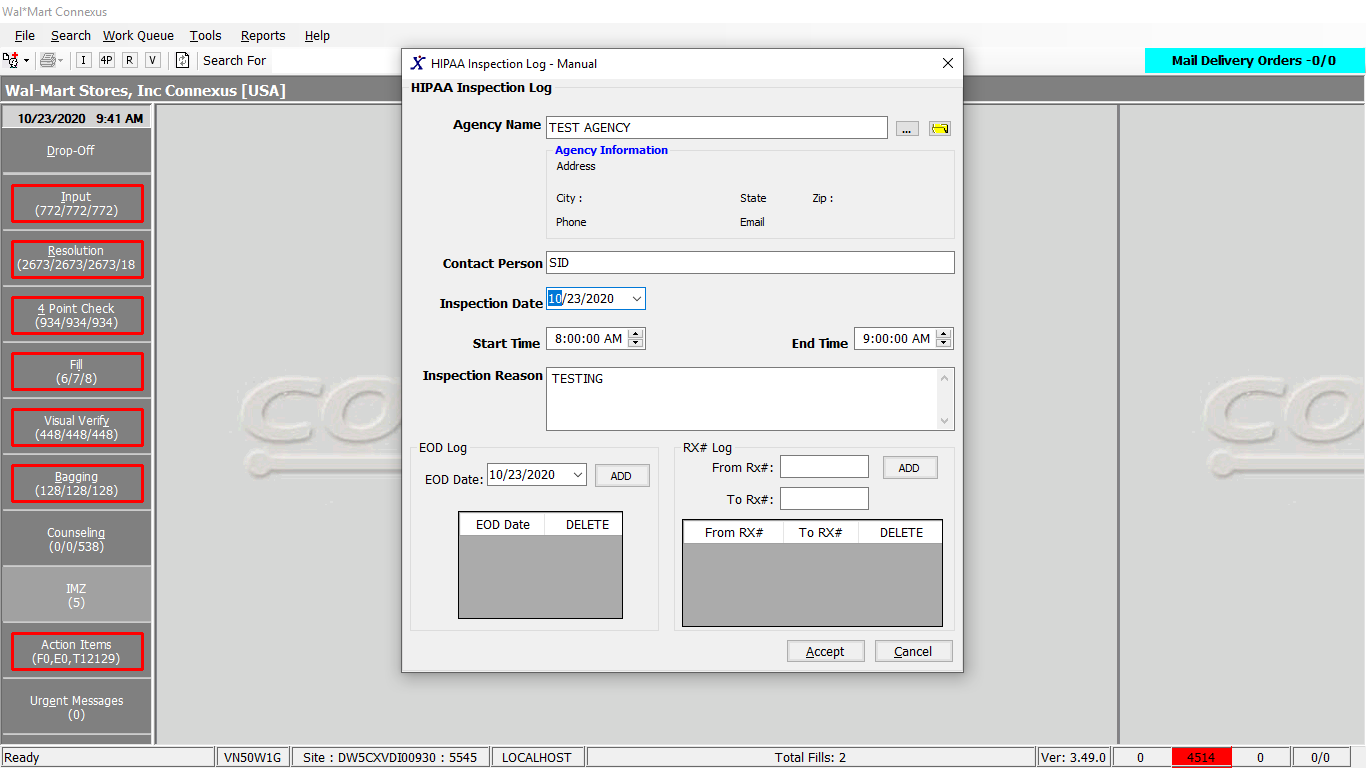
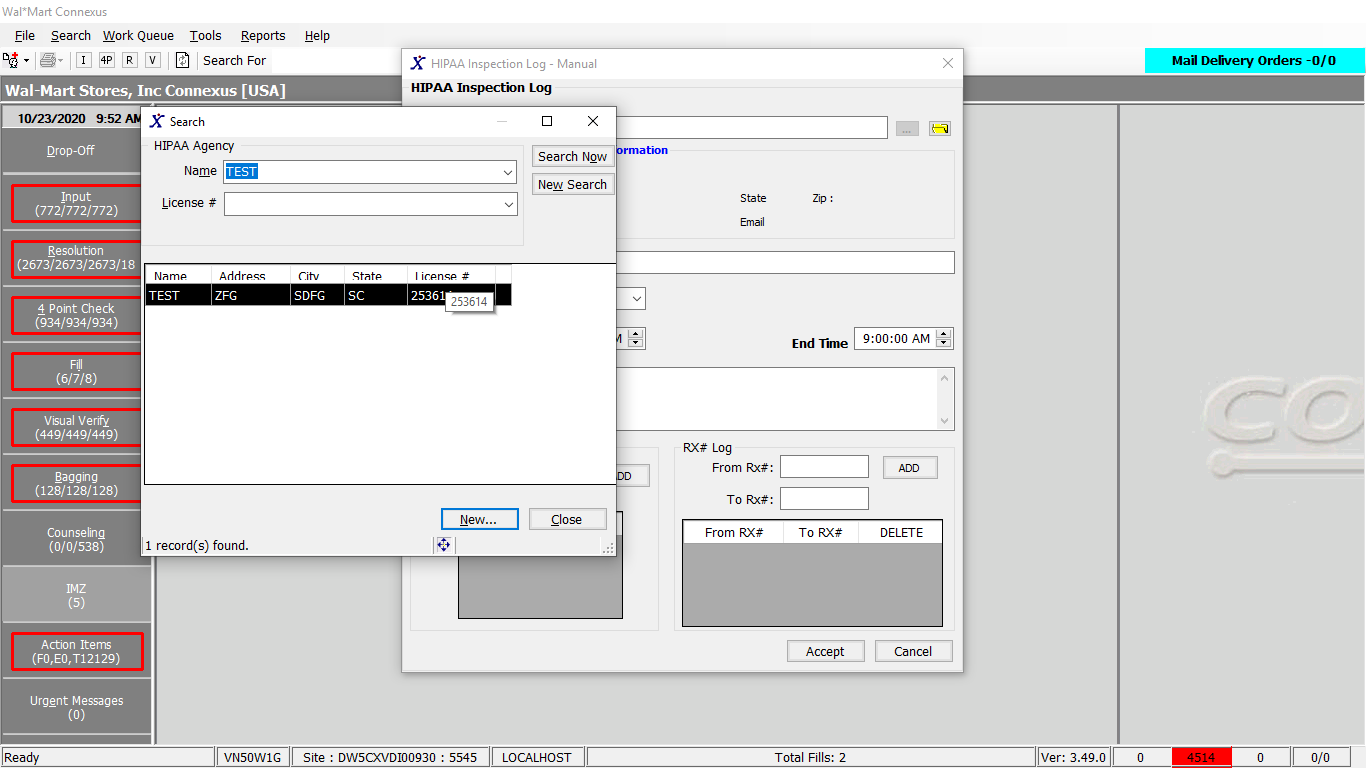
## Reports Menu/HIPAA/HIPAA Inspection Log:

1)Request:

🡪The initial page after Hipaa inspection log click event is:

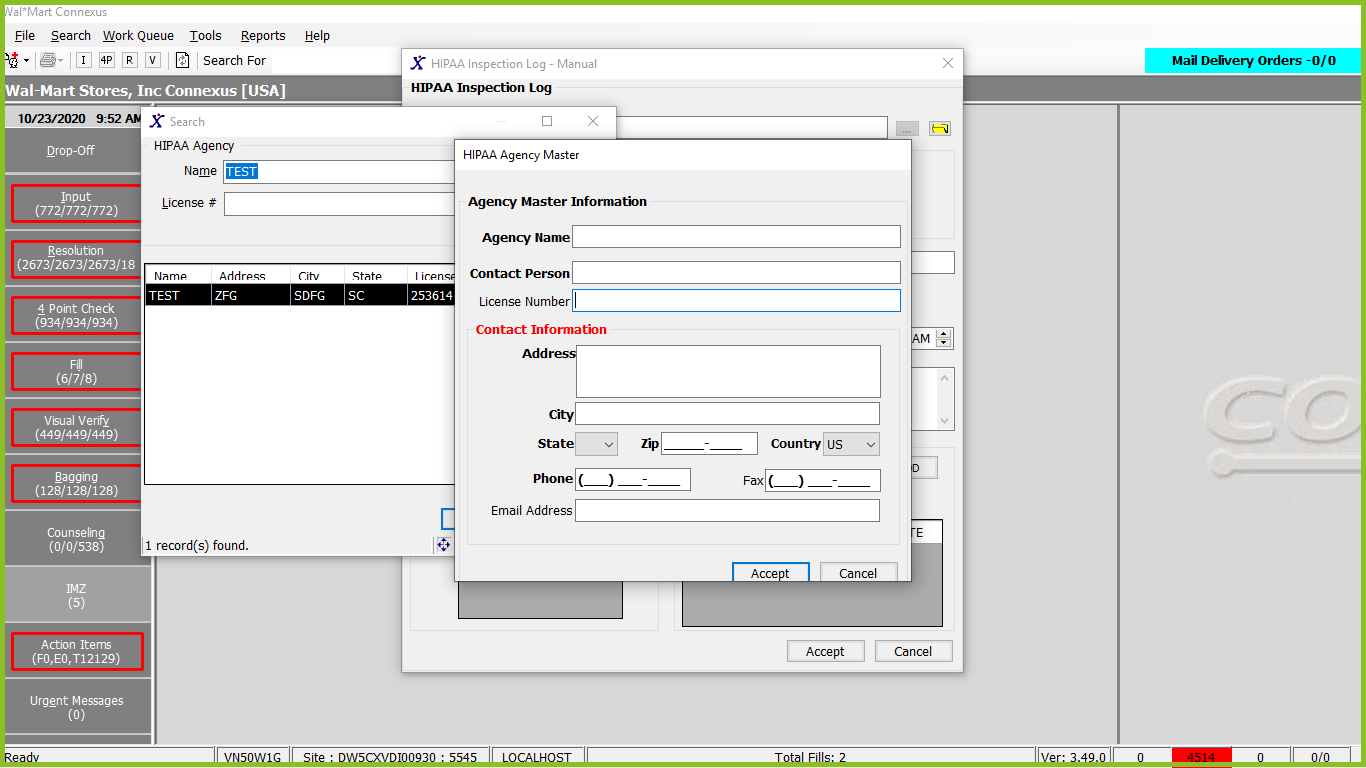


🡪The Agency name must be verified. That is either it should already be in the data base or the new agency Should be created by entering the agency details. Refer the below snapshots.



Note: The search criteria for agency are “Name” and “LicenceId”.

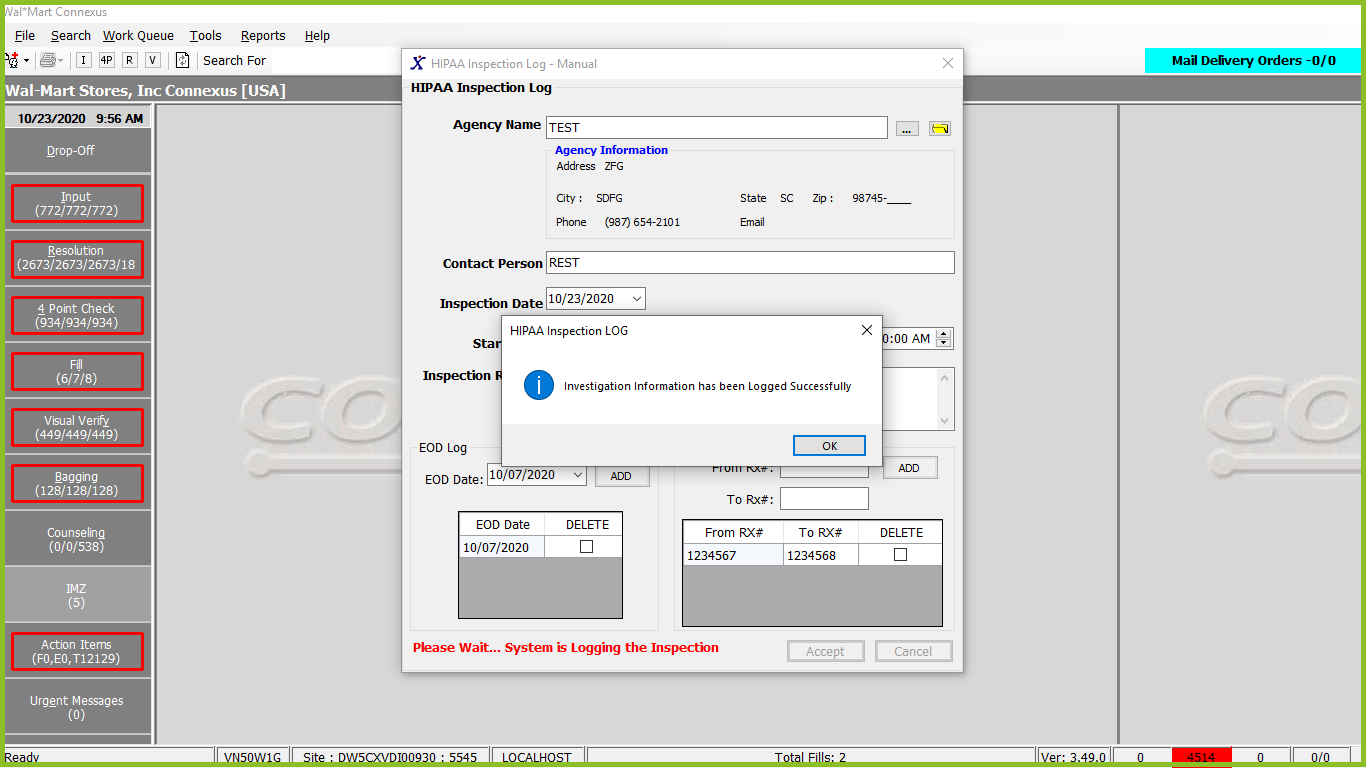
🡪If the log is for new agency then it should be created.



2)Miscellaneous points:

🡪The EOD date and Rx range are optional. If they are entered they must also be saved into database.

3)Response: The response of after “Accept” click event will be successful insertion message.



3)DAL method and stored procedure:

Public Function InsertInspectionTransactionLog(ByVal siteId As Integer,

ByVal transactionLogInfo As InspectionTransactionLogBE,

ByVal rxValue As String, ByVal eodValue As String,

ByVal inspectionValue As String) As Integer Implements IHIPAADAL.InsertInspectionTransactionLog

Using tracer As New Tracer(ConnexusCommonConstants.ConnexusTracer)

Dim paramList As New List(Of DbParameter)

paramList.AddParameter(DbType.Int32, transactionLogInfo.RequestingPartyId)

paramList.AddParameter(DbType.Int16, transactionLogInfo.RequestorTypeCode)

paramList.AddParameter(DbType.DateTime, transactionLogInfo.RequestTS)

paramList.AddParameter(DbType.String, transactionLogInfo.RequestorName)

paramList.AddParameter(DbType.DateTime, transactionLogInfo.InspectionStartTS)

paramList.AddParameter(DbType.DateTime, transactionLogInfo.InspectionEndTS)

paramList.AddParameter(DbType.DateTime, transactionLogInfo.DisclosureTS)

paramList.AddParameter(DbType.String, transactionLogInfo.UserId)

paramList.AddParameter(DbType.String, transactionLogInfo.RequestText)

paramList.AddParameter(DbType.Int16, transactionLogInfo.RequestTypeCode)

paramList.AddParameter(DbType.String, eodValue)

If String.IsNullOrEmpty(rxValue) Then

paramList.AddParameter(DbType.String, Nothing)

Else

paramList.AddParameter(DbType.String, rxValue)

End If

If String.IsNullOrEmpty(inspectionValue) Then

paramList.AddParameter(DbType.String, Nothing)

Else

paramList.AddParameter(DbType.String, inspectionValue)

End If

Dim inspectionTransactionLog As Integer

Using dr As IDataReader = ExecuteReader(DatabaseEnum.rx21c, siteId, HipaaQueries.InsertInspectionTransactionLogCall, paramList)

If dr.Read Then

inspectionTransactionLog = dr.GetValueOrDefault(Of Int32)(0)

End If

End Using

Return inspectionTransactionLog

End Using

End Function

--------------End of function-------------------------------------------------------------------

🡪The Following procedure has been called in the above highlighted line.

Friend Const InsertInspectionTransactionLogCall As String = "{Call insertinspectiontranslog\_new(?,?,?,?,?,?,?,?,?,?,?,?,?)}"

Procedure: insertinspectiontranslog\_new

**CREATE** **PROCEDURE** insertinspectiontranslog\_new(irqst\_part\_id **integer**,

srqstor\_type\_cd **smallint**,

dtrequest\_ts **datetime** **year** **to** **second**,

crequestor\_namechar(30),

dtinspection\_start\_ts **datetime** **year** **to** **second**,

dtinspection\_end\_ts **datetime** **year** **to** **second**,

dtdisclosure\_ts **datetime** **year** **to** **second**,

ccreate\_userid **char**(10),crequest\_txt **varchar**(255),

sreport\_type\_code **smallint**,

cSQLEOD LIST(**lvarchar** **not** **null**),

cSQLRX **lvarchar**,

CSQLINSPECTIONTIME **lvarchar**,

iDebug **smallint** **DEFAULT** 0)**returning** **integer** ;

**define** in\_trans **smallint**;

**define** sql\_err\_code **integer**;

**define** isam\_err\_code **integer**;

**define** err\_desc **varchar**(255);

**define** ihipaa\_tran\_id **integer**;

**define** lstmt **lvarchar**;

**define** lRXRetval **lvarchar**;

**define** lINSPECTIONTIMERetVal **lvarchar**;

**define** lEODRetVal **lvarchar**;

**define** lTmpRetval **lvarchar**;

**define** lSQLEOD **lvarchar**;

**define** lExecEODSQL **lvarchar**;

**define** CONST\_SUBJ\_AGENCY **integer**;

**define** CONST\_WORK\_SEQNO **integer**;

**define** CONST\_FAX\_SEQNO **integer**;

**define** lRemedyRqstTxt **lvarchar**;

-- SCCSID: @(#) name:InsertInspectionTransLog\_NEW.sql stored\_procedure scid=version:1.11 date\_modified:05/03/08 derived\_by: \_\_PWR\_CHG\_USER\_\_

**on** **exception** **set** sql\_err\_code,isam\_err\_code,err\_desc

**if** in\_trans=0

**then**

**rollback** **work**;

**return** -1;

**end** **if**

**raise** **exception** sql\_err\_code,isam\_err\_code,err\_desc;

**end** **exception**;

**on** **exception** **in** (-958)

**let** in\_trans=0;

**delete** **from** hirqr\_tmp;

**end** **exception** **with** **resume**;

**on** **exception** **in** (-316)

**let** in\_trans=0;

**drop** **index** hirqr\_tmp\_idx;

**CREATE** **INDEX** hirqr\_tmp\_idx **ON** hirqr\_tmp(rx\_id);

**end** **exception** **with** **resume**;

**on** **exception** **in** (-535,-744)

**let** in\_trans=1;

**end** **exception** **with** **resume**;

IFiDebug = 1 **THEN**

**SET** **DEBUG** **FILE** **TO** '/tmp/insertinspectiontranslog\_new.trace';

**TRACE** **ON**;

**END** **IF**

**let** in\_trans=1;

**CREATE** **TEMP** **TABLE** hirqr\_tmp

(rx\_id **INTEGER** **not** **null**,rx\_version\_seq\_nbr **smallint** **not** **null**)

**WITH** **NO** **LOG**;

**LET** CONST\_SUBJ\_AGENCY =308;

**LET** CONST\_WORK\_SEQNO = 2;

**LET** CONST\_FAX\_SEQNO = 3;

**LET** lRemedyRqstTxt = "";

**LET** lstmt = "";

**LET** lSQLEOD = ' select distinct rx.rx\_id, rx.rx\_version\_seq\_nbr ' ;

**LET** lSQLEOD = lSQLEOD || ' ' || ' FROM rx\_fill rxf, rx rx, patient pat,prescriber pres, rx\_input rxi, ';

**LET** lSQLEOD = lSQLEOD || ' ' || ' OUTER rx\_order\_detail rod, OUTER subject\_address sa, ';

**LET** lSQLEOD = lSQLEOD || ' ' || 'OUTER (prescriber\_clinic pcli, OUTER subject\_address sa1),OUTER prescriber\_license presl, ';

**LET** lSQLEOD = lSQLEOD || ' ' || 'OUTER prescriber\_license prestate, OUTER subject\_comm sbjc, ';

--- LET lSQLEOD = lSQLEOD || ' ' || 'OUTER pharmacy\_offering:pharmacy\_product pp, OUTER Pharmacy\_code\_txt pct ';

**LET** lSQLEOD= lSQLEOD || ' ' || 'OUTER pharmacy\_offering:pharmacy\_product pp, OUTER daw\_text pct ';

**LET** lSQLEOD = lSQLEOD || ' ' || 'Where rxf.rx\_id = rx.rx\_id And rxf.fill\_type\_code <> 7003 ';

**LET** lSQLEOD = lSQLEOD || ' ' || 'AND rxf.rx\_version\_seq\_nbr =rx.rx\_version\_seq\_nbr AND rx.input\_id = rxi.input\_id';

**LET** lSQLEOD = lSQLEOD || ' ' || 'AND rx.rx\_version\_seq\_nbr = 1 AND rxi.input\_source\_code = 3800 ';

**LET** lSQLEOD = lSQLEOD || ' ' || 'AND rx.prescriber\_id = pres.prescriber\_id AND rx.prescriber\_id = presl.prescriber\_id ';

-- LET lSQLEOD = lSQLEOD || ' ' || 'AND presl.license\_type\_code = 800 AND rx.prescriber\_id = prestate.prescriber\_id ';

**LET** lSQLEOD = lSQLEOD || ' ' || 'AND presl.license\_type\_code in (800,31101) AND rx.prescriber\_id = prestate.prescriber\_id ';

-- LET lSQLEOD = lSQLEOD || ' ' || 'AND prestate.license\_type\_code = 801 AND rx.patient\_id = pat.patient\_id ';

**LET** lSQLEOD = lSQLEOD || ' ' || 'AND prestate.license\_type\_code in (801, 31101) AND rx.patient\_id = pat.patient\_id ';

**LET** lSQLEOD = lSQLEOD || ' ' || 'AND pres.prescriber\_id = sbjc.subject\_id AND sbjc.subject\_type\_code = 303 ';

**LET** lSQLEOD = lSQLEOD || ' ' || 'AND sbjc.comm\_method\_code = 2 AND sbjc.comm\_seq\_nbr = 1 ';

**LET** lSQLEOD = lSQLEOD || ' ' || 'AND rod.rx\_fill\_id = rxf.rx\_fill\_id AND rod.rx\_id = rx.rx\_id ';

**LET** lSQLEOD = lSQLEOD || ' ' || 'AND rod.request\_type\_code IN (1600, 1602,1607) ';

**LET** lSQLEOD = lSQLEOD || ' ' || 'AND rxf.rx\_fill\_seq\_nbr = 1 AND rx.rx\_nbr IS NOT NULL ';

LETlSQLEOD = lSQLEOD || ' ' || 'AND sa.subject\_id = pat.patient\_id AND sa.subject\_type\_code = 300 ';

**LET** lSQLEOD = lSQLEOD || ' ' || 'AND pres.prescriber\_id = pcli.prescriber\_id AND pcli.primary\_clinic\_ind = ''Y'' ';

**LET** lSQLEOD = lSQLEOD || ' ' ||'AND pcli.clinic\_id = sa1.subject\_id AND sa1.subject\_type\_code = 303 ';

**LET** lSQLEOD = lSQLEOD || ' ' || 'AND ( ( rx.status\_code IN ( 20601, 20606 ) and rxf.status\_code <> 20403 ) OR ';

**LET** lSQLEOD = lSQLEOD || ' ' || '( rx.status\_code = 20604 )) AND pp.product\_mds\_fam\_id = rx.product\_mds\_fam\_id ';

--- LET lSQLEOD = lSQLEOD || ' ' || 'AND pct.code\_id = rxf.daw\_code AND pct.language\_code = 101 ';

**LET** lSQLEOD = lSQLEOD || ' ' || 'AND pct.daw\_code = rxf.daw\_code AND pct.language\_code = 101';

**LET** lSQLEOD = lSQLEOD || ' ' || 'AND DATE(rx.version\_seq\_nbr\_ts) = ';

**FOREACH** **SELECT** a.execfromlist **INTO** lExecEODSQL **FROM** **table**(cSQLEOD)a(execfromlist)

**IF** lExecEODSQL = 'NO EODS' **then**

**EXIT** **FOREACH**;**END** **IF**

**LET** lstmt = ' insert into hirqr\_tmp (rx\_id,rx\_version\_seq\_nbr)';

**LET** lstmt = lstmt || ' ' || lSQLEOD ;

**LET** lstmt = lstmt || ' ' || lExecEODSQL ;

**LET** lEODRetVal = **exec**(lstmt);

**END** **FOREACH**

**LET** lstmt = "";

**if** cSQLRX **IS** **NOT** **NULL** **then**

**LET** lstmt = ' insert into hirqr\_tmp (rx\_id,rx\_version\_seq\_nbr)';

**LET** lstmt = lstmt || ' ' || ' select distinct rx.rx\_id, rx.rx\_version\_seq\_nbr ' ;

**LET** lstmt = lstmt ||' ' || cSQLRX ;

**LET** lRXRetval = **exec**(lstmt);

**end** **if**

**LET** lstmt = "";

**if** CSQLINSPECTIONTIME **IS** **NOT** **NULL** **then**

**LET** lstmt = ' insert into hirqr\_tmp (rx\_id,rx\_version\_seq\_nbr)';

**LET** lstmt = lstmt || ' ' || ' select distinct rx.rx\_id, rx.rx\_version\_seq\_nbr ' ;

**LET** lstmt = lstmt || ' ' || CSQLINSPECTIONTIME ;

**LET** lINSPECTIONTIMERetVal = **exec**(lstmt);

**end** **if**

**CREATE** **INDEX** hirqr\_tmp\_idx **ON** hirqr\_tmp(rx\_id);

**let** in\_trans=0;

**if** in\_trans=0

**then**

**begin** **work**;

**end** **if**

-- Insert intoHippa Info Request Table

**INSERT** **INTO** Hipaa\_info\_Request

(rqstg\_party\_id, requestor\_type\_cd, request\_ts, requestor\_name,

request\_start\_ts, request\_end\_ts, disclosure\_ts, phm\_signature\_id,

create\_ts, create\_userid, request\_txt, relationship\_code,request\_type\_code)

**VALUES**

(irqst\_part\_id,srqstor\_type\_cd, dtrequest\_ts, crequestor\_name,

dtinspection\_start\_ts,dtinspection\_end\_ts, dtdisclosure\_ts,**NULL**,

**CURRENT**,ccreate\_userid,crequest\_txt,**NULL**,sreport\_type\_code);

**LET** ihipaa\_tran\_id= dbinfo("SQLCA.SQLERRD1");

**LET** lstmt = ' insert into hipaa\_info\_rqst\_rx (rx\_id,rx\_version\_seq\_nbr,hipaa\_info\_rqst\_id)';

**LET** lstmt = lstmt || ' ' || ' select distinct rx\_id, rx\_version\_seq\_nbr, ' || ihipaa\_tran\_id ;

**LET** lstmt = lstmt || ' from hirqr\_tmp ' ;

**LET** lTmpRetval = **exec**(lstmt);

**Select** 'Agency Name:' || **trim**(hipaa\_agency\_name) || ',

Address: ' || **trim**(nvl(sa.address\_line\_1,"")) || ',' || **trim**(nvl(sa.address\_line\_2,"")) ||

', City: ' || **trim**(nvl(sa.city\_name,"")) ||

',State: ' || **trim**(nvl(sa.state\_prov\_code,"")) ||

',Zip: ' || **trim**(nvl(sa.postal\_code,"")) ||

',Country: ' || **trim**(nvl(sa.country\_code,"")) ||

',Office Phone: ' || **trim**(nvl(sc1.communication\_id,"")) ||

',Fax: ' || **trim**(nvl(sc2.communication\_id,""))

**into**

lRemedyRqstTxt

**from**

hipaa\_agency,

**outer** subject\_address sa ,

**outer** subject\_comm sc1,

**outer** subject\_comm sc2

**where**

hipaa\_agency\_id = irqst\_part\_id

**and** sa.subject\_id = hipaa\_agency.hipaa\_agency\_id

**and** sa.subject\_type\_code = CONST\_SUBJ\_AGENCY

**and** sc1.subject\_id = hipaa\_agency\_id

**and** sc1.subject\_type\_code = CONST\_SUBJ\_AGENCY

**and** sc1.comm\_seq\_nbr = CONST\_WORK\_SEQNO

**and** sc2.subject\_id = hipaa\_agency\_id

**and** sc2.subject\_type\_code = CONST\_SUBJ\_AGENCY

**and** sc2.comm\_seq\_nbr = CONST\_FAX\_SEQNO ;

**LET** lRemedyRqstTxt = lRemedyRqstTxt || ' Inspection Reason: ' || crequest\_txt ;

**if** in\_trans=0

**then**

**commit** **work**;

**end** **if**

**drop** **index** hirqr\_tmp\_idx;

**drop** **table** hirqr\_tmp;

{ here it's going to invoke the shell program that is supposedly creating an AUTO REMEDY ticket

}

-- SYSTEM '/usr/**local**/bin/srxAutoRemedyForHIPAA.sh ' || """ || crequest\_txt || """ ;

SYSTEM 'nohup /usr/**local**/bin/srxAutoRemedyForHIPAA.sh '|| """" || lRemedyRqstTxt || """ & " ;

IF iDebug = 1 THEN

TRACE OFF;

END IF

return 1;

END PROCEDURE;