**MCCF EDI TAS US-9 (USEB-16)**

System Design Document

IB\*2.0\*608



Department of Veterans Affairs

**February 2018**

Version 2.0

**User Story Number:** USEB-16

**User Story Name:** Transmitting SNF Claims with Appropriate Revenue Codes

**Product Backlog ID:** #146

**Rally ID:** US-9

# Design/Resolution:

To resolve this request, the following rules need to be followed:

* + Claim must be a SNF Claim
  + UB-04 Bill Classification must be either 1, 2 or 3
  + Receiver ID must be “ENVOYH” or “PARTA”
  + Payer Responsibility Sequence must be “P”
  + Payer Primary ID must be “12M61”

If the above rules are met, then if the VCx record has a value code of “80”, add new INS and LDAT output lines.

* Institute a new variable (IBXSV) to handle the data manipulation for the eligible SNF claims in the 364.7 [478] Output Formatter record:

**From:**

PIECE 1 RECORD ID 'BGN ' 364.6[582] 364.7[478] 364.5[131] Length=4 Max Lines=0

>S IBXDATA=""

>S:'$O(^TMP("IBXMSG",$J,""),-1) IBXDATA="BGN " S:$G(IBXDATA)="" IBXNOREQ=1 S

^TMP("IBTX",$J,IBXIEN)=1

PIECE 2 Segment Delimiter 364.6[583] 364.7[479] 36

4.5[136] Length=1 Max Lines=0

>Constant Value: "~"

>I $D(^TMP("IBHDR",$J)) K IBXDATA S IBXNOREQ=1

**To:**

PIECE 1 RECORD ID 'BGN ' 364.6[582] 364.7[478] 364.5[131] Length=4 Max Lines=0

>S IBXDATA=""

>K IBXSV S:'$O(^TMP("IBXMSG",$J,""),-1) IBXDATA="BGN " S:$G(IBXDATA)="" IBXNOREQ=1 S ^TMP("IBTX",$J,IBXIEN)=1

PIECE 2 Segment Delimiter 364.6[583] 364.7[479] 36

4.5[136] Length=1 Max Lines=0

>Constant Value: "~"

>I $D(^TMP("IBHDR",$J)) K IBXDATA S IBXNOREQ=1

* To determine if a claim is a valid SNF claim to generate the appropriate “INS” and “LDAT” line items for viewing and transmitting purposes, based upon the claims VALUE CODEs, the following Output Formatter record (#364.7 record 105) needs to be modified:

**From:**

PIECE 1 RECORD ID 'VC1 ' 364.6[244] 364.7[235] 364.5[131] Length=4 Max Lines=0

>S IBXDATA=""

>S IBXDATA="VC1 "

PIECE 1.9 VALUE CODE GROUP 364.6[161] 364.7[106] 364.5[374]

\*\*\*\*\* CALCULATE ONLY, NO OUTPUT \*\*\*\*\*

>K IBXDATA D VC1^IBCVC(IBXIEN,.IBFL)

>K IBXDATA N Z K:$D(IBFL(39))=1 IBFL(39) S Z=0 F S Z=$O(IBFL(39,Z)) K:'Z IBFL(39) Q:'Z S IBXSAVE("VALUE",Z)=IBFL(39,Z)

PIECE 2 Value Code 364.6[88] 364.7[105] 364.5[5] Length=3 Max Lines=0

>Constant Value: ""

> K IBXDATA N Z,Z0,R S (Z,R)=0 F S Z=$O(IBXSAVE("VALUE",Z)) Q:'Z S Z0=$P(IBXSAVE("VALUE",Z),

U) I Z0'="" S IBXDATA(Z)=Z0 D ID^IBCEF2(Z,"VC"\_$$M^IBCEF2(.R))

**To:**

PIECE 1 RECORD ID 'VC1 ' 364.6[244] 364.7[235] 364.5[131] Length=4 Max Lines=0

>S IBXDATA=""

>S IBXDATA="VC1 "

PIECE 1.9 VALUE CODE GROUP 364.6[161] 364.7[106] 364.5[374]

\*\*\*\*\* CALCULATE ONLY, NO OUTPUT \*\*\*\*\*

>K IBXDATA D VC1^IBCVC(IBXIEN,.IBFL)

>K IBXDATA N Z K:$D(IBFL(39))=1 IBFL(39) S Z=0 F S Z=$O(IBFL(39,Z)) K:'Z IBFL(39) Q:'Z S IBXSAVE("VALUE",Z)=IBFL(39,Z)

PIECE 2 Value Code 364.6[88] 364.7[105] 364.5[5] Length=3 Max Lines=0

>Constant Value: ""

> K IBXDATA,VC80 N R,VZ,Z,Z0 S (Z,R)=0,VC80=$$VC80CK^IBCEF2(IBIFN) F S Z=$O(IBXSAVE("VALUE",

Z)) Q:'Z S VZ=IBXSAVE("VALUE",Z) S:+VC80 IBXSV("VC80",Z)=$P(VZ,U,2,3) S Z0=$P(VZ,U) I Z0'="" S IBXDATA(Z)=Z0 D ID^IBCEF2(Z,"VC"\_$$M^IBCEF2(.R))

[NOTE: VC80 means the Claim is eligible because it is a SNF, has the right BILL TYPE, RECEIVER ID, PAYER RESPONSIBILITY SEQUENCE and PAYER PRIMARY ID]

* The routine ^IBCEF2, will be modified to validate a claim based on the rules above:

| **Routines** | **Activities** | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Routine Name** | IBCEF2 | | | | | | | | |
| **Enhancement Category** | New | | Modify | Delete | | | No Change | | |
| **RTM** |  | | | | | | | | |
| **Related Options** | None | | | | | | | | |
| **Related Routines** | **Routines “Called By”** | | | | | **Routines “Called”** | | | |
|  |  | | | | |  | | | |
| **Data Dictionary (DD) References** | None | | | | | | | | |
| **Related Protocols** | None | | | | | | | | |
| **Related Integration Control Registrations (ICRs)** | None | | | | | | | | |
| **Data Passing** | Input | Output Reference | | | Both | | | Global Reference | Local |
| **Input Attribute Name and Definition** | Name:  Definition: | | | | | | | | |
| **Output Attribute Name and Definition** | Name:  Definition: | | | | | | | | |
| **Current Logic** | | | | | | | | | |
| IBCEF2 ;ALB/TMP - FORMATTER SPECIFIC BILL FUNCTIONS ;8/6/03 10:54am  ;;2.0;INTEGRATED BILLING;\*\*52,85,51,137,232,155,296,349,403,400,432,488,461,547,592\*\*;21-MAR-94;Build 25  ;;Per VA Directive 6402, this routine should not be modified.  ;        ; ID(LN,VAL) ; Set EXTRACT GLOBAL for multi-valued record  ; ids for Austin  ; LN = the line # being extracted  ; VAL = the value of the element being extracted  ;  ; Assumes IBXPG exists  ;  Q:LN<2  D SETGBL^IBCEFG(IBXPG,LN,1,VAL,.IBXSIZE)   Q  ;        ; AMTOUT(A,B,C,IBXSAVE) ; format output amount  ;  N Z,K,IBZ,IBARR K IBXDATA S (IBZ,K)=0,IBARR="IBXSAVE("""\_A\_""")" F  S IBZ=$O(@IBARR@(IBZ)) Q:'IBZ  S K=K+1,Z=0 F  S Z=$O(@IBARR@(IBZ,Z)) Q:'Z  I $P($G(@IBARR@(IBZ,Z,B)),U,C) S IBXDATA(K)=$$DOLLAR^IBCEFG1($G(IBXDATA(K))+$P(@IBARR@(IBZ,Z,B),U,C))  Q | | | | | | | | | |
| **Modified Logic (Changes are in bold)** | | | | | | | | | |
| IBCEF2 ;ALB/TMP - FORMATTER SPECIFIC BILL FUNCTIONS ;8/6/03 10:54am  ;;2.0;INTEGRATED BILLING;\*\*52,85,51,137,232,155,296,349,403,400,432,488,461,547,592,608\*\*;21-MAR-94;Build 25  ;;Per VA Directive 6402, this routine should not be modified.  ;        ; ID(LN,VAL) ; Set EXTRACT GLOBAL for multi-valued record  ; ids for Austin  ; LN = the line # being extracted  ; VAL = the value of the element being extracted  ;  ; Assumes IBXPG exists  ;  Q:LN<2  D SETGBL^IBCEFG(IBXPG,LN,1,VAL,.IBXSIZE)  ;  ;/Begin IB\*2.0\*608 - US9 - vd  I +$$VC80 D VC80I^IBCEF22(LN,+$G(VC80),$G(IBXSV("VC80",LN))) ; Process for 'SNF' claims  ;/End IB\*2.0\*608  Q  ;        ; AMTOUT(A,B,C,IBXSAVE) ; format output amount  ;  N Z,K,IBZ,IBARR K IBXDATA S (IBZ,K)=0,IBARR="IBXSAVE("""\_A\_""")" F  S IBZ=$O(@IBARR@(IBZ)) Q:'IBZ  S K=K+1,Z=0 F  S Z=$O(@IBARR@(IBZ,Z)) Q:'Z  I $P($G(@IBARR@(IBZ,Z,B)),U,C) S IBXDATA(K)=$$DOLLAR^IBCEFG1($G(IBXDATA(K))+$P(@IBARR@(IBZ,Z,B),U,C))  Q   ;  ;/Beginning of IB\*2.0\*608 (US9) - vd SNF(IBIFN) ; Check to see if the claim is a SNF (Skilled Nursing Facility) Claim.  ; Returned Values:  ; SNF = 0 if claim is not a SNF Claim.  ; SNF = 1 if claim is a SNF Claim.  N SNF  S SNF=0  I $$GET1^DIQ(399,IBIFN\_", ",.24,"I")=2 S SNF=1 ; Claim is a SNF Claim  Q SNF  ; VC80CK(IBIFN) ; Determine if the Claim is eligible for Value Code 80 Revenue Code Claim Lines.  N BLTYPX,COB,IB0,NUM,PPAYID,RCVRID,VC80SW  S VC80SW=0  I '$$SNF(IBIFN) Q VC80SW   ; Not a SNF Claim.  S RCVRID=$$RECVR^IBCEF2(IBIFN) I "^ENVOYH^PARTA^"'[(U\_RCVRID\_U) Q VC80SW   ; Not ENVOYH or PARTA  S COB=$$COB^IBCEF(IBIFN) I COB'="P" Q VC80SW   ; Payer Responsibility Sequence not equal to "P".  D ALLPAYID(IBIFN,.NUM,1) S PPAYID=$G(NUM(1)) Q:(PPAYID'="12M61") VC80SW  ; Primary Payer not equal to "12M61"  S IB0=$G(^DGCR(399,IBIFN,0))  S BLTYPX=$P(IB0,U,24)\_$P(IB0,U,5)  I BLTYPX<21!(BLTYPX>23) Q VC80SW  ; Not a valid Bill Type.  S VC80SW=1 ; If we got this far.the claim is eligible for Value Code 80 Revenue Code Claim Lines.  Q VC80SW  ; INS ; Called by the Output Formatter [#364.7, 176]  N A,Z  S Z=0,A=$G(^TMP($J,"IBLCT"))  F  S Z=$O(IBXDATA(Z)) D  Q:'Z  . K:'Z&($D(IBXDATA)=1) IBXDATA  . S:'Z ^TMP($J,"IBLCT")=A Q:'Z  . S A=A+1 M IBXSAVE("INPT",Z)=IBXDATA(Z)  . K IBXDATA(Z) S IBXDATA(Z)=A  . D:Z>1 ID^IBCEF2(Z,"INS ")  ;  I +$G(VC80) D  . S Z=$O(IBXDATA(""),-1)+1  . D ID^IBCEF2(Z,"INS ")  . D VC80I^IBCEF22(A,$G(IBXSV("VC80",A))) ; Process for 'SNF' claims & the last claim line  . S A=A+1  . S IBXDATA(Z)=A  Q  ; INS05 ; Called by the Output Formatter [#364.7, 178]  ;K IBXDATA S IBXNOREQ='$$REQ^IBCEF1(3,"",IBXIEN) N Z S Z=0 F S Z=$O(IBXSAVE("INPT",Z)) Q:'Z S:$P(IBXSAVE("INPT",Z),U,4)'="" IBXDATA(Z)=$P(IBXSAVE("INPT",Z),U,4)  K IBXDATA  S IBXNOREQ='$$REQ^IBCEF1(3,"",IBXIEN)  N LAST,Z S (LAST,Z)=0  F  S Z=$O(IBXSAVE("INPT",Z)) Q:'Z  D  Q:LAST  . I +$G(VC80) S:$O(IBXSAVE("INPT",Z))="" LAST=1,IBXDATA(Z)=$P(IBXSAVE("INPT",Z),U,4) Q:LAST  . S:$P(IBXSAVE("INPT",Z),U,4)'="" IBXDATA(Z)=$P(IBXSAVE("INPT",Z),U,4)  Q  ; INS07 ; Called by the Output Formatter [#364.7, 181]  ;N Z S Z=0 F S Z=$O(IBXSAVE("INPT",Z)) Q:'Z S:$P(IBXSAVE("INPT",Z),U,9)'=""&($P(IBXSAVE("INPT",Z),U,2)'="") IBXSAVE("PMOD",Z)=$P(IBXSAVE("INPT",Z),U,9),IBXDATA(Z)=$P(IBXSAVE("PMOD",Z),",")  N LAST,Z S (LAST,Z)=0  F  S Z=$O(IBXSAVE("INPT",Z)) Q:'Z  D  Q:LAST  . I +$G(VC80) S:$O(IBXSAVE("INPT",Z))="" LAST=1,IBXDATA(Z)="" Q:LAST  . S:$P(IBXSAVE("INPT",Z),U,9)'=""&($P(IBXSAVE("INPT",Z),U,2)'="") IBXSAVE("PMOD",Z)=$P(IBXSAVE("INPT",Z),U,9),IBXDATA(Z)=$P(IBXSAVE("PMOD",Z),",")  Q  ; INS09 ; Called by the Output Formatter [#364.7, 180]  ;K IBXDATA N Z S Z=0 F S Z=$O(IBXSAVE("INPT",Z)) Q:'Z S IBXDATA(Z)=$S($P(IBXSAVE("INPT",Z),U,5)="":$$DOLLAR^IBCEFG1("0.00"),1:$$DOLLAR^IBCEFG1($P(IBXSAVE("INPT",Z),U,5)))  K IBXDATA  N LAST,Z S (LAST,Z)=0  F  S Z=$O(IBXSAVE("INPT",Z)) Q:'Z  D  Q:LAST  . I +$G(VC80) S:$O(IBXSAVE("INPT",Z))="" LAST=1,IBXDATA(Z)=0 Q:LAST  . S IBXDATA(Z)=$S($P(IBXSAVE("INPT",Z),U,5)="":$$DOLLAR^IBCEFG1("0.00"),1:$$DOLLAR^IBCEFG1($P(IBXSAVE("INPT",Z),U,5)))  Q  ; INS12 ; Called by the Output Formatter [#364.7, 482]  ;K IBXDATA N Z S Z=0 F S Z=$O(IBXSAVE("INPT",Z)) Q:'Z S:$P(IBXSAVE("INPT",Z),U,6)'="" IBXDATA(Z)=$$DOLLAR^IBCEFG1($P(IBXSAVE("INPT",Z),U,6))  K IBXDATA  N LAST,Z S (LAST,Z)=0  F  S Z=$O(IBXSAVE("INPT",Z)) Q:'Z  D  Q:LAST  . I +$G(VC80) S:$O(IBXSAVE("INPT",Z))="" LAST=1,IBXDATA(Z)="" Q:LAST  . S:$P(IBXSAVE("INPT",Z),U,6)'="" IBXDATA(Z)=$$DOLLAR^IBCEFG1($P(IBXSAVE("INPT",Z),U,6))  Q  ; INS13 ; Called by the Output Formatter [#364.7, 805]  ;K IBXDATA N Z,Z0 S Z=0 F S Z=$O(IBXSAVE("INPT",Z)) Q:'Z S:$P(IBXSAVE("INPT",Z),U)'="" Z0=$P(IBXSAVE("INPT",Z),U) S IBXDATA(Z)=$S(Z0="":"",Z0'<100&(Z0'>219):"DA",1:"UN") K:IBXDATA(Z)="" IBXDATA(Z) K IBXDATA  K IBXDATA  N LAST,Z,Z0 S (LAST,Z)=0  F  S Z=$O(IBXSAVE("INPT",Z)) Q:'Z  D  Q:LAST  . I +$G(VC80) S:$O(IBXSAVE("INPT",Z))="" LAST=1,IBXDATA(Z)=$P(IBXSAVE("INPT",Z),U,13) Q:LAST  . S:$P(IBXSAVE("INPT",Z),U)'="" Z0=$P(IBXSAVE("INPT",Z),U)  . S IBXDATA(Z)=$S(Z0="":"",Z0'<100&(Z0'>219):"DA",1:"UN") K:IBXDATA(Z)="" IBXDATA(Z)  Q  ;/End IB\*2.0\*608 (US9) - vd  ; | | | | | | | | | |

* For the ‘INS ’ record that needs to be created, the following Output Formatter record (#364.7 records 176, 178, 179, 180, 181, 189, 482 and 805) needs to be modified to generate an “INS” line item for viewing and transmitting purposes:

**From:**

PIECE 1 RECORD ID 'INS ' 364.6[252] 364.7[242] 364.5[131] Length=4 Max Lines=0

>S IBXDATA=""

>S IBXDATA="INS "

PIECE 2 Service Line Counter 364.6[123] 364.7[176] 364.5[8] Length=6 Max Lines=0

>S IBXDATA="" I $$FT^IBCEF(IBXIEN)=3 K:$$INSPRF^IBCEF(IBXIEN)=1 IBXSAVE("DATE

") S:'$$INPAT^IBCEF(IBXIEN,1) IBXSAVE("DATE")=$$SERVDT^IBCEF(IBXIEN,8,1) D HOS^I

BCEF2(IBXIEN)

>N A,Z S Z=0,A=$G(^TMP($J,"IBLCT")) F S Z=$O(IBXDATA(Z)) K:'Z&($D(IBXDATA)=1

) IBXDATA S:'Z ^TMP($J,"IBLCT")=A Q:'Z S A=A+1 M IBXSAVE("INPT",Z)=IBXDATA(Z) K

IBXDATA(Z) S IBXDATA(Z)=A D:Z>1 ID^IBCEF2(Z,"INS ")



PIECE 4 Procedure Code 364.6[125] 364.7[189] 364.5[5] Length=10 Max Lines=0

>Constant Value: ""

>K IBXDATA N Z S Z=0 F S Z=$O(IBXSAVE("INPT",Z)) Q:'Z S:$P(IBXSAVE("INPT",Z

),U,2)'="" IBXDATA(Z)=$$PRCD^IBCEF1($P(IBXSAVE("INPT",Z),U,2)\_";ICPT(")

PIECE 5 Service Unit Count 364.6[126] 364.7[178] 364.5[5] Length=6 Max Lines=0

>Constant Value: ""

>K IBXDATA S IBXNOREQ='$$REQ^IBCEF1(3,"",IBXIEN) N Z S Z=0 F S Z=$O(IBXSAVE(

"INPT",Z)) Q:'Z S:$P(IBXSAVE("INPT",Z),U,4)'="" IBXDATA(Z)=$P(IBXSAVE("INPT",Z)

,U,4)

PIECE 6 BLANK 364.6[127] 364.7[179] 364.5[5] Length=1 Max Lines=0

>Constant Value: ""

>S IBXDATA=""

PIECE 7 Procedure Modifier(1) 364.6[128] 364.7[181] 364.5[5] Length=2 Max Lines=0

>Constant Value: ""

>N Z S Z=0 F S Z=$O(IBXSAVE("INPT",Z)) Q:'Z S:$P(IBXSAVE("INPT",Z),U,9)'=""

&($P(IBXSAVE("INPT",Z),U,2)'="") IBXSAVE("PMOD",Z)=$P(IBXSAVE("INPT",Z),U,9),IBX

DATA(Z)=$P(IBXSAVE("PMOD",Z),",")



PIECE 9 Service Line Charge Amt 364.6[130] 364.7[180] 364.5[5] Length=18 Max Lines=0

>Constant Value: ""

>K IBXDATA N Z S Z=0 F S Z=$O(IBXSAVE("INPT",Z)) Q:'Z S IBXDATA(Z)=$S($P(IB

XSAVE("INPT",Z),U,5)="":$$DOLLAR^IBCEFG1("0.00"),1:$$DOLLAR^IBCEFG1($P(IBXSAVE("

INPT",Z),U,5)))



PIECE 12 Service Line Non-covered Charge Amt 364.6[579] 364.7[482] 364.5[5] Length=18 Max Lines=0

>Constant Value: ""

>K IBXDATA N Z S Z=0 F S Z=$O(IBXSAVE("INPT",Z)) Q:'Z S:$P(IBXSAVE("INPT",Z

),U,6)'="" IBXDATA(Z)=$$DOLLAR^IBCEFG1($P(IBXSAVE("INPT",Z),U,6))

PIECE 13 Units/Basis for Measurement Code 364.6[864] 364.7[805] 364.5[5] Length=2 Max Lines=0

>Constant Value: ""

>K IBXDATA N Z,Z0 S Z=0 F S Z=$O(IBXSAVE("INPT",Z)) Q:'Z S:$P(IBXSAVE("INPT

",Z),U)'="" Z0=$P(IBXSAVE("INPT",Z),U) S IBXDATA(Z)=$S(Z0="":"",Z0'<100&(Z0'>219

):"DA",1:"UN") K:IBXDATA(Z)="" IBXDATA(Z)



**To:**

PIECE 1 RECORD ID 'INS ' 364.6[252] 364.7[242] 364.5[131] Length=4 Max Lines=0

>S IBXDATA=""

>S IBXDATA="INS "

PIECE 2 Service Line Counter 364.6[123] 364.7[176] 364.5[8] Length=6 Max Lines=0

>S IBXDATA="" I $$FT^IBCEF(IBXIEN)=3 K:$$INSPRF^IBCEF(IBXIEN)=1 IBXSAVE("DATE

") S:'$$INPAT^IBCEF(IBXIEN,1) IBXSAVE("DATE")=$$SERVDT^IBCEF(IBXIEN,8,1) D HOS^I

BCEF2(IBXIEN)

>D INS^IBCEF2



PIECE 4 Procedure Code 364.6[125] 364.7[189] 364.5[5] Length=10 Max Lines=0

>Constant Value: ""

>K IBXDATA N Z,PCX S Z=0 F S Z=$O(IBXSAVE("INPT",Z)) Q:'Z S PCX=$P(IBXSAVE(

"INPT",Z),U,2) S:PCX'="" IBXDATA(Z)=$S(+$G(VC80):PCX,1:$$PRCD^IBCEF1(PCX\_";ICPT(

"))

PIECE 5 Service Unit Count 364.6[126] 364.7[178] 364.5[5] Length=6 Max Lines=0

>Constant Value: ""

>D INS05^IBCEF2

PIECE 6 BLANK 364.6[127] 364.7[179] 364.5[5] Length=1 Max Lines=0

>Constant Value: ""

>S IBXDATA="" I +$G(VC80) S Z=$O(IBXSAVE("INPT",""),-1) S:$P(IBXSAVE("INPT",Z

),U,2)="AAA00" IBXDATA(Z)=$P(IBXSAVE("INPT",Z),U,6)

PIECE 7 Procedure Modifier(1) 364.6[128] 364.7[181] 364.5[5] Length=2 Max Lines=0

>Constant Value: ""

>D INS07^IBCEF2



PIECE 9 Service Line Charge Amt 364.6[130] 364.7[180] 364.5[5] Length=18 Max Lines=0

>Constant Value: ""

>D INS09^IBCEF2



PIECE 12 Service Line Non-covered Charge Amt 364.6[579] 364.7[482] 364.5[5] Length=18 Max Lines=0

>Constant Value: ""

>D INS12^IBCEF2

PIECE 13 Units/Basis for Measurement Code 364.6[864] 364.7[805] 364.5[5] Length=2 Max Lines=0

>Constant Value: ""

>D INS13^IBCEF2

* Create the VC80I^IBCEF22 module of code to extract the data for the new INS record for Value Code 80

| **Routines** | **Activities** | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Routine Name** | IBCEF22 | | | | | | | | |
| **Enhancement Category** | New | | Modify | Delete | | | No Change | | |
| **RTM** |  | | | | | | | | |
| **Related Options** | None | | | | | | | | |
| **Related Routines** | **Routines “Called By”** | | | | | **Routines “Called”** | | | |
|  |  | | | | |  | | | |
| **Data Dictionary (DD) References** | None | | | | | | | | |
| **Related Protocols** | None | | | | | | | | |
| **Related Integration Control Registrations (ICRs)** | None | | | | | | | | |
| **Data Passing** | Input | Output Reference | | | Both | | | Global Reference | Local |
| **Input Attribute Name and Definition** | Name:  Definition: | | | | | | | | |
| **Output Attribute Name and Definition** | Name: | | | | | | | | |
| **Current Logic** | | | | | | | | | |
| IBCEF22 ;ALB/TMP - FORMATTER SPECIFIC BILL FUNCTIONS ;06-FEB-96  ;;2.0;INTEGRATED BILLING;\*\*51,137,135,155,309,349,389,432,488,516,577\*\*;21-MAR-94;Build 38  ;;Per VA Directive 6402, this routine should not be modified.  ;  ; OVERFLOW FROM ROUTINE IBCEF2        ; SPLIT    ; Split codes into multiple lines as needed => baa ; 488  ; The max line $ amount for a printed claim is less than the max line $ amount for an electronically transmitted claim.  ; However, since the clearinghouse can drop to print for a myriad of reasons at any time, the lines may need to be split  ; so they can all fit on a printed claim line just in case. In addition, since some claims are sent to primary payers as   ; electronic claims but printed for secondary claims, the lines numbers need to be the same going out to ensure the   ; COB data is correct applied (previous payments adj, etc are applied to the correct line.)  N IBS,IBSS,DATA,CHRG,UNTS,TOT,LNS,MOD,CPT,LNK,RLNK,IBSS1,LTOT,LUNT,REC,LST,FST  S IBS="",IBLN=0  F  S IBS=$O(IBX1(IBS)) Q:IBS=""  D  . S LST=$O(IBX1(IBS,""),-1) ;we have to go through each level so must reset for each  . S LNK=0  . F  S LNK=$O(IBX1(IBS,LNK)) Q:('LNK!(LNK>LST)) S IBSS="" F  S IBSS=$O(IBX1(IBS,LNK,IBSS)) Q:IBSS=""  D  .. S DATA=IBX1(IBS,LNK,IBSS,2)  .. S CHRG=$P(DATA,U,2)  .. S UNTS=$P(DATA,U,3)  .. I UNTS=1 Q  ; if only one unit can't split  .. S TOT=UNTS\*CHRG  .. I TOT<=9999999.99 Q  ; if the total is less tham max we don't need to split  .. S LNS=TOT\9999999.99  .. S MOD=TOT#9999999.99  .. I MOD S LNS=LNS+1  .. I CHRG>4999999.995 S LNS=UNTS  ; if the charge is greater than half the mas can't put more than one on a line.  .. S LUNT=UNTS\LNS  .. S MOD=UNTS#LNS  .. I MOD S LUNT=LUNT+1  .. F L=1:1:LNS D  ... N Q  ... S Q=$O(IBX1(IBS,""),-1)+1  ... I L=1 S Q=LNK  ... M IBX1(IBS,Q,IBSS)=IBX1(IBS,LNK,IBSS)  ... S $P(IBX1(IBS,Q,IBSS,2),U,3)=LUNT,$P(IBX1(IBS,Q,IBSS,2),U,4)=LUNT\*CHRG  ... S $P(IBX1(IBS,Q,IBSS),U,1)=LUNT,$P(IBX1(IBS,Q,IBSS),U,2)=LUNT\*CHRG  ... I L>1 S $P(IBX1(IBS,Q,IBSS,2),U,9)=""  ... S UNTS=UNTS-LUNT,LUNT=$S(UNTS>LUNT:LUNT,1:UNTS)  Q | | | | | | | | | |
| **Modified Logic (Changes are in bold)** | | | | | | | | | |
| IBCEF22 ;ALB/TMP - FORMATTER SPECIFIC BILL FUNCTIONS ;06-FEB-96  ;;2.0;INTEGRATED BILLING;\*\*51,137,135,155,309,349,389,432,488,516,577,608\*\*;21-MAR-94;Build 38  ;;Per VA Directive 6402, this routine should not be modified.  ;  ; OVERFLOW FROM ROUTINE IBCEF2        ; SPLIT    ; Split codes into multiple lines as needed => baa ; 488  ; The max line $ amount for a printed claim is less than the max line $ amount for an electronically transmitted claim.  ; However, since the clearinghouse can drop to print for a myriad of reasons at any time, the lines may need to be split  ; so they can all fit on a printed claim line just in case. In addition, since some claims are sent to primary payers as   ; electronic claims but printed for secondary claims, the lines numbers need to be the same going out to ensure the   ; COB data is correct applied (previous payments adj, etc are applied to the correct line.)  N IBS,IBSS,DATA,CHRG,UNTS,TOT,LNS,MOD,CPT,LNK,RLNK,IBSS1,LTOT,LUNT,REC,LST,FST  S IBS="",IBLN=0  F  S IBS=$O(IBX1(IBS)) Q:IBS=""  D  . S LST=$O(IBX1(IBS,""),-1) ;we have to go through each level so must reset for each  . S LNK=0  . F  S LNK=$O(IBX1(IBS,LNK)) Q:('LNK!(LNK>LST)) S IBSS="" F  S IBSS=$O(IBX1(IBS,LNK,IBSS)) Q:IBSS=""  D  .. S DATA=IBX1(IBS,LNK,IBSS,2)  .. S CHRG=$P(DATA,U,2)  .. S UNTS=$P(DATA,U,3)  .. I UNTS=1 Q  ; if only one unit can't split  .. S TOT=UNTS\*CHRG  .. I TOT<=9999999.99 Q  ; if the total is less tham max we don't need to split  .. S LNS=TOT\9999999.99  .. S MOD=TOT#9999999.99  .. I MOD S LNS=LNS+1  .. I CHRG>4999999.995 S LNS=UNTS  ; if the charge is greater than half the mas can't put more than one on a line.  .. S LUNT=UNTS\LNS  .. S MOD=UNTS#LNS  .. I MOD S LUNT=LUNT+1  .. F L=1:1:LNS D  ... N Q  ... S Q=$O(IBX1(IBS,""),-1)+1  ... I L=1 S Q=LNK  ... M IBX1(IBS,Q,IBSS)=IBX1(IBS,LNK,IBSS)  ... S $P(IBX1(IBS,Q,IBSS,2),U,3)=LUNT,$P(IBX1(IBS,Q,IBSS,2),U,4)=LUNT\*CHRG  ... S $P(IBX1(IBS,Q,IBSS),U,1)=LUNT,$P(IBX1(IBS,Q,IBSS),U,2)=LUNT\*CHRG  ... I L>1 S $P(IBX1(IBS,Q,IBSS,2),U,9)=""  ... S UNTS=UNTS-LUNT,LUNT=$S(UNTS>LUNT:LUNT,1:UNTS)  Q   ;  ; /Begin IB\*2.0\*608 - US9 - vd VC80I(LN,VC80REC) ; Extracts the data for the "INS" record for VALUE CODE 80 Line item.  ; INPUT: LN = Line counter  ; VC80REC = Value Code 80 Line Item.  ;  N AMT,VC80LN,VC80VAL  S AMT=$P(VC80REC,U,2) ; Amount  S VC80LN=LN+1 ; Get the next available line number.  S VC80VAL=$P(VC80REC,U) ; Value Code  ;  S $P(IBXSAVE("INPT",VC80LN),U,1)="0022"  S $P(IBXSAVE("INPT",VC80LN),U,2)="AAA00"  S $P(IBXSAVE("INPT",VC80LN),U,4)=$S(VC80VAL=80:AMT,1:0)  S $P(IBXSAVE("INPT",VC80LN),U,6)=0  S $P(IBXSAVE("INPT",VC80LN),U,9)=0  S $P(IBXSAVE("INPT",VC80LN),U,13)=$S(VC80VAL=80:"DA",1:"UN")  Q  ; /End IB\*2.0\*608  ; | | | | | | | | | |

* For the ‘LDAT’ records that need to be created, the following Output Formatter records (#364.7 record 1752 and 1765) need to be modified to generate an “LDAT” line item for viewing and transmitting purposes:

**From:**

PIECE 1 RECORD ID 'LDAT' 364.6[2050] 364.7[1750] 364.5[131] Length=4 Max Lines=0

>S IBXDATA=""

>S IBXDATA="LDAT"

PIECE 1.9 LDAT DATA EXTRACT 364.6[2052] 364.7[1752] 364.5[5]

\*\*\*\*\* CALCULATE ONLY, NO OUTPUT \*\*\*\*\*

>Constant Value: ""

>K IBXSAVE("LDAT") D GETLDAT^IBCEF11(IBXIEN)



PIECE 99.9 Cleanup 364.6[2065] 364.7[1765] 364.5[5]

\*\*\*\*\* CALCULATE ONLY, NO OUTPUT \*\*\*\*\*

>Constant Value: ""

>K IBXSAVE("LDAT")

**To:**

PIECE 1 RECORD ID 'LDAT' 364.6[2050] 364.7[1750] 364.5[131] Length=4 Max Lines=0

>S IBXDATA=""

>S IBXDATA="LDAT"

PIECE 1.9 LDAT DATA EXTRACT 364.6[2052] 364.7[1752] 364.5[5]

\*\*\*\*\* CALCULATE ONLY, NO OUTPUT \*\*\*\*\*

>Constant Value: ""

>K IBXSAVE("LDAT") D GETLDAT^IBCEF11(IBXIEN) D:$D(IBXSV(“VC80”)) VC80L^IBCEF11(IBXIEN,

+$G(IBXSV(“VC80LN”))



PIECE 99.9 Cleanup 364.6[2065] 364.7[1765] 364.5[5]

\*\*\*\*\* CALCULATE ONLY, NO OUTPUT \*\*\*\*\*

>Constant Value: ""

>K IBXSAVE("LDAT"),IBXSV(“VC80”),IBXSV(“VC80LN”)

* Create the VC80L^IBCEF11 module of code to extract the data for the new LDAT record for Value Code 80

| **Routines** | **Activities** | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Routine Name** | IBCEF11 | | | | | | | | |
| **Enhancement Category** | New | | Modify | Delete | | | No Change | | |
| **RTM** |  | | | | | | | | |
| **Related Options** | None | | | | | | | | |
| **Related Routines** | **Routines “Called By”** | | | | | **Routines “Called”** | | | |
|  |  | | | | |  | | | |
| **Data Dictionary (DD) References** | None | | | | | | | | |
| **Related Protocols** | None | | | | | | | | |
| **Related Integration Control Registrations (ICRs)** | None | | | | | | | | |
| **Data Passing** | Input | Output Reference | | | Both | | | Global Reference | Local |
| **Input Attribute Name and Definition** | Name:  Definition: | | | | | | | | |
| **Output Attribute Name and Definition** | Name:  Definition: | | | | | | | | |
| **Current Logic** | | | | | | | | | |
| IBCEF11 ;ALB/TMP - FORMATTER SPECIFIC BILL FUNCTIONS - CONT ;30-JAN-96  ;;2.0;INTEGRATED BILLING;\*\*51,137,155,309,335,348,349,371,432,447,473,516,577,592\*\*;21-MAR-94;Build 25  ;;Per VA Directive 6402, this routine should not be modified.  ;         ; GETLDAT(IBXIEN) ; Extract data for 837 transmission LDAT record  ; IBXIEN - ien in file 399  ; Sets up IBXSAVE("LDAT",n) array:  ; Attachment report type ^ Attachment report transmission code ^ Attachment control number ^   ; OB Anesthesia Additional Units ^ Purchase Service Provider ID ^ Purchase Service Amount ^  N CPIEN,FTYPE,IBXDATA,IDS,IBIDS,NODE1,PSAMNT,PSPID,Z,PCE1,LINE  I '+$G(IBXIEN) Q  K IBXSAVE("LDAT")  S FTYPE=$$FT^IBCEF(IBXIEN)  ;JWS;IB\*2.0\*592; added Dental form  I FTYPE=2!(FTYPE=7) D OUTPT(IBXIEN,0)  I FTYPE=3 D HOS^IBCEF2(IBXIEN)  D ALLIDS^IBCEFP(IBXIEN,.IDS,1)  S (PSPID,PSAMNT)=""  ; IB\*2.0\*473/TAZ - Convert PROVIDER code to function call to PSID^IBCEFP  I $$SUB1OK^IBCEP8A(IBXIEN),(FTYPE=2) D  . D PSID^IBCEFP(IBXIEN,.IDS,.IBIDS)  . S PSPID=$G(IBIDS(0)) I PSPID="" S PSPID=$P($G(IBIDS(1)),U,1)  ;IB\*2.0\*473/TAZ - END  S Z=0 F  S Z=$O(IBXDATA(Z)) Q:'Z  D  . S CPIEN=+$G(IBXDATA(Z,"CPLNK")) ;I 'CPIEN Q  . I FTYPE=2,$$SUB1OK^IBCEP8A(IBXIEN) S PSAMNT=$$DOLLAR^IBCEFG1($P($G(IBXDATA(Z)),U,11))  . S (PCE1,NODE1)=""  . I CPIEN D  . . S NODE1=$G(^DGCR(399,IBXIEN,"CP",CPIEN,1))  . . S PCE1=$$GET1^DIQ(399.0304,CPIEN\_","\_IBXIEN\_",",71)  . . Q  . ; MRD;IB\*2.0\*516 - Added addl. procedure description as piece 7   . ; of IBXSAVE, which will exist only if the procedure ends in '99'  . ; or is an 'NOC/NOS' procedure.  . ;JWS;IB\*2.0\*592;do not include NOC description here for Dental claims  . S IBXSAVE("LDAT",Z)=PCE1\_U\_$P(NODE1,U,3)\_U\_$P(NODE1,U)\_U\_$P(NODE1,U,5)\_U\_$G(PSPID)\_U\_$G(PSAMNT)\_U\_$S(FTYPE=7:"",1:$P(NODE1,U,4))  . Q  Q | | | | | | | | | |
| **Modified Logic (Changes are in bold)** | | | | | | | | | |
| IBCEF11 ;ALB/TMP - FORMATTER SPECIFIC BILL FUNCTIONS - CONT ;30-JAN-96  ;;2.0;INTEGRATED BILLING;\*\*51,137,155,309,335,348,349,371,432,447,473,516,577,592,608\*\*;21-MAR-94;Build 25  ;;Per VA Directive 6402, this routine should not be modified.  ;         ; GETLDAT(IBXIEN) ; Extract data for 837 transmission LDAT record  ; IBXIEN - ien in file 399  ; Sets up IBXSAVE("LDAT",n) array:  ; Attachment report type ^ Attachment report transmission code ^ Attachment control number ^   ; OB Anesthesia Additional Units ^ Purchase Service Provider ID ^ Purchase Service Amount ^  N CPIEN,FTYPE,IBXDATA,IDS,IBIDS,NODE1,PSAMNT,PSPID,Z,PCE1,LINE  I '+$G(IBXIEN) Q  K IBXSAVE("LDAT")  S FTYPE=$$FT^IBCEF(IBXIEN)  ;JWS;IB\*2.0\*592; added Dental form  I FTYPE=2!(FTYPE=7) D OUTPT(IBXIEN,0)  I FTYPE=3 D HOS^IBCEF2(IBXIEN)  D ALLIDS^IBCEFP(IBXIEN,.IDS,1)  S (PSPID,PSAMNT)=""  ; IB\*2.0\*473/TAZ - Convert PROVIDER code to function call to PSID^IBCEFP  I $$SUB1OK^IBCEP8A(IBXIEN),(FTYPE=2) D  . D PSID^IBCEFP(IBXIEN,.IDS,.IBIDS)  . S PSPID=$G(IBIDS(0)) I PSPID="" S PSPID=$P($G(IBIDS(1)),U,1)  ;IB\*2.0\*473/TAZ - END  S Z=0 F  S Z=$O(IBXDATA(Z)) Q:'Z  D  . S CPIEN=+$G(IBXDATA(Z,"CPLNK")) ;I 'CPIEN Q  . I FTYPE=2,$$SUB1OK^IBCEP8A(IBXIEN) S PSAMNT=$$DOLLAR^IBCEFG1($P($G(IBXDATA(Z)),U,11))  . S (PCE1,NODE1)=""  . I CPIEN D  . . S NODE1=$G(^DGCR(399,IBXIEN,"CP",CPIEN,1))  . . S PCE1=$$GET1^DIQ(399.0304,CPIEN\_","\_IBXIEN\_",",71)  . . Q  . ; MRD;IB\*2.0\*516 - Added addl. procedure description as piece 7   . ; of IBXSAVE, which will exist only if the procedure ends in '99'  . ; or is an 'NOC/NOS' procedure.  . ;JWS;IB\*2.0\*592;do not include NOC description here for Dental claims  . S IBXSAVE("LDAT",Z)=PCE1\_U\_$P(NODE1,U,3)\_U\_$P(NODE1,U)\_U\_$P(NODE1,U,5)\_U\_$G(PSPID)\_U\_$G(PSAMNT)\_U\_$S(FTYPE=7:"",1:$P(NODE1,U,4))  . Q  Q   ;  ;/Beginning of IB\*2.0\*608 (US9) - vd LDAT ;  N Z K IBXDATA  S Z=0 F  S Z=$O(IBXSAVE("LDAT",Z)) Q:'Z  D  . S IBXDATA(Z)=Z D:Z>1 ID^IBCEF2(Z,"LDAT")  ;  I +$G(VC80) D  . S Z=$O(IBXDATA(""),-1)+1  . D ID^IBCEF2(Z,"LDAT")  . D VC80L(IBXIEN,Z) ; Process for 'SNF' claims & the last claim line  . S A=Z  . S IBXDATA(Z)=A  Q  ; VC80L(IBIFN,LN) ; Extracts the data for the "LDAT" record for VALUE CODE 80 Line item.  ; IBIFN = Claims internal id number  ; LN =   N VC80LN  S VC80LN=LN  S $P(IBXSAVE("LDAT",VC80LN),U,11)="6R"  S $P(IBXSAVE("LDAT",VC80LN),U,12)=VC80LN\_"\_"\_$$GET1^DIQ(399,IBIFN\_", ",.01)  Q  ;/Ending of IB\*2.0\*608 - vd  ; | | | | | | | | | |

* Include the necessary Output Formatter references in the Post-Install routine (^IBY608PR) for this build:

| **Routines** | **Activities** | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Routine Name** | IBY608PR | | | | | | | | |
| **Enhancement Category** | New | | Modify | Delete | | | No Change | | |
| **RTM** |  | | | | | | | | |
| **Related Options** | None | | | | | | | | |
| **Related Routines** | **Routines “Called By”** | | | | | **Routines “Called”** | | | |
|  |  | | | | |  | | | |
| **Data Dictionary (DD) References** | None | | | | | | | | |
| **Related Protocols** | None | | | | | | | | |
| **Related Integration Control Registrations (ICRs)** | None | | | | | | | | |
| **Data Passing** | Input | Output Reference | | | Both | | | Global Reference | Local |
| **Input Attribute Name and Definition** | Name:  Definition: | | | | | | | | |
| **Output Attribute Name and Definition** | Name:  Definition: | | | | | | | | |
| **Current Logic** | | | | | | | | | |
|  | | | | | | | | | |
| **Modified Logic (Changes are in bold)** | | | | | | | | | |
| IBY608PR ;EDE/JRA - Pre-Installation for IB patch 608 ; 10/12/17 2:12 pm  ;;2.0;INTEGRATED BILLING;\*\*608\*\*;21-MAR-94;Build 5  ;;Per VA Directive 6402, this routine should not be modified.  ;  ; delete all output formatter (O.F.) data elements included in build  ;D DELOF ;WCJ;temporarily suspend all entries from being deleted and included in the build  Q  ; INCLUDE(FILE,Y) ; function to determine if O.F. entry should be included in the build  ; FILE=5,6,7 indicating file 364.x or FILE=8 indicating file 350.8 (IB ERROR)  ; Y=ien to file  Q 0 ;WCJ;temporarily suspend all entries from being included in the build  NEW OK,LN,TAG,DATA  S OK=0  F LN=2:1 S TAG="ENT"\_FILE\_"+"\_LN,DATA=$P($T(@TAG),";;",2) Q:DATA=""  I $F(DATA,U\_Y\_U) S OK=1 Q  Q OK  ;  ;Delete edited entries to insure clean install of new entries  ;Delete obsolete entries. DELOF   ; Delete included OF entries  NEW FILE,DIK,LN,TAG,TAGLN,DATA,PCE,DA,Y  F FILE=5:1:8 S DIK=$S(FILE=8:"^IBE(350.",1:"^IBA(364.")\_FILE\_"," D  . F TAG="ENT"\_FILE,"DEL"\_FILE D  .. F LN=2:1 S TAGLN=TAG\_"+"\_LN,DATA=$P($T(@TAGLN),";;",2) Q:DATA=""  D  ... F PCE=2:1 S DA=$P(DATA,U,PCE) Q:'DA  D  .... I FILE=8,$D(^IBE(350.8,DA,0)) D ^DIK  .... Q:FILE=8  .... I $D(^IBA("364."\_FILE,DA,0)) D ^DIK  Q  ;  ; Example for ENT5, ENT6, ENT7, ENT8, DEL5, DEL6, and DEL7:  ;;^195^254^259^269^324^325^  ; Note: Must have beginning and ending up-carat  ;  ;-----------------------------------------------------------------------  ; 364.5 O.F. entries added:  ;  ; 396 N-CMN RECORD ID 'LQ '  ; ENT5 ;O.F. entries in file 364.5 to be added  ;  ;;^396^  ;  ;-----------------------------------------------------------------------  ; 364.6 O.F. entries added:  ;  ; 2383 CMN RECORD ID 'LQ '  ; 2384 CMN FORM TYPE QUALIFIER  ; 2385 SERVICE LINE #  ; 2386 CMN FORM TYPE  ; 2387 CMN INDUSTRY CODE  ; 2388 CMN CERTIFICATION TYPE  ; 2390 CMN CERTIFICATION TYPE QUAL  ; 2391 CMN PATIENT HEIGHT (IN)  ; 2392 CMN PATIENT HEIGHT MODIFIER  ; 2393 CMN PATIENT WEIGHT (LBS)  ; 2394 CMN PATIENT WEIGHT MODIFIER  ; 2395 CMN MONTHS DME EQUIPMENT NEEDED  ; 2396 CMN DATE THERAPY STARTED  ; 2397 CMN DATE THERAPY STARTED QUALIFIER  ; 2398 CMN LAST CERTIFICATION DATE  ; 2399 CMN LAST CERTIFICATION DATE QUALIFIER  ; 2400 CMN RECERTIFICATION/REVISION DATE  ; 2401 CMN REPLACEMENT ITEM?  ; 2402 CMN ABG PO2 (MMHG)  ; 2403 CMN DATE OF LAST ABG TEST  ; 2404 CMN O2 SATURATION %  ; 2405 CMN DATE LAST O2 SATURATION TEST  ; 2406 CMN EDEMA DUE TO CHF PRESENT?  ; 2407 CMN COR PULMONARY HYPERTENSION?  ; 2408 CMN HEMATOCRIT > 56%?  ; 2409 CMN PT CONDITION AT TEST TIME  ; 2410 CMN TEST CONDITIONS  ; 2411 CMN PORTABLE O2 INDICATOR  ; 2412 CMN HIGHEST O2 FLOW RATE  ; 2413 CMN LAST 4 LPM ABG PO2 (MMHG)  ; 2414 CMN DATE OF 4 LPM ABG PO2  ; 2415 CMN LAST 4 LPM O2 SATURATION %  ; 2416 CMN DT LAST 4 LPM O2 SATURATION  ; 2417 CMN EQUIPMENT/COST DESCRIPTION  ; 2418 CMN SMALL BOWEL ABSORPTION DOC?  ; 2419 CMN ENTERAL NUTRITION BY TUBE?  ; 2420 CMN PROCEDURE CALORIES  ; 2422 CMN METHOD OF ADMINISTRATION  ; 2423 CMN SEVERE MALABSORPTION DOC?  ; 2424 CMN DAYS PER WEEK ADMINISTERED  ; 2425 CMN AMINO ACID (ML/DAY)  ; 2426 CMN AMINO ACID CONCENTRATION %  ; 2427 CMN AMINO ACID PROTEIN (GM/DAY)  ; 2428 CMN DEXTROSE (ML/DAY)  ; 2429 CMN DEXTROSE CONCENTRATION %  ; 2430 CMN LIPIDS (ML/DAY)  ; 2431 CMN ROUTE OF ADMINISTRATION  ; 2433 LQ DATA EXTRACT  ; ENT6 ;O.F. entries in file 364.6 to be added  ;  ;;^2383^2384^2385^2386^2387^2388^2390^2391^2392^2393^2394^2395^2396^2397^  ;;^2398^2399^2400^2401^2402^2403^2404^2405^2406^2407^2408^2409^2410^2411^  ;;^2412^2413^2414^2415^2416^2417^2418^2419^2420^2422^2423^2424^2425^2426^  ;;^2427^2428^2429^2430^2431^2433^  ;  ;-----------------------------------------------------------------------  ; 364.7 O.F. entries added:  ;  ; 105 VC1 VALUE CODE (837 Transaction) (PC 2) (US9)  ; 176 INS SERVICE LINE COUNTER (PC 2) (US9)  ; 178 INS SERVICE UNIT COUNT (PC 5) (US9)  ; 179 INS BLANK (PC 6) (US9)  ; 180 INS SERVICE LINE CHARGE AMT (PC 9) (US9)  ; 181 INS PROCEDURE MODIFIER (1) (PC 7) (US9)  ; 189 INS PROCEDURE CODE (PC 4) (US9)  ; 478 BGN N-RECORD ID (PC1) (US9)  ; 482 INS SERVICE LINE NON-COVERED CHARGE AMT (PC 12) (US9)  ; 805 INS UNITS/BASIS FOR MEASUREMENT CODE (PC 13) (US9)  ; 1752 LDAT DATA EXTRACT (837 Transaction) (PC 1.9) (US9)  ; 1765 LDAT CLEANUP (837 Transaction) (PC 99.9) (US9)  ; 1969 CMN RECORD ID 'LQ '  ; 1970 SERVICE LINE #  ; 1971 CMN FORM TYPE QUALIFIER  ; 1972 CMN FORM TYPE  ; 1973 CMN INDUSTRY CODE  ; 1974 CMN CERTIFICATION TYPE  ; 1975 CMN CERTIFICATION TYPE QUAL  ; 1976 CMN PATIENT HEIGHT (IN)  ; 1977 CMN PATIENT HEIGHT MODIFIER  ; 1978 CMN PATIENT WEIGHT (LBS)  ; 1979 CMN PATIENT WEIGHT MODIFIER  ; 1980 CMN MONTHS DME EQUIPMENT NEEDED  ; 1981 CMN DATE THERAPY STARTED  ; 1982 CMN DATE THERAPY STARTED QUALIFIER  ; 1983 CMN LAST CERTIFICATION DATE  ; 1984 CMN LAST CERTIFICATION DATE QUALIFIER  ; 1985 CMN RECERTIFICATION/REVISION DATE  ; 1986 CMN REPLACEMENT ITEM?  ; 1987 CMN ABG PO2 (MMHG)  ; 1988 CMN DATE OF LAST ABG TEST  ; 1989 CMN O2 SATURATION %  ; 1990 CMN DATE LAST O2 SATURATION TEST  ; 1991 CMN EDEMA DUE TO CHF PRESENT?  ; 1992 CMN COR PULMONARY HYPERTENSION?  ; 1993 CMN HEMATOCRIT > 56%?  ; 1994 CMN PT CONDITION AT TEST TIME  ; 1995 CMN TEST CONDITIONS  ; 1996 CMN PORTABLE O2 INDICATOR  ; 1997 CMN HIGHEST O2 FLOW RATE  ; 1998 CMN LAST 4 LPM ABG PO2 (MMHG)  ; 1999 CMN DT LAST 4 LPM O2 SATURATION  ; 2000 CMN EQUIPMENT/COST DESCRIPTION  ; 2001 CMN SMALL BOWEL ABSORPTION DOC?  ; 2002 CMN ENTERAL NUTRITION BY TUBE?  ; 2003 CMN PROCEDURE CALORIES  ; 2005 CMN METHOD OF ADMINISTRATION  ; 2006 CMN SEVERE MALABSORPTION DOC?  ; 2007 CMN DAYS PER WEEK ADMINISTERED  ; 2008 CMN AMINO ACID (ML/DAY)  ; 2009 CMN AMINO ACID CONCENTRATION %  ; 2010 CMN AMINO ACID PROTEIN (GM/DAY)  ; 2011 CMN DEXTROSE (ML/DAY)  ; 2012 CMN DEXTROSE CONCENTRATION %  ; 2013 CMN LIPIDS (ML/DAY)  ; 2014 CMN ROUTE OF ADMINISTRATION  ; 2016 CMN DATE OF 4 LPM ABG PO2  ; 2017 CMN LAST 4 LPM O2 SATURATION %  ; 2018 LQ DATA EXTRACT  ; ENT7 ; O.F. entries in file 364.7 to be added  ;  ;;^105^176^178^179^180^181^189^478^482^805^1752^1765^  ;;^1969^1970^1971^1972^1973^1974^1975^1976^1977^1978^1979^1980^  ;;^1981^1982^1983^1984^1985^1986^1987^1988^1989^1990^1991^1992^  ;;^1993^1994^1995^1996^1997^1998^1999^2000^2001^2002^2003^2005^  ;;^2006^2007^2008^2009^2010^2011^2012^2013^2014^2016^2017^2018^  ;  ;-----------------------------------------------------------------------  ; 350.8 O.F. entries added:  ;  ; 238 IB CMN REQ  ; 239 IB CMN NOT REQ BUT DATA  ; 240 IB CMN FORM TYPE  ; 241 IB CMN NO DATA NODE  ; 243 IB CMN BAD DATA NODE  ; 244 IB CMN CERT TYPE  ; 245 IB CMN PT HEIGHT  ; 246 IB CMN THERAPY DT  ; 247 IB CMN LAST CERT DT  ; 248 IB CMN RECERT/REVISION DT  ; 250 IB CMN LAST ABG PO2  ; 251 IB CMN LAST ABG PO2 DT  ; 259 IB CMN O2 SAT DT  ; 261 IB CMN 4 LPM ABG DT  ; 271 IB CMN 4 LPM O2 SAT % DT  ; 272 IB CMN ERRORS HEADER  ; ENT8 ;O.F. entries in file 350.8 to be added  ;  ;;^238^239^240^241^243^244^245^246^247^248^250^251^259^261^271^272^  ;  ;-----------------------------------------------------------------------  ; 364.5 entries deleted:  ; DEL5    ; remove O.F. entries in file 364.5 (not re-added)  ;  ;;  ;  ;-----------------------------------------------------------------------  ; 364.6 entries deleted:  ;  ; DEL6    ; remove O.F. entries in file 364.6 (not re-added)  ;  ;;  ;  ;-----------------------------------------------------------------------  ; 364.7 entries deleted:  ;  ; DEL7    ; remove O.F. entries in file 364.7 (not re-added)  ;  ;;  ;  ;----------------------------------------------------------------------- | | | | | | | | | |