**MCCF EDI TAS US583**

System Design Document



Department of Veterans Affairs

**March 2017**

Version 1.0

**User Story Number:** USRX-29

**User Story Name:** Reject Information Screen to display the Tax ID and PCN

**Product Backlog ID:** 29

**Backlog Priority:** Low

**Initial Sizing Estimate:**

**Rational ID:** tbd

**Rally ID:** US583

# Design – Summary

Modify the Reject Information screen to display the Tax ID and PCN. The Reject Information Screen is available from two options: Third Party Payer Rejects – Worklist (PSO REJECTS WORKLIST) and Third Party Payer Rejects – View/Process (PSO REJECTS VIEW/PROCESS).

Modify action REJ OPECC Reject Information to display the Tax ID. The REJ action is available from the ECME User Screen (BPS USER SCREEN).

List of Components:

* File: REJECT INFO Sub-File (#52.25)
* Routine: PSOREJU2
* Routine: BPSSCRRJ
* Routine: PSOREJP1
* Routine: PSO7P478
* Routine: BPSNCPD3
* Routine: PSOREJUT

# Design – Detail

| Subroutine Name | **DVINFO^PSOREJU2** | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify | | | Delete | | No Change | |
| Story | US583 | | | | | | | |
| Related Menu Options or ListMan Actions | Third Party Payer Rejects – Worklist [PSO REJECTS WORKLIST]  Third Party Payer Rejects – View/Process [PSO REJECTS VIEW/PROCESS] | | | | | | | |
| Related Routines | This Subroutine is Called By | | | This Subroutine Calls | | | | |
| HDR^PSOREJP1  HDR^PSOREJP5  DISPLAY^PSOREJU3 | | | $$DIVNCPDP^BPSBUTL  $$GET1^DIQ  $$RXSITE^PSOBPSUT | | | | |
| Related Integration Control Registrations (ICRs) | TBD; a new ICR will be created to govern the use of $$TAXID^IBCEF75. | | | | | | | |
| Data Passing | Parameter Input | | Parameter Output | | | Function Return Value | | Global Modified |
| Input Attribute Name and Definition | Name: RX  Definition: Rx IEN (#52)  New  Modify  Delete  No Change  Name: RFL  Definition: Refill #  New  Modify  Delete  No Change  Name: LM  Definition: ListManager Format (1-Yes, 2-No) Default-0  New  Modify  Delete  No Change | | | | | | | |
| Output Attribute Name and Definition | Name: $$DVINFO  Definition: String of Division information  New  Modify  Delete  No Change | | | | | | | |
| Current Logic | | | | | | | | |
| DVINFO(RX,RFL,LM) ; Returns header displayable Division Information  ;Input: (r) RX - Rx IEN (#52)  ; (o) RFL - Refill # (Default: most recent)  ; (o) LM - ListManager format? (1 - Yes / 0 - No) - Default: 0  N TXT,DVINFO,NCPNPI,DVIEN  S DVIEN=+$$RXSITE^PSOBPSUT(RX,RFL)  S DVINFO="Division : "\_$$GET1^DIQ(59,DVIEN,.01)  ;Display both NPI and NCPDP numbers - PSO\*7.0\*421  S NCPNPI=$$DIVNCPDP^BPSBUTL(DVIEN)  S $E(DVINFO,33)="NPI: "\_$P(NCPNPI,U,2)  S $E(DVINFO,$S($G(LM):59,1:52))="NCPDP: "\_$P(NCPNPI,U)  Q DVINFO | | | | | | | | |
| Modified Logic (Changes are highlighted) | | | | | | | | |
| DVINFO(RX,RFL,LM) ; Returns header displayable Division Information  ;Input: (r) RX - Rx IEN (#52)  ; (o) RFL - Refill # (Default: most recent)  ; (o) LM - ListManager format? (1 - Yes / 0 - No) - Default: 0  N TXT,DVINFO,NCPNPI,DVIEN  S DVIEN=+$$RXSITE^PSOBPSUT(RX,RFL)  S DVINFO="Division : "\_$E($$GET1^DIQ(59,DVIEN,.01),1,15)  ;Display both NPI and NCPDP numbers - PSO\*7.0\*421  S NCPNPI=$$DIVNCPDP^BPSBUTL(DVIEN)  S $E(DVINFO,28)="NPI: "\_$P(NCPNPI,U,2)  S $E(DVINFO,44))="NCPDP: "\_$P(NCPNPI,U)  S $E(DVINFO,62)=”TAX ID: “\_$$TAXID^IBCEF75  Q DVINFO | | | | | | | | |

| Subroutine Name | **DVINFO^BPSSCRRJ** | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify | | | Delete | | No Change | |
| Story | US583 | | | | | | | |
| Related Menu Options or ListMan Actions | REJ Action available from the ECME User Screen [BPS USER SCREEN]. | | | | | | | |
| Related Routines | This Subroutine is Called By | | | This Subroutine Calls | | | | |
| HDR^BPSSCRRJ | | | $$DIVNCPDP^BPSBUTL  $$GET1^DIQ  $$RXSITE^PSOBPSUT | | | | |
| Related Integration Control Registrations (ICRs) | TBD; a new ICR will be created to govern the use of $$TAXID^IBCEF75. | | | | | | | |
| Data Passing | Parameter Input | | Parameter Output | | | Function Return Value | | Global Modified |
| Input Attribute Name and Definition | Name: RX  Definition: Rx IEN (#52)  New  Modify  Delete  No Change  Name: RFL  Definition: Refill #  New  Modify  Delete  No Change | | | | | | | |
| Output Attribute Name and Definition | Name: $$DVINFO  Definition: String of Division information  New  Modify  Delete  No Change | | | | | | | |
| Current Logic | | | | | | | | |
| DVINFO(RX,RFL); header division data  ;Input: (r) RX - Rx IEN (#52)  ; (o) RFL - Refill #  N DVIEN,DVINFO,NCPNPI  S DVINFO="Division : "\_$$GET1^DIQ(9002313.59,BPORI59,11) ; Pharmacy Division name from BPS Transaction  ;Display both NPI and NCPDP numbers  S DVIEN=+$$RXSITE^PSOBPSUT(RX,RFL) ; ICR# 4701  S NCPNPI=$$DIVNCPDP^BPSBUTL(DVIEN)  S $E(DVINFO,33)="NPI: "\_$P(NCPNPI,U,2)  S $E(DVINFO,59)="NCPDP: "\_$P(NCPNPI,U,1)  Q DVINFO | | | | | | | | |
| Modified Logic (Changes are highlighted) | | | | | | | | |
| DVINFO(RX,RFL); header division data  ;Input: (r) RX - Rx IEN (#52)  ; (o) RFL - Refill #  N DVIEN,DVINFO,NCPNPI  S DVINFO="Division : "\_$E($$GET1^DIQ(9002313.59,BPORI59,11),1,15) ; Pharmacy Division name from BPS Transaction  ;Display both NPI and NCPDP numbers  S DVIEN=+$$RXSITE^PSOBPSUT(RX,RFL) ; ICR# 4701  S NCPNPI=$$DIVNCPDP^BPSBUTL(DVIEN)  S $E(DVINFO,28)="NPI: "\_$P(NCPNPI,U,2)  S $E(DVINFO,44)="NCPDP: "\_$P(NCPNPI,U,1)  S $E(DVINFO,62)=”TAX ID: “\_$$TAXID^IBCEF75  Q DVINFO | | | | | | | | |

| Subroutine Name | **GET^PSOREJU2** | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify | | | Delete | | No Change | |
| Story | US583 | | | | | | | |
| Related Menu Options or ListMan Actions | Third Party Payer Rejects – Worklist [PSO REJECTS WORKLIST]  Third Party Payer Rejects – View/Process [PSO REJECTS VIEW/PROCESS] | | | | | | | |
| Related Routines | This Subroutine is Called By | | | This Subroutine Calls | | | | |
| SETTMP^PSOREJP0  RUN^PSOREJP0  INIT^PSOREJP1  OPN^PSOREJP2  PSOCOB^PSOREJP3  DISPLAY^PSOREJU3  MULTI^PSOREJU4  SINGLE^PSOREJU4  FIND^PSOREJUT | | | GETS^DIQ  $$LSTRFL^PSOBPSU1 | | | | |
| Related Integration Control Registrations (ICRs) | N/A | | | | | | | |
| Data Passing | Parameter Input | | Parameter Output | | | Function Return Value | | Global Modified |
| Input Attribute Name and Definition | Name: RX  Definition: Rx IEN (#52)  New  Modify  Delete  No Change  Name: RFL  Definition: Refill #  New  Modify  Delete  No Change  Name: REJDATA  Definition: Array where Reject Fields are returned  New  Modify  Delete  No Change  Name: REJID  Definition: REJECT IEN in the PRESCRIPTION File  New  Modify  Delete  No Change  Name: OKCL  Definition: CLOSED REJECTS flag  New  Modify  Delete  No Change  Name: CODE  Definition: Only REJECTS with this CODE should be returned  New  Modify  Delete  No Change  Name: RRRFLAG  Definition: Reject Resolution Required flag  New  Modify  Delete  No Change | | | | | | | |
| Output Attribute Name and Definition | Name: REJDATA  Definition: Array in which reject fields will be returned.  New  Modify  Delete  No Change | | | | | | | |
| Current Logic | | | | | | | | |
| GET(RX,RFL,REJDATA,REJID,OKCL,CODE,RRRFLG) ; get reject data from subfile 52.25  ; Input: (r) RX - Rx IEN (#52)  ; (o) RFL - Refill # (Default: most recent)  ; (r) REJDATA(REJECT IEN,FIELD) - Array where these Reject fields will be returned:  ; "BIN" - Payer BIN number  ; "CODE" - Reject Code (79 or 88)  ; "DATE/TIME" - DATE/TIME Reject was detected  ; "PAYER MESSAGE" - Message returned by the payer  ; "REASON" - Reject Reason description (from payer)  ; "INSURANCE NAME" - Patient's Insurance Company Name  ; "INSURANCE POINTER" - Patient Insurance Company Pointer  ; "COB" - Coordination of Benefits  ; "GROUP NAME" - Patient's Insurance Group Name  ; "GROUP NUMBER" - Patient's Insurance Group Number  ; "CARDHOLDER ID" - Patient's Insurance Cardholder ID  ; "PLAN CONTACT" - Plan's Contact (eg., "1-800-...")  ; "PLAN PREVIOUS FILL DATE" - Last time Rx was paid by payer  ; "STATUS" - REJECTS status ("OPEN/UNRESOLVED" or "CLOSED/RESOLVED")  ; "DUR TEXT" - Payer's DUR description  ; "DUR ADD MSG TEXT" - Payer's DUR additional description  ; "OTHER REJECTS" - Other Rejects on the same response  ; "REASON SVC CODE" - Reason for Service Code  ; If REJECT is closed, the following fields will be returned:  ; "CLA CODE" - Clarification Code submitted  ; "PRIOR AUTH TYPE" - Prior Authorization Type  ; "PRIOR AUTH NUMBER" - Prior Authorization Type  ; "CLOSED DATE/TIME" - DATE/TIME Reject was closed  ; "CLOSED BY" - Name of the user responsible for closing Reject  ; "CLOSE REASON" - Reason for closing Reject (text)  ; "CLOSE COMMENTS" - User entered comments at close  ; (o) REJID - REJECT IEN in the PRESCRIPTION file for retrieve this REJECT  ; (o) OKCL - If set to 1, CLOSED REJECTs will also be returned  ; (o) CODE - Only REJECTs with this CODE should be returned  ; (o) RRRFLG - If set to 1 with CODE present, also return Reject Resolution Required REJECTs  ; If set to 1 and CODE not passed, then only return RRR REJECTs  ;  N REJS,ARRAY,REJFLD,IDX,COM,Z  ;  I '$D(RFL) S RFL=$$LSTRFL^PSOBPSU1(RX)  ;  K REJDATA  I '$O(^PSRX(RX,"REJ",0)) Q  ;  K REJS S RFL=+$G(RFL)  I $G(REJID) D  . I +$P($G(^PSRX(RX,"REJ",REJID,0)),"^",4)'=RFL Q  . I '$G(OKCL),$P($G(^PSRX(RX,"REJ",REJID,0)),"^",5) Q  . S REJS(REJID)=""  E D  . S IDX=999  . F S IDX=$O(^PSRX(RX,"REJ",IDX),-1) Q:'IDX D  . . I +$P($G(^PSRX(RX,"REJ",IDX,0)),"^",4)'=RFL Q  . . I '$G(OKCL),$P($G(^PSRX(RX,"REJ",IDX,0)),"^",5) Q  . . S REJS(IDX)=""  I '$D(REJS) Q  ;  S IDX=0  F S IDX=$O(REJS(IDX)) Q:'IDX D  . N SKIP  . K ARRAY D GETS^DIQ(52.25,IDX\_","\_RX\_",","\*","","ARRAY")  . K REJFLD M REJFLD=ARRAY(52.25,IDX\_","\_RX\_",")  . ;  . ; check CODE and RRRFLG to see if we want this reject data  . S SKIP=0 ; default is to include it  . I $G(CODE)'="",REJFLD(.01)'=CODE S SKIP=1 ; CODE exists and doesn't match this reject  . I SKIP,$G(RRRFLG),$G(REJFLD(30))="YES" S SKIP=0 ; but include these if RRRFLG is true and this is an RRR reject  . I $G(CODE)="",$G(RRRFLG),$G(REJFLD(30))'="YES" S SKIP=1 ; want only RRR rejects in this case  . I SKIP Q ; get out if we're skipping this one  . ;  . S REJDATA(IDX,"CODE")=$G(REJFLD(.01))  . S REJDATA(IDX,"DATE/TIME")=$G(REJFLD(1))  . S REJDATA(IDX,"PAYER MESSAGE")=$G(REJFLD(2))  . S REJDATA(IDX,"REASON")=$G(REJFLD(3))  . S REJDATA(IDX,"PHARMACIST")=$G(REJFLD(4))  . S REJDATA(IDX,"INSURANCE NAME")=$G(REJFLD(20))  . S REJDATA(IDX,"INSURANCE POINTER")=$G(REJFLD(33)) ;PSO\*427  . S REJDATA(IDX,"COB")=$G(REJFLD(27))  . S REJDATA(IDX,"GROUP NAME")=$G(REJFLD(6))  . S REJDATA(IDX,"GROUP NUMBER")=$G(REJFLD(21))  . S REJDATA(IDX,"BIN")=$G(REJFLD(29))  . S REJDATA(IDX,"CARDHOLDER ID")=$G(REJFLD(22))  . S REJDATA(IDX,"PLAN CONTACT")=$G(REJFLD(7))  . S REJDATA(IDX,"PLAN PREVIOUS FILL DATE")=$G(REJFLD(8))  . S REJDATA(IDX,"STATUS")=$G(REJFLD(9))  . S REJDATA(IDX,"OTHER REJECTS")=$G(REJFLD(17))  . S REJDATA(IDX,"DUR TEXT")=$G(REJFLD(18))  . S REJDATA(IDX,"DUR ADD MSG TEXT")=$G(REJFLD(28))  . S REJDATA(IDX,"REASON SVC CODE")=$G(REJFLD(14))  . S REJDATA(IDX,"RESPONSE IEN")=$G(REJFLD(16))  . S REJDATA(IDX,"RRR FLAG")=$G(REJFLD(30)) ;PSO\*421  . S REJDATA(IDX,"RRR THRESHOLD AMT")=$G(REJFLD(31)) ;PSO\*421  . S REJDATA(IDX,"RRR GROSS AMT DUE")=$G(REJFLD(32)) ;PSO\*421  . I '$G(OKCL) Q  . S REJDATA(IDX,"CLOSED DATE/TIME")=$G(REJFLD(10))  . S REJDATA(IDX,"CLOSED BY")=$G(REJFLD(11))  . S REJDATA(IDX,"CLOSE REASON")=$G(REJFLD(12))  . S REJDATA(IDX,"CLOSE COMMENTS")=$G(REJFLD(13))  . S REJDATA(IDX,"COD1")=$G(REJFLD(14))  . S REJDATA(IDX,"COD2")=$G(REJFLD(15))  . S REJDATA(IDX,"COD3")=$G(REJFLD(19))  . S REJDATA(IDX,"CLA CODE")=$G(REJFLD(24))  . S REJDATA(IDX,"PRIOR AUTH TYPE")=$G(REJFLD(25))  . S REJDATA(IDX,"PRIOR AUTH NUMBER")=$G(REJFLD(26))  . S COM=0 F S COM=$O(^PSRX(RX,"REJ",IDX,"COM",COM)) Q:'COM D  . . S Z=^PSRX(RX,"REJ",IDX,"COM",COM,0)  . . S REJDATA(IDX,"COMMENTS",COM,"DATE/TIME")=$P(Z,"^")  . . S REJDATA(IDX,"COMMENTS",COM,"USER")=$P(Z,"^",2)  . . S REJDATA(IDX,"COMMENTS",COM,"COMMENTS")=$P(Z,"^",3)  Q | | | | | | | | |
| Modified Logic (Changes are highlighted) | | | | | | | | |
| GET(RX,RFL,REJDATA,REJID,OKCL,CODE,RRRFLG) ; get reject data from subfile 52.25  ; Input: (r) RX - Rx IEN (#52)  ; (o) RFL - Refill # (Default: most recent)  ; (r) REJDATA(REJECT IEN,FIELD) - Array where these Reject fields will be returned:  ; "BIN" - Payer BIN number  ; “PCN” – Processor Control Number  ; "CODE" - Reject Code (79 or 88)  ; "DATE/TIME" - DATE/TIME Reject was detected  ; "PAYER MESSAGE" - Message returned by the payer  ; "REASON" - Reject Reason description (from payer)  ; "INSURANCE NAME" - Patient's Insurance Company Name  ; "INSURANCE POINTER" - Patient Insurance Company Pointer  ; "COB" - Coordination of Benefits  ; "GROUP NAME" - Patient's Insurance Group Name  ; "GROUP NUMBER" - Patient's Insurance Group Number  ; "CARDHOLDER ID" - Patient's Insurance Cardholder ID  ; "PLAN CONTACT" - Plan's Contact (eg., "1-800-...")  ; "PLAN PREVIOUS FILL DATE" - Last time Rx was paid by payer  ; "STATUS" - REJECTS status ("OPEN/UNRESOLVED" or "CLOSED/RESOLVED")  ; "DUR TEXT" - Payer's DUR description  ; "DUR ADD MSG TEXT" - Payer's DUR additional description  ; "OTHER REJECTS" - Other Rejects on the same response  ; "REASON SVC CODE" - Reason for Service Code  ; If REJECT is closed, the following fields will be returned:  ; "CLA CODE" - Clarification Code submitted  ; "PRIOR AUTH TYPE" - Prior Authorization Type  ; "PRIOR AUTH NUMBER" - Prior Authorization Type  ; "CLOSED DATE/TIME" - DATE/TIME Reject was closed  ; "CLOSED BY" - Name of the user responsible for closing Reject  ; "CLOSE REASON" - Reason for closing Reject (text)  ; "CLOSE COMMENTS" - User entered comments at close  ; (o) REJID - REJECT IEN in the PRESCRIPTION file for retrieve this REJECT  ; (o) OKCL - If set to 1, CLOSED REJECTs will also be returned  ; (o) CODE - Only REJECTs with this CODE should be returned  ; (o) RRRFLG - If set to 1 with CODE present, also return Reject Resolution Required REJECTs  ; If set to 1 and CODE not passed, then only return RRR REJECTs  ;  N REJS,ARRAY,REJFLD,IDX,COM,Z  ;  I '$D(RFL) S RFL=$$LSTRFL^PSOBPSU1(RX)  ;  K REJDATA  I '$O(^PSRX(RX,"REJ",0)) Q  ;  K REJS S RFL=+$G(RFL)  I $G(REJID) D  . I +$P($G(^PSRX(RX,"REJ",REJID,0)),"^",4)'=RFL Q  . I '$G(OKCL),$P($G(^PSRX(RX,"REJ",REJID,0)),"^",5) Q  . S REJS(REJID)=""  E D  . S IDX=999  . F S IDX=$O(^PSRX(RX,"REJ",IDX),-1) Q:'IDX D  . . I +$P($G(^PSRX(RX,"REJ",IDX,0)),"^",4)'=RFL Q  . . I '$G(OKCL),$P($G(^PSRX(RX,"REJ",IDX,0)),"^",5) Q  . . S REJS(IDX)=""  I '$D(REJS) Q  ;  S IDX=0  F S IDX=$O(REJS(IDX)) Q:'IDX D  . N SKIP  . K ARRAY D GETS^DIQ(52.25,IDX\_","\_RX\_",","\*","","ARRAY")  . K REJFLD M REJFLD=ARRAY(52.25,IDX\_","\_RX\_",")  . ;  . ; check CODE and RRRFLG to see if we want this reject data  . S SKIP=0 ; default is to include it  . I $G(CODE)'="",REJFLD(.01)'=CODE S SKIP=1 ; CODE exists and doesn't match this reject  . I SKIP,$G(RRRFLG),$G(REJFLD(30))="YES" S SKIP=0 ; but include these if RRRFLG is true and this is an RRR reject  . I $G(CODE)="",$G(RRRFLG),$G(REJFLD(30))'="YES" S SKIP=1 ; want only RRR rejects in this case  . I SKIP Q ; get out if we're skipping this one  . ;  . S REJDATA(IDX,"CODE")=$G(REJFLD(.01))  . S REJDATA(IDX,"DATE/TIME")=$G(REJFLD(1))  . S REJDATA(IDX,"PAYER MESSAGE")=$G(REJFLD(2))  . S REJDATA(IDX,"REASON")=$G(REJFLD(3))  . S REJDATA(IDX,"PHARMACIST")=$G(REJFLD(4))  . S REJDATA(IDX,"INSURANCE NAME")=$G(REJFLD(20))  . S REJDATA(IDX,"INSURANCE POINTER")=$G(REJFLD(33)) ;PSO\*427  . S REJDATA(IDX,"COB")=$G(REJFLD(27))  . S REJDATA(IDX,"GROUP NAME")=$G(REJFLD(6))  . S REJDATA(IDX,"GROUP NUMBER")=$G(REJFLD(21))  . S REJDATA(IDX,"BIN")=$G(REJFLD(29))  . S REJDATA(IDX,”PCN”)=$G(REJFLD(34))  . S REJDATA(IDX,"CARDHOLDER ID")=$G(REJFLD(22))  . S REJDATA(IDX,"PLAN CONTACT")=$G(REJFLD(7))  . S REJDATA(IDX,"PLAN PREVIOUS FILL DATE")=$G(REJFLD(8))  . S REJDATA(IDX,"STATUS")=$G(REJFLD(9))  . S REJDATA(IDX,"OTHER REJECTS")=$G(REJFLD(17))  . S REJDATA(IDX,"DUR TEXT")=$G(REJFLD(18))  . S REJDATA(IDX,"DUR ADD MSG TEXT")=$G(REJFLD(28))  . S REJDATA(IDX,"REASON SVC CODE")=$G(REJFLD(14))  . S REJDATA(IDX,"RESPONSE IEN")=$G(REJFLD(16))  . S REJDATA(IDX,"RRR FLAG")=$G(REJFLD(30)) ;PSO\*421  . S REJDATA(IDX,"RRR THRESHOLD AMT")=$G(REJFLD(31)) ;PSO\*421  . S REJDATA(IDX,"RRR GROSS AMT DUE")=$G(REJFLD(32)) ;PSO\*421  . I '$G(OKCL) Q  . S REJDATA(IDX,"CLOSED DATE/TIME")=$G(REJFLD(10))  . S REJDATA(IDX,"CLOSED BY")=$G(REJFLD(11))  . S REJDATA(IDX,"CLOSE REASON")=$G(REJFLD(12))  . S REJDATA(IDX,"CLOSE COMMENTS")=$G(REJFLD(13))  . S REJDATA(IDX,"COD1")=$G(REJFLD(14))  . S REJDATA(IDX,"COD2")=$G(REJFLD(15))  . S REJDATA(IDX,"COD3")=$G(REJFLD(19))  . S REJDATA(IDX,"CLA CODE")=$G(REJFLD(24))  . S REJDATA(IDX,"PRIOR AUTH TYPE")=$G(REJFLD(25))  . S REJDATA(IDX,"PRIOR AUTH NUMBER")=$G(REJFLD(26))  . S COM=0 F S COM=$O(^PSRX(RX,"REJ",IDX,"COM",COM)) Q:'COM D  . . S Z=^PSRX(RX,"REJ",IDX,"COM",COM,0)  . . S REJDATA(IDX,"COMMENTS",COM,"DATE/TIME")=$P(Z,"^")  . . S REJDATA(IDX,"COMMENTS",COM,"USER")=$P(Z,"^",2)  . . S REJDATA(IDX,"COMMENTS",COM,"COMMENTS")=$P(Z,"^",3)  Q | | | | | | | | |

| Subroutine Name | **INS^PSOREJP1** | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify | | | Delete | | No Change | |
| Story | US583 | | | | | | | |
| Related Menu Options or ListMan Actions | Third Party Payer Rejects – Worklist [PSO REJECTS WORKLIST]  Third Party Payer Rejects – View/Process [PSO REJECTS VIEW/PROCESS] | | | | | | | |
| Related Routines | This Subroutine is Called By | | | This Subroutine Calls | | | | |
| INIT^PSOREJP1 | | | SETLN^PSOREJP1 | | | | |
| Related Integration Control Registrations (ICRs) | N/A | | | | | | | |
| Data Passing | Parameter Input | | Parameter Output | | | Function Return Value | | Global Modified |
| Input Attribute Name and Definition | Name:  Definition:  New  Modify  Delete  No Change | | | | | | | |
| Output Attribute Name and Definition | Name:  Definition:  New  Modify  Delete  No Change | | | | | | | |
| Current Logic | | | | | | | | |
| INS; - Insurance Information  D SETLN()  D SETLN("INSURANCE Information",1,1)  N PSOINS,PSOINS1,I  S PSOINS=$G(DATA(REJ,"INSURANCE NAME"))  F I=1:1:(50-($L(PSOINS)+18)) S PSOINS=PSOINS\_" "  S PSOINS1=$G(DATA(REJ,"COB"))  I PSOINS1="SECONDARY" S PSOINS=PSOINS\_"Coord. Of Benefits: "\_PSOINS1  D SETLN("Insurance : "\_PSOINS,,,18)  D SETLN("Contact : "\_$G(DATA(REJ,"PLAN CONTACT")),,,18)  D SETLN("BIN : "\_$G(DATA(REJ,"BIN")),,,18)  D SETLN("Group Number : "\_$G(DATA(REJ,"GROUP NUMBER")),,,18)  D SETLN("Cardholder ID : "\_$G(DATA(REJ,"CARDHOLDER ID")),,1,18)  Q | | | | | | | | |
| Modified Logic (Changes are highlighted) | | | | | | | | |
| INS; - Insurance Information  D SETLN()  D SETLN("INSURANCE Information",1,1)  N PSOINS,PSOINS1,I,PSOBINPCN  S PSOINS=$G(DATA(REJ,"INSURANCE NAME"))  F I=1:1:(50-($L(PSOINS)+18)) S PSOINS=PSOINS\_" "  S PSOINS1=$G(DATA(REJ,"COB"))  I PSOINS1="SECONDARY" S PSOINS=PSOINS\_"Coord. Of Benefits: "\_PSOINS1  D SETLN("Insurance : "\_PSOINS,,,18)  D SETLN("Contact : "\_$G(DATA(REJ,"PLAN CONTACT")),,,18)  S PSOBINPCN=$G(DATA(REJ,”BIN”))\_”/ “\_$G(DATA(REJ,”PCN”))  D SETLN("BIN/ PCN : "\_PSOBINPCN,,,18)  D SETLN("Group Number : "\_$G(DATA(REJ,"GROUP NUMBER")),,,18)  D SETLN("Cardholder ID : "\_$G(DATA(REJ,"CARDHOLDER ID")),,1,18)  Q | | | | | | | | |

| Subroutine Name | **POST^PSO7P478** | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify | | | Delete | | No Change | |
| Story | US583 | | | | | | | |
| Related Menu Options or ListMan Actions | N/A | | | | | | | |
| Related Routines | This Subroutine is Called By | | | This Subroutine Calls | | | | |
| KIDS system; Kernel Installation and Distribution System; called upon patch installation | | | BMES^XPDUTL  PCN^PSO7P478 | | | | |
| Related Integration Control Registrations (ICRs) | N/A | | | | | | | |
| Data Passing | Parameter Input | | Parameter Output | | | Function Return Value | | Global Modified |
| Input Attribute Name and Definition | Name: N/A  Definition:  New  Modify  Delete  No Change | | | | | | | |
| Output Attribute Name and Definition | Name: N/A  Definition:  New  Modify  Delete  No Change | | | | | | | |
| Current Logic | | | | | | | | |
| N/A – new routine for patch; post install. | | | | | | | | |
| Modified Logic (Changes are highlighted) | | | | | | | | |
| POST ;post-install functions are coded here.  D BMES^XPDUTL(" Starting post-install of PSO\*7\*478")  D PCN  D BMES^XPDUTL(" Finished post-install of PSO\*7\*478")  Q | | | | | | | | |

| Subroutine Name | **PCN^PSO7P478** | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify | | | Delete | | No Change | |
| Story | US583 | | | | | | | |
| Related Menu Options or ListMan Actions | N/A | | | | | | | |
| Related Routines | This Subroutine is Called By | | | This Subroutine Calls | | | | |
| POST^PSO7P478 | | | ^DIE  DURRESP^BPSNCPD3  BMES^XPDUTL  MES^XPDUTL | | | | |
| Data Dictionary (DD) References | File #52.25 – PRESCRIPTION, REJECT INFO sub-file.  Some existing entries are being updated in this sub-file. | | | | | | | |
| Related Integration Control Registrations (ICRs) | Call to DURRESP^BPSNCPD3 supported by IA #4560. | | | | | | | |
| Data Passing | Parameter Input | | Parameter Output | | | Function Return Value | | Global Modified |
| Input Attribute Name and Definition | Name: N/A  Definition:  New  Modify  Delete  No Change | | | | | | | |
| Output Attribute Name and Definition | Name: N/A  Definition:  New  Modify  Delete  No Change | | | | | | | |
| Current Logic | | | | | | | | |
| N/A – new routine for patch; post install. | | | | | | | | |
| Modified Logic (Changes are highlighted) | | | | | | | | |
| PCN ;Update PCN on PRESCRIPTION reject multiple  ;  ; Reference to BPSNCPD3 supported by IA 4560  ;  N CNT,COB,DAT,DUR,RX,RN,RSPIEN,DA,DR,DIE  D BMES^XPDUTL(" Updating PCN Numbers")  S CNT=0  S DAT=0 F S DAT=$O(^PSRX("REJDAT",DAT)) Q:'DAT D  . S RX="" F S RX=$O(^PSRX("REJDAT",DAT,RX)) Q:'RX D  .. S RN="" F S RN=$O(^PSRX("REJDAT",DAT,RX,RN)) Q:'RN D  ... I $L($P($G(^PSRX(RX,"REJ",RN,2)),"^",10))=10 Q  ... S RSPIEN=$P($G(^PSRX(RX,"REJ",RN,0)),"^",11) I 'RSPIEN Q  ... S COB=$P($G(^PSRX(RX,"REJ",RN,2)),"^",7) I COB="" S COB=1  ... K DUR D DURRESP^BPSNCPD3(RSPIEN,.DUR,COB) ; ICR# 4560  ... I 'DUR(COB,"PCN") Q  ... S DIE="^PSRX("\_RX\_",""REJ"",",DA(1)=RX,DA=RN,DR=34\_"////"\_DUR(COB,"PCN")  ... D ^DIE K DA,DR,DIE  ... S CNT=CNT+1  D MES^XPDUTL(" - "\_CNT\_" entries updated")  D MES^XPDUTL(" - Done with updating PCN Numbers")  Q | | | | | | | | |

| Subroutine Name | **DURRESP^BPSNCPD3** | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify | | | Delete | | No Change | |
| Story | US583 | | | | | | | |
| Related Menu Options or ListMan Actions | Third Party Payer Rejects – Worklist [PSO REJECTS WORKLIST]  Third Party Payer Rejects – View/Process [PSO REJECTS VIEW/PROCESS] | | | | | | | |
| Related Routines | This Subroutine is Called By | | | This Subroutine Calls | | | | |
| DUR1^BPSNCPD3  SMA^PSOREJP1  ISDUR^PSOREJP5  INIT^PSOREJP5 | | | ADDMESS^BPSSCRLG  $$GET1^DIQ  GETS^DIQ | | | | |
| Related Integration Control Registrations (ICRs) | IA #4560 | | | | | | | |
| Data Passing | Parameter Input | | Parameter Output | | | Function Return Value | | Global Modified |
| Input Attribute Name and Definition | Name: DURIEN  Definition: Claim Response IEN. Ptr to BPS RESPONSES (#9002313.03)  New  Modify  Delete  No Change  Name: BPRXCOB  Definition: Payer Sequence  New  Modify  Delete  No Change | | | | | | | |
| Output Attribute Name and Definition | Name: DUR  Definition: Array of DUR related information for a specific claim response in the BPS RESPONSES file.  New  Modify  Delete  No Change | | | | | | | |
| Current Logic | | | | | | | | |
| DURRESP(DURIEN,DUR,BPRXCOB) ;  ;Input Variables:  ; DURIEN - Claim Response IEN. Pointer to the BPS RESPONSES File #9002313.03  ; BPRXCOB - (Optional) The Payer Sequence:  ; 1 - Primary (default)  ; 2 - Secondary  ;  ;Output Variables:  ; DUR - Array of DUR related information for a specific claim response in the  ; BPS RESPONSES file in the following format (INSN is the Payer Sequence):  ;  ; DUR(INSN,"RESPONSE IEN") - Pointer to the RESPONSE file (#9002313.03) for  ; the claim submission  ; DUR(INSN,"MESSAGE") - The Transmission level specific data, Message field 504  ; DUR(INSN,"PAYER MESSAGE") - Message returned from the payer in the Transaction  ; level  ; DUR(INSN,"STATUS") - Status of the claim (i.e. REJECTED CLAIM, PAYABLE)  ;  ; The following four fields are redundant with the fields in the DUR PPS  ; array but are provided for backwards compatibility.  ; DUR(INSN,"REASON") - Reason for Service Code pointer to BPS NCPDP REASON FOR  ; SERVICE CODE file (#9002313.23)  ; DUR(INSN,"PREV FILL DATE") - Plan's Previous Fill Date  ; DUR(INSN,"DUR FREE TEXT DESC") - Drug Utilization Review (DUR) description  ; and/or claims rejection free text information from the payer  ; DUR(INSN,"DUR ADD MSG TEXT") - Drug Utilization Review (DUR) additional free  ; text information from the payer  ;  ; The following fields are from the DUR PPS RESPONSE multiple.  ; DUR(INSN,"DUR PPS",SEQ,"DUR PPS RESPONSE") - Total number of DUR PPS  ; responses from the payer  ; DUR(INSN,"DUR PPS",SEQ,"REASON FOR SERVICE CODE") - Code identifying the  ; type of utilization conflict detected or the reason for the pharmacist  ; professional service  ; DUR(INSN,"DUR PPS",SEQ,"CLINICAL SIGNIFICANCE CODE") - Code identifying  ; the significance or severity level of a clinical event as contained  ; in the originating data base  ; DUR(INSN,"DUR PPS",SEQ,"OTHER PHARMACY INDICATOR") - Code for the type of  ; pharmacy dispensing the conflicting drug  ; DUR(INSN,"DUR PPS",SEQ,"PREVIOUS DATE OF FILL") - Date prescription was  ; previously filled  ; DUR(INSN,"DUR PPS",SEQ,"QUANTITY OF PREVIOUS FILL") - Amount expressed in  ; metric decimal units of the conflicting agent that was previously filled  ; DUR(INSN,"DUR PPS",SEQ,"DATABASE INDICATOR") - Code identifying the source  ; of drug information used for DUR processing  ; DUR(INSN,"DUR PPS",SEQ,"OTHER PRESCRIBER INDICATOR") - Code comparing the  ; prescriber of the current prescription to the prescriber of the previously  ; filled conflicting prescription  ; DUR(INSN,"DUR PPS",SEQ,"DUR FREE TEXT MESSAGE") - Text that provides  ; additional detail regarding a DUR conflict  ; DUR(INSN,"DUR PPS",SEQ,"DUR ADDITIONAL TEXT") - Descriptive information that  ; further defines the referenced DUR alert  ; DUR(INSN,"REJ CODE LST") - List of rejection code(s) returned by the payer  ; separated by commas (i.e. 79,14)  ; DUR(INSN,"REJ CODES",SEQ,REJ CODE) - Array of rejection code descriptions  ; where REJ CODE correlates to DUR(INSN,"REJ CODE LST") value(s) and SEQ  ; equals a sequential number  ;  I '$G(DURIEN) Q  S BPRXCOB=+$G(BPRXCOB)  I BPRXCOB=0 S BPRXCOB=1 ;default is Primary  N ADDMESS,I,DUR1,CLMIEN  S DUR(BPRXCOB,"RESPONSE IEN")=DURIEN  ;  ;Get BIN from claim  S CLMIEN=$$GET1^DIQ(9002313.03,DURIEN,.01,"I")  S DUR(BPRXCOB,"BIN")=$$GET1^DIQ(9002313.02,CLMIEN\_",",101) ; BIN Number  ;  ; Get the Transmission specific data (Message)  S DUR(BPRXCOB,"MESSAGE")=$$GET1^DIQ(9002313.03,DURIEN\_",",504,"E")  ;  ; Get the Additional Message Information from the transaction  D ADDMESS^BPSSCRLG(DURIEN,1,.ADDMESS)  M DUR(BPRXCOB,"PAYER MESSAGE")=ADDMESS  ;  ; Get the other transaction level data  D GETS^DIQ(9002313.0301,"1,"\_DURIEN\_",","501;567.01\*","E","DUR1","ERROR")  S DUR(BPRXCOB,"STATUS")=$G(DUR1(9002313.0301,"1,"\_DURIEN\_",",501,"E"))  ;Status of Response  ;  ; The following four fields are redundant with the fields in the DUR PPS    ; multiple but are needed for backwards compatibility with the OP code  S DUR(BPRXCOB,"REASON")=$G(DUR1(9002313.1101,"1,1,"\_DURIEN\_",",439,"E")) ;Reason for Service Code  S DUR(BPRXCOB,"PREV FILL DATE")=$G(DUR1(9002313.1101,"1,1,"\_DURIEN\_",",530,"E")) ;Previous Date of Fill  S DUR(BPRXCOB,"DUR FREE TEXT DESC")=$G(DUR1(9002313.1101,"1,1,"\_DURIEN\_",",544,"E")) ;DUR Free Text Message from Payer  S DUR(BPRXCOB,"DUR ADD MSG TEXT")=$G(DUR1(9002313.1101,"1,1,"\_DURIEN\_",",570,"E")) ;DUR Additional Message Text from Payer  ;  ; Get DUR PPS RESPONSE multiple values  S DUR(BPRXCOB,"DUR PPS RESPONSE")=""  F I=1:1 Q:'$D(DUR1(9002313.1101,I\_",1,"\_DURIEN\_",",.01)) D  . S DUR(BPRXCOB,"DUR PPS RESPONSE")=I  . S DUR(BPRXCOB,"DUR PPS",I,"DUR PPS RESPONSE")=$G(DUR1(9002313.1101,I\_",1,"\_DURIEN\_",",.01,"E"))  . S DUR(BPRXCOB,"DUR PPS",I,"REASON FOR SERVICE CODE")=$G(DUR1(9002313.1101,I\_",1,"\_DURIEN\_",",439,"E"))  . S DUR(BPRXCOB,"DUR PPS",I,"CLINICAL SIGNIFICANCE CODE")=$G(DUR1(9002313.1101,I\_",1,"\_DURIEN\_",",528,"E"))  . S DUR(BPRXCOB,"DUR PPS",I,"OTHER PHARMACY INDICATOR")=$G(DUR1(9002313.1101,I\_",1,"\_DURIEN\_",",529,"E"))  . S DUR(BPRXCOB,"DUR PPS",I,"PREVIOUS DATE OF FILL")=$G(DUR1(9002313.1101,I\_",1,"\_DURIEN\_",",530,"E"))  . S DUR(BPRXCOB,"DUR PPS",I,"QUANTITY OF PREVIOUS FILL")=$G(DUR1(9002313.1101,I\_",1,"\_DURIEN\_",",531,"E"))  . S DUR(BPRXCOB,"DUR PPS",I,"DATABASE INDICATOR")=$G(DUR1(9002313.1101,I\_",1,"\_DURIEN\_",",532,"E"))  . S DUR(BPRXCOB,"DUR PPS",I,"OTHER PRESCRIBER INDICATOR")=$G(DUR1(9002313.1101,I\_",1,"\_DURIEN\_",",533,"E"))  . S DUR(BPRXCOB,"DUR PPS",I,"DUR FREE TEXT MESSAGE")=$G(DUR1(9002313.1101,I\_",1,"\_DURIEN\_",",544,"E"))  . S DUR(BPRXCOB,"DUR PPS",I,"DUR ADDITIONAL TEXT")=$G(DUR1(9002313.1101,I\_",1,"\_DURIEN\_",",570,"E"))  ;  ; Get DUR reject codes and description and store in DUR  D GETS^DIQ(9002313.0301,"1,"\_DURIEN\_",","511\*","I","DUR1","ERROR") ;get DUR codes and descriptions  S DUR(BPRXCOB,"REJ CODE LST")=""  F I=1:1 Q:'$D(DUR1(9002313.03511,I\_",1,"\_DURIEN\_",")) D  . N REJX,REJN  . S REJX=$G(DUR1(9002313.03511,I\_",1,"\_DURIEN\_",",.01,"I")) Q:REJX="" ; external reject code  . S REJN=+$O(^BPSF(9002313.93,"B",REJX,0)) Q:'REJN ; internal reject code ien  . S DUR(BPRXCOB,"REJ CODES",I,REJX)=$P($G(^BPSF(9002313.93,REJN,0)),U,2) ; reject code description  . S DUR(BPRXCOB,"REJ CODE LST")=DUR(BPRXCOB,"REJ CODE LST")\_","\_REJX  S DUR(BPRXCOB,"REJ CODE LST")=$E(DUR(BPRXCOB,"REJ CODE LST"),2,9999)  Q | | | | | | | | |
| Modified Logic (Changes are highlighted) | | | | | | | | |
| DURRESP(DURIEN,DUR,BPRXCOB) ;  ;Input Variables:  ; DURIEN - Claim Response IEN. Pointer to the BPS RESPONSES File #9002313.03  ; BPRXCOB - (Optional) The Payer Sequence:  ; 1 - Primary (default)  ; 2 - Secondary  ;  ;Output Variables:  ; DUR - Array of DUR related information for a specific claim response in the  ; BPS RESPONSES file in the following format (INSN is the Payer Sequence):  ;  ; DUR(INSN,"RESPONSE IEN") - Pointer to the RESPONSE file (#9002313.03) for  ; the claim submission  ; DUR(INSN,”PCN”) – Processor Control Number  ; DUR(INSN,"MESSAGE") - The Transmission level specific data, Message field 504  ; DUR(INSN,"PAYER MESSAGE") - Message returned from the payer in the Transaction  ; level  ; DUR(INSN,"STATUS") - Status of the claim (i.e. REJECTED CLAIM, PAYABLE)  ;  ; The following four fields are redundant with the fields in the DUR PPS  ; array but are provided for backwards compatibility.  ; DUR(INSN,"REASON") - Reason for Service Code pointer to BPS NCPDP REASON FOR  ; SERVICE CODE file (#9002313.23)  ; DUR(INSN,"PREV FILL DATE") - Plan's Previous Fill Date  ; DUR(INSN,"DUR FREE TEXT DESC") - Drug Utilization Review (DUR) description  ; and/or claims rejection free text information from the payer  ; DUR(INSN,"DUR ADD MSG TEXT") - Drug Utilization Review (DUR) additional free  ; text information from the payer  ;  ; The following fields are from the DUR PPS RESPONSE multiple.  ; DUR(INSN,"DUR PPS",SEQ,"DUR PPS RESPONSE") - Total number of DUR PPS  ; responses from the payer  ; DUR(INSN,"DUR PPS",SEQ,"REASON FOR SERVICE CODE") - Code identifying the  ; type of utilization conflict detected or the reason for the pharmacist  ; professional service  ; DUR(INSN,"DUR PPS",SEQ,"CLINICAL SIGNIFICANCE CODE") - Code identifying  ; the significance or severity level of a clinical event as contained  ; in the originating data base  ; DUR(INSN,"DUR PPS",SEQ,"OTHER PHARMACY INDICATOR") - Code for the type of  ; pharmacy dispensing the conflicting drug  ; DUR(INSN,"DUR PPS",SEQ,"PREVIOUS DATE OF FILL") - Date prescription was  ; previously filled  ; DUR(INSN,"DUR PPS",SEQ,"QUANTITY OF PREVIOUS FILL") - Amount expressed in  ; metric decimal units of the conflicting agent that was previously filled  ; DUR(INSN,"DUR PPS",SEQ,"DATABASE INDICATOR") - Code identifying the source  ; of drug information used for DUR processing  ; DUR(INSN,"DUR PPS",SEQ,"OTHER PRESCRIBER INDICATOR") - Code comparing the  ; prescriber of the current prescription to the prescriber of the previously  ; filled conflicting prescription  ; DUR(INSN,"DUR PPS",SEQ,"DUR FREE TEXT MESSAGE") - Text that provides  ; additional detail regarding a DUR conflict  ; DUR(INSN,"DUR PPS",SEQ,"DUR ADDITIONAL TEXT") - Descriptive information that  ; further defines the referenced DUR alert  ; DUR(INSN,"REJ CODE LST") - List of rejection code(s) returned by the payer  ; separated by commas (i.e. 79,14)  ; DUR(INSN,"REJ CODES",SEQ,REJ CODE) - Array of rejection code descriptions  ; where REJ CODE correlates to DUR(INSN,"REJ CODE LST") value(s) and SEQ  ; equals a sequential number  ;  I '$G(DURIEN) Q  S BPRXCOB=+$G(BPRXCOB)  I BPRXCOB=0 S BPRXCOB=1 ;default is Primary  N ADDMESS,I,DUR1,CLMIEN  S DUR(BPRXCOB,"RESPONSE IEN")=DURIEN  ;  ;Get BIN from claim  S CLMIEN=$$GET1^DIQ(9002313.03,DURIEN,.01,"I")  S DUR(BPRXCOB,"BIN")=$$GET1^DIQ(9002313.02,CLMIEN\_",",101) ; BIN Number  ;  ; Get PCN from claim  S DUR(BPRXCOB,”PCN”)=$$GET1^DIQ(9002313.02,CLMIEN\_”,”,104) ; PCN Number  ;  ; Get the Transmission specific data (Message)  S DUR(BPRXCOB,"MESSAGE")=$$GET1^DIQ(9002313.03,DURIEN\_",",504,"E")  ;  ; Get the Additional Message Information from the transaction  D ADDMESS^BPSSCRLG(DURIEN,1,.ADDMESS)  M DUR(BPRXCOB,"PAYER MESSAGE")=ADDMESS  ;  ; Get the other transaction level data  D GETS^DIQ(9002313.0301,"1,"\_DURIEN\_",","501;567.01\*","E","DUR1","ERROR")  S DUR(BPRXCOB,"STATUS")=$G(DUR1(9002313.0301,"1,"\_DURIEN\_",",501,"E"))  ;Status of Response  ;  ; The following four fields are redundant with the fields in the DUR PPS    ; multiple but are needed for backwards compatibility with the OP code  S DUR(BPRXCOB,"REASON")=$G(DUR1(9002313.1101,"1,1,"\_DURIEN\_",",439,"E")) ;Reason for Service Code  S DUR(BPRXCOB,"PREV FILL DATE")=$G(DUR1(9002313.1101,"1,1,"\_DURIEN\_",",530,"E")) ;Previous Date of Fill  S DUR(BPRXCOB,"DUR FREE TEXT DESC")=$G(DUR1(9002313.1101,"1,1,"\_DURIEN\_",",544,"E")) ;DUR Free Text Message from Payer  S DUR(BPRXCOB,"DUR ADD MSG TEXT")=$G(DUR1(9002313.1101,"1,1,"\_DURIEN\_",",570,"E")) ;DUR Additional Message Text from Payer  ;  ; Get DUR PPS RESPONSE multiple values  S DUR(BPRXCOB,"DUR PPS RESPONSE")=""  F I=1:1 Q:'$D(DUR1(9002313.1101,I\_",1,"\_DURIEN\_",",.01)) D  . S DUR(BPRXCOB,"DUR PPS RESPONSE")=I  . S DUR(BPRXCOB,"DUR PPS",I,"DUR PPS RESPONSE")=$G(DUR1(9002313.1101,I\_",1,"\_DURIEN\_",",.01,"E"))  . S DUR(BPRXCOB,"DUR PPS",I,"REASON FOR SERVICE CODE")=$G(DUR1(9002313.1101,I\_",1,"\_DURIEN\_",",439,"E"))  . S DUR(BPRXCOB,"DUR PPS",I,"CLINICAL SIGNIFICANCE CODE")=$G(DUR1(9002313.1101,I\_",1,"\_DURIEN\_",",528,"E"))  . S DUR(BPRXCOB,"DUR PPS",I,"OTHER PHARMACY INDICATOR")=$G(DUR1(9002313.1101,I\_",1,"\_DURIEN\_",",529,"E"))  . S DUR(BPRXCOB,"DUR PPS",I,"PREVIOUS DATE OF FILL")=$G(DUR1(9002313.1101,I\_",1,"\_DURIEN\_",",530,"E"))  . S DUR(BPRXCOB,"DUR PPS",I,"QUANTITY OF PREVIOUS FILL")=$G(DUR1(9002313.1101,I\_",1,"\_DURIEN\_",",531,"E"))  . S DUR(BPRXCOB,"DUR PPS",I,"DATABASE INDICATOR")=$G(DUR1(9002313.1101,I\_",1,"\_DURIEN\_",",532,"E"))  . S DUR(BPRXCOB,"DUR PPS",I,"OTHER PRESCRIBER INDICATOR")=$G(DUR1(9002313.1101,I\_",1,"\_DURIEN\_",",533,"E"))  . S DUR(BPRXCOB,"DUR PPS",I,"DUR FREE TEXT MESSAGE")=$G(DUR1(9002313.1101,I\_",1,"\_DURIEN\_",",544,"E"))  . S DUR(BPRXCOB,"DUR PPS",I,"DUR ADDITIONAL TEXT")=$G(DUR1(9002313.1101,I\_",1,"\_DURIEN\_",",570,"E"))  ;  ; Get DUR reject codes and description and store in DUR  D GETS^DIQ(9002313.0301,"1,"\_DURIEN\_",","511\*","I","DUR1","ERROR") ;get DUR codes and descriptions  S DUR(BPRXCOB,"REJ CODE LST")=""  F I=1:1 Q:'$D(DUR1(9002313.03511,I\_",1,"\_DURIEN\_",")) D  . N REJX,REJN  . S REJX=$G(DUR1(9002313.03511,I\_",1,"\_DURIEN\_",",.01,"I")) Q:REJX="" ; external reject code  . S REJN=+$O(^BPSF(9002313.93,"B",REJX,0)) Q:'REJN ; internal reject code ien  . S DUR(BPRXCOB,"REJ CODES",I,REJX)=$P($G(^BPSF(9002313.93,REJN,0)),U,2) ; reject code description  . S DUR(BPRXCOB,"REJ CODE LST")=DUR(BPRXCOB,"REJ CODE LST")\_","\_REJX  S DUR(BPRXCOB,"REJ CODE LST")=$E(DUR(BPRXCOB,"REJ CODE LST"),2,9999)  Q | | | | | | | | |

| Subroutine Name | **SAVE^PSOREJUT** | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify | | | Delete | | No Change | |
| Story | US583 | | | | | | | |
| Related Menu Options or ListMan Actions | Third Party Payer Rejects – Worklist [PSO REJECTS WORKLIST]  Third Party Payer Rejects – View/Process [PSO REJECTS VIEW/PROCESS] | | | | | | | |
| Related Routines | This Subroutine is Called By | | | This Subroutine Calls | | | | |
| SYNC2^PSOREJUT  OPN^PSOREJP2 | | | NOW^%DTC  $$ADDCOMM^BPSBUTL  FILE^DICN  $$LSTRFL^PSOBPSU1  $$RXSITE^PSOBPSUT  SAVECOM^PSOREJP3  $$EVAL^PSOREJU4 | | | | |
| Related Integration Control Registrations (ICRs) | N/A | | | | | | | |
| Data Passing | Parameter Input | | Parameter Output | | | Function Return Value | | Global Modified |
| Input Attribute Name and Definition | Name: RX  Definition: Rx IEN (#52)  New  Modify  Delete  No Change  Name: RFL  Definition: Refill #  New  Modify  Delete  No Change  Name: REOPEN  Definition: Claim ReOpened Flag  New  Modify  Delete  No Change  Name: REJ  Definition: REJECT Information Array  New  Modify  Delete  No Change | | | | | | | |
| Output Attribute Name and Definition | Name: REJ  Definition: Reject Information Array  New  Modify  Delete  No Change | | | | | | | |
| Current Logic | | | | | | | | |
| SAVE(RX,RFL,REJ,REOPEN) ; - Saves DUR Information in the file 52  ; Input: (r) RX - Rx IEN (#52)  ; (o) RFL - Refill # (Default: most recent)  ; (o) REOPEN - value of 1 means claim being reopened; null or no value passed means reopen claim functionality not being used  ; (r) REJ - Array containing information about the REJECT on the following subscripts:  ; "BIN" - BIN Number  ; "CODE" - Reject Code (79 or 88)  ; "DATE/TIME" - Date/Time Reject Detected  ; "PAYER MESSAGE" - Message returned by Payer (up to 140 chars long)  ; "REASON" - Reject Reason (up to 100 chars long)  ; "DUR TEXT" - Payer's DUR description  ; "DUR ADD MSG TEXT" - Payer's DUR additional message text description  ; "INSURANCE NAME" - Patient's Insurance Company Name  ; "INSURANCE POINTER" - Patient's Insurance Company IEN  ; "GROUP NAME" - Patient's Insurance Group Name  ; "GROUP NUMBER" - Patient's Insurance Group Number  ; "CARDHOLDER ID" - Patient's Insurance Cardholder ID  ; "COB" - Coordination of Benefits  ; "PLAN CONTACT" - Patient's Insurance Plan Contact (1-800)  ; "PREVIOUS FILL" - Plan's Previous Fill Date  ; "OTHER REJECTS" - Other Rejects with same Response  ; "PHARMACIST" - Pharmacist DUZ  ; "RESPONSE IEN" - Pointer to the RESPONSE file in ECME  ; "REASON SVC CODE" - Reason for Service Code (pointer to BPS NCPDP REASON FOR SERVICE CODE)  ; "RE-OPENED" - Re-Open Flag  ; "RRR FLAG" - Reject Resolution Required indicator (expecting 1/0 into SAVE)  ; "RRR THRESHOLD AMT" - Reject Resolution Required Dollar Threshold  ; "RRR GROSS AMT DUE" - Reject Resolution Required Gross Amount Due  ;Output: REJ("REJECT IEN")  N %,DIC,DR,DA,X,DINUM,DD,DO,DLAYGO,ERR  I '$D(RFL) S RFL=$$LSTRFL^PSOBPSU1(RX)  I '$G(PSODIV) S PSODIV=$$RXSITE^PSOBPSUT(RX,RFL)  S REJ("BIN")=$E($G(REJ("BIN")),1,6)  S REJ("CODE")=$G(REJ("CODE"))  ;  ; convert REJ("RRR FLAG") into internal format (1/0) if necessary. When coming into SAVE from the Re-open Reject  ; action, this flag is in the external format (YES/NO). esg - 3/29/16 - PSO\*7\*448  I $G(REJ("RRR FLAG"))="YES" S REJ("RRR FLAG")=1  I $G(REJ("RRR FLAG"))="NO" S REJ("RRR FLAG")=0  ;  ;Ignore this additional Check if reject is Reject Resolution Required reject - PSO\*7\*421  I '$G(REJ("RRR FLAG")),REJ("CODE")'=79&(REJ("CODE")'=88)&('$G(PSOTRIC))&('$G(REOPEN)) S ERR=$$EVAL^PSOREJU4(PSODIV,REJ("CODE"),$G(OPECC)) Q:'+ERR  S REJ("PAYER MESSAGE")=$E($G(REJ("PAYER MESSAGE")),1,140),REJ("REASON")=$E($G(REJ("REASON")),1,100)  S REJ("DUR TEXT")=$E($G(REJ("DUR TEXT")),1,100),REJ("DUR ADD MSG TEXT")=$E($G(REJ("DUR ADD MSG TEXT")),1,100),REJ("GROUP NAME")=$E($G(REJ("GROUP NAME"))  ,1,30)  S REJ("INSURANCE NAME")=$E($G(REJ("INSURANCE NAME")),1,30),REJ("PLAN CONTACT")=$E($G(REJ("PLAN CONTACT")),1,30)  S REJ("GROUP NUMBER")=$E($G(REJ("GROUP NUMBER")),1,30),REJ("OTHER REJECTS")=$E($G(REJ("OTHER REJECTS")),1,15)  S REJ("CARDHOLDER ID")=$E($G(REJ("CARDHOLDER ID")),1,20),REJ("COB")=$G(REJ("COB"))  I $G(REJ("DATE/TIME"))="" D NOW^%DTC S REJ("DATE/TIME")=%  S DIC="^PSRX("\_RX\_",""REJ"",",DA(1)=RX,DIC(0)=""  S X=$G(REJ("CODE")),DINUM=$O(^PSRX(RX,"REJ",9999),-1)+1  S DIC("DR")="1///"\_$G(REJ("DATE/TIME"))\_";2///"\_REJ("PAYER MESSAGE")\_";3///"\_REJ("REASON")\_";4////"\_$G(REJ("PHARMACIST"))\_";5///"\_RFL  S DIC("DR")=DIC("DR")\_";6///"\_REJ("GROUP NAME")\_";7///"\_REJ("PLAN CONTACT")\_";8///"\_$G(REJ("PREVIOUS FILL"))  S DIC("DR")=DIC("DR")\_";9///0;14///"\_$G(REJ("REASON SVC CODE"))\_";16///"\_$G(REJ("RESPONSE IEN"))  S DIC("DR")=DIC("DR")\_";17///"\_$G(REJ("OTHER REJECTS"))\_";18///"\_REJ("DUR TEXT")\_";20///"\_REJ("INSURANCE NAME")  S DIC("DR")=DIC("DR")\_";21///"\_REJ("GROUP NUMBER")\_";22///"\_REJ("CARDHOLDER ID")\_";23///"\_$G(REJ("RE-OPENED"))  S DIC("DR")=DIC("DR")\_";27///"\_REJ("COB")  S DIC("DR")=DIC("DR")\_";28///"\_REJ("DUR ADD MSG TEXT")  S DIC("DR")=DIC("DR")\_";29///"\_REJ("BIN")  ;Update Reject Resolution Required fields - PSO\*7\*421  I $G(REJ("RRR FLAG")) D  .S DIC("DR")=DIC("DR")\_";30///"\_REJ("RRR FLAG")  .S DIC("DR")=DIC("DR")\_";31///"\_REJ("RRR THRESHOLD AMT")  .S DIC("DR")=DIC("DR")\_";32///"\_REJ("RRR GROSS AMT DUE")  S DIC("DR")=DIC("DR")\_";33///"\_REJ("INSURANCE POINTER")  F L +^PSRX(RX):5 Q:$T H 15  K DD,DO D FILE^DICN K DD,DO S REJ("REJECT IEN")=+Y  S REJ("OVERRIDE MSG")=$G(DATA("OVERRIDE MSG"))  ;Comments use POSTMASTER as user for auto transfers - PSO\*7\*421  I REJ("OVERRIDE MSG")'="" D  .N ORIGIN S ORIGIN=$G(DUZ)  .S:REJ("OVERRIDE MSG")["Automatically transferred" ORIGIN=.5  .D SAVECOM^PSOREJP3(RX,REJ("REJECT IEN"),REJ("OVERRIDE MSG"),$G(REJ("DATE/TIME")),ORIGIN)  .;Insert comment for Transfer and RRR Rejects - PSO\*7\*421  .I REJ("OVERRIDE MSG")["Automatically transferred" D  ..N X,TXT  ..S TXT="Auto Send to Pharmacy Worklist due to Transfer Reject Code"  ..I $G(REJ("RRR FLAG")) S TXT="Auto Send to Pharmacy Worklist due to Reject Resolution Required Code"  ..I $G(PSOTRIC) S TXT="Auto Send to Pharmacy Worklist & OPECC - CVA/TRI"  ..S X=$$ADDCOMM^BPSBUTL(RX,RFL,TXT,1) ; IA 4719  L -^PSRX(RX)  Q | | | | | | | | |
| Modified Logic (Changes are highlighted) | | | | | | | | |
| SAVE(RX,RFL,REJ,REOPEN) ; - Saves DUR Information in the file 52  ; Input: (r) RX - Rx IEN (#52)  ; (o) RFL - Refill # (Default: most recent)  ; (o) REOPEN - value of 1 means claim being reopened; null or no value passed means reopen claim functionality not being used  ; (r) REJ - Array containing information about the REJECT on the following subscripts:  ; "BIN" - BIN Number  ; “PCN” – PCN Number  ; "CODE" - Reject Code (79 or 88)  ; "DATE/TIME" - Date/Time Reject Detected  ; "PAYER MESSAGE" - Message returned by Payer (up to 140 chars long)  ; "REASON" - Reject Reason (up to 100 chars long)  ; "DUR TEXT" - Payer's DUR description  ; "DUR ADD MSG TEXT" - Payer's DUR additional message text description  ; "INSURANCE NAME" - Patient's Insurance Company Name  ; "INSURANCE POINTER" - Patient's Insurance Company IEN  ; "GROUP NAME" - Patient's Insurance Group Name  ; "GROUP NUMBER" - Patient's Insurance Group Number  ; "CARDHOLDER ID" - Patient's Insurance Cardholder ID  ; "COB" - Coordination of Benefits  ; "PLAN CONTACT" - Patient's Insurance Plan Contact (1-800)  ; "PREVIOUS FILL" - Plan's Previous Fill Date  ; "OTHER REJECTS" - Other Rejects with same Response  ; "PHARMACIST" - Pharmacist DUZ  ; "RESPONSE IEN" - Pointer to the RESPONSE file in ECME  ; "REASON SVC CODE" - Reason for Service Code (pointer to BPS NCPDP REASON FOR SERVICE CODE)  ; "RE-OPENED" - Re-Open Flag  ; "RRR FLAG" - Reject Resolution Required indicator (expecting 1/0 into SAVE)  ; "RRR THRESHOLD AMT" - Reject Resolution Required Dollar Threshold  ; "RRR GROSS AMT DUE" - Reject Resolution Required Gross Amount Due  ;Output: REJ("REJECT IEN")  N %,DIC,DR,DA,X,DINUM,DD,DO,DLAYGO,ERR  I '$D(RFL) S RFL=$$LSTRFL^PSOBPSU1(RX)  I '$G(PSODIV) S PSODIV=$$RXSITE^PSOBPSUT(RX,RFL)  S REJ("BIN")=$E($G(REJ("BIN")),1,6)  S REJ(“PCN”)=$G(REJ(“PCN”))  S REJ("CODE")=$G(REJ("CODE"))  ;  ; convert REJ("RRR FLAG") into internal format (1/0) if necessary. When coming into SAVE from the Re-open Reject  ; action, this flag is in the external format (YES/NO). esg - 3/29/16 - PSO\*7\*448  I $G(REJ("RRR FLAG"))="YES" S REJ("RRR FLAG")=1  I $G(REJ("RRR FLAG"))="NO" S REJ("RRR FLAG")=0  ;  ;Ignore this additional Check if reject is Reject Resolution Required reject - PSO\*7\*421  I '$G(REJ("RRR FLAG")),REJ("CODE")'=79&(REJ("CODE")'=88)&('$G(PSOTRIC))&('$G(REOPEN)) S ERR=$$EVAL^PSOREJU4(PSODIV,REJ("CODE"),$G(OPECC)) Q:'+ERR  S REJ("PAYER MESSAGE")=$E($G(REJ("PAYER MESSAGE")),1,140),REJ("REASON")=$E($G(REJ("REASON")),1,100)  S REJ("DUR TEXT")=$E($G(REJ("DUR TEXT")),1,100),REJ("DUR ADD MSG TEXT")=$E($G(REJ("DUR ADD MSG TEXT")),1,100),REJ("GROUP NAME")=$E($G(REJ("GROUP NAME"))  ,1,30)  S REJ("INSURANCE NAME")=$E($G(REJ("INSURANCE NAME")),1,30),REJ("PLAN CONTACT")=$E($G(REJ("PLAN CONTACT")),1,30)  S REJ("GROUP NUMBER")=$E($G(REJ("GROUP NUMBER")),1,30),REJ("OTHER REJECTS")=$E($G(REJ("OTHER REJECTS")),1,15)  S REJ("CARDHOLDER ID")=$E($G(REJ("CARDHOLDER ID")),1,20),REJ("COB")=$G(REJ("COB"))  I $G(REJ("DATE/TIME"))="" D NOW^%DTC S REJ("DATE/TIME")=%  S DIC="^PSRX("\_RX\_",""REJ"",",DA(1)=RX,DIC(0)=""  S X=$G(REJ("CODE")),DINUM=$O(^PSRX(RX,"REJ",9999),-1)+1  S DIC("DR")="1///"\_$G(REJ("DATE/TIME"))\_";2///"\_REJ("PAYER MESSAGE")\_";3///"\_REJ("REASON")\_";4////"\_$G(REJ("PHARMACIST"))\_";5///"\_RFL  S DIC("DR")=DIC("DR")\_";6///"\_REJ("GROUP NAME")\_";7///"\_REJ("PLAN CONTACT")\_";8///"\_$G(REJ("PREVIOUS FILL"))  S DIC("DR")=DIC("DR")\_";9///0;14///"\_$G(REJ("REASON SVC CODE"))\_";16///"\_$G(REJ("RESPONSE IEN"))  S DIC("DR")=DIC("DR")\_";17///"\_$G(REJ("OTHER REJECTS"))\_";18///"\_REJ("DUR TEXT")\_";20///"\_REJ("INSURANCE NAME")  S DIC("DR")=DIC("DR")\_";21///"\_REJ("GROUP NUMBER")\_";22///"\_REJ("CARDHOLDER ID")\_";23///"\_$G(REJ("RE-OPENED"))  S DIC("DR")=DIC("DR")\_";27///"\_REJ("COB")  S DIC("DR")=DIC("DR")\_";28///"\_REJ("DUR ADD MSG TEXT")  S DIC("DR")=DIC("DR")\_";29///"\_REJ("BIN")  S DIC(“DR”)=DIC(“DR”)\_”;34///”\_REJ(“PCN”)  ;Update Reject Resolution Required fields - PSO\*7\*421  I $G(REJ("RRR FLAG")) D  .S DIC("DR")=DIC("DR")\_";30///"\_REJ("RRR FLAG")  .S DIC("DR")=DIC("DR")\_";31///"\_REJ("RRR THRESHOLD AMT")  .S DIC("DR")=DIC("DR")\_";32///"\_REJ("RRR GROSS AMT DUE")  S DIC("DR")=DIC("DR")\_";33///"\_REJ("INSURANCE POINTER")  F L +^PSRX(RX):5 Q:$T H 15  K DD,DO D FILE^DICN K DD,DO S REJ("REJECT IEN")=+Y  S REJ("OVERRIDE MSG")=$G(DATA("OVERRIDE MSG"))  ;Comments use POSTMASTER as user for auto transfers - PSO\*7\*421  I REJ("OVERRIDE MSG")'="" D  .N ORIGIN S ORIGIN=$G(DUZ)  .S:REJ("OVERRIDE MSG")["Automatically transferred" ORIGIN=.5  .D SAVECOM^PSOREJP3(RX,REJ("REJECT IEN"),REJ("OVERRIDE MSG"),$G(REJ("DATE/TIME")),ORIGIN)  .;Insert comment for Transfer and RRR Rejects - PSO\*7\*421  .I REJ("OVERRIDE MSG")["Automatically transferred" D  ..N X,TXT  ..S TXT="Auto Send to Pharmacy Worklist due to Transfer Reject Code"  ..I $G(REJ("RRR FLAG")) S TXT="Auto Send to Pharmacy Worklist due to Reject Resolution Required Code"  ..I $G(PSOTRIC) S TXT="Auto Send to Pharmacy Worklist & OPECC - CVA/TRI"  ..S X=$$ADDCOMM^BPSBUTL(RX,RFL,TXT,1) ; IA 4719  L -^PSRX(RX)  Q | | | | | | | | |

| Subroutine Name | **SYNC2^PSOREJUT** | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify | | | Delete | | No Change | |
| Story | US583 | | | | | | | |
| Related Menu Options or ListMan Actions | Third Party Payer Rejects – Worklist [PSO REJECTS WORKLIST]  Third Party Payer Rejects – View/Process [PSO REJECTS VIEW/PROCESS] | | | | | | | |
| Related Routines | This Subroutine is Called By | | | This Subroutine Calls | | | | |
| WRKLST^PSOREJU4 | | | $$CLEAN^PSOREJU1  $$DAT^PSOREJU1  $$OTH^PSOREJU1  $$REASON^PSOREJU2 | | | | |
| Related Integration Control Registrations (ICRs) | N/A | | | | | | | |
| Data Passing | Parameter Input | | Parameter Output | | | Function Return Value | | Global Modified |
| Input Attribute Name and Definition | Name: N/A  Definition:  New  Modify  Delete  No Change | | | | | | | |
| Output Attribute Name and Definition | Name: N/A  Definition:  New  Modify  Delete  No Change | | | | | | | |
| Current Logic | | | | | | | | |
| SYNC2 ;  S (IDX,CODE)="" F S IDX=$O(REJS(IDX)) Q:IDX="" D  . F S CODE=$O(REJS(IDX,CODE)) Q:CODE="" K DATA D  . . ;Additional check for Reject Resolution Required - PSO\*7\*421  . . I 'OPECC&(CODE'[79)&(CODE'[88) D  . . .I '+$G(REJRRR(IDX)) S DATA("OVERRIDE MSG")="Automatically transferred due to override for reject code." Q  . . .;Reject Resolution Required fields  . . .S DATA("RRR FLAG")=1  . . .S DATA("RRR GROSS AMT DUE")=$P(REJRRR(IDX),U,2)  . . .S DATA("RRR THRESHOLD AMT")=$P(REJRRR(IDX),U,3)  . . .S DATA("OVERRIDE MSG")="Automatically transferred due to Reject Resolution Required reject code"  . . I OPECC&(CODE'[79)&(CODE'[88) S DATA("OVERRIDE MSG")="Transferred by "\_$S(CODE["eT":"",CODE["eC":"",1:"OPECC.") ;cnf,PSO\*7.0\*358  . . I $D(COMMTXT) S:COMMTXT'="" DATA("OVERRIDE MSG")=DATA("OVERRIDE MSG")\_" "\_$$CLEAN^PSOREJU1($P(COMMTXT,":",2))  . . S DATA("DUR TEXT")=$$CLEAN^PSOREJU1($G(REJ(IDX,"DUR FREE TEXT DESC")))  . . S DATA("DUR ADD MSG TEXT")=$$CLEAN^PSOREJU1($G(REJ(IDX,"DUR ADD MSG TEXT")))  . . ; In NCPDP D0, the Payer Additional Message is a repeating field and we want to display as much of the  . . ; data on the reject information screen as possible so we put the messages together up to the field  . . ; length of 140  . . N CNT,MSG  . . S CNT="",DATA("PAYER MESSAGE")=""  . . F S CNT=$O(REJ(IDX,"PAYER MESSAGE",CNT)) Q:CNT=""!($L(DATA("PAYER MESSAGE"))>140) D  . . . S MSG=$$CLEAN^PSOREJU1(REJ(IDX,"PAYER MESSAGE",CNT))  . . . I MSG]"" S DATA("PAYER MESSAGE")=DATA("PAYER MESSAGE")\_MSG\_" "  . . ; Call CLEAN again to strip the extra trailing spaces we might have added  . . S DATA("PAYER MESSAGE")=$$CLEAN^PSOREJU1(DATA("PAYER MESSAGE"))  . . S DATA("CODE")=CODE,DATA("REASON")=$$CLEAN^PSOREJU1($G(REJ(IDX,"REASON")))  . . S DATA("PHARMACIST")=$G(USR),DATA("INSURANCE NAME")=$$CLEAN^PSOREJU1($G(REJ(IDX,"INSURANCE NAME")))  . . S DATA("INSURANCE POINTER")=$$CLEAN^PSOREJU1($G(REJ(IDX,"INSURANCE POINTER")))  . . S DATA("GROUP NAME")=$$CLEAN^PSOREJU1($G(REJ(IDX,"GROUP NAME"))),DATA("GROUP NUMBER")=$$CLEAN^PSOREJU1($G(REJ(IDX,"GROUP NUMBER")))  . . S DATA("CARDHOLDER ID")=$$CLEAN^PSOREJU1($G(REJ(IDX,"CARDHOLDER ID"))),DATA("PLAN CONTACT")=$$CLEAN^PSOREJU1($G(REJ(IDX,"PLAN CONTACT")))  . . S DATA("PREVIOUS FILL")=$$CLEAN^PSOREJU1($$DAT^PSOREJU1($G(REJ(IDX,"PREVIOUS FILL DATE"))))  . . S DATA("OTHER REJECTS")=$$CLEAN^PSOREJU1($$OTH^PSOREJU1(CODE,$G(REJ(IDX,"REJ CODE LST"))))  . . S DATA("RESPONSE IEN")=+$$CLEAN^PSOREJU1($G(REJ(IDX,"RESPONSE IEN")))  . . S DATA("REASON SVC CODE")=$$REASON^PSOREJU2($G(REJ(IDX,"REASON"))),DATA("COB")=IDX  . . S DATA("MESSAGE")=$$CLEAN^PSOREJU1($G(REJ(IDX,"MESSAGE")))  . . S DATA("DUR RESPONSE DATA")=$$CLEAN^PSOREJU1($G(REJ(IDX,"DUR RESPONSE DATA")))  . . S DATA("BIN")=$$CLEAN^PSOREJU1($G(REJ(IDX,"BIN")))  . . D SAVE(RX,RFL,.DATA)  L -^PSRX("REJ",RX)  Q | | | | | | | | |
| Modified Logic (Changes are highlighted) | | | | | | | | |
| SYNC2 ;  S (IDX,CODE)="" F S IDX=$O(REJS(IDX)) Q:IDX="" D  . F S CODE=$O(REJS(IDX,CODE)) Q:CODE="" K DATA D  . . ;Additional check for Reject Resolution Required - PSO\*7\*421  . . I 'OPECC&(CODE'[79)&(CODE'[88) D  . . .I '+$G(REJRRR(IDX)) S DATA("OVERRIDE MSG")="Automatically transferred due to override for reject code." Q  . . .;Reject Resolution Required fields  . . .S DATA("RRR FLAG")=1  . . .S DATA("RRR GROSS AMT DUE")=$P(REJRRR(IDX),U,2)  . . .S DATA("RRR THRESHOLD AMT")=$P(REJRRR(IDX),U,3)  . . .S DATA("OVERRIDE MSG")="Automatically transferred due to Reject Resolution Required reject code"  . . I OPECC&(CODE'[79)&(CODE'[88) S DATA("OVERRIDE MSG")="Transferred by "\_$S(CODE["eT":"",CODE["eC":"",1:"OPECC.") ;cnf,PSO\*7.0\*358  . . I $D(COMMTXT) S:COMMTXT'="" DATA("OVERRIDE MSG")=DATA("OVERRIDE MSG")\_" "\_$$CLEAN^PSOREJU1($P(COMMTXT,":",2))  . . S DATA("DUR TEXT")=$$CLEAN^PSOREJU1($G(REJ(IDX,"DUR FREE TEXT DESC")))  . . S DATA("DUR ADD MSG TEXT")=$$CLEAN^PSOREJU1($G(REJ(IDX,"DUR ADD MSG TEXT")))  . . ; In NCPDP D0, the Payer Additional Message is a repeating field and we want to display as much of the  . . ; data on the reject information screen as possible so we put the messages together up to the field  . . ; length of 140  . . N CNT,MSG  . . S CNT="",DATA("PAYER MESSAGE")=""  . . F S CNT=$O(REJ(IDX,"PAYER MESSAGE",CNT)) Q:CNT=""!($L(DATA("PAYER MESSAGE"))>140) D  . . . S MSG=$$CLEAN^PSOREJU1(REJ(IDX,"PAYER MESSAGE",CNT))  . . . I MSG]"" S DATA("PAYER MESSAGE")=DATA("PAYER MESSAGE")\_MSG\_" "  . . ; Call CLEAN again to strip the extra trailing spaces we might have added  . . S DATA("PAYER MESSAGE")=$$CLEAN^PSOREJU1(DATA("PAYER MESSAGE"))  . . S DATA("CODE")=CODE,DATA("REASON")=$$CLEAN^PSOREJU1($G(REJ(IDX,"REASON")))  . . S DATA("PHARMACIST")=$G(USR),DATA("INSURANCE NAME")=$$CLEAN^PSOREJU1($G(REJ(IDX,"INSURANCE NAME")))  . . S DATA("INSURANCE POINTER")=$$CLEAN^PSOREJU1($G(REJ(IDX,"INSURANCE POINTER")))  . . S DATA("GROUP NAME")=$$CLEAN^PSOREJU1($G(REJ(IDX,"GROUP NAME"))),DATA("GROUP NUMBER")=$$CLEAN^PSOREJU1($G(REJ(IDX,"GROUP NUMBER")))  . . S DATA("CARDHOLDER ID")=$$CLEAN^PSOREJU1($G(REJ(IDX,"CARDHOLDER ID"))),DATA("PLAN CONTACT")=$$CLEAN^PSOREJU1($G(REJ(IDX,"PLAN CONTACT")))  . . S DATA("PREVIOUS FILL")=$$CLEAN^PSOREJU1($$DAT^PSOREJU1($G(REJ(IDX,"PREVIOUS FILL DATE"))))  . . S DATA("OTHER REJECTS")=$$CLEAN^PSOREJU1($$OTH^PSOREJU1(CODE,$G(REJ(IDX,"REJ CODE LST"))))  . . S DATA("RESPONSE IEN")=+$$CLEAN^PSOREJU1($G(REJ(IDX,"RESPONSE IEN")))  . . S DATA("REASON SVC CODE")=$$REASON^PSOREJU2($G(REJ(IDX,"REASON"))),DATA("COB")=IDX  . . S DATA("MESSAGE")=$$CLEAN^PSOREJU1($G(REJ(IDX,"MESSAGE")))  . . S DATA("DUR RESPONSE DATA")=$$CLEAN^PSOREJU1($G(REJ(IDX,"DUR RESPONSE DATA")))  . . S DATA("BIN")=$$CLEAN^PSOREJU1($G(REJ(IDX,"BIN")))  . . S DATA(“PCN”)=$$CLEAN^PSOREJU1($G(REJ(IDX,”PCN”)))  . . D SAVE(RX,RFL,.DATA)  L -^PSRX("REJ",RX)  Q | | | | | | | | |

The field PCN will be added to sub-file #52.25, Reject Info, of file #52, Prescription.

|  |  |
| --- | --- |
| **Field Attributes** | **Values** |
| **FIELD NAME** | PCN |
| **FIELD NUMBER** | 34 |
| **NODE;PIECE** | 2;10 |
| **FIELD TYPE** | Free Text |
| **FIELD LENGTH** | 10 |
| **INPUT TRANSFORM** | K:$L(X)>10!($L(X)<10) X |
| **HELP PROMPT** | Answer must be 10 characters in length. |
| **DESCRIPTION** | Number assigned by the processor. |
| **TECHNICAL DESCRIPTION** | REJECT INFO Sub-File (#52.25) |

ICR agreement to govern use of $$TAXID^IBCEF75. Called from DVINFO^PSOREJU2 and DVINFO^BPSSCRRJ.

|  |  |
| --- | --- |
| Integration Control Registration # | **TBD** |
| Name | FEDERAL TAX ID FOR SITE |
| Enhancement Category | New  Modify  Expand |
| Custodial Package | Integrated Billing |
| Subscribing Package(s) | E Claims Mgmt Engine  Outpatient Pharmacy |
| Usage | Controlled Subscription |
| Type | Routine |
| Routine | IBCEF75 |
| Component | $$TAXID() |
| Input Variables | Name: n/a  Description: |
| Output Variables | Name: TAXID  Description: Federal Tax Number for the facility. |

# Back Out/Rollback Procedure

This story includes modifications to existing routines and files.

To back out this enhancement, all the routines which are a part of this patch should be backed up prior to installation. Reverting to the previous version of these routines would back out this enhancement. Alternatively, the previous version of these routines could be distributed via an emergency patch.

Rolling back the changes to the database would involve the following steps, which could be performed by a developer or IRM at each site or via an emergency patch:

* Delete any values in field #34, PCN, from the Reject Info sub-file of the Prescription File. This could be performed via FileMan or from a routine written for the purpose.
* Delete field #34, PCN, from the data dictionary for sub-file #52.25, REJECT INFO. This should be performed via FileMan.