

WIKAREK Jarosław <sup>1</sup>

## ASPEKTY IMPLEMENTACYJNE SYSTEMU DO WSPOMAGANIA DECYZJI

*W artykule przedstawiono założenia i podstawowe struktury systemu wspomaganie decyzji w postaci dodatkowych warstw zintegrowanego systemu zarządzania klasy ERP/SCM. Każda decyzja systemu może być interpretowana jako odpowiedź na odpowiednio sformułowane pytanie, które spełnia ograniczenia. Innowacyjnym mechanizmem zastosowanym w proponowanym podejściu jest automatyczna generacja zapytań w postaci predykatów CLP lub modeli MIP.*

## IMPLEMENTATION ASPECTS OF THE DECISION SUPPORT SYSTEM

*The paper presents the concepts and the basic structure of decision support system in the form of the additional layers of an ERP/SCM system. Any decision can be interpreted as a response to the properly formulated question that meets the existing constraints. Novel mechanism was introduced to the automatic generation of questions in the form of CLP (Constraint Logic Programming) predicates or MIP (Mixed Integer Programming) models.*

### 1. WPROWADZENIE

W artykule przedstawiono aspekty implementacyjne wraz z podstawowymi strukturami systemu wspomaganie decyzji zarządzania łańcuchem dostaw (SCM-*Supply Chain Management*). Ogólną koncepcję systemu wspomaganie decyzji dla SCM zaproponowano w [3]. Zaprezentowana koncepcja została zaimplementowana w oparciu o środowisko SQL i zintegrowana z solverami MIP (Mixed Integer Programming) [1] i CLP (Constraint Logic Programming) [2] w postaci wielowarstwowej aplikacji bazy danych.

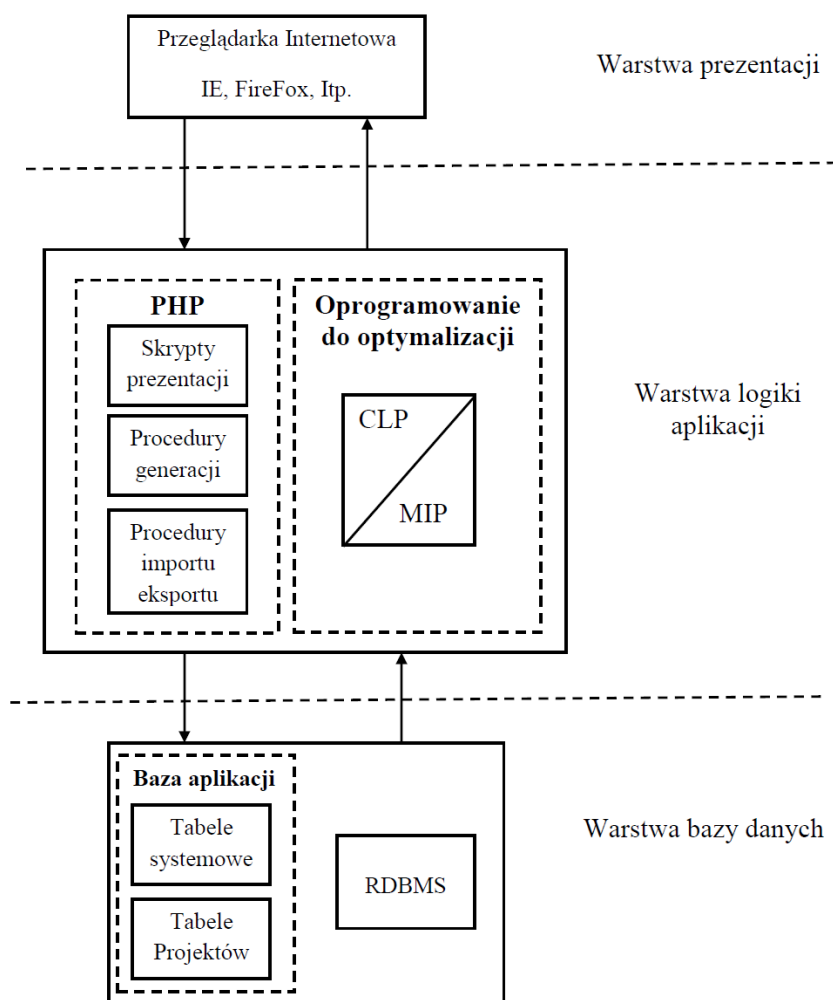
### 2. ARCHITEKTURA SYSTEMU WSPOMAGANIA DECYZJI

Do implementacji systemu wspomaganie decyzji (SWD) zaproponowano architekturę wielowarstwową w postaci trójwarstwowej aplikacji bazy danych. Jest ona powszechnie stosowaną architekturą w aplikacjach internetowych, stąd jej duża popularność. Architektura trójwarstwowa wprowadza logiczny; funkcjonalny podział

---

<sup>1</sup> Politechnika Świętokrzyska, Wydział Elektrochniki Automatyki i Informatyki; 25-314 Kielce Al. 1000 PP 7.  
tel: +48 42 3424200 Fax: +48 41 3424214, e-mail: j.wikarek@tu.kielce.pl

aplikacji na trzy warstwy: warstwę podstawową (bazy danych) – system zarządzania bazą danych (SZBD) wraz z tabelami, warstwę pośrednią (logiki aplikacji) – serwer WWW wraz z logiką przetwarzania, logiką całej aplikacji oraz warstwę końcową (prezentacji) – klienta (komputer lub inne urządzenie obsługujące przeglądarkę internetową). Na Rys.1. przedstawiono schemat architektury dla SWD zarządzania łańcuchem dostaw, którego koncepcję funkcjonowania oraz podstawowe założenia zawarto w [3].



Rys.1. Trójwarstwowa architektura SWD

## 2.1 Warstwa bazy danych

Warstwę bazy danych tworzą oprócz SZBD dwa zbiory tabel. Są to tabele systemowe (TS) oraz tabele modeli (TM). W tabelach systemowych, których schemat – Entity Relationship Diagram (ERD) pokazano na Rys.2. a strukturę szczegółową tabel wraz z kluczami w Tab. 1. zapisywana jest struktura modeli decyzyjnych. Na podstawie danych z tych tabel generowane są odpowiednie pliki tekstowe z modelami decyzyjnymi. Format generowanych plików jest związany również z odpowiednimi wpisami w tabelach (TS). Przykładowe fragmenty tabel (TS) z wybranymi wpisami przedstawiono odpowiednio w Tab. 2, Tab.3, Tab.4, Tab.5, Tab.6. W obecnej wersji SWD modele generowane są dla dwóch środowisk. Pierwsze to środowisko CLP reprezentowane przez system (solver) ECLIPSE [4]. Drugie to środowisko MIP w którym zastosowano oprogramowanie (solver) LINGO [5].

Tab. 1. Tabele systemowe (TS).

Tabela	Opis	Nazwa pola	Klucz	Opis kolumny
sys_tysl	Typy linii	KD_IDLIN	PK	Identyfikator typu linii
		ID_NAZWA		Nazwa typu linii
sys_wynik	Wynik zadziałania	KD_WYNIK	PK	Identyfikator typu wyniku
		ID_OPIS_		Opis typu wynik
gen_proje	Projekty w systemie	KD_IDPRO	PK	Identyfikator projektu
		ID_NAZWA		Nazwa projektu
		ID_SKROT		Skrót nazwy
		ID_OPIS_		Opis projektu
gen_funkc	Funkcje w czasie obsługi projektu	KD_IDPRO	PK(FK)	Identyfikator projektu
		KD_IDFUN	PK	Identyfikator funkcji
		FD_WYNIK	FK	Identyfikator typu wyniku
		ID_OPIS_		Opis funkcji
gen_fu_li	Kody funkcji w czasie obsługi projektu	KD_IDPRO	PK(FK)	Identyfikator projektu
		KD_IDFUN	PK(FK)	Identyfikator funkcji
		KD_ID_KO	PK	Numer linii
		FD_IDLIN	FK	Identyfikator typu linii
		ID_DANA_		Kod linii
gen_tabel	Tabele projektu	KD_IDPRO	PK(FK)	Identyfikator projektu
		KD_ID_TA	PK	Nazwa tabeli w projekcie
		ID_ID_AL		Ogólna nazwa tabeli
		ID_OPIS_		Opis tabeli

Tab. 2. Tabela gen\_funkc

KD_IDPRO	KD_IDFUN	FD_WYNIK	ID_OPIS_
00001	00001	00001	'Generacja modelu MIP'
00001	00002	00001	'Generacja dane CLP'
00001	00003	00001	'Funkcje do modelu CLP'
00001	00004	00001	'Generacja szukaj do CLP'
.....	.....	.....	.....

Tab. 3. Tabela *gen\_proje*

KD_IDPRO	ID_NAZWA	ID_SKROT	ID_OPIS
00001	'Transport'	'Zagadnienie Transportowe'	'Zagadnienie Transportowe'
00002	'Projekty'	'Zarządzanie projektami'	'Zarządzanie projektami'
00003	'Łańcuch'	'SCM'	'Zarządzanie łańcuchem dostaw'
.....	.....	.....	.....

Tab. 4. Tabela *sys\_wynik*

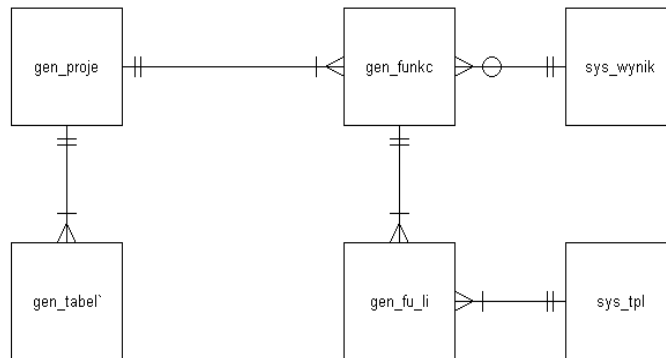
KD_WYNIK	ID_OPIS
00001	'Wynik do pliku'
00002	'Wynik z pliku'

Tab. 5. Tabela *gen\_fu\_li*

KD_IDPRO	KD_IDFUN	KD_ID_KO	FD_IDLIN	ID_DANA
00004	00001	00010	00008	modele2/dystrybucja.lng
.....	.....	.....	.....	.....
00004	00001	00030	00003	select kd_idmag from uni_maga
.....	.....	.....	.....	.....
00004	00001	00170	00002	select id_zapot from uni_zapo order by kd_prod, kd_klie

Tab. 6. Tabela *sys\_typ*

KD_IDLIN	ID_NAZWA	KD_IDLIN	ID_NAZWA
00001	'Tekst do przepisania'	00008	'Utwórz plik'
00002	'Licz wiersz.'	00009	'Zamknij plik'
00003	'Wynik kursor'	00010	'Kursor pętla'
00004	'Funkcja do uruchom'	00011	'Pętla dwuwymiarowa'
00005	'Program zewnętrzny'	00012	'Pęta dwu odwrotna'
00006	'Dane do bazy'	00013	'Import 2 wymiary'
.....	.....	.....	.....



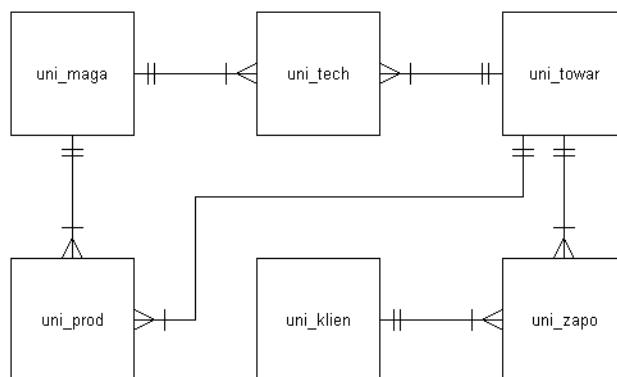
Rys. 2. Schemat ERD dla tabel systemowych (TS) systemu wspomagania decyzji (uproszczona notacja Martin'a).

Zbiór tabel modeli (TM) uzupełnia warstwę bazy danych SWD. Ich znaczenie jest bardzo ważne bowiem dane zawarte w tych tabelach uzupełniają i konkretyzują model decyzyjny. Wpisy w tabelach systemowych (TS), zmieniają się dosyć rzadko. Tak naprawdę tylko w dwóch przypadkach. Pierwszy dotyczy w ogóle wprowadzenia danego modelu decyzyjnego, drugi kiedy zmienia się struktura modelu np. dochodzą nowe ograniczenia lub zmienia się ich charakter. Natomiast wpisy w tabelach modeli (TM) ulegają zmianie bardzo często. Zasilane są bowiem danymi operacyjnymi z systemów zintegrowanych, gdzie dane zmieniają się na bieżąco. Dodatkowo mogą być uzupełnione danymi do symulacji decyzji. Na podstawie różnych serii danych z tabel modeli (TM) mogą być generowane różne decyzje tego samego typu /np. codziennych realizacji dostaw, symulowanych obciążeń punktów dystrybucyjnych/. Fragment schematu tabel modeli (TM) wraz z powiązaniem – Entity Relationship Diagram (ERD) pokazano na Rys.3. a strukturę szczegółową tabel wraz z kluczami w Tab. 7.

Tab. 7. Tabele modeli (TM).

Nazwa tabeli	Opis	Symbol	Nazwa pola	Klucz	Opis kolumny
<b>uni_maga</b> (Magazyny)	Magazyny pośredniczące	i	KD_IDMAG	PK	Identyfikator magazynu
		nazwa	ID_NAZWA		Nazwa magazynu
		f	ID_KOSZT		Koszty ogólne magazynu
		w	ID_POJEM		Pojemność magazynu
		z	IW_UDZIA		Czy w dostawach bierze udział magazyn
<b>uni_towar</b> (Produkty)	Wytwarzane produkty	j	KD_PROD	PK	Identyfikator produktu
		nazwa	ID_NAZWA		Nazwa produktu
		s	ID_PRZES		Przeźreń potrzebna do przechowania produktu
<b>uni_klien</b> (Klienci)	Odbiorcy klienci	k	KD_KLIE	PK	Identyfikator klienta
		nazwa	ID_NAZWA		Nazwa produktu
<b>uni_zapo</b> (Zapotrzebowanie klienta na wyrób)	Zapotrzebowanie klienta na wyrób	j	KD_PROD	PK (FK)	Identyfikator produktu
		k	KD_KLIE	PK (FK)	Identyfikator klienta
		d	ID_ZAPOT		Zapotrzebowanie odbiorcy na towar
<b>unit_tech</b> (Lokalizacje)	Możliwość przechowywania towarów w magazynach	i	KD_IDMAG	PK (FK)	Identyfikator magazynu
		j	KD_PROD	PK (FK)	Identyfikator produktu
		r	ID_REAL		Czy magazyn może dystrybuować towar
<b>uni_prod</b> (Realizacja)	Realizacja dostaw produktu do klienta przez magazyn	i	KD_IDMAG	PK (FK)	Identyfikator magazynu
		j	KD_PROD	PK (FK)	Identyfikator produktu
		k	KD_KLIE	PK (FK)	Identyfikator klienta
		c	ID_KOSZT		Koszt realizacji dostawy
		x	ID_WIELK		Wielkość dostawy

PK- klucz główny, FK- klucz obcy



Rys. 3. Schemat ERD dla tabel modeli (TM) fragment dla modelu zarządzania łańcuchem dostaw (uproszczona notacja Martin'a).

## 2.2 Warstwa logiki aplikacji

Warstwa logiki aplikacji spełnia w prezentowanej implementacji SWD kilka funkcji. Po pierwsze umożliwia generację dynamicznych stron [7] co wynika z przyjętej wielowarstwowej architektury systemu (Rys.1.). Po drugie właśnie w tej warstwie zaimplementowano mechanizm automatycznej generacji modeli decyzyjnych/pytań na podstawie tabel systemowych (TS) oraz tabel modeli (TM) SWD. W tej warstwie również jest zaimplementowana funkcja przesyłania wygenerowanych modeli do oprogramowania MIP/CLP gdzie są rozwiązywane modele czyli uzyskiwane decyzje. W bieżącej implementacji wykorzystano pakiet oprogramowania LINGO jako środowisko MIP [5] oraz Eclipse jako środowisko CLP [4]. Oprogramowanie LINGO firmy LINDO służy do modelowania i optymalizacji problemów liniowych, nieliniowych i całkowitoliczbowych. Posiada swój metajęzyk modelowania. Jest to oprogramowanie komercyjne, ale posiada również wersje darmowe posiadające restrykcje na liczbę zmiennych decyzyjnych oraz ograniczeń. Środowisko ECLIPSE to darmowe oprogramowanie opierające się na paradygmacie programowania deklaratywnego. Wywodzi się z oprogramowania w logice z języka Prolog, który został wzbogacony o rozwiązywanie ograniczeń. Stąd jest określane jako programowanie w logice z ograniczeniami (CLP) [6].

Na Rys.4 przedstawiono plik tekstowy z wygenerowanym automatycznie przez SWD w formacie pakietu LINGO modelem zarządzania łańcuchem dostaw [3]. Model jest gotowy do przesłania i rozwiązania w środowisku optymalizacyjnym pakietu.

Warstwa logiki aplikacji została zaimplementowana w środowisku php [7] z wykorzystaniem frameworku Joomla [8].

```

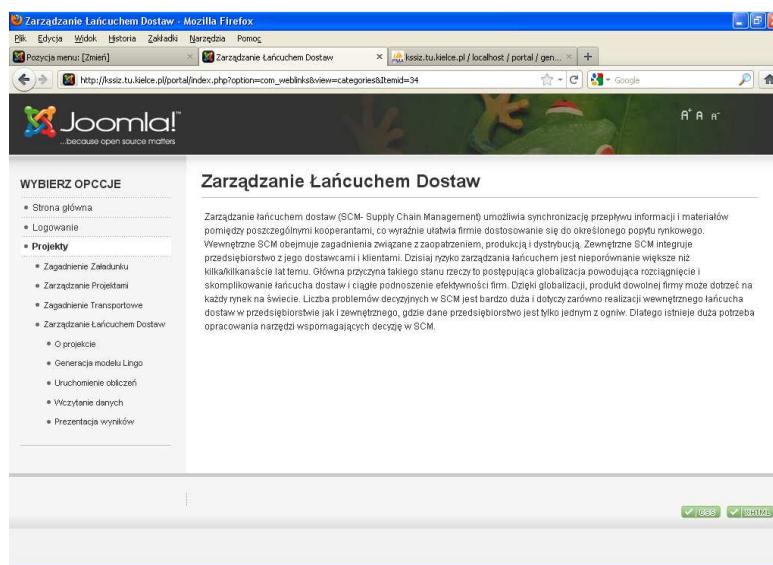
Model:
! Indeksy;
! i - magazyny/centra dystrybucyjne j - towary k - odbiorcy/klienci;
! Zmienne Decyzyjne;
!  $X_{ijk}$       Ile produktu j do klienta k jest dostarczane z
                wykorzystaniem magazynu/centrum i (od 0 do 1) procentowo ;
!  $Z_i$       Czy w dostawach bierze udział magazyn/centrum i;
Sets:
! Magazyny/centra dystrybucyjne;
Maga /1..4/:f,z,w;
! f - koszt ogólne magazynu/centrum i w - pojemność magazynu/centrum i;
! Towary;
towar /1..10/:s;! s - przestrzeń potrzebna do przechowania produktu j ;
! Klienci;
klient /1..8/;
!Wyroby dla klienta;
zapo (towar,klient):d;!d-Zapotrzebowanie odbiorcy/klienta k na towar j;
!Towary w magazynach;
tech (maga,towar):r;
! r - czy magazyn/centrum i może dystrybuować towar j;
!Realizacja dostaw produktu j do klienta k przez magazyn i;
prod (maga,towar,klient):x,c;
! c - koszt realizacji dostaw przez centrum i produktów j dla odbiorcy k;
EndSets
Data:
c = @file('koszt'); w = @file('pojemnosc');f = @file('koszt_og');
s = @file('przestrzen'); r = @file('udzial');
! Parametr p - maksymalna ilość magazynów/centrów do realizacji dostaw;
p = 4;
EndData
! Koszty utrzymania centrów i koszty realizacji dostaw;
Min=@sum(maga(i):f(i)*z(i))
      +@sum(prod(i,j,k):c(i,j,k)*x(i,j,k));
! Ograniczenie 2 - każdy produkt j dla każdego klienta k ma być w całości
dostarczony przy wykorzystaniu centrów i;
@for(towar(j): @for(klient(k): @sum(maga(i):r(i,j)*x(i,j,k))=1));
! Ograniczenie 3 - nie wolno przekroczyć dopuszczalnej liczby użycia
magazynów/centrów;
@sum(maga(i):z(i))<=p;
! Ograniczenie 4 - jeśli magazyn dostarcza produkt j do klienta to musi
pracować;
@for(prod(i,j,k):x(i,j,k)<=z(i));
! Ograniczenie 5 - pojemność centrom;
@for(maga(i):@sum(towar(j):@sum(klient(k):s(j)*d(j,k)*x(i,j,k)))<=w(i));
! Binarność;
@for(maga(i):@bin(z(i)));
!Przedziałowość zmiennej x
@for(prod(i,j,k): x(i,j,k)<=1);
End

```

Rys. 4. Plik tekstowy z dwuwarstwowym modelem zarządzania łańcuchem dostaw w formacie pakietu LINGO.

### 2.3 Warstwa prezentacji

Warstwa prezentacji została opracowana przy wykorzystaniu frameworku Joomla [8]. Stanowi ją cienki klient w postaci przeglądarki internetowej. Na Rys.5 przedstawiono okno główne SWD. Jest to wersja robocza systemu a raczej rodzaj szkieletu. Zapewniono w tej wersji implementacji kontakt z bazą danych, obustronny kontakt ze środowiskami MIP/CLP oraz możliwość łatwego dodawania nowych modeli decyzyjnych. Dopracowania w kolejnych wersjach wymaga sposób zadawania pytań do systemu oraz prezentacji wyników Rys.6, Rys.7.



Rys. 5. Okno główne SWD (zarządzanie łańcuchem dostaw)

The screenshot shows a Joomla! website interface in a Mozilla Firefox browser. The page title is 'Wyniki optymalizacji - Pakiet Lingo - Udział centrów'. The table displays optimization results for the Lingo package, showing the center share for four different centers. The table has three columns: 'L.P.' (Serial Number), 'Centrum' (Center), and 'Czy uczestniczy' (Participates). The data is as follows:

L.P.	Centrum	Czy uczestniczy
1	00001	1
2	00002	1
3	00003	1
4	00004	1

Rys.6. Okno prezentacja wyników cz.1 (zarządzanie łańcuchem dostaw)



Wyniki optymalizacji - Pakiet Lingo - Zarządzanie Łańcuchem dostaw

Lp.	Magazyn	Produkt	Klient	Wielkość dostawy
61	00004	00002	00002	1
62	00004	00002	00003	1
63	00004	00002	00004	1
64	00004	00002	00005	1
65	00004	00002	00006	1
66	00004	00002	00007	1
67	00004	00009	00001	1
68	00004	00009	00002	1
69	00004	00009	00005	1
70	00004	00009	00006	1
71	00004	00009	00007	1
72	00004	00010	00001	1
73	00004	00010	00002	1
74	00004	00010	00003	1
75	00004	00010	00005	1
76	00004	00010	00006	1
77	00004	00010	00007	1
78	00004	00010	00008	1

Rys. 7. Okno prezentacja wyników cz.2 (zarządzanie łańcuchem dostaw)

### 3. WNIOSKI

W artykule przedstawiono koncepcje podstawowych struktur oraz architekturę SWD. Przedstawiono również aspekty implementacyjne, zastosowane narzędzia oraz pierwszą wersję systemu. Bardzo duża elastyczność zaproponowanego rozwiązania wynika z zastosowania:

- relacyjnej bazy danych jako podstawy systemu;
- mechanizmów automatycznej generacji modeli i predykatów;
- mechanizmów mapowania danych – czyli automatycznego zasilania tabel z danymi z bazy danych systemu ERP/SCM;
- uniwersalnej wielowarstwowej architektury aplikacji mogącej pracować w dowolnym środowisku sieciowym.

### 4. LITERATURA

- [1] Schrijver A.: *Theory of Linear and Integer Programming*, John Wiley & sons, 2000.
- [2] Apt K.: *Principles of constraint programming*, Cambridge University, 2003.
- [3] Sitek P.: *Koncepcja systemu do wspomaganie decyzji zarządzania łańcuchem dostaw*, Transcomp 2011, LOGISTYKA (w druku).
- [4] <http://www.eclipse.org>
- [5] <http://www.lindo.com>
- [6] Frühwirth, Thom, Slim Abdennadher: *Essentials of constraint programming*, Springer, 2003.
- [7] <http://www.php.net/docs.php>
- [8] <http://www.joomla.org>