**MCCF EDI TAS US3529**

System Design Document



Department of Veterans Affairs

**June 2018**

Version 1.05

**User Story Number:** USRX-

**User Story Name:** Most Recent version of NCPDP Telecom Standard

**Product Backlog ID:**

**Backlog Priority:**

**Initial Sizing Estimate:**

**Rational ID:**

**Rally ID:**

**Epic Taxonomy:** eBiz Compliance

# Design – Summary

The National Council for Prescription Drug Programs (NCPDP) publishes updates to the fields on a quarterly basis.  The annual implementation schedule incorporates up to four (4) publications each year.  The new annual implementations become effective each October.  NCPDP may publish emergency updates which are required to be implemented any date prior to the scheduled annual implementation date, but must be at least 180 days from the publication date. This user story includes the updates for all four of the publications from 2017.

List of Components:

* File: BPS CLAIMS
* File: BPS NCPDP FIELD DEFS
* File: BPS NCPDP SEGMENTS
* File: BPS RESPONSES
* File: BPS REQUESTS
* File: BPS TRANSACTION
* File: BPS PAYER RESPONSE OVERRIDES
* File: BPS NCPDP PRESCRIBER PLACE OF SERVICE
* File: BPS NCPDP BENEFIT STAGE INDICATOR
* File: BPS NCPDP LTPAC DISPENSE FREQUENCY
* File: BPS NCPDP PATIENT PAY COMPONENT QUALIFIER
* File: BPS NCPDP OTHER PAYER PROGRAM TYPE
* Routine: BPS24PRE
* Routine: BPSBUTL
* Routine: BPSECX0
* Routine: BPSFLD01
* Routine: BPSOSCD
* Routine: BPSOSCE
* Routine: BPSOSHF
* Routine: BPSOSRX3
* Routine: BPSOSSG
* Routine: BPSPRRX3
* Routine: BPSPRRX6
* Routine: BPSSCRL1
* Routine: BPSSCRLG
* Routine: BPSTEST
* Routine: BPSTEST2
* Routine: IBJTRX
* Routine: PSOREJP1
* Routine: PSOREJP5

# Design – Detail

In reviewing the latest NCPDP Implementation Guide (October 2017), it was determined that updates must be made to the VistA files corresponding to Data Elements.

Because both the field def number and name are identifier fields in file 9002313.91, BPS NCPDP FIELD DEFS, field name updates for existing fields must be performed in the BPS patch Pre-Installation routine ^BPS24PRE. The name is also a identifier field in file 9002313.9, BPS NCPDP SEGMENTS, therefore field name updates for an existing field must be performed in the BPS patch Pre-Installation routine ^BPS24PRE. Modifying the field names in both of these files in the pre-install routine prevents duplicate entries from being created by KIDS when the data file is added to the system.

| Subroutine Name | **PRE^BPS24PRE** | | | | |
| --- | --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify | | Delete | No Change |
| Related Integration Control Registrations (ICRs) | 10141 for MES^XPDUTL | |  |  |  |
| Current Logic | | | | | |
| PRE ; Pre-install functions are coded here.  ;  D MES^XPDUTL(" Starting pre-install of BPS\*1.0\*24")  D UPDOPT  D BMES^XPDUTL(" Finished pre-install of BPS\*1.0\*24")  Q | | | | | |
| Modified Logic (Changes are highlighted) | | | | | |
| PRE ; Pre-install functions are coded here.  ;  D MES^XPDUTL(" Starting pre-install of BPS\*1.0\*24")  D UPDOPT  ; Update descriptions in BPS NCPDP FIELD DEFS file #9002313.91  D FIELDS   ; Update segment name in BPS NCPDP SEGMENTS file #9002313.9  D UPDSEG  D BMES^XPDUTL(" Finished pre-install of BPS\*1.0\*24")  Q | | | | | |

| Subroutine Name | **FIELDS^BPS24PRE** | | | |
| --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify | Delete | No Change |
| Current Logic | | | | |
| n/a – new subroutine | | | | |
| Modified Logic (Changes are highlighted) | | | | |
| FIELDS ; Update Fields Defs with new descriptions  N LINE,DATA,NUM,NAME,DA,DIE,DR,CNT  D MES^XPDUTL(" - Updating BPS NCPDP FIELD DEFS")  S CNT=0  F LINE=1:1 S DATA=$P($T(NFLDS+LINE),";;",2,99) Q:DATA=""  D  . S NUM=$P(DATA,";",1),NAME=$P(DATA,";",2)  . S DIE=9002313.91   . S DA=$O(^BPSF(DIE,"B",NUM,""))  . I 'DA D MES^XPDUTL(" - No IEN found for entry "\_NUM\_",field: "\_NAME) Q  . S DR=".03////^S X=NAME",CNT=CNT+1  . D ^DIE  D MES^XPDUTL(" - "\_CNT\_" entries updated")  D MES^XPDUTL(" - Done with BPS NCPDP FIELD DEFS")  D MES^XPDUTL(" ")  Q  ; NFLDS ; Updated field names  ;;101;IIN NUMBER  ;;348;BASIS OF CALC - REGULATORY FEE  ;;349;BASIS OF CALC - % TAX  ;;433;PATIENT PAID AMOUNT REPORTED  ;;481;REGULATORY FEE AMT SUBMITTED  ;;482;PERCENT TAX AMT SUBMITTED  ;;483;PERCENT TAX RATE SUBMITTED  ;;484;PERCENTAGE TAX BASIS SBMTTD  ;;530;PREVIOUS DATE OF SERVICE  ;;531;QTY OF PREVIOUS DISPENSING  ;;551;FORMULARY ALT PRODUCT COUNT  ;;552;FORMULARY ALT ID QUALIFIER  ;;553;FORMULARY ALTERNATIVE ID  ;;554;FORMULARY ALT INCENTIVE  ;;555;FORMULARY ALT COST SHARE  ;;556;FORMULARY ALT DESCRIPTION  ;;557;PERCENTAGE TAX EXEMPT INDICATOR  ;;558;REGULATORY FEE AMOUNT PAID  ;;559;PERCENTAGE TAX AMOUNT PAID  ;;560;PERCENTAGE TAX RATE PAID  ;;561;PERCENTAGE TAX BASIS PAID  ;;568;PAYER/HEALTH PLAN ID QUALIFIER  ;;569;PAYER/HEALTH PLAN ID  ;;575;PATIENT PERCENTAGE TAX AMOUNT  ;;A28;ADJUDICATED PROGRAM TYPE   ;;A29;REPORTED ADJUDICATED PROG TYPE  ;;B56;LAST KNOWN IIN NUMBER  ; | | | | |

The following entries in the BPS NCPDP FIELD DEFS file need to be edited to accurately reflect the NCPDP name/description.

101-A1 – IIN NUMBER

Current VistA Name: BIN NUMBER

Updated VistA Name: IIN NUMBER

348-HK – BASIS OF CALCULATION-REGULATORY FEE

Current VistA Name: BASIS OF CALC – FLAT SALES TAX

Updated VistA Name: BASIS OF CALC – REGULATORY FEE

349-HM – BASIS OF CALCULATION-PERCENTAGE TAX

Current VistA Name: BASIS OF CALC - % SALES TAX

Updated VistA Name: BASIS OF CALC - % TAX

433-DX – PATIENT PAID AMOUNT REPORTED

Current VistA Name: PATIENT PAID AMOUNT SUBMITTED

Updated VistA Name: PATIENT PAID AMOUNT REPORTED

481-HA - REGULATORY FEE AMOUNT SUBMITTED

Current VistA Name: FLAT SALES TAX AMT SUBMITTED

Updated VistA Name: REGULATORY FEE AMT SUBMITTED

482-GE – PERCENT TAX AMOUNT SUBMITTED

Current VistA Name: PERCENT SALES TAX AMT SBMTTD

Updated VistA Name: PERCENT TAX AMT SUBMITTED

483-HE - PERCENT TAX RATE SUBMITTED

Current VistA Name: PERCENT SALES TAX RATE SBMTTD

Updated VistA Name: PERCENT TAX RATE SUBMITTED

484-JE - PERCENTAGE TAX BASIS SUBMITTED

Current VistA Name: PERCENT SALES TAX BASIS SBMTTD

Updated VistA Name: PERCENTAGE TAX BASIS SBMTTD

530-FU - PREVIOUS DATE OF SERVICE

Current VistA Name: PREVIOUS DATE OF FILL

Updated VistA Name: PREVIOUS DATE OF SERVICE

531-FV - QUANTITY OF PREVIOUS DISPENSING

Current VistA Name: QUANTITY OF PREVIOUS FILL

Updated VistA Name: QTY OF PREVIOUS DISPENSING

551-9F - FORMULARY ALTERNATIVE PRODUCT COUNT

Current VistA Name: PREFERRED PRODUCT COUNT

Updated VistA Name: FORMULARY ALT PRODUCT COUNT

552-AP - FORMULARY ALTERNATIVE ID QUALIFIER

Current VistA Name: PREFERRED PRODUCT ID QUALIFIER

Updated VistA Name: FORMULARY ALT ID QUALIFIER

553-AR - FORMULARY ALTERNATIVE ID

Current VistA Name: PREFERRED PRODUCT ID

Updated VistA Name: FORMULARY ALTERNATIVE ID

554-AS - FORMULARY ALTERNATIVE INCENTIVE

Current VistA Name: PREFERRED PRODUCT INCENTIVE

Updated VistA Name: FORMULARY ALT INCENTIVE

555-AT - FORMULARY ALTERNATIVE COST SHARE

Current VistA Name: PREF PRODUCT COST SHARE INCNTV

Updated VistA Name: FORMULARY ALT COST SHARE

556-AU - FORMULARY ALTERNATIVE DESCRIPTION

Current VistA Name: PREFERRED PRODUCT DESCRIPTION

Updated VistA Name: FORMULARY ALT DESCRIPTION

557-AV – PERCENTAGE TAX EXEMPT INDICATOR

Current VistA Name: TAX EXEMPT INDICATOR

Updated VistA Name: PERCENTAGE TAX EXEMPT INDICATOR

558-AW - REGULATORY FEE AMOUNT PAID

Current VistA Name: FLAT SALES TAX AMOUNT PAID

Updated VistA Name: REGULATORY FEE AMOUNT PAID

559-AX – PERCENTAGE TAX AMOUNT PAID

Current VistA Name: PERCENTAGE SALES TAX AMT PAID

Updated VistA Name: PERCENTAGE TAX AMOUNT PAID

560-AY - PERCENTAGE TAX RATE PAID

Current VistA Name: PERCENTAGE SALES TAX RATE PAID

Updated VistA Name: PERCENTAGE TAX RATE PAID

561-AZ - PERCENTAGE TAX BASIS PAID

Current VistA Name: PERCENT SALES TAX BASIS PAID

Updated VistA Name: PERCENTAGE TAX BASIS PAID

568-J7 - PAYER/HEALTH PLAN ID QUALIFIER

Current VistA Name: PAYER ID QUALIFIER

Updated VistA Name: PAYER/HEALTH PLAN ID QUALIFIER

569-J8 - PAYER/HEALTH PLAN ID

Current VistA Name: PAYER ID

Updated VistA Name: PAYER/HEALTH PLAN ID

575-EQ - PATIENT PERCENTAGE TAX AMOUNT

Current VistA Name: PATIENT SALES TAX

Updated VistA Name: PATIENT PERCENTAGE TAX AMOUNT

A28-ZR - ADJUDICATED PROGRAM TYPE

Current VistA Name: ADJUDICATED PAYMENT TYPE

Updated VistA Name: ADJUDICATED PROGRAM TYPE

A29-ZS - REPORTED ADJUDICATED PROGRAM TYPE

Current VistA Name: REPORTED PAYMENT TYPE

Updated VistA Name: REPORTED ADJUDICATED PROG TYPE

B56-3E - LAST KNOWN IIN NUMBER

Current VistA Name: LAST KNOWN BIN NUMBER

Updated VistA Name: LAST KNOWN IIN NUMBER

The following entry in the BPS NCPDP SEGMENTS file will need to be edited to accurately reflect the NCPDP name.

Current Segment Name: RESPONSE COORDINATION OF BENEFITS/OTHER PAYERS

Updated Segment Name: RESPONSE OTHER PAYERS

| Subroutine Name | **UPDSEG^BPS24PRE** | | | |
| --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify | Delete | No Change |
| Current Logic | | | | |
| n/a – new subroutine | | | | |
| Modified Logic (Changes are highlighted) | | | | |
| UPDSEG ; Update Segment with new   N DA,DIE,DR,NAME,ONAME  D MES^XPDUTL(" - Updating BPS NCPDP SEGMENTS")  S ONAME="RESPONSE COORDINATION OF BENEF"  S NAME="RESPONSE OTHER PAYERS"  S DIE=9002313.9  S DA=$O(^BPSF(DIE,"B",ONAME,""))  I 'DA D MES^XPDUTL(" - No IEN found for entry "\_ONAME\_",field: "\_NAME) Q  S DR=".01////^S X=NAME"  D ^DIE   D MES^XPDUTL(" - entry updated")  D MES^XPDUTL(" - Done with BPS NCPDP SEGMENTS")  D MES^XPDUTL(" ")  Q  ; | | | | |

The five (5) tables below describe how all the new field definitions are going to be added to file 9002313.91, BPS NCPDP FIELD DEFS, how the new outgoing fields are going to be added to file 9002313.02, BPS CLAIMS, and how the new incoming fields are going to be added to file 9002313.03 , BPS RESPONSES.

1 The table below describes how the new incoming fields are going to be added to file 9002313.91, BPS NCPDP FIELD DEFS.

| **.01 -**  **NO** | **.03 - NAME** | **.04 -**  **FORMAT** | **.06**  **ID** | **.07 -**  **LENGTH** | **1 – VISTA**  **FIELD NO** | **1.01 – STANDARD NCPDP FIELD NAME** | **3 – RESPONSE SEGMENT** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 931 | Maximum Age Qualifier | A/N | F8 | 1 | 931 | Maximum Age Qualifier | RESPONSE CLAIM |
| 932 | Maximum Age | N | GA | 3 | 932 | Maximum Age | RESPONSE CLAIM |
| 933 | Maximum Amount | N | GB | 10 | 933 | Maximum Amount | RESPONSE CLAIM |
| 934 | Maximum Amount Qualifier | A/N | GC | 2 | 934 | Maximum Amount Qualifier | RESPONSE CLAIM |
| 935 | Maximum Amount Time Period | A/N | GF | 2 | 935 | Maximum Amount Time Period | RESPONSE CLAIM |
| 936 | Max Amt Time Period Start Date | N | GG | 8 | 936 | Maximum Amount Time Period Start Date | RESPONSE CLAIM |
| 937 | Max Amt Time Period End Date | N | GH | 8 | 937 | Maximum Amount Time Period End Date | RESPONSE CLAIM |
| 938 | Max Amt Time Period Units | N | GJ | 4 | 938 | Maximum Amount Time Period Units | RESPONSE CLAIM |
| 943 | Minimum Age Qualifier | A/N | GQ | 1 | 943 | Minimum Age Qualifier | RESPONSE CLAIM |
| 944 | Minimum Age | N | GR | 3 | 944 | Minimum Age | RESPONSE CLAIM |
| C58 | Benefit Type Opportunity | A/N | AE | 2 | 2158 | Benefit Type Opportunity | RESPONSE CLAIM |
| C59 | Benefit Type Opportunity Count | N | AF | 1 | 2159 | Benefit Type Opportunity Count | RESPONSE CLAIM |
| C66 | Help Desk Business Unit Type | A/N | BA | 2 | 2166 | Help Desk Business Unit Type | RESPONSE STATUS |
| C67 | Help Desk Bus Unit Type Count | N | BB | 2 | 2167 | Help Desk Business Unit Type Count | RESPONSE STATUS |
| C68 | Help Desk Contact Information | A/N | BC | 255 | 2168 | Help Desk Contact Information | RESPONSE STATUS |
| C69 | Help Desk Contact Info Ext | A/N | BD | 8 | 2169 | Help Desk Contact Information Extension | RESPONSE STATUS |
| C70 | Help Desk Contact Info Qual | A/N | BF | 2 | 2170 | Help Desk Contact Information Qualifier | RESPONSE STATUS |
| C71 | Help Desk Support Type | A/N | BG | 2 | 2171 | Help Desk Support Type | RESPONSE STATUS |
| C72 | Help Desk Support Type Count | N | BH | 2 | 2172 | Help Desk Support Type Count | RESPONSE STATUS |
| C80 | Intermediary Help Desk Type | A/N | G8 | 2 | 2180 | Intermediary Help Desk Business Unit Type | RESPONSE INTERMEDIARY |
| C81 | Intermediary HlpDsk Bus Count | N | G9 | 2 | 2181 | Intermediary Help Desk Business Unit Type Count | RESPONSE INTERMEDIARY |
| C82 | Intermediary HlpDsk Contact | A/N | JP | 255 | 2182 | Intermediary Help Desk Contact Information | RESPONSE INTERMEDIARY |
| C83 | Intermediary HlpDsk Extension | A/N | JR | 8 | 2183 | Intermediary Help Desk Contact Information Extension | RESPONSE INTERMEDIARY |
| C84 | Intermediary HlpDsk Qualifier | A/N | KA | 2 | 2184 | Intermediary Help Desk Contact Information Qualifier | RESPONSE INTERMEDIARY |
| C85 | Intermediary HlpDsk Suppt Type | A/N | KB | 2 | 2185 | Intermediary Help Desk Support Type | RESPONSE INTERMEDIARY |
| C86 | Intermediary HlpDsk Typ  Count | N | KC | 2 | 2186 | Intermediary Help Desk Support Type Count | RESPONSE INTERMEDIARY |
| C93 | Patient Pay Component Amount | N | KN | 8 | 2193 | Patient Pay Component Amount | RESPONSE PRICING |
| C94 | Patient Pay Component Count | N | KP | 2 | 2194 | Patient Pay Component Count | RESPONSE PRICING |
| C95 | Patient Pay Component Qual | A/N | KQ | 2 | 2195 | Patient Pay Component Qualifier | RESPONSE PRICING |
| C96 | Payer/Health Plan ID Count | N | KR | 1 | 2196 | Payer/Health Plan ID Count | RESPONSE INSURANCE |
| D15 | Subrogtn Requestr Reconcil ID | A/N | KY | 30 | 2215 | Subrogation Requestors Reconciliation ID | RESPONSE CLAIM |
| D19 | Minimum Amount | N | M1 | 10 | 2219 | Minimum Amount | RESPONSE CLAIM |
| D20 | Minimum Amount Qualifier | A/N | M2 | 3 | 2220 | Minimum Amount Qualifier | RESPONSE CLAIM |
| D23 | Other Payer Name | A/N | M5 | 70 | 2223 | Other Payer Name | RESPONSE OTHER PAYERS |
| D24 | Remaining Amount | N | M6 | 10 | 2224 | Remaining Amount | RESPONSE CLAIM |
| D25 | Remaining Amount Qualifier | A/N | M7 | 3 | 2225 | Remaining Amount Qualifier | RESPONSE CLAIM |
| D41 | Other Payer Relationship Type | A/N | PQ | 3 | 2241 | Other Payer Relationship Type | RESPONSE OTHER PAYERS |
| D42 | Formulary Alt Benefit Tier | A/N | PV | 2 | 2242 | Formulary Alternative Plan Benefit Tier | RESPONSE CLAIM |
| D43 | Formulary Alt Reason Code | A/N | PZ | 2 | 2243 | Formulary Alternative Reason Code | RESPONSE CLAIM |
| D44 | Formulry Alt Req Therapy Count | A/N | P0 | 2 | 2244 | Formulary Alternative Required Therapy Indicator Count | RESPONSE CLAIM |
| D45 | Formulry Alt Therapy Indicator | A/N | P1 | 2 | 2245 | Formulary Alternative Required Therapy Indicator | RESPONSE CLAIM |
| D46 | Formulry Alt Therapy Time Qual | A/N | P2 | 2 | 2246 | Formulary Alternative Required Therapy Time Period Qualifier | RESPONSE CLAIM |
| D47 | Formulry Alt Therapy Duration | A/N | P3 | 3 | 2247 | Formulary Alternative Required Therapy Time Period Duration | RESPONSE CLAIM |
| D48 | Formulary Alt Therapy Start Date | N | P4 | 8 | 2248 | Formulary Alternative Required Therapy Time Period Start Date | RESPONSE CLAIM |
| D49 | Formulry Alt Therapy End Date | N | P5 | 9 | 2249 | Formulary Alternative Required Therapy Time Period End Date | RESPONSE CLAIM |
| D50 | Other Payer Benefit Class | A/N | P6 | 10 | 2250 | Other Payer Benefit Classification | RESPONSE OTHER PAYERS |
| D54 | Plan Override Indicator | A/N | RC | 6 | 2254 | Plan Benefit Override Indicator | RESPONSE CLAIM |
| D55 | Plan Override Value Count | N | RD | 1 | 2255 | Plan Benefit Override Value Count | RESPONSE CLAIM |
| D56 | Plan Benefit Override Value | A/N | RF | 10 | 2256 | Plan Benefit Override Value | RESPONSE CLAIM |
| D62 | Regulatory Fee Exmpt Indicator | A/N | RM | 1 | 2262 | Regulatory Fee Exempt Indicator | RESPONSE PRICING |
| D65 | Patient Regulatory Fee  Amount | N | RS | 8 | 2265 | Patient Regulatory Fee  Amount | RESPONSE PRICING |

All of the new incoming fields listed above will have the same value for the GET CODE field #10 in file 9002313.91, BPS NCPDP FIELD DEFS. The value for the GET CODE field #10 will be "; This is a response-only field which does not use the GET, FORMAT, or SET code".

2a The table below describes how the new outgoing fields are going to be added to the non-coding fields in file 9002313.91, BPS NCPDP FIELD DEFS. There is another table that will contain the coding fields.

| **.01 -**  **NO** | **.03 - NAME** | **.04 -**  **FORMAT** | **.06**  **ID** | **.07 -**  **LENGTH** | **1 – VISTA**  **FIELD NO** | **1.01 – STANDARD NCPDP FIELD NAME** | **2 – REQUEST SEGMENT** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| C49 | Other Payer Reconciliation ID | A/N | 9V | 30 | 2149 | Other Payer Reconciliation ID | COORDINATION OF BENEFITS/OTHER |
| C60 | Compound Level of Complexity | A/N | AG | 2 | 2160 | Compound Level of Complexity | COMPOUND |
| C90 | LTPAC Billing Methodology | N | KH | 1 | 2190 | LTPAC Billing Methodology | CLAIM |
| C91 | LTPAC Dispense Frequency | N | KK | 2 | 2191 | LTPAC Dispense Frequency | CLAIM |
| C92 | Number LTPAC Dispensing Events | N | KM | 3 | 2192 | Number Of LTPAC Dispensing Events | CLAIM |
| C98 | Preparation Environment Code | N | KT | 3 | 2198 | Preparation Environment Event Code | CLAIM |
| C99 | Preparation Environment Type | A/N | KU | 2 | 2199 | Preparation Environment Type | CLAIM |
| D01 | Prescriber DEA Number | A/N | KV | 15 | 2201 | Prescriber DEA Number | PRESCRIBER |
| D02 | Total Prescribed Qty Remaining | N | KW | 10 | 2202 | Total Prescribed Quantity Remaining | CLAIM |
| D14 | Subrogation Amount Requested | N | KX | 8 | 2214 | Subrogation Amount Requested | PRICING |
| D16 | Submission Type Code Count | N | KZ | 1 | 2216 | Submission Type Code Count | CLAIM |
| D17 | Submission Type Code | A/N | K8 | 2 | 2217 | Submission Type Code | CLAIM |
| D18 | Do Not Dispense Before Date | N | K9 | 8 | 2218 | Do Not Dispense Before Date | CLAIM |
| D21 | Multiple Rx/Svc Group ID | A/N | M3 | 35 | 2221 | Multiple Prescription/Service Order Group ID | CLAIM |
| D22 | Mult Rx/Svc Group Reason Code | A/N | M4 | 2 | 2222 | Multiple Prescription/Service Order Group Reason Code | CLAIM |
| D51 | Other Payr Tax Exempt Indicatr | A/N | P7 | 1 | 2250 | Other Payer Percentage Tax Exempt Indicator | COORDINATION OF BENEFITS/OTHER |
| D52 | Other Payr Fee Exmpt Indicatr | A/N | P8 | 1 | 2251 | Other Payer Regulatory Fee Exempt Indicator | COORDINATION OF BENEFITS/OTHER |
| D53 | Other Payr Fee Type Count | N | P9 | 1 | 2253 | Other Payer Regulatory Fee Type Count | COORDINATION OF BENEFITS/OTHER |
| D57 | Prescriber Place of Service | N | RG | 2 | 2257 | Prescriber Place of Service | PRESCRIBER |
| D63 | Other Payer Fee Type Code | A/N | RN | 2 | 2263 | Other Payer Regulatory Fee Type Code | COORDINATION OF BENEFITS/OTHER REQUEST |

2b The table below describes how the new outgoing fields are going to be added to what are considered the coding fields in file 9002313.91, BPS NCPDP FIELD DEFS. In the table below the FORMAT CODE field #40 was not included because it has the same value as D0 FORMAT #20.

| **.01-NO** | **10 – GET CODE** | **20 – D0 FORMAT** | **30 – SET CODE** |
| --- | --- | --- | --- |
| C49 | ; Handled by SET2149^BPSFLD01 | ; Handled by SET2149^BPSFLD01 | D SET2149^BPSFLD01 |
| C60 | S BPS("X")="" | S BPS("X")=$$ANFF^BPSECFM($G(BPS(“X”)),2) | S $P(^BPSC(BPS(9002313.02),400,BPS(9002313.0201),"C50"),U,10)=BPS("X") |
| C90 | S BPS("X")="" | S BPS("X")=$$NFF^BPSECFM($G(BPS("X")),1) | S $P(^BPSC(BPS(9002313.02),400,BPS(9002313.0201),"C80"),U,10)=BPS("X") |
| C91 | S BPS("X")="" | S BPS("X")=$$NFF^BPSECFM($G(BPS("X")),2) | S $P(^BPSC(BPS(9002313.02),400,BPS(9002313.0201),"C90"),U,1)=BPS("X") |
| C92 | S BPS("X")="" | S BPS("X")=$$NFF^BPSECFM($G(BPS("X")),3) | S $P(^BPSC(BPS(9002313.02),400,BPS(9002313.0201),"C90"),U,2)=BPS("X") |
| C98 | S BPS("X")="" | S BPS("X")=$$NFF^BPSECFM($G(BPS("X")),3) | S $P(^BPSC(BPS(9002313.02),400,BPS(9002313.0201),"C90"),U,8)=BPS("X") |
| C99 | S BPS("X")="" | S BPS("X")=$$ANFF^BPSECFM($G(BPS(“X”)),2) | S $P(^BPSC(BPS(9002313.02),400,BPS(9002313.0201),"C90"),U,9)=BPS("X") |
| D01 | S BPS("X")=$G(BPS("RX",BPS(9002313.0201), "Prescriber DEA")) | S BPS("X")=$$ANFF^BPSECFM($G(BPS(“X”)),15) | S $P(^BPSC(BPS(9002313.02),400,BPS(9002313.0201),"D00"),U,1)=BPS("X") |
| D02 | D FLDD02^BPSOSSG | ; Handled by FLDD02^BPSOSSG | D FLDD02^BPSOSSG |
| D14 | S BPS("X")="" | S BPS("X")=$$NFF^BPSECFM($G(BPS("X")),8) | S $P(^BPSC(BPS(9002313.02),400,BPS(9002313.0201),"D10"),U,4)=BPS("X") |
| D16 | S BPS("X")="" | S BPS("X")=$$NFF^BPSECFM($G(BPS("X")),1) | S $P(^BPSC(BPS(9002313.02),400,BPS(9002313.0201),"D10"),U,6)=BPS("X") |
| D17 | S BPS("X")="" | S BPS("X")=$$ANFF^BPSECFM($G(BPS(“X”)),2) | S $P(^BPSC(BPS(9002313.02),400,BPS(9002313.0201),"D10"),U,7)=BPS("X") |
| D18 | S BPS("X")="" | S BPS("X")=$$NFF^BPSECFM($G(BPS("X")),8) | S $P(^BPSC(BPS(9002313.02),400,BPS(9002313.0201),"D10"),U,8)=BPS("X") |
| D21 | S BPS("X")="" | S BPS("X")=$$ANFF^BPSECFM($G(BPS(“X”)),35) | S $P(^BPSC(BPS(9002313.02),400,BPS(9002313.0201),"D20"),U,1)=BPS("X") |
| D22 | S BPS("X")="" | S BPS("X")=$$ANFF^BPSECFM($G(BPS(“X”)),2) | S $P(^BPSC(BPS(9002313.02),400,BPS(9002313.0201),"D20"),U,2)=BPS("X") |
| D51 | S BPS("X")="" | S BPS("X")=$$ANFF^BPSECFM($G(BPS(“X”)),1) | S $P(^BPSC(BPS(9002313.02),400,BPS(9002313.0201),"D50"),U,1)=BPS("X") |
| D52 | S BPS("X")="" | S BPS("X")=$$ANFF^BPSECFM($G(BPS(“X”)),1) | S $P(^BPSC(BPS(9002313.02),400,BPS(9002313.0201),"D50"),U,2)=BPS("X") |
| D53 | S BPS("X")="" | S BPS("X")=$$NFF^BPSECFM($G(BPS("X")),1) | S $P(^BPSC(BPS(9002313.02),400,BPS(9002313.0201),"D50"),U,3)=BPS("X") |
| D57 | S BPS("X")="" | S BPS("X")=$$NFF^BPSECFM($G(BPS("X")),2) | S $P(^BPSC(BPS(9002313.02),400,BPS(9002313.0201),"D50"),U,7)=BPS("X") |
| D63 | S BPS("X")="" | S BPS("X")=$$ANFF^BPSECFM($G(BPS(“X”)),2) | S $P(^BPSC(BPS(9002313.02),400,BPS(9002313.0201),"D60"),U,3)=BPS("X") |

3 The table below describes how new fields appearing on both outgoing and incoming segments are going to be added to file 9002313.91, BPS NCPDP FIELD DEFS.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **.01 -**  **NO** | **.03 – NAME** | **.04 -**  **FORMAT** | **.06**  **ID** | **.07 -**  **LENGTH** | **1 – VISTA**  **FIELD NO** | **1.01 – STANDARD NCPDP FIELD NAME** | **2 – REQUEST SEGMENT**  **3 – RESPONSE SEGMENT** |
| C47 | Other Payer Program Type | A/N | 9T | 3 | 2147 | Other Payer Adjudicated Program Type | RESPONSE OTHER PAYERS SEGMENT |
| C50 | Benefit Stage Indicator Count | N | 9W | 1 | 2150 | Benefit Stage Indicator Count | COORDINATION OF BENEFITS/OTHER , RESPONSE PRICING |
| C51 | Benefit Stage Indicator | A/N | 9X | 2 | 2151 | Benefit Stage Indicator | COORDINATION OF BENEFITS/OTHER, RESPONSE PRICING |
| D60 | Regulatory Fee Count | N | RK | 1 | 2260 | Regulatory Fee Count | PRICING,  RESPONSE PRICING |
| D61 | Regulatory Fee Type Code | A/N | RL | 2 | 2261 | Regulatory Fee Type Code | PRICING,  RESPONSE PRICING |

The D0 LENGTH #.08 has the same value as the LENGTH #.07 and the D0 FORMAT #.09 has the same value as the FORMAT #.04 for the BPS NCPDP FIELD DEFS tables above.

4 The table below describes how the new outgoing fields are going to be added to file 9002313.0201, BPS CLAIMS, TRANSACTIONS sub-file.

\* - the NCPDP field number is not part of file 9002313.0201 it was added only as a reference

| **NCPDP field # \*** | **NAME** | **FIELD**  **NO** | **NODE;**  **PIECE** | **FIELD TYPE**  **/ LENGTH** | **DESCRIPTION** |
| --- | --- | --- | --- | --- | --- |
| C47-9T | OTHER PAYER PROGRAM TYPE | 2147 | C40;7 | POINTER / 9002313.38 | The type of prescription benefit plan/program associated with the Other Payer. |
| C49-9V | OTHER PAYER RECONCILIATION ID | 2149 | C40;9 | FREE TEXT / 30 | Reconciliation ID (B98-34) as reported by the Other Payer for Paid/Accepted transactions OR for Information Reporting transactions, the designated default value for reporting a previous payer's rejected response as designated in the Other Payer Reject Code value(s) reported in the COB claim. |
| C50-9W | BENEFIT STAGE INDICATOR COUNT | 2150 | C40;10 | NUMERIC  / 1 | Count of Benefit Stage Indicator (C51-9X) occurrences. |
| C51-9X | BENEFIT STAGE INDICATOR | 2151 | C50;1 | POINTER / 9002313.35 | Code identifying the Benefit Stage(s) which applied to the claim at the time of adjudication. |
| C60-AG | COMPOUND LEVEL OF COMPLEXITY | 2160 | C50;10 | SET OF CODES | Value used by the pharmacy to indicate the complexity involved in the preparation of the compounded prescription. 1=Low Level Complexity, 10=Mid-Level Comp Non-Hazardous, 20=Mid-Level Comp Hazardous, 30=High Level Non-Hazardous, 40=High Level Hazardous, 50=High Level Non-Hazardous Sterile, 60=High Level Hazardous Sterile |
| C90-KH | LTPAC BILLING METHODOLOGY | 2190 | C80;10 | SET OF CODES  / 1 | Code indicating the billing methodology used for the claim.  1=Full quantity (dispensed on date of service),  2=Post-consumption (One or more dispensings make up the total quantity on the claim has been dispensed.), 3=Pre-consumption (One or more dispensings make up the total quantity on the claim, but all dispsensings that make up the total quantity on the claim have not yet occurred.) |
| C91-KK | LTPAC DISPENSE FREQUENCY | 2191 | C90;1 | POINTER / 9002313.36 | Code indicating the frequency of dispensing medication to a LTPAC patient. |
| C92-KM | NUMBER LTPAC DISPENSING EVENTS | 2192 | C90;2 | NUMERIC  / 3 | Value indicating the number of times pharmacy dispensed product or service for the claim period requested. |
| C98-KT | PREPARATION ENVIRONMENT CODE | 2198 | C90;8 | SET OF CODES | Event which required a special preparation environment.  Values: 1=Pill Splitting, 2=Other |
| C99-KU | PREPARATION ENVIRONMENT TYPE | 2199 | C90;9 | FREE TEXT  / 2 | Code identifying the environment in which the medication was prepared. 1=Specific Pressure Environment Not Required/Not Used, 2=Positive Pressure Non-sterile/Non-hazardous, 3=Negative Pressure Non-sterile/Hazardous, 4=Positive Pressure Sterile/Non-hazardous, 5=Negative Pressure Sterile/Hazardous |
| D01-KV | PRESCRIBER DEA NUMBER | 2201 | D00;1 | FREE TEXT  / 15 | ID assigned to a health care provider (e.g. Practitioner, Hospital, Manufacturer, etc.) by the US Drug Enforcement Administration, allowing them to distribute, dispense, administer, or conduct research with respect to controlled substances in the course of professional practice or research. |
| D02-KW | TOTAL PRESCRIBED QTY REMAINING | 2202 | D00;2 | NUMERIC  / 10 | Accumulated Total Prescribed Quantity Remaining as of the date of service. Calculated value based on: (Prescribed Quantity (460-ET)) \* (Number Of Refills Authorized (415-DF) + 1) - Accumulated Quantity Dispensed (442-E7) for each previous fill. |
| D14-KX | SUBROGATION AMOUNT REQUESTED | 2214 | D10;4 | NUMERIC  / 8 | Amount paid by the plan to the pharmacy. |
| D16-KZ | SUBMISSION TYPE CODE COUNT | 2216 | D10;6 | NUMERIC  / 1 | Count of the Submission Type Code (D17-K8) occurrences. |
| D17-K8 | SUBMISSION TYPE CODE | 2217 | D10;7 | SET OF CODES | Code identifying the type of submission as needed for appropriate transaction processing.  AA=340B, AB=Split Billing, AC=Encounter |
| D18-K9 | DO NOT DISPENSE BEFORE DATE | 2218 | D10;8 | NUMERIC  / 8 | The earliest date the prescriber indicates a prescribed drug can be dispensed. |
| D21-M3 | MULTIPLE RX/SVC GROUP ID | 2221 | D20;1 | FREE TEXT  / 35 | Unique ID assigned by the prescriber or pharmacy system to link multiple product orders together. |
| D22-M4 | MULTIPLE RX/SVC GROUP REASON CODE | 2222 | D20;2 | SET OF CODES | Indicates the reason for the quantity dispensed and/or days supply as a result of the prescriber issuing grouped prescriptions. 1=Injectable Therapy, 2=Loading Qty, 3=Maintenance Qty, 4=Unbreakable Pkg Multiple Locations, 5=Trial Fill, 6=Non-Commercially Available Dose, 7=Bundled Health Care Service |
| D51-P7 | OTHER PAYR TAX EXEMPT INDICATR | 2251 | D50;1 | SET OF CODES | Code indicating the source of the percentage tax exempt status of the other payer's claim. 1=Other Payer/Plan Is Tax Exempt, 2=Other Payer Religious Organization, 3=Other Payer Tax Exempt Certificate |
| D52-P8 | OTHER PAYR FEE EXMPT INDICATR | 2252 | D50;2 | SET OF CODES | Code Indicating the source of the regulatory fee exempt status of the other payer's claim.  1=Other Payer/Plan is Regulatory Fee Exempt,  2=Other Payer Religious Organization, 3=Other Payer Regulatory Fee Exempt Certificate |
| D53-P9 | OTHER PAYR FEE TYPE COUNT | 2253 | D50;3 | NUMERIC / 1 | Count of Other Payer Regulatory Fee Type Code. |
| D57-RG | PRESCRIBER PLACE OF SERVICE | 2257 | D50;7 | POINTER / 9002313.34 | Code identifying the place where the patient encounter occurred as reported by the prescriber. |
| D60-RK | REGULATORY FEE COUNT | 2260 | D50;10 | NUMERIC / 1 | Count of Regulatory Fee Type Code (D61-RL). |
| D61-RL | REGULATORY FEE TYPE CODE | 2261 | D60;1 | FREE TEXT  / 2 | Code identifying the type of regulatory fee. AA=LA RS 46:2625, AB=other |
| D63-RS | OTHER PAYER FEE TYPE CODE | 2263 | D60;3 | FREE TEXT  / 2 | Code identifying the type of Regulatory Fee reported by the other payer. AA=LA RS 46:2625, AB=Other |

5 The table below describes how the new incoming fields are going to be added to file 9002313.0301, BPS RESPONSES, RESPONSES sub-file.

\* - the NCPDP field number is not part of file 9002313.0301 it was added only as a reference

| **NCPDP field # \*** | **NAME** | **FIELD**  **NO** | **NODE;PIECE** | **FIELD TYPE / LENGTH** | **DESCRIPTION** |
| --- | --- | --- | --- | --- | --- |
| 931-F8 | MAXIMUM AGE QUALIFIER | 931 | 930;1 | SET OF CODES | Qualifies Maximum Age (932-GA)  D-days (24-hr periods), Y-years (12-month periods) |
| 932-GA | MAXIMUM AGE | 932 | 930;2 | NUMBER / 3 | Maximum age at which the product/service is covered (inclusive). Qualified by Maximum Age Qualifier (931-F80). |
| 933-GB | MAXIMUM AMOUNT | 933 | 930;3 | FREE TEXT / 10 | Qualified by Maximum Amount Qualifier (934-GC).  OUTPUT TRANSFORM: S Y=$J($$NFF^BPSECFM(Y,10),10,3) |
| 934-GC | MAXIMUM AMOUNT QUALIFIER | 934 | 930;4 | SET OF CODES | Qualifies Maximum Amount (933-GB).  DL=Dollar Amount, DS=Days Supply, FL=Fills, QY=Quantity |
| 935-GF | MAXIMUM AMOUNT TIME PERIOD | 935 | 930;5 | SET OF CODES | Type of time period associated with the overall Maximum Amount Qualifier (934-GC).  CM=Calendar Month, CQ=Calendar Quarter, CY=Calendar Year, DY=Days, LT=Lifetime, PD=Per Dispensing, SP=Specific Date Range |
| 936-GG | MAX AMT TIME PERIOD START DATE | 936 | 930;6 | FREE TEXT / 8 | Starting date of Specific Date Range. |
| 937-GH | MAX AMT TIME PERIOD END DATE | 937 | 930;7 | FREE TEXT / 8 | Ending date of Specific Date Range. |
| 938-GJ | MAX AMT TIME PERIOD UNITS | 938 | 930;8 | NUMBER / 4 | Number of units associated with the overall Maximum Amount Time Period (935-GF). |
| 943-GQ | MINIMUM AGE QUALIFIER | 943 | 940;3 | FREE TEXT / 1 | Qualifies Minimum Age (944-GR).  D=Days, Y=Years |
| 944-GR | MINIMUM AGE | 944 | 940;4 | NUMBER / 3 | Qualified by Minimum Age Qualifier (943-GQ). |
| C47-9T | OTHER PAYER PROGRAM TYPE | 2147 | C40;7 | FREE TEXT / 3 | The type of prescription benefit plan/program associated with the Other Payer. |
| C50-9W | BENEFIT STAGE INDICATOR COUNT | 2150 | C40;10 | NUMBER / 1 | Count of Benefit Stage Indicator (C51-9X) occurrences. |
| C51-9X | BENEFIT STAGE INDICATOR | 2151 | C50;1 | POINTER / 9002313.35 | Code identifying the Benefit Stage(s) which applied to the claim at the time of adjudication. |
| C58-AE | BENEFIT TYPE OPPORTUNITY | 2158 | C50;8 | FREE TEXT / 2 | The type of additional benefit the patient is eligible for.  1=Flu Vaccine Benefit, 2=90 Day At Retail |
| C59-AF | BENEFIT TYPE OPPORTUNITY COUNT | 2159 | C50;9 | NUMERIC / 1 | Count of Benefit Type Opportunity (C58-AE) that follow. |
| C66-BA | HELP DESK BUSINESS UNIT TYPE | 2166 | C60;6 | FREE TEXT / 2 | The type of help desk support that is available. 1=Pharmacy Help Desk, 2=Clinical/PA, 3=Health Plan, 4=Eligibility-3rd Party Liability, 5=Other |
| C67-BB | HELP DESK BUS UNIT TYPE COUNT | 2167 | C60;7 | NUMERIC / 2 | The number of Help Desk Business Unit Types (C66-BA). |
| C68-BC | HELP DESK CONTACT INFORMATION | 2168 | C60;8 | FREE TEXT / 255 | Value of the specific help desk contact information (e.g.: Telephone Number, Fax Number, URL). |
| C69-BD | HELP DESK CONTACT INFO EXT | 2169 | C60;9 | FREE TEXT / 8 | Extension of the help desk contact information. |
| C70-BF | HELP DESK CONTACT INFO QUAL | 2170 | C60;10 | SET OF CODES | Code qualifying the value in the Help Desk Contact Information (C68-BC). 1=Telephone Number, 2=Fax Number, 3=URL, 4=Other |
| C71-BG | HELP DESK SUPPORT TYPE | 2171 | C70;1 | SET OF CODES | The intended entity responsible for contacting the help desk. 1=Pharmacy, 2=Prescriber, 3=Member Services, 4=Other Payer-N1 Reporting, 5=Other |
| C72-BH | HELP DESK SUPPORT TYPE COUNT | 2172 | C70;2 | NUMERIC / 2 | The number of Help Desk Contact Support Types. |
| C80-G8 | INTERMEDIARY HELP DESK TYPE | 2180 | C70;10 | SET OF CODES | The type of intermediary help desk support that is available. 1-Intermediary/Switch, 2=REMS-Risk Evaluation Mitigation Strategy, 3=PDMP-Prescription Drug Monitoring Program, 4=Other |
| C81-G9 | INTERMEDIARY HLPDSK BUS COUNT | 2181 | C80;1 | NUMERIC / 2 | The number of Intermediary Help Desk Business Unit Types (C80-G8). |
| C82-JP | INTERMEDIARY HLPDSK CONTACT | 2182 | C80;2 | FREE TEXT / 255 | Value of the specific intermediary help desk contact information (e.g.: Telephone Number, Fax Number, URL). |
| C83-JR | INTERMEDIARY HLPDSK EXTENSION | 2183 | C80;3 | FREE TEXT / 8 | Extension of the intermediary help desk contact information. |
| C84-KA | INTERMEDIARY HLPDSK QUALIFIER | 2184 | C80;4 | SET OF CODES | Code qualifying the value in the Intermediary Help Desk Contact Information (C82-JP). 1=Telephone Number, 2=Fax Number, 3=URL, 4=Other |
| C85-KB | INTERMEDIARY HLPDSK SUPPT TYPE | 2185 | C80;5 | SET OF CODES | The intended entity responsible for contacting the intermediary help desk. 1=Pharmacy, 2=Prescriber, 3=Member Services, 4=Other |
| C86-KC | INTERMEDIARY HLPDSK TYP COUNT | 2186 | C80;6 | NUMERIC / 2 | The number of Intermediary Help Desk Contact Support Types. |
| C93-KN | PATIENT PAY COMPONENT AMOUNT | 2193 | C90;3 | FREE TEXT / 8 | The cost share amount attributed to the component of Patient Pay Amount (505-F5).  OUTPUT TRANSFORM:  S Y="$"\_$J($$DFF2EXT^BPSECFM(Y),7,2) |
| C94-KP | PATIENT PAY COMPONENT COUNT | 2194 | C90;4 | NUMERIC / 2 | Count of Patient Pay Component Qualifier(s) (C95-KQ) AND Patient Pay Component Amount(s) (C93-KN) occurrences. |
| C95-KQ | PATIENT PAY COMPONENT QUAL | 2195 | C90;5 | POINTER / 9002313.37 | Code qualifying the Patient Pay Component Amount (C93-KN). |
| C96-KR | PAYER/HEALTH PLAN ID COUNT | 2196 | C90;6 | NUMERIC / 1 | Count of Payer/Health Plan ID Occurrences. |
| D15-KY | SUBROGTN REQUESTR RECONCIL ID | 2215 | D10;5 | FREE TEXT / 30 | The Reconciliation ID returned to the pharmacy on the paid claim transaction from the original/requesting payer. |
| D19-M1 | MINIMUM AMOUNT | 2219 | D10;9 | FREE TEXT / 10 | Minimum amount for a quantity limit as specified in the Minimum Amount Qualifier (D20-M2).  OUTPUT TRANSFORM: S Y=$J($$NFF^BPSECFM(Y,10),10,3) |
| D20-M2 | MINIMUM AMOUNT QUALIFIER | 2220 | D10;10 | SET OF CODES | Qualifies the amount in the Minimum Amount (D19-M1) field. MDL=Min Dollar Amt, MDS=Min Day Supply, MFL=Min Fills, MQY=Min Qty |
| D23-M5 | OTHER PAYER NAME | 2223 | D20;3 | FREE TEXT / 70 | Name of the other payer. |
| D24-M6 | REMAINING AMOUNT | 2224 | D20;4 | FREE TEXT / 10 | Qualified by Remaining Amount Qualifier (D25-M7).  Remaining Benefit Amount as it relates to benefit dollars is represented in Remaining Benefit Amount (514-FE) field.  OUTPUT TRANSFORM: S Y=$J($$NFF^BPSECFM(Y,10),10,3) |
| D25-M7 | REMAINING AMOUNT QUALIFIER | 2225 | D20;5 | SET OF CODES | Qualifies Remaining Amount (D24-M6). RDS=Remaining Days Supply, RFL=Remaining Fills, RQY=Remaining Qty |
| D41-PQ | OTHER PAYER RELATIONSHIP TYPE | 2241 | D40;1 | SET OF CODES | Identifies the relationship of the other payer information returned by the reporting payer or entity for coordination with the Other Payer Coverage Type (338-5C) value for that payer. CP=COB Coverage Prior to Responding Payer, CS=COB Coverage Subsequent to Responding Payer, MX=Mutually Exclusive Benefits, CC=Change in Coverage, RP=Responding Payer, CE=Centralized Eligibility |
| D42-PV | FORMULARY ALT BENEFIT TIER | 2242 | D40;2 | SET OF CODES | Patient cost share tier applied to the formulary alternative product. 1=Tier 1, 2=Tier 2, 3=Tier 3, 4=Tier 4, 5=Tier 5, 9=Other |
| D43-PZ | FORMULARY ALT REASON CODE | 2243 | D40;3 | SET OF CODES | Reason for formulary alternative option(s).  1=Step Therapy Required, 2=Prescribed Drug Is Non-Formulary, 3=Prescribed Drug Requires Prior Authorization, 4=Preferred Product |
| D44-P0 | FORMULRY ALT REQ THERAPY COUNT | 2244 | D40;4 | FREE TEXT / 2 | Count of Formulary Alternative Required Therapy Indicator Occurrences. |
| D45-P1 | FORMULRY ALT THERAPY INDICATOR | 2245 | D40;5 | SET OF CODES | Code identifying required treatment with specified formulary alternative. 1=Required Treatment not otherwise specified, 2=Required Treatment, minimum time period duration required, 3=Required Treatment within specified Time Period Range |
| D46-P2 | FORMULRY ALT THERAPY TIME QUAL | 2246 | D40;6 | SET OF CODES | Code qualifying Formulary Alternative Required Therapy Time Period Duration. 1=Days, 2=Calendar Month, 3=Calendar Quarters, 4=Calendar Years, 5=Specified Date Range |
| D47-P3 | FORMULRY ALT THERAPY DURATION | 2247 | D40;7 | FREE TEXT / 3 | Minimum amount of time period duration for required therapy. Qualified by Formulary Alternative Required Therapy Time Period Qualifier (D46-P2). |
| D48-P4 | FORMULRY ALT THERAPY START DATE | 2248 | D40;8 | DATE / 8 | Start date for the required therapy time period range. |
| D49-P5 | FORMULRY ALT THERAPY END DATE | 2249 | D40;9 | DATE / 8 | End date for the required therapy time period range. |
| D50-P6 | OTHER PAYER BENEFIT CLASS | 2250 | D40;10 | FREE TEXT/ 10 | Identifies the benefit classification for other payer information associated to the patient. BEHAVORIAL, DENTAL, DME, MEDICAL, RX, VISION, UNKNOWN |
| D54-RC | PLAN OVERRIDE INDICATOR | 2254 | D50;4 | FREE TEXT / 6 | Restricted to valid NCPDP Telecommunication request fields (field number-ID). Example: 420-DK, 462-EV, etc. |
| D55-RD | PLAN OVERRIDE VALUE COUNT | 2255 | D50;5 | NUMERIC / 1 | Count of plan benefit override values. |
| D56-RF | PLAN BENEFIT OVERRIDE VALUE | 2256 | D50;6 | FREE TEXT / 10 | Identifies the plan supported value(s) for the field identified in Plan Benefit Override Indicator (D54-RC). |
| D60-RK | REGULATORY FEE COUNT | 2260 | D50;10 | NUMERIC / 1 | Count of Regulatory Fee Type Code (D61-RL). |
| D61-RL | REGULATORY FEE TYPE CODE | 2261 | D60;1 | FREE TEXT / 2 | Code identifying the type of regulatory fee.  AA=LA RS 46:2625, AB=other |
| D62-RM | REGULATORY FEE EXMPT INDICATOR | 2262 | D60;2 | SET OF CODES | Code indicating the source of the regulatory fee exempt status of the claim. 1=Payer/Plan Is Regulatory Fee Exempt, 2=Religious Organization, 3=Regulatory Fee Exempt Certificate |
| D65-RS | PATIENT REGULATORY FEE AMOUNT | 2265 | D60;5 | NUMERIC / 8 | Patient regulatory fee amount obligation or portion thereof when benefit is set up to directly pass regulatory fee onto the patient. |

The new field definitions, listed in the table(s), will be added to File 9002313.91, BPS NCPDP FIELD DEFS, in our development environment. The file will be included in our patch, so that the new fields will be included in the KIDS Build. When the KIDS Build is installed at the sites, the new fields will be automatically added to File 9002313.91 on their system.

The data dictionary for file 9002313.91, BPS NCPDP FIELD DEFS, will be sent with data to the target sites. The table below describes how the file is to be included in the KIDS Build.

|  |  |
| --- | --- |
| File Name | **BPS NCPDP FIELD DEFS** |
| Send Full or Partial DD | FULL |
| Update the Data Dictionary | YES |
| Send Security Codes | YES |
| Screen to Determine DD Update | n/a |
| Data Comes with File | YES |
| Site’s Data | OVERWRITE |
| Resolve Pointers | NO |
| May User Override Data Update | NO |
| Data List | n/a |
| Screen to Select Data | n/a |

The new fields, listed in the table(s) above, will be added to File 9002313.02, BPS CLAIMS, in our test environment, and that file will be included in our patch, so that the new fields will be included in the KIDS Build. When the KIDS Build is installed, the new fields will be automatically added to File 9002313.02.

The data dictionary for file 9002313.0201, BPS CLAIMS-TRANSACTIONS sub-file, will be sent without data to the target sites. The table below describes how the file is to be included in the KIDS Build.

|  |  |
| --- | --- |
| File Name | **BPS CLAIMS** |
| Send Full or Partial DD | PARTIAL |
| Data Dictionary Number | 9002313.0201 |
| Field Number | 2147, 2149, 2150, 2151, 2160, 2190, 2191, 2192, 2198, 2199, 2201, 2202, 2214, 2216, 2217, 2218, 2221, 2222, 2251, 2252, 2253, 2257, 2260, 2261, 2263 |
| Update the Data Dictionary | YES |
| Send Security Codes | NO |
| Screen to Determine DD Update | N/A |
| Data Comes with File | NO |

The new fields, listed in the table(s) above, will be added to File 9002313.03, BPS RESPONSES, in our test environment, and that file will be included in our patch, so that the new fields will be included in the KIDS Build. When the KIDS Build is installed, the new fields will be automatically added to File 9002313.03.

The data dictionary for file 9002313.0301, BPS RESPONSES-RESPONSES sub-file, will be sent without data to the target sites. The table below describes how the file is to be included in the KIDS Build.

|  |  |
| --- | --- |
| File Name | **BPS RESPONSES** |
| Send Full or Partial DD | PARTIAL |
| Data Dictionary Number | 9002313.0301 |
| Field Number | 931, 932, 933, 934, 935, 936, 937, 938, 943, 944, 2147, 2150, 2151, 2158, 2159, 2166, 2167, 2168, 2169, 2170, 2171, 2172, 2180, 2181, 2182, 2183, 2184, 2185, 2186, 2193, 2194, 2195, 2196, 2215, 2219, 2220, 2223, 2224, 2225, 2241, 2242, 2243, 2244, 2245, 2246, 2247, 2248, 2249, 2250, 2254, 2255, 2256, 2261, 2262, 2265 |
| Update the Data Dictionary | YES |
| Send Security Codes | NO |
| Screen to Determine DD Update | N/A |
| Data Comes with File | NO |

A new File 9002313.34, BPS NCPDP PRESCRIBER PLACE OF SERVICE, will need to be added. The information about the new file is described in the table below.

|  |  |
| --- | --- |
| File Name | BPS NCPDP PRESCRIBER PLACE OF SERVICE |
| File Number | 9002313.34 |
| Internal Global Reference | ^BPS(9002313.34, |
| Description | NCPDP field D57-RG PRESCRIBER PLACE OF SERVICE CODE  This file is used to store the possible NCPDP PRESCRIBER PLACE OF SERVICE CODES, that can be sent with a claim. No local changes should ever be made to this file. The data stored in this file is based on the NCPDP standards and are nationally distributed.  Per VHA Directive 2004-038, this file definition should not be modified. |

This new file will contain the list of Place of Service Codes for NCDPD field D57-RG Prescriber Place of Service. The codes that are being added are described in the table below.

|  |  |
| --- | --- |
| **.01 - CODE** | **.02 - NAME** |
| 1 | PHARMACY |
| 2 | TELEHEALTH |
| 3 | SCHOOL |
| 4 | HOMELESS SHELTER |
| 5 | INDIDAN HEALTH SERVICE FREE-STANDING FACILITY |
| 6 | INDIAN HEALTH SERVICE PROVIDER-BASED FACILITY |
| 7 | TRIBAL 638 FREE-STANDING FACILITY |
| 8 | TRIBAL 638 PROVIDER-BASED FACILITY |
| 9 | PRISON/CORRECTIONAL FACILITY |
| 11 | OFFICE |
| 12 | HOME |
| 13 | ASSISTED LIVING FACILITY |
| 14 | GROUP HOME |
| 15 | MOBILE UNIT |
| 16 | TEMPORARY LODGING |
| 17 | WALK-IN RETAIL HEALTH CLINIC |
| 18 | PLACE OF EMPLOYMENT-WORKSITE |
| 19 | OFF CAMPUS-OUTPATIENT HOSPITAL |
| 20 | URGENT CARE FACILITY |
| 21 | INPATIENT HOSPITAL |
| 22 | ON CAMPUS-OUTPATIENT HOSPITAL |
| 23 | EMERGENCY ROOM-HOSPITAL |
| 24 | AMBULATORY SURGICAL CENTER |
| 25 | BIRTHING CENTER |
| 26 | MILIARY TREATMENT FACILITY |
| 31 | SKILLED NURSING FACILITY |
| 32 | NURSING FACILITY |
| 33 | CUSTODIAL CARE FACILITY |
| 34 | HOSPICE |
| 41 | AMBULANCE-LAND |
| 42 | AMBULANCE-AIR OR WATER |
| 49 | INDEPENDENT CLINIC |
| 50 | FEDERALLY QUALIFIED HEALTH CENTER |
| 51 | INPATIENT PSYCHIATRIC FACILITY |
| 52 | PSYCHIATRIC FACILITY-PARTIAL HOSPITALIZATION |
| 53 | COMMUNITY MENTAL HEALTH CENTER |
| 54 | INTERMEDIATE CARE FACILITY/INDIVIDUALS WITH INTELLECTUAL DISABILITIES |
| 55 | RESIDENTAIL SUBSTANCE ABUSE TREATMENT FACILITY |
| 56 | PSYCHIATRIC RESIDENTIAL TREATMENT CENTER |
| 57 | NON-RESIDENTIAL SUBSTANCE ABUSE TREATMENT FACILITY |
| 60 | MASS IMMUNIZATION CENTER |
| 61 | COMPREHENSIVE INPATIENT REHABILITATION FACILITY |
| 62 | COMPREHENSIVE OUTPATIENT REHABILITATION FACILITY |
| 65 | END-STAGE RENAL DISEASE TREATMENT FACILITY |
| 71 | PUBLIC HEALTH CLINIC |
| 72 | RURAL HEALTH CLINIC |
| 81 | INDEPENDENT LABORATORY |
| 99 | OTHER PLACE OF SERVICE |

The new place of service codes will be added to File 9002313.34, BPS NCPDP PRESCRIBER PLACE OF SERVICE, in our development environment. That file will be included in our patch, so that the new codes will be included in the KIDS Build.

The data dictionary for the new file 9002313.34, BPS NCPDP PRESCRIBER PLACE OF SERVICE, will be sent with data to the target sites. The table below describes how the file is to be included in the KIDS Build.

|  |  |
| --- | --- |
| File Name | **BPS NCPDP PRESCRIBER PLACE OF SERVICE** |
| Send Full or Partial DD | FULL |
| Update the Data Dictionary | YES |
| Send Security Codes | YES |
| Screen to Determine DD Update | n/a |
| Data Comes with File | YES |
| Site’s Data | OVERWRITE |
| Resolve Pointers | NO |
| May User Override Data Update | NO |
| Data List | n/a |
| Screen to Select Data | n/a |

A new File 9002313.35, BPS NCPDP BENEFIT STAGE INDICATOR, will need to be added. The information about the new file is described in the table below.

|  |  |
| --- | --- |
| File Name | BPS NCPDP BENEFIT STAGE INDICATOR |
| File Number | 9002313.35 |
| Internal Global Reference | ^BPS(9002313.35, |
| Description | NCPDP field C51-9X BENEFIT STAGE INDICATOR  This file is used to store the possible NCPDP BENEFIT STAGE INDICATOR, that can be sent on a claim or returned by the payer. No local changes should ever be made to this file. The data stored in this file is based on the NCPDP standards and are nationally distributed.  Per VHA Directive 2004-038, this file definition should not be modified. |

This new file will contain the list of Benefit Stage Indicators for NCDPD field C51-9X Benefit Stage Indicator. The indicators that are being added are described in the table below.

|  |  |
| --- | --- |
| **.01 - CODE** | **.02 - NAME** |
| 1 | MEDICARE PART D DEDUCTIBLE |
| 2 | MEDICARE PART D INITIAL BENEFIT |
| 3 | MEDICARE PART D COVERAGE GAP |
| 4 | MEDICARE PART D CATASTROPHIC COVERAGE |
| 50 | NOT PAID UNDER PART D, PAID UNDER PART C BENEFIT (FOD MA-PD PLAN) |
| 51 | NOT PAID UNDER PART D, PAID UNDER PART C BENEFIT (FOR MA-PD PLAN). BENE IF A QUALIFIED MEDICARE BENE-PHARMACY SHOULD NOT ATTEMPT TO COLLECT COST-SHARE, SHOULD ATTEMPT TO BILL COB TO MEDICAID COVERAGE. |
| 61 | PART D DRUG NOT PAID BY PART D BENEFIT. PAID AS OR UNDER A CO-ADMINISTERED BENEFIT ONLY. |
| 62 | NON-PART D/NON-QUAL DRUG NOT PD BY PART D PLAN BENEFIT |
| 63 | NON-PART D/NON-QUAL DRUG NO PAID BY PART D PLAN BENEFIT. PAID UNDER MEDICARE BENE ONLY OF THE MMP PLAN. |
| 70 | PART D DRUG NOT PAID BY PART D PLAN BENEFIT, PAID BY THE BENEFICIARY UNDER PLAN-SPONSORED NEGOTIATED PRICING. |
| 80 | NON-PART D/NON-QUAL DRUG NOT PAID BY PART D PLAN BENEFIT, HOSPICE BENEFIT OR ANY OTHER COMPONENT OF MEDICARE; PD BY THE BENE UNDER PLAN-SPON NEGOTIATED PRICING. |
| 90 | ENHANCE OR OTC DRUG (PDE VALUE OF E/O) NOT APPLICABLE TO THE PART D DRUG SPEND, BUT IS COVERED BY THE PART D PLAN. |

The new place of service codes will be added to File 9002313.35, BPS NCPDP BENEFIT STAGE INDICATOR, in our development environment. That file will be included in our patch, so that the new codes will be included in the KIDS Build.

The data dictionary for the new file 9002313.35, BPS NCPDP BENEFIT STAGE INDICATOR will be sent with data to the target sites. The table below describes how the file is to be included in the KIDS Build.

|  |  |
| --- | --- |
| File Name | **BPS NCPDP BENEFIT STAGE INDICATOR** |
| Send Full or Partial DD | FULL |
| Update the Data Dictionary | YES |
| Send Security Codes | YES |
| Screen to Determine DD Update | n/a |
| Data Comes with File | YES |
| Site’s Data | OVERWRITE |
| Resolve Pointers | NO |
| May User Override Data Update | NO |
| Data List | n/a |
| Screen to Select Data | n/a |

A new File 9002313.36, BPS NCPDP LTPAC DISPENSE FREQUENCY, will need to be added. The information about the new file is described in the table below.

|  |  |
| --- | --- |
| File Name | BPS NCPDP |
| File Number | 9002313.36 |
| Internal Global Reference | ^BPS(9002313.36, |
| Description | NCPDP field C91-KK LTPAC DISPENSE FREQUENCY  This file is used to store the possible NCPDP LTPAC DISPENSE FREQUENCY, that can be sent on a claim. No local changes should ever be made to this file. The data stored in this file is based on the NCPDP standards and are nationally distributed.  Per VHA Directive 2004-038, this file definition should not be modified. |

This new file will contain the list of LTPAC Dispense Frequency’s for NCDPD field C91-KK LTPAC Dispense Frequency. The frequency’s that are being added are described in the table below.

|  |  |
| --- | --- |
| **.01 - CODE** | **.02 - DESCRIPTION** |
| 1 | Medication dispensed in a day-supply increment equal to the billed days supply |
| 2 | 7 days - dispenses medication in 7-day supplies |
| 3 | 4 days - dispenses medication in 4-day supplies |
| 4 | 3 days - dispenses medication in 3-day supplies |
| 5 | 2 days - dispenses medication in 2-day supplies |
| 6 | 1 day - dispenses medication in 1-day supplies |
| 7 | 4-3 days - dispenses medication in 4-day, then 3-day supplies |
| 8 | 2-2-3 days - dispenses medication in 2-day, then 2-day, then 3-day supplies |
| 9 | Daily and 3-day weekend - dispensed daily during the week and combines multiple days dispensing for weekends |
| 10 | Per shift dispensing (multiple med passes) |
| 11 | Per med pass dispensing |
| 12 | PRN on demand |
| 13 | 7-day or less cycle not otherwise represented |
| 14 | 14 days dispensing - dispenses medication in 14-day supplies |
| 15 | 8-14-Day dispensing cycle not otherwise represented |

The new place of service codes will be added to File 9002313.36, BPS NCPDP LTPAC DISPENSE FREQUENCY, in our development environment. That file will be included in our patch, so that the new codes will be included in the KIDS Build.

The data dictionary for the new file 9002313.36, BPS NCPDP LTPAC DISPENSE FREQUENCY will be sent with data to the target sites. The table below describes how the file is to be included in the KIDS Build.

|  |  |
| --- | --- |
| File Name | **BPS NCPDP LTPAC DISPENSE FREQUENCY** |
| Send Full or Partial DD | FULL |
| Update the Data Dictionary | YES |
| Send Security Codes | YES |
| Screen to Determine DD Update | n/a |
| Data Comes with File | YES |
| Site’s Data | OVERWRITE |
| Resolve Pointers | NO |
| May User Override Data Update | NO |
| Data List | n/a |
| Screen to Select Data | n/a |

A new File 9002313.37, BPS NCPDP PATIENT PAY COMPONENT QUALIFIER, will need to be added. The information about the new file is described in the table below.

|  |  |
| --- | --- |
| File Name | BPS NCPDP |
| File Number | 9002313.37 |
| Internal Global Reference | ^BPS(9002313.37, |
| Description | NCPDP field C95-KQ PATIENT PAY COMPONENT QUALIFIER  This file is used to store the possible NCPDP PATIENT PAY COMPONENT QUALIFIER, that can be returned by the payer. No local changes should ever be made to this file. The data stored in this file is based on the NCPDP standards and are nationally distributed.  Per VHA Directive 2004-038, this file definition should not be modified. |

This new file will contain the list of Patient Pay Component Qualifiers for NCDPD field C95-KQ Patient Pay Component Qualifier. The qualifiers that are being added are described in the table below.

|  |  |
| --- | --- |
| **.01 - CODE** | **.02 - NAME** |
| 1 | Amount Applied to Periodic Deductible |
| 2 | Amount Attributed to Product Selection/Brand Drug |
| 3 | Amount Attributed to Percentage Tax |
| 4 | Amount Exceeding Periodic Benefit Maximum |
| 5 | Amount of Copay |
| 7 | Amount of Coinsurance |
| 8 | Amount Attributed to Product Selection/Non-Preferred Formulary Selection |
| 9 | Amount Attributed to Health Plan Assistance Amount |
| 10 | Amount Attributed to Provider Network Selection |
| 11 | Amount Attributed to Product Selection/Brand Non-Preferred Formulary Selection |
| 12 | Amount Attributed to Coverage Gap |
| 13 | Amount Attributed to Processor Fee |
| 14 | Amount Attributed to Grace Period |
| 15 | Amount Attributed to Catastrophic Benefit |
| 16 | Amount Attributed to Unbalanced Patient Pay OPPRA |
| 17 | Amount Attributed to Regulatory Fee |

The new place of service codes will be added to File 9002313.37, BPS NCPDP PATIENT PAY COMPONENT QUALIFIER, in our development environment. That file will be included in our patch, so that the new codes will be included in the KIDS Build.

The data dictionary for the new file 9002313.37, BPS NCPDP PATIENT PAY COMPONENT QUALIFIER, will be sent with data to the target sites. The table below describes how the file is to be included in the KIDS Build.

|  |  |
| --- | --- |
| File Name | **BPS NCPDP PATIENT PAY COMPONENT QUALIFIER** |
| Send Full or Partial DD | FULL |
| Update the Data Dictionary | YES |
| Send Security Codes | YES |
| Screen to Determine DD Update | n/a |
| Data Comes with File | YES |
| Site’s Data | OVERWRITE |
| Resolve Pointers | NO |
| May User Override Data Update | NO |
| Data List | n/a |
| Screen to Select Data | n/a |

A new File 9002313.38, BPS NCPDP OTHER PAYER PROGRAM TYPE, will need to be added. The information about the new file is described in the table below.

|  |  |
| --- | --- |
| File Name | BPS NCPDP |
| File Number | 9002313.38 |
| Internal Global Reference | ^BPS(9002313.38, |
| Description | NCPDP field C47-9T OTHER PAYER ADJUDICATED PROGRAM TYPE  This file is used to store the possible NCPDP OTHER PAYER ADJUDICATED PROGRAM TYPE, that can be returned by the payer. No local changes should ever be made to this file. The data stored in this file is based on the NCPDP standards and are nationally distributed.  Per VHA Directive 2004-038, this file definition should not be modified. |

This new file will contain the list of Other Payer Adjudicated Program Types for NCDPD field C47-9T Other Payer Adjudicated Program Type. The qualifiers that are being added are described in the table below.

|  |  |
| --- | --- |
| **.01 - CODE** | **.02 - DESCRIPTION** |
| 0 | Cash |
| 1 | Medicaid Title XIX |
| 2 | Medicare |
| 3 | Commercial |
| 4 | Workers Compensation |
| 5 | Self-Pay: Discount Program |
| 6 | Manufacturer Sponsored Patient Pay Reduction Program |
| 7 | Manufacturer Free Product |
| 8 | Veterans Health Administration (VA) |
| 9 | Unknown |
| 10 | Hospice - Non Medicare |
| 11 | Medicaid Managed Care |
| 12 | Medicare Part A |
| 13 | Medicare Advantage |
| 14 | Medicare Part D PDP |
| 15 | Self Pay: Cash |
| 16 | Medicare Part B |
| 17 | Indian Health Services |
| 18 | ADAP/Ryan White |
| 19 | Black Lung |
| 20 | Casualty Insurance |
| 21 | CHIP Title XXI |
| 22 | Health Marketplace Exchange Qualified Health Plan |
| 23 | HRSA 340B Indigent Program |
| 24 | Independent Charity Patient Assistance Program |
| 25 | Manufacturer Patient Assistance Program |
| 26 | Medicare - Medicaid Plan (MMP) |
| 27 | SPAP |
| 28 | Tricare |
| 29 | Other Federal Payer |
| 30 | Programs of All-Inclusive Care for the Elderly (PACE) |
| 99 | Other |

The new place of service codes will be added to File 9002313.38, BPS NCPDP OTHER PAYER PROGRAM TYPE, in our development environment. That file will be included in our patch, so that the new codes will be included in the KIDS Build.

The data dictionary for the new file 9002313.38, BPS NCPDP OTHER PAYER PROGRAM TYPE, will be sent with data to the target sites. The table below describes how the file is to be included in the KIDS Build.

|  |  |
| --- | --- |
| File Name | **BPS NCPDP OTHER PAYER ADJ PROGRAM TYPE** |
| Send Full or Partial DD | FULL |
| Update the Data Dictionary | YES |
| Send Security Codes | YES |
| Screen to Determine DD Update | n/a |
| Data Comes with File | YES |
| Site’s Data | OVERWRITE |
| Resolve Pointers | NO |
| May User Override Data Update | NO |
| Data List | n/a |
| Screen to Select Data | n/a |

MEDINFO^BPSOSCD will need to be modified to include the PRESCRIBER DEA NUMBER (D01-KV) field in the BPS(“RX”) array. The Get code for field D01-KV will use it when the claim is being built.

| Subroutine Name | **MEDINFO^BPSOSCD** | | | |
| --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify | Delete | No Change |
| Current Logic | | | | |
| BPSOSCD ;BHAM ISC/FCS/DRS/DLF - Set BPS() "RX" nodes for current medication ;06/01/2004  ;;1.0;E CLAIMS MGMT ENGINE;\*\*1,3,2,5,7,8,10,11,15,19,20,23\*\*;JUN 2004;Build 27  ;;Per VA Directive 6402, this routine should not be modified.  ;  . . .  ;MEDINFO, Set BPS("RX)" nodes for current medication  ; Called from BPSOSCA for every transaction in the multiple  ; IEN59 = IEN in BPS TRANSACTION (#9002313.59)  ; IEN5902 = IEN for Insurance multiple of BPS Transactions  ; MEDN = Index number of medication being processed  ; BPS array shared by all of the BPSOSC\* routines, created in BPSOSCA  ; VAINFO created in BPSOSCB  MEDINFO(IEN59,IEN5902,MEDN) ;  ; Verify Parameters  I $G(IEN59)="" Q  I $G(IEN5902)="" Q  I $G(MEDN)="" Q  ;  N %,BPS0,DRUGIEN,IENS,J,NDC,NPI,OSITEIEN,PRICING,PROVIEN,RTN,RXI,RXIEN,RXRFIEN,VANATURE,VAOIEN,X,ADFEE  ;  ;RXIEN=Rx IEN, RXRFIEN=Fill Number, IENS=FileMan style IENS  S BPS0=$G(^BPST(IEN59,1)),RXIEN=$P(BPS0,U,11),RXRFIEN=$P(BPS0,U,1),IENS=IEN5902\_","\_IEN59\_","  . . .  ; Get Provider Info  S PROVIEN=+$$RXAPI1^BPSUTIL1(RXIEN,4,"I")  S BPS("RX",MEDN,"Prescriber IEN")=PROVIEN  I PROVIEN'="" D  .S X=$$GET1^DIQ(200,PROVIEN,.01)  .D NAMECOMP^XLFNAME(.X)  .S BPS("RX",MEDN,"Prescriber Last Name")=X("FAMILY")  .S BPS("RX",MEDN,"Prescriber First Name")=X("GIVEN") ; NCPDP field 364-2J  .S BPS("RX",MEDN,"Prescriber Phone #")=$$ACPHONE^IBNCPDPI ; DBIA 4721, Agent Cashier Phone Number  .S BPS("RX",MEDN,"Prescriber Billing Location")=""  .S NPI=$$NPI^BPSNPI("Individual\_ID",+PROVIEN)  .I NPI<0 S NPI=""  .S BPS("RX",MEDN,"Prescriber NPI")=$P(NPI,U)  .S BPS("RX",MEDN,"Primary Care Provider NPI")=$P(NPI,U)  .S BPS("RX",MEDN,"Provider NPI")=$P(NPI,U)  .;  .S X=$$PRVADRS(IEN59,PROVIEN) ; provide address info  .S BPS("RX",MEDN,"Prescriber Street Address")=$P(X,U)\_$S($P(X,U,5)]"":" ",1:"")\_$P(X,U,5) ; NCPDP field 365-2K  .S BPS("RX",MEDN,"Prescriber Street Address Line 1")=$P(X,U) ; NCPDP field B27-7U  .S BPS("RX",MEDN,"Prescriber Street Address Line 2")=$P(X,U,5) ; NCPDP field B28-8U  .S BPS("RX",MEDN,"Prescriber City Address")=$P(X,U,2) ; NCPDP field 366-2M  .S BPS("RX",MEDN,"Prescriber State/Province Address")=$P(X,U,3) ; NCPDP field 367-2N  .S BPS("RX",MEDN,"Prescriber Zip/Postal Zone")=$TR($P(X,U,4)," -") ; NCPDP field 368-2P  .S BPS("RX",MEDN,"Prescriber Country")=$$COUNTRY($P(X,U,3),$P(X,U,6)) ;NCPDP field B42-3C  ;  ; Stop if Eligibility as we do not need any of the claim data below  I BPS("Transaction Code")="E1" Q  ;  . . .  ;  ;;;  D LOG^BPSOSL(IEN59,"BPSOSCD-MEDINFO-Contents of BPS") ;;; Remove after testing ;;;  D LOGARRAY^BPSOSL(IEN59,"BPS",1000) ;;; Remove after testing ;;;  ;;;  Q | | | | |
| Modified Logic (Changes are highlighted) | | | | |
| BPSOSCD ;BHAM ISC/FCS/DRS/DLF - Set BPS() "RX" nodes for current medication ;06/01/2004  ;;1.0;E CLAIMS MGMT ENGINE;\*\*1,3,2,5,7,8,10,11,15,19,20,23,24\*\*;JUN 2004;Build 27  ;;Per VA Directive 6402, this routine should not be modified.  ;  . . .    ;MEDINFO, Set BPS("RX)" nodes for current medication  ; Called from BPSOSCA for every transaction in the multiple  ; IEN59 = IEN in BPS TRANSACTION (#9002313.59)  ; IEN5902 = IEN for Insurance multiple of BPS Transactions  ; MEDN = Index number of medication being processed  ; BPS array shared by all of the BPSOSC\* routines, created in BPSOSCA  ; VAINFO created in BPSOSCB  MEDINFO(IEN59,IEN5902,MEDN) ;  ; Verify Parameters  I $G(IEN59)="" Q  I $G(IEN5902)="" Q  I $G(MEDN)="" Q  ;  N %,BPS0,DRUGIEN,IENS,J,NDC,NPI,OSITEIEN,PRICING,PROVIEN,RTN,RXI,RXIEN,RXRFIEN,VANATURE,VAOIEN,X,ADFEE  ;  ;RXIEN=Rx IEN, RXRFIEN=Fill Number, IENS=FileMan style IENS  S BPS0=$G(^BPST(IEN59,1)),RXIEN=$P(BPS0,U,11),RXRFIEN=$P(BPS0,U,1),IENS=IEN5902\_","\_IEN59\_","  ;  . . .  ; Get Provider Info  S PROVIEN=+$$RXAPI1^BPSUTIL1(RXIEN,4,"I")  S BPS("RX",MEDN,"Prescriber IEN")=PROVIEN  I PROVIEN'="" D  .S X=$$GET1^DIQ(200,PROVIEN,.01)  .D NAMECOMP^XLFNAME(.X)  .S BPS("RX",MEDN,"Prescriber Last Name")=X("FAMILY")  .S BPS("RX",MEDN,"Prescriber First Name")=X("GIVEN") ; NCPDP field 364-2J  .S BPS("RX",MEDN,"Prescriber Phone #")=$$ACPHONE^IBNCPDPI ; DBIA 4721, Agent Cashier Phone Number  .S BPS("RX",MEDN,"Prescriber Billing Location")=""  .S NPI=$$NPI^BPSNPI("Individual\_ID",+PROVIEN)  .I NPI<0 S NPI=""  .S BPS("RX",MEDN,"Prescriber NPI")=$P(NPI,U)  .S BPS("RX",MEDN,"Primary Care Provider NPI")=$P(NPI,U)  .S BPS("RX",MEDN,"Provider NPI")=$P(NPI,U)  .;  .S BPS("RX",MEDN,"Prescriber DEA")=$$GET1^DIQ(200,PROVIEN,53.2) ; NCPDP field D01-KV  .S X=$$PRVADRS(IEN59,PROVIEN) ; provide address info  .S BPS("RX",MEDN,"Prescriber Street Address")=$P(X,U)\_$S($P(X,U,5)]"":" ",1:"")\_$P(X,U,5) ; NCPDP field 365-2K  .S BPS("RX",MEDN,"Prescriber Street Address Line 1")=$P(X,U) ; NCPDP field B27-7U  .S BPS("RX",MEDN,"Prescriber Street Address Line 2")=$P(X,U,5) ; NCPDP field B28-8U  .S BPS("RX",MEDN,"Prescriber City Address")=$P(X,U,2) ; NCPDP field 366-2M  .S BPS("RX",MEDN,"Prescriber State/Province Address")=$P(X,U,3) ; NCPDP field 367-2N  .S BPS("RX",MEDN,"Prescriber Zip/Postal Zone")=$TR($P(X,U,4)," -") ; NCPDP field 368-2P  .S BPS("RX",MEDN,"Prescriber Country")=$$COUNTRY($P(X,U,3),$P(X,U,6)) ;NCPDP field B42-3C  ;  ; Stop if Eligibility as we do not need any of the claim data below  I BPS("Transaction Code")="E1" Q  ;  . . .  ;  ;;;  D LOG^BPSOSL(IEN59,"BPSOSCD-MEDINFO-Contents of BPS") ;;; Remove after testing ;;;  D LOGARRAY^BPSOSL(IEN59,"BPS",1000) ;;; Remove after testing ;;;  ;;;  Q | | | | |

A new subroutine FLDD02^BPSOSSG will be created to be used by the Set code for field TOTAL PRESCRIBED QUANTITY (D02-K) in BPS NCPDP FIELD DEFS field 2202.

| Subroutine Name | **FLDD02^BPSOSSG** | | | |
| --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify | Delete | No Change |
| Current Logic | | | | |
| n/a – new subroutine | | | | |
| Modified Logic (Changes are highlighted) | | | | |
| BPSOSSG ;BHAM ISC/SD/lwj/FLS - Special gets for formats ;06/01/2004  ;;1.0;E CLAIMS MGMT ENGINE;\*\*1,5,10,11,20,24\*\*;JUN 2004;Build 27  ;;Per VA Directive 6402, this routine should not be modified.  ;  . . .  FLDD02 ; Total Prescribed Quantity Remaining field (D02-KW)  ; called by SET CODE in BPS NCPDP Field DEFS for field 2202 (D02-KW)  ;  I '$G(BPS(9002313.02)) S BPS(9002313.02)=$G(BPS02)  Q:'$G(BPS(9002313.02)) ; must have BPS Claims IEN  Q:'$G(BPS(9002313.0201)) ; must have Transaction subfile IEN  ;  N I,PREVFILLS,REFILLS,RTS,RXIEN,TOTALDISP,TOTALQTY,QTY  ;  S REFILLS=$G(BPS("RX",BPS(9002313.0201),"# Refills"))  S QTY=$G(BPS("RX",BPS(9002313.0201),"Quantity"))  S TOTALQTY=QTY\*(REFILLS+1) ; Total quantity for the prescription  S PREVFILLS=$G(BPS("RX",BPS(9002313.0201),"Refill #"))  ;  ; Determine if any previous fills were returned to stock.  S RXIEN=$G(BPS("RX",BPS(9002313.0201),"RX IEN"))  I RXIEN S RTS=0,I=0 D  . F S I=$O(^PSRX(RXIEN,"RTS",I)) Q:'I S RTS=RTS+1  ;  ; Subtract and return to stock fills (RTS) from the number of previous fills (PREVFILLS).  S TOTALDISP=(PREVFILLS-RTS)\*QTY ; Total dispensed for all previous fills  ; D02-KW Total Prescribed Quantity Remaining  S BPS("X")=TOTALQTY-TOTALDISP  S $P(^BPSC(BPS(9002313.02),400,BPS(9002313.0201),"D00"),U,2)="KW"\_$$NFF^BPSECFM(BPS("X"),10)  Q | | | | |

| Subroutine Name | **NEWCLAIM^BPSOSCE** | | | |
| --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify | Delete | No Change |
| Current Logic | | | | |
| BPSOSCE ;BHAM ISC/FCS/DRS/DLF - New entry in 9002313.02 ;06/01/2004  ;;1.0;E CLAIMS MGMT ENGINE;\*\*1,5,7,8,10,11,15,19\*\*;JUN 2004;Build 18  ;;Per VA Directive 6402, this routine should not be modified.  ;  . . .  NEWCLAIM(START,END,TOTAL) ; function, returns null on success, else error  ;  N BPSIEN,CLAIMID,COUNT,DA,DIC,DIK,DLAYGO,ERROR,INDEX,NODE0,ROU,SEG,X,Y  S ROU=$T(+0),START=+$G(START),END=+$G(END),TOTAL=+$G(TOTAL)  ;  ;Create new record in Claim Submission File (9002313.02)  ; try for exclusive access for 1 min. before logging error  F  L +^XTMP(ROU,"NEWCLAIM"):60 Q:$T  D  .N A S A=$$IMPOSS^BPSOSUE("L","RTI","Single-threaded routine",,,ROU)  ; Generate Claim ID  S CLAIMID=$$CLAIMID^BPSECX1($G(BPS("RX",START,"IEN59")))  I CLAIMID="" D  .S ERROR="320^VA Claim ID not created"  .D LOG(ROU\_"-Failed to create Claim ID")  ;  ; Create claim record  D:'$G(ERROR)  .S DLAYGO=9002313.02,DIC="^BPSC(",DIC(0)="LXZ",X=CLAIMID  .D ^DIC Q:Y>0 ; less than zero is error  .S ERROR="321^Failed to create claim record"  .D LOG(ROU\_"-Failed to create an entry in file 9002313.02")  ;  L -^XTMP(ROU,"NEWCLAIM")  ;  Q:$G(ERROR) ERROR  ;  ; Update BPS and Log it  S BPS(9002313.02)=+Y  ; Needed for Turn-Around Stats - Do NOT delete/alter!!  D LOG(ROU\_"-Created claim ID "\_CLAIMID\_" (IEN "\_BPS(9002313.02)\_")")  ;  ; Update zero node of the claim  S NODE0=$G(^BPSC(BPS(9002313.02),0))  S $P(NODE0,U,2)=$G(BPS("NCPDP","IEN")) ; Electronic Payor (Payer Sheet)  S $P(NODE0,U,4)=2 ; Transmit Flag - 2 is 'Yes (Point of Sale)'  S $P(NODE0,U,6)=$$NOWFM^BPSOSU1() ; Created On  S ^BPSC(BPS(9002313.02),0)=NODE0  ;  ; Update Patient Name  S $P(^BPSC(BPS(9002313.02),1),U,1)=$G(BPS("Patient","Name"))  S $P(^BPSC(BPS(9002313.02),1),U,4)=$G(BPS("Insurer","IEN"))  ; Update TRANSACTION field  S $P(^BPSC(BPS(9002313.02),0),U,8)=$G(BPS("RX",START,"IEN59"))  ;  ; Count of meds in claim  S BPS("Transaction Count")=TOTAL  ;  ; Process the 'non-multiple' segments (Header, Patient, Cardholder)  F SEG=100:10:120 D XLOOP^BPSOSCF(BPS("NCPDP","IEN"),SEG)  ;  ; zero node for MEDICATIONS SUB-FIELD (#9002313.0201)  S:'$D(^BPSC(BPS(9002313.02),400,0)) ^(0)="^9002313.0201PA^^"  S COUNT=0 F INDEX=START:1:END D  .; Create zero node for entry in multiple  .S COUNT=COUNT+1,NODE0=""  .S $P(NODE0,U)=INDEX,$P(NODE0,U,4)=$G(BPS("RX",INDEX,"Drug Name")),$P(NODE0,U,5)=$G(BPS("RX",INDEX,"RX IEN"))  .S ^BPSC(BPS(9002313.02),400,INDEX,0)=NODE0  .S BPS(9002313.0201)=INDEX  .; Process entries in medication multiple  .F SEG=130:10:300 D XLOOP^BPSOSCF(BPS("NCPDP","IEN"),SEG,INDEX) ; BPS\*1\*19 - add Intermediary and Last Known 4Rx segments  .;  .; Update the indices  .S ^BPSC(BPS(9002313.02),400,"B",INDEX,INDEX)=""  .; Update top-level node of the multiple  .S NODE0=$G(^BPSC(BPS(9002313.02),400,0))  .S $P(NODE0,U,3)=COUNT,$P(NODE0,U,4)=COUNT,^BPSC(BPS(9002313.02),400,0)=NODE0  ;  ; Cross-Reference Claim Submission Record  S DIK="^BPSC(",DA=BPS(9002313.02) D IX1^DIK  ;  Q ""  ; Return null on success | | | | |
| Modified Logic (Changes are highlighted) | | | | |
| BPSOSCE ;BHAM ISC/FCS/DRS/DLF - New entry in 9002313.02 ;06/01/2004  ;;1.0;E CLAIMS MGMT ENGINE;\*\*1,5,7,8,10,11,15,19,24\*\*;JUN 2004;Build 18  ;;Per VA Directive 6402, this routine should not be modified.  ;  . . .  NEWCLAIM(START,END,TOTAL) ; function, returns null on success, else error  ;  N BPSIEN,CLAIMID,COUNT,DA,DIC,DIK,DLAYGO,ERROR,INDEX,NODE0,ROU,SEG,X,Y  S ROU=$T(+0),START=+$G(START),END=+$G(END),TOTAL=+$G(TOTAL)  ;  ;Create new record in Claim Submission File (9002313.02)  ; try for exclusive access for 1 min. before logging error  F  L +^XTMP(ROU,"NEWCLAIM"):60 Q:$T  D  .N A S A=$$IMPOSS^BPSOSUE("L","RTI","Single-threaded routine",,,ROU)  ; Generate Claim ID  S CLAIMID=$$CLAIMID^BPSECX1($G(BPS("RX",START,"IEN59")))  I CLAIMID="" D  .S ERROR="320^VA Claim ID not created"  .D LOG(ROU\_"-Failed to create Claim ID")  ;  ; Create claim record  D:'$G(ERROR)  .S DLAYGO=9002313.02,DIC="^BPSC(",DIC(0)="LXZ",X=CLAIMID  .D ^DIC Q:Y>0 ; less than zero is error  .S ERROR="321^Failed to create claim record"  .D LOG(ROU\_"-Failed to create an entry in file 9002313.02")  ;  L -^XTMP(ROU,"NEWCLAIM")  ;  Q:$G(ERROR) ERROR  ;  ; Update BPS and Log it  S BPS(9002313.02)=+Y  ; Needed for Turn-Around Stats - Do NOT delete/alter!!  D LOG(ROU\_"-Created claim ID "\_CLAIMID\_" (IEN "\_BPS(9002313.02)\_")")  ;  ; Update zero node of the claim  S NODE0=$G(^BPSC(BPS(9002313.02),0))  S $P(NODE0,U,2)=$G(BPS("NCPDP","IEN")) ; Electronic Payor (Payer Sheet)  S $P(NODE0,U,4)=2 ; Transmit Flag - 2 is 'Yes (Point of Sale)'  S $P(NODE0,U,6)=$$NOWFM^BPSOSU1() ; Created On  S ^BPSC(BPS(9002313.02),0)=NODE0  ;  ; Update Patient Name  S $P(^BPSC(BPS(9002313.02),1),U,1)=$G(BPS("Patient","Name"))  S $P(^BPSC(BPS(9002313.02),1),U,4)=$G(BPS("Insurer","IEN"))  ; Update TRANSACTION field  S $P(^BPSC(BPS(9002313.02),0),U,8)=$G(BPS("RX",START,"IEN59"))  ;  ; Count of meds in claim  S BPS("Transaction Count")=TOTAL  ;  ; Process the 'non-multiple' segments (Header, Patient, Cardholder)  F SEG=100:10:120 D XLOOP^BPSOSCF(BPS("NCPDP","IEN"),SEG)  ;  ; zero node for MEDICATIONS SUB-FIELD (#9002313.0201)  S:'$D(^BPSC(BPS(9002313.02),400,0)) ^(0)="^9002313.0201PA^^"  S COUNT=0 F INDEX=START:1:END D  .; Create zero node for entry in multiple  .S COUNT=COUNT+1,NODE0=""  .S $P(NODE0,U)=INDEX,$P(NODE0,U,4)=$G(BPS("RX",INDEX,"Drug Name")),$P(NODE0,U,5)=$G(BPS("RX",INDEX,"RX IEN"))  .S ^BPSC(BPS(9002313.02),400,INDEX,0)=NODE0  .S BPS(9002313.0201)=INDEX  .; Process entries in medication multiple  .F SEG=130:10:300 D XLOOP^BPSOSCF(BPS("NCPDP","IEN"),SEG,INDEX) ; BPS\*1\*19 - add Intermediary and Last Known 4Rx segments  .; Process entries in "D00" node of the Claim  .; Prescriber DEA Number D01-KV  .S BPSIEN=$O(^BPSF(9002313.91,"C","PRESCRIBER DEA NUMBER",""))  .I BPSIEN'="" D XFLDCODE^BPSOSCF(150,BPSIEN,"GFS") ; calls the GET, FORMAT and SET for NCPDP Field in file 9002313.91  .; Total Prescribed Qty Remaining D02-KW  .S BPSIEN=$O(^BPSF(9002313.91,"C","TOTAL PRESCRIBED QTY REMAINING",""))  .D FLDD02^BPSOSSG  .;  .; Update the indices  .S ^BPSC(BPS(9002313.02),400,"B",INDEX,INDEX)=""  .; Update top-level node of the multiple  .S NODE0=$G(^BPSC(BPS(9002313.02),400,0))  .S $P(NODE0,U,3)=COUNT,$P(NODE0,U,4)=COUNT,^BPSC(BPS(9002313.02),400,0)=NODE0  ;  ; Cross-Reference Claim Submission Record  S DIK="^BPSC(",DA=BPS(9002313.02) D IX1^DIK  ;  Q ""  ; Return null on success | | | | |

The new field Patient Pay Component Amount (C93-KN) will be added to the Third Party Joint Inquiry [IBJ THIRD PARTY JOINT INQUIRY] under the Patient Responsibility Amounts section.

| Subroutine Name | **INIT^IBJTRX** | | | |
| --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify | Delete | No Change |
| Current Logic | | | | |
| IBJTRX ;ALB/ESG - TPJI ePharmacy ECME claim information ;22-Oct-2010  ;;2.0;INTEGRATED BILLING;\*\*435,452,494,521\*\*;21-MAR-94;Build 33  ;;Per VA Directive 6402, this routine should not be modified.  ;  . . .  INIT ; -- init variables and list array  N IBM1,ECME,ECMEAP,RXORG,DOCIEN,PHARMNPI,DOCNPI,RESPIEN,ZR,RSPSUB,ZM,BPSM,BPSMCOB,IBLINE,ZC,ZCTOT,ZCN  N IBZ,IBRXDATA,IBRXIEN,IBRXFILL,IBCOBN,IBBPS,IB0,IBS,IBHPD,IBVL,IBCPY,IBM0  K ^TMP("IBJTRX",$J)  S VALMCNT=0  ;  S IBZ=+$O(^IBA(362.4,"C",IBIFN,0))  S IBRXDATA=$G(^IBA(362.4,IBZ,0))  S IBRXIEN=+$P(IBRXDATA,U,5) ; RX ien ptr file 52  S IBRXFILL=+$P(IBRXDATA,U,10) ; rx fill#  S IBCOBN=+$$COBN^IBCEF(IBIFN) ; current payer sequence #  S IBBPS=$$CLAIM^BPSBUTL(IBRXIEN,IBRXFILL,IBCOBN) ; DBIA 4719  ;  S IBM1=$G(^DGCR(399,IBIFN,"M1"))  S IB0=$G(^DGCR(399,IBIFN,0))  S IBS=$G(^DGCR(399,IBIFN,"S"))  S ECME=$P($P(IBM1,U,8),";",1) ; ECME#  S ECMEAP=$P(IBM1,U,9) ; ECME approval number  S RXORG=$$RXSITE^IBCEF73A(IBIFN) ; pharmacy file 4 ien  S DOCIEN=$$RXAPI1^IBNCPUT1(IBRXIEN,4,"I") ; ien of doctor who wrote the Rx (52,4)  S (PHARMNPI,DOCNPI)=""  I RXORG S PHARMNPI=$P($$NPI^XUSNPI("Organization\_ID",RXORG),U,1) ; pharmacy NPI  I DOCIEN S DOCNPI=$P($$NPI^XUSNPI("Individual\_ID",DOCIEN),U,1) ; doctor NPI  I PHARMNPI'>0 S PHARMNPI="No NPI on file"  I DOCNPI'>0 S DOCNPI="No NPI on file"  ;  S RESPIEN=+$P(IBBPS,U,3) ; BPS response file ien  I RESPIEN D  . ; IB\*2.0\*521 - add HPID from response to TPJI screen  . S IBM0=$G(^DGCR(399,IBIFN,"M")),IBCPY=$S($P(IB0,U,21)="P":$P(IBM0,U),$P(IB0,U,21)="S":$P(IBM0,U,2),1:$P(IBM0,"^",3))  . I $P($G(^BPSR(RESPIEN,560)),U,8)="01" S IBHPD=$P($G(^BPSR(RESPIEN,560)),U,9) S IBVL=$$HOD^IBCNHUT1(IBHPD,IBCPY)  . S ZR=RESPIEN\_","  . S RSPSUB=+$O(^BPSR(RESPIEN,1000,0))  . I RSPSUB D  .. S ZM=RSPSUB\_","\_RESPIEN\_","  .. D GETS^DIQ(9002313.0301,ZM,"129;133:137;505;506;507;509;517:520;571;572","IEN","BPSM") ; get selected $ amount fields  .. D GETS^DIQ(9002313.0301,ZM,"355.01\*","IEN","BPSMCOB") ; get cob/other payer data fields  .. Q  . Q  ;  S IBLINE=$$SETL("",ECME,"ECME No",25,11,1)  S IBLINE=$$SETL(IBLINE,PHARMNPI,"Pharmacy NPI",14,15,40)  D SET(IBLINE) . . .   ;  D SET(" ")  S IBLINE=$$SETL("",,"Patient Responsibility Amounts",,31,1)  D SET(IBLINE,"3;2;30")  ;  S IBLINE=$$SETL("",$$AMT($G(BPSM(9002313.0301,ZM,517,"E"))),"Deductible",10,13,1)  S IBLINE=$$SETL(IBLINE,$$AMT($G(BPSM(9002313.0301,ZM,572,"E"))),"Coinsurance",10,13,27)  S IBLINE=$$SETL(IBLINE,$$AMT($G(BPSM(9002313.0301,ZM,518,"E"))),"Amount of Copay",9,18,52)  D SET(IBLINE)  ;  S IBLINE=$$SETL("",$$AMT($G(BPSM(9002313.0301,ZM,137,"E"))),"Coverage Gap",10,13,1)  S IBLINE=$$SETL(IBLINE,$$AMT($G(BPSM(9002313.0301,ZM,571,"E"))),"Processor Fee",10,13,27)  S IBLINE=$$SETL(IBLINE,$$AMT($G(BPSM(9002313.0301,ZM,520,"E"))),"Exceed Benefit Max",9,18,52)  D SET(IBLINE)  ;  S IBLINE=$$SETL("",$$AMT($G(BPSM(9002313.0301,ZM,129,"E"))),"Health Plan-funded Assistance Amount",15,39,1)  D SET(IBLINE)  ;  D SET(" ")  S IBLINE=$$SETL("",,"Product Selection Amounts",,26,1)  D SET(IBLINE,"3;2;25")  ;  . . .   ; INITX ;  D SET(" "),SET(" ")  Q | | | | |
| Modified Logic (Changes are highlighted) | | | | |
| IBJTRX ;ALB/ESG - TPJI ePharmacy ECME claim information ;22-Oct-2010  ;;2.0;INTEGRATED BILLING;\*\*435,452,494,521,617\*\*;21-MAR-94;Build 33  ;;Per VA Directive 6402, this routine should not be modified.  ;  . . .  INIT ; -- init variables and list array  N IBM1,ECME,ECMEAP,RXORG,DOCIEN,PHARMNPI,DOCNPI,RESPIEN,ZR,RSPSUB,ZM,BPSM,BPSMCOB,IBLINE,ZC,ZCTOT,ZCN  N IBZ,IBRXDATA,IBRXIEN,IBRXFILL,IBCOBN,IBBPS,IB0,IBS,IBHPD,IBVL,IBCPY,IBM0  K ^TMP("IBJTRX",$J)  S VALMCNT=0  ;  S IBZ=+$O(^IBA(362.4,"C",IBIFN,0))  S IBRXDATA=$G(^IBA(362.4,IBZ,0))  S IBRXIEN=+$P(IBRXDATA,U,5) ; RX ien ptr file 52  S IBRXFILL=+$P(IBRXDATA,U,10) ; rx fill#  S IBCOBN=+$$COBN^IBCEF(IBIFN) ; current payer sequence #  S IBBPS=$$CLAIM^BPSBUTL(IBRXIEN,IBRXFILL,IBCOBN) ; DBIA 4719  ;  S IBM1=$G(^DGCR(399,IBIFN,"M1"))  S IB0=$G(^DGCR(399,IBIFN,0))  S IBS=$G(^DGCR(399,IBIFN,"S"))  S ECME=$P($P(IBM1,U,8),";",1) ; ECME#  S ECMEAP=$P(IBM1,U,9) ; ECME approval number  S RXORG=$$RXSITE^IBCEF73A(IBIFN) ; pharmacy file 4 ien  S DOCIEN=$$RXAPI1^IBNCPUT1(IBRXIEN,4,"I") ; ien of doctor who wrote the Rx (52,4)  S (PHARMNPI,DOCNPI)=""  I RXORG S PHARMNPI=$P($$NPI^XUSNPI("Organization\_ID",RXORG),U,1) ; pharmacy NPI  I DOCIEN S DOCNPI=$P($$NPI^XUSNPI("Individual\_ID",DOCIEN),U,1) ; doctor NPI  I PHARMNPI'>0 S PHARMNPI="No NPI on file"  I DOCNPI'>0 S DOCNPI="No NPI on file"  ;  S RESPIEN=+$P(IBBPS,U,3) ; BPS response file ien  I RESPIEN D  . ; IB\*2.0\*521 - add HPID from response to TPJI screen  . S IBM0=$G(^DGCR(399,IBIFN,"M")),IBCPY=$S($P(IB0,U,21)="P":$P(IBM0,U),$P(IB0,U,21)="S":$P(IBM0,U,2),1:$P(IBM0,"^",3))  . I $P($G(^BPSR(RESPIEN,560)),U,8)="01" S IBHPD=$P($G(^BPSR(RESPIEN,560)),U,9) S IBVL=$$HOD^IBCNHUT1(IBHPD,IBCPY)  . S ZR=RESPIEN\_","  . S RSPSUB=+$O(^BPSR(RESPIEN,1000,0))  . I RSPSUB D  .. S ZM=RSPSUB\_","\_RESPIEN\_","  .. D GETS^DIQ(9002313.0301,ZM,"129;133:137;505;506;507;509;517:520;571;572;2193,"IEN","BPSM") ; get selected $ amount fields  .. D GETS^DIQ(9002313.0301,ZM,"355.01\*","IEN","BPSMCOB") ; get cob/other payer data fields  .. Q  . Q  ;  S IBLINE=$$SETL("",ECME,"ECME No",25,11,1)  S IBLINE=$$SETL(IBLINE,PHARMNPI,"Pharmacy NPI",14,15,40)  D SET(IBLINE) . . .   ;  D SET(" ")  S IBLINE=$$SETL("",,"Patient Responsibility Amounts",,31,1)  D SET(IBLINE,"3;2;30")  ;  S IBLINE=$$SETL("",$$AMT($G(BPSM(9002313.0301,ZM,517,"E"))),"Deductible",10,13,1)  S IBLINE=$$SETL(IBLINE,$$AMT($G(BPSM(9002313.0301,ZM,572,"E"))),"Coinsurance",10,13,27)  S IBLINE=$$SETL(IBLINE,$$AMT($G(BPSM(9002313.0301,ZM,518,"E"))),"Amount of Copay",9,18,52)  D SET(IBLINE)  ;  S IBLINE=$$SETL("",$$AMT($G(BPSM(9002313.0301,ZM,137,"E"))),"Coverage Gap",10,13,1)  S IBLINE=$$SETL(IBLINE,$$AMT($G(BPSM(9002313.0301,ZM,571,"E"))),"Processor Fee",10,13,27)  S IBLINE=$$SETL(IBLINE,$$AMT($G(BPSM(9002313.0301,ZM,520,"E"))),"Exceed Benefit Max",9,18,52)  D SET(IBLINE)  ;  S IBLINE=$$SETL("",$$AMT($G(BPSM(9002313.0301,ZM,129,"E"))),"Health Plan-funded Assistance Amount",15,39,1)   D SET(IBLINE)   S IBLINE=$$SETL("",$$AMT($G(BPSM(9002313.0301,ZM,2193,"E"))),"Patient Pay Component Amount",15,39,1)  D SET(IBLINE)  ;  D SET(" ")  S IBLINE=$$SETL("",,"Product Selection Amounts",,26,1)  D SET(IBLINE,"3;2;25")  ; . . .   ; INITX ;  D SET(" "),SET(" ")  Q | | | | |

The CRI/Claim Response Inquiry will automatically display the new outgoing fields from the claim request whenever they are populated and the new incoming fields on the claim response whenever it is populated. No change to existing logic is necessary. This is also true of the VER, which contains the CRI.

The tables below describe how the Prescriber DEA Number (D01-KV) and Total Prescribed Quantity Remaining (D02-KW) are being added to the *Claim Log*.

| Subroutine Name | **DISPCLM^BPSSCRL1** | | | |
| --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify | Delete | No Change |
| Current Logic | | | | |
| DISPCLM(BPLN,BP59,BPIEN02,BP57,BPSTYPE,BPSDTALT) ;  N BPSCRLNS S BPSCRLNS=17 ;(see "BPS LSTMN LOG" LM template: Bottom=21, Top = 4, 21-4=17)  N BPX,BPLN0,BPCNT,BPSTR1,BPSTYP2,BPNFLDT  S BPLN0=BPLN  S BPSTYP2=$S(BPSTYPE="C":"CLAIM REQUEST",BPSTYPE="R":"REVERSAL",1:"")  S BPSTR1="Transmission Information ("\_BPSTYP2\_")(#"\_BPIEN02\_")"  D SETLINE^BPSSCRLG(.BPLN,BPSTR1\_$$LINE^BPSSCRU3(79-$L(BPSTR1),"-"))  D SETLINE^BPSSCRLG(.BPLN,"Created on: "\_$$CREATEDT^BPSSCRLG(BPIEN02,BPSDTALT))  D SETLINE^BPSSCRLG(.BPLN,"VA Claim ID: "\_$P($G(^BPSC(+BPIEN02,0)),U,1))  D SETLINE^BPSSCRLG(.BPLN,"Submitted By: "\_$$SUBMTBY^BPSSCRLG(BP57))  D SETLINE^BPSSCRLG(.BPLN,"Transaction Type: "\_$$TRTYPE^BPSSCRU5($$TRCODE^BPSSCRLG(BPIEN02)))  D SETLINE^BPSSCRLG(.BPLN,"Date of Service: "\_$$DOSCLM^BPSSCRLG(BPIEN02))  ;Display Next Available Fill Date - BPS\*1.0\*15  S BPNFLDT=$$NFLDT^BPSBUTL(BPRXIEN,BPREF,$$RXCOB57^BPSSCRLG(BP57))  D:BPNFLDT SETLINE^BPSSCRLG(.BPLN,"Next Available Fill Date: "\_$$FMTE^XLFDT(BPNFLDT,"2ZM"))  D SETLINE^BPSSCRLG(.BPLN,"NDC Code: "\_$$LNDC^BPSSCRU5(BPIEN02))  ;  S BPUNITS=$$UNITS^BPSSCRLG(BPIEN02)  ; if BPUNITS is null get the BPUNITS from the PRESCRIPTION file (#52)  I BPUNITS="( )" S BPUNITS=$$GETUNIT(BPRXIEN,$G(BPREF))  D SETLINE^BPSSCRLG(.BPLN,"Quantity Submitted on Claim: "\_$$QTY^BPSSCRLG(BPIEN02)\_" "\_BPUNITS)  ;  D SETLINE^BPSSCRLG(.BPLN,"Days Supply: "\_$$DAYSSUPL^BPSSCRLG(BPIEN02))  D SETLINE^BPSSCRLG(.BPLN,"Division: "\_$$DIV^BPSSCRLG(BP57))  D SETLINE^BPSSCRLG(.BPLN,"NPI#: "\_$$NPI^BPSSCRLG(BPIEN02))  D SETLINE^BPSSCRLG(.BPLN,"ECME Pharmacy: "\_$$DIVNAME^BPSSCRDS($$LDIV^BPSSCRLG(BP57)))  S BPX="Rx Qty: "\_$$BILLQTY^BPSSCRLG(BP57)\_" "\_$$BILLUNT^BPSSCRLG(BP57)  S BPX=BPX\_" Unit Cost: "\_$$UNTPRICE^BPSSCRLG(BP57)  S BPX=BPX\_" Gross Amt Due: "\_$$TOTPRICE^BPSSCRLG(BPIEN02)  D SETLINE^BPSSCRLG(.BPLN,BPX)  S BPX="Ingredient Cost: "\_$$INGRCST^BPSSCRLG(BPIEN02)  S BPX=BPX\_" Dispensing Fee: "\_$$DISPFEE^BPSSCRLG(BPIEN02)  D SETLINE^BPSSCRLG(.BPLN,BPX)  S BPX="U&C Charge: "\_$$UCCHRG^BPSSCRLG(BPIEN02)  S BPX=BPX\_" Admin Fee: "\_$$ADMNFEE^BPSSCRLG(BPIEN02)  D SETLINE^BPSSCRLG(.BPLN,BPX)  D SETLINE^BPSSCRLG(.BPLN,"")  D SETLINE^BPSSCRLG(.BPLN,"Insurance Name: "\_$$INSUR57^BPSSCRLG(BP57))  D SETLINE^BPSSCRLG(.BPLN,"Group Name: "\_$$GRPNM^BPSSCRLG(BPIEN02))  D SETLINE^BPSSCRLG(.BPLN,"Rx Coordination of Benefits: "\_$$RXCOB57^BPSSCRLG(BP57))  D SETLINE^BPSSCRLG(.BPLN,"Pharmacy Plan ID: "\_$$PHPLANID^BPSSCRLG(BP57))  D SETLINE^BPSSCRLG(.BPLN,"BIN: "\_$$BIN^BPSSCRLG(BPIEN02))  D SETLINE^BPSSCRLG(.BPLN,"PCN: "\_$$PCN^BPSSCRLG(BPIEN02))  D SETLINE^BPSSCRLG(.BPLN,"NCPDP Version: "\_$$GETVER^BPSSCRLG(BPIEN02))  D SETLINE^BPSSCRLG(.BPLN,"Group ID: "\_$$GRPID^BPSSCRLG(BPIEN02))  D SETLINE^BPSSCRLG(.BPLN,"Cardholder ID: "\_$$CRDHLDID^BPSSCRLG(BPIEN02))  D SETLINE^BPSSCRLG(.BPLN,"Patient Relationship Code: "\_$$PATRELSH^BPSSCRLG(BPIEN02))  D SETLINE^BPSSCRLG(.BPLN,"Cardholder First Name: "\_$$CRDHLDFN^BPSSCRLG(BPIEN02,BP57))  D SETLINE^BPSSCRLG(.BPLN,"Cardholder Last Name: "\_$$CRDHLDLN^BPSSCRLG(BPIEN02,BP57))  ; BPS\*1\*22  D SETLINE^BPSSCRLG(.BPLN,"Facility ID Qualifier: "\_$$FACIDQ^BPSSCRLG(BPIEN02))  F BPCNT=BPLN:1:BPLN0+BPSCRLNS D SETLINE^BPSSCRLG(.BPLN,"")  S BPLN0=BPLN  D SETLINE^BPSSCRLG(.BPLN,"Billing Request Payer Sheet: "\_$$B1PYRIEN^BPSSCRU5(BP57))  D SETLINE^BPSSCRLG(.BPLN,"Reversal Payer Sheet: "\_$$B2PYRIEN^BPSSCRU5(BP57))  D SETLINE^BPSSCRLG(.BPLN,"VA Claim ID: "\_$P($G(^BPSC(+BPIEN02,0)),U,1))  D SETLINE^BPSSCRLG(.BPLN,"")  Q | | | | |
| Modified Logic (Changes are highlighted) | | | | |
| DISPCLM(BPLN,BP59,BPIEN02,BP57,BPSTYPE,BPSDTALT) ;  N BPSCRLNS S BPSCRLNS=17 ;(see "BPS LSTMN LOG" LM template: Bottom=21, Top = 4, 21-4=17)  N BPX,BPLN0,BPCNT,BPSTR1,BPSTYP2,BPNFLDT  S BPLN0=BPLN  S BPSTYP2=$S(BPSTYPE="C":"CLAIM REQUEST",BPSTYPE="R":"REVERSAL",1:"")  S BPSTR1="Transmission Information ("\_BPSTYP2\_")(#"\_BPIEN02\_")"  D SETLINE^BPSSCRLG(.BPLN,BPSTR1\_$$LINE^BPSSCRU3(79-$L(BPSTR1),"-"))  D SETLINE^BPSSCRLG(.BPLN,"Created on: "\_$$CREATEDT^BPSSCRLG(BPIEN02,BPSDTALT))  D SETLINE^BPSSCRLG(.BPLN,"VA Claim ID: "\_$P($G(^BPSC(+BPIEN02,0)),U,1))  D SETLINE^BPSSCRLG(.BPLN,"Submitted By: "\_$$SUBMTBY^BPSSCRLG(BP57))  D SETLINE^BPSSCRLG(.BPLN,"Transaction Type: "\_$$TRTYPE^BPSSCRU5($$TRCODE^BPSSCRLG(BPIEN02)))  D SETLINE^BPSSCRLG(.BPLN,"Date of Service: "\_$$DOSCLM^BPSSCRLG(BPIEN02))  ;Display Next Available Fill Date - BPS\*1.0\*15  S BPNFLDT=$$NFLDT^BPSBUTL(BPRXIEN,BPREF,$$RXCOB57^BPSSCRLG(BP57))  D:BPNFLDT SETLINE^BPSSCRLG(.BPLN,"Next Available Fill Date: "\_$$FMTE^XLFDT(BPNFLDT,"2ZM"))  D SETLINE^BPSSCRLG(.BPLN,"NDC Code: "\_$$LNDC^BPSSCRU5(BPIEN02))  ;  S BPUNITS=$$UNITS^BPSSCRLG(BPIEN02)  ; if BPUNITS is null get the BPUNITS from the PRESCRIPTION file (#52)  I BPUNITS="( )" S BPUNITS=$$GETUNIT(BPRXIEN,$G(BPREF))  D SETLINE^BPSSCRLG(.BPLN,"Quantity Submitted on Claim: "\_$$QTY^BPSSCRLG(BPIEN02)\_" "\_BPUNITS)  ;  D SETLINE^BPSSCRLG(.BPLN,"Days Supply: "\_$$DAYSSUPL^BPSSCRLG(BPIEN02))  D SETLINE^BPSSCRLG(.BPLN,"Division: "\_$$DIV^BPSSCRLG(BP57))  D SETLINE^BPSSCRLG(.BPLN,"NPI#: "\_$$NPI^BPSSCRLG(BPIEN02))   D SETLINE^BPSSCRLG(.BPLN,"Prescriber DEA Number: "\_$$PDEA^BPSSCRLG(BPIEN02))  D SETLINE^BPSSCRLG(.BPLN,"ECME Pharmacy: "\_$$DIVNAME^BPSSCRDS($$LDIV^BPSSCRLG(BP57)))   D SETLINE^BPSSCRLG(.BPLN,"Total Prescribed Quantity Remaining: "\_$$TOTPQR^BPSSCRLG(BPIEN02))  S BPX="Rx Qty: "\_$$BILLQTY^BPSSCRLG(BP57)\_" "\_$$BILLUNT^BPSSCRLG(BP57)  S BPX=BPX\_" Unit Cost: "\_$$UNTPRICE^BPSSCRLG(BP57)  S BPX=BPX\_" Gross Amt Due: "\_$$TOTPRICE^BPSSCRLG(BPIEN02)  D SETLINE^BPSSCRLG(.BPLN,BPX)  S BPX="Ingredient Cost: "\_$$INGRCST^BPSSCRLG(BPIEN02)  S BPX=BPX\_" Dispensing Fee: "\_$$DISPFEE^BPSSCRLG(BPIEN02)  D SETLINE^BPSSCRLG(.BPLN,BPX)  S BPX="U&C Charge: "\_$$UCCHRG^BPSSCRLG(BPIEN02)  S BPX=BPX\_" Admin Fee: "\_$$ADMNFEE^BPSSCRLG(BPIEN02)  D SETLINE^BPSSCRLG(.BPLN,BPX)  D SETLINE^BPSSCRLG(.BPLN,"")  D SETLINE^BPSSCRLG(.BPLN,"Insurance Name: "\_$$INSUR57^BPSSCRLG(BP57))  D SETLINE^BPSSCRLG(.BPLN,"Group Name: "\_$$GRPNM^BPSSCRLG(BPIEN02))  D SETLINE^BPSSCRLG(.BPLN,"Rx Coordination of Benefits: "\_$$RXCOB57^BPSSCRLG(BP57))  D SETLINE^BPSSCRLG(.BPLN,"Pharmacy Plan ID: "\_$$PHPLANID^BPSSCRLG(BP57))  D SETLINE^BPSSCRLG(.BPLN,"BIN: "\_$$BIN^BPSSCRLG(BPIEN02))  D SETLINE^BPSSCRLG(.BPLN,"PCN: "\_$$PCN^BPSSCRLG(BPIEN02))  D SETLINE^BPSSCRLG(.BPLN,"NCPDP Version: "\_$$GETVER^BPSSCRLG(BPIEN02))  D SETLINE^BPSSCRLG(.BPLN,"Group ID: "\_$$GRPID^BPSSCRLG(BPIEN02))  D SETLINE^BPSSCRLG(.BPLN,"Cardholder ID: "\_$$CRDHLDID^BPSSCRLG(BPIEN02))  D SETLINE^BPSSCRLG(.BPLN,"Patient Relationship Code: "\_$$PATRELSH^BPSSCRLG(BPIEN02))  D SETLINE^BPSSCRLG(.BPLN,"Cardholder First Name: "\_$$CRDHLDFN^BPSSCRLG(BPIEN02,BP57))  D SETLINE^BPSSCRLG(.BPLN,"Cardholder Last Name: "\_$$CRDHLDLN^BPSSCRLG(BPIEN02,BP57))  ; BPS\*1\*22  D SETLINE^BPSSCRLG(.BPLN,"Facility ID Qualifier: "\_$$FACIDQ^BPSSCRLG(BPIEN02))  F BPCNT=BPLN:1:BPLN0+BPSCRLNS D SETLINE^BPSSCRLG(.BPLN,"")  S BPLN0=BPLN  D SETLINE^BPSSCRLG(.BPLN,"Billing Request Payer Sheet: "\_$$B1PYRIEN^BPSSCRU5(BP57))  D SETLINE^BPSSCRLG(.BPLN,"Reversal Payer Sheet: "\_$$B2PYRIEN^BPSSCRU5(BP57))  D SETLINE^BPSSCRLG(.BPLN,"VA Claim ID: "\_$P($G(^BPSC(+BPIEN02,0)),U,1))  D SETLINE^BPSSCRLG(.BPLN,"")  Q | | | | |

A new tag was created to get the data for the Prescriber DEA Number, which is described below.

| Subroutine Name | **PDEA^BPSSCRLG** | | | |
| --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify | Delete | No Change |
| Input Attribute Name and Definition | Name: BPIEN02  Definition: IEN to #9002313.02  New  Modify  Delete  No Change | | | |
| Output Attribute Name and Definition | Name: $$PDEA  Definition: Prescriber DEA Number  New  Modify  Delete  No Change | | | |
| Current Logic | | | | |
| N/A | | | | |
| Modified Logic (Changes are highlighted) | | | | |
| BPSSCRLG ;BHAM ISC/SS - ECME LOGINFO ;05-APR-05  ;;1.0;E CLAIMS MGMT ENGINE;\*\*1,5,7,8,10,11,15,18,20,22,24\*\*;JUN 2004;Build 27  ;;Per VA Directive 6402, this routine should not be modified.  ;  . . .  ;Prescriber DEA Number  PDEA(BPIEN02) ;  Q $E($P($G(^BPSC(BPIEN02,400,1,"D00")),U,1),3,18)  ; | | | | |

A new tag was created to get the data for the Total Prescribed Quantity Remaining, which is described below.

| Subroutine Name | **TOTPQR^BPSSCRLG** | | | |
| --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify | Delete | No Change |
| Input Attribute Name and Definition | Name: BPIEN02  Definition: IEN to #9002313.02  New  Modify  Delete  No Change | | | |
| Output Attribute Name and Definition | Name: $$TOTPQR  Definition: Total Prescribed Quantity Remaining  New  Modify  Delete  No Change | | | |
| Current Logic | | | | |
| N/A | | | | |
| Modified Logic (Changes are highlighted) | | | | |
| ;Total Prescribed Quantity Remaining  TOTPQR(BPIEN02) ;  N X  S X=$E($P($G(^BPSC(BPIEN02,400,1,"D00")),U,2),3,99)  Q +X  ; | | | | |

The table below lists new NCPDP fields that will appear on the *Reject Information Screen* only if a value exists. These new fields will appear in the same location as the Next Avail Fill label.

|  |  |  |
| --- | --- | --- |
| **FIELD #** | **ID** | **FIELD NAME** |
| 931 | F8 | Maximum Age Qualifier |
| 932 | GA | Maximum Age |
| 933 | GB | Maximum Amount |
| 934 | GC | Maximum Amount Qualifier |
| 935 | GF | Maximum Amount Time Period |
| 936 | GG | Maximum Amount Time Period Start Date |
| 937 | GH | Maximum Amount Time Period End Date |
| 938 | GJ | Maximum Amount Time Period Units |
| 943 | GQ | Minimum Age Qualifier |
| 944 | GR | Minimum Age |
| D19 | M1 | Minimum Amount |
| D20 | M2 | Minimum Amount Qualifier |
| D24 | M6 | Remaining Amount |
| D25 | M7 | Remaining Amount Qualifier |

A portion of the code from REJ^PSOREJP1 will be moved to REJ^PSOREJP5, due to the size of PSOREJP1. REJ^PSOREJP5 will be modified to add the code from REJ^PSOREJP1 and to display the NCPDP fields above on the *Reject Information Screen* if there is a value.

| Subroutine Name | **REJ^PSOREJP1** | | | |
| --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify | Delete | No Change |
| Current Logic | | | | |
| PSOREJP1 ;BIRM/MFR - Third Party Reject Display Screen ;04/29/05  ;;7.0;OUTPATIENT PHARMACY;\*\*148,247,260,281,287,289,290,358,359,385,403,421,427,448,478,482\*\*;DEC 1997;Build 27  . . .  REJ ; - DUR Information  N TYPE,PFLDT,TREJ,TDATA,PSOET,PSONAF,PSOCOB,PSOTXT,PSOECME S TDATA=""  ;  ; LH;PSO\*7\*448 - Display 'RESUBMISSION' where 'BACK-BILL' currently  ; displays if the claim was resubmitted from the ECME User Screen.  ; To facilitate this, the function $$RESUBMIT^BPSBUTL was created.  ;  ; Back Bill indicator - PSO\*7\*421  S PSOTXT="",PSOCOB=$G(DATA(REJ,"COB")),PSOCOB=$S(PSOCOB="SECONDARY":2,PSOCOB="TERTIARY":3,1:1)  I $$BBILL^BPSBUTL(RX,FILL,PSOCOB) S PSOTXT=" BACK-BILL"  E I $$RESUBMIT^BPSBUTL(RX,FILL,PSOCOB) S PSOTXT=" RESUBMISSION" ; IA 4719.  D SETLN("REJECT Information ("\_$$ELIGTCV(RX,FILL)\_") "\_PSOTXT,1,1)  S PSOECME=$$STATUS^PSOBPSUT(RX,FILL)  I PSOECME="E PAYABLE" D  . D SETLN("Reject Type : ",,,18)  . D SETLN("Reject Status : \*\* E PAYABLE \*\*",,,18)  . Q  E D  . S TYPE=$S($G(DATA(REJ,"CODE"))=79:"79 - REFILL TOO SOON",1:"")  . I TYPE="" S TYPE=DATA(REJ,"CODE")\_" - "\_$E($$EXP(DATA(REJ,"CODE")),1,23)\_"-"  . D SETLN("Reject Type : "\_TYPE\_" received on "\_$$FMTE^XLFDT($G(DATA(REJ,"DATE/TIME"))),,,18)  . ;cnf, PSO\*7\*358, if TRICARE/CHAMPVA non-billable then reset Status line  . S PSOET=$$PSOET^PSOREJP3(RX,FILL)  . I PSOET D SETLN("Status : NO CLAIM SUBMITTED")  . I 'PSOET D SETLN("Reject Status : "\_$G(DATA(REJ,"STATUS"))\_" - "\_PSOECME,,,18)  . Q  S PSONAF=$$NFLDT^BPSBUTL(RX,FILL) ; IA 4719  I PSONAF'="" D SETLN("Next Avail Fill: "\_$$FMTE^XLFDT(PSONAF),,,18) ; PSO\*7\*421  D SET("PAYER MESSAGE",63)  D SET("REASON",63)  S PFLDT=$$FMTE^XLFDT($G(DATA(REJ,"PLAN PREVIOUS FILL DATE")))  D SET("DUR TEXT",63,$S(PFLDT="":1,1:0))  I PFLDT'="" D SETLN("Last Fill Date : "\_PFLDT\_" (from payer)",,1,18)  Q | | | | |
| Modified Logic (Changes are highlighted) | | | | |
| PSOREJP1 ;BIRM/MFR - Third Party Reject Display Screen ;04/29/05  ;;7.0;OUTPATIENT PHARMACY;\*\*148,247,260,281,287,289,290,358,359,385,403,421,427,448,478,482,512\*\*;DEC 1997;Build 27  . . .  REJ ; - DUR Information  N TYPE,PFLDT,TREJ,TDATA,PSOET,PSONAF,PSOCOB,PSOTXT,PSOECME,PSOADD  S TDATA=""  ;  ; LH;PSO\*7\*448 - Display 'RESUBMISSION' where 'BACK-BILL' currently  ; displays if the claim was resubmitted from the ECME User Screen.  ; To facilitate this, the function $$RESUBMIT^BPSBUTL was created.  ;  ; Back Bill indicator - PSO\*7\*421  S PSOTXT="",PSOCOB=$G(DATA(REJ,"COB")),PSOCOB=$S(PSOCOB="SECONDARY":2,PSOCOB="TERTIARY":3,1:1)  I $$BBILL^BPSBUTL(RX,FILL,PSOCOB) S PSOTXT=" BACK-BILL"  E I $$RESUBMIT^BPSBUTL(RX,FILL,PSOCOB) S PSOTXT=" RESUBMISSION" ; IA 4719.  D SETLN("REJECT Information ("\_$$ELIGTCV(RX,FILL)\_") "\_PSOTXT,1,1)  S PSOECME=$$STATUS^PSOBPSUT(RX,FILL)  I PSOECME="E PAYABLE" D  . D SETLN("Reject Type : ",,,18)  . D SETLN("Reject Status : \*\* E PAYABLE \*\*",,,18)  . Q  E D  . S TYPE=$S($G(DATA(REJ,"CODE"))=79:"79 - REFILL TOO SOON",1:"")  . I TYPE="" S TYPE=DATA(REJ,"CODE")\_" - "\_$E($$EXP(DATA(REJ,"CODE")),1,23)\_"-"  . D SETLN("Reject Type : "\_TYPE\_" received on "\_$$FMTE^XLFDT($G(DATA(REJ,"DATE/TIME"))),,,18)  . ;cnf, PSO\*7\*358, if TRICARE/CHAMPVA non-billable then reset Status line  . S PSOET=$$PSOET^PSOREJP3(RX,FILL)  . I PSOET D SETLN("Status : NO CLAIM SUBMITTED")  . I 'PSOET D SETLN("Reject Status : "\_$G(DATA(REJ,"STATUS"))\_" - "\_PSOECME,,,18)  . Q  ; code moved to PSOREJP5  D REJ^PSOREJP5  Q | | | | |

| Subroutine Name | **REJ^PSOREJP5** | | | |
| --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify | Delete | No Change |
| Current Logic | | | | |
| n/a – new subroutine | | | | |
| Modified Logic (Changes are highlighted) | | | | |
| PSOREJP5 ;ALB/BNT - Third Party Reject Additional Reject Information Screen ;02/14/11  ;;7.0;OUTPATIENT PHARMACY;\*\*359,421,512\*\*;DEC 1997;Build 15  ;   ; Reference to $$BBILL^BPSBUTL and $$RESUBMIT^BPSBUTL supported by IA 4719  ; Reference to BPSNCPD3 supported by IA 4560  ;  . . .  REJ ; - DUR Information - called from REJ^PSOREJP1  ; this code moved from PSOREJP1, routine was too large  ;  S PSONAF=$$NFLDT^BPSBUTL(RX,FILL) ; IA 4719  I PSONAF'="" D SETLN^PSOREJP1("Next Avail Fill: "\_$$FMTE^XLFDT(PSONAF),,,16) ; PSO\*7\*421  S PSOADD=$$ADDFLDS^BPSBUTL(RX,FILL) ; IA 4719  I '$G(PSOADD) S PSOADD=""  I $P(PSOADD,U)'="" D SETLN^PSOREJP1("Maximum Age Qualifier: "\_$P(PSOADD,U),,,22)  I $P(PSOADD,U,2)'="" D SETLN^PSOREJP1("Maximum Age: "\_$P(PSOADD,U,2),,,12)  I $P(PSOADD,U,3)'="" D SETLN^PSOREJP1("Maximum Amount: "\_$P(PSOADD,U,3),,,15)  I $P(PSOADD,U,4)'="" D SETLN^PSOREJP1("Maximum Amount Qualifier: "\_$P(PSOADD,U,4),,,25)  I $P(PSOADD,U,5)'="" D SETLN^PSOREJP1("Maximum Amount Time Period: "\_$P(PSOADD,U,5),,,27)  I $P(PSOADD,U,6)'="" D SETLN^PSOREJP1("Maximum Amount Time Period Start Date: "\_$$FMTE^XLFDT($P(PSOADD,U,6)),,,38)  I $P(PSOADD,U,7)'="" D SETLN^PSOREJP1("Maximum Amount Time Period End Date: "\_$$FMTE^XLFDT($P(PSOADD,U,7)),,,36)  I $P(PSOADD,U,8)'="" D SETLN^PSOREJP1("Maximum Amount Time Period Units: "\_$P(PSOADD,U,8),,,33)  I $P(PSOADD,U,9)'="" D SETLN^PSOREJP1("Minimum Age Qualifier: "\_$P(PSOADD,U,9),,,22)  I $P(PSOADD,U,10)'="" D SETLN^PSOREJP1("Minimum Age: "\_$P(PSOADD,U,10),,,12)  I $P(PSOADD,U,11)'="" D SETLN^PSOREJP1("Minimum Amount: "\_$P(PSOADD,U,11),,,15)  I $P(PSOADD,U,12)'="" D SETLN^PSOREJP1("Minimum Amount Qualifier: "\_$P(PSOADD,U,12),,,25)  I $P(PSOADD,U,13)'="" D SETLN^PSOREJP1("Remaining Amount: "\_$P(PSOADD,U,13),,,17)  I $P(PSOADD,U,14)'="" D SETLN^PSOREJP1("Remaining Amount Qualifier: "\_$P(PSOADD,U,14),,,27)  ;  D SET^PSOREJP1("PAYER MESSAGE",63)  D SET^PSOREJP1("REASON",63)  S PFLDT=$$FMTE^XLFDT($G(DATA(REJ,"PLAN PREVIOUS FILL DATE")))  D SET^PSOREJP1("DUR TEXT",63,$S(PFLDT="":1,1:0))  I PFLDT'="" D SETLN^PSOREJP1("Last Fill Date : "\_PFLDT\_" (from payer)",,1,18)  Q | | | | |

ADDFLDS^BPSBUTL will be created to get the data for the NCPDP fields that were listed above so they can be displayed on the *Reject Information Screen*.

| Subroutine Name | **ADDFLDS^BPSBUTL** | | | |
| --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify | Delete | No Change |
| Input Attribute Name and Definition | Name: RX  Definition: RX IEN (req)  New  Modify  Delete  No Change  Name: FIL  Definition: Fill Number (req)  New  Modify  Delete  No Change | | | |
| Output Attribute Name and Definition | Name: $$ADDFLDS  Definition: a string separated by “^” containing the output of the BPSARR array  New  Modify  Delete  No Change | | | |
| Current Logic | | | | |
| N/A | | | | |
| Modified Logic (Changes are highlighted) | | | | |
| BPSBUTL ;BHAM ISC/MFR/VA/DLF - IB Communication Utilities ;06/01/2004  ;;1.0;E CLAIMS MGMT ENGINE;\*\*1,3,2,5,7,8,9,10,11,15,20,24\*\*;JUN 2004;Build 27  ;;Per VA Directive 6402, this routine should not be modified.  . . .  ADDFLDS(RX,FIL) ;Returns Additional NCPDP fields from ECME  ;Input:  ; RX (req) --> RX IEN  ; FIL (req) --> Fill number  ;Output:  ; BPSREC --> a string separated by "^" containing the output of the BPSARR array  ;  Q:'$G(RX)!($G(FIL)="") ""  N BPSAMT,BPSARR,BPSCNT,BPSEDT,BPSFLDS,BPSREC,BPSSDT,IEN03,IEN59  ;  S IEN59=$$IEN59^BPSOSRX(RX,FIL,1)  S IEN03=+$P($G(^BPST(IEN59,0)),U,5)  ;  I '$G(IEN03) Q "" ; Quit if INE03 is not found  ;  S BPSFLDS="931;932;933;934;935;936;937;938;943;944;2219;2220;2224;2225"  D GETS^DIQ(9002313.0301,"1,"\_IEN03,BPSFLDS,"IE","BPSARR")  ;  S BPSREC="",BPSCNT=0  F I=1:1:$L(BPSFLDS,";") S BPSCNT=BPSCNT+1 D  . S $P(BPSREC,U,BPSCNT)=BPSARR(9002313.0301,"1,"\_IEN03\_",",+$P(BPSFLDS,";",I),"E")  ;  ; Convert Dates to Fileman Format (936,937)  S BPSSDT=BPSARR(9002313.0301,"1,"\_IEN03\_",",936,"I") ; MAX AMT START DATE  S:BPSSDT BPSSDT=BPSSDT-17000000,$P(BPSREC,U,6)=BPSSDT  S BPSEDT=BPSARR(9002313.0301,"1,"\_IEN03\_",",937,"I") ; MAX AMT END DATE  S:BPSEDT BPSEDT=BPSEDT-17000000,$P(BPSREC,U,7)=BPSEDT  ;  Q BPSREC | | | | |

The table below lists new NCPDP fields that will be added to File 9002313.32, BPS PAYER RESPONSE OVERRIDES, so that the fields can be tested on an incoming response using the testing tool. The routine BPSTEST will also be modified in order for these fields to be included in the incoming response from the payer.

|  |  |  |
| --- | --- | --- |
| **FIELD #** | **ID** | **FIELD NAME** |
| 931 | F8 | Maximum Age Qualifier |
| 932 | GA | Maximum Age |
| 933 | GB | Maximum Amount |
| 934 | GC | Maximum Amount Qualifier |
| 935 | GF | Maximum Amount Time Period |
| 936 | GG | Max Amt Time Period Start Date |
| 937 | GH | Max Amt Time Period End Date |
| 938 | GJ | Max Amt Time Period Units |
| 943 | GQ | Minimum Age Qualifier |
| 944 | GR | Minimum Age |
| C47 | 9T | Other Payer Program Type |
| C93 | KN | Patient Pay Component Amount |
| C94 | KP | Patient Pay Component Count |
| C95 | KQ | Patient Pay Component Qual |
| D19 | M1 | Minimum Amount |
| D20 | M2 | Minimum Amount Qualifier |
| D23 | M5 | Other Payer Name |
| D24 | M6 | Remaining Amount |
| D25 | M7 | Remaining Amount Qualifier |
| D41 | PQ | Other Payer Relationship Type |

The new fields will be added to the data dictionary of File 9002313.32, BPS PAYER RESPONSE OVERRIDES, in our development environment, and that file will be included in our patch, so that the new field will be included in the KIDS Build. When the KIDS Build is installed at the sites, the new fields will be automatically added to File 9002313.32 on their system.

The data dictionary for file 9002313.32, BPS PAYER RESPONSE OVERRIDES, will be sent without data to the target sites. The table below describes how the file is to be included in the KIDS Build.

|  |  |
| --- | --- |
| File Name | **BPS PAYER RESPONSE OVERRIDES** |
| Send Full or Partial DD | PARTIAL |
| Update the Data Dictionary | YES |
| Send Security Codes | YES |
| Screen to Determine DD Update | n/a |
| Data Comes with File | NO |

The following two tables describe how the new fields are being added to file 9002313.32, BPS PAYER RESPONSE OVERRIDES.

The table below describes the Field Name, Field Number, Node;Piece, Field Type and Field length. A description will also need to be added to this file, the description will be the same as the one used when the new fields were added to the BPS NCPDP FIELD DEFS file.

\* - the NCPDP field number is not part of file 9002313.32 it was added only as a reference

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **NCPDP FIELD #\*** | **FIELD NAME** | **FIELD**  **NO** | **NODE;**  **PIECE** | **FIELD TYPE/LENGTH** |
| 931-F8 | MAXIMUM AGE QUALIFIER | 4.01 | 4;1 | SET / 1  ‘D’ FOR DAYS; ‘Y’ FOR YEARS |
| 932-GA | MAXIMUM AGE | 4.02 | 4;2 | NUMBER / 3 |
| 933-GB | MAXIMUM AMOUNT | 4.03 | 4;3 | NUMBER / 10 |
| 934-GC | MAXIMUM AMOUNT QUALIFIER | 4.04 | 4;4 | SET / 2  ‘DL’ FOR DOLLAR AMOUNT; ‘DS’ FOR DAYS SUPPLY; ‘FL’ FOR FILLS;  ‘QY’ FOR QUANTITY |
| 935-GF | MAXIMUM AMOUNT TIME PERIOD | 4.05 | 4;5 | SET / 2  ‘CM’ FOR CALENDAR MONTH;,  ‘CQ’ FOR CALENDAR QUARTER;  ‘CY’ FOR CALENDAR YEAR;,  ‘DY’ FOR DAYS;,  ‘LT’ FOR LIFETIME,  ‘PD’ FOR PER DISPENSING,  ‘SP’ FOR SPECIFIC DATE RANGE |
| 936-GG | MAX AMT TIME PERIOD START DATE | 4.06 | 4;6 | DATE / 8 |
| 937-GH | MAX AMT TIME PERIOD END DATE | 4.07 | 4;7 | DATE / 8 |
| 936-GJ | MAX AMT TIME PERIOD UNITS | 4.08 | 4;8 | NUMBER / 4 |
| 943-GQ | MINIMUM AGE QUALIFIER | 4.09 | 4;9 | SET / 1  ‘D’ FOR DAYS,  ‘Y’ FOR YEARS |
| 944-GR | MINIMUM AGE | 4.1 | 4;10 | NUMBER / 3 |
| C47-9T | OTHER PAYER PROGRAM TYPE | 4.11 | 4;11 | POINTER / 9002313.38 |
| C93-KN | PATIENT PAY COMPONENT AMOUNT | 4.12 | 4;12 | NUMBER / 8 |
| C94-KP | PATIENT PAY COMPONENT COUNT | 4.13 | 4;13 | NUMBER / 2 |
| C95-KP | PATIENT PAY COMPONENT QUAL | 4.14 | 4;14 | POINTER / 9002313.37 |
| D19-M1 | MINIMUM AMOUNT | 4.15 | 4;15 | NUMBER /10 |
| D20-M2 | MINIMUM AMOUNT QUALIFIER | 4.16 | 4;16 | SET / 3  ‘MDL’ FOR MIN DOLLAR AMT;  ‘MDS’ FOR MIN DAY SUPPLY;  ‘MFL’ FOR MIN FILLS;  ‘MQY’ FOR MIN QTY |
| D23-M5 | OTHER PAYER NAME | 4.17 | 4;17 | FREE TEXT / 70 |
| D24-M6 | REMAINING AMOUNT | 4.18 | 4;18 | NUMBER / 10 |
| D25-M7 | REMAINING AMOUNT QUALIFIER | 4.19 | 4;19 | SET / 3  ‘RFL’ FOR REMAINING FILLS;  ‘RQY’ FOR REMAINING QTY;  ‘RDS’ FOR REMAINING DAY SUPPLY |
| D41-PQ | OTHER PAYER RELATIONSHIP TYPE | 4.2 | 4;20 | SET / 3  ‘CP’ FOR COB COVERAGE PRIOR TO RESPONDING PAYER,  ‘CS’ FOR COB COVERAGE SUBSEQUENT TO RESPONDING PAYER,  ’MX’ FOR MUTUALLY EXCLUSIVE BENEFITS,  ‘CC’ FOR CHANGE IN COVERAGE,  ‘RP’ FOR RESPONDING PAYER,  ‘CE’ FOR CENTRALIZED ELIGIBILITY |

The table below describes the Help Prompt and Description for the new fields being added.

\* - the NCPDP field number is not part of file 9002313.32 it was added only as a reference

|  |  |  |  |
| --- | --- | --- | --- |
| **NCPDP FIELD # \*** | **FIELD NAME** | **HELP PROMPT** | **DESCRIPTION** |
| 931-F8 | MAXIMUM AGE QUALIFIER | Enter override value for the Maximum Age Qualifier. | This is the override value that will be used for the payer response for billing requests. The value will be used to populate the MAXIMUM AGE QUALIFIER (#931) field of the BPS RESPONSES (#9002313.03) file. Corresponds to NCPDP code 931-F8 and qualifies MAXIMUM AGE (932-GA). |
| 932-GA | MAXIMUM AGE | Enter override value between 0 and 999, for the Maximum Age. | This is the override value that will be used for the payer response for billing requests. The value will be used to populate the MAXIMUM AGE (#932) field of the BPS RESPONSES (#9002313.03) file. Corresponds to NCPDP code 932-GA. |
| 933-GB | MAXIMUM AMOUNT | Enter between 0 and 999999.999, 3 decimal digits. | This is the override value that will be used for the payer response for billing requests. The value will be used to populate the MAXIMUM AMOUNT (#933) field of the BPS RESPONSES (#9002313.03) file. Corresponds to NCPDP code 933-GB. |
| 934-GC | MAXIMUM AMOUNT QUALIFIER | Enter override value for the Maximum Amount Qualifier. | This is the override value that will be used for the payer response for billing requests. The value will be used to populate the MAXIMUM AMOUNT QUALIFIER (#934) field of the BPS RESPONSES (#9002313.03) file. Corresponds to NCPDP code 934-GC and qualifies MAXIMUM AMOUNT (933-GB). |
| 935-GF | MAXIMUM AMOUNT TIME PERIOD | Enter override value for the Maximum Amount Time Period. | This is the override value that will be used for the payer response for billing requests. The value will be used to populate the MAXIMUM AMOUNT TIME PERIOD (#935) field of the BPS RESPONSES (#9002313.03) file.  Corresponds to NCPDP code 935-GF. |
| 936-GG | MAX AMT TIME PERIOD START DATE | Enter the date for the Maximum Amount Time Period Start Date. | This is the override value that will be used for the payer response for billing requests. The value will be used to populate the MAXIMUM AMOUNT TIME PERIOD START DATE (#936) field of the BPS RESPONSES (#9002313.03) file.  Corresponds to NCPDP code 936-GG. |
| 937-GH | MAX AMT TIME PERIOD END DATE | Enter the date for the Maximum Amount Time Period End Date. | This is the override value that will be used for the payer response for billing requests. The value will be used to populate the MAXIMUM AMOUNT TIME PERIOD END DATE (#937) field of the BPS RESPONSES (#9002313.03) file.  Corresponds to NCPDP code 937-GH. |
| 943-GQ | MAX AMT TIME PERIOD UNITS | Enter an amount between 0 and 9999, 0 decimal digits, for the Maximum Amount Time Period Units. | This is the override value that will be used for the payer response for billing requests. The value will be used to populate the MAXIMUM AMOUNT TIME PERIOD UNITS (#938) field of the BPS RESPONSES (#9002313.03) file.  Corresponds to NCPDP code 938-GJ. |
| 943-GQ | MINIMUM AGE QUALIFIER | Enter override value for the Minimum Age Qualifier. | This is the override value that will be used for the payer response for billing requests. The value will be used to populate the MINIMUM AGE QUALIFIER (#943) field of the BPS RESPONSES (#9002313.03) file. Corresponds to NCPDP code 934-GQ and qualifies MINIMUM AGE (944-GR). |
| 944-GR | MINIMUM AGE | Enter override value between 0 and 999, for the Minimum Age. | This is the override value that will be used for the payer response for billing requests. The value will be used to populate the MINIMUM AGE (#944) field of the BPS RESPONSES (#9002313.03) file. Corresponds to NCPDP code 944-GR. |
| C47-9T | OTHER PAYER PROGRAM TYPE | Enter override value for the Other Payer Adjudicated Program Type. | This is the override value that will be used for the payer response for billing requests. The value will be used to populate the OTHER PAYER ADJUDICATED PROGRAM TYPE (#2147) field of the BPS RESPONSES (#9002313.03) file. Corresponds to NCPDP code C47-9T. |
| C93-KN | PATIENT PAY COMPONENT AMOUNT | Enter a dollar amount between 0 and 999999.99, 2 decimal digits. | This is the override value that will be used for the payer response for billing requests. The value will be used to populate the PATIENT PAY COMPONENT AMOUNT (#2193) field of the BPS RESPONSES (#9002313.03) file. Corresponds to NCPDP code C93-KN. |
| C94-KP | PATIENT PAY COMPONENT COUNT | Enter a value between 0 and 99, for the Patient Pay Component Count. | This is the override value that will be used for the payer response for billing requests. The value will be used to populate the PATIENT PAY COMPONENT COUNT (#2194) field of the BPS RESPONSES (#9002313.03) file. Corresponds to NCPDP code C94-KP. |
| C95-KP | PATIENT PAY COMPONENT QUAL | Enter override value for the Patient Pay Component Qualifier. | This is the override value that will be used for the payer response for billing requests. The value will be used to populate the PATIENT PAY COMPONENT QUALIFIER (#2195) field of the BPS RESPONSES (#9002313.03) file. Corresponds to NCPDP code C95-KQ and qualifies PATIENT PAY COMPONENT AMOUNT (C93-KN). |
| D19-M1 | MINIMUM AMOUNT | Enter an amount between 0 and 999999.999, 3 decimal digits. | This is the override value that will be used for the payer response for billing requests. The value will be used to populate the MINIMUM AMOUNT (#2219) field of the BPS RESPONSES (#9002313.03) file. Corresponds to NCPDP code D19-M1. |
| D20-M2 | MINIMUM AMOUNT QUALIFIER | Enter override value for the Minimum Amount Qualifier. | This is the override value that will be used for the payer response for billing requests. The value will be used to populate the MAXIMUM AMOUNT QUALIFIER (#2220) field of the BPS RESPONSES (#9002313.03) file. Corresponds to NCPDP code D20-M2 and qualifies MINIMUM AMOUNT (D19-M1). |
| D23-M5 | OTHER PAYER NAME | Enter override value for the Other Payer Name. | This is the override value that will be used for the payer response for billing requests. The value will be used to populate the OTHER PAYER NAME (#2223) field of the BPS RESPONSES (#9002313.03) file. Corresponds to NCPDP code D23-M5. |
| D24-M6 | REMAINING AMOUNT | Enter between 0 and 999999.999, 3 decimal digits. | This is the override value that will be used for the payer response for billing requests. The value will be used to populate the REMAINING AMOUNT (#2224) field of the BPS RESPONSES (#9002313.03) file. Corresponds to NCPDP code D24-M6. |
| D25-M7 | REMAINING AMOUNT QUALIFIER | Enter override value for the Remaining Amount Qualifier. | This is the override value that will be used for the payer response for billing requests. The value will be used to populate the REMAINING AMOUNT QUALIFIER (#2225) field of the BPS RESPONSES (#9002313.03) file.  Corresponds to NCPDP code D25-M7 and qualifies REMAINING AMOUNT (D24-M6). |
| D41-PQ | OTHER PAYER RELATIONSHIP TYPE | Enter override value for the Other Payer Relationship Type. | This is the override value that will be used for the payer response for billing requests. The value will be used to populate the OTHER PAYER RELATIONSHIP TYPE (#2241) field of the BPS RESPONSES (#9002313.03) file. Corresponds to NCPDP code D41-PQ. |

The subroutines GETOVER and SETOVER in BPSTEST will need to be modified to allow the user to enter the new fields we added above to the BPS PAYER RESPONSE OVERRIDES file.

| Subroutine Name | **GETOVER^BPSTEST** | | | |
| --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify | Delete | No Change |
| Current Logic | | | | |
| BPSTEST ;OAK/ELZ - ECME TESTING TOOL ;11/15/07 09:55  ;;1.0;E CLAIMS MGMT ENGINE;\*\*6,7,8,10,11,15,19,20,22,23\*\*;JUN 2004;Build 27  ;;Per VA Directive 6402, this routine should not be modified.  ;  ; Look at BPSTEST1 for additional documentation of the Testing Tool  ;  GETOVER(KEY1,KEY2,BPSORESP,BPSWHERE,BPSTYPE,BPPAYSEQ) ;  ; called by BPSNCPDP to enter overrides for a particular RX  ; INPUT  ; KEY1 - Prescription IEN/Patient IEN  ; KEY2 - Fill Number/Policy Number  ; BPSORESP - Previous response when this claim was processed  ; BPSWHERE - RX Action passed into BPSNCPDP  ; BPSTYPE - R (Reversal), S (Submission), E (Eligibility)  ; BPPAYSEQ - payer sequence 1 - primary, 2 - secondary  ; OUTPUT  ; None - Table BPS PAYER RESPONSE OVERRIDE entry is created.  ;  N BPSTRANS,BPSTIEN,BPSSRESP,DIC,X,Y,DIR,DIK,DA  ;  ; Check if testing is enabled  I '$$CHECK() Q  ;  ; Option can not be run for Date of Death option as it causes errors  I $G(XQY0)["DG DEATH ENTRY" W !,"The testing tool can not be run from Date of Death option" Q  ;  ; Do not run for background jobs  I $D(ZTQUEUED)!(",AREV,CRLB,CRLR,CRLX,CRRL,PC,PL,"[(","\_BPSWHERE\_",")) Q  ;  ; Create Transaction Number  S BPSTRANS=$$IEN59^BPSOSRX(KEY1,KEY2,$S($G(BPPAYSEQ)>0:+BPPAYSEQ,1:1))  ;  . . .  .. ; Additional overrides for E0-E6 (BPS\*1\*19)  .. D PROMPT(BPSTIEN,2.01,"04") ; % sales tax basis pd  .. D PROMPT(BPSTIEN,2.02,11) ; other amount paid qualifier  .. D PROMPT(BPSTIEN,2.03,"01") ; payer id qualifier  .. D PROMPT(BPSTIEN,2.04,"") ; help desk phone# ext  .. D PROMPT(BPSTIEN,2.05,"") ; pro service fee cont/reim amt  .. D PROMPT(BPSTIEN,2.06,"") ; other payer help desk phone# ext  .. D PROMPT(BPSTIEN,2.07,"") ; response intermed auth type id  .. D PROMPT(BPSTIEN,2.08,"") ; response intermed auth id  .. D PROMPT(BPSTIEN,3.01,"") ; response intermed message  .. ;  .. ; E7 overrides (BPS\*1\*20)  .. D PROMPT(BPSTIEN,.11,"") ; quan limit per specific time period  .. D PROMPT(BPSTIEN,.12,"") ; quan limit time period  .. D PROMPT(BPSTIEN,.13,"") ; days supp limit per specific time period  .. D PROMPT(BPSTIEN,.14,"") ; days supp limit time period  .. ; Overrides to test functionality - BPS\*1\*22  .. D PROMPT(BPSTIEN,2.09,"") ; reconciliation id  .. ;  .. D PROMPT(BPSTIEN,2.1,"") ; Patient Pay Amount  ;  W ! D PROMPT(BPSTIEN,.07,0)  Q | | | | |
| Modified Logic (Changes are highlighted) | | | | |
| BPSTEST ;OAK/ELZ - ECME TESTING TOOL ;11/15/07 09:55  ;;1.0;E CLAIMS MGMT ENGINE;\*\*6,7,8,10,11,15,19,20,22,23,24\*\*;JUN 2004;Build 27  ;;Per VA Directive 6402, this routine should not be modified.  ;  ; Look at BPSTEST1 for additional documentation of the Testing Tool  ;  GETOVER(KEY1,KEY2,BPSORESP,BPSWHERE,BPSTYPE,BPPAYSEQ) ;  ; called by BPSNCPDP to enter overrides for a particular RX  ; INPUT  ; KEY1 - Prescription IEN/Patient IEN  ; KEY2 - Fill Number/Policy Number  ; BPSORESP - Previous response when this claim was processed  ; BPSWHERE - RX Action passed into BPSNCPDP  ; BPSTYPE - R (Reversal), S (Submission), E (Eligibility)  ; BPPAYSEQ - payer sequence 1 - primary, 2 - secondary  ; OUTPUT  ; None - Table BPS PAYER RESPONSE OVERRIDE entry is created.  ;  N BPSTRANS,BPSTIEN,BPSSRESP,DIC,X,Y,DIR,DIK,DA  ;  ; Check if testing is enabled  I '$$CHECK() Q  ;  ; Option can not be run for Date of Death option as it causes errors  I $G(XQY0)["DG DEATH ENTRY" W !,"The testing tool can not be run from Date of Death option" Q  ;  ; Do not run for background jobs  I $D(ZTQUEUED)!(",AREV,CRLB,CRLR,CRLX,CRRL,PC,PL,"[(","\_BPSWHERE\_",")) Q  ;  ; Create Transaction Number  S BPSTRANS=$$IEN59^BPSOSRX(KEY1,KEY2,$S($G(BPPAYSEQ)>0:+BPPAYSEQ,1:1))  ;  . . .  .. ; Additional overrides for E0-E6 (BPS\*1\*19)  .. D PROMPT(BPSTIEN,2.01,"04") ; % sales tax basis pd  .. D PROMPT(BPSTIEN,2.02,11) ; other amount paid qualifier  .. D PROMPT(BPSTIEN,2.03,"01") ; payer id qualifier  .. D PROMPT(BPSTIEN,2.04,"") ; help desk phone# ext  .. D PROMPT(BPSTIEN,2.05,"") ; pro service fee cont/reim amt  .. D PROMPT(BPSTIEN,2.06,"") ; other payer help desk phone# ext  .. D PROMPT(BPSTIEN,2.07,"") ; response intermed auth type id  .. D PROMPT(BPSTIEN,2.08,"") ; response intermed auth id  .. D PROMPT(BPSTIEN,3.01,"") ; response intermed message  .. ;  .. ; E7 overrides (BPS\*1\*20)  .. D PROMPT(BPSTIEN,.11,"") ; quan limit per specific time period  .. D PROMPT(BPSTIEN,.12,"") ; quan limit time period  .. D PROMPT(BPSTIEN,.13,"") ; days supp limit per specific time period  .. D PROMPT(BPSTIEN,.14,"") ; days supp limit time period  .. ; Overrides to test functionality - BPS\*1\*22  .. D PROMPT(BPSTIEN,2.09,"") ; reconciliation id  .. ;  .. D PROMPT(BPSTIEN,2.1,"") ; Patient Pay Amount  .. ;  .. D PROMPT(BPSTIEN,4.01,"") ; Maximum Age Qualifier  .. D PROMPT(BPSTIEN,4.02,"") ; Maximum Age  .. D PROMPT(BPSTIEN,4.03,"") ; Maximum Amount  .. D PROMPT(BPSTIEN,4.04,"") ; Maximum Amount Qualifier  .. D PROMPT(BPSTIEN,4.05,"") ; Maximum Amount Time Period  .. D PROMPT(BPSTIEN,4.06,"") ; Maximum Amount Time Period Start Date  .. D PROMPT(BPSTIEN,4.07,"") ; Maximum Amount Time Period End Date  .. D PROMPT(BPSTIEN,4.08,"") ; Maximum Amount Time Period Units  .. D PROMPT(BPSTIEN,4.09,"") ; Minimum Age Qualifier  .. D PROMPT(BPSTIEN,4.1,"") ; Minimum Age  .. D PROMPT(BPSTIEN,4.11,"") ; Other Payer Adjudicated Program Type  .. D PROMPT(BPSTIEN,4.12,"") ; Patient Pay Component Amount  .. D PROMPT(BPSTIEN,4.13,"") ; Patient Pay Component Count  .. D PROMPT(BPSTIEN,4.14,"") ; Patient Pay Component Qualifier  .. D PROMPT(BPSTIEN,4.15,"") ; Minimum Amount  .. D PROMPT(BPSTIEN,4.16,"") ; Minimum Amount Qualifier  .. D PROMPT(BPSTIEN,4.17,"") ; Other Payer Name  .. D PROMPT(BPSTIEN,4.18,"") ; Remaining Amount  .. D PROMPT(BPSTIEN,4.19,"") ; Remaining Amount Qualifier  .. D PROMPT(BPSTIEN,4.2,"") ; Other Payer Relationship Type  ;  W ! D PROMPT(BPSTIEN,.07,0)  Q | | | | |

With the changes for this user story the routine BPSTEST will be too large, exceed SAC standards, so the code was moved to SETOVER^BPSTEST2. This is described in the tables below.

| Subroutine Name | **SETOVER^BPSTEST** | | | |
| --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify | Delete | No Change |
| Current Logic | | | | |
| SETOVER(BPSTRANS,BPSTYPE,BPSDATA) ;  ; called by BPSECMPS to set the override data  ; Input  ; BPSTRANS - Transaction IEN  ; BPSTYPE - B1 for submission, B2 for reversals  ; Output  ; BPSDATA - Passed by reference and updated with appropriate overrides  ;  N BPSTIEN,BPSRRESP,BPSSRESP,BPSPAID,BPSRCNT,BPSRIEN,BPSRCODE,BPSRCD,BPSCOPAY,BPSXXXX,BPSUNDEF  N BPSAJPAY,BPSNFLDT,BPSX  N BPS506,BPS507,BPS513,BPS517  ;  ; Check the Test Flag in set in BPS SETUP  I '$$CHECK() Q  ;  ; If a eligibility, check for specific reversal overrides and set  I BPSTYPE="E1" D  Q  . S BPSRRESP=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.08,"I")  . ;  . ; If the response is Stranded, force an <UNDEF> error  . I BPSRRESP="S" S BPSXXXX=BPSUNDEF  . I BPSRRESP]"" S BPSDATA(1,112)=BPSRRESP  . S BPSDATA(9002313.03,9002313.03,"+1,",501)=$S(BPSRRESP="R":"R",1:"A")  . ;   . ; If the response is accepted, delete the reject code count and codes  . I BPSRRESP="A" K BPSDATA(1,510),BPSDATA(1,511)  . ;   . ; If the response is rejected, delete the rejections returned by payers  . ; and put in the ones entered by the user  . I BPSRRESP="R" D  .. K BPSDATA(1,509),BPSDATA(1,511)  .. S BPSRCNT=0  .. S BPSRIEN=0 F  S BPSRIEN=$O(^BPS(9002313.32,BPSTIEN,1,BPSRIEN)) Q:+BPSRIEN=0 D  ... S BPSRCODE=$P($G(^BPS(9002313.32,BPSTIEN,1,BPSRIEN,0)),"^",1)  ... ; Increment counter and store  ... I BPSRCODE]"" D  .... S BPSRCD=$$GET1^DIQ(9002313.93,BPSRCODE\_",",.01,"E")  .... I BPSRCD]"" S BPSRCNT=BPSRCNT+1,BPSDATA(1,511,BPSRCNT)=BPSRCD  .. ; Store total number of rejections  .. S BPSDATA(1,510)=BPSRCNT  ;  ; If a reversal, check for specific reversal overrides and set  I BPSTYPE="B2" D  . S BPSRRESP=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.05,"I")  . ;  . ; If the response is Stranded, force an <UNDEF> error  . I BPSRRESP="S" S BPSXXXX=BPSUNDEF  . I BPSRRESP]"" S BPSDATA(1,112)=$S(BPSRRESP="D":"S",1:BPSRRESP)  . S BPSDATA(9002313.03,9002313.03,"+1,",501)=$S(BPSRRESP="R":"R",1:"A")  . ;  . ; If the response is accepted or duplicate, kill the reject code count and codes  . I BPSRRESP="A"!(BPSRRESP="D") K BPSDATA(1,510),BPSDATA(1,511)  . ;  . ; If the response is rejected, set the reject codes  . I BPSRRESP="R" D SETREJ(BPSTRANS)  ;  ; If a submission, check for specific submission overrides and set   I BPSTYPE="B1" D  . ; Get submission response  . S BPSSRESP=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.03,"I")  . ;  . ; If the response is Stranded, force an <UNDEF> error  . I BPSSRESP="S" S BPSXXXX=BPSUNDEF  . ;  . ; If BPSSRESP exists, file it  . I BPSSRESP]"" D  .. S BPSDATA(1,112)=BPSSRESP  .. S BPSDATA(9002313.03,9002313.03,"+1,",501)=$S(BPSSRESP="R":"R",1:"A")  .. ; If payable or duplicate, get the BPSPAID amount and file it if it  .. ; exists. Also delete any reject codes  .. I BPSSRESP="P"!(BPSSRESP="D") D  ... ;  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",2.1,"I") ; 505-F5 Patient Pay Amount  ... I BPSX]"" S BPSDATA(1,"505")=$$DFF^BPSECFM(BPSX,10)  ... ;  ... S BPSPAID=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.04,"I")  ... I BPSPAID]"" S BPSDATA(1,509)=$$DFF^BPSECFM(BPSPAID,8) ; 509 Total amount paid  ... ;  ... K BPSDATA(1,510),BPSDATA(1,511) ; kill Reject Count (510) and Reject Code (511)  ... ;  ... S BPSCOPAY=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.06,"I")  ... I BPSCOPAY]"" S BPSDATA(1,518)=$$DFF^BPSECFM(BPSCOPAY,8) ; 518 Copay Amount  ... ;  ... S BPS506=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.15,"I")  ... I BPS506]"" S BPSDATA(1,506)=$$DFF^BPSECFM(BPS506,8) ; 506 Ingredient Cost Paid  ... ;  ... S BPS507=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.16,"I")  ... I BPS507]"" S BPSDATA(1,507)=$$DFF^BPSECFM(BPS507,8) ; 507 Dispensing Fee Paid  ... ;  ... S BPS513=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.17,"I")  ... I BPS513]"" S BPSDATA(1,513)=$$DFF^BPSECFM(BPS513,8) ; 513 Remaining Deductible Amount  ... ;  ... S BPS517=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.18,"I")  ... I BPS517]"" S BPSDATA(1,517)=$$DFF^BPSECFM(BPS517,8) ; 517 Amount Applied to Periodic Deductible  ... Q  .. ;  .. I BPSSRESP="P"!(BPSSRESP="D")!(BPSSRESP="R") D  ... ; D1-D9 fields (BPS\*1\*15)  ... S BPSAJPAY=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.1,"I") ; Adjudicated Payment Type  ... I BPSAJPAY]"" S BPSDATA(1,1028)=$$NFF^BPSECFM(BPSAJPAY,2)  ... S BPSNFLDT=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.09,"I") ; Override Next Available Fill  ... I BPSNFLDT]"" S BPSDATA(1,2004)=$$DTF1^BPSECFM(BPSNFLDT)  ... ;  ... ; E0-E6 overrides (BPS\*1\*19)  ... ; PERCENTAGE SALES TAX BASIS PAID  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",2.01,"I")  ... I BPSX]"" S BPSDATA(1,561)=BPSX  ... ; OTHER AMOUNT PAID QUALIFIER and associated field  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",2.02,"I")  ... I BPSX]"" S BPSDATA(1,564,1)=$$NFF^BPSECFM(BPSX,2),BPSDATA(1,565,1)=$$DFF^BPSECFM(5.64,8),BPSDATA(1,563)=1  ... ; PAYER ID QUALIFIER  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",2.03,"I")  ... I BPSX]"" S BPSDATA(9002313.03,9002313.03,"+1,",568)=BPSX  ... ; HELP DESK TELEPHONE NUMBER EXTENSION  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",2.04,"I")  ... I BPSX]"" S BPSDATA(1,"2022")=$$NFF^BPSECFM(BPSX,8)  ... ; PROFESSIONAL SERVICE FEE CONTRACTED/REIMURSEMENT AMOUNT  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",2.05,"I")  ... I BPSX]"" S BPSDATA(1,"2033")=$$DFF^BPSECFM(BPSX,8)  ... ; OTHER PAYER HELPDESK TELEPHONE EXTENSION  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",2.06,"I")  ... I BPSX]"" S BPSDATA(1,"2023",1)=$$NFF^BPSECFM(BPSX,8),BPSDATA(1,338,1)="01"  ... ; RESPONSE INTERMEDIARY AUTHORIZATION TYPE ID and associated fields  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",2.07,"I")  ... I BPSX]"" S BPSDATA(1,"2053",1)=$$NFF^BPSECFM(BPSX,2),BPSDATA(1,2052)=1  ... ; RESPONSE INTERMEDIARY AUTHORIZATION ID and associated fields  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",2.08,"I")  ... I BPSX]"" S BPSDATA(1,"2054",1)=$$ANFF^BPSECFM(BPSX,20),BPSDATA(1,2052)=1  ... ; INTERMEDIARY MESSAGE and associated fields  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",3.01,"I")  ... I BPSX]"" S BPSDATA(1,"2051",1)=$$ANFF^BPSECFM(BPSX,200),BPSDATA(1,2052)=1  ... ; (BPS\*1\*22)  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",2.09,"I") ; B98-34 reconciliation id  ... I BPSX]"" S BPSDATA(1,"2098")=$$ANFF^BPSECFM(BPSX,30)  ... ;  ... ; E7 overrides (BPS\*1\*20)  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.11,"I") I BPSX'="" D ; B88-3R quantity limit per spec time period  .... S BPSDATA(1,2087)=1 ; count field  .... S BPSDATA(1,2088,1)=$$NFF^BPSECFM(BPSX,10) ; data from override file  .... Q  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.12,"I") I BPSX'="" D ; B89-3S quantity limit time period  .... S BPSDATA(1,2087)=1 ; count field  .... S BPSDATA(1,2089,1)=$$NFF^BPSECFM(BPSX,5) ; data from override file  .... Q  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.13,"I") I BPSX'="" D ; B91-3W days supply limit per spec time period  .... S BPSDATA(1,2090)=1 ; count field  .... S BPSDATA(1,2091,1)=$$NFF^BPSECFM(BPSX,3) ; data from override file  .... Q  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.14,"I") I BPSX'="" D ; B92-3X days supply limit time period  .... S BPSDATA(1,2090)=1 ; count field  .... S BPSDATA(1,2092,1)=$$NFF^BPSECFM(BPSX,5) ; data from override file  .... Q  ... Q  .. ;  .. ; If rejected, get the rejection code and file them  .. ; Also, delete the BPSPAID amount  .. I BPSSRESP="R" D  ... ; Delete old rejections and BPSPAID amount  ... K BPSDATA(1,509),BPSDATA(1,511)  ... ; Loop through rejections and store  ... S BPSRCNT=0  ... S BPSRIEN=0 F S BPSRIEN=$O(^BPS(9002313.32,BPSTIEN,1,BPSRIEN)) Q:+BPSRIEN=0 D  .... S BPSRCODE=$P($G(^BPS(9002313.32,BPSTIEN,1,BPSRIEN,0)),"^",1)  .... ; Increment counter and store  .... I BPSRCODE]"" D  ..... S BPSRCD=$$GET1^DIQ(9002313.93,BPSRCODE\_",",.01,"E")  ..... I BPSRCD]"" S BPSRCNT=BPSRCNT+1,BPSDATA(1,511,BPSRCNT)=BPSRCD  ... ; Store total number of rejections  ... S BPSDATA(1,510)=BPSRCNT  Q | | | | |
| Modified Logic (Changes are highlighted) | | | | |
| SETOVER(BPSTRANS,BPSTYPE,BPSDATA) ;  ; called by BPSECMPS to set the override data  ; Input  ; BPSTRANS - Transaction IEN  ; BPSTYPE - B1 for submission, B2 for reversals  ; Output  ; BPSDATA - Passed by reference and updated with appropriate overrides  ;  N BPSTIEN,BPSRRESP,BPSSRESP,BPSPAID,BPSRCNT,BPSRIEN,BPSRCODE,BPSRCD,BPSCOPAY,BPSXXXX,BPSUNDEF  N BPSAJPAY,BPSNFLDT,BPSX  N BPS506,BPS507,BPS513,BPS517  ;  ; Check the Test Flag in set in BPS SETUP  I '$$CHECK() Q   ;  ; Check if the Transaction Number is defined in BPS RESPONSE OVERRIDES  S BPSTIEN=$O(^BPS(9002313.32,"B",BPSTRANS,""))  I BPSTIEN="" Q  ;  ; If a eligibility, check for specific reversal overrides and set  ; If a reversal, check for specific reversal overrides and set  ; If a submission, check for specific submission overrides and set  ; the code for the above checks was moved to SETOVER^BPSTEST2  D SETOVER^BPSTEST2  Q | | | | |

The code that is in Bold below was the code that will be added for this user story.

| Subroutine Name | **SETOVER^BPSTEST2** | | | |
| --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify | Delete | No Change |
| Current Logic | | | | |
| n/a – new subroutine | | | | |
| Modified Logic (Changes are highlighted) | | | | |
| BPSTEST2 ;AITC/CKB - ECME TESTING TOOL ;5/31/2018  ;;1.0;E CLAIMS MGMT ENGINE;\*\*24\*\*;  ;;Per VA Directive 6402, this routine should not be modified.  ;  Q  ; SETOVER ;  ; the following code was from SETOVER^BPSTEST and is called by SETOVER^BPSTEST  ;  ; If a eligibility, check for specific reversal overrides and set  I BPSTYPE="E1" D  Q  . S BPSRRESP=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.08,"I")  . ;  . ; If the response is Stranded, force an <UNDEF> error  . I BPSRRESP="S" S BPSXXXX=BPSUNDEF  . I BPSRRESP]"" S BPSDATA(1,112)=BPSRRESP  . S BPSDATA(9002313.03,9002313.03,"+1,",501)=$S(BPSRRESP="R":"R",1:"A")  . ;   . ; If the response is accepted, delete the reject code count and codes  . I BPSRRESP="A" K BPSDATA(1,510),BPSDATA(1,511)  . ;   . ; If the response is rejected, delete the rejections returned by payers  . ; and put in the ones entered by the user  . I BPSRRESP="R" D  .. K BPSDATA(1,509),BPSDATA(1,511)  .. S BPSRCNT=0  .. S BPSRIEN=0 F  S BPSRIEN=$O(^BPS(9002313.32,BPSTIEN,1,BPSRIEN)) Q:+BPSRIEN=0 D  ... S BPSRCODE=$P($G(^BPS(9002313.32,BPSTIEN,1,BPSRIEN,0)),"^",1)  ... ; Increment counter and store  ... I BPSRCODE]"" D  .... S BPSRCD=$$GET1^DIQ(9002313.93,BPSRCODE\_",",.01,"E")  .... I BPSRCD]"" S BPSRCNT=BPSRCNT+1,BPSDATA(1,511,BPSRCNT)=BPSRCD  .. ; Store total number of rejections  .. S BPSDATA(1,510)=BPSRCNT  ;  ; If a reversal, check for specific reversal overrides and set  I BPSTYPE="B2" D  . S BPSRRESP=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.05,"I")  . ;  . ; If the response is Stranded, force an <UNDEF> error  . I BPSRRESP="S" S BPSXXXX=BPSUNDEF  . I BPSRRESP]"" S BPSDATA(1,112)=$S(BPSRRESP="D":"S",1:BPSRRESP)  . S BPSDATA(9002313.03,9002313.03,"+1,",501)=$S(BPSRRESP="R":"R",1:"A")  . ;  . ; If the response is accepted or duplicate, kill the reject code count and codes  . I BPSRRESP="A"!(BPSRRESP="D") K BPSDATA(1,510),BPSDATA(1,511)  . ;  . ; If the response is rejected, set the reject codes  . I BPSRRESP="R" D SETREJ^BPSTEST(BPSTRANS)  ;  ; If a submission, check for specific submission overrides and set   I BPSTYPE="B1" D  . ; Get submission response  . S BPSSRESP=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.03,"I")  . ;  . ; If the response is Stranded, force an <UNDEF> error  . I BPSSRESP="S" S BPSXXXX=BPSUNDEF  . ;  . ; If BPSSRESP exists, file it  . I BPSSRESP]"" D  .. S BPSDATA(1,112)=BPSSRESP  .. S BPSDATA(9002313.03,9002313.03,"+1,",501)=$S(BPSSRESP="R":"R",1:"A")  .. ; If payable or duplicate, get the BPSPAID amount and file it if it  .. ; exists. Also delete any reject codes  .. I BPSSRESP="P"!(BPSSRESP="D") D  ... ;  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",2.1,"I") ; 505-F5 Patient Pay Amount  ... I BPSX]"" S BPSDATA(1,"505")=$$DFF^BPSECFM(BPSX,10)  ... ;  ... S BPSPAID=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.04,"I")  ... I BPSPAID]"" S BPSDATA(1,509)=$$DFF^BPSECFM(BPSPAID,8) ; 509 Total amount paid  ... ;  ... K BPSDATA(1,510),BPSDATA(1,511) ; kill Reject Count (510) and Reject Code (511)  ... ;  ... S BPSCOPAY=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.06,"I")  ... I BPSCOPAY]"" S BPSDATA(1,518)=$$DFF^BPSECFM(BPSCOPAY,8) ; 518 Copay Amount  ... ;  ... S BPS506=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.15,"I")  ... I BPS506]"" S BPSDATA(1,506)=$$DFF^BPSECFM(BPS506,8) ; 506 Ingredient Cost Paid  ... ;  ... S BPS507=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.16,"I")  ... I BPS507]"" S BPSDATA(1,507)=$$DFF^BPSECFM(BPS507,8) ; 507 Dispensing Fee Paid  ... ;  ... S BPS513=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.17,"I")  ... I BPS513]"" S BPSDATA(1,513)=$$DFF^BPSECFM(BPS513,8) ; 513 Remaining Deductible Amount  ... ;  ... S BPS517=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.18,"I")  ... I BPS517]"" S BPSDATA(1,517)=$$DFF^BPSECFM(BPS517,8) ; 517 Amount Applied to Periodic Deductible  ... Q  .. ;  .. I BPSSRESP="P"!(BPSSRESP="D")!(BPSSRESP="R") D  ... ; D1-D9 fields (BPS\*1\*15)  ... S BPSAJPAY=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.1,"I") ; Adjudicated Payment Type  ... I BPSAJPAY]"" S BPSDATA(1,1028)=$$NFF^BPSECFM(BPSAJPAY,2)  ... S BPSNFLDT=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.09,"I") ; Override Next Available Fill  ... I BPSNFLDT]"" S BPSDATA(1,2004)=$$DTF1^BPSECFM(BPSNFLDT)  ... ;  ... ; E0-E6 overrides (BPS\*1\*19)  ... ; PERCENTAGE SALES TAX BASIS PAID  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",2.01,"I")  ... I BPSX]"" S BPSDATA(1,561)=BPSX  ... ; OTHER AMOUNT PAID QUALIFIER and associated field  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",2.02,"I")  ... I BPSX]"" S BPSDATA(1,564,1)=$$NFF^BPSECFM(BPSX,2),BPSDATA(1,565,1)=$$DFF^BPSECFM(5.64,8),BPSDATA(1,563)=1  ... ; PAYER ID QUALIFIER  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",2.03,"I")  ... I BPSX]"" S BPSDATA(9002313.03,9002313.03,"+1,",568)=BPSX  ... ; HELP DESK TELEPHONE NUMBER EXTENSION  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",2.04,"I")  ... I BPSX]"" S BPSDATA(1,"2022")=$$NFF^BPSECFM(BPSX,8)  ... ; PROFESSIONAL SERVICE FEE CONTRACTED/REIMURSEMENT AMOUNT  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",2.05,"I")  ... I BPSX]"" S BPSDATA(1,"2033")=$$DFF^BPSECFM(BPSX,8)  ... ; OTHER PAYER HELPDESK TELEPHONE EXTENSION  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",2.06,"I")  ... I BPSX]"" S BPSDATA(1,"2023",1)=$$NFF^BPSECFM(BPSX,8),BPSDATA(1,338,1)="01"  ... ; RESPONSE INTERMEDIARY AUTHORIZATION TYPE ID and associated fields  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",2.07,"I")  ... I BPSX]"" S BPSDATA(1,"2053",1)=$$NFF^BPSECFM(BPSX,2),BPSDATA(1,2052)=1  ... ; RESPONSE INTERMEDIARY AUTHORIZATION ID and associated fields  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",2.08,"I")  ... I BPSX]"" S BPSDATA(1,"2054",1)=$$ANFF^BPSECFM(BPSX,20),BPSDATA(1,2052)=1  ... ; INTERMEDIARY MESSAGE and associated fields  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",3.01,"I")  ... I BPSX]"" S BPSDATA(1,"2051",1)=$$ANFF^BPSECFM(BPSX,200),BPSDATA(1,2052)=1  ... ; (BPS\*1\*22)  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",2.09,"I") ; B98-34 reconciliation id  ... I BPSX]"" S BPSDATA(1,"2098")=$$ANFF^BPSECFM(BPSX,30) **... ;  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",4.01,"I") ; 931-F8 maximum age qualifier  ... I BPSX]"" S BPSDATA(1,931)=$$ANFF^BPSECFM(BPSX,1)  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",4.02,"I") ; 932-GA maximum age  ... I BPSX]"" S BPSDATA(1,932)=$$NFF^BPSECFM(BPSX,3)  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",4.03,"I") ; 933-GB maximum amount  ... I BPSX]"" S BPSDATA(1,933)=$$DFF^BPSECFM(BPSX,10)  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",4.04,"I") ; 934-GC maximum amt qualifier  ... I BPSX]"" S BPSDATA(1,934)=$$ANFF^BPSECFM(BPSX,2)  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",4.05,"I") ; 935-GF maximum amt time period  ... I BPSX]"" S BPSDATA(1,935)=$$ANFF^BPSECFM(BPSX,2)  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",4.06,"I") ; 936-GG maximum amt time period start date  ... I BPSX]"" S BPSDATA(1,936)=$$DTF1^BPSECFM(BPSX)  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",4.07,"I") ; 937-GH maximum amt time period end date  ... I BPSX]"" S BPSDATA(1,937)=$$DTF1^BPSECFM(BPSX)  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",4.08,"I") ; 938-GJ maximum amt time period units  ... I BPSX]"" S BPSDATA(1,938)=$$NFF^BPSECFM(BPSX,4)  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",4.09,"I") ; 943-GQ minimum age qualifier  ... I BPSX]"" S BPSDATA(1,943)=$$ANFF^BPSECFM(BPSX,1)  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",4.1,"I") ; 944-GR minimum age  ... I BPSX]"" S BPSDATA(1,944)=$$NFF^BPSECFM(BPSX,3)  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",4.11,"I") ; C47-9T other payer adjudicate prog type  ... I BPSX]"" S BPSDATA(1,2147)=$$ANFF^BPSECFM(BPSX,30)  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",4.12,"I") ; C93-KN patient pay component amount  ... I BPSX]"" S BPSDATA(1,2193)=$$DFF^BPSECFM(BPSX,8)  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",4.13,"I") ; C94-KP patient pay component count  ... I BPSX]"" S BPSDATA(1,2194)=$$NFF^BPSECFM(BPSX,4)  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",4.14,"I") ; C95-KQ patient payer component qualifier  ... I BPSX]"" S BPSDATA(1,2195)=$$ANFF^BPSECFM(BPSX,2)  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",4.15,"I") ; D19-M1 minimum amount  ... I BPSX]"" S BPSDATA(1,2219)=$$DFF^BPSECFM(BPSX,10)  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",4.16,"I") ; D20-M2 minimum amount qualifier  ... I BPSX]"" S BPSDATA(1,2220)=$$ANFF^BPSECFM(BPSX,3)  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",4.17,"I") ; D23-M5 other payer name  ... I BPSX]"" S BPSDATA(1,2223)=$$ANFF^BPSECFM(BPSX,30)  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",4.18,"I") ; D24-M6 remaining amount  ... I BPSX]"" S BPSDATA(1,2224)=$$DFF^BPSECFM(BPSX,10,3)  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",4.19,"I") ; D25-M7 remaining amount qualifier  ... I BPSX]"" S BPSDATA(1,2225)=$$ANFF^BPSECFM(BPSX,3)  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",4.2,"I") ; D41-PA other payer relationship type  ... I BPSX]"" S BPSDATA(1,2241)=$$ANFF^BPSECFM(BPSX,3)**  **... ;**  ... ; E7 overrides (BPS\*1\*20)  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.11,"I") I BPSX'="" D     ; B88-3R quantity limit per spec time period  .... S BPSDATA(1,2087)=1 ; count field  .... S BPSDATA(1,2088,1)=$$NFF^BPSECFM(BPSX,10) ; data from override file  .... Q  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.12,"I") I BPSX'="" D     ; B89-3S quantity limit time period  .... S BPSDATA(1,2087)=1 ; count field  .... S BPSDATA(1,2089,1)=$$NFF^BPSECFM(BPSX,5) ; data from override file  .... Q  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.13,"I") I BPSX'="" D     ; B91-3W days supply limit per spec time period  .... S BPSDATA(1,2090)=1 ; count field  .... S BPSDATA(1,2091,1)=$$NFF^BPSECFM(BPSX,3) ; data from override file  .... Q  ... S BPSX=$$GET1^DIQ(9002313.32,BPSTIEN\_",",.14,"I") I BPSX'="" D     ; B92-3X days supply limit time period  .... S BPSDATA(1,2090)=1 ; count field  .... S BPSDATA(1,2092,1)=$$NFF^BPSECFM(BPSX,5) ; data from override file  .... Q  ... Q  .. ;  .. ; If rejected, get the rejection code and file them  .. ; Also, delete the BPSPAID amount  .. I BPSSRESP="R" D  ... ; Delete old rejections and BPSPAID amount  ... K BPSDATA(1,509),BPSDATA(1,511)  ... ; Loop through rejections and store  ... S BPSRCNT=0  ... S BPSRIEN=0 F  S BPSRIEN=$O(^BPS(9002313.32,BPSTIEN,1,BPSRIEN)) Q:+BPSRIEN=0 D  .... S BPSRCODE=$P($G(^BPS(9002313.32,BPSTIEN,1,BPSRIEN,0)),"^",1)  .... ; Increment counter and store  .... I BPSRCODE]"" D  ..... S BPSRCD=$$GET1^DIQ(9002313.93,BPSRCODE\_",",.01,"E")  ..... I BPSRCD]"" S BPSRCNT=BPSRCNT+1,BPSDATA(1,511,BPSRCNT)=BPSRCD  ... ; Store total number of rejections  ... S BPSDATA(1,510)=BPSRCNT  ;  Q | | | | |

GETBPS3^BPSECX0 will need to be modified to include all the new outgoing fields to ensure that they will be added to the BPS array, even if the field is always blank. The BPS array is used when the claim is built during the claim submission process and when the user performs the RED Resubmit with Edits action.

| Subroutine Name | **GETBPS3^BPSECX0** | | | |
| --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify | Delete | No Change |
| Current Logic | | | | |
| BPSECX0 ;BHAM ISC/FCS/DRS/VA/DLF - Retrieve Claim submission record ;05/17/2004  ;;1.0;E CLAIMS MGMT ENGINE;\*\*1,5,8,10,15,19,23\*\*;JUN 2004;Build 27  ;;Per VA Directive 6402, this routine should not be modified.  . . .  ;Retrieve data in TRANSACTIONS multiple in BPS CLAIMS  ; CLAIMIEN = ien in BPS CLAIMS (#9002313.02)  ; TRXIEN = ien in TRANSACTIONS (#9002313.0201)  ; BPS - Passed by reference  ; returns: BPS(9002313.0201,TRXIEN,field #,"I") = internal format value  GETBPS3(CLAIMIEN,TRXIEN,BPS) ;called from BPSECA1  ;  Q:$G(CLAIMIEN)="" Q:$G(TRXIEN)="" ; must have both  ;  N D0,DA,DIC,DIQ,DIQ2,DR  ;  ; There are other alphanumeric fields that could be added but since they are for segments that are not  ; supported by E1, B1, B3 transactions and/or not segments not used by VA. These can be added later, if  ; needed for those segments.  S DIC=9002313.02,DR="400",DR(9002313.0201)="113:996" ; all TRANSACTION fields  S DR(9002313.0201)=DR(9002313.0201)\_";1023:1032"  S DR(9002313.0201)=DR(9002313.0201)\_";2024:2032;2039:2043"  S DR=DR\_";1093;2013:2021;2034;2035;2037"  S DR=DR\_";2056:2061;2095:2097;2101;2102"  S DA=CLAIMIEN,DA(9002313.0201)=TRXIEN ; IEN and sub-file IEN  S DIQ="BPS",DIQ(0)="I" ; "I" for internal format  D EN^DIQ1  ;  ; Copy Prescriber Phone Number (498.12) to field 498 as this is where BPSOSH2  ; expects to find it. This works for now but if we implement the Prior Auth  ; segment (which has multiple field labelled 498), a more complete solution  ; will need to be found  S BPS(9002313.0201,TRXIEN,498,"I")=$G(BPS(9002313.0201,TRXIEN,498.12,"I"))  Q  ; | | | | |
| Modified Logic (Changes are highlighted) | | | | |
| BPSECX0 ;BHAM ISC/FCS/DRS/VA/DLF - Retrieve Claim submission record ;05/17/2004  ;;1.0;E CLAIMS MGMT ENGINE;\*\*1,5,8,10,15,19,23,24\*\*;JUN 2004;Build 27  ;;Per VA Directive 6402, this routine should not be modified.  . . .  ;Retrieve data in TRANSACTIONS multiple in BPS CLAIMS  ; CLAIMIEN = ien in BPS CLAIMS (#9002313.02)  ; TRXIEN = ien in TRANSACTIONS (#9002313.0201)  ; BPS - Passed by reference  ; returns: BPS(9002313.0201,TRXIEN,field #,"I") = internal format value  GETBPS3(CLAIMIEN,TRXIEN,BPS) ;called from BPSECA1  ;  Q:$G(CLAIMIEN)="" Q:$G(TRXIEN)="" ; must have both  ;  N D0,DA,DIC,DIQ,DIQ2,DR  ;  ; There are other alphanumeric fields that could be added but since they are for segments that are not  ; supported by E1, B1, B3 transactions and/or not segments not used by VA. These can be added later, if  ; needed for those segments.  S DIC=9002313.02,DR="400",DR(9002313.0201)="113:996" ; all TRANSACTION fields  S DR(9002313.0201)=DR(9002313.0201)\_";1023:1032"  S DR(9002313.0201)=DR(9002313.0201)\_";2024:2032;2039:2043"  S DR(9002313.0201)=DR(9002313.0201)\_";1093;2013:2021;2034;2035;2037"  S DR(9002313.0201)=DR(9002313.0201)\_";2056:2061;2095:2097;2101;2102"  ; new fields added in 2017 NCPDP updates  S DR(9002313.0201)=DR(9002313.0201)\_";2147;2149;2150;2151;2160;2190;2191"  S DR(9002313.0201)=DR(9002313.0201)\_";2191;2192;2198;2199;2201;2202;2214"  S DR(9002313.0201)=DR(9002313.0201)\_";2216;2217;2218;2221;2222;2251;2252"  S DR(9002313.0201)=DR(9002313.0201)\_";2253;2257;2260;2261;2263"  ;  S DA=CLAIMIEN,DA(9002313.0201)=TRXIEN ; IEN and sub-file IEN  S DIQ="BPS",DIQ(0)="I" ; "I" for internal format  D EN^DIQ1  ;  ; Copy Prescriber Phone Number (498.12) to field 498 as this is where BPSOSH2  ; expects to find it. This works for now but if we implement the Prior Auth  ; segment (which has multiple field labelled 498), a more complete solution  ; will need to be found  S BPS(9002313.0201,TRXIEN,498,"I")=$G(BPS(9002313.0201,TRXIEN,498.12,"I"))  Q  ; | | | | |

This story calls for taking a field (Reconciliation ID) from the response to a primary claim and sending it on the secondary claim request. The system already does this with other fields. For example, the amount paid on the primary claim comes in on the response as field 509-F9, Total Amount Paid, and is sent on the secondary claim request as field 431-DV, Other Payer Amount Paid. To accomplish this, the following happens.

* The system saves the field in the BPS RESPONSES file when the response is received and processed.
* When the secondary claim is initiated, the system puts the field on the BPS REQUEST.
* From there, it is copied to the BPS TRANSACTION.
* It is pulled from the BPS TRANSACTION and put onto the claim in BPS CLAIMS when the claim is created.
* The value stored in BPS CLAIMS is put onto the claim request that is sent to the payer.

Incoming BPS BPS BPS BPS Outgoing

Claim 🡪 RESPONSES 🡪 REQUESTS 🡪 TRAN- 🡪 CLAIMS 🡪 Claim

Response ACTION Request

The new functionality called for in this story will work in the same way. Some of the existing logic will handle the new fields without any modifications.

This section of the SDD will generally follow the above flow, from capturing the Reconciliation ID on the incoming claim response to sending the Other Payer Reconciliation ID field on the outgoing claim request. At the end of the SDD are sections covering the CRI/VER screens, and the PRO Option and RED Action.

**Fields in Each File**

The file BPS RESPONSES already has the field necessary for this story. The new field will need to be added to the files BPS REQUESTS, BPS TRANSACTION, and BPS CLAIMS.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| File | Field | New/Existing | Length | Format |
| Incoming Claim Response | B98-34 RECONCILIATION ID | Existing | 30 | free text |
| BPS RESPONSES | RECONCILIATION ID (#2098) | Existing | 30 | free text |
| BPS REQUESTS | OTHER PAYER RECONCILIATION ID (#.11) | New | 30 | free text |
| BPS TRANSACTION | OTHER PAYER RECONCILIATION ID (#.11) | New | 30 | free text |
| BPS CLAIMS | OTHER PAYER RECONCILIATION ID (#2149) | New | 30 | free text |
| Outgoing Claim Request | C49-9V OTHER PAYER RECONCILIATION ID | New | 30 | free text |

A new field will be added to file# 9002313.77, BPS REQUESTS, sub-file# 9002313.778, COB OTHER PAYERS, and a new field will be added to file# 9002313.59, BPS TRANSACTION, sub-file# 9002313.5914, COB OTHER PAYERS, and a new field will be added to file# 9002313.02, sub-file# 9002313.0401, COB OTHER PAYMENTS.

|  |  |
| --- | --- |
| Field Attributes | Values |
| FILE | 9002313.77, BPS REQUESTS |
| SUB-FILE | 9002313.778, COB OTHER PAYERS |
| FIELD NAME | OTHER PAYER RECONCILIATION ID |
| FIELD NUMBER | .11 |
| NODE;PIECE | ^BPS(9002313.77,D0,8,D1,0);11 |
| FIELD TYPE | Free Text |
| FIELD LENGTH | 30 |
| HELP PROMPT | Enter an ID between 1 and 30 characters. |
| DESCRIPTION | NCPDP field C49-9V – Other Payer Reconciliation ID. Reconciliation ID (B98-34) as reported by the Other Payer for Paid/Accepted transactions OR for Information Reporting transactions, the designated default value for reporting a previous payer's rejected response as designated in the Other Payer Reject Code value(s) reported in the COB claim. |

|  |  |
| --- | --- |
| Field Attributes | Values |
| FILE | 9002313.59, BPS TRANSACTION |
| SUB-FILE | 9002313.5914, COB OTHER PAYERS |
| FIELD NAME | OTHER PAYER RECONCILIATION ID |
| FIELD NUMBER | .11 |
| NODE;PIECE | ^BPST(D0,14,D1,0);11 |
| FIELD TYPE | Free Text |
| FIELD LENGTH | 30 |
| HELP PROMPT | Enter an ID between 1 and 30 characters. |
| DESCRIPTION | NCPDP field C49-9V – Other Payer Reconciliation ID. Reconciliation ID (B98-34) as reported by the Other Payer for Paid/Accepted transactions OR for Information Reporting transactions, the designated default value for reporting a previous payer's rejected response as designated in the Other Payer Reject Code value(s) reported in the COB claim. |

|  |  |
| --- | --- |
| Field Attributes | Values |
| FILE | 9002313.02, BPS CLAIMS |
| SUB-FILE | 9002313.0401, COB OTHER PAYMENTS |
| FIELD NAME | OTHER PAYER RECONCILIATION ID |
| FIELD NUMBER | 2149 |
| NODE;PIECE | ^BPSC(D0,400,D1,337,D2,0);11 |
| FIELD TYPE | Free Text |
| FIELD LENGTH | 30 |
| HELP PROMPT | Enter an ID between 1 and 30 characters. |
| DESCRIPTION | NCPDP field C49-9V – Other Payer Reconciliation ID. Reconciliation ID (B98-34) as reported by the Other Payer for Paid/Accepted transactions OR for Information Reporting transactions, the designated default value for reporting a previous payer's rejected response as designated in the Other Payer Reject Code value(s) reported in the COB claim. |

The tables below describe how the files BPS REQUESTS, BPS TRANSACTION and BPS CLAIMS will be added to the build.

|  |  |
| --- | --- |
| File Name | BPS REQUESTS |
| Send Full or Partial DD | PARTIAL |
| Data Dictionary Number | 9002313.77 |
| Sub-File Number | 9002313.778 |
| Field Number | .11 |
| Update the Data Dictionary | YES |
| Send Security Codes | NO |
| Screen to Determine DD Update | N/A |
| Data Comes with File | NO |

|  |  |
| --- | --- |
| File Name | BPS TRANSACTION |
| Send Full or Partial DD | PARTIAL |
| Data Dictionary Number | 9002313.59 |
| Sub-File Number | 9002313.5914 |
| Field Number | .11 |
| Update the Data Dictionary | YES |
| Send Security Codes | NO |
| Screen to Determine DD Update | N/A |
| Data Comes with File | NO |

|  |  |
| --- | --- |
| File Name | BPS CLAIMS |
| Send Full or Partial DD | PARTIAL |
| Data Dictionary Number | 9002313.02 |
| Sub-File Number | 9002313.0401 |
| Field Number | 2149 |
| Update the Data Dictionary | YES |
| Send Security Codes | NO |
| Screen to Determine DD Update | N/A |
| Data Comes with File | NO |

**Capturing the Incoming Reconciliation ID**

The field Reconciliation ID, B98-34, is already being captured in file# 9002313.03, BPS REPONSES, field # 2098, RECONCILIATION ID. No change necessary.

**BPS REQUESTS**

The claim submission process can be initiated from many places within the system, but they all eventually call EN^BPSNCPDP. When that function is called, sometimes COB data has already been pulled and set into a local array (passed into EN^BPSNCPDP as the 16th parameter, BPSPRDAT). If that array has already been created, it will automatically include the Patient Pay Amount from the primary claim response.

(The two procedures which pass the COB array into EN^BPSNCPDP are

* SUBMCLM^BPSPRRX2
  + Called from
    - SECONDRY^BPSPRRX
      * COB array built in $$PRIMDATA^BPSPRRX6 and $$SECDATA^BPSPRRX6; see SECONDRY^BPSPRRX for calls
    - PRIMARY^BPSPRRX4
      * News the array BPSDAT in PRIMARY^BPSPRRX4
      * Never sets up the array
    - SECNOPRM^BPSPRRX5
      * COB array built in $$PRIMDATA^BPSPRRX6 and $$SECDATA^BPSPRRX6; see SECNOPRM^BPSPRRX for calls
* DOSELCTD^BPSRES
  + Calls PROMPTS^BPSRES
    - COB array built in $$PRIMDATA^BPSPRRX6 and $$SECDATA^BPSPRRX6; see PROMPTS^BPSRES for calls

(So in all instances, the COB array is set up in $$PRIMDATA^BPSPRRX6 and $$SECDATA^BPSPRRX6.

($$PRIMDATA builds COBARRAY(“OTHER PAYER”,BPSPIEN,0) from fields on BPS RESPONSES. $$SECDATA builds BPSPRDAT(“OTHER PAYER”,BPSPIEN,0) from fields on BPS TRANSACTION.)

When resubmitting a secondary claim, SECDATA^BPSPRRX6 pulls COB data from the BPS TRANSACTION entry. SECDATA^BPSPRRX6 will not need to be updated; it will pull the new field, OTHER PAYER-PAT RESP AMOUNT as written.

SECDATA+44 . S BPSPRDAT("OTHER PAYER",COBPIEN,0)=$G(^BPST(IEN59SEC,14,COBPIEN,0))

MKRQST^BPSOSRX3 (called only by REQST^BPSOSRX) creates an entry in the BPS REQUESTS file. That function will be modified to pull the RECONCILIATION ID from the primary claim’s response when creating the new entry in BPS REQUESTS for the secondary claim.

| Subroutine Name | **MKRQST^BPSOSRX3** | | | |
| --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify |  |  |
| Routines Which Call This Subroutine | REQST^BPSOSRX | | | |
| Current Logic | | | | |
| BPSOSRX3 ;ALB/SS - ECME REQUESTS ;02-JAN-08  ;;1.0;E CLAIMS MGMT ENGINE;\*\*7,8,10,11,23\*\*;JUN 2004;Build 27  ;;Per VA Directive 6402, this routine should not be modified.  ;  . . .  MKRQST(BPREQTYP,KEY1,KEY2,MOREDATA,BPIENS78,BPCOBIND,BILLNDC,BPSKIP) ;  N BPIEN77,BPCOB,BPQ,BPIEN772,BPERRMSG,BPIEN59,BPIEN78,BPZ  N RETVAL,STAT,TYPE,RESULT,SUBMITDT,BPNOW,BPACTTYP,BP77LCK  N DUR,BPIEN771,BPCNT,BPSDUPL  ...  ; store secondary billing related data entered by the user - esg 6/8/10  S BPQ=0,BPERRMSG=""  I BPCOBIND=2 D  . N AMTIEN,BPIEN1,BPIEN2,BPIEN778,BPZ,BPZ1,BPZ2,OPAMT,OPAPQ,OPAYD,OPPRA,OPREJ,PIEN,REJIEN  . S PIEN=0 F S PIEN=$O(MOREDATA("OTHER PAYER",PIEN)) Q:'PIEN!BPQ D  .. S OPAYD=$G(MOREDATA("OTHER PAYER",PIEN,0)) Q:OPAYD=""  .. ;  .. ; count up the number of multiples we have in each set  .. S BPZ=0 F BPZ1=0:1 S BPZ=$O(MOREDATA("OTHER PAYER",PIEN,"P",BPZ)) Q:'BPZ  .. S BPZ=0 F BPZ2=0:1 S BPZ=$O(MOREDATA("OTHER PAYER",PIEN,"R",BPZ)) Q:'BPZ  .. I BPZ1,BPZ2 S BPQ=1,BPERRMSG="Can't have both payments and rejects for the same OTHER PAYER" Q  .. ;  .. ; add a new entry to subfile 9002313.778  .. S BPIEN778=$$INSITEM^BPSUTIL2(9002313.778,BPIEN77,PIEN,PIEN,"",,0)  .. I BPIEN778<1 S BPERRMSG="Can't create entry in COB OTHER PAYERS multiple of the BPS REQUESTS file",BPQ=1 Q  .. S BPERRMSG="Can't populate field in COB OTHER PAYERS multiple" ; just in case BPQ is set below  .. ;  .. ; set the rest of the pieces at this level  .. I $P(OPAYD,U,2)'="" I $$FILLFLDS^BPSUTIL2(9002313.778,.02,PIEN\_","\_BPIEN77,$P(OPAYD,U,2))<1 S BPQ=1 Q  .. I $P(OPAYD,U,3)'="" I $$FILLFLDS^BPSUTIL2(9002313.778,.03,PIEN\_","\_BPIEN77,$P(OPAYD,U,3))<1 S BPQ=1 Q  .. I $P(OPAYD,U,4)'="" I $$FILLFLDS^BPSUTIL2(9002313.778,.04,PIEN\_","\_BPIEN77,$P(OPAYD,U,4))<1 S BPQ=1 Q  .. I $P(OPAYD,U,5)'="" I $$FILLFLDS^BPSUTIL2(9002313.778,.05,PIEN\_","\_BPIEN77,$P(OPAYD,U,5))<1 S BPQ=1 Q  .. I $$FILLFLDS^BPSUTIL2(9002313.778,.06,PIEN\_","\_BPIEN77,BPZ1)<1 S BPQ=1 Q  .. I $$FILLFLDS^BPSUTIL2(9002313.778,.07,PIEN\_","\_BPIEN77,BPZ2)<1 S BPQ=1 Q  .. S BPERRMSG=""  .. ;  ... | | | | |
| Modified Logic (Changes are highlighted) | | | | |
| BPSOSRX3 ;ALB/SS - ECME REQUESTS ;02-JAN-08  ;;1.0;E CLAIMS MGMT ENGINE;\*\*7,8,10,11,23,24\*\*;JUN 2004;Build 27  ;;Per VA Directive 6402, this routine should not be modified.  ;  . . .  MKRQST(BPREQTYP,KEY1,KEY2,MOREDATA,BPIENS78,BPCOBIND,BILLNDC,BPSKIP) ;  N BPIEN77,BPCOB,BPQ,BPIEN772,BPERRMSG,BPIEN59,BPIEN78,BPZ  N RETVAL,STAT,TYPE,RESULT,SUBMITDT,BPNOW,BPACTTYP,BP77LCK  N DUR,BPIEN771,BPCNT,BPSDUPL  ...  ; store secondary billing related data entered by the user - esg 6/8/10  S BPQ=0,BPERRMSG=""  I BPCOBIND=2 D  . N AMTIEN,BPIEN1,BPIEN2,BPIEN778,BPZ,BPZ1,BPZ2,OPAMT,OPAPQ,OPAYD,OPPRA,OPREJ,PIEN,REJIEN  . S PIEN=0 F S PIEN=$O(MOREDATA("OTHER PAYER",PIEN)) Q:'PIEN!BPQ D  .. S OPAYD=$G(MOREDATA("OTHER PAYER",PIEN,0)) Q:OPAYD=""  .. ;  .. ; count up the number of multiples we have in each set  .. S BPZ=0 F BPZ1=0:1 S BPZ=$O(MOREDATA("OTHER PAYER",PIEN,"P",BPZ)) Q:'BPZ  .. S BPZ=0 F BPZ2=0:1 S BPZ=$O(MOREDATA("OTHER PAYER",PIEN,"R",BPZ)) Q:'BPZ  .. I BPZ1,BPZ2 S BPQ=1,BPERRMSG="Can't have both payments and rejects for the same OTHER PAYER" Q  .. ;  .. ; add a new entry to subfile 9002313.778  .. S BPIEN778=$$INSITEM^BPSUTIL2(9002313.778,BPIEN77,PIEN,PIEN,"",,0)  .. I BPIEN778<1 S BPERRMSG="Can't create entry in COB OTHER PAYERS multiple of the BPS REQUESTS file",BPQ=1 Q  .. S BPERRMSG="Can't populate field in COB OTHER PAYERS multiple" ; just in case BPQ is set below  .. ;  .. ; set the rest of the pieces at this level  .. I $P(OPAYD,U,2)'="" I $$FILLFLDS^BPSUTIL2(9002313.778,.02,PIEN\_","\_BPIEN77,$P(OPAYD,U,2))<1 S BPQ=1 Q  .. I $P(OPAYD,U,3)'="" I $$FILLFLDS^BPSUTIL2(9002313.778,.03,PIEN\_","\_BPIEN77,$P(OPAYD,U,3))<1 S BPQ=1 Q  .. I $P(OPAYD,U,4)'="" I $$FILLFLDS^BPSUTIL2(9002313.778,.04,PIEN\_","\_BPIEN77,$P(OPAYD,U,4))<1 S BPQ=1 Q  .. I $P(OPAYD,U,5)'="" I $$FILLFLDS^BPSUTIL2(9002313.778,.05,PIEN\_","\_BPIEN77,$P(OPAYD,U,5))<1 S BPQ=1 Q  .. I $$FILLFLDS^BPSUTIL2(9002313.778,.06,PIEN\_","\_BPIEN77,BPZ1)<1 S BPQ=1 Q  .. I $$FILLFLDS^BPSUTIL2(9002313.778,.07,PIEN\_","\_BPIEN77,BPZ2)<1 S BPQ=1 Q  .. I $P(OPAYD,U,11)'="" I $$FILLFLDS^BPSUTIL2(9002313.778,.11,PIEN\_","\_BPIEN77,$P(OPAYD,U,11))<1 S BPQ=1 Q  .. S BPERRMSG=""  .. ;  ... | | | | |

**BPS TRANSACTION**

Data is pulled from the BPS REQUESTS entry and put into the MOREDATA array in READMORE^BPSOSRX4. No change is necessary to the existing code, since the new field is on the same node already referenced in this line:

READMORE+50

… S MOREDATA("OTHER PAYER",COBPIEN,0)=$G(^BPS(9002313.77,BPIEN77,8,COBPIEN,0))

In SECBIL59^BPSPRRX6, data in the MOREDATA array is used to populate fields on the BPS TRANSACTION. The new field will need to be added to this code.

| Subroutine Name | **SECBIL59^BPSPRRX6** | | | |
| --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify |  |  |
| Routines Which Call This Subroutine | INIT^BPSOSIY | | | |
| Current Logic | | | | |
| BPSPRRX6 ;ALB/SS - ePharmacy secondary billing ;12-DEC-08  ;;1.0;E CLAIMS MGMT ENGINE;\*\*8,10,11,19,23\*\*;JUN 2004;Build 27  ;;Per VA Directive 6402, this routine should not be modified.  ;  . . .  SECBIL59(MOREDATA,IEN59) ;  ; Populate secondary billing fields in BPS TRANSACTION  ; MOREDATA array filed into 9002313.59  N BPTYPE,BPSTIME,BPCOB  N AMTIEN,BPIEN1,BPIEN2,BPZ5914,BPZ,BPZ1,BPZ2,OPAMT,OPAPQ,OPAYD,OPPRA,OPREJ,PIEN,REJIEN,BPQ  I +$G(IEN59)=0 Q  ;  I $L($G(MOREDATA("337-4C"))) I $$FILLFLDS^BPSUTIL2(9002313.59,1204,IEN59,MOREDATA("337-4C"))<1 D LOG^BPSOSL(IEN59,$T(+0)\_"-Cannot populate (#1204) of (#9002313.59)") ; cob other payments count  I $L($G(MOREDATA("308-C8"))) I $$FILLFLDS^BPSUTIL2(9002313.59,1205,IEN59,MOREDATA("308-C8"))<1 D LOG^BPSOSL(IEN59,$T(+0)\_"-Cannot populate (#1205) of (#9002313.59)") ; other coverage code  ;  ; store secondary billing related data entered by the user - esg 6/14/10  S BPQ=0  S PIEN=0 F S PIEN=$O(MOREDATA("OTHER PAYER",PIEN)) Q:'PIEN!BPQ D  . S OPAYD=$G(MOREDATA("OTHER PAYER",PIEN,0)) Q:OPAYD=""  . ;  . ; count up the number of multiples we have in each set  . S BPZ=0 F BPZ1=0:1 S BPZ=$O(MOREDATA("OTHER PAYER",PIEN,"P",BPZ)) Q:'BPZ  . S BPZ=0 F BPZ2=0:1 S BPZ=$O(MOREDATA("OTHER PAYER",PIEN,"R",BPZ)) Q:'BPZ  . I BPZ1,BPZ2 S BPQ=1 D LOG^BPSOSL(IEN59,$T(+0)\_"-Cannot have both payments and rejects for the same OTHER PAYER.") Q  . ;  . ; add a new entry to subfile 9002313.5914  . S BPZ5914=$$INSITEM^BPSUTIL2(9002313.5914,IEN59,PIEN,PIEN,"",,0)  . I BPZ5914<1 S BPQ=1 D LOG^BPSOSL(IEN59,$T(+0)\_"-Can't create entry in COB OTHER PAYERS multiple of the BPS TRANSACTION file") Q  . ;  . ; set the rest of the pieces at this level  . I $P(OPAYD,U,2)'="" I $$FILLFLDS^BPSUTIL2(9002313.5914,.02,PIEN\_","\_IEN59,$P(OPAYD,U,2))<1 S BPQ=1 D LOG^BPSOSL(IEN59,$T(+0)\_"-Cannot populate (#.02) of (#9002313.5914)") Q  . I $P(OPAYD,U,3)'="" I $$FILLFLDS^BPSUTIL2(9002313.5914,.03,PIEN\_","\_IEN59,$P(OPAYD,U,3))<1 S BPQ=1 D LOG^BPSOSL(IEN59,$T(+0)\_"-Cannot populate (#.03) of (#9002313.5914)") Q  . I $P(OPAYD,U,4)'="" I $$FILLFLDS^BPSUTIL2(9002313.5914,.04,PIEN\_","\_IEN59,$P(OPAYD,U,4))<1 S BPQ=1 D LOG^BPSOSL(IEN59,$T(+0)\_"-Cannot populate (#.04) of (#9002313.5914)") Q  . I $P(OPAYD,U,5)'="" I $$FILLFLDS^BPSUTIL2(9002313.5914,.05,PIEN\_","\_IEN59,$P(OPAYD,U,5))<1 S BPQ=1 D LOG^BPSOSL(IEN59,$T(+0)\_"-Cannot populate (#.05) of (#9002313.5914)") Q  . I $$FILLFLDS^BPSUTIL2(9002313.5914,.06,PIEN\_","\_IEN59,BPZ1)<1 S BPQ=1 D LOG^BPSOSL(IEN59,$T(+0)\_"-Cannot populate (#.06) of (#9002313.5914)") Q  . I $$FILLFLDS^BPSUTIL2(9002313.5914,.07,PIEN\_","\_IEN59,BPZ2)<1 S BPQ=1 D LOG^BPSOSL(IEN59,$T(+0)\_"-Cannot populate (#.07) of (#9002313.5914)") Q  . ;  . ; now loop thru the other payer payment array  . S AMTIEN=0 F S AMTIEN=$O(MOREDATA("OTHER PAYER",PIEN,"P",AMTIEN)) Q:'AMTIEN!BPQ D  .. S OPAMT=$G(MOREDATA("OTHER PAYER",PIEN,"P",AMTIEN,0))  .. S OPAPQ=$P(OPAMT,U,2) ; 342-HC other payer amt paid qualifier (ncpdp 5.1 blank is OK)  .. S OPPRA=$P(OPAMT,U,3) ; 352-NQ, Other Payer-Patient Responsibility Amount  .. S OPAMT=+OPAMT ; 431-DV other payer amt paid  .. ;  .. ; add a new entry to subfile 9002313.59141  .. S BPIEN1=$$INSITEM^BPSUTIL2(9002313.59141,PIEN\_","\_IEN59,OPAMT,AMTIEN,"",,0)  .. I BPIEN1<1 S BPQ=1 D LOG^BPSOSL(IEN59,$T(+0)\_"-Can't create entry in 9002313.59141 subfile") Q  .. ;  .. ; set piece 2  .. I OPAPQ'="" I $$FILLFLDS^BPSUTIL2(9002313.59141,.02,AMTIEN\_","\_PIEN\_","\_IEN59,OPAPQ)<1 D  ... S BPQ=1 D LOG^BPSOSL(IEN59,$T(+0)\_"-Cannot populate (#.02) of (#9002313.59141)")  ... Q  .. ;  .. ; set piece 3  .. I OPPRA'="" I $$FILLFLDS^BPSUTIL2(9002313.59141,.03,AMTIEN\_","\_PIEN\_","\_IEN59,OPPRA)<1 D  ... S BPQ=1 D LOG^BPSOSL(IEN59,$T(+0)\_"-Cannot populate (#.03) of (#9002313.59141)")  ... Q  .. ;  .. Q  . ;  . ; now loop thru the other payer reject array  . S REJIEN=0 F S REJIEN=$O(MOREDATA("OTHER PAYER",PIEN,"R",REJIEN)) Q:'REJIEN!BPQ D  .. S OPREJ=$G(MOREDATA("OTHER PAYER",PIEN,"R",REJIEN,0)) Q:OPREJ="" Q:$P(OPREJ,U,1)=""  .. ;  .. ; add a new entry to subfile 9002313.59142  .. S BPIEN2=$$INSITEM^BPSUTIL2(9002313.59142,PIEN\_","\_IEN59,$P(OPREJ,U,1),REJIEN,"",,0)  .. I BPIEN2<1 S BPQ=1 D LOG^BPSOSL(IEN59,$T(+0)\_"-Can't create entry in 9002313.59142 subfile") Q  .. Q  . Q  Q | | | | |
| Modified Logic (Changes are highlighted) | | | | |
| BPSPRRX6 ;ALB/SS - ePharmacy secondary billing ;12-DEC-08  ;;1.0;E CLAIMS MGMT ENGINE;\*\*8,10,11,19,23,24\*\*;JUN 2004;Build 27  ;;Per VA Directive 6402, this routine should not be modified.  ;  . . .  SECBIL59(MOREDATA,IEN59) ;  ; Populate secondary billing fields in BPS TRANSACTION  ; MOREDATA array filed into 9002313.59  N BPTYPE,BPSTIME,BPCOB  N AMTIEN,BPIEN1,BPIEN2,BPZ5914,BPZ,BPZ1,BPZ2,OPAMT,OPAPQ,OPAYD,OPPRA,OPREJ,PIEN,REJIEN,BPQ  I +$G(IEN59)=0 Q  ;  I $L($G(MOREDATA("337-4C"))) I $$FILLFLDS^BPSUTIL2(9002313.59,1204,IEN59,MOREDATA("337-4C"))<1 D LOG^BPSOSL(IEN59,$T(+0)\_"-Cannot populate (#1204) of (#9002313.59)") ; cob other payments count  I $L($G(MOREDATA("308-C8"))) I $$FILLFLDS^BPSUTIL2(9002313.59,1205,IEN59,MOREDATA("308-C8"))<1 D LOG^BPSOSL(IEN59,$T(+0)\_"-Cannot populate (#1205) of (#9002313.59)") ; other coverage code  ;  ; store secondary billing related data entered by the user - esg 6/14/10  S BPQ=0  S PIEN=0 F S PIEN=$O(MOREDATA("OTHER PAYER",PIEN)) Q:'PIEN!BPQ D  . S OPAYD=$G(MOREDATA("OTHER PAYER",PIEN,0)) Q:OPAYD=""  . ;  . ; count up the number of multiples we have in each set  . S BPZ=0 F BPZ1=0:1 S BPZ=$O(MOREDATA("OTHER PAYER",PIEN,"P",BPZ)) Q:'BPZ  . S BPZ=0 F BPZ2=0:1 S BPZ=$O(MOREDATA("OTHER PAYER",PIEN,"R",BPZ)) Q:'BPZ  . I BPZ1,BPZ2 S BPQ=1 D LOG^BPSOSL(IEN59,$T(+0)\_"-Cannot have both payments and rejects for the same OTHER PAYER.") Q  . ;  . ; add a new entry to subfile 9002313.5914  . S BPZ5914=$$INSITEM^BPSUTIL2(9002313.5914,IEN59,PIEN,PIEN,"",,0)  . I BPZ5914<1 S BPQ=1 D LOG^BPSOSL(IEN59,$T(+0)\_"-Can't create entry in COB OTHER PAYERS multiple of the BPS TRANSACTION file") Q  . ;  . ; set the rest of the pieces at this level  . I $P(OPAYD,U,2)'="" I $$FILLFLDS^BPSUTIL2(9002313.5914,.02,PIEN\_","\_IEN59,$P(OPAYD,U,2))<1 S BPQ=1 D LOG^BPSOSL(IEN59,$T(+0)\_"-Cannot populate (#.02) of (#9002313.5914)") Q  . I $P(OPAYD,U,3)'="" I $$FILLFLDS^BPSUTIL2(9002313.5914,.03,PIEN\_","\_IEN59,$P(OPAYD,U,3))<1 S BPQ=1 D LOG^BPSOSL(IEN59,$T(+0)\_"-Cannot populate (#.03) of (#9002313.5914)") Q  . I $P(OPAYD,U,4)'="" I $$FILLFLDS^BPSUTIL2(9002313.5914,.04,PIEN\_","\_IEN59,$P(OPAYD,U,4))<1 S BPQ=1 D LOG^BPSOSL(IEN59,$T(+0)\_"-Cannot populate (#.04) of (#9002313.5914)") Q  . I $P(OPAYD,U,5)'="" I $$FILLFLDS^BPSUTIL2(9002313.5914,.05,PIEN\_","\_IEN59,$P(OPAYD,U,5))<1 S BPQ=1 D LOG^BPSOSL(IEN59,$T(+0)\_"-Cannot populate (#.05) of (#9002313.5914)") Q  . I $$FILLFLDS^BPSUTIL2(9002313.5914,.06,PIEN\_","\_IEN59,BPZ1)<1 S BPQ=1 D LOG^BPSOSL(IEN59,$T(+0)\_"-Cannot populate (#.06) of (#9002313.5914)") Q  . I $$FILLFLDS^BPSUTIL2(9002313.5914,.07,PIEN\_","\_IEN59,BPZ2)<1 S BPQ=1 D LOG^BPSOSL(IEN59,$T(+0)\_"-Cannot populate (#.07) of (#9002313.5914)") Q  . I $P(OPAYD,U,11)'="" I $$FILLFLDS^BPSUTIL2(9002313.5914,.11,PIEN\_","\_IEN59,$P(OPAYD,U,11))<1 S BPQ=1 D LOG^BPSOSL(IEN59,$T(+0)\_"-Cannot populate (#.11) of (#9002313.5914)") Q  . ;  . ; now loop thru the other payer payment array  . S AMTIEN=0 F S AMTIEN=$O(MOREDATA("OTHER PAYER",PIEN,"P",AMTIEN)) Q:'AMTIEN!BPQ D  .. S OPAMT=$G(MOREDATA("OTHER PAYER",PIEN,"P",AMTIEN,0))  .. S OPAPQ=$P(OPAMT,U,2) ; 342-HC other payer amt paid qualifier (ncpdp 5.1 blank is OK)  .. S OPPRA=$P(OPAMT,U,3) ; 352-NQ, Other Payer-Patient Responsibility Amount  .. S OPAMT=+OPAMT ; 431-DV other payer amt paid  .. ;  .. ; add a new entry to subfile 9002313.59141  .. S BPIEN1=$$INSITEM^BPSUTIL2(9002313.59141,PIEN\_","\_IEN59,OPAMT,AMTIEN,"",,0)  .. I BPIEN1<1 S BPQ=1 D LOG^BPSOSL(IEN59,$T(+0)\_"-Can't create entry in 9002313.59141 subfile") Q  .. ;  .. ; set piece 2  .. I OPAPQ'="" I $$FILLFLDS^BPSUTIL2(9002313.59141,.02,AMTIEN\_","\_PIEN\_","\_IEN59,OPAPQ)<1 D  ... S BPQ=1 D LOG^BPSOSL(IEN59,$T(+0)\_"-Cannot populate (#.02) of (#9002313.59141)")  ... Q  .. ;  .. ; set piece 3  .. I OPPRA'="" I $$FILLFLDS^BPSUTIL2(9002313.59141,.03,AMTIEN\_","\_PIEN\_","\_IEN59,OPPRA)<1 D  ... S BPQ=1 D LOG^BPSOSL(IEN59,$T(+0)\_"-Cannot populate (#.03) of (#9002313.59141)")  ... Q  .. ;  .. Q  . ;  . ; now loop thru the other payer reject array  . S REJIEN=0 F S REJIEN=$O(MOREDATA("OTHER PAYER",PIEN,"R",REJIEN)) Q:'REJIEN!BPQ D  .. S OPREJ=$G(MOREDATA("OTHER PAYER",PIEN,"R",REJIEN,0)) Q:OPREJ="" Q:$P(OPREJ,U,1)=""  .. ;  .. ; add a new entry to subfile 9002313.59142  .. S BPIEN2=$$INSITEM^BPSUTIL2(9002313.59142,PIEN\_","\_IEN59,$P(OPREJ,U,1),REJIEN,"",,0)  .. I BPIEN2<1 S BPQ=1 D LOG^BPSOSL(IEN59,$T(+0)\_"-Can't create entry in 9002313.59142 subfile") Q  .. Q  . Q  Q | | | | |

**BPS CLAIMS**

COB^BPSOSCD populates portions of the BPS array with data from the BPS TRANSACTION. After the BPS array is built, it is used to create an entry in BPS CLAIMS. No change is necessary to the existing code, since the new field is on the same node already referenced in the line below, however a comment will be modified in COB^BPSOSCD.

COB+14^BPSOSCD

. S BPS("RX",MEDN,"OTHER PAYER",COBPIEN,0)=$G(^BPST(IEN59,14,COBPIEN,0))

| Subroutine Name | **COB^BPSOSCD** | | | |
| --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify |  |  |
| Routines Which Call This Subroutine | MEDINFO^BPSOSCD | | | |
| Current Logic | | | | |
| COB(IEN59,MEDN) ; process the COB fields and build the COB array  ; Code for Benefit Stages multiple not implemented yet (except by  ; certification)  ;  ; build array of COB secondary claim data from the BPS Transaction file - esg - 6/16/10  N COBPIEN,APDIEN,REJIEN,DATA  K BPS("RX",MEDN,"OTHER PAYER")  ;  ; Field 337-4C COB OTHER PAYMENTS COUNT (9002313.59,1204) moved into [1] below  S BPS("RX",MEDN,"OTHER PAYER",0)=$P($G(^BPST(IEN59,12)),U,4)  ;  S COBPIEN=0 F S COBPIEN=$O(^BPST(IEN59,14,COBPIEN)) Q:'COBPIEN D  . ; Note that this will set pieces 1-7. Piece 8 is reserved for  . ; Payer-Patient Responsibility Count and is set by the certification code  . S BPS("RX",MEDN,"OTHER PAYER",COBPIEN,0)=$G(^BPST(IEN59,14,COBPIEN,0))  . ;  . ; retrieve data from other payer amount paid multiple  . S APDIEN=0 F S APDIEN=$O(^BPST(IEN59,14,COBPIEN,1,APDIEN)) Q:'APDIEN D  .. S DATA=$G(^BPST(IEN59,14,COBPIEN,1,APDIEN,0))  .. S BPS("RX",MEDN,"OTHER PAYER",COBPIEN,"P",APDIEN,0)=  $P(DATA,"^",1)\_"^"\_$$GET1^DIQ(9002313.2,$P(DATA,"^",2),.01)  .. S BPS("RX",MEDN,"OTHER PAYER",COBPIEN,"PP",APDIEN,0)=$P(DATA,"^",3)  .. I +$P(DATA,"^",3) S $P(BPS("RX",MEDN,"OTHER PAYER",COBPIEN,"PP",APDIEN,0) ,"^",2)="06"  .. Q  . ;  . ; retrieve data from other payer reject multiple  . S REJIEN=0 F S REJIEN=$O(^BPST(IEN59,14,COBPIEN,2,REJIEN)) Q:'REJIEN D  .. S BPS("RX",MEDN,"OTHER PAYER",COBPIEN,"R",REJIEN,0)=$G(^BPST(IEN59,14,COBPIEN,2,REJIEN,0))  Q | | | | |
| Modified Logic (Changes are highlighted) | | | | |
| COB(IEN59,MEDN) ; process the COB fields and build the COB array  ; Code for Benefit Stages multiple not implemented yet (except by  ; certification)  ;  ; build array of COB secondary claim data from the BPS Transaction file - esg - 6/16/10  N COBPIEN,APDIEN,REJIEN,DATA  K BPS("RX",MEDN,"OTHER PAYER")  ;  ; Field 337-4C COB OTHER PAYMENTS COUNT (9002313.59,1204) moved into [1] below  S BPS("RX",MEDN,"OTHER PAYER",0)=$P($G(^BPST(IEN59,12)),U,4)  ;  S COBPIEN=0 F S COBPIEN=$O(^BPST(IEN59,14,COBPIEN)) Q:'COBPIEN D  . ; Note that this will set pieces 1-7 and 11. Piece ~~8~~9 is reserved for  . ; ~~Payer-Patient Responsibility Count~~ Benefit Stage Count and is set by the certification code  . S BPS("RX",MEDN,"OTHER PAYER",COBPIEN,0)=$G(^BPST(IEN59,14,COBPIEN,0))  . ;  . ; retrieve data from other payer amount paid multiple  . S APDIEN=0 F S APDIEN=$O(^BPST(IEN59,14,COBPIEN,1,APDIEN)) Q:'APDIEN D  .. S DATA=$G(^BPST(IEN59,14,COBPIEN,1,APDIEN,0))  .. S BPS("RX",MEDN,"OTHER PAYER",COBPIEN,"P",APDIEN,0)=  $P(DATA,"^",1)\_"^"\_$$GET1^DIQ(9002313.2,$P(DATA,"^",2),.01)  .. S BPS("RX",MEDN,"OTHER PAYER",COBPIEN,"PP",APDIEN,0)=$P(DATA,"^",3)  .. I +$P(DATA,"^",3) S $P(BPS("RX",MEDN,"OTHER PAYER",COBPIEN,"PP",APDIEN,0) ,"^",2)="06"  .. Q  . ;  . ; retrieve data from other payer reject multiple  . S REJIEN=0 F S REJIEN=$O(^BPST(IEN59,14,COBPIEN,2,REJIEN)) Q:'REJIEN D  .. S BPS("RX",MEDN,"OTHER PAYER",COBPIEN,"R",REJIEN,0)=$G(^BPST(IEN59,14,COBPIEN,2,REJIEN,0))  Q | | | | |

The Set code for each field in the file BPS NCPDP FIELD DEFS sets the value of that field into the corresponding field in the BPS CLAIMS file. SET2149^BPSFLD01 will be created to serve as the Set code for field C49-9V.

| Subroutine Name | **SET2149^BPSFLD01** | | | |
| --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify |  |  |
| Current Logic | | | | |
| n/a – new subroutine | | | | |
| Modified Logic (Changes are highlighted) | | | | |
| BPSFLD01 ;ALB/SS - ePharmacy secondary billing - COB fields processing ;27-FEB-09  ;;1.0;E CLAIMS MGMT ENGINE;\*\*8,10,23,24\*\*;JUN 2004;Build 27  ;;Per VA Directive 6402, this routine should not be modified.  ;  . . .  SET2149 ; C49-9V Other Payer-Patient Reconciliation ID  I '$G(BPSOPIEN) Q  S $P(^BPSC(BPS(9002313.02),400,BPS(9002313.0201),337,BPSOPIEN,0),U,11)=BPS("X")  Q | | | | |

The many COB fields in BPS NCPDP FIELD DEFS do not each have their own Get code. Rather, COB^BPSOSHF pulls all the fields and executes the format and set code for each. This procedure will be modified to make sure the Other Payer Reconciliation ID field will be populated if there is a value in Reconciliation ID from the primary claim.

| Subroutine Name | **COB^BPSOSHF** | | | |
| --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify |  |  |
| Routines Which Call This Subroutine | XLOOP^BPSOSCF | | | |
| Current Logic | | | | |
| BPSOSHF ;BHAM ISC/SD/lwj/DLF - Get/Format/Set value for repeating segments ;06/01/2004  ;;1.0;E CLAIMS MGMT ENGINE;\*\*1,5,8,10,11,23\*\*;JUN 2004;Build 27  ;;Per VA Directive 6402, this routine should not be modified.  ;  . . .  COB(FORMAT,NODE,MEDN) ; COB fields processing, NODE=160  ;---------------------------------------------------------------  ; The COB data is stored in the following local array:  ;  ; BPS("RX",MEDN,"OTHER PAYER",.....  ;  ; Array built in routine BPSOSCD.  ; Special note - Overrides are not allowed on this multiple.  ; "Special" code is not accounted for either.  ;---------------------------------------------------------------  ;  N FIELD,FLD,OVERRIDE,FLAG,ORD,NCPFLD,BPD,BPD1,BPD2,PCE,BPSOPIEN,BPSOAIEN,BPSORIEN,BPSCOUNT  S FLAG="FS"  ;  ; Quit if there is no data in the array  Q:'$D(BPS("RX",MEDN,"OTHER PAYER"))  ;  ; next we need to figure out which fields on this format are really  ; needed, then we will loop through and populate them  ;  D GETFLDS(FORMAT,NODE,.FIELD)  ;  ; re-sort this list by the NCPDP field#  ; NCPFLD(NCPDP FIELD#) = internal field#  K NCPFLD S ORD=0 F S ORD=$O(FIELD(ORD)) Q:'ORD S FLD=$P(FIELD(ORD),U,2) I FLD'="" S NCPFLD(FLD)=+FIELD(ORD)  ;  ; see if 337-4C is needed  S FLD=337  I $D(NCPFLD(FLD)) D  . S BPS("X")=$P($G(BPS("RX",MEDN,"OTHER PAYER",0)),U,1) ; get  . I BPS("X")="" Q  . D XFLDCODE^BPSOSCF(NODE,NCPFLD(FLD),FLAG) ; format/set  . Q  ;  ; now lets get, format and set the rest of the COB fields  S BPSOPIEN=0 F S BPSOPIEN=$O(BPS("RX",MEDN,"OTHER PAYER",BPSOPIEN)) Q:'BPSOPIEN D  . S BPD=$G(BPS("RX",MEDN,"OTHER PAYER",BPSOPIEN,0))  . ; Note that piece 9 (Benefit Stage Count) is only set by Certification Code  . F PCE=1:1:7,9 D  .. S FLD=$S(PCE=1:337,PCE=2:338,PCE=3:339,PCE=4:340,PCE=5:443,PCE=6:341,PCE=7:471,PCE=9:392,1:0) Q:'FLD  .. I '$D(NCPFLD(FLD)) Q ; field not needed  .. I $P(BPD,U,PCE)="" Q ; data is nil  .. S BPS("X")=$P(BPD,U,PCE) ; get  .. D XFLDCODE^BPSOSCF(NODE,NCPFLD(FLD),FLAG) ; format/set  .. Q  . ;  . ; Now look at the other payer amount paid fields  . S BPSOAIEN=0 F S BPSOAIEN=$O(BPS("RX",MEDN,"OTHER PAYER",BPSOPIEN,"P",BPSOAIEN)) Q:'BPSOAIEN D  .. S BPD1=$G(BPS("RX",MEDN,"OTHER PAYER",BPSOPIEN,"P",BPSOAIEN,0))  .. F PCE=1,2 D  ... S FLD=$S(PCE=1:431,PCE=2:342,1:0) Q:'FLD  ... I '$D(NCPFLD(FLD)) Q ; field not needed  ... I $P(BPD1,U,PCE)="" Q ; data is nil  ... S BPS("X")=$P(BPD1,U,PCE) ; get  ... D XFLDCODE^BPSOSCF(NODE,NCPFLD(FLD),FLAG) ; format/set  .. Q  . ;  . ; Now look at the other payer reject code fields  . S BPSORIEN=0 F S BPSORIEN=$O(BPS("RX",MEDN,"OTHER PAYER",BPSOPIEN,"R",BPSORIEN)) Q:'BPSORIEN D  .. S BPD2=$G(BPS("RX",MEDN,"OTHER PAYER",BPSOPIEN,"R",BPSORIEN,0))  .. S FLD=472  .. I '$D(NCPFLD(FLD)) Q ; field not needed  .. I BPD2="" Q ; data is nil  .. S BPS("X")=BPD2 ; get  .. D XFLDCODE^BPSOSCF(NODE,NCPFLD(FLD),FLAG) ; format/set  .. Q  . . .  COBX ;  Q | | | | |
| Modified Logic (Changes are highlighted) | | | | |
| BPSOSHF ;BHAM ISC/SD/lwj/DLF - Get/Format/Set value for repeating segments ;06/01/2004  ;;1.0;E CLAIMS MGMT ENGINE;\*\*1,5,8,10,11,23,24\*\*;JUN 2004;Build 27  ;;Per VA Directive 6402, this routine should not be modified.  ;  . . .  COB(FORMAT,NODE,MEDN) ; COB fields processing, NODE=160  ;---------------------------------------------------------------  ; The COB data is stored in the following local array:  ;  ; BPS("RX",MEDN,"OTHER PAYER",.....  ;  ; Array built in routine BPSOSCD.  ; Special note - Overrides are not allowed on this multiple.  ; "Special" code is not accounted for either.  ;---------------------------------------------------------------  ;  N FIELD,FLD,OVERRIDE,FLAG,ORD,NCPFLD,BPD,BPD1,BPD2,PCE,BPSOPIEN,BPSOAIEN,BPSORIEN,BPSCOUNT  S FLAG="FS"  ;  ; Quit if there is no data in the array  Q:'$D(BPS("RX",MEDN,"OTHER PAYER"))  ;  ; next we need to figure out which fields on this format are really  ; needed, then we will loop through and populate them  ;  D GETFLDS(FORMAT,NODE,.FIELD)  ;  ; re-sort this list by the NCPDP field#  ; NCPFLD(NCPDP FIELD#) = internal field#  K NCPFLD S ORD=0 F S ORD=$O(FIELD(ORD)) Q:'ORD S FLD=$P(FIELD(ORD),U,2) I FLD'="" S NCPFLD(FLD)=+FIELD(ORD)  ;  ; see if 337-4C is needed  S FLD=337  I $D(NCPFLD(FLD)) D  . S BPS("X")=$P($G(BPS("RX",MEDN,"OTHER PAYER",0)),U,1) ; get  . I BPS("X")="" Q  . D XFLDCODE^BPSOSCF(NODE,NCPFLD(FLD),FLAG) ; format/set  . Q  ;  ; now lets get, format and set the rest of the COB fields  S BPSOPIEN=0 F S BPSOPIEN=$O(BPS("RX",MEDN,"OTHER PAYER",BPSOPIEN)) Q:'BPSOPIEN D  . S BPD=$G(BPS("RX",MEDN,"OTHER PAYER",BPSOPIEN,0))  . ; Note that piece 9 (Benefit Stage Count) is only set by Certification Code  . F PCE=1:1:7,9,11 D  .. S FLD=$S(PCE=1:337,PCE=2:338,PCE=3:339,PCE=4:340,PCE=5:443,PCE=6:341,PCE=7:471,PCE=9:392, PCE=11:2149,1:0) Q:'FLD  .. I '$D(NCPFLD(FLD)) Q ; field not needed  .. I $P(BPD,U,PCE)="" Q ; data is nil  .. S BPS("X")=$P(BPD,U,PCE) ; get  .. D XFLDCODE^BPSOSCF(NODE,NCPFLD(FLD),FLAG) ; format/set  .. Q  . . .  COBX ;  Q | | | | |

**The Outgoing Claim Request**

After the entry in BPS CLAIMS is created, the system uses the data in that entry to build the claim to be sent to the payer. Before looping through the segments and fields on the payer sheet, the system copies most of the fields from the BPS CLAIMS record into the BPS array. That is done in GETBPS5^BPSECX0 and GETBPS7^BPSECX0. GETBPS5^BPSEXCX0 will need to be modified to pick up the new field.

A small error in GETBPS5^BPSEXC0 will also be corrected with this build. The field number 993 was incorrectly listed as 393.

| Subroutine Name | **GETBPS5^BPSECX0** | | | |
| --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify |  |  |
| Routines Which Call This Subroutine | XLOOP^BPSOSCF | | | |
| Current Logic | | | | |
| ;Retrieve COB OTHER PAYMENTS multiple data  ; CLAIMIEN = ien in BPS CLAIMS (#9002313.02)  ; TRXIEN = ien in TRANSACTIONS (#9002313.0201)  ; BPCOBIEN= ien in COB OTHER PAYMENTS (#9002313.0401)  ; BPS - Passed by reference  ; Output: BPS(9002313.0401,BPCOBIEN,field #,"I") = Value  GETBPS5(CLAIMIEN,TRXIEN,BPCOBIEN,BPS) ;EP - from BPSECA1  ;  Q:$G(CLAIMIEN)="" Q:$G(TRXIEN)="" Q:$G(BPCOBIEN)=""  ;  N BPREJCT,BPSCNT,BPSPAMT,BPSOTHR,D0,DA,DIC,DIQ,DIQ2,DR  ;  S DIC=9002313.02  S DA=CLAIMIEN  S DA(9002313.0201)=TRXIEN  S DA(9002313.0401)=BPCOBIEN  S DR="400" ; field (#400) TRANSACTIONS  S DR(9002313.0201)=337.01 ;field (#337.01) COB OTHER PAYMENTS  S DR(9002313.0401)=".01;338;339;340;341;443;471;353;392;393" ; fields  S DIQ="BPS",DIQ(0)="I"  D EN^DIQ1  ;  ; Loop through PAYER AMT and get the data  S BPSPAMT=$P($G(^BPSC(CLAIMIEN,400,TRXIEN,337,BPCOBIEN,1,0)),U,4)  F BPSCNT=1:1:BPSPAMT D GETBPS6(CLAIMIEN,TRXIEN,BPCOBIEN,BPSCNT,.BPS)  ;  ; Loop through OTHER PAYER REJECT CODE multiple and get the data  S BPREJCT=$P($G(^BPSC(CLAIMIEN,400,TRXIEN,337,BPCOBIEN,2,0)),U,4)  F BPSCNT=1:1:BPREJCT D GETBPS7(CLAIMIEN,TRXIEN,BPCOBIEN,BPSCNT,.BPS)  ;  ; Loop through PAYER-PATIENT RESP and get the data  S BPSPAMT=$P($G(^BPSC(CLAIMIEN,400,TRXIEN,337,BPCOBIEN,3,0)),U,4)  F BPSCNT=1:1:BPSPAMT D GETBPS8(CLAIMIEN,TRXIEN,BPCOBIEN,BPSCNT,.BPS)  ;  ; Loop through BENEFIT STAGES and get the data  S BPSPAMT=$P($G(^BPSC(CLAIMIEN,400,TRXIEN,337,BPCOBIEN,4,0)),U,4)  F BPSCNT=1:1:BPSPAMT D GETBPS9(CLAIMIEN,TRXIEN,BPCOBIEN,BPSCNT,.BPS)  Q | | | | |
| Modified Logic (Changes are highlighted) | | | | |
| ;Retrieve COB OTHER PAYMENTS multiple data  ; CLAIMIEN = ien in BPS CLAIMS (#9002313.02)  ; TRXIEN = ien in TRANSACTIONS (#9002313.0201)  ; BPCOBIEN= ien in COB OTHER PAYMENTS (#9002313.0401)  ; BPS - Passed by reference  ; Output: BPS(9002313.0401,BPCOBIEN,field #,"I") = Value  GETBPS5(CLAIMIEN,TRXIEN,BPCOBIEN,BPS) ;EP - from BPSECA1  ;  Q:$G(CLAIMIEN)="" Q:$G(TRXIEN)="" Q:$G(BPCOBIEN)=""  ;  N BPREJCT,BPSCNT,BPSPAMT,BPSOTHR,D0,DA,DIC,DIQ,DIQ2,DR  ;  S DIC=9002313.02  S DA=CLAIMIEN  S DA(9002313.0201)=TRXIEN  S DA(9002313.0401)=BPCOBIEN  S DR="400" ; field (#400) TRANSACTIONS  S DR(9002313.0201)=337.01 ;field (#337.01) COB OTHER PAYMENTS  S DR(9002313.0401)=".01;338;339;340;341;443;471;353;392;~~393;~~993;2149" ; fields  S DIQ="BPS",DIQ(0)="I"  D EN^DIQ1  ;  ; Loop through PAYER AMT and get the data  S BPSPAMT=$P($G(^BPSC(CLAIMIEN,400,TRXIEN,337,BPCOBIEN,1,0)),U,4)  F BPSCNT=1:1:BPSPAMT D GETBPS6(CLAIMIEN,TRXIEN,BPCOBIEN,BPSCNT,.BPS)  ;  ; Loop through OTHER PAYER REJECT CODE multiple and get the data  S BPREJCT=$P($G(^BPSC(CLAIMIEN,400,TRXIEN,337,BPCOBIEN,2,0)),U,4)  F BPSCNT=1:1:BPREJCT D GETBPS7(CLAIMIEN,TRXIEN,BPCOBIEN,BPSCNT,.BPS)  ;  ; Loop through PAYER-PATIENT RESP and get the data  S BPSPAMT=$P($G(^BPSC(CLAIMIEN,400,TRXIEN,337,BPCOBIEN,3,0)),U,4)  F BPSCNT=1:1:BPSPAMT D GETBPS8(CLAIMIEN,TRXIEN,BPCOBIEN,BPSCNT,.BPS)  ;  ; Loop through BENEFIT STAGES and get the data  S BPSPAMT=$P($G(^BPSC(CLAIMIEN,400,TRXIEN,337,BPCOBIEN,4,0)),U,4)  F BPSCNT=1:1:BPSPAMT D GETBPS9(CLAIMIEN,TRXIEN,BPCOBIEN,BPSCNT,.BPS)  Q | | | | |

After the values in BPS CLAIMS are copied into the BPS array, the system uses the data in that array to build the claim to be sent to the payer. This is largely done in XLOOP^BPSOSH2. This subroutine loops through each field on each segment on the payer sheet, and for each field it pulls the data from BPS CLAIMS and adds it to the segment. XLOOP^BPSOSH2 calls PROCCOB^BPSOSH2 to pull the COB fields, and both of those procedures are already pulling the Patient Responsibility fields.

**CRI/Claim Response Inquiry Screen**

The CRI/Claim Response Inquiry will automatically display the Other Payer Patient Responsibility fields from the claim request whenever they are populated and the Patient Pay Amount on the claim response whenever it is populated. No change to existing logic is necessary. This is also true of the VER, which contains the CRI.

**PRO Option and RED Action**

Both the PRO option and the RED action display to the user a list of COB fields if the claim is secondary. The Other Payer Reconciliation ID field will be added to this display.

Both PRO and RED rely on PRIMDATA^BPSPRRX6 and GETOPAP^BPSPRRX6 to pull the COB fields from the BPS RESPONSE of the primary claim. No change to PRIMDATA^BPSPRRX6 is necessary, but GETOPAP^BPSPRRX6 will be updated to include the Reconciliation ID field.

If the system is not able to pull COB fields from the primary claim, then both PRO and RED rely on SECDATA^BPSPRRX6 to pull the COB fields from the BPS TRANSACTION of the secondary claim if the system is attempting to resubmit a secondary claim. No change to SECDATA^BPSPRRX6 is necessary. As it is currently written, SECDATA^BPSPRRX6 will automatically pull the new field OTHER PAYER RECONCILIATION ID from the BPS TRANSACTION file after that field has been added to the file.

SECDATA+44^BPSPRRX6

. S BPSPRDAT("OTHER PAYER",COBPIEN,0)=$G(^BPST(IEN59SEC,14,COBPIEN,0))

| Subroutine Name | **GETOPAP^BPSPRRX6** | | | |
| --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify |  |  |
| Related Menu Options or ListMan Actions | PRO Option  RED Action | | | |
| Routines Which Call This Subroutine | PRIMDATA^BPSPRRX6 | | | |
| Current Logic | | | | |
| GETOPAP(BPSRESP,BPSDAT) ;  ; Get the Other Payer Amount Paid values and qualifiers  ; Input:  ; BPSRESP = IEN of BPS RESPONSE file  ; BPSDAT(N)= Array of Other Payer fields (passed by reference)  ; [1] Paid Amount  ; [2] Qualifier  ; [3] Other Payer Patient Responsibility Amount  ;  I '$G(BPSRESP) Q  I '$D(^BPSR(BPSRESP,1000)) Q  N CNT,BPS505,BPS509,BPS559,BPS558,BPS523,BPS563,BPS562,BPS521,BPSQUAL,BPSAMNT,BPSTAX,BPSOAP,BPSX  S CNT=0  ; Set up D.0 fields for COB segment  S BPS509=$$DFF2EXT^BPSECFM($P($G(^BPSR(BPSRESP,1000,1,500)),U,9))  ; If Total Amount Paid is a negative number, set it to zero.  ; Zero Pay amount is allowed  I BPS509<0 S BPS509=0  ;  ; Cognitive Services Qualifier/Professional Service Fee Paid  S BPS562=$$DFF2EXT^BPSECFM($P($G(^BPSR(BPSRESP,1000,1,560)),U,2))  I BPS562<0 S BPS562=0  I +BPS562 S CNT=CNT+1,BPSDAT(CNT)=BPS562\_U\_"06"  ;  ; Incentive Qualifier/Incentive Amt Paid  S BPS521=$$DFF2EXT^BPSECFM($P($G(^BPSR(BPSRESP,1000,1,500)),U,21))  I BPS521<0 S BPS521=0  I +BPS521 S CNT=CNT+1,BPSDAT(CNT)=BPS521\_U\_"05"  ; Subtract Incentive Qualifier from Paid Amount for Drug Benefit  S BPS509=BPS509-BPS521  ;  ; Default all Tax values to zero for negative values  S BPS559=$$DFF2EXT^BPSECFM($P($G(^BPSR(BPSRESP,1000,1,550)),U,9)) ; Percentage Sales Tax Paid  I BPS559<0 S BPS559=0  S BPS558=$$DFF2EXT^BPSECFM($P($G(^BPSR(BPSRESP,1000,1,550)),U,8)) ; Flat Sales Tax Paid  I BPS558<0 S BPS558=0  S BPS523=$$DFF2EXT^BPSECFM($P($G(^BPSR(BPSRESP,1000,1,500)),U,23)) ; Amount Attributed to Sales Tax  I BPS523<0 S BPS523=0  ;  ; Sales Tax Qualifier  S BPSTAX=BPS559+BPS558-BPS523  I BPSTAX<0 S BPSTAX=0  I +BPSTAX S CNT=CNT+1,BPSDAT(CNT)=BPSTAX\_U\_"10"  ; Subtract Sales Tax Qualifier from Paid Amount for Drug Benefit  S BPS509=BPS509-BPSTAX  ;  ; Set OTHER AMOUNT PAID multiples  S BPS563=0 F S BPS563=$O(^BPSR(BPSRESP,1000,1,563.01,BPS563)) Q:BPS563="" D  . S BPSQUAL=$P($G(^BPSR(BPSRESP,1000,1,563.01,BPS563,1)),U,1)  . ; Quit if qualifier = 99 since there is no NCPDP mapping for this qualifier  . Q:BPSQUAL']""!(BPSQUAL=99)  . S BPSAMNT=$$DFF2EXT^BPSECFM($P($G(^BPSR(BPSRESP,1000,1,563.01,BPS563,1)),U,2))  . ; Default negative amounts to zero  . I BPSAMNT<0 S BPSAMNT=0  . I $D(BPSOAP(BPSQUAL)) S BPSOAP(BPSQUAL)=BPSOAP(BPSQUAL)+BPSAMNT  . I '$D(BPSOAP(BPSQUAL)) S BPSOAP(BPSQUAL)=BPSAMNT  . ; Subtract Amount if Qualifier is 01, 02, 03, 04, 09 or 11  . I "010203040911"[BPSQUAL S BPS509=BPS509-BPSAMNT  I $D(BPSOAP) D  . S BPSX="" F S BPSX=$O(BPSOAP(BPSX)) Q:BPSX="" D  . . S CNT=CNT+1,BPSDAT(CNT)=BPSOAP(BPSX)\_U\_$$GETPDIEN(BPSX)  ; Set Drug Benefit Qualifier  I BPS509<0 S BPS509=0  ; Set Patient Pay Amount  S BPS505=$$DFF2EXT^BPSECFM($P($G(^BPSR(BPSRESP,1000,1,500)),U,5))  ;  S CNT=CNT+1,BPSDAT(CNT)=BPS509\_U\_$$GETPDIEN("07")\_U\_BPS505  Q | | | | |
| Modified Logic (Changes are highlighted) | | | | |
| GETOPAP(BPSRESP,BPSDAT) ;  ; Get the Other Payer Amount Paid values and qualifiers  ; Input:  ; BPSRESP = IEN of BPS RESPONSE file  ; BPSDAT(N)= Array of Other Payer fields (passed by reference)  ; [1] Paid Amount  ; [2] Qualifier  ; [3] Other Payer Patient Responsibility Amount  ; [4] Other Payer Reconciliation ID  ;  I '$G(BPSRESP) Q  I '$D(^BPSR(BPSRESP,1000)) Q  N CNT,BPS505,BPS509,BPS559,BPS558,BPS523,BPS563,BPS562,BPS521,BPS2098,BPSQUAL,BPSAMNT,BPSTAX,BPSOAP,BPSX  S CNT=0  ; Set up D.0 fields for COB segment  S BPS509=$$DFF2EXT^BPSECFM($P($G(^BPSR(BPSRESP,1000,1,500)),U,9))  ; If Total Amount Paid is a negative number, set it to zero.  ; Zero Pay amount is allowed  I BPS509<0 S BPS509=0  ;  ; Cognitive Services Qualifier/Professional Service Fee Paid  S BPS562=$$DFF2EXT^BPSECFM($P($G(^BPSR(BPSRESP,1000,1,560)),U,2))  I BPS562<0 S BPS562=0  I +BPS562 S CNT=CNT+1,BPSDAT(CNT)=BPS562\_U\_"06"  ;  ; Incentive Qualifier/Incentive Amt Paid  S BPS521=$$DFF2EXT^BPSECFM($P($G(^BPSR(BPSRESP,1000,1,500)),U,21))  I BPS521<0 S BPS521=0  I +BPS521 S CNT=CNT+1,BPSDAT(CNT)=BPS521\_U\_"05"  ; Subtract Incentive Qualifier from Paid Amount for Drug Benefit  S BPS509=BPS509-BPS521  ;  ; Default all Tax values to zero for negative values  S BPS559=$$DFF2EXT^BPSECFM($P($G(^BPSR(BPSRESP,1000,1,550)),U,9)) ; Percentage Sales Tax Paid  I BPS559<0 S BPS559=0  S BPS558=$$DFF2EXT^BPSECFM($P($G(^BPSR(BPSRESP,1000,1,550)),U,8)) ; Flat Sales Tax Paid  I BPS558<0 S BPS558=0  S BPS523=$$DFF2EXT^BPSECFM($P($G(^BPSR(BPSRESP,1000,1,500)),U,23)) ; Amount Attributed to Sales Tax  I BPS523<0 S BPS523=0  ;  ; Sales Tax Qualifier  S BPSTAX=BPS559+BPS558-BPS523  I BPSTAX<0 S BPSTAX=0  I +BPSTAX S CNT=CNT+1,BPSDAT(CNT)=BPSTAX\_U\_"10"  ; Subtract Sales Tax Qualifier from Paid Amount for Drug Benefit  S BPS509=BPS509-BPSTAX  ;  ; Set OTHER AMOUNT PAID multiples  S BPS563=0 F S BPS563=$O(^BPSR(BPSRESP,1000,1,563.01,BPS563)) Q:BPS563="" D  . S BPSQUAL=$P($G(^BPSR(BPSRESP,1000,1,563.01,BPS563,1)),U,1)  . ; Quit if qualifier = 99 since there is no NCPDP mapping for this qualifier  . Q:BPSQUAL']""!(BPSQUAL=99)  . S BPSAMNT=$$DFF2EXT^BPSECFM($P($G(^BPSR(BPSRESP,1000,1,563.01,BPS563,1)),U,2))  . ; Default negative amounts to zero  . I BPSAMNT<0 S BPSAMNT=0  . I $D(BPSOAP(BPSQUAL)) S BPSOAP(BPSQUAL)=BPSOAP(BPSQUAL)+BPSAMNT  . I '$D(BPSOAP(BPSQUAL)) S BPSOAP(BPSQUAL)=BPSAMNT  . ; Subtract Amount if Qualifier is 01, 02, 03, 04, 09 or 11  . I "010203040911"[BPSQUAL S BPS509=BPS509-BPSAMNT  I $D(BPSOAP) D  . S BPSX="" F S BPSX=$O(BPSOAP(BPSX)) Q:BPSX="" D  . . S CNT=CNT+1,BPSDAT(CNT)=BPSOAP(BPSX)\_U\_$$GETPDIEN(BPSX)  ; Set Drug Benefit Qualifier  I BPS509<0 S BPS509=0  ; Set Patient Pay Amount  S BPS505=$$DFF2EXT^BPSECFM($P($G(^BPSR(BPSRESP,1000,1,500)),U,5))  ; Set Reconciliation ID  S BPS2098=$$ANFF^BPSECFM($P($G(^BPSR(BPSRESP,1000,1,"B98")),U,1)),30)  ;  S CNT=CNT+1,BPSDAT(CNT)=BPS509\_U\_$$GETPDIEN("07")\_U\_BPS505\_U\_BPS2098  Q | | | | |

Both the PRO option and the RED action rely on DISPSEC^BPSPRRX3 to display the COB fields. DISPSEC^BPSPRRX3 will be updated to include the new field.

| Subroutine Name | **DISPSEC^BPSPRRX3** | | | |
| --- | --- | --- | --- | --- |
| Enhancement Category | New | Modify |  |  |
| Related Menu Options or ListMan Actions | PRO Option  RED Action | | | |
| Routines Which Call This Subroutine | PROMPTS^BPSPRRX3 | | | |
| Current Logic | | | | |
| BPSPRRX3 ;ALB/SS - ePharmacy secondary billing ;16-DEC-08  ;;1.0;E CLAIMS MGMT ENGINE;\*\*8,10,11,19,23\*\*;JUN 2004;Build 27  ;;Per VA Directive 6402, this routine should not be modified.  ;  . . .  DISPSEC(BPSPRARR) ;  ; Validate and Display the current secondary insurance information and prompt to edit.  ; Input:  ; BPSPRARR - Array of COB data, passed by reference  ;  N BPSPIEN,BPSCOB,BPSCOV,BPX,BPSCOV,DATA  ;  ; Other Payer IEN defaults to 1 since we don't do tertiary  S BPSPIEN=1,BPSCOB="SECONDARY"  ;  ; Get Coverage Code  S BPSCOV=$G(BPSPRARR("308-C8"))  I BPSCOV="02" S BPSCOV="02 (OTHER COVERAGE EXISTS - PAYMENT COLLECTED)"  E I BPSCOV="03" S BPSCOV="03 (OTHER COVERAGE EXISTS - THIS CLAIM NOT COVERED)"  E S BPSCOV="04 (OTHER COVERAGE EXISTS - PAYMENT NOT COLLECTED)"  ;  ; Write Data  W !!,"Data for Secondary Claim"  W !,"------------------------"  W !,"Insurance: "\_$G(BPSPRARR("INS NAME"))\_" COB: "\_BPSCOB  W !,"Rate Type: "\_$$GET1^DIQ(399.3,$G(BPSPRARR("RTYPE"))\_",",.01,,,,)  W !,"Other Coverage Code: "\_BPSCOV  W !,"Other Payer Coverage Type: 01 (PRIMARY)"  W !,"Other Payer ID Qualifier: 03 (BANK INFORMATION NUMBER (BIN))"  W !,"Other Payer ID: "\_$P($G(BPSPRARR("OTHER PAYER",BPSPIEN,0)),U,4)  W !,"Other Payer Date: "\_$$FMTE^XLFDT($P($G(BPSPRARR("OTHER PAYER",BPSPIEN,0)),U,5))  ;  ; Write Paid Amounts if previous claim if they are there  I $D(BPSPRARR("OTHER PAYER",BPSPIEN,"P")) D  . S BPX=0 F S BPX=$O(BPSPRARR("OTHER PAYER",BPSPIEN,"P",BPX)) Q:BPX="" D  . . S DATA=BPSPRARR("OTHER PAYER",BPSPIEN,"P",BPX,0)  . . W !,"Other Payer Paid Qualifier: "\_$$GET1^DIQ(9002313.2,$P(DATA,U,2),.01)\_" ("\_$$GET1^DIQ(9002313.2,$P(DATA,U,2),.02)\_")"  . . W !,"Other Payer Amount Paid: $"\_$FN($P(DATA,U,1),",",2)  . . I $P(DATA,U,3)'="" D  . . . W !,"Other Payer Patient Responsibility Amount Qualifier:"  . . . W !?40,"06 (AMT REPORTED BY PRIOR PAYER)"  . . . W !,"Other Payer Patient Responsibility Amount: $"\_$FN($P(DATA,U,3),",",2)  ;  ; Write Reject Codes if previous claims if they are there  I $D(BPSPRARR("OTHER PAYER",BPSPIEN,"R")) D  . S BPX=0 F S BPX=$O(BPSPRARR("OTHER PAYER",BPSPIEN,"R",BPX)) Q:BPX="" D  . . W !,"Other Payer Reject Code: "\_$$TRANREJ^BPSECFM($G(BPSPRARR("OTHER PAYER",BPSPIEN,"R",BPX,0)))  Q | | | | |
| Modified Logic (Changes are highlighted) | | | | |
| BPSPRRX3 ;ALB/SS - ePharmacy secondary billing ;16-DEC-08  ;;1.0;E CLAIMS MGMT ENGINE;\*\*8,10,11,19,23,24\*\*;JUN 2004;Build 27  ;;Per VA Directive 6402, this routine should not be modified.  ;  . . .  DISPSEC(BPSPRARR) ;  ; Validate and Display the current secondary insurance information and prompt to edit.  ; Input:  ; BPSPRARR - Array of COB data, passed by reference  ;  N BPSPIEN,BPSCOB,BPSCOV,BPX,BPSCOV,DATA  ;  ; Other Payer IEN defaults to 1 since we don't do tertiary  S BPSPIEN=1,BPSCOB="SECONDARY"  ;  ; Get Coverage Code  S BPSCOV=$G(BPSPRARR("308-C8"))  I BPSCOV="02" S BPSCOV="02 (OTHER COVERAGE EXISTS - PAYMENT COLLECTED)"  E I BPSCOV="03" S BPSCOV="03 (OTHER COVERAGE EXISTS - THIS CLAIM NOT COVERED)"  E S BPSCOV="04 (OTHER COVERAGE EXISTS - PAYMENT NOT COLLECTED)"  ;  ; Write Data  W !!,"Data for Secondary Claim"  W !,"------------------------"  W !,"Insurance: "\_$G(BPSPRARR("INS NAME"))\_" COB: "\_BPSCOB  W !,"Rate Type: "\_$$GET1^DIQ(399.3,$G(BPSPRARR("RTYPE"))\_",",.01,,,,)  W !,"Other Coverage Code: "\_BPSCOV  W !,"Other Payer Coverage Type: 01 (PRIMARY)"  W !,"Other Payer ID Qualifier: 03 (BANK INFORMATION NUMBER (BIN))"  W !,"Other Payer ID: "\_$P($G(BPSPRARR("OTHER PAYER",BPSPIEN,0)),U,4)  W !,"Other Payer Date: "\_$$FMTE^XLFDT($P($G(BPSPRARR("OTHER PAYER",BPSPIEN,0)),U,5))  ;  ; Write Paid Amounts if previous claim if they are there  I $D(BPSPRARR("OTHER PAYER",BPSPIEN,"P")) D  . S BPX=0 F S BPX=$O(BPSPRARR("OTHER PAYER",BPSPIEN,"P",BPX)) Q:BPX="" D  . . S DATA=BPSPRARR("OTHER PAYER",BPSPIEN,"P",BPX,0)    . . W !,"Other Payer Reconciliation ID: "\_$P(DATA,U,4)  . . W !,"Other Payer Paid Qualifier: "\_$$GET1^DIQ(9002313.2,$P(DATA,U,2),.01)\_" ("\_$$GET1^DIQ(9002313.2,$P(DATA,U,2),.02)\_")"  . . W !,"Other Payer Amount Paid: $"\_$FN($P(DATA,U,1),",",2)  . . I $P(DATA,U,3)'="" D  . . . W !,"Other Payer Patient Responsibility Amount Qualifier:"  . . . W !?40,"06 (AMT REPORTED BY PRIOR PAYER)"  . . . W !,"Other Payer Patient Responsibility Amount: $"\_$FN($P(DATA,U,3),",",2)  ;  ; Write Reject Codes if previous claims if they are there  I $D(BPSPRARR("OTHER PAYER",BPSPIEN,"R")) D  . S BPX=0 F S BPX=$O(BPSPRARR("OTHER PAYER",BPSPIEN,"R",BPX)) Q:BPX="" D  . . W !,"Other Payer Reject Code: "\_$$TRANREJ^BPSECFM($G(BPSPRARR("OTHER PAYER",BPSPIEN,"R",BPX,0)))  Q | | | | |