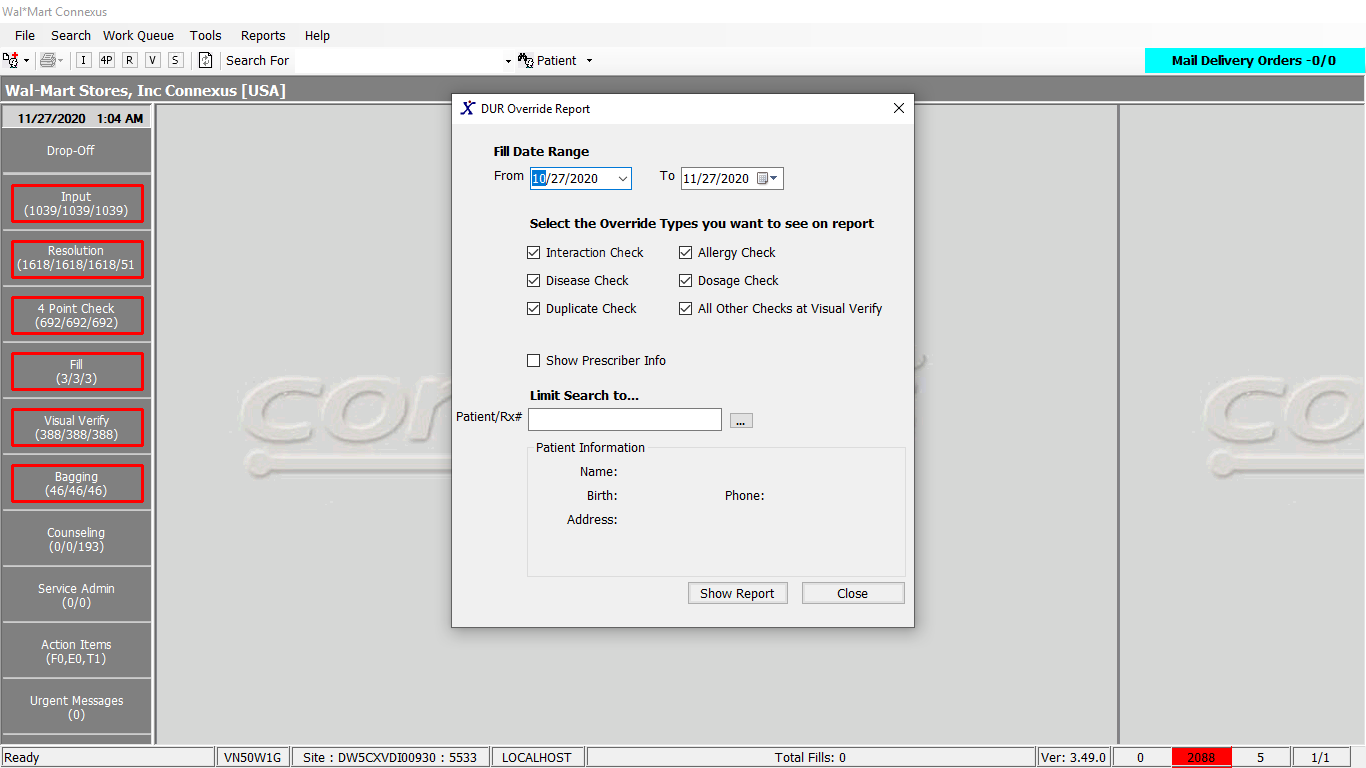
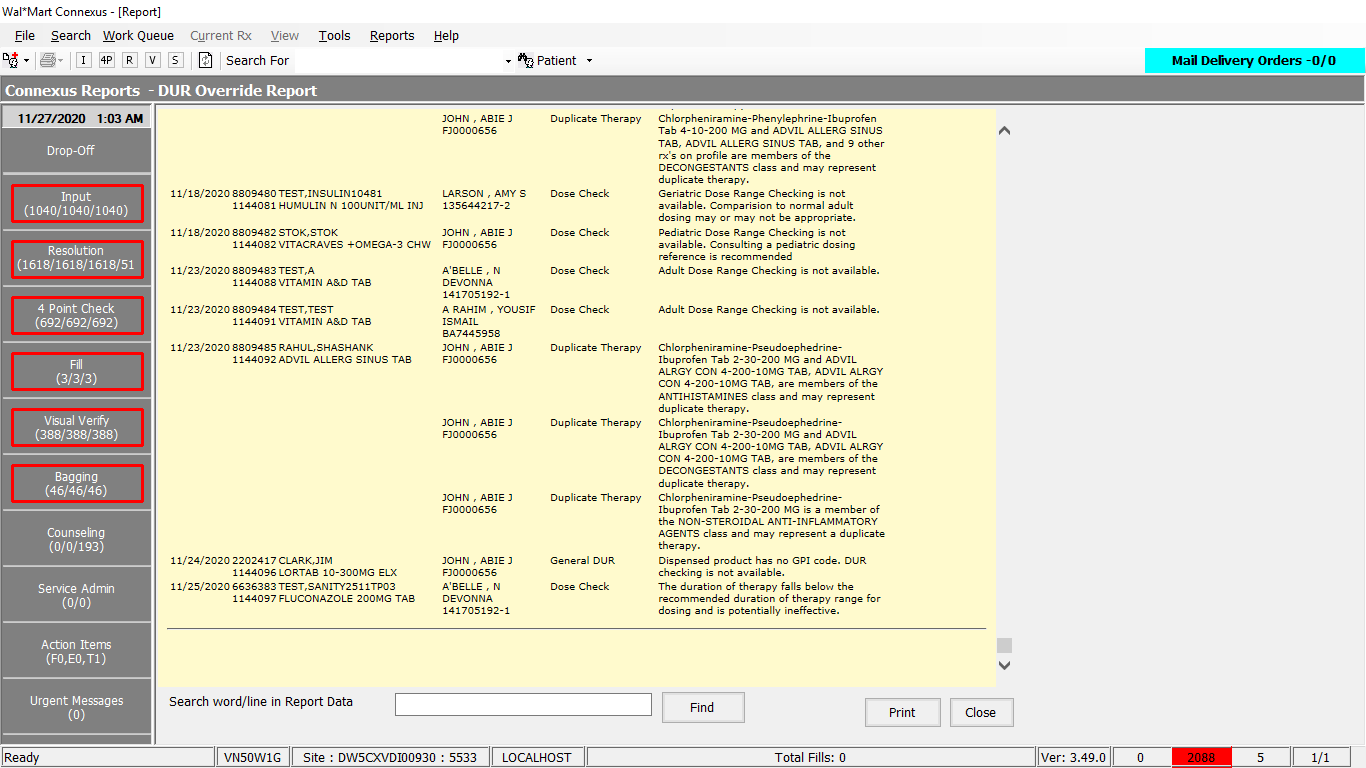
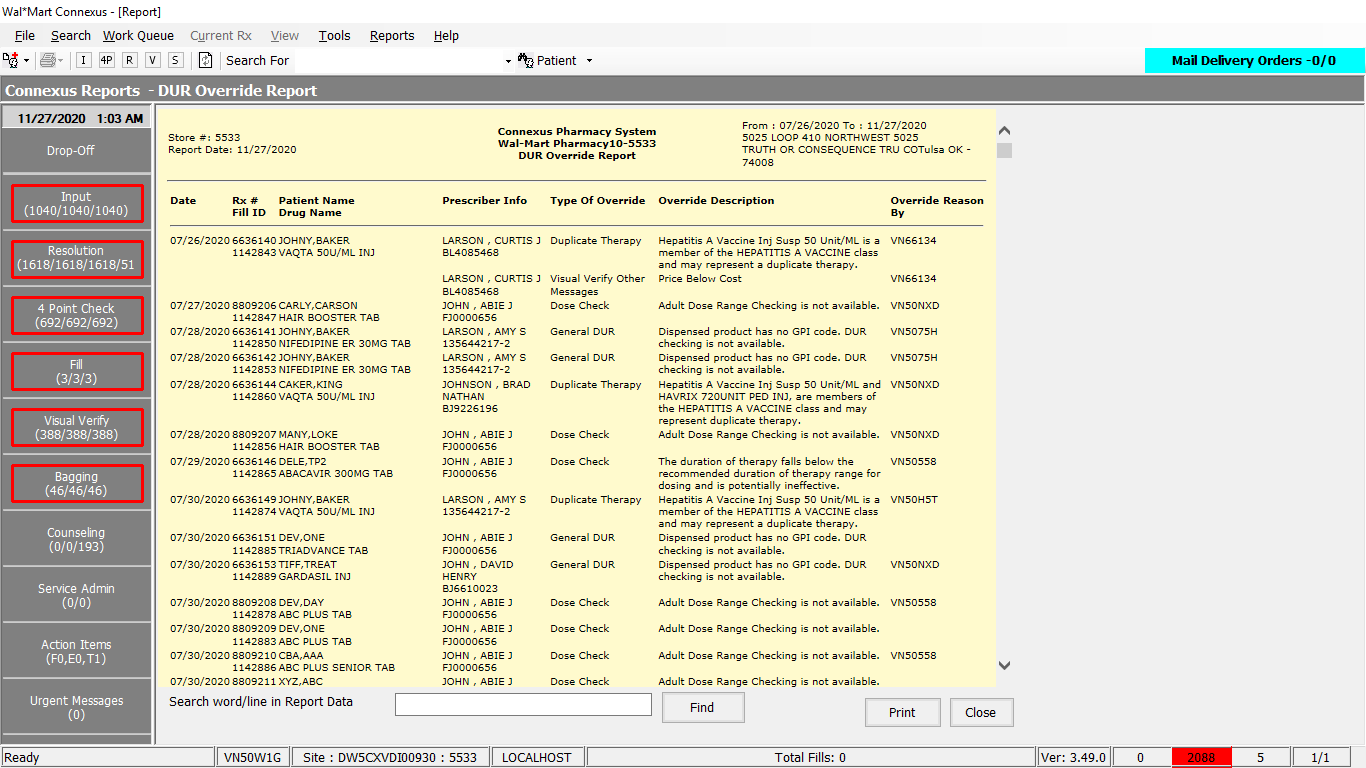
## Reports/ Rx Reports/ DUR Override Summary

Request:



Response:



Queries:

1. Friend Const GetDurOverrideDataSelectPart1 As String = "SELECT distinct rx.rx\_id, rx.rx\_version\_seq\_nbr, rf.rx\_fill\_id, rf.fill\_date, rx.rx\_nbr, TRIM(NVL(pa.last\_name,'')) || ',' || TRIM(NVL(pa.first\_name,'')) || ' ' || TRIM(NVL(pa.middle\_name,'')) PatientName, pp.short\_name, pct.code\_desc override\_type, rxc.conflict\_msg\_txt, rxc.resolution\_userid, pr.last\_name || ', ' || pr.first\_name || ' ' || pr.middle\_name pres\_name,NVL(rpl.license\_nbr,pl.license\_nbr) DEA, sc1.communication\_id pres\_phone "

2. Friend Const GetDurOverrideDataSelectPart2 As String = " , CASE WHEN rxc.dur\_reason\_code = 7 THEN ( SELECT cl.comment\_text FROM comment\_line cl,conflict\_comment cc WHERE cl.comment\_id=cc.comment\_id AND cc.conflict\_seq\_nbr=rxc.conflict\_seq\_nbr AND cc.rx\_id=rxc.rx\_id AND rxc.comment\_ind = 'Y') ELSE (SELECT drt.dur\_reason\_desc FROM dur\_reason\_txt drt, phm\_problem\_dur\_rsn ppd WHERE drt.dur\_reason\_code=ppd.dur\_reason\_code AND drt.dur\_reason\_code=rxc.dur\_reason\_code AND ppd.phm\_problem\_code=rxc.phm\_problem\_code) End Reason "

3. Friend Const GetDurOverrideDataSelectPart3 As String = " FROM rx, rx\_conflict rxc, pharmacy\_code\_txt pct, pharmacy\_offering:pharmacy\_product pp, rx\_fill rf, rx\_fill\_item rfi, pharmacy\_offering:pharmacy\_item pi, patient pa, prescriber pr, prescriber\_clinic pcli, OUTER subject\_comm sc1, OUTER prescriber\_license pl, OUTER rx\_prsbr\_license rpl WHERE rx.rx\_id = rxc.rx\_id AND rxc.conflict\_type\_code = pct.code\_id AND rx.patient\_id = pa.patient\_id AND rx.prescriber\_id = pr.prescriber\_id AND pr.prescriber\_id = pl.prescriber\_id AND pl.license\_seq\_nbr = (SELECT MIN(license\_seq\_nbr) FROM prescriber\_license pl1 WHERE pl1.prescriber\_id = pl.prescriber\_id AND (pl1.expiration\_date IS NULL OR pl1.expiration\_date >= TODAY) AND (pl1.effective\_date is NULL OR pl1.effective\_date <= TODAY)) AND rpl.rx\_id = rx.rx\_id AND rpl.prescriber\_id = rx.prescriber\_id AND rpl.license\_type\_code = 800 AND (((SELECT count(rfa1.activity\_code) FROM rx\_fill\_activity rfa1 WHERE rfa1.rx\_fill\_id = rf.rx\_fill\_id AND rfa1.activity\_code = 15116) > 0 AND rpl.license\_nbr like 'X%') OR ((SELECT count(rfa2.activity\_code) FROM rx\_fill\_activity rfa2 WHERE rfa2.rx\_fill\_id = rf.rx\_fill\_id AND rfa2.activity\_code = 15115) > 0 AND rpl.license\_nbr not like 'X%') OR ((SELECT count(rfa2.activity\_code) FROM rx\_fill\_activity rfa2 WHERE rfa2.rx\_fill\_id = rf.rx\_fill\_id AND rfa2.activity\_code in (15115,15116)) = 0)) AND pcli.primary\_clinic\_ind = 'Y' AND pcli.clinic\_id = sc1.subject\_id AND pr.prescriber\_id = pcli.prescriber\_id AND sc1.subject\_type\_code = 303 AND sc1.comm\_method\_code = 2 AND sc1.comm\_seq\_nbr = 2 AND rx.rx\_id = rf.rx\_id AND rx.rx\_version\_seq\_nbr = rf.rx\_version\_seq\_nbr AND rf.rx\_fill\_id = rxc.current\_rx\_fill\_id AND rfi.rx\_fill\_id = rf.rx\_fill\_id AND rfi.item\_mds\_fam\_id = pi.mds\_fam\_id AND pi.product\_mds\_fam\_id = pp.product\_mds\_fam\_id AND rxc.conflict\_type\_code IN ("

4. Public Const GetDateTime As String = "Select CURRENT dtmCurrent FROM systables where tabname = 'systables'"

DAL:

Public Function GetDurOverrideData(ByVal siteId As Integer,

Optional ByVal getDurOverrideDataInputInfo As \_

GetDurOverrideDataInputInfoBE = Nothing) \_

As List(Of GetDurOverrideDataInfoBE) Implements IReportDAL.GetDurOverrideData

Using (New Tracer(ConnexusCommonConstants.ConnexusTracer))

Dim sqlString As New StringBuilder

Dim DUR As New StringBuilder

Dim getDUROverrideDataInfoBEList As New List(Of GetDurOverrideDataInfoBE)

Dim paramList As New List(Of DbParameter)

If getDurOverrideDataInputInfo.Interaction Then

DUR.Append(CStr(DURConflict.CrossCheck).Trim)

Else

DUR.Append("0")

End If

'QA40: Added by - Infosys for LR - Rx Origin

Dim result As String

result = ReportDAL.GetBUPhmParm(siteId, ServicesCommonConstants.CheckDurOn)

'QA40: End by - Infosys for LR - Rx Origin

If getDurOverrideDataInputInfo.Allergy Then

DUR.Append("," & DURConflict.AllergyCheck)

End If

If getDurOverrideDataInputInfo.Disease Then

DUR.Append("," & DURConflict.DrugDisease)

End If

If getDurOverrideDataInputInfo.Dosage Then

DUR.Append("," & DURConflict.DoseCheck)

End If

If getDurOverrideDataInputInfo.Duplicate Then

DUR.Append("," & DURConflict.DuplicateTherapy)

End If

If getDurOverrideDataInputInfo.Other Then

DUR.Append("," & DURConflict.GeneralDUR & ",")

DUR.Append(DURConflict.GPIInterchange & "," & DURConflict.MedicationCrossCheck & ",")

DUR.Append(DURConflict.VisualVerifyOtherMessages)

End If

sqlString.Append(ReportQueries.GetDurOverrideDataSelectPart1)

'QA40: Added by akavoor- Infosys for LR - Log DUR

If result.Trim().ToUpper(CultureInfo.InvariantCulture) = "TRUE" Then

sqlString.Append(ReportQueries.GetDurOverrideDataSelectPart2)

End If

'End

sqlString.Append(ReportQueries.GetDurOverrideDataSelectPart3)

sqlString.Append(DUR)

sqlString.Append(")")

sqlString.Append(" AND rf.fill\_date >= ? ")

sqlString.Append(" AND rf.fill\_date <= ? ")

paramList.AddParameter(DbType.Date, getDurOverrideDataInputInfo.StartFillDate, "fillDate")

paramList.AddParameter(DbType.Date, getDurOverrideDataInputInfo.EndFillDate, "fillDate")

If getDurOverrideDataInputInfo.RxNbr > 0 Then

sqlString.Append(" AND rx.rx\_nbr = ? ")

paramList.AddParameter(DbType.Int32, getDurOverrideDataInputInfo.RxNbr, "rxNbr")

ElseIf getDurOverrideDataInputInfo.PatientId > 0 Then

sqlString.Append(" AND pa.patient\_id = ? ")

paramList.AddParameter(DbType.Int32, getDurOverrideDataInputInfo.PatientId, "patientId")

End If

If getDurOverrideDataInputInfo.Sort.Equals("F") Then

sqlString.Append(" ORDER BY rf.fill\_date, rx\_nbr, rf.rx\_fill\_id ")

Else

sqlString.Append(" ORDER BY pp.short\_name")

End If

Dim dataReader As IDataReader = ExecuteReader(DatabaseEnum.rx21c, siteId,

sqlString.ToString, paramList)

Dim getDUROverrideDataBE As GetDurOverrideDataInfoBE

While dataReader.Read

getDUROverrideDataBE = New GetDurOverrideDataInfoBE

getDUROverrideDataBE.CodeDesc = dataReader.GetValueOrDefault(Of String)("override\_type").SafeTrim()

getDUROverrideDataBE.ConflictMsgTxt = dataReader.GetValueOrDefault(Of String)("conflict\_msg\_txt").SafeTrim()

getDUROverrideDataBE.FillDate = dataReader.GetValueOrDefault(Of Date)("fill\_date")

getDUROverrideDataBE.LicenseNbr = dataReader.GetValueOrDefault(Of String)("DEA").SafeTrim()

If getDUROverrideDataBE.LicenseNbr IsNot Nothing Then

If getDUROverrideDataBE.LicenseNbr.Length > ServicesCommonConstants.PrescriberDEALength AndAlso Not getDUROverrideDataBE.LicenseNbr.Contains("-") Then

getDUROverrideDataBE.LicenseNbr = getDUROverrideDataBE.LicenseNbr.Substring(0, ServicesCommonConstants.PrescriberDEALength) & "-" & getDUROverrideDataBE.LicenseNbr.Substring(ServicesCommonConstants.PrescriberDEALength)

End If

End If

getDUROverrideDataBE.PatientName = dataReader.GetValueOrDefault(Of String)("PatientName").SafeTrim()

getDUROverrideDataBE.PresName = dataReader.GetValueOrDefault(Of String)("pres\_name").SafeTrim()

getDUROverrideDataBE.PresPhone = dataReader.GetValueOrDefault(Of String)("pres\_phone").SafeTrim()

getDUROverrideDataBE.ResolutionUserId = dataReader.GetValueOrDefault(Of String)("resolution\_userid").SafeTrim()

getDUROverrideDataBE.RxFillId = dataReader.GetValueOrDefault(Of Int32)("rx\_fill\_id")

getDUROverrideDataBE.RxId = dataReader.GetValueOrDefault(Of Int32)("rx\_id")

getDUROverrideDataBE.RxNbr = dataReader.GetValueOrDefault(Of Int32)("rx\_nbr")

getDUROverrideDataBE.RxVersionSeqNbr = dataReader.GetValueOrDefault(Of Int16)("rx\_version\_seq\_nbr")

getDUROverrideDataBE.ShortName = dataReader.GetValueOrDefault(Of String)("short\_name").SafeTrim()

If result.Trim().ToUpper(CultureInfo.InvariantCulture) = "TRUE" Then

getDUROverrideDataBE.Reason = dataReader.GetValueOrDefault(Of String)("reason").SafeTrim()

End If

getDUROverrideDataInfoBEList.Add(getDUROverrideDataBE)

End While

If getDUROverrideDataInfoBEList.Count > 0 Then

Return getDUROverrideDataInfoBEList

Else

Return Nothing

End If

End Using

End Function

Public Shared ReadOnly Property GetDBDateTime(ByVal siteId As Integer) As DateTime

Get

Using (New Tracer(ConnexusCommonConstants.ConnexusTracer))

Dim dbDateTime As DateTime

Dim offset As TimeSpan

Dim isOffsetValid As Boolean

'Get the offset time from cache

isOffsetValid = TimeSpan.TryParse(Convert.ToString(CnxCache(CStr(CacheKeys.OffsetTime))), offset)

If (Not isOffsetValid) Then

'get it from db

'execute query

dbDateTime = CDate(ExecuteScalar(DatabaseEnum.rx21c, siteId, BaseQueries.GetDateTime))

offset = dbDateTime.Subtract(DateTime.Now)

'Add it to Cache

CnxCache.Add(CStr(CacheKeys.OffsetTime), offset, CacheHelper.GetExpirationPolicy(CStr(CacheKeys.OffsetTime)))

End If

Return DateTime.Now.Add(offset)

End Using

End Get

End Property