APPENDIX II

This appendix supports chapter 4 and presents a database script associated with some prototype modules. The database script presented in this appendix is responsible for the:

- creation of the Risk Profiles Repository;
- creation of the Real Online Transactions Repository;
- instantiation of both databases to allow the deployment of the prototype in the implementation of the case study;
- triggers implemented in the operational information system of the case study. Just a few triggers have been chosen to be presented herein. The presented triggers are sufficient to understand the structure and reasoning of the implementation of all other triggers.

Creation of the Risk Profiles Repository

```
   if not exists(select * from sys.databases where name = 'rpr_db')
       create database rpr_db;
   go
   use rpr_db;
   go

   /** Drop Tables   **/
   if exists (select * from INFORMATION_SCHEMA.TABLES where TABLE_NAME = 'Tev_TriggerEvent')
       drop table Tev_TriggerEvent;
   if   exists   (select   *   from   INFORMATION_SCHEMA.TABLES   where   TABLE_NAME   =
'Tpi_TriggerParentInfo')
       drop table Tpi_TriggerParentInfo;
   if   exists   (select   *   from   INFORMATION_SCHEMA.TABLES   where   TABLE_NAME   =
'Toi_TriggerOccurrenceInfo')
       drop table Toi_TriggerOccurrenceInfo;
   if   exists   (select   *   from   INFORMATION_SCHEMA.TABLES   where   TABLE_NAME   =
'Tgo_TriggerOccurrence')
       drop table Tgo_TriggerOccurrence;
   if exists (select * from INFORMATION_SCHEMA.TABLES where TABLE_NAME = 'Tgr_Trigger')
       drop table Tgr_Trigger;
   if exists (select * from INFORMATION_SCHEMA.TABLES where TABLE_NAME = 'Ttp_TriggerType')
       drop table Ttp_TriggerType;
   if exists (select * from INFORMATION_SCHEMA.TABLES where TABLE_NAME = 'Usr_User')
       drop table Usr_User;
   if exists (select * from INFORMATION_SCHEMA.TABLES where TABLE_NAME = 'Evp_EventParent')
       drop table Evp_EventParent;
   if   exists   (select   *   from   INFORMATION_SCHEMA.TABLES   where   TABLE_NAME   =
'Eur_EventUserRole')
       drop table Eur_EventUserRole;
   if   exists   (select   *   from   INFORMATION_SCHEMA.TABLES   where   TABLE_NAME   =
'Evt_TransactionEvent')
       drop table Evt_TransactionEvent;
   if   exists   (select   *   from   INFORMATION_SCHEMA.TABLES   where   TABLE_NAME   =
'Tpf_TransactionProfile')
       drop table Tpf_TransactionProfile;
   if exists (select * from INFORMATION_SCHEMA.TABLES where TABLE_NAME = 'Tst_Transaction')
       drop table Tst_Transaction;
   if exists (select * from INFORMATION_SCHEMA.TABLES where TABLE_NAME = 'Ptp_ProfileType')
       drop table Ptp_ProfileType;
   if exists (select * from INFORMATION_SCHEMA.TABLES where TABLE_NAME = 'Etp_EventType')
       drop table Etp_EventType ;
   if exists (select * from INFORMATION_SCHEMA.TABLES where TABLE_NAME = 'Url_UserRole')
       drop table Url_UserRole;


   /** Create Tables **/
```

```
/*** Risk Profile Repository ***/
CREATE TABLE Url_UserRole
(
      Url_id                    INT                         NOT NULL,
     Url_description            NVARCHAR(100)               NOT NULL,
     CONSTRAINT PK_UserRole     PRIMARY KEY(Url_id)
);

CREATE TABLE Etp_EventType
(
     Etp_id                     INT                 NOT NULL,
     Etp_description            NVARCHAR(100)       NOT NULL,
     CONSTRAINT PK_EventType    PRIMARY KEY(Etp_id)
);

CREATE TABLE Ptp_ProfileType
(
     Ptp_id                     INT                         NOT NULL  ,
     Ptp_description            NVARCHAR(100)               NOT NULL,
     CONSTRAINT PK_ProfileType  PRIMARY KEY(Ptp_id)
);

CREATE TABLE Tst_Transaction
(
     Tst_id                     INT                 NOT NULL       IDENTITY,
     Tst_description            NVARCHAR(100)       NOT NULL,
     CONSTRAINT PK_Transaction  PRIMARY KEY(Tst_id)
);

CREATE TABLE Tpf_TransactionProfile
(
     Tpf_id                              INT            NOT NULL       IDENTITY,
     Tpf_description                     NVARCHAR(100)  NOT NULL,
     TpfTst_id                           INT            NOT NULL,
     TpfPtp_id                           INT            NOT NULL,
     CONSTRAINT PK_ProfileTransaction    PRIMARY KEY(Tpf_id),
     CONSTRAINT FK_Tpf_Transaction       FOREIGN KEY(TpfTst_id)
                                         REFERENCES     Tst_Transaction,
     CONSTRAINT FK_Tpf_ProfileType       FOREIGN KEY(TpfPtp_id)
                                         REFERENCES     Ptp_ProfileType
);

CREATE TABLE Evt_TransactionEvent
(
     Evt_id                              INT            NOT NULL       IDENTITY,
     Evt_description                     NVARCHAR(100),
     EvtTpf_id                           INT            NOT NULL,
```

```
    EvtEtp_id                           INT            NOT NULL,
    CONSTRAINT PK_TransactionEvent      PRIMARY KEY(Evt_id),
    CONSTRAINT FK_Evt_ProfileTransaction FOREIGN KEY(EvtTpf_id)
                                        REFERENCES     Tpf_TransactionProfile,
    CONSTRAINT FK_Evt_EventType         FOREIGN KEY(EvtEtp_id)
                                        REFERENCES     Etp_EventType
);

CREATE TABLE Eur_EventUserRole
(
    Eur_id                      INT            NOT NULL       IDENTITY,
    EurEvt_id                           INT    NOT NULL,
    EurUrl_id                           INT    NOT NULL,
    CONSTRAINT PK_EventUserRole         PRIMARY KEY(Eur_id),
    CONSTRAINT FK_Eur_TransactionEvent  FOREIGN KEY(EurEvt_id)
                                        REFERENCES     Evt_TransactionEvent,
    CONSTRAINT FK_Eur_UserRole          FOREIGN KEY(EurUrl_id)
                                        REFERENCES     Url_UserRole
);

CREATE TABLE Evp_EventParent
(
    Evp_id                      INT            NOT NULL       IDENTITY,
    EvpEvt_idFather             INT            NOT NULL,
    EvpEvt_idSon                INT            NOT NULL,
    Evp_RequiredFather          SMALLINT       NULL,
    Evp_RequiredSon              SMALLINT      NULL,
    Evp_EventParent             INT,
    CONSTRAINT PK_EventParent   PRIMARY KEY(Evp_id),
    CONSTRAINT FK_Evp_Father    FOREIGN KEY(EvpEvt_idFather)
                                REFERENCES     Evt_TransactionEvent,
    CONSTRAINT FK_Evp_Son       FOREIGN KEY(EvpEvt_idSon)
                                REFERENCES     Evt_TransactionEvent
);
```

Creation of the Real Online Transactions Repository

```
CREATE TABLE Usr_User
(
    Usr_id                      INT                         NOT NULL,
    Usr_description             NVARCHAR(100)               NOT NULL,
    UsrUrl_id                   INT                         NOT NULL,
    CONSTRAINT PK_User          PRIMARY KEY(Usr_id),
    CONSTRAINT FK_Usr_UserRole  FOREIGN KEY(UsrUrl_id) REFERENCES     Url_UserRole
);
```

```
CREATE TABLE Ttp_TriggerType
(
    Ttp_id                      INT                         NOT NULL,
    Ttp_description             NVARCHAR(100)               NOT NULL,
    CONSTRAINT PK_TriggerType   PRIMARY KEY(Ttp_id)

);

CREATE TABLE Tgr_Trigger
(
    Tgr_id                              INT                         NOT NULL,
    Tgr_description                     NVARCHAR(100)               NOT NULL,
    TgrTtp_id                           INT                         NOT NULL,
    CONSTRAINT PK_Trigger               PRIMARY KEY(Tgr_id),
    CONSTRAINT FK_Tgr_TriggerType FOREIGN KEY(TgrTtp_id) REFERENCES     Ttp_TriggerType
);

CREATE TABLE Tgo_TriggerOccurrence
(
    Tgo_id                              INT             NOT NULL            IDENTITY,
    TgoTgr_id                           INT             NOT NULL,
    Tgo_dateTime                        DATETIME        NOT NULL,
    Tgo_dbTable                         NVARCHAR(100)   NOT NULL,
    TgoUsr_id                           INT             NOT NULL,
    CONSTRAINT PK_TriggerOcurrence      PRIMARY KEY(Tgo_id),
    CONSTRAINT FK_Tgo_Trigger           FOREIGN KEY(TgoTgr_id)
                                        REFERENCES      Tgr_Trigger,
    CONSTRAINT FK_Tgo_User              FOREIGN KEY(TgoUsr_id)
                                        REFERENCES      Usr_User
);

CREATE TABLE Toi_TriggerOccurrenceInfo
(
    Toi_id                                  INT                     NOT NULL
    IDENTITY,
    ToiTgo_id                               INT                     NOT NULL,
    Toi_attribute                           NVARCHAR(100)           NOT NULL,
    Toi_value                               NVARCHAR(500)           NOT NULL,
    Toi_description                         NVARCHAR(500)           NOT NULL,
    CONSTRAINT PK_TriggerOcurrenceInfo      PRIMARY KEY(Toi_id),
    CONSTRAINT FK_TriggerOcurrence_Info     FOREIGN KEY(ToiTgo_id)
                                            REFERENCES      Tgo_TriggerOccurrence
);



CREATE TABLE Tpi_TriggerParentInfo
```

```
(
    Tpi_id                              INT            NOT NULL              IDENTITY,
    TpiTgo_id                           INT            NOT NULL,
    Tpi_attribute                       NVARCHAR(100)  NOT NULL,
    Tpi_value                           NVARCHAR(500)          NOT NULL,
    Tpi_description                     NVARCHAR(500)          NOT NULL,
    CONSTRAINT PK_TriggerParentInfo     PRIMARY KEY(Tpi_id),
    CONSTRAINT FK_Trigger_ParentInfo    FOREIGN KEY(TpiTgo_id)
                                        REFERENCES     Tgo_TriggerOccurrence
);

CREATE TABLE Atg_AlterTrigger
(
    Atg_id                                      INT            NOT NULL       IDENTITY,
    Atg_attributeName                           NVARCHAR(100)  NOT NULL,
    Atg_oldValue                                NVARCHAR(500)  NOT NULL,
    Atg_newValue                                NVARCHAR(500)  NOT NULL,
    AtgTgo_id                                   INT            NOT NULL,
    CONSTRAINT PK_AlterTrigger                  PRIMARY KEY(Atg_id),
    CONSTRAINT FK_Atg_TriggerOccurrence         FOREIGN KEY(AtgTgo_id)
                                                REFERENCES     Tgo_TriggerOccurrence
);

CREATE TABLE Tev_TriggerEvent
(
    Tev_id                          INT                   NOT NULL              IDENTITY,
    TevTgr_id                       INT                   NOT NULL,
    TevEvt_id                       INT                   NOT NULL,
    CONSTRAINT PK_TriggerEvent      PRIMARY KEY(Tev_id),
    CONSTRAINT FK_Tev_Trigger       FOREIGN KEY(TevTgr_id) REFERENCES     Tgr_Trigger,
    CONSTRAINT FK_Tev_Event         FOREIGN KEY(TevEvt_id)
                                    REFERENCES     Evt_TransactionEvent
);
```

Instantiation of Databases

```
INSERT INTO  URL_USERROLE     (URL_ID, URL_DESCRIPTION)                          VALUES
('1','admin');
INSERT INTO  URL_USERROLE     (URL_ID, URL_DESCRIPTION)                          VALUES
('9999','desconhecido');

INSERT INTO  USR_USER         (USR_ID,USR_DESCRIPTION, USRURL_ID)   VALUES
('1','User 1','1');
INSERT INTO  USR_USER         (USR_ID,USR_DESCRIPTION, USRURL_ID)   VALUES
('9999','desconhecido','9999');
```

```
INSERT INTO  TTP_TRIGGERTYPE (TTP_ID, TTP_DESCRIPTION)                          VALUES
('1','insert');
INSERT INTO  TTP_TRIGGERTYPE (TTP_ID, TTP_DESCRIPTION)                          VALUES
('2','update');
INSERT INTO  TTP_TRIGGERTYPE (TTP_ID, TTP_DESCRIPTION)                          VALUES
('3','delete');

INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES     ('1','inserir
encomenda','1');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES     ('2','inserir
guia de remessa','1');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES     ('3','inserir
guia de transporte','1');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES     ('4','inserir
orçamento','1');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES     ('5','inserir
fatura pro-forma','1');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES     ('6','inserir
fatura','1');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES     ('7','inserir
venda a dinheiro','1');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES     ('8','inserir
nota de credito','1');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES     ('9','inserir
nota de debito','1');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('10','eliminação
de encomenda','3');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('11','eliminação
de guia de remessa','3');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('12','eliminação
de guia de transporte','3');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('13','eliminação
de orçamento','3');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('14','eliminação
de fatura pro-forma','3');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('15','eliminação
de fatura','3');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('16','eliminação
de venda a dinheiro','3');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('17','eliminação
de nota de credito','3');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('18','eliminação
de nota de debito','3');

INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('19','eliminação
de linhas de encomenda','3');
```

```
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('20','eliminação
de linhas de guia de remessa','3');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('21','eliminação
de linhas de guia de transporte','3');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('22','eliminação
de linhas de orçamento','3');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('23','eliminação
de linhas de  fatura pro-forma','3');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('24','eliminação
de linhas de fatura','3');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('25','eliminação
de linhas de venda a dinheiro','3');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('26','eliminação
de linhas de nota de credito','3');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('27','eliminação
de linhas de nota de debito','3');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('28','eliminação
de linhas descritivas de encomenda','3');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('29','eliminação
de linhas descritivas de guia de remessa','3');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('30','eliminação
de linhas descritivas de guia de transporte','3');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('31','eliminação
de linhas descritivas de orçamento','3');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('32','eliminação
de linhas descritivas de fatura pro-forma','3');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('33','eliminação
de linhas descritivas de fatura','3');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('34','eliminação
de linhas descritivas de venda a dinheiro','3');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('35','eliminação
de linhas descritivas de nota de credito','1');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('36','eliminação
de linhas descritivas de nota de debito','3');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES  ('37','alteração
de encomenda - cabeçalho','2');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES  ('38','alteração
de guia de remessa - cabeçalho ','2');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('39', 'alteração
de guia de transporte - cabeçalho ','2');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('40', 'alteração
de orçamento- cabeçalho ','2');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('41', 'alteração
de fatura pro-forma - cabeçalho ','2');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('42', 'alteração
de fatura - cabeçalho ','2');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('43', 'alteração
de venda a dinheiro - cabeçalho ','2');
```

```
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('44', 'alteração
de nota de credito - cabeçalho ','2');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('45', 'alteração
de nota de debito - cabeçalho ','2');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES  ('46','alteração
de encomenda – linhas de artigo','2');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES  ('47','alteração
de guia de remessa - linhas de artigo','2');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('48', 'alteração
de guia de transporte - linhas de artigo','2');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('49', 'alteração
de orçamento - linhas de artigo ','2');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('50', 'alteração
de fatura pro-forma - linhas de artigo ','2');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('51', 'alteração
de fatura - linhas de artigo ','2');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('52', 'alteração
de venda a dinheiro - linhas de artigo ','2');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('53', 'alteração
de nota de credito - linhas de artigo ','2');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('54', 'alteração
de nota de debito - linhas de artigo ','2');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES  ('55','alteração
de encomenda – linhas descritivas','2');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES  ('56','alteração
de guia de remessa - linhas descritivas ','2');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('57', 'alteração
de guia de transporte - linhas descritivas ','2');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('58', 'alteração
de orçamento - linhas descritivas','2');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('59', 'alteração
de fatura pro-forma - linhas descritivas ','2');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('60', 'alteração
de fatura - linhas descritivas','2');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('61', 'alteração
de venda a dinheiro - linhas descritivas','2');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('62', 'alteração
de nota de credito - linhas descritivas','2');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('63', 'alteração
de nota de debito - linhas descritivas','2');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES    ('64','inserir
recibo','1');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES  ('65','alteração
de recibo - cabeçalho','2');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('66','eliminação
de recibo','3');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES  ('67','alteração
de recibo - linha','2');
```

```
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('68','eliminação
de recibo - linha','3');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES   ('69','inserção
de movimento contabilistico','1');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES  ('70','alteração
de movimento contabilistico - cabeçalho','2');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('71','eliminação
de movimento contabilistico - cabeçalho','3');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES  ('72','alteração
de movimento contabilistico - linha','2');
INSERT INTO  TGR_TRIGGER      (TGR_ID, TGR_DESCRIPTION, TGRTTP_ID)  VALUES ('73','eliminação
de movimento contabilistico - linha','3');
Go


INSERT INTO  ptp_profiletype (ptp_id, ptp_description) VALUES (0,'negative');
INSERT INTO  ptp_profiletype (ptp_id, ptp_description) VALUES (1,'positive');

INSERT INTO  etp_eventtype (etp_id, etp_description) VALUES (1,'request');
INSERT INTO  etp_eventtype (etp_id, etp_description) VALUES (2,'promise');
INSERT INTO  etp_eventtype (etp_id, etp_description) VALUES (3,'execute');
INSERT INTO  etp_eventtype (etp_id, etp_description) VALUES (4,'state');
INSERT INTO  etp_eventtype (etp_id, etp_description) VALUES (5,'accept');
INSERT INTO  etp_eventtype (etp_id, etp_description) VALUES (6,'decline');
INSERT INTO  etp_eventtype (etp_id, etp_description) VALUES (7,'quit');
INSERT INTO  etp_eventtype (etp_id, etp_description) VALUES (8,'reject');
INSERT INTO  etp_eventtype (etp_id, etp_description) VALUES (9,'stop');
INSERT INTO  tst_transaction (tst_description) VALUES ('Comercial');

INSERT  INTO    tpf_transactionprofile  (tpf_description,  tpftst_id,  tpfptp_id)  VALUES
('Comercial – FAT + MOV', 1, 1);
INSERT  INTO    tpf_transactionprofile  (tpf_description,  tpftst_id,  tpfptp_id)  VALUES
('Comercial - ORC + ENC + GRM + FAT + MOV', 1, 1);
INSERT  INTO    tpf_transactionprofile  (tpf_description,  tpftst_id,  tpfptp_id)  VALUES
('Comercial - ORC + ENC + FAT + MOV', 1, 1);
INSERT  INTO    tpf_transactionprofile  (tpf_description,  tpftst_id,  tpfptp_id)  VALUES
('Comercial - ENC + FAT + MOV', 1, 1);
INSERT  INTO    tpf_transactionprofile  (tpf_description,  tpftst_id,  tpfptp_id)  VALUES
('Comercial - ENC + GRM + FAT + MOV', 1, 1);
INSERT  INTO    tpf_transactionprofile  (tpf_description,  tpftst_id,  tpfptp_id)  VALUES
('Comercial – FAT + RE + MOV', 1, 1);
INSERT  INTO    tpf_transactionprofile  (tpf_description,  tpftst_id,  tpfptp_id)  VALUES
('Comercial - ORC + ENC + GRM + FAT + RE + MOV', 1, 1);
INSERT  INTO    tpf_transactionprofile  (tpf_description,  tpftst_id,  tpfptp_id)  VALUES
('Comercial - ORC + ENC + FAT + RE+ MOV', 1, 1);
INSERT  INTO    tpf_transactionprofile  (tpf_description,  tpftst_id,  tpfptp_id)  VALUES
('Comercial - ENC + FAT + RE + MOV', 1, 1);
```

```
INSERT INTO tpf_transactionprofile (tpf_description, tpftst_id, tpfptp_id) VALUES
('Comercial - ENC + GRM + FAT + RE + MOV', 1, 1);
INSERT INTO tpf_transactionprofile (tpf_description, tpftst_id, tpfptp_id) VALUES
('Comercial – FAT + RE + MOV + AltFAT', 1, 0);
INSERT INTO tpf_transactionprofile (tpf_description, tpftst_id, tpfptp_id) VALUES
('Comercial – ENC + FAT + RE + MOV + AltFAT', 1, 0);
INSERT INTO tpf_transactionprofile (tpf_description, tpftst_id, tpfptp_id) VALUES
('Comercial – FAT + AltFAT', 1, 0);
INSERT INTO tpf_transactionprofile (tpf_description, tpftst_id, tpfptp_id) VALUES
('Comercial – FAT + RE + MOV + ElimFAT', 1, 0);
INSERT INTO tpf_transactionprofile (tpf_description, tpftst_id, tpfptp_id) VALUES
('Comercial – ENC + FAT + RE + MOV + ElimFAT', 1, 0);
INSERT INTO tpf_transactionprofile (tpf_description, tpftst_id, tpfptp_id) VALUES
('Comercial – FAT + ElimFAT', 1, 0);
INSERT INTO tpf_transactionprofile (tpf_description, tpftst_id, tpfptp_id) VALUES
('Comercial – FAT + AltLinhasFAT', 1, 0);
INSERT INTO tpf_transactionprofile (tpf_description, tpftst_id, tpfptp_id) VALUES
('Comercial – FAT + RE + MOV + AltLinhasFAT', 1, 0);
INSERT INTO tpf_transactionprofile (tpf_description, tpftst_id, tpfptp_id) VALUES
('Comercial – ENC + FAT + RE + MOV + ElimLinhasFAT', 1, 0);
INSERT INTO tpf_transactionprofile (tpf_description, tpftst_id, tpfptp_id) VALUES
('Comercial – FAT + ElimLinhasFAT', 1, 0);


INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da fatura', 1, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES ('Registo
Contabilistico', 1, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação do orçamento', 2, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da encomenda', 2, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da guia de remessa', 2, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da fatura', 2, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES ('Registo
Contabilistico', 2, 3);

INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação do orçamento', 3, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da encomenda', 3, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da fatura', 3, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES ('Registo
Contabilistico', 3, 3);
```

```
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da encomenda', 4, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da fatura', 4, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES ('Registo
Contabilistico', 4, 3);

INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da encomenda', 5, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da guia de remessa', 5, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da fatura', 5, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES ('Registo
Contabilistico', 5, 3);

INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da fatura', 6, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação do recibo', 6, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES ('Registo
Contabilistico', 6, 3);

INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação do orçamento', 7, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da encomenda', 7, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da guia de remessa', 7, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da fatura', 7, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação do recibo', 7, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES ('Registo
Contabilistico', 7, 3);

INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação do orçamento', 8, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da encomenda', 8, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da fatura', 8, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES ('Registo
Contabilistico', 8, 3);

INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da encomenda', 9, 3);
```

```
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da fatura', 9, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES ('Registo
Contabilistico', 9, 3);

INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da encomenda', 10, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da guia de remessa', 10, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da fatura', 10, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES ('Registo
Contabilistico', 10, 3);

INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da fatura', 11, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação do recibo', 11, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES ('Registo
Contabilistico', 11, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Alteração cabeçalho da fatura', 11, 3);

INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da encomenda', 12, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da fatura', 12, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação do recibo', 12, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES ('Registo
Contabilistico', 12, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Alteração cabeçalho da fatura', 12, 3);

INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da fatura', 13, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Alteração cabeçalho da fatura', 13, 3);

INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da fatura', 14, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação do recibo', 14, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES ('Registo
Contabilistico', 14, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Eliminação da fatura', 14, 3);
```

```
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da encomenda', 15, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da fatura', 15, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação do recibo', 15, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES ('Registo
Contabilistico', 15, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Eliminação da fatura', 15, 3);

INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da fatura', 16, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Eliminação da fatura', 16, 3);

INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da fatura', 17, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Alteração de linhas da fatura', 17, 3);

INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da fatura', 18, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação do recibo', 18, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES ('Registo
Contabilistico', 18, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Alteração de linhas da fatura', 18, 3);

INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da encomenda', 19, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da fatura', 19, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação do recibo', 19, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES ('Registo
Contabilistico', 19, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Eliminação de linhas de fatura', 19, 3);

INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação da fatura', 20, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Eliminação de linhas da fatura', 20, 3);

INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação do recibo', 8, 3);
```

```
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação do recibo', 9, 3);
INSERT INTO evt_transactionevent (evt_description, evttpf_id, evtetp_id) VALUES
('Apresentação do recibo', 10, 3);
go

INSERT INTO evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (1, 2, 10);
INSERT INTO evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (3, 4, 1);
INSERT INTO evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (4, 5, 1);
INSERT INTO evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (5, 6, 1);
INSERT INTO evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (6, 7, 10);
INSERT INTO evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (8, 9, 1);
INSERT INTO evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (9, 10, 1);
INSERT INTO evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (10, 11, 10);
INSERT INTO evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (12, 13, 1);
INSERT INTO evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (13, 14, 10);
INSERT INTO evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (15, 16, 1);
INSERT INTO evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (16, 17, 1);
INSERT INTO evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (17, 18, 10);
INSERT INTO evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (19, 20, 2);
INSERT INTO evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (20, 21, 10);
INSERT INTO evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (22, 23, 1);
INSERT INTO evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (23, 24, 1);
INSERT INTO evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (24, 25, 1);
INSERT INTO evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (25, 26, 2);
INSERT INTO evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (26, 27, 10);
INSERT INTO evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (28, 29, 1);
```

```
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (29, 30, 1);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (30, 74, 2);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (74, 31, 10);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (32, 33, 1);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (33, 75, 2);

INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (75, 34, 10);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (35, 36, 1);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (36, 37, 1);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (37, 76, 2);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (76, 38, 10);

INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (39, 40, 2);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (40, 41, 10);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson) VALUES (41, 42);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (43, 44, 1);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (44, 45, 2);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (45, 46, 10);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson) VALUES (46, 47);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson) VALUES (48, 49);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (50, 51, 2);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (51, 52, 10);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson) VALUES (52, 53);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (54, 55, 1);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (55, 56, 2);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (56, 57, 10);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson) VALUES (57, 58);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson) VALUES (59, 60);
```

```
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson) VALUES (61, 62);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (63, 64, 2);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (64, 65, 10);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson) VALUES (65, 66);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (67, 68, 1);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (68, 69, 2);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson, evp_timeinterval)
VALUES (69, 70, 10);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson) VALUES (70, 71);
INSERT INTO  evp_eventparent (evpevt_idfather, evpevt_idson) VALUES (72, 73);


INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (1, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (2, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (3, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (4, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (5, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (6, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (7, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (8, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (9, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (10, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (11, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (12, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (13, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (14, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (15, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (16, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (17, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (18, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (19, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (20, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (21, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (22, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (23, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (24, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (25, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (26, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (27, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (28, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (29, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (30, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (31, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (32, 1);
```

```
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (33, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (34, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (35, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (36, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (37, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (38, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (39, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (40, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (41, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (42, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (43, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (44, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (45, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (46, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (47, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (48, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (49, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (50, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (51, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (52, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (53, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (54, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (55, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (56, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (57, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (58, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (59, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (60, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (61, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (62, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (63, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (64, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (65, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (66, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (67, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (68, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (69, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (70, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (71, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (72, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (73, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (74, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (75, 1);
INSERT INTO  eur_eventuserrole (eurevt_id, eururl_id) VALUES (76, 1);

INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (6, 1);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (69, 2);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (4, 3);
```

```
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (1, 4);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (2, 5);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (6, 6);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (69, 7);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (4, 8);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (1, 9);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (6, 10);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (69, 11);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (1, 12);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (6, 13);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (69, 14);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (1, 15);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (2, 16);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (6, 17);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (69, 18);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (6, 19);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (64, 20);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (69, 21);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (4, 22);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (1, 23);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (2, 24);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (6, 25);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (64, 26);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (69, 27);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (4, 28);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (1, 29);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (6, 30);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (69, 31);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (1, 32);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (6, 33);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (69, 34);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (1, 35);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (2, 36);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (6, 37);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (69, 38);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (6, 39);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (64, 40);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (69, 41);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (42, 42);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (1, 43);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (6, 44);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (64, 45);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (69, 46);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (42, 47);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (6, 48);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (42, 49);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (6, 50);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (64, 51);
```

```
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (69, 52);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (15, 53);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (1, 54);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (6, 55);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (64, 56);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (69, 57);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (15, 58);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (6, 59);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (15, 60);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (6, 61);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (51, 62);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (6, 63);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (64, 64);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (69, 65);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (51, 66);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (1, 67);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (6, 68);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (64, 69);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (69, 70);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (24, 71);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (6, 72);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (24, 73);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (64, 74);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (64, 75);
INSERT INTO  tev_triggerevent (tevtgr_id, tevevt_id) VALUES (64, 76);
```

Triggers

```
-- =============================================
-- Author:          Rui Marques
-- Create date:    13th March, 2013
-- Description:     Trigger for documents insertion
-- =============================================
CREATE TRIGGER [dbo].[TGR_INSERT_DOCUMENTS]
   ON  [dbo].[CabecDoc]
   AFTER INSERT
AS
BEGIN
     -- SET NOCOUNT ON added to prevent extra result sets from
     -- interfering with SELECT statements.
     SET NOCOUNT ON;

     DECLARE
               @id    int,
               @td    nvarchar(5),
               @s     nvarchar(2),
               @nr    int,
               @dt    date,
               @des   nvarchar(50),
               @tgr   int,
               @usr   nvarchar(50),
               @usrId int

     SET     @td = (SELECT TipoDoc FROM inserted)
     SET     @s  = (SELECT Serie FROM inserted)
     SET     @nr = (SELECT NumDoc FROM inserted)
     SET     @dt = (SELECT Data FROM inserted)
     SET     @usr = (SELECT Utilizador FROM inserted)
     SET     @usrId =(SELECT usr_id from rpr_db.dbo.usr_user where usr_description = @usr)
     IF  @usrId is null
            SET @usrId = 9999

     SET @des = CASE
                                  WHEN @td='ECL' then 'Encomenda a cliente'
                                  WHEN @td='FA' then 'Fatura'
                                  WHEN @td='FP' then 'Fatura pró-forma'
                                  WHEN @td='GR' then 'Guia de remessa'
                                  WHEN @td='GT' then 'Guia de transporte'
                                  WHEN @td='NC' then 'Nota de crédito'
                                  WHEN @td='NB' then 'Nota de débito'
                                  WHEN @td='ORC' then 'Orçamento'
                                  WHEN @td='VD' then 'Venda a dinheiro'
```

```
                        END
        SET @tgr = CASE
                                WHEN @td='ECL' then 1
                                WHEN @td='FA' then 6
                                WHEN @td='FP' then 5
                                WHEN @td='GR' then 2
                                WHEN @td='GT' then 3
                                WHEN @td='NC' then 8
                                WHEN @td='NB' then 9
                                WHEN @td='ORC' then 4
                                WHEN @td='VD' then 7
                                ELSE 0
                        END


         IF @tgr <> 0
               BEGIN
                       INSERT    INTO        rpr_db.dbo.tgo_triggeroccurrence    (tgotgr_id,
                         tgo_datetime, tgo_dbtable, Tgousr_id)
                       VALUES    (@tgr,    convert(nvarchar,CURRENT_TIMESTAMP),    'CabecDoc',
                         @usrid)
                       set @id = scope_identity()
                       INSERT   INTO      rpr_db.dbo.toi_triggeroccurrenceinfo   (toitgo_id,
                         toi_attribute, toi_value, toi_description)
                       VALUES (@id, 'tipodoc', @td, 'tipo de documento criado - ' + @des)
                       INSERT   INTO      rpr_db.dbo.toi_triggeroccurrenceinfo   (toitgo_id,
                         toi_attribute, toi_value, toi_description)
                       VALUES (@id, 'serie', @s, 'serie da ' + @des)
                       INSERT   INTO      rpr_db.dbo.toi_triggeroccurrenceinfo   (toitgo_id,
                         toi_attribute, toi_value, toi_description)
                       VALUES (@id, 'nroficial', @nr, 'numero da ' + @des)
                       INSERT   INTO      rpr_db.dbo.toi_triggeroccurrenceinfo   (toitgo_id,
                         toi_attribute, toi_value, toi_description)
                       VALUES (@id, 'datadoc', @dt, 'data da ' + @des)
                               END
                       END
GO
```

```
-- =============================================
-- Author:          Rui Marques
-- Create date:     March 20th, 2013
-- Description:     Trigger for document updates
-- =============================================
CREATE TRIGGER [dbo].[TGR_UPDATE_DOCUMENTHEADER]
   ON  [dbo].[CabecDoc]
   AFTER UPDATE
AS
BEGIN

     SET NOCOUNT ON;

     DECLARE
        @id    int,                -- id of trigger that have occurred
        @td    nvarchar(5),        -- type of document
        @s     nvarchar(2),        -- document serie
        @nr    int,                -- document oficial number
        @dt    date,               -- document date
        @tgr   int,
        @des   nvarchar(50),
        @usr   nvarchar(50),
        @usrId int

     SET     @td = (SELECT TipoDoc FROM deleted)
     SET     @s  = (SELECT Serie FROM deleted)
     SET     @nr = (SELECT NumDoc FROM deleted)
     SET     @dt = (SELECT Data FROM deleted)
     SET     @usr = (SELECT Utilizador FROM inserted)
     SET   @usrId = (SELECT usr_id from rpr_db.dbo.usr_user where usr_description = @usr)
     IF @usrId is null
            SET @usrId = 9999

     SET @des = CASE
                                  WHEN @td='ECL' then 'Encomenda a cliente'
                                  WHEN @td='FA' then 'Fatura'
                                  WHEN @td='FP' then 'Fatura pró-forma'
                                  WHEN @td='GR' then 'Guia de remessa'
                                  WHEN @td='GT' then 'Guia de transporte'
                                  WHEN @td='NC' then 'Nota de crédito'
                                  WHEN @td='NB' then 'Nota de débito'
                                  WHEN @td='ORC' then 'Orçamento'
                                  WHEN @td='VD' then 'Venda a dinheiro'
                            END
     SET @tgr = CASE
                                  WHEN @td='ECL' then 37
                                  WHEN @td='FA' then 42
                                  WHEN @td='FP' then 41
```

```
                                WHEN @td='GR' then 38
                                WHEN @td='GT' then 39
                                WHEN @td='NC' then 44
                                WHEN @td='NB' then 45
                                WHEN @td='ORC' then 40
                                WHEN @td='VD' then 43
                                ELSE 0
                        END

    if @tgr <> 0
    begin
            INSERT  INTO    rpr_db.dbo.tgo_triggeroccurrence  (tgotgr_id,  tgo_datetime,
                tgo_dbtable, Tgousr_id)
            VALUES (@tgr,convert(nvarchar,CURRENT_TIMESTAMP),'CabecDoc', @usrId);
                  set @id = scope_identity()
            INSERT     INTO         rpr_db.dbo.toi_triggeroccurrenceinfo     (toitgo_id,
                toi_attribute, toi_value, toi_description)
            VALUES (@id, 'tipodoc', @td, 'tipo de documento - ' + @des);
            INSERT     INTO         rpr_db.dbo.toi_triggeroccurrenceinfo     (toitgo_id,
                toi_attribute, toi_value, toi_description)
            VALUES (@id, 'serie', @s, 'serie ' + @des);
            INSERT     INTO         rpr_db.dbo.toi_triggeroccurrenceinfo     (toitgo_id,
                toi_attribute, toi_value, toi_description)
            VALUES (@id, 'nroficial', @nr, 'numero ' + @des);
            INSERT     INTO         rpr_db.dbo.toi_triggeroccurrenceinfo     (toitgo_id,
                toi_attribute, toi_value, toi_description)
            VALUES (@id, 'datadoc', @dt, 'data ' + @des);

    end

END
```

```
-- ============================================
-- Author:          Rui Marques
-- Create date:     March 19th, 2013
-- Description:     Trigger for document deletetion
-- ============================================
CREATE TRIGGER [dbo].[TGR_DELETE_DOCUMENTS]
   ON  [dbo].[CabecDoc]
   AFTER DELETE
AS
BEGIN
     SET NOCOUNT ON;

     DECLARE
        @id    int,
        @td    nvarchar(5),
        @s     nvarchar(2),
        @nr    int,
        @dt    date,
        @des   nvarchar(50),
        @tgr   int,
        @usr   nvarchar(50),
        @usrId int

     SET     @td = (SELECT TipoDoc FROM deleted)
     SET     @s  = (SELECT Serie FROM deleted)
     SET     @nr = (SELECT NumDoc FROM deleted)
     SET     @dt = (SELECT Data FROM deleted)
     SET     @usr = (SELECT Utilizador FROM deleted)
     SET   @usrId = (SELECT usr_id from rpr_db.dbo.usr_user where usr_description = @usr)
     IF @usrId is null
            SET @usrId = 9999

     SET @des = CASE
                                  WHEN @td='ECL' then 'Encomenda a cliente'
                                  WHEN @td='FA' then 'Fatura'
                                  WHEN @td='FP' then 'Fatura pró-forma'
                                  WHEN @td='GR' then 'Guia de remessa'
                                  WHEN @td='GT' then 'Guia de transporte'
                                  WHEN @td='NC' then 'Nota de crédito'
                                  WHEN @td='NB' then 'Nota de débito'
                                  WHEN @td='ORC' then 'Orçamento'
                                  WHEN @td='VD' then 'Venda a dinheiro'
                            END
     SET @tgr = CASE
                                  WHEN @td='ECL' then 10
                                  WHEN @td='FA' then 15
                                  WHEN @td='FP' then 14
                                  WHEN @td='GR' then 11
```

```
                                WHEN @td='GT' then 12
                                WHEN @td='NC' then 17
                                WHEN @td='NB' then 18
                                WHEN @td='ORC' then 13
                                WHEN @td='VD' then 16
                                ELSE 0
                        END
    if @tgr <> 0
    begin
            INSERT INTO   rpr_db.dbo.tgo_triggeroccurrence (tgotgr_id, tgo_datetime,
                tgo_dbtable, Tgousr_id)
            VALUES (@tgr, convert(nvarchar,CURRENT_TIMESTAMP), 'CabecDoc', @usrId)
            set @id = scope_identity()
            INSERT    INTO        rpr_db.dbo.toi_triggeroccurrenceinfo    (toitgo_id,
                toi_attribute, toi_value, toi_description)
            VALUES (@id, 'tipodoc', @td, 'tipo de documento - ' + @des);
            INSERT    INTO        rpr_db.dbo.toi_triggeroccurrenceinfo    (toitgo_id,
                toi_attribute, toi_value, toi_description)
            VALUES (@id, 'serie', @s, 'serie - ' + @des);
            INSERT    INTO        rpr_db.dbo.toi_triggeroccurrenceinfo    (toitgo_id,
                toi_attribute, toi_value, toi_description)
            VALUES (@id, 'nroficial', @nr, 'numero - ' + @des);
            INSERT    INTO        rpr_db.dbo.toi_triggeroccurrenceinfo    (toitgo_id,
                toi_attribute, toi_value, toi_description)
            VALUES (@id, 'datadoc', @dt, 'data - ' + @des);
    end

END
```