

## Wstęp

System Golem ma „otwartą” architekturę – oznacza to że każdy (oczywiście każdy kto potrafi) może ingerować w system na poziomie baz danych. Może zapisywać w bazie dane do tworzenia zleceń, odczytywać wyniki pomiarów etc.

Trzeba w tym miejscu bardzo wyraźnie powiedzieć: sama znajomość tabel bez znajomości systemu Golem i sposobu jego pracy i działania to za mało aby dokonać integracji z innym oprogramowaniem.

Zdarza się (wiemy o tym z licznych telefonów) że osoba X z firmy Y próbuje dokonać insertu danych do Golema albo interpretacji danych z Golema li tylko na podstawie opisu tabel.

Tak się nie da !

Golem znacząco różni się od typowych systemów i aby cokolwiek zrobić trzeba rozumieć jak cały system jest zbudowany.

Trzeba na przykład mieć świadomość tego że w systemie poza bazą SQL jest jeszcze podręczna baza stacji do której dostęp ma tylko program stacji a np. zmiana statusu realizowana jest przez stację na podstawie rozkazów z programu klienckiego i w bazie SQL zapisywany jest jedynie wynik.

Nie da się więc zmienić statusu przez „zapisanie czegoś” bezpośrednio w bazie danych.

### Pliki baz danych

Golem_mes.fdb	główny plik systemu zawierający konfigurację i dane pomiarowe
Foto_baze.fdb	pomocnicza baza danych zawierająca zdjęcia
Golem_station.fdb	pomocnicza baza z binarnym obrazem trendów i liczników stacji – tej bazy nie należy otwierać !!!!

### Logowanie do bazy

Bazy danych systemu golem obsługiwane są przez serwer FireBird. System nasz nie korzysta z ról – hasła są danymi w bazie. Jednak aby połączyć się z bazą wymagane jest hasło i login. Jest to domyślne hasło systemu.

Login: **SYSDBA** hasło: **MASTERKEY**

### Tabele konfiguracji

tabela CFG\_CFG

Główna tabela konfiguracji systemu, tabela CFG\_CFG jest NIETYKALNA, jej nieautoryzowana zmiana spowoduje destabilizację systemu.

tabela CFGV\_STATUS i CFGV\_STATUSR

Tabela z definicjami statusów i statusów rozszerzonych. Tabele są tylko do odczytu – uzupełniane po zmianie konfiguracji.

Tabele te mogą się przydać do tworzenia złączeń, tak samo jak dane z tabeli SV\_CRR gdzie są np. nazwy maszyn (nadzorców)

ID – id rekordu

SID – numer statusu

Name – nazwa statusu

### Wspólne pola nagłówkowe dla wielu tabel

Nadzorca

Pole SV (SMALLINT) określa zawsze numer nadzorca którego dotyczy zapis.

W tabeli SV\_CURRENT numerem nadzorca jest ID rekordu

Czas

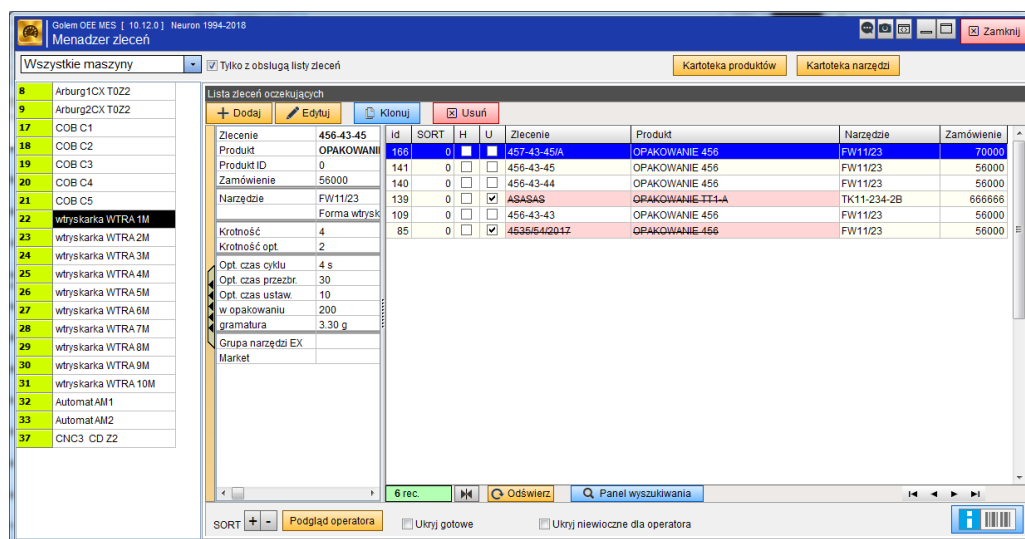
W wielu tabelach poza polami określającymi czas typu Timestamp występują pola określające czas jako rok, miesiąc, dzień, godzina i zmiana robocza co pozwala na filtrowanie danych tak po polach czas jak i po tych polach

TYR	SMALLINT,	rok
TMC	SMALLINT,	miesiąc
TDY	SMALLINT,	dzień
THR	SMALLINT,	godzina
TSHIFT	SMALLINT,	zmiana robocza ( numer zmiany)

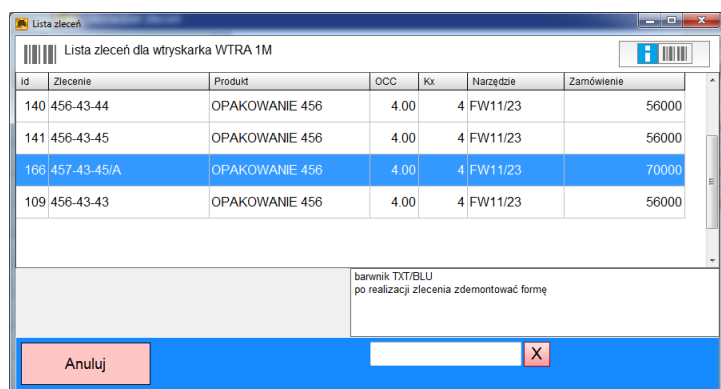
## Zlecenie produkcyjne

Cele integracji systemu Golem z innym oprogramowaniem mogą być różne, ale najczęściej chodzi o „wpisanie” do systemu zleceń produkcyjnych. Aby tego dokonać konfigurujemy nadzorcę w taki sposób który pozwala na wybór zlecenia produkcyjnego z listy zleceń.

Zlecenia umieszczamy w tabeli TASK\_WAITING. To ta sama tabela w której system przechowuje zlecenia (wzorce zleceń) widoczne w menadżerze zleceń:



Umieszczenie w tabeli zlecenia ( z odpowiednimi parametrami TUSED i THIDE) spowoduje że będzie ono widoczne dla operatora w panelu operatorskim:



Co się stanie gdy operator dokona wyboru zlecenia z tej listy ?

- 1) do stacji zostanie wysłane polecenie stworzenia zlecenia z numerem id z tabeli task\_waiting
- 2) stacja założy nowy rekord w tabeli task z przepisаныmi wszystkimi parametrami z tabeli task\_waiting
- 3) stacja zmieni pole TUSED w tabeli task\_waiting na 1 (rekord oznaczony jako użyty )
- 4) stacja ustawia numer ID zlecenia (id nowego rekordu w tabeli task ) oraz wszystkie parametry zlecenia w tabeli sv\_crr
- 5) stacja wpisuje komunikat do tabeli sv\_ev ( powiadomienie o stworzeniu nowego zlecenia )
- 6) stacja informuje programy klienckie o konieczności zmodyfikowania widocznego zlecenia na podstawie zmian w sv\_crr

### Produkt a zlecenie

System zawiera kartotekę produktów (tabela task\_product). Dodając z poziomu programu nowe zlecenie musimy najpierw wybrać produkt z którego definicji zostaną przepisane odpowiednie parametry potrzebne do stworzenia nowego zlecenia.

Zwróćmy uwagę na słowo „przepisane”. Pomimo tego że w polu product\_id zostanie zapisane id produktu to nie jest to relacja w rozumieniu baz danych – zlecenie nie jest złączeniem z tabelą task\_product – zlecenie (tabela task czy task\_waiting) ma zestaw identycznych pól z parametrami które są przepisane z tabeli produktu.

Dzięki temu użytkownik Golema może stworzyć zlecenie z użyciem produktu dla którego przypisano czas cyklu 17.76 sekundy i zmienić ten czas np. na 22s.

Główną decyzją jaką musimy podjąć to decyzja czy będziemy zapisywać do tabeli task\_waiting zlecenia bez odnoszenia się do tabeli task\_product (pozostawimy pole product\_id z wartością 0) godząc się na utratę pewnych funkcjonalności systemu takich jak analiza realizacji poszczególnych produktów czy karta informacyjna produktu czy zdecydujemy się na pełną obsługę relacji produkt -> zlecenie ( choć przypomnijmy, nie ma tu relacji w sensie złączenia tabel)

Możemy wybrać też wariant pośredni – kartoteka produktów będzie utrzymywana w systemie Golem a system nadrzędny będzie miał jej kopie - będzie znał numer ID – i zapisze ten numer w tabeli task\_waiting.

Gdyby tak się zdarzyło że mamy w systemie produkt który nie jest zdefiniowany w systemie nadrzędnym to będzie to jedynie skutkowało brakiem odniesienia zlecenia do produktu.

## Pole ID\_OOP w tabelach task i task\_waiting

To bardzo ważne pole z punktu widzenia integracji – wpisujemy w nie ID zlecenia z nadrzędnego systemu które to ID zostanie umieszczone w tabeli task co pozwoli na proste odszukanie ID zlecenia systemu Golem na podstawie ID własnego.

Postępujemy według następującego schematu:

Mamy własny numer ID zlecenia, nazwijmy go MasterID.

Wpisując zlecenie do tabeli przypisujemy numer MasterID polu ID\_OOP.

Podczas tworzenia zlecenia ID z ID\_OOP tabeli task\_waiting przepisywany jest do tabeli do pola ID\_OOP w tabeli task.

Dzięki temu zapytanie :

```
select ID from task where ID_OOP = MasterID
```

zwróci numer ID zlecenia nadany przez system Golem który możemy użyć np. do agregacji danych z tabeli sv\_bdata :

```
select sum( D.cd0 )  
from sv_bdata D  
where D.id_task = pozyskany_numer_zlecenia
```

da nam w wyniku sumę produktu dla zlecenia.

## Tabele związane z produktem i zleceniem

task_product	kartoteka produktów
task	kartoteka zleceń produkcyjnych
taks_waiting	tabela ze zleceniami oczekującymi – na podstawie danych z tej tabeli system tworzy zlecenie w tabeli task
task_summary	tabela z podsumowaniem ZAKONCZONYCH zleceń – system po zakończeniu zleceń wylicza sumy, OEE itp. i zapisuje w tej tabeli

bezpośrednio ze zleceniami związane są też tabele:

tools	kartoteka narzędzi
toolsex	kartoteka narzędzi eksploatacyjnych
markets	kartoteka marketów

## Tabela TASK\_PRODUCT

tabela przechowuje kartotekę produktów

ID	INTEGER NOT NULL,	
M	SMALLINT,	pomocnicze pole do filtrowania
A	SMALLINT,	1 to produkt archiwalny
ID_G	INTEGER,	rezerwa
ID_EAN	VARCHAR(50),	ID jako kod kreskowy EAN13
ID_OPP	INTEGER,	zewnętrzny numer ID
ID_KP	INTEGER,	rezerwa
STATUS	SMALLINT,	status produktu
NAME	VARCHAR(100),	nazwa produktu
NAMEL	VARCHAR(200),	nazwa rozszerzona
SNOTE	VARCHAR(1000),	krótki opis / uwagi
TOOL_ID	INTEGER,	id z kartoteki narzędzi
TARGET	INTEGER,	ilość zamówiona (rezerwa)
OP	INTEGER,	operacje
KR	DOUBLE PRECISION,	krotność
OCC	DOUBLE PRECISION,	optymalny czas cyklu
OWYD	INTEGER,	rezerwa
PAKIET	DOUBLE PRECISION,	ilość w opakowaniu
GRAM	DOUBLE PRECISION,	gramatura
OPT_P	INTEGER,	optymalny czas przezbrajania
OPT_U	INTEGER,	optymalny czas ustawiania
OPT_KR	INTEGER,	optymalna krotność
DETAIL_NR	VARCHAR(100),	nazwa/symbol detalu
CODE_NR	VARCHAR(100),	nazwa/symbol programu
TECHNOLOGY	BLOB,	rezerwa
FC_ID	INTEGER,	rezerwa
KI_USE	SMALLINT,	1 to użyj karty informacyjnej
KI_DESC	BLOB,	karta info – opis 1
KI_DESC2	BLOB,	karta info – opis 2
KI_DESC_N	VARCHAR(200),	karta info – nazwa opisu 1
KI_DESC2_N	VARCHAR(200),	karta info – nazwa opisu 2
KI_F1	INTEGER,	karta info – id zdjęcia 1
KI_F2	INTEGER,	karta info – id zdjęcia 2
KI_F3	INTEGER,	karta info – id zdjęcia 3
KI_F4	INTEGER,	karta info – id zdjęcia 4
KI_FN1	VARCHAR(200),	karta info – nazwa zdjęcia 1
KI_FN2	VARCHAR(200),	karta info – nazwa zdjęcia 2
KI_FN3	VARCHAR(200),	karta info – nazwa zdjęcia 3
KI_FN4	VARCHAR(200),	karta info – nazwa zdjęcia 4
KI_FILE1	INTEGER,	id pdf1
KI_FILE2	INTEGER,	id pdf2
KI_FILE3	INTEGER,	id pdf3
KI_FILE4	INTEGER,	id pdf4

ALLSV	SMALLINT,	wszystkie maszyny – pole do użytku wewnętrznego
TEXP_LIST	INTEGER,	id listy narzędzi eksploatacyjnych
MARKET_LIST	INTEGER	id marketu

## Tabela TASK

Kartoteka zleceń produkcyjnych

ID	INTEGER NOT NULL,	
ID_EAN	VARCHAR(50),	// kod kreskowy EAN13 generowany z pola ID
ID_OPP	INTEGER,	// id dla systemu zewnętrznego
ID_KP	INTEGER,	// rezerwa
SV	SMALLINT,	// nr nadzorca
STATUS	SMALLINT,	// status: 0- w realizacji; 1- zakończone; 2- zawieszono
OP_ID	INTEGER,	// id operatora który zainicjował zlecenie
TBEGIN	TIMESTAMP,	// czas rozpoczęcia
TEND	TIMESTAMP,	// czas zakończenia
TPLAN	TIMESTAMP,	// rezerwa
NAME	VARCHAR(100),	// nazwa zlecenia
PRODUCT	VARCHAR(100),	// nazwa produktu
PRODUCT_M	SMALLINT,	// sterowanie powiązaniem z tabelą task_product
PRODUCT_ID	INTEGER,	// id kartoteki produktów
SNOTE	VARCHAR(1000),	// notatka o zleceniu / produkcie
TOOL_ID	INTEGER,	// id narzędzia
TARGET	INTEGER,	// ilość zamówiona
OP	INTEGER,	// ilość operacji
KR	DOUBLE PRECISION,	// krotność
OCC	DOUBLE PRECISION,	// optymalny czas cyklu
OWYD	INTEGER,	// rezerwa
PAKIET	DOUBLE PRECISION,	// ilość w opakowaniu
GRAM	DOUBLE PRECISION,	// waga wyrobu
OPT_P	INTEGER,	// optymalny czas przeobrażenia
OPT_U	INTEGER,	// optymalny czas ustawiania
OPT_KR	INTEGER,	// optymalna krotność
DETAIL_NR	VARCHAR(100),	// numer detalu
CODE_NR	VARCHAR(100),	// numer programu
TECHNOLOGY	BLOB SUB_TYPE 1 SEGMENT SIZE 80	// rezerwa
TEXP_LIST	INTEGER,	// id listy narzędzi eksploatacyjnych
MARKET_LIST	INTEGER,	// id listy marketów
KI_USE	SMALLINT	// dostępna karta informacyjna produktu
RATING	SMALLINT,	// subiektywna ocena
SUSPEND_C	INTEGER,	// licznik zawieszonych zleceń
SUSPEND_TIME	TIMESTAMP,	// czas ostatniego zawieszenia zlecenia
SUSPEND_MAP	BLOB SUB_TYPE 0 SEGMENT SIZE 80,	// binarny obraz liczników zawieszonych zleceń
RESTORE_TIME	TIMESTAMP,	// czas ostatniego odzwieszenia zlecenia
SUSPEND_T	INTEGER	// rezerwa

);

## Tabela TASK\_SUMMARY

tabela przechowuje podsumowanie zlecenia generowane po jego zakończeniu. Podsumowanie zawiera ilość wykonaną, braki, wskaźniki OEE dla zlecenia oraz wartości początkowe i końcowe parametrów zlecenia które mogą być zmieniane w trakcie jego realizacji.

ID	INTEGER NOT NULL,	
TASK_ID	INTEGER,	id zlecenia z tabeli TASK
PRODUKT	INTEGER,	ilość wyprodukowana
BRAKI	INTEGER,	ilość braków
ODPAD	INTEGER,	ilość odpadu
OEE	DOUBLE PRECISION,	OEE
OEE_D	DOUBLE PRECISION,	Dostępność
OEE_W	DOUBLE PRECISION,	Wykorzystanie
CCTIME	INTEGER,	całkowity czas w sekundach
CPTIME	INTEGER,	czas efektywnej pracy w sekundach
E_TARGET	INTEGER,	końcowa wartość zamówienia
E_KR	DOUBLE PRECISION,	końcowa krotność
E_OCC	DOUBLE PRECISION,	końcowy czas cyklu
E_GRAM	DOUBLE PRECISION,	końcowa gramatura
B_TARGET	INTEGER,	początkowa wartość zamówienia
B_KR	DOUBLE PRECISION,	początkowa krotność
B_OCC	DOUBLE PRECISION,	początkowy czas cyklu
B_GRAM	DOUBLE PRECISION,	początkowa gramatura

## Tabela TASK\_WAITING

Tabela przechowuje zlecenia oczekujące. To do tej tabeli zapisujemy zlecenia z zewnątrz – operator zobaczy ją na liście zleceń do wyboru.

ID	INTEGER NOT NULL,	
ID_OPP	INTEGER,	// id dla systemu zewnętrznego
ID_KP	INTEGER,	// rezerwa
SV	SMALLINT,	// numer nadzorcy
STATUS	SMALLINT,	// rezerwa
MODE	SMALLINT,	// wartość 0 – zlecenie oczekujące, wartość 1 dla operacji wewnętrznych
TORDER	SMALLINT,	// pole sortujące widok operatora
TUSED	SMALLINT,	// zlecenie użyte do realizacji
THIDE	SMALLINT,	// ukryj w widoku operatora
TPLAN	TIMESTAMP,	// rezerwa
NAME	VARCHAR(100),	// nazwa zlecenia
PRODUCT	VARCHAR(100),	// nazwa produktu
PRODUCT_M	SMALLINT,	
PRODUCT_ID	INTEGER,	// id produktu z kartoteki produktów
SNOTE	VARCHAR(1000),	// uwagi do realizacji
OP_ID	INTEGER,	// id operatora który zainicjował zlecenie ( tryb bezpośredni )
TOOL_ID, TARGET, OP, KR, OCC, OWYD, PAKIET, GRAM,		// parametry zlecenia
OPT_P, OPT_U, OPT_KR, DETAIL_NR, CODE_NR, KI_USE		// tożsame z tabelą TASK
TECHNOLOGY	BLOB SUB_TYPE 1 SEGMENT SIZE 80,	// rezerwa
TEXP_LIST	INTEGER,	// id listy narzędzi eksploatacyjnych
MARKET_LIST	INTEGER,	// id listy marketów
FC_ID	INTEGER	//rezerwa
KI_USE	SMALLINT,	//rezerwa
TEXP_LIST	INTEGER,	// id
MARKET_LIST	INTEGER	

## Pozostałe tabele

### Tabela TOOLS

Kartoteka narzędzi

ID	INTEGER NOT NULL,	
M	SMALLINT,	
FOTO1	INTEGER,	id 1 zdjęcia
FOTO2	INTEGER,	id 2 zdjęcia
ID_G	INTEGER,	id grupy (słownik)
ID_FW	INTEGER,	rezerwa
ID_BARCODE	VARCHAR(50),	kod kreskowy ean z numeru ID
STATE	SMALLINT,	status: 0-sprawne, 1-Ograniczone, 2-niedostępne, 3-zlikwidowane
AR	INTEGER,	rezerwa
SYMBOL	VARCHAR(100),	symbol
NAME	VARCHAR(200),	nazwa
BARCODE	VARCHAR(100),	kod ean
NOTE	BLOB SUB_TYPE 1 SEGMENT SIZE 80,	opis
KBASE	DOUBLE PRECISION,	krotność bazowa
KOPT	DOUBLE PRECISION,	krotność optymalna
KC	DOUBLE PRECISION,	krotność
OCC	DOUBLE PRECISION,	optymalny czas cyklu
OPT_P	INTEGER,	optymalny czas przezbrajania
OPT_U	INTEGER,	optymalny czas ustawiania
INSPECT	SMALLINT,	podlega inspekcji
OLD_INSPECT	TIMESTAMP,	czas ostatniej inspekcji
MAXC	INTEGER	maksymalna ilość cykli

## Tabela SV\_CRR

Tabela przechowuje informacje o aktualnym stanie maszyn. Tabela ma stałą ilość rekordów a ID rekordu jest jednocześnie numerem nadzorca

ID	INTEGER NOT NULL,	// ID=numer SV
SV_NAME	VARCHAR(100),	// nazwa nadzorca
STATUS	SMALLINT,	// numer statusu
STATUSR	SMALLINT,	// numer statusu rozszerzonego
STATUSR_ID	SMALLINT,	// ID statusu rozszerzonego
STATUS_CHANGE	TIMESTAMP,	// czas ostatniej zmiany statusu
UOP_ID	INTEGER,	// id operatora
UOP_CHANGE	TIMESTAMP,	// ostatnia zmiana operatora
ULD_ID	INTEGER,	// rezerwa
ULD_CHANGE	TIMESTAMP,	// rezerwa
UUR_ID	INTEGER,	// id pracownika utrzymania ruchu
UUR_CHANGE	TIMESTAMP,	// czas ostatniej zmiany pracownika utrzymania ruchu
USETUP_ID	INTEGER,	// id „ustawiacza”
USETUP_CHANGE	TIMESTAMP,	// czas ostatniej zmiany „ustawiacza”
TASK_ID	INTEGER,	// id zlecenia
TASK_P_ID	INTEGER,	// id produktu
TASK_ID_OPP	INTEGER,	// zewnętrzne ID
TASK_CHANGE	TIMESTAMP,	// czas zmiany zlecenia
TASK_NAME	VARCHAR(100),	// nazwa zlecenia
TASK_P_NAME	VARCHAR(100),	// nazwa produktu
TASK_P_NAME_R	VARCHAR(600),	// nazwa produktu rozszerzona
TASK_TOOL_ID	INTEGER,	// id narzędzia
TASK_TOOL_NAME_R	VARCHAR(400),	// nazwa narzędzia
TASK_TOOL_NAME	VARCHAR(100),	// symbol narzędzia
TASK_TARGET	INTEGER,	// ilość zamówiona
TASK_TARGET_START	INTEGER,	// początkowa ilość zamówiona
TASK_KR_START	DOUBLE PRECISION,	// początkowa krotność
TASK_KR	DOUBLE PRECISION,	// krotność
TASK_OP	INTEGER,	// ilość operacji
TASK_OCC	DOUBLE PRECISION,	// optymalny czas cyklu
TASK_OCC_START	DOUBLE PRECISION,	// początkowy, optymalny czas cyklu
TASK_OWYD	INTEGER,	// rezerwa
TASK_PAKIET	DOUBLE PRECISION,	// ilość w opakowaniu
TASK_GRAM	DOUBLE PRECISION,	// gramatura
TASK_GRAM_START	DOUBLE PRECISION,	// początkowa gramatura
TASK_OPT_P	INTEGER,	// optymalny czas przezbrajania
TASK_OPT_U	INTEGER,	// optymalny czas ustawiania
TASK_OPT_KR	INTEGER,	// optymalna krotność
TASK_DETAIL_NR	VARCHAR(100),	// parametr detal
TASK_CODE_NR	VARCHAR(100),	// parametr kod
TASK_TECHNOLOGY	BLOB SUB_TYPE 1 SEGMENT SIZE 80,	//rezerwa
TASK_FC_ID	INTEGER,	//
TASK_FC_OK	SMALLINT,	//
TASK_KI_USE	SMALLINT,	//
TASK_TEXP_LIST	INTEGER,	// id listy narzędzi eksploatacyjnych
TASK_MARKET_LIST	INTEGER,	// id marketu
TASK_TEXP_NAME	VARCHAR(400),	// nazwa grupy narzędzi eksploatacyjnych
TASK_MARKET_NAME	VARCHAR(400),	// nazwa marketu
TASK_RESUME_NO	INTEGER,	// ilość wznowień zlecenia
TASK_RESUME_DT	TIMESTAMP	// czas ostatniego wznowienia zlecenia

## Tabela SV\_COMMENTS

ID	INTEGER NOT NULL,	
SV	SMALLINT,	//numer nadzorca
NOTE	BLOB	//treść notatki

Tabela przechowuje notatki (komentarze) dla tabeli SV\_EV wskazywane polem ID\_CMT

## Tabela SV\_EV

Tabela przechowuje informacje o wszystkich ( poza załącz / wyłącz) zdarzeniach w systemie

ID	INTEGER NOT NULL,	
SV	SMALLINT,	numer nadzorca
ID_U	INTEGER,	id użytkownika ( np. osoby która zmieniła status)
ID_OP	INTEGER,	id operatora przypisanego do maszyny podczas zdarzenia
ID_TASK	INTEGER,	id zlecenia realizowanego przez maszynę podczas zdarzenia
CTIME	TIMESTAMP,	czas wystąpienia zdarzenia
TYR , TMC, TDY, THR, TSHIFT	SMALLINT,	nagłówek czasu wg zasad ogólnych
MODE	SMALLINT,	rodzaj zdarzenia
MODEG	SMALLINT,	grupa zdarzenia
P	SMALLINT,	1 określa że zdarzenie jest powiadomieniem

ID_CMT	INTEGER,	id komentarza z tabeli SV_COMMENTS jeśli ze zdarzeniem powiązany jest komentarz
OLD_ST	SMALLINT,	status przed zdarzeniem
NEW_ST	SMALLINT,	status po zdarzeniu
PAR1	INTEGER,	opcjonalne parametry
PAR2	INTEGER,	o wartości zależnej od kontekstu określanego polem MODE
MARK	SMALLINT,	rezerwa
MARK_T	TIMESTAMP,	rezerwa
MARK_CMT	INTEGER	rezerwa
MARK_UID	INTEGER	rezerwa

Kontekst określany wartością pola MODE:

0 – komentarz użytkownika  
1 – zmiana statusu

10 – zmiana operatora  
11 – //rezerwa  
12 – zmiana UR  
13 – zmiana ustawiacza  
14 – skasuj operatora  
15 – // rezerwa  
16 – skasuj UR  
17 – skasuj ustawiacza

20 – wznowienie pracy p1 – czas w sekundach  
21 – awaria trwa od xx czasu [ 0 ]  
22 – postój nieplanowany trwa do xx czasu [ 0 ]

30 – andon – odwołanie wezwania  
31 – andon – wezwanie pomocy  
32 – andon – wezwanie pomocy technicznej  
33 – andon – wezwanie pomocy logistycznej  
34 – andon – wezwanie pomocy U1  
35 – andon – wezwanie pomocy U2

Pole MODEG ( grupuje mode)

1 – komentarz produkcja  
2 – komentarz awarie  
3 – komentarz logistyka  
4 – komentarz jakość, dodanie braków przez operatora, korekta braków  
5 – komentarze technologia / produkt  
6 – status poza awarią i pracą po awarii  
7 – status awaria i praca po awarii  
8 – zmiana operatora i ustawiacza  
9 – zmiana pracownika UR  
10 – andon inny  
11 – andon techniczny, odwołanie andon techniczny  
12 – andon logistyczny, korekta stanu marketu  
13 – odwołanie wezwania  
14 – zmiana zlecenia  
15 – zmiana parametrów zlecenia, zmiana narzędzia  
16 – powiadomienia „wznowienie pracy”  
17 – powiadomienia status awaria > Tx  
18 – powiadomienia status postój nieplanowany > Tx  
19 – mikroraport  
20 – ręczne dodanie produktu  
21 – ręczna korekta produktu  
22 – inne komentarze

40 – nowe zlecenie  
41 – koniec zlecenia  
42- zmiana ilości zamówionej  
43- zmiana czasu cyklu  
44 – zmiana krotności  
45 – zmiana gramatury  
47- osiągnięto ilość zamówioną  
48 – do końca zlecenia X %  
49 – formalne potwierdzenie zlecenia  
50 – punkt kontrolny zlecenia  
51 - punkt kontrolny zlecenia – koniec zmiany X  
52 – zawieszenie zlecenia  
53 – odwieszenie zlecenia  
54 – instalacja narzędzia ( zmiana narzędzia )  
55 – deinstalacja narzędzia  
59 – pseudo zlecenie  
  
60 – market alarm  
61 – market - operacja  
62 – utylizacja narzędzia Ex  
70 - braki  
71 – braki – korekta  
75- produkt ręcznie  
76- ręczna korekta produktu

klasyfikacja funkcjonalna powiadomień ( wartości pola modeG)

produkcja:	1,6,7,8,10,14,16,17,18,19,21,22
ur:	2,7,9,11,17
produkt:	5,14,15,19,21,22
braki:	4
logistyka:	3,12
inne: ??	

## Tabela SV\_BDATA

Tabela przechowuje dane procesowe zapisywane co godzinę oraz po każdej operacji zmiany statusu, zlecenia, operatora etc.

ID	INTEGER NOT NULL,	
SV	SMALLINT, // numer nadzorca	
MODE	SMALLINT,	
TBEGIN	TIMESTAMP,	
TEND	TIMESTAMP,	
TYR, TMC, TDY, THR, TSHIFT	SMALLINT,	
ID_TASK	INTEGER,	
ID_TASKP	INTEGER,	
ID_TOOL	INTEGER,	
ID_UOP	INTEGER,	
ID_UUR	INTEGER,	
ID_ULD	INTEGER,	
ID_SET	INTEGER,	
SNO	SMALLINT,	// numer statusu
SRNO	SMALLINT,	// numer status rozszerzonego
SRID	SMALLINT,	// id status rozszerzonego
SCOUNT	SMALLINT,	// rezerwa
STIME	SMALLINT,	
S0	SMALLINT,	//postój planowany
S1	SMALLINT,	//konserwacja
S2	SMALLINT,	//przezbajanie
S3	SMALLINT,	//ustawianie
S4	SMALLINT,	//postój nieplanowany
S5	SMALLINT,	//awaria
S6	SMALLINT,	//efektywny czas pracy
S7	SMALLINT,	//mikropostoje
S8	SMALLINT,	//czas nieoznaczony
S9	SMALLINT,	//przerwa techniczna 1
S10	SMALLINT,	//przerwa techniczna 2
S11	SMALLINT,	//przezbajanie ponadnormatywne
S12	SMALLINT,	//ustawianie ponadnormatywne
CS0	SMALLINT,	//postój planowany count
CS1	SMALLINT,	//konserwacja count
CS2	SMALLINT,	//przezbajanie count
CS3	SMALLINT,	//ustawianie count
CS4	SMALLINT,	//postój nieplanowany count
CS5	SMALLINT,	//awaria count
CI0	INTEGER,	//cykle absolutne
CI1	INTEGER,	//operacje
CI2	INTEGER,	//cykle
CI3	INTEGER,	//takt
CI4	INTEGER,	//andon1 time
CI5	INTEGER,	//andon2 time
CI6	INTEGER,	//andon3 time
CI7	INTEGER,	//andon4 time
CI8	INTEGER,	//andon5 time
CI9	INTEGER,	// motogodziny
CI10	INTEGER,	//rezerwa
CI11	INTEGER,	//czas naprawy
CI12	INTEGER,	//czas przezbajania
CW0	SMALLINT,	//andon1 cykle
CW1	SMALLINT,	//andon2 cykle
CW2	SMALLINT,	//andon3 cykle
CW3	SMALLINT,	//andon4 cykle
CW4	SMALLINT,	//andon5 cykle
CD0	DOUBLE PRECISION,	//produkt
CD1	DOUBLE PRECISION,	//produkt abs
CD2	DOUBLE PRECISION,	//kwh
CD3	DOUBLE PRECISION,	//materiał1
CD4	DOUBLE PRECISION,	//licznik CAVITY
CDS0	DOUBLE PRECISION,	//braki
CDS1	DOUBLE PRECISION,	//odpad
CDS2	DOUBLE PRECISION,	//pakiety
CDS3	DOUBLE PRECISION,	//rezerwa
CDS4	DOUBLE PRECISION,	//kwh - postój planowany
CDS5	DOUBLE PRECISION,	//kwh – postój nieplanowany
CDS6	DOUBLE PRECISION,	//kwh – przezbajanie i ustawianie
CDS7	DOUBLE PRECISION,	//kwh – awarie