

Opis relacji i tabel w bazach programu MagAD-SQL

Bardzo wszystkich adminów baz przepraszam za to, co jest bazą MagAD. Program ten jednak rozwinął się z poprzednika, programu MAGPX i z konieczności musi dźwigać jego spuściznę. Każdy system bazodanowy projektowany jest w sposób X, a w trakcie rozwoju przybywają kolejne elementy. Tu i tak nie wygląda tak strasznie, wierzcie mi. W niżej przedstawionym opisie nie starałem się specjalnie o formę, chodziło o przedstawienie konkretnych informacji. Oprócz tego to, co najdziwniejsze, że każdy magazyn ma swój „set” tabel, okazało się idealnym rozwiązaniem przy dużych rozproszonych instalacjach, gdzie oddział ma swój magazyn i tylko dane w tym magazynie synchronizuje, co nie powoduje konfliktów przy przesyłaniu danych z kilku oddziałów równocześnie. Sa wdrożenia, że tych oddziałów jest ok. 300 ...

Struktura baz danych MAGPX jest taka sama, ale znak „_” w nazwach tablic należy zastąpić „\” - w ten sposób powstaną nazwy tablic MAGPX. Niektóre tablice słownikowe mają na końcu cyferki - zamiast nich w MAGPX są różne znaczki, ale na pewno zorientujecie się, która tablica jest która.

Motorem bazy MAGPX jest Paradox 4.0, MagAD-SQL PostgreSQL od 7.4 wzwyż. Od wersji MagAD 3.4 wzwyż zalecany jest PostgreSQL od 8.4 wzwyż. Bazy Paradox mają system sortowania Mazovia, PostgreSQL muszą mieć dowolne ustawienia polskie, bazy w kodowaniu najlepiej LATIN2, ale mogą być także UTF lub Win CP1250. W Unix/Linux najkorzystniejszy układ clustra to dla wszystkich ustawień narodowych pl_PL, bazy w kodowaniu LATIN2, drugi to UTF8. W Windows ustawienia polskie, bazy w kodowaniu Win CP1250 drugi ustawienie to UTF8.

Niniejsze opisy są, jak instrukcja programu MagAD i MAGPX na licencji GNU Wolnej Dokumentacji, która to licencja załączona jest w dokumentacji.

W opisie zastosowano typy pól, które są używane w MagAD - oto ich odpowiedniki dla systemu baz danych Paradox 4.0 i tablicy tables_list mojego frameworka, który stworzyłem dla programu MagAD:

PostgreSQL	Paradox	Framework	uwagi
char, character varying (x)	Ax	Ax	
smallint	S	S	
date	D	D	
integer	N	I	zawierające czterobajtowe liczby całkowite (longint)
numeric (28,7)	N	X	zawierające liczby zmiennoprzecinkowe
bigint	N	Y	zawierające liczby całkowite do 15 cyfr znaczących
text	Bx	Bx	dane zakodowane szesnastkowo – każde dwa znaki to cyfra szesnastkowa – do 16kb znaków
text	Mx	Mx	czysty tekst – do 16kb znaków

MagAD-SQL używa jednej roli (użytkownika SZBD PostgreSQL) do podłączania się do bazy. Rola musi być właścicielem bazy. Operatorzy zdefiniowani w tablicy dane_uzytkown_uzytkown nie mają nic wspólnego z rolami (użytkownikami) w systemie baz danych PostgreSQL.

Czytając schematy relacji należy mieć na uwadze to, że na rysunkach niekiedy nie są widoczne wszystkie nazwy pól w danej tabeli. Aby dostać informację nt. pól danej tabeli i ich opis, należy sięgać do szczegółowych opisów tabel z niniejszego opracowania.

Table "public.tables_list" - tablica sterująca napisanego przeze mnie frameworka do obróbki baz MagAD-SQL, zawieraj opisy tabel w bazie			
Column	Type	Modifiers	
nazt	character varying(255)	not null	- nazwa tabeli
nazp	character varying(25)	not null	- nazwa pola
typp	character varying(4)		- specjalny kod typu pola
k	character(1)	default '':bpchar	- gwiazdka oznacza, że pole nazp jest elementem klucza głównego
i	character(1)	default '':bpchar	- gwiazdka oznacza, że pole nazp jest elementem indeksu założonego na jedno pole (kolumnę)
ci	character varying(25)	default '':character varying	- nazwa indeksu złożonego z kilku pól
temp	character(1)	default '':bpchar	- czy tablica jest tymczasowa - jeśli jest w polu jest gwiazdka
maxrec	integer	default 0	- oszacowana w jakimś momencie ilość rekordów
lp	integer	default 0	- kolejny numer pola w ramach tabeli
cilp	integer	default 0	- numer pola w inksie złożonym (indeks złożony może być tylko jeden dla jednej tablicy)
lck	character(1)	default 'N':bpchar	- czy jest zablokowana do zapisu (nie używane, ale zapisywane)
typi	smallint	default 0	- jeśli na polu określonym w rekordzie, założono indeks jednopółowy, to typ indeksu: 1,2 - z powtórzeniami, 3,4 - bez powtórzeń
typci	smallint	default 0	- jeśli na polu określonym w rekordzie, założono indeks złożony, to typ indeksu: 1,2 - z powtórzeniami, 3,4 - bez powtórzeń
il_zm	integer	default -2147483647	- wartość zwiększana o 1, jeśli do tablicy „poszedł” zapis - w MagAD istotny jest fakt jej zmiany, a nie wartości; jeśli wpisaliśmy coś do rekordu jakiejś tabeli to dla lp=1 musimy zmienić wartość il_zm

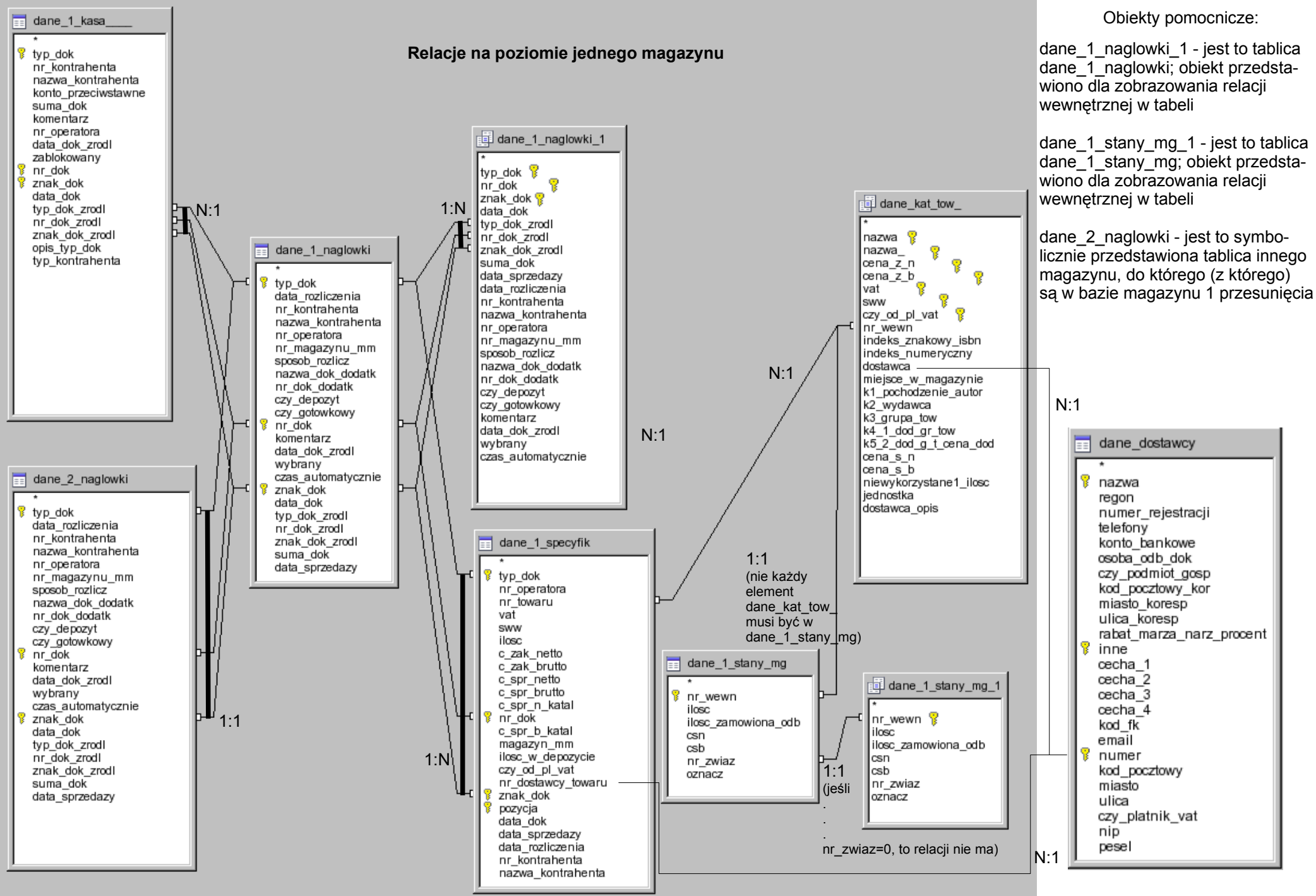
Indexes:

"tables_list_pkey" PRIMARY KEY, btree (nazt, nazp)
"tables_list_lp" btree (lp)

Uszkodzenie/utrata tej tablicy spowoduje dysfunkcje programu MagAD, co akurat nigdy się nie zdarzyło, ale odzyskać można ją w prosty sposób instalując pustą bazę programu, konfigurując go tak, jak używający uszkodzonej bazy, w opcji „Inne - Inne katalogi” wchodząc do każdego takiego katalogu (np. vat itp.), potem zakładając wszystkie magazyny (istotne, aby istniał magazyn o danym numerze, reszta danych jest zbędna). To wszystko w pustej bazie. Potem wystarczy tablicę tables_list przekopiować do właściwej bazy.

Dla kogoś, kto analizuje bazę, czy operuje innymi narzędziami, niż mój framework, tablica wcale nie musiałaby istnieć.

Relacje na poziomie jednego magazynu



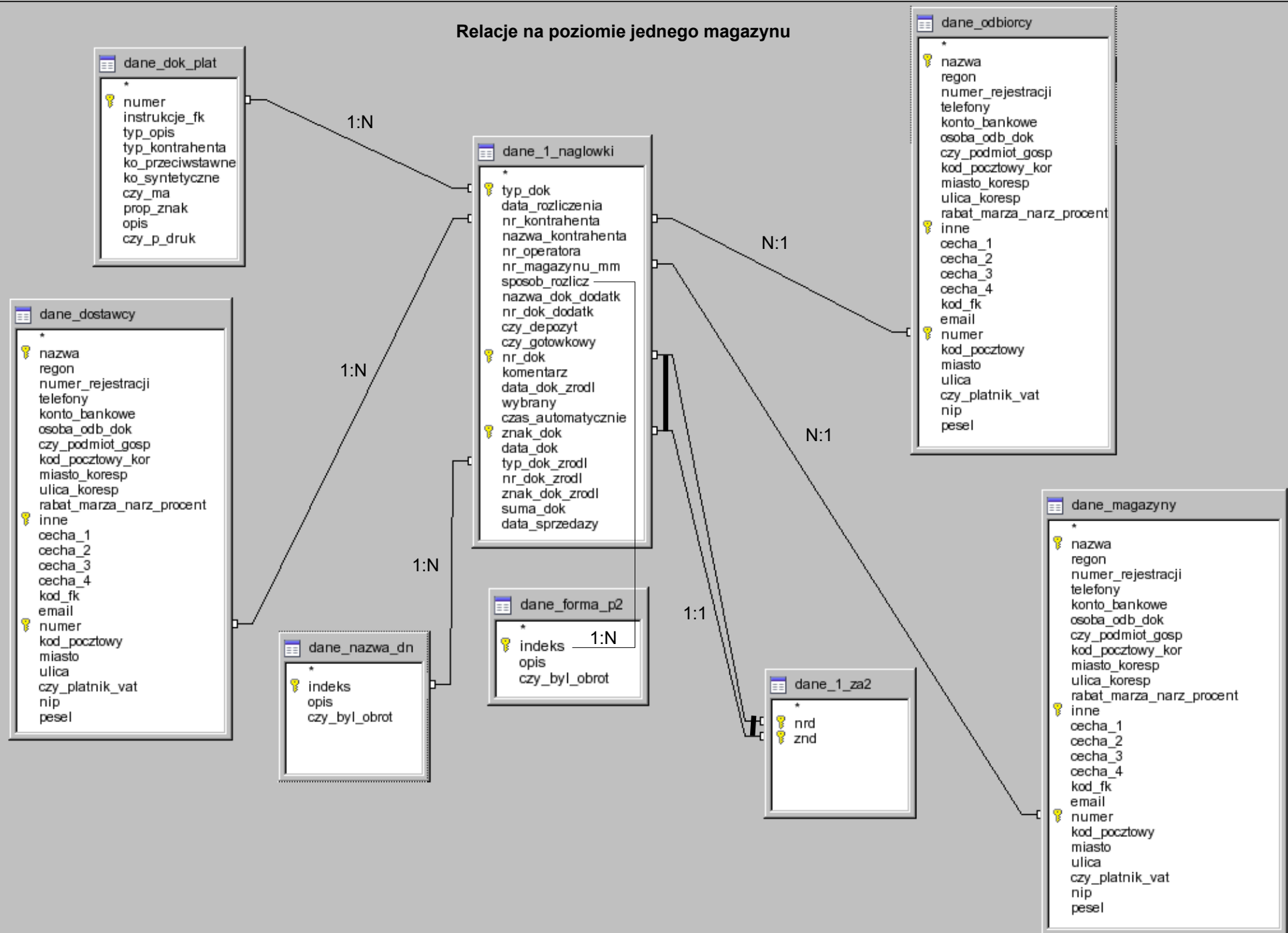
Obiekty pomocnicze:

dane_1_naglowki_1 - jest to tablica dane_1_naglowki; obiekt przedstawiono dla zobrazowania relacji wewnętrznej w tabeli

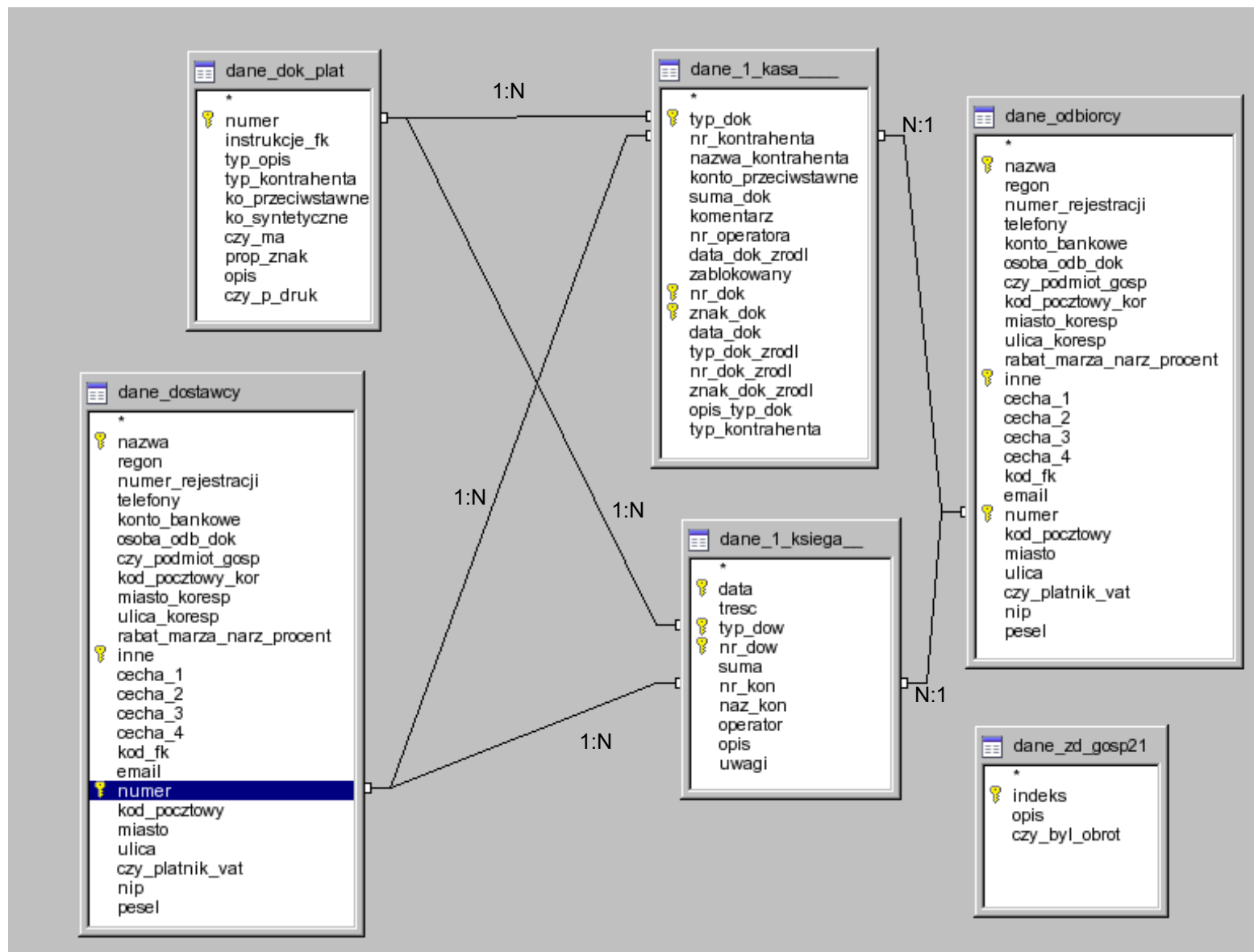
dane_1_stany_mg_1 - jest to tablica dane_1_stany_mg; obiekt przedstawiono dla zobrazowania relacji wewnętrznej w tabeli

dane_2_naglowki - jest to symbolicznie przedstawiona tablica innego magazynu, do którego (z którego) są w bazie magazynu 1 przesunięcia

Relacje na poziomie jednego magazynu



Relacje na poziomie jednego magazynu



Tablica dane_zd_gosp21 służy jako słownik do pobierania informacji z pola opis do dane_1_ksiega__.opis . W dane_1_ksiega__.opis operator może jednak dowolnie zmodyfikować tekst

Opis relacji na poziomie jednego magazynu:

Tablice magazynu tworzone są w momencie założenia magazynu w programie MagAD. W momencie usunięcia magazynu przez Kierownika Systemu mogą zostać utracone relacje pomiędzy przesunięciami, więc nie powinno się magazynów usuwać, chyba, że dany magazyn nie ma przesunięć z niego i do niego.

Na rysunku powyżej tablice o nazwach zawierających po słowie „dane_” numer to tablice z danymi magazynu o określonym numerze. Tu ten numer magazynu symbolizuje cyfra „1”. Jeśli numer magazynu ma do 8 cyfr, przedstawiany jest po prostu, jako cyfra, ale jeśli ma więcej, będzie podzielony - pierwsze osiem cyfr, znak podkreślenia, reszta cyfr. W tej (niespotykanej w praktyce) sytuacji, kiedy numer magazynu ma więcej osiem cyfr, nazwa tabeli nagłówek dokumentów towarowych (na rysunku dane_1_naglowki) będzie następującej postaci: dane_12345678_12_naglowki (czyli numer magazynu będzie 1234567812).

Niezbyt to wygodne, ale aby odpytać wyznaczone magazyny np. o sprzedaż, trzeba połączyć dane z kilku zapytań - oddzielnie do każdego magazynu, który nas interesuje...

Powiązania na poziomie kartotek towarowych

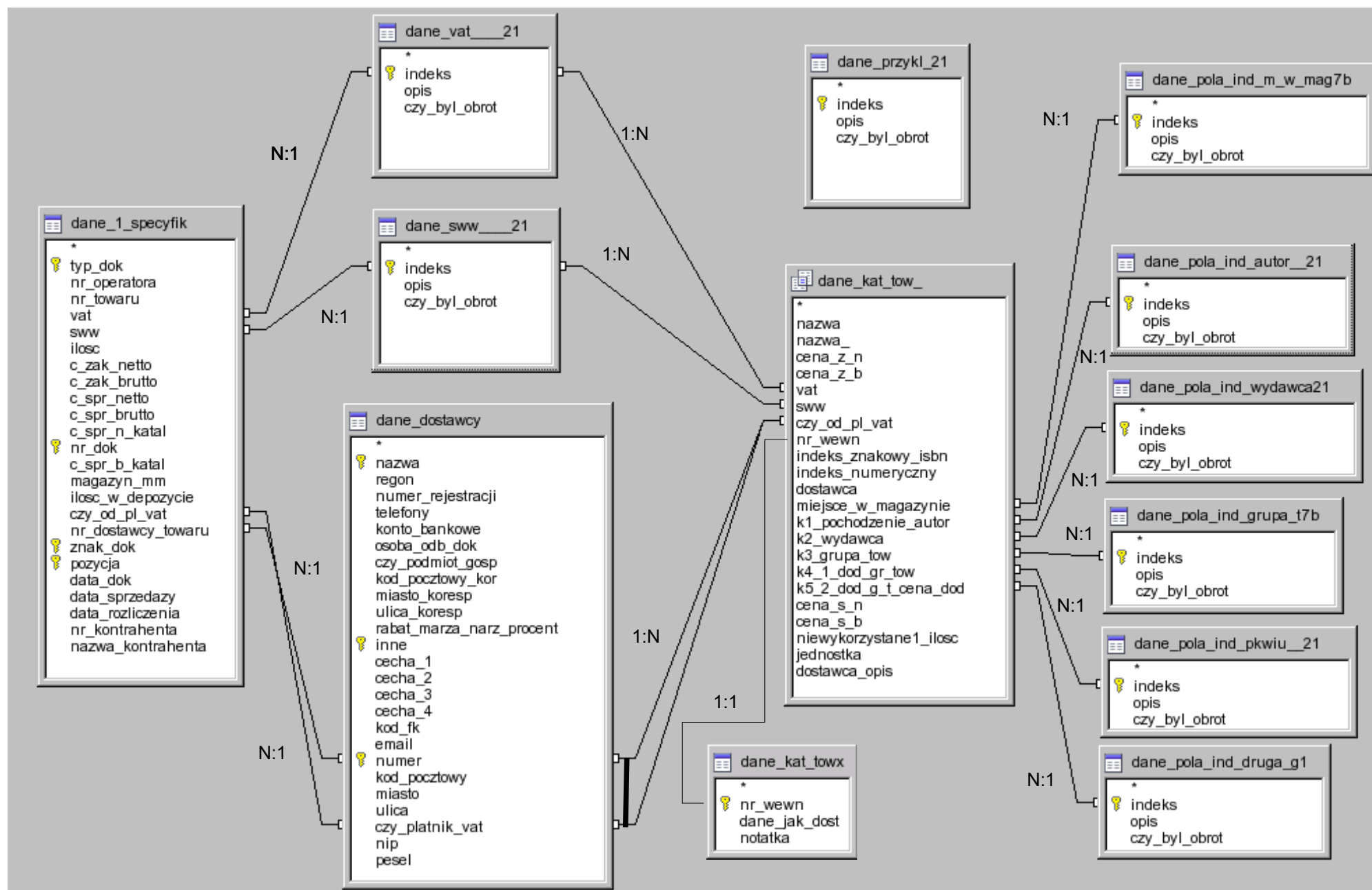


Table "public.dane_1_naglowki" - nagłówki dokumentów towarowych w magazynie 1		
Column	Type	Modifiers - Opis
typ_dok	smallint	not null - typ dokumentu - patrz zawartość tabeli dane_dok_plat
nr_dok	integer	not null - numer dokumentu - część cyfrowa
znak_dok	character varying(4)	not null - numer dokumentu - dodatkowe znaki
data_dok	date	- data wystawienia dokumentu
typ_dok_zrodl	smallint	- typ dokumentu źródłowego
nr_dok_zrodl	integer	- numer dokumentu źródłowego - część cyfrowa
znak_dok_zrodl	character varying(4)	- numer dokumentu źródłowego
suma_dok	numeric(28,7)	- wyliczona na podstawie specyfikacji suma dokumentu
data_sprzedazy	date	- data sprzedaży dla dok. sprzedaży i zakupu
data_rozliczenia	date	- termin płatności dla dok. sprzedaży i zakupu
nr_kontrahenta	integer	- kod cyfrowy kontrahenta (patrz relacje nagłówków z kontrahentami)
nazwa_kontrahenta	character varying(17)	- wpisany przy zapisie dokumentu początkowy fragment nazwy kontrahenta
nr_operatora	integer	- kod operatora, który wprowadził dokument - relacja z tablica dane_uzytkown_uzytkown.numerw
nr_magazynu_mm	integer	- kod magazynu do/z którego jest adresowane przesunięcie (typ_dok=15,18)
sposob_rozlicz	character varying(15)	- sposób rozliczenia - relacja N:1 z tablica formy płatności
nazwa_dok_dodat	character varying(15)	- typ dokumentu dostawcy (np. dla PZ zakupu typ dok. zakupu)
nr_dok_dodat	character varying(17)	- numer dokumentu dostawcy (np. dla PZ zakupu numer dok. zakupu)
czy_depozyt	character(1)	- czy dokument jest w całości, lub części depozytowy
czy_gotowkowy	character(1)	- czy dokument jest nieterminowy =T, terminowy=N
komentarz	character varying(47)	- krótki komentarz indywidualny dla każdego dokumentu
data_dok_zrodl	date	- data dokumentu źródłowego
zablokowany	character(1)	- czy dokument jest zablokowany do edycji, w niektórych bazach pole może mieć nazwę „wybrany”
czas_automatycznie	double precision	- czas pierwszego utworzenia dokumentu od północy w sekundach, przy zmianie czegokolwiek niezmienny
dod_inf	character varying(12)	- pole nieedytowalne z poziomu MagAD; dla dokumentów powiązanych z płatnościami w zależności od konfiguracji MagAD
rab_kwot	numeric(28,7)	- może przechowywać upust kwotowy dla danego dokumentu, o ile opcja upustów jest włączona
czy_mm_zwr	character(1)	W konfiguracji
uz_modyfik	integer	- wykorzystywane przez specyficzną wersję programu
czy_zam_na_dok	character(1)	- numer użytkownika, który ostatnio zmodyfikował dokument - relacja z tablica dane_uzytkown_uzytkown.numerw
vwyb	character varying(15)	- czy dokument był zamieniany na inny typ dokumentu (MagAD ma opcję automatycznych zamian dokumentów)
vinf	character varying(17)	- do dowolnego wykorzystania – MagAD modyfikuje te pola tylko w wersji VIOLA, która nie jest standardowa
nag1	character varying(15)	- j.w.
nag2	character varying(15)	- dodatkowa dana tekstowa, może być włączona do edycji lub nie, może mieć listę wyb. lub nie – w zał. od konfiguracji w pliku „dane/lista_na.txt”
nag3	character varying(15)	- j.w.
nag4	character varying(15)	- j.w.
nag5	character varying(15)	- j.w.
czy_anulowany	character(1)	- czy dokument jest anulowany T/N – jeśli T, to specyfikacja nie istnieje w tablicy „dane_1_specyfika”, ale „dane_dokanuls”
suma_dok_anul	numeric(28,7)	- suma dokumentu w momencie anulowania
czy_wyslany	smallint	- czy dokument wysłano email 1-tak inne liczby - nie

Indexes:

"dane_1_naglowki_pkey" PRIMARY KEY, btree (typ_dok, nr_dok, znak_dok)
 "dane_1_naglowki_oidindeks" UNIQUE, btree (oid)
 "dane_1_naglowki_data_dok" btree (data_dok)
 "dane_1_naglowki_data_dok_u" btree (data_dok, typ_dok, nr_dok, znak_dok)
 "dane_1_naglowki_dok_zrodl" btree (typ_dok_zrodl, nr_dok_zrodl, znak_dok_zrodl)
 "dane_1_naglowki_nr_kontrahenta" btree (nr_kontrahenta)
 "dane_1_naglowki_nr_kontrahenta_u" btree (nr_kontrahenta, typ_dok, nr_dok, znak_dok)

Numery dokumentów w MagAD/MAGPX mają w sobie zakodowany rok i numer dokumentu - najbardziej znaczący bajt czterobajтового integer zawiera kod roku, pozostałe bajty numer dokumentu. Kod roku to rok-1996, czyli np. kod roku 2011 to 2011-1996=15.

Czyli dana z pola "nr_dok" skonstruowana jest tak:

bajt3 | bajt2 | bajt1 | bajt0

kod.r. | n u m e r

Jeżeli wybieramy dane z pola „nr_dok”, to:

Za pomocą polecenia SQL rok można uzyskać przykładowo:

```
SELECT 1996+(T.NR_DOK>>24) AS ROK FROM DANE_1_NAGLOWKI AS T
```

Numer dokumentu w roku:

```
SELECT T.NR_DOK-((T.NR_DOK>>24)<<24) AS NR FROM DANE_1_NAGLOWKI AS T
```


Aspekty nadawania numerów dokumentów – maksymalny + 1 oraz pierwszy wolny – na przykładzie

Nadanie pierwszego maksymalnego numeru dokumentu jest trywialne (na przykład dokument MM /typ=15/ w roku 2023:

```
SELECT MAX(nr_dok-((nr_dok>>24)<<24)))+1 AS P
FROM DANE_2_NAGLOWKI
WHERE ((typ_dok=15) and (data_dok>='2023-01-01') and (data_dok<='2023-12-31'))
```

Zwrócony zostanie jeden rekord z wartością maksymalną. Jeżeli nie zostanie zwrócony żaden rekord, przyjmujemy wartość numeru dokumentu = 1 .

Nadanie pierwszego wolnego dokumentu, ze względu na nieco różne działanie wstawiania inkrementowanych sekwencji w zależności od wersji PostgreSQL po testach może obyc się tylko tak:

```
create temporary sequence XX start 1;

create temporary table xx1 as SELECT NEXTVAL('XX') AS NXX,(nr_dok-((nr_dok>>24)<<24)) AS P
FROM DANE_2_NAGLOWKI
WHERE ((typ_dok=15) and (data_dok>='2023-01-01') and (data_dok<='2023-12-31')) AND ((nr_dok-((nr_dok>>24)<<24))>=1);

drop sequence xx;

delete from xx1 using (select p from xx1) as xx2 where xx1.nxx=xx2.p;

select nxx from xx1 order by nxx limit 1;
-- zapytanie powyżej zwróci pierwszy wolny numer, a jeżeli nie zwróci żadnego rekordu, oznacza to, że pierwszy wolny numer należy przyjąć MAX+1

Drop table xx1;
```

Przykład, który przedstawiłem powyżej realizuje już dla dowolnego przypadków bazie funkcja, która w materiałach do MagAD

Jest w pliku "f_magad4.sql":

```
function z_pier_woln_nr(tn varchar,fn varchar,od_wartosci bigint,varunek varchar,czydoble int) returns bigint
Jej opis:
-- funkcja pobiera z tabeli tn z pola fn pierwszy wolny numer >= od_wartosci dla warunku warunek
-- czydouble=1 wskazuje, że pole zwraca wartosc zmiennoprzecinkowa i wartosci z niego beda zaokraglane do 1 przy testowaniu
-- fn moze byc wyrazeniem
-- tn moze byc zapytaniem, ale wtedy musi byc to zapytanie typu:
-- (select nr*2 as nr1 from jakas_tablica) as t1
-- i wtedy pole fn musi byc zalezne od nr1, czyli moze byc tm nr1, albo np. nr1-1
-- funkcja jest wgrywana w upg37_38
```

Kodowanie numeru dokumentu w SQL (symbolicznie):

```
nr_dok=(rok_dokumentu-1996)<<24|numer_niezakodowany
Używamy więc operatorów binarnych „<<” (przesunięcie bitowe w lewo w kierunku bitów starszych) i „|” (bitowy OR)
Na przykład, jeśli do tabeli gdzie r zawierać będzie rok, a n normalny, niezakodowany numer:
create table t1(nr int default 0,r int default 0,n int default 0);
dodamy dane:
insert into t1 values (0,2011,1);
insert into t1 values (0,2011,2);
insert into t1 values (0,2011,3);
i wykonamy:
update t1 set nr=(r-1996)<<24|n;
to w wyniku w polu nr otrzymamy zakodowane numery:
select * from t1 order by n;
  nr      | r      | n
-----+-----+---
268435457 | 2011   | 1
268435458 | 2011   | 2
268435459 | 2011   | 3
```

Algorytm kodowania numeru dokumentu (r-rok,n-numer do zakodowania) w języku Pascal:

```
function m_cnr(r:integer;n:longint):longint;
var
p:record
  c,b,a:byte;
  rok:byte;
end;
w:longint;
Begin
m_cnr:=0;
r:=r-1996;
if (r<0)or(r>127)or(n<0)or(n>16777215) then begin
writeln('Logiczne uszkodzenie numeru dokumentu (kodowanie).');exit;
end;

with p do begin
rok:=r;
a:=n div 65536;
b:=(n-longint(a)*65536) div 256;
c:=n-longint(a)*65536-longint(b)*256;
end;
move(p,w,4);
m_cnr:=w;
end;
```

Algorytm dekodowania numeru dokumentu (nw. funkcja zwraca false, gdy dekodowanie jest nieudane, cn-numer,r-rok,nr-zdekodowany numer) (język Pascal):

```
function test_dcnr(var nr:longint;var r:integer;cn:longint):boolean;
var
p:record
  c,b,a:byte;
  rok:byte;
end;
begin
move(cn,p,4);
with p do begin
test_dcnr:=rok<=127;
if rok<=127 then begin
r:=rok;r:=r+1996;
nr:=longint(c)+longint(b)*256+longint(a)*65536;
end else begin
nr:=0;r:=0;
end;
end;
end;
```

Table "public.dane_forma_p2" - lista form płatności

Column	Type	Modifiers
indeks	character varying(15)	not null - klucz „nick” tekstowy
opis	character varying(55)	- opis drukowany na dokumentach
czy_był_obrot	character(1)	- nie używane

Indexes:
"dane_forma_p2_pkey" PRIMARY KEY, btree (indeks)
"dane_forma_p2_oidindeks" UNIQUE, btree (oid)

Table "public.dane_1_stany_mg" - pozycja stanu w magazynie

Column	Type	Modifiers
nr_wewn	integer	not null
ilosc	numeric(28,7)	
ilosc_zamowiona_odb	numeric(28,7)	
csn	numeric(28,7)	
csb	numeric(28,7)	
nr_zwiaz	integer	
oznacz	character(1)	

- numer systemowy kartoteki towarowej/usługi
 - aktualna ilość w magazynie nr 1
 - aktualna ilość towarów/usług zamówionych w danym magazynie
 - cena sprzedaży netto kartoteki w magazynie 1
 - cena sprzedaży brutto kartoteki w magazynie 1
 - numer systemowy kartoteki, z którą kartoteka nr_wewn ma powiązanie
 - czy towar został oznaczony w danym magazynie do późniejszego stworzenia pliku roboczego - wartość T oznacza, że towar jest oznaczony

Indexes:

"dane_1_stany_mg_pkey" PRIMARY KEY, btree (nr_wewn)

"dane_1_stany_mg_oidindeks" UNIQUE, btree (oid)

Uwaga - „powiązania” kartotek służą do tego, aby po wybraniu do dokumentu jednej z tych kartotek, na specyfikację dokumentu „wskakiwała” równolegle druga, jeśli dla pierwszej podano rabat, to druga wskakuje od razu z tym rabatem. Powiązanie funkcjonuje w obrębie jednego magazynu, przy przesunięciach przenosi się do magazynu, do którego są przesuwane kartoteki. Twary mogą tworzyć łańcuchy powiązań - ostatnia powiązana pozycja jest powiązana z pierwszą, co „zamyka” łańcuch.

Table "public.dane_1_specyfik" - zawartość towarowa (specyfikacje) dokumentów towarowych.

Column	Type	Modifiers
typ_dok	smallint	not null - patrz dane_1_naglowki
nr_dok	integer	not null - patrz dane_1_naglowki
znak_dok	character varying(4)	not null - patrz dane_1_naglowki
pozycja	integer	not null - numer pozycji na dokumencie - dane niewidzialna, tworzona automatycznie
data_dok	date	- patrz dane_1_naglowki
data_sprzedazy	character(1)	- nie używane
data_rozliczenia	character(1)	- nie używane
nr_kontrahenta	character(1)	- nie używane
nazwa_kontrahenta	character(1)	- nie używane
nr_operatora	character(1)	- nie używane
nr_towaru	integer	- nr systemowy kartoteki (patrz np. dane_1_stany_mg, dane_kat_tow_)
vat	character varying(5)	- VAT (redundancja z dane_kat_tow_)
sww	character varying(10)	- pole widziane w programie po standardowej instalacji, jako gr.tow. (redundancja z dane_kat_tow_)
ilosc	numeric(28,7)	- ilość danej pozycji na dokumencie
c_zak_netto	numeric(28,7)	- cena zakupu netto (redundancja z dane_kat_tow_ lub cena zakupu w momencie wystawienia dokumentu)
c_zak_brutto	numeric(28,7)	- analogicznie do powyższego, cena brutto
c_spr_netto	numeric(28,7)	- cena sprzedaży netto na dokumencie
c_spr_brutto	numeric(28,7)	- cena sprzedaży brutto na dokumencie
c_spr_n_katal	numeric(28,7)	- cena sprzedaży netto na dokumencie, od której udzielono rabatu
c_spr_b_katal	numeric(28,7)	- cena sprzedaży netto na dokumencie, od której udzielono rabatu
magazyn_mm	character(1)	- nie używane
ilosc_w_depozycie	numeric(28,7)	- ile na danych dokumencie jest danej kartoteki oznaczone, jako ilość w depozycie
czy_od_pl_vat	character(1)	- czy dostawca jest płatnikiem VAT (redundancja z pierwszym znakiem z analogicznego pola w dane_kat_tow_)
nr_dostawcy_towaru	integer	- numer dostawcy (redundancja z dane_kat_tow_)

Indexes:

"dane_1_specyfik_pkey" PRIMARY KEY, btree (typ_dok, nr_dok, znak_dok, pozycja)

"dane_1_specyfik_oidindeks" UNIQUE, btree (oid)

"dane_1_specyfik_nr_towaru" btree (nr_towaru)

W programie kilka pól może przenosić ceny sprzedaży (patrz dane_kat_tow_ i od nich może być udzielany rabat. Jeśli rabatu na dana pozycję nie ma, zachodzi $c_spr_netto=c_spr_n_katal$, $c_spr_brutto=c_spr_b_katal$.

Pola nie używane wypełniane są nieraz różnymi znaczkami, nie radzę z nich korzystać do zapisu w innych aplikacjach.

Ilość w depozycie to ilość towaru z danego dokumentu, która nie została w jakiejś formie rozliczona, w najprostszym układzie po prostu nie zapłacona.

Tablicy „dane_1_specyfik” towarzyszy druga o identycznej strukturze „dane_1_spekont”, gdzie zapisywane są kontrolne specyfikacje dokumentów (edycja F3 na liście dokumentów). Po kontrolnym wpisaniu MagAD dokonuje sprawdzenia, czy wpisana specyfikacja zgadza się z oryginalną.

W wersjach MagAD od 3.5 w górę tablica służy albo to ww. Celu, albo do zapisywania specyfikacji dokumentów anulowanych.

Dokumenty anulowane znajdują się we wspólnej dla wszystkich magazynów tabeli „dane_dokanuls”, której struktura jest identyczna, jak „dane_1_specyfik”, Z tą różnicą, że posiada na początku pole:

mag_dok	integer	not null
---------	---------	----------

a indeksy tablicy są takie:

"dane_dokanuls_pkey" PRIMARY KEY, btree (mag_dok, typ_dok, nr_dok, znak_dok, pozycja)

"dane_dokanuls_oidindeks" UNIQUE, btree (oid)

Table "public.dane_1_kasa____" dokumenty płatności		
Column	Type	Modifiers
typ_dok	smallint	not null - typ dokumentu - patrz zawartość dane_dok_plat
nr_dok	integer	not null - numer dokumentu
znak_dok	character varying(4)	not null - dodatkowe znaki, rozszerzające numer dokumentu
data_dok	date	- data wystawienia dokumentu
typ_dok_zrodl	smallint	- typ dokumentu źródłowego (dok. towarowy, do którego jest płatność), może być [2..9,16]
nr_dok_zrodl	integer	- numer dokumentu źródłowego
znak_dok_zrodl	character varying(4)	- dodatkowe znaki, rozszerzające numer dokumentu źródłowego
opis_typ_dok	character varying(7)	- opis typu dokumentu, pobrany z dane_dok_plat w momencie wystawienia
typ_kontrahenta	character(1)	- typ kontrahenta ['d','0'], pobrany z dane_dok_plat w momencie wystawienia
nr_kontrahenta	integer	- numer kontrahenta (w zależności od typu kontrahenta jest to numer z dane_dostawcy, albo dane_odbiorcy
nazwa_kontrahenta	character varying(17)	- początkowy fragment nazwy kontrahenta pobrany w momencie dopisania/edycji dokumentu z dane_dostawcy, albo dane_odbiorcy
konto_przeciwstawne	character varying(3)	- pobrane w momencie wystawienia z dane_dok_plat
suma_dok	numeric(28,7)	- suma dokumentu
komentarz	character varying(36)	- krótki komentarz do dokumentu
nr_operatora	integer	- numer operatora, który wystawił dokument
data_dok_zrodl	date	- data dokumentu źródłowego
zablokowany	character(1)	- czy dokument jest zablokowany do edycji

Indexes:

"dane_1_kasa____pkey" PRIMARY KEY, btree (typ_dok, nr_dok, znak_dok)
 "dane_1_kasa____oidindeks" UNIQUE, btree (oid)
 "dane_1_kasa____data_dok" btree (data_dok)
 "dane_1_kasa____data_dok_u" btree (data_dok, typ_dok, nr_dok, znak_dok)
 "dane_1_kasa____dok_zrodl" btree (typ_dok_zrodl, nr_dok_zrodl, znak_dok_zrodl)
 "dane_1_kasa____nr_kontrahenta" btree (nr_kontrahenta)
 "dane_1_kasa____nr_kontrahenta_u" btree (nr_kontrahenta, typ_dok, nr_dok, znak_dok)

UWAGA - jeżeli typ_dok_zrodl = 0, to dokument płatności nie jest przyporządkowany do dokumentu towarowego

Table "public.dane_1_ksiega__" dane księgi przychodów i rozchodów w magazynie

Column	Type	Modifiers
data	date	not null - data wpisu
typ_dow	smallint	not null - typ dowodu - patrz typy z dane_dok_plat
nr_dow	character varying(17)	not null - numer dowodu
suma	numeric(28,7)	- suma dowodu
nr_kon	integer	- numer kontrahenta (w zależności od typu kontrahenta jest to numer z dane_dostawcy, albo dane_odbiorcy w zależności, jak dokument poprzez typ_dok ma w dok_plat przyporządkowanego kontrahenta - czy odbiorcę, czy dostawcę)
naz_kon	character varying(26)	- początkowy fragment nazwy kontrahenta pobrany w momencie dopisania/edycji dokumentu z dane_dostawcy, albo dane_odbiorcy
operator	integer	- numer operatora, który wystawił dokument
opis	text	- opis operacji (do 256 znaków)
uwagi	text	- uwagi (do 256 znaków)
tresc	text	- ewentualna całkowita treść dokumentu (z księgi dzięki temu można drukować dowolne dokumenty, tworzone z małym edytorku tekstowym)

Indexes:

"dane_1_ksiega__pkey" PRIMARY KEY, btree (data, typ_dow, nr_dow)

"dane_1_ksiega__oidindeks" UNIQUE, btree (oid)

Nazwy pól w katalogu towarów są tak sugestywne, że części z nich nie ma sensu wyjaśniać.

Table "public.dane_kat_tow_"		
Column	Type	Modifiers
nazwa	character varying(31)	not null
nazwa_	character varying(30)	not null
cena_z_n	numeric(28,7)	not null
cena_z_b	numeric(28,7)	not null
vat	character varying(5)	not null
sww	character varying(10)	not null
		- w podstawowej konfiguracji MagAD to pole widoczne jest w formularzach ekranu, jako „gr.gł.” definicja jest w pliku konfiguracyjnym „list_i2.txt” lub w opcji konfigurowania widocznych W MagAD nazw pól (patrz niżej); kiedyś pole służyło do przechowywania SWW; w tej chwili definiuje się jedno z pól z podstawowego zestawu (pola od k1.. do k5.., jako przenoszące PKWiU, W podstawowej konfiguracji jest to „k4_1_dod_gr_tow”
czy_od_pl_vat	character varying(13)	not null
		- w tym polu gromadzona jest informacja (pierwszy znak, czy dostawca towaru jest płatnikiem VAT (jeden znak T lub N, a za nim powtórzony w formie tekstu numer systemowy dostawcy
nr_wewn	integer	
indeks_znakowy_isbn	character varying(15)	- to pole może być skonfigurowane w czasie instalacji na wielkość od 15 do 26 znaków
indeks_numeryczny	bigint	
dostawca	integer	- numer dostawcy
miejsce_w_magazynie	character varying(9)	\ - pole może pomimo tekstowego charakteru, w zależności od konfiguracji, zawierać ceną detaliczną, rok, datę
k1_pochodzenie_autor	character varying(15)	\
k2_wydawca	character varying(15)	\
k3_grupa_tow	character varying(15)	\ - pola o różnych funkcjonalnościach, których nazwy widziane w MagAD znajdują się w plikach konfigura-
k4_1_dod_gr_tow	character varying(15)	/ cyjnych „lista_in.txt” i „lista_i2.txt”, a także w programie przy konfigurowaniu tych nazw
k5_2_dod_g_t_cena_dod	character varying(15)	/ - pole może zawierać także dodatkową cenę sprzedaży w wielkości netto lub brutto w zał. od konfiguracji
cena_s_n	numeric(28,7)	
cena_s_b	numeric(28,7)	
niewykorzystane1_ilosc	numeric(28,7)	- tylko MagAD-SQL - jeśli to pole jest NULL, to pozycja ma status nieaktywnej i domyślnie nie jest pokazywana w opcji „Wprowadzanie”; w katalogu towarów nie ma żadnej ilości, pole jest historyczne,
dla zgodności kodu MagAD		
jednostka	character varying(7)	
ilosc_w_opak	numeric(28,7)	
dostawca_opis	character varying(20)	- wycięty podczas dopisywania kartoteki z nazwy dostawcy krótki fragment nazwy, jest używany tylko informacyjne, do analiz stosuje się pole „dostawca”
niewykorzystane2_il_zam	numeric(28,7)	- nieużywane, utrzymywane w celu wstecznej zgodności w opcji „wprowadzanie”, chyba, że chwilowo wymusimy jej widoczność
blok_rab	char default 'N'	czy pozycja zablokowana do rabatowania (T,N)
ozn	char default 'N'	czy pozycja jest oznaczona (T,N) – w opcji Wprowadzanie działa Ctrl+F – oznaczanie pozycji z głównego katalogu – używane głównie do analiz własnych

Indexes:

"dane_kat_tow__pkey" PRIMARY KEY, btree (nazwa, nazwa_, cena_z_n, cena_z_b, vat, sww, czy_od_pl_vat)
 "dane_kat_tow__nr_wewn" UNIQUE, btree (nr_wewn)
 "dane_kat_tow__oidindeks" UNIQUE, btree (oid)
 "dane_kat_tow__dostawca" btree (dostawca)
 "dane_kat_tow__dostawca_u" btree (dostawca, nazwa, nazwa_, cena_z_n, cena_z_b, vat, sww, czy_od_pl_vat)
 "dane_kat_tow__indeks_numeryczny" btree (indeks_numeryczny)
 "dane_kat_tow__indeks_numeryczny_u" btree (indeks_numeryczny, nazwa, nazwa_, cena_z_n, cena_z_b, vat, sww, czy_od_pl_vat)
 "dane_kat_tow__indeks_znakowy_isbn" btree (indeks_znakowy_isbn)
 "dane_kat_tow__indeks_znakowy_isbn_u" btree (indeks_znakowy_isbn, nazwa, nazwa_, cena_z_n, cena_z_b, vat, sww, czy_od_pl_vat)
 "dane_kat_tow__k1_pochodzenie_autor" btree (k1_pochodzenie_autor)
 "dane_kat_tow__k1_pochodzenie_autor_u" btree (k1_pochodzenie_autor, nazwa, nazwa_, cena_z_n, cena_z_b, vat, sww, czy_od_pl_vat)
 "dane_kat_tow__k2_wydawca" btree (k2_wydawca)
 "dane_kat_tow__k2_wydawca_u" btree (k2_wydawca, nazwa, nazwa_, cena_z_n, cena_z_b, vat, sww, czy_od_pl_vat)
 "dane_kat_tow__k3_grupa_tow" btree (k3_grupa_tow)
 "dane_kat_tow__k3_grupa_tow_u" btree (k3_grupa_tow, nazwa, nazwa_, cena_z_n, cena_z_b, vat, sww, czy_od_pl_vat)
 "dane_kat_tow__niewykorzystane1_ilosc" btree (niewykorzystane1_ilosc)
 "dane_kat_tow__nr_wewn_u" btree (nr_wewn, nazwa, nazwa_, cena_z_n, cena_z_b, vat, sww, czy_od_pl_vat)

Table "public.dane_uzytkown_0_plikrob_" - plik roboczy (tymczasowa robocza lista towarowa - struktura identyczna z dane_kat_tow_ (poza kluczem i indeksami). Cyfra w nazwie to numer operatora, do którego plik roboczy należy. W pliku roboczym mogą znajdować się kartoteki, których nie ma w dane_kat_tow_. Przy pobraniu pliku roboczego na specyfikację dokumentu brakujące w dane_kat_tow_ kartoteki zostaną uzupełnione. W zależności od konfiguracji MagAD może być tworzona, jako tymczasowa lub zwykła. Jeśli tymczasowa, możliwa jest praca, jako jeden operator programu w kilku sesjach (na kilku klientach – tak naprawdę - bez ograniczeń ilości klientów) na jednej bazie)

Column	Type	Modifiers
nazwa	character varying(31)	
nazwa_	character varying(30)	
cena_z_n	numeric(28,7)	
cena_z_b	numeric(28,7)	
vat	character varying(5)	
sww	character varying(10)	
czy_od_pl_vat	character varying(13)	
nr_wewn	integer	
indeks_znakowy_isbn	character varying(15)	
indeks_numeryczny	bigint	
dostawca	integer	
miejsce_w_magazynie	character varying(9)	
k1_pochodzenie_autor	character varying(15)	
k2_wydawca	character varying(15)	
k3_grupa_tow	character varying(15)	
k4	character varying(15)	
k5	character varying(15)	
cena_s_n	numeric(28,7)	
cena_s_b	numeric(28,7)	
ilosc	numeric(28,7)	
jednostka	character varying(7)	
ilosc_w_opak	numeric(28,7)	
dostawca_opis	character varying(20)	
il_zamowiona	numeric(28,7)	- pole jest wykorzystywane do przenoszenia informacji o połączeniach dokonanych w pliku roboczym, jeśli plik roboczy pochodzi z analizy towarowej, może być tu zapisany stan aktualny danej kartoteki
ldod1	numeric(29,7)	domyślnie 0 – pole jest wykorzystywane technicznie przez MagAD i nie należy go wypełniać samemu, bo MagAD wyczyści nasze wpisy.

Indexes:

"dane_uzytkown_0_plikrob__oidindeks" UNIQUE, btree (oid)

Table "public.dane_uzytkown_0_praca____" specyfikacja dokumentu aktualnie wypisywanego/poprawianego danego operatora. Cyfra w nazwie to numer tego operatora. Na specyfikacji mogą być tylko towary obecne w dane_kat_tow_. Struktura - jak pliku roboczego.

Column	Type	Modifiers
nazwa	character varying(31)	
nazwa_	character varying(30)	
cena_z_n	numeric(28,7)	
cena_z_b	numeric(28,7)	
vat	character varying(5)	
swv	character varying(10)	
czy_od_pl_vat	character varying(13)	
nr_wewn	integer	
indeks_znakowy_isbn	character varying(15)	
indeks_numeryczny	bigint	
dostawca	integer	
miejsce_w_magazynie	character varying(9)	
k1_pochodzenie_autor	character varying(15)	
k2_wydawca	character varying(15)	
k3_grupa_tow	character varying(15)	
k4	character varying(15)	
k5	character varying(15)	
cena_s_n	numeric(28,7)	
cena_s_b	numeric(28,7)	
ilosc	numeric(28,7)	
jednostka	character varying(7)	
ilosc_w_opak	numeric(28,7)	
dostawca_opis	character varying(20)	
il_zamowiona	numeric(28,7)	- nie wykorzystywane

Indexes:

"dane_uzytkown_0_praca____oidindeks" UNIQUE, btree (oid)

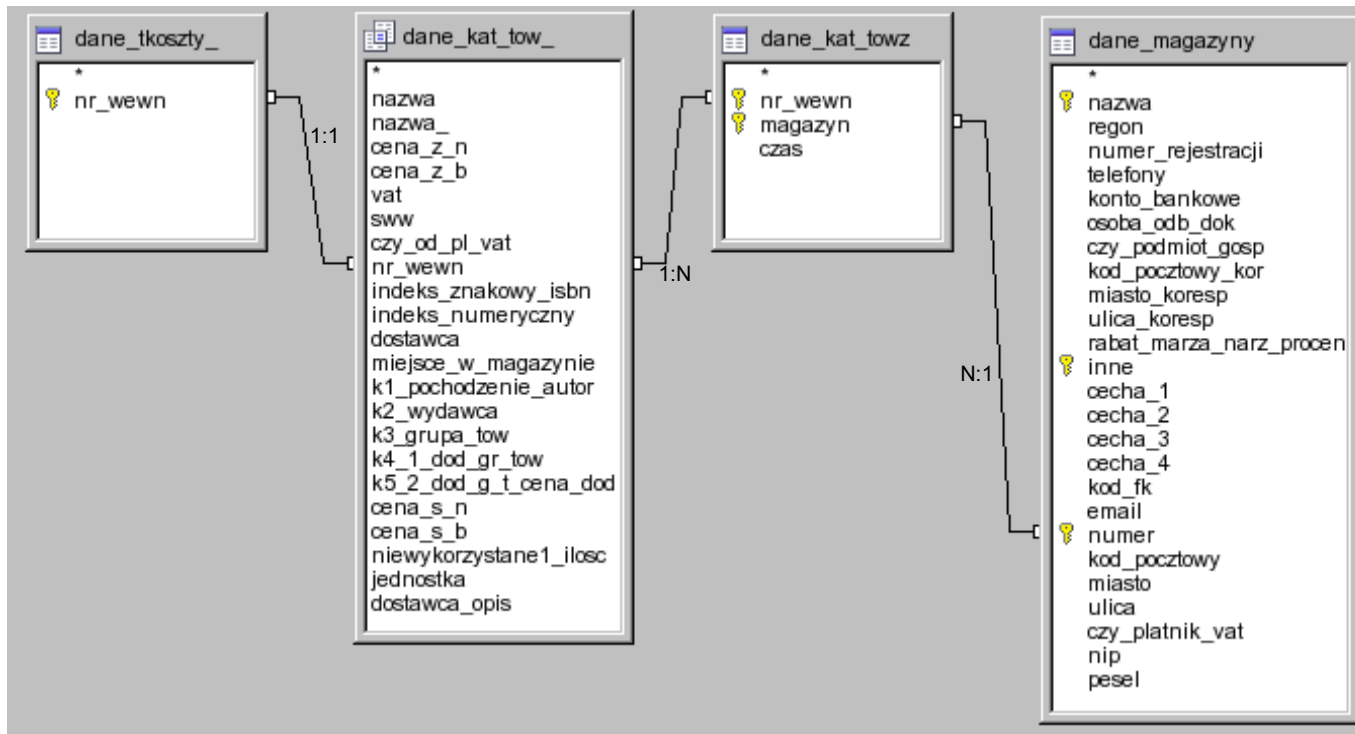
Table "public.dane_kat_towz" - tablica, w której, o ile tak jest ustawione w konfiguracji, zapisywane są informacje, że kartoteka o nr_wewn została zmieniona. Potrzebne jest to w wypadku, kiedy program pracuje na zasadach synchronizacji danych oddziałów (wpisanych do bazy centrali jako magazyny o numerach „magazyn”). Jeśli kartoteka jest zmieniana, usuwane są wszystkie jej wpisy i dopisywany jeden z magazyn=0. Kiedy jednostka zsynchronizuje dane, dodawany jest rekord z jej numerem magazynu - i wtedy już ta jednostka po raz wtóry nie synchronizuje danych. Pole czas zapisywane jest automatycznie wartością domyślną i służy jedynie informacja adminowi.

Column	Type	Modifiers
nr_wewn	integer	not null
magazyn	integer	not null
czas	timestamp without time zone	default now()

Indexes:

"dane_kat_towz_pkey" PRIMARY KEY, btree (nr_wewn, magazyn)

"dane_kat_towz_oidindeks" UNIQUE, btree (oid)



W tabeli dane_tkoszty_gromadzone są numery systemowe (tzw. „wewnętrzne”) kartotek, które są kosztowe (np. do wpisywania kosztu wysyłki na fakturze).

Table "public.dane_1_za2" - dokumenty towarowe zaznaczone (w opcji „Wprowadzanie i zmiany” widać te „ptaszki”)

Column	Type	Modifiers
nrd	integer	not null - numer dokumentu
znd	character varying(4)	not null - znaki rozszerzające numer dokumentu

Indexes:

"dane_1_za2_pkey" PRIMARY KEY, btree (nrd, znd)

Tablica jest tworzona dynamicznie, dla każdego rodzaju dokumentu oddzielna, mogą więc być tablice:

dane_1_za1, dane_1_za2, dane_1_za3, , dane_1_za18

Table "public.dane_dok_plat" - nerw bazy - opisy funkcjonowania poszczególnych rodzajów dokumentów

Column	Type	Modifiers	
numer	smallint	not null	- numer typu dokumentu
typ_opis	character varying(7)		- skrócony opis typu dokumentu (pełne nazwy dokumentów od 1 do 18 znajdują się w pliku konfiguracyjnym programu (konfigur.txt lub konfigur.csv)
typ_kontrahenta	character(1)		- typ kontrahenta ['d' - dostawca, '0' - odbiorca, dla dokumentów typu 12 zawsze 'w', dla 15,18 zawsze 'W']
ko_przeciwstawne	character varying(3)		- w programie projektowano zrobienie podstawowych analiz księgowych, zaniechane - w przypadku księgi przychodów i rozchodów określa, czy do sumowania księgi wchodzi na +, czy -
ko_syntetyczne	character varying(3)		- w programie projektowano zrobienie podstawowych analiz księgowych, zaniechane - w przypadku księgi przychodów i rozchodów (KPR) określa, do której kolumny księgi dokument (dowód) wchodzi - patrz opis KPR
czy_ma	character(1)		- czy do bilansów dany typ dokumentu jest wpływem, czy wypływem
prop_znak	character varying(4)		- znak proponowany przy dopisywaniu dokumentu, o ile opcja używania proponowanych znaków jest włączona w konfiguracji
opis	character varying(30)		- czy dokument jest zablokowany do wystawiania (SIC!), pole może mieć nazwę „czy_nie_uzywac” i być char(1), literka „T” w polu oznacza, że typ dokumentu (rodzaj) jest zablokowany do wystawiania
czy_p_druk	character(1)		- czy dokument ma być drukowany na drukarce dodatkowej, o ile używanie takich drukarek jest ustawione, tylko dla dokumentów 1 - 18 (towarowe)
instrukcje_fk	character varying(250)		- dane konieczne do prawidłowego rozksięgowania dokumentów przy eksporcie do różnych systemów FK (KH, KPR)
Indexes:			
"dane_dok_plat_pkey" PRIMARY KEY, btree (numer)			
"dane_dok_plat_oidindeks" UNIQUE, btree (oid)			

Dokumentów towarowych jest 18 typów, następnie od 100 do 1000 to standardowe dokumenty płatności w MagAD, powyżej 1000 dokumenty kasowe samodzielnie dopisywane przez operatorów (w programie może więc istnieć praktycznie nieograniczona ilość dokumentów kasowych)

Table "public.dane_forma_p2"		
Column	Type	Modifiers
indeks	character varying(15)	not null
opis	character varying(55)	
czy_byl_obrot	character(1)	

Indexes:

- "dane_forma_p2_pkey" PRIMARY KEY, btree (indeks)
- "dane_forma_p2_oidindeks" UNIQUE, btree (oid)

Przykładowa zawartość tabeli dane_dok_plat

numer	typ_opis	typ_kontrahenta	ko_przeciwstawne	ko_syntetyczne	czy_ma	prop_znak	opis	czy_p_druk	instrukcje_fk
1	PA det.	0	730	200	T			N	
2	FA VAT	0	730	200	N			T	
3	FA KOR.	0	730	200	T			N	
4	FA vat.	0	730	200	N			T	
5	FA kor.	0	730	200	T			N	
6	PZ/Fzak	d	330	202	T			N	
7	PZ/Rzak	d	330	202	T			N	
8	WZkorFz	d	330	202	N			N	
9	WZkorRz	d	330	202	N			N	
10	PZ vat	d			T				
11	Pz	d			T				
12	B0	w			T				
13	WZ/dos.	D			N				
14	WZ/odb.	0			N				
15	MM roz.	W			N			T	
16	Zam.odb	0			N				
17	Zam.dos	d			T				
18	MM prz.	w			T				
100	KP	0	101	200	T				
101	KP	d	101	202	T				
102	KW	0	101	200	N				
103	KW	d	101	202	N				
104	WB	0	131	200	T				
105	WB	d	131	202	N				
106	.B0/k	0	131	200	T				
107	.B0/k	d	131	202	T				
108	KP/ods.	0	131	200	T				
109	WB/ods.	0	131	200	T				

Powyżej przedstawiono zawartość tabeli dane_dok_plat. Tabela ta zawiera numery typów dokumentów, które zapisywane są w tablicach dane_1_naglowki, dane_1_ksiega__, dane_1_kasa____. KP, KW - wiadomo. WB - przelewy, .B0/k - bilans otwarcia kontrahenta, z rozszerzeniem „ods.” dokumentu wpłat odsetek, które nie wchodzą do bilansów, chyba, że zadeklarowaliśmy przy robieniu bilansu zobowiązań i należności dla dokumentów towarowych naliczanie odsetek.

Table "public.dane_dostawcy" - kontrahenci, którzy sa naszymi dostawcami - pola mają na tyle sugestywne nazwy, że ograniczę się do opisu kilku

Column	Type	Modifiers
nazwa	character varying(74)	not null - w przypadku odp. konfiguracji programu może zawierać „nick” kontrahenta
inne	character varying(150)	not null - w przypadku odp. konfiguracji programu może zawierać drukowane dane kontrahenta
numer	integer	not null - systemowy numer kontrahenta
kod_pocztowy	character varying(7)	
miasto	character varying(20)	
ulica	character varying(45)	
czy_platnik_vat	character(1)	
nip	character varying(26)	
pesel	character varying(11)	
regon	character varying(14)	
numer_rejestracji	character varying(15)	
telefony	character varying(25)	
konto_bankowe	character varying(74)	
osoba_odb_dok	character varying(44)	
czy_podmiot_gosp	character(1)	- nie używane
kod_pocztowy_kor	character varying(7)	
miasto_koresp	character varying(20)	
ulica_koresp	character varying(45)	
rabat_marża_narz_procent	numeric(28,7)	- jeśli <> 0.0 proponowana marża/narzut w wypadku wpisywania towarów danego dostawcy
cecha_1	character varying(15)	- dodatkowa cecha kontrahenta, wg której można tworzyć zestawienia
cecha_2	character varying(15)	- jw.
cecha_3	character varying(15)	- jw.
cecha_4	character varying(15)	- jw.
kod_fk	character varying(20)	- kod w systemie FK (konto kontrahenta)
email	character varying(150)	- do wersji 3.1 włącznie pole nie istniało
zmieniony	character (1)	- jeśli T, oznacza, że oprócz wprowadzenia go jako nowego, dane były modyfikowane
aktywny	character (1)	- jeśli N, to przy wybieraniu dostawcy do dokumentu nie jest pokazywany na liście

Indexes:

"dane_dostawcy_pkey" PRIMARY KEY, btree (nazwa, inne, numer)
 "dane_dostawcy_numer" UNIQUE, btree (numer)
 "dane_dostawcy_oidindeks" UNIQUE, btree (oid)
 "dane_dostawcy_miasto" btree (miasto)
 "dane_dostawcy_nip" btree (nip)

Do wersji 3.1 taka funkcjonalność nie działa, ale jeśli któras z cech dodatkowych nazywa się „www”, to przy naciśnięciu odp klawisza uruchamiana jest przeglądarka ze stroną kontrahenta

Table "public.dane_odbiorcy" - znaczenia pól analogiczne do dane_dostawcy, z wyjątkiem opisanych poniżej

Column	Type	Modifiers
nazwa	character varying(74)	not null
inne	character varying(150)	not null
numer	integer	not null
kod_pocztowy	character varying(7)	
miasto	character varying(20)	
ulica	character varying(45)	
czy_platnik_vat	character(1)	- nie używane
nip	character varying(26)	
pesel	character varying(11)	
region	character varying(14)	
numer_rejestracji	character varying(15)	
telefony	character varying(25)	
konto_bankowe	character varying(74)	
osoba_odb_dok	character varying(44)	
czy_podmiot_gosp	character(1)	- czy podmiot jest firmą; przy wystawianiu drugiego rodzaju faktur VAT te, które mają kontrahentów z „T” w tym polu, przechodzą przez drukarkę fiskalną, o ile tak ustawiono
kod_pocztowy_kor	character varying(7)	
miasto_koresp	character varying(20)	
ulica_koresp	character varying(45)	- sugerowany przez program rabat dla kontrahenta
rabat_marza_narz_procent	numeric(28,7)	
cecha_1	character varying(15)	
cecha_2	character varying(15)	
cecha_3	character varying(15)	
cecha_4	character varying(15)	
kod_fk	character varying(20)	
email	character varying(150)	
zmieniony	character (1)	- jeśli T, oznacza, że oprócz wprowadzenia go jako nowego, dane były modyfikowane
aktywny	character (1)	- jeśli N, to przy wybieraniu odbiorcy do dokumentu nie jest pokazywany na liście

Indexes:

"dane_odbiorcy_pkey" PRIMARY KEY, btree (nazwa, inne, numer)
 "dane_odbiorcy_numer" UNIQUE, btree (numer)
 "dane_odbiorcy_oidindeks" UNIQUE, btree (oid)
 "dane_odbiorcy_miasto" btree (miasto)
 "dane_odbiorcy_nip" btree (nip)

Table "public.dane_magazyny"		
Column	Type	Modifiers
nazwa	character varying(74)	not null
inne	character varying(150)	not null
numer	integer	not null
kod_pocztowy	character varying(7)	
miasto	character varying(20)	
ulica	character varying(45)	
czy_platnik_vat	character(1)	

- czy magazyn jest główny - wpis „T” oznacza, że tak - wtedy przy odpowiednim ustawieniu konfiguracji program, jako dane firmy może drukować dane tego magazynu na dokumentach, w przeciwnym razie drukowane są dane magazynu, w którym wystawiono dokument; jeśli żaden magazyn nie jest główny, to za główny jest przyjmowany o najniższym numerze, jeżeli jest kilka głównych, za rzeczywisty główny przyjmowany jest jeden z tych głównych, który ma najniższy numer; dla magazynu głównego drukowane są automatycznie zestawienia „dziennie”, „tygodniowe” i „miesięczne” - patrz opis programu

nip	character varying(26)	
pesel	character varying(11)	
regon	character varying(14)	
numer_rejestracji	character varying(15)	
telefony	character varying(25)	
konto_bankowe	character varying(74)	
osoba_odb_dok	character varying(44)	
czy_podmiot_gosp	character(1)	
kod_pocztowy_kor	character varying(7)	
miasto_koresp	character varying(20)	
ulica_koresp	character varying(45)	
rabat_marza_narz_procent	numeric(28,7)	
cecha_1	character varying(15)	
cecha_2	character varying(15)	
cecha_3	character varying(15)	
cecha_4	character varying(15)	
kod_fk	character varying(20)	
email	character varying(150)	
zmieniony	character (1)	- jeśli T, oznacza, że oprócz wprowadzenia go jako nowego, dane były modyfikowane
aktywny	character (1)	- jeśli N, to przy wybieraniu magazynu do dokumentu nie jest pokazywany na liście

Indexes:

"dane_magazyny_pkey" PRIMARY KEY, btree (nazwa, inne, numer)
 "dane_magazyny_numer" UNIQUE, btree (numer)
 "dane_magazyny_oidindeks" UNIQUE, btree (oid)
 "dane_magazyny_cecha_1" btree (cecha_1)
 "dane_magazyny_cecha_2" btree (cecha_2)
 "dane_magazyny_cecha_3" btree (cecha_3)
 "dane_magazyny_cecha_4" btree (cecha_4)
 "dane_magazyny_miasto" btree (miasto)
 "dane_magazyny_miasto_koresp" btree (miasto_koresp)
 "dane_magazyny_nip" btree (nip)

Table "public.dane_nazwa_dn" - nazwy dokumentów dodatkowych

Column	Type	Modifiers
indeks	character varying(15)	not null - typ dokumentu dodatkowego (np. oryginalnego do PZ zakupu)
opis	character varying(55)	- opis ww. typu
czy_byl_obrot	character(1)	- nie używane, ale zapisywane

Indexes:

"dane_nazwa_dn_pkey" PRIMARY KEY, btree (indeks)

"dane_nazwa_dn_oidindeks" UNIQUE, btree (oid)

Table "public.dane_zd_gosp21" - opis zdarzenia gospodarczego, wpisywany do księgi przychodów i rozchodów

Column	Type	Modifiers
indeks	character varying(15)	not null
opis	character varying(55)	
czy_był_obrot	character(1)	

Indexes:

"dane_zd_gosp21_pkey" PRIMARY KEY, btree (indeks)

"dane_zd_gosp21_oidindeks" UNIQUE, btree (oid)

Program korzysta z pola opis, z którego można pobrać opis do pola dane_1_ksiega__.opis, ale potem swobodnie wyedytować.

Relacje operatorów programu z tablicami pojedynczego magazynu

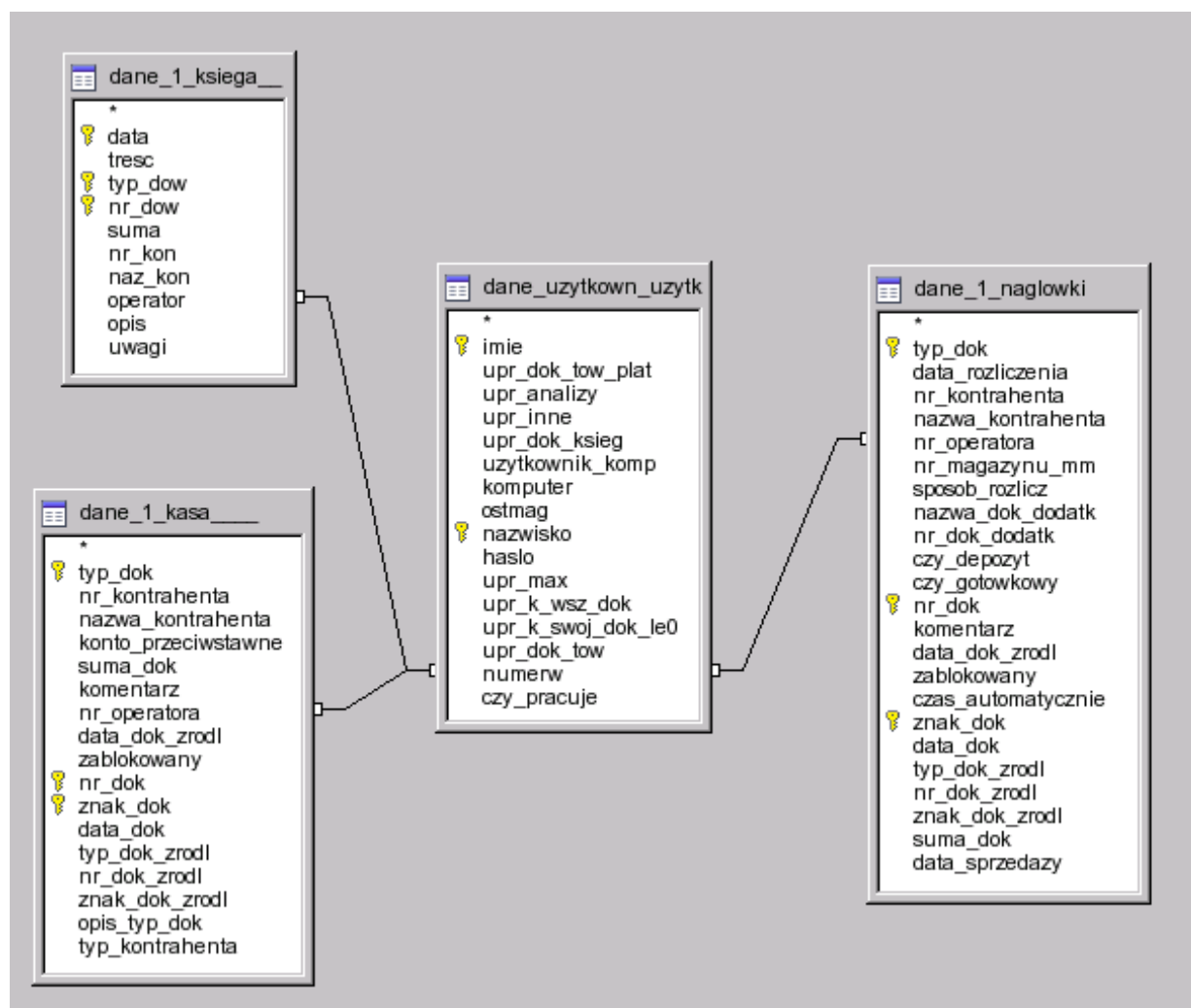


Table "public.dane_uzytkown_uzytkown" lista operatorów programu

Column	Type	Modifiers
imie	character varying(20)	not null
nazwisko	character varying(30)	not null
haslo	character varying(10)	
upr_max	smallint	- pole uprawnień 1
upr_k_wsz_dok	smallint	- pole uprawnień 2
upr_k_swoj_dok_le0	smallint	- pole uprawnień 3
upr_dok_tow	smallint	- pole uprawnień 4
numerw	integer	- numer systemowy operatora
czy_pracuje	character(1)	- wpisane „T” oznacza, że operator pracuje, albo nie wyszedł z programu poprawnie, albo program nie był w stanie po jakimś błędzie zapisać informacji o zakończeniu pracy operatora
upr_dok_tow_plat	smallint	- pole uprawnień nie wykorzystane
upr_analizy	smallint	- pole uprawnień nie wykorzystane
upr_inne	smallint	- pole uprawnień nie wykorzystane
upr_dok_ksieg	smallint	- pole uprawnień nie wykorzystane
uzytkownik_komp	character varying(20)	- wpisywana jest zawartość zmiennej systemowej USER
komputer	character varying(42)	- wpisywana jest zawartość zmiennej systemowej HOST
ostmag	integer	- numer ostatniego magazynu, w którym operator pracował - przy następnym włączeniu magazynem aktualnym dla tego operatora będzie magazyn o numerze zapisanym w tym polu
domdos	integer	- numer ostatniego dostawcy, używanego do wpisania towaru lub dokumentu od dostawcy
czy_nieaktywny	character(1)	- jeśli N, nie jest p[okazywane na liście przy wejściu do programu
osttypdok	smallint	- ostatnio wprowadzany typ dokumentu (patrz tablica „dane_dok_plat”
ostopcja	smallint	- ostatnio używana opcja (pole techniczne)
ostmagdo	integer	- ostatnio używany magazyn przeznaczenia dla MM
pid_magad	integer	- numer PID programu, w którym pracuje dany użytkownik (tylko Linuks i MacOS X)
email	character varying(250)	- email użytkownika
notatka	character varying(250)	- notatka na temat użytkownika
uz_dr_fisk	character varying(250)	- port drukarki fiskalnej, na której pracuje użytkownik (program nie pozwala na jednym komputerze na używanie drukarki fiskalnej więcej, niż jednemu użytkownikowi; jeśli ktoś inny chce pracować, musi się wylogować z Programu, inaczej program odbierze mu możliwość pracy na drukarce fiskalnej

Indexes:

"dane_uzytkown_uzytkown_pkey" PRIMARY KEY, btree (imie, nazwisko)

"dane_uzytkown_uzytkown_numerw" UNIQUE, btree (numerw)

"dane_uzytkown_uzytkown_oidindeks" UNIQUE, btree (oid)

MagAD-SQL używa jednej roli (użytkownika SZBD PostgreSQL) do podłączania się do bazy. Rola musi być właścicielem bazy. Operatorzy zdefiniowani w tablicy dane_uzytkown_uzytkown nie mają nic wspólnego z rolami (użytkownikami) w systemie baz danych PostgreSQL. Operator „Kierownik Systemu” ma zawsze uprawnienia maksymalne plus dodatki (patrz dokumentacja programu), jego „imię” w bazie MagAD jest wpisane jako #001zzKierownik Systemu, w MAGPX #255Kierownik systemu. #xxx oznacza numer znaku ASCII przedstawiony jako liczba dziesiętna xxx. Dzięki tym manewrom Kierownik Systemu najczęściej jest na końcu listy operatorów (serwery bazy na systemach Linuks i MacOS X – w Windows najczęściej wystąpi na pierwszej pozycji).

pola uprawnień integer:

- 1 - and 1<>1 czy max, czy nie
 - and 2<>2 czy może dopisywać i edytować użytkowników (bez kasowania)
 - and 4<>4 czy może blokować i odblokowywać dokumenty
- 2 - and 1<>1 czy może kasować, anulować i zmieniać dok.innych użytkowników
- 3 - and 4<>4-> tylko sprzedaż
 - and 8<>8-> nie może wykonywać analiz, oprócz podsumowania kasy
 - and 16<>16-> nie może wykonywać analiz i podsumowania kasy
 - and 2<>2-> nie może schodzić ze stanem na <0
 - and 1<>1-> nie może edytować, anulować i kasować dokumentów swoich
 - and 32<>32-> nie może oglądać cen i wartości zakupu
 - and 64<>64-> nie może zmieniać dokumentów z inną datą, niż dzisiejsza w komputerze
 - and 128<>128-> nie może edytować elementów klucza głównego katalogu towarów (jeśli globalnie ustawiony jest brak takiej możliwości, to i tak nie może)
- 4 - and 1<>1 LINUX prawo tylko do odczytu danych
 - and 2<>2 LINUX - nie ma prawa do zestawień dziennych, tygodniowych, miesięcznych (tych na Ctrl+F6)
 - and 4<>4 nie ma prawa do ptaszkowania pozycji
 - and 8<>8 nie ma prawa do kasowania dokumentów
 - and 16<>16 nie ma prawa do zamykania remanentu
 - and 32<>32 nie ma prawa do dopisywania dokumentów FA,FAkor.we wprowadzaniu i zmianach
 - and 64<>64 nie ma prawa do rabatowania
 - and 128<>128 nie ma prawa do wchodzenia F3 w kontrolę dokumentów lub do anulowania
 - and 256<>256 nie ma prawa do ingerencji w numery i daty dokumentów

Tablice słownikowe katalogu towarów

Tablice słownikowe po standardowej instalacji w hurtowni książek:

```
public.dane_sww____21
public.dane_vat____21
public.dane_pola_ind_autor__21
public.dane_pola_ind_druga_g1
public.dane_pola_ind_grupa_t7b
public.dane_pola_ind_m_w_mag7b
public.dane_pola_ind_pkwiu__21
public.dane_pola_ind_wydawca21
```

Wszystkie te tablice mają jednakową strukturę:

Column	Type	Modifiers
indeks	character varying(15)	not null - wartość klucza
opis	character varying(55)	- program praktycznie nie korzysta z tego opisu
czy_byl_obrot	character(1)	

Indexes:

- "dane_sww____21_pkey" PRIMARY KEY, btree (indeks)
- "dane_sww____21_oidindeks" UNIQUE, btree (oid)

Ww. tablice słownikowe mogą mieć różne nazwy, w zależności od ustawień MagAD-SQL w pliku dane/lista_in.txt .

Oprócz tego jest jeszcze słownikowa tabela dane_przykl_21, z której opisu można (ale nie trzeba) skorzystać przy wypełnianiu nazw. Tablica ta zawiera listę przykładowych nazw (bywa, że nazwy asortymentu są podobne).

Tabela dane_matr_roz, zawierająca wpisy tworzone przy okazji wpisywania rozliczeń sprzedaży wybranego dostawcy (opcja Analiz specjalnych):

data_od date NOT NULL, - data początkowa rozliczenia
data_do date NOT NULL, - data końcowa rozliczenia
dostawca integer NOT NULL, - numer dostawcy (relacja z dane_dostawcy.numer)
suma numeric(28,7), - suma brutto zakupu sprzedanych towarów
numer integer, - numer rozliczenia

Definicje ograniczeń w bazie dla ww. tabeli:

```
CONSTRAINT dane_matr_roz_pkey PRIMARY KEY (data_od , data_do , dostawca )  
CREATE UNIQUE INDEX dane_matr_roz_oidindeks ON dane_matr_roz USING btree (oid )
```

Dla dostawcy, zdefiniowanego, jako „Matras” w konfiguracji programu dodatkowo tworzona jest tabela „dane_matr_roz”, zawierająca typy i numery eksportów do Matras'a. Przy każdym eksporcie numer narasta. Kiedy eksport obejmuje poprzedni rok, program MagAD lub specjalna Aplikacja, służąca do rozliczeń z „Matras'em” proponuje rozpoczęcie numerowania od nowa.

Tablica danych sterujących dla eksportu do programu „Homer”, instalowana za pomocą programu „upg34_35” przy podaniu opcji 60.

Tabela "public.dane_homer_naglowki"		
Kolumna	Typ	Modyfikatory
lp	integer	
nag	character varying(255)	

Dla wpisów lp od 1 do 6 w wierszach znajdują się następujące informacje w polu „nag”:

- 1 i 2) specyficzne dla Homera informacje w XML
- 3) rodzaj dokumentu przyjęcia w programie Homer
- 4) skrót opisu płatności w programie „Homer”
- 5) opis płatności w programie „Homer”
- 6) ilość dni prolongaty płatności

Tablica zawierać może np dane:

```
select * from DANE_HOMER_NAGLOWKI order by lp;
```

lp	nag
1	<?xml-stylesheet type="text/xsl" href="http://services.azymut.pl/schema/homer/hanmag/im_1.xsl"?>
2	<import xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="http://services.azymut.pl/schema/homer/hanmag/im_1.xsd">
3	PK
4	przelew
5	Przelew
6	14

(6 wierszy)

Program MagAD może prowadzić następujące logi zmian w bazie (wszystkie logi dostępne są w niestandardowych kompilacjach programu, normalnie dostępny jest log zmian wprowadzanych w dokumentach):

- 1) log zmian wprowadzanych w dokumentach gromadzący informację o tym, że dany dokument był zmieniany, a także informację, co było zmieniane – czy nagłówek, czy też specyfikacja dokumentu
- 2) log sesji (wchodzenia i wychodzenia) użytkowników do programu
- 3) szczegółowy log zmian wprowadzanych w katalogu towarów.

Log zmian dokumentów:

Table "public.dane_dok_popr"		
Column	Type	znaczenie
nr_mag	integer	numer magazynu, w którym dokonano zmiany/usunięcia
typ_dok	smallint	numer typu dokumentu (opisy typów w tabeli dane_dok_plat)
nr_dok	integer	numer dokumentu zakodowany z rokiem (patrz opisy tabeli dane_1_naglowki)
zn_dok	character varying(4)	znak dokumentu
nr_zmiany	integer	numer kolejnej zmiany dla danego dokumentu
nr_oper	integer	numer operatora, który wykonał zmianę
rodz_zmiany	character(1)	rodzaj zmiany (S-zmiana specyfikacji,D-dopisanie dokumentu,N-zmiana danych w nagłówku,U-usunięcie)
opis	character varying(50)	opis (m.in opis typu, data, suma nagłówkowa)
data_dok	date	data dokumentu
czas	timestamp without time zone	czas wykonania zmiany

Indexes:

"dane_dok_popr_pkey" PRIMARY KEY, btree (nr_mag, typ_dok, nr_dok, zn_dok, nr

_zmiany)

"dane_dok_popr_oidindeks" UNIQUE, btree (oid)

Log sesji użytkowników:

Table "public.dane_uzytkown_logowani"		
Column	Type	Znaczenie
nr_oper	integer	numer użytkownika, który rozpoczął sesję - w wypadku nieodnalezienia numeru sesji przy wyjściu z programu =-1
nr_oper_wy	integer	numer użytkownika, który zakończył sesję
id_sesji	integer	numer sesji - w wypadku nieodnalezienia numeru sesji przy wyjściu z programu =-1
id_sesji_poprz	integer	numer sesji
pid	integer	ID procesu programu
pidpsql	integer	ID procesu postmaster uruchomionego dla sesji przez serwer bazy
pidpsql_central	integer	ID procesu postmaster uruchomionego dla sesji przez serwer bazy centrali (o ile mamy do czynienia z wersją Synchronizująca dane z centrala i ta synchronizacja jest uruchomiona)
uzytkownik_komp	character varying(50)	nazwa systemowego użytkownika (\$USER)
komputer	character varying(50)	systemowa nazwa komputera (\$HOSTNAME)
oper_haslo	character varying(10)	hasło, z którym zalogował się operator (uzytkownik MagAD)
oper_naz	character varying(51)	nazwa operatora
czas	timestamp without time zone	czas rozpoczęcia sesji
czas_wy	timestamp without time zone	czas zakończenia sesji

Indexes:

"dane_uzytkown_logowani_oidindeks" UNIQUE, btree (oid)

"dane_uzytkown_logowani_id_sesji" btree (id_sesji)

Log zmian w katalogu towarów:

Pierwsze 24 pola mają te samo znaczenie, co w katalogu towarów.

Każda najmniejsza zmiana, usunięcie itp. Pozostawia w logu ślad dla tego samego ID_ZMIANY - wpisana jest postać towaru sprzed zmiany i po zmianie. Nie są ujmowane w nim grupowe zmiany VAT na pozycjach.

typ_zmiany	smallint	5000 - usunięcie pojedynczego towaru, 3000 - korygowanie powtórzeń indeksów, 4000 - zerowanie indeksów większych od podawanej wartości, 6000 - usuwanie towarów bez obrotu, 1000 - zmiana w tabeli stanów (np. cena sprzedaży), może być połączona ze zmianą w katalogu towarów, 2000 - zmiana pojedynczego towaru w katalogu towarów (w tablicy dane_kat_tow_), -1000 - zmiana cen sprzedaży na takie, jakie są w pliku roboczym (z poziomu zmian w pliku roboczym)
mag_akt	integer	numer magazynu, który był aktualny w momencie wykonania zmiany
uzytkownik	integer	numer użytkownika, który wykonał zmianę
id_zmiany	integer	numer kolejny zmiany - jeśli w trakcie zmiany zmieniane jest wiele towarów, mają to samo ID_ZMIANY
id_wpisu	integer	not null default nextval('dane_kat_towl_id_wpisu_seq'::regclass) - autonumer wpisu do logu
czas_zmiany	timestamp without time zone	czas wykonania zmiany

Indexes:

"dane_kat_towl_oidindeks" UNIQUE, btree (oid)

"dane_kat_towl_id_zmiany" btree (id_zmiany)

Struktura tablicy „dane_rap_remanentowe” (jej znaczenie dla wersji synchronizującej dane opisane jest w pliku „rozn34_35.pdf”):

Column	Type	Modifiers
nr_mag	integer	not null (numer magazynu, którym jest jednostka podległa w centrali)
typ_rem	smallint	not null (typ remanentu: 0-remanent zwykły, 1-cząstkowy)
nr_rem	integer	not null (numer remanentu w ramach magazynu w nr_mag i typu remanentu)
poz_rem	integer	not null (pozycja w remanencie o zadanym nr_rem w ramach magazynu w nr_mag i typu remanentu)
czas_pocz	timestamp	data i godzina początku remanentu
czas_konc	timestamp	data i godzina końca remanentu
nazwa	character varying(31)	-> dane_kat_tow_.nazwa
nazwa_	character varying(30)	-> dane_kat_tow_.nazwa_
brak	numeric(28,7)	jeśli > 0 to brak remanentowy
nadw	numeric(28,7)	jeśli > 0 to nadwyżka remanentowa
indeks	bigint	-> dane_kat_tow_.indeks_numeryczny
nr_wewn	integer	-> dane_kat_tow_.nr_wewn
c_sp_br	numeric(28,7)	-> dane_NR_MAG_stany_mg.csb
c_sp_net	numeric(28,7)	-> dane_NR_MAG_stany_mg.csn
c_z_br	numeric(28,7)	-> dane_kat_tow_.cena_z_b
c_z_net	numeric(28,7)	-> dane_kat_tow_.cena_z_n
k5_2_dod_g_t_cena_dod	character varying(15)	-> dane_kat_tow_.k5_2_dod_g_t_cena_dod (w tym polu może być skonfigurowana w MagAD dodatkowa cena sprzedaży - w zależności od konfiguracji w wielkości netto lub brutto, stąd to pole się tu znalazło)
miejsce_w_magazynie	character varying(9)	-> dane_kat_tow_.k5_2_dod_g_t_cena_dod (w tym polu może być skonfigurowana w MagAD dodatkowa cena sprzedaży - brutto, stąd to pole się tu znalazło)
vat	varchar(5)	-> dane_kat_tow_.vat
il_ks	numeric(28,7)	-> dane_NR_MAG_stany_mg.ilosc
il_rem	numeric(28,7)	-> dane_NR_MAG_stany_mg.il_rem_akt
wyeksportowany	char(1)	czy remanent został wyeksportowany (T lub N)

Wszystkie odniesienia danych „->” należy rozumieć, jako dane, które w chwili zamykania remanentu znajdowały się w tablicach, do których jest to odniesienie. Symbol NR_MAG oznacza numer magazynu, w którym przeprowadzany był remanent (dane pobierane są z bazy lokalnej jednostki podległej). Wszystkie ceny zasadniczo w bazie pierwotnej oraz VAT mogą ulec zmianie, stąd dane te są eksportowane do tej tablicy. Zawartość pól: czas_pocz, czas_konc, wyeksportowany - oczywiście dla danego klucza głównego są takie same.

Tylko w wersji synchronizującej dane z innymi serwerami MagAD.

Tablica przechowująca dane o włączenia systemu automatycznych zmian specyfikacji sprzedaży (np..promocji).

Znaczenie pól dokładnie opisane jest w pliku „rozn_magpx_34_35.pdf”, poza kilkoma, których opis jest poniżej.

Tabela tworzona jest automatycznie na podstawie listy magazynów.

Table "public.dane_inf_mods"			
Column	Type	Modifiers	Opis
nr_m	integer	not null	numer magazynu oddziału w centrali
naz_mag	character varying(35)		numer magazynu oddziału w centrali (pierwsze 35 znaków tej nazwy)
ozn_pocz	timestamp without time zone		
pocz	timestamp without time zone		
ozn_konc	timestamp without time zone		
konc	timestamp without time zone		
symb_zm_wpr	character varying(10)	domyślnie ''::character varying	

Indexes:

"dane_inf_mods_pkey" PRIMARY KEY, btree (nr_m)

Tablica przechowująca dane o wykonywanych przez MagAD niektórych zapytaniach, które można zmienić w tej tabeli, w polu „tresc”.

Zawiera np.. dane systemu automatycznych zmian specyfikacji sprzedaży (np..promocji).

Tablica jest w relacji N:1 z „dane_rodzaje_sazss” (pole „dane_skry_sql.symb_zm_wpr” i „dane_rodzaje_sazss.symbol”), ale „dane_rodzaje_sazss” jest słownikiem, w którym powinny, ale nie muszą być wszystkie symbole zmian wprowadzania. Centala posiada niepusta tabelę „dane_inf_mods”, „dane_rodzaje_sazss” i „dane_skry_sql”.

Jednostka podległa importuje (przy okazji importu przesunięć zaopiarowania) jedną, przeznaczoną dla niej opcję systemu zmian wprowadzania.

W jednostce podległej tablica „dane_inf_mods” i „dane_rodzaje_sazss” może być pusta, a w „dane_skry_sql” są tylko rekordy z jedną wartością „symb_zm_wpr”.

Table "public.dane_skry_sql"			
Column	Type	Modifiers	Opis
uzyj	boolean		czy używać danej linii wpisu
symb	character varying(50)	not null	symbol wpisu (w jego ramach wykonywane będą rozkazy SQL w kolejności lp)
lp	double precision	not null	kolejność wykonania polecenia SQL zawartego w kolumnie „tresc”
tresc	text		polecenie SQL – teksty i symbole mogą być wymieniane przez program MagAD –
są traktowane, jak pewna forma zmiennych i są podawane z poziomu procedury wywołującej zapytania z zakresu danego „symbolu”			
symb_zm_wpr	character varying(10)	niepusty domyślnie ''::character varying	

Indexes:

"dane_skry_sql_pkey" PRIMARY KEY, btree (symb, lp)

Tabela zawierające słownik rodzajów zmian systemu wprowadzania sprzedaży.

Komentarze są na poprzedniej stronie.

Tabela "public.dane_rodzaje_sazss"		
Kolumna	Typ	Modyfikatory
symbol	character varying(10)	niepusty
opis	character varying(250)	
opis_	character varying(250)	
tabl_dod	character varying(70)	domyślnie '::character varying

Indeksy:

"dane_rodzaje_sazss_pkey" PRIMARY KEY, btree (symbol)

Tablica istotna w jednostce podległej. W tabeli zbierane są informacje, dotyczące towarów, które napływają do jednostki podległej i są w promocji.

Pole „kod_sazss” zawiera dane z pola „symb_zm_wpr”, które w danej chwili zawiera tabela „dane_inf_mods” w pou „symb_zm_wpr”.

Raport z tabeli automatycznie pokazywany jest dla wartości „czas_zatw” NULL przy starcie programu.

Tabela stanowi rodzaj logu, więc nie jest zmieniana, jeśli ktoś w jednostce zmieni/skasuje dokument przesunięcia (jest to operacja zasadniczo merytorycznie Nieuzasadniona). Zaimportowane dokumenty są blokowane do zmian, więc program „broni się” przed niepożądaną operacją. Użytkownicy o maksymalnych Uprawnieniach mogą jednak zdjąć znacznik blokady i wyedytować/skasować dokument. Zakłada się jednak, że tylko osoby wiedzące, jakie są tego konsekwencje Merytoryczne, będą to robić.

Tabela "public.dane_nowe_sazss"		
Kolumna	Typ	Modyfikatory
indeks	character varying(15)	indeks_numeryczny z kat.towarów
nazwa1	character varying(31)	nazwa z kat.towarów
nazwa2	character varying(30)	nazwa_ z kat.towarów
cena_br	character varying(36)	cena sprzedaży brutto w jednostce podległej
opis	character varying(250)	opis (np. podane rabaty, jeśli SAZSS to promocja)
kod_sazss	character varying(10)	j.w.
nr_wewn	character varying(10)	nr_wewn z kat.towarów
mag_skl	integer	numer magazynu jednostki podległej w jej bazie
nr_przes	integer	nr przesunięcia zatowarowania
znak_przes	character varying(4)	znak przesunięcia zatowarowania
data_przes	date	data przesunięcia zatowarowania
czas_zatw	timestamp without time zone	czas zatwierdzenia przez operatora, opisanego w polach poniżej
zatw_uz_nr	integer	numer operatora zatwierdzającego
zatw_uz_naz	character varying(55)	nazwa operatora zatwierdzającego
czas_importu	timestamp without time zone	domyślnie now() - czas, w którym importowano przesunięcie