

*PROJEKT SKRÓCENIA CZASU W JAKIM KLIENT OCZEKUJE NA
POTWIERDZENIE TERMINU REALIZACJI ZAMÓWIENIA.*

*Etap
Potwierdzanie zamówień klientowi .*

*Specyfikacja systemu służącego do przyśpieszenia procesu
potwierdzania zamówień klientowi i redukcji problemów
związanych z dostawami materiałów.*

Autor Radek Kosobucki
Brodnica dnia 07-11-2017r.

Spis treści :

1. Wstęp
2. Tryby pracy serwera
3. Podstawowe funkcje serwera potwierdzeń
4. Funkcje w aplikacji dla zakupów – porównanie ze starą wersją
5. Rodzaje komunikatów
6. Bezpieczeństwo
7. Co Dalej ?

1.Wstęp

Opisywany system powstał w ramach projektu modyfikacji procedur obiegu informacji o złożonych zamówieniach pomiędzy firmą SITS a klientem. Jednym z ważniejszych problemów podnoszonych podczas spotkań grupy było stworzenie mechanizmów przyspieszających określenie terminu możliwej realizacji zamówienia wraz z większą kontrolą nad procesem. W toku spotkań zdecydowano o rozszerzeniu istniejącego „systemu informacji o zagrożeniach w realizacji” o nowe funkcje związane z automatycznym potwierdzaniem zamówień do klientów.

Po analizie istniejącego rozwiązania pod kątem możliwości rozszerzenia o zakres danych potrzebnych do odwzorowania wszystkich zagrożeń związanych z implementacją funkcji automatycznego potwierdzania zdecydowano o:

- zmianie podstawowej bazy danych z ACCESS na POSTEGRESQL.
- rezygnacji z rozproszonych obliczeń generowanych przez skrypty (wiele procesów wymiany danych przyspieszających wykonanie) oraz rezygnacji z przeliczania danych w aplikacji ZAKUPY (model wymiany danych Peer to Peer) na poczet nowocześniejszego i generującego mniej problemów rozwiązania KLIENT-SERWER.
- utworzeniu od nowa aplikacji serwera zajmującej się importem danych z ORACLE , przetwarzaniem danych , wysyłaniem informacji drogą e-mail , generowaniem raportów i danych źródłowych dla Klientów (aplikacja ZakupyNEW lub serwer WWW utworzony przy pomocy języka node.js), automatycznym przesuwaniem zleceń produkcyjnych, potwierdzaniem terminów zamówień klientowi.
- zmodyfikowaniu aplikacji ZakupyNEW w celu umożliwienia obsługi rozszerzonych funkcjonalności i wymiany danych z nową bazą danych POSTEGRESQL .

W toku podejmowanych działań uzyskano znaczne rozszerzenie funkcjonalności wraz z przyspieszeniem realizacji poszczególnych etapów obliczeń i mniejszą awaryjnością systemu.

2. Tryby pracy serwera

a) Tryby pracy ciągłej

Serwer skompilowany jako usługa systemowa pracuje w dwóch trybach :

– w momencie gdy na którymkolwiek komputerze uruchomiony jest program do raportowania problemów zakupowych (nowa wersja aplikacji „ZakupyNEW”) to obliczenia uruchamiane są w odstępach cztero – minutowych.

- w przypadku braku aktywnych użytkowników, serwer przechodzi w tryb monitorowania – uruchamiając się w odstępach godzinnych.

Brak możliwości ręcznego uruchomienia serwera.

Po stronie bazodanowej serwis w interwałach pięcio-sekundowych wykonuje zapytanie w bazie POSTEGRESQL / zakupy:

```
Select case when d.state='STOP' then 0 else case when b.in_progress=true then case
when b.start_update-interval '5 minutes'<current_timestamp then 0 else case when
c.il>0 then case when a.start_update+interval '4 minutes' < current_timestamp then 1
else 0 end else case when a.start_update+interval '1 hour' < current_timestamp then
1 else 0 end end end else case when c.il>0 then case when a.start_update+interval '4
minutes' < current_timestamp then 1 else 0 end else case when a.start_update+interval
'1 hour' < current_timestamp then 1 else 0 end end end end stat from (select
start_update from datatbles where table_name='server_progress') a,(select
in_progress,start_update from datatbles where table_name='ifs_CONN') b,(SELECT
count(application_name) il FROM pg_catalog.pg_stat_activity where
application_name='CLIENT') c,(SELECT typ ,case when current_timestamp between
(current_date + cast(to_char(start_idle,'HH24:MI:SS') as time)-interval '5 minutes')
and (current_date + cast(to_char(stop_idle,'HH24:MI:SS') as time)+interval '5 minutes'
) then 'STOP' else 'RUN' end state from serv_idle where
current_timestamp<=current_date + cast(to_char(stop_idle,'HH24:MI:SS') as
time)+interval '15 minutes' order by current_date +
cast(to_char(stop_idle,'HH24:MI:SS') as time) limit 1 ) d”
```

W wyniku którego sprawdzana jest aktywnych ilość użytkowników, czy serwis jest w trakcie działania oraz czy nie zostały włączone sygnały związane z zatrzymaniem działania.

b) Przerwy w działaniu serwera:

W programie występują cztery tryby przerwania działania:

- **Tryb automatyczny** – wprowadzony został dla ominięcia problemów związanych z wewnętrznymi procedurami IFS oraz obliczeń systemu premiowego (raporty produkcja na czas dla magazynu materiałów i reszty produkcji). W głównej bazie serwera POSTEGRSQL w tabeli ‘public.serv_idle’ zapisana jest lista codziennych wyłączeń serwera. Na pięć minut przed planowanym rozpoczęciem i pięć minut po zakończeniu przerwy serwer wchodzi w stan czuwania. (połączenia z główną bazą danych POSTEGRESQL są możliwe / łączenie z bazą danych ORACLE jest zabronione).

Struktura tabeli public.serv_idle:

typ character varying(50) COLLATE pg_catalog."default", start_idle timestamp without time zone, stop_idle timestamp without time zone

Na tą chwilę zaplanowane są następujące przerwy:

	typ character varying (50)	start_idle timestamp without time zone	stop_idle timestamp without time zone
1	Aktualizacja struktur	0001-01-01 22:15:00 BC	0001-01-01 23:30:00 BC
2	MRP	0001-01-01 03:00:00 BC	0001-01-01 03:30:00 BC
3	CRP	0001-01-01 04:30:00 BC	0001-01-01 05:00:00 BC
4	Termin MAG	0001-01-01 18:00:00 BC	0001-01-01 19:00:00 BC
5	Termin ALL	0001-01-01 00:00:00 BC	0001-01-01 01:00:00 BC

- Wstrzymanie działania związane z przesunięciami dat produkcji lub wystąpienia błędów w trakcie obliczeń serwisu.

W momencie wystąpienia zmian uniemożliwiających określenie poprawnych wyników serwis zatrzymuje działanie na okres siedmiu minut.. (żadne maile ,przesunięcia dat produkcji ,potwierdzenia zamówień do klientów nie są wykonywane)

- Przerwa działania serwisu związana z pracami konserwacyjnymi w ORACLE / IFS.

Ze względów bezpieczeństwa wyłącznie serwisu potwierdzeń powinno nastąpić na 5 – 10 minut przed planowanymi pracami w ORACLE /IFS.

Do wymuszonego przerwania działania serwisu można doprowadzić dwoma sposobami:

Przy pomocy aplikacji monitorującej działanie serwisu Server_control poprzez użycie przycisku „Zatrzymaj SERWER” - aplikacja w ciągu 5 sekund przekaże do usługi systemowej sygnał zatrzymania (brak możliwości zatrzymania działania serwisu w momencie pracy serwera).

Ponowne uruchomienie serwisu nastąpi po naciśnięciu przycisku „Uruchom serwer”

Magazyn danych dla Zakupy

Logi z obliczeń na serwerze

start_date	end_date	errors_found
2017-11-02 11:06	2017-11-02 11:07	0
2017-11-02 11:02	2017-11-02 11:03	0
2017-11-02 10:57	2017-11-02 10:59	0
2017-11-02 10:53	2017-11-02 10:55	0
2017-11-02 10:49	2017-11-02 10:51	0

Status magazynów danych

table_name	last_modify	start_update	in_progress	updt_errors
mag	2017-11-02 11:06	2017-11-02 11:10	<input checked="" type="checkbox"/>	<input type="checkbox"/>
worker2	2017-11-02 11:07	2017-11-02 11:10	<input checked="" type="checkbox"/>	<input type="checkbox"/>
worker3	2017-11-02 11:07	2017-11-02 11:10	<input checked="" type="checkbox"/>	<input type="checkbox"/>
worker1	2017-11-02 11:07	2017-11-02 11:10	<input checked="" type="checkbox"/>	<input type="checkbox"/>
worker5	2017-11-02 11:07	2017-11-02 11:10	<input checked="" type="checkbox"/>	<input type="checkbox"/>
worker4	2017-11-02 11:07	2017-11-02 11:10	<input checked="" type="checkbox"/>	<input type="checkbox"/>
worker_all	2017-11-02 11:07	2017-11-02 11:10	<input checked="" type="checkbox"/>	<input type="checkbox"/>
work_cal	2017-11-02 11:10	2017-11-02 11:10	<input type="checkbox"/>	<input type="checkbox"/>
server_progress	2017-11-02 11:07	2017-11-02 11:10	<input checked="" type="checkbox"/>	<input type="checkbox"/>
cust_ord	2017-11-02 11:06	2017-11-02 11:10	<input checked="" type="checkbox"/>	<input type="checkbox"/>
day_qty	2017-11-02 11:07	2017-11-02 11:07	<input type="checkbox"/>	<input type="checkbox"/>
day_qty_fs	2017-11-02 11:07	2017-11-02 11:07	<input type="checkbox"/>	<input type="checkbox"/>
TR_sendm	2017-11-02 11:07	2017-11-02 11:07	<input type="checkbox"/>	<input type="checkbox"/>
send_mail	2017-11-02 11:07	2017-11-02 11:07	<input type="checkbox"/>	<input type="checkbox"/>

Aktywni użytkownicy

username	state	application_name	backend_start	state_change	datname
ZANPRZ		CLIENT			zakupy
TOMKOP		CLIENT			zakupy
RADKOS	active	SERVER_CONT...	2017-11-02 10:13	2017-11-02 11:10	zakupy
RADKOS	idle	ZAKUPY_CONN...	2017-11-02 11:10	2017-11-02 11:10	zakupy
RADKOS	idle	ZAKUPY_CONN...	2017-11-02 11:10	2017-11-02 11:10	zakupy
RADKOS	idle	ZAKUPY_CONN...	2017-11-02 11:10	2017-11-02 11:10	zakupy
RADKOS	idle	CLIENT	2017-11-02 08:46	2017-11-02 11:10	zakupy
RADKOS	idle	ZAKUPY_CONN...	2017-11-02 11:10	2017-11-02 11:10	zakupy

Zatrzymaj SERWER

Uruchom SERWER

Lista adresatów e-mail

Status serwera

STATUS SERWERA :
WORK

LICZBA POŁĄCZEŃ : 17

Dane z daty :
2017-11-02 11:10:05

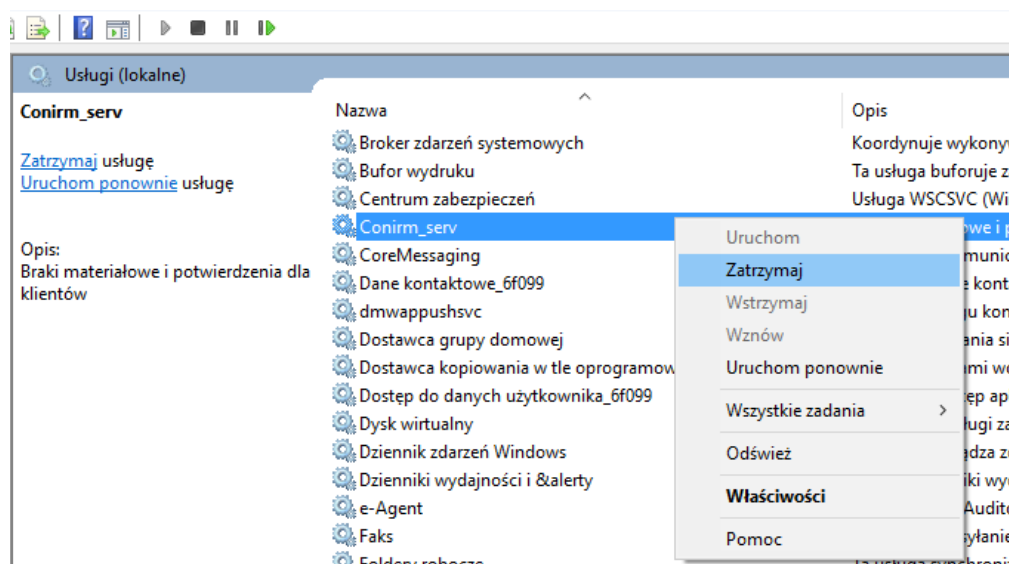
Przy pomocy klienta POSTEGRESQL.

Zatrzymanie serwisu po uruchomieniu komendy: `"UPDATE public.datatbles SET last_modify=current_timestamp, start_update=current_timestamp, in_progress=true WHERE table_name ='ifs_CONN'"`

Ponowne uruchomienie po komendzie: `"UPDATE public.datatbles SET last_modify=current_timestamp, start_update=current_timestamp, in_progress=false WHERE table_name ='ifs_CONN'"`

- Przerwa działania serwisu związana z pracami konserwacyjnymi w POSTEGRESQL.

W przypadku prac nad bazą danych POSTEGRESQL znajdującej się w lokalizacji 10.0.1.29 należy zatrzymać usługę systemową „Confirm_serv”. W wyniku zatrzymania wewnętrzny zegar usługi zostanie zatrzymany i biblioteka przestanie sprawdzać konieczność ponownego uruchamiania obliczeń. Uwaga : brak możliwości zatrzymania działania serwisu w momencie pracy serwera- przed rozpoczęciem czynności konserwacyjnych należy upewnić się jaki jest status serwisu w aplikacji Server_control .



Ponowne uruchomienie serwisu po włączeniu funkcji uruchom za menu podręcznego w oknie usługi.

3. Podstawowe funkcje serwera potwierdzeń

Serwer został skompilowany w konfiguracji „ANY CPU” umożliwiającej ominięcie problemów związanych z różnymi systemami / rodzajami procesorów , jako aplikacja konsolowa która przy pomocy biblioteki **Topshelf** **wer 4.0.0.0** może być uruchamiana jako usługa systemowa – nie wymagająca żadnych poświadczeń do domeny SITS (uwierzytelnienia na poziomie baz ORACLE i POSTEGRESQL wystarczają do poprawnego działania serwisu). Dla potrzeb usługi systemowej przewidziano tylko dwa zdarzenia **WhenStarted** ,**WhenStopped** – praca w trybie wielu instancji serwisu została zablokowana (specjalna właściwość biblioteki Topshelf). Sytuacja jeśli przypadkowo serwis zostanie uruchomiony na dwóch komputerach jednocześnie może doprowadzić do błędów w działaniu i ostatecznie zablokować wykonanie – dla ominięcia problemu w 99% przypadków, usługa sprawdza ilość połączeń wykazaną w tabeli **pg_catalog.pg_stat_activity** z nazwą aplikacji **CONFIRM_SERVICE** jeśli ich ilość będzie większa od jeden działanie zostanie zablokowane. Do wszystkich niezbędnych obliczeń wykorzystano bazę danych **POSTEGRESQL ver9.6** . Do łączenia z bazą danych ORACLE wykorzystano natywną bibliotekę **Oracle.ManagedDataAccess.Client** w **wersji 4.122.1.0** natomiast do łączenia z bazą danych POSTEGRESQL wykorzystana została biblioteka **Npgsql wersja 3.2.5.0** , do wysyłania poczty e-mail wykorzystano systemową bibliotekę **System.Net.Mail**, format wysyłanej poczty zgodny ze standardem **XHTML 1.0 Transitional**. Użycie w/w bibliotek w sposób bardzo istotny przyspieszyło czasy wykonania poszczególnych funkcji serwisu.

Dla potrzeb kontroli i utrzymania poprawnej kolejności wykonania wszystkich funkcji programu ,utworzono tabelę **public.datatbles** w której każde z wewnętrznych zadań i transakcji raportuje zdarzenia rozpoczęcia / zakończenia/ zaistnienia błędu.

Dodatkowo serwis posiada tablicę logów działania w której można podejrzeć kolejność wykonania wszystkich zadań ,czasy wykonania, występujące błędy wraz z tzw. Wartością ‘**throwed exception**’ ,oraz zdarzenia serwera mail (tabela **public.server_query**). Czas przechowywania logów ze względu na wielkość został ograniczony do 24 godzin.

Serwer przy pomocy asynchronicznie uruchamianych funkcji oblicza jednocześnie kilka typów danych wynikowych:

- na podstawie danych źródłowych ORACLE dotyczących danych magazynowych (tabela **public.mag**) oraz bilansów potrzeb z matrycą bilansów dla wszystkich indeksów (tabela **public.demands**) dla aplikacji ZakupyNEW uruchamiane są raporty dotyczące między innymi : nie pasujących dostaw lub ich braków (widok **public.zak_dat** tabela źródłowa **public.data**) wraz z obróbką i aktualizacją statusów sygnałów od kupców zgromadzonych w tabeli **public.potw** oraz skalkulowane ilości sztuk z brakami na poszczególnych gniazdach (zmaterializowany widok **public.braki_gniazd** ,tabela **public.day_qty**) ,raport z wartościami zamówień (zmaterializowany widok **public.bilans_val**), brakujące indeksy z określonym zagrożeniem dla realizacji (zmaterializowany widok **public.braki_poreal**, **public.formatka**, **public.formatka_bil**)

- dla potrzeb serwera braków materiałowych obliczane są macierze potencjalnych zagrożeń w poszczególnych zleceniach produkcyjnych (procesy **WORKER** tabela **public.ord_demands**), do obliczeń związanych z przeliczaniem leadtime pobierane są z IFS dane dotyczące obowiązującego kalendarza (tabela **public.work_cal**), informacje dotyczące zamówień klienta pobierane są z ORACLE przy pomocy zapytania którego wynik można podejrzeć w tabeli **public.cust_ord**. Na podstawie powyższych tabel z ORACLE przeprowadzana jest kalkulacja zamówień / zleceń dla których występują braki materiałowe według zasad:

Przed rozpoczęciem obliczeń dokonujemy kontroli zaimportowanych danych dotyczących zleceń vs pobrane zsumowane potrzeby materiałowe (ord_demands vs demands) jeśli coś nie pasuje dla wybranych indeksów uruchamiamy procedury aktualizujące (procesy nazwane WORKER)

a)Zaczynamy obrabiać zaimportowane z IFS dane:

- dla celów określenia linii tworzących nie rozłączne zestawy dokonujemy operacji UPDATE dla zamówień klienta. Kryterium czy dane linie tworzą zestaw jest określenie czy na tej samej dacie obiecanej znajdują się różne indeksy zgromadzone w ramach tego samego numeru linii w zamówieniu. Wyniki w których status linii jest różny od „Aktywna” lub linia nie jest powiązana z nagłówkiem DOP bądź indeks nie jest produkowany w SITS (nie indeksy zaczynające się na 5,6,2) są pomijane.
- tworzymy podstawowy widok do obliczeń zawierający dane z tabel dla nie pasujących dostaw lub ich braków (public.data) oraz zamówień klienta aktywnych w systemie (public.cust_ord) .

b) Dla każdego zlecenia:

- obliczamy daty końcowych realizacji w oparciu o dostawy brakujących komponentów.
- sprawdzamy czy przesunięcie związane z inną niż oczekiwaną dostawą spowoduje zmianę daty obiecanej klientowi.
- określamy ilość problemów materiałowych występujących w danym DOP i zleceniu produkcyjnym.
- dla każdej dostawy i indeksu określamy bilans i stan magazynowy
- tworzymy macierz obliczeniową w której występują numery indeksów, dostaw, brakujących ilości.

c) w oparciu o tak spreparowane dane dokonujemy:

- sortowania wszystkich incydentów materiałowych po jak najdalszej obliczonej dacie dostawy końcowej (przyjęto logikę iż najważniejsze niedobory materiałowe powodują przesunięcia zleceń na jak najodleglejsze terminy)
- dodajemy kolejny poziom sortowania ustawiając na początku zlecenia dla których nowa data produkcji nie będzie miała wpływu na zmianę daty obiecanej, kolejne zlecenia dla których nastąpi zmiana daty obiecanej sortujemy według metody FIFO.

d) zaczynamy obliczenia bilansów i wskazujemy zlecenia dla których występuje brak materiałowy dla poszczególnych rekordów według określonych w punkcie c) metod sortowania, w przypadku gdy rekord związany z nagłówkiem lub zleceniem zostaje zaliczony do braku i zawiera więcej niż jeden brak materiałowy to system zaczyna sprawdzanie bilansów komponentów dla wszystkich znajdujących się poniżej rekordu braków dla danego DOP / zlecenia produkcyjnego. Dla przyspieszenia obliczeń wykorzystujemy wspomnianą w pkt. b) macierz obliczeniową.

e) przeliczone rekordy z wykazaniem brakiem materiałowym kopiujemy do tymczasowej tabeli public.braki_tmp

f) na podstawie danych z tabeli public.braki_tmp dokonujemy aktualizacji tabeli public.braki

g) historia zmian w tabeli public.braki gromadzona jest w tabeli public.braki_hist.

h) Zapisy w tabeli public.braki_hist kasowane są po upływie miesiąca.

g) na podstawie aktualnych braków (public.braki), danych dotyczących serii zero, zgromadzonych zestawów tworzymy zmaterializowany widok (public.to_mail) będący podstawą do określenia bieżących sygnałów.

h) Na podstawie widoku public.to_mail przeprowadzamy aktualizację i dodanie nowych rekordów w tabeli public.mail. Dokonujemy sprawdzenia aktualności komunikatów dla reszty sygnałów zgromadzonych w tabeli mail.

i) Ze względu na kluczową rolę Tabeli public.mail w procesach wysyłania wiadomości e-mail, przesuwania zleceń i potwierdzania zamówień do zdarzeń związanych z aktualizacjami ,modyfikacjami ,usunięciami dodano wyzwalacz zapisujący historię zmian rekordów do tabeli public.mail_hist

j) Zapisy w tabeli public.mail_hist kasowane są po upływie miesiąca od wprowadzenia.

k) Na podstawie tabel public.mail ,public.Kontakty ,public.cust_ord generowany jest widok fill_sendmail na podstawie którego zapełniana jest tabela public.send_mail gromadząca wszystkie informacje do sygnałów wysyłanych na wybrane adresy e-mail. Na podstawie tej tabeli system wie który adresat otrzymał już daną wiadomość , blokując wysłanie e-mail'a o tej samej treści częściej niż raz dziennie. (chyba że nastąpią jakieś zmiany).

h) Na podstawie tabeli cust_ord i send_mail generowana jest lista zamówień do potwierdzenia w IFS. (nie potwierdzone zamówienia które nie występują w send_mail)

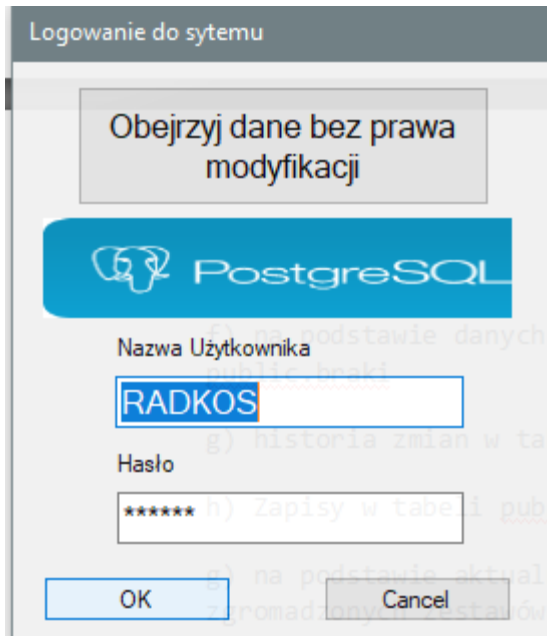
Dla zamówień w których wystąpił problem z potwierdzeniem generowany jest sygnał e-mail do koordynatora rynku oraz stosowny zapis w tabeli public.conf_mail_null

i) Na podstawie tabeli public.mail generowana jest lista zleceń DOP do przesunięcia. Każde przesunięte zlecenie wraz z datą źródłową i nowo planowaną oraz kodem przyczyny zapisywane jest w tabeli public.mod_date.

4. Funkcje w aplikacji dla zakupów – porównanie ze starą wersją.

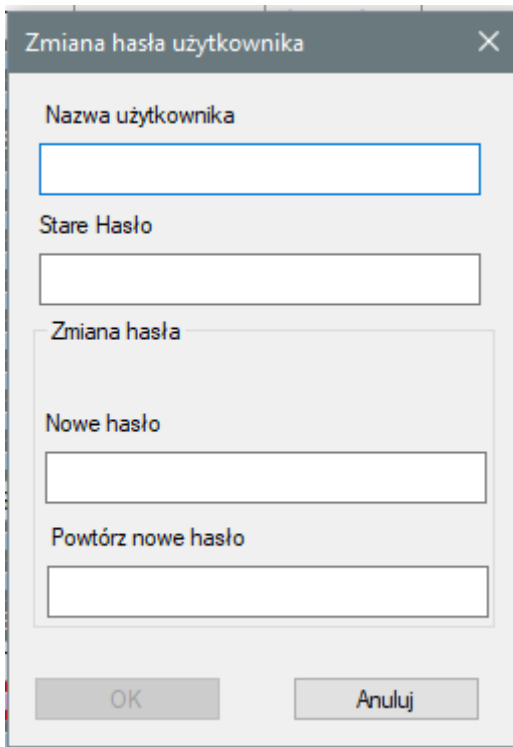
Ze względu na zmianę źródłowej bazy danych do aplikacji dodano:

a) ekran logowania – dzięki któremu rozwiązano problem bezpieczeństwa danych





The screenshot shows a dialog box titled "Logowanie do systemu". At the top, there is a button labeled "Obejrzyj dane bez prawa modyfikacji". Below this is the PostgreSQL logo. The dialog contains two input fields: "Nazwa Użytkownika" with the text "RADKOS" entered, and "Hasło" with masked characters "*****". At the bottom, there are two buttons: "OK" and "Cancel".

b) ekran zmiany hasła do POSTEGRESQL



The screenshot shows a dialog box titled "Zmiana hasła użytkownika". It contains four input fields: "Nazwa użytkownika", "Stare Hasło", "Nowe hasło", and "Powtórz nowe hasło". At the bottom, there are two buttons: "OK" and "Anuluj".

c) dodano listwę z możliwością wpisania dodatkowych informacji – które będą wyświetlane we wiadomościach e-mail.(możliwość wyboru z listy komunikatów lub wpisanie informacji własnej)

Dodatkowa informacja:		 										
<div> <div></div> <div>Komponent nie zamawiany</div> <div>Tkanina wycofana z kolekcji</div> <div>Brak możliwości przyspieszenia</div> <div>Problemy u dostawcy</div> <div>Proszę o użycie wersji niepalnej</div> <div>Przeprowadzono inwentaryzację na indeksie</div> <div>Komponent zniszczony/zagubiony na produkcji</div> </div>		<table border="1"> <tr> <td>zaj</td> <td>dmd_ty</td> </tr> <tr> <td>\$</td> <td>Zlec;</td> </tr> <tr> <td>\$</td> <td>BRAK</td> </tr> <tr> <td>\$</td> <td>DOP:Z</td> </tr> <tr> <td>\$</td> <td>Zam.:Z</td> </tr> </table>	zaj	dmd_ty	\$	Zlec;	\$	BRAK	\$	DOP:Z	\$	Zam.:Z
zaj	dmd_ty											
\$	Zlec;											
\$	BRAK											
\$	DOP:Z											
\$	Zam.:Z											

d) Rozbudowano panel raportujący. Do wszystkich wykresów dołożono zdarzenia , które po kliknięciu na podzbiór danych w wykresie pokazują dane szczegółowe w formie tabelarycznej. Dla większości wykresów kołowych dołożono możliwość przeglądania wyników dotyczących wskazanych dat.



e)Dołożono link do serwisu WWW systemu premiowego produkcji.

f)Dołożono komunikat związany z brakami zamówień w wyznaczonym leadtime tzw. „braki w gwarantowanej dacie”.

5. Rodzaje komunikatów

a.) Dane podstawowe

Serwer kontroluje dostawy oraz zamówienia i ich integralność pod kątem:

- braków materiałowych związanych z datami dostaw lub nie wystarczającymi ilościami.
- zagrożeń materiałowych związanych z brakami zamówień zakupu.
- opóźnieniami w dostawach.
- dostawami realizowanymi w dniu produkcji.
- ustalonymi datami serii zero.
- planowanymi datami realizacji zamówień dla indeksów w których planowana jest seria zero (realizacja kolejnych partii produktu możliwa po trzech tygodniach od Serii zero).
- integralnością dat produkcji w ramach określonego zestawu.
- konfliktu ustalonych dat produkcji względem daty obiecanej.
- konieczności przesunięcia daty obiecanej dla zamówienia z potwierdzonym brakiem materiałowym.
- konieczności przesunięcia daty obiecanej lub zmiany konfiguracji dla zamówienia z sygnałem do zmiany komponentu na alternatywny.
- konieczności przesunięcia daty wysyłki dla zamówienia z potwierdzonym brakiem materiałowym.
- konieczności przesunięcia daty wysyłki dla zamówienia z sygnałem do zmiany komponentu na alternatywny w przypadku gdy osoba zajmująca się zamówieniem nie zdecydowała się na podmianę konfiguracji i dokonała przesunięcia daty obiecanej na termin w którym brak materiałowy zostanie uzupełniony.
- braku możliwości realizacji / potwierdzenia zamówienia z komponentem wycofanym z siatki zakupów (poza kolekcją) dla którego występują nie wystarczające ilości na magazynie
- braku możliwości potwierdzenia zamówienia w związku ze źle uzupełnionymi odniesieniami w zamówieniu klienta.
- braku możliwości potwierdzenia ze względu na *status zamówienia, błędne daty obiecane, braki materiałowe, użycie komponentów wycofanych z kolekcji / nie zamawianych*

Powyższe komunikaty w zależności od docelowych adresatów pokazywane są w aplikacji ZakupyNEW lub wysyłane w formie wiadomości e-mail.

W przypadku komunikatów związanymi z dostawami – widoczne są one w aplikacji ZakupyNEW (możliwość pobrania z lokalizacji <http://10.0.1.29:3000/installNew/setup.exe>)

b.) Konfiguracja serwisów e-mail

Konfiguracja serwisu e-mail związana z adresatami wiadomości możliwa jest z poziomu aplikacji Server_control.

Po naciśnięciu przycisku „Lista adresatów e-mail” uzyskujemy dostęp do tabeli public.kontakty.

Magazyn danych dla Zakupy

Logi z obliczeń na serwerze

start_date	end_date	errors_found
2017-11-03 13:45	2017-11-03 13:46	0
2017-11-03 13:41	2017-11-03 13:42	0
2017-11-03 13:37	2017-11-03 13:38	0
2017-11-03 13:33	2017-11-03 13:34	0
2017-11-03 13:29	2017-11-03 13:30	0

Logi Started at 2017-11-03 09:23:37
CREATE END 00:00:00.0506743
START cust_ord 00:00:00.0516768
START DMND 00:00:00.0922034
START ACC 00:00:00.0922034
START KALUPDATE 00:00:00.0942029
Start KAL 00:00:00.5045211
READY KAL 00:00:01.0765980
READY ACC 00:00:01.9637861
START MAG 00:00:01.9878062
READY KALUPDATE 00:00:08.2640635
Start_update WRK_CENTER
Cust_ord start_load FetchSize:131072 Started:00:00:11.0107562
END WRK_CENTER

Status magazynów danych

table_name	last_modify	start_update	in_progress	updt_errors
mag	2017-11-03 13:49	2017-11-03 13:49	<input type="checkbox"/>	<input type="checkbox"/>
worker2	2017-11-03 13:46	2017-11-03 13:49	<input checked="" type="checkbox"/>	<input type="checkbox"/>
worker3	2017-11-03 13:46	2017-11-03 13:49	<input checked="" type="checkbox"/>	<input type="checkbox"/>
worker1	2017-11-03 13:46	2017-11-03 13:49	<input checked="" type="checkbox"/>	<input type="checkbox"/>
worker5	2017-11-03 13:46	2017-11-03 13:49	<input checked="" type="checkbox"/>	<input type="checkbox"/>
worker4	2017-11-03 13:46	2017-11-03 13:49	<input checked="" type="checkbox"/>	<input type="checkbox"/>
worker_all	2017-11-03 13:46	2017-11-03 13:49	<input checked="" type="checkbox"/>	<input type="checkbox"/>
work_cal	2017-11-03 13:49	2017-11-03 13:49	<input type="checkbox"/>	<input type="checkbox"/>
cust_ord	2017-11-03 13:50	2017-11-03 13:49	<input type="checkbox"/>	<input type="checkbox"/>
server_progress	2017-11-03 13:46	2017-11-03 13:49	<input checked="" type="checkbox"/>	<input type="checkbox"/>
day_qty	2017-11-03 13:46	2017-11-03 13:46	<input type="checkbox"/>	<input type="checkbox"/>
day_qty_fs	2017-11-03 13:46	2017-11-03 13:46	<input type="checkbox"/>	<input type="checkbox"/>
TR_sendm	2017-11-03 13:46	2017-11-03 13:46	<input type="checkbox"/>	<input type="checkbox"/>
send_mail	2017-11-03 13:46	2017-11-03 13:46	<input type="checkbox"/>	<input type="checkbox"/>

Zatrzymaj SERVER

Uruchom SERVER

Lista adresatów e-mail

Status serwera

STATUS SERWERA :
WORK

LICZBA POŁĄCZEŃ : 15

Dane z daty :
2017-11-03 13:49:23

Organizacja danych wygląda następująco:

W kolumnie country_coor wprowadzamy kod kraju ,w polu ifs_user wprowadzamy ID użytkownika z IFS , natomiast w kolumnie mail - adres mailowy

Lista Adresatów e-mail

country_coor	ifs_user	mail
DK	ILOKWE	ilona.ostrowska@sits.pl
EE	AGNMIE	agnieszka.mierzejewska@sits.pl
EE	AMESWI	amelia.swiezawska@sits.pl
EG	EGIPT	handlowy@sits.pl
ES	JOARZE	joanna.rzepnikowska@sits.pl
ES	MONPIO	monika.piotrowska-rzenska@sits.pl
FI	ADAHAR	adam.harajda@sits.pl
FI	ILOKWE	ilona.ostrowska@sits.pl
FI	ANNBAB	anna.babinska@sits.pl
FR	JOARZE	joanna.rzepnikowska@sits.pl
FR	LUCWEB	lucyna.weber@sits.pl
FR	MONPIO	monika.piotrowska-rzenska@sits.pl
GB	PATBUD	patrycjusz.budka@sits.pl
GB	KARSAR	karolina.samowska@sits.pl
GR	GRECJA	handlowy@sits.pl
HK	HONG KONG	handlowy@sits.pl
HU	ADAHAR	adam.harajda@sits.pl
HU	ANNBAB	anna.babinska@sits.pl
HU	ILOKWE	ilona.ostrowska@sits.pl
IE	PATBUD	patrycjusz.budka@sits.pl
IE	KARSAR	karolina.samowska@sits.pl
IL	IZRAEL	handlowy@sits.pl

Edycja, dodawanie rekordów, usuwanie wierszy i kopiowanie danych do tabeli umieszczonej w formularzu Lista Adresatów e-mail jest intuicyjna i podlega takim samym zasadom jak w przypadku tabel Access (do zdarzeń DataGridView dołączono stosowne komendy inicjujące UPDATE, INSERT, DELETE na tabeli źródłowej POSTEGRESQL).

System w przypadku powstania sygnału gromadzi listę adresatów według klucza kraju zamówienia oraz użytkownika wprowadzającego zamówienie.

W przypadku zamówień do koordynatorów SERII_ZERO kolumna country_coor powinna być równa 'Seria Zero'



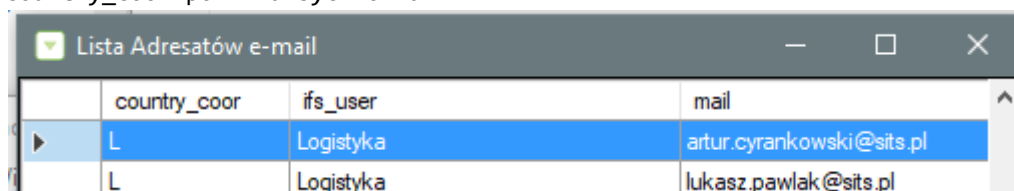
	country_coor	ifs_user	mail
▶	Seria Zero	PIOTUS	piotr.tuszynski@sits.pl

W przypadku adresatów dla sygnałów zamówień rozpoczynających się od litery 'Z' lub sygnałów związanych z brakami możliwości potwierdzenia (zakupy) kolumna country_coor powinna być równa 'Z'



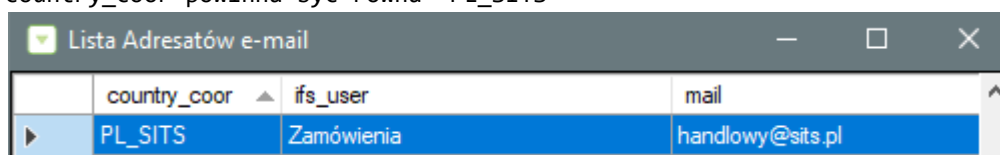
	country_coor	ifs_user	mail
▶	Z	Zakupy	zakupy@sits.pl

W przypadku adresatów dla sygnałów dotyczących zmiany daty wysyłki kolumna country_coor powinna być równa 'L'



	country_coor	ifs_user	mail
▶	L	Logistyka	artur.cyrankowski@sits.pl
	L	Logistyka	lukasz.pawlak@sits.pl

W przypadku adresatów zajmujących się wewnętrznymi zamówieniami SITS kolumna country_coor powinna być równa 'PL_SITS'



	country_coor	ifs_user	mail
▶	PL_SITS	Zamówienia	handlowy@sits.pl

Dla sygnałów związanych z nieokreślonymi koordynatorami dla kraju skonfigurowanie poczty e-mail powinno wyglądać.



	country_coor	ifs_user	mail
▶	TR	TURCJA	handlowy@sits.pl
	SK	SŁOWACJA	handlowy@sits.pl
	SI	SŁOWENIA	handlowy@sits.pl
	QA	KATAR	handlowy@sits.pl
	LU	LUKSEMBURG	handlowy@sits.pl
	CA	KANADA	handlowy@sits.pl
	AR	ARGENTYNA	handlowy@sits.pl

c.) Rodzaje sygnałów w aplikacji ZakupyNEW

Sygnały w aplikacji ZakupyNEW generowane są przez kupców w wyniku zdarzeń występujących bezpośrednio w IFS i stanowią bazę do obliczeń serwera braków i potwierdzeń.

Z poziomu aplikacji wyróżniamy sygnały uruchamiające zdarzenia wewnątrz serwera potwierdzeń i przesunięć materiałowych:

- **POTWIERDZONE** - kupiec potwierdza brak możliwości skorygowania planu dostaw na indeksie w wyznaczonym zakresie dat. Sygnał dla serwera o konieczności przesunięcia zleceń produkcyjnych. Informacja mailem w zależności od zakłóceń (zmiana daty obiecaną, zmiana daty wysyłki, zmiana daty wysyłki lub obiecaną dla linii stanowiących zestaw, prośba o zmianę planowanej serii ZERO lub zmianę zamówień po serii zero). Jeśli w wyniku obliczeń zagrożone są zamówienia nie potwierdzone wysyłane są informacje związane z brakiem możliwości potwierdzenia (po upływie dnia od aktywacji).

- **CZEKAM NA INFORMACJE** - kupiec próbuje wynegocjować nowe terminy, ilości w wyznaczonym zakresie dat. Serwer nie przesuwa zamówień zagrożonych realizacją lub wysyłką, dopuszczalne są korekty dat produkcji w stosunku do których brak jakichkolwiek uwag, jeśli w wyniku obliczeń zagrożone są zamówienia nie potwierdzone wysyłane są informacje związane z brakiem możliwości potwierdzenia i rozmowach prowadzonych przez kupca (po upływie dnia od aktywacji). Ważność tego typu sygnału ustawiona została na 48 godzin - po upływie tego terminu informacja jest automatycznie kasowana.

- **UŻYJ KOMPONENTU ALTERNATYWNEGO** - kupiec proponuje zmianę indeksu w zleceniach, lub zmianę daty obiecaną (jeśli istnieje taka możliwość). Informację o konieczności zamiany konfiguracji otrzymują koordynatorzy rynku i osoby wprowadzające zamówienie (tzw. Koordynatorzy zamówienia) w stosunku do zamówień dla których serwer wyliczy brak pierwotnego materiału i stwierdzi że zamówienie zagrożone jest realizacją w zakresie obiecaną klientowi daty wykonania. Podobne sygnały generowane są w przypadku linii stanowiących zestaw, i zamówień serii ZERO. W stosunku do zamówień które można przesunąć bez ryzyka zakłóceń związanych z datą obiecaną serwer dokonuje przesunięcia zleceń na termin planowanej dostawy pierwotnego materiału (w przypadku gdy przeplanowanie wchodzi w interakcje z datami wysyłki, generowany jest stosowny monit o konieczności zmiany daty wysyłki do działu logistyki). Jeśli w wyniku obliczeń zagrożone są zamówienia nie potwierdzone wysyłane są informacje związane z brakiem możliwości potwierdzenia (po upływie dnia od aktywacji).

- **KOMPONENT NIE ZAMAWIANY** - informacja o użyciu komponentu wycofanego - generowany sygnał dla zamówień w których serwer wykaże ujemny bilans (wystąpienie braku). Konieczność anulacji zamówienia lub zmiany komponentów. Jeśli w wyniku obliczeń zagrożone są zamówienia nie potwierdzone wysyłane są informacje związane z brakiem możliwości potwierdzenia i rozmowach prowadzonych przez kupca (po upływie dnia od aktywacji). W przypadku gdy mamy do czynienia z permanentnym wycofaniem komponentu data oczekiwanego czasu realizacji powinna być nie mniejsza od wartości 720 dni. (serwis będzie monitorował wszystkie zamówienia ustawione w prawie dwu letnim horyzoncie czasowym)

- **BRAK** - brak sygnału serwer niczego nie przesuwa wysyła jedynie informacje o brakach możliwości automatycznego potwierdzenia

Wyróżniamy następujące typy zdarzeń dla kupca materiałowego:

- *brakujące ilości* - zdarzenie wywoływane w momencie ujawnienia ujemnego bilansu materiałowego dla pozycji w sytuacji gdy w późniejszym czasie istnieją zaplanowane dostawy. W aplikacji domyślnie sformatowane jest na kolor czerwony. Dla tego typu zdarzenia istnieje możliwość uruchomienia sygnałów : POTWIERDZONE , CZEKAM NA INFORMACJE , UŻYJ KOMPONENTU ALTERNATYWNEGO , KOMPONENT NIE ZAMAWIANY ,BRAK

- *braki w gwarantowanej dacie* - zdarzenie wywoływane w momencie ujawnienia ujemnego bilansu materiałowego dla pozycji w terminie nie przekraczającym gwarantowanej daty dostawy oraz sytuacji braku dostaw w późniejszym czasie. W aplikacji domyślnie sformatowane jest na kolor żółty dla nie opóźnionych potrzeb materiałowych , w przypadku gdy w wyniku braku materiałowego mamy do czynienia z opóźnieniem realizacji sygnał przybiera kolor różowy (dziś mija dzień kiedy materiał jest potrzebny) lub brązowy (nastąpiło opóźnienie w planowanej realizacji). W przypadku zaistnienia takiego sygnału system oczekuje na utworzenie zamówienia zakupu lub przesłanie sygnału CZEKAM NA INFORMACJE , KOMPONENT NIE ZAMAWIANY.

- *opóźniona dostawa* - zdarzenie utworzone w sytuacji dostaw z nieaktualną / błędną datą. W aplikacji domyślnie sformatowane jest na kolor biały dla dostaw nie generujących braków, w przypadku gdy w wyniku braku materiałowego mamy do czynienia z opóźnieniem realizacji sygnał przybiera brązowy (nastąpiło opóźnienie w planowanej realizacji). Należy poprawić dane dotyczące dostawy.

- *dzisiejsza dostawa* - zdarzenie związane z bieżącymi dostawami. W aplikacji domyślnie sformatowane jest na kolor biały dla dostaw nie generujących braków, w przypadku gdy w wyniku braku materiałowego mamy do czynienia z opóźnieniem realizacji sygnał przybiera kolor różowy (dziś mija dzień kiedy materiał jest potrzebny) - konieczna weryfikacja

- *dostawa na dzisiejsze ilości* - dostawa nastąpi w dniu na kiedy komponent jest potrzebny. W aplikacji domyślnie sformatowane jest na niebieski. Jeśli istnieje taka możliwość bezpieczniej by było nieznacznie przyspieszyć dostawę.

d.) Rodzaje sygnałów w aplikacji na poziomie serwera braków materiałowych i potwierdzeń.

W serwerze potwierdzeń generujemy możemy wyróżnić trzy poziomy przetwarzania sygnałów:

- sygnały pochodzące z aplikacji ZakupyNEW - przetwarzane przy pomocy mechanizmów zaimplementowanych dla tabeli public.braki (macierz zamówień z aktualnymi brakami).

- do informacji związanymi z aktualnymi brakami materiałowymi dokładane są sygnały wewnętrzne związane z korektami dat produkcji (pilnowanie integralności dla zestawów, przesunięcia dat produkcji dla zleceń w ramach Serii Zero), sprawdzana jest aktualność przesuniętych wcześniej zleceń (określenie czy występuje niezgodność daty obiecanej lub daty wysyłki względem bieżącej daty produkcji). Wyniki powyższych działań możemy prześledzić w tabeli public.mail.

- Na podstawie sygnałów z tabeli public.mail i listy adresatów e-mail generowane są informacje mailowe do zamówień oraz uruchamiane są funkcje związane z automatycznymi przesunięciami zleceń i potwierdzaniem bez problemowych zamówień klienta.

6. Bezpieczeństwo

Ze względu na dostępność aplikacji na serwerze WWW (warunek konieczny dla automatycznych aktualizacji aplikacji ZakupyNEW) utworzono system oparty na poświadczeniach.

Na poziomie bazy POSTGRESQL powstały następujące role grup /loginów

- NoPostgresql – login umożliwia przeglądanie tabel ,widoków niezbędnych do uzyskania informacji na poziomie aplikacji ZakupyNEW i serwera WWW. Brak możliwości generowania komend UPDATE, DELETE, INSERT i potwierdzania sygnałów dotyczących dostaw, brak możliwości zmian Schematu bazy danych.
- PLANNERS – grupa z której dziedziczą aktywni użytkownicy aplikacji ZakupyNEW, umożliwia przeglądanie tabel ,widoków niezbędnych do uzyskania informacji na poziomie aplikacji ZakupyNEW i serwera WWW. Dla niektórych tabel włączone komendy UPDATE, DELETE, INSERT i możliwe potwierdzania sygnałów dotyczących dostaw, brak możliwości zmian Schematu bazy danych.
- postgres – login na poziomie SUPERUSER'a – może dowolnie modyfikować i tworzyć bazę.

7. Co dalej.

a) Rozszerzenie funkcjonalności systemu

Serwer w stosunku do wcześniejszych rozwiązań zaczął kontrolować problemy materiałowe związane ze zleceniami powstałymi jako zamówienia magazynowe (IKEA) – istnieje możliwość rozszerzenia funkcjonalności o automatyczne przetwarzanie problemów związanych z zamówieniami MTS (Made to STOCK).

b) Portal WWW z zamówieniami.

Umożliwienie sprawdzenia wszystkich swoich zamówień pod kątem przeprowadzonych działań. (bieżące sygnały o brakach, potencjalne problemy materiałowe, historyczne sygnały o brakach, przesunięcia produkcji wraz z przyczynami ,sprawdzenie czy zamówienie jest potwierdzone a jeśli nie to z jakiej przyczyny).