feromonu zostaje zaktualizowana w momencie przebycia przez każdego z osobników określonej trasy;
- 2) feromon pozostawia tylko najlepszy w danym cyklu osobnik — wartość feromonu jest aktualizowana dopiero po zakończeniu całego cyklu działania algorytmu.

Po każdym pełnym cyklu algorytmu sprawdzany jest warunek zatrzymania. Jeżeli został on spełniony, to algorytm kończy swoje działanie i zapisywane jest najlepsze odnalezione rozwiązanie. W przeciwnym przypadku rozpoczyna się kolejny cykl. Warunek zatrzymania jest albo narzuceniem określonej liczby cykli do realizacji, albo brakiem poprawy osiągniętego rozwiązania w określonej liczbie cykli (tzw. stagnacja rozwiązania). Schemat działania AM prezentuje rysunek 9.19.

Rysunek 9.19. Schemat działania algorytmu mrówkowego



Źródło: Opracowanie własne.

## 9.4.2. Model decyzyjny algorytmu mrówkowego

Przykładem zastosowania algorytmu mrówkowego w systemie SWD jest harmonogramowanie produkcji przepływowej, przedstawione w [Kostrubiec 2002]. Celem AM jest optymalizacja harmonogramu ze względu na minimalizację czasu wykonania wszystkich zadań.

Algorytm AM został skonstruowany na podstawie macierzy kwadratowej  $\varphi$  o rozmiarze odpowiadającym liczbie zleceń do uszeregowania w harmonogramie:

$$\Phi = \begin{bmatrix} \varphi_{1,1} & \varphi_{1,2} & \cdots & \varphi_{1,N} \\ \varphi_{2,1} & \varphi_{2,2} & \cdots & \varphi_{2,N} \\ \cdots & \cdots & \cdots & \cdots \\ \varphi_{N,1} & \varphi_{N,2} & \cdots & \varphi_{N,N} \end{bmatrix}. \quad [9.7]$$

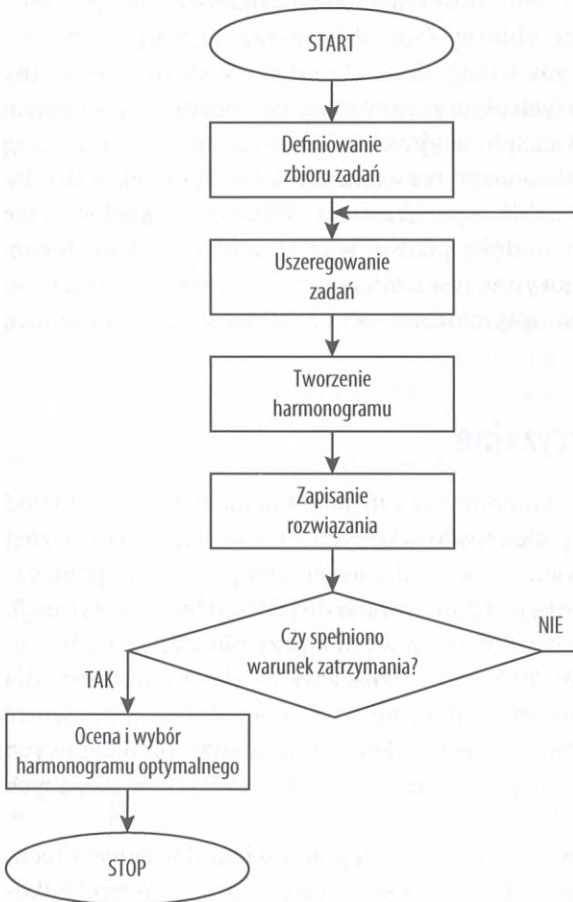
Trasa budowana przez mrówkę składa się ze zleceń oraz ich uszeregowania. Poprzez określenie czasów uruchomienia zleceń uszeregowanie jest przekształcane w harmonogramy zadań. Każdy harmonogram zapisywany jest do zbiorowej pamięci kolonii jako kolejna trasa feromonowa macierzy  $\Phi$ . Na podstawie harmonogramu określana jest także wartość kryterium oceny, która stanowi podstawę wyboru harmonogramu optymalnego. Schemat działania algorytmu został przedstawiony na rysunku 9.20.

Szeregowanie zleceń w harmonogramie jest tworzone na podstawie wcześniejszego śladu feromonowego w dwóch możliwych wariantach — tylko przy uwzględnieniu pozycji bieżącej osobnika (tzw. bieżąca reprezentacja uszeregowania) lub na podstawie ważonej ilości feromonu odpowiadającego umieszczeniu osobnika na pozycjach od pierwszej do bieżącej. W przykładzie harmonogramowania zadań produkcji przepływowej feromonem jest prawdopodobieństwo wyboru zadania  $i$  na pozycji  $j$ . W ten sposób tworzone są kolejne wersje harmonogramu wykonania wszystkich zleceń w cyklu produkcji.

Następnie każdy harmonogram zostaje poddany ocenie ze względu na minimalizację kryterium czasu jego całkowitej realizacji. Na podstawie tej oceny dokonywana jest aktualizacja wartości feromonu na ścieżkach. Możliwe są dwie strategie aktualizacji: zwykła lub elitarna. Strategia zwykła oznacza aktualizację wartości feromonu dla wszystkich osobników ścieżki. Wybór strategii elitarniej jest równoznaczny z aktualizacją wyłącznie wartości feromonu dla najlepszego osobnika na ścieżce (o najmniejszej wartości funkcji oceny). Dobór właściwej strategii aktualizacji jest poszukiwaniem stanu równowagi między zdolnością eksplorowania nowych obszarów przestrzeni rozwiązań a umiejętnością ich eksploatacji, czyli wyszukiwania rozwiązań optymalnych.

Na podstawie zbudowanego algorytmu możliwe jest znalezienie rozwiązania optymalnego wśród wielu wariantów rozwiązań dopuszczalnych z zastrzeżeniem spełnienia warunków ograniczających i optymalną wartością kryterium oceny klasyfikacji przypadków.

Rysunek 9.20. Schemat działania algorytmu mrówkowego dla harmonogramowania produkcji przepływowej



Źródło: Opracowanie własne.

### 9.4.3. Przykłady zastosowań algorytmów mrówkowych

AM znajdują zastosowanie przy rozwiązywaniu trudnych problemów, często nierozwiązywalnych za pomocą algorytmów deterministycznych. Wykorzystuje się je m.in. do rozwiązywania problemów plecakowych, szeregowania zadań lub trasowania pakietów w sieciach bezprzewodowych i optycznych.

Algorytmy służą do rozwiązywania trudnych problemów kombinatorycznej optymalizacji, takich jak np. problem komiwojażera. Znajdują również zastosowanie w przypadku rozwiązywania dyskretnych problemów optymalizacyjnych, np. do wyznaczania tras pojazdów, sortowania sekwencyjnego, wyznaczania tras w sieci komputerowej czy telekomunikacyjnej.

Różnorodność tych zadań powoduje, że powinny być do nich dobierane różne typy algorytmów mrówkowych. Do tych najbardziej znanych można zaliczyć: gęstościowy (ang. *Density Ant System*), ilościowy (ang. *Quantity Ant System*), cykliczny (ang. *Cycle Ant System*), elitarny (ang. *Elitist Ant System*), graniczny (ang. *Max-Min Ant System*), rankingowy (ang. *Rank-Based Ant System*) i kolonialny (ang. *Ant Colony System*). Każdy z tych algorytmów może być oceniany pod kątem jego własności eksploatacyjnych i eksploracyjnych. Własności eksploatacyjne są mierzone jakością najlepszego znalezionej rozwiązania w każdym cyklu działania algorytmu oraz czasu trwania cyklu wyszukiwania. Własności eksploracyjne algorytmu są oceniane pod kątem średniej jakości wszystkich znalezionych rozwiązań oraz ich entropii. Dobór algorytmu jest uzależniony od wielkości i wartości zbioru obiektów oraz typu zadania optymalizacyjnego, które ma być za pomocą tego algorytmu rozwiązane.

## 9.5. Hybrydowe modele decyzyjne

**Hybrydowe modele decyzyjne** są kompozycją lub uogólnieniem różnych metod i technik stosowanych w procesach podejmowania decyzji z zakresu metod sztucznej inteligencji. Model hybrydowy zawiera wyspecjalizowane komponenty wspomagające proces decyzyjny, a jego struktura jest dopasowana do potrzeb bieżącej sytuacji. Przykładowo, do stosowanych rozwiązań hybrydowych należy połączenie sieci neuronowych i algorytmów genetycznych, które wyznaczają w SWD parametry dla funkcji aktywacji sieci. Podejście to wzmacnia rolę SWD w organizacji, ponieważ umożliwia objęcie swoim działaniem znacznie szerszych obszarów problemowych oraz uzyskanie efektu synergicznego dopasowania modeli decyzyjnych do bieżących potrzeb użytkowników.

Idea hybrydowego SWD (HSWD) bazuje na integracji wielu dostępnych technik, metod i narzędzi w jednej spójnej strukturze stanowiącej logicznie uporządkowaną całość, dążącą do rozwiązania problemu decyzyjnego. Te połączenia tworzy się w celu osiągnięcia bardziej zadowalającego efektu niż przy wykorzystaniu każdej techniki oddzielnie.

### 9.5.1. Idea i struktura hybrydowych modeli decyzyjnych

Znane są dwa główne podejścia do tworzenia systemów hybrydowych: zastosowanie **obliczeń inteligentnych** (ang. *Computational Intelligence* — CI) lub wykorzystanie **miękkich technik obliczeniowych** (ang. *Soft Computing* — SC) [Sroka, Wolny 2009, s. 129]. HSWD w podejściu CI (obliczeniowo inteligentne) przetwarzają dane wyłącznie numeryczne, wykorzystując metody rozpoznawania wzorców. Wykazują się przy tym szybkością działania, odpornością na błędy obliczeniowe oraz zdolnością do dostosowywania się. Natomiast nie potrafią posługiwać się wiedzą. Podejście SF, zachowując wszystkie własności CI, koncentruje się wła-

śnie na przetwarzaniu wiedzy. Dodatkowo, oprócz metod obliczeniowych wykorzystywanych przez CI, stosuje się również: uczenie maszynowe, teorię chaosu oraz wnioskowanie probabilistyczne.

HSWD integrują odmienne techniki sztucznej inteligencji (np. logikę rozmytą, sieci neuronowe, algorytmy ewolucyjne, wnioskowanie probabilistyczne, uczenie maszynowe, sztuczne życie czy drzewa regresyjno-decyzyjne). Istnieje wiele możliwych kombinacji tych technik w HSWD oraz dużo możliwych sposobów ich integracji. W strukturze systemu możliwe jest zastosowanie jednego, dwóch lub wielu technik i algorytmów jednocześnie. Wśród typów hybrydyzacji modeli HSWD wyróżnia się [Sroka, Wolny 2009, s. 135–140]:

- modele niezależne — w trakcie rozwiązywania problemu decyzyjnego poszczególne komponenty systemu działają równolegle, nie wchodząc we wzajemne interakcje; stwarza to możliwość obiektywnej oceny i porównania wyników, pozyskanych przez zastosowanie różnych technik analityczno-decyzyjnych;
- modele transformowalne — komponenty nadal są niezależne, ale wykorzystywane w nich techniki obsługują różne fazy rozwiązywania danego problemu decyzyjnego; zatem uzyskane z nich wyniki sumarycznie składają się na ostateczne rozwiązanie;
- modele swobodnie powiązane — integrują komponenty systemu poprzez komunikację opartą na plikach tekstowych oraz zewnętrzne pliki wymiany danych; powiązanie może być realizowane na poziomie: przygotowania danych do przetwarzania, transformacji danych między kolejnymi etapami przetwarzania, wzajemnej wymiany danych między komponentami systemu lub komunikacji z użytkownikiem systemu;
- modele ściśle powiązane — komponenty systemu integrowane są poprzez struktury danych, czasem również poprzez zewnętrzne pliki wymiany danych;
- modele w pełni zintegrowane — pełna integracja komponentów oznacza współdzielenie struktur danych oraz reprezentacji wiedzy w systemie.

Ze względu na sposób integracji technik w systemie wyróżnia się HSWD [Sroka, Wolny 2009, s. 140–141]:

- z wymianą funkcji — poszczególne funkcje systemu są realizowane poprzez zastosowanie różnych technik i metod w taki sposób, aby uzyskać jak najlepsze rezultaty funkcjonowania całego systemu;
- wzajemnie komunikujące się — system składa się z niezależnych modułów, odpowiedzialnych za rozwiązanie poszczególnych zadań wydzielonych wewnątrz problemu decyzyjnego; moduły komunikują się ze sobą poprzez wzajemną wymianę danych i informacji;
- polimorficzne — wykorzystuje się jedną technikę, która imituje działanie wielu różnych technik; zaletą jest jednorodność architektury systemu przy jednoczesnym rozszerzeniu jego własności funkcjonalnych.

## 9.5.2. Przykłady rozwiązań hybrydowych modeli decyzyjnych

Wśród najczęściej stosowanych połączeń w HSWD wyróżnia się [Sroka, Wolny 2009, s. 141–148]: systemy ekspertowe i sieci neuronowe, systemy rozmyte i systemy ekspertowe, systemy rozmyte i sieci neuronowe, algorytmy genetyczne i sieci neuronowe, algorytmy genetyczne i systemy rozmyte, algorytmy genetyczne i systemy ekspertowe oraz hybrydy systemu CBR (ang. *Case Based Reasoning* — wnioskowanie na podstawie przypadków).

Połączenie systemów ekspertowych z sieciami neuronowymi jest bardzo popularnym modelem HSWD, który znalazł zastosowanie w zagadnieniach giełdowych (np. prognozowanie kursów akcji, szacowanie ryzyka inwestycyjnego). W tej kombinacji systemy ekspertowe lepiej radzą sobie w sytuacji możliwości zastosowania logicznego, symbolicznego podejścia do rozwiązania problemu, a sieci neuronowe stosowane są w przypadku przetwarzania danych numerycznych przy braku znajomości reguł rozwiązania problemu.

Najczęściej preferowanym podejściem w połączeniu systemów rozmytych i systemów ekspertowych jest pełna integracja technik, prowadząca do powstania rozmytego systemu ekspertowego. Otrzymany w ten sposób efekt synergii polega na uzyskaniu możliwości przetwarzania informacji niepełnych i nieprecyzyjnych systemu rozmytego, przy zachowaniu zasad interakcyjnej komunikacji z użytkownikiem systemu ekspertowego. Ten typ hybrydowego modelu decyzyjnego znalazł swoje zastosowanie w marketingu przy diagnozie otoczenia marketingowego oraz wspieraniu przygotowywania strategii marketingowych.

Systemy rozmyte w połączeniu z sieciami neuronowymi wnoszą do HSWD wysoką tolerancję na nieprecyzyjność danych i informacji, zdolność uczenia się na zgromadzonych zbiorach danych oraz możliwość jawnej interpretacji wiedzy w systemie. Przykładem zastosowania takiego rozwiązania jest wspomaganie planowania strategicznego w przedsiębiorstwie czy analiza portfelowa akcji notowanych na giełdach papierów wartościowych.

Integracja algorytmów genetycznych i sieci neuronowych jest ukierunkowana na optymalizację architektury i eksploatacji sieci neuronowych, a w szczególności na [Gwiazda 1998]: optymalizację typologii sieci, selekcję algorytmów uczenia sieci, optymalizację parametrów kontrolnych sieci oraz tworzenie nowych parametrów na bazie już istniejących. Znane są zastosowania takiego połączenia do szacowania ryzyka kredytowego czy prognozowania finansowych szeregów czasowych.

W modelach hybrydowych łączących algorytmy genetyczne i systemy rozmyte komponent rozmyty zapewnia możliwość czytelnej reprezentacji wiedzy, a algorytm genetyczny wspomaga optymalizację działania systemu rozmytego. Połączenie takie było wykorzystane do efektywnego rozwiązania problemów odnoszących się do optymalizacji alokacji zasobów w procesie produkcyjnym oraz jako narzędzie symulacyjne wspomagające procesy diagnostyczne.



Algorytmy genetyczne w systemach ekspertowych stosowane są do parametryzacji i odkrywania reguł, a w szczególności do [Sroka, Wolny 2009, s. 147]: szacowania współczynników pewności reguł, uzyskiwania wartości zmiennej wyznaczającej liczbę możliwych rozwiązań, wspomaganie przetwarzania danych numerycznych, parametryzacji rozwiązywanego problemu w postaci wartości numerycznych (możliwych do przetworzenia przez sieci neuronowe). Z kolei systemy ekspertowe mogą dostarczać heurystyk poprawiających funkcjonowanie algorytmów genetycznych w zakresie [Medsker 1998, s. 203]: nadzorowania procesów przez system ekspertowy, budowy interfejsów komunikacji z użytkownikiem, tworzenia systemu wyjaśnień oraz jawnej reprezentacji wiedzy, wyznaczania heurystyk ograniczających przestrzeń rozwiązań algorytmów genetycznych. Przykładem zastosowania takiego połączenia jest selekcja obiektów w zadaniach klasyfikacji i grupowania.

Połączenie systemu CBR i innych technologii inteligentnych jest stosowane w kilku różnych kombinacjach. Jedną z możliwości jest kompozycja systemu CBR i systemu ekspertowego w postaci modeli niezależnych lub transformalnych. W pierwszym przypadku możliwa jest podwójna weryfikacja uzyskanego rozwiązania, pogłębiająca wiedzę o problemie. W drugim przypadku jeden z komponentów dostarcza wiedzę o problemie dla drugiego. Innym wariantem rozwiązania jest połączenie systemu CBR z systemem rozmytym, co skutkuje możliwością współpracy z informacjami niepełnymi i nieprecyzyjnymi. Połączenie z sieciami neuronowymi wspomaga proces wyszukiwania początkowego zbioru przypadków, a algorytmy genetyczne wspierają optymalizację technik reprezentacji przypadków oraz ich wyszukiwania.

## 9.6. Pytania i zadania kontrolne

### 9.6.1. Pytania kontrolne

1. Na podstawie modelu McCullocha–Pittsa omów zasadę działania sztucznego neuronu.
2. Omów proces uczenia sztucznej sieci neuronowej.
3. Co oznacza pojęcie „epoka uczenia w sztucznych sieciach neuronowych”?
4. Omów trzy podstawowe typy uczenia w sztucznych sieciach neuronowych.
5. Na czym polega optymalizacja działania sztucznych sieci neuronowych z wykorzystaniem algorytmu genetycznego?
6. Jaki jest związek między algorytmem genetycznym a algorytmem ewolucyjnym?
7. Jakie są różnice między operacją krzyżowania a mutacją?
8. Jakimi metodami i w jakim celu realizowana jest selekcja osobników populacji w algorytmie genetycznym?
9. Do czego wykorzystywany jest ślad feromonowy w algorytmie mrówkowym?

10. Omów strategie aktualizacji wartości feromonu na ścieżkach osobników w algorytmie mrówkowym.
11. Od czego zależą i jakie mają znaczenie własności eksploracyjne i eksploatacyjne algorytmu mrówkowego?
12. Co to jest system hybrydowy i dlaczego jest stosowany?
13. Jaka jest różnica między zastosowaniem obliczeń inteligentnych CI i miękkich technik obliczeniowych SF w HSWD?
14. Jakie są podstawowe typy hybrydyzacji HSWD?
15. Jakimi wyróżniamy typy systemów hybrydowych wykorzystujących modele wnioskowania na podstawie przypadków CBR?

### 9.6.2. Przykładowe zadania

1. Dobierz najlepszy model sztucznej sieci neuronowej do następujących zastosowań:
  - a) rozpoznawanie skupień i wzorców w danych,
  - b) systemy sterujące w automatyce i robotyce,
  - c) zagadnienie klasyfikacji i segmentacji danych,
  - d) aproksymacja funkcji wielu zmiennych,
  - e) rozproszone systemy sterujące,
  - f) odtwarzanie uszkodzonych lub niekompletnych obrazów kamer cyfrowych,
  - g) rozpoznawanie obrazów.
2. Zaprojektuj strukturę systemu wspomaganego decyzji z wykorzystaniem modelu sztucznej sieci neuronowej.
3. Wskaż obszary zastosowania algorytmu mrówkowego w procesie wspomaganego podejmowania decyzji logistycznych w przedsiębiorstwie.
4. Zaprojektuj strukturę systemu wspomaganego decyzji z wykorzystaniem algorytmów genetycznych.
5. Korzystając z metody analogii, zaprojektuj predyktora do określenia zbioru pracowników najmujących się do pracy w przedsiębiorstwie.
6. Zaproponuj zastosowanie i architekturę HSWD łączącego sieci neuronowe z algorytmem genetycznym. Na czym polega efekt synergii w takim systemie?
7. Zaproponuj zastosowanie i architekturę HSWD łączącego sieci neuronowe z algorytmem mrówkowym. Na czym polega efekt synergii w takim systemie?

# Integracja w systemach SWD

W rozdziale przedstawiono trzy typy rozwiązań SWD, w których dominującym elementem jest zagadnienie integracji — systemy Business Intelligence (BI), systemy ekspertowe (SE) oraz systemy hybrydowe (HSWD).

Coraz bardziej powszechną grupą SWD w zastosowaniach biznesowych staje się grupa systemów Business Intelligence (BI) wyposażona w metody i rozwiązania integrujące dane z zakresu całej organizacji. Zadaniem systemów BI jest dostarczenie kompletnej i aktualnej informacji we właściwym czasie oraz w oczekiwanej przez użytkownika formie. Wymiernym wskaźnikiem efektywności BI jest stopień i poziom integracji danych oraz zdolność do ich wielowymiarowego przetwarzania analitycznego, wspomagającego proces podejmowania decyzji.

Zadaniem SE jest wspomaganie procesów decyzyjnych użytkowników o różnym stopniu przygotowania merytorycznego oraz automatyczne monitorowanie procesów i generowanie wariantów decyzji koniecznych do zachowania ciągłości działania organizacji. Jest to możliwe dzięki temu, że dominującym elementem architektury każdego SE jest integracja warstwy danych z warstwą wiedzy. Użytkownicy otrzymują z SE gotowe rozwiązania, na poziomie zgodnym z ich przygotowaniem merytorycznym oraz bieżącymi potrzebami.

Systemy HSWD integrują różne modele i metody wspomagające podejmowanie decyzji w celu poprawienia ich sumarycznej skuteczności, wiarygodności i powtarzalności. Zastosowanie HSWD ma na celu poprawę uzyskiwanych z systemu wyników w stosunku do rozwiązań niehybrydowych.

## 10.1. Business Intelligence (BI)

**Business Intelligence (BI)** to grupa dynamicznie rozwijających się SWD. Ich rolą jest dostarczanie kompletnej, aktualnej i prawdziwej informacji we właściwym czasie oraz w oczekiwanej przez użytkownika formie i postaci. Do niedawna były to systemy

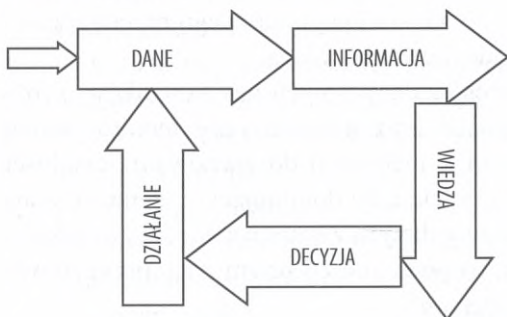
funkcjonujące w oparciu o dane historyczne. Obecnie coraz popularniejsze stają się operacyjne systemy BI, których zadaniem jest bieżące śledzenie zmian zachodzących w systemach transakcyjnych i błyskawiczne reagowanie na pojawiające się zagrożenia.

### 10.1.1. Koncepcja i istota działania BI

Systemy BI są obecnie najszybciej rozwijającą się technologią w grupie systemów wspomagania decyzji. Według Gartner Group [Amanowicz 2007]: „(...) Rozwiązaniem klasy BI można określić system zapewniający zestaw technologii oraz produktów udostępniających użytkownikom informacje niezbędne do prowadzenia działalności gospodarczej oraz podejmowania strategicznych decyzji biznesowych (...)”. Zdaniem A. Milera z firmy Oracle [Miller 2006]: „(...) Business Intelligence to zespół technologii do pozyskiwania, gromadzenia, udostępniania i analizowania informacji o przedsiębiorstwie (...)”. Według A. Kicingera z SAS Institute Polska [Jaworska 2005]: „(...) Business Intelligence to zintegrowana korporacyjna architektura informacyjna, z interfejsami dedykowanymi różnym grupom użytkowników, zapewniająca szybki, łatwy i efektywny dostęp do wiarygodnych informacji w procesie podejmowania decyzji (...)”.

Działanie BI opiera się na uporządkowanej sekwencji transformacji zgromadzonych przez przedsiębiorstwo danych, prezentowanej na rysunku 10.1.

**Rysunek 10.1. Idea działania systemu informacji zarządczej**

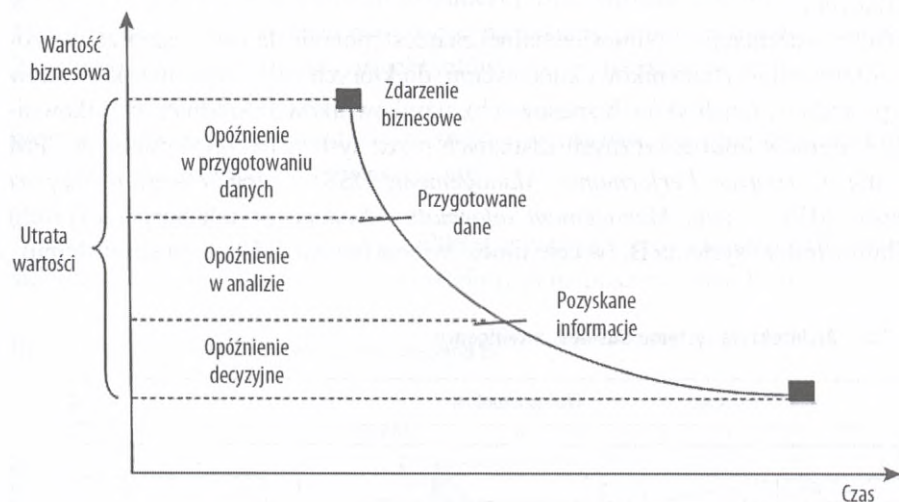


Źródło: [Rostek 2008a].

Rozwiązania BI, jako element efektywnego zarządzania firmą, pozwalają na dostęp do danych pochodzących z wielu, najczęściej rozproszonych źródeł danych. Następnie umożliwiają analizę tych danych dzięki metodom klasyfikacji, prognozowaniu i symulacji, co pozwala na pozyskanie nowej, nieznannej dotychczas i użytecznej informacji biznesowej. Informacje te zasilają zasoby wiedzy przedsiębiorstwa, a wiedza stanowi podstawę do podejmowania konkretnych decyzji zarządczych. Na koniec wdrożenie podjętych decyzji w praktyce biznesowej zapewnia efektywne zarządzanie firmą i możliwość uzyskania przewagi konkurencyjnej na rynku.

Dzięki automatyzacji etapów: gromadzenia i analizy danych oraz dystrybuowania informacji uzyskuje się ograniczenie trzech typów opóźnień: w przygotowaniu danych, w wykonaniu analizy oraz w podjęciu decyzji. Brak tych opóźnień wpływa z kolei na efektywność podejmowania decyzji (rysunek 10.2; [Hackathorn 2004]).

**Rysunek 10.2. Utrata wartości biznesowej podejmowanej decyzji zarządczej**



Źródło: [Jaworska 2005].

Zrealizowanie przedstawionych zadań systemu BI wymaga stworzenia architektury zapewniającej: integrację i transformację danych źródłowych, przetwarzanie analityczne tych danych, a następnie swobodną dystrybucję pozyskanych informacji.

### 10.1.2. Struktura i metody BI

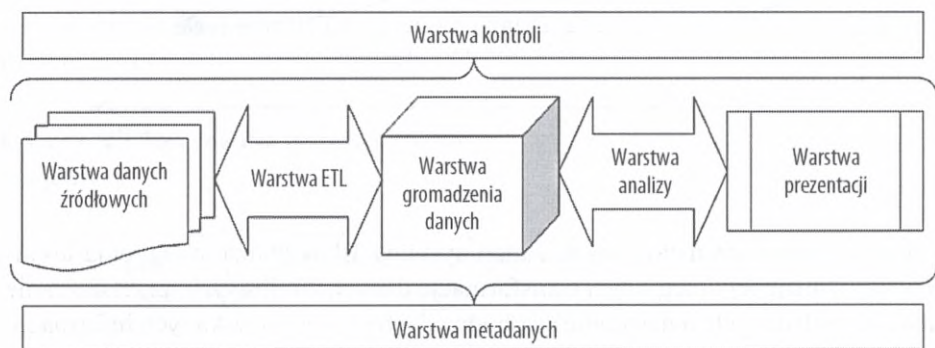
Architektura systemu BI składa się z następujących warstw (rysunek 10.3):

- warstwy danych źródłowych — obejmującej wszystkie dostępne i użyteczne dla analiz biznesowych źródła danych, w skład których wchodzi: transakcyjne BD, systemy informatyczne, witryny internetowe, pliki w różnych formatach, akty i przepisy prawne, archiwa papierowe itp.;
- warstwy ETL<sup>1</sup> odpowiedzialnej za pozyskiwanie danych, weryfikację ich jakości (profilowanie, standaryzacja, deduplikacja, wzbogacanie danych), odpowiednie przygotowanie danych do umieszczenia w strukturach analitycznych

<sup>1</sup> ETL (ang. *Extract, Transformation, Load* — ekstrakcja, transformacja, ładowanie) — operacja zasilania hurtowni danymi, składająca się z ekstrakcji danych z zasobów źródłowych (ekstrakcja), przygotowania ich do umieszczenia w strukturach hurtowni (transformacja) oraz załadowania do modelu danych hurtowni (ładowanie).

- (transformacja, wyznaczanie wartości obliczanych, wstępna agregacja) oraz łądowanie danych do warstwy gromadzącej je;
- warstwy gromadzenia danych — obejmującej repozytorium (lub repozytoria) analityczne, do których można zaliczyć: repozytoria danych operacyjnych, HD, składnice danych, kostki OLAP itp.;
  - warstwy analizy — odpowiedzialnej za udostępnienie narzędzi i obszaru analizy danych;
  - warstwy prezentacji — odpowiedzialnej za udostępnianie danych oraz przetworzonej informacji użytkownikowi końcowemu, do których zaliczamy: użytkowników bezpośrednich (analityków biznesowych), użytkowników pośrednich (użytkowników systemów informatycznych zasilanych przez system BI, np.: systemów CPM — ang. *Corporate Performance Management*; DSS — ang. *Decision Support System*; MIS — ang. *Management Information System*; portali korporacyjnych) i administratora systemu BI (w celu umożliwienia mu zarządzania pracą systemu).

Rysunek 10.3. Architektura systemu Business Intelligence



Źródło: Opracowanie własne.

Podstawowa struktura systemu jest integrowana przez dwie warstwy towarzyszące (rysunek 10.3):

- 1) warstwę kontroli — odpowiedzialną za bezpieczeństwo dostępu do danych oraz procesów pracy w systemie BI;
- 2) warstwę metadanych (zwaną również warstwą logiczną systemu) — odpowiedzialną za gromadzenie i rejestrowanie wszelkich definicji, procesów, zdarzeń i operacji zachodzących podczas działania systemu.

Przedstawiona architektura gwarantuje osiągnięcie podstawowych celów wdrożenia systemu BI:

- umożliwia integrację danych w obszarze całego przedsiębiorstwa,
- przechowuje dane w postaci przygotowanej do prowadzenia analiz biznesowych,

- udostępnia wyniki analiz bez opóźnień czasowych wszystkim uprawnionym odbiorcom, przy zachowaniu odpowiedniego poziomu bezpieczeństwa.

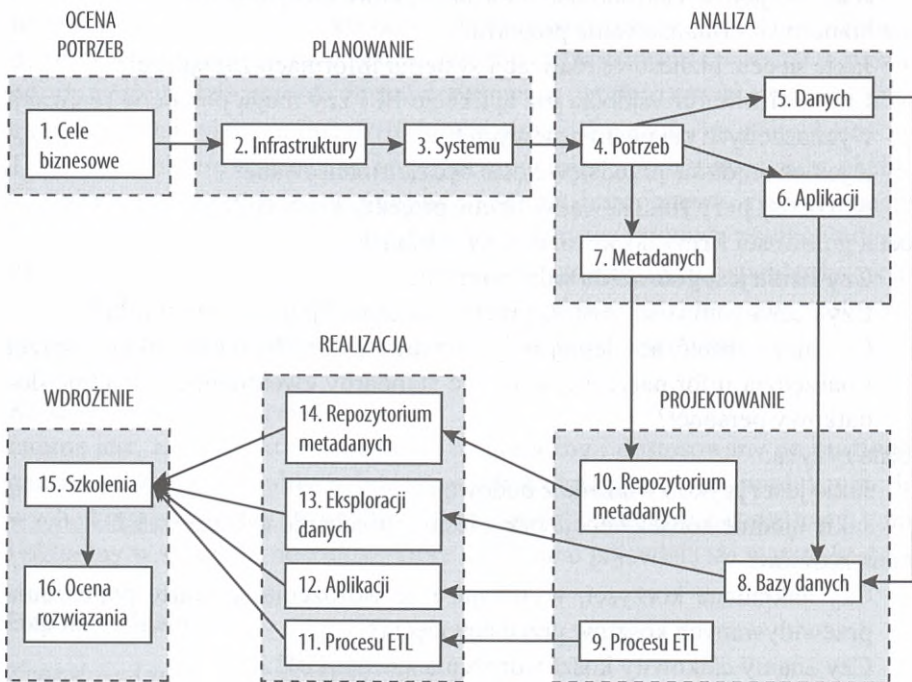
### 10.1.3. Projektowanie rozwiązań BI

Można wskazać trzy warunki konieczne udanego wdrożenia systemu BI, nazywanego również systemem informacji zarządczej (SIZ; [Rostek 2008b]):

- 1) doświadczenie i wiedza merytoryczna zespołu projektowo-wdrożeniowego;
- 2) intensywne współpraca projektantów i użytkowników przez cały okres realizacji i wdrażania systemu;
- 3) zastosowanie odpowiedniej metodyki wdrożenia, uwzględniającej możliwość podziału projektu na mniejsze etapy.

Według L.T. Moss i S. Atre [2003] metodyka projektowania BI powinna się składać z sześciu etapów (I–VI) podzielonych na poszczególne kroki (rysunek 10.4).

Rysunek 10.4. Metodyka projektowania systemu BI



Źródło: Opracowanie własne na podstawie: [Moss, Atre 2003].

## Etap I — Ocena potrzeb

### *Krok 1 — Wyznaczenie celów biznesowych*

System informacji zarządczej jest bardzo złożonym projektem, który musi być właściwie przygotowany i prowadzony. BI to nie tylko narzędzia informatyczne, ale także zbiór technik, rozwiązań biznesowych i wiele procesów, które należy odpowiednio wdrożyć. Łączy się to ze zmianami organizacyjnymi, zmianami w systemie zarządzania oraz poważnymi zmianami w strukturze i architekturze systemów informatycznych firmy. Wymienione zmiany pociągają za sobą wysokie nakłady finansowe i — jak wszystkie poważne wdrożenia informatyczne — duże ryzyko inwestycyjne. Dlatego konieczne jest szczegółowe zdefiniowanie celów oraz wskazanie wymiernych i niewymiernych korzyści wdrożenia.

Pytania, które muszą zostać zadane, oraz problemy, które muszą zostać rozwiązane na tym etapie, to [Moss, Atre 2003]:

- 1) dostęp do informacji:
  - Skąd pochodzą informacje, które są niezbędne do podejmowania decyzji?
  - Jakie informacje już otrzymaliśmy, a jakich jeszcze nam brakuje?
  - Jaka jest jakość i aktualność informacji, które otrzymujemy?
- 2) cele biznesowe i finansowanie projektu:
  - Jakie są cele biznesowe realizacji systemu informacji zarządczej?
  - Czy w firmie funkcjonują już aplikacje BI i czy mogą pomóc w realizacji wyznaczonych celów biznesowych?
  - Z jakich środków przedsięwzięcie będzie finansowane?
  - Kto do tej pory finansował w firmie projekty klasy BI?
- 3) ocena gotowości firmy do wdrożenia systemu BI:
  - Czy firma jest gotowa do wdrożenia BI?
  - Czy ocena gotowości została przeprowadzona i jakie są jej wyniki?
  - Co należy zrobić, aby lepiej przygotować się do wdrożenia: zakupić sprzęt i narzędzia informatyczne, wdrożyć standardy ewentualnie zatrudnić dodatkowy personel?
- 4) ocena ryzyka:
  - Jakie jest ryzyko związane z budową BI?
  - Jakie ujemne konsekwencje przyniesie zaniechanie wdrożenia BI?
- 5) ocena kosztów:
  - Czy zakładane korzyści, wynikające ze wdrożenia są warte ponoszenia przewidywanych kosztów przedsięwzięcia?
  - Czy znamy całkowity koszt wdrożenia systemu BI?
  - Czy musimy kupić nowy sprzęt komputerowy?
  - Czy musimy zmodernizować naszą sieć komputerową?
  - Czy potrzebne są nowe narzędzia informatyczne?
  - Czy musimy zatrudnić doradcę/doradców z obszaru BI?



- 6) zwrot nakładów inwestycyjnych — obliczenie współczynnika ROI (ang. *Return on Investment*) w oparciu o wpływ wdrożonego systemu na:
- efektywność pracy działu obsługi klienta,
  - zadowolenie klienta,
  - wzrost zysku firmy,
  - efektywność decyzji strategicznych,
  - obniżenie kosztów,
  - wzrost udziału firmy na rynku.

Przedstawioną listę należy traktować jako plan analizy, która musi poprzedzać w firmie podjęcie decyzji o wdrożeniu bądź zaniechaniu wdrożenia BI. Lista zawiera szeroki zakres niezbędnych pytań i badań pozwalający określić gotowość firmy do wdrożenia oraz rzeczywistą możliwość realizacji takiej inwestycji.

## **Etap II — Planowanie**

### *Krok 2 — Projekt infrastruktury*

Ponieważ system informacji zarządczej obejmuje swoim projektem całą organizację, więc również cała infrastruktura przedsiębiorstwa będzie musiała być dostosowana do realizacji tego projektu. Część elementów infrastruktury pozostanie niezmieniona, ale część będzie musiała zostać dopasowana zgodnie ze wskazaniami specyfikacji projektowej systemu. Infrastruktura obejmuje dwie główne grupy komponentów:

- 1) infrastrukturę techniczną: sprzęt komputerowy, oprogramowanie, okablowanie sieciowe, urządzenia peryferyjne, systemy bazodanowe, systemy operacyjne, komponenty sieciowe, repozytoria metadanych, aplikacje użytkowników itp.;
- 2) infrastrukturę nietechniczną: standardy metadanych, standardy eksploracji danych, logiczny model BD, metodyki, podręczniki, procedury testowe, procedury kontroli zmian, procedury zarządzania zmianami itp.

### *Krok 3 — Projekt systemu*

Istotne jest, aby projekt był możliwie dynamiczny i dostosowany do możliwości organizacyjnych, personalnych, budżetowych, technologicznych, produkcyjno-handlowych i kooperacyjnych firmy. Oprócz tego projekt musi być szczegółowy, kompletny i aktualny w stosunku do zmian, które na bieżąco pojawiają się w przedsiębiorstwie.

## **Etap III — Analiza**

### *Krok 4 — Analiza potrzeb biznesowych*

Dostosowanie projektu, z jednej strony, do potrzeb, a z drugiej — do możliwości organizacyjnych i finansowych przedsiębiorstwa jest jednym z najtrudniejszych aspektów projektowania i wdrażania BI. Trudno jest ograniczyć naturalne dążenie użytkownika do zbudowania systemu, który uwzględniłby od razu wszystkie jego potrzeby informacyjne, a ponadto, jak wynika z doświadczeń, takie kompleksowe

wdrożenia rzadko kończą się powodzeniem. Należy podzielić projekt wdrożenia BI na mniejsze etapy, które będą obsługiwały określone obszary tematyczne, a jednocześnie każdy kolejny etap wdrożenia będzie stanowił element nauki i dla projektantów, i dla użytkowników systemu.

#### *Krok 5 — Analiza danych źródłowych*

Dużym wyzwaniem dla każdego projektu wdrożenia BI jest jakość danych źródłowych. Zła jakość danych, problemy z ich integracją, trudności w uzyskaniu dostępu do baz źródłowych — wszystko to powoduje wydłużenie czasu poświęconego na ten etap projektowania i często również wzrost kosztów całego projektu. Dane, które będą zasilaly BI, decydują o jego efektywności i użyteczności dla klienta. Dlatego należy przydzielić na ten etap możliwie najwięcej czasu i najwięcej uwagi zespołu projektowo-wdrożeniowego.

#### *Krok 6 — Prototyp aplikacji*

System analityczny, który będzie funkcjonował w ramach projektu, można z powodzeniem wykonać metodą prototypowania, ponieważ stanowi on kombinację gotowych aplikacji i narzędzi informatycznych. Prototyp pozwala łatwiej porozumieć się projektantom i użytkownikom oraz szybciej uzgodnić ostateczną formę i funkcjonalność systemu analitycznego.

#### *Krok 7 — Analiza repozytoriów metadanych*

Zwiększenie liczby narzędzi informatycznych wykorzystywanych w systemie oznacza zwiększenie liczby metadanych technicznych w stosunku do metadanych biznesowych, które są obsługiwane przez te narzędzia. Metadane techniczne muszą być zmapowane na metadane biznesowe, a następnie przesłane do repozytorium metadanych. Wszystkie metadane, które będą wprowadzane i przechowywane w repozytorium, muszą zostać udokumentowane w logicznym modelu metadanych. Dodatkowo wszystkie wymagania dotyczące metadanych powinny być dostępne dla użytkownika poprzez funkcję pomocy.

### **Etap IV — Projektowanie**

#### *Krok 8 — Projekt BD*

Bazy źródłowe dla systemu informacji zarządczej mogą mieć różną strukturę, mogą nie uwzględniać wielowymiarowości, a także nie odpowiadać rzeczywistym potrzebom informacyjnym użytkowników systemu. Projekt struktury BD na potrzeby HD musi, uwzględniając dostępne zasoby danych źródłowych, zorganizować je w taki sposób, aby zaspokajały wymagania informacyjne wszystkich modułów analityczno-raportujących projektowanego systemu.

#### *Krok 9 — Projekt procesu ETL*

Proces ETL to proces zasilania HD, obejmujący: pozyskanie danych ze źródeł, przygotowanie ich do umieszczenia w hurtowni i wsadowe załadowanie do struktur HD; zwany jest również procesem integracji danych i jest najtrudniejszym elementem całego projektu. Jednocześnie w znacznym stopniu decyduje o sukcesie wdrożenia

i użyteczności biznesowej całego projektu. Na stopień złożoności tego procesu mają wpływ takie elementy, jak:

- liczba i jakość źródeł danych zasilających bazę systemu,
- dostęp do danych źródłowych,
- zakres informacyjny objęty przez projektowany system,
- liczba i stopień złożoności przygotowywanych raportów,
- wymagania użytkowników systemu (liczba i stopień złożoności raportów *ad hoc*).

Dobrze zaprojektowany i zrealizowany proces ETL może w zamian ukazać nieznaną dotąd moc informacyjną gromadzonych przez przedsiębiorstwo danych.

#### *Krok 10 — Projekt repozytorium metadanych*

Jeżeli decydujemy się na zakup gotowego produktu, to projekt repozytorium metadanych polega jedynie na uwzględnieniu (rozszerzeniu) tych cech logicznego metamodelu, które są niezbędne do realizacji projektowanego systemu, a nie zostały przewidziane przez licencjodawcę. W przypadku dedykowanego repozytorium metadanych projekt musi odpowiadać wymaganiom metamodelu logicznego, który jest oparty bądź na strukturze relacyjnej, bądź na strukturze obiektowej.

### **Etap V — Realizacja**

#### *Krok 11 — Wdrożenie procesu ETL*

Wiele narzędzi dostępnych na rynku jest przeznaczonych do wdrożenia procesu ETL. W zależności od stopnia złożoności procesu czyszczenia i transformacji danych (krok 5) oraz analizy danych (krok 9) różne narzędzia będą bardziej lub mniej przydatne w konkretnym przypadku. Niezbędne jest szczegółowe określenie wymagań i złożoności realizacji ww. procesów, aby prawidłowo dobrać rozwiązanie ETL.

#### *Krok 12 — Realizacja aplikacji*

Jeśli tylko tworzony w kroku 6 prototyp aplikacji spełni funkcjonalne wymogi systemu, można rozpocząć pracę nad rozwojem środowiska dostępu i analizy. Budowana aplikacja może być jedynie prostym przekształceniem prototypu w pełni funkcjonalny system albo znacznie bardziej złożonym nowym projektem, opartym nawet na zupełnie innych narzędziach informatycznych. W obydwu jednak przypadkach ten etap przebiega równoległe z wdrożeniem procesu ETL.

#### *Krok 13 — Eksploracja danych*

Wiele przedsiębiorstw stosujących BI nie wykorzystuje pełnych możliwości ich środowiska analitycznego, ograniczając się jedynie do zbioru predefiniowanych raportów — najczęściej wybieranych przez użytkowników. Tymczasem prawdziwa „siła systemu” i maksymalny zwrot inwestycji tkwi w informacjach ukrytych w danych przedsiębiorstwa. Nie są one widoczne w grupie raportów standardowych, ale są możliwe do pozyskania, jeżeli skorzysta się z metod zaawansowanej analizy wielowymiarowej, np. eksploracji danych. Wiąże się to oczywiście z ko-

niecznością zakupu i zastosowania narzędzi dedykowanych analizom eksploatacyjnym.

#### *Krok 14 — Realizacja repozytorium metadanych*

Jeżeli repozytorium metadanych jest tworzone dla systemu, a nie jest kupowany gotowy produkt, to do jego przygotowania zazwyczaj jest powoływany oddzielny zespół projektowy. Jest to uzasadnione tym, że projekt rozwoju repozytorium stanowi na ogół obszerny subprojekt, który musi uwzględnić: strategiczne cele przedsiębiorstwa, uwarunkowania biznesowe i rynkowe, przewidywane działania konkurencji i wiele innych aspektów rozwoju działalności przedsiębiorstwa determinujących rozwój BI, a więc i repozytorium metadanych.

### **Etap VI — Wdrożenie**

#### *Krok 15 — Szkolenie*

Po zakończeniu testów aplikacji BI należy rozpocząć intensywne szkolenie wszystkich osób, które będą korzystały z aplikacji bądź metadanych systemu. Szkolenia powinny dotyczyć: obsługi aplikacji, utrzymania i rozwoju BD, harmonogramowania i uruchamiania procesów ETL, planowania i kontroli skalowalności BD.

#### *Krok 16 — Ocena wdrożenia*

Bardzo ważne jest, żeby przy kolejnych etapach realizacji systemu korzystać z doświadczeń zdobytych w poprzednich etapach wdrażania. Jeżeli nastąpiło znaczące przekroczenie budżetu lub terminu realizacji projektu, to przyczyny powinny być szczegółowo przeanalizowane, aby nie powtórzyły się w kolejnych etapach. Jakiegokolwiek narzędzia, metody i praktyki, które nie przydały się lub przeszkadzały przy prawidłowej realizacji projektu, powinny zostać wyeliminowane przed rozpoczęciem kolejnego wdrożenia.

Niektóre etapy wdrożenia mogą być wykonywanych równolegle, co pozwoli na skrócenie czasu realizacji. Czas ten ulega też skróceniu dzięki rosnącemu doświadczeniu zespołu projektowo-wdrożeniowego. Są to elementy, na których można budować sukces i rosnącą efektywność kolejnych wdrożeń.

Projekt BI musi być rozwijany przez cały okres jego eksploatacji i — aby z powodzeniem wspierał proces zarządzania przedsiębiorstwem — powinien:

- obejmować cały cykl życia informacji, tj. pozyskiwanie, składowanie, czyszczenie, integrację i analizę;
- wykorzystywać nie tylko technologię do zarządzania informacją, ale także cały proces zarządzania nią;
- utrzymywać wysoką jakość danych, bezpieczny dostęp do nich oraz jasne reguły odpowiedzialności za te dane.

W praktyce biznesowej systemy BI są wykorzystywane w następujących celach [Olszak, Ziemia 2003]: możliwości swobodnej i elastycznej realizacji złożonych analiz, poprawy sprawności operacyjnej, zwiększenia efektywności zarządzania

oraz tworzenia i usprawniania relacji z klientami. Obszarami zarządzania, które mogą wykorzystywać rozwiązania BI, są [Olszak, Ziemba 2003]:

- produkcja — np. w zakresie: analizy zdolności produkcyjnych, analizy efektywności maszyn i urządzeń, wspomaganie elastycznej gospodarki materiałowej;
- logistyka — np. w zakresie: analizy efektywności przemieszczania, analizy procesów zaopatrzenia, wspomaganie optymalizacji wyboru dostawców i warunków dostaw;
- marketing — np. w zakresie: wspomaganie projektowania kampanii marketingowych, a następnie analizy ich efektywności, wspomaganie projektowania kampanii promocyjnych i programów lojalnościowych;
- sprzedaż — np. w zakresie: wspomaganie projektowania rozmieszczenia towarów czy tworzenia pakietów usług, analizy rynku i przygotowania kampanii wprowadzenia nowych towarów i usług, analizy sprzedaży i sezonowości sprzedaży, wspierania działań reprezentantów handlowych;
- obsługa klienta — np. w zakresie: analizy zachowań i potrzeb klienta, analizy rentowności i lojalności klienta, przygotowywanie oferty dedykowanej;
- finanse i kontroling — wspieranie zarządzania kosztami oraz badanie efektywności tego zarządzania, analiza obecnej kondycji finansowej oraz przygotowywanie prognoz, wspieranie przygotowywania budżetu, analiza efektywności inwestycji;
- kadry i płace — zarządzanie procesem szkoleń, planowanie rozwoju i naboru kadry, analiza efektywności działań pracowników.

Systemy BI stają się obecnie integralną częścią rozwiązań informatycznych wykorzystywanych w przedsiębiorstwach. W kompleksowych systemach klasy ERP czy CRM rozwiązania BI stanowią obowiązkowy element struktury rozwiązania. Bogata oferta rynkowa narzędzi BI (również z grupy opensource) coraz częściej pozwala z nich korzystać nie tylko dużym korporacjom, lecz także przedsiębiorstwom małym i średnim.

## 10.2. System ekspertowy

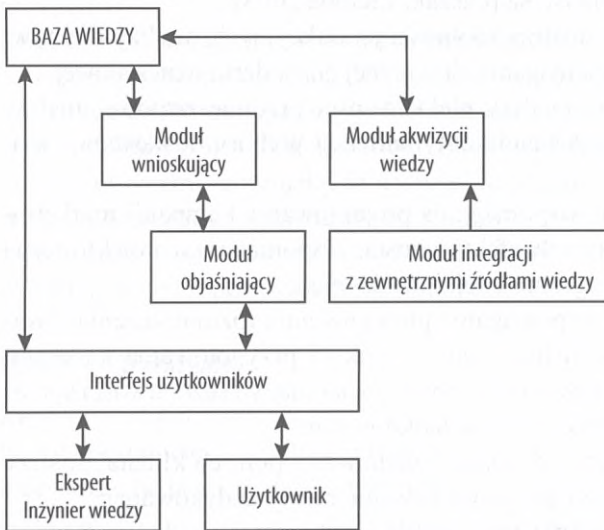
Zasadniczym celem tworzenia SE jest skomputeryzowane rozwiązywanie problemów na poziomie eksperta wysokiej klasy.

Wiedza z SE powinna być łatwo dostępna i powinna umożliwiać personelowi o niewielkich umiejętnościach wykonywanie powierzonych zadań (po konsultacji z SE), które w przeciwnym razie musiałyby być rozwiązywane przez rzeczoznawców.

### 10.2.1. Struktura i istota działania SE

Architekturę systemu ekspertowego przedstawiono na rysunku 10.5. Zawiera on takie moduły, jak:

Rysunek 10.5. Architektura systemu ekspertowego



Źródło: Opracowanie własne.

- podsystem akwizycji wiedzy spełniający funkcje: gromadzenia, przesyłania i przetwarzania wiedzy eksperta-rzeczoznawcy, pochodzącej bezpośrednio od eksperta bądź z udokumentowanych źródeł wiedzy zawierających zapis ekspertryzy w języku programu komputerowego w celu stworzenia lub poszerzenia bazy wiedzy;
- baza wiedzy (BW);
- maszyna wnioskująca;
- użytkownik;
- interfejs użytkownika;
- obszar pracy;
- podsystem wyjaśniająco-uzasadniający;
- system aktualizacji i uściślenia wiedzy.

W BW znajdują się informacje o wybranej dziedzinie wiedzy i stanach środowiska, które zostały przekazane przez eksperta. BW tworzy inżynier wiedzy. Musi ona być zapisana w postaci sformalizowanej, zrozumiałej dla maszyny wnioskującej. Wiedza w BW zapisana jest za pomocą symboli. Dzięki temu łatwo można prześledzić sposób dochodzenia systemu do rozwiązania oraz przeanalizować poprawność BW. Jest to duża zaleta w porównaniu z niesymbolicznymi metodami AI, gdzie przeanalizowanie drogi wnioskowania jest bardzo utrudnione i z tego powodu często niezrozumiałe i nieakceptowane przez użytkownika SE [Mulawka 1997; Sroka, Wolny 2009].

Bardzo ważnym procesem w projektowaniu i tworzeniu SE jest proces weryfikacji BW, mający na celu zapewnienie, że tworzona BW jest wolna od typowych błędów i anomalii. Walidacja z kolei jest równie ważnym procesem mającym na celu wykazanie, że SWD jest wystarczająco kompetentny. Podstawową techniką weryfikacji jest wykrywanie błędów i anomalii, realizowane poprzez formalną analizę struktury BW. Wykrywane błędy i anomalie są najczęściej dziedzinowo niezależne, a dodatkowo, uwarunkowane zastosowaniem wytyczne i ograniczenia stanowią metawiedzę dla weryfikacji. Podstawową techniką walidacji jest testowanie działającego systemu, ukierunkowane na wykazanie zgodności wyników jego działania z rezultatami osiąganymi przez ekspertów dziedzinowych. W procesie walidacji znajdują zastosowanie techniki testowania znane z inżynierii oprogramowania, a przedstawione bliżej w rozdziale 7.

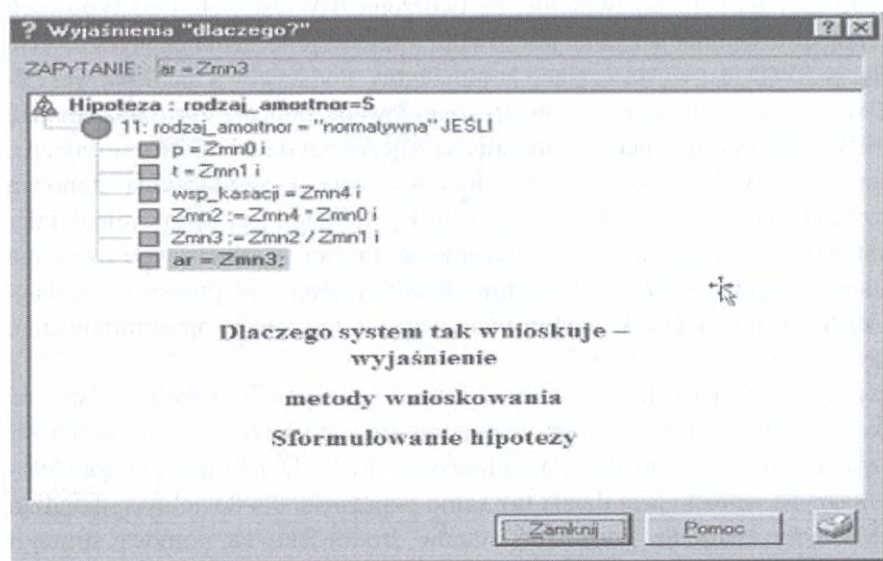
Maszyna wnioskująca służy do wyszukiwania rozwiązań w przestrzeni stanów środowiska. Jednym z podstawowych postulatów tworzenia tzw. szkieletowych SE jest to, aby maszyna wnioskująca była oddzielona od BW. Dzięki temu w szkieletowych SE maszyna wnioskująca działa tak samo poprawnie dla dowolnych dziedzin wiedzy. Maszyna analizuje przestrzeń stanów środowiska za pomocą strategii przeszukiwań i heurystyk, stosując ustalone dla niej metody wnioskowania. Strategie poszukiwań określają, w jaki sposób maszyna wnioskująca ma sprawdzać prawdziwość kolejnych stanów środowiska. Heurystyki pomagają zoptymalizować poszukiwania rozwiązań i pominąć ścieżki, które nie rokują nadziei na odnalezienie rozwiązania. Dzięki temu zostaje skrócony czas dochodzenia do wyniku. Metody wnioskowania decydują, w jaki sposób zachodzi proces myślenia, czy jest to np. wnioskowanie wstecz (indukcja), wprzód (dedukcja) lub inne.

Użytkownik jest najważniejszym podmiotem SE i bez uwzględnienia jego preferencji, specyfiki i oczekiwań nie da się zbudować i doskonalić SE. Interfejs użytkownika umożliwi komunikację z SE; pozwala generować wynik, a także przedstawia objaśnienia do procesu wnioskowania. Możliwości komunikacyjne interfejsu są uzależnione od rodzaju systemu operacyjnego, pod jakim działa SE, i narzędzi programowania, za pomocą których zaprojektowano SE.

Obszar pracy to obszar ekranu, w którym użytkownik może dokonywać modyfikacji stanów środowiska SE, np. pole dialogowe, pola wyboru itp. Obszar ten powinien być zaprojektowany według zasad opisanych w rozdziale 3.

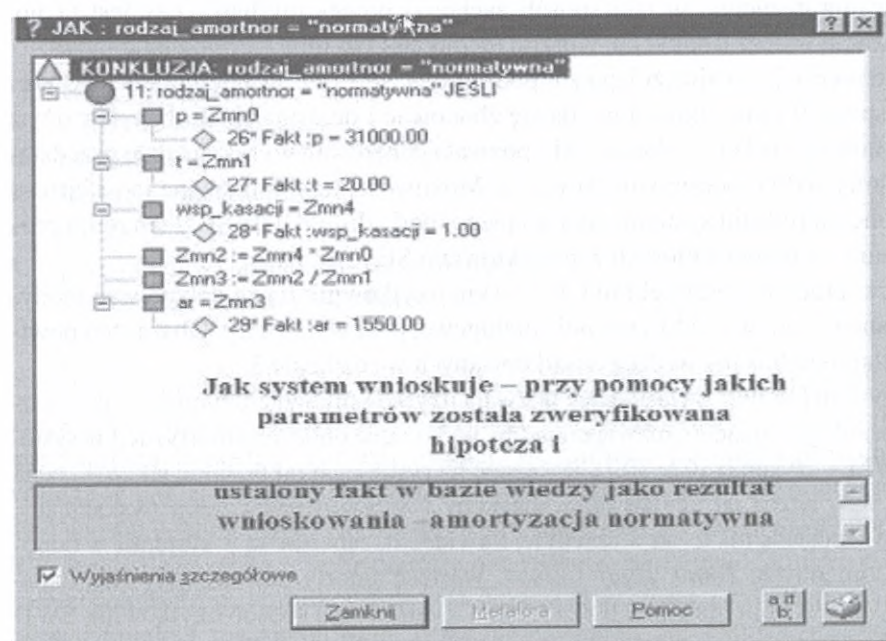
Podsystem (moduł) wyjaśniający pozwala użytkownikowi zrozumieć, w jaki sposób SE poszukuje i znajduje rozwiązania (np. wykonanie obliczeń amortyzacji w systemie MOWM — rysunki 10.6–10.7). Pokazane na rysunkach fakty, takie jak:  $p$  — cena maszyny,  $t$  — liczba lat użytkowania maszyny i *współczynnik kasacji* (wskazujący, jaką wartość posiada maszyna w chwili jej likwidacji), zapisane są w systemie w odpowiednich zmiennych:  $Zmn0$ ,  $Zmn1$  i  $Zmn4$ . Wartość amortyzacji normatywnej oznaczona jako  $ar$  jest obliczana jako iloraz  $Zmn2$  i  $Zmn1$ . W ten sposób użytkownik SWD może śledzić przebieg procesu przetwarzania zmiennych modelu swojego gospodarstwa rolnego i uzyskać odpowiedź na pytanie, jak dana wielkość została obliczona.

Rysunek 10.6. Objaśnienie metody obliczenia amortyzacji ciągnika rolniczego w SE MOWM



Źródło: Opracowanie własne.

Rysunek 10.7. Objaśnienie szczegółowe sposobu obliczenia amortyzacji ciągnika rolniczego w SE MOWM



Źródło: Opracowanie własne.



System aktualizacji i uściślenia wiedzy jest odpowiednią techniką przeprowadzenia procesu walidacji SE. Z konstrukcyjnego punktu widzenia istnieje potrzeba posiadania danych o systemie rzeczywistym, które pozwolą budować teorie szczegółowe, użyteczne w budowie modeli i testowaniu ich założeń. Drugi typ danych zawiera dane systemu rzeczywistego (wejście, wyjście), które są wykorzystywane w procesie operacyjnej walidacji modelu, polegającej na porównaniu zachowania się systemu rzeczywistego z zachowaniami modelu. Jeżeli brak jest danych o systemie rzeczywistym, to zazwyczaj nie jest możliwa do uzyskania wysoka wiarygodność modelu. Problem wiarygodności danych odnosi się do ich odpowiedniości (adekwatności), dokładności, spójności oraz kompletności. Ponadto, jeżeli dokonywana jest transformacja danych systemu rzeczywistego (np. agregacja, dezagregacja), to procedura ta powinna również podlegać sprawdzeniu jej poprawności.

### 10.2.2. Projektowanie SE

W procesie projektowania SE należy posługiwać się sprawdzonymi standardami postępowania w zakresie dotyczącym konkretyzacji rozwiązań projektowych SWD, wykorzystując metody i narzędzia projektowania SWD, co zostało omówione szerzej w rozdziałach 5 i 6. Opracowując koncepcję systemu, należy najpierw zdefiniować problem decyzyjny i cechy go opisujące oraz ustalić relacje między nimi. Następnie konieczne jest ustalenie dziedziny zainteresowań oraz składu zespołu projektowego, który powinni tworzyć: inżynier wiedzy, ekspert, personel pomocniczy i oczywiście przedstawiciele użytkowników. Kolejnym etapem procesu projektowania jest ustalenie strategii budowy systemu w oparciu o zasoby własne lub outsourcing i wybranie strategii projektowania, np. przyrostowej, kaskadowej lub spiralnej. Należy uwzględnić uwarunkowania rozwoju architektury SE, jak również sposób komunikacji poszczególnych modułów i zadania, które mają one pełnić w oparciu o dokonaną wraz z użytkownikami SE specyfikację funkcjonalności i weryfikację niesprzeczności modelu architektury SWD.

Kolejnym etapem jest ustalenie źródeł wiedzy w postaci zbiorów wiedzy udokumentowanej, jak również wiedzy ekspertów oraz przy współpracy z inżynierami wiedzy dokonanie formalizacji wiedzy na język programowania, np. FNDL, aby opracować BW. Inne przydatne w procesie formalizacji języki i narzędzia to m.in. języki sztucznej inteligencji (PROLOG i LISP) oraz specjalizowane języki i środowiska realizacyjne (np. OPS5, CLISP czy Level 5). Wyróżnia się pięć faz projektowania BW: fazę identyfikacji problemu, fazę projektu koncepcyjnego, fazę formalizacji reprezentacji wiedzy, fazę implementacji BW i fazę sprawdzenia projektu.

Następnie należy wybrać odpowiedni język programowania lub pakiety narzędziowe, np. pakiet AITECH-CAKE, który jest komputerowym systemem wspomaganiania inżynierii wiedzy. Jest on przeznaczony do komputerowego wspomaganiania procesu realizacji dziedzinowych aplikacji szkieletowego systemu ekspertowego PC-Shell. Dzięki wykorzystaniu systemu CAKE można realizować aplikacje sys-

temu PC-Shell bez dokładnej znajomości języka opisu bazy wiedzy. Na każdym etapie pracy system oferuje wygodne narzędzia wspomagające, eliminujące konieczność żmudnego wprowadzania kodu. Zapis BW w postaci binarnej zapewnia, z jednej strony, ochronę zgromadzonej wiedzy przed niepowołanym dostępem, z drugiej zaś — poprawia efektywność wykonania aplikacji w środowisku systemu PC-Shell. Dzięki systemowi uprawnień i haseł można ograniczyć dostęp do aplikacji zarówno na etapie jej tworzenia, jak i na etapie wykonywania. Edytor BW zawiera edytor bloku atrybutów, edytor bloku faktów, bloku uprawnień i bloku sterującego. Następnie translator opisu języka BW pozwala tłumaczyć dane z wymienionych edytorów do bazy postaci tekstowej. Takie właściwości opisywanego pakietu zapewniają efektywny technicznie i ekonomicznie sposób projektowania, uwzględniający uwarunkowania wdrożeniowo-eksploatacyjne, co pozwala zrealizować projekt w postaci funkcjonującej aplikacji, np. dla potrzeb analizy finansowej procesów gospodarczych [Michalik 1999].

Konieczne jest zatem opracowanie założeń SE w odniesieniu do konkretnego sprzętu, systemu operacyjnego, języka programowania, narzędzi wspomagających, jak również strategii rozwiązywania problemów, baz wiedzy, podziału pracy i diagnostyki. W przedostatniej fazie projektowania SE niezbędne jest również przygotowanie studium wykonalności projektu poprzez ustalenie i zweryfikowanie wielu wskaźników ekonomicznych, technicznych i użytkowych uzasadniających te działania. Etap prototypowania powinien polegać na cyklicznym procesie doskonalenia projektu za pomocą technologii zwinnych, począwszy od fazy potrzeb użytkownika, poprzez tworzenie prototypu i jego ocenę, a skończywszy na powtórnej konfrontacji z potrzebami użytkownika systemu, prowadzonymi cyklicznie do momentu uzyskania satysfakcji przez użytkownika. Na tym etapie rozwoju dokonuje się modernizacja i sprawdzenie zmienionej architektury; następuje hierarchiczne porządkowanie BW, metawiedzy i określane są wyjątki.

Na etapie eksploatacji i użytkowania systemu korygowane są błędy i wprowadzane ulepszenia oraz rozbudowywana jest BW, która lepiej pomaga weryfikować wiarygodność wprowadzanych do niej faktów.

Ze względu na wykorzystywanie SE do rozwiązywania problemów trudnych jednym z kluczowych wymagań konstrukcyjnych jest pewność ich działania. Wymaganie to jest szczególnie ważne wtedy, gdy niesprawność SE stanowi zagrożenie dla życia ludzkiego. Dlatego konieczne jest przeprowadzenie oceny SE z wykorzystaniem zbioru różnych wskaźników niezawodnościowych.

Panuje opinia, że system działa poprawnie wtedy, kiedy po wykryciu błędu może być sprowadzony do stanu zapobiegającego rozprzestrzenianiu się skutków błędu. Niezawodność działania SE wynika z niezawodności działania sprzętu i oprogramowania. W tym celu wykorzystuje się sprawdzone w ramach „klasycznych rozwiązań” metody unikania potencjalnych błędów poprzez takie działania, jak:

- stosowanie najnowszych osiągnięć nauki,
- posługiwanie się sprawnymi i sprawdzonymi narzędziami programowymi,

- tworzenie strukturalnego oprogramowania,
- stosowanie specjalnych rozwiązań układowych,
- tworzenie programów nadmiarowych,
- tworzenie  $n$  wersji programu,
- nadzorowanie działania SE przez niezależny system zabezpieczający, którym również może być SE<sup>2</sup>.

Do mechanizmów doskonalących działanie SE można zaliczyć m.in.:

- kontroler niesprzeczności nowo wprowadzanych do bazy wiedzy reguł z regułami w niej zawartymi;
- kontroler zgodności reguł z nowo wprowadzanymi faktami;
- mechanizm oceny częstości stosowania poszczególnych reguł;
- mechanizm rozbudowy istniejącej bazy reguł poza zakres danej bazy.

Dwa pierwsze elementy mieszczą się w module niesprzeczności, pozostałe należy wbudować jako dodatkową strukturę uczącą. Adaptacyjność nie jest wymagana w przypadku zamkniętych dziedzin wiedzy, będzie natomiast konieczna w systemach dla tych dziedzin wiedzy, w których występuje duży stopień niepewności i niedostatecznie jest jeszcze rozwinięta BW.

### 10.2.3. Przykłady zastosowań SE

System doradczy Ma'ayan, zbudowany ze względu na wspomaganie decyzji w zarządzaniu przedsiębiorstwami rolnymi, wykorzystuje nieskomplikowany i przyjazny dla użytkownika model zarządzania. Jego zaletą jest eliminowanie modeli, które wymagają wyższego stopnia zaawansowania wiedzy z zakresu zarządzania. Jest prosty i łatwy do zrozumienia, ponieważ zawiera raporty spełniające oczekiwania rolników. System ten redukuje czas zastosowania modelu, głównie dlatego, że rolnik może utrzymywać swój tradycyjny sposób rejestracji danych wejściowych oraz format ich zapisu. Inną bezsprzeczną zaletą jest możliwość pozyskiwania zróżnicowanych raportów na podstawie przeszłych i nowych rekordów. Zaleca się, aby rozpocząć pracę od obsługi modułu rejestracji nakładów pracy, a następnie, wraz z coraz lepszym poznawaniem systemu, dodawać nowe moduły. Wcześniejsze doświadczenia z wdrożeń opisywanego systemu wskazały, że 20% całkowitego nakładu czasu wystarcza do zgromadzenia 80% potrzebnych informacji do wspomaganie decyzji i prowadzenia ewidencji, co jest zgodne z zasadą Pareto. Doświadczenia z pracy z systemem Ma'ayan wskazują, że 80% skomputeryzowanej informacji wystarcza<sup>3</sup>, aby zaspokoić podstawowe potrzeby informacyjne rolnika.

<sup>2</sup> Zob. [www.nt.if.pwr.wroc.pl/up/przewodnikspd/SPD6\\_3.htm](http://www.nt.if.pwr.wroc.pl/up/przewodnikspd/SPD6_3.htm).

<sup>3</sup> Zob. [www.agriculture.gov.ie/farmerschemespayments/crosscompliance/farmadvisorysystem](http://www.agriculture.gov.ie/farmerschemespayments/crosscompliance/farmadvisorysystem).

Program komputerowy zastosowany w systemie doradczym Ma'ayan jest elastyczny i nie jest zdefiniowany dla danej uprawy roślinnej lub struktury zasiewów, co często powoduje nieelastyczność i nieadekwatność narzędzi ICT do zaspokajanych potrzeb informacyjnych. Raporty są sporządzane na podstawie bieżących danych i adekwatnych warunków dzięki prostocie i szybkości metody rejestrowania i przetwarzania danych, natomiast obsługa skomplikowanego oprogramowania zwykle wymaga szkolenia i wsparcia technicznego. System pomocy w Ma'ayan jest uproszczony, tzn. że działa na życzenie użytkownika za pośrednictwem telefonu lub poczty elektronicznej.

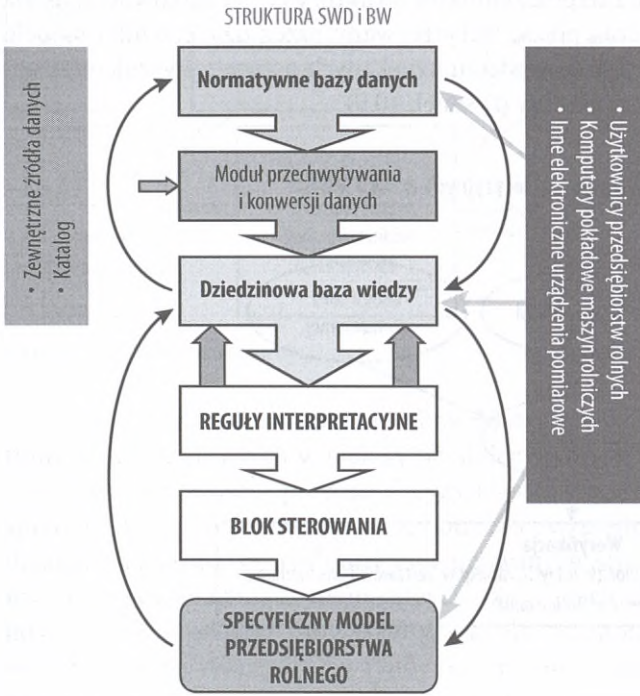
Trudności w adaptacji i wdrożeniu mogą wynikać z faktu, że rolnicy już wykorzystują inne narzędzia komputerowe, np. MS Excel, lub z osobistych preferencji lub uwarunkowań (np. część użytkowników uważa, że Ma'ayan nie jest dość przyjazny i nie zaspokaja ich specyficznych wymagań). Kolejną przyczyną może być również rozczarowanie rolników innymi narzędziami wspomagania decyzji lub słaby stopień rozpropagowania możliwości systemu na rynku. Na podstawie obserwacji można stwierdzić także, że rolnicy niewspółpracujący z doradcami rolnymi nie wykorzystują specjalistycznego oprogramowania w ogóle, a zwłaszcza systemu Ma'ayan. Takich rolników jest jednak niewiele.

Zaproponowana na podstawie badań metoda klasy SWD [Bojar 2005] obejmuje BD i dziedzinową BW (rysunek 10.8), pozwalając gromadzić, weryfikować i przetwarzać dane wejściowe modelu w taki sposób, aby uzyskać raporty przydatne w analizie decyzji zarządzających. Koniecznym warunkiem wykorzystania odpowiednich algorytmów przetwarzania jest uzyskanie precyzyjnych i adekwatnych danych wejściowych. Dotyczy to zarówno specyficznych danych obiektowych analizowanego przedsiębiorstwa, jak i danych normatywnych niezbędnych do znalezienia konkretnego problemu decyzyjnego. Ponieważ przedsiębiorstwa rolne w kraju i za granicą nie posiadają wystarczająco szczegółowych systemów informacyjnych, zaproponowano rozwiązanie uwidocznione na rysunku 10.8.

Normatywne BD za pośrednictwem modułu przechwytywania i konwersji danych są zasilane ze źródeł danych zewnętrznych i wewnętrznych, stanowiąc źródło danych dla dziedzinowej BW. Baza ta jest poszerzana i weryfikowana poprzez zastosowanie reguł interpretacyjnych i funkcje bloku sterowania programem, stając się jednocześnie źródłem danych dla specyficznego modelu przedsiębiorstwa rolnego, które z kolei w sprzężeniu zwrotnym służy wzbogacaniu dziedzinowej BW i normatywnych BD. W ten sposób opracowany SWD może pełnić funkcje uzupełniania i poszerzania systemu informacyjnego przedsiębiorstwa rolnego, uwzględniając także sprzęt zmechanizowany nie zawsze zapewniający automatyczny pomiar nakładów (czasu, materiału, powierzchni).

Przedstawiony model wspomagania decyzji (rysunek 10.8) wykorzystuje dane wejściowe znajdujące się w BD oraz BW. Uaktualniana i weryfikowana BW dostarcza parametry eksploatacyjno-techniczne i ekonomiczne niezbędne do oceny wyboru maszyn. Zgromadzone w niej fakty są interpretowane na podstawie reguł

**Rysunek 10.8. Strukturalne elementy SWD dla potrzeb tworzenia modeli przedsiębiorstw rolnych wspomagających decyzje dotyczące wyboru maszyn**



Źródło: Opracowanie własne.

wnioskowania, w których wykorzystano wyniki badań własnych oraz innych ośrodków zajmujących się problematyką optymalnego doboru maszyn.

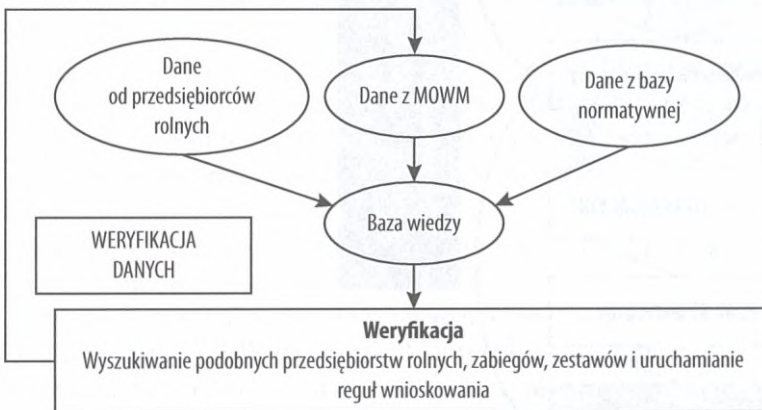
Ustalenie wiarygodnych danych modeli przedsiębiorstw rolnych jest związane z działaniem odpowiednich reguł interpretacyjnych (wnioskowania) umożliwiających nie tylko weryfikację parametrów modelu, ale także działanie algorytmów ich przetwarzania.

W opracowanej Metodzie Oceany Wykorzystania Maszyn (MOWM) [Bojar 2005] można wyróżnić następujące grupy reguł:

- wyboru i przechwytywania danych z bazy wiedzy, np. jeśli zabieg = orka, to system wybiera zestaw do orki (np. ciągnik360 + pługU023/1 + gospodarz);
- wariantowania obliczanych kosztów, np. jeśli roczny czas wykorzystania jest krótszy od czasu normatywnego, to system oblicza amortyzację metodą normatywną (użytkownik może wybrać metodę degresywną lub czasową);
- weryfikacji poprawności tworzenia zestawów, np. jeśli zapotrzebowanie maszyny na moc ciągnika jest dużo niższe od jego mocy, to system generuje stosowany komunikat o nieekonomicznym wykorzystaniu mocy w zestawie;

- obliczania nakładów czasu wykonania zabiegu agrotechnicznego, np. jeśli niezbędnym zabiegiem jest orka, to system sprawdza wydajność eksploatacyjną maszyny wiodącej w zestawie wybranym do wykonania tej orki, a następnie na podstawie wielkości pola i współczynników struktury czasu całkowitego ustala orientacyjny całkowity czas pracy, weryfikowany przez użytkownika modelu oraz na podstawie podpowiedzi systemu uzyskanych poprzez wyszukanie adekwatnych wartości w bazie wiedzy (rysunek 10.9).

Rysunek 10.9. Moduł weryfikacji parametrów decyzyjnych w SWD MOWM



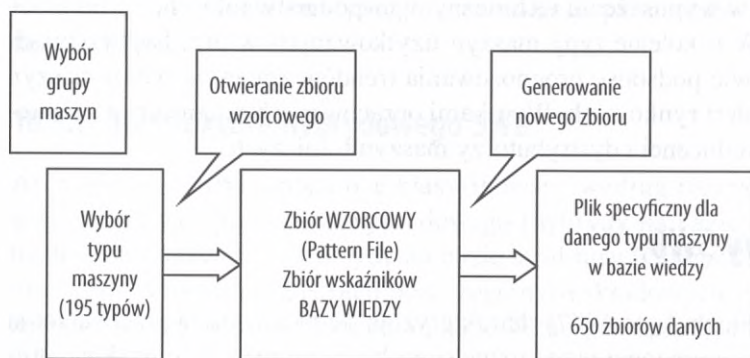
Źródło: Opracowanie własne.

Tworzenie zbiorów danych o nowych środkach mechanizacji wprowadzanych do systemu następuje metodą analogii poprzez użytkownika-eksperta. Polega to na generowaniu nowego zbioru specyficznego typu maszyny występującego w przedsiębiorstwie rolnym na podstawie istniejącego zbioru wzorcowego (rysunek 10.10).

Doświadczenia badawcze w obszarze tworzenia SWD ze względu na potrzeby optymalnego wyboru maszyn w przedsiębiorstwach rolnych wskazują, że trudno jest znaleźć źródła danych, które zarówno w wymiarze ilościowym, jak i jakościowym byłyby wystarczające do osiągnięcia celów SWD MOWM [Bojar 2005]. Dlatego też założono, że w miarę poszerzania baz danych i bazy wiedzy utworzonej na potrzeby opracowanego systemu niezbędne jest zastosowanie automatycznych procedur weryfikacyjnych. Procedury te pozwalają tworzyć z dużym prawdopodobieństwem wiarygodności specyficzne dane modeli przedsiębiorstw rolnych na podstawie wyszukanych, porównywalnych parametrów dotyczących realizowanych zabiegów technologicznych, które generują określone nakłady i związane z nimi koszty, będące podstawą oceny trafności analizowanych decyzji (rysunek 10.10). W miarę powiększania zasobów bazy wiedzy i baz danych systemu MOWM proces uściślenia danych będzie coraz bardziej adekwatny do rozwiązywanych problemów decyzyjnych.

### Rysunek 10.10. Tworzenie zbiorów danych o środkach mechanizacji

— moduł akwizycji wiedzy



Źródło: Opracowanie własne.

Konieczność poszerzania zasobów systemu o kolejne modele przedsiębiorstw rolnych i analizowanych w nich sytuacji decyzyjnych uwarunkowana jest zakresem i wielowariantowością procesu decyzyjnego związanego z problematyką wyboru sprzętu rolniczego. Składa się na to potrzeba uwzględnienia w ramach prowadzonej działalności produkcji roślinnej zróżnicowanych struktur procesów technologicznych oraz powierzchni użytków rolnych, rzutujących na obszar prowadzonej działalności i skalę produkcji. To z kolei przekłada się na konieczność uwzględnienia na wejściu modelu różnej liczby typów ciągników, maszyn samobieżnych i środków transportu, jak również liczby i kwalifikacji operatorów maszyn oraz pracowników wykonujących zabiegi agrotechniczne.

Zaletą zaproponowanego rozwiązania jest obecność w MOWM modułu objaśniającego, który na każdym etapie analizy problemu decyzyjnego pozwala użytkownikowi śledzić metodę uzyskania określonego wyniku. Czyni to użytkownika w pełni świadomym i aktywnym podmiotem procesu wspomaganego decyzji oraz pozwala lepiej strukturalizować problem niż inne metody [Wójcicki 2003]. Ze względu na złożony charakter problemu optymalnego wyboru sprzętu ma to bardzo duże znaczenie z punktu widzenia praktyki zarządzania przedsiębiorstwami.

Opracowany SWD pozwala spełnić wymagania ekonomiczno-organizacyjnej oceny procesu wytwórczego, zastępując i ograniczając zakres niezbędnych zasobów informacyjnych przedsiębiorstw rolnych do jego funkcjonowania poprzez uzupełnienie informacji z BW. Opracowana metoda MOWM pozwala w skuteczny sposób uzyskiwać rozwiązania dla potrzeb wyboru maszyn, np. metody stosowane wcześniej. Wykorzystując dziedzinową BW, zmniejsza się pracochłonność modelowania przedsiębiorstw rolnych i uzyskiwania kolejnych wariantów wyników. Wygenerowane rozwiązania mogą być różnicowane w zależności od zmiany maszyny w zestawie, całego zestawu, procesu technologicznego, poszerzenia powierzchni danej działalności, wpływu zakupu lub świadczenia usług mechanizacyjnych, wspólnego

użytkowania maszyn itp. Prezentowane podejście pozwala także dobrze strukturalizować złożony problem wyboru maszyn. Może być wykorzystywane do oceny stanu i planowania zmian w wyposażeniu technicznym gospodarstw rolnych.

Poszerzanie BW o kolejne typy maszyn użytkowanych w przedsiębiorstwach rolnych może stanowić podstawę prognozowania trendów zmian na rynku maszyn rolniczych metodą ofert rynkowych. Wynikami opracowanych prognoz będą zainteresowani również producenci i dystrybutorzy maszyn rolniczych.

### 10.3. Hybrydowy SWD

Hybrydowy SWD [Finch, Lees 1997] charakteryzuje się możliwością przetwarzania wielu typów wiedzy, wyrażonej przez różne sposoby jej reprezentacji, wykorzystującej zróżnicowane metody wnioskowania i agregacji informacji, o odrębnych sposobach ich przedstawiania. Przesłanką projektowania i wdrażania struktur hybrydowych jest możliwość zestawiania w jednym systemie metod i analiz wzajemnie się uzupełniających, które dotychczas nie mogły występować razem, np. analiza danych ilościowych i jakościowych, wykorzystanie algorytmów genetycznych i metod wnioskowania, konekcyjność i regułowa reprezentacja wiedzy lub wykorzystanie symulacji i wizualizacji.

Już w opiniach formułowanych pod koniec lat 80. ubiegłego wieku pojawiają się takie, które wśród najważniejszych zastosowań SWD dostrzegają hybrydowość, mającą na celu wykorzystanie wielu technik i połączenie elementów z wielu technologii we wspólnym rozwiązaniu [O'Keefe 1988]. W połowie lat 90. m.in. J. Liebowitz [Liebowitz, Letsky 2000] wyrażał pogląd, że funkcjonujący samodzielnie SE jest rzadkością, natomiast powszechne stało się integrowanie SE z sieciami neuronowymi, analitycznymi SWD, narzędziami CASE (ang. *Computer Aided Software Engineering*) i multimediami.

Proponowana przez N. Matsatsinisa i A.P. Samarasa [Matsatsinis, Samaras 2000] hybryda systemu SWD (HSWD) zawiera moduł SE, który służy: wsparciu interpretacji rezultatów pochodzących z modeli matematycznych, pomocy przy formułowaniu podstawowych założeń leżących u podstaw zastosowania tych modeli oraz rekomendowaniu określonych wariantów decyzyjnych, z którymi spotyka się decydent.

Jak stwierdza A. Dutta [Dutta 1996]: „(..) hybrydowość SWD polega na pragmatycznej i skoordynowanej (równoległej, szeregowej, hierarchicznej czy prowadzonej poprzez komunikację międzymodułową) współpracy, współzawodnictwie w wykonywaniu poszczególnych zadań lub nadzorowaniu pracy modułów implementujących techniki sztucznej inteligencji, prognozowania, uczenia maszynowego, sieci neuronowych, algorytmów genetycznych, symulacji, regułowych systemów eksperckich (...)”.



HSWD zawsze charakteryzują się dużym stopniem złożoności funkcjonalnej, strukturalnej i metodologicznej. Stąd też występują częste trudności w ich konfigurowaniu i konieczność umiejętnej synchronizacji parametrów połączonych modeli decyzyjnych.

### 10.3.1. Architektura hybrydowego SWD

Architektury HSWD mogą być klasyfikowane według różnych kryteriów, np. ze względu na: typ połączenia hybrydowego (hybrydy na bazie funkcji, hybrydy na bazie komunikacji oraz hybrydy na bazie struktury [Goonatilake, Khebbal 1995]) oraz stopień i sposób integracji poszczególnych składowych systemu (modele niezależne, modele transformujące, modele luźno powiązane, modele zintegrowane [Medsker 1995]).

W przypadku hybrydy na bazie funkcji w modelu HSWD znajduje zastosowanie więcej niż jedna technika analityczna [Radosiński 1998]. Przykładowo, w sieci neuronowej procedura wyznaczania wag może przyjmować postać algorytmu genetycznego. Z kolei sieć neuronowa może być użyta do wyboru populacji początkowej w modelu opartym na algorytmie genetycznym.

Hybryda na bazie komunikacji wymaga wzajemnego połączenia co najmniej dwóch niezależnych technik analitycznych kanałami transmisji danych [Radosiński 1998]. Dzięki elastycznej wymianie danych struktury te są w stanie lepiej wykonywać zaplanowane zadania.

Hybryda na bazie struktury to system o jednolitej strukturze, który łączy w sobie cechy różnych technik inteligentnych [Radosiński 1998], np.: SE, algorytmów genetycznych, zbiorów rozmytych itp.

Niezależne systemy hybrydowe funkcjonalnie stanowią odrębne moduły, których wzajemna komunikacja i wymiana danych jest znacznie ograniczona [Medsker 1995]. Rozwiązania tego typu są tworzone w celach symulacyjnych i porównawczych.

System hybrydowy oparty na modelu transformującym wykorzystuje mechanizm, w którym przetworzone wyniki pracy jednego modułu są przekazywane do innego modułu jako jego dane wejściowe [Medsker 1995]. Rozwiązanie takie jest skuteczne w przypadkach, kiedy dane wejściowe jednego z modułów HSWD wymagają wcześniejszego ich przygotowania i przetworzenia.

W systemach hybrydowych o luźno powiązanej architekturze poszczególne moduły systemu komunikują się ze sobą za pośrednictwem plików [Medsker 1995]. Pliki przekazują wyniki pracy jednego modułu do innych modułów, wykorzystujących je jako dane wejściowe.

Natomiast przekazywanie danych między modułami w systemach hybrydowych o architekturze zintegrowanej odbywa się bezpośrednio w pamięci operacyjnej komputera [Medsker 1995]. W pełni zintegrowanym HSWD występuje bardzo silne

powiązanie każdego modułu z resztą systemu, a poszczególne moduły współdziałały między sobą zarówno wiedzę, jak i model oraz struktury danych [Medsker 1995].

### 10.3.2. Problemy projektowania i eksploatacji HSWD

Podstawowym celem wdrożenia HSWD jest wspieranie [Zabawa 2005]: diagnozy stanu bieżącego, poszukiwania optymalnych rozwiązań zdiagnozowanych problemów oraz korzystania z eksperckiej bazy wiedzy przedsiębiorstwa. Istotą systemu HSWD jest integracja wewnątrz systemu kilku różnych technik, o wzajemnie uzupełniających się funkcjonalnościach. Ta najsilniejsza zaleta systemu hybrydowego jest również jego największą wadą.

W opinii Z. Zhanga [Zhang 2002] projektowanie i doskonalenie systemów hybrydowych jest zadaniem trudnym, ponieważ składają się one z dużej liczby modułów, między którymi zachodzą wzajemne powiązania. Dostępne narzędzia informatyczne nie wspierają procesu budowy zintegrowanych HSWD właśnie ze względu na ich złożoność oraz możliwość wystąpienia licznych, niepożądanych interakcji, które są niezwykle trudne do przewidzenia i zdiagnozowania. Łatwo znaleźć przykłady kombinacji nawet tylko dwóch technik (np. sieci neuronowych i logiki rozmytej), których zastosowanie w wielu przypadkach nie jest zachęcające z powodu braku oczekiwanego wzrostu dokładności rozwiązań w akceptowalnym czasie [Zabawa 2005]. A. Dutta [Dutta 1996] dowodzi, że głównym problemem podczas tworzenia systemów hybrydowych jest konieczność dostosowania metod integracji poszczególnych komponentów do dziedziny zastosowań. Dlatego też niezbędne jest opracowanie wielu narzędzi wspierających integrację procedur, struktur i danych w HSWD. Bez takiego pakietu wspomagającego trudno będzie oczekiwać znaczącego rozwoju tej grupy systemów. Niemniej w fazie testowej, eksperymentalnej lub prototypowej opisywanych jest wiele przykładów zastosowań HSWD.

### 10.3.3. Przykłady zastosowań HSWD

System hybrydowy łączący w sobie techniki sieci neuronowej, wnioskowania na podstawie przypadków oraz analizy dyskryminacyjnej (analiza statystyczna) został przedstawiony w pracy H. Jo i I. Han [Jo, Han 1996]. HSWD został wykorzystany w celu oszacowania zagrożenia firmy bankructwem. Klasyfikację przeprowadzano na podstawie wartości funkcji dyskryminacji. Po dokonaniu wielu eksperymentów z bazą danych dotyczących firm, które uległy bankructwu, autorzy doszli do wniosku, że integracja wybranych metod daje lepsze wyniki (jeśli chodzi o błąd klasyfikacji) niż każda z nich zastosowana oddzielnie.

Tradycyjna metoda rachunku kosztów, przypisująca koszty działań poszczególnym produktom, ma dwie słabe strony: brak uznanych kryteriów reguł wyboru nośników kosztów oraz założenie o liniowym charakterze funkcji kosztów (koszty cał-

kowite mają być proporcjonalne do pojedynczego nośnika kosztów). K. Kim i I. Han [Kim, Han 1997] wykazali, że problem daje się rozwiązać po zastosowaniu połączenia szeregowego algorytmu genetycznego oraz sztucznej sieci neuronowej. W pierwszym etapie wykorzystano procedurę algorytmu genetycznego, natomiast w drugim — sztuczną sieć neuronową ze wsteczną propagacją błędów oraz funkcją aktywacji typu sigma. Zadaniem algorytmu genetycznego było zidentyfikowanie suboptymalnych nośników kosztów i wyznaczenie liczby ukrytych neuronów w sieci neuronowej. Optymalizacja została przeprowadzona za pomocą algorytmu genetycznego. Odbywała się ona w obrębie populacji tych chromosomów, które reprezentowały określone parametry modelu. Funkcją przystosowania była średnia różnica, czyli odchylenie między wartościami oczekiwanych i przewidywanych kosztów rozpatrywanego produktu. Jak wykazały badania, w rozwiązaniu wykorzystującym architekturę hybrydową uzyskano dokładniejsze oszacowanie kosztów przy jednoczesnym zmniejszeniu liczby neuronów w warstwie ukrytej, czyli przy uproszczeniu struktury całej sieci.

Polskim HSWD, dostępnym obecnie na rynku jest Aitech DSS<sup>4</sup>, autorstwa K. Michalika. Jest to uniwersalny system hybrydowy do analizy i interpretacji dowolnych danych zarówno ilościowych, jak i jakościowych. Podstawowe jego funkcje to:

- pobieranie danych ze źródeł zewnętrznych;
- agregacja i przetwarzanie danych (np. automatyczne obliczanie wskaźników ekonomicznych, prognozy, symulacje);
- wizualizacja danych wejściowych i otrzymanych wyników;
- tworzenie prognoz wybranych czynników ekonomicznych;
- łatwe definiowanie dowolnych scenariuszy analizy danych;
- automatyczna ocena (interpretacja) danych i wskaźników;
- automatyczna publikacja danych, przebiegu i wyników analizy w formie raportów w formacie dokumentów Microsoft Word.

W komunikacji z użytkownikiem Aitech DSS wykorzystuje mechanizm scenariuszy udostępniających zbiór zhierarchizowanych metod, umożliwiających rozwiązanie problemu. Obecna wersja systemu zawiera 14 typów metod, które można podzielić na: metody wizualnego wprowadzania i prezentacji danych (arkusz kalkulacyjny, arkusz czasowy, dialog, wykres), metodę automatycznego importu i eksportu danych do BD oraz metody obliczeniowe i interpretujące dane (SE, sieć neuronowa), a także metodę raportowania.

Aitech DSS znajduje zastosowanie w następujących obszarach tematycznych:

- bieżący monitoring finansowy w zakresie płynności, rentowności i obsługi zadłużenia;
- wczesne ostrzeżenie o powstających zagrożeniach w sytuacji finansowej;

<sup>4</sup> <http://aitech.pl/content/view/65/54/lang,ISO-8859-2/> (pobranie danych: 30.03.2010).

- planowanie finansowe oparte na metodach statystycznych oraz na założeniach ekspertów;
- wielowariantowa ocena efektywności projektów inwestycyjnych połączona z symulacjami zmian warunków, wejściowych przy założeniach i w czasie trwania inwestycji;
- budżetowanie i kontroling;
- ocena ryzyka związanego z kredytem bankowym.

## 10.4. Pytania i zadania kontrolne

### 10.4.1. Pytania kontrolne

1. Omów elementy struktury systemu ekspertowego (SE).
2. Czym się powinien charakteryzować obszar pracy SE?
3. Scharakteryzuj mechanizm działania maszyny wnioskującej.
4. Omów mechanizmy doskonalące działanie SE.
5. Omów fazy projektowania BW w SE.
6. Na czym polega technika walidacji w SE?
7. Omów działania prowadzące do unikania potencjalnych błędów w projektowaniu SE.
8. Wyjaśnij zalety szkieletowych SE.
9. Wyjaśnij rolę podsystemu akwizycji wiedzy w SE.
10. Wyjaśnij rolę podsystemu wyjaśniająco-uzasadniającego w SE.
11. Zdefiniuj pojęcie BI.
12. Omów ideę działania systemu BI i jego wpływ na efektywność zarządzania przedsiębiorstwem.
13. Jakie elementy wpływają na utratę wartości biznesowej podejmowanej decyzji zarządczej?
14. Omów architekturę systemu BI.
15. Jakie obszary działalności firmy mogą być wspierane przez rozwiązania BI i w jakim zakresie?
16. Jaka jest podstawowa cecha wyróżniająca współczesne HSWD?
17. Omów klasyfikację HSWD ze względu na typ połączenia hybrydowego.
18. Omów klasyfikację HSWD ze względu na stopień i sposób integracji składowych struktury systemu.
19. Jakie są podstawowe problemy projektowania i wdrażania HSWD?
20. Przedstaw jedno rozwiązanie dostępne na rynku HSWD.

### 10.4.2. Przykładowe zadania

1. Dla wybranej funkcjonalności zarządzania przedsiębiorstwem produkcyjnym zaprojektuj ramową strukturę SE, wskazując podstawowe grupy danych wejściowych w BW oraz najważniejsze zestawienia wynikowe.
2. Podaj przynajmniej cztery przykłady zapisu metawiedzy z otoczenia przedsiębiorstw w obszarze regulacji prawnych i wskaźników makroekonomicznych.
3. Dla wybranego przedsiębiorstwa wskaż obszary zastosowania systemu BI w podziale na zastosowania operacyjne i analityczne.
4. Dla wybranego przedsiębiorstwa wskaż obszary zastosowania HSWD. Na podstawie wiedzy z wcześniejszych rozdziałów książki zaproponuj możliwe i celowe kombinacje technik i metod do wykorzystania w HSWD.

# Przykłady praktycznych zastosowań SWD

W rozdziale przedstawiono dwa, zasadniczo odmienne, obszary możliwych zastosowań SWD: zarządzanie rynkiem ubezpieczeniowym oraz zarządzanie produkcją rolną. Każdy z rozważanych obszarów został przedstawiony jako studium przypadku. Wskazano na: specyficzne cechy różnych SWD, związane z badaniem ich przydatności eksperymenty symulacyjne oraz korzyści płynące z wykorzystania branżowo różnych rozwiązań.

## 11.1. System Business Intelligence w zarządzaniu ryzykiem ubezpieczeniowym

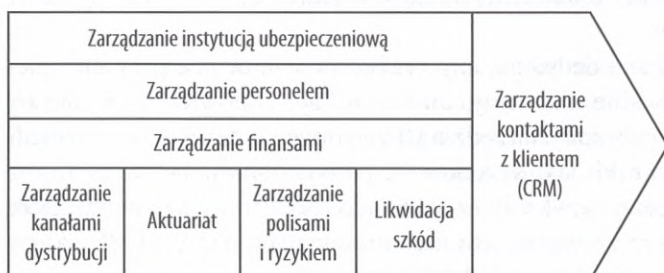
Obecnie w Polsce nie ma niez informatyzowanych firm ubezpieczeniowych, ale wiele z nich pracuje na wyspecjalizowanych aplikacjach o wąskim zakresie zastosowań, bez możliwości łatwej i szybkiej wymiany danych. Co prawda, dane, gromadzone na poziomie transakcyjnym, są bardzo obszerne i dotyczą prawie każdego aspektu działalności firmy, ale mnogość wykorzystywanych rozwiązań informatycznych powoduje utrudnienie korzystanie ze zgromadzonych w nich zasobów. Nawet w przypadku systemu zintegrowanego dane często są rozproszone w wielu modułach i niełatwo z nich skorzystać, jeśli celem jest uzyskanie przekrojowych sprawozdań zarządczych.

Pojawiają się zatem nowe potrzeby w zakresie migracji danych, integracji istniejących rozwiązań, a przede wszystkim swobodnego dostępu do zintegrowanej i przetworzonej informacji, która wpłynie na jakość i efektywność zarządzania firmą. W tej sytuacji ważną grupą oprogramowania aplikacyjnego, stosowanego w sektorze ubezpieczeń, stają się narzędzia klasy Business Intelligence (BI), które umożliwiają budowę zintegrowanych systemów informacji zarządczej (BI), udostępniających kompletną, aktualną i prawdziwą informację, wspomagającą proces zarządzania w firmie.

### 11.1.1. Cechy funkcjonalne systemu BI

Funkcjonalność kompleksowego systemu BI dla rynku ubezpieczeniowego powinna obejmować wszystkie kluczowe aspekty zarządzania firmą ubezpieczeniową (rysunek 11.1). Biorąc pod uwagę problematykę rozdziału, skoncentrujemy się na omówieniu zarządzania ryzykiem ubezpieczeniowym w trzech aspektach: analizy aktuarialnych, oceny ryzyka i zarządzania polisami oraz likwidacji szkód.

Rysunek 11.1. Kluczowe obszary zarządzania instytucją ubezpieczeniową



Źródło: [Sobieszczyk 2007].

Funkcja aktuariatu jest kluczowa w zapewnieniu prawidłowego funkcjonowania firmy ubezpieczeniowej. Dotyczy szacowania ryzyka w odniesieniu do majątku, który jest ubezpieczony. W przypadku ubezpieczeń zdrowotnych i życiowych polega ona na kalkulowaniu prawdopodobieństwa wypadku albo śmierci, bazując na zróżnicowanych charakterystykach demograficznych i środowiskowych. Zadania działu aktuariatu są nadzwyczaj złożone i mają strategiczne znaczenie dla całej instytucji ubezpieczeniowej. Aktuariat korzysta ze skomplikowanych modeli matematycznych, które wykorzystuje do obliczania przyszłych składek ubezpieczeniowych i alokowania części w celu reasekuracji. Zastosowanie narzędzi BI może w tym przypadku znacząco usprawnić proces zarządzania w obszarze [Sobieszczyk 2007]: modelowania ryzyka, reasekuracji oraz analizy rentowności.

Modelowanie ryzyka, przy zastosowaniu narzędzi BI, polega na budowaniu modeli analitycznych identyfikujących profile ryzyka ubezpieczeniowego w różnych segmentach klientów. Modele te zawierają miary ryzyka, takie jak: średnia kwota szkody, częstotliwość występowania szkód czy wskaźnik strat. Miary ryzyka są wyznaczane dla każdego segmentu klientów, a następnie są podstawą do kalkulowania odpowiedniej wielkości składki ubezpieczeniowej dla określonej grupy klientów.

Instytucja reasekuracyjna przejmuje na siebie część ryzyka ubezpieczycieli w zamian za część składki ubezpieczeniowej płaconej przez klienta. W przypadku szkody instytucja reasekuracyjna wypłaci odpowiednią kwotę odszkodowania. Aktuariat musi podejmować decyzję o ustaleniu odpowiedniej kwoty reasekuracji, takiej żeby dochód był maksymalizowany przy akceptowalnym poziomie ryzyka

ubezpieczeniowego. Dzięki narzędziom BI można budować szacunkowe modele, które precyzyjnie określą wymagany poziom reasekuracji, odnoszący się do danych dotyczących wypłaconych odszkodowań. Tak zbudowane modele identyfikują te polisy, które powinny być przeznaczone do reasekuracji, bazując na doświadczeniach w zakresie strat z podobnych polis w przeszłości.

Rentowność istniejących produktów ubezpieczeniowych może być śledzona w odniesieniu do różnorodnych czynników, takich jak: linie produktowe, regiony geograficzne, agencje, segmenty klientów, kanały dystrybucji itp. Bazując na historycznych analizach rentowności, aktuariusze budują modele eksploracji danych szacujące popyt na nowe produkty oraz identyfikujące najbardziej rentowne segmenty klientów dla tych produktów.

Zespoły oceniające ryzyko decydują, czy ryzyko podejmowane przy ubezpieczeniu klienta jest akceptowalne dla firmy, oraz określają odpowiednią kwotę tej składki, która powinna być pobrana. Narzędzia BI wspomagają te analizy, uwzględniając w badaniu takie czynniki, jak: szkodowość polisy, poniesione straty i inne dane, mające związek z oceną ryzyka i zarządzaniem polisami. Do analiz, które mogą być wykonywane także ze wsparciem zaawansowanych narzędzi BI, należy analiza składki i analiza strat [Sobieszczyk 2007].

Analiza składki ubezpieczeniowej pozwala na śledzenie efektywności poziomu składki w odniesieniu do: poszczególnych produktów, linii produktów, regionów geograficznych, agencji, poszczególnych agentów lub oddziałów firmy. Wyniki tych analiz są pomocne w określeniu optymalnej wielkości składki, co może wpłynąć na atrakcyjność oferowanych ubezpieczeń i konkurencyjność firmy na rynku.

Analiza strat jest szczególnie pomocna w przypadku tych produktów, dla których kwota składki ubezpieczeniowej okazuje się mniejsza niż koszt ich obsługi. Taka strata może być spowodowana niedokładnym oszacowaniem początkowego ryzyka lub większym, niż zakładano, kosztem związanym z obsługą i administracją produktu. Analizy tego typu są użyteczne przy poprawie poziomu rentowności i ulepszaniu programów oceny ryzyka, a także pomagają ubezpieczycielom chronić ich środki finansowe.

Szybkie i skuteczne radzenie sobie ze szkodami tworzy podstawy dobrego systemu kontaktów z klientami. W tym samym czasie ubezpieczyciel musi rozwiązywać problemy wciąż rosnącej liczby wyłudzeń odszkodowań. Oszustwa ubezpieczeniowe są obecnie bardzo powszechne, a większość z tych dodatkowych kosztów, ponoszonych przez ubezpieczycieli z powodu oszustw, jest, niestety, przenoszona na konsumentów jako podwyższona składka ubezpieczeniowa. Zatem problem prawidłowej i bezpiecznej likwidacji szkód jest dla towarzystwa ubezpieczeniowego bardzo istotny. Analizy BI, które mogą wspomóc działanie instytucji w zakresie likwidacji szkód, to [Sobieszczyk 2007]: analiza szkód i odszkodowań, wykrywanie oszustw, szacowanie szkód i odszkodowań.



Analiza szkód i odszkodowań jest jedną z podstawowych analiz w obszarze BI dla ubezpieczeń. Dotyczą one analiz danych z obszaru odszkodowań, połączonych z innymi źródłami danych, takimi jak informacje o polisach i ocenie ryzyka. Są one wykorzystywane przede wszystkim do oceny skuteczności procesu likwidacji szkód, która jest bezpośrednio powiązana z satysfakcją klienta.

Wykrywanie oszustw podczas likwidacji szkód może być osiągnięte przez wielowymiarowe analizy danych o odszkodowaniach połączonych z innymi wewnętrznymi i zewnętrznymi danymi, takimi jak historie płatności i ocena ryzyka. Pomocne są w tym przypadku narzędzia eksploracji danych, które wykorzystuje się do budowy modeli, wykrywających zmieniające się schematy oszustw ubezpieczeniowych.

Szacowanie szkód i odszkodowań jest trudnym zadaniem ze względu na to, że rzeczywista wartość szkód nie jest znana przed momentem wprowadzenia nowych produktów na rynek. W takich przypadkach oszacowana wartość musi być odłożona w postaci rezerw gotówkowych, a takie fundusze nie mogą być używane w procesie długoterminowych inwestycji. Dlatego precyzja tych szacunków ma znaczący wpływ na rentowność instytucji ubezpieczeniowej. Wielowymiarowe analizy BI mogą być używane do analizowania danych o odszkodowaniach w poszczególnych segmentach klientów, z uwzględnieniem specyficznych warunków gospodarczych i geograficznych, w celu lepszego oszacowania przyszłych odszkodowań. Do osiągnięcia wyższej dokładności tych analiz można wykorzystać narzędzia eksploracji danych i ich zaawansowane modele szacowania odszkodowań.

### 11.1.2. Architektura systemu BI

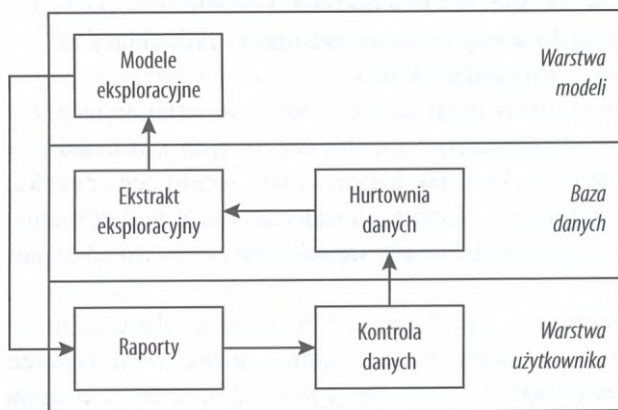
Architektura systemu zaprojektowanego na potrzeby eksperymentów symulacyjnych w jednym z czołowych polskich towarzystw ubezpieczeniowych [Rostek 2004] miała charakter prototypu i była zgodna z klasycznym schematem SWD (rysunek 11.2): warstwa użytkownika, baza danych, warstwa modeli eksploracyjnych.

Elementem inicjującym działanie systemu są wyniki standardowych analiz i raportów, przygotowywanych w dziale analitycznym. Pojawiające się nietypowe sytuacje i problemy decyzyjne, które nie poddają się analizie oraz interpretacji standardowymi metodami, są kierowane do systemu BI.

Wyselekcjonowane do badania dane HD są podstawą do przygotowania ekstraktu analitycznego, który jest wprowadzany do przygotowanego modelu eksploracyjnego. Wyniki analiz eksploracyjnych są selekcjonowane, interpretowane, a następnie dokumentowane w raportach, które trafiają do zainteresowanego decydenta.

Warunkiem wysokiej efektywności działania systemu BI jest okresowe testowanie jakości modeli. Jest to konieczne ze względu na zmienność i nieliniowość zarówno schematów oszustw ubezpieczeniowych, jak i preferencji klientów produktów ubezpieczeniowych.

**Rysunek 11.2. Architektura prototypu systemu BI dla polskiego towarzystwa ubezpieczeniowego**



Źródło: Opracowanie własne.

### 11.1.3. Eksperymenty symulacyjne

Na podstawie prototypu systemu BI zostały przeprowadzone eksperymenty symulacyjne, testujące przydatność systemu w zastosowaniu korporacyjnym [Rostek 2005]. Prototyp systemu BI został zaprojektowany i wykonany w środowisku SAS 9.1, a realizowane eksperymenty podzielono na dwie grupy:

- 1) eksperymenty dotyczące ubezpieczeń długoterminowych — polisy na życie i polisy posagowe;
- 2) eksperymenty dotyczące ubezpieczeń majątkowych — polisy AC.

Eksperymenty oparte na danych obejmujących długi horyzont czasowy dotyczyły badań związanych z analizą ubezpieczeń życiowych i posagowych. Polisy takie zawierane są na okres od kilku do kilkudziesięciu lat, a cechą charakterystyczną ich analizy jest fakt, że badaniu poddawana jest jedynie część historii ich ubezpieczenia. Jest to zjawisko tzw. ucinania obserwacji, kiedy o znacznej grupie obserwacji możemy jedynie powiedzieć, że trwały jeszcze zanim zakończyło się badanie. Zjawisko występowania obserwacji uciętych<sup>1</sup> wymaga zastosowania specyficznych technik eksploracyjnych, charakterystycznych np. dla medycyny. Są to tzw. metody analizy przeżycia (np. metoda Kaplana–Meiera [Kaplan, Meier 1958] czy regresja Coxa [Cox, Oakes 1984]), które nie stanowią podstawowych technik analizy eksploracyjnej.

<sup>1</sup> Obserwacja ucięta — obserwacja, o której wiadomo, że w chwili zakończenia badania była jeszcze aktywna i nie potrafimy dokładnie określić, kiedy stanie się nieaktywna. Z obserwacji uciętych korzysta analiza przeżycia, badająca np. czas przeżycia pacjentów po operacji.

Wszystkie eksperymenty tej grupy polegały na dokonaniu klasyfikacji klientów ze względu na ryzyko zerwania przez nich polisy ubezpieczeniowej. Ponieważ docelowo badanie obejmowało różne rodzaje ubezpieczeń, więc każdy z typów został poddany oddzielnej analizie, ale opartej na takim samym modelu eksploracyjnym.

Eksperymenty oparte na danych obejmujących krótki horyzont czasowy dotyczyły badań związanych z analizą polis AC samochodów osobowych. Polisy takie zawierane są na okres jednego roku z możliwością przedłużenia ich na kolejny rok. Eksperymenty tej grupy dotyczyły:

- klasyfikacji klientów polis indywidualnych i pakietowych ze względu na prawdopodobieństwo przedłużenia lub polisy AC na kolejny rok;
- wyodrębnienia cech ryzyka wystąpienia szkody w grupie polis AC.

Ze względu na rozbieżny charakter realizowanych zadań badawczych podzielono je na wiele eksperymentów opartych na dwóch podstawowych modelach eksploracyjnych:

- 1) modelu klasyfikacji klientów polis AC,
- 2) modelu klasyfikacji polis ze względu na ryzyko wystąpienia szkody.

W przypadku modelu klasyfikacji klientów polis AC eksperyment obejmował analizę ubezpieczeń AC zawieranych na jeden rok, kiedy badanie dotyczy całej historii życia polisy. W tym przypadku nie wystąpiła konieczność analizy obserwacji uciętych. Standardowe techniki eksploracyjne (np. regresja logistyczna i drzewa decyzyjne) okazały się wystarczające do opisanego zjawiska.

Klasyfikacja polis i wyodrębnienie czynników zwiększających lub zmniejszających ryzyko wystąpienia szkody ubezpieczeniowej jest, obok klasyfikacji klientów, jednym z najczęściej realizowanych zadań eksploracyjnych w zakładach ubezpieczeniowych. Do budowy modelu eksploracyjnego, służącego tej analizie, ponownie wykorzystano metodę regresji logistycznej (podstawa modelu) i drzewa decyzyjnego (metoda uzupełniająca).

Projekt modelu eksploracyjnego, polegający na najwłaściwszym doborze metod analitycznych do rozpatrywanego zagadnienia, jest pierwszym etapem realizacji badania (rysunek 11.3).

Kolejnym etapem jest podział ekstraktu analitycznego na trzy części:

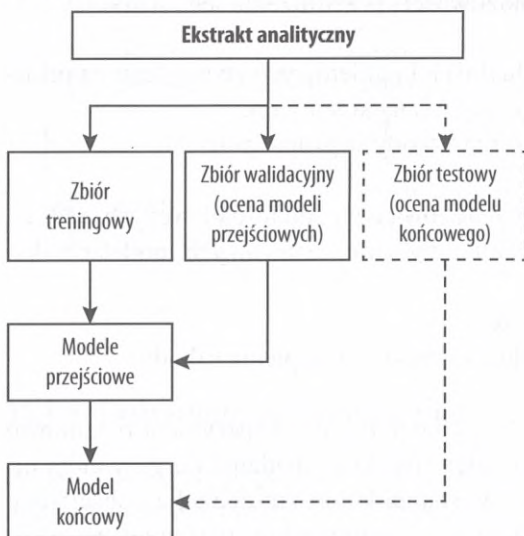
- 1) zbiór treningowy (model jest poddawany treningowi);
- 2) zbiór walidacyjny, który służy do selekcji różnych wariantów modelu eksploracyjnego;
- 3) zbiór testowy (ocena modelu przyjętego do eksploatacji).

Obowiązkowe jest zastosowanie zbioru treningowego i walidacyjnego, zbiór testowy jest opcjonalny.

Następnie spośród różnych możliwych kombinacji metod analitycznych oraz zestawów zmiennych tworzone są kolejne warianty modelu eksploracyjnego i diagno-

zowana jest jego użyteczność dla odbiorcy. Poszukiwany jest model, którego wyniki użyteczności są możliwie najwyższe, a jednocześnie spełnione są kryteria ich poprawności i wiarygodności.

**Rysunek 11.3. Przygotowywanie modelu eksploracyjnego do eksploatacji**



Źródło: Opracowanie własne.

Model przyjęty do eksploatacji nosi nazwę modelu końcowego i stanowi element bazy modeli systemu BI. Nie oznacza to jednak, że jego użycie będzie niezmiennie efektywne w całym okresie eksploataowania systemu BI. Obowiązkiem analityka jest okresowa weryfikacja jakości modelu oraz wiarygodności jego wyników. Szczególnie dotyczy to tych obszarów, gdzie następuje duża zmienność wyników modelu w czasie oraz są silnie zróżnicowane wzorce i zachowania. Przykładowo, modele identyfikacji oszustw ubezpieczeniowych muszą być często modyfikowane, chyba że zostały zbudowane przy zastosowaniu metody gwarantującej zdolność uczenia się (np. sztuczne sieci neuronowe).

#### 11.1.4. Charakterystyki eksploatacyjne systemu BI

System BI w wersji prototypowej został uruchomiony na pojedynczym stanowisku komputerowym klasy PC 2GB RAM. Analizy prowadzono na ekstrakcie zawierającym ok. 200 tys. rekordów, o objętości poniżej 50 MB. Poważnym utrudnieniem była zbyt mała moc obliczeniowa komputera, która uniemożliwiała np. wykorzystanie modelu sieci neuronowych, bardzo wskazanego do zastosowania w rozwiązaniu opisanego problemu. Przeprowadzone eksperymenty symulacyjne wykazały wyso-

ką skuteczność modeli w testowanym obszarze i podniosły efektywność zarządzania w zakresie: znajomości klientów poszczególnych grup produktów ubezpieczeniowych, analizy ryzyka ubezpieczeniowego, przewidywania odchodzenia klientów do konkurencji, wykrywania oszustw i nadużyć oraz szacowania wielkości składek ubezpieczeniowych. Dlatego została podjęta decyzja o rozbudowie tego systemu do wersji pełnej.

W wersji docelowej projekt przewidywał zastosowanie, zamiast pojedynczego stanowiska komputerowego, serwera analitycznego o znacznie większej mocy obliczeniowej, umożliwiającego wykorzystanie każdej dostępnej w pakietach SAS metody analizy eksploracyjnej danych. Działanie systemu, opartego na architekturze przedstawionej na rysunku 11.3 oraz korzystającego z modeli opracowanych w wersji prototypowej systemu, jest następujące:

- uruchamiany jest proces uaktualniania ekstraktu eksploracyjnego, czyli ładowany jest nowy zbiór danych, pobranych z hurtowni danych;
- uruchamiane są modele eksploracyjne i przeprowadzana jest analiza na nowym ekstrakcie danych;
- sprawdzane są wyniki analizy pod kątem ich poprawności statystycznej oraz przydatności w procesie podejmowania decyzji;
- jeżeli ocena uzyskanych wyników analizy jest pozytywna, to przygotowywane są raporty dokumentujące wyniki przeprowadzonych analiz;
- analiza jakości otrzymanych raportów może dodatkowo skutkować wymuszeniem kontroli danych oraz działań naprawczo-konserwujących w odniesieniu do zasobów hurtowni danych.

Jeżeli ocena wyników analizy pod kątem ich poprawności statystycznej oraz przydatności w procesie podejmowania decyzji nie jest satysfakcjonująca, to modele są modyfikowane przez pracowników IT. Również jeżeli przygotowane raporty wynikowe analizy przestają być wystarczające dla użytkowników, należy je przebudować lub stworzyć nowe.

Najbardziej zaawansowani użytkownicy systemu BI otrzymają uprawnienia do samodzielnego tworzenia modeli eksploracyjnych i modyfikowania raportów wyników w zakresie przygotowanego ekstraktu eksploracyjnego. Pozostałe osoby będą korzystały z pakietu standardowych raportów, przygotowanych przez pracowników IT na zamówienie użytkowników systemu.

## **11.2. System ekspertowy w zarządzaniu produkcją rolniczą**

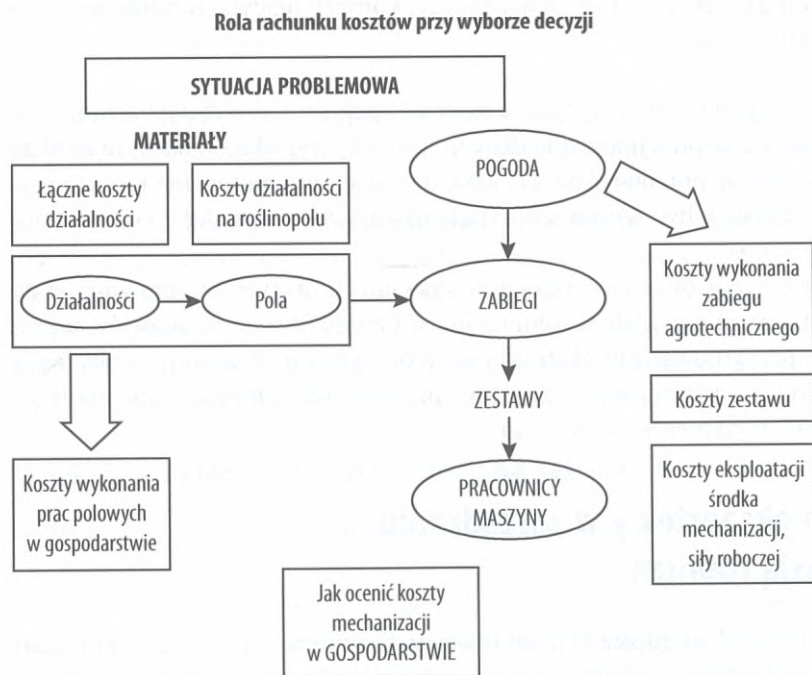
Ogólna koncepcja SE obsługującego dział inwestycji wspomagających zakup i użytkowanie sprzętu rolniczego związana jest z uszczegółowionym modelem w warunkach funkcjonowania przedsiębiorstw produkcji rolniczej w różnych obszarach

zastosowań praktycznych (zmiana parku maszynowego, powiększenie areału upraw, rodzaju upraw itp).

### 11.2.1. Cechy funkcjonalne SE

SE zbudowany na potrzeby zarządzania produkcją roślinną musi uwzględniać sytuację problemową wynikającą z potrzeb użytkowników, a więc powinien brać pod uwagę takie komponenty, jak: pola, działalności, materiały, warunki pogodowe, zabiegi agrotechniczne oraz zestawy pracowników maszyn i innych elementów sprzętu rolniczego. Wyszczególnione elementy będą miały wpływ na organizacyjno-ekonomiczne decyzje dotyczące zarządzania produkcją roślinną — zatem ich uwzględnienie w architekturze SE jest niezbędne (rysunek 11.4). Przepisanie wyszczególnionym elementom nakładów czasu oraz środków finansowych umożliwi optymalne planowanie prac polowych oraz pozwoli zminimalizować koszt przy zachowaniu wysokiej jakości realizowanych procesów technologicznych produkcji polowej. Zasadniczą trudność stanowi ustalenie parametrów technologicznych z uwagi na dużą ich zależność od warunków pogodowych podwyższających ryzyko konieczności zmiany planów prac polowych.

**Rysunek 11.4. Strukturalne elementy SWD dla potrzeb tworzenia modeli przedsiębiorstw rolnych wspomagających decyzje dotyczące wyboru maszyn**



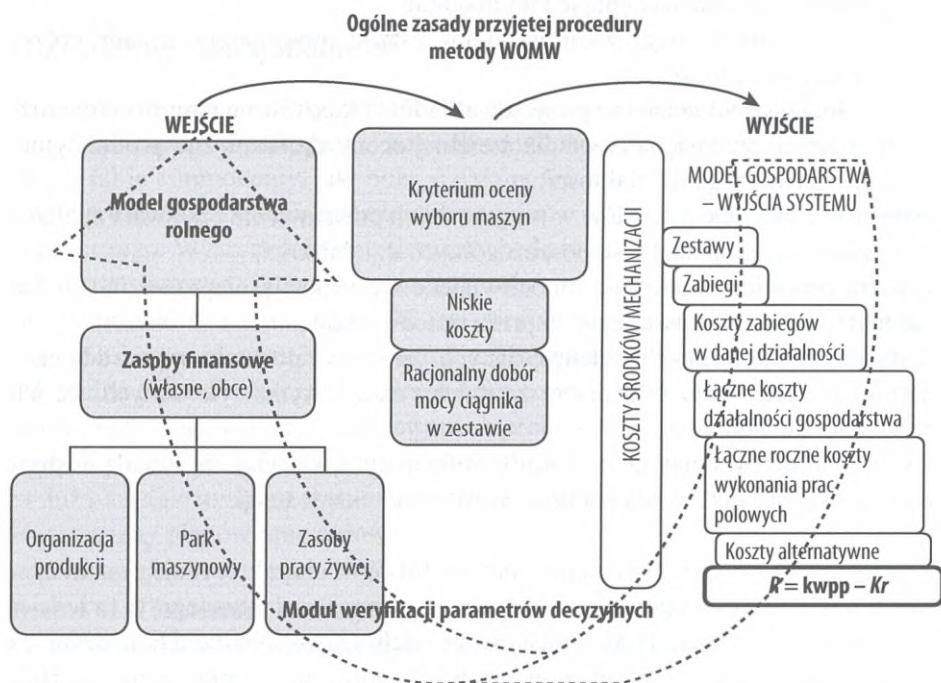
Opracowana aplikacja MOWM [Bojar 2005] pozwala jako SE realizować funkcjonalności ważne w procesie wspomagania decyzji menedżerów przedsiębiorstw rolnych, ponieważ spełnia następujące wymagania:

- jest dostosowana do organizacji produkcji, zarządzania i poziomu kadr, co zapewnia jej skuteczność;
- uwzględnia wszystkie istotne pozycje nakładów i kosztów na danym etapie rozwoju i potrzeb przedsiębiorstwa dla rzetelnej oceny ekonomiczno-produkcyjnej i organizacyjnej jego działalności;
- zapewnia ewidencję nakładów w miejscach ich powstawania i pozwala analizować rzeczywistą opłacalność poszczególnych działalności;
- zawiera procedury kontrolne umożliwiające weryfikację nieprawdziwych lub błędnych danych wprowadzanych przez pracowników;
- maksymalnie upraszcza ewidencję danych, zgodnie z podstawową zasadą ekonomiki pozyskiwania informacji, że należy ustalać tylko tyle danych, ile ich bezwzględnie potrzeba;
- wykorzystuje różne narzędzia i środki informatyki, kierując się zasadą dostępności i niskich kosztów ich zakupu, wdrożenia i eksploatacji.

Zamieszczona procedura działania metody MOWM charakteryzuje system ekspertowy umożliwiający racjonalny wybór sprzętu zmechanizowanego w przedsiębiorstwie rolnym (rysunek 11.5). Podstawowe cechy funkcjonalne SE decydujące o jego zastosowaniach w zarządzaniu produkcją rolniczą to: przyjazny interfejs użytkownika, regionalne bazy danych normatywnych, monitoring warunków pogodowych, rozbudowany moduł objaśniający, nawiązujący do zasad projektowania interfejsu użytkownika, sformułowanych w rozdziale 3. Należy podkreślić, że stosowanie zasad przyjaznego interfejsu — doskonałego w sposób ewolucyjny przy współpracy z użytkownikiem i pozwalającego intuicyjnie uczyć się obsługi oprogramowania — było naczelną zasadą przy opracowywaniu aplikacji MOWM. Wiele komunikatów anonsujących wyniki rozwiązań czy też informacje z modułu objaśniającego były dyskutowane z użytkownikami, a następnie modyfikowane tak, aby nie było błędów w ich interpretacji. Warunek poprawności semiozy znaków został w ten sposób spełniony.

Wielkość BD normatywnych aplikacji MOWM określa 888 zbiorów BD empirycznych, obejmujących parametry techniczno-eksploatacyjne środków mechanizacji, metadane klasyfikacyjne zabiegów, a także średnie nakłady czasu i kosztów. Podstawą do ustalenia wartości normatywnych są również dane zawarte w prawie 1000 zbiorach BW. W skład zakresu danych normatywnych powinny wejść także parametry o pogodzie z regionu, które rzutują bezpośrednio na przebieg procesów technologicznych w produkcji roślinnej. Regionalne bazy danych w rolnictwie, w odróżnieniu od przemysłu, są bardzo drogie, ponieważ zmienność czynników jest większa (np. wpływ warunków pogodowych, glebowych i roślinnych na koszty upraw kombajnu zbożowego).

Rysunek 11.5. Strukturalne elementy SWD dla potrzeb tworzenia modeli przedsiębiorstw rolnych wspomagających decyzje dotyczące wyboru maszyn



Źródło: Jak rysunku 11.4.

Rozbudowany moduł objaśniający w aplikacji MOWM jest wynikiem uwzględnienia dwóch przesłanek: po pierwsze, przekonanie pesymistycznie nastawionych użytkowników do efektów stosowania aplikacji IT (dla potrzeb podejmowania decyzji) wymaga przystępnego tłumaczenia zasad działania zaimplementowanych modeli; po drugie, moduł objaśniający dostarcza również dodatkowych informacji wyjaśniających uzyskane rozwiązania. Z tych powodów, po wielu konsultacjach i modyfikacjach, zostały opracowane komunikaty objaśniające sposób poszukiwania rozwiązań i jednocześnie poszerzających wiedzę o istocie problemu decyzyjnego (np. jaką metodę należy zastosować, aby ustalić koszty modernizacji sprzętu).

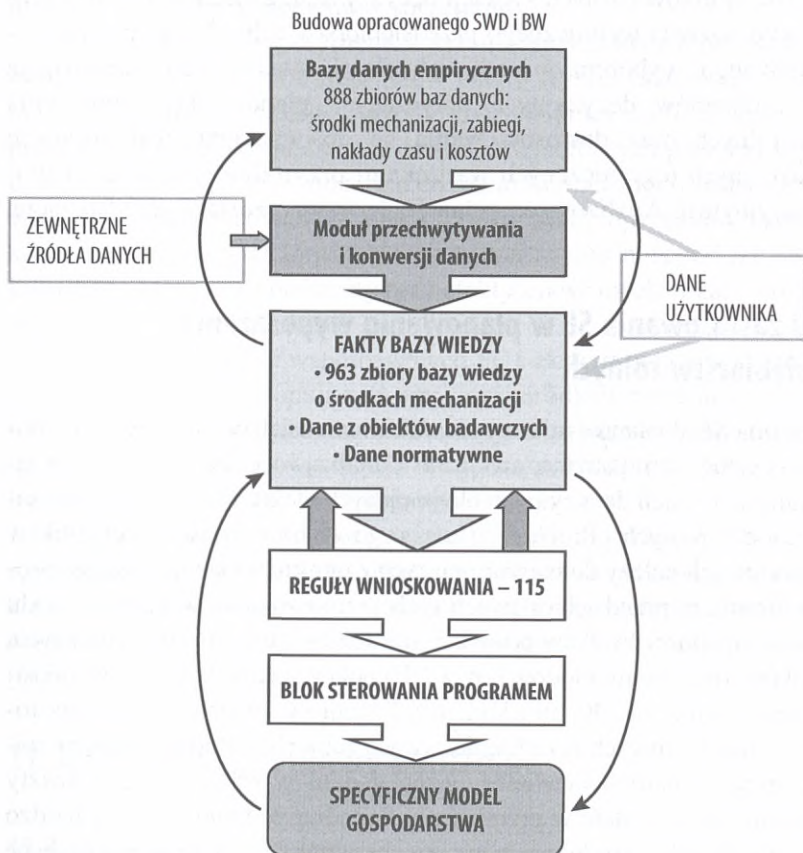
### 11.2.2. Architektura SE

W architekturze SE przedstawionej na rysunku 11.6 [Bojar 2005] można wyróżnić takie komponenty, jak: BD, BW, reguły wnioskowania, blok sterowania programem, moduł przechwytywania i konwersji danych, a także model gospodarstwa rolnego.



Szczególnie istotną funkcję pełni moduł przechwytywania i konwersji danych, który odpowiada za przekształcenie i ujednolicenie danych pochodzących z różnych źródeł zewnętrznych (także od użytkownika) w taki sposób, aby reprezentacja wiedzy w systemie zabezpieczała przed błędami formalnymi.

**Rysunek 11.6. Strukturalne elementy SWD dla potrzeb tworzenia modeli przedsiębiorstw rolnych wspomagających decyzje dotyczące wyboru maszyn**



Źródło: Jak rysunku 11.4.

Bardzo ważną rolę odgrywa również moduł interpretacyjny (grupy reguł wnioskowania) spełniających takie funkcje, jak:

- wybór i przechwytywanie danych z BW;
- automatyczne generowanie zbiorów danych o nowych typach środków mechanizacji;
- wariantowanie i wybór modeli obliczanych kosztów w zależności od preferencji użytkownika;

- weryfikacja poprawności tworzenia zestawów;
- obliczanie nakładów czasu wykonania zabiegu agrotechnicznego na podstawie norm techniczno-eksploatacyjnych oraz współczynników organizacyjnych korygujących ich wpływ na uzyskane wartości czasu.

### 11.2.3. Eksperymenty symulacyjne

Symulacje i analiza skutków różnych sytuacji decyzyjnych, związanych z optymalnym wyborem wyposażenia technicznego przedsiębiorstw rolnych (np. powiększaniem areалу uprawnego, wyborem nowych technologii, maszyn itd.), wspomagają rozwiązywanie problemów decyzyjnych dotyczących planowania wyposażenia przedsiębiorstw rolnych oraz dostosowywania go do wymagań realizowanych procesów produkcyjnych (ograniczanych warunkami przyrodniczymi oraz ekonomiczno-organizacyjnymi). Analizowane sytuacje decyzyjne zostały przedstawione w punkcie 11.2.4.

### 11.2.4. Wyniki zastosowania SE w planowaniu wyposażenia przedsiębiorstw rolnych

Wyniki zastosowania SE w planowaniu wyposażenia przedsiębiorstw rolnych są, podobnie jak eksperymenty symulacyjne, związane z analizą i rozwiązaniami dotyczącymi zróżnicowanych sytuacji decyzyjnych obejmujących efekty planowania wyposażenia przedsiębiorstw rolnych. Problem właściwego doboru maszyn, ciągników i środków transportowych należy do najważniejszych z punktu widzenia kosztów produkcji rolnej, ponieważ w przedsiębiorstwach tych przewożone są w każdym cyklu produkcyjnym znaczne ilości środków produkcji (nawozów naturalnych i sztucznych, nasion, sadzeniaków) oraz ziemiopłodów (np. z 1 ha pola uprawnego buraków cukrowych trzeba zwieźć średnio ok. 30–50 t korzeni). Ponadto w przeciętnie kilkunastolub w kilkudziesięciohektarowych przedsiębiorstwach rolnych w Polsce maszyny rolnicze używane są przez bardzo niewielką liczbę godzin w roku i dlatego koszty amortyzacji oraz inne koszty stale w przeliczeniu na jedną godzinę pracy są bardzo wysokie. Dzieje się tak, gdyż produkcja rolna ma charakter sezonowy, a nowoczesne maszyny są bardziej wydajne i droższe.

SE oraz eksperymenty symulacyjne mogą odzwierciedlać sytuacje decyzyjne dotyczące różnorodnych aspektów zmian, które przedsiębiorca rolny planuje w horyzoncie krótko-, średnio- lub długookresowym. Szybkość uzyskiwania zróżnicowanych wariantów decyzyjnych i wyników badań wrażliwości zmian modelowanej sytuacji decyzyjnej (na wprowadzane parametry decyzyjne) stanowi o przewadze SE nad tradycyjnymi metodami rachunkowymi.

Przedsiębiorca rolny może analizować różne warianty — scenariusze decyzyjne poprzez porównanie rozwiązań uzyskanych z modeli SE lub modeli symulacyjnych. Dotyczą one oceny wykorzystania środków mechanizacji w różnych okresach agro-

technicznych (np. w okresie siewów wiosennych), dla różnych wariantów struktury i skali prowadzonych działalności produkcji roślinnej (np. 80% zboża i 20% rośliny oleiste) oraz struktury parku maszynowego, np. dla różnego areалу upraw, różnych roślin uprawnych, różnych zestawów ciągnikowo-maszynowych i kombajnowych, w różnych okresach agrotechnicznych. Przydatnymi miernikami mogą być koszty mechanizacji, liczba godzin wykorzystania maszyny lub ciągnika w roku albo w poszczególnych okresach, niedobory lub nadmiar dysponowanych godzin dyspozycyjnych poszczególnych elementów sprzętu itd.

Rozwiązania z eksperymentów symulacyjnych lub z modułu wyjścia SE umożliwiają także porównanie mierników oceny wykorzystania środków mechanizacji przy uwzględnieniu różnego poziomu kosztów pracy osób zatrudnionych w procesie produkcji. Przyjęty na podstawie danych statystycznych poziom dysparytetu dochodów ludności wiejskiej w stosunku do ludności miejskiej warunkuje koszty wykonywanych zabiegów technologicznych. Opłata pracy operatorów maszyn i pracowników pomocniczych także jest zróżnicowana w zależności od poziomu niezbędnych kwalifikacji i wpływa na koszty oraz efektywność mechanizacji procesów produkcji roślinnej.

Inna grupa decyzji wspomaganych SE i eksperymentami symulacyjnymi dotyczy oceny i poprawy stopnia wykorzystania środków mechanizacji poprzez wspólne użytkowanie maszyn lub wzrost świadczenia usług mechanizacyjnych, wpływając na zmniejszenie kosztów stałych utrzymania maszyn w przeliczeniu na większą liczbę ich właścicieli lub jednostek wykonanej pracy.

Kolejna grupa wskaźników wartościowych uzyskanych z SE i modeli symulacyjnych dotyczy określenia rozmiaru świadczonych usług lub rozmiaru łącznego poziomu rocznego wykorzystania maszyn w celu uzyskania pożądanej poprawy wykorzystania środków mechanizacji zarówno w ujęciu kosztowym, jak i organizacyjnym [Bojar 2005; Bojar 2001].

## 11.3. Pytania i problemy kontrolne

### 11.3.1. Pytania kontrolne

1. Wymień kluczowe obszary funkcjonalne systemu BI dedykowanego instytucjom ubezpieczeniowym.
2. Omów zakres zastosowania rozwiązań BI w dziale aktuariatu towarzystwa ubezpieczeniowego.
3. Wskaż przykłady zastosowania systemu BI w obszarze zarządzania kontaktami z klientem towarzystwa ubezpieczeniowego.
4. Z jakich etapów składa się poprawnie skonstruowana eksploracja danych?
5. Co stanowi główne źródło danych dla systemu BI?

6. Jakie dwa podsystemy w architekturze SE wspomagającego decyzje menedżerów w produkcji rolnej mają szczególnie duże znaczenie i dlaczego?
7. Jakie znaczenie w tworzeniu i uaktualnianiu baz danych normatywnych ma komercjalizacja SE w kontekście wykorzystania subsystemu akwizycji wiedzy ze szczególnym uwzględnieniem specyfiki produkcji rolnej?
8. Jakie cechy funkcjonalne SE decydują o przydatności ich zastosowań w zarządzaniu produkcją rolną?
9. Wyjaśnij rolę modułu objaśniającego w akceptacji aplikacji MOWM przez jego użytkowników.
10. Wyjaśnij rolę grupy reguł wnioskowania w module interpretacyjnym MOWM.

### 11.3.2. Problemy zastosowań SWD

1. Podaj przykłady zastosowań systemów BI w inżynierii produkcji.
2. Jakiego typu decyzje z zakresu inżynierii produkcji mogą być wspierane przez systemy BI?
3. Przygotuj przykłady analiz wspomagających zarządzanie planowaniem i produkcją w wybranym przedsiębiorstwie.
4. Jakiego typu decyzje mogą być wspomagane przez SE MOWM i eksperymenty symulacyjne w zarządzaniu produkcją rolniczą?
5. Podaj inne przykłady zastosowań SWD w rolnictwie.

## Podsumowanie

Problematyka metod oraz technologii wspomaganie decyzji jest dynamicznie rozwijającą się gałęzią nauki i techniki. Staje się również koniecznością dla tych przedsiębiorstw i organizacji, które chcą być konkurencyjne na zmiennym rynku. Pozwala na efektywne poszukiwanie szans, omijanie zagrożeń oraz optymalizację zużycia zasobów, tak niezwykle ważnych w sytuacji niekorzystnej koniunktury gospodarczej. Jednocześnie rozwój technologii informatycznych powoduje, że SWD staje się coraz bardziej dostępnym i powszechnie wykorzystywanym narzędziem, dynamicznie wpływającym na poprawę jakości zarządzania w tych organizacjach, które z niego korzystają.

Perspektywicznie jakościowo nieporównywalnie większe możliwości wspomaganie decyzji strategicznych niż obecne metody SWD stwarza koncepcja kognitywistyki, której metody pogłębionej analizy danych wzorowanej na funkcjonowaniu ludzkiego mózgu miałyby, zdaniem L. Ogieli i R. Tadeusiewicza [2011], dokonać przełomu w przejściu od udzielania odpowiedzi na pytania menedżerów przez SWD do wytyczania kadrze zarządczej zupełnie nowych dróg rozwoju ich firm.

Ambicją autorów podręcznika było przedstawienie jak najpełniejszego zakresu zagadnień dotyczących SWD oraz ich wpływu na funkcjonowanie organizacji gospodarczych. Wiązało się to z koniecznością wskazania licznych źródeł literaturowych, dzięki którym Czytelnik może poszerzyć i pogłębić swoją wiedzę z danego obszaru. Autorzy mają nadzieję, że Czytelnik skorzysta z przygotowanej biblioteki odwołań, traktując tę publikację jako wstęp do własnych poszukiwań i przemyśleń związanych z SWD.



## Bibliografia

- Aamodt A., Plaza E. [1994], *Case-Based Reasoning: Foundational Issues. Methodological Variations, and System Approaches*, „AI-Com, Artificial Intelligence Communications” vol. 7, no. 1.
- Aczel E.A. [2000], *Statystyka w zarządzaniu*, WN PWN, Warszawa.
- Adelman L. [1991], *Experiments, Quasi-experiments and Case Studies: a Review of Empirical Methods for Evaluating Decision Support Systems*, „IEEE Transactions on Systems, Man and Cybernetics” vol. 21, no. 2.
- Alter S. [1980], *Decision Support Systems: Current Practices and Continuing Challenges*, Addison-Wesley, Reading, MA.
- Amanowicz A. [2007], *Business Intelligence — od wiedzy do trafnych decyzji biznesowych*, „Strategie Biznesu” nr 2 (29), SAP Polska.
- Ambler W.S. [2002], *User Interface Prototypes*, „Boxes and Arrows. Essential (Low Fidelity) Bob Baxley Defining Interaction Design” no. 11.
- Arabas J. [2001], *Wykłady z algorytmów ewolucyjnych*, WNT, Warszawa.
- Back T., Schwefel H.P. [1993], *An Overview of Evolutionary Algorithms for Parameter Optimization*, „Evolutionary Computation” vol. 1, no.1.
- Barfield L. [1993], *User Interface Concepts & Design*, Addison-Wesley, New York.
- Bassiliades N., Vlahayas I. [2000], *Knowledge-Base Systems*, w: Leondies C.T. (red.), *Knowledge-Base Systems. Techniques and Applications*, t. 1, Academic Press, New York.
- Bellman R.E. [1967], *Programowanie dynamiczne*, PWN, Warszawa.
- Berry M.J., Linoff G.S. [2004], *Data Mining Techniques for Marketing, Sales, and Customer Relationship Management*, J. Wiley Publishing Inc., Indianapolis.
- Beynon-Davies P. [1999], *Inżynieria systemów informacyjnych*, WNT, Warszawa.
- Bezulski G. [2009], *Sieci neuronowe i prognozowanie*, w: Łyka J (red.), *Wybrane modele matematyczne w ekonomii. Prognozowanie i jakość*, Uniwersytet Ekonomiczny, Wrocław.
- Biddle R., Noble J. [2003], *From Essential Use Cases to Objects*, w: Constantine L.L. (red.), *For Use 2002 Conference Proceedings*, Ampersand Press.
- Bielecki W.T. [2001], *Informatyzacja zarządzania*, PWE, Warszawa.
- Bielski M. [1997], *Organizacje. Istota. Struktury. Procesy*, Uniwersytet Łódzki, Łódź.

- Billewicz G., Ziemia E. [1999], *Wykorzystanie narzędzi typu CASE w analizie danych i procesów*, w: Gołuchowski J., Sroka H. (red.), *Systemy wspomagania organizacji*, Akademia Ekonomiczna, Katowice.
- Boehm B., Abts Ch., Brown A.W. [2000], *Software Cost Estimation with COCOMO II*, Prentice Hall, Englewood Cliffs, NJ.
- Bojar W. [2001], *The Cost Budgeting in the Crop Production Connected with Organizational Schemes (Budżetowanie kosztów w produkcji roślinnej z uwzględnieniem organizacji pracy)*, Proceedings XXIX CIOSTA-CIGR V Congress titled: *Farm Work Science Facing Challenges of the XXI Century*, Wageningen Pers Edition, Krakow, Poland.
- Bojar W. [2005], *Studium wyboru maszyn w gospodarstwach rolniczych w świetle rozwoju systemów wspomagania decyzji*, „Rozprawy” nr 114, Akademia Techniczno-Rolnicza, Bydgoszcz.
- Bojar W. [2010], *Problemy zarządzania sieciami biznesu w świetle koordynacji współpracy przedsiębiorców*, „Studies & Proceedings of Polish Association for Knowledge Management” no. 32.
- Bonczek R.H., Holsapple C.W., Whinston A.B. [1981], *Foundations of Decision Support Systems*, Academic Press, New York.
- Boryczka M., Słowiński R. [1988], *Derivation of Optimal Decision Algorithms from Decision Tables Using Rough Sets*, „Bulletin of the Polish Academy of Sciences”, Series Technical Sciences, vol. 36.
- Budziński R., Szaraneck A. [2006], *Zastosowanie reguł asocjacyjnych do eksploracji baz danych*, „Studia i Materiały Polskiego Stowarzyszenia Zarządzania Wiedzą” nr 5.
- Budziński R., Wawrzyniak A. [2006], *Identyfikacja outsourcingu informatycznego w jednostkach samorządowych*, „Studia i Materiały Polskiego Stowarzyszenia Zarządzania Wiedzą” nr 5.
- Cadle J., Yeates D. [2004], *Zarządzanie procesem tworzenia systemów informacyjnych*, WNT, Warszawa.
- Cassandras Ch.G. [1993], *Discrete Event Systems: Modeling and Performance Analysis*, Irwin and Aksen Associates, Boston.
- Chmielarz W. [2005], *Aspekty integracji systemów informatycznych wspomagających podejmowanie decyzji z elementami zarządzania wiedzą*, „Studia i Materiały Polskiego Stowarzyszenia Zarządzania Wiedzą” nr 4.
- Chodak G., Kwaśnicki W. [2002], *Zastosowanie algorytmów genetycznych w prognozowaniu popytu*, „Gospodarka Materiałowa i Logistyka” nr 4.
- Cieślak M. (red.). [2004], *Prognozowanie gospodarcze. Metody i zastosowania*, WN PWN, Warszawa.
- Codd E., Codd S., Smalley C. [1993], *Providing OLAP to User-Analysts: An IT Mandate*, Hyperion Solutions Corporation, Sunnyvale–Manchester–Singapore.
- Connolly T., Begg C. [2004], *Systemy baz danych — projektowanie, wdrażanie i zarządzanie w praktyce*, t. 2, Wydawnictwo RM, Warszawa.
- Constantine L.L., Lockwood L.A. [1999], *Software for Use: A Practical Guide to the Essential Models and Methods of Usage — Centered Design*, Prentice Hall, NJ.
- Constantine L.L., Lockwood L.A. [2001], *Process Agility and Software Usability*, „The Management Forum. Soft Development” vol. 9, no. 6.
- Constantine L.L., Lockwood L.A. [2002], *Instructive Interaction*, „User Experience” vol. 1, no. 3.
- Cormen T.H., Leiserson C.L., Rivest R.L. [1997], *Wprowadzenie do algorytmów*, WNT, Warszawa.



- Coulouris G., Dollimore J., Kindberg T. [1998], *Systemy rozproszone — podstawy i projektowanie*, WNT, Warszawa.
- Cox D.R., Oakes D. [1984], *Analysis of Survival Data*, Chapman & Hall, London–New York.
- Coyle J.J., Bardi E.J., Langley J.C. Jr [2002], *Zarządzanie logistyczne*, PWE, Warszawa.
- Czermiński J. [2002], *Systemy wspomaganie decyzji w zarządzaniu przedsiębiorstwem*, TNOiK, Toruń.
- Davenport T., Prusak L. [1998], *Working Knowledge*, Harvard Business School Press, Boston.
- Delobel C., Adiba M. [1989], *Relacyjne bazy danych*, WNT, Warszawa.
- DeSanctis G., Gallupe R.B. [1987], *A Foundation for the Study of Group Decision Support Systems*, „Management Science” vol. 33, no. 5.
- Dietrich J. [2003], *The Mandrax Manual*, Institute of Information Sciences & Technology, Te Kura Putaiao o Hangarau-a-Mohiotanga Massey University Palmerston North New Zealand, <http://mandarax.sourceforge.net/docs/mandarax.pdf>.
- d’Inverno M. [2006], *Theory and Application of Intelligent Agent Systems*, The Third International Conference on Informatics in Control, Automation and Robotics, Intelligent Control Systems and Optimization, Setúbal, Portugal.
- Dorigo M., Stützle T. [2004], *Ant Colony Optimization*, MIT Press, Cambridge, MA.
- Drelichowski L., Bobek S., Bojar W., Chęsy W., Cilski M., Czechumski W., Feoli E., Fronczak E., Ganis P., Graul C., Gruden T., Gvozdenović M., Kołodziejowski M., Lewandowski R., Łagodziński M., Oszuścik G., Siwiec J., Sternad S., Wawrzyniak K., Zarzycki H. [2012], *Methodological Aspects and Ase Studies of Business Intelligence Applications Tools in Knowledge Management*, „Studies & Proceedings of Polish Association for Knowledge Management” no. 59.
- Duan Y., Burrelli P. [1995], *A Hybrid System for Strategic Marketing Planning*, „Marketing Intelligence and Planning” vol. 13, no. 11.
- Duch W., Mandziuk J. [2004], *Quo Vadis Computational Intelligence?*, „Machine Intelligence. Quo Vadis? Advances in Fuzzy Systems–Applications and Theory” vol. 21, World Scientific.
- Dudycz H. [2007], *Znaczenie wizualizacji w prezentacji informacji w portalu korporacyjnym*, Akademia Ekonomiczna, Wrocław.
- Dutta A. [1996], *Integrating AI and Optimization for Decision Support: a Survey*, „Decision Support Systems” vol. 18, Elsevier Science B.V.
- Duzinkiewicz K. [2009/2010], *Metody sztucznej inteligencji w sterowaniu*.
- Dziuba D. [2005], *Przesłanki rozwoju sieciowych systemów wspomaganie decyzji udostępnianych w modelu biznesowym ASP*, w: Oleński J., Olejniczak Z., Nowak J.S (red.), *Informatyka. Strategie i zarządzanie wiedzą*, PTI, Katowice.
- Encyklopedia PWN, <http://encyklopedia.pwn.pl/haslo/3995573/wiedza.html> (pobranie danych: 17.02.2012).
- eProceedings Informing Science Conference*, Krakow, Poland, 19–22 June.
- Everett R.R., Zraket C.A., Bennington H.D. [1957], *SAGE – A Data Processing System for Air Defense*, IRE, Washington DC.
- Finch D.J., Lees P.F. [1997], *Hybrid Knowledge-Based System for Chemical Incident Management*, „Expert Systems with Application” vol. 12, Elsevier Science Ltd.
- Fitzpatrick K., Bronstein C. [2006], *Ethics in Public Relations: Responsible Advocacy*, Sage Publications, Thousand Oaks, California.
- Frączkowski K. [2003], *Zarządzanie projektem informatycznym*, Politechnika Wrocławska, Wrocław.

- Frydrychowicz W., Szymańska K. [2008], *Zagadnienie sztucznych sieci neuronowych w dynamicznych procesach niestandardowej ekonomii*, „Scientific Bulletin of Chełm. Section of Mathematics and Computer Science” no. 18.
- Fuglewicz P., Stapor K., Trojan A. [1996], *CASE dla ludzi*, LUPUS, Warszawa.
- Gadomski A.M. [1999], *Global TOGA Meta-Theory*, ENEA'S e-paper.
- Gajda J.B. [2001], *Prognozowanie i symulacja a decyzje gospodarcze*, C.H. Beck, Warszawa.
- Geoffrion A.M. [1991], *A Prototype Structured Modeling Environment*, „Management Science” vol. 37, no. 12.
- Gerrity T.P. Jr. [1971], *Design of Man-Machine Decision Systems: An Application to Portfolio Management*, „Sloan Management Review” vol. 12, no. 2.
- Glushko R.J., Lindsay T. [2009], *Designing Service Systems by Bridging the „Front Stage” and „Back Stage”*, „Information Systems and eBusiness Management” vol. 7.
- Goczyła K., Zawadzka T. [2006], *Ontologie w sieci semantycznej*, „Studia Informatica. Zeszyt Naukowy Politechniki Śląskiej” nr 67, Politechnika Śląska, Gliwice.
- Godyń I., Chmielowski W.Z. [2008], *Zastosowanie wnioskowania rozmytego do prognozowania zmienności wodochłonności i zużycia wody w gospodarce*, „Zeszyty Naukowe — Informatyka”, Wydawnictwo Krakowskiej Szkoły Wyższej, Kraków.
- Goldberg E.D. [2003], *Algorytmy genetyczne i ich zastosowania*, WNT, Warszawa.
- Gołoś P. [2010], *Wdrażanie system informatycznego*, w: Zawila-Niedźwiecki J., Rostek K., Gąsiorkiewicz A. (red.), *Informatyka gospodarcza*, t. 2, C.H. Beck, Warszawa.
- Gołuchowski J. [1997], *Inteligentne systemy diagnoz ekonomicznych*, Akademia Ekonomiczna, Katowice.
- Goonatilake S., Khebbal S. (red.) [1995], *Intelligent Hybrid Systems*, J. Wiley & Sons, London.
- Gorbatov V.A. (red.) [1978], *Awtomatizacija projektirowanija slożnych logiczeskich struktur*, Energia, Moskwa.
- Gorbatov V.A. [1979], *Semanticeskaja teoria projektirowanija awtomatow*, Energia, Moskwa.
- Gotterbarn D. [2001], *Reducing Software Failures: Addressing the Ethical Risks of the Software Development Lifecycle*, w: *The 5th International Conference on The Social and Ethical Impacts of ICT ETHICOMP 2001*, vol. 2.
- Górniak J., Wawnicki J. [2003], *Pierwsze kroki w analizie danych. SPSS-PL for Windows*, Wydawnictwo SPSS, Warszawa.
- Granat J. [2009], *Systemy wspomagania decyzji i projektowanie*, Wydział Elektroniki i Technik Informatycznych, Politechnika Warszawska, Warszawa.
- Guzik B. [2005], *Ekonometria*, Akademia Ekonomiczna, Poznań.
- Gwiazda T.D. [1998], *Algorytmy genetyczne. Zastosowanie w finansach*, WSPiZ, Warszawa.
- Hackathorn R. [2004], *The BI Watch: Real-Time to Real-Value*, „DM Review Magazine” no. 14.
- Hanczar P. [2007], *Symulacja — narzędzie analizy przepływu towarów w systemie dystrybucyjnym*, „Logistyka” nr 5.
- Hand D.J., Mannila H., Smyth P. [2001], *Principles of Data Mining (Adaptive Computation and Machine Learning)*, MIT Press, Cambridge, MA.
- Harmon P., King D. [1985], *Expert Systems*, J. Wiley & Sons, New York.
- Hayes F. [2002], *The Story So Far*, „Computerworld” nr 36 (16).
- Hecht-Nielsen R. [1990], *Neurocomputing*, Addison-Wesley, New York.
- Hellwig Z. [1990], *Taksonometria ekonomiczna, jej osiągnięcia, zadania i cele*, w: *Taksonomia — teoria i jej zastosowania*, Akademia Ekonomiczna, Kraków.

- Herbst H. [1996], *Business Rules in Systems Analysis: a Meta-model and Repository System*, „Information Systems” vol. 21, no. 2.
- Holland J.H. [1975], *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, Michigan.
- Inmon W.H. [2005], *Building the Data Warehouse*, wyd. 4, J. Wiley Publishing Inc., Indianapolis.
- Jaworska E. [2005], *Wycisnąć z danych co się da*, „Manufacturing Systems Information. Polska” nr 12.
- Jaźwiński J., Żuber J. [2000], *Zasady wyznaczania zestawu obsługowego kontroli stanu systemu transportowego z wykorzystaniem opinii ekspertów*, materiały konferencyjne, XXVIII Zimowa Szkoła Niezawodności, Szczyrk.
- Jerram P., Gosney M. [1993], *Multimedia Power Tools*, Random House, New York.
- Jo H., Han I. [1996], *Integration of Case-Based Forecasting, Neural Network, and Discriminant Analysis for Bankruptcy Prediction*, „Expert Systems with Application” vol. 11, no. 4, Elsevier Science Ltd.
- Kahneman D., Slovic P., Tversky A. [1982], *Judgment Under Uncertainty: Heuristics and Biases*, Cambridge University Press, New York.
- Kaplan E.L., Meier P. [1958], *Nonparametric Estimation from Incomplete Observations*, „Journal of the American Statistical Association” vol. 53, no. 282.
- Kasperski M.J. [2003], *Sztuczna inteligencja*, Helion, Gliwice.
- Kasprzyk J. [2001], *Wieloetapowe sterowanie rozmyte*, WNT, Warszawa.
- Keen P.G.W., Scott-Morton M.S. [1978], *Decision Support Systems: An Organizational Perspective*, Addison-Wesley, Michigan.
- Kendall M.G. [1948], *Rank Correlation Methods*, Charles Griffin & Co. Ltd, London.
- Kim D.P., Nazaretov V.M., Krupa T. [1991], *Techniczna imitacja intelektu*, WNT, Warszawa.
- Kim K., Han I. [1997], *Application of a Hybrid Genetic Algorithm and Neural Network Approach in Activity-based costing*, „Expert Systems with Application” vol. 24, no. 1, Elsevier Science Ltd.
- Kimball R., Caserta J. [2004], *The Data Warehouse ETL Toolkit. Practical Techniques for Extracting, Cleaning, Conforming and Delivering Data*, J. Wiley Publishing Inc., Indianapolis.
- Kisielnicki J. [2004], *Zarządzanie wiedzą we współczesnych organizacjach*, w: Abramowicz W., Nowicki A., Owoc M. (red.), *Zarządzanie wiedzą w systemach informacyjnych*, Akademia Ekonomiczna, Wrocław.
- Kisielnicki J. [2008a], *MIS — systemy informatyczne zarządzania*, Placet, Warszawa.
- Kisielnicki J. [2008b], *Zarządzanie*, PWE, Warszawa.
- Kisielnicki J., Sroka H. [2005], *Systemy informacyjne biznesu*, Placet, Warszawa.
- Klimek M. [2006], *Wybrane możliwości wspomagania inwestycji giełdowych przy użyciu algorytmów genetycznych*, V Konferencja Matematyki i Informatyki Stosowanej. PWSZ, Chełm.
- Klimkiewicz M. [2005], *Zastosowanie zbiorów przybliżonych do diagnostyki aparatury paliwowej silników o zapłonie samoczynnym*, „Inżynieria Rolnicza” nr 14 (74).
- Klimkiewicz M., Moczulska K. [2008], *Zastosowanie zbiorów przybliżonych do analizy satysfakcji klienta serwisu pojazdów*, „Inżynieria Rolnicza” nr 1 (99).
- Klonowski Z.J. [2004], *Systemy informatyczne zarządzania przedsiębiorstwem. Modele rozwoju i właściwości funkcjonalne*, Politechnika Wrocławska, Wrocław.
- Klosow O., Jasiński K. [2007], *Wykorzystanie algorytmu genetycznego do generowania twarzy ludzkiej*, VI Konferencja Matematyki i Informatyki Stosowanej, PWSZ, Chełm.

- Knosala R. [1989], *Methoden zur Bewertung von Bauelementen als Voraussetzung für die Entwicklung von Baukastensystemen*, Technische Universität Dresden, Diss. B.
- Komorowski J., Polkowski L., Skowron A. [1999], *Rought Sets: Tutorial*, ESSLLI'99, Utrecht.
- Korczak J., Lipinski P. [2001], *Evolution Strategies: Principles & Prototypes, Research Report 2001/05*, Laboratoire des Sciences de l'Image, de l'Informatique et de la Télédéttection, CNRS, Université Louis Pasteur, Illkirch.
- Kosiński R.A. [2007], *Sztuczne sieci neuronowe. Dynamika nieliniowa i chaos*, wyd. 3, WNT, Warszawa.
- Kostrubala S., Stolecki G., Twardowski Z. [2004], *Inteligentna linia analityczna jako platforma budowy systemów wczesnego ostrzegania w strategicznym zarządzaniu organizacją gospodarczą*, w: Niedzielska E., Dudycz H., Dyczkowski M. (red.), „Prace Naukowe. Nowoczesne Technologie Informacyjne w Zarządzaniu” nr 1081, Akademia Ekonomiczna, Wrocław.
- Kostrubiec A. [2002], *Zastosowanie algorytmu mrówkowego w harmonogramowaniu produkcji przepływowej*, Wydział Zarządzania i Ekonomii, Politechnika Gdańska, Gdańsk, [www.zic.pg.gda.pl/~akost/pubs/kostrubiec2002.pdf](http://www.zic.pg.gda.pl/~akost/pubs/kostrubiec2002.pdf).
- Kotarbiński T. [1999], *Prakseologia*, Ossolineum, Wrocław.
- Kowalczyk, T. [2008], *On Derivation of Maximal Spearman Rho and Maximal Kendall Tau for Bivariate Distributions (O wyznaczaniu maksymalnego rho Spearmana i maksymalnego tau Kendalla dla rozkładów dwuwymiarowych)*, „Prace Instytutu Podstaw Informatyki Polskiej Akademii Nauk” nr 1011.
- Kozielecki J. [1977], *Psychologiczna teoria decyzji*, PWN, Warszawa.
- Koźmiński A.K., Piotrowski W. (red.) [1996], *Zarządzanie. Teoria i praktyka*, WN PWN, Warszawa.
- Koźmiński A.K., Zawislak M. [1982], *Pewność i gra. Wstęp do teorii zachowań organizacyjnych*, PWE, Warszawa.
- Krawiec K., Stefanowski J. [2003], *Uczenie maszynowe i sieci neuronowe*, Politechnika Poznańska, Poznań.
- Król Z.P. [2006], *Przegląd i komputerowa implementacja algorytmów genetycznych w oprogramowaniu edukacyjnym*, Wydział Elektryczny, Politechnika Warszawska, Warszawa, [www.zby.waw.pl/PW\\_mgr/dyplom.pdf](http://www.zby.waw.pl/PW_mgr/dyplom.pdf), praca dyplomowa.
- Krupa T. [2004], *Strategia informatyzacji w dużych organizacjach gospodarczych*, „Zeszyt Naukowy IOSP PW. Informatyka w Przedsiębiorstwie. Wybrane Zagadnienia” nr 16, Politechnika Warszawska, Warszawa.
- Krupa T. [2006a], *Elementy organizacji — zasoby i zadania*, WNT, Warszawa.
- Krupa T. [2006b], *Sieciowe modele procesów zdarzeń*, w: Knosala R. (red.), *Komputerowo zintegrowane zarządzanie*, PTZP, Opole.
- Krupa T. [2009], *Events and Event Processes*, „Foundations of Management. International Journal” vol. 1, no. 2, Faculty of Management, Warsaw University of Technology, Warsaw.
- Krzywka A. [2005], *Informacja — wiedza — mądrość*, „Edukacja i Dialog” nr 2 (165).
- Kuźniarz L., Piasecki M. [1998], *Zarys obiektowej metodologii analizy i projektowania multimedialnego interfejsu użytkownika*, w: Daniłowicz Cz. (red.), *Multimedialne i sieciowe systemy informacyjne*, Politechnika Wroclawska, Wrocław.
- Kwaśnicka H. [1999], *Obliczenia ewolucyjne w sztucznej inteligencji*, Politechnika Wroclawska, Wrocław.
- Kwiatkowska A.M. [2007], *Systemy wspomagania podejmowania decyzji. Jak korzystać z wiedzy i informacji w praktyce*, WN PWN, Warszawa.

- Lasdon L.S., Waren A.D., Jain A., Ratner M. [1978], *Design and Testing of a Generalized Reduced Gradient Code for Nonlinear Programming*, „ACM Transactions on Mathematical Software (TOMS)” vol. 4, no. 1, March.
- Law A.M., McComas M.G. [2001], *Building Valid Models: How to Build Valid and Credible Simulation Models*, Proceedings of the 2001 Winter Simulation Conference, ACM Press, Arlington, VA.
- Lee J.A.N. [1996], *Computer Looking Back*, „Computer” vol. 29, no.1.
- Leffingwell D., Widrig D. [2003], *Zarządzanie wymaganiami*, WNT, Warszawa.
- Lewicka M. [2000], *Myślenie i rozumowanie*, w: Strelau J. (red.), *Psychologia*, t. 2, Gdańskie Wydawnictwo Psychologiczne, Gdańsk.
- Liebowitz J., Letsky C. [2000], *Developing Your First Expert System*, International Society for Intelligent Systems, Rockville.
- Lipiec-Zajchowska M., Manikowski A., Sikora-Wisniewska E., Zajchowski J., Koślacz P. [2003], *Wspomaganie procesów decyzyjnych*, t. 1 (*Statystyka*), t. 2 (*Ekonometria*), C.H. Beck, Warszawa.
- Lobel L., Brust A.J., Forte S. [2010], *Programowanie Microsoft SQL Server 2008*, t. 1 i 2, APN Promise, Warszawa.
- Łazewski M., Gołębiowski M. [2006], *Własność intelektualna*, w: *Vademecum innowacyjnego przedsiębiorcy*, t. 3., STIM – SOOIPP, Warszawa.
- Łodziński A. [2006], *Komputerowy system wyboru decyzji wielokryterialnej*, V Konferencja Informatyki Stosowanej, Chełm.
- Magiera R. [2007], *Modele i metody statystyki matematycznej*, GIS, Wrocław.
- Maj K., Krupa T. [2008], *Sieci neuronowe do przetwarzania informacji narzędziem wspomagającym procesy zarządzania w elektroenergetyce*, w: Sitarski K. (red.), *Przyszłość systemów informatycznych w zarządzaniu przedsiębiorstwem*, PTZP, Opole.
- Maj K., Krupa T. [2010], *The Management Method Preventing a Crisis Situation in an Electrical Energy Utility*, „Foundations of Management. International Journal” vol. 2, no. 2, Faculty of Management, Warsaw University of Technology, Warsaw.
- Makarewicz G., Zawieska W. [2003], *Zastosowanie algorytmów genetycznych do aktywnej redukcji hałasu*, „Bezpieczeństwo Pracy” nr 1.
- Marciszewski W. [2002], *Wykłady. Logika predykatów świat modułów i zbiór relacji*, rozdz. 4, Collegium Civitas, Warszawa.
- Martyniak Z. [1986], *Organizacja i zarządzanie. 50 problemów teorii i praktyki*, Książka i Wiedza, Warszawa.
- Matsatsinis N.F., Doumpos M., Zopounidis C. [1997], *Knowledge Acquisition and Representation for Expert Systems in the Field of Financial Analysis*, „Expert Systems with Application” vol.12, Elsevier Science Ltd.
- Matsatsinis N.F., Samaras A. P. [2000], *Brand Choice Model Selection Based on Consumers' Multicriteria Preferences and Experts' Knowledge*, „Computers & Operations Research” vol. 27.
- McCown R.L. [2002], *Locating Agricultural Decision Support Systems in the Troubled Past and Socio-technical Complexity of Models for Management*, „Agricultural Systems” vol. 74 (1), no. 13.
- McCulloch W.S., Pitts W.H. [1943], *A Logical Calculus of the Ideas Immanent in Nervous Activity*, „Bulletin of Mathematical Biophysics” no. 5.
- Medsker L.R. [1994], *Hybrid Neural Network and Expert Systems*, Kluwer Academic Publisher, Boston.
- Medsker L.R. [1998], *Hybrid Intelligent Systems*, Kluwer Academic Publisher, Boston.

- Michalewicz Z. [2003], *Algorytmy genetyczne + struktury danych = programy ewolucyjne*, WNT, Warszawa.
- Michalik K. [1999], *Szkieletowy system ekspertowy PC-Shell. Podręcznik inżyniera wiedzy*, AITECH, Katowice.
- Miler A. [2006], *Sposób na pozyskiwanie i weryfikację informacji biznesowej*, Oracle Polska, Katowice.
- Minsky M. [1975], *Framework for Representing Knowledge*, w: Winston P.H. (red.), *The Psychology of Computer Vision*, McGraw-Hill, New York.
- Moore J.H., Weatherford L.R. [2001], *Decision Modeling with Microsoft Excel*, Prentice Hall, NJ.
- Morzy T. [1999], *Eksploracja danych: problemy i rozwiązania*. Instytut Informatyki, Politechnika Poznańska. V Konferencja PLOUG, Zakopane.
- Moss L.T., Atre S. [2003], *Business Intelligence Roadmap: The Complete Project Lifecycle for Decision-Support Applications*, Addison-Wesley, Boston.
- Mulawka J.J. [1997], *Systemy ekspertowe*, WNT, Warszawa.
- Murphy N. [2009], *Mechanical vs. Digital: a GUI isn't Always the Answer: User Interface Design is Not Always an Either/or Decision*, „Embedded Systems Design” vol. 22.
- Nazaretow W.M., Kim D.P., Krupa T. [1991], *Techniczna imitacja intelektu*, WNT, Warszawa.
- Niedzielska E., Dudycz H., Dyczkowski M. [2005], *Nowoczesne technologie informacyjne w zarządzaniu NTIZ'05 2005*, Akademia Ekonomiczna, Wrocław.
- Nonaka I., Takeuchi H. [2000], *Kreowanie wiedzy w organizacji*, Poltext, Warszawa.
- O'Keefe D.J. [1988], *Some Methodological and Meta-theoretical Implications of Message Design Logic*, Paper presented at the annual convention of the Speech Communication Association, New Orleans LA.
- Ogiela L., Tadeusiewicz R. [2011], *Cognitive Science — The Key to the Mind: Natural and Artificial (Kognitywistyka — klucz do umysłu naturalnego i sztucznego)*, w: *Informatyka i psychologia w społeczeństwie informacyjnym*, Akademia Górniczo-Hutnicza, Kraków.
- Oleński J. [2003], *Ekonomika informacji. Metody*, PWE, Warszawa.
- Olszak C.M. [1988], *Struktury systemów wspomaganie decyzji*, w: *Informatyka w zarządzaniu przedsiębiorstwem*, INFOGRYF, Szczecin.
- Olszak C.M. [1999], *Metody multimedialne w projektowaniu interfejsu dla potrzeb systemów wspomaganie decyzji*, w: Baborski A. (red.), *Pozyskiwanie wiedzy z baz danych*, Akademia Ekonomiczna, Wrocław.
- Olszak C.M. [2000], *Zarys metodologii multimedialnych systemów wspomaganie decyzji w zarządzaniu*, Akademia Ekonomiczna, Katowice.
- Olszak C.M. [2007], *Tworzenie i wykorzystywanie systemów Business Intelligence na potrzeby współczesnej organizacji*, Akademia Ekonomiczna, Katowice.
- Olszak C.M., Sroka H. [1999], *Multimedia in Decision Support Systems*, w: Zupanic J., Wojtkowski W., Wojtkowski W.G., Wrycza S., *Evolution and Challenges in System Development*, Kluwer Academic–Pleum Publishers, New York.
- Olszak C.M., Ziemba E. [2003], *Systemy Business Intelligence narzędziem wspomagającym pracę menedżerów*, w: Kubiak B.F., Korowicki A. (red.), *Human–Computer Interaction*, Uniwersytet Gdański, Gdańsk.
- Ostrowska T.M. [2002], *Relacyjne systemy bazodanowe — podstawy projektowania i eksploatacji*, Politechnika Warszawska, Warszawa.
- Partyka M.A. [1999a], *Algorytm Quine'a–McCluskeya minimalizacji indywidualnych cząstkowych wielowartościowych funkcji logicznych*, „Studia i Monografie” nr 109, Politechnika Opolska, Opole.

- Partyka M.A. [1999b], *Logika systemów projektowania na przykładzie CAD układów maszynowych*, „Studia i Monografie” nr 105, Politechnika Opolska, Opole.
- Partyka M.A. [2001], *Metodologia projektowania — wybrane zagadnienia projektowania technicznego*, Politechnika Opolska, Opole.
- Pawlak Z. [1982], *Rough Sets*, „International Journal of Information and Computer Sciences” vol. 11, no. 5.
- Pawlak Z., Skowron A. [2007], *Rudiments of Rough Sets*, „Information Sciences” no. 177.
- Piegat A. [1999], *Modelowanie i sterowanie rozmyte*, Exit, Warszawa.
- Pokojski J. (red.) [2003], *Zastosowanie metody case-based reasoning w projektowaniu maszyn*, WNT, Warszawa.
- Power D.J. [2000], *Web-Based and Model-Driven Decision Support Systems. Concepts and Issues*, American Conference on Information Systems, California.
- Power D.J. [2001], *Supporting Decision-Makers: An Expanded Framework*, w: Harriger A. (red.), *e-Proceedings Informing Science Conference*, Kraków.
- Power D.J. [2002], *Decision Support Systems: Concepts and Resources for Managers*, Greenwood Publishing Group, Westport, CT.
- Power D.J. [2003], *Defining Decision Support Constructs*, w: Bui T., Sroka H., Stanek S., Gluchowski J. (red.), *Proceedings on the 7th International Conference of The International Society for Decision Support Systems DSS in the Uncertainty of the Internet Age*, The Karol Adamiecki University of Economics, Katowice.
- Power D.J. [2006], *What Was the First Computerized Decision Support System (DSS)?*, „Decision Support System News” vol. 7, no. 27.
- Power D.J. [2007], *What is the Modern Support System?*, „Decision Support System News” vol. 8, no. 26.
- Prędko B., Słowinski R., Stefanowski J., Susmaga R., Wilk S. [1998], *ROSE — Software Implementation of the Rough Set Theory*, w: Polkowski L., Skowron A. (red.), *Rough Sets and Current Trends in Computing, Lecture Notes in Artificial Intelligence*, vol. 1424, Springer Verlag, Berlin.
- Quillian R.M. [1968], *Semantic Memory*, w: Minsky M. (red.), *Semantic Information Processing*, MIT Press, Cambridge, MA.
- Radośniński E. [1998], *Inteligentne techniki hybrydowe w analizie ekonomicznej firmy*, w: Miśiąg W. (red.), *Symulacja systemów gospodarczych: prace szkoły Antałówka'98*, WSPiZ, Warszawa.
- Riesbeck C., Schank R. [1989], *Inside Case-Based Reasoning*, Taylor & Francis Group, London.
- Robinson J.A. [1965], *A Machine-Oriented Logic Based on the Resolution Principle*, „Communications of the ACM” no. 5.
- Rockart J.F. [1979], *Chief Executives Define Their Own Data Needs*, „Harvard Business Review” vol. 57.
- Rojek I. [2007], *Inteligentny system wspomaganie decyzji dla sterowania siecią wodociagową*, w: *Technologie przetwarzania danych, materiały konferencyjne KKNTPD 2007*, Poznań.
- Rostek K. [2004], *Proces pozyskiwania wiedzy i eksploracji danych w zastosowaniach biznesowych na przykładzie ubezpieczeń*, w: Krupa T. (red.), *Informatyka w przedsiębiorstwie — wybrane zagadnienia*, „Prace Naukowe Organizacja i Zarządzanie Przemysłem” z. 16.
- Rostek K. [2005], *Modelowanie procesów eksploracji danych w zastosowaniach biznesowych na przykładzie ubezpieczeń*, Wydział Inżynierii Produkcji, Politechnika Warszawska, Warszawa, praca doktorska.

- Rostek K. [2008a], *Business Intelligence i Corporate Performance Management (Kompleksowe zarządzanie efektywnością firmy)*, w: Kisielnicki J. (red.), *Informatyka dla przyszłości*, Wydział Zarządzania, Uniwersytet Warszawski, Warszawa.
- Rostek K. [2008b], *Metodyka wdrożenia systemu Business Intelligence w ubezpieczeniach*, w: Marciniak S., Ostaszewski J. (red.), *Nowoczesne instrumenty zarządzania*, Oficyna Wydawnicza SGH, Warszawa.
- Rostek K. [2010a], *Data Analytical Processing In Data Warehouses*, „Foundations of Management. International Journal” vol. 2, no. 1(3), Politechnika Warszawska, Warszawa.
- Rostek K. [2010b], *Projektowanie analitycznych baz danych*, w: Zawila-Niedźwiecki J., Rostek K., Gąsioriewicz A. (red.), *Informatyka gospodarcza*, t. 1, C.H. Beck, Warszawa.
- Rostek K. [2010c], *Projektowanie analitycznych baz danych*, w: Zawila-Niedźwiecki J., Rostek K., Gąsioriewicz A. (red.), *Informatyka gospodarcza*, t. 2, C.H. Beck, Warszawa.
- Roszkowski M. [2004], *Sieci neuronowe jako przykład współczesnej technologii informatycznej. KEI'2004*, PWSZ, Chełm.
- Roy B. [1996], *Multicriteria Methodology for Decision Aiding*, Wolters Kluwer, New York.
- Rozporządzenie Rady (WE) nr 207/2009 z dnia 26 lutego 2009 r. w sprawie wspólnotowego znaku towarowego.
- Rudas I.J. [2002], *Hybrid Systems (Integration of Neural Networks, Fuzzy Logic, Expert Systems, and Genetic Algorithms)*, w: *Encyclopedia of Information Systems*, Academic Press, Waltham, MA.
- Rutkowska D., Pliński M., Rutkowski L. [1999], *Sieci neuronowe, algorytmy genetyczne i systemy rozmyte*, WN PWN, Warszawa.
- Rutkowski L. [2005], *Metody i techniki sztucznej inteligencji*, WN PWN, Warszawa.
- Rutkowski L. [2012], *Metody i techniki sztucznej inteligencji. Inteligencja obliczeniowa*, wyd. 2, WN PWN, Warszawa.
- Rybarski J., Tadeusiewicz R. [2006], *Problem optymalnego doboru firm inwestujących w danym regionie jako informatyczny binarny problem plecakowy*, „Informatyka Teoretyczna i Stosowana” t. 5, nr 9.
- Sadowski W. (red.) [1985], *Elementy ekonometrii i programowania matematycznego*, PWN, Warszawa.
- Scott-Morton M.S. [1971], *Management Decision Systems. Computer-based Support for Decision Making*, Division of Research, Graduate School of Business Administration, Harvard University, Boston.
- Senczyna S. (red.) [2000], *Analiza i modelowanie systemu informacyjnego przedsiębiorstwa*, Politechnika Śląska, Gliwice.
- Setlak G. [2004], *Harmonogramowanie produkcji na wydziale montażu za pomocą algorytmów ewolucyjnych*, „Technologia i Automatyzacja Montażu” nr 2.
- Shim J.P., Warkentin M., Courtney J.F., Power D.J., Sharda R., Carlsson Ch. [2002], *Past, Present and Future of Decision Support Technology*, „Decision Support Systems” vol. 33, Elsevier Science B.V.
- Sikorski M. [2010a], *Interakcja człowiek–komputer*, PJWSTK, Warszawa.
- Sikorski M. [2010b], *Projektowanie interfejsu użytkownika*, w: Zawila-Niedźwiecki J., Rostek K., Gąsioriewicz A. (red.), *Informatyka gospodarcza*, t. 1, C.H. Beck, Warszawa.
- Sikorski M., Garnik I., Redlarski K. [2011], *Badania User Experience w projektowaniu interakcji użytkownik–system*, „Prace Naukowe. Informatyka Ekonomiczna” nr 20, Uniwersytet Ekonomiczny, Wrocław.
- Sikorski M., Wichary M. [2004], *Wygnij swój komputer: najnowsze trendy w projektowaniu interfejsu użytkownika*, „Computerworld” nr 23.



- Silberschatz A., Galvin P.B. [2000], *Podstawy systemów operacyjnych*, WNT, Warszawa.
- Silberschatz A., Korth H.F., Sudarshan S. [2006], *Database System Concepts*, McGraw-Hill, Columbus.
- Skorupska I. [2004], *Metody sztucznej inteligencji w rozwiązywaniu zadań optymalizacji*, w: *Informatyka — sztuka czy rzemiosło (KNWS' 04)*, materiały konferencyjne, Uniwersytet Zielonogórski, Zielona Góra.
- Smith B.C. [1999], w: Honderich T. (red.), *Encyklopedia filozofii*, Zysk i S-ka, Poznań.
- Sobieszczyk T. [2007], *Analiza rentowności w branży ubezpieczeniowej z wykorzystaniem narzędzi Business Intelligence*, DC Business Intelligence Solutions, Sopot–Gdynia.
- Sokołowski S.J. [2003], *Logika w racjonalnym działaniu. Zastosowanie praktyczne*, WSPiZ, Warszawa.
- Sommerville I. [2003], *Inżynieria oprogramowania*, WNT, Warszawa.
- Sprague R.H. [1994], *A Framework for the Development of Decision Support Systems*, w: Gray P., King W., McLean E., Watson H., *Management of Information System*, The Dryden Press, London.
- Sprague R.H., Carlson E. [1982], *Building Effective Decision Support Systems*, Prentice Hall, Englewood Cliffs, NJ.
- Sroka H. [1986], *Systemy wspomaganie decyzji kierowniczych*, Akademia Ekonomiczna, Katowice.
- Sroka H. [1994], *Systemy ekspertowe. Komputerowe wspomaganie decyzji w zarządzaniu i finansach*, Akademia Ekonomiczna, Katowice.
- Sroka H., Stanek S. [1999], *Systemy wspomaganie organizacji*, w: Gołuchowski J., Sroka H. (red.), *Systemy wspomaganie organizacji*, Akademia Ekonomiczna, Katowice.
- Sroka H., Wolny W. (red.) [2009], *Inteligentne systemy wspomaganie decyzji*, Akademia Ekonomiczna, Katowice.
- Stachowiak K. [2002], *Wielokryterialna analiza decyzyjna w badaniach przestrzenno-ekonomicznych*, Bogucki Wydawnictwo Naukowe, Poznań.
- Standish Group [1994–2009], *Chaos Report*, Standish Group Pub., Boston.
- Stanek S. [1999], *Metodologia budowy komputerowych systemów wspomaganie organizacji*, Akademia Ekonomiczna, Katowice.
- Stanek S., Sroka H. [2001], *Building Creativity into OSS Design Framework*, w: *Decision Support in the New Millenium*, „Journal of Decision Systems Special Issue” vol. 10, no. 3–4.
- Stokalski B. [2010], *Zarządzanie ryzykiem procesów związanych z informatyzacją*, w: Zawiła-Niedźwiecki J., Rostek K., Gąsioriewicz A. (red.), *Informatyka gospodarcza*, t. 4, C.H. Beck, Warszawa.
- Stolecki G., Kostrubała S., Twardowski Z. [2003], *The Multidimensional Intelligent Analytical Platform in Hybrid Controlling Decision Support System Development*, w: Abramowicz W. (red.), *Proc. 7th International Conference on Business Information Systems*, Akademia Ekonomiczna, Poznań.
- Straffin P.D. [2004], *Teoria gier*, Scholar, Warszawa.
- Strzykowski S. [2005], *System tworzenia aktualizacji i dostępu do informacji o skazanych elementem wiedzy w zwalczaniu przestępczości*, „Studia i Materiały Polskiego Stowarzyszenia Zarządzania Wiedzą” nr 4.
- Surma J. [2010], *Rola analogii w podejmowaniu decyzji w zarządzaniu strategicznym małymi i średnimi przedsiębiorstwami*, Oficyna Wydawnicza SGH, Warszawa.
- Suttonm R.I., Pleffer J. [2002], *Wiedza a działanie. Przeszkody w wykorzystywaniu zasobów wiedzy w organizacji*, Oficyna Ekonomiczna, Kraków.

- Szarfenberg R. [2007], *Paradoks nieuchronnej niesprawiedliwości w polityce społecznej*, w: Karwat M. (red.), *Paradoksy polityki*, Elipsa, Warszawa.
- Szmuc T. [2002], *Precyzyjna reprezentacji wiedzy niepełnej — zbiory rozmyte i przybliżone*, „Geoinformatica Polonica” t. 4.
- Szyjewski Z. [1993], *Wspomaganie tworzenia systemów informatycznych zarządzania*, Uniwersytet Szczeciński, Szczecin.
- Traktat Światowej Organizacji Własności Intelektualnej o Prawie Autorskim, Genewa, 20 grudnia 1996 r. (Dz.U. z 2005 r., nr 3, poz. 12).
- Tadeusiewicz R. [1993], *Sieci neuronowe*, seria: *Problemy Współczesnej Nauki i Techniki. Informatyka*, Akademicka Oficyna Wydawnicza RM, Warszawa.
- Triantaphyllou E. [2000], *Multi-Criteria Decision Making Methods: A Comparative Study*, Kluwer Academic Publisher, Dordrecht.
- Trzaskalik T. [2006], *Metody wielokryterialne na polskim rynku finansowym*, PWE, Warszawa.
- Tyszka T. [1986], *Analiza decyzyjna i psychologia decyzji*, PWN, Warszawa.
- Tyszka T., Zaleśkiewicz T. [2002], *Racjonalność decyzji*, PWE, Warszawa.
- Ullman J.D., Widom J. [2002], *Podstawowy wykład z systemów baz danych*, WNT, Warszawa.
- Ustawa z dnia 4 lutego 1994 r. o prawie autorskim i prawach pokrewnych (Dz.U. z 2000 r., nr 80, poz. 904).
- Ustawa z dnia 30 czerwca 2000 r. — Prawo własności przemysłowej (tekst jednolity: Dz.U. z 2003 r., nr 119, poz. 1117 z późn. zm.).
- Valenta M., Śnieżyński B., Zygmunt A. [2000], *Consus — eksperymentalny shell regułowych systemów ekspertowych*, w: Bubnicki Z., Grzech A. (red.), *Inżynieria wiedzy i systemy ekspertowe*, t. 2, Politechnika Wroclawska, Wrocław.
- Van thienen J., Wets G. [1995], *Integration of the Decision Table Formalism with a Relational Database Environment*, „Information Systems” vol. 20, no. 7.
- Von Neumann J., Morgenstern O. [1944], *Teoria gier i zachowania ekonomicznego*, Princeton University Press, Princeton NJ.
- Wabnic D. [2007], *Sztuczne sieci neuronowe*, PWSZ, Leszno.
- Watkins M. [2005], *Sztuka negocjacji w biznesie. Innowacyjne podejścia prowadzące do przelomu*, Helion, Gliwice.
- West D. [2001], *Introduction to Graph Theory*, wyd. 2, Prentice Hall—University of Illinois, Urbana.
- Wierzchoń S.T. [2001], *Sztuczne systemy immunologiczne. Teoria i zastosowania*, Akademicka Oficyna Wydawnicza EXIT, Warszawa.
- Wilk A., Müller L. [1997], *Teoria podobieństwa w badaniach modeli fizycznych i matematycznych*. „Monografia”, Politechnika Śląska, Gliwice.
- Witkowski T. [2010], *Systemy informatyczne wspomaganie podejmowania decyzji*, w: Zawiła-Niedźwiecki J., Rostek K., Gąsioriewicz A. (red.), *Informatyka gospodarcza*, t. 3, cz. 2 (*Informatyka w zarządzaniu wiedzą*), C.H. Beck, Warszawa.
- Wolny W., Zadora P. [2008], *Redefinicja idei inteligentnych systemów wspomaganie decyzji*, XXIII Konferencja pt. *Systemy Wspomaganie Organizacji*, Katowice.
- Wołoszyn P. [2010], *Obserwacja zachowań społecznych w multiagentowym modelu populacji*, „Zeszyty Naukowe AE w Krakowie” nr 814.
- Wołoszyn P., Wołoszyn J. [2011], *Dobór par sprzedawca-kupujący a równowaga ofert w symulacyjnym modelu serwisu aukcyjnego*, „Zeszyty Naukowe AE w Krakowie” nr 873.

- Wrona S., Zadora P. [1997], *Przetwarzanie wiedzy pochodzącej z przeszłości w celu rozwiązania problemów klasyfikacyjnych*, w: Sroka H., Stanek S., *Inteligentne systemy wspomagania decyzji w zarządzaniu. Transformacje systemów*, Akademia Ekonomiczna, Katowice.
- Wrycza S. [1999], *Analiza i projektowanie systemów informatycznych zarządzania. Metody, techniki, narzędzia*, WN PWN, Warszawa.
- Zabawa J. [2005], *Podejście hybrydowe w analizie ekonomicznej przedsiębiorstwa*, Wydział Informatyki i Zarządzania, Politechnika Wrocławska, Wrocław, praca doktorska.
- Zadeh L. [1965], *Fuzzy Sets*, „Information and Control” no. 8.
- Zadeh L. [1994], *Soft Computing and Fuzzy Logic*, „IEEE Software” vol. 11, no. 6.
- Zadora P. [1999], *Wnioskowanie na podstawie przypadków jako metoda wspomagająca proces podejmowania decyzji*, Akademia Ekonomiczna, Katowice, [www.swo.ae.katowice.pl/\\_pdf/139.pdf](http://www.swo.ae.katowice.pl/_pdf/139.pdf), praca doktorska pod kierunkiem prof. zw. dr. hab. H. Sroki.
- Zduńczuk J., Przystupa W. [2008], *Wykorzystanie algorytmów genetycznych i mrówkowych w problemach transportowych*, „Inżynieria Rolnicza” nr 7(105).
- Zhang Z., Zhang C. [2002], *An Agent-Based Hybrid Intelligent System for Financial Investment Planning*, Proceedings of PRICAI 2002, LNAI-2417, Springer Verlag.
- Ziomba P., Piwowarski M. [2008], *Metody analizy wielokryterialnej we wspomaganii porównywania produktów w Internecie*, „Metody Informatyki Stosowanej” nr 2.



# Indeks

## A

- Agile Modelling zob. Modelowanie zwinne
- AHP 264–265
- Aitech DSS 337–338
- Algorytm
  - genetyczny (AG) 157, 282–289, 298–303
    - budowa i schemat funkcjonowania 298–303
    - krzyżowanie uśredniające 286–288
    - krzyżowanie wymieniające 286–288
    - model decyzyjny 301–302
    - mutacja 284–289, 298–301
    - schemat działania 283–289
    - selekcja osobników 283–285, 298–302
    - zastosowania 303
  - mrówkowy (AM) 278, 282, 289–290, 303–308
    - budowa i schemat funkcjonowania 303–305
    - model decyzyjny 305–307
    - przykłady zastosowań 307–308
    - zasady (schemat) działania 289–290, 303–305
    - zastosowania 307–308
- Algorytmy
  - optymalizacyjne 188
    - wnioskowania logicznego 60–62
- Analityczna baza danych (BD) 71, 168,
- Analityczne przetwarzanie danych 15, 18, 59, 146, 169, 173–174, 178–180, 195, 200
- Analiza
  - decyzyjna 193, 210–211
    - wielokryterialna 256–257
    - typu „co, jeśli” 62, 216–218
    - wielowymiarowa 58
- Analogia naturalna 282–290
- Aplikacja

- konstruktor 167–168
- kreator 167–168

## B

- Baza
  - danych (BD) 62–72, 153, 198, 218–229
    - analityczna 71, 168
    - architektura 63–64
    - definicja 62–70
    - fakty 70–72
    - model fizyczny 66–67
    - model konceptualny 64–65
    - przetwarzanie transakcyjne 68
    - reguły decyzyjne 70–72
    - relacyjna 153, 198, 218–229
    - transakcyjna 63, 68, 168–170, 315
  - wiedzy (BW) 155–159, 163–164, 182–188, 273
    - fakty 155, 183
    - formuły rachunku zdań 163–164
    - metody reprezentowania wiedzy 157
    - przesłanka 158–159, 185, 273
    - reguła decyzyjna 155
    - regułowa 186–188
    - rodzaje 156–157
    - twierdzenia 157–158
- Bezpieczeństwo
  - interfejsu 85–86
  - teleinformatyczne 270–271
- Blackboard Systems 194–195
- Business Intelligence (BI) 313–323, 340–347
  - architektura 315–317, 343–344
  - cechy funkcjonalne 341–343
  - charakterystyki eksploatacyjne 346–347
  - eksperymenty symulacyjne 344–346

koncepcja i istota działania 314–315  
zarządzanie ryzykiem ubezpieczeniowym 340

**C**

Case Based Reasoning zob. Metoda CBR  
CLIPS 198, 229, 231–232  
COCOMO 244

**D**

Dane 182–183  
eksploracja 173–174, 201  
parsowanie 196  
przetwarzanie 18–20, 70, 151–152, 166, 172, 182, 311  
analityczne 15, 18, 59, 146, 169, 173–174, 178–180, 195, 200  
składnica 169, 175–177  
Data mining zob. Przetwarzanie danych  
Decyzje 22–24, 98–99, 134–141  
alternatywne, konstruowanie zbioru 134–141  
metoda hierarchicznego wspomaganie 98–99  
typologia 24  
Delphi 232–233  
Drzewo decyzyjne 95–97, 181, 187–188, 216  
procedura dekompozycji 97  
Dyferencjał semantyczny 206–207

**E**

EIS zob. System informowania kierownictwa  
Eksperymenty symulacyjne 345–346  
techniki kalkulacyjne 58–59  
Eksploracja danych 173–174, 201  
Electre 265  
Encja 64–66, 171–172  
ETL 315

**F**

Fakty 70–71, 152, 183  
przetwarzanie 70–71  
tabela 152  
Fazy procesu decyzyjnego 21–24  
Feromon zob. Ślad feromonowy  
Formuły rachunku  
predykatów 281  
zdań 163–164  
Funkcja boolowska 134–138

**G**

Graf 90–96, 131–133, 290–291, 304  
definicja 91–92  
dwudzielny 131–133  
Grafy  
sposoby zadania 90–96

**H**

Heurystyka zob. Metody heurystyczne  
HSWD zob. Hybrydowy SWD  
Hurtownia danych (HD) 151–154, 168–182  
architektura gwiazdy 151–153  
architektura konstelacji faktów 153–154  
definicje 168–170  
model fizyczny 172  
model logiczny 171  
model pojęciowy 170  
modelowanie fizyczne 170–172  
modelowanie logiczne 170–172  
projektowanie 174–178  
przetwarzanie analityczne 172–174, 247  
schemat płatka śniegu 152–153  
struktura i istota działania 168–170  
tabela faktów 152  
Hybrydowe modele decyzyjne 308–311  
idea i struktura 308–309  
rozwiązania 310–311  
typy hybrydyzacji 309  
Hybrydowy  
SWD 313, 334–338  
architektura 335–336  
problemy projektowania i eksploatacji 336  
zastosowania 336–338  
system sztucznej inteligencji 194–195

**I**

Incyzyn kartezjański zbiorów rozmytych 106  
Implikant prosty 135–137  
Informacje 182–183  
Inteligencja obliczeniowa 262  
Inteligentny SWD 261–262  
Interfejs  
bezpieczeństwo 85–86  
definicja 75  
użytkownika 19, 69, 74–80, 83–88, 165–168, 324–325, 349  
definicja 74–75

- język specyfikacji wymagań funkcjonalnych 77–80  
 projektowanie 83–88, 165–168  
 wymagania eksploatacyjne 80–83  
 wymagania funkcjonalne 75–80  
 Inżynieria wiedzy, narzędzia 198–201  
 Iteracja 59–60, 245–246, 284–285, 304
- J**
- Język komunikacji relacyjnej bazy danych 153, 198, 218–229
- K**
- Klasy 184–185  
 Klasyfikacja  
 metode sztucznej inteligencji 279–280  
 modele symulacyjne 29–33  
 Klatki 184–185  
 Klauzula 46, 185, 281–282  
 Horna 46  
 Kodeks etyki komputerowej 270–271  
 Konsola 166  
 Konstruktor aplikacji 167–168  
 Konstruowanie zbioru decyzji alternatywnych 134–141  
 Korelacja rangowa 203–207  
 współczynnik 206–207  
 Kostka 135  
 Kreator aplikacji 167–168  
 Kreatywny SWD, architektura 194  
 Krzyżowanie 286–288, 300–301  
 uśredniające 286–288  
 wymieniające 286–288
- L**
- Licencja 270  
 LISP 231, 327  
 Logiczne metody optymalizacyjne 46–51  
 Logika  
 kwantyfikatorów 281  
 rachunku zdań 157  
 Logistyka 242–243
- M**
- Maksymalizacja subiektywnie oczekiwanej użyteczności zob. SEU  
 Mandarax 229, 232  
 Manualne metody pozyskiwania wiedzy 199  
 MAUT 264–265  
 Mądrość 182–183  
 Metaklasa 156, 185–186,  
 Metoda  
 analogii 199, 254, 274, 282, 332  
 hierarchicznego wspomaganie decyzji 98–99  
 Oceany Wykorzystania Maszyn (MOWM) 331–333, 349–350  
 (system) CBR 271–272, 310–311  
 Metody  
 ewolucyjne 283  
 heurystyczne 55–58  
 strategia najpierw najlepszy 57  
 strategia w głąb 56  
 strategia wszcz 56–57  
 strategia zachłanna 57  
 optymalizacji wielokryterialnej 263–264  
 optymalizacyjne 39–57, 188, 247  
 logiczne 46–51  
 statystyczne 51–55, 188  
 zastosowanie 39–57  
 rangowe 207–208  
 reprezentowania wiedzy w bazie wiedzy (BW) 157  
 sztucznej inteligencji (AI) 193–194, 278–290  
 klasyfikacja 279–280  
 Miękkie techniki obliczeniowe zob. Obliczenia miękkie  
 Model  
 abstrakcyjny 30  
 cyklu życia SI 202–203  
 decyzyjny 22  
 deterministyczny 30  
 fizyczny 30  
 normatywny 30  
 statyczny 30  
 symboliczny 30–31  
 systemowy przedsiębiorstwa 240–241  
 średniej ruchomej ważonej 32–33  
 Modele  
 idealne 30  
 materialne 29  
 optymalizacyjne 35–42  
 sformalizowane 30  
 symulacyjne, klasyfikacja 29–33  
 wnioskowania 109  
 Modelowanie zwinne 76–77  
 MS Excel 208–209  
 Solver 37–39  
 Mutacja 284–289, 298–301  
 kodu binarnego 288–289

**N**

## Narzędzia

- CASE 78–79, 334
- inżynierii wiedzy 198–201

**O**

- Obiekt 158, 162–163, 183–186, 212–213, 280–282
- Obliczenia
  - miękkie 262, 308–309
  - twarde 262
- Obszerzacja ucięta 344
- Odkrywanie wiedzy z danych 198–201
- OLAP zob. Analityczne przetwarzanie danych, Wielowymiarowa kostka
- Operacje analizy wielowymiarowej 58–60
- Optymalizacja wielokryterialna 263
  - metody 263–264

**P**

- Parsowanie danych 196
- Patent 269
- PC–Shell 229–230, 235, 327–328
- Platforma Eclipse 87, 165, 169
- Podejmowanie decyzji
  - psychologiczne aspekty 266–268
  - stopień niepewności 266
  - stopień swobody wyboru 267
- Podgraf 91–92
- Podkostka 135
- Pojęcia (koncepty) zob. Klasy
- Postać normalna 134–137, 228
- Poziomy
  - decyzji
    - proces zarządzania 25
    - stopień ryzyka decyzyjnego 26
    - stopień strukturalizowania 25–26
    - procesów decyzyjnych 24–27
- Pozyskiwanie wiedzy
  - manualne metody 199
  - półautomatyczne metody 199–200
- Półautomatyczne metody pozyskiwania wiedzy 199–200
- Prawo własności intelektualnej 268–270
- Predykat (y) 47, 60, 69, 130–131, 160–161, 164, 189, 280–282
  - rachunek 69, 160–161, 189, 281
  - tautologia logiki 164, 281–282
- Problemy decyzyjne 21–22, 263–264
  - hierarchiczne 90–99

- koncepcja, zasady charakteryzacji 125–133
  - Markowa 210
  - płaskie 90–99
  - zasada charakteryzacji w modelowaniu 125–141
- Procedura
  - dekompozycji drzewa decyzyjnego 97
  - wielokryterialnego wyboru 256–257
- Proces
  - decyzyjny 23–27, 34–36
    - fazy 21–24
    - modele i symulacje 34–36
    - poziomy 24–27
  - przygotowania BD na potrzeby SWD 69–70
- Projektowanie
  - interfejsu użytkownika 83–88, 165–168
  - rozwiązań 317–323
- Promethee 265
- Prototypowanie SWD 202–203
- Przekrój (przecięcie) zbiorów rozmytych 103
- Przeźreń tolerancji 272–273
- Przetwarzanie
  - analityczne hurtowni danych (HD) 172–174, 247
  - danych 18–20, 70, 151–152, 166, 172, 182, 311
  - faktów 70–71
  - transakcyjne baz danych (BD) 68
  - wiedzy 120, 124, 156, 272–273

**R**

- Rachunek
  - predykatów 60, 160–161, 189, 281
    - formuły 281
  - zdań
    - formuły 163–164
    - logika 157
- Ramki 184–186
- Reguła 46–47, 185
  - decyzyjna 55, 60, 155, 267–268
  - dysjunkcji 60–61
  - koniunkcji 60
- Regułowa
  - baza wiedzy (BW) 186–188
  - reprezentacja wiedzy 158–160
- Regułowy system ekspertowy (SE) 189, 197–198, 229
- Reguły
  - stosowane spontanicznie 60
  - wnioskowania 108
- Relacja
  - binarna 248
  - rozmyta 107



- równoważności 248–249
- Relacyjna baza danych (BD), język komunikacji 153, 198, 218–229
- Reprezentacja wiedzy 183–185, 210, 280, 334
- regułowa 158–160
- Równowaga 53
- Nasha 53
- Równoważenie syntaktyczne 129
- ## S
- Selekcja osobników 283–285, 298–302
- Semantyka
- logiki 130
- refleksywna 130–131
- rzutowa 130–131
- SEU 267
- Sieci
- neuronowe 282–283, 290–297
- model decyzyjny 296–297
- strojenie (uczenie) 294–296
- struktura 291–294
- sztuczne 282–283
- zastosowania 297
- semantyczne 158–164, 183–184
- SIK zob. System informowania kierownictwa
- Składnica danych 169, 175–177
- SMART 264
- SoDIS 270–271
- SPHINX 229–230, 235
- Sposoby zadania grafów 90–96
- SQL 153, 198, 218–229
- Statystyczne metody optymalizacyjne 51–55, 188
- Stwierdzenia 157–158, 183
- Suma zbiorów rozmytych 104
- SWD
- architektura 150–168, 194–197
- definicje 13–19
- funkcjonowanie organizacji 259–260
- hybrydowy 313, 334–338
- inteligentny 261–262
- kreatywny 194
- metoda systemu idealnego (IDEALS) 147–148
- metody projektowania 192–197
- model kaskadowy 251–252
- model sieciowy 149–150
- model spiralny 251
- narzędzia projektowania 198–218
- obszary zastosowań 261–271
- prawne aspekty zastosowań 268–271
- projektowanie, języki programowania wysokiego poziomu 229–235
- systemy zarządzania 219–229
- prototypowanie 202–203
- przyrostowa strategia projektowania 148–149
- rozwój 19–21
- spiralna strategia projektowania 149
- strategie projektowania 144–149
- struktura 15–19
- systemy zarządzania 219–229
- tworzenie modelu 251
- wdrożenie systemu 241–246
- weryfikacja prototypów 203–208
- wymagania eksploatacyjne 74, 80–83
- wymagania funkcjonalne 74–80
- SWOT 260
- Symulacja komputerowa 34–36
- System
- doradcy Ma'ayan 76, 329–330
- ekspertowy (SE) 76, 89, 189, 197–198, 229, 273–276, 310–311, 323–334, 348–352
- architektura 323–327, 350–352
- cechy funkcjonalne 348–350
- eksperymenty symulacyjne 352
- projektowanie 327–329
- przykłady zastosowań 329–334
- regułowy 189, 197–198, 229
- zarządzanie produkcją rolniczą 347–348
- informacji zarządczej (BI) 18, 314
- informacyjny
- projektowanie ewolucyjne 240
- projektowanie kaskadowe 240
- projektowanie przyrostowe 240
- projektowanie spiralne 240
- informowania kierownictwa (EIS) 14, 18
- wieloagentowy 290
- Systemy
- rozproszone 69–70
- wspomagania decyzji zob. SWD
- wspomagania kierownictwa (SWK) 14
- Szeregi czasowe 32–33
- Sztuczne sieci neuronowe 282–283
- ## Ś
- Ślady feromonowe (zapachowe) 278, 289–290, 303–304, 306
- ## T
- Tabela faktów 152
- Tablica
- decyzyjna (TD) 118–125, 210–216
- definicja 118–119
- redukt 123

reguła deterministyczna 120  
 reguła niedeterministyczna 120  
 Quine'a 137–138  
 semantyczna 132–133, 139–140  
 Tautologia 164  
 logiki predykatów 164, 281–282  
 Techniki iteracyjne zob. Iteracja  
 Teoria  
 chaosu 262  
 gier 51  
 sztucznej inteligencji 278–279  
 użyteczności 264  
 zbiorów przybliżonych 197  
 Terminy 160–161, 281  
 T-norma 106–107  
 Transakcja 68  
 Transakcyjna baza danych (BD) 63, 68, 168–170, 315  
 Typologia decyzji 24  
 Typy uczenia się 295–296

## U

Uczenie maszynowe 262  
 Uczenie się, typy 295–296  
 Uniwersalny Model Interfejsu Użytkownika 76, 82–83  
 UTA 265

## W

Wdrożenie SWD 241–246  
 przygotowanie finansowe 242–243  
 przygotowanie logistyczne 242–243  
 zarządzanie projektem 244–246  
 Wektor binarny 248, 285  
 Wektory wiedzy 160  
 Wiedza 120, 124, 156, 160, 182–183, 272–274  
 ekspercka 273–274  
 przetwarzanie 120, 124, 156, 272–273  
 reprezentacja 183  
 systemowa 273–274  
 wektory 160  
 Wielokryterialna  
 analiza decyzyjna 256–257  
 procedura wyboru 256–257

Wielokryterialne podejmowanie decyzji 259, 263–266  
 model funkcjonalny 264–265  
 model relacyjny 265–266  
 Wielowymiarowa kostka OLAP 178  
 Własność intelektualna  
 prawo 268–270  
 zarządzanie 268–271  
 Wnioskowanie  
 „do przodu” 164  
 logiczne, algorytmy 60–62  
 probabilistyczne 262  
 „wstecz” 164  
 Współczynnik  
 korelacji rangowej 206–207  
 stopnia pewności (CF) 158, 183  
 Współczynniki sezonowe 32

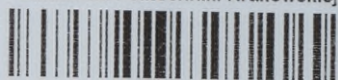
## Z

Zadanie  
 optymalizacji wielokryterialnej 263–264  
 optymalizacyjne 301–302  
 Zarządzanie własnością intelektualną 268–271  
 Zbiory  
 przybliżone 113–128  
 operacje wnioskowania 118–124  
 procesy decyzyjne 124–125  
 redukt 117–118, 122–123  
 relacja nierozróżnialności 114–116  
 teoria 113  
 współczynnik aproksymacji 116–117  
 rozmyte 99–113  
 definicja 99–100  
 funkcja przynależności 100–101  
 iloczyn kartezjański 106  
 nośnik 101  
 operacje 103–106  
 operacje wnioskowania 108–110  
 przekrój (przecięcie) 103  
 reguły wnioskowania 108  
 relacje rozmyte 106–108  
 suma 104  
 własności 101–103  
 Zbiór Pareto 249–250



Cykl podręczników akademickich dla kierunku „zarządzanie i inżynieria produkcji” w Polsce kompendium wiedzy dla studentów w zakresie kształcenia. Poszczególne podręczniki obejmują całość wiedzy z zakresu i pytaniami kontrolnymi. Trudu przygotowania cyklu poświęcił zespół wykładowców na kierunku „zarządzanie i inżynieria produkcji”.

Biblioteka Politechniki Krakowskiej



II-362132

Czyt. Gł.

Prof. Ryszard  
Redaktor naukowy cyklu

Celem podręcznika jest zarys problematyki systemów wspomagania decyzji (SWD) zarówno w zakresie teoretycznych podstaw konstrukcji i budowy, jak i praktycznych aspektów metod ich wdrażania oraz użytkowania. Utrwaleniu wiedzy służą pytania i zadania kontrolne na końcu każdego rozdziału. Szeroki zakres źródeł literaturowych umożliwi samodzielne pogłębienie zawartej w nich wiedzy. Podręcznik jest przeznaczony głównie dla studentów zarządzania i inżynierii produkcji studiów I i II stopnia na uczelniach wyższych różnych typów, a także dla studentów logistyki, administracji i kierunków pokrewnych. Może zainteresować również menedżerów w firmach i praktyków gospodarczych.

**Dr hab. inż. Waldemar Bojar, prof. UTP** jest kierownikiem Katedry Inżynierii Zarządzania Wydziału Zarządzania UTP w Bydgoszczy. Jest także członkiem Komitetu Inżynierii Produkcji PAN przy Wydziale IV. Pełni funkcję redaktora naczelnego „Studies & Proceedings of Polish Association for Knowledge Management” oraz przewodniczącego Oddziału Polskiego Towarzystwa Zarządzania Produkcją w Bydgoszczy. Działalność naukowa obejmuje m.in.: systemy wspomagania decyzji, a zwłaszcza zastosowania metod sztucznej inteligencji i systemów ekspertowych w zarządzaniu produkcją, tworzenie i weryfikację dziedzinowych baz danych oraz baz wiedzy dotyczących wyposażenia technicznego i zasobów przedsiębiorstw oraz procesy dostosowawcze rolnictwa i agrobiznesu do współczesnych wyzwań.

**Dr inż. Katarzyna Rostek** jest adiunktem na Wydziale Zarządzania Politechniki Warszawskiej. Specjalizuje się w zagadnieniach związanych z systemami wspomagania decyzji, analityką biznesową oraz bazami i hurtowniami danych. W tym zakresie prowadzi zajęcia dydaktyczne oraz uczestniczy w realizacji projektów naukowo-badawczych oraz badawczo-rozwojowych. Jest członkiem Rady Wydziału Zarządzania, Polskiego Towarzystwa Zarządzania Produkcją oraz Stowarzyszenia Polish Sybase User Group. Opublikowała kilkadziesiąt artykułów w recenzowanych czasopismach naukowych oraz monografiach. Jest członkiem zespołu redakcyjnego „Public Administration Research”. Otrzymała indywidualne i zespołowe nagrody naukowe i dydaktyczne JM Rektora Politechniki Warszawskiej.

**Dr hab. Leszek Knopik, prof. UTP** pracuje w Katedrze Informatyki w Zarządzaniu, na Wydziale Zarządzania UTP w Bydgoszczy. Jest członkiem Polskiego Towarzystwa Zarządzania Produkcją. Zainteresowania naukowe dotyczą zastosowania metod probabilistycznych, a zwłaszcza teorii niezawodności, metod statystyki matematycznej i badań operacyjnych w zagadnieniach badania efektywności działania złożonych obiektów technicznych. Działalność naukowa obejmuje także problemy modelowania procesów i systemów eksploatacji oraz sterowania procesami realizowanymi w złożonych systemach eksploatacji w kontekście maksymalizacji zysków z pracy systemu. Prowadzi badania dotyczące niezawodności, gotowości i efektywności działania systemów transportowych. Jest autorem ponad 60 prac naukowych opublikowanych w kraju i za granicą.

Biblioteka Politechniki Krakowskiej



100000336374

ISBN 978-83-208-2076-8



9 788320 820768 >

Księgarnia internetowa PWE  
www.pwe.com.pl

Cena zł 64,90  
(w tym 5% VAT)