

Department of Veterans Affairs

EDI New Standards and Operating Rules VHA Provider-side

Technical Compliance Requirements

Patch IB*2.0*488

Patch PRCA*4.5*300

System Design Document



February 2014

Version 3.0

Revision History

Note: The revision history cycle begins once changes or enhancements are requested after the System Design Document has been baselined.

| Date | Version | Description | Author |
|------------|---------|---|---------------------------|
| 12/20/2013 | 1.0 | Initial | eBilling Development Team |
| 02/03/2014 | 2.0 | Modified version to address CBO's comments. | eBilling Development Team |
| 02/18/2014 | 3.0 | Made a correction to what the code would look like for the change to the Accounts Receivable Bill No. (based on Debbie Brun's comment). | eBilling Development Team |

Artifact Rationale

The System Design Document (SDD) is a dual-use document that provides the conceptual design as well as the as-built design. This document will be updated as the product is built, to reflect the as-built product. Per the Project Management Accountability System (PMAS) Guide, the SDD with conceptual design is required prior to the Milestone 1 Review. The as-built for each delivery must be incorporated prior to the Milestone 2 Review.

Table of Contents

| | |
|---|----------|
| 1. Introduction | 1 |
| 1.1. Purpose of this Document | 1 |
| 1.2. Identification | 1 |
| 1.3. Scope | 1 |
| 1.4. Relationship to Other Plans | 3 |
| 1.5. Methodology, Tools, and Techniques | 4 |
| 1.6. Constraining Policies, Directives and Procedures | 4 |
| 1.7. Constraints | 4 |
| 1.8. Design Trade-offs | 4 |
| 1.9. User Characteristics | 5 |
| 1.10. User Problem Statement | 5 |
| 2. Background | 5 |
| 2.1. Overview of the System | 5 |
| 2.2. Overview of the Business Process | 5 |
| 2.3. Assumptions | 6 |
| 2.4. Legacy System Retirement | 6 |
| 3. Conceptual Design | 6 |
| 3.1. Conceptual Application Design | 6 |
| 3.1.1. Application Context | 6 |
| 3.1.2. High-Level Application Design | 6 |
| 3.1.3. Application Locations | 6 |
| 3.1.4. Application Users | 6 |
| 3.2. Conceptual Data Design | 6 |
| 3.2.1. Project Conceptual Data Model | 7 |
| 3.2.2. Database Information | 7 |
| 3.2.3. User Interface Data Mapping | 8 |
| 3.3. Conceptual Infrastructure Design | 8 |
| 3.3.1. System Criticality and High Availability | 8 |
| 3.3.2. Special Technology | 8 |
| 3.3.3. Technology Locations | 9 |
| 3.3.4. Conceptual Infrastructure Diagram | 9 |
| 4. System Architecture | 9 |
| 4.1. Hardware Architecture | 9 |
| 4.2. Software Architecture | 9 |

| | |
|---|------------|
| 4.3. Communications Architecture..... | 9 |
| 5. Data Design | 9 |
| 5.1. DBMS Files | 9 |
| 5.2. Non-DBMS Files | 10 |
| 6. Detailed Design | 10 |
| 6.1. Hardware Detailed Design..... | 10 |
| 6.2. Software Detailed Design..... | 10 |
| 6.2.1. Conceptual Design | 10 |
| 6.2.2. Specific Requirements | 12 |
| 7. External Interface Design | 153 |
| 7.1. Interface Architecture..... | 154 |
| 7.2. Interface Detailed Design | 154 |
| 8. Human-Machine Interface | 154 |
| 8.1. Interface Design Rules | 154 |
| 8.2. Inputs | 154 |
| 8.3. Outputs | 154 |
| 8.4. Navigation Hierarchy | 154 |
| 8.4.1. Screen [x.1] | 154 |
| 8.4.2. Screen [x.2] | 154 |
| 8.4.3. Screen [x.3] | 155 |
| 9. System Integrity Controls | 155 |
| 10. Approval Signatures..... | 156 |
| A. Additional Information..... | 158 |
| A.1. RTM..... | 158 |
| A.2. Packaging and Installation..... | 158 |
| A.3. Design Metrics | 158 |
| A.4. Acronym List and Glossary | 158 |
| A-4.1 Acronyms..... | 158 |
| A-4.2 Definitions..... | 159 |
| A-4.3 References..... | 160 |

1. Introduction

The purpose of this Software Design Document (SDD) is to describe the changes needed to fulfill the requirements for Electronic Data Interchange (EDI) New Standards and Operating Rules VHA Provider-side projects related to the Veterans Health Information Systems and Technology Architecture (VistA) Integrated Billing (IB) version 2.0 and Accounts Receivable (AR) version 4.5 applications specifically IB Patch IB*2*488 and AR Patch PRCA*4.5*300.

The target audience for this SDD includes Office of Enterprise Development (OED), Product Development (PD), Product Support (PS), Software Quality Assurance (SQA), the Program Management Office (PMO), the Chief Business Office (CBO) Office and the software end-users.

1.1. Purpose of this Document

The purpose of this document is to describe in sufficient detail how the proposed system is to be constructed. The SDD translates the requirement specifications into a document from which the developers can create the actual system. It identifies the top-level system architecture, and identifies hardware, software, communication, and interface components.

1.2. Identification

The software that this SDD applies to are VistA's Integrated Billing version 2.0 and Accounts Receivable version 4.5 applications. It will use the following standards:

- American National Standards Institute (ANSI)
- International Organization for Standardization (ISO)
- Veterans Administration Standards and Conventions (VA SAC)

Refer to Appendix A for the standard Acronyms (Appendix A 4.1), Definitions (Appendix A 4.2) and References (Appendix A 4.3).

1.3. Scope

The section describes the project scope which includes changes to VistA's Integrated Billing (IB) and Accounts Receivable (AR) modules.

Table 1: Integrated Billing (IB*2.0*488) Inclusions

| Includes |
|----------|
|----------|

| Includes |
|--|
| <p>Enter/Edit Billing Information:</p> <ul style="list-style-type: none"> • Remove the ability to force a claim to be printed at the Health Care Clearing House (HCCH) for institutional claims • Remove the ability to force a claim to be printed at the Health Care Clearing House (HCCH) for professional claims • Provide a Fatal Warning message to the user when EDI – Inst Payer Primary ID on an institutional claim equals HPRNT/SPRNT • Provide a Fatal Warning message to the user when EDI – Inst Payer Primary ID on a professional claim equals SPRNT/HPRNT • Provide the ability to prevent the authorization of a professional claim with no procedures • Provide the ability to prevent the authorization of an outpatient, institutional claim with no procedures • Provide the ability to display the Coordination of Benefits (COB) sequence with the patient's insurance plans when available when a partial entry is entered for the insurance policy. • Remove the Line Level EPSDT Indicator for professional claims • Remove the Line Level Attending Physician is not a Hospice Employee for professional claims |
| <p>Provider Maintenance:</p> <ul style="list-style-type: none"> • Provide the ability to enter only a 9 digit ZIP, with the optional entry of a hyphen, for a non-VA facility address • Provide the ability to enter only a physical location in Address Line 1 for a non-VA facility |
| <p>Insurance Company Editor:</p> <ul style="list-style-type: none"> • Remove the ability to set the EDI – Inst Payer Primary ID equal to HPRNT/IPRNT/SPRNT/PPRNT • Remove the ability to set the EDI – Prof Payer Primary ID equal to SPRNT/PPRNT/IPRNT/HPRNT • Provide the functionality to set the Default value to EDI – Transmit?: to YES-LIVE when a new Insurance Company is created |
| <p>MRA Management Worklist (MRW):</p> <ul style="list-style-type: none"> • Provide the ability to view Message Storage Errors for inbound Medicare-equivalent MRAs in the MRW in a manner an end-user can understand |

| Includes |
|--|
| <p>Transactions:</p> <ul style="list-style-type: none"> • Provide the ability to transmit up to 12 diagnoses on a Professional 837 claim transaction • Prevent the transmission of the following Property and Casualty data in the Service Facility loop of an institutional/professional 837 claim transaction: <ul style="list-style-type: none"> ○ P&C Contact Name ○ P&C Contact Telephone Number ○ P&C Contact Telephone Number Extension • Provide the ability to transmit a Service Line Charge Amt (INS, Piece 9) with a maximum length equal to 18 numeric in an institutional 837 claim transaction. • Provide the ability to transmit a Service Line Non-Covered Charge Amt (INS, Piece 12) with a maximum length equal to 18 numeric in an Institutional 837 claim transaction • Provide the ability to transmit institutional claims with the Rate Type = Worker's Comp. • Provide the ability to transmit professional claims with the Rate Type = Worker's Comp. • Prevent the transmission of institutional/professional 837 claim transactions with an Assignment Code (CL1A, Piece 5) equal to 'C' or null by substituting the value 'A' • Provide the ability to transmit institutional/professional claims with Line Item Charge Amounts equal to \$0.00 • Provide the ability to transmit diagnosis pointers corresponding with diagnoses on professional 837 transactions |
| <p>Third Party Joint Inquiry (TPJI):</p> <ul style="list-style-type: none"> • Display message storage error messages generated when VistA is unable to upload inbound X12N 5010 Health Care Claim Payment/Advice (835) messages in a manner an end-user can understand |
| <p>CMS – 1500 Form:</p> <ul style="list-style-type: none"> • Modify the print logic for the CMS – 1500 form to comply with the new National Uniform Claim Committee (NUCC) standards |
| Excludes |
| <p>Enter/Edit Billing Information:</p> <ul style="list-style-type: none"> • Provide the ability to display the Coordination of Benefits (COB) sequence with the patient's insurance plans when available when ?? is entered for Help |

Table 2: Accounts Receivable (PRCA*4.5*300) Inclusions

| Includes |
|---|
| <p>Data Dictionary:</p> <ul style="list-style-type: none"> • Provide the ability to define a claim number as 6-10 characters |

1.4. Relationship to Other Plans

This SDD contains relationships between VistA's Integrated Billing (IB) and Accounts Receivable (AR) modules specifically the Data Dictionary and the enhancing of the Claim Number to be 6-10 characters in length. Beyond that, there are no reliability requirements specific to this development effort. The IB and AR modules are integrated parts of the overall VistA system that exists at each site and will be subject to the normal reliability standards.

1.5. Methodology, Tools, and Techniques

The methodologies, tools and techniques that are to be used in this development effort include the following:

- Cache Cube
- MUMPS (M)
- Kernel Installation and Distribution System (KIDS)
- FileMan
- MailMan
- ScreenMan
- ListMan
- 837 Health Care Claim: Professional ASC X12 Standards for Electronic Data Exchange Technical Report Type 3 – May 2006
- 837 Health Care Claim: Institutional ASC X12 Standards for Electronic Data Exchange Technical Report Type 3 – May 2006
- NUCC, 1500 Health Insurance Claim Form Reference Instruction Manual for Form Version 02/12

1.6. Constraining Policies, Directives and Procedures

The constraining policies, directives and procedures that are to be followed by this development effort include the Veteran Administration's Standards and Conventions (VA SAC).

1.7. Constraints

The constraints that exist for this development effort include the following:

- The software language of MUMPS (M)
- PMAS project management methodology
- ProPath

1.8. Design Trade-offs

This section is not applicable to this SDD.

1.9. User Characteristics

The IB and AR modules are designed to be used by Billing Supervisors, Billing Clerks, Accounts Receivable Supervisors and Accounts Receivable Clerks as well as Utilization Review and Insurance Verification personnel.

1.10. User Problem Statement

The following are a list of the User Problems/Concerns that this development effort is addressing:

- The CMS-1500 Claim Form requires multiple modifications and enhancements implemented by April 1, 2014.
- The Message Storage Errors (MSEs) need to be made more legible and understandable on the MRA Management Worklist (MRW).
- The Health Care Claim Transactions (837) needs multiple modifications to it.

2. Background

2.1. Overview of the System

The IB and AR modules are designed to be used by Billing Supervisors, Billing Clerks, Accounts Receivable Supervisors and Accounts Receivable Clerks as well as Utilization Review and Insurance Verification personnel.

2.2. Overview of the Business Process

The following features of the IB and AR modules will be affected by this development effort:

For Integrated Billing

- Enter/Edit Billing Information
- Provider Maintenance
- Insurance Company Enter/Edit
- MRA Management Worklist
- Third Party Joint Inquiry (TPJI)
- CMS-1500 Claim Form
- X12N 5010 Health Care Claim (837)

For Accounts Receivable:

- Data Dictionary

2.3. Assumptions

This section is not applicable to this SDD.

2.4. Legacy System Retirement

This section is not applicable to this SDD.

3. Conceptual Design

This section is not applicable to this SDD.

3.1. Conceptual Application Design

This section is not applicable to this SDD.

3.1.1. Application Context

This section is not applicable to this SDD.

3.1.2. High-Level Application Design

This section is not applicable to this SDD.

3.1.3. Application Locations

This section is not applicable to this SDD.

3.1.4. Application Users

This section is not applicable to this SDD.

3.2. Conceptual Data Design

3.2.1. Project Conceptual Data Model

The Conceptual Data Model is found in the [Entity Relationship Diagrams \(ERD\)](#).

3.2.2. Database Information

3.2.2.1. Accounts Receivable Bill Number Length

| | |
|---|---|
| File Name and Number | ACCOUNTS RECEIVABLE (#430) |
| Enhancement Category | <input type="checkbox"/> New <input checked="" type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change |
| Requirements Traceability Matrix | 2.6.2.1 |
| Related Options | |
| Data Dictionary (DD) References | |
| Related Protocols | |
| Related Integration Control Registrations (ICRs) Agreements | |
| File Documentation | |
| File Auditing, Security, and Archiving | |
| Field Name | BILL NO. |
| Field Description | ACCOUNTS RECEIVABLE BILL NUMBER |
| Field # | .01 |
| Node # | ^PRCA(430,0) |
| Piece # | 1 |
| New Field | <input type="checkbox"/> Yes <input checked="" type="checkbox"/> No |
| Data Type | <input type="checkbox"/> Date/Time <input type="checkbox"/> Numeric <input type="checkbox"/> Set of Codes <input checked="" type="checkbox"/> Free Text <input type="checkbox"/> Pointer to a File <input type="checkbox"/> Variable-Pointer |
| Identifier | <input type="checkbox"/> Yes <input checked="" type="checkbox"/> No |
| Uneditable Field | <input type="checkbox"/> Yes <input checked="" type="checkbox"/> No |
| Mandatory Field | <input type="checkbox"/> Yes <input checked="" type="checkbox"/> No |
| Field Documentation or Help Changes Necessary | <input checked="" type="checkbox"/> Yes <input type="checkbox"/> No |
| Field Definition | Bill Number |
| Input/Output Transform | CURRENT: K:X["!!!!"](\$A(X)=45) X I \$D(X) K:\$P(X,"-",1)"?3N!(\$P(X,"-",2)?7UN) X I \$D(X) K:\$L(X)>14!(\$L(X)<11) X TO BE CHANGED TO: K:X["!!!!"](\$A(X)=45) X I \$D(X) K:\$P(X,"-",1)"?3N!(\$P(X,"-",2)?6UN.1UN) X I \$D(X) K:\$L(X)>14!(\$L(X)<10) X |

| | | | | |
|-------------------------------|----------------------------------|----------------------------------|-----------------------------------|--------------------------------|
| Cross-Reference (id and type) | <input type="checkbox"/> Regular | <input type="checkbox"/> Kwic | <input type="checkbox"/> Mnemonic | <input type="checkbox"/> Mumps |
| No cross reference | <input type="checkbox"/> Soundex | <input type="checkbox"/> Trigger | <input type="checkbox"/> Bulletin | |

3.2.3. User Interface Data Mapping

This section is not applicable to this SDD.

3.2.3.1. Application Screen Interface

This section is not applicable to this SDD.

3.2.3.1.1. <Insert name of screen>

This section is not applicable to this SDD.

3.2.3.2. Application Report Interface

This section is not applicable to this SDD.

3.2.3.2.1. <Insert name of report>

This section is not applicable to this SDD.

3.2.3.3. Unmapped Data Element

This section is not applicable to this SDD.

3.3. Conceptual Infrastructure Design

This section is not applicable to this SDD.

3.3.1. System Criticality and High Availability

This section is not applicable to this SDD.

3.3.2. Special Technology

This section is not applicable to this SDD.

3.3.3. Technology Locations

This section is not applicable to this SDD.

3.3.4. Conceptual Infrastructure Diagram

This section is not applicable to this SDD.

3.3.4.1. Location of Environments and External Interfaces

This section is not applicable to this SDD.

3.3.4.2. Conceptual Production String Diagram

This section is not applicable to this SDD.

4. System Architecture

This section is not applicable to this SDD.

4.1. Hardware Architecture

This section is not applicable to this SDD.

4.2. Software Architecture

This section is not applicable to this SDD.

4.3. Communications Architecture

This section is not applicable to this SDD.

5. Data Design

This section is not applicable to this SDD.

5.1. DBMS Files

This section is not applicable to this SDD.

5.2. Non-DBMS Files

This section is not applicable to this SDD.

6. Detailed Design

This section is not applicable to this SDD.

6.1. Hardware Detailed Design

This section is not applicable to this SDD.

6.2. Software Detailed Design

This section is not applicable to this SDD.

6.2.1. Conceptual Design

This section is not applicable to this SDD.

6.2.1.1. Product Perspective

This section is not applicable to this SDD.

6.2.1.1.1. User Interfaces

Users of VistA use terminal emulation software to access VistA as if they were using a VT320/400/500 terminal. The VistA user interface is a two color, roll and scroll interface developed in M.

6.2.1.1.2. Hardware Interfaces

This section is not applicable to this SDD.

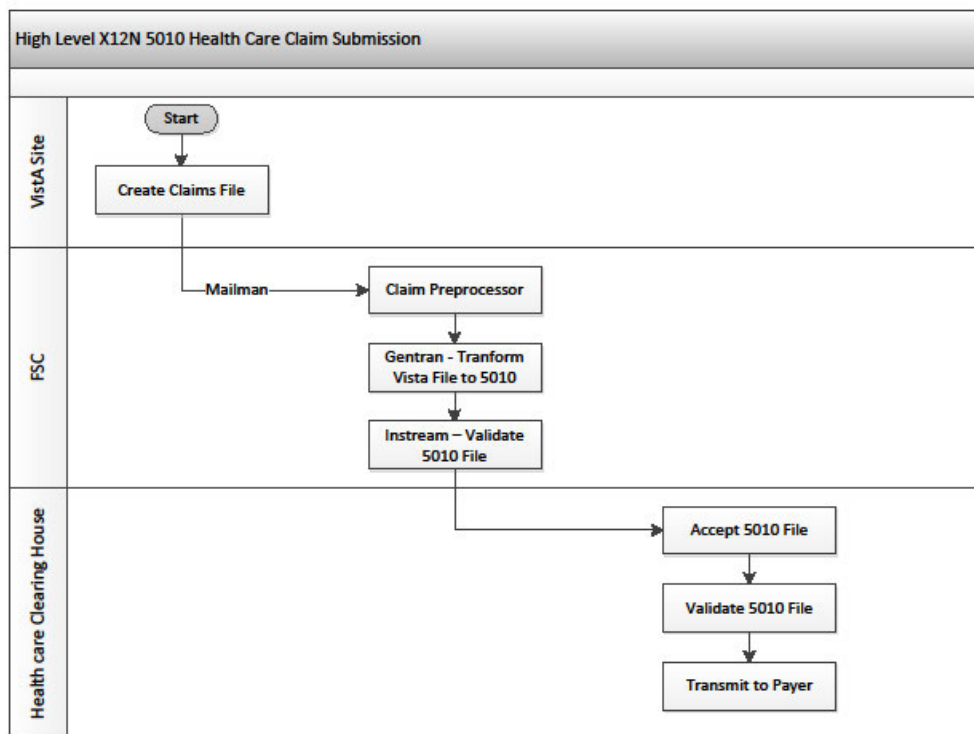
6.2.1.1.3. Software Interfaces

The IB module communicates with other VistA modules such as Accounts Receivable, Appointment Scheduling, Admission/Discharge/Transfer, Claims Tracking, and Charge Master. These are existing interfaces.

The AR module communicates with other VistA modules such as IB and IFCAP, as well as external systems such as the Financial Management System (FMS).

The Claim Scrubber, a 3rd party piece of software developed by DSS, Inc. which uses the 837 message format to communicate between Enter/Edit Billing Information and the scrubber. DSS will need to make changes to correspond to the changes made to the 837 map as part of this project.

6.2.1.1.4. Communications Interfaces



6.2.1.1.5. Memory Constraints

This section is not applicable to this SDD.

6.2.1.1.6. Special Operations

This section is not applicable to this SDD.

6.2.1.2. Product Features

This section is not applicable to this SDD.

6.2.1.3. User Characteristics

The IB and AR software is designed to be used by Billing Supervisors, Billing Clerks, Accounts Receivable Supervisors and Accounts Receivable Clerks, as well as Utilization Review and Insurance Verification personnel.

6.2.1.4. Dependencies and Constraints

This section is not applicable to this SDD.

6.2.2. Specific Requirements

For a high level summary of the features and modifications that will be made to the eBilling module as part of this project, refer to Table 1 (Scope Inclusions) of Section 1.3 Scope.

6.2.2.1. System Feature: Enter/Edit Billing Information

This section covers the design for the Functional Requirements listed in Section 2.6.1 of the RSD.

6.2.2.1.1. Functional Requirements: Enter/Edit Billing Information

| SRS REQ ID | REQ Title | Comments / Notes |
|-----------------------|--|-------------------------|
| 2.6.1.1 | Remove Force Print at HCCH – Institutional | |
| 2.6.1.2 | Remove Force Print at HCCH – Professional | |
| 2.6.1.3 | Fatal Error for HPRNT/SPRNT Values – Institutional | |
| 2.6.1.4 | Fatal Error for SPRNT/HPRNT Values - Professional | |
| 2.6.1.5 | Fatal Error for No Procedures – Professional | |
| 2.6.1.6 | Fatal Error for No Procedures – Institutional | |
| 2.6.1.7 | Display Payer Sequence – Partial Entry | |
| 2.6.1.8 | Remove Line Level EPSDT Indicator – Professional | |
| 2.6.1.9 | Remove Line Level Attending Physician is not a Hospice Employee – Professional | |

6.2.2.1.2. Routines (Entry Points)

| | | | |
|---|--|---|--|
| Routine Name | IBCBB1 | | |
| Enhancement Category | <input type="checkbox"/> New <input checked="" type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change | | |
| Requirement Traceability Matrix | 2.6.1.1, 2.6.1.2, 2.6.1.3, 2.6.1.4, 2.6.1.5, 2.6.1.6 | | |
| Related Options | Enter/Edit Billing Information [IB EDIT BILLING INFO] | | |
| Related Routines | Routines "Called By" | Routines "Called" | |
| | ^IBACUS2, ^IBCA3, ^IBCB1, ^IBCB2, ^IBCBB, ^IBCONS1, ^IBNCPDP2, ^IBNCPDP5, ^IBOA31 | ^IBOUTL, ^IBEFUNC, ^PRCAFN, ^IBCEF74, ^IBCEF, ^DILFD, ^IBCBB1, ^IBCEF, ^IBCEP2A, ^IBCRU3, ^IBCBB12, ^IBCEF1, ^IBCBB3, ^DIQ, ^IBCBB13, ^IBCEF4, ^IBCEMU2 | |
| Data Dictionary (DD) References | BILL/CLAIMS File (#399) | | |
| Related Protocols | | | |
| Related Integration Control Registrations (ICRs) | | | |
| Data Passing | <input type="checkbox"/> Input <input type="checkbox"/> Output Reference <input checked="" type="checkbox"/> Both <input type="checkbox"/> Global Reference <input type="checkbox"/> Local | | |
| Input Attribute Name and Definition | | | |
| Output Attribute Name and Definition | | | |
| Current Logic | | | |
| <pre> IBCBB1 ;ALB/AAS - CONTINUATION OF EDIT CHECK ROUTINE ;2-NOV-89 ;;2.0;INTEGRATED BILLING; **27,52,80,93,106,51,151,148,153,137,232,280,155,320,343,349,363,371,395,384,432,447** ;21 -MAR-94;Build 80 ;;Per VHA Directive 2004-038, this routine should not be modified. ; ;MAP TO DGCRBB1 ; % ;Bill Status N Z,Z0,Z1,IBFT I \$S(+IBST=0:1,1:"^1^2^3^4^7^"[(U_IBST_U)) S IBER=IBER_"IB045;" ; ;Statement Covers From I IBFDT="" S IBER=IBER_"IB061;" I IBFDTJ"" ,IBFDT'?7N&(IBFDT'?7N1"." .N) S IBER=IBER_"IB061;" I IBFDT>IBTDT S IBER=IBER_"IB061;" ; from must be on or before the to date S IBFFY=\$\$FY^IBOUTL(IBFDT) ; if input - from date must not be prior to admit date. I \$\$\$INPAT^IBCEF(IBIFN,1),(IBFDT<(\$P(\$G(^DGPT(+P(IBND0,U,8),0)),U,2)\1)) S IBER=IBER_"IB061;" </pre> | | | |

| Routine Name | IBCBB1 |
|--------------|---|
| | <pre> ; ;Statement Covers To I IBTDT="" S IBER=IBER_"IB062;" I IBTDT]",IBTDT'?7N&(IBTDT'?7N1".N) S IBER=IBER_"IB062;" I IBTDT>DT!(IBTDT<IBFDT) S IBER=IBER_"IB062;" ; to date must not be >than today's date S IBTFY=\$\$FY^IBOUTL(IBTDT) ; ;Total Charges ; IB*2.0*447/TAZ Removed this error so that zero dollar revenue codes can process on the 837 ;I +IBTC'>0!(+IBTC'=IBTC) S IBER=IBER_"IB064;" ; ;Billable charges for secondary claim I \$\$MCRONBIL^IBEFUNC(IBIFN)&((\$P(IBNDU1,U,1)-\$P(IBNDU1,U,2))>0) S IBER=IBER_"IB094;" ;Fiscal Year 1 S IBFFY=\$\$FY^IBOUTL(IBFDT) ; ;Check provider link for current user, enterer, reviewer and Authorizor I '\$D(^VA(200,DUZ,0)) S IBER=IBER_"IB048;" I IBEU]",'\$D(^VA(200,IBEU,0)) S IBER=IBER_"IB048;" I IBRU]",'\$D(^VA(200,IBRU,0)) S IBER=IBER_"IB060;" I IBAU]",'\$D(^VA(200,IBAU,0)) S IBER=IBER_"IB041;" ; I IBER="",+\$\$STA^PRCAFN(IBIFN)=104 S IBER=IBER_"IB040;" ; If ins bill, must have valid COB sequence I \$P(IBND0,U,11)="I",\$S(\$P(IBND0,U,21))="":1,1:"PST"[\$P(IBND0,U,21)) S IBER=IBER_"IB324;" ; ; Check for valid sec provider id for current ins S Z=0 F S Z=\$O(^DGCR(399,IBIFN,"PRV",Z)) Q:Z S Z0=\$G(^Z(0)),Z1=+\$\$COBN^IBCEF(IBIFN) I \$P(Z0,U,4+Z1)="", \$P(Z0,U,11+Z1)="" D . I '\$\$SECIDCK^IBCEF74(IBIFN,Z1,\$P(Z0,U,11+Z1),Z) D WARN^IBCBB11("Prov secondary id type for the "_\$P("PRIMARY^SECONDARY^TERTIARY",U,Z1)_" "\$\$EXTERNAL^DILFD(399.0222,.01,+,+Z0)_" is invalid/won't transmit") ; Check NPIs D NPICHK^IBCBB11 ; ; Check multiple rx NPIs D RXNPI^IBCBB11(IBIFN) ; ; Check taxonomies D TAXCHK^IBCBB11 ; ; Check for Physician Name K IBXDATA D F^IBCEF("N-ATT/REND PHYSICIAN NAME",,,,IBIFN) ; IB*2.0*432 - CMS1500 no longer needs a claim level rendering S IBFT=\$\$FT^IBCEF(IBIFN) I IBFT=2,\$P(\$G(IBXDATA),U)="" S IBER=IBER_"IB303;" ; N FUNCTION,IBINS ; IB*2.0*432 - CMS1500 no longer needs a claim level rendering ;S FUNCTION=\$S(\$\$FT^IBCEF(IBIFN)=3:4,1:3) </pre> |

| Routine Name | IBCBB1 |
|--------------|---|
| | <pre> S FUNCTION=\$S(IBFT=3:4,1:3) I IBFT'=2,IBER["IB303;" D . F IBINS=1:1:3 D .. S Z=\$\$GETTYP^IBCEP2A(IBIFN,IBINS) .. I Z,\$P(Z,U,2) D ; Rendering/attending prov secondary id required ... N IBID,IBOK,Q0 ... D PROVINF^IBCEF74(IBIFN,IBINS,.IBID,1,"C") ; check all as though they were current ... S IBOK=0 ... S Q0=0 F S Q0=\$O(IBID(1,FUNCTION,Q0)) Q:'Q0 I \$P(IBID(1,FUNCTION,Q0),U,9)=+Z S IBOK=1 Q ... I 'IBOK S IBER=IBER_\$S(IBINS=1:"IB236;",IBINS=2:"IB237;",IBINS=3:"IB238;",1:"") ; ; Patch 432 enh5:The IB system shall no longer prevent users from authorizing(fatal error message)a claim because the system cannot find the providersSSNorEIN ; D PRIIDCHK^IBCBB11 ; N IBM,IBM1 S IBM=\$G(^DGCR(399,IBIFN,"M")) S IBM1=\$G(^DGCR(399,IBIFN,"M1")) I \$P(IBM,U),\$P(\$G(^DIC(36,\$P(IBM,U,4)),U,6),\$P(IBM1,U,2))="" S IBER=IBER_"IB244;" I \$P(IBM,U,2),\$P(\$G(^DIC(36,\$P(IBM,U,2,4)),U,6),\$P(IBM1,U,3))="" S IBER=IBER_"IB245;" I \$P(IBM,U,3),\$P(\$G(^DIC(36,\$P(IBM,U,3,4)),U,6),\$P(IBM1,U,4))="" S IBER=IBER_"IB246;" ; ; If outside facility, check for ID and qualifier in 355.93 ; 5/15/06 - esg - hard error IB243 turned into warning message instead S Z=\$P(\$G(^DGCR(399,IBIFN,"U2")),U,10) I Z D . I \$P(\$G(^IBA(355.93,Z,0)),U,9)=""!(\$P(\$G(^IBA(355.93,Z,0)),U,13)="") D .. N Z1,Z2 .. S Z1="Missing Lab or Facility Primary ID for non-VA facility, " .. S Z2=\$\$EXTERNAL^DILFD(399,232,,Z) .. I \$L(Z2)'>19 D WARN^IBCBB11(Z1_Z2) Q .. D WARN^IBCBB11(Z1),WARN^IBCBB11(" "_Z2) .. Q . Q ; ; Must be one and only one division on bill S IBZ=\$\$MULTDIV^IBCBB11(IBIFN,IBND0) ; I IBZ S IBER=IBER_\$S(IBZ=1:"IB095;",IBZ=2:"IB104;",1:"IB105;") ; Allow multi-divisional for OP insttutional claims I IBZ,\$\$INPAT^IBCEF(IBIFN)!(\$\$INSPRF^IBCEF(IBIFN)) S IBER=IBER_\$S(IBZ=1:"IB095;",IBZ=2:"IB104;",1:"IB105;") ; Still need error msg on OP Institutional if No Default division I IBZ=3,\$\$INPAT^IBCEF(IBIFN),\$INSPRF^IBCEF(IBIFN) S IBER=IBER_"IB105;" ; Division address must be defined in institution file I \$P(IBND0,U,22) D . N Z,Z0,Z1 . S Z0=\$G(^DIC(4,+P(\$G(^DG(40.8,+P(IBMND0,U,22),0)),U,7),0)) . S Z1=\$G(^DIC(4,+P(\$G(^DG(40.8,+P(IBMND0,U,22),0)),U,7),1)) . I \$P(Z0,U,2)="" S IBER=IBER_"IB097;" Q . F Z=1,3,4 I \$P(Z1,U,Z)="" S IBER=IBER_"IB097;" Q </pre> |

| Routine Name | IBCBB1 |
|--------------|--|
| | <pre> ; ; IB*2.0*432 Check ambulance addresses, COB Non-covered amt. & Attachment Control I \$\$AMBCK^IBCBB11(IBIFN)=1 S IBER=IBER_"IB329;" I \$\$COBAMT^IBCBB11(IBIFN)=1 S IBER=IBER_"IB330;" I \$\$TMCK^IBCBB11(IBIFN)=1 S IBER=IBER_"IB331;" I \$\$ACCK^IBCBB11(IBIFN)=1 S IBER=IBER_"IB332;" I \$\$COBMRA^IBCBB11(IBIFN)=1 S IBER=IBER_"IB342;" I \$\$COBSEC^IBCBB11(IBIFN)=1 S IBER=IBER_"IB343;" ; ;CHAMPVA Rate Type and Primary Insurance Carriers Type of Coverage must match S (IBRTCHV,IBPICHV)=0 I \$P(\$G(^DGCR(399.3,+IBAT,0)),U,1)="CHAMPVA" S IBRTCHV=1 I \$P(\$G(^IBE(355.2,\$P(\$G(^DIC(36,+IBNDMP,0)),U,13),0)),U,1)="CHAMPVA" S IBPICHV=1 I (+IBRTCHV!+IBPICHV)&('IBRTCHV!'IBPICHV) S IBER=IBER_"IB085;" ; N IBZPRC,IBZPRCUB D F^IBCEF("N-ALL PROCEDURES","IBZPRC",,IBIFN) ; Procedure Clinic is required for Surgical Procedures Outpt Facility Charges I +\$P(IBND0,U,27)=2,\$\$BILLRATE^IBCRU3(IBAT,IBCL,IBEVD,"RC OUTPATIENT") D . N Z,Z0,Z1,ZE S (ZE,Z)=0 F S Z=\$O(^DGCR(399,IBIFN,"CP",Z)) Q:'Z D I +ZE S IBER=IBER_"IB320;" Q .. S Z0=\$G(^DGCR(399,IBIFN,"CP",Z,0)),Z1=+Z0 I Z0['";ICPT(" Q .. I '((Z1<10000)&(Z1>69999))&((Z1<93501)&(Z1>93533)) Q .. I '\$P(Z0,U,7) S ZE=1 ; ; Extract procedures for UB-04 D F^IBCEF("N-UB-04 PROCEDURES","IBZPRCUB",,IBIFN) ; Does this bill have ANY prescriptions associated with it? ; Must bill prescriptions separately from other charges ; ; DEM;432 - Call line level provider edit checks. D LNPROV^IBCBB12(IBIFN) ; DEM;432 - If there are line provider edits, then routine LNPROV^IBCBB12(IBIFN) updates IBER string. ; DEM;432 - Call to Other Operating/Operating Provider edit checks. I \$\$OPPROVCK^IBCBB12(IBIFN)=1 S IBER=IBER_"IB337;" ; DEM;432 ; DEM;432 - Line level Attachment Control edits. I \$\$LNTMCK^IBCBB11(IBIFN)=1 S IBER=IBER_"IB331;" ; DEM;432 I \$\$LNACCK^IBCBB11(IBIFN)=1 S IBER=IBER_"IB332;" ; DEM;432 ; I \$\$ISRX^IBCEF1(IBIFN) D . N IBZ,IBRXDEF . S IBRXDEF=\$P(\$G(^IBE(350.9,1,1)),U,30),IBZ=0 . F S IBZ=\$O(IBZPRCUB(IBZ)) Q:'IBZ I IBZPRCUB(IBZ)+\$P(IBZPRCUB(IBZ),U)=IBRXDEF S IBER=IBER_"IB102;" Q . K IBZ ; ; Check that COB sequences are not skipped K Z F Z=1:1:3 S:+\$G(^DGCR(399,IBIFN,"I"_Z)) Z(Z)=" F Z=0:1:2 S Z0=\$O(Z(Z)) Q:'Z0 I Z0'=(Z+1) S IBER=IBER_"IB322;" Q </pre> |

| Routine Name | IBCBB1 |
|--------------|--|
| | <pre> K Z ; HD64676 IB*2*371 - OK for payer sequence to be blank when the Rate ; Type is either Interagency or Sharing Agreement I \$P(\$G(^DGCR(399,IBIFN,0)),U,21)="",\$P(\$G(^DGCR(399,IBIFN,0)),U,7)'=4,\$P(\$G(^DGCR(399,IBIFN,0)),U,7)'=9 S IBER=IBER_"IB323;" K IBXDATA D F^IBCEF("N-PROCEDURE CODING METHD",,,IBIFN) ; Coding method should agree with types of procedure codes S IBOK=\$S(\$O(IBZPRC(0))!(IBXDATA=""):1,1:0) I 'IBOK S IBOK=1,IBZ=0 F S IBZ=\$O(IBZPRC(IBZ)) Q:'IBZ I IBZPRC(IBZ),\$P(IBZPRC(IBZ),U)'\$S(IBXDATA=9:"ICD",1:"ICP") S IBOK=0 Q I 'IBOK D WARN^IBCBB11("Coding Method does not agree with all procedure codes found on bill") D EDITMRA^IBCBB3(.IBQUIT,.IBER,IBIFN,IBFT) Q:\$G(IBQUIT) ; ;Other things that could be added: Rev Code - calculating charges ; Diagnosis Coding, if MT copay - check for other co-payments ; I \$P(IBNDTX,U,8),\$REQMRA^IBEFUNC(IBIFN) S IBER=IBER_"IB121;" ; can't force MRAs to print I \$P(IBNDTX,U,8)!\$P(IBNDTX,U,9) D WARN^IBCBB11(\$S(\$REQMRA^IBEFUNC(IBIFN)&(\$P(IBNDTX,U,9)):"MRA Secondary ",1:"")_"Bill has been forced to print "_\$S(\$P(IBNDTX,U,8)=1!(\$P(IBNDTX,U,9)=1):"locally",1:"at clearinghouse")) N IBXZ,IBIZ F IBIZ=12,13,14 S IBXZ=\$P(IBNDM,U,IBIZ) I +IBXZ S IBXZ=\$P(\$G(^DPT(DFN,.312,IBXZ,0)),U,18) I +IBXZ S IBXZ=\$G(^IBA(355.3,+IBXZ,0)) I +\$P(IBXZ,U,12) D . D WARN^IBCBB11(\$P(\$G(^DIC(36,+IBXZ,0)),U,1)" requires Amb Care Certification") ; D VALNDC^IBCBB11(IBIFN,DFN) ;validate NDC# ; ;Build AR array if no errors and MRA not needed or already rec'd I IBER="", \$S(\$NEEDMRA^IBEFUNC(IBIFN)!(\$REQMRA^IBEFUNC(IBIFN)):0,1:1) D ARRAY ; ;Check ROI N ROIERR S ROIERR=0 I \$P(\$G(^DGCR(399,IBIFN,"U")),U,5)=1,+\$P(\$G(^DGCR(399,IBIFN,"U")),U,7)=0 S ROIERR=1 ; screen 7 sensitive record and no ROI I \$\$ROICLK^IBCBB11(IBIFN,DFN,+IBNDMP) S ROIERR=1 ; check file for sensitive Rx and missing ROI I ROIERR S IBER=IBER_"IB328;" ; ;Verify Line Charges Match Claim Total Charge. IB*2.0*447 BI I +\$\$GET1^DIQ(399,IBIFN_",",201)'=+\$S(1BLNTOT^IBCBB13(IBIFN) S IBER=IBER_"IB344;" ; ;Test for valid EIN/SY ID Values. IB*2.0*447 BI I \$\$IBSYEI^IBCBB13(IBIFN) S IBER=IBER_"IB345;" ; ;Test for a missing ICN. IB*2.0*447 BI I \$\$IBMICN^IBCBB13(IBIFN) S IBER=IBER_"IB346;" ; ;Test for a ZERO charge amounts. IB*2.0*447 BI I \$\$IBRCCHK^IBCBB13(IBIFN) D WARN^IBCBB11("Claim contains revenue codes with no associated charges.") </pre> |

| Routine Name | IBCBB1 |
|--------------|--|
| | <pre> ; ;Test for missing "Patient reason for visit". IB*2.0*447 BI I \$\$\$FT^IBCEF(IBIFN)=3,\$\$INPAT^IBCEF(IBIFN),\$IBPRV3^IBCBB13(IBIFN) S IBER=IBER_"IB347;" ; ;Test for missing Payer ID. IB*2.0*447 BI ;I \$\$\$BMPID^IBCBB13(IBIFN) S IBER=IBER_"IB348;" ;Changed Error to Warning. IB*2.0*447 TAZ I \$\$\$BMPID^IBCBB13(IBIFN) D WARN^IBCBB11("Not all payers have Payer IDs.") ; ;Test for missing "Priority (Type) of Admission" for UB-04. IB*2.0*447 BI I \$\$\$FT^IBCEF(IBIFN)=3,\$\$GET1^DIQ(399,IBIFN_",",158)=" S IBER=IBER_"IB349;" ; END ;Don't kill IBIFN, IBER, DFN I \$Q(^TMP(\$J,"BILL-WARN",0)),\$G(IBER)=" S IBER="WARN" ;Warnings only K IBBNO,IBEVDI,IBLOC,IBCL,IBTF,IBAT,IBWHO,IBST,IBFDT,IBTDT,IBTC,IBFY,IBFY1,IBAU,IBRU,IBE U,IBARTP,IBFYC,IBMRA,IBTOB,IBTOB12,IBNDU2,IBNDUF3,IBNDUF31,IBNDTX K IBNDS,IBND0,IBNDU,IBNDM,IBNDMP,IBNDU1,IBFFY,IBTFY,IBFT,IBRTCHV,IBPICHV,IBXDATA,IBO K I \$D(IBER),IBER="" W !,"No Errors found for National edits" Q ; ARRAY ;Build PRCASV(array) N IBCOBN,X K PRCASV Q:\$\$\$MCRWNR^IBEFUNC(+\$\$\$CURR^IBCEF2(IBIFN)) S IBCOBN=\$\$\$COBN^IBCEF(IBIFN) S X=IBIFN S PRCASV("BDT")=DT,PRCASV("ARREC")=IBIFN S PRCASV("APR")=DUZ S PRCASV("PAT")=DFN,PRCASV("CAT")=\$P(^DGCR(399.3,IBAT,0),"^",6) I IBWHO="i" S PRCASV("DEBTOR")=+IBNDMP_"DIC(36," S PRCASV("DEBTOR")=\$S(IBWHO="p":DFN_"DPT(",IBWHO="o":\$P(IBNDM,"^",11)_"DIC(4,"IBWHO=" "i":PRCASV("DEBTOR"),1:"") S PRCASV("CARE")=\$E(\$\$TOB^IBCEF1(IBIFN),1,2) S PRCASV("FY")=\$\$FY^IBOUTL(DT)_U_(\$P(IBNDU1,U)-\$P(IBNDU1,U,2)) ;S PRCASV("FY")=\$P(IBNDU1,U,9)_U_(\$P(IBNDU1,U,2))":(\$P(IBNDU1,U,10)- \$P(IBNDU1,U,2)),1:\$P(IBNDU1,U,10))_S(\$P(IBNDU1,U,11))":U_\$P(IBNDU1,U,11)_U_\$P(IBNDU1,U,1 2),1:"") PLUS I IBWHO="i",\$P(IBNDM,"^",2),\$D(^DIC(36,\$P(IBNDM,"^",2),0)) S PRCASV("2NDINS")=\$P(IBNDM,"^",2) I IBWHO="i",\$P(IBNDM,"^",3),\$D(^DIC(36,\$P(IBNDM,"^",3),0)) S PRCASV("3RDINS")=\$P(IBNDM,"^",3) ; N IBX S IBX=\$P(IBND0,U,21),IBX=\$S(IBX="P":^"I1",IBX="S":^"I2",IBX="T":^"I3",1:"") Q:IBX="" N IBNDI1 Q:\$D(^DGCR(399,IBIFN,IBX)) S IBNDI1=^(IBX) S:\$P(IBNDI1,"^",3)]"" PRCASV("GPNO")=\$P(IBNDI1,"^",3) S:\$P(IBNDI1,"^",15)]"" PRCASV("GPNM")=\$P(IBNDI1,"^",15) S:\$P(IBNDI1,"^",17)]"" PRCASV("INPA")=\$P(IBNDI1,"^",17) S:\$P(IBNDI1,"^",2)]"" PRCASV("IDNO")=\$P(IBNDI1,"^",2),PRCASV("INID")=PRCASV("IDNO") ; Check that this is a secondary or tertiary bill and insurance for previous </pre> |

| Routine Name | IBCBB1 |
|---|--------|
| <pre> ; COB sequence is Medicare WNR and MRA is active --> send data elements to AR I IBCOBN>1,\$\$WNRBILL^IBEFUNC(IBIFN,IBCOBN-1),\$SEDIACV^IBCEF4(2) D MRA Q ; ; MRA N IBEOB S IBEOB=0 ; ; K PRCASV("MEDURE"),PRCASV("MEDCA") ; Get EOB data F S IBEOB=\$O(^IBM(361.1,"B",IBIFN,IBEOB)) Q:'IBEOB D . D MRACALC^IBCEMU2(IBEOB,IBIFN,1,.PRCASV) Q ;MRA ; ; ;; PREGNANCY DX CODES: V22**-V24**, V27**-V28**, 630**-677** ;; FLU SHOTS PROCEDURE CODES: 90724, G0008, 90732, G0009 </pre> | |
| Modified Logic (Changes are in bold) | |
| <pre> IBCBB1 ;ALB/AAS - CONTINUATION OF EDIT CHECK ROUTINE ;2-NOV-89 ;;2.0;INTEGRATED BILLING;**27,52,80,93,106,51,151,148,153,137,232,280,155,320,343,349,363,371,395,384,432,447,488 **;21-MAR-94;Build 80 ;;Per VHA Directive 2004-038, this routine should not be modified. ; ;MAP TO DGCRBB1 ; ; ; *** Begin IB*2.0*488 ; ; NI S I="" S X=+\$G(^DGCR(399,IBIFN,"MP")) I 'X,\$\$MCRWNR^IBEFUNC(+\$\$CURR^IBCEF2(IBIFN)) S X=+\$G(^CURR^IBCEF2(IBIFN)) I X,+\$G(^DIC(36,X,3)) S I=\$P(^3,U,\$S(\$\$FT^IBCEF(IBIFN)=2:2,1:4)) S I=\$\$UP^XLFSTR(I) I (I=""&((I["SPRNT")!(!["HPRNT"])&(\$G(IBER)["IB488"]))) D . S IBER=\$G(IBER)_ "IB488;" ; ; I \$P(\$G(^DGCR(399,IBIFN,"TX")),U,8)=2 D . S IBER=\$G(IBER)_ "IB489;" ; ; ; Cause a fatal error if the claim has no procedures & is NOT a UB-04 Inpatient claim. I +\$O(^DGCR(399,IBIFN,"CP",0))=0 D .I \$\$INPAT^IBCEF(IBIFN,1)&(\$\$FT^IBCEF(IBIFN)=3) Q ; inpatient UB-04 check .I IBER["IB352" Q .S IBER=IBER_ "IB352;" ; *** End IB*2.0*488 ; ; % ;Bill Status N Z,Z0,Z1,IBFT I \$S(+IBST=0:1,1:"^1^2^3^4^7^")[(U_IBST_U)) S IBER=IBER_ "IB045;" ; ; ;Statement Covers From </pre> | |

| Routine Name | IBCBB1 |
|--|--------|
| <pre> I IBFDT="" S IBER=IBER_"IB061;" I IBFDTJ"" ,IBFDT'?7N&(IBFDT'?7N1"."N) S IBER=IBER_"IB061;" I IBFDT>IBTDT S IBER=IBER_"IB061;" ; from must be on or before the to date S IBFFY=\$\$FY^IBOUTL(IBFDT) ; if inpat - from date must not be prior to admit date. I \$\$INPAT^IBCEF(IBIFN,1),(IBFDT<(\$P(\$G(^DGPT(+P(IBND0,U,8),0)),U,2)\1)) S IBER=IBER_"IB061;" ; ;Statement Covers To I IBTDT="" S IBER=IBER_"IB062;" I IBTDTJ"" ,IBTDT'?7N&(IBTDT'?7N1"."N) S IBER=IBER_"IB062;" I IBTDT>DT!(IBTDT<IBFDT) S IBER=IBER_"IB062;" ; to date must not be >than today's date S IBTFY=\$\$FY^IBOUTL(IBTDT) ; ;Total Charges ; IB*2.0*447/TAZ Removed this error so that zero dollar revenue codes can process on the 837 ;I +IBTC'>0!(+IBTC'=IBTC) S IBER=IBER_"IB064;" ; ;Billable charges for secondary claim I \$\$MCRONBIL^IBEFUNC(IBIFN)&((\$P(IBNDU1,U,1)-\$P(IBNDU1,U,2))>0) S IBER=IBER_"IB094;" ;Fiscal Year 1 S IBFFY=\$\$FY^IBOUTL(IBFDT) ; ;Check provider link for current user, enterer, reviewer and Authorizer I '\$D(^VA(200,DUZ,0)) S IBER=IBER_"IB048;" I IBEUJ"" '\$D(^VA(200,IBEU,0)) S IBER=IBER_"IB048;" I IBRUJ"" '\$D(^VA(200,IBRU,0)) S IBER=IBER_"IB060;" I IBAUJ"" '\$D(^VA(200,IBAU,0)) S IBER=IBER_"IB041;" ; I IBER="" ,\$\$STA^PRCAFN(IBIFN)=104 S IBER=IBER_"IB040;" ; If ins bill, must have valid COB sequence I \$P(IBND0,U,11)="i" ,\$\$(\$P(IBND0,U,21)="" :1,1:"PST"[P(IBND0,U,21)) S IBER=IBER_"IB324;" ; ; Check for valid sec provider id for current ins S Z=0 F S Z=\$O(^DGCR(399,IBIFN,"PRV",Z)) Q:'Z S Z0=\$G(^Z,0),Z1=+\$\$COBN^IBCEF(IBIFN) I \$P(Z0,U,4+Z1)="" , \$P(Z0,U,11+Z1)="" D . I '\$\$SECIDCK^IBCEF74(IBIFN,Z1,\$P(Z0,U,11+Z1),Z) D WARN^IBCBB11("Prov secondary id type for the "_\$P("PRIMARY^SECONDARY^TERTIARY",U,Z1)_" "\$\$EXTERNAL^DILFD(399.0222,.01,,+Z0)_" is invalid/won't transmit") ; Check NPIs D NPICHK^IBCBB11 ; ; Check multiple rx NPIs D RXNPI^IBCBB11(IBIFN) ; ; Check taxonomies D TAXCHK^IBCBB11 ; ; Check for Physician Name K IBXDATA D F^IBCEF("N-ATT/REND PHYSICIAN NAME",,,IBIFN) ; IB*2.0*432 - CMS1500 no longer needs a claim level rendering </pre> | |

| Routine Name | IBCB1 |
|--------------|---|
| | <pre> S IBFT=\$\$FT^IBCEF(IBIFN) I IBFT=2,\$P(\$G(IBXDATA),U)="" S IBER=IBER_"IB303;" ; N FUNCTION,IBINS ; IB*2.0*432 - CMS1500 no longer needs a claim level rendering ;S FUNCTION=\$S(\$\$FT^IBCEF(IBIFN)=3:4,1:3) S FUNCTION=\$S(IBFT=3:4,1:3) I IBFT=2,IBER["IB303;" D . F IBINS=1:1:3 D .. S Z=\$\$GETTYP^IBCEP2A(IBIFN,IBINS) .. I Z,\$P(Z,U,2) D ; Rendering/attending prov secondary id required ... N IBID,IBOK,Q0 ... D PROVIN^IBCEF74(IBIFN,IBINS,.IBID,1,"C") ; check all as though they were current ... S IBOK=0 ... S Q0=0 F S Q0=\$O(IBID(1,FUNCTION,Q0)) Q:'Q0 I \$P(IBID(1,FUNCTION,Q0),U,9)=+Z S IBOK=1 Q ... I 'IBOK S IBER=IBER_\$S(IBINS=1:"IB236;",IBINS=2:"IB237;",IBINS=3:"IB238;",1:"") ; ; Patch 432 enh5:The IB system shall no longer prevent users from authorizing(fatal error message)a claim because the system cannot find the providersSSNorEIN ; D PRIIDCHK^IBCB11 ; N IBM,IBM1 S IBM=\$G(^DGCR(399,IBIFN,"M")) S IBM1=\$G(^DGCR(399,IBIFN,"M1")) I \$P(IBM,U),\$P(\$G(^DIC(36,\$P(IBM,U),4)),U,6),\$P(IBM1,U,2)="" S IBER=IBER_"IB244;" I \$P(IBM,U,2),\$P(\$G(^DIC(36,\$P(IBM,U,2),4)),U,6),\$P(IBM1,U,3)="" S IBER=IBER_"IB245;" I \$P(IBM,U,3),\$P(\$G(^DIC(36,\$P(IBM,U,3),4)),U,6),\$P(IBM1,U,4)="" S IBER=IBER_"IB246;" ; ; If outside facility, check for ID and qualifier in 355.93 ; 5/15/06 - esg - hard error IB243 turned into warning message instead S Z=\$P(\$G(^DGCR(399,IBIFN,"U2")),U,10) I Z D . I \$P(\$G(^IBA(355.93,Z,0)),U,9)=""!(\$P(\$G(^IBA(355.93,Z,0)),U,13)="") D .. N Z1,Z2 .. S Z1="Missing Lab or Facility Primary ID for non-VA facility, " .. S Z2=\$\$EXTERNAL^DILFD(399,232,,Z) .. I \$L(Z2)>19 D WARN^IBCB11(Z1_Z2) Q .. D WARN^IBCB11(Z1),WARN^IBCB11(" "_Z2) .. Q . Q ; ; Must be one and only one division on bill S IBZ=\$\$MULTDIV^IBCB11(IBIFN,IBND0) ; I IBZ S IBER=IBER_\$S(IBZ=1:"IB095;",IBZ=2:"IB104;",1:"IB105;") ; Allow multi-divisional for OP institutional claims I IBZ,\$\$INPAT^IBCEF(IBIFN)!(\$\$INSPRF^IBCEF(IBIFN)) S IBER=IBER_\$S(IBZ=1:"IB095;",IBZ=2:"IB104;",1:"IB105;") ; Still need error msg on OP Institutional if No Default division I IBZ=3,\$\$INPAT^IBCEF(IBIFN),\$INSPRF^IBCEF(IBIFN) S IBER=IBER_"IB105;" ; Division address must be defined in institution file </pre> |

| Routine Name | IBCBB1 |
|--------------|---|
| | <pre> I \$P(IBND0,U,22) D . N Z,Z0,Z1 . S Z0=\$G(^DIC(4,\$P(\$G(^DG(40.8,\$P(IBND0,U,22),0)),U,7),0)) . S Z1=\$G(^DIC(4,\$P(\$G(^DG(40.8,\$P(IBND0,U,22),0)),U,7),1)) . I \$P(Z0,U,2)=" S IBER=IBER_"IB097;" Q . F Z=1,3,4 I \$P(Z1,U,Z)=" S IBER=IBER_"IB097;" Q ; ; IB*2.0*432 Check ambulance addresses, COB Non-covered amt. & Attachment Control I \$\$AMBCK^IBCBB11(IBIFN)=1 S IBER=IBER_"IB329;" I \$\$COBAMT^IBCBB11(IBIFN)=1 S IBER=IBER_"IB330;" I \$\$TMCK^IBCBB11(IBIFN)=1 S IBER=IBER_"IB331;" I \$\$ACCK^IBCBB11(IBIFN)=1 S IBER=IBER_"IB332;" I \$\$COBMRA^IBCBB11(IBIFN)=1 S IBER=IBER_"IB342;" I \$\$COBSEC^IBCBB11(IBIFN)=1 S IBER=IBER_"IB343;" ; ;CHAMPVA Rate Type and Primary Insurance Carriers Type of Coverage must match S (IBRTCHV,IBPICHV)=0 I \$P(\$G(^DGCR(399.3,+IBAT,0)),U,1)="CHAMPVA" S IBRTCHV=1 I \$P(\$G(^IBE(355.2,\$P(\$G(^DIC(36,+IBNDMP,0)),U,13),0)),U,1)="CHAMPVA" S IBPICHV=1 I (+IBRTCHV!+IBPICHV)&('IBRTCHV!'IBPICHV) S IBER=IBER_"IB085;" ; N IBZPRC,IBZPRCUB D F^IBCEF("N-ALL PROCEDURES","IBZPRC",,IBIFN) ; Procedure Clinic is required for Surgical Procedures Outpt Facility Charges I +\$P(IBND0,U,27)=2,\$\$BILLRATE^IBCRU3(IBAT,IBCL,IBEVD,"RC OUTPATIENT") D . N Z,Z0,Z1,ZE S (ZE,Z)=0 F S Z=\$O(^DGCR(399,IBIFN,"CP",Z)) Q:'Z D I +ZE S IBER=IBER_"IB320;" Q .. S Z0=\$G(^DGCR(399,IBIFN,"CP",Z,0)),Z1=+Z0 I Z0[';ICPT(" Q .. I '((Z1<10000)&(Z1>69999))&'((Z1<93501)&(Z1>93533)) Q .. I '\$P(Z0,U,7) S ZE=1 ; ; Extract procedures for UB-04 D F^IBCEF("N-UB-04 PROCEDURES","IBZPRCUB",,IBIFN) ; Does this bill have ANY prescriptions associated with it? ; Must bill prescriptions separately from other charges ; ; DEM;432 - Call line level provider edit checks. D LNPROV^IBCBB12(IBIFN) ; DEM;432 - If there are line provider edits, then routine LNPROV^IBCBB12(IBIFN) updates IBER string. ; DEM;432 - Call to Other Operating/Operating Provider edit checks. I \$\$OPPROVCK^IBCBB12(IBIFN)=1 S IBER=IBER_"IB337;" ; DEM;432 ; DEM;432 - Line level Attachment Control edits. I \$\$LNTMCK^IBCBB11(IBIFN)=1 S IBER=IBER_"IB331;" ; DEM;432 I \$\$LNACCK^IBCBB11(IBIFN)=1 S IBER=IBER_"IB332;" ; DEM;432 ; I \$\$ISRX^IBCEF1(IBIFN) D . N IBZ,IBRXDEF . S IBRXDEF=\$P(\$G(^IBE(350.9,1,1)),U,30),IBZ=0 . F S IBZ=\$O(IBZPRCUB(IBZ)) Q:'IBZ I IBZPRCUB(IBZ)+\$P(IBZPRCUB(IBZ),U)=IBRXDEF S IBER=IBER_"IB102;" Q </pre> |

| Routine Name | IBCB1 |
|--------------|---|
| | <pre> . K IBZ ; ; Check that COB sequences are not skipped K Z F Z=1:1:3 S:+\$G(^DGCR(399,IBIFN,"I" _Z)) Z(Z)=" F Z=0:1:2 S Z0=\$O(Z(Z)) Q:'Z0 I Z0'=(Z+1) S IBER=IBER_"IB322;" Q K Z ; HD64676 IB*2*371 - OK for payer sequence to be blank when the Rate ; Type is either Interagency or Sharing Agreement I \$P(\$G(^DGCR(399,IBIFN,0)),U,21)="",\$P(\$G(^DGCR(399,IBIFN,0)),U,7)'=4,\$P(\$G(^DGCR(399,IBIFN,0)),U,7)'=9 S IBER=IBER_"IB323;" K IBXDATA D F^IBCEF("N-PROCEDURE CODING METHD",,,IBIFN) ; Coding method should agree with types of procedure codes S IBOK=\$S(\$O(IBZPRC(0))!(IBXDATA="")):1,1:0) I 'IBOK S IBOK=1,IBZ=0 F S IBZ=\$O(IBZPRC(IBZ)) Q:'IBZ I IBZPRC(IBZ),\$P(IBZPRC(IBZ),U)'[\$S(IBXDATA=9:"ICD",1:"ICP") S IBOK=0 Q I 'IBOK D WARN^IBCB11("Coding Method does not agree with all procedure codes found on bill") D EDITMRA^IBCB11(.IBQUIT,.IBER,IBIFN,IBFT) Q:\$G(IBQUIT) ; ;Other things that could be added: Rev Code - calculating charges ; Diagnosis Coding, if MT copay - check for other co-payments ; I \$P(IBNDTX,U,8),\$REQMRA^IBEFUNC(IBIFN) S IBER=IBER_"IB121;" ; can't force MRAs to print I \$P(IBNDTX,U,8)!\$P(IBNDTX,U,9) D . Q:\$P(IBNDTX,U,8)=2 ; Don't want to do this for option 2 any more. . D WARN^IBCB11(\$S(\$REQMRA^IBEFUNC(IBIFN)&(\$P(IBNDTX,U,9)):"MRA Secondary ",1:"")_"Bill has been forced to print " _\$S(\$P(IBNDTX,U,8)=1!(\$P(IBNDTX,U,9)=1):"locally",1:"at clearinghouse")) N IBXZ,IBIZ F IBIZ=12,13,14 S IBXZ=\$P(IBNDM,U,IBIZ) I +IBXZ S IBXZ=\$P(\$G(^DPT(DFN,.312,IBXZ,0)),U,18) I +IBXZ S IBXZ=\$G(^IBA(355.3,+IBXZ,0)) I +\$P(IBXZ,U,12) D . D WARN^IBCB11(\$P(\$G(^DIC(36,+IBXZ,0)),U,1)" requires Amb Care Certification") ; D VALNDC^IBCB11(IBIFN,DFN) ;validate NDC# ; ;Build AR array if no errors and MRA not needed or already rec'd I IBER="", \$S(\$NEEDMRA^IBEFUNC(IBIFN)!(\$REQMRA^IBEFUNC(IBIFN)):0,1:1) D ARRAY ; ;Check ROI N ROIERR S ROIERR=0 I \$P(\$G(^DGCR(399,IBIFN,"U")),U,5)=1,+ \$P(\$G(^DGCR(399,IBIFN,"U")),U,7)=0 S ROIERR=1 ; screen 7 sensitive record and no ROI I \$\$ROICHK^IBCB11(IBIFN,DFN,+IBNDMP) S ROIERR=1 ; check file for sensitive Rx and missing ROI I ROIERR S IBER=IBER_"IB328;" ; ;Verify Line Charges Match Claim Total Charge. IB*2.0*447 BI I +\$\$GET1^DIQ(399,IBIFN_",",201)'=+\$\$IBLNTOT^IBCB13(IBIFN) S IBER=IBER_"IB344;" ; ;Test for valid EIN/SY ID Values. IB*2.0*447 BI I \$\$IBSYEI^IBCB13(IBIFN) S IBER=IBER_"IB345;" </pre> |

| Routine Name | IBCBB1 |
|--------------|--|
| | <pre> ; ;Test for a missing ICN. IB*2.0*447 BI I \$\$IBMICN^IBCBB13(IBIFN) S IBER=IBER_"IB346;" ; ;Test for a ZERO charge amounts. IB*2.0*447 BI I \$\$IBRCCHK^IBCBB13(IBIFN) D WARN^IBCBB11("Claim contains revenue codes with no associated charges.") ; ;Test for missing "Patient reason for visit". IB*2.0*447 BI I \$\$FT^IBCEF(IBIFN)=3,\$\$INPAT^IBCEF(IBIFN),\$IBPRV3^IBCBB13(IBIFN) S IBER=IBER_"IB347;" ; ;Test for missing Payer ID. IB*2.0*447 BI ;I \$\$IBMPID^IBCBB13(IBIFN) S IBER=IBER_"IB348;" ;Changed Error to Warning. IB*2.0*447 TAZ I \$\$IBMPID^IBCBB13(IBIFN) D WARN^IBCBB11("Not all payers have Payer IDs.") ; ;Test for missing "Priority (Type) of Admission" for UB-04. IB*2.0*447 BI I \$\$FT^IBCEF(IBIFN)=3,\$\$GET1^DIQ(399,IBIFN_"",",158)="" S IBER=IBER_"IB349;" ; END ;Don't kill IBIFN, IBER, DFN I \$O(^TMP(\$J,"BILL-WARN",0)),\$G(IBER)="" S IBER="WARN" ;Warnings only K IBBNO,IBEVDI,IBLOC,IBCL,IBTF,IBAT,IBWHO,IBST,IBFDT,IBTDT,IBTC,IBFY,IBFY1,IBAU,IBRU,IBE U,IBARTP,IBFYC,IBMRA,IBTOB,IBTOB12,IBNDU2,IBNDUF3,IBNDUF31,IBNDTX K IBNDS,IBND0,IBNDU,IBNDM,IBNDMP,IBNDU1,IBFFY,IBTFY,IBFT,IBRTCHV,IBPICHV,IBXDATA,IBO K I \$D(IBER),IBER="" W !,"No Errors found for National edits" Q ; ARRAY ;Build PRCASV(array) N IBCOBN,X K PRCASV Q:\$\$MCRWNR^IBEFUNC(+\$\$CURR^IBCEF2(IBIFN)) S IBCOBN=\$\$COBN^IBCEF(IBIFN) S X=IBIFN S PRCASV("BDT")=DT,PRCASV("ARREC")=IBIFN S PRCASV("APR")=DUZ S PRCASV("PAT")=DFN,PRCASV("CAT")=\$P(^DGCR(399.3,IBAT,0),"^",6) I IBWHO="i" S PRCASV("DEBTOR")=+IBNDMP_" ";DIC(36," S PRCASV("DEBTOR")=\$\$(IBWHO="p":DFN_" ";DPT("IBWHO="o":\$P(IBNDM,"^",11)_" ";DIC(4,"IBWHO= "i":PRCASV("DEBTOR"),1:"") S PRCASV("CARE")=\$E(\$\$TOB^IBCEF1(IBIFN),1,2) S PRCASV("FY")=\$\$FY^IBOUTL(DT)_U_(\$P(IBNDU1,U)-\$P(IBNDU1,U,2)) ;S PRCASV("FY")=\$P(IBNDU1,U,9)_U_\$\$(\$P(IBNDU1,U,2))":(\$P(IBNDU1,U,10)- \$P(IBNDU1,U,2),1:\$P(IBNDU1,U,10))_\$\$(\$P(IBNDU1,U,11))":U_\$P(IBNDU1,U,11)_U_\$P(IBNDU1,U,1 2),1:"") PLUS I IBWHO="i",\$P(IBNDM,"^",2),\$D(^DIC(36,\$P(IBNDM,"^",2),0)) S PRCASV("2NDINS")=\$P(IBNDM,"^",2) I IBWHO="i",\$P(IBNDM,"^",3),\$D(^DIC(36,\$P(IBNDM,"^",3),0)) S PRCASV("3RDINS")=\$P(IBNDM,"^",3) ; N IBX S IBX=\$P(IBND0,U,21),IBX=\$S(IBX="P": "I1",IBX="S": "I2",IBX="T": "I3",1:"") Q:IBX="" </pre> |

| Routine Name | IBCBB1 |
|--|--------|
| <pre> N IBNDI1 Q:\$D(^DGCR(399,IBFN,IBX)) S IBNDI1=^(IBX) S:\$P(IBNDI1,"^",3)]"" PRCASV("GPNO")=\$P(IBNDI1,"^",3) S:\$P(IBNDI1,"^",15)]"" PRCASV("GPNM")=\$P(IBNDI1,"^",15) S:\$P(IBNDI1,"^",17)]"" PRCASV("INPA")=\$P(IBNDI1,"^",17) S:\$P(IBNDI1,"^",2)]"" PRCASV("IDNO")=\$P(IBNDI1,"^",2),PRCASV("INID")=PRCASV("IDNO") ; Check that this is a secondary or tertiary bill and insurance for previous ; COB sequence is Medicare WNR and MRA is active --> send data elements to AR I IBCOBN>1,\$\$WNRBILL^IBEFUNC(IBFN,IBCOBN-1,\$\$EDIACTV^IBCEF4(2) D MRA Q ; MRA N IBEOB S IBEOB=0 ; K PRCASV("MEDURE"),PRCASV("MEDCA") ; Get EOB data F S IBEOB=\$O(^IBM(361.1,"B",IBFN,IBEOB)) Q:'IBEOB D . D MRACALC^IBCEMU2(IBEOB,IBFN,1,.PRCASV) Q ;MRA ; ; ;: PREGNANCY DX CODES: V22**-V24**, V27**-V28**, 630**-677** ;: FLU SHOTS PROCEDURE CODES: 90724, G0008, 90732, G0009 </pre> | |

| Routine Name | IBXS3 | |
|--|---|---|
| Enhancement Category | <input type="checkbox"/> New <input checked="" type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change | |
| Requirement Traceability Matrix | 2.6.1.8 | |
| Related Options | Enter/Edit Billing Information [IB EDIT BILLING INFO] – Screen 3 | |
| Related Routines | Routines “Called By” | Routines “Called” |
| | | ^DIR2, ^DIE17, ^DIED, ^DIEQ, ^DIE0, ^DIE2, ^DIE3, ^DIKC1, ^DIC, ^IBCEP3, ^IBCU1, ^DICR, ^IBCEU7, ^DIE, ^DILF, ^IBCNS2, ^IBXS31, ^IBXS32, ^IBXS33, ^IBCEP7B, ^IBXS34, ^IBXS35, ^IBXS36 |
| Data Dictionary (DD) References | | |
| Related Protocols | | |
| Related Integration Control Registrations (ICRs) | | |
| Data Passing | <input type="checkbox"/> Input <input type="checkbox"/> Output Reference <input type="checkbox"/> Both <input type="checkbox"/> Global Reference <input type="checkbox"/> Local | |
| Input Attribute Name and Definition | | |

| | |
|--|--------------|
| Routine Name | IBXS3 |
| Output Attribute Name and Definition | |
| Current Logic | |
| <pre> . . . 22 D:\$D(DG)>9 F^DIE17,DE S DQ=22,DW="M;12",DV="FXO",DU="",DLB="PRIMARY INSURANCE POLICY",DIFLD=112 S DQ(22,2)="S Y(0)=Y S Y=\$\$TRANS^IBCNS2(\$G(DA,D0),Y)" S DE(DW)="C22^IBXS3",DE(DW,"INDEX")=1 G RE C22 G C22S:\$D(DE(22))[0 K DB S X=DE(22),DIC=DIE K DIV S DIV=X,D0=DA,DIV(0)=D0 S Y(1)=\$S(\$D(^DGCR(399,D0,"M")):^("M"),1:"") S X=\$P(Y(1),U,1),X=X S DIU=X K Y S X="" X ^DD(399,112,1,1,2.4) S X=DE(22),DIC=DIE D KIX^IBCNS2(DA,"I1") S X=DE(22),DIC=DIE X ^DD(399,112,1,3,2.3) I X S X=DIV S Y(1)=\$S(\$D(^DGCR(399,D0,"MP")):^("MP"),1:"") S X=\$P(Y(1),U,2),X=X S DIU=X K Y S X="" X ^DD(399,112,1,3,2.4) C22S S X="" G:DG(DQ)=X C22F1 K DB S X=DG(DQ),DIC=DIE K DIV S DIV=X,D0=DA,DIV(0)=D0 S Y(1)=\$S(\$D(^DGCR(399,D0,"M")):^("M"),1:"") S X=\$P(Y(1),U,1),X=X S DIU=X K Y X ^DD(399,112,1,1,1.1) X ^DD(399,112,1,1,1.4) S X=DG(DQ),DIC=DIE D IX^IBCNS2(DA,"I1") S X=DG(DQ),DIC=DIE X ^DD(399,112,1,3,1.3) I X S X=DIV S Y(1)=\$S(\$D(^DGCR(399,D0,"MP")):^("MP"),1:"") S X=\$P(Y(1),U,2),X=X S DIU=X K Y S X=DIV S X=\$\$BPP^IBCNS2(DA,1) X ^DD(399,112,1,3,1.4) C22F1 S DIEZRXR(399,DIENS)=\$\$OREF^DILF(\$NA(@\$\$CREF^DILF(DIE))) F DIXR=78 S DIEZRXR(399,DIXR)="" Q X22 K:X[""!(\$A(X)=45) X I \$D(X) K:\$L(X)>20!(\$L(X)<1) X D:\$D(X) DD^IBCNS2(X,DA,1) I \$D(X),X'?.ANP K X Q ; . . . </pre> | |
| Modified Logic (Changes are in bold) | |
| <p>Need to implement similar code as the following from GETLN^IBCNS3 so that when the IBXS3 routine is compiled it will display the payer sequence as part of each insurance policy on the pick list:</p> <pre> . . </pre> | |

| Routine Name | IBXS3 |
|---|-------|
| <pre> . ; GETLN(IBPOL0,IBDATE) ; get single line of primary data on insurance policy ; input: IBPOL0 = line from array, zero node of patient policy (2,..312) ; IBDATE = date to check coverage, default today ; output: formatted line of data for insurance policy in TMP(\$J,"IBCNS") ; N IBX,IBLINE S IBLINE=" " S IBPOL0=\$G(IBPOL0) ; S IBX=\$G(^DIC(36,+IBPOL0,0)),IBX=\$S(\$P(IBX,U,1)'=""\$P(IBX,U,1),1:"UNKNOWN") S IBLINE=\$\$FRMLN(IBX,IBLINE,11,0) S IBX=\$P(IBPOL0,U,20),IBX=\$S(IBX=1:"p",IBX=2:"s",IBX=3:"t",1:"") S IBLINE=\$\$FRMLN(IBX,IBLINE,1,14) . . . </pre> | |

| | | | |
|--|--|--------------------------------|--|
| Routine Name | IBCSC8 | | |
| Enhancement Category | <input type="checkbox"/> New <input checked="" type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change | | |
| Requirement Traceability Matrix | 2.6.7.1, 2.6.7.2 | | |
| Related Options | Enter/Edit Billing Information [IB EDIT BILLING INFO] – Screens 8 & 10 | | |
| Related Routines | Routines “Called By” | Routines “Called” | |
| | ^IBCSC7, ^IBCSCP | ^IBCSCU, %DTC, ^XLFDT, ^IBCSCP | |
| Data Dictionary (DD) References | BILL/CLAIMS File (#399) | | |
| Related Protocols | | | |
| Related Integration Control Registrations (ICRs) | | | |
| Data Passing | <input type="checkbox"/> Input <input type="checkbox"/> Output Reference <input checked="" type="checkbox"/> Both <input type="checkbox"/> Global Reference <input type="checkbox"/> Local | | |
| Input Attribute Name and Definition | | | |
| Output Attribute Name and Definition | | | |
| Current Logic | | | |

| Routine Name | IBCSC8 |
|--|--------|
| <pre> IBCSC8 ;ALB/MJB/AAS - MCCR SCREEN 8 (BILLING - CLAIM INFORMATION SCREEN) ;27 MAY 88 10:15 ;;2.0;INTEGRATED BILLING;**432,447**;21-MAR-94;Build 80 ;;Per VHA Directive 2004-038, this routine should not be modified. ; ; EN D ^IBCSCU S IBSR=8,IBSR1="" S IB("U4")=\$G(^DGCR(399,IBFN,"U4")),IB("U5")=\$G(^DGCR(399,IBFN,"U5")),IB(" U6")=\$G(^DGCR(399,IBFN,"U6")),IB("U8")=\$G(^DGCR(399,IBFN,"U8")) D H^IBCSCU ; DEM - IBV is set in EDI^IBCB => S IBAC=1,IBV=0 D EN G Q:'IBAC1,EDI ; IBV=0, or IBV=1 as a flag if field on screen is required ; or not. <Field #> indicates field is not required. ; [Field #] ; Make some sections NOT available for UB04 form S IBT=\$P(\$G(^DGCR(399,IBFN,0)),U,19) S IBV1=\$S(IBT=3:"001011",IBV:"111111",1:"000000") ; S Z=1,IBW=1 X IBWW W " COB Non-Covered Charge Amt: " S X=\$P(IB("U4"),U),X2="2\$" I X="" D COMMA^%DTC W X S Z=2 X IBWW W " Property Casualty Information" W !,?4,"Claim Number: ",\$P(IB("U4"),U,2),?41,"Contact Name: ",\$P(IB("U4"),U,9) W !,?4,"Date of 1st Contact: ",\$FMTE^XLFD(\$P(IB("U4"),U,3)),?41,"Contact Phone: ",\$P(IB("U4"),U,10)," ",\$P(IB("U4"),U,11) ; Start IB*2.0*447 BI ;S Z=3 X IBWW W " Ambulance Information" ;W !,?41,"D/O Location: ",\$P(IB("U6"),U) ;W !,?4,"P/U Address1: ",\$P(IB("U5"),U,2),?41,"D/O Address1: ",\$P(IB("U6"),U,2) ;W !,?4,"P/U Address2: ",\$P(IB("U5"),U,3),?41,"D/O Address2: ",\$P(IB("U6"),U,3) ;W !,?4,"P/U City: ",\$P(IB("U5"),U,4),?41,"D/O City: ",\$P(IB("U6"),U,4) ;W !,?4,"P/U State/Zip: " W:\$P(IB("U5"),U,5)'="" \$P(\$G(^DIC(5,\$P(IB("U5"),U,5),0)),U,2) ;W:\$P(IB("U5"),U,6)]" "/"_\$P(IB("U5"),U,6) ;W ?41,"D/O State/Zip: " W:\$P(IB("U6"),U,5)'="" \$P(\$G(^DIC(5,\$P(IB("U6"),U,5),0)),U,2) ;W:\$P(IB("U6"),U,6)]" "/"_\$P(IB("U6"),U,6) ;;W !,?4,"P/U Country/SubDiv: ",\$P(IB("U5"),U),?41,"D/O Country/SubDiv: " S Z=3 X IBWW W " Surgical Codes for Anesthesia Claims" W !,?4,"Primary Code: " W:\$P(IB("U4"),U,7)'="" \$P(\$G(^ICPT(\$P(IB("U4"),U,7),0)),U) W ?41,"Secondary Code: " W:\$P(IB("U4"),U,8)'="" \$P(\$G(^ICPT(\$P(IB("U4"),U,8),0)),U) S Z=4 X IBWW W " Paperwork Attachment Information" W !,?4,"Report Type: " W:\$P(IB("U8"),U,2)'="" \$P(\$G(^IBE(353.3,\$P(IB("U8"),U,2),0)),U) W ?41,"Transmission Method: ",\$P(IB("U8"),U,3) W !,?4,"Attachment Control #: ",\$P(IB("U8"),U) S Z=5 X IBWW W " Disability Start Date: ",\$FMTE^XLFD(\$P(IB("U4"),U,4)),?41,"Disability End Date: ",\$FMTE^XLFD(\$P(IB("U4"),U,5)) S Z=6 X IBWW W " Assumed Care Date: ",\$FMTE^XLFD(\$P(IB("U4"),U,13)),?41,"Relinquished Care Date: ",\$FMTE^XLFD(\$P(IB("U4"),U,14)) ; End IB*2.0*447 BI W ! REV G ^IBCSCP </pre> | |

| Routine Name | IBCS8 |
|--|-------|
| ;IBCS8 | |
| Modified Logic (Changes are in bold) | |
| IBCS8 ;ALB/MJB/AAS - MCCR SCREEN 8 (BILLING - CLAIM INFORMATION SCREEN) ;27 MAY 88 10:15 ;;2.0;INTEGRATED BILLING;**432,447**;21-MAR-94;Build 80 ;;Per VHA Directive 2004-038, this routine should not be modified. ; ; EN D ^IBCSU S IBSR=8,IBSR1="" S IB("U4")=\$G(^DGCR(399,IBFN,"U4")),IB("U5")=\$G(^DGCR(399,IBFN,"U5")),IB("U6")=\$G(^DGCR(399,IBFN,"U6")),IB("U8")=\$G(^DGCR(399,IBFN,"U8")) D H^IBCSU ; DEM - IBV is set in EDI^IBCB => S IBAC=1,IBV=0 D EN G Q:IBAC1,EDI ; IBV=0, or IBV=1 as a flag if field on screen is required ; or not. <Field #> indicates field is not required. ; [Field #] ; Make some sections NOT available for UB04 form S IBT=\$P(\$G(^DGCR(399,IBFN,0)),U,19) S IBV1=\$S(IBT=3:"001011",IBV:"111111",1:"000000") ; S Z=1,IBW=1 X IBWW W " COB Non-Covered Charge Amt: " S X=\$P(IB("U4"),U),X2="2\$" I X="" D COMMA^%DTC W X S Z=2 X IBWW W " Property Casualty Information" W !,?4,"Claim Number: ",\$P(IB("U4"),U,2),?41,"Contact Name: ",\$P(IB("U4"),U,9) W !,?4,"Date of 1st Contact: ",\$FMTE^XLFD(\$P(IB("U4"),U,3)),?41,"Contact Phone: ",\$P(IB("U4"),U,10)," ",\$P(IB("U4"),U,11) ; Start IB*2.0*447 BI ;S Z=3 X IBWW W " Ambulance Information" ;W !,?41,"D/O Location: ",\$P(IB("U6"),U) ;W !,?4,"P/U Address1: ",\$P(IB("U5"),U,2),?41,"D/O Address1: ",\$P(IB("U6"),U,2) ;W !,?4,"P/U Address2: ",\$P(IB("U5"),U,3),?41,"D/O Address2: ",\$P(IB("U6"),U,3) ;W !,?4,"P/U City: ",\$P(IB("U5"),U,4),?41,"D/O City: ",\$P(IB("U6"),U,4) ;W !,?4,"P/U State/Zip: " W:\$P(IB("U5"),U,5)'="" \$P(\$G(^DIC(5,\$P(IB("U5"),U,5),0)),U,2) ;W:\$P(IB("U5"),U,6)]'" /"_\$P(IB("U5"),U,6) ;W ?41,"D/O State/Zip: " W:\$P(IB("U6"),U,5)'="" \$P(\$G(^DIC(5,\$P(IB("U6"),U,5),0)),U,2) ;W:\$P(IB("U6"),U,6)]'" /"_\$P(IB("U6"),U,6) ;;W !,?4,"P/U Country/SubDiv: ",\$P(IB("U5"),U),?41,"D/O Country/SubDiv: " S Z=3 X IBWW W " Surgical Codes for Anesthesia Claims" W !,?4,"Primary Code: " W:\$P(IB("U4"),U,7)'="" \$P(\$G(^ICPT(\$P(IB("U4"),U,7),0)),U) W ?41,"Secondary Code: " W:\$P(IB("U4"),U,8)'="" \$P(\$G(^ICPT(\$P(IB("U4"),U,8),0)),U) S Z=4 X IBWW W " Paperwork Attachment Information" W !,?4,"Report Type: " W:\$P(IB("U8"),U,2)'="" \$P(\$G(^IBE(353.3,\$P(IB("U8"),U,2),0)),U) W ?41,"Transmission Method: ",\$P(IB("U8"),U,3) W !,?4,"Attachment Control #: ",\$P(IB("U8"),U) S Z=5 X IBWW W " Disability Start Date: ",\$FMTE^XLFD(\$P(IB("U4"),U,4)),?41,"Disability End Date: ",\$FMTE^XLFD(\$P(IB("U4"),U,5)) S Z=6 X IBWW W " Assumed Care Date: ",\$FMTE^XLFD(\$P(IB("U4"),U,13)),?41,"Relinquished Care Date: ",\$FMTE^XLFD(\$P(IB("U4"),U,14)) ; End IB*2.0*447 BI | |

| Routine Name | IBCSC8 |
|--|--------|
| W ! ; /Beginning of IB*2.0*488 (vd) I \$\$FT^IBCU3(IBIFN)'=2 G REV ; If not CMS-1500, skip over the following lines of code. I \$P(IB("U2"),U,14)'="" W !,?4,"Homebound : ",\$\$EXPAND^IBTRE(399,236,\$P(IB("U2"),U,14)) I \$P(IB("U2"),U,15)'="" W !,?4,"Date Last Seen : ",\$\$EXPAND^IBTRE(399,237,\$P(IB("U2"),U,15)) I \$P(IB("U2"),U,16)'="" W !,?4,"Spec Prog Indicator: " S IBZ=\$\$EXPAND^IBTRE(399,238,\$P(IB("U2"),U,16)) W \$S(IBZ'="" :IBZ,\$\$WNRBILL^IBEFUNC(IBIFN):"31",1:"") ; /End of IB488 (vd) REV G ^IBCSCP ;IBCSC8 | |

| | | | |
|---|---|---|--|
| Routine Name | IBCSC10H | | |
| Enhancement Category | <input type="checkbox"/> New <input checked="" type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change | | |
| Requirement Traceability Matrix | 2.6.1.8, 2.6.1.9 | | |
| Related Options | Enter/Edit Billing Information [IB EDIT BILLING INFO] – Screens 8 & 10 | | |
| Related Routines | Routines “Called By” | Routines “Called” | |
| | ^IBCSC10, ^IBCSC102, ^IBCSCP, ^IBCU7B, ^IBXSA21, ^IBXSAH1 | ^IBCSCU, ^IBACSV, ^IBCEU, ^IBCEF, ^IBCEF74, ^IBCEF73A, ^IBTRE, ^DIQ, ^IBCEP8A, ^DIE, ^IBEFUNC, ^IBCEF79, ^IBCEF4, ^DILFD, ^IBCSCP | |
| Data Dictionary (DD) References | | | |
| Related Protocols | | | |
| Related Integration Control Registrations (ICRs) | | | |
| Data Passing | <input type="checkbox"/> Input <input type="checkbox"/> Output Reference <input type="checkbox"/> Both <input type="checkbox"/> Global Reference <input type="checkbox"/> Local | | |
| Input Attribute Name and Definition | | | |
| Output Attribute Name and Definition | | | |
| Current Logic | | | |
| IBCSC10H ;ALB/ARH - MCCR SCREEN 10 (BILL SPECIFIC INFO) CMS-1500 ;4/21/92 ;;2.0;INTEGRATED BILLING;**432**;21-MAR-94;Build 192 ;;Per VHA Directive 2004-038, this routine should not be modified. ; CMS-1500 screen 10 ; ; | | | |

| Routine Name | IBCSC10H |
|--|----------|
| <pre> ; MAP TO DGCRSC8H ; ; DEM;432 - Moved IBCSC8* billing screen routines to IBCSC10* billing screen ; routines and created a new billing screen 8 routine IBCSC8. ; EN ; N I,IB,Y,Z D ^IBCSCU S IBSR=10,IBSR1="H",IBV1="000000000" S:IBV IBV1="111111111" F I="U","U1","UF2","UF3","UF32","U2","M","TX",0,"U3" S IB(I)=\$G(^DGCR(399,IBIFN,I)) N IBZ,IBPRV,IBDATE,IBREQ,IBMRASEC,IBZ1,IBZCNT ; S IBDATE=\$\$BDATE^IBACSV(IBIFN) ; Date of service for the bill S IBPRV="" D GETPRV^IBCEU(IBIFN,"ALL",.IBPRV) K IB("PRV") S IBZ=0 F S IBZ=\$O(IBPRV(IBZ)) Q:'IBZ I \$O(IBPRV(IBZ,0))!\$D(IBPRV(IBZ,"NOTOPT")) M IB("PRV",IBZ)=IBPRV(IBZ) ; D H^IBCSCU ; ; Section 1 S Z=1,IBW=1 X IBWW W " Unable To Work From: " S Y=\$P(IB("U"),U,16) X ^DD("DD") W \$(Y'="" :Y,1:IBUN) W !?4,"Unable To Work To : " S Y=\$P(IB("U"),U,17) X ^DD("DD") W \$(Y'="" :Y,1:IBUN) ; ; Section 2 S Z=2,IBW=1 X IBWW I \$\$INPAT^IBCEF(IBIFN) W " Admitting Dx : " S IBZ=\$\$ICD9^IBACSV(+IB("U2"),IBDATE) W \$(IBZ'="" :\$P(IBZ,U) - " _\$P(IBZ,U,3),1:IBUN),! S IBZCNT=0,IBZ(IBZCNT)=" I \$P(IB("UF3"),U,4)]"" S IBZ(IBZCNT)="P: " _\$P(IB("UF3"),U,4),IBZCNT=IBZCNT+1 I \$P(IB("UF3"),U,5)]"" S IBZ(IBZCNT)="S: " _\$P(IB("UF3"),U,5),IBZCNT=IBZCNT+1 I \$P(IB("UF3"),U,6)]"" S IBZ(IBZCNT)="T: " _\$P(IB("UF3"),U,6) S:IBZ(0)="" IBZ(0)=IBUN W ?4,"ICN/DCN(s) : ",IBZ(0) F IBZCNT=1:1 Q:'\$D(IBZ(IBZCNT)) W !?25,IBZ(IBZCNT) K IBZ S IBZ=\$\$CKPROV^IBCEU(IBIFN,3) S IBZCNT=0,IBZ(IBZCNT)=" I \$P(IB("U"),U,13)]"" S IBZ(IBZCNT)="P: " _\$P(IB("U"),U,13),IBZCNT=IBZCNT+1 I \$P(IB("U2"),U,8)]"" S IBZ(IBZCNT)="S: " _\$P(IB("U2"),U,8),IBZCNT=IBZCNT+1 I \$P(IB("U2"),U,9)]"" S IBZ(IBZCNT)="T: " _\$P(IB("U2"),U,9),IBZCNT=IBZCNT+1 I \$P(IB("UF32"),U,1)]"" S IBZ(IBZCNT)="P: " _\$P(IB("UF32"),U,1),IBZCNT=IBZCNT+1 I \$P(IB("UF32"),U,2)]"" S IBZ(IBZCNT)="S: " _\$P(IB("UF32"),U,2),IBZCNT=IBZCNT+1 I \$P(IB("UF32"),U,3)]"" S IBZ(IBZCNT)="T: " _\$P(IB("UF32"),U,3) S:IBZ(0)="" IBZ(0)=IBUN W !,?3," Auth/Referral : ",IBZ(0) F IBZCNT=1:1 Q:'\$D(IBZ(IBZCNT)) W !?25,IBZ(IBZCNT) K IBZ S IBZ="" ; </pre> | |

| Routine Name | IBCSC10H |
|--------------|--|
| | <pre> ; Section 3 S Z=3,IBW=1 X IBWW W " Providers : ",\$(O(IB("PRV",0)):IBU,1:"") I \$D(IB("PRV")) D ; at least 1 provider found . N IBQ,A,A1,IBARR,IBTAX,IBNOTAX,IBSPEC,IBNOSPEC . S IBZ=0 . D DEFSEC^IBCEF74(IBIFN,IBARR) . ; PRXM/KJH - Add Taxonomy code to display for patch 343. Moved secondary IDs slightly (below). . S IBTAX=\$\$PROVTAX^IBCEF73A(IBIFN,IBNOTAX) . S IBSPEC=\$\$SPECTAX^IBCEF73A(IBIFN,IBNOSPEC) . F S IBZ=\$O(IB("PRV",IBZ)) Q:'IBZ D .. S IBQ="" .. W !,?5,"- " .. S A=\$\$EXPAND^IBTRE(399.0222,.01,IBZ) .. I \$P(\$G(IB("PRV",IBZ,1)),U,4)'="" S A1=" ("_\$E(\$P(IB("PRV",IBZ,1),U,4),1,3)_)" ,A=\$E(A,1,16-\$L(A1))_A1 .. W \$E(A_\$J("",16),1,16),": " .. I '\$P(\$G(IB("PRV",IBZ,1)),U,3),\$P(\$G(IB("PRV",IBZ,1)),U)'="" W IBU Q .. I \$P(\$G(IB("PRV",IBZ,1)),U)'="" W:\$G(IB("PRV",IBZ)) \$E(\$P(IB("PRV",IBZ,1),U)_\$J("",16),1,16) W:\$G(IB("PRV",IBZ)) "(OLD BOX 31 DATA) "_\$P(IB("PRV",IBZ,1),U) .. I \$P(\$G(IB("PRV",IBZ,1)),U)'="" , \$P(\$G(IB("PRV",IBZ)),U)'="" W \$E(\$P(IB("PRV",IBZ),U)_\$J("",16),1,16) .. W " Taxonomy: ",\$(P(IBTAX,U,IBZ))'="" :\$P(IBTAX,U,IBZ),1:IBU),\$(P(IBSPEC,U,IBZ))'="" : ("\$P(IBSPEC,U,IBZ)_") ,1:"") .. F A=1:1:3 I \$G(IBARR(IBZ,A))'="" S IBQ=IBQ_"["_\$E("PST",A)_"]" _IBARR(IBZ,A)_ " .. I \$L(IBQ) W !,?30,\$E(IBQ,1,49) ; K IB("PRV") ; ; Section 4 S Z=4,IBW=1 X IBWW W " Other Facility (VA/non): " S IBZ=\$\$EXPAND^IBTRE(399,232,+\$P(IB("U2"),U,10)) W \$(IBZ'="" :\$E(IBZ,1,23),\$\$PSRV^IBCEU(IBIFN):IBU,1:IBUN) I IBZ'="" D . ; PRXM/KJH - Add Taxonomy code to display for patch 343. . W ?53,"Taxonomy: " . S IBZ=\$\$GET1^DIQ(8932.1,+\$P(IB("U3"),U,3),"X12 CODE") W \$(IBZ'="" :IBZ,1:IBU) . S IBZ=\$\$GET1^DIQ(8932.1,+\$P(IB("U3"),U,3),"SPECIALTY CODE") W \$(IBZ'="" : (" _IBZ_"),1:"") . Q ; ; clia# display - IB patch 320 S (IBZ,IBZ1)=\$P(IB("U2"),U,13) ; retrieve CLIA# from database ; I IBZ'="" D . NEW CLIAREQ,DEFCLIA,DIE,DA,DR . S CLIAREQ=\$\$CLIAREQ^IBCEP8A(IBIFN) . I 'CLIAREQ S IBZ1=IBUN Q ; clia# not needed . S DEFCLIA=\$\$CLIA^IBCEP8A(IBIFN) ; default clia# for claim . I DEFCLIA="" S IBZ1=IBU Q ; no default found . I \$G(IBMDOTCN) K IBMDOTCN S IBZ1=IBU Q ; user @-deleted clia# . S IBZ1=DEFCLIA ; display and stuff default clia# </pre> |

| Routine Name | IBCSC10H |
|--------------|--|
| | <pre> . S DIE=399,DA=IBIFN,DR="235///" _DEFCLIA D ^DIE ; stuff in default . Q ; ; W !,?4,"Lab CLIA # : ",IBZ1 ; ; Mammo# display IB patch 320 S (IBZ,IBZ1)=\$P(IB("U3"),U,1) ; retrieve mammo# from database ; ; If mammo# is there, but should not be, then blank it out I IBZ'="", \$\$XRAY^IBCEP8A(IBIFN) D . NEW DIE,DA,DR . S IBZ1=IBUN ; mammo# not needed . S DIE=399,DA=IBIFN,DR="242///@" D ^DIE . Q ; ; I IBZ="" S IBZ1=IBUN W !?4,"Mammography Cert # : ",IBZ1 ; ; ; Section 5 S Z=5,IBW=1 X IBWW W " Chiropractic Data : " S Y=\$P(IB("U3"),U,5) X ^DD("DD") W \$(Y'="" : "INITIAL TREATMENT ON " _Y,1:IBUN) ; ; ; Section 6 S Z=6,IBW=1 X IBWW W " Form Locator 19 : " S IBZ=\$P(\$G(^DGCR(399,IBIFN,"UF31")),U,3) W \$(IBZ'="" : IBZ,1:IBUN) I \$P(IB("U2"),U,14)'="" W !,?4,"Homebound : ", \$\$EXPAND^IBTRE(399,236,\$P(IB("U2"),U,14)) I \$P(IB("U2"),U,15)'="" W !,?4,"Date Last Seen : ", \$\$EXPAND^IBTRE(399,237,\$P(IB("U2"),U,15)) I \$P(IB("U2"),U,16)'="" W !,?4,"Spec Prog Indicator: " S IBZ=\$\$EXPAND^IBTRE(399,238,\$P(IB("U2"),U,16)) W \$(IBZ'="" : IBZ,\$\$WNRBILL^IBEFUNC(IBIFN):"31",1:"") ; ; ; Section 7 S Z=7,IBW=1 X IBWW W " Billing Provider : " K IBZ D GETBP^IBCEF79(IBIFN,"",+\$\$B^IBCEF79(IBIFN),"CMS-1500 SCREEN 8",.IBZ) S IBZ=\$G(IBZ("CMS-1500 SCREEN 8","NAME")) W \$(IBZ'="" : IBZ,1:IBU) ; billing provider name W !?3," Taxonomy Code : " S IBZ=\$\$GET1^DIQ(8932.1,+P(IB("U3"),U,11),"X12 CODE") W \$(IBZ'="" : IBZ,1:IBU) S IBZ=\$\$GET1^DIQ(8932.1,+P(IB("U3"),U,11),"SPECIALTY CODE") W \$(IBZ'="" : (" _IBZ_ "),1:"") ; ; ; Section 8 S Z=8,IBW=1 X IBWW S IBREQ=+\$REQMRA^IBEFUNC(IBIFN) S:IBREQ IBREQ=1 S IBMRASEC=\$\$MRASEC^IBCEF4(IBIFN) W " ",\$(IBREQ:"Force To Print? : ",1:"Force MRA Sec Prt? : ") S IBZ=\$\$EXTERNAL^DILFD(399,27+IBREQ,+P(IB("TX"),U,8+IBREQ)) I IBMRASEC,\$P(IB("TX"),U,8),\$P(IB("TX"),U,9) S IBZ="FORCED TO PRINT BY MRA PRIMARY",\$P(IB("TX"),U,8)=0 </pre> |

| Routine Name | IBCSC10H |
|---|----------|
| <pre> W \$S(IBZ'=""&(\$P(IB("TX"),U,8+IBREQ)'=""):IBZ,\$\$TXMT^IBCEF4(IBIFN):"[NOT APPLICABLE - NOT TRANSMITTABLE]",IBREQ:"NO FORCED PRINT",1:IBZ) ; ; Section 9 S Z=9,IBW=1 X IBWW W " Provider ID Maint : (Edit Provider ID information)",! G ^IBCSCP Q Q ; WRT1(IBCRED) ; Write credentials mismatch W !,*7," **Warning** Credentials differ from those found in NEW PERSON or IB NON VA",!,\$J("",14),"BILLING PROVIDER file (",\$S(IBCRED="":"none",1:IBCRED),")" W !,\$J("",14),"Changes will print local, but only credentials on file transmit" Q ; NSAME(DA) ; Returns 1 if div on bill is not the default billing facility Q (\$P(\$G(^IBE(350.9,1,0)),U,2)'=\$P(\$G(^DG(40.8,\$P(^DGCR(399,DA,0),U,22),0)),U,7)) ; ;IBCSC10H </pre> | |
| Modified Logic (Changes are in bold> | |
| <pre> IBCSC10H ;ALB/ARH - MCCR SCREEN 10 (BILL SPECIFIC INFO) CMS-1500 ;4/21/92 ;;2.0;INTEGRATED BILLING;**432**;21-MAR-94;Build 192 ;;Per VHA Directive 2004-038, this routine should not be modified. ; CMS-1500 screen 10 ; ; ; MAP TO DGCRC8H ; ; DEM;432 - Moved IBCSC8* billing screen routines to IBCSC10* billing screen ; routines and created a new billing screen 8 routine IBCSC8. ; EN ; N I,IB,Y,Z D ^IBCSCU S IBSR=10,IBSR1="H",IBV1="000000000" S:IBV IBV1="111111111" F I="U","U1","UF2","UF3","UF32","U2","M","TX",0,"U3" S IB(I)=\$G(^DGCR(399,IBIFN,I)) N IBZ,IBPRV,IBDATE,IBREQ,IBMRASEC,IBZ1,IBZCNT ; S IBDATE=\$\$BDATE^IBACSV(IBIFN) ; Date of service for the bill S IBPRV="" D GETPRV^IBCEU(IBIFN,"ALL",.IBPRV) K IB("PRV") S IBZ=0 F S IBZ=\$O(IBPRV(IBZ)) Q:'IBZ I \$O(IBPRV(IBZ,0))!\$D(IBPRV(IBZ,"NOTOPT")) M IB("PRV",IBZ)=IBPRV(IBZ) ; ; D H^IBCSCU ; ; Section 1 S Z=1,IBW=1 X IBWW W " Unable To Work From: " S Y=\$P(IB("U"),U,16) X ^DD("DD") W \$S(Y'="" :Y,1:IBUN) </pre> | |

| Routine Name | IBCSC10H |
|---|----------|
| <pre> W !?4,"Unable To Work To : " S Y=\$P(IB("U"),U,17) X ^DD("DD") W \$(Y'="" :Y,1:IBUN) ; ; Section 2 S Z=2,IBW=1 X IBWW I \$\$INPAT^IBCEF(IBIFN) W " Admitting Dx : " S IBZ=\$\$ICD9^IBACSV(+IB("U2"),IBDATE) W \$(IBZ'="" :\$P(IBZ,U)_ - "_\$P(IBZ,U,3),1:IBUN),! S IBZCNT=0,IBZ(IBZCNT)=" I \$P(IB("UF3"),U,4)]"" S IBZ(IBZCNT)="P: "_\$P(IB("UF3"),U,4),IBZCNT=IBZCNT+1 I \$P(IB("UF3"),U,5)]"" S IBZ(IBZCNT)="S: "_\$P(IB("UF3"),U,5),IBZCNT=IBZCNT+1 I \$P(IB("UF3"),U,6)]"" S IBZ(IBZCNT)="T: "_\$P(IB("UF3"),U,6) S:IBZ(0)="" IBZ(0)=IBUN W ?4,"ICN/DCN(s) : ",IBZ(0) F IBZCNT=1:1 Q:'\$D(IBZ(IBZCNT)) W !?25,IBZ(IBZCNT) K IBZ S IBZ=\$\$CKPROV^IBCEU(IBIFN,3) S IBZCNT=0,IBZ(IBZCNT)=" I \$P(IB("U"),U,13)]"" S IBZ(IBZCNT)="P: "_\$P(IB("U"),U,13),IBZCNT=IBZCNT+1 I \$P(IB("U2"),U,8)'="" S IBZ(IBZCNT)="S: "_\$P(IB("U2"),U,8),IBZCNT=IBZCNT+1 I \$P(IB("U2"),U,9)'="" S IBZ(IBZCNT)="T: "_\$P(IB("U2"),U,9),IBZCNT=IBZCNT+1 I \$P(IB("UF32"),U,1)'="" S IBZ(IBZCNT)="P: "_\$P(IB("UF32"),U,1),IBZCNT=IBZCNT+1 I \$P(IB("UF32"),U,2)'="" S IBZ(IBZCNT)="S: "_\$P(IB("UF32"),U,2),IBZCNT=IBZCNT+1 I \$P(IB("UF32"),U,3)'="" S IBZ(IBZCNT)="T: "_\$P(IB("UF32"),U,3) S:IBZ(0)="" IBZ(0)=IBUN W !,?3," Auth/Referral : ",IBZ(0) F IBZCNT=1:1 Q:'\$D(IBZ(IBZCNT)) W !?25,IBZ(IBZCNT) K IBZ S IBZ="" ; ; Section 3 S Z=3,IBW=1 X IBWW W " Providers : ",\$(S(\$O(IB("PRV",0)):IBU,1:")) I \$D(IB("PRV")) D ; at least 1 provider found . N IBQ,A,A1,IBARR,IBTAX,IBNOTAX,IBSPEC,IBNOSPEC . S IBZ=0 . D DEFSEC^IBCEF74(IBIFN,.IBARR) . ; PRXM/KJH - Add Taxonomy code to display for patch 343. Moved secondary IDs slightly (below). . S IBTAX=\$\$PROVTAX^IBCEF73A(IBIFN,.IBNOTAX) . S IBSPEC=\$\$SPECTAX^IBCEF73A(IBIFN,.IBNOSPEC) . F S IBZ=\$O(IB("PRV",IBZ)) Q:'IBZ D .. S IBQ="" .. W !,?5," - " .. S A=\$\$EXPAND^IBTRE(399.0222,.01,IBZ) .. I \$P(\$G(IB("PRV",IBZ,1)),U,4)'="" S A1=" ("_\$(P(IB("PRV",IBZ,1),U,4),1,3)_)" ,A=\$E(A,1,16-\$L(A1))_A1 .. W \$(A_\$J("",16),1,16),": " .. I '\$P(\$G(IB("PRV",IBZ,1)),U,3),\$P(\$G(IB("PRV",IBZ,1)),U)'="" W IBU Q .. I '\$P(\$G(IB("PRV",IBZ,1)),U)'="" W:'\$G(IB("PRV",IBZ)) \$E(\$P(IB("PRV",IBZ,1),U)_\$J("",16),1,16) W:\$G(IB("PRV",IBZ)) (OLD BOX 31 DATA) "_\$P(IB("PRV",IBZ,1),U) .. I '\$P(\$G(IB("PRV",IBZ,1)),U)'="" , \$P(\$G(IB("PRV",IBZ)),U)'="" W \$(P(IB("PRV",IBZ),U)_\$J("",16),1,16) .. W " Taxonomy: ",\$(P(IBTAX,U,IBZ))'="" :\$P(IBTAX,U,IBZ),1:IBU,\$\$(P(IBSPEC,U,IBZ))'="" : (" _\$P(IBSPEC,U,IBZ)_)" ,1: "") .. F A=1:1:3 I \$G(IBARR(IBZ,A))'="" S IBQ=IBQ_"["_\$(PST,A)_"]_"_IBARR(IBZ,A)_ " </pre> | |

| Routine Name | IBCSC10H |
|--|----------|
| <pre> .. I \$L(IBQ) W !,?30,\$E(IBQ,1,49) ; K IB("PRV") ; ; Section 4 S Z=4,IBW=1 X IBWW W " Other Facility (VA/non): " S IBZ=\$\$EXPAND^IBTRE(399,232,\$P(IB("U2"),U,10)) W \$\$S(IBZ'=""\$E(IBZ,1,23),\$\$PSRV^IBCEU(IBIFN):IBU,1:IBUN) I IBZ="" D . ; PRXM/KJH - Add Taxonomy code to display for patch 343. . W ?53,"Taxonomy: " . S IBZ=\$\$GET1^DIQ(8932.1,\$P(IB("U3"),U,3),"X12 CODE") W \$\$S(IBZ'=""\$IBZ,1:IBU) . S IBZ=\$\$GET1^DIQ(8932.1,\$P(IB("U3"),U,3),"SPECIALTY CODE") W \$\$S(IBZ'=""\$ ("_IBZ_"),1:"") . Q ; ; clia# display - IB patch 320 S (IBZ,IBZ1)=\$P(IB("U2"),U,13) ; retrieve CLIA# from database ; I IBZ="" D . NEW CLIAREQ,DEFCLIA,DIE,DA,DR . S CLIAREQ=\$\$CLIAREQ^IBCEP8A(IBIFN) . I 'CLIAREQ S IBZ1=IBUN Q ; clia# not needed . S DEFCLIA=\$\$CLIAREQ^IBCEP8A(IBIFN) ; default clia# for claim . I DEFCLIA="" S IBZ1=IBU Q ; no default found . I \$G(IBMDDOTCN) K IBMDDOTCN S IBZ1=IBU Q ; user @-deleted clia# . S IBZ1=DEFCLIA ; display and stuff default clia# . S DIE=399,DA=IBIFN,DR="235///" _DEFCLIA D ^DIE ; stuff in default . Q ; W !,?4,"Lab CLIA # : ",IBZ1 ; ; Mammo# display IB patch 320 S (IBZ,IBZ1)=\$P(IB("U3"),U,1) ; retrieve mammo# from database ; ; If mammo# is there, but should not be, then blank it out I IBZ'=""\$, \$\$XRAY^IBCEP8A(IBIFN) D . NEW DIE,DA,DR . S IBZ1=IBUN ; mammo# not needed . S DIE=399,DA=IBIFN,DR="242///@" D ^DIE . Q ; I IBZ="" S IBZ1=IBUN W !?4,"Mammography Cert # : ",IBZ1 ; ; Section 5 S Z=5,IBW=1 X IBWW W " Chiropractic Data : " S Y=\$P(IB("U3"),U,5) X ^DD("DD") W \$\$S(Y'=""\$ "INITIAL TREATMENT ON " _Y,1:IBUN) ; ; Section 6 </pre> | |

| Routine Name | IBCSC10H |
|--------------|---|
| | <pre> S Z=6,IBW=1 X IBWW W " Form Locator 19 : " S IBZ=\$P(\$G(^DGCR(399,IBIFN,"UF31")),U,3) W \$S(IBZ'="" :IBZ,1:IBUN) ; / Beginning of IB*2.0*488 – Moved this lines of code to ^IBCSC8 (vd) ; I \$P(IB("U2"),U,14)'="" W !,?4,"Homebound : ",,\$\$EXPAND^IBTRE(399,236,\$P(IB("U2"),U,14)) ; I \$P(IB("U2"),U,15)'="" W !,?4,"Date Last Seen : ",,\$\$EXPAND^IBTRE(399,237,\$P(IB("U2"),U,15)) ; I \$P(IB("U2"),U,16)'="" W !,?4,"Spec Prog Indicator: " S IBZ=\$\$EXPAND^IBTRE(399,238,\$P(IB("U2"),U,16)) W \$S(IBZ'="" :IBZ,\$\$WNRBILL^IBEFUNC(IBIFN):"31",1:"") ; /End of IB*2.0*488 (vd) ; ; Section 7 S Z=7,IBW=1 X IBWW W " Billing Provider : " K IBZ D GETBP^IBCEF79(IBIFN,"",+\$\$B^IBCEF79(IBIFN),"CMS-1500 SCREEN 8",.IBZ) S IBZ=\$G(IBZ("CMS-1500 SCREEN 8","NAME")) W \$S(IBZ'="" :IBZ,1:IBU) ; billing provider name W !?3," Taxonomy Code : " S IBZ=\$\$GET1^DIQ(8932.1,+\$\$P(IB("U3"),U,11),"X12 CODE") W \$S(IBZ'="" :IBZ,1:IBU) S IBZ=\$\$GET1^DIQ(8932.1,+\$\$P(IB("U3"),U,11),"SPECIALTY CODE") W \$S(IBZ'="" : ("_IBZ_"),1:"") ; ; Section 8 S Z=8,IBW=1 X IBWW S IBREQ=+\$\$REQMRA^IBEFUNC(IBIFN) S:IBREQ IBREQ=1 S IBMRASEC=\$\$MRASEC^IBCEF4(IBIFN) W " ",,\$S('IBREQ:"Force To Print? : ",1:"Force MRA Sec Prt? : ") S IBZ=\$\$EXTERNAL^DILFD(399,27+IBREQ,,+\$\$P(IB("TX"),U,8+IBREQ)) I IBMRASEC,\$\$P(IB("TX"),U,8),\$P(IB("TX"),U,9) S IBZ="FORCED TO PRINT BY MRA PRIMARY",,\$P(IB("TX"),U,8)=0 W \$S(IBZ'=""&(\$P(IB("TX"),U,8+IBREQ)'=""):IBZ,\$\$TXMT^IBCEF4(IBIFN):"[NOT APPLICABLE - NOT TRANSMITTABLE]",IBREQ:"NO FORCED PRINT",1:IBZ) ; ; Section 9 S Z=9,IBW=1 X IBWW W " Provider ID Maint : (Edit Provider ID information)",! G ^IBCSCP Q Q ; WRT1(IBCRED) ; Write credentials mismatch W !,*7," **Warning** Credentials differ from those found in NEW PERSON or IB NON VA",!,,\$J("",14),"BILLING PROVIDER file (",\$S(IBCRED="" : "none",1:IBCRED),")" W !,\$J("",14),"Changes will print local, but only credentials on file transmit" Q ; NSAME(DA) ; Returns 1 if div on bill is not the default billing facility Q (\$P(\$G(^IBE(350.9,1,0)),U,2)'=\$P(\$G(^DG(40.8,+\$\$P(^DGCR(399,DA,0),U,22),0)),U,7)) ; ;IBCSC10H </pre> |

6.2.2.1.3. Unique Record(s) for CMS – 1500

IB DATA ELEMENT : HOMEBOUND (HCFA 1500)

| Field Name(s) | Current Value | New Value |
|---------------------------------|---|--|
| Requirement Traceability Matrix | 2.6.7.1, 2.6.7.2 | |
| .01 Name | HOMEBOUND | |
| .02 Security Level | N | |
| .03 Type of Element | SET | |
| .04 Element Category | 236 | |
| .05 Base File | 399 | |
| 3 Description | This is the indicator to tell the CMS-1500 to print the homebound statement in box 19. Refer to the MEDICARE regulations on when to fill in this field. | This is to indicate that the patient is homebound or institutionalized. Refer to MEDICARE regulations on when to use this field. |

IB DATA ELEMENT : DATE LAST SEEN (HCFA 1500)

| Field Name(s) | Current Value | New Value |
|---------------------------------|---|--|
| Requirement Traceability Matrix | 2.6.7.1, 2.6.7.2 | |
| .01 Name | DATE LAST SEEN | |
| .02 Security Level | N | |
| .03 Type of Element | DATE | |
| .04 Element Category | 237 | |
| .05 Base File | 399 | |
| 3 Description | This data will print in box 19 of the CMS-1500 for printed claims. Refer to the MEDICARE regulations on when to fill in this field. | This is the date a patient was last seen. Refer to MEDICARE regulations on when to use this field. |

IB DATA ELEMENT : SPECIAL PROGRAM INDICATOR (HCFA 1500)

| Field Name(s) | Current Value | New Value |
|---------------------------------|---------------------------|-----------|
| Requirement Traceability Matrix | 2.6.7.1, 2.6.7.2 | |
| .01 Name | SPECIAL PROGRAM INDICATOR | |
| .02 Security Level | N | |

| Field Name(s) | Current Value | New Value |
|----------------------|--|---|
| .03 Type of Element | SET | |
| .04 Element Category | 238 | |
| .05 Base File | 399 | |
| 3 Description | This is an indicator to tell the CMS-1500 to print the statement associated with the special program in box 19. Refer to the MEDICARE regulations on when to fill in this field. | This is the Special Program with which a claim is associated. Refer to MEDICARE regulations to decide when to use this field. |

6.2.2.2. System Feature: Data Dictionary

This section covers the design for the Functional Requirements listed in Section 2.6.2 of the RSD.

6.2.2.2.1. Functional Requirements: Data Dictionary

| SRS REQ ID | REQ Title | Comments / Notes |
|------------|--|---|
| 2.6.2.1 | Bill Number Length – Accounts Receivable | Refer to section 3.2.2.1 – Accounts Receivable Bill Number Length |

6.2.2.2.2. Routines (Entry Points)

This section does not require any routines to be modified.

6.2.2.2.3. Unique Records for Accounts Receivable File (#430)

Refer to section 3.2.2.1 of this document for required modification to the Accounts Receivable Bill Number Length.

6.2.2.3. System Feature: Provider Maintenance

This section covers the design for the Functional Requirements listed in Section 2.6.3 of the RSD.

6.2.2.3.1. Functional Requirements: Enhancements to the Provider Maintenance

| SRS REQ ID | REQ Title | Comments / Notes |
|---------------|---------------------------------|------------------|
| 2.6.3.1 | Outside Facility ZIP Code | |
| 2.6.3.2 | Outside Facility Address Line 1 | |

6.2.2.3.2. Routines (Entry Points)

| | | |
|---|--|--|
| Routine Name | IBCEP8B | |
| Enhancement Category | <input type="checkbox"/> New <input checked="" type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change | |
| Requirement Traceability Matrix | 2.6.3.1, 2.6.3.2 | |
| Related Options | Provide ID Maintenance [IBCE PROVIDER MAINT] | |
| Related Routines | Routines "Called By" | Routines "Called" |
| | ^IBCEP8 | ^IBCEP81, ^DILFD, ^DIQ, ^IBCEP8C1, ^VALM10 |
| Data Dictionary (DD) References | IB NON/OTHER VA BILLING PROVIDER File (# 355.93) | |
| Related Protocols | IBCE PRVNVA MAINT | |
| Related Integration Control Registrations (ICRs) | | |
| Data Passing | <input type="checkbox"/> Input <input type="checkbox"/> Output Reference <input checked="" type="checkbox"/> Both <input type="checkbox"/> Global Reference <input type="checkbox"/> Local | |
| Input Attribute Name and Definition | | |
| Output Attribute Name and Definition | | |
| Current Logic | | |
| <pre> I \$P(Z,U,2)'=2 D .,IB*2.0*476 - END added prompt to allow OPTION FB PAID TO IB to make updates or not . S IBCT=IBCT+1 . S Z1=\$J("Address: ",15)_\$P(Z,U,5) D SET1(.IBLCT,Z1,IBCT) . I \$P(Z,U,10) D .. S IBCT=IBCT+1 .. S Z1=\$J("",15)_\$P(Z,U,10) . S IBCT=IBCT+1 . S Z1=\$J("",15)_\$P(Z,U,6)_\$S(\$P(Z,U,6)'="":", ",1:"")_\$\$(\$P(Z,U,7):\$EXTERNAL^DILFD(355.93,.07,"",\$P(Z,U,7))_" ",1:"")_\$P(Z,U,8) </pre> | | |

| Routine Name | IBCEP8B |
|---|---------|
| . D SET1(.IBLCT,Z1,IBCT) . ; start contact changes here | |
| Modified Logic (Changes are in bold) | |
| <pre> I \$P(Z,U,2)'=2 D .;IB*2.0*476 - END added prompt to allow OPTION FB PAID TO IB to make updates or not . S IBCT=IBCT+1 . S Z1=\$J("Name: ",19)_\$P(Z,U) D SET1(.IBLCT,Z1,IBCT) .. * 33 .;; Begin IB*2.0*488 -RBN .. * 33 . N XX,BADADD,BADZIP,MSG . S MSG=" " . S (BADADD,BADZIP)=0 . S XX=\$P(Z,U,5) . I \$L(XX)>30!(\$L(XX)<1)!(XX?.E1"BOX"."."#"." "1N.E) D .. S BADADD=1 . S XX=\$P(Z,U,8) . I \$L(XX)'=9,(\$L(XX)'=10) D .. S BADZIP=1 . I (XX?9N),(\$E(XX,6,9)["0000"]) D .. S BADZIP=1 . I (XX?5N1"-4N),(\$E(XX,7,10)["0000"]) D .. S BADZIP=1 .. * 33 .;; End IB*2.0*488 .. * 33 . S IBCT=IBCT+1 . S Z1=\$J("Address: ",19)_\$P(Z,U,5) D SET1(.IBLCT,Z1,IBCT) . I \$P(Z,U,10) D .. S IBCT=IBCT+1 .. S Z1=\$J("",19)_\$P(Z,U,10) ; This is the street2 of the address - NOT displayed . S IBCT=IBCT+1 . S Z1=\$J("",19)_\$P(Z,U,6)_\$\$(\$P(Z,U,6)'=""", "1:"")_\$\$(\$P(Z,U,7):\$\$EXTERNAL^DILFD(355.93,.07,"",\$P(Z,U,7))_" "1:"")_ \$P(Z,U,8) . D SET1(.IBLCT,Z1,IBCT) .. * 33 .;; Begin IB*2.0*488 - RBN .. * 33 . I BADADD S MSG=MSG_"Address cannot be a PO BOX" </pre> | |

| | |
|--|----------------|
| Routine Name | IBCEP8B |
| . I BADZIP S MSG=\$S(MSG'=" ":MSG_ " & ",1:MSG) S MSG=MSG_"ZIP must be 9 digits not ending in 0000 or 5 digits 1 hyphen and 4 digits not ending in 0000" . I BADADD!BADZIP D .. S IBCT=IBCT+1 .. S Z1=" " .. D SET1(.IBLCT,Z1,IBCT) .. S IBCT=IBCT+1 .. D SET1(.IBLCT,MSG,IBCT) .. S IBCT=IBCT+1 .. S Z1=" " .. D SET1(.IBLCT,Z1,IBCT) .. .;; End IB*2.0*488 .;; | |

6.2.2.4. System Feature: Insurance Company Entry/Edit

This section covers the design for the Functional Requirements listed in Section 2.6.4 of the RSD.

6.2.2.4.1. Functional Requirements: Insurance Company Entry/Edit

| SRS REQ ID | REQ Title | Comments / Notes |
|------------|---|------------------|
| 2.6.4.1 | Payer Primary ID – Institutional | |
| 2.6.4.2 | Payer Primary ID – Professional | |
| 2.6.4.3 | Value for EDI – Transmit? – New Insurance Company | |

6.2.2.4.2. Routines (Entry Points)

| | | | |
|---------------------------------|--|--------------------------|--|
| Routine Name | IBCNSC01 | | |
| Enhancement Category | <input type="checkbox"/> New <input checked="" type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change | | |
| Requirement Traceability Matrix | 2.6.4.3 | | |
| Related Options | Insurance Company Entry/Edit [IBCN INSURANCE CO EDIT] | | |
| Related Routines | Routines “Called By” | Routines “Called” | |

| | | |
|---|--|--|
| Routine Name | IBCNSC01 | |
| | ^IBCNSC, ^IBCNSM6, ^IBCNSP0 | ^IBCNSP, ^IBTRE, ^IBCNSU31, ^DIC, ^DIQ, ^DILFD, ^XLFD |
| Data Dictionary (DD) References | INSURANCE COMPANY File (#36) | |
| Related Protocols | | |
| Related Integration Control Registrations (ICRs) | | |
| Data Passing | <input type="checkbox"/> Input <input type="checkbox"/> Output Reference <input checked="" type="checkbox"/> Both <input type="checkbox"/> Global Reference <input type="checkbox"/> Local | |
| Input Attribute Name and Definition | | |
| Output Attribute Name and Definition | | |
| Current Logic | | |
| <pre> IBCNSC01 ;ALB/NLR - INSURANCE COMPANY EDIT ;6/1/05 10:06am ;;2.0;INTEGRATED BILLING;**52,137,191,184,232,320,349,371,399,416,432**;;21-MAR-94;Build 192 ;;Per VHA Directive 2004-038, this routine should not be modified. ; PARAM ; -- Insurance company parameters region N OFFSET,START,IBCNS0,IBCNS03,IBCNS06,IBCNS08,IBCNS13,IBCNS3 S IBCNS0=\$G(^DIC(36,+IBCNS,0)),IBCNS3=\$G(^3) S IBCNS03=\$P(IBCNS0,"^",3),IBCNS06=\$P(IBCNS0,"^",6),IBCNS08=\$P(IBCNS0,"^",8) S IBCNS13=\$G(^DIC(36,+IBCNS,.13)) S START=1,OFFSET=2 D SET^IBCNSP(START,OFFSET+25," Billing Parameters ",IORVON,IORVOFF) ; D SET^IBCNSP(START+1,OFFSET+1,"Signature Required?: "_\$S(+IBCNS03:"YES",1:"NO")) D SET^IBCNSP(START+2,OFFSET+10,"Reimburse?: _"\$E(\$\$EXPAND^IBTRE(36,1,\$P(IBCNS0,"^",2)),1,21)) D SET^IBCNSP(START+3,OFFSET+3,"Mult. Bedsections: _"\$S(+IBCNS06:"YES",IBCNS06=0:"NO",1:"")) D SET^IBCNSP(START+4,OFFSET+6,"One Opt. Visit: "_\$S(+IBCNS08:"YES",1:"NO")) D SET^IBCNSP(START+5,OFFSET+4,"Diff. Rev. Codes: "_\$P(IBCNS0,"^",7)) D SET^IBCNSP(START+6,OFFSET+1,"Amb. Sur. Rev. Code: "_\$P(IBCNS0,"^",9)) D SET^IBCNSP(START+7,OFFSET+1,"Rx Refill Rev. Code: "_\$P(IBCNS0,"^",15)) D SET^IBCNSP(START+8,OFFSET+3,"Filing Time Frame: _"\$P(IBCNS0,"^",12)_\$S(+\$P(IBCNS0,"^",18):" ("_\$FTFN^IBCNSU31(+IBCNS)_)",1:"")) ; S OFFSET=45 D SET^IBCNSP(START+1,OFFSET+4,"Type Of Coverage: _"\$E(\$\$EXPAND^IBTRE(36,.13,\$P(IBCNS0,U,13))) D SET^IBCNSP(START+2,OFFSET+7,"Billing Phone: "_\$P(IBCNS13,"^",2)) D SET^IBCNSP(START+3,OFFSET+2,"Verification Phone: "_\$P(IBCNS13,"^",4)) D SET^IBCNSP(START+4,OFFSET+2,"Precert Comp. Name: </pre> | | |

| Routine Name | IBCNSC01 |
|--------------|--|
| | <pre> "_\$P(\$G(^DIC(36,\$P(IBCNS13,"^",9),0)), "^",1)) D SET^IBCNSP(START+5,OFFSET+7,"Precert Phone: " _\$\$PHONE(IBCNS13)) I +IBCNS3=2 D SET^IBCNSP(START+6,OFFSET,"Max # Test Bills/Day: " _\$P(IBCNS3,U,6)) ; S START=11,OFFSET=2 D SET^IBCNSP(START,OFFSET+28," EDI Parameters ",IORVON,IORVOFF) D SET^IBCNSP(START+1,OFFSET+13,"Transmit?: " _\$(+IBCNS3=1:"YES-LIVE",+IBCNS3=2:"TEST ONLY",1:"NO")) D SET^IBCNSP(START+2,OFFSET+1,"Inst Payer Primary ID: " _\$P(IBCNS3,U,4)) D SET^IBCNSP(START+3,OFFSET,"Inst Payer Sec ID Qual: " _\$\$GET1^DIQ(36,+IBCNS,6.01)) D SET^IBCNSP(START+4,OFFSET+5,"Inst Payer Sec ID: " _\$\$GET1^DIQ(36,+IBCNS,6.02)) D SET^IBCNSP(START+5,OFFSET,"Inst Payer Sec ID Qual: " _\$\$GET1^DIQ(36,+IBCNS,6.03)) D SET^IBCNSP(START+6,OFFSET+5,"Inst Payer Sec ID: " _\$\$GET1^DIQ(36,+IBCNS,6.04)) D SET^IBCNSP(START+7,OFFSET+12,"Bin Number: " _\$P(\$G(^DIC(36,+IBCNS,3)), "^",3)) ; ; S OFFSET=41 D SET^IBCNSP(START+1,OFFSET+8," Insurance Type: " _\$\$EXPAND^IBTRE(36,3.09,\$P(IBCNS3,U,9))) D SET^IBCNSP(START+2,OFFSET+1," Prof Payer Primary ID: " _\$P(IBCNS3,U,2)) D SET^IBCNSP(START+3,OFFSET," Prof Payer Sec ID Qual: " _\$\$GET1^DIQ(36,+IBCNS,6.05)) D SET^IBCNSP(START+4,OFFSET+5," Prof Payer Sec ID: " _\$\$GET1^DIQ(36,+IBCNS,6.06)) D SET^IBCNSP(START+5,OFFSET," Prof Payer Sec ID Qual: " _\$\$GET1^DIQ(36,+IBCNS,6.07)) D SET^IBCNSP(START+6,OFFSET+5," Prof Payer Sec ID: " _\$\$GET1^DIQ(36,+IBCNS,6.08)) ;IB*2.0*432/TAZ Added fields 6.09 and 6.1 D SET^IBCNSP(START+7,OFFSET-3," Prnt Sec/Tert Auto Claims: " _\$\$GET1^DIQ(36,+IBCNS,6.09)) D SET^IBCNSP(START+8,OFFSET-5," Prnt Med Sec Claims w/o MRA: " _\$\$GET1^DIQ(36,+IBCNS,6.1)) Q ; . . . </pre> |
| | Modified Logic (Changes are in bold) |
| | <pre> IBCNSC01 ;ALB/NLR - INSURANCE COMPANY EDIT ;6/1/05 10:06am ;;2.0;INTEGRATED BILLING;**52,137,191,184,232,320,349,371,399,416,432**;;21-MAR-94;Build 192 ;;Per VHA Directive 2004-038, this routine should not be modified. ; PARAM ; -- Insurance company parameters region N OFFSET,START,IBCNS0,IBCNS03,IBCNS06,IBCNS08,IBCNS13,IBCNS3 S IBCNS0=\$G(^DIC(36,+IBCNS,0)),IBCNS3=\$G(^3) S IBCNS03=\$P(IBCNS0,"^",3),IBCNS06=\$P(IBCNS0,"^",6),IBCNS08=\$P(IBCNS0,"^",8) S IBCNS13=\$G(^DIC(36,+IBCNS,.13)) S START=1,OFFSET=2 D SET^IBCNSP(START,OFFSET+25," Billing Parameters ",IORVON,IORVOFF) ; D SET^IBCNSP(START+1,OFFSET+1,"Signature Required?: " _\$(+IBCNS03:"YES",1:"NO")) D SET^IBCNSP(START+2,OFFSET+10,"Reimburse?: </pre> |

| Routine Name | IBCNSC01 |
|--------------|--|
| | <pre> "_\$E(\$\$EXPAND^IBTRE(36,1,\$P(IBCNS0,"^",2)),1,21)) D SET^IBCNSP(START+3,OFFSET+3,"Mult. Bedsections: "\$S(+IBCNS06:"YES",IBCNS06=0:"NO",1:"")) D SET^IBCNSP(START+4,OFFSET+6,"One Opt. Visit: "\$S(+IBCNS08:"YES",1:"NO")) D SET^IBCNSP(START+5,OFFSET+4,"Diff. Rev. Codes: "\$P(IBCNS0,"^",7)) D SET^IBCNSP(START+6,OFFSET+1,"Amb. Sur. Rev. Code: "\$P(IBCNS0,"^",9)) D SET^IBCNSP(START+7,OFFSET+1,"Rx Refill Rev. Code: "\$P(IBCNS0,"^",15)) D SET^IBCNSP(START+8,OFFSET+3,"Filing Time Frame: "\$P(IBCNS0,"^",12)_\$S(+\$P(IBCNS0,"^",18):("\$\$_\$FTFN^IBCNSU31(+IBCNS)_",1:"")) ; S OFFSET=45 D SET^IBCNSP(START+1,OFFSET+4,"Type Of Coverage: "\$\$_\$EXPAND^IBTRE(36,13,\$P(IBCNS0,U,13))) D SET^IBCNSP(START+2,OFFSET+7,"Billing Phone: "\$P(IBCNS13,"^",2)) D SET^IBCNSP(START+3,OFFSET+2,"Verification Phone: "\$P(IBCNS13,"^",4)) D SET^IBCNSP(START+4,OFFSET+2,"Precert Comp. Name: "\$P(\$G(^DIC(36,\$P(IBCNS13,"^",9),0)),"^",1)) D SET^IBCNSP(START+5,OFFSET+7,"Precert Phone: "\$\$_\$PHONE(IBCNS13)) I +IBCNS3=2 D SET^IBCNSP(START+6,OFFSET,"Max # Test Bills/Day: "\$P(IBCNS3,U,6)) ; S START=11,OFFSET=2 D SET^IBCNSP(START,OFFSET+28," EDI Parameters ",IORVON,IORVOFF) D SET^IBCNSP(START+1,OFFSET+13,"Transmit?: "\$S(IBCNS3="":YES-LIVE",+IBCNS3=1:"YES- LIVE",+IBCNS3=2:"TEST ONLY",1:"NO")) D SET^IBCNSP(START+2,OFFSET+1,"Inst Payer Primary ID: "\$P(IBCNS3,U,4)) D SET^IBCNSP(START+3,OFFSET,"Inst Payer Sec ID Qual: "\$\$_\$GET1^DIQ(36,+IBCNS,6.01)) D SET^IBCNSP(START+4,OFFSET+5,"Inst Payer Sec ID: "\$\$_\$GET1^DIQ(36,+IBCNS,6.02)) D SET^IBCNSP(START+5,OFFSET,"Inst Payer Sec ID Qual: "\$\$_\$GET1^DIQ(36,+IBCNS,6.03)) D SET^IBCNSP(START+6,OFFSET+5,"Inst Payer Sec ID: "\$\$_\$GET1^DIQ(36,+IBCNS,6.04)) D SET^IBCNSP(START+7,OFFSET+12,"Bin Number: "\$P(\$G(^DIC(36,+IBCNS,3)),"^",3)) ; ; S OFFSET=41 D SET^IBCNSP(START+1,OFFSET+8," Insurance Type: "\$\$_\$EXPAND^IBTRE(36,3.09,\$P(IBCNS3,U,9))) D SET^IBCNSP(START+2,OFFSET+1," Prof Payer Primary ID: "\$P(IBCNS3,U,2)) D SET^IBCNSP(START+3,OFFSET," Prof Payer Sec ID Qual: "\$\$_\$GET1^DIQ(36,+IBCNS,6.05)) D SET^IBCNSP(START+4,OFFSET+5," Prof Payer Sec ID: "\$\$_\$GET1^DIQ(36,+IBCNS,6.06)) D SET^IBCNSP(START+5,OFFSET," Prof Payer Sec ID Qual: "\$\$_\$GET1^DIQ(36,+IBCNS,6.07)) D SET^IBCNSP(START+6,OFFSET+5," Prof Payer Sec ID: "\$\$_\$GET1^DIQ(36,+IBCNS,6.08)) ;IB*2.0*432/TAZ Added fields 6.09 and 6.1 D SET^IBCNSP(START+7,OFFSET-3," Prnt Sec/Tert Auto Claims: "\$\$_\$GET1^DIQ(36,+IBCNS,6.09)) D SET^IBCNSP(START+8,OFFSET-5," Prnt Med Sec Claims w/o MRA: "\$\$_\$GET1^DIQ(36,+IBCNS,6.1)) Q ; . . . </pre> |

6.2.2.4.3. Unique Record(s) for INSURANCE COMPANY FILE (#36)

IB DATA ELEMENT : EDI ID NUMBER - PROF

| Field Name(s) | Current Value | New Value |
|---------------------------------|--|---|
| Requirement Traceability Matrix | 2.6.4.1, 2.6.4.2 | |
| Base File | 36 | |
| .01 Name | EDI ID NUMBER – PROF | |
| .02 Security Level | N | |
| .03 Type of Element | FREE TEXT | |
| .04 Element Category | 3.02 | |
| .05 Input Transform | K:\$L(X)>30!(\$L(X)<1) X I \$D(X) K:'\$\$EDIKEY^IBCNSC X | Need to add logic to prevent the entering of any PRNT values from within Insurance Company Entry/Edit. |
| 3 Description | This is the ID number used to identify the insurance company for professional claim transmissions. | This is the ID number used to identify the Payer on professional claim transmissions. SPRNT, HPRNT, PPRNT, IPRNT are not a valid Payer IDs. |

IB DATA ELEMENT : EDI ID NUMBER - INST

| Field Name(s) | Current Value | New Value |
|---------------------------------|---|--|
| Requirement Traceability Matrix | 2.6.4.1, 2.6.4.2 | |
| Base File | 36 | |
| .01 Name | EDI ID NUMBER – INST | |
| .02 Security Level | N | |
| .03 Type of Element | FREE TEXT | |
| .04 Element Category | 3.04 | |
| .05 Input Transform | K:\$L(X)>30!(\$L(X)<1) X I \$D(X) K:'\$\$EDIKEY^IBCNSC X | Need to add logic to prevent the entering of any PRNT values from within Insurance Company Entry/Edit. |
| 3 Description | This is the id number used to identify the insurance company for institutional claim transmissions. | This is the ID number used to identify the Payer on institutional claim transmissions. HPRNT, SPRNT, IPRNT, PPRNT are not a valid Payer IDs. |

6.2.2.5. System Feature: MRA Management Worklist (MRW)

This section covers the design for the Functional Requirements listed in Section 2.6.5 of the RSD.

6.2.2.5.1. Functional Requirements: MRA Management Worklist (MRW)

| SRS REQ ID | REQ Title | Comments / Notes |
|------------|---------------------------------------|------------------|
| 2.6.5.1 | Display Message Storage Errors in MRW | |

6.2.2.5.2. Routines (Entry Points)

| | | |
|--|---|--|
| Routine Name | IBCEOB | |
| Enhancement Category | <input type="checkbox"/> New <input checked="" type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change | |
| Requirement Traceability Matrix | 2.6.5.1 | |
| Related Options | MRA Management Worklist [MRW] | |
| Related Routines | Routines "Called By" | Routines "Called" |
| | ^IBCEOB0, ^IBCEOB1, ^IBCESRV3 | ^IBCESRV2, ^IBCEM, ^DILF, ^IBCEOB1, ^IBCEU, ^DIE, ^VASITE, ^IBCEF, ^IBCEFUNC, ^IBCEOB01, ^IBCEOB00, ^IBCEOB0, ^XMRENT, ^IBCEMU1, ^DICN, ^IBJTU4, ^PRCAFN, ^BCNSBL2, ^BCAPP |
| Data Dictionary (DD) References | | |
| Related Protocols | | |
| Related Integration Control Registrations (ICRs) | | |
| Data Passing | <input type="checkbox"/> Input <input type="checkbox"/> Output Reference <input type="checkbox"/> Both <input type="checkbox"/> Global Reference <input type="checkbox"/> Local | |
| Input Attribute Name and Definition | | |
| Output Attribute Name and Definition | | |
| Current Logic | | |
| ; HDR ; -- header code | | |

| | |
|---|---------------|
| Routine Name | IBCEOB |
| Q ; | |
| Modified Logic (Changes are in bold) | |
| ; HDR ; -- header code I '\$G(IBMRRANOT) S VALMSG="!=Data Mismatch/MSE Enter ?? for more actions" Q ; | |

| | | | | |
|--|---|--|--|--|
| Routine Name | IBCECOB1 | | | |
| Enhancement Category | <input type="checkbox"/> New | <input checked="" type="checkbox"/> Modify | <input type="checkbox"/> Delete | <input type="checkbox"/> No Change |
| Requirement Traceability Matrix | 2.6.5.1 | | | |
| Related Options | MRA Management Worklist [MRW] | | | |
| Related Routines | Routines “Called By” | | Routines “Called” | |
| | ^IBCAPR, ^IBCAPR2, ^IBCECOB, ^IBCECOB2, ^IBCECOB5, ^IBCECOB6, ^IBCECSA6, ^IBCEMQC, ^IBCEMU2, ^IBCEMU4, ^IBCEU1, ^IBCNSBL2, ^IBJTBA1 | | ^VALM10, ^IBCAPP2, ^IBCEMU4, ^IBEFUNC, ^IBCEF, ^XLFDT, ^IBCEF4, ^PRCAFN, ^IBCEU0, ^DILFD, ^XLFSTR, ^VALM1, ^IBOUTL, ^%ZISS, ^IBCECOB2, ^DIQ, ^DID, ^IBCAPP, ^DIE | |
| Data Dictionary (DD) References | | | | |
| Related Protocols | | | | |
| Related Integration Control Registrations (ICRs) | | | | |
| Data Passing | <input type="checkbox"/> Input | <input type="checkbox"/> Output Reference | <input type="checkbox"/> Both | <input type="checkbox"/> Global Reference <input type="checkbox"/> Local |
| Input Attribute Name and Definition | | | | |
| Output Attribute Name and Definition | | | | |
| Current Logic | | | | |
| IBCECOB1 ;ALB/CXW - IB COB MANAGEMENT SCREEN/REPORT ;14-JUN-99 ;;2.0;INTEGRATED BILLING;**137,155,288,348,377,417,432,447**;;21-MAR-94;Build 80 ;;Per VHA Directive 2004-038, this routine should not be modified. | | | | |

| Routine Name | IBCECOB1 |
|--------------|---|
| | <pre> ; ; IBMRANOT = 1 when dealing with the COB Management Worklist. ; It is set by the entry action in the option file. ; BLD ; Build list entrypoint N I,IBFND,IBB,IBIFN,IB364,IBDA1,IBDTN,IBDA,IBDAY,IBHIS,IBNDS,IBEUT,IBAPY,IBOAM,IBDT,IBMUT,IB BPY,IBINS,IBNDM,IBQ,IBNDI1,IBNDI2,IBNDI3,Z,Z0,IBSEQ,IB3611,IBINS1,IBINS2,IBEXPY,IBNBAL,IBPTR SP,IBAMT,IBMRACNT,IBPTNM,IBSRVC,IBPY,IBB364 N IBEOBREV,IBDENDUP,EOBTYPE K ^TMP("IBCECOB",\$J),^TMP("IBCECOB1",\$J),^TMP("IBCOBST",\$J),^TMP("IBCOBSTX",\$J) D CLEAN^VALM10 ; kill data and video control arrays S VALMCNT=0,IBHIS="" ; IB*2.0*432 IF not MRA, use new CAP index on 399 file D:\$G(IBMRANOT)=1 CAP^IBCAPP2 ; since 0 is a valid Review Status, init w/null S IBEOBREV="" ; get EOB's w/Review Status of 0, 1, 1.5 or 2; If 3 or higher, not needed I \$G(IBMRANOT)=1 F S IBEOBREV=\$O(^IBM(361.1,"AMRA",1,IBEOBREV)) Q:IBEOBREV="" Q:IBEOBREV>2 D . S IBDA="A" F S IBDA=\$O(^IBM(361.1,"AMRA",1,IBEOBREV,IBDA),-1) Q:'IBDA D BLD1 ; no data accumulated I \$O(^TMP("IBCOBST",\$J,""))="" D NMAT Q ; display accumulated data D SCRN Q BLD1 ; I '\$\$ELIG(IBDA) Q S IBDENDUP=\$\$DENDUP^IBCEMU4(IBDA,\$G(IBMRANOT)) I '\$G(IBMRADUP),IBDENDUP Q ; don't include denied MRAs/EOBs for Duplicate Claim/Service S IB3611=\$G(^IBM(361.1,IBDA,0)) S IBIFN=+IB3611,IB364=\$P(IB3611,U,19),IBDT=+\$P(IB3611,U,6) I \$D(^TMP("IBCOBSTX",\$J,IBIFN)) Q ;show each bill once on the worklist S IBB=\$G(^DGCR(399,IBIFN,0)) S IBNDS=\$G(^DGCR(399,IBIFN,"S")),IBNDI1=\$G(^("I1")),IBNDI2=\$G(^("I2")),IBNDI3=\$G(^("I3")),IBNDM=\$ G(^("M")) S IBMUT=+\$P(IBNDS,U,8),IBEUT=+\$P(IBNDS,U,2) S IBINS="",IBSEQ=\$P(IB3611,U,15) F I=1:1:3 S Z="IBNDI"_I I @Z D . N Q . S Q=(IBSEQ=I) . I Q S IBINS1=+@Z_U_\$P(\$G(^DIC(36,+@Z,0)),U) . S IBINS=IBINS_\$S(IBINS="":",1:",")_ \$P(\$G(^DIC(36,+@Z,0)),U) ; Get the payer/insurance company that comes after Medicare WNR ; If WNR is Primary, get the secondary ins. co. ; If WNR is secondary, get the tertiary ins. co. D I \$P(\$G(IBINS2),U,2)="" S \$P(IBINS2,U,2)="UNKNOWN" . I \$\$WNRBILL^IBEFUNC(IBIFN,1) S IBINS2=+IBNDI2_U_\$P(\$G(^DIC(36,+IBNDI2,0)),U) Q . S IBINS2=+IBNDI3_U_\$P(\$G(^DIC(36,+IBNDI3,0)),U) S IBFND=0 ; biller entry not ALL and no biller, then get entered/edited by user </pre> |

| Routine Name | IBCECOB1 |
|--|----------|
| <pre> I \$D(^TMP("IBBIL",\$J)) D Q:'IBFND . S IBFND=\$S(\$D(^TMP("IBBIL",\$J,IBMUT)):IBMUT,\$D(^TMP("IBBIL",\$J,IBEUT)):IBEUT,1:0) S Z=\$S(IBFND:IBFND,IBMUT:IBMUT,1:IBEUT) S IBMUT=\$P(\$G(^VA(200,+Z,0)),U)_"~"_Z S:\$P(IBMUT,"~",2) IBMUT="UNKNOWN~0" S IBBPY=+\$COBN^IBCEF(IBIFN),IBQ=1 ;IBQ;1=EOB without subsequent insurer,0=COB,2=0 balance D ;I IBQ Q . ;Check for no reimbursable subsequent insurance . F I=IBBPY+1:1:3 D Q:'IBQ .. S Z="IBNDI"_I,Z=\$G(@Z) .. I \$P(\$G(^DIC(36,+Z,0)),U,2)="N" S IBQ=0 Q . ;Check if next ins doesn't exist or next bill# already created . S Z="IBNDI"_(IBBPY+1),Z=\$G(@Z) . I Z,\$P(\$G(^DGCR(399,IBIFN,"M1")),U,5+IBBPY) S IBQ=0 ; ; Days since transmission of latest bill in COB - IBDAY S IBDAY=+\$P(\$G(^DGCR(399,IBIFN,"TX")),U,2) I IBDAY S IBDAY=\$\$FMDIFF^XLFD(T,IBDAY,1) ; if no Last Electronic Extract Date on file 399, get it from file 364 I 'IBDAY D I IBDAY S IBDAY=\$\$FMDIFF^XLFD(T,IBDAY,1) ;calc. the difference . S IBB364=\$\$LAST364^IBCEF4(IBIFN) I IBB364'="" S IBDAY=+\$P(\$P(\$G(^IBA(364,IBB364,0)),U,4),".",1) ; S IBAPY=\$\$TPR^PRCAFN(IBIFN) ; payment on this bill from A/R S IBEXPY=+\$G(^IBM(361.1,IBDA,1)) ; payer paid amount ; IB*2.0*447 add excess indicator to MRW screen and adjust calcs to include percentages S IBPTRSP=\$S(\$\$MS EDT^IBCEMU4(IBIFN)'="":\$\$MSPRE^IBCEMU4(IBIFN,1),1:\$\$PREOBTOT^IBCEU0(IBIFN)) ; patient resp. function S IBPY=\$S(IBAPY:IBAPY,1:IBEXPY) S IBOAM=+\$G(^DGCR(399,IBIFN,"U1")) ; total charges for bill S IBNBAL=IBOAM-IBPY I IBNBAL'>0 S IBQ=2 S IBPTNM=\$P(\$G(^DPT(+ \$P(\$G(^DGCR(399,IBIFN,0)),U,2,0)),U) I IBPTNM="" S IBPTNM="UNKNOWN" S IBSRVC=\$P(\$G(^DGCR(399,IBIFN,"U")),U) S Z0=\$S(IBSRT="B":IBMUT,IBSRT="D":- IBDAY,IBSRT="I":\$P(IBINS2,U,2)_"~"_\$P(IBINS2,U),IBSRT="M":\$\$EXTERNAL^DILFD(361.1,.13,"",\$P(IB3611,"^",13)),IBSRT="R":-IBPTRSP,IBSRT="P":IBPTNM,IBSRT="S":+IBSRVC,1:+IBDT) S:((IBSRT="M")&(Z0="")) Z0="UNKNOWN" ;USE UNKNOWN IF NOT SET - BI;IB*2.0*432 S ^TMP("IBCOBST",\$J,Z0,IBIFN)=IBSRVC_U_IBOAM_U_IBAPY_U_\$S(IBNBAL>0:IBNBAL,1:0)_U_\$P(IBB,U,5)_U_\$P(IBB,U,19)_U_IBBPY_U_\$P(IBMUT,"~")_U_IBINS_U_IBDA_U_\$HIS(IBIFN)_U_IBDAY_U_IB DT_U_IBQ_U_IB364_U_IBSEQ_U_IBEXPY_U_IBPTRSP S ^TMP("IBCOBST",\$J,Z0,IBIFN,1)=\$\$EXTERNAL^DILFD(361.1,.13,"",\$P(IB3611,"^",13))_"_\$\$FMTE^XLFD(\$P(\$P(IB3611,"^",6),".")_"^"_\$P(IB3611,"^",16) S ^TMP("IBCOBSTX",\$J,IBIFN)=IBDA ;keep track of compiled IBIFN's ; ; Save some data when there are multiple MRA's on file for this bill S IBMRACNT=\$\$MRACNT^IBCEMU1(IBIFN,\$G(IBMRA NOT)) ;WCJ IB*2.0*432 I IBMRACNT>1 S \$P(^TMP("IBCOBST",\$J,Z0,IBIFN,1),U,1)="Multiple "_\$S(\$G(IBMRA NOT):"EOBs",1:"MRA's")_" on file" ;WCJ IB*2.0*432 S \$P(^TMP("IBCOBST",\$J,Z0,IBIFN,1),U,3)=IBMRACNT S \$P(^TMP("IBCOBST",\$J,Z0,IBIFN,1),U,4)=IBDENDUP </pre> | |

| Routine Name | IBCECOB1 |
|--|----------|
| <pre> Q ; HIS(IBIFN) ; COB history N A,B,IBST,IBBIL,IBHIS S IBHIS="",A=0 F S A=\$O(^IBM(361.1,"ABS",IBIFN,A)) Q:'A S B=0 F S B=\$O(^IBM(361.1,"ABS",IBIFN,A,B)) Q:'B D . S IBST=\$P(\$G(^IBM(361.1,B,0)),U,4),IBBIL=\$P(\$G(^DGCR(399,IBIFN,"M1")),U,4+A) ;WCJ IB*2.0*432 added \$G . Q:IBBIL="" . S IBHIS=IBHIS_\$S(IBHIS="":",1:",";")_\$S(A=1:"PRIMARY",A=2:"SECONDARY",1:"TERTIARY")_" "\$S(IBST:"MRA",1:"EOB")_" RECEIVED - "_IBBIL Q IBHIS ; NMAT ;No COB list S VALMCNT=2,IBCNT=2 S ^TMP("IBCECOB",\$J,1,0)=" " S ^TMP("IBCECOB",\$J,2,0)=" No "\$S(\$G(IBMRA NOT)=1:"EOB's",1:"MRA's")_" Matching Selection Criteria Were Found" Q ; SCRN ; N IBX,IBCNT,IBIFN,IBDA,IB,X,IBS1,IBPAT,Z,IBK,IBFORM S IBCNT=0 S IBS1=\$S(IBSRT="B": "BILLER",IBSRT="D": "Days Since Last Transmission",IBSRT="L": "Date Last "\$S(\$G(IBMRA NOT): "EOB",1:"MRA")_" Received",IBSRT="I": "SECONDARY INSURANCE COMPANY",IBSRT="M": "\$S(\$G(IBMRA NOT): "EOB",1:"MRA")_" Status",1:"") S IBX="" F S IBX=\$O(^TMP("IBCOBST",\$J,IBX)) Q:IBX="" D . I IBSRT="B"!(IBSRT="I")!(IBSRT="M") D .. D:IBCNT SET("",IBCNT+1) .. D SET(IBS1_"": "_\$P(IBX,"~"),IBCNT+1) . S IBIFN=0 F S IBIFN=\$O(^TMP("IBCOBST",\$J,IBX,IBIFN)) Q:'IBIFN D .. S IB=\$G(^TMP("IBCOBST",\$J,IBX,IBIFN)) .. S Z=\$G(^DPT(+ \$P(\$G(^DGCR(399,IBIFN,0)),U,2),0)) .. S IBPAT=\$\$LJ^XL FSTR(\$E(\$P(Z,U),1,18),18," ")_" "\$E(\$P(Z,U,9),6,9) .. S IBDA=\$P(IB,U,10) ;361.1-ien .. S IBQ=\$P(IB,U,14),IB364=\$P(IB,U,15) .. ; IB*2.0*447 shorten form column to I for Instutlional and P for Professional .. ;S IBFORM=\$\$EXTERNAL^DILFD(399,.19,+\$P(IB,U,6)) .. ;I +\$P(IB,U,6)=2 S IBFORM=1500 ; for space reasons .. S IBFORM=\$\$(+\$P(IB,U,6)=2:"P",1:"I") .. S IBPTRSP=\$P(IB,U,18) .. S IBAMT=\$P(IB,U,2) .. S IBCNT=IBCNT+1 .. S X="" .. S X=\$\$SETFLD^VALM1(IBCNT,X,"NUMBER") .. S X=\$\$SETFLD^VALM1(\$\$BN1^PRCAFN(IBIFN)_\$S(\$P(\$G(^DGCR(399,IBIFN,"TX")),U,10)=1:"*",1:""),X," BILL") .. S X=\$\$SETFLD^VALM1(\$\$DAT1^IBOUTL(\$P(IB,U)),X,"SERVICE") .. S X=\$\$SETFLD^VALM1(IBPAT,X,"PATNM") </pre> | |

| Routine Name | IBCECOB1 |
|--------------|--|
| | <pre> .. S X=\$\$SETFLD^VALM1(\$\$RJ^XLFSTR(\$FN(IBPTRSP,"",2),9,""),X,"PTRESP") .. S X=\$\$SETFLD^VALM1(\$\$RJ^XLFSTR(\$FN(IBAMT,"",2),9,""),X,"IBAMT") .. S X=\$\$SETFLD^VALM1(\$E(\$\$TYPE^IBJTLA1(\$P(IB,U,5)))_"/_IBFORM,X,"BTYPE") .. D SET(X,IBCNT,IBIFN,IBDA,IBQ,IB364,IBX,IB) .. ;For R (Pt Resp), P (Pt Name) and S (Service Date) don't display sub-headers .. I "BIMRPS"[IBSRT D ... S Z=\$S(IBSRT="L":\$DAT1^IBOUTL(IBX),IBSRT="D":-IBX,1:IBX) ... D SET(" "_IBS1_"": "_Z,IBCNT) .. S X=\$\$SETSTR^VALM1("Insurers: "_\$P(IB,U,9),"",7,74) .. D SET(X,IBCNT,IBIFN,IBDA,IBQ,IB364,IBX,IB) .. ; .. ; line 3 of display: MRA status/date/split claim indicator .. S X=\$\$SETSTR^VALM1(\$S(\$G(IBMRANOT):"EOB",1:"MRA")_" Status: ", "",5,13) .. S IBK=\$G(^TMP("IBCOBST",\$J,IBX,IBIFN,1)) .. S X=\$\$SETSTR^VALM1(\$P(IBK,U,1),X,18,63) .. I \$P(IBK,U,2)=2 S X=\$\$SETSTR^VALM1("*** SPLIT CLAIM ***",X,63,18) .. I \$P(IBK,U,4),\$P(IBK,U,2)=2,\$P(IBK,U,3)=1 S X=\$\$SETSTR^VALM1("*** Denied for Duplicate **",X,54,27) .. D SET(X,IBCNT,IBIFN,IBDA,IBQ,IB364,IBX,IB) .. ; .. ; conditionally update video attributes of line 3 .. I '\$D(IOINH) D ENS^%ZISS .. ; split claim .. I \$P(IBK,U,2)=2 D CNTRL^VALM10(VALMCNT,63,17,IOINH,IOINORM) .. ; multiple mra's on file .. I \$P(IBK,U,3)>1 D CNTRL^VALM10(VALMCNT,18,22,IOINH,IOINORM) .. ; Denied for Duplicate - no split claim and single MRA only .. I \$P(IBK,U,4),\$P(IBK,U,2)=2,\$P(IBK,U,3)=1 D CNTRL^VALM10(VALMCNT,54,26,IOINH,IOINORM) .. Q Q ; SET(X,CNT,IBIFN,IBDA,IBQ,IB364,IBX,IB) ;set up list manager screen array S VALMCNT=VALMCNT+1 S ^TMP("IBCECOB",\$J,VALMCNT,0)=X S ^TMP("IBCECOB",\$J,"IDX",VALMCNT,CNT)="" I \$G(IBIFN),\$G(^TMP("IBCECOB",\$J,CNT))="" S ^TMP("IBCECOB",\$J,CNT)=VALMCNT_U_IBIFN_U_IB36 4_U_IBDA_U_IBQ_U_IBX,^TMP("IBCECOB1",\$J,CNT)=IB Q ; FTYPE(Y) ;type classification Q \$E(\$P(\$G(^IBE(353,Y,0)),U),1,8) ; PTRESPI(IBEOb) ; Function - Computes the Patient's Responsibility based on IBEOb ; of 361.1 for Claims/Bills with form type 3=UB ; Input IBEOb - a single EOB ien; Required ; Output - Function Returns IBPTRES - Patient Responsibility Amount for the EOB ; N IBPTRES,IBC,EOBADJ S IBPTRES=0,IBEOb=+\$G(IBEOb) I 'IBEOb Q IBPTRES ;PTRESPI </pre> |

| Routine Name | IBCECOB1 |
|--------------|--|
| | <pre> ; filing error Q:\$D(^IBM(361.1,IBEOB,"ERR")) IBPTRES ; ; get claim level adjustments K EOBADJ M EOBADJ=^IBM(361.1,IBEOB,10) S IBPTRES=\$\$CALCPR^IBCEU0(.EOBADJ) ; ; get line level adjustments S IBC=0 F S IBC=\$O(^IBM(361.1,IBEOB,15,IBC)) Q:'IBC D . K EOBADJ M EOBADJ=^IBM(361.1,IBEOB,15,IBC,1) . S IBPTRES=IBPTRES+\$\$CALCPR^IBCEU0(.EOBADJ) Q IBPTRES ; ELIG(IBEOB) ; Function to determine if an EOB entry is eligible for ; inclusion on the MRA or COB management worklist or not. ; IBEOB - ien into file 361.1 (required) ; Returns 1 if EOB should appear on the worklist ; Returns 0 if EOB should not appear on the worklist ; NEW ELIG,IB3611,IBIFN S ELIG=0,IBEOB=+\$G(IBEOB) S IB3611=\$G(^IBM(361.1,IBEOB,0)) I \$P(IB3611,U,4)=1 G ELIGX ; eob type must be correct for this worklist I \$P(IB3611,U,16)>2 G ELIGX ; review status must be <= 2 S IBIFN=+IB3611 I \$P(\$G(^DGCR(399,IBIFN,0)),U,13)'=2 G ELIGX ; Request MRA bill status I \$D(^IBM(361.1,IBEOB,"ERR")) G ELIGX ; filing errors ; S ELIG=1 ; this EOB is eligible for the worklist ; ELIGX ; Q ELIG ; WLRMV ; REMOVE FROM EOB WORK LIST ; IBDA(IBDA)=IBIFN^IB364^ien of 361.1^user selection seq^user name~duz# N IBIFN,IBDA,DIR,DTOUT,DUOUT,DA,DIE,DR,X D SEL^IBCECOB2(.IBDA,1) S VALMBCK="R" S IBDA=\$O(IBDA(0)) I 'IBDA Q S IBIFN=\$P(IBDA(IBDA),U,1) I 'IBIFN Q S DIR("A",1)=" " S DIR("A",2)=" Bill #: " _ \$\$GET1^DIQ(399,IBIFN_,",",.01,"E") S DIR("A",3)=" Patient: " _ \$\$GET1^DIQ(399,IBIFN_,",",.02,"E") S DIR("A",4)=" Bill Type: " _ \$\$GET1^DIQ(399,IBIFN_,",",.05,"E") S DIR("A",5)="Bill Dates: " _ \$\$GET1^DIQ(399,IBIFN_,",",151,"E") _ " - " _ \$\$GET1^DIQ(399,IBIFN_,",",152,"E") S DIR("A",6)=" " S DIR("A")="Are you sure remove this claim from the worklist? " S DIR("B")="NO" S DIR(0)="YA" D ^DIR K DIR </pre> |

| Routine Name | IBCECOB1 |
|--|----------|
| <pre> I \$D(DTOUT)!\$D(DUOUT)!Y Q ;FLAG IF USER ANSWERS YES S X=\$\$WLRMVF^IBCECOB1(IBIFN,"RM") D BLD^IBCECOB1 Q ; WLRMVF(IBIFN,METHOD,BKFL) ; ; BKFL = 1 means background process, remove NOT initiated by a user N SOC,SOCNT,SOCLIST,STATUS,IBDUZ S STATUS=0 Q:\$G(DUZ) STATUS_"^MISSING DUZ" Q:\$G(IBIFN) STATUS_"^MISSING IBIFN" Q:\$D(^DGCR(399,IBIFN)) STATUS_"^INVALID IBIFN" ; if this is a background process, set user who removed to AUTHORIZER,IB REG S IBDUZ=\$S(\$G(BKFL)=1:\$BREG^IBCAPP(),1:\$G(DUZ)) ; GET DICTIONARY SET OF CODES. ; SOC("POINTER")="RM:REMOVE ACTION;PC:PROCESS COB ACTION;CL:CLONE ACTION;" D FIELD^DID(399,38,"","POINTER","SOC") S SOC=\$G(SOC("POINTER")) F SOCNT=1:1:\$L(SOC,";")-1 S SOCLIST(\$P(\$P(SOC,";"),SOCNT),":",1))="" Q:\$D(SOCLIST(METHOD))=0 STATUS_"^INVALID METHOD" S DA=IBIFN S DIE="^DGCR(399," S DR="35////4" ; AUTO PROCESS, NO LONGER ON WORKLIST S:IBDUZ'=-1 DR=DR_"_"^37////^IBDUZ ; WHO REMOVED FROM WORKLIST S DR=DR_"_"^38////^METHOD ; METHOD USED TO REMOVE FROM WORKLIST S DR=DR_"_"^39//NOW" ; DATE STAMP WHEN REMOVED FOR WORKLIST D ^DIE S STATUS=1 Q STATUS </pre> | |
| Modified Logic (Changes are in bold) | |
| <pre> IBCECOB1 ;ALB/CXW - IB COB MANAGEMENT SCREEN/REPORT ;14-JUN-99 ;;2.0;INTEGRATED BILLING;**137,155,288,348,377,417,432,447,488**,21-MAR-94;Build 80 ;;Per VHA Directive 2004-038, this routine should not be modified. ; ; IBMRANOT = 1 when dealing with the COB Management Worklist. ; It is set by the entry action in the option file. ; BLD ; Build list entryptpoint N I,IBFND,IBB,IBIFN,IB364,IBDA1,IBDTN,IBDA,IBDAY,IBHIS,IBNDS,IBEUT,IBAPY,IBOAM,IBDT,IBMUT,IB BPY,IBINS,IBNDM,IBQ,IBNDI1,IBNDI2,IBNDI3,Z,Z0,IBSEQ,IB3611,IBINS1,IBINS2,IBEXPY,IBNBAL,IBPTR SP,IBAMT,IBMRACNT,IBPTNM,IBSRVC,IBPY,IBB364 N IBEOBREV,IBDENDUP,EOBTYPE,MSEFLG K ^TMP("IBCECOB",\$J),^TMP("IBCECOB1",\$J),^TMP("IBCOBST",\$J),^TMP("IBCOBSTX",\$J) D CLEAN^VALM10 ; kill data and video control arrays S (VALMCNT,MSEFLG)=0,IBHIS="" ; IB*2.0*432 IF not MRA, use new CAP index on 399 file </pre> | |

| Routine Name | IBCECOB1 |
|--------------|--|
| | <pre> D:\$G(IBM(RANOT)=1 CAP^IBCAPP2 ; since 0 is a valid Review Status, init w/null S IBEOBREV="" ; get EOB's w/Review Status of 0, 1, 1.5 or 2; If 3 or higher, not needed I \$G(IBM(RANOT))=1 F S IBEOBREV=\$O(^IBM(361.1,"AMRA",1,IBEOBREV)) Q:IBEOBREV="" Q:IBEOBREV>2 D . S IBDA="A" F S IBDA=\$O(^IBM(361.1,"AMRA",1,IBEOBREV,IBDA),-1) Q:'IBDA D BLD1 ; no data accumulated I \$O(^TMP("IBCOBST",\$J,""))="" D NMAT Q ; display accumulated data D SCR N Q BLD1 ; ;;;I '\$\$ELIG(IBDA) Q S MSEFLG=\$\$ELIG(IBDA) Q:'MSEFLG S IBDENDUP=\$\$DENDUP^IBCEMU4(IBDA,\$G(IBM(RANOT))) I '\$G(IBM(RADUP),IBDENDUP) Q ; don't include denied MRAs/EOBs for Duplicate Claim/Service S IB3611=\$G(^IBM(361.1,IBDA,0)) S IBIFN=+IB3611,IB364=\$P(IB3611,U,19),IBDT=+\$P(IB3611,U,6) I \$D(^TMP("IBCOBSTX",\$J,IBIFN)) Q ;show each bill once on the worklist S IBB=\$G(^DGCR(399,IBIFN,0)) S IBNDS=\$G(^DGCR(399,IBIFN,"S")),IBNDI1=\$G(^("I1")),IBNDI2=\$G(^("I2")),IBNDI3=\$G(^("I3")),IBNDM=\$ G(^("M")) S IBMUT=+\$P(IBNDS,U,8),IBEUT=+\$P(IBNDS,U,2) S IBINS="",IBSEQ=\$P(IB3611,U,15) F I=1:1:3 S Z="IBNDI"_I I @Z D . N Q . S Q=(IBSEQ=I) . I Q S IBINS1=+@Z_U_\$P(\$G(^DIC(36,+@Z,0)),U) . S IBINS=IBINS_\$S(IBINS="":",1:",")_ \$P(\$G(^DIC(36,+@Z,0)),U) ; Get the payer/insurance company that comes after Medicare WNR ; If WNR is Primary, get the secondary ins. co. ; If WNR is secondary, get the tertiary ins. co. D I \$P(\$G(IBINS2),U,2)="" S \$P(IBINS2,U,2)="UNKNOWN" . I \$WNRBILL^IBEFUNC(IBIFN,1) S IBINS2=+IBNDI2_U_\$P(\$G(^DIC(36,+IBNDI2,0)),U) Q . S IBINS2=+IBNDI3_U_\$P(\$G(^DIC(36,+IBNDI3,0)),U) S IBFND=0 ; biller entry not ALL and no biller, then get entered/edited by user I \$D(^TMP("IBBIL",\$J)) D Q:'IBFND . S IBFND=\$S(\$D(^TMP("IBBIL",\$J,IBMUT)):IBMUT,\$D(^TMP("IBBIL",\$J,IBEUT)):IBEUT,1:0) S Z=\$S(IBFND:IBFND,IBMUT:IBMUT,1:IBEUT) S IBMUT=\$P(\$G(^VA(200,+Z,0)),U)_"~"_Z S:\$P(IBMUT,"~",2) IBMUT="UNKNOWN~0" S IBBPY=+\$\$COBN^IBCEF(IBIFN),IBQ=1 ;IBQ;1=EOB without subsequent insurer,0=COB,2=0 balance D ;I IBQ Q . ;Check for no reimbursable subsequent insurance . F I=IBBPY+1:1:3 D Q:'IBQ .. S Z="IBNDI"_I,Z=\$G(@Z) .. I \$P(\$G(^DIC(36,+Z,0)),U,2)="N" S IBQ=0 Q </pre> |

| Routine Name | IBCECOB1 |
|--------------|---|
| | <pre> ; Check if next ins doesn't exist or next bill# already created . S Z="IBNDI"_(IBBPY+1),Z=\$G(@Z) . I Z,\$P(\$G(^DGCR(399,IBIFN,"M1")),U,5+IBBPY) S IBQ=0 ; ; Days since transmission of latest bill in COB - IBDAY S IBDAY=+\$P(\$G(^DGCR(399,IBIFN,"TX")),U,2) I IBDAY S IBDAY=\$\$FMDIFF^XLFD(T,IBDAY,1) ; if no Last Electronic Extract Date on file 399, get it from file 364 I 'IBDAY D I IBDAY S IBDAY=\$\$FMDIFF^XLFD(T,IBDAY,1) ; calc. the difference . S IBB364=\$\$LAST364^IBCEF4(IBIFN) I IBB364="" S IBDAY=+\$P(\$P(\$G(^IBA(364,IBB364,0)),U,4),".",1) ; S IBAPY=\$\$TPR^PRCAFN(IBIFN) ; payment on this bill from A/R S IBEXPY=+\$G(^IBM(361.1,IBDA,1)) ; payer paid amount ; IB*2.0*447 add excess indicator to MRW screen and adjust calcs to include percentages S IBPTRSP=\$S(\$\$MS EDT^IBCEMU4(IBIFN)'="" : \$\$MSPRE^IBCEMU4(IBIFN,1),1 : \$\$PREOBTOT^IBCEU0(IBIFN)) ; patient resp. function S IBPY=\$S(IBAPY:IBAPY,1:IBEXPY) S IBOAM=+\$G(^DGCR(399,IBIFN,"U1")) ; total charges for bill S IBNBAL=IBOAM-IBPY I IBNBAL>0 S IBQ=2 S IBPTNM=\$P(\$G(^DPT(+\$P(\$G(^DGCR(399,IBIFN,0)),U,2),0)),U) I IBPTNM="" S IBPTNM="UNKNOWN" S IBSRVC=\$P(\$G(^DGCR(399,IBIFN,"U")),U) S Z0=\$S(IBSRT="B":IBMUT,IBSRT="D":- IBDAY,IBSRT="I":\$P(IBINS2,U,2)_"~"_\$P(IBINS2,U),IBSRT="M":\$\$EXTERNAL^DILFD(361.1,.13,"",\$P(IB3 611,"^",13)),IBSRT="R":-IBPTRSP,IBSRT="P":IBPTNM,IBSRT="S":+IBSRVC,1:+IBDT) S:(IBSRT="M")&(Z0="") Z0="UNKNOWN" ; USE UNKNOWN IF NOT SET - BI;IB*2.0*432 I \$D(^TMP("IBCOBST",\$J,Z0,IBIFN)),\$P(^TMP("IBCOBST",\$J,Z0,IBIFN),U,19)=-1 S MSEFLG=-1 ; If a MSE was previously found for IBIFN, we want to insure that we don't ignore that by resetting the 19th piece to something else. ; S ^TMP("IBCOBST",\$J,Z0,IBIFN)=IBSRVC_U_IBOAM_U_IBAPY_U_\$S(IBNBAL>0:IBNBAL,1:0)_U_\$P(IB B,U,5)_U_\$P(IBM,U,19)_U_IBBPY_U_\$P(IBMUT,"~")_U_IBINS_U_IBDA_U_\$HIS(IBIFN)_U_IBDAY_U_IB DT_U_IBQ_U_IB364_U_IBSEQ_U_IBEXPY_U_IBPTRSP_U_MSEFLG S ^TMP("IBCOBST",\$J,Z0,IBIFN,1)=\$\$EXTERNAL^DILFD(361.1,.13,"",\$P(IB3611,"^",13))_"", "\$FMT^XLFD(\$P(\$P(IB3611,"^",6),"."))_"^"_\$P(IB3611,"^",16) S ^TMP("IBCOBSTX",\$J,IBIFN)=IBDA ; keep track of compiled IBIFN's ; ; Save some data when there are multiple MRA's on file for this bill S IBMRACNT=\$\$MRACNT^IBCEMU1(IBIFN,\$G(IBMRA NOT)) ; WCJ IB*2.0*432 I IBMRACNT>1 S \$P(^TMP("IBCOBST",\$J,Z0,IBIFN,1),U,1)="Multiple "\$S(\$G(IBMRA NOT):"EOBs",1:"MRA's")_" on file" ; WCJ IB*2.0*432 S \$P(^TMP("IBCOBST",\$J,Z0,IBIFN,1),U,3)=IBMRACNT S \$P(^TMP("IBCOBST",\$J,Z0,IBIFN,1),U,4)=IBDENDUP Q ; HIS(IBIFN) ; COB history N A,B,IBST,IBBIL,IBHIS S IBHIS="",A=0 F S A=\$O(^IBM(361.1,"ABS",IBIFN,A)) Q:'A S B=0 F S B=\$O(^IBM(361.1,"ABS",IBIFN,A,B)) Q:'B D . S IBST=\$P(\$G(^IBM(361.1,B,0)),U,4),IBBIL=\$P(\$G(^DGCR(399,IBIFN,"M1")),U,4+A) ; WCJ IB*2.0*432 added \$G </pre> |

| Routine Name | IBCECOB1 |
|--|----------|
| <pre> . Q:IBBIL="" . S IBHIS=IBHIS_\$\$(IBHIS=""::",1:");_\$\$\$(A=1:"PRIMARY",A=2:"SECONDARY",1:"TERTIARY")_ "\$\$_\$(IBST:"MRA",1:"EOB")_ " RECEIVED - "_IBBIL Q IBHIS ; NMAT ;No COB list S VALMCNT=2,IBCNT=2 S ^TMP("IBCECOB",\$J,1,0)=" " S ^TMP("IBCECOB",\$J,2,0)=" No "\$\$_\$(G(IBMTRANOT)=1:"EOB's",1:"MRA's")_ " Matching Selection Criteria Were Found" Q ; SCRN ; N IBX,IBCNT,IBIFN,IBDA,IB,X,IBS1,IBPAT,Z,IBK,IBFORM,MSEFLG S IBCNT=0 S IBS1=\$\$(IBSRT="B":"BILLER",IBSRT="D":"Days Since Last Transmission",IBSRT="L":"Date Last "\$\$_\$(G(IBMTRANOT):"EOB",1:"MRA")_ " Received",IBSRT="I":"SECONDARY INSURANCE COMPANY",IBSRT="M":\$\$\$(G(IBMTRANOT):"EOB",1:"MRA")_ " Status",1:"") S IBX="" F S IBX=\$O(^TMP("IBCOBST",\$J,IBX)) Q:IBX="" D . I IBSRT="B"!(IBSRT="I")!(IBSRT="M") D .. D:IBCNT SET("",IBCNT+1) .. D SET(IBS1_"": "_\$P(IBX,"~"),IBCNT+1) . S IBIFN=0 F S IBIFN=\$O(^TMP("IBCOBST",\$J,IBX,IBIFN)) Q:IBIFN D .. S IB=\$G(^TMP("IBCOBST",\$J,IBX,IBIFN)) .. S Z=\$G(^DPT(+P(\$G(^DGCR(399,IBIFN,0)),U,2),0)) .. S IBPAT=\$\$LJ^XLFFSTR(\$E(\$P(Z,U),1,18),18," ")_ " _\$E(\$P(Z,U,9),6,9) .. S IBDA=\$P(IB,U,10) ;361.1-ien .. S IBQ=\$P(IB,U,14),IB364=\$P(IB,U,15) .. ; IB*2.0*447 shorten form column to I for Institutional and P for Professional .. ;S IBFORM=\$\$EXTERNAL^DILFD(399,.19,+\$P(IB,U,6)) .. ;I +\$P(IB,U,6)=2 S IBFORM=1500 ; for space reasons .. S IBFORM=\$\$(+\$P(IB,U,6)=2:"P",1:"I") .. S IBPTRSP=\$P(IB,U,18) .. S MSEFLG=\$P(IB,U,19) .. S IBAMT=\$P(IB,U,2) .. S IBCNT=IBCNT+1 .. S X="" .. S X=\$\$SETFLD^VALM1(IBCNT,X,"NUMBER") .. ;;S X=\$\$SETFLD^VALM1(\$\$BN1^PRCAFN(IBIFN)_\$\$(\$P(\$G(^DGCR(399,IBIFN,"TX")),U,10)=1:"*",1:""),X,"BIL L") .. S X=\$\$SETFLD^VALM1(\$\$(MSEFLG=-1:"!",1:" ")_\$\$BN1^PRCAFN(IBIFN)_\$\$(\$P(\$G(^DGCR(399,IBIFN,"TX")),U,10)=1:"*",1:""),X,"BILL") ; per IB*2.0*488 .. S X=\$\$SETFLD^VALM1(\$\$DAT1^IBOUTL(\$P(IB,U)),X,"SERVICE") .. S X=\$\$SETFLD^VALM1(IBPAT,X,"PATNM") .. S X=\$\$SETFLD^VALM1(\$\$RJ^XLFFSTR(\$FN(IBPTRSP,"",2),9," "),X,"PTRESP") .. S X=\$\$SETFLD^VALM1(\$\$RJ^XLFFSTR(\$FN(IBAMT,"",2),9," "),X,"IBAMT") .. S X=\$\$SETFLD^VALM1(\$E(\$\$TYPE^IBJTLA1(\$P(IB,U,5)))_ "/"_IBFORM,X,"BTYPE") .. D SET(X,IBCNT,IBIFN,IBDA,IBQ,IB364,IBX,IB) .. ;For R (Pt Resp), P (Pt Name) and S (Service Date) don't display sub-headers </pre> | |

| Routine Name | IBCECOB1 |
|--|----------|
| <pre> .. I "BIMRPS"[IBSRT D .. S Z=\$\$(IBSRT="L":\$\$DAT1^IBOUTL(IBX),IBSRT="D":-IBX,1:IBX) .. D SET(" _IBS1_": "_Z,IBCNT) .. S X=\$\$SETSTR^VALM1("Insurers: " _\$P(IB,U,9),"",7,74) .. D SET(X,IBCNT,IBIFN,IBDA,IBQ,IB364,IBX,IB) .. ; .. ; line 3 of display: MRA status/date/split claim indicator .. S X=\$\$SETSTR^VALM1(\$\$(G(IBMRAOT):"EOB",1:"MRA")_ " Status: ", "",5,13) .. S IBK=\$G(^TMP("IBCOBST",\$J,IBX,IBIFN,1)) .. S X=\$\$SETSTR^VALM1(\$P(IBK,U,1),X,18,63) .. I \$P(IBK,U,2)=2 S X=\$\$SETSTR^VALM1("*** SPLIT CLAIM ***",X,63,18) .. I \$P(IBK,U,4),\$P(IBK,U,2)'=2,\$P(IBK,U,3)=1 S X=\$\$SETSTR^VALM1("*** Denied for Duplicate ***",X,54,27) .. D SET(X,IBCNT,IBIFN,IBDA,IBQ,IB364,IBX,IB) .. ; .. ; conditionally update video attributes of line 3 .. I '\$D(IOINHI) D ENS^%ZISS .. ; split claim .. I \$P(IBK,U,2)=2 D CNTRL^VALM10(VALMCNT,63,17,IOINHI,IOINORM) .. ; multiple mra's on file .. I \$P(IBK,U,3)>1 D CNTRL^VALM10(VALMCNT,18,22,IOINHI,IOINORM) .. ; Denied for Duplicate - no split claim and single MRA only .. I \$P(IBK,U,4),\$P(IBK,U,2)'=2,\$P(IBK,U,3)=1 D CNTRL^VALM10(VALMCNT,54,26,IOINHI,IOINORM) .. Q Q ; SET(X,CNT,IBIFN,IBDA,IBQ,IB364,IBX,IB) ;set up list manager screen array S VALMCNT=VALMCNT+1 S ^TMP("IBCECOB",\$J,VALMCNT,0)=X S ^TMP("IBCECOB",\$J,"IDX",VALMCNT,CNT)="" I \$G(IBIFN),\$G(^TMP("IBCECOB",\$J,CNT))="" S ^TMP("IBCECOB",\$J,CNT)=VALMCNT_U_IBIFN_U_IB36 4_U_IBDA_U_IBQ_U_IBX,^TMP("IBCECOB1",\$J,CNT)=IB Q ; FTYPE(Y) ;type classification Q \$E(\$P(\$G(^IBE(353,Y,0)),U),1,8) ; PTRESPI(IBEOb) ; Function - Computes the Patient's Responsibility based on IBEOb ; of 361.1 for Claims/Bills with form type 3=UB ; Input IBEOb - a single EOB ien; Required ; Output - Function Returns IBPTRES - Patient Responsibility Amount for the EOB ; N IBPTRES,IBC,EOBADJ S IBPTRES=0,IBEOb=+\$G(IBEOb) I 'IBEOb Q IBPTRES ;PTRESPI ; filing error Q:\$D(^IBM(361.1,IBEOb,"ERR")) IBPTRES ; ; get claim level adjustments K EOBADJ M EOBADJ=^IBM(361.1,IBEOb,10) </pre> | |

| Routine Name | IBCECOB1 |
|--------------|---|
| | <pre> S IBPTRES=\$\$CALCPR^IBCEU0(.EOBADJ) ; ; get line level adjustments S IBC=0 F S IBC=\$O(^IBM(361.1,IBEOB,15,IBC)) Q:'IBC D . K EOBAJ M EOBAJ=^IBM(361.1,IBEOB,15,IBC,1) . S IBPTRES=IBPTRES+\$\$CALCPR^IBCEU0(.EOBADJ) Q IBPTRES ; ELIG(IBEOB) ; Function to determine if an EOB entry is eligible for ; inclusion on the MRA or COB management worklist or not. ; IBEOB - ien into file 361.1 (required) ; Returns 1 if EOB should appear on the worklist ; Returns 0 if EOB should not appear on the worklist ; Returns -1 if EOB contains Message Storage Errors ; NEW ELIG,IB3611,IBIFN S ELIG=0,IBEOB=+\$G(IBEOB) S IB3611=\$G(^IBM(361.1,IBEOB,0)) I \$P(IB3611,U,4)=1 G ELIGX ; eob type must be correct for this worklist I \$P(IB3611,U,16)>2 G ELIGX ; review status must be <= 2 S IBIFN=+IB3611 I \$P(\$G(^DGCR(399,IBIFN,0)),U,13)=2 G ELIGX ; Request MRA bill status I \$D(^IBM(361.1,IBEOB,"ERR")) S ELIG=\$S('\$G(IBMRRANOT):-1,1:ELIG) G ELIGX ; filing errors - contains Message Storage Errors ; S ELIG=1 ; this EOB is eligible for the worklist ; ELIGX ; Q ELIG ; WLRMV ; REMOVE FROM EOB WORK LIST ; IBDA(IBDA)=IBIFN^IB364^ien of 361.1^user selection seq^user name~duz# N IBIFN,IBDA,DIR,DTOUT,DUOUT,DA,DIE,DR,X D SEL^IBCECOB2(.IBDA,1) S VALMBCK="R" S IBDA=\$O(IBDA(0)) I 'IBDA Q S IBIFN=\$P(IBDA(IBDA),U,1) I 'IBIFN Q S DIR("A",1)=" " S DIR("A",2)=" Bill #: " _ \$\$GET1^DIQ(399,IBIFN_,",",.01,"E") S DIR("A",3)=" Patient: " _ \$\$GET1^DIQ(399,IBIFN_,",",.02,"E") S DIR("A",4)=" Bill Type: " _ \$\$GET1^DIQ(399,IBIFN_,",",.05,"E") S DIR("A",5)="Bill Dates: " _ \$\$GET1^DIQ(399,IBIFN_,",",151,"E") _ " - " _ \$\$GET1^DIQ(399,IBIFN_,",",152,"E") S DIR("A",6)=" " S DIR("A")="Are you sure remove this claim from the worklist? " S DIR("B")="NO" S DIR(0)="YA" D ^DIR K DIR I \$D(DTOUT)!\$D(DUOUT)!Y Q ;FLAG IF USER ANSWERS YES S X=\$\$WLRMV^IBCECOB1(IBIFN,"RM") </pre> |

| Routine Name | IBCECOB1 |
|--|----------|
| D BLD^IBCECOB1 Q ; WLRMVF(IBIFN,METHOD,BKFL) ; ; BFKL = 1 means background process, remove NOT initiated by a user N SOC,SOCNT,SOCLIST,STATUS,IBDUZ S STATUS=0 Q:\$G(DUZ) STATUS_ "^MISSING DUZ" Q:\$G(IBIFN) STATUS_ "^MISSING IBIFN" Q:\$D(^DGCR(399,IBIFN)) STATUS_ "^INVALID IBIFN" ; if this is a background process, set user who removed to AUTHORIZER,IB REG S IBDUZ=\$S(\$G(BKFL)=1:\$IBREG^IBCAPP(),1:\$G(DUZ)) ; GET DICTIONARY SET OF CODES. ; SOC("POINTER")="RM:REMOVE ACTION;PC:PROCESS COB ACTION;CL:CLONE ACTION;" D FIELD^DID(399,38,"","POINTER","SOC") S SOC=\$G(SOC("POINTER")) F SOCNT=1:1:\$L(SOC,";")-1 S SOCLIST(\$P(\$P(SOC,";"),SOCNT),":",1))="" Q:\$D(SOCLIST(METHOD))=0 STATUS_ "^INVALID METHOD" S DA=IBIFN S DIE="^DGCR(399," S DR="35////4" ; AUTO PROCESS, NO LONGER ON WORKLIST S:IBDUZ'=-1 DR=DR_"_"^"37////" _IBDUZ ; WHO REMOVED FROM WORKLIST S DR=DR_"_"^"38////" _METHOD ; METHOD USED TO REMOVE FROM WORKLIST S DR=DR_"_"^"39///NOW" ; DATE STAMP WHEN REMOVED FOR WORKLIST D ^DIE S STATUS=1 Q STATUS | |

| Routine Name | IBCECOB2 | |
|---------------------------------|---|--|
| Enhancement Category | <input type="checkbox"/> New <input type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change | |
| Requirement Traceability Matrix | 2.6.5.1 | |
| Related Options | MRA Management Worklist [MRW] | |
| Related Routines | Routines "Called By" | Routines "Called" |
| | ^IBCAPP1, ^IBCECOB, ^IBCECOB1, ^IBCECOB5, ^IBCECOB6, ^IBCECSA4, ^IBCEMSR3, ^IBCEMU4, ^IBCEXTR2 | ^VALM, ^IBCECSA, ^IBCECSA4, ^VALM1, ^IBCEM03, ^IBCEMRAA, ^BCB1, ^XUSRB, ^IBCEU0, ^IBCEM3, ^IBCECOB1, ^IBCCC, ^IBCEM, ^IBCEF, ^IBCCCB0, ^IBCCCB, ^IBCB2, ^IBCB1, ^%ZISS, ^IBCEFG4, ^IBCEMU4, ^IBCEU1, ^IBJTU4, ^DIR |
| Data Dictionary (DD) | | |

| | |
|---|---|
| Routine Name | IBCECOB2 |
| References | |
| Related Protocols | |
| Related Integration Control Registrations (ICRs) | |
| Data Passing | <input type="checkbox"/> Input <input type="checkbox"/> Output Reference <input type="checkbox"/> Both <input type="checkbox"/> Global Reference <input type="checkbox"/> Local |
| Input Attribute Name and Definition | |
| Output Attribute Name and Definition | |
| Current Logic | |
| IBCECOB2 ;ALB/CXW - IB COB MANAGEMENT SCREEN ;16-JUN-1999 ;;2.0;INTEGRATED BILLING; **137,155,433,432,447** ;21-MAR-1994;Build 80 ;;Per VHA Directive 2004-038, this routine should not be modified. ; EDI ;history detail display N IBIFN,IBDA D SEL(.IBDA,1) S IBDA=+\$O(IBDA(0)),IBIFN=+\$G(IBDA(IBDA)) D EDI1(IBIFN) S VALMBCK="R" Q ; EDI1(IBIFN) ; N DFN Q:'IBIFN S DFN=\$P(\$G(^DGCR(399,IBIFN,0)),U,2) D EN^VALM("IBJT EDI STATUS") K:\$D(IBFASTXT) IBFASTXT Q ; EDI2(IBIFN) ; N DFN Q:'IBIFN S DFN=\$P(\$G(^DGCR(399,IBIFN,0)),U,2) D EN^VALM("IBJT EDI STATUS ALONE") K:\$D(IBFASTXT) IBFASTXT Q ; CSA ;claims status awaiting resolution N IBDAX D EN^IBCECSA I \$D(IBFASTXT) K IBFASTXT S VALMBCK="R" Q ; | |

| Routine Name | IBCECOB2 |
|--------------|---|
| | <pre> RVEOB ;Review EOB D FULL^VALM1 W ! N IBDA,IBIFN,IBCMT,IBSEL D SEL(.IBDA,1) S IBSEL=+\$O(IBDA(0)) S IBDA=\$G(IBDA(IBSEL)) S IBIFN=\$P(IBDA,U),IBDA=\$P(IBDA,U,3) I 'IBIFN G VEOBQ S IBCMT=\$G(^TMP("IBCECOB1",\$J,IBSEL)) I IBCMT'="" D EN^VALM("IBCEM MRA REVIEW") VEOBQ K ^TMP("IBCECOC",\$J) S VALMBCK="R" Q ; TPJI ;Third Party joint Inquiry N IBDA,IBIFN D SEL(.IBDA,1) S IBDA=+\$O(IBDA(0)),IBIFN=+\$G(IBDA(IBDA)) I IBDA="" G TPJIQ D TPJI1(IBIFN) TPJIQ S VALMBCK="R" Q ; TPJI1(IBIFN) ; N DFN,IBNOTPJI Q:'IBIFN S DFN=\$P(\$G(^DGCR(399,IBIFN,0)),U,2),IBNOTPJI=1 D EN^VALM("IBJT CLAIM INFO") K:\$D(IBFASTXT) IBFASTXT Q ; PBILL ;Print bill N IBIFN,IBDA,IBRESUB D SEL(.IBDA,1) S IBDA=\$O(IBDA(0)),IBIFN=+\$G(IBDA(+IBDA)) I IBDA="" G PBOUT S IBRESUB=\$\$RESUB^IBCECSA4(IBIFN,1,"P") I IBRESUB>0 W !,*7,"This is not a transmittable bill or review not needed" D PAUSE^VALM1 G PBOUT I IBRESUB=2 D G PBOUT . N IB364 . S IB364=+\$P(\$G(IBDA(IBDA)),U,2) . D PRINT1^IBCEM03(IBIFN,.IBDA,IB364) D PBILL1(IBIFN) PBOUT S VALMBCK="R" Q ; PMRA ;Print MRA N IBIFN,IBDA D SEL(.IBDA,1) S IBDA=\$O(IBDA(0)),IBIFN=+\$G(IBDA(+IBDA)) </pre> |

| Routine Name | IBCECOB2 |
|--------------|--|
| | <pre> G:'IBIFN PRMQ D MRA^IBCEMRAA(.IBIFN) PRMQ S VALMBCK="R" Q PBILL1(IBIFN) ; N IBAC1,IBAC,DFN Q:'IBIFN S DFN=\$P(\$G(^DGCR(399,IBIFN,0)),U,2) S IBAC=4,IBAC1=1 D 4^IBCB1 D FULL^VALM1,PAUSE^VALM1 Q ; CANCEL ;Cancel bill ; IBDA(IBDA)=IBIFN^IB364^ien of 361.1^user selection seq^user name~duz# ; N IBIFN,IBDA,IB364,IBEObIFN,X,IBDENCT ; ; Check for security key I '\$\$CHK^XUSRB("IB AUTHORIZE") D G CANCELQ . D FULL^VALM1 S VALMBCK="R" . W !!?5,"You don't hold the proper security key to access this function." . W !?5,"The necessary key is IB AUTHORIZE. Please see your manager." . D PAUSE^VALM1 . Q ; D SEL(.IBDA,1) S IBDA=\$O(IBDA(0)),IBIFN=+\$G(IBDA(+IBDA)),IB364=\$P(\$G(IBDA(+IBDA)),U,2) S IBEObIFN=\$P(\$G(IBDA(+IBDA)),U,3) ; ; IB*2.0*432 - if not mra, only allow cancel of denied claims. If no EOB, check AR status instead I 'IBEObIFN,\$G(IBMRA NOT)=1,\$P(\$\$ARSTATA^IBJTU4(IBIFN),U)="COLLECTED/CLOSED" D G CANCE LQ . D FULL^VALM1 S VALMBCK="R" . W !!,*7,"You can only cancel denied claims. This claim is in a COLLECTED/CLOSED status" . W !,"Use Remove Action to remove claim from this worklist." . D PAUSE^VALM1 . Q ; ; IB*2.0*432 - if not mra, only allow cancel of claims with multiple EOBs if none have processed. I \$G(IBMRA NOT)=1,\$\$DENCHK(IBIFN,.IBDENCT),\$G(IBDENCT)>1 D G CANCELQ . D FULL^VALM1 S VALMBCK="R" . W !!,*7,"Multiple EOBs exist for this claim and at least one has EOB status of PROCESSED." . W !,"Use Remove Action to remove claim from this worklist." . D PAUSE^VALM1 . Q ; ; IB*2.0*432 - if not mra, only allow cancel of denied claims I IBEObIFN,\$G(IBMRA NOT)=1,\$P(\$G(^IBM(361.1,IBEObIFN,0)),U,13)'=2 D G CANCELQ . D FULL^VALM1 S VALMBCK="R" </pre> |

| Routine Name | IBCECOB2 |
|--------------|---|
| | <pre> . W !!?5,*7,"You can only cancel denied claims." . D PAUSE^VALM1 . Q ; I IBDA D . I '\$\$LOCK^IBCEU0(361.1,IBEOBIFN) Q . D CANCEL^IBCEM3(.IBDA,IBIFN,IB364) . D UNLOCK^IBCEU0(361.1,IBEOBIFN) S VALMBCK="R" ; ; for non-MRA claims cancelled from worklist, set field 38 I \$G(IBMRANOT)=1,\$P(\$G(^DGCR(399,IBIFN,0)),U,13)=7 S X=\$\$WLRMV^IBCECOB1(\$S(\$G(IBIFN)'="" :IBIFN,1:+\$G(IBDA(IBDA))),,"CA") I \$G(IBDA)'="" D BLD^IBCECOB1 CANCELQ Q ; CRD ; Correct Rejected/Denied claim protocol action N IBCNCRD S IBCNCRD=1 CLONE ; 'Copy/cancel bill' protocol action N IBDA,IBQ,IBEOBIFN,IBKEY,X,IBDENCT ; ; Check for security key ;I '\$\$CHK^XUSRB("IB AUTHORIZE") D G CLONEQ S IBKEY=\$S(\$G(IBCNCRD)=1:"IB AUTHORIZE",1:"IB CLON") I '\$\$CHK^XUSRB(IBKEY) D G CLONEQ . D FULL^VALM1 S VALMBCK="R" . ;W !!?5,"You don't hold the proper security key to access this function." . ;W !?5,"The necessary key is IB AUTHORIZE. Please see your manager." . W !!?5,"You must hold the "_IBKEY_" security key to access this function." . W !?5,"Please see your manager." . D PAUSE^VALM1 . Q ; D SEL(.IBDA,1) S IBDA=\$O(IBDA("")) I IBDA="" G CLONEQ ; ; IB*2.0*432 - if not mra, only allow cancel of claims with multiple EOBs if none have processed. I \$G(IBMRANOT)=1,\$\$DENCHK(+IBDA(IBDA),.IBDENCT),\$G(IBDENCT)>1 D G CANCELQ . D FULL^VALM1 S VALMBCK="R" . W !!,*7,"Multiple EOBs exist for this claim and at least one has EOB status of PROCESSED." . W !,"Use Remove Action to remove claim from this worklist." . D PAUSE^VALM1 . Q ; S IBEOBIFN=\$P(\$G(IBDA(+IBDA)),U,3) I '\$\$LOCK^IBCEU0(361.1,IBEOBIFN) G CLONEQ D COPYCLON(+\$G(IBDA(IBDA)), \$P(\$G(IBDA(+IBDA)),U,2),.IBQ) D UNLOCK^IBCEU0(361.1,IBEOBIFN) </pre> |

| Routine Name | IBCECOB2 |
|--------------|---|
| | <pre> ; ; for non-MRA claims cloned or corrected from worklist, set field 38 I \$G(IBMRANOT)=1,\$G(IBQ)'="" S X=\$\$WLRMV^IBCECOB1(+\$G(IBDA(IBDA)),\$S(\$G(IBCNCRD)=1:"CR",1:"CL")) ; ; CLONEQ ; S VALMBCK="R" D:\$G(IBQ)'="" BLD^IBCECOB1 Q ; ; COPYCLON(IBIFN,IB364,IBQ) ; Generic entry point for clone a bill from EDI processing ; IBIFN = original bill ien ; IB364 = the ien of the transmission bill entry in file 364 ; IBQ = If bill is not cancelled, this is returned as null ; - pass by reference - ; ; N IBQUIT,IBCCCC,IBHV,Y,IBCAN,IBCE,IBDA,IBCNCOPY ;I '\$\$CANCK\$^IBCEM3("CC",IBIFN) S IBQ="" G CCQ I \$G(IBCNCRD)=1,\$\$CANCK\$^IBCEM3("CC",IBIFN) S IBQ="" G CCQ ; ;S IBCAN=2,IBCE("EDI")=1,Y=IBIFN,IBCCCC=0,IBHV("IBIFN")=IBIFN,IBHV("IBIFN1")="",IBCNCOPY=1 S IBCAN=2,IBCE("EDI")=1,Y=IBIFN,IBCCCC=0,IBHV("IBIFN")=IBIFN,IBHV("IBIFN1")="" I \$G(IBCNCRD)=1 S IBCNCOPY=1 D ^IBCCC I \$G(IBCNCRD)=1 D CRD^IBCCC ;D ^IBCCC S IBIFN=IBHV("IBIFN") K IBCE("EDI") S IBQ=1 I \$P(\$G(^DGCR(399,IBIFN,0)),U,13)'=7 S IBQ="" I IBHV("IBIFN1") D . N IBU . S IBU="R" . S IBNIEN=+IBHV("IBIFN1") . I "23"[\$P(\$G(^DGCR(399,+IBHV("IBIFN1"),0)),U,13) D .. W:\$G(IBCCEAUTO) !,*7,"Please note: the new bill was not AUTHORIZED.",!,"It can only be accessed now via the normal, non-EDI functions.",!,"Status of new bill is ",\$EXPAND^IBTRE(399,.13,\$P(^DGCR(399,IBHV("IBIFN1"),0),U,13)) S IBU="C" . D UPDED^IBCEM(IB364,IBU) ; ; I '\$G(IBCCEAUTO) D PAUSE^VALM1 CCQ Q ; ; PRO ; Copy for secondary/tertiary bill N VALMY,IBDA,Z,IBIFN,IBIFNH,IB364,IBCE,IBNCN ;I '\$P(\$G(^IBE(350.9,1,8)),U,12) D G PROQ I '\$P(\$G(^IBE(350.9,1,8)),U,12),\$G(IBMRANOT)'=1 D G PROQ . D FULL^VALM1 . W !?5,"MRA's may not be processed at this time." . W !?5,"The IB site parameter ""Allow MRA Processing?"" is set to NO." . D PAUSE^VALM1 . Q </pre> |

| Routine Name | IBCECOB2 |
|--------------|--|
| | <pre> D SEL(.IBDA,1) S Z=\$O(IBDA(0)),Z=\$G(IBDA(+Z)) G:'Z PROQ S IBIFN=\$P(Z,U),IB364=\$P(Z,U,2),IBDA=\$P(Z,U,3),IBIFNH=IBIFN I 'IBIFN G PROQ I '\$\$LOCK^IBCEU0(361.1,IBDA) G PROQ D COBCOPY(IBIFN,IB364,2,IBDA,"BLD^IBCECOB1",.IBNCN) D UNLOCK^IBCEU0(361.1,IBDA) ; ; for non-MRA claims copied from work list, set field 38 I \$G(IBMRAIEN)=1,\$G(IBMRAIEN)'="" (\$G(IBMRAIEN)=\$G(IBMRAIEN)) D .S X=\$\$WLRMV^IBCECOB1(\$G(IBMRAIEN),"PC") .;I \$P(\$G(^DGCR(399,IBMRAIEN,"S")),U,9)'=1 D .;W:'\$G(IBMRAIEN) !,*7,"Please note: the new bill was not AUTHORIZED.",!,"It can only be accessed now via the normal, non-EDI functions.",!,"Status of new bill is ",\$\$EXPAND^IBTRE(399,.13,\$P(^DGCR(399,IBMRAIEN,0),U,13)) .;D PAUSE^VALM1 .D:\$G(IBMRAIEN)=1 BLD^IBCECOB1 .Q ; ; PROQ S VALMBCK="R" Q ; COBCOPY(IBIFN,IB364,IBFROM,IBIEN,IBBLD,IBNCN) ; Generic entry point for EDI COB copy ; IBIFN = original bill ien ; IB364 = the ien of the transmission bill entry in file 364 ; IBFROM = 1 if called from CSA, 2 if called from COB/EOB processing ; IBIEN = entry in 361 (IBFROM=1) or 361.1 (IBFROM=2) being processed ; IBBLD = the name of the entypoint that will rebuild the display ; IBNCN = by reference, returns the new claim ien if user completed the Copy process ; N IBCBASK,IBCBCOPY,IBCAN,IBIFNH,IBNSTAT,IBOSTAT,IBPRCOB,IBSECHK,IBLMVAR,IBAC,IBMRAIEN, IBDA,IBAUTO N IBCOB,IBCOBIL,IBCOBN,IBINS,IBINSN,IBINSOLD,IBMRAIO,IBMRAO,IBNMOLD,IBQUIT S (IBCBASK,IBCBCOPY,IBCAN,IBAUTO)=1,(IBPRCOB,IBSECHK)=0,(IBMRAIEN,IBDA)=IBIEN I \$G(IBMRAIEN)'=1,'IB364!'IBIFN W !,"Transmission record is missing for this bill" D PAUSE^VALM1 G COBCOPX ; S IBIFNH=IBIFN I IBFROM=2 S IBPRCOB=1 ; IB*2.0*447 Check PR to include excess and percentages where applicable ;I \$\$(\$G(IBMRAIEN)=1:\$TOT(IBIFN)'>0,1:\$PREOBTOT^IBCEU0(IBIFN,\$G(IBMRAIEN))>0) D G COBCOPX I \$\$TOT(IBIFN,\$G(IBMRAIEN))>0 D G COBCOPX . D FULL^VALM1 . W !?5,"There is no " _\$\$(\$G(IBMRAIEN)=1:"balance remaining",1:"patient responsibility and/or excess charges")_" for this claim." . W !?5,"This claim may not be processed." . D PAUSE^VALM1 . Q ; ; </pre> |

| Routine Name | IBCECOB2 |
|--------------|--|
| | <pre> I \$G(IBDA)'="" , \$P(\$G(^IBM(361.1,IBDA,0)),U,16)="1.5" D G COBCOPX . W !,"This claim has already been processed as a sec/tert claim." . W !,"You will need to complete the authorization process for this claim." . D PAUSE^VALM1 . D AUTH . Q ; ; If multiple EOBs and one is processed, make sure collected closed. I \$G(IBMTRANOT), \$CCCHK(IBIFN)<0 D G COBCOPX . W !,"Multiple EOBs exist for this claim and at least one has EOB status of PROCESSED." . W !,"Claim cannot be sent to next payer until AR status is Collected/Closed." . D PAUSE^VALM1 . Q ; ; Get out if no next payer I \$P(\$G(^DGCR(399,IBIFN,"I"_\$COBN^IBCEF(IBIFN)+1))),U,1) D G COBCOPX . W !,"There is no next payer for this bill" . D PAUSE^VALM1 . Q ; D DSPRB^IBCCCB0(IBIFN) ; display related bills S IBCE("EDI")=1 D CHKB^IBCCCB ; process COB, create secondary bill S IBNCN=\$G(IBCE("EDI","NEW")) ; get new claim ien S IBIFN=IBIFNH I IBSECHK G COBCOPX ; ; if user came from CBW, no need to view and authorize a 2nd time (already happens in IBCCCB) Q:\$G(IBMTRANOT)=1 S IBV=1 D VIEW^IBCB2 ; display billing screens D AUTH ; authorize bill COBCOPX ; Q ; AUTH ; procedure to authorize the claim and refresh the screen K ^UTILITY(\$J) S IBAC=1,IBQUIT=0 D 3^IBCB1 I '\$D(IOUON)!\$D(IORVON) D ENS^%ZISS I \$P(\$G(^IBM(361.1,IBMRAIEN,0)),U,16)=3 D UPDEDI^IBCEM(IB364,"Z") I \$G(IBBLD)'="" D @IBBLD D PAUSE^VALM1 AUTHX ; Q ; RES ;Resubmit bill by print N IBDA,IBIFN,IB364 D SEL(.IBDA,1) S IBDA=+\$O(IBDA(0)),IBIFN=+\$G(IBDA(+IBDA)),IB364=+\$P(\$G(IBDA(IBDA)),U,2) I 'IBIFN G RESQ D PRINT1^IBCEM03(IBIFN,.IBDA,IB364) D PAUSE^VALM1 </pre> |

| Routine Name | IBCECOB2 |
|--------------|---|
| | <pre> I \$G(IBDA)'="" D BLD^IBCECOB1 RESQ S VALMBCK="R" Q ; EBI ;View an unauthorized transmitted bill N IBFLG,IBDA,IBIFN,IB364,DFN K ^TMP(\$J,"IBBILL") D FULL^VALM1 ; D SEL(.IBDA,1) S IBDA=+\$O(IBDA("")) S IBIFN=+\$G(IBDA(IBDA)),IB364=+\$P(\$G(IBDA(IBDA)),U,2),DFN=\$P(\$G(^DGCR(399,IBIFN,0)),U,2) G:'IBIFN EDITQ S IBV=1 D VIEW^IBCB2 I '\$D(IOUON)!'\$D(IORVON) D ENS^%ZISS D BLD^IBCECOB1 EDITQ S VALMBCK="R" Q ; SEL(IBDA,ONE) ; Select entry(s) from list ; IBDA = array returned if selections made ; IBDA(n)=ien of bill selected (file 399) ; ONE = if set to 1, only one selection can be made at a time N IB K IBDA D FULL^VALM1 D EN^VALM2(\$G(XQORNOD(0)),\$S('\$G(ONE):"',1:"S")) S IBDA=0 F S IBDA=\$O(VALMY(IBDA)) Q:'IBDA S IBDA(IBDA)=\$P(\$G(^TMP("IBCECOB",\$J,+IBDA)),U,2,6) Q ; EXIT ; Exit out of COB D FASTEXIT^IBCEFG4 I \$G(IBFASTXT)=1 S IBFASTXT=5 Q ; TOT(IBIFN,IBMRANOT) ; calculate if any balance remaining on non-MRA claim ; IBIFN = claim ien ; IBMRANOT = MRW/CBW flag (1=user came from CBW) added with IB*2.0*447 N IBPRTOT,IBBLD,IBCBN,IBU2 I \$G(IBMRANOT)'=1 Q \$\$(\$\$MSEDT^IBCEMU4(IBIFN)'="" :\$\$MSPRE^IBCEMU4(IBIFN),1:\$PREOBTOT^IBCEU0(IBIFN)) ; total up the payer paid amounts, if this is a 2ndary claim, be sure to account for what the primary paid also S IBU2=\$G(^DGCR(399,IBIFN,"U2")),IBCBN=\$\$COBN^IBCEF(IBIFN),IBPRTOT=\$\$EOBTOT^IBCEU1(IBIFN,IBCBN) S:IBPRTOT<0 IBPRTOT=0 ; don't allow negative prior payment or offset S:IBCBN=2 IBPRTOT=IBPRTOT+\$P(IBU2,U,4) S:IBCBN=3 IBPRTOT=IBPRTOT+\$P(IBU2,U,4)+\$P(IBU2,U,5) S:IBPRTOT<0 IBPRTOT=0 ; don't allow negative prior payment or offset ; Subtract payer paid amount from Total Charges from BILLS/CLAIMS (#399) file, don't allow neg </pre> |

| Routine Name | IBCECOB2 |
|--------------|--|
| | <pre> S IBBLD=\$P(\$G(^DGCR(399,IBIFN,"U1")),U,1)-IBPRTOT S:IBBLD<0 IBBLD=0 Q IBBLD ; CCCHK(IBIFN) ; If there are multiple EOBs on file for this claim, then one of them must be processed and AR status must be collected closed to process. ; returns 1 if true ; 0 if there are not multiple EOBs or mulitple EOBs and none are processed (all denials) ; -1 if false N IBDA,IBCT,IBPROC,IBARSTAT,IBEONBNDX,IBEOB S IBCT=0,IBPROC=0 F IBEONBNDX="B","C" D .S IBDA=0 F S IBDA=\$O(^IBM(361.1,IBEONBNDX,IBIFN,IBDA)) Q:'+IBDA D ..Q:\$D(IBEONBNDX) ..Q:\$P(\$G(^IBM(361.1,IBDA,0)),U,4)=1 ; only count EOBs ..S IBEONBNDX="B",IBCT=IBCT+1 ..I \$P(\$G(^IBM(361.1,IBDA,0)),U,13)=1 S IBPROC=1 I IBCT<2 Q 0 ; less than 2 EOBs I 'IBPROC Q 0 ; no EOBs with status processed S IBARSTAT=\$\$ARSTAT^IBJTU4(IBIFN) ; get status of AR I \$P(IBARSTAT,U)="COLLECTED/CLOSED" Q 1 Q -1 ; DENCHK(IBIFN,IBCT) ; Make sure all EOBs from this claim are denied. ; Input: IBIFN - IEN to 399 ; IBCT - by reference. Return count of EOBs. ; Output: returns 1 if there is at least one EOB and that none of the EOBs are processed. ; otherwise 0 ; N IBDA,IBPROC,IBEONBNDX,IBEOB S IBCT=0,IBPROC=0 F IBEONBNDX="B","C" D .S IBDA=0 F S IBDA=\$O(^IBM(361.1,IBEONBNDX,IBIFN,IBDA)) Q:'+IBDA D ..Q:\$D(IBEONBNDX) ..Q:\$P(\$G(^IBM(361.1,IBDA,0)),U,4)=1 ; only count EOBs ..S IBEONBNDX="B",IBCT=IBCT+1 ..I \$P(\$G(^IBM(361.1,IBDA,0)),U,13)=1 S IBPROC=1 I IBCT,'IBPROC Q 1 ; there is at least one EOB and none of the EOBs are processed. Q 0 ; </pre> |
| | <p>Modified Logic (Changes are in bold)</p> <pre> IBCECOB2 ;ALB/CXW - IB COB MANAGEMENT SCREEN ;16-JUN-1999 ;;2.0;INTEGRATED BILLING;**137,155,433,432,447,488**;;21-MAR-1994;Build 80 ;;Per VHA Directive 2004-038, this routine should not be modified. ; EDI ;history detail display N IBIFN,IBDA D SEL(.IBDA,1) </pre> |

| Routine Name | IBCECOB2 |
|--------------|---|
| | <pre> S IBDA=+\$O(IBDA(0)),IBIFN=+\$G(IBDA(IBDA)) D EDI1(IBIFN) S VALMBCK="R" Q ; EDI1(IBIFN) ; N DFN Q:'IBIFN S DFN=\$P(\$G(^DGCR(399,IBIFN,0)),U,2) D EN^VALM("IBJT EDI STATUS") K:\$D(IBFASTXT) IBFASTXT Q ; EDI2(IBIFN) ; N DFN Q:'IBIFN S DFN=\$P(\$G(^DGCR(399,IBIFN,0)),U,2) D EN^VALM("IBJT EDI STATUS ALONE") K:\$D(IBFASTXT) IBFASTXT Q ; CSA ;claims status awaiting resolution N IBDA D EN^IBCECSA I \$D(IBFASTXT) K IBFASTXT S VALMBCK="R" Q ; RVEOB ;Review EOB D FULL^VALM1 W ! N IBDA,IBIFN,IBCMT,IBSEL D SEL(.IBDA,1) S IBSEL=+\$O(IBDA(0)) S IBDA=\$G(IBDA(IBSEL)) S IBIFN=\$P(IBDA,U),IBDA=\$P(IBDA,U,3) I 'IBIFN G VEOBQ S IBCMT=\$G(^TMP("IBCECOB1",\$J,IBSEL)) I IBCMT="" D EN^VALM("IBCEM MRA REVIEW") VEOBQ K ^TMP("IBCECOC",\$J) S VALMBCK="R" Q ; TPJI ;Third Party joint Inquiry N IBDA,IBIFN D SEL(.IBDA,1) S IBDA=+\$O(IBDA(0)),IBIFN=+\$G(IBDA(IBDA)) I IBDA="" G TPJIQ D TPJI1(IBIFN) TPJIQ S VALMBCK="R" Q </pre> |

| Routine Name | IBCECOB2 |
|--------------|--|
| | <pre> ; TPJ11(IBIFN) ; N DFN,IBNOTPJI Q:'IBIFN S DFN=\$P(\$G(^DGCR(399,IBIFN,0)),U,2),IBNOTPJI=1 D EN^VALM("IBJT CLAIM INFO") K:\$D(IBFASTXT) IBFASTXT Q ; PBILL ;Print bill N IBIFN,IBDA,IBRESUB D SEL(.IBDA,1) S IBDA=\$O(IBDA(0)),IBIFN=+\$G(IBDA(+IBDA)) I IBDA="" G PBOUT S IBRESUB=\$RESUB^IBCECSA4(IBIFN,1,"P") I IBRESUB>0 W !,*7,"This is not a transmittable bill or review not needed" D PAUSE^VALM1 G PBOUT I IBRESUB=2 D G PBOUT . N IB364 . S IB364=+\$P(\$G(IBDA(IBDA)),U,2) . D PRINT1^IBCEM03(IBIFN,.IBDA,IB364) D PBILL1(IBIFN) PBOUT S VALMBCK="R" Q ; PMRA ;Print MRA N IBIFN,IBDA,IBDAX D SEL(.IBDA,1) S IBDA=\$O(IBDA(0)),IBIFN=+\$G(IBDA(+IBDA)),IBDAX=\$P(IBDA(+IBDA),U,3) G:'IBIFN PRMQ I '\$G(IBM(RANOT),\$D(^IBM(361.1,IBDAX,"ERR")),\$\$WARNMSE G PRMQ ; Claim contains Message Storage Errors D MRA^IBCEMRAA(.IBIFN) PRMQ S VALMBCK="R" Q PBILL1(IBIFN) ; N IBAC1,IBAC,DFN Q:'IBIFN S DFN=\$P(\$G(^DGCR(399,IBIFN,0)),U,2) S IBAC=4,IBAC1=1 D 4^IBCB1 D FULL^VALM1,PAUSE^VALM1 Q ; CANCEL ;Cancel bill ; IBDA(IBDA)=IBIFN^IB364^ien of 361.1^user selection seq^user name~duz# ; N IBIFN,IBDA,IB364,IBEObIFN,X,IBDENCT ; ; Check for security key I '\$\$KCHK^XUSRB("IB AUTHORIZE") D G CANCELQ </pre> |

| Routine Name | IBCECOB2 |
|--------------|---|
| | <pre> . D FULL^VALM1 S VALMBCK="R" . W !!?5,"You don't hold the proper security key to access this function." . W !?5,"The necessary key is IB AUTHORIZE. Please see your manager." . D PAUSE^VALM1 . Q ; D SEL(.IBDA,1) S IBDA=\$O(IBDA(0)),IBIFN=+\$G(IBDA(+IBDA)),IB364=\$P(\$G(IBDA(+IBDA)),U,2) S IBEOBIFN=\$P(\$G(IBDA(+IBDA)),U,3) ; ; IB*2.0*432 - if not mra, only allow cancel of denied claims. If no EOB, check AR status instead I 'IBEOBIFN,\$G(IBMCRANOT)=1,\$P(\$\$ARSTATA^IBJTU4(IBIFN),U)="COLLECTED/CLOSED" D G CANCE LQ . D FULL^VALM1 S VALMBCK="R" . W !!,*7,"You can only cancel denied claims. This claim is in a COLLECTED/CLOSED status" . W !,"Use Remove Action to remove claim from this worklist." . D PAUSE^VALM1 . Q ; ; IB*2.0*432 - if not mra, only allow cancel of claims with multiple EOBs if none have processed. I \$G(IBMCRANOT)=1,\$\$DENCHK(IBIFN,.IBDENCT),\$G(IBDENCT)>1 D G CANCELQ . D FULL^VALM1 S VALMBCK="R" . W !!,*7,"Multiple EOBs exist for this claim and at least one has EOB status of PROCESSED." . W !,"Use Remove Action to remove claim from this worklist." . D PAUSE^VALM1 . Q ; ; IB*2.0*432 - if not mra, only allow cancel of denied claims I IBEOBIFN,\$G(IBMCRANOT)=1,\$P(\$G(^IBM(361.1,IBEOBIFN,0)),U,13)'=2 D G CANCELQ . D FULL^VALM1 S VALMBCK="R" . W !!?5,*7,"You can only cancel denied claims." . D PAUSE^VALM1 . Q ; I IBDA D . I '\$\$LOCK^IBCEU0(361.1,IBEOBIFN) Q . D CANCEL^IBCEM3(.IBDA,IBIFN,IB364) . D UNLOCK^IBCEU0(361.1,IBEOBIFN) S VALMBCK="R" ; ; for non-MRA claims cancelled from worklist, set field 38 I \$G(IBMCRANOT)=1,\$P(\$G(^DGCR(399,IBIFN,0)),U,13)=7 S X=\$\$WLRMV^IBCECOB1(\$S(\$G(IBIFN)'="" :IBIFN,1:+\$G(IBDA(IBDA))),,"CA") I \$G(IBDA)'="" D BLD^IBCECOB1 CANCELQ Q ; CRD ; Correct Rejected/Denied claim protocol action N IBCNCRD S IBCNCRD=1 CLONE ; 'Copy/cancel bill' protocol action </pre> |

| Routine Name | IBCECOB2 |
|--------------|--|
| | <pre> N IBDA,IBQ,IBEObIFN,IBKEY,X,IBDENCT ; ; Check for security key ;I '\$\$KCHK^XUSRB("IB AUTHORIZE") D G CLONEQ S IBKEY=\$S(\$G(IBCNCRD)=1:"IB AUTHORIZE",1:"IB CLON") I '\$\$KCHK^XUSRB(IBKEY) D G CLONEQ . D FULL^VALM1 S VALMBCK="R" . ;W !!?5,"You don't hold the proper security key to access this function." . ;W !?5,"The necessary key is IB AUTHORIZE. Please see your manager." . W !!?5,"You must hold the "_IBKEY_" security key to access this function." . W !?5,"Please see your manager." . D PAUSE^VALM1 . Q ; D SEL(.IBDA,1) S IBDA=\$O(IBDA("")) I IBDA="" G CLONEQ ; ; IB*2.0*432 - if not mra, only allow cancel of claims with multiple EOBS if none have processed. I \$G(IBMRRANOT)=1,\$\$DENCHK(+IBDA(IBDA),.IBDENCT),\$G(IBDENCT)>1 D G CANCELQ . D FULL^VALM1 S VALMBCK="R" . W !,*7,"Multiple EOBS exist for this claim and at least one has EOB status of PROCESSED." . W !,"Use Remove Action to remove claim from this worklist." . D PAUSE^VALM1 . Q ; ; S IBEObIFN=\$P(\$G(IBDA(+IBDA)),U,3) I '\$\$LOCK^IBCEU0(361.1,IBEObIFN) G CLONEQ D COPYCLON(+\$G(IBDA(IBDA)),P(\$G(IBDA(+IBDA)),U,2),.IBQ) D UNLOCK^IBCEU0(361.1,IBEObIFN) ; ; for non-MRA claims cloned or corrected from worklist, set field 38 I \$G(IBMRRANOT)=1,\$G(IBQ)'="" S X=\$\$WLRMV^IBCECOB1(+\$G(IBDA(IBDA)),\$S(\$G(IBCNCRD)=1:"CR",1:"CL")) ; CLONEQ ; S VALMBCK="R" D:\$G(IBQ)'="" BLD^IBCECOB1 Q ; COPYCLON(IBIFN,IB364,IBQ) ; Generic entry point for clone a bill from EDI processing ; IBIFN = original bill ien ; IB364 = the ien of the transmission bill entry in file 364 ; IBQ = If bill is not cancelled, this is returned as null ; - pass by reference - ; N IBQUIT,IBCCCC,IBHV,Y,IBCAN,IBCE,IBDA,IBCNCOPY ;I '\$\$CANCK^IBCEM3("CC",IBIFN) S IBQ="" G CCQ I \$G(IBCNCRD)=1,\$\$CANCK^IBCEM3("CC",IBIFN) S IBQ="" G CCQ ; </pre> |

| Routine Name | IBCECOB2 |
|--------------|--|
| | <pre> ;S IBCAN=2,IBCE("EDI")=1,Y=IBIFN,IBCCCC=0,IBHV("IBIFN")=IBIFN,IBHV("IBIFN1")="",IBCNCOPY=1 S IBCAN=2,IBCE("EDI")=1,Y=IBIFN,IBCCCC=0,IBHV("IBIFN")=IBIFN,IBHV("IBIFN1")="" I \$G(IBCNCRD)=1 S IBCNCOPY=1 D ^IBCCC I \$G(IBCNCRD)=1 D CRD^IBCCC ;D ^IBCCC S IBIFN=IBHV("IBIFN") K IBCE("EDI") S IBQ=1 I \$P(\$G(^DGCR(399,IBIFN,0)),U,13)=7 S IBQ="" I IBHV("IBIFN1") D . N IBU . S IBU="R" . S IBNIEN=+IBHV("IBIFN1") . I "23"[\$P(\$G(^DGCR(399,+IBHV("IBIFN1"),0)),U,13) D .. W:\$G(IBCCEAUTO) !,*7,"Please note: the new bill was not AUTHORIZED.",!,"It can only be accessed now via the normal, non-EDI functions.",!,"Status of new bill is ",\$EXPAND^IBTRE(399,.13,\$P(^DGCR(399,IBHV("IBIFN1"),0),U,13)) S IBU="C" . D UPDEDI^IBCEM(IB364,IBU) ; I '\$G(IBCCEAUTO) D PAUSE^VALM1 CCQ Q ; PRO ; Copy for secondary/tertiary bill N VALMY,IBDA,Z,IBIFN,IBIFNH,IB364,IBCE,IBNCN ;I '\$P(\$G(^IBE(350.9,1,8)),U,12) D G PROQ I '\$P(\$G(^IBE(350.9,1,8)),U,12),\$G(IBMTRANOT)=1 D G PROQ . D FULL^VALM1 . W !?5,"MRA's may not be processed at this time." . W !?5,"The IB site parameter ""Allow MRA Processing?"" is set to NO." . D PAUSE^VALM1 . Q D SEL(.IBDA,1) S Z=\$O(IBMTRANOT),Z=\$G(IBMTRANOT) G:Z PROQ S IBIFN=\$P(Z,U),IB364=\$P(Z,U,2),IBDA=\$P(Z,U,3),IBIFNH=IBIFN I 'IBIFN G PROQ I '\$G(IBMTRANOT),\$D(^IBM(361.1,IBDA,"ERR")),\$\$WARNMSE G PROQ ; Claim contains Message Storage Errors I '\$\$LOCK^IBCEU0(361.1,IBDA) G PROQ D COBCOPY(IBMTRANOT,IB364,2,IBDA,"BLD^IBCECOB1",.IBNCN) D UNLOCK^IBCEU0(361.1,IBDA) ; ; for non-MRA claims copied from work list, set field 38 I \$G(IBMTRANOT)=1,\$G(IBMNCN)'="",(\$G(IBMNCN)'=\$G(IBMIFN)) D .S X=\$\$WLRMVF^IBCECOB1(\$G(IBMIFN),"PC") .;I \$P(\$G(^DGCR(399,+IBMNCN,"S")),U,9)=1 D .;W:\$G(IBCCEAUTO) !,*7,"Please note: the new bill was not AUTHORIZED.",!,"It can only be accessed now via the normal, non-EDI functions.",!,"Status of new bill is ",\$EXPAND^IBTRE(399,.13,\$P(^DGCR(399,IBMNCN,0),U,13)) .;D PAUSE^VALM1 .D:\$G(IBMTRANOT)=1 BLD^IBCECOB1 .Q </pre> |

| Routine Name | IBCECOB2 |
|--|----------|
| <pre> ; PROQ S VALMBCK="R" Q ; COBCOPY(IBIFN,IB364,IBFROM,IBIEN,IBBLD,IBNCN) ; Generic entry point for EDI COB copy ; IBIFN = original bill ien ; IB364 = the ien of the transmission bill entry in file 364 ; IBFROM = 1 if called from CSA, 2 if called from COB/EOB processing ; IBIEN = entry in 361 (IBFROM=1) or 361.1 (IBFROM=2) being processed ; IBBLD = the name of the entypoint that will rebuild the display ; IBNCN = by reference, returns the new claim ien if user completed the Copy process ; N IBCBASK,IBCBCOPY,IBCAN,IBIFNH,IBNSTAT,IBOSTAT,IBPRCOB,IBSECHK,IBLMVAR,IBAC,IBMRAIE N,IBDA,IBAUTO N IBCOB,IBCOBIL,IBCOBN,IBINS,IBINSN,IBINSOLD,IBMRAIO,IBMRAO,IBNMOLD,IBQUIT S (IBCBASK,IBCBCOPY,IBCAN,IBAUTO)=1,(IBPRCOB,IBSECHK)=0,(IBMRAIEN,IBDA)=IBIEN I \$G(IBMTRANOT)'=1,'IB364!'IBIFN W !,"Transmission record is missing for this bill" D PAUSE^VALM1 G COBCOPX ; S IBIFNH=IBIFN I IBFROM=2 S IBPRCOB=1 ; IB*2.0*447 Check PR to include excess and percentages where applicable ;I \$\$S(\$G(IBMTRANOT)=1:\$TOT(IBIFN)'>0,1:\$PREOBTOT^IBCEU0(IBIFN,\$G(IBMTRANOT))>0) D G COBCOPX I \$TOT(IBIFN,\$G(IBMTRANOT))>0 D G COBCOPX . D FULL^VALM1 . W !?5,"There is no "_\$S(\$G(IBMTRANOT)=1:"balance remaining",1:"patient responsibility and/or excess charges")_" for this claim." . W !?5,"This claim may not be processed." . D PAUSE^VALM1 . Q ; I \$G(IBDA)'=""\$,P(\$G(^IBM(361.1,IBDA,0)),U,16)="1.5" D G COBCOPX . W !,"This claim has already been processed as a sec/tert claim." . W !,"You will need to complete the authorization process for this claim." . D PAUSE^VALM1 . D AUTH . Q ; ; If multiple EOBs and one is processed, make sure collected closed. I \$G(IBMTRANOT),\$\$CCCHK(IBIFN)<0 D G COBCOPX . W !,"Multiple EOBs exist for this claim and at least one has EOB status of PROCESSED." . W !,"Claim cannot be sent to next payer until AR status is Collected/Closed." . D PAUSE^VALM1 . Q ; ; Get out if no next payer I '\$P(\$G(^DGCR(399,IBIFN,"I"_\$COBN^IBCEF(IBIFN)+1))),U,1) D G COBCOPX . W !,"There is no next payer for this bill" . D PAUSE^VALM1 </pre> | |

| Routine Name | IBCECOB2 |
|--------------|--|
| | <pre> . Q ; D DSPRB^IBCCCB0(IBIFN) ; display related bills S IBCE("EDI")=1 D CHKB^IBCCCB ; process COB, create secondary bill S IBNCN=\$G(IBCE("EDI","NEW")) ; get new claim ien S IBIFN=IBIFNH I IBSECHK G COBCOPX ; ; if user came from CBW, no need to view and authorize a 2nd time (already happens in IBCCCB) Q:\$G(IBMRAOT)=1 S IBV=1 D VIEW^IBCB2 ; display billing screens D AUTH ; authorize bill COBCOPX ; Q ; AUTH ; procedure to authorize the claim and refresh the screen K ^UTILITY(\$J) S IBAC=1,IBQUIT=0 D 3^IBCB1 I '\$D(IOUSON)!\$D(IORVON) D ENS^%ZISS I \$P(\$G(^IBM(361.1,IBMRAIEN,0)),U,16)=3 D UPDEDI^IBCEM(IB364,"Z") I \$G(IBBLD)'="" D @IBBLD D PAUSE^VALM1 AUTHX ; Q ; RES ;Resubmit bill by print N IBDA,IBIFN,IB364 D SEL(.IBDA,1) S IBDA=+\$O(IBDA(0)),IBIFN=+\$G(IBDA(+IBDA)),IB364=+\$P(\$G(IBDA(IBDA)),U,2) I 'IBIFN G RESQ D PRINT1^IBCEM03(IBIFN,.IBDA,IB364) D PAUSE^VALM1 I \$G(IBDA)'="" D BLD^IBCECOB1 RESQ S VALMBCK="R" Q ; EBI ;View an unauthorized transmitted bill N IBFLG,IBDA,IBIFN,IB364,DFN K ^TMP(\$J,"IBBILL") D FULL^VALM1 ; D SEL(.IBDA,1) S IBDA=+\$O(IBDA("")) S IBIFN=+\$G(IBDA(IBDA)),IB364=+\$P(\$G(IBDA(IBDA)),U,2),DFN=\$P(\$G(^DGCR(399,IBIFN,0)),U,2) G:'IBIFN EDITQ S IBV=1 D VIEW^IBCB2 I '\$D(IOUSON)!\$D(IORVON) D ENS^%ZISS D BLD^IBCECOB1 EDITQ S VALMBCK="R" Q </pre> |

| Routine Name | IBCECOB2 |
|--------------|---|
| | <pre> ; SEL(IBDA,ONE) ; Select entry(s) from list ; IBDA = array returned if selections made ; IBDA(n)=ien of bill selected (file 399) ; ONE = if set to 1, only one selection can be made at a time N IB K IBDA D FULL^VALM1 D EN^VALM2(\$G(XQORNOD(0)),\$S('\$G(ONE):"',1:"S")) S IBDA=0 F S IBDA=\$O(VALMY(IBDA)) Q:'IBDA S IBDA(IBDA)=\$P(\$G(^TMP("IBCECOB",\$J,+IBDA)),U,2,6) Q ; EXIT ; Exit out of COB D FASTEXIT^IBCEFG4 I \$G(IBFASTXT)=1 S IBFASTXT=5 Q ; TOT(IBIFN,IBMRANOT) ; calculate if any balance remaining on non-MRA claim ; IBIFN = claim ien ; IBMRANOT = MRW/CBW flag (1=user came from CBW) added with IB*2.0*447 N IBPRTOT,IBBLD,IBCBN,IBU2 I \$G(IBMRANOT)=1 Q \$\$(\$MSEDT^IBCEMU4(IBIFN)'="":\$MSPRE^IBCEMU4(IBIFN),1:\$PREOBTOT^IBCEU0(IBIFN)) ; total up the payer paid amounts, if this is a 2ndary claim, be sure to account for what the primary paid also S IBU2=\$G(^DGCR(399,IBIFN,"U2")),IBCBN=\$G(COBN^IBCEF(IBIFN),IBPRTOT=\$EGBTOT^IBCEU1(IBIFN,IBCBN) S:IBPRTOT<0 IBPRTOT=0 ; don't allow negative prior payment or offset S:IBCBN=2 IBPRTOT=IBPRTOT+\$P(IBU2,U,4) S:IBCBN=3 IBPRTOT=IBPRTOT+\$P(IBU2,U,4)+\$P(IBU2,U,5) S:IBPRTOT<0 IBPRTOT=0 ; don't allow negative prior payment or offset ; Subtract payer paid amount from Total Charges from BILLS/CLAIMS (#399) file, don't allow neg S IBBLD=\$P(\$G(^DGCR(399,IBIFN,"U1")),U,1)-IBPRTOT S:IBBLD<0 IBBLD=0 Q IBBLD ; CCCHK(IBIFN) ; If there are multiple EOBs on file for this claim, then one of them must be processed and AR status must be collected closed to process. ; returns 1 if true ; 0 if there are not multiple EOBs or mulitple EOBs and none are processed (all denials) ; -1 if false N IBDA,IBCT,IBPROC,IBARSTAT,IBEOBNDX,IBEOB S IBCT=0,IBPROC=0 F IBEOBNDX="B","C" D .S IBDA=0 F S IBDA=\$O(^IBM(361.1,IBEOBNDX,IBIFN,IBDA)) Q:'+IBDA D ..Q:\$D(IEOB(IBDA)) ..Q:\$P(\$G(^IBM(361.1,IBDA,0)),U,4)=1 ; only count EOBs ..S IBEOB(IBDA)="",IBCT=IBCT+1 ..I \$P(\$G(^IBM(361.1,IBDA,0)),U,13)=1 S IBPROC=1 I IBCT<2 Q 0 ; less than 2 EOBs </pre> |

| Routine Name | IBCECOB2 |
|--|----------|
| <pre> I 'IBPROC Q 0 ; no EOBs with status processed S IBARSTAT=\$\$ARSTATA^IBJTU4(IBIFN) ; get status of AR I \$P(IBARSTAT,U)="COLLECTED/CLOSED" Q 1 Q -1 ; DENCHK(IBIFN,IBCT) ; Make sure all EOBs from this claim are denied. ; Input: IBIFN - IEN to 399 ; IBCT - by reference. Return count of EOBs. ; Output: returns 1 if there is at least one EOB and that none of the EOBS are processed. ; otherwise 0 ; N IBDA,IBPROC,IBEObNDX,IBEOb S IBCT=0,IBPROC=0 F IBEObNDX="B","C" D .S IBDA=0 F S IBDA=\$O(^IBM(361.1,IBEObNDX,IBIFN,IBDA)) Q:'+IBDA D ..Q:\$D(IBEOb(IBDA)) ..Q:\$P(\$G(^IBM(361.1,IBDA,0)),U,4)=1 ; only count EOBs ..S IBEOb(IBDA)="",IBCT=IBCT+1 ..I \$P(\$G(^IBM(361.1,IBDA,0)),U,13)=1 S IBPROC=1 I IBCT,IBPROC Q 1 ; there is at least one EOB and none of the EOBS are processed. Q 0 ; ; WARNMSE() ; Display MSE Warning and check if we should continue. D FULL^VALM1 N DIR,X,Y S DIR("A",1)="WARNING: Claim has a Data Mismatch/Message Storage Error;" S DIR("A",2)="this option may not function as needed." S DIR("A")="Continue? ",DIR("B")="NO",DIR(0)="YA" D ^DIR I Y>0 Q 1 ; Okay to continue. Q 0 ; </pre> | |

| Routine Name | IBCEOB | |
|---------------------------------|--|---|
| Enhancement Category | <input type="checkbox"/> New <input checked="" type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change | |
| Requirement Traceability Matrix | 2.6.5.1 | |
| Related Options | MRA Management Worklist [MRW] | |
| Related Routines | Routines "Called By" | Routines "Called" |
| | ^IBCEOB0, ^IBCEOB00, ^IBCEOB01, ^IBCEOB1, ^IBCESRV3 | ^IBCESRV2, ^IBCEM, ^DILF, ^IBCEOB1, ^IBCEU, ^DIE, ^IBCEF, ^IBEFUNC, ^IBCEOB01, ^IBCEOB00, ^IBCEOB0, ^XMRENT, ^DICN, ^IBJTU4, |

| Routine Name | IBCEOB | | | | |
|--|--------------------------------|---|-------------------------------|---|--------------------------------|
| | | ^PRCAFN, ^IBCNSBL2, ^IBCAPP, ^IBCEMU1, | | | |
| Data Dictionary (DD) References | | | | | |
| Related Protocols | | | | | |
| Related Integration Control Registrations (ICRs) | | | | | |
| Data Passing | <input type="checkbox"/> Input | <input type="checkbox"/> Output Reference | <input type="checkbox"/> Both | <input type="checkbox"/> Global Reference | <input type="checkbox"/> Local |
| Input Attribute Name and Definition | | | | | |
| Output Attribute Name and Definition | | | | | |
| Current Logic | | | | | |
| <pre>. . . ; UPD3611(IBEOb,IBTDA,IBAR) ; From flat file 835 format, add EOB record ; IBEOb = the ien of the entry in file 361.1 being updated ; IBTDA = the ien in the source file ; IBAR = 1 if being called from AR N HIPAA,IBA1,IBFILE,IBEGBL,Z,IBREC,Q S IBFILE=\$S('\$G(IBAR):'^IBA(364.2,"_IBTDA_",2),1:"^TMP("_\$J_",""RCDP-EOB"", "_IBTDA_"")') S IBEGBL=\$S('\$G(IBAR):'IBCERR-EOB',1:"RCDPERR-EOB") S HIPAA=0 I \$G(IBAR),'\$HDR^IBCEOB1(\$G(^TMP(\$J,"RCDPEOB","HDR")),IBEGBL,IBEOb,.HIPAA) Q S IBA1=0 F S IBA1=\$O(@IBFILE@(IBA1)) Q:'IBA1 S IB0=\$S('\$G(IBAR):\$P(\$G(^IBA1,0)),##RAW DATA: ",2),1:\$G(@IBFILE@(IBA1,0))) I IB0'="" D . S IBREC=+IB0 . I IBREC'=37 K ^TMP(\$J,37) . I IBREC S IB="S IBOK=\$\$_IBREC_(IB0,IBEGBL,IBEOb)",Q=IBREC_"^IBCEOB" I \$T(@Q)'="" X IB S:'IBOK ^TMP(IBEGBL, \$J,+\$(^TMP(IBEGBL,\$J,""),-1)+1)=\$S('\$G(IBAR):' ##RAW DATA: ",1:"")_IB0 ; If a DENIED non MRA EOB with no filing errors is updated, put on the CBW worklist if the ; claim isn't already COLLECTED/CLOSED and there is a subsequent payer (incl. Tricare & ChampVA) I IBEOb,'\$(^TMP(IBEGBL,\$J,0)) D .N IB361,IBIFN,IBX,IBTXT,IBPYMT .; must be non-MRA EOB and DENIED .S IB361=\$G(^IBM(361.1,IBEOb,0)),IBIFN=\$P(IB361,U) Q:\$P(IB361,U,4)'=0 .Q:\$P(IB361,U,13)'=2 .Q:\$P(\$\$ARSTATA^IBJTU4(IBIFN,U)="COLLECTED/CLOSED"</pre> | | | | | |

| Routine Name | IBCEOB |
|---|---|
| | <pre> .; payment on this bill from A/R IA#380 OR payer paid amount from EOB .S IBPYMT=\$\$TPR^PRCAFN(IBIFN) S:IBPYMT="" IBPYMT=+\$G(^IBM(361.1,IBEOB,1)) .; check for subsequent payer .S IBX=\$\$EOB^IBCNSBL2(\$G(IBIFN),+\$G(^DGCR(399,IBIFN,"U1")), \$G(IBPYMT),.IBTXT) Q:\$D(IBTXT) .D PUTONWL^IBCAPP(\$P(IB361,U),"IB804:EOB Claim Status must be PROCESSED") ; </pre> |
| Modified Logic (Changes are in bold) | |
| | <pre> . . . ; UPD3611(IBEOB,IBTDA,IBAR) ; From flat file 835 format, add EOB record ; IBEOB = the ien of the entry in file 361.1 being updated ; IBTDA = the ien in the source file ; IBAR = 1 if being called from AR N HIPAA,IBA1,IBFILE,IBEGBL,Z,IBREC,Q S IBFILE=\$S('\$G(IBAR):'^IBA(364.2,"_IBTDA_",2),1:^TMP("_\$J_", "RCDP-EOB", "_IBTDA_")) S IBEGBL=\$S('\$G(IBAR):'^IBCERR-EOB",1:"RCDPERR-EOB") S HIPAA=0 I \$G(IBAR),\$\$HDR^IBCEOB1(\$G(^TMP(\$J,"RCDPEOB","HDR")),IBEGBL,IBEOB,.HIPAA) Q S IBA1=0 F S IBA1=\$O(@IBFILE@(IBA1)) Q:'IBA1 S IB0=\$S('\$G(IBAR):\$P(\$G(^IBA1,0)),##RAW DATA: ",2),1:\$G(@IBFILE@(IBA1,0))) I IB0="" D . S IBREC=+IB0 . I IBREC'=37 K ^TMP(\$J,37) .;;I IBREC S IB="S IBOK=\$\$_IBREC_"(IB0,IBEGBL,IBEOB),Q=IBREC_"^IBCEOB" I \$T(@Q)'="" X IB S:'IBOK ^TMP(IGBGL,\$J,+O(^TMP(IGBGL,\$J,""),-1)+1)=\$S('\$G(IBAR):" ##RAW DATA: ",1:"")_IB0 . I IBREC S IB="S IBOK=\$\$_IBREC_"(IB0,IBEGBL,IBEOB),Q=IBREC_"^IBCEOB" I \$T(@Q)'="" X IB S:'IBOK ^TMP(IGBGL, \$J,+O(^TMP(IGBGL,\$J,""),-1)+1)="-----" ; If a DENIED non MRA EOB with no filing errors is updated, put on the CBW worklist if the ; claim isn't already COLLECTED/CLOSED and there is a subsequent payer (incl. Tricare & ChampVA) I IBEOB,\$O(^TMP(IGBGL,\$J,0)) D .N IB361,IBIFN,IBX,IBTXT,IBPYMT .; must be non-MRA EOB and DENIED .S IB361=\$G(^IBM(361.1,IBEOB,0)),IBIFN=\$P(IB361,U) Q:\$P(IB361,U,4)'=0 .Q:\$P(IB361,U,13)'=2 .Q:\$P(\$\$ARSTATA^IBJTU4(IBIFN),U)="COLLECTED/CLOSED" .; payment on this bill from A/R IA#380 OR payer paid amount from EOB .S IBPYMT=\$\$TPR^PRCAFN(IBIFN) S:IBPYMT="" IBPYMT=+\$G(^IBM(361.1,IBEOB,1)) .; check for subsequent payer .S IBX=\$\$EOB^IBCNSBL2(\$G(IBIFN),+\$G(^DGCR(399,IBIFN,"U1")), \$G(IBPYMT),.IBTXT) Q:\$D(IBTXT) .D PUTONWL^IBCAPP(\$P(IB361,U),"IB804:EOB Claim Status must be PROCESSED") ; Q F S IBEOB=\$O(^IBM(361.1,"B",IBIFN,IBEOB)) Q:'IBEOB D Q:DUP </pre> |

| | |
|---|---------------|
| Routine Name | IBCEOB |
| . S CHKSUM2=+\$P(\$G(^IBM(361.1,IBEOB,100)),U,5) ; checksum of old EOB . I 'CHKSUM2 Q . I CHKSUM1=CHKSUM2 S DUP=IBEOB Q ; comparison . Q DUPX ; Q DUP ; | |

| | | |
|--|---|--|
| Routine Name | IBCEOB00 | |
| Enhancement Category | <input type="checkbox"/> New <input checked="" type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change | |
| Requirement Traceability Matrix | 2.6.5.1 | |
| Related Options | MRA Management Worklist [MRW] | |
| Related Routines | Routines “Called By” | Routines “Called” |
| | ^IBCEOB, ^IBCEOB0, ^IBCEOB1 | ^IBCEOB, ^DIE, ^IBCEOB1, ^IBEFUNC, ^DICN |
| Data Dictionary (DD) References | | |
| Related Protocols | | |
| Related Integration Control Registrations (ICRs) | | |
| Data Passing | <input type="checkbox"/> Input <input type="checkbox"/> Output Reference <input type="checkbox"/> Both <input type="checkbox"/> Global Reference <input type="checkbox"/> Local | |
| Input Attribute Name and Definition | | |
| Output Attribute Name and Definition | | |
| Current Logic | | |
| <div><div>.</div><div>.</div><div>.</div><div>;</div><div>DET40(IB0,ARRAY) ; Format important details of record 40 for error</div><div>; IB0 = data on 40 record (some pieces pre-formatted)</div><div>; ARRAY(n)=formatted line is returned if passed by ref</div><div>N Q</div><div>S ARRAY(1)="Payer reported the following was billed to them:"</div><div>S ARRAY(2)=" " _ \$\$(\$P(IB0,U,21)="NU": "Rev Cd",1:"Proc") _ ": " _ \$\$(\$P(IB0,U,10)'="":\$P(IB0,U,10),1:"Same</div></div> | | |

| Routine Name | IBCEOB00 |
|--------------------------------------|--|
| | <pre> as adjudicated")_ " Chg: " _\$J(\$P(IB0,U,15)/100,"",2)_ " Units: " _\$S(\$P(IB0,U,16):\$P(IB0,U,16),1:1) S ARRAY(3)=" Svc Date(s): " _\$S(\$P(IB0,U,19)'="":\$FDT(\$P(IB0,U,19)),1:"??")_ \$S(\$P(IB0,U,20)'="":"- _ \$FDT(\$P(IB0,U,20)),1:"") I \$P(IB0,U,11)'="" S ARRAY(3)=ARRAY(3)_ " Mods: " F Q=11:1:14 I \$P(IB0,U,Q)'="" S ARRAY(3)=ARRAY(3)_ \$P(IB0,U,Q)_ \$S(Q=14:"",\$P(IB0,U,Q+1)'="":",",1:"") S ARRAY(4)="Payer reported adjudication on:" S ARRAY(5)=" " _\$S(\$P(IB0,U,21)="NU": "Rev Cd",1:"Proc")_ ": _ \$S(\$P(IB0,U,3)'="":\$P(IB0,U,3),1:\$P(IB0,U,4)) S ARRAY(5)=ARRAY(5)_ " Type: " _\$P(IB0,U,21)_ \$S(\$P(IB0,U,21)'="NU": "Rev Cd: " _\$P(IB0,U,4),1:"")_ " Units: " _\$S(\$P(IB0,U,18):\$P(IB0,U,18)/100,1:1)_ " Amt: " _\$J(\$P(IB0,U,17)/100,"",2) I \$P(IB0,U,5)'="" S ARRAY(5)=ARRAY(5)_ " Mods: " F Q=5:1:8 I \$P(IB0,U,Q)'="" S ARRAY(5)=ARRAY(5)_ \$P(IB0,U,Q)_ \$S(Q=8:"",\$P(IB0,U,Q+1)'="":",",1:"") Q ; </pre> |
| Modified Logic (Changes are in bold) | |
| | <pre> . . . ; DET40(IB0,ARRAY,IBZDATA,ERRTYP) ; Format important details of record 40 for error ; IB0 = data on 40 record (some pieces pre-formatted) ; ARRAY(n)=formatted line is returned if passed by ref ; IBZDATA = an array of the 835 data elements. N CPT,EOBCHG,IBBNDL,OCHG,OCHG2,OPROC,OREVCD,Q S EOBCHG=+\$DOLLAR^IBCEOB(\$P(IB0,U,15)) S IBBNDL=\$S(\$P(IB0,U,10)'="":1,1:0) I ERRTYP="ERR-04" D Q . ; PLEASE NOTE...Should any of the following data gathering be changed...please . ; make sure that the same changes are made in the routine IBCEOB1 because these . ; 2 routines need to be in sync. . S OCHG=\$P(IBZDATA,U,3)*\$P(IBZDATA,U,4) . S OCHG2=\$P(IBZDATA,U,5) . I OCHG'=EOBCHG,OCHG2=EOBCHG S OCHG=OCHG2 . S OPROC="" ,CPT=\$P(IBZDATA,U,2) I CPT'? .N,CPT'="" S CPT=\$O(^ICPT("B",CPT,"")) . S OPROC=\$\$PRCD^IBCEF1(+CPT_";ICPT(") . S OREVCD=+\$P(\$G(^DGCR(399.2,+IBZDATA,0)),U) . . . S ARRAY(1)="Payer reported the following was billed via the Claim (837):" . ;;;S ARRAY(2)=" " _\$S(\$P(IB0,U,21)="NU": "Rev Cd",1:"Proc")_ ": _ \$S(\$P(IB0,U,10)'="":\$P(IB0,U,10),1:"Same as adjudicated")_ " Chg: " _\$J(\$P(IB0,U,15)/100,"",2)_ " Units: _ \$S(\$P(IB0,U,16):\$P(IB0,U,16),1:1) . ;;;S ARRAY(3)=" Svc Date(s): " _\$S(\$P(IB0,U,19)'="":\$FDT(\$P(IB0,U,19)),1:"??")_ \$S(\$P(IB0,U,20)'="":"- _ \$FDT(\$P(IB0,U,20)),1:"") . S ARRAY(2)=" Proc:" _OPROC . S ARRAY(2)=ARRAY(2)_ " Mods:" _\$P(IBZDATA,U,9) . S ARRAY(2)=ARRAY(2)_ " Rev Cd:" _OREVCD_ " Chg:" _OCHG_ Units:" _\$S(\$P(IBZDATA,U,4):\$P(IBZDATA,U,4),1:1) </pre> |

| Routine Name | IBCEOB00 |
|---|----------|
| <pre> . S ARRAY(3)="Payer reported adjudication via the ERA (835) as follows:" . S ARRAY(4)=" Proc:"_\$_S('IBBNDL:\$P(IB0,U,3),1:\$P(IB0,U,10)) . S ARRAY(4)=ARRAY(4)_ " Mods:" I \$P(IB0,U,5)'="" F Q=5:1:8 I \$P(IB0,U,Q)'="" S ARRAY(4)=ARRAY(4)_\$_P(IB0,U,Q)_\$_S(Q=8:"",\$P(IB0,U,Q+1)'=""_",",1:"") .;;;S ARRAY(5)=ARRAY(5)_ " Type: " _\$_P(IB0,U,21)_\$_S(\$P(IB0,U,21)'="NU": " Rev Cd: " _\$_P(IB0,U,4),1:"")_ Units: " _\$_S(\$P(IB0,U,18):\$P(IB0,U,18)/100,1:1)_ " Amt: " _\$_J(\$P(IB0,U,17)/100,"",2) . S ARRAY(4)=ARRAY(4)_ " Rev Cd:"_\$_S('IBBNDL:\$P(IB0,U,4),1:\$P(IB0,U,10)) . S ARRAY(4)=ARRAY(4)_ " Chg:"_EOBCHG_ " Units:" _\$_P(IB0,U,16) . S ARRAY(5)=" Amt:" _\$_J(\$P(IB0,U,17)/100,"",2) . S ARRAY(6)=" " .;;; S ARRAY(4)=ARRAY(4)_ " Paid Amt: " _\$_J(\$P(IB0,U,17)/100,"",2)_ " Units: " _\$_S(\$P(IB0,U,16):\$P(IB0,U,16)/100,1:1) ; ; ERRTYP must be "ERR-1500" N IBAMIN,UNITS,MODS ; PLEASE NOTE...Should any of the following data gathering be changed...please ; make sure that the same changes are made in the routine IBCEOB1 because these ; 2 routines need to be in sync. S IBAMIN="" I \$P(IBZDATA,U,12)'="" S IBAMIN=\$P(IBZDATA,U,12) S UNITS=\$\$_S('IBAMIN:\$P(IBZDATA,U,9),1:IBAMIN/15) S OPROC="" ,CPT=\$P(IBZDATA,U,5) I CPT?.N,CPT'="" S CPT=\$O(^ICPT("B",CPT,"")) S OPROC=\$\$_PRCD^IBCEF1(+CPT_"",ICPT("")) S OCHG=\$P(IBZDATA,U,8)*\$P(IBZDATA,U,9) ; S ARRAY(1)="Payer reported the following was billed via the Claim (837):" .;;;S ARRAY(2)=" " _\$_S(\$P(IB0,U,21)'="NU": "Rev Cd",1:"Proc")_ " : " _\$_S(\$P(IB0,U,10)'="" :\$P(IB0,U,10),1:"Same as adjudicated")_ " Chg: " _\$_J(\$P(IB0,U,15)/100,"",2)_ " Units: " _\$_S(\$P(IB0,U,16):\$P(IB0,U,16),1:1) .;;;S ARRAY(3)=" Svc Date(s): " _\$_S(\$P(IB0,U,19)'="" :\$\$_FDT(\$P(IB0,U,19)),1:"??")_\$_S(\$P(IB0,U,20)'="" :"- " _\$_FDT(\$P(IB0,U,20)),1:"") S ARRAY(2)=" Proc:"_OPROC S ARRAY(2)=ARRAY(2)_ " Mods:" _\$_MODLST^IBEFUNC2(\$P(IBZDATA,U,10)) S ARRAY(2)=ARRAY(2)_ " Rev Cd: Chg:" _\$_J(OCHG,"",2)_ " Units:" _UNITS S ARRAY(3)="Payer reported adjudication via the ERA (835) as follows:" S ARRAY(4)=" Proc:"_\$_S('IBBNDL:\$P(IB0,U,3),1:\$P(IB0,U,10))_ " Mods:" S MODS=\$\$_S('IBBNDL:\$P(IB0,U,5,8),1:\$P(IB0,U,11,14)) I \$P(MODS,U,1)'="" F Q=1:1:4 I \$P(MODS,U,Q)'="" S ARRAY(4)=ARRAY(4)_\$_P(MODS,U,Q)_\$_S(Q=4:"",\$P(MODS,U,Q+1)'=""_",",1:"") .;;;S ARRAY(5)=ARRAY(5)_ " Type: " _\$_P(IB0,U,21)_\$_S(\$P(IB0,U,21)'="NU": " Rev Cd: " _\$_P(IB0,U,4),1:"")_ Units: " _\$_S(\$P(IB0,U,18):\$P(IB0,U,18)/100,1:1)_ " Amt: " _\$_J(\$P(IB0,U,17)/100,"",2) S ARRAY(4)=ARRAY(4)_ " Rev Cd: Chg:" _\$_J(\$P(IB0,U,15)/100,"",2) S ARRAY(4)=ARRAY(4)_ " Units:" _\$_P(IB0,U,16) S ARRAY(5)=" Amt:" _\$_J(\$P(IB0,U,17)/100,"",2) S ARRAY(6)=" " .;;;S ARRAY(4)=ARRAY(4)_ " Rev Cd: Paid Amt: " _\$_J(\$P(IB0,U,17)/100,"",2) Q ; </pre> | |

| | | | | |
|--|---|---|--|--|
| Routine Name | IBCEOB0 | | | |
| Enhancement Category | <input type="checkbox"/> New <input checked="" type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change | | | |
| Requirement Traceability Matrix | 2.6.5.1 | | | |
| Related Options | MRA Management Worklist [MRW] | | | |
| Related Routines | Routines "Called By" | Routines "Called" | | |
| | ^IBCEOB | ^IBCEOB1, ^IBCEU, ^IBCEOB00, ^DICN, ^IBCEOB, ^DIE, ^IBEFUNC | | |
| Data Dictionary (DD) References | | | | |
| Related Protocols | | | | |
| Related Integration Control Registrations (ICRs) | | | | |
| Data Passing | <input type="checkbox"/> Input <input type="checkbox"/> Output Reference <input type="checkbox"/> Both <input type="checkbox"/> Global Reference <input type="checkbox"/> Local | | | |
| Input Attribute Name and Definition | | | | |
| Output Attribute Name and Definition | | | | |
| Current Logic | | | | |
| <p>IBCEOB0 ;ALB/TMP/PJH - 835 EDI EOB MSG PROCESSING ; 8/24/10 7:23pm ;;2.0;INTEGRATED BILLING;**135,280,155,431**;;21-MAR-94;Build 106 ;;Per VHA Directive 2004-038, this routine should not be modified. Q ; ; LINE() ;Extract Provider Line Reference from 42 record N SUB,NODE,VAL S VAL="",SUB=IBA1 ; from loop in UPD3611^IBCEOB F S SUB=\$O(@IBFILE@(SUB)) Q:SUB="" D Q:+NODE>42 .S NODE=\$G(@IBFILE@(SUB,0)) .S:NODE["RAW DATA" NODE=\$P(NODE," ",3,99) .Q:+NODE'=42 S VAL=\$P(NODE,U,5) Q VAL ; ; 30(IB0,IBE0B,IBOK) ; Process record type 30 for EOB ; IB0 = the record being processed ; IBE0B = the ien of the EOB entry in file 361.1 ; IBOK = Returned as 1 if record filed OK, 0 if error occurred ; ; N A S A="3;4.01;0;1;1^5;4.02;0;1;1^6;4.03;1;0;0^7;4.05;1;0;0^8;4.06;1;0;0^9;4.07;1;0;0^10;4.08;1;0;0^11;4.09;1;0;0^12;4.1;1;0;0^13;4.11;1;0;0^14;4.19;0;1;1" ; ;</p> | | | | |

| Routine Name | IBCEOB0 |
|--------------|---|
| | <pre> S IBOK=\$\$STORE^IBCEOB1(A,IB0,IBEOB) I 'IBOK S ^TMP(IBEGBL,\$J,\$O(^TMP(IBEGBL,\$J,""),-1)+1)="Bad MEDICARE Inpt Adjudication data" Q30 Q ; 40(IB0,IBEOB,IBOK) ; Process record type 40 for EOB ; IB0 = the record being processed ; IBEOB = the ien of the EOB entry in file 361.1 ; IBOK = Returned as 1 if record filed OK, 0 if error occurred ; ; IBZDATA is also assumed to exist or if not, it is created in FINDLN ; N A,LEVEL,IBSEQ,IBDA,IBPC,IBLREF,IBIFN,Q,X,Y,DA,DD,DO,DIC,DLAYGO,PLREF K ^TMP(\$J,40) ; the entry # for corresponding 41, 42, and 45 records ; S IBIFN=+\$G(^IBM(361.1,IBEOB,0)) L +^IBM(361.1,IBEOB,15):0 I \$T S IBSEQ=+\$O(^IBM(361.1,IBEOB,15," "),-1)+1 I '\$G(IBSEQ) S ^TMP(IBEGBL,\$J,\$O(^TMP(IBEGBL,\$J,""),-1)+1)="Record lock failure - could not acquire next service line number" G Q40 ; ; Update the 40 record data a little bit (pieces 3/4/16) I \$P(IB0,U,21)="NU" S \$P(IB0,U,4)=\$P(IB0,U,3),\$P(IB0,U,3)="" S \$P(IB0,U,16)=\$S(+\$P(IB0,U,16):\$P(IB0,U,16)/100,1:+\$P(IB0,U,18)/100) I \$P(IB0,U,4)?1.N S \$P(IB0,U,4)=\$P(IB0,U,4) ; ; Find the line item from original bill for this adjustment S PLREF=\$S('HIPAA:\$P(IB0,U,22),1:\$LINE()) ; old format from 40 record, new format from 42 S IBLREF=+\$FINDLN^IBCEOB1(IB0,IBEOB,IBZDATA,+PLREF) I 'IBLREF D G Q40 . N Z,Z0,CT . S ^TMP(IBEGBL,\$J,\$O(^TMP(IBEGBL,\$J,""),-1)+1)="Service line detail could not be matched to a billed item" . D DET40^IBCEOB00(IB0,Z0) . S CT=+\$O(^TMP(IBEGBL,\$J,""),-1),Z=0 F S Z=\$O(Z0(Z)) Q:'Z S CT=CT+1,^TMP(IBEGBL,\$J,CT)=Z0(Z) ; S DIC="^IBM(361.1,"_IBEOB_",15,"DIC(0)="L",DLAYGO=361.115,DA(1)=IBEOB S X=IBSEQ S DIC("DR")=".12/////"_IBLREF_\$S(\$P(IBLREF,U,2)=""::"1:",".15/////"_P(IBLREF,U,2))_",".16/////"_\$\$DATE^ IBCEU(\$P(IB0,U,19))_\$S(\$P(IB0,U,20):",".17/////"_\$\$DATE^IBCEU(\$P(IB0,U,20)),1:"") D FILE^DICN K DIC,DO,DD,DLAYGO ;Add a new LINE LEVEL ADJUSTMENT ('SVC') I Y<0 S ^TMP(IBEGBL,\$J,\$O(^TMP(IBEGBL,\$J,""),-1)+1)="Could not add a LINE LEVEL ADJUSTMENT ("_IBSEQ_)" G Q40 ; L -^IBM(361.1,IBEOB,15) ; S LEVEL=15.1,LEVEL(0)=+Y,LEVEL(1)=IBEOB,LEVEL("DIE")="^IBM(361.1,"_IBEOB_",15," S A="3;.04;0;0;0^4;.1;0;0;0^9;.09;0;0;0^17;.03;1;0;0^18;.11;0;1;D2^21;.18;0;0;0" I \$P(IB0,U,18),\$P(IB0,U,16) S \$P(A,U,5)="16;.11;0;1;1" I \$\$STORE^IBCEOB1(A,IB0,IBEOB,LEVEL) S ^TMP(\$J,40)=LEVEL(0),IBOK=1 I '\$G(IBOK) S ^TMP(IBEGBL,\$J,\$O(^TMP(IBEGBL,\$J,""),-1)+1)="Bad data for line level adjustment </pre> |

| Routine Name | IBCEOB0 |
|--------------|---|
| | <pre> " _IBSEQ G Q40 ; ; Store modifiers in multiple S DIC="^IBM(361.1," _IBEOB_" ,15," _LEVEL(0)" ,2," ,DIC(0)="L",DLAYGO=361.1152,DA(2)=IBEOB,DA(1) =LEVEL(0) F Q=5:1:8 S X=\$P(IB0,U,Q) I X'="" D FILE^DICN K DO,DD I Y<0 S IBOK=0 Q K DLAYGO,DIC,DR,DA I '\$G(IBOK) S ^TMP(IBEGBL,\$J,\$O(^TMP(IBEGBL,\$J,""),-1)+1)="Could not file modifier data for line level adjustment " _IBSEQ G Q40 Q40 Q ; 41(IB0,IBEOB,IBOK) ; Process record type 41 for EOB ; IB0 = the record being processed ; IBEOB = the ien of the EOB entry in file 361.1 ; IBOK = Returned as 1 if record filed OK, 0 if error occurred ; N DA,DR,DIE,X,Y,Z,Z0,CT I '\$G(^TMP(\$J,40)) D G Q41 . S ^TMP(IBEGBL,\$J,\$O(^TMP(IBEGBL,\$J,""),-1)+1)="Service line adjustment has no corresponding service line" . D DET4X^IBCEOB00(41,IB0,.Z0) . S CT=\$O(^TMP(IBEGBL,\$J,""),-1),Z=0 F S Z=\$O(Z0(Z)) Q:'Z S CT=CT+1,^TMP(IBEGBL,\$J,CT)=Z0(Z) ; S DR="",IBOK=1 S DA=+^TMP(\$J,40),DA(1)=IBEOB S DIE="^IBM(361.1," _DA(1)" ,15," I +\$P(IB0,U,3) S DR=" .13//"_ \$\$DOLLAR^IBCEOB(\$P(IB0,U,3)) I +\$P(IB0,U,4) S DR=DR_\$S(DR="":",1:");" .14//"_ \$\$DOLLAR^IBCEOB(\$P(IB0,U,4)) I DR="" D ^DIE S IBOK=(\$D(Y)=0) I '\$G(IBOK) S ^TMP(IBEGBL,\$J,\$O(^TMP(IBEGBL,\$J,""),-1)+1)="Bad data for service line adjustment-2" ; ; For Medicare MRA's only: ; If the Allowed Amount field is present, then we need to file an ; adjustment: Group code PR, Reason code AAA, Amount, Quantity, and ; Reason Text. This is data normally found on the 45 record, so we're ; going to create our own "45" record and file it. ; I \$P(\$G(^IBM(361.1,IBEOB,0)),U,4)=1,+\$P(IB0,U,3) D . N IB45,IBSAV40 . S IB45=45_U_\$P(IB0,U,2)_U_"PR"_U_"AAA"_U_\$P(IB0,U,3)_U_"0000000001" . S IB45=IB45_U_"Allowed Amount" . S IBSAV40=\$G(^TMP(\$J,40)) . D 45(IB45,IBEOB,IBOK) . S ^TMP(\$J,40)=IBSAV40 . I '\$G(IBOK) S ^TMP(IBEGBL,\$J,\$O(^TMP(IBEGBL,\$J,""),-1)+1)="Could not file the PR-AAA adjustment for the Allowed Amount at line " _+^TMP(\$J,40) . Q ; Q41 Q </pre> |

| Routine Name | IBCEOB0 |
|--------------|---|
| | <pre> ; 42(IB0,IBEOB,IBOK) ; Process record type 42 for EOB ; IB0 = the record being processed ; IBEOB = the ien of the EOB entry in file 361.1 ; IBOK = Returned as 1 if record filed OK, 0 if error occurred ; N DO,DD,DLAYGO,DIC,DA,X,Y,Z,Z0,CT S IBOK=0 I '\$G(^TMP(\$J,40)) D G Q42 . S ^TMP(IBEGBL,\$J,+\$(^TMP(IBEGBL,\$J,""),-1)+1)="Service line adjustment has no corresponding service line" . D DET4X^IBCEOB00(42,IB0,.Z0) . S CT=+\$(^TMP(IBEGBL,\$J,""),-1),Z=0 F S Z=\$(Z0(Z)) Q:'Z S CT=CT+1,^TMP(IBEGBL,\$J,CT)=Z0(Z) ; K DO,DD,DLAYGO S IBOK=1 S DA(1)=+^TMP(\$J,40),DA(2)=IBEOB S X=+\$(^IBM(361.1,DA(2),15,DA(1),4," "),- 1)+1,DIC="^IBM(361.1,"_DA(2)_"_15,"_DA(1)_"_4,"_DIC(0)="L",DLAYGO=361.1154 S DIC("DR")=\$S(\$P(IB0,U,3)'="" : ".02////"_\$P(IB0,U,3),1:"") I \$P(IB0,U,4)'="" S:\$L(DIC("DR")) DIC("DR")=DIC("DR")_"_" S DIC("DR")=DIC("DR")_"_.03////"_\$TR(\$P(IB0,U,4),","," ") D FILE^DICN K DO,DD,DLAYGO I Y>0 S IBOK=0 I '\$G(IBOK) S ^TMP(IBEGBL,\$J,+\$(^TMP(IBEGBL,\$J,""),-1)+1)="Bad data for service line adjustment-3" ; ; For Medicare MRA's only: ; Process and store the line level remark code as an LQ kludge line ; level adjustment. ; I \$P(\$G(^IBM(361.1,IBEOB,0)),U,4)=1,\$P(IB0,U,3)'="" D . N IB45,IBSAV40 . S IB45=45_U_\$P(IB0,U,2)_U_"LQ"_U_\$P(IB0,U,3)_U_0_U_0_U_\$P(IB0,U,4) . S IBSAV40=\$G(^TMP(\$J,40)) . D 45(IB45,IBEOB,IBOK) . S ^TMP(\$J,40)=IBSAV40 . I '\$G(IBOK) S ^TMP(IBEGBL,\$J,+\$(^TMP(IBEGBL,\$J,""),-1)+1)="Could not file the LQ-remark code adjustment at line "_+^TMP(\$J,40) . Q Q42 Q ; 45(IB0,IBEOB,IBOK) ; Process record type 45 for EOB ; IB0 = the record being processed ; IBEOB = the ien of the EOB entry in file 361.1 ; IBOK = Returned as 1 if record filed OK, 0 if error occurred ; N IBDA,LEVEL,A,Z0,CT,Z I '\$G(^TMP(\$J,40)) D G Q45 . S ^TMP(IBEGBL,\$J,+\$(^TMP(IBEGBL,\$J,""),-1)+1)="Service line adjustment has no corresponding </pre> |

IBCEOB0

| Routine Name | IBCEOB0 |
|--------------|--|
| | <pre> ; IBOK = Returned as 1 if record filed OK, 0 if error occurred ; S IBOK=0 N AGC,IBDA,LEVEL,A,Z0,CT,Z I '\$G(^TMP(\$J,40)) D G Q46 . S ^TMP(IBEGBL,\$J,\$O(^TMP(IBEGBL,\$J,""),-1)+1)="Service line adjustment has no corresponding service line" . D DET4X^IBCEOB00(46,IB0,.Z0) . ;S CT=+\$O(^TMP(IBEGBL,\$J,""),-1),Z=0 F S Z=\$O(Z0(Z)) Q:'Z S CT=CT+1,^TMP(IBEGBL,\$J,CT)=Z0(Z) ; S AGC=\$P(^TMP(\$J,40),U,2) I AGC="" S ^TMP(IBEGBL,\$J,\$O(^TMP(IBEGBL,\$J,""),-1)+1)="Service line adjustment is missing its group code" G Q46 ; S IBDA(2)=+^TMP(\$J,40) S IBDA(1)=+\$O(^IBM(361.1,IBE0B,15,IBDA(2),1,"B",AGC,0)) ; ; ;Add a new entry at the Payer Policy level I \$G(IBDA(1)) D . S DIC="^IBM(361.1,"_IBE0B_",15,"_IBDA(2)_",1,"_IBDA(1)_",2,"DIC(0)="L",DLAYGO=361.11511,DA(1)=I BDA(1),DA(2)=IBDA(2),DA(3)=IBE0B . S DIC("P")=\$\$GETSPEC^IBEFUNC(361.1151,1) . S X=\$P(IB0,U,3) . D FILE^DICN K DIC,DO,DD,DLAYGO . I Y<0 K IBDA S ^TMP(IBEGBL,\$J,\$O(^TMP(IBEGBL,\$J,""),-1)+1)="Could not add payer policy (" _\$P(IB0,U,4)_") for adjustment group code (" _\$P(IB0,U,3)_") at line adjustment " _+^TMP(\$J,40) Q . S IBDA=+Y,IBOK=1 ; Q46 Q ; </pre> |
| | Modified Logic (Changes are in bold) |
| | <pre> IBCEOB0 ;ALB/TMP/PJH - 835 EDI EOB MSG PROCESSING ; 8/24/10 7:23pm ;;2.0;INTEGRATED BILLING;**135,280,155,431,488**;21-MAR-94;Build 106 ;;Per VHA Directive 2004-038, this routine should not be modified. Q ; LINE() ;Extract Provider Line Reference from 42 record N SUB,NODE,VAL S VAL="",SUB=IBA1 ; from loop in UPD3611^IBCEOB F S SUB=\$O(@IBFILE@(SUB)) Q:SUB="" D Q:~SUB>42 .S NODE=\$G(@IBFILE@(SUB,0)) .S:NODE["RAW DATA" NODE=\$P(NODE," ",3,99) .Q:~SUB=42 S VAL=\$P(NODE,U,5) Q VAL ; </pre> |

| Routine Name | IBCEOB0 |
|--------------|---|
| | <pre> 30(IB0,IBEOB,IBOK) ; Process record type 30 for EOB ; IB0 = the record being processed ; IBEOB = the ien of the EOB entry in file 361.1 ; IBOK = Returned as 1 if record filed OK, 0 if error occurred ; N A S A="3;4.01;0;1;1^5;4.02;0;1;1^6;4.03;1;0;0^7;4.05;1;0;0^8;4.06;1;0;0^9;4.07;1;0;0^10;4.08;1;0;0^11;4.09;1;0;0^12;4.1;1;0;0^13;4.11;1;0;0^14;4.19;0;1;1" ; S IBOK=\$\$STORE^IBCEOB1(A,IB0,IBEOB) I 'IBOK S ^TMP(IBEGBL,\$J,\$O(^TMP(IBEGBL,\$J,""),-1)+1)="Bad MEDICARE Inpt Adjudication data" Q30 Q ; 40(IB0,IBEOB,IBOK) ; Process record type 40 for EOB ; IB0 = the record being processed ; IBEOB = the ien of the EOB entry in file 361.1 ; IBOK = Returned as 1 if record filed OK, 0 if error occurred ; ; IBZDATA is also assumed to exist or if not, it is created in FINDLN ; N A,LEVEL,IBSEQ,IBDA,IBPC,IBLREF,IBIFN,Q,X,Y,DA,DD,DO,DIC,DLAYGO,PLREF K ^TMP(\$J,40) ; the entry # for corresponding 41, 42, and 45 records ; S IBIFN=+\$G(^IBM(361.1,IBEOB,0)) L +^IBM(361.1,IBEOB,15):0 I \$T S IBSEQ=+\$O(^IBM(361.1,IBEOB,15,""),-1)+1 I '\$G(IBSEQ) S ^TMP(IBEGBL,\$J,\$O(^TMP(IBEGBL,\$J,""),-1)+1)="Record lock failure - could not acquire next service line number" G Q40 ; ; Update the 40 record data a little bit (pieces 3/4/16) I \$P(IB0,U,21)="NU" S \$P(IB0,U,4)=\$P(IB0,U,3),\$P(IB0,U,3)=" " S \$P(IB0,U,16)=\$S(+\$P(IB0,U,16):\$P(IB0,U,16)/100,1:+\$P(IB0,U,18)/100) I \$P(IB0,U,4)?1.N S \$P(IB0,U,4)=\$P(IB0,U,4) ; ; Find the line item from original bill for this adjustment S PLREF=\$S('HIPAA:\$P(IB0,U,22),1:\$LINE()) ; old format from 40 record, new format from 42 S IBLREF=\$\$FINDLN^IBCEOB1(IB0,IBEOB,IBZDATA,+PLREF) I '+IBLREF D G Q40 . N Z,Z0,CT,EFLAG,ETEXT,IBZ,ERRTYP . S EFLAG=0,ETEXT="",IBZ=+\$P(IBLREF,U,2),ERRTYP=\$P(IBLREF,U,3) . ;; S ^TMP(IBEGBL,\$J,\$O(^TMP(IBEGBL,\$J,""),-1)+1)="Service line detail could not be matched to a billed item" . S ^TMP(IBEGBL,\$J,\$O(^TMP(IBEGBL,\$J,""),-1)+1)=" " . S ^TMP(IBEGBL,\$J,\$O(^TMP(IBEGBL,\$J,""),-1)+1)="A discrepancy exists between data elements in the Claim (837)" . S ^TMP(IBEGBL,\$J,\$O(^TMP(IBEGBL,\$J,""),-1)+1)="compared to the ERA (835) in record type 40:" . S ^TMP(IBEGBL,\$J,\$O(^TMP(IBEGBL,\$J,""),-1)+1)=" " . S EFLAG=\$P(IBLREF,U,4),ETEXT=\$P("REVENUE CODE^PROCEDURE CODE^UNITS^CHARGES^MODIFIERS",U,EFLAG) . I ETEXT="" S ETEXT="DATA" . S ^TMP(IBEGBL,\$J,\$O(^TMP(IBEGBL,\$J,""),-1)+1)="*** " _ETEXT_ " MISMATCH" </pre> |

| Routine Name | IBCEOB0 |
|--|---------|
| <pre> . S ^TMP(IBEGBL,\$J,+\$(^TMP(IBEGBL,\$J,""),-1)+1)=" " . D DET40^IBCEOB00(IB0,Z0,\$G(IBZDATA(IBZ)),ERRTYP) . S CT=+\$(^TMP(IBEGBL,\$J,""),-1),Z=0 F S Z=\$(Z0(Z)) Q:'Z S CT=CT+1,^TMP(IBEGBL,\$J,CT)=Z0(Z) ; S DIC="^IBM(361.1,"_IBEOB_",15,"DIC(0)="L",DLAYGO=361.115,DA(1)=IBEOB S X=IBSEQ S DIC("DR")=".12////_"_IBLREF_\$S(\$P(IBLREF,U,2)="":",1:",".15////_"_P(IBLREF,U,2))_",".16////_"_\$\$DATE^ IBCEU(\$P(IB0,U,19))_\$\$(\$P(IB0,U,20):",".17////_"_\$\$DATE^IBCEU(\$P(IB0,U,20)),1:","") D FILE^DICN K DIC,DO,DD,DLAYGO ;Add a new LINE LEVEL ADJUSTMENT ('SVC') I Y<0 S ^TMP(IBEGBL,\$J,+\$(^TMP(IBEGBL,\$J,""),-1)+1)="Could not add a LINE LEVEL ADJUSTMENT ("_IBSEQ_)" G Q40 ; L -^IBM(361.1,IBEOB,15) ; S LEVEL=15.1,LEVEL(0)=+Y,LEVEL(1)=IBEOB,LEVEL("DIE")="^IBM(361.1,"_IBEOB_",15," S A="3;.04;0;0;0^4;.1;0;0;0^9;.09;0;0;0^17;.03;1;0;0^18;.11;0;1;D2^21;.18;0;0;0" I '\$P(IB0,U,18),\$P(IB0,U,16) S \$P(A,U,5)="16;.11;0;1;1" I \$\$STORE^IBCEOB1(A,IB0,IBEOB,LEVEL) S ^TMP(\$J,40)=LEVEL(0),IBOK=1 I '\$G(IBOK) S ^TMP(IBEGBL,\$J,+\$(^TMP(IBEGBL,\$J,""),-1)+1)="Bad data for line level adjustment "_IBSEQ G Q40 ; ; Store modifiers in multiple S DIC="^IBM(361.1,"_IBEOB_",15,"_LEVEL(0)_"",2,"DIC(0)="L",DLAYGO=361.1152,DA(2)=IBEOB,DA(1) =LEVEL(0) F Q=5:1:8 S X=\$P(IB0,U,Q) I X'="" D FILE^DICN K DO,DD I Y<0 S IBOK=0 Q K DLAYGO,DIC,DR,DA I '\$G(IBOK) S ^TMP(IBEGBL,\$J,+\$(^TMP(IBEGBL,\$J,""),-1)+1)="Could not file modifier data for line level adjustment "_IBSEQ G Q40 Q40 Q ; 41(IB0,IBEOB,IBOK) ; Process record type 41 for EOB ; IB0 = the record being processed ; IBEOB = the ien of the EOB entry in file 361.1 ; IBOK = Returned as 1 if record filed OK, 0 if error occurred ; N DA,DR,DIE,X,Y,Z,Z0,CT I '\$G(^TMP(\$J,40)) D G Q41 . S ^TMP(IBEGBL,\$J,+\$(^TMP(IBEGBL,\$J,""),-1)+1)="Service line adjustment (record type 41) has no corresponding service line" . D DET4X^IBCEOB00(41,IB0,Z0) . S CT=+\$(^TMP(IBEGBL,\$J,""),-1),Z=0 F S Z=\$(Z0(Z)) Q:'Z S CT=CT+1,^TMP(IBEGBL,\$J,CT)=Z0(Z) ; S DR="",IBOK=1 S DA=+^TMP(\$J,40),DA(1)=IBEOB S DIE="^IBM(361.1,"_DA(1)_"",15," I +\$P(IB0,U,3) S DR=".13//_"_\$\$DOLLAR^IBCEOB(\$P(IB0,U,3)) I +\$P(IB0,U,4) S DR=DR_\$\$(\$P(DR,"":",1:","")_".14//_"_\$\$DOLLAR^IBCEOB(\$P(IB0,U,4)) I DR'="" D ^DIE S IBOK=(\$D(Y)=0) </pre> | |

| Routine Name | IBCEOB0 |
|--------------|---|
| | <pre> I '\$G(IBOK) S ^TMP(IBEGBL,\$J,+\$(^TMP(IBEGBL,\$J,""),-1)+1)="Bad data for service line adjustment-2 (record type 41)" ; ; For Medicare MRA's only: ; If the Allowed Amount field is present, then we need to file an ; adjustment: Group code PR, Reason code AAA, Amount, Quantity, and ; Reason Text. This is data normally found on the 45 record, so we're ; going to create our own "45" record and file it. ; I \$P(\$G(^IBM(361.1,IBEGBL,0)),U,4)=1,+\$(P(IB0,U,3) D . N IB45,IBSAV40 . S IB45=45_U_\$P(IB0,U,2)_U_"PR"_U_"AAA"_U_\$P(IB0,U,3)_U_"0000000001" . S IB45=IB45_U_"Allowed Amount" . S IBSAV40=\$G(^TMP(\$J,40)) . D 45(IB45,IBEGBL,IBOK) . S ^TMP(\$J,40)=IBSAV40 . I '\$G(IBOK) S ^TMP(IBEGBL,\$J,+\$(^TMP(IBEGBL,\$J,""),-1)+1)="Could not file the PR-AAA adjustment for the Allowed Amount at line "_+^TMP(\$J,40) . Q ; Q41 Q ; 42(IB0,IBEGBL,IBOK) ; Process record type 42 for EOB ; IB0 = the record being processed ; IBEGBL = the ien of the EOB entry in file 361.1 ; IBOK = Returned as 1 if record filed OK, 0 if error occurred ; N DO,DD,DLAYGO,DIC,DA,X,Y,Z,Z0,CT S IBOK=0 I '\$G(^TMP(\$J,40)) D G Q42 . S ^TMP(IBEGBL,\$J,+\$(^TMP(IBEGBL,\$J,""),-1)+1)="Service line adjustment (record type 42) has no corresponding service line" . D DET4X^IBCEOB00(42,IB0,Z0) . S CT=+\$(^TMP(IBEGBL,\$J,""),-1),Z=0 F S Z=Z0(Z) Q:'Z S CT=CT+1,^TMP(IBEGBL,\$J,CT)=Z0(Z) ; K DO,DD,DLAYGO S IBOK=1 S DA(1)=+^TMP(\$J,40),DA(2)=IBEGBL S X=+\$(^IBM(361.1,DA(2),15,DA(1),4,""),- 1)+1,DIC="^IBM(361.1,"_DA(2)_"_15,"_DA(1)_"_4,"",DIC(0)="L",DLAYGO=361.1154 S DIC("DR")=\$S(\$P(IB0,U,3)'="" : ".02////"_\$P(IB0,U,3),1:"") I \$P(IB0,U,4)'="" S:\$L(DIC("DR")) DIC("DR")=DIC("DR")_"_" S DIC("DR")=DIC("DR")_"_.03////"_\$TR(\$P(IB0,U,4),","," ") D FILE^DICN K DO,DD,DLAYGO I Y>0 S IBOK=0 I '\$G(IBOK) S ^TMP(IBEGBL,\$J,+\$(^TMP(IBEGBL,\$J,""),-1)+1)="Bad data for service line adjustment-3 (record type 42)" ; ; For Medicare MRA's only: </pre> |

| Routine Name | IBCEOB0 |
|--------------|--|
| | <pre> ; Process and store the line level remark code as an LQ kludge line ; level adjustment. ; I \$P(\$G(^IBM(361.1,IBEOB,0)),U,4)=1,\$P(IB0,U,3)="" D . N IB45,IBSAV40 . S IB45=45_U_\$P(IB0,U,2)_U_"LQ"_U_\$P(IB0,U,3)_U_0_U_0_U_\$P(IB0,U,4) . S IBSAV40=\$G(^TMP(\$J,40)) . D 45(IB45,IBEOB,.IBOK) . S ^TMP(\$J,40)=IBSAV40 . I '\$G(IBOK) S ^TMP(IBEGBL,\$J,\$O(^TMP(IBEGBL,\$J,""),-1)+1)="Could not file the LQ-remark code adjustment at line "_+^TMP(\$J,40) . Q Q42 Q ; 45(IB0,IBEOB,IBOK) ; Process record type 45 for EOB ; IB0 = the record being processed ; IBEOB = the ien of the EOB entry in file 361.1 ; IBOK = Returned as 1 if record filed OK, 0 if error occurred ; N IBDA,LEVEL,A,Z0,CT,Z I '\$G(^TMP(\$J,40)) D G Q45 . S ^TMP(IBEGBL,\$J,\$O(^TMP(IBEGBL,\$J,""),-1)+1)="Service line adjustment (record type 45) has no corresponding service line" . D DET4X^IBCEOB00(45,IB0,.Z0) . S CT=+\$O(^TMP(IBEGBL,\$J,""),-1),Z=0 F S Z=\$O(Z0(Z)) Q:'Z S CT=CT+1,^TMP(IBEGBL,\$J,CT)=Z0(Z) ; I \$P(IB0,U,3)="" S \$P(^TMP(\$J,40),U,2)=\$P(IB0,U,3) I \$P(IB0,U,3)="" S \$P(IB0,U,3)=\$P(^TMP(\$J,40),U,2) I \$P(IB0,U,3)="" S ^TMP(IBEGBL,\$J,\$O(^TMP(IBEGBL,\$J,""),-1)+1)="Service line adjustment (record type 45) is missing its group code" G Q45 ; S IBDA(2)=+^TMP(\$J,40) S IBDA(1)=+\$O(^IBM(361.1,IBEOB,15,IBDA(2),1,"B",\$P(IB0,U,3),0)) ; I 'IBDA(1) D ;Needs a new entry at group level . N X,Y,DA,DD,DO,DIC,DLAYGO . S DIC=^IBM(361.1,"_IBEOB_",15,"_IBDA(2)_",1,"",DIC(0)="L",DLAYGO=361.1151,DA(2)=IBEOB,DA(1)=IB DA(2) . S DIC("P")=\$\$GETSPEC^IBEFUNC(361.115,1) . S X=\$P(IB0,U,3) . D FILE^DICN K DIC,DO,DD,DLAYGO . I Y<0 K IBDA S ^TMP(IBEGBL,\$J,\$O(^TMP(IBEGBL,\$J,""),-1)+1)="Could not add adjustment group code ("_ \$P(IB0,U,3)_") at line adjustment "_+^TMP(\$J,40) Q . S IBDA(1)=+Y ; ;Add a new entry at the reason code level I \$G(IBDA(1)) D . </pre> |

| Routine Name | IBCEOB0 |
|--|---------|
| <pre> S DIC="^IBM(361.1,"_IBEOB_",15,"_IBDA(2)_" ,1,"_IBDA(1)_" ,1," ,DIC(0)="L",DLAYGO=361.11511,DA(1)=I BDA(1),DA(2)=IBDA(2),DA(3)=IBEOB . S DIC("P")=\$\$GETSPEC^IBEFUNC(361.1151,1) . S X=\$P(IB0,U,4) . D FILE^DICN K DIC,DO,DD,DLAYGO . I Y<0 K IBDA S ^TMP(IBEGBL,\$J,\$O(^TMP(IBEGBL,\$J,""),-1)+1)="Could not add reason code ("_\$P(IB0,U,4)_") for adjustment group code ("_\$P(IB0,U,3)_") at line adjustment "_+^TMP(\$J,40) Q . S IBDA=+Y ; I \$G(IBDA) D . S LEVEL=15,LEVEL("DIE")="^IBM(361.1,"_IBEOB_",15,"_IBDA(2)_" ,1,"_IBDA(1)_" ,1," . S LEVEL(0)=IBDA,LEVEL(1)=IBDA(1),LEVEL(2)=IBDA(2),LEVEL(3)=IBEOB . S A="5;.02;1;0;0^6;.03;0;1;1^7;.04;0;1;0" . S IBOK=\$\$STORE^IBCEOB1(A,IB0,IBEOB,.LEVEL) . I 'IBOK S ^TMP(IBEGBL,\$J,\$O(^TMP(IBEGBL,\$J,""),-1)+1)="Bad data for reason code ("_\$P(IB0,U,4)_"), adjustment group code ("_\$P(IB0,U,3)_") at line adjustment "_+^TMP(\$J,40) Q ; Q45 Q ; 46(IB0,IBEOB,IBOK) ; Process record type 46 for EOB ; IB0 = the record being processed ; IBEOB = the ien of the EOB entry in file 361.1 ; IBOK = Returned as 1 if record filed OK, 0 if error occurred ; S IBOK=0 N AGC,IBDA,LEVEL,A,Z0,CT,Z I '\$G(^TMP(\$J,40)) D G Q46 . S ^TMP(IBEGBL,\$J,\$O(^TMP(IBEGBL,\$J,""),-1)+1)="Service line adjustment (record type 46) has no corresponding service line" . D DET4X^IBCEOB00(46,IB0,Z0) . ;S CT=+\$O(^TMP(IBEGBL,\$J,""),-1),Z=0 F S Z=\$O(Z0(Z)) Q:'Z S CT=CT+1,^TMP(IBEGBL,\$J,CT)=Z0(Z) ; S AGC=\$P(^TMP(\$J,40),U,2) I AGC="" S ^TMP(IBEGBL,\$J,\$O(^TMP(IBEGBL,\$J,""),-1)+1)="Service line adjustment (record type 46) is missing its group code" G Q46 ; S IBDA(2)=+^TMP(\$J,40) S IBDA(1)=+\$O(^IBM(361.1,IBEOB,15,IBDA(2),1,"B",AGC,0)) ; ; ;Add a new entry at the Payer Policy level I \$G(IBDA(1)) D . S DIC="^IBM(361.1,"_IBEOB_",15,"_IBDA(2)_" ,1,"_IBDA(1)_" ,2," ,DIC(0)="L",DLAYGO=361.11511,DA(1)=I BDA(1),DA(2)=IBDA(2),DA(3)=IBEOB . S DIC("P")=\$\$GETSPEC^IBEFUNC(361.1151,1) . S X=\$P(IB0,U,3) . D FILE^DICN K DIC,DO,DD,DLAYGO . I Y<0 K IBDA S ^TMP(IBEGBL,\$J,\$O(^TMP(IBEGBL,\$J,""),-1)+1)="Could not add payer policy ("_\$P(IB0,U,4)_") for adjustment group code ("_\$P(IB0,U,3)_") at line adjustment "_+^TMP(\$J,40) Q </pre> | |

| | |
|--|----------------|
| Routine Name | IBCEOB0 |
| . S IBDA=+Y,IBOK=1 ; Q46 Q ; ; | |

| | | |
|--|---|--|
| Routine Name | IBCEOB1 | |
| Enhancement Category | <input type="checkbox"/> New <input checked="" type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change | |
| Requirement Traceability Matrix | 2.6.5.1 | |
| Related Options | MRA Management Worklist [MRW] | |
| Related Routines | Routines "Called By" | Routines "Called" |
| | ^IBCEOB, ^IBCEOB0, ^IBCEOB00 | ^DIE, ^IBCEF, ^IBCEOB, ^IBCEMU1, ^IBCEF1, ^IBCEOB00, ^IBCEFG1, ^IBEFUNC2 |
| Data Dictionary (DD) References | | |
| Related Protocols | | |
| Related Integration Control Registrations (ICRs) | | |
| Data Passing | <input type="checkbox"/> Input <input type="checkbox"/> Output Reference <input type="checkbox"/> Both <input type="checkbox"/> Global Reference <input type="checkbox"/> Local | |
| Input Attribute Name and Definition | | |
| Output Attribute Name and Definition | | |
| Current Logic | | |
| IBCEOB1 ;ALB/TMP/PJH - 835 EDI EOB MSG PROCESSING ; 7/13/10 5:32pm ;;2.0;INTEGRATED BILLING;**137,135,155,296,356,349,431**;21-MAR-94;Build 106 ;;Per VHA Directive 2004-038, this routine should not be modified. Q ; ; STORE(A,IB0,IBE0B,LEVEL) ; ; A = the string of data to extract and try to store ; each ^ piece is a field to store ; within each ^ piece, there are 5 ';' pieces: ; 1 = piece to extract from the data string; ; 2 = field to update; ; 3 = flag for dollar amt (1=YES); ; 4 = flag for 4-slash stuff without transform (1=YES); | | |

| Routine Name | IBCEOB1 |
|--------------|---|
| | <pre> ; 5 = flag for numeric/non-dollar amt (1=Yes, 0=No, ; Dn = the field is numeric with 'n' decimal places ; IB0 = the record being processed ; IBEOB = the ien of the EOB entry in file 361.1 ; LEVEL = the array that contains the DIE and DA values if stuffing at a ; level other than the top level ; ; N B,IBPC,IBFLD,DA,DR,DIE,X,Y S DR="," ; ; I '\$G(LEVEL) S DIE="^IBM(361.1,"DA=IBEOB ; ; I '\$G(LEVEL) D . N Q . S DIE=\$G(LEVEL("DIE")) . S Q=0 F S Q=\$O(LEVEL(Q)) Q:Q S DA(Q)=LEVEL(Q) . S DA=LEVEL(0) ; ; I '\$G(DA) F B=1:1:\$L(A,U) D . S IBPC=\$P(A,U,B),IBFLD=\$P(IBPC,"",2) . I \$P(IB0,U,+IBPC)'="" ,IBFLD D .. N VAL .. ; For dollar amts, add full cents; For numerics, strip leading .. ; 0's; For non-numeric/non-dollar amts, make any ; in data into .. S VAL=\$S(\$P(IBPC,"",3):\$DOLLAR^IBCEOB(\$P(IB0,U,+IBPC)),\$P(IBPC,"",5):+\$P(IB0,U,+IBPC),\$P(\$P(I BPC,"",5),"D",2):\$P(IB0,U,+IBPC)/(10**\$P(\$P(IBPC,"",5),"D",2)),1:\$STR(\$P(IB0,U,+IBPC),"", ")) .. I \$P(IBPC,"",3),VAL S VAL=\$P(VAL,"")_"_"\$E(\$P(VAL,"",2)"00",1,2) .. S DR=DR_IBFLD_"//"_\$S(\$P(IBPC,"",4):"/",1:"")_VAL_"", ; ; S DR=\$P(DR,"",2,\$L(DR,"")-1) I DR'="" D ^DIE Q (\$D(Y)=0) ;Successfully stored all the data it was sent if \$D(Y)=0 ; ; HDR(IB0,IBEGBL,IBEOB,HIPAA) ; Store header data for EOB ; IB0 = the record being processed from the msg ; IBEOB = the ien of the EOB entry in file 361.1 ; ; N IBDT,IBDTP,DA,DR,DIE,X,Y K IBXSAVE("XTRA"),IBZSAVE ; S HIPAA=+\$P(IB0,U,16) ;HIPAA Version code S IBDT=\$P(IB0,U,3),IBDT=\$E(IBDT,1,4)-1700_\$E(IBDT,5,8)_"_"\$P(IB0,U,4) S IBDTP=\$P(IB0,U,9) I IBDTP S IBDTP=\$E(IBDTP,1,4)-1700_\$E(IBDTP,5,8) S DR=\$S(\$P(IB0,U,7)'="" :".03//"_\$P(IB0,U,7)_"",1:"")_"_.05//"_IBDT_"_.04//"_(\$P(IB0,U,5)="Y")_"_.15//"_ \$\$COBN^IBCEF(+\$G(^IBM(361.1,IBEOB,0)))_"_.07//"_\$P(IB0,U,8)_\$S(IBDTP:"_.06//"_IBDTP,1:"") S DIE="^IBM(361.1,"DA=IBEOB D ^DIE I \$D(Y)'=0 D </pre> |

| Routine Name | IBCEOB1 |
|--------------|--|
| | <pre> . S ^TMP(IBEGBL,\$J,+\$(^TMP(IBEGBL,\$J,""),-1)+1)="Bad header data" Q (\$D(Y)=0) ; FINDLN(IB0,IBEOB,IBZDATA,PLREF) ; Find corresponding billed line for the adj ; IB0 = the record being processed ; NOTE: pieces 3,4,16 are already reformatted ; IBEOB = the ien of the EOB entry in file 361.1 ; IBZDATA = the array from the output formatter containing line ; items for the bill. This is passed in so this data only has ; to be extracted once for each bill (the first time in, it ; will be undefined) ; PLREF = Provider Line Reference ; OUTPUT = Line # in the original bill that this adjustment relates to ; ^ paid procedure code if different from original procedure OR ; paid rev code if different from original and no proc code ; N IBLN,IBLN1,IBBNDL,OCHG,OCHG2,OPROC,OREVCD,IBIFN,IBXARRAY,IBXARRY N IBXERR,UNITS,UNITS2,UNITS3,IBMOD,Z,Z0,CPT,EOBCHG,IBZVLA,IBAMIN ; S (IBLN,IBLN1)="",IBIFN=+\$(^IBM(361.1,IBEOB,0)) S EOBCHG=+\$\$DOLLAR^IBCEOB(\$P(IB0,U,15)) ; charges on EOB 40 record ; ; if original procedure exists and is different than the 835 procedure, ; the procedure or revenue code originally billed will be in piece 10 ; of the '40' record of the 835 flat file. Otherwise, pc 10 is null. S IBBNDL=\$(\$P(IB0,U,10))=""':1,1:0) ; ; If this is a split MRA, build array of Vista line#'s from other split MRA's I \$P(\$G(^IBM(361.1,IBEOB,0)),U,4)=1,\$\$SPLIT^IBCEMU1(IBEOB) D . N IEN S IEN=0 . F S IEN=\$O(^IBM(361.1,"B",IBIFN,IEN)) Q:'IEN I IEN'=IBEOB D .. I \$P(\$G(^IBM(361.1,IEN,0)),U,4)'=1 Q ; not an MRA .. I '\$\$SPLIT^IBCEMU1(IEN) Q ; not a split EOB .. M IBZVLA=^IBM(361.1,IEN,15,"AC") .. Q . Q ; I \$P(\$G(^DGCR(399,IBIFN,0)),U,19)=3 D G FINDLN ; UB-04 format . I '\$D(IBZDATA) D F^IBCEF("N-UB-04 SERVICE LINE (EDI)","IBZDATA",,IBIFN) . I +PLREF,\$D(IBZDATA(+PLREF)) S IBLN=+PLREF_U_\$P(IB0,U,10) Q ; . S Z=0 F S Z=\$O(IBZDATA(Z)) Q:'Z D Q:IBLN .. ; Quit if processing an MRA and this VistA line# has already been filed .. I \$P(\$G(^IBM(361.1,IBEOB,0)),U,4)=1,\$D(^IBM(361.1,IBEOB,15,"AC",Z)) Q .. ; Quit if split MRA and this VistA line# has already been filed .. I \$D(IBZVLA(Z)) Q .. I \$G(IBZDATA(Z))="" Q .. S OCHG=\$P(IBZDATA(Z),U,3)*\$P(IBZDATA(Z),U,4) ; Total charge from bill .. S OCHG2=+\$P(IBZDATA(Z),U,5) .. I OCHG'=EOBCHG,OCHG2=EOBCHG S OCHG=OCHG2 ; update OCHG </pre> |

| Routine Name | IBCEOB1 |
|--------------|--|
| | <pre> .. ; .. S CPT=\$P(IBZDATA(Z),U,2) ; proc from bill .. I CPT'?.N,CPT'="" S CPT=\$O(^ICPT("B",CPT,"")) ; non-numeric proc .. S OPROC=\$\$PRCD^IBCEF1(+CPT_";ICPT(") ; ext proc code .. S OREVCD=+\$P(\$G(^DGCR(399.2,+IBZDATA(Z),0)),U) ; Rev cd from bill .. ; .. ; if not bundled/unbundled .. I 'IBBNDL D Q ... I OPROC="",OREVCD,OREVCD'=\$P(IB0,U,4) Q ; revenue code ... I OPROC="",OPROC'=\$P(IB0,U,3) Q ; procedure code ... I +\$P(IBZDATA(Z),U,4)'=\$P(IB0,U,16) Q ; original units ... I +OCHG'=EOBCHG Q ; original charges ... I '\$\$MODMATCH(\$P(IBZDATA(Z),U,9),\$P(IB0,U,5,8)),\$\$MODMATCH(\$P(\$P(IBZDATA(Z),U,9),"",1),\$P(IB0,U,5)) Q ; modifiers ... S IBLN=Z ... Q .. ; .. ; if bundled/unbundled .. I IBBNDL D Q ... I OPROC="",OREVCD,OREVCD'=\$P(IB0,U,10) Q ; revenue code ... I OPROC="",OPROC'=\$P(IB0,U,10) Q ; procedure code ... I +\$P(IBZDATA(Z),U,4)'=\$P(IB0,U,16) Q ; original units ... I +OCHG'=EOBCHG Q ; original charges ... I '\$\$MODMATCH(\$P(IBZDATA(Z),U,9),\$P(IB0,U,11,14)),\$\$MODMATCH(\$P(\$P(IBZDATA(Z),U,9),"",1),\$P(IB0,U,11)) Q ; modifiers ... S IBLN=Z_U_\$S(OPROC'="" :OPROC,1:OREVCD) ... Q .. Q .. ; When dealing with Inpatient UB-04's, check for revenue code roll-ups .. I 'IBLN,\$\$INPAT^IBCEF(IBIFN,1) D RCRU^IBCEOB00(.IBZDATA,IB0,.IBLN) .. ; If only 1 rev code and charges are the same, assume a match .. I 'IBLN,\$P(\$G(^IBM(361.1,IBEOB,0)),U,4),\$O(IBZDATA("")), 1)=\$O(IBZDATA("")),+OCHG'=EOBCHG S IBLN=+\$O(IBZDATA(""))_U_ OREVCD .. ; .. ; At this point, we can assume the claim is CMS-1500 format I '\$D(IBZDATA) D F^IBCEF("N-HCFA 1500 SERVICE LINE (EDI)",,IBZDATA",,IBIFN) I PLREF,\$D(IBZDATA(PLREF)) S IBLN=PLREF_U_\$P(IB0,U,10) G FINDLNX S Z=0 F S Z=\$O(IBZDATA(Z)) Q:'Z D Q:IBLN .. ; Quit if processing an MRA and this VistA line# has already been filed .. I \$P(\$G(^IBM(361.1,IBEOB,0)),U,4)=1,\$D(^IBM(361.1,IBEOB,15,"AC",Z)) Q .. ; Quit if split MRA and this VistA line# has already been filed .. I \$D(IBZVLA(Z)) Q .. S OCHG=\$P(IBZDATA(Z),U,8)*\$P(IBZDATA(Z),U,9) ; charge from bill .. S IBAMIN="" .. I \$P(IBZDATA(Z),U,12)'="" S IBAMIN=\$P(IBZDATA(Z),U,12) ;anesthesia minutes .. S UNITS=\$S('IBAMIN:\$P(IBZDATA(Z),U,9),1:IBAMIN/15) .. ; original units from bill or anesthesia minutes calculation .. I \$P(UNITS,".",2) S UNITS=\$FN(UNITS,"",1) ; round to a single decimal place for fractional units </pre> |

| Routine Name | IBCEOB1 |
|--------------|--|
| | <pre> . I \$P(\$P(IB0,U,16),",",2) S \$P(IB0,U,16)=\$FN(\$P(IB0,U,16),"",1) . S UNITS2=\$P(IBZDATA(Z),U,9) ; just the units . ; UNITS3 is the number of anesthesia minutes divided by 10, or nil. . ; Solution to get around the Trailblazers bug for MRAs . S UNITS3="" . I IBAMIN'=0 S UNITS3=IBAMIN/10 . ; . S CPT=\$P(IBZDATA(Z),U,5) ; proc from bill . I CPT'?.N,CPT'="" S CPT=\$O(^ICPT("B",CPT,"")) ; non-numeric proc . S OPROC=\$\$PRCD^IBCEF1(+CPT_",ICPT(") ; ext proc code . Q:OPROC'=\$S('IBNDL:\$P(IB0,U,3),1:\$P(IB0,U,10)) . ; . S MODOK=0 . I \$\$DOLLAR^IBCEFG1(OCHG)=+\$P(IB0,U,15),UNITS=\$P(IB0,U,16)!(UNITS2=\$P(IB0,U,16))!(UNITS3=\$P(IB0,U,16))!(IBAMIN=\$P(IB0,U,16)),\$\$(\$P(IB0,U,19):\$P(IB0,U,19)=\$P(IBZDATA(Z),U),1:1) D .. ;Original procedure/chg/units/date match to get here .. ;Check matching original modifiers .. S MODOK=\$\$MODMATCH(\$\$MODLST^IBEFUNC2(\$P(IBZDATA(Z),U,10)),\$('IBNDL:\$P(IB0,U,5,8),1:\$P (IB0,U,11,14))) .. I 'MODOK,IBLN1 S IBLN1=Z_\$(IBNDL:U_OPROC,1:"") .. Q:'MODOK .. S IBLN=Z_\$(IBNDL:U_OPROC,1:"") I 'IBLN,IBLN1 S IBLN=IBLN1 ; FINDLNX ; Q IBLN ; MODMATCH(IB,MODLST) ; Match modifiers ; IB = the list of modifiers iens from the bill, comma delimited ; MODLST = the 4 '^' pieces of the reported modifiers ; N MODOK,Q,Z0,IBMOD,MMOD S MODOK=1 I \$TR(IB,",")'="" F Q=1:1:\$L(IB,",") S Z0=\$P(IB,",",Q) I Z0'="" S IBMOD(Z0)=\$G(IBMOD(Z0))+1 I \$TR(MODLST,U)'="" ,\$(IBMOD(""))'="" G MODQ ; No modifiers used ; ; No match if no VistA modifiers, but there are MRA modifiers I \$TR(MODLST,U)'="" ,\$(IBMOD(""))'="" S MODOK=0 G MODQ ; ; Evaluate each MRA modifier F Z0=1:1:4 D . S MMOD=\$P(MODLST,U,Z0) Q:MMOD="" ; individual MRA modifier . I '\$D(IBMOD(MMOD)) Q ; not in array so just quit . S IBMOD(MMOD)=IBMOD(MMOD)-1 ; decrement array counter . I 'IBMOD(MMOD) KILL IBMOD(MMOD) ; if 0, then kill array entry . Q ; I \$(IBMOD(""))'="" S MODOK=0 ; All submitted mods not matched </pre> |

| Routine Name | IBCEOB1 |
|--|---------|
| MODQ Q MODOK ; | |
| Modified Logic (Changes are in bold) | |
| IBCEOB1 ;ALB/TMP/PJH - 835 EDI EOB MSG PROCESSING ; 7/13/10 5:32pm ;;2.0;INTEGRATED BILLING;**137,135,155,296,356,349,431,488**;;21-MAR-94;Build 106 ;;Per VHA Directive 2004-038, this routine should not be modified. Q ; STORE(A,IB0,IBEOB,LEVEL) ; ; A = the string of data to extract and try to store ; each ^ piece is a field to store ; within each ^ piece, there are 5 ';' pieces: ; 1 = piece to extract from the data string; ; 2 = field to update; ; 3 = flag for dollar amt (1=YES); ; 4 = flag for 4-slash stuff without transform (1=YES); ; 5 = flag for numeric/non-dollar amt (1=Yes, 0=No, ; Dn = the field is numeric with 'n' decimal places ; IB0 = the record being processed ; IBEOB = the ien of the EOB entry in file 361.1 ; LEVEL = the array that contains the DIE and DA values if stuffing at a ; level other than the top level ; N B,IBPC,IBFLD,DA,DR,DIE,X,Y S DR="," ; I \$G(LEVEL) S DIE="^IBM(361.1,"DA=IBEOB ; I \$G(LEVEL) D . N Q . S DIE=\$G(LEVEL("DIE")) . S Q=0 F S Q=\$O(LEVEL(Q)) Q:'Q S DA(Q)=LEVEL(Q) . S DA=LEVEL(0) ; I \$G(DA) F B=1:1:\$L(A,U) D . S IBPC=\$P(A,U,B),IBFLD=\$P(IBPC,";",2) . I \$P(IB0,U,+IBPC)'="" ,IBFLD D .. N VAL .. ; For dollar amts, add full cents; For numerics, strip leading .. ; 0's; For non-numeric/non-dollar amts, make any ; in data into .. S VAL=\$S(\$P(IBPC,";",3):\$DOLLAR^IBCEOB(\$P(IB0,U,+IBPC)),\$P(IBPC,";",5):+\$P(IB0,U,+IBPC),\$P(\$P(I BPC,";",5),"D",2):\$P(IB0,U,+IBPC)/(10**\$P(\$P(IBPC,";",5),"D",2)),1:\$TR(\$P(IB0,U,+IBPC),"", ")) .. I \$P(IBPC,";",3),VAL S VAL=\$P(VAL,".")_"_"\$E(\$P(VAL,".",2)"00",1,2) .. S DR=DR_IBFLD_"///"_\$S(\$P(IBPC,";",4):"/",1:"")_VAL_"", ; S DR=\$P(DR,"",2,\$L(DR,"")-1) | |

| Routine Name | IBCEOB1 |
|--------------|---|
| | <pre> I DR'="" D ^DIE Q (\$D(Y)=0) ;Successfully stored all the data it was sent if \$D(Y)=0 ; HDR(IB0,IBEGBL,IBEBOB,HIPAA) ; Store header data for EOB ; IB0 = the record being processed from the msg ; IBEBOB = the ien of the EOB entry in file 361.1 ; N IBDT,IBDTP,DA,DR,DIE,X,Y K IBXSAVE("XTRA"),IBZSAVE ; S HIPAA=+\$P(IB0,U,16) ;HIPAA Version code S IBDT=\$P(IB0,U,3),IBDT=\$E(IBDT,1,4)-1700_\$E(IBDT,5,8)_"_"\$P(IB0,U,4) S IBDTP=\$P(IB0,U,9) I IBDTP S IBDTP=\$E(IBDTP,1,4)-1700_\$E(IBDTP,5,8) S DR=\$S(\$P(IB0,U,7)'="" :".03////_"\$P(IB0,U,7)_"",1:"")_"05////_"_IBDT_"":.04////_"_(\$P(IB0,U,5)="Y")_"":.15////_"_ \$\$COBN^IBCEF(+\$G(^IBM(361.1,IBEBOB,0)))_"":.07////_"\$P(IB0,U,8)\$S(IBDTP:"":.06////_"_IBDTP,1:"") S DIE="^IBM(361.1,"DA=IBEBOB D ^DIE I \$D(Y)'=0 D . S ^TMP(IBEGBL,\$J,+\$(^TMP(IBEGBL,\$J,""),-1)+1)="Bad header data" Q (\$D(Y)=0) ; FINDLN(IB0,IBEBOB,IBZDATA,PLREF) ; Find corresponding billed line for the adj ; IB0 = the record being processed ; NOTE: pieces 3,4,16 are already reformatted ; IBEBOB = the ien of the EOB entry in file 361.1 ; IBZDATA = the array from the output formatter containing line ; items for the bill. This is passed in so this data only has ; to be extracted once for each bill (the first time in, it ; will be undefined) ; PLREF = Provider Line Reference ; OUTPUT = Line # in the original bill that this adjustment relates to ; ^ paid procedure code if different from original procedure OR ; paid rev code if different from original and no proc code ; N IBLN,IBLN1,IBBNDL,OCHG,OCHG2,OPROC,OREVCD,IBIFN,IBXARRAY,IBXARRY N IBXERR,UNITS,UNITS2,UNITS3,IBMOD,Z,Z0,CPT,EOBCHG,IBZVLA,IBAMIN,ERRTYP ; S (IBLN,IBLN1,ERRTYP)="",IBIFN=+\$G(^IBM(361.1,IBEBOB,0)) ; PLEASE NOTE...Should any of the following data gathering be changed...please ; make sure that the same changes are made in the routine IBCEOB00 because these ; 2 routines need to be in sync. S EOBCHG=+\$\$DOLLAR^IBCEOB(\$P(IB0,U,15)) ; charges on EOB 40 record ; ; if original procedure exists and is different than the 835 procedure, ; the procedure or revenue code originally billed will be in piece 10 ; of the '40' record of the 835 flat file. Otherwise, pc 10 is null. S IBBNDL=\$S(\$P(IB0,U,10)'="" :1,1:0) ; ; If this is a split MRA, build array of Vista line#'s from other split MRA's </pre> |

| Routine Name | IBCEOB1 |
|--|---------|
| <pre> I \$P(\$G(^IBM(361.1,IBEOB,0)),U,4)=1,\$\$SPLIT^IBCEMU1(IBEOD) D . N IEN S IEN=0 . F S IEN=\$O(^IBM(361.1,"B",IBIFN,IEN)) Q:'IEN I IEN'=IBEOD D .. I \$P(\$G(^IBM(361.1,IEN,0)),U,4)=1 Q ; not an MRA .. I '\$\$SPLIT^IBCEMU1(IEN) Q ; not a split EOB .. M IBZVLA=^IBM(361.1,IEN,15,"AC") .. Q . Q ; I \$P(\$G(^DGCR(399,IBIFN,0)),U,19)=3 D G FINDLN ; UB-04 format . S ERRTYP="ERR-04" . I '\$D(IBZDATA) D F^IBCEF("N-UB-04 SERVICE LINE (EDI)","IBZDATA",,IBIFN) . I +PLREF,\$D(IBZDATA(+PLREF)) S IBLN=+PLREF_U_\$P(IB0,U,10) Q ; . S Z=0 F S Z=\$O(IBZDATA(Z)) Q:'Z D Q:IBLN .. ; Quit if processing an MRA and this VistA line# has already been filed .. I \$P(\$G(^IBM(361.1,IBEOB,0)),U,4)=1,\$D(^IBM(361.1,IBEOB,15,"AC",Z)) Q .. ; Quit if split MRA and this VistA line# has already been filed .. I \$D(IBZVLA(Z)) Q .. I \$G(IBZDATA(Z))="" Q .. ; .. ; PLEASE NOTE...Should any of the following data gathering be changed...please .. ; make sure that the same changes are made in the routine IBCEOB00 because these .. ; 2 routines need to be in sync. .. S OCHG=\$P(IBZDATA(Z),U,3)*\$P(IBZDATA(Z),U,4) ; Total charge from bill .. S OCHG2=+\$P(IBZDATA(Z),U,5) .. I OCHG'=EOBCHG,OCHG2=EOBCHG S OCHG=OCHG2 ; update OCHG .. ; .. S CPT=\$P(IBZDATA(Z),U,2) ; proc from bill .. I CPT'?.N,CPT'="" S CPT=\$O(^ICPT("B",CPT,"")) ; non-numeric proc .. S OPROC=\$\$PRCD^IBCEF1(+CPT_";ICPT(") ; ext proc code .. S OREVCN=+\$P(\$G(^DGCR(399.2,+IBZDATA(Z),0)),U) ; Rev cd from bill .. ; .. ; if not bundled/unbundled .. I 'IBBNDL D Q ... I OPROC="",OREVCN,OREVCN'=\$P(IB0,U,4) S IBLN=IBLN_U_Z_U_ERRTYP_U_1 Q ; revenue code ... I OPROC="",OPROC'=\$P(IB0,U,3) S IBLN=IBLN_U_Z_U_ERRTYP_U_2 Q ; procedure code ... I +\$P(IBZDATA(Z),U,4)=\$P(IB0,U,16) S IBLN=IBLN_U_Z_U_ERRTYP_U_3 Q ; original units ... I +OCHG'=EOBCHG S IBLN=IBLN_U_Z_U_ERRTYP_U_4 Q ; original charges ... I '\$\$MODMATCH(\$P(IBZDATA(Z),U,9),\$P(IB0,U,5,8)),\$\$MODMATCH(\$P(\$P(IBZDATA(Z),U,9),"",1),\$P(IB 0,U,5)) S IBLN=IBLN_U_Z_U_ERRTYP_U_5 Q ; modifiers ... S IBLN=Z ... Q .. ; .. ; if bundled/unbundled .. I IBBNDL D Q ... I OPROC="",OREVCN,OREVCN'=\$P(IB0,U,10) S IBLN=IBLN_U_Z_U_ERRTYP_U_1 Q ; revenue code </pre> | |

| Routine Name | IBCEOB1 |
|--|---------|
| <pre> ... I OPROC='',"OPROC'=\$P(IB0,U,10) S IBLN=IBLN_U_Z_U_ERRTYP_U_2 Q ; procedure code ... I +\$P(IBZDATA(Z),U,4)=\$P(IB0,U,16) S IBLN=IBLN_U_Z_U_ERRTYP_U_3 Q ; original units ... I +OCHG'=EOBCHG S IBLN=IBLN_U_Z_U_ERRTYP_U_4 Q ; original charges ... I '\$\$MODMATCH(\$P(IBZDATA(Z),U,9),\$P(IB0,U,11,14)),\$\$MODMATCH(\$P(\$P(IBZDATA(Z),U,9),",",1),\$P(IB0,U,11)) S IBLN=IBLN_U_Z_U_ERRTYP_U_5 Q ; modifiers ... S IBLN=Z_U_\$S(OPROC='":OPROC,1:OREVCD) ... Q .. Q . ; When dealing with Inpatient UB-04's, check for revenue code roll-ups . I 'IBLN,\$\$INPAT^IBCEF(IBIFN,1) D RCRU^IBCEOB00(.IBZDATA,IB0,.IBLN) . ; If only 1 rev code and charges are the same, assume a match . I 'IBLN,\$P(\$G(^IBM(361.1,IBEOB,0)),U,4),\$O(IBZDATA("))- 1)=\$O(IBZDATA("))+OCHG=EOBCHG S IBLN=+\$O(IBZDATA("))_U_OREVCD ; . ; At this point, we can assume the claim is CMS-1500 format I '\$D(IBZDATA) D F^IBCEF("N-HCFA 1500 SERVICE LINE (EDI)","IBZDATA",,IBIFN) I PLREF,\$D(IBZDATA(PLREF)) S IBLN=PLREF_U_\$P(IB0,U,10) G FINDLNX S Z=0 F S Z=\$O(IBZDATA(Z)) Q:'Z D Q:IBLN . S ERRTYP="ERR-1500" . ; Quit if processing an MRA and this VistA line# has already been filed . I \$P(\$G(^IBM(361.1,IBEOB,0)),U,4)=1,\$D(^IBM(361.1,IBEOB,15,"AC",Z)) Q . ; Quit if split MRA and this VistA line# has already been filed . I \$D(IBZVLA(Z)) Q . ; . ; PLEASE NOTE...Should any of the following data gathering be changed...please . ; make sure that the same changes are made in the routine IBCEOB00 because these . ; 2 routines need to be in sync. . S OCHG=\$P(IBZDATA(Z),U,8)*\$P(IBZDATA(Z),U,9) ; charge from bill . S IBAMIN="" . I \$P(IBZDATA(Z),U,12)'="" S IBAMIN=\$P(IBZDATA(Z),U,12) ;anesthesia minutes . S UNITS=\$S('IBAMIN:\$P(IBZDATA(Z),U,9),1:IBAMIN/15) . ; original units from bill or anesthesia minutes calculation . I \$P(UNITS,".",2) S UNITS=\$FN(UNITS,"",1) ; round to a single decimal place for fractional units . I \$P(\$P(IB0,U,16),".",2) S \$P(IB0,U,16)=\$FN(\$P(IB0,U,16),"",1) . S UNITS2=\$P(IBZDATA(Z),U,9) ; just the units . ; UNITS3 is the number of anesthesia minutes divided by 10, or nil. . ; Solution to get around the Trailblazers bug for MRAs . S UNITS3="" . I IBAMIN'=0 S UNITS3=IBAMIN/10 . ; . S CPT=\$P(IBZDATA(Z),U,5) ; proc from bill . I CPT'?.N,CPT'="" S CPT=\$O(^ICPT("B",CPT,"")) ; non-numeric proc . S OPROC=\$\$PRCD^IBCEF1(+CPT_",ICPT(") ; ext proc code . ;;;Q:OPROC'=\$S('IBNDL:\$P(IB0,U,3),1:\$P(IB0,U,10)) . I OPROC'=\$S('IBNDL:\$P(IB0,U,3),1:\$P(IB0,U,10)) S IBLN=IBLN_U_Z_U_ERRTYP_U_2 Q ; Mis- matched Proc Code. . ; . S MODOK=0 3333 </pre> | |

| Routine Name | IBCEOB1 |
|--|---------|
| <pre> \$\$\$DOLLAR^IBCEFG1(OCHG)=+\$P(IB0,U,15),UNITS=\$P(IB0,U,16)!(UNITS2=\$P(IB0,U,16))!(UNITS3=\$P(IB0,U,16))!(IBAMIN=\$P(IB0,U,16)),\$\$S(\$P(IB0,U,19):\$P(IB0,U,19)=\$P(IBZDATA(Z),U),1:1) D . I UNITS'=\$P(IB0,U,16),UNITS2'=\$P(IB0,U,16),UNITS3'=\$P(IB0,U,16),IBAMIN'=\$P(IB0,U,16) S IBLN=IBLN_U_Z_U_ERRTYP_U_3 Q . I \$\$\$DOLLAR^IBCEFG1(OCHG)'=+\$P(IB0,U,15) S IBLN=IBLN_U_Z_U_ERRTYP_U_4 Q . I \$\$S(\$P(IB0,U,19):\$P(IB0,U,19)=\$P(IBZDATA(Z),U),1:1) D .. ;Original procedure/chg/units/date have matched in order to get here .. ;Check matching original modifiers .. S MODOK=\$\$MODMATCH(\$\$MODLST^IBEFUNC2(\$P(IBZDATA(Z),U,10)),\$(IBBNDL:\$P(IB0,U,5,8),1:\$P (IB0,U,11,14))) .. I 'MODOK S IBLN=IBLN_U_Z_U_ERRTYP_U_5 .. I 'MODOK,'IBLN1 S IBLN1=Z_\$\$S(IBBNDL:U_OPROC,1: "") .. Q:'MODOK .. S IBLN=Z_\$\$S(IBBNDL:U_OPROC,1: "") I 'IBLN,IBLN1 S IBLN=IBLN1 ; FINDLN ; Q IBLN ; MODMATCH(IB,MODLST) ; Match modifiers ; IB = the list of modifiers iens from the bill, comma delimited ; MODLST = the 4 '^ pieces of the reported modifiers ; N MODOK,Q,Z0,IBMOD,MMOD S MODOK=1 I \$TR(IB,"")'="" F Q=1:1:\$L(IB,"") S Z0=\$P(IB,"",Q) I Z0'="" S IBMOD(Z0)=\$G(IBMOD(Z0))+1 I \$TR(MODLST,U)'="" , \$O(IBMOD(""))'="" G MODQ ; No modifiers used ; ; No match if no VistA modifiers, but there are MRA modifiers I \$TR(MODLST,U)'="" , \$O(IBMOD(""))'="" S MODOK=0 G MODQ ; ; Evaluate each MRA modifier F Z0=1:1:4 D . S MMOD=\$P(MODLST,U,Z0) Q:MMOD="" ; individual MRA modifier . I '\$D(IBMOD(MMOD)) Q ; not in array so just quit . S IBMOD(MMOD)=IBMOD(MMOD)-1 ; decrement array counter . I 'IBMOD(MMOD) KILL IBMOD(MMOD) ; if 0, then kill array entry . Q ; I \$O(IBMOD(""))'="" S MODOK=0 ; All submitted mods not matched MODQ Q MODOK ; </pre> | |

6.2.2.6. System Feature: Third Party Joint Inquiry (TPJI)

This section covers the design for the Functional Requirements listed in Section 2.6.6 of the RSD.

6.2.2.6.1. Functional Requirements: Third Party Joint Inquiry (TPJI)

| SRS REQ ID | REQ Title | Comments / Notes |
|------------|--|------------------|
| 2.6.6.1 | Display Message Storage Errors in TPJI | |

6.2.2.6.2. Routines (Entry Points)

| | | | | |
|---|---|--|--|--|
| Routine Name | IBJTA1 | | | |
| Enhancement Category | <input type="checkbox"/> New <input checked="" type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change | | | |
| Requirement Traceability Matrix | 2.6.6.1 | | | |
| Related Options | Third Party Joint Inquiry [IBJ THIRD PARTY JOINT INQUIRY] | | | |
| Related Routines | Routines "Called By" | Routines "Called" | | |
| | | ^VALM1, ^IBJTU2, ^IBJTLA1, ^IBJTLA, ^VALM10, ^IBJTCA1, ^IBJTCA, ^IBJU1, ^XLFD, ^IBJTLB1, ^IBJTLB, ^IBCECOB6, ^RCJIBFN3, ^IBJTTC, ^XPDUTL, ^DIR, ^GMTSDVR | | |
| Data Dictionary (DD) References | | | | |
| Related Protocols | | | | |
| Related Integration Control Registrations (ICRs) | | | | |
| Data Passing | <input type="checkbox"/> Input <input type="checkbox"/> Output Reference <input type="checkbox"/> Both <input type="checkbox"/> Global Reference <input type="checkbox"/> Local | | | |
| Input Attribute Name and Definition | | | | |
| Output Attribute Name and Definition | | | | |
| Current Logic | | | | |
| <pre> . . . ; EOBERR ; Display information about any 361.1 message storage or filing errors I '\$O(^IBM(361.1,IBI,"ERR",0)) Q S IBSTR=\$\$SETLN^IBJTBA(" ** MESSAGE STORAGE ERRORS </pre> | | | | |

| | |
|---|---------------|
| Routine Name | IBJTA1 |
| <pre> **", "", 1, 79), IBLN=\$\$SET^IBJTBA(IBSTR, IBLN) S Z=0 F S Z=\$O(^IBM(361.1, IBI, "ERR", Z)) Q:'Z S IBSTR=\$\$SETLN^IBJTBA(\$G(^Z, 0)), "", 1, 79), IBLN=\$\$SET^IBJTBA(IBSTR, IBLN) Q </pre> | |
| Modified Logic (Changes are in bold) | |
| <pre> . . . ; EOBERR ; Display information about any 361.1 message storage or filing errors I '\$O(^IBM(361.1, IBI, "ERR", 0)) Q S IBSTR=\$\$SETLN^IBJTBA("***** DATA MISMATCH ERRORS FOUND *****", "", 1, 79), IBLN=\$\$SET^IBJTBA(IBSTR, IBLN) S Z=0 F S Z=\$O(^IBM(361.1, IBI, "ERR", Z)) Q:'Z S IBSTR=\$\$SETLN^IBJTBA(\$G(^Z, 0)), "", 1, 79), IBLN=\$\$SET^IBJTBA(IBSTR, IBLN) Q </pre> | |

6.2.2.7. System Feature: CMS-1500 Form

This section covers the design for the Functional Requirements listed in Section 2.6.7 of the RSD.

6.2.2.7.1. Functional Requirements: CMS-1500 Form

| SRS REQ ID | REQ Title | Comments / Notes |
|------------|-----------------------------------|------------------|
| 2.6.7.1 | Obsolete CMS – 1500 Data Elements | |
| 2.6.7.2 | New CMS – 1500 Data Elements | |

6.2.2.7.2. Routines (Entry Points)

| | | |
|---------------------------------|--|--------------------------|
| Routine Name | IBCEU3 | |
| Enhancement Category | <input type="checkbox"/> New <input checked="" type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change | |
| Requirement Traceability Matrix | 2.6.7.1, 2.6.7.2 | |
| Related Options | FORMATTER/EXTRACT BILL FUNCTIONS | |
| Related Routines | Routines "Called By" | Routines "Called" |

| Routine Name | IBCEU3 | |
|--|---|---|
| | IBCBB2, IBCBB9, IBCEF74, 8BCEF80, IBCEFP, IBCIADD1, IBCINPT, IBCIUT1, IBXSAH, IBXSC8H4 | \$\$GET1^DIQ, ^DIR, \$\$COBCT^IBCEF, \$\$FT^IBCEF, F^IBCEF3, F^IBCEF, B24^IBCEF3, \$\$CFIDS^IBCEF77, \$\$DOL^IBCEF77, \$\$SPEC^IBCEU, GETPRV^IBCEU, \$\$DATE^IBCF2, \$\$FO^IBCNEUT1, \$\$MODLST^IBEFUNC |
| Data Dictionary (DD) References | BILL/CLAIMS File (#399) RATE TYPE File (#399.3) | |
| Related Protocols | | |
| Related Integration Control Registrations (ICRs) | N/A | |
| Data Passing | <input type="checkbox"/> Input <input type="checkbox"/> Output Reference <input type="checkbox"/> Both <input type="checkbox"/> Global Reference <input type="checkbox"/> Local | |
| Input Attribute Name and Definition | | |
| Output Attribute Name and Definition | | |
| Current Logic | | |
| <pre>; BOX19(IBIFN) ; Returns the text that should print in box 19 of the CMS-1500 ; for bill ien IBIFN ; Data is derived from a combo of data throughout ; the system and is limited to 80 characters. The hierarchy for ; including data is as follows (until 80 characters have been used): ; DATE LAST SEEN and REFERRING PHYSICIAN ID# (physical therapy) ; specialty codes = 025,065,073,067,048 ; LAST X-RAY DATE (chiropractic) specialty code = 35 ; HOMEBOUND INDICATOR (independent lab renders an EKG or obtains ; a specimen from a homebound patient) ; NO ASSIGNMENT OF BENEFITS (if no assignment of benefits indicated) ; Hearing aid testing (if applicable) ; ATTENDING PHYSICIAN NOT HOSPICE EMPLOYEE (if applicable) ; SPECIAL PROGRAM indicator if Medicare demonstration project for ; lung volume reduction surgery study is set ; COMMENTS FOUND IN BOX 19 DATA FIELD FOR THE CLAIM ; REMARKS FOUND IN BILL COMMENT FOR THE CLAIM, INCLUDING PROSTHETICS ; DETAIL ; N IBGO,IBHOSP,IBID,IBLSDT,IBXDATA,IB19,IBHAID,IBXRAY,IBSPEC,Z,Z0,IBSUB,IBPRT,IBREM,IBSPI S IB19="",IBGO=1 S IBSUB=\$S('\$G(^TMP("IBTX",\$J,IBIFN)):"BOX24",1:"OUTPT")</pre> | | |

| Routine Name | IBCEU3 |
|--------------|--|
| | <pre> I \$D(IBXSAVE(IBSUB)) N IBXSAVE S IBPRT=(IBSUB["24"]) ; S IBSPEC=\$\$BILLSPEC(IBIFN) G:'IBPRT NPRT ; Check for chiropractic services I \$P(\$G(^DGCR(399,IBIFN,"U3")),U,5)'="" S:\$P(\$G(^DGCR(399,IBIFN,"U3")),U,4)'="" IBGO=\$\$LENOK("Last X-ray: " _\$STR(\$\$DATE^IBCF2(\$P(^DGCR(399,IBIFN,"U3")),U,4))," ","/"),.IB19) G:'IBGO BOX19Q ; I "^25^65^73^67^48^"[(U_IBSPEC_U) D . K IBXDATA D F^IBCEF("N-DATE LAST SEEN",,,IBIFN) . I IBXDATA'="" S IBID="" ,IBLSDT=\$\$DATE^IBCF2(IBXDATA,0,1) D I IBLSDT'="" S IBGO=\$\$LENOK("Date Last Seen:" _IBLSDT_IBID,.IB19) .. ; Only print if specialty is OT or PT or proc for routine foot care .. D F^IBCEF("N-REFERRING PROVIDER ID",,,IBIFN) I IBXDATA'="" S IBID=" By:" _IBXDATA ; G:'IBGO BOX19Q K IBXDATA D F^IBCEF("N-HOMEBOUND",,,IBIFN) I IBXDATA G:\$LENOK("Homebound",.IB19) BOX19Q ; K IBXDATA D F^IBCEF("N-ASSIGN OF BENEFITS INDICATOR",,,IBIFN) I "Nn0"[IBXDATA&(IBXDATA'="") G:\$LENOK("Patient refuses to assign benefits",.IB19) BOX19Q ; I '\$D(IBXSAVE(IBSUB)) D B24^IBCEF3(.IBXSAVE,IBIFN,\$S(\$G(IBNOSHOW)=0:0,1:1)) ; S (IBHAID,IBHOSP,IBXRAY)=0 ; S Z=0 F S Z=\$O(IBXSAVE(IBSUB,Z)) Q:'Z D G:'IBGO BOX19Q . I \$D(IBXSAVE(IBSUB,Z,"RX")), \$P(IBXSAVE(IBSUB,Z,"RX"),U,3)'="" S IBGO=\$\$LENOK("NOC Drug:" _\$P(IBXSAVE(IBSUB,Z,"RX"),U,2) _" Units:" _+\$P(IBXSAVE(IBSUB,Z,"RX"),U,6),.IB19) . ; . Q:'IBGO . I 'IBHAID,\$P(IBXSAVE(IBSUB,Z),U,5)="V5010",\$\$COBCT^IBCEF(IBIFN)>1 D Q .. S IBHAID=1,IBGO=\$\$LENOK("Testing for hearing aid",.IB19) Q . ; . Q:'IBGO . I 'IBHOSP,\$P(\$G(IBXSAVE(IBSUB,Z,"AUX")),U,3) S IBHOSP=1,IBGO=\$\$LENOK("Attending physician,not hospice employee",.IB19) Q G:'IBGO BOX19Q K IBXDATA D F^IBCEF("N-SPECIAL PROGRAM",,,IBIFN) I IBXDATA=30 G:\$LENOK("Medicare demonstration project for lung volume reduction surgery study",.IB19) BOX19Q ; </pre> |

| Routine Name | IBCEU3 |
|--------------|---|
| | <pre> ; SPECIAL PROGRAM INDICATOR field code. S IBSPI=\$\$GET1^DIQ(399,IBIFN_"",238,"E") I IBSPI="" S IBGO=\$\$LENOK(IBSPI,.IB19) ; G:'IBGO BOX19Q NPRT K IBXDATA D F^IBCEF("N-HCFA 1500 BOX 19 RAW DATA",,,IBIFN) S IBREM=0 I IBXDATA="" G:'\$\$LENOK("Remarks:"_IBXDATA,.IB19) BOX19Q S IBREM=1 K IBXDATA D F^IBCEF("N-BILL REMARKS",,,IBIFN) I IBXDATA="" G:'\$\$LENOK(\$\$('IBREM:"Remarks:",1:"")_IBXDATA,.IB19) BOX19Q ; BOX19Q Q IB19 ; LENOK(IBDATA,IB19) ; Add text IBDATA to box 19 string (IB19 passed by ref) ; Check length of box 19 data - truncate at 96 (max length) ; Returns 0 if max length reached or exceeded, otherwise, 1 N OK S OK=1 S IB19=IB19_\$\$S(IB19=""_"",1:"")_G(IBDATA) I \$L(IB19)<96 S OK=0,IB19=\$E(IB19,1,96) G LENOKQ LENOKQ Q OK ; ASK19(IBIFN) ; Ask to display CMS-1500 box 19 data for current IBIFN N DIR,DIC,X,Y,DIE,DR,Z S DIR(0)="YA",DIR("B")="NO",DIR("A")="DISPLAY THE FULL CMS-1500 BOX 19?: " D ^DIR I Y=1 S Z=\$\$BOX19(IBIFN) W !,?4,"19",?20,\$E(Z,1,32) W:\$L(Z)>32 !,?4,\$E(Z,33,80),! Q ; </pre> |
| | Modified Logic (Changes are in bold) |
| | <pre> ; BOX19(IBIFN) ; Nex Box 19. This is for workman's comp only? ; This returns the Paperwork Attachment ; Information in the following format: ; • PWKNNFX12348907CHEY<3 Spaces>Next set if more than one on claim ; This information can be at either the Line Level or the Claim Level. ; Check all Lines first and print as many as possible – 83 characters ; maximum. Then check the Claim Level N IBRTP,LN,U8,IBBX19,IB19,LN,DATA,I,DEL S IB19="",DEL="" ; Get rate type S IBRTP=\$P(\$G(DGCR(399,IBIFN,0)),U,7) </pre> |

| Routine Name | IBCEU3 |
|---|--------|
| <pre> ; Get data entered for box 19 S IBBX19=\$P(\$G(^DGCR(399,IBIFN,"UF31")),U,3) ; check the line Level first I IB RTP=11 D .S LN=0 .F S LN=\$O(^DGCR(399,IBIFN,"CP",LN)) Q:LN="" D ..S DATA=\$G(^DGCR(399,IBIFN,"CP",LN,1)) ..I DATA="" S IB19=\$S(IB19="": "",1:DEL)_"PWK"_\$P(DATA,U,2)_\$P(DATA,U,3)_\$P(DATA,U,1) .; check the Claim Level next .S DATA="" .S DATA=\$G(^DGCR(399,IBIFN,"U8")) .I DATA="" S IB19=\$S(IB19="": "",1:DEL)_"PWK"_\$P(DATA,U,2)_\$P(DATA,U,3)_\$P(DATA,U,1) ; If any room left add user entered box 19 info I \$L(IB19)<84 D .F I=1:1:\$L(IBBX19,DEL) S DATA=\$P(IBBX19,DEL,I) D ..I \$L(IB19_DEL_DATA)<84 S IB19=\$S(IB19="": "",1:DEL)_DATA ; S IB19=\$E(IB19,1,83) ; ONLY 83 CHARCTERS ALLOWED ; Q IB19 OBOX19(IBIFN) ; THIS IS NO LONGER USED. IT WAS REPLACE WITH ABOVE. ; Returns the text that should print in box 19 of the CMS-1500 ; LENOK(IBDATA,IB19) ; Add text IBDATA to box 19 string (IB19 passed by ref) ; Check length of box 19 data - truncate at 83 (max length) ; Returns 0 if max length reached or exceeded, otherwise, 1 N OK S OK=1 S IB19=IB19_\$S(IB19'="" : " ",1:"")_\$G(IBDATA) I \$L(IB19)'<83 S OK=0,IB19=\$E(IB19,1,83) G LENOKQ LENOKQ Q OK ; ASK19(IBIFN) ; Ask to display CMS-1500 box 19 data for current IBIFN N DIR,DIC,X,Y,DIE,DR,Z S DIR(0)="YA",DIR("B")="NO",DIR("A")="DISPLAY THE FULL CMS-1500 BOX 19?: " D ^DIR I Y=1 S Z=\$BOX19(IBIFN) W !,?4,"19",?20,\$E(Z,1,32) W:\$L(Z)>32 !,?4,\$E(Z,33,83),! Q ; </pre> | |

| | | | | | |
|---|---|---|--|--|--|
| Routine Name | IBCF2 | | | | |
| Enhancement Category | <input type="checkbox"/> New <input checked="" type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change | | | | |
| Requirement Traceability Matrix | 2.6.7.1, 2.6.7.2 | | | | |
| Related Options | Print Authorized Bills [IB BATCH PRINT BILLS] | | | | |
| Related Routines | Routines "Called By" | Routines "Called" | | | |
| | IBCBB2, IBCEF, IBCEF11, IBCEU3, IBCF2, IBCF21, IBCF22, IBCF23, IBCF23A, IBCF2TP, IBCF3, IBCF33, IBCFP1, IBCOMD1, IBYPENV | ^%ZIS, HOME^%ZIS, ^%ZISC, ^%ZTLOAD, ^DIE, BSTAT^IBCDC, F^IBCEF, \$\$EXTRACT^IBCEFG, \$\$NAME^IBCEFG, \$\$NAME^IBCEFG1, ENFMT^IBCF, ^IBCF21, ^IBCF2P, \$\$FTN^IBCU3, \$\$NEEDMRA^IBEFUNC, ADD^VADPT,ZIPOUT^VAFADDR, \$\$FMTE^XLFD | | | |
| Data Dictionary (DD) References | BILL/CLAIMS File (#399) BILL FORM TYPE File (#353) PATIENT File (#2) | | | | |
| Related Protocols | | | | | |
| Related Integration Control Registrations (ICRs) | N/A | | | | |
| Data Passing | <input type="checkbox"/> Input <input type="checkbox"/> Output Reference <input type="checkbox"/> Both <input type="checkbox"/> Global Reference <input type="checkbox"/> Local | | | | |
| Input Attribute Name and Definition | | | | | |
| Output Attribute Name and Definition | | | | | |
| Current Logic | | | | | |
| <pre> ARRAY ; F IBI=1:1:6 S IBFLD(0,IBI)="" F IBI=1:1:21,23:1:26,28:1:33 S IBFLD(IBI)="" F IBI=10,16,18 F IBJ="A","B" S IBFLD(IBI_IBJ)="" F IBI="10BS","10C","11AX","11B","11C","11D","1A","3D","3X","5S","5T","8E","8M","9A","9BD","9BX","9C","9D","17A" S IBFLD(IBI)="" Q </pre> | | | | | |
| Modified Logic (Changes are in bold) | | | | | |
| <pre> ARRAY ; F IBI=1:1:6 S IBFLD(0,IBI)="" F IBI=1:1:21,23:1:26,28:1:33 S IBFLD(IBI)="" F IBI=10,16,18 F IBJ="A","B" S IBFLD(IBI_IBJ)="" F IBI="10BS","10C","10D","11AX","11B","11C","11D","1A","3D","3X","5S","5T","8E","8M","9A","9BD","9BX","9C","9D","17A" S IBFLD(IBI)="" Q </pre> | | | | | |

| | | | | |
|--|---|--|--|--|
| Routine Name | IBCF21 | | | |
| Enhancement Category | <input type="checkbox"/> New <input checked="" type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change | | | |
| Requirement Traceability Matrix | 2.6.7.1, 2.6.7.2 | | | |
| Related Options | Print Authorized Bills [IB BATCH PRINT BILLS] | | | |
| Related Routines | Routines "Called By" | Routines "Called" | | |
| | IBCF2 | \$\$DATE^IBCF2, \$\$STATE^IBCF2, ^IBCF22, ELIG^VADPT, OAD^VADPT | | |
| Data Dictionary (DD) References | BILL/CLAIMS File (#399) MCCR UTILITY File (#399.1) | | | |
| Related Protocols | | | | |
| Related Integration Control Registrations (ICRs) | N/A | | | |
| Data Passing | <input type="checkbox"/> Input <input type="checkbox"/> Output Reference <input type="checkbox"/> Both <input type="checkbox"/> Global Reference <input type="checkbox"/> Local | | | |
| Input Attribute Name and Definition | | | | |
| Output Attribute Name and Definition | | | | |
| Current Logic | | | | |
| <pre> INS1 G INS2:IB(IBPRIM)="!"(\$D(^DIC(36,+IB(IBPRIM),0))) F IBI=\$P(IB(IBPRIM),U,2),\$P(IB(IBPRIM),U,3) I IBI="" S IBFLD("1A")=IBI Q ;policy number S IBFLD(4)=\$S(IBRIP=1:"SAME",1:\$P(IB(IBPRIM),U,17)) ; insureds name S IBFLD(6)=\$S(\$P(IB(IBPRIM),U,16):IBRIP,1:\$P(IB(IBPRIM),U,16)) ; patient relationship to insured I IBRIP=1!(IBRIP=2) S IBFLD(7)="SAME" ; insured's address ; I \$P(IB(IBPRIM),U,2)="" S IBFLD(11)=\$P(IB(IBPRIM),U,3) ; group number I IBRIP=1 S IBFLD("11AD")=IBFLD("3D"),IBFLD("11AX")=IBFLD("3X") I +IBRIP=1,IBFLD("8E")="E" S VAOA("A")=5 D OAD^VADPT S IBFLD("11B")=VAOA(9) K VAOA ;employer I +IBRIP=2 D . I IBFLD("3X")="" S X="MFM",IBFLD("11AX")=\$E(X,\$F(X,IBFLD("3X"))) . I IBSPE="E" S VAOA("A")=6 D OAD^VADPT S IBFLD("11B")=VAOA(9) K VAOA ;spouses employer S IBFLD("11C")=\$P(IB(IBPRIM),U,15) ; INS2 G COND:IB(IBSECD)="!"(\$D(^DIC(36,+IB(IBSECD),0))) ; secondary insurance S IBFLD("11D")=1 S IBFLD(9)=\$P(IB(IBSECD),U,17) I IBFLD(9)="" ,IBFLD(9)=\$P(IB(IBPRIM),U,17) S IBFLD(9)="SAME" ;secondary insureds nam F IBI=\$P(IB(IBSECD),U,2),\$P(IB(IBSECD),U,3) I IBI="" S IBFLD("9A")=IBI Q ;policy number I +IBRIS=1 D . S IBFLD("9BD")=IBFLD("3D"),IBFLD("9BX")=IBFLD("3X") . I IBFLD("8E")="E" S VAOA("A")=5 D OAD^VADPT S IBFLD("9C")=VAOA(9) K VAOA ;employer I +IBRIS=2 D . I IBFLD("3X")="" S X="MFM",IBFLD("9BX")=\$E(X,\$F(X,IBFLD("3X"))) </pre> | | | | |

| Routine Name | IBCF21 |
|--------------|---|
| | <pre> . I IBSPE="E" S VAOA("A")=6 D OAD^VADPT S IBFLD("9C")=VAOA(9) K VAOA ;spouses employer I IBFLD("9A")=\$P(IB(IBSECD),U,3) S IBFLD("9D")=\$P(IB(IBSECD),U,15) ;group name I IBFLD("9D")="" S IBFLD("9D")=\$P(\$G(^DIC(36,+IB(IBSECD),0)),U) ;company name ; ; COND ;condition related to employment, auto accident (place), other accident S IBI=0 F S IBI=\$O(^DGCR(399,IBIFN,"CC",IBI)) Q:'IBI S X=\$G(^IBI,0)) I +X D . S Y=\$G(^DGCR(399.1,+X,0)) Q:Y="" I \$P(Y,U,2)="02" S IBFLD("10A")=1 S IBI=0 F S IBI=\$O(^DGCR(399,IBIFN,"OC",IBI)) Q:'IBI S X=\$G(^IBI,0)) I +X D . S Y=\$G(^DGCR(399.1,+X,0)) Q:Y="" . I \$P(Y,U,9)=1 S IBFLD("10A")=1 . I \$P(Y,U,9)=2 S IBFLD("10B")=1 S X=\$\$STATE^IBCF2(\$P(X,U,3)) I X="" S IBFLD("10BS")=X . I \$P(Y,U,9)=3 S IBFLD("10C")=1 . I \$P(Y,U,1)="ONSET OF SYMPTOMS/ILLNESS" S IBFLD(15)=\$\$DATE^IBCF2(\$P(X,U,2),1) ; see DATES+1^IBCF22 ; ; K IBRI1,IBRI2,IBRI3,IBCOB,IBPRIM,IBSECD,IBRIP,IBRIS D ^IBCF22 Q </pre> |
| | Modified Logic (Changes are in bold) |
| | <pre> INS1 G INS2:IB(IBPRIM)=""!('\$D(^DIC(36,+IB(IBPRIM),0))) F IBI=\$P(IB(IBPRIM),U,2),\$P(IB(IBPRIM),U,3) I IBI="" S IBFLD("1A")=IBI Q ;policy number S IBFLD(4)=\$S(IBRIP=1:"SAME",1:\$P(IB(IBPRIM),U,17)) ; insureds name S IBFLD(6)=\$S(\$P(IB(IBPRIM),U,16):IBRIP,1:\$P(IB(IBPRIM),U,16)) ; patient relationship to insured I IBRIP=1!(IBRIP=2) S IBFLD(7)="SAME" ; insured's address ; I \$P(IB(IBPRIM),U,2)="" S IBFLD(11)=\$P(IB(IBPRIM),U,3) ; group number I IBRIP=1 S IBFLD("11AD")=IBFLD("3D"),IBFLD("11AX")=IBFLD("3X") I +IBRIP=1,IBFLD("8E")="E" S VAOA("A")=5 D OAD^VADPT S IBFLD("11B")=VAOA(9) K VAOA ;employer I +IBRIP=2 D . I IBFLD("3X")="" S X="MFM",IBFLD("11AX")=\$E(X,\$F(X,IBFLD("3X"))) . I IBSPE="E" S VAOA("A")=6 D OAD^VADPT S IBFLD("11B")=VAOA(9) K VAOA ;spouses employer S IBFLD("11C")=\$P(IB(IBPRIM),U,15) ; INS2 G COND:IB(IBSECD)=""!('\$D(^DIC(36,+IB(IBSECD),0))) ; secondary insurance S IBFLD("11D")=1 S IBFLD(9)=\$P(IB(IBSECD),U,17) I IBFLD(9)="" ,IBFLD(9)=\$P(IB(IBPRIM),U,17) S IBFLD(9)="SAME" ;secondary insureds nam F IBI=\$P(IB(IBSECD),U,2),\$P(IB(IBSECD),U,3) I IBI="" S IBFLD("9A")=IBI Q ;policy number I +IBRIS=1 D . S IBFLD("9BD")=IBFLD("3D"),IBFLD("9BX")=IBFLD("3X") . I IBFLD("8E")="E" S VAOA("A")=5 D OAD^VADPT S IBFLD("9C")=VAOA(9) K VAOA ;employer ;I +IBRIS=2 D ;I IBFLD("3X")="" S X="MFM",IBFLD("9BX")=\$E(X,\$F(X,IBFLD("3X"))) ;I IBSPE="E" S VAOA("A")=6 D OAD^VADPT S IBFLD("9C")=VAOA(9) K VAOA ;spouses employer I IBFLD("9A")=\$P(IB(IBSECD),U,3) S IBFLD("9D")=\$P(IB(IBSECD),U,15) ;group name I IBFLD("9D")="" S IBFLD("9D")=\$P(\$G(^DIC(36,+IB(IBSECD),0)),U) ;company name ; ; ; </pre> |

| | |
|---|---------------|
| Routine Name | IBCF21 |
| COND ;condition related to employment, auto accident (place), other accident S IBI=0 F S IBI=\$O(^DGCR(399,IBIFN,"CC",IBI)) Q:'IBI S X=\$G(^IBI,0)) I +X D . S Y=\$G(^DGCR(399.1,+X,0)) Q:Y="" I \$P(Y,U,2)="02" S IBFLD("10A")=1 S IBI=0 F S IBI=\$O(^DGCR(399,IBIFN,"OC",IBI)) Q:'IBI S X=\$G(^IBI,0)) I +X D . S Y=\$G(^DGCR(399.1,+X,0)) Q:Y="" . I \$P(Y,U,9)=1 S IBFLD("10A")=1 . I \$P(Y,U,9)=2 S IBFLD("10B")=1 S X=\$\$STATE^IBCF2(\$P(X,U,3)) I X'="" S IBFLD("10BS")=X . I \$P(Y,U,9)=3 S IBFLD("10C")=1 . I \$P(Y,U,1)="ONSET OF SYMPTOMS/ILLNESS" S IBFLD(15)=\$\$DATE^IBCF2(\$P(X,U,2),1) ; see DATES+1^IBCF22 ; K IBRI1,IBRI2,IBRI3,IBCOB,IBPRIM,IBSECD,IBRIP,IBRIS D ^IBCF22 Q | |

| | | | |
|---|---|---|--|
| Routine Name | IBCF22 | | |
| Enhancement Category | <input type="checkbox"/> New <input checked="" type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change | | |
| Requirement Traceability Matrix | 2.6.7.1, 2.6.7.2 | | |
| Related Options | Print Authorized Bills [IB BATCH PRINT BILLS] | | |
| Related Routines | Routines “Called By” | Routines “Called” | |
| | ^IBCBB2 ^IBCF21 | \$\$ICD9^IBACSV, F^IBCEF, \$\$DATE^IBCF2, \$\$STATE^IBCF2, ^IBCF23, SET^IBCSC4D, \$\$NEEDMRA^IBEFUNC, \$\$BN1^PRCAFN, INP^VADPT | |
| Data Dictionary (DD) References | BILL/CLAIMS File (#399) PATIENT MOVEMENT File (#405) IB SITE PARAMETERS File (#350.9) | | |
| Related Protocols | | | |
| Related Integration Control Registrations (ICRs) | N/A | | |
| Data Passing | <input type="checkbox"/> Input <input type="checkbox"/> Output Reference <input type="checkbox"/> Both <input type="checkbox"/> Global Reference <input type="checkbox"/> Local | | |
| Input Attribute Name and Definition | | | |
| Output Attribute Name and Definition | | | |
| Current Logic | | | |
| IBCF22 ;ALB/ARH - HCFA 1500 19-90 DATA (gather other data) ;12-JUN-93 | | | |

| Routine Name | IBCF22 |
|--------------|---|
| | <pre> ;;2.0;INTEGRATED BILLING;**52,80,122,51,210**,21-MAR-94 ;;Per VHA Directive 10-93-142, this routine should not be modified. ; ; requires DFN, IBIFN, IB(0) F IBI="C","U","U1","U2","UF2" S IB(IBI)=\$G(^DGCR(399,IBIFN,IBI)) S IBFLD(12)="PUBLIC LAW 99-272/SECTION 1729 TITLE 38" S IBFLD(13)="PUBLIC LAW 99-272" DATES S IBFLD(14)=\$\$DATE(\$\$EVENT(IBIFN)) I \$G(IBFLD(15))="",IBIFN'=\$P(IB(0),U,17) S IBFLD(15)=\$\$DATE(\$P(\$G(^DGCR(399,+P(IB(0),U,17),0)),U,3)) S IBFLD("16A")=\$\$DATE(\$P(IB("U"),U,16)),IBFLD("16B")=\$\$DATE(\$P(IB("U"),U,17)) S:\$NEEDMRA^IBEFUNC(IBIFN) IBFLD(17)="Dept. Of Veterans Affairs" I \$P(IB(0),U,5)<3 S IBFLD("18A")=\$\$DATE(\$P(IB("U"),U,1)),IBFLD("18B")=\$\$DATE(\$P(IB("U"),U,2)) I \$P(IB(0),U,5)>2 S VAINDT=\$P(IB(0),U,3) D INP^VADPT I +VAIN(1) D . S IBFLD("18A")=\$\$DATE(VAIN(7)),IBFLD("18B")=\$\$DATE(+\$G(^DGPM(+P(\$G(^DGPM(+VAIN(1),0)),U,17),0))) K VAINDT,VAIN S IBFLD(19)="THE UNDERSIGNED CERTIFIES TREATMENT IS NOT FOR A SERVICE-CONNECTED CONDITION" S IBFLD(20)=0 ; DX ;S X=14 F IBI="21A","21B","21C","21D" S IBFLD(IBI)=\$P(\$G(^ICD9(+P(IB("C"),U,X),0)),U,1),X=X+1 ;F IBI="21A","21B","21C","21D" S IBFLD(IBI)="" ;N IBINDXX D SET^IBCSC4D(IBIFN,"",.IBINDXX) S X=0,Y="21@" D ;. F S X=\$O(IBINDXX(X)) Q:'X S Y=\$O(IBFLD(Y)) Q:+Y'=21 S IBFLD(Y)=\$P(\$G(^ICD9(+IBINDXX(X),0)),U,1) ; N IBDXX,IBPOX D SET^IBCSC4D(IBIFN,.IBDXX,.IBPOX) S X=0 F IBI=1:1:4 S IBFLD(21,IBI)="" I IBI'>\$P(IBPOX,U,2) D . S X=\$O(IBPOX(X)) Q:X="" . S IBFLD(21,IBI)=\$P(\$G(^ICD9^IBACSV(+IBPOX(X)),U) . S IBDXI(+G(IBDXX(+IBPOX(X))))=IBI ; S IBFLD(23)=\$P(IB("U"),U,13) EVENT(IBIFN,IBXSAVE,IBERR,IBD) ; The event date for box 14 on the ; HCFA 1500 ; IBIFN = bill ien ; IBXSAVE = the array returned by the output formatter for data element ; N-OCCURRENCE CODES ; Returns IBERR=1 if passed by reference meaning more than one condition </pre> |

| Routine Name | IBCF22 |
|---|--------|
| <pre> ; has been found ; IBD("LMP"), IBD("ACC"), IBD("ONS"), IBD("EVT") returned with ; Last menstrual period date, accident date, date of onset, ; event date if IBD passed by reference ; Function returns the appropriate date ; N Z,Z0,IBX,IBF,A ; ; Default if no applicable occurrence codes found is event date on bill S IBX=\$P(\$G(^DGCR(399,IBIFN,0)),U,3),IBF=0 S IBD("EVT")=IBX ; I '\$D(IBXSAVE("OCC")) D F^IBCEF("N-OCCURRENCE CODES",,,IBIFN) S Z=0 F S Z=\$O(IBXSAVE("OCC",Z)) Q:'Z S Z0(+IBXSAVE("OCC",Z))=\$P(IBXSAVE("OCC",Z),U,2) I \$O(Z0(5.99),-1) D . S A=\$O(Z0(5.99),-1),IBF=IBF+1 ;Accident codes 1-5 . S IBD("ACC")=Z0(A) S:IBF>1 IBX=Z0(A) I \$D(Z0(10)) S IBF=IBF+1,IBD("LMP")=IBX S:IBF>1 IBX=Z0(10) ;Last Menstrual period I \$D(Z0(11)) S (IBD("ONS"),IBX)=Z0(11),IBF=IBF+1 ;Onset of Illness ; S IBERR=(IBF>1) Q IBX ; DATE(X) ; format date(X) as MM DD YYYY Q \$\$DATE^IBCF2(X,1) </pre> | |
| Modified Logic (Changes are in bold) | |
| <pre> IBCF22 ;ALB/ARH - HCFA 1500 19-90 DATA (gather other data) ;12-JUN-93 ;;2.0;INTEGRATED BILLING;**52,80,122,51,210**;21-MAR-94 ;;Per VHA Directive 10-93-142, this routine should not be modified. ; ; requires DFN, IBIFN, IB(0) F IBI="C","U","U1","U2","UF2" S IB(IBI)=\$G(^DGCR(399,IBIFN,IBI)) S IBFLD(12)="PUBLIC LAW 99-272/SECTION 1729 TITLE 38" S IBFLD(13)="PUBLIC LAW 99-272" DATES ; S IBFLD(14)=\$\$QUAL(IBIFN,14),IBFLD(14.1)=\$P(IBFLD(14),U,2) S IBFLD(14)=\$\$DATE(\$P(IBFLD(14),U,1)) S IBFLD(15)=\$\$QUAL(IBIFN,15),IBFLD(15.1)=\$P(IBFLD(15),U,2) S IBFLD(15)=\$\$DATE(\$P(IBFLD(15),U,1)) S IBFLD("16A")=\$\$DATE(\$P(IB("U"),U,16)),IBFLD("16B")=\$\$DATE(\$P(IB("U"),U,17)) S:\$NEEDMRA^IBEFUNC(IBIFN) IBFLD(17)="Dept. Of Veterans Affairs" I \$P(IB(0),U,5)<3 S IBFLD("18A")=\$\$DATE(\$P(IB("U"),U,1)),IBFLD("18B")=\$\$DATE(\$P(IB("U"),U,2)) I \$P(IB(0),U,5)>2 S VAINDT=\$P(IB(0),U,3) D INP^VADPT I +VAIN(1) D . S IBFLD("18A")=\$\$DATE(VAIN(7)),IBFLD("18B")=\$\$DATE(+\$G(^DGPM(+\$P(\$G(^DGPM(+VAIN(1),0)),U,1 </pre> | |

| Routine Name | IBCF22 |
|--|--------|
| <pre> 7),0))) K VAINDT,VAIN S IBFLD(19)="THE UNDERSIGNED CERTIFIES TREATMENT IS NOT FOR A SERVICE-CONNECTED CONDITION" S IBFLD(20)=0 ; ; DX ;S X=14 F IBI="21A","21B","21C","21D" S IBFLD(IBI)=\$P(\$G(^ICD9(+P(IB("C"),U,X),0)),U,1),X=X+1 ;F IBI="21A","21B","21C","21D" S IBFLD(IBI)=" ;N IBINDXX D SET^IBCSC4D(IBIFN,"",.IBINDXX) S X=0,Y="21@" D ; F S X=\$O(IBINDXX(X)) Q:'X S Y=\$O(IBFLD(Y)) Q:+Y'=21 S IBFLD(Y)=\$P(\$G(^ICD9(+IBINDXX(X),0)),U,1) ; ; N IBDXX,IBPOX D SET^IBCSC4D(IBIFN,.IBDXX,.IBPOX) S X=0 F IBI=1:1:12 S IBFLD(21,IBI)=" IBI'>\$P(IBPOX,U,2) D . S X=\$O(IBPOX(X)) Q:X="" . S IBFLD(21,IBI)=\$P(\$G(^ICD9^IBACSV(+IBPOX(X)),U) . S IBDXI(+G(IBDXX(+IBPOX(X))))=IBI ; ; S IBFLD(23)=\$P(IB("U"),U,13) DATE(X) ; format date(X) as MM DD YYYY Q \$\$DATE^IBCF2(X,1) ; ; QUAL(IBIFN,IBXBOX,IBXSAVE,IBD) ; The event date for box 14 on the ; HCFA 1500 ; IBIFN = bill ien ; IBXBOX = BOX 14 OR BOX 15 of CMS-1500 form ; IBXSAVE = the array returned by the output formatter for data element ; N-OCCURRENCE CODES ; ; IBD("LMP"), IBD("ACC"), IBD("ONS"), IBD("EVT") returned with ; Last menstrual period date, accident date, date of onset, ; event date if IBD passed by reference ; Function returns the appropriate date ; ; N Z,Z0,IBX,IBF,A ; ; I '\$D(IBXSAVE("OCC")) D F^IBCEF("N-OCCURRENCE CODES",,,IBIFN) ; ; I IBXBOX=14 D ; Default if no applicable occurrence codes found is event date on bill . I '\$D(Z0(11)) S (IBD("ONS"),IBX)=Z0(11),IBF=IBF+1,IBX=IBX_U_431 ;Onset of Illness . I '\$D(Z0(10)) S IBF=IBF+1,IBD("LMP")=IBX S:IBF>1 IBX=Z0(10)_U_484 ;Last Menstrual period ; ; I IBXBOX=15 D . S IBX="" . D ACC I IBX'="" Q . D LXRY I IBX'="" Q . D AMCC I IBX'="" Q . D SCPT I IBX'="" Q . D INTTRT I IBX'="" Q </pre> | |

| Routine Name | IBCF22 |
|---|--------|
| <pre> .D LVC I IBX="" Q .D AMCC ; Q IBX ; ACC ;Accident 439 .S Z=0 F S Z=\$O(IBXSAVE("OCC",Z)) Q:'Z S Z0(+IBXSAVE("OCC",Z))=\$P(IBXSAVE("OCC",Z),U,2) .I \$O(Z0(5.99),-1) D .. S A=\$O(Z0(5.99),-1),IBF=IBF+1 ;Accident codes 1-5 .. S IBD("ACC")=Z0(A) S:IBF>1 IBX=Z0(A) .. I IBX="" S IBX=IBX_U_"439" Q ; LXRY ; Last X-Ray 455 S IBX=\$P(\$G(^DGCR(399,IBIFN,"U3")),U,4) I IBX="" S IBX=IBX_U_"455",IBD("AMC")=IBX ; AMCC ;Acute Manifestation of Chronic Condition 453 S IBX=\$P(\$G(^DGCR(399,IBIFN,"U3")),U,6) I IBX="" S IBX=IBX_U_"453",IBD("AMC")=IBX Q ; SCPT ; Prescription 471 N IBRX,RXNM,RXDT D SET^BCSC5A(IBIFN,.IBRX) I 'IBRX Q S RXNM=\$O(IBRX("")) I RXNM="" Q S RXDT=\$O(IBRX(RXNM,"")) I RXDT="" Q S IBX=RXDT_U_"471" Q ; LVC ;Latest Visit or Consultation 304 S IBXDATA="" D F^IBCEF("N-DATE LAST SEEN",,,IBIFN) I IBXDATA="" S IBD("LVC")=IBXDATA,IBX=IBXDATA_U_"304" Q ; INTTRT ;Initial Treatment 454 S IBX=\$P(\$G(^DGCR(399,IBIFN,"U3")),U,5) I IBX="" S IBX=IBX_U_"454",IBD("INT")=IBX Q ; AMCC ;Acute Manifestation of Chronic Condition 453 S IBX=\$P(\$G(^DGCR(399,IBIFN,"U3")),U,6) I IBX="" S IBX=IBX_U_"453",IBD("AMC")=IBX Q .. I IBX="" S IBX=IBX_U_"439" Q ; </pre> | |

| Routine Name | IBCF2P |
|--------------|--------|
|--------------|--------|

| | | | | |
|--|---|--------------------------|--|--|
| Routine Name | IBCF2P | | | |
| Enhancement Category | <input type="checkbox"/> New <input checked="" type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change | | | |
| Requirement Traceability Matrix | 2.6.7.1, 2.6.7.2 | | | |
| Related Options | Print Authorized Bills [IB BATCH PRINT BILLS] | | | |
| Related Routines | Routines "Called By" | Routines "Called" | | |
| | IBCF2, IBYPENV | \$\$MOD^ICPTMOD | | |
| Data Dictionary (DD) References | IB SITE PARAMETERS File (#350.9) | | | |
| Related Protocols | | | | |
| Related Integration Control Registrations (ICRs) | N/A | | | |
| Data Passing | <input type="checkbox"/> Input <input type="checkbox"/> Output Reference <input type="checkbox"/> Both <input type="checkbox"/> Global Reference <input type="checkbox"/> Local | | | |
| Input Attribute Name and Definition | | | | |
| Output Attribute Name and Definition | | | | |
| Current Logic | | | | |
| IBCF2P ;ALB/ARH - PRINT HCFA 1500 12-90 FORM ; 17-JUL-93 ;;2.0;INTEGRATED BILLING;**8,52,133**;21-MAR-94 ;;Per VHA Directive 10-93-142, this routine should not be modified. ; PRINT ; print the form, IBFLD required S IBADDM=+\$P(\$G(^IBE(350.9,1,1)),U,27),IBPAGE=1 F IBI=1:1:6 W !,?IBADDM,\$G(IBFLD(0,IBI)) ;mailing address LINE8 ; insured's ID number W !!,?49,\$E(IBFLD("1A"),1,28) LINE10 ; patient name, DOB, sex; insured's name ("SAME" if patient) W !!,\$E(IBFLD(2),1,28),?30,IBFLD("3D"),?(\$S(IBFLD("3X")="M":41,1:46)), "X",?49,\$E(IBFLD(4),1,28) LINE12 ; patient's address (street); pt. rel to ins.; insured's address W !!,\$E(IBFLD(5,1),1,28),?(\$S(+IBFLD(6)=1:32,+IBFLD(6)=2:37,+IBFLD(6)=3:41,1:46)), "X",?49,\$E(IBFLD(7),1,28) LINE14 ; patient addr city, state code, marital status; insured's city, state W !!,\$E(IBFLD(5,2),1,24),?25,IBFLD("5S"),?(\$S(IBFLD("8M")="S":34,IBFLD("8M")="M":40,1:46)), "X",?49 LINE16 ; patient zip code, phone; patient employment status; insured's zip code, phone W !!,\$E(IBFLD(5,3),1,12),?14,IBFLD("5T"),?34,\$S(IBFLD("8E")="E": "X",1:""),?49 LINE18 ; other insured's name; insured's policy group W !!,\$E(IBFLD(9),1,28),?49,\$E(IBFLD(11),1,28) LINE20 ; other insured's policy number; condition related to employment?; insured's date of birth and sex W !!,\$E(IBFLD("9A"),1,28),?(\$S(+IBFLD("10A"):34,1:40)), "X" W ?53,IBFLD("11AD") I IBFLD("11AX")="" W ?(\$S(IBFLD("11AX")="M":67,1:74)), "X" LINE22 ; other insured's DOB, sex; patient auto accident & place; insured's employer W !!,?1,IBFLD("9BD") I IBFLD("9BX")="" W ?(\$S(IBFLD("9BX")="M":17,1:23)), "X" | | | | |

| Routine Name | IBCF2P |
|--------------|--|
| | <pre> W ?(\$S(+IBFLD("10B"):34,1:40)), "X", ?44, IBFLD("10BS"), ?49, \$E(IBFLD("11B"), 1, 28) LINE24 ; other insured's employer; patient other accident; insured's insurance plan name W !, \$E(IBFLD("9C"), 1, 28), ?(\$S(+IBFLD("10C"):34,1:40)), "X", ?49, \$E(IBFLD("11C"), 1, 28) LINE26 ; other insured's plan name; is there another benefit plan W !, \$E(IBFLD("9D"), 1, 28), ?(\$S(+IBFLD("11D"):51,1:56)), "X" LINE29 ; patient's signature; insured's signature (use PL 99-272, SECTION 1729 TITLE 38) W !, ?3, IBFLD(12), ?56, IBFLD(13) LINE32 ; date of current illness; date of similar illness; dates unable to work W !, ?1, IBFLD(14), ?36, IBFLD(15), ?53, IBFLD("16A"), ?67, IBFLD("16B") LINE34 ; name of referring physician; ID# referring physician; hospitalization dates related to services W !, ?53, IBFLD("18A"), ?67, IBFLD("18B") LINE35 ; "not for SC" note line 1 W !, ?14, \$E(IBFLD(19), 1, 31) _ "-" LINE36 ; "not for SC" note line 2; outside lab (now defaults to "no" in IBEHCFA) W !, \$E(IBFLD(19), 32, 999), ?56, "X" LINE38 ; diagnosis codes 1 and 2; field 22 (MEDICAID) left blank W !, ?2, IBFLD(21, 1), ?30, IBFLD(21, 3) LINE40 ; diagnosis codes 3 and 4; field 23 (prior authorization #) W !, ?2, IBFLD(21, 2), ?30, IBFLD(21, 4), ?49, IBFLD(23) LINE44 ; lines 44, 46, 48, 50, 52, 54 all the same W ! S IBI=+\$P(IBFLD(24), U, 2) F IBJ=1:1:6 S IBI=IBI+1 D . W ! I \$D(IBFLD(24, IBI_ "A")) W ?25, \$E(IBFLD(24, IBI_ "A"), 1, 15) . W ! I \$D(IBFLD(24, IBI)) D S IBFLD(24)=IBFLD(24)-1 .. W \$P(IBFLD(24, IBI), U, 1), ?9, \$P(IBFLD(24, IBI), U, 2), ?18, \$P(IBFLD(24, IBI), U, 3), ?21, \$P(IBFLD(24, IBI), U, 4), ? 25, \$E(\$P(IBFLD(24, IBI), U, 5), 1, 15) .. I \$P(IBFLD(24, IBI), U, 9) W ?32, \$P(\$MOD^CPTMOD(+\$P(IBFLD(24, IBI), U, 9), "I", DT), U, 2) .. W ?41, \$P(IBFLD(24, IBI), U, 6), ?48, \$S(+\$P(IBFLD(24, IBI), U, 7):\$J(\$P(IBFLD(24, IBI), U, 7), 9, 2), 1:""), ?57, \$J(\$ P(IBFLD(24, IBI), U, 8), 3) S \$P(IBFLD(24), U, 2)=IBI ; ; LINE56 W !, IBFLD(25), ?18, "X", ?22, IBFLD(26), ?49, \$J(IBFLD(28), 10, 2) W:IBFLD(29) ?62, \$J(IBFLD(29), 7, 2), ?71, \$J(IBFLD(30), 7, 2) LINE58 W !, ?22, \$E(IBFLD(32, 1), 1, 26), ?49, \$E(IBFLD(33, 1), 1, 26) LINE69 W !, \$E(IBFLD(31), 1, 21), ?22, \$E(IBFLD(32, 2), 1, 26), ?49, \$E(IBFLD(33, 2), 1, 26) LINE60 W !, \$E(IBFLD(31), 22, 42) W ?22, \$E(IBFLD(32, 3), 1, (26-2-\$L(IBFLD(32, "X")))) I IBFLD(32, "X")="" W " ", "_IBFLD(32, "X") W ?49, \$E(IBFLD(33, 3), 1, (26-2-\$L(IBFLD(33, "X")))) I IBFLD(33, "X")="" W " ", "_IBFLD(33, "X") LINE61 W !, \$E(IBFLD(31), 43, 63), ?49, IBFLD(33, 4) ; I +IBFLD(24)>0 D G LINE38 ;multiple pages . S IBPAGE=IBPAGE+1 . W @IOF, I, ?IBADDM, "PAGE ", IBPAGE, !!!!!, ?49, IBFLD("1A"), !, IBFLD(2) . F IBI=1:1:26 W ! ; ; END K IBADDM, IBPAGE Q </pre> |
| | <p>Modified Logic (Changes are in Highlighted)</p> <pre> IBCF2P ;ALB/ARH - PRINT HCFA 1500 12-90 FORM ; 17-JUL-93 ;;2.0;INTEGRATED BILLING;**8,52,133**;21-MAR-94 ;;Per VHA Directive 10-93-142, this routine should not be modified. ; ; </pre> |

| Routine Name | IBCF2P |
|--------------|---|
| | <p>PRINT ; print the form, IBFLD required S IBADDM=+\$P(\$G(^IBE(350.9,1,1)),U,27),IBPAGE=1 F IBI=1:1:6 W !,?IBADDM,\$G(IBFLD(0,IBI)) ;mailing address LINE8 ; insured's ID number W !,?49,\$E(IBFLD("1A"),1,28) LINE10 ; patient name, DOB, sex; insured's name ("SAME" if patient) W !,\$E(IBFLD(2),1,28),?30,IBFLD("3D"),?(\$S(IBFLD("3X")="M":41,1:46)), "X",?49,\$E(IBFLD(4),1,28) LINE12 ; patient's address (street); pt. rel to ins.; insured's address W !,\$E(IBFLD(5,1),1,28),?(\$S(+IBFLD(6)=1:32,+IBFLD(6)=2:37,+IBFLD(6)=3:41,1:46)), "X",?49,\$E(IBFLD(7),1,28) LINE14 ; patient addr city, state code, marital status; insured's city, state W !,\$E(IBFLD(5,2),1,24),?25,IBFLD("5S"),?49 LINE16 ; patient zip code, phone; patient employment status; insured's zip code, phone W !,\$E(IBFLD(5,3),1,12),?14,IBFLD("5T"),?49 LINE18 ; other insured's name; insured's policy group W !,\$E(IBFLD(9),1,28),?49,\$E(IBFLD(11),1,28) LINE20 ; other insured's policy number; condition related to employment?; insured's date of birth and sex W !,\$E(IBFLD("9A"),1,28),?(\$S(+IBFLD("10A"):34,1:40)), "X" W ?53,IBFLD("11AD") I IBFLD("11AX")="" W ?(\$S(IBFLD("11AX")="M":67,1:74)), "X" LINE22 ; other insured's DOB, sex; patient auto accident & place; insured's employer ;W !,?1,IBFLD("9BD") I IBFLD("9BX")="" W ?(\$S(IBFLD("9BX")="M":17,1:23)), "X" W ?(\$S(+IBFLD("10B"):34,1:40)), "X",?44,IBFLD("10BS"),?49,\$P(IBFLD("11B",U,2),?51,\$P(IBFLD("11B"),U,2) LINE24 ; other insured's employer; patient other accident; insured's insurance plan name W !,?(\$S(+IBFLD("10C"):34,1:40)), "X",?49,\$E(IBFLD("11C"),1,28) LINE26 ; other insured's plan name; is there another benefit plan W !,\$E(IBFLD("9D"),1,28),?30,IBFLD(10D),?(\$S(+IBFLD("11D"):51,1:56)), "X" LINE29 ; patient's signature; insured's signature (use PL 99-272, SECTION 1729 TITLE 38) W !!!,?3,IBFLD(12),?56,IBFLD(13) LINE32 ; date of current illness; date of similar illness; dates unable to work W !!!,?1,IBFLD(14),?16,IBFLD(14.1),?31,IBFLD(15.1),?36,IBFLD(15),?53,IBFLD("16A"),?67,IBFLD("16B") LINE34 ; name of referring physician; ID# referring physician; hospitalization dates related to services W !,?53,IBFLD("18A"),?67,IBFLD("18B") LINE35 ; "not for SC" note line 1 W !,?14,\$E(IBFLD(19),1,23)_"-" LINE36 ; "not for SC" note line 2; outside lab (now defaults to "no" in IBEHCFA) W !,\$E(IBFLD(19),24,999),?56,"X" LINE38 ; diagnosis indicator (0 = ICD9, 1 = ICD10) W !,?42,IBFLD(21,0) LINE39 ; diagnosis codes 1 - 4; field 22 (MEDICAID) left blank W !,?3,IBFLD(21,1),?16,IBFLD(21,2),?29,IBFLD(21,3),?41,IBFLD(21,4) LINE40 ; diagnosis codes 5 - 8; field 23 (prior authorization #) W !,?3,IBFLD(21,5),?16,IBFLD(21,6),?29,IBFLD(21,7),?41,IBFLD(21,8)),?49,IBFLD(23 LINE41 ; diagnosis codes 9 - 12; field 23 (prior authorization #) W !,?3,IBFLD(21,9),?16,IBFLD(21,10),?29,IBFLD(21,11),?41,IBFLD(21,12) LINE44 ;lines 44,46,48,50,52,54 all the same W !! S IBI=+\$P(IBFLD(24),U,2) F IBJ=1:1:6 S IBI=IBI+1 D . W ! I \$D(IBFLD(24,IBI_"A")) W ?25,\$E(IBFLD(24,IBI_"A"),1,15) . W ! I \$D(IBFLD(24,IBI)) D S IBFLD(24)=IBFLD(24)-1 .. W \$P(IBFLD(24,IBI),U,1),?9,\$P(IBFLD(24,IBI),U,2),?18,\$P(IBFLD(24,IBI),U,3),?21,\$P(IBFLD(24,IBI),U,4),? 25,\$E(\$P(IBFLD(24,IBI),U,5),1,15) .. I \$P(IBFLD(24,IBI),U,9) W ?32,\$P(\$MOD^ICPTMOD(+\$P(IBFLD(24,IBI),U,9),"I",DT),U,2)</p> |

| Routine Name | IBCF2P |
|--|--------|
| <pre> .. W ?41,\$P(IBFLD(24,IBI),U,6),?48,\$S(+\$P(IBFLD(24,IBI),U,7):\$J(\$P(IBFLD(24,IBI),U,7),9,2),1:""),?57,\$J(\$ P(IBFLD(24,IBI),U,8),3) S \$P(IBFLD(24),U,2)=IBI ; LINE56 W !,IBFLD(25),?18,"X",?22,IBFLD(26),?49,\$J(IBFLD(28),10,2) W:IBFLD(29) ?62,\$J(IBFLD(29),7,2) ;,?71,\$J(IBFLD(30),7,2) LINE58 W !,?22,\$E(IBFLD(32,1),1,26),?49,\$E(IBFLD(33,1),1,26) LINE69 W !,\$E(IBFLD(31),1,21),?22,\$E(IBFLD(32,2),1,26),?49,\$E(IBFLD(33,2),1,26) LINE60 W !,\$E(IBFLD(31),22,42) W ?22,\$E(IBFLD(32,3),1,(26-2-\$L(IBFLD(32,"X")))) I IBFLD(32,"X")'="" W " , "_IBFLD(32,"X") W ?49,\$E(IBFLD(33,3),1,(26-2-\$L(IBFLD(33,"X")))) I IBFLD(33,"X")'="" W " , "_IBFLD(33,"X") LINE61 W !,\$E(IBFLD(31),43,63),?49,IBFLD(33,4) ; I +IBFLD(24)>0 D G LINE38 ;multiple pages . S IBPAGE=IBPAGE+1 . W @IOF,I,?IBADDM,"PAGE ",IBPAGE,!!!!!,?49,IBFLD("1A"),!,IBFLD(2) . F IBI=1:1:26 W ! ; END K IBADDM,IBPAGE Q </pre> | |

| Routine Name | IBCF2P | |
|--|---|------------------------------------|
| Enhancement Category | <input type="checkbox"/> New <input checked="" type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change | |
| Requirement Traceability Matrix | 2.6.7.1, 2.6.7.2 | |
| Related Options | Print Authorized Bills [IB BATCH PRINT BILLS] Print Bill [IB PRINT BILL] | |
| Related Routines | Routines "Called By" | Routines "Called" |
| | | ^%ZIS, HOME^%ZIS, ^%ZISC, ^%ZTLOAD |
| Data Dictionary (DD) References | IB SITE PARAMETERS File (#350.9) | |
| Related Protocols | | |
| Related Integration Control Registrations (ICRs) | N/A | |
| Data Passing | <input type="checkbox"/> Input <input type="checkbox"/> Output Reference <input type="checkbox"/> Both <input type="checkbox"/> Global Reference <input type="checkbox"/> Local | |
| Input Attribute Name and Definition | | |
| Output Attribute Name and Definition | | |

| Routine Name | IBCF2TP |
|---|---------|
| Current Logic | |
| <pre> IBCF2TP ;ALB/RLW - HCFA 1500 12-90 TEST PATTERN ; 1-JAN-92 ;;Version 2.0 ; INTEGRATED BILLING ;; 21-MAR-94 ;;Per VHA Directive 10-93-142, this routine should not be modified. ZIS ; .*** ; ;S XRTL=\$ZU(0),XRTN="IBCF2TP-1" D T0^%ZOSV ;start rt clock ;S DGPGM="PRINT^IBCF2TP",DGVAR="^" D ZIS^DGUTQ I POP G Q S %ZIS="QM" D ^%ZIS G:POP Q I \$D(IO("Q")) K IO("Q") S ZTRTN="PRINT^IBCF2TP",ZTDESC="IB - TEST HCFA 1500 PRINT" D ^%ZTLOAD K ZTSK D HOME^%ZIS G Q U IO D PRINT Q K DGPGM,DGVAR,POP,IBADDM .*** ; ;I \$D(XRT0) S:\$D(XRTN) XRTN="IBCF2TP" D T1^%ZOSV ;stop rt clock Q PRINT ; print test pattern .*** ; ;I \$D(XRT0) S:\$D(XRTN) XRTN="IBCF2TP" D T1^%ZOSV ;stop rt clock ;S XRTL=\$ZU(0),XRTN="IBCF2TP-2" D T0^%ZOSV ;start rt clock N X S X="" CARRIER ; S IBADDM=+\$P(\$G(^IBE(350.9,1,1)),U,27) W:\$E(IOST,1,2) ["C-" @IOF W !,?IBADDM,"INSURANCE CARRIER NAME",!?IBADDM,"CARRIER ADDRESS LINE 1",!?IBADDM,"CARRIER ADDRESS LINE 2",!?IBADDM,"CARRIER ADDRESS LINE 3",!?IBADDM,"CARRIER CITY, STATE ZIP" LINE2 W !!!,X,?7,X,?14,X,?23,X,?30,X,?38,X,?44,X,?49,"SUBSCRIBER ID#" LINE4 W !!,"PATIENT NAME",?30,"MM DD YY",?41,X,?46,X,?49,"INSURED'S NAME" LINE6 W !!,"PATIENT ADDRESS STREET",?32,X,?37,X,?41,X,?46,X,?49,"INSURED'S ADDRESS STREET" LINE8 W !!,"PATIENT ADDRESS CITY",?25,"ST",?34,X,?40,X,?46,X,?49,"INSURED'S ADDRESS CITY",?73,"ST" LINE10 W !!,"PT ZIP CODE",?14,"999",?18,"999-9999",?34,X,?40,X,?46,X,?49,"INS ZIP CODE",?64,"999",?68,"999-9999" LINE12 W !!,"OTHER INSURED'S NAME",?49,"INSURED'S POLICY GROUP" LINE14 W !!,"OTHER POLICY NUMBER",?34,X,?40,X,?53,"MM DD YY",?67,X,?74,X LINE16 W !!,"MM DD YY",?17,X,?23,X,?34,X,?40,X,?44,"ST",?49,"INSURED'S EMPLOYER" LINE18 W !!,"OTHER'S EMPLOYER",?34,X,?40,X,?49,"INSURANCE PLAN NAME" LINE20 W !!,"OTHER'S INSURANCE PLAN",?51,X,?56,X LINE26 W !!!!!,?1,"MM DD YY",?36,"MM DD YY",?53,"MM DD YY",?67,"MM DD YY" LINE28 W !!,"REFERRING PHYSICIAN",?27,"PHYSICIAN ID",?53,"MM DD YY",?67,"MM DD YY" LINE30 W !!,"?51,X,?56,X,?61,"9999.99",?70,"9999.99" LINE32 W !!,"?2,"X99.99",?29,"X99.99" LINE34 W !!,"?2,"X99.99",?29,"X99.99" LINE38 ;lines 38,40,42,44,46,48 all the same W !! F I=1:1:6 W !!,"MM DD YY",?9,"MM DD YY",?18,X,?21,X,?25,"CPT",?32,"MODIF",?41,"DIAG",?50,"9999.99",?58,X,?61,X,?64,X,?67,X,?70,"BC/BS #" LINE50 W !!,"FEDERAL TAX ID",?18,X,?22,"PAT ACCT#",?37,X,?42,X,?52,"9999.99",?62,"9999.99",?71,"9999.99" LINE52 W !!,"?22,"VAMC",?49,"AGENT CASHIER (999) 999-9999" </pre> | |

| Routine Name | IBCF2TP |
|--|---------|
| LINE53 W !,?22,"STREET ADDRESS",?49,"STREET ADDRESS" LINE54 W !,?22,"CITY, STATE ZIP",?49,"CITY, STATE ZIP" I \$D(ZTQUEUED) S ZTREQ="@ " Q D ^%ZISC .*** ; ;I \$D(XRT0) S:\$D(XRTN) XRTN="IBCF2TP" D T1^%ZOSV ;stop rt clock Q | |
| Modified Logic (Changes are in Highlighted) | |
| IBCF2TP ;ALB/RLW - HCFA 1500 12-90 TEST PATTERN ; 1-JAN-92 ;;Version 2.0 ; INTEGRATED BILLING ;; 21-MAR-94 ;;Per VHA Directive 10-93-142, this routine should not be modified. ZIS ; .*** ; ;S XRTL=\$ZU(0),XRTN="IBCF2TP-1" D T0^%ZOSV ;start rt clock ;S DGPGM="PRINT^IBCF2TP",DGVAR="^" D ZIS^DGUTQ I POP G Q S %ZIS="QM" D ^%ZIS G:POP Q I \$D(IO("Q")) K IO("Q") S ZTRTN="PRINT^IBCF2TP",ZTDESC="IB - TEST HCFA 1500 PRINT" D ^%ZTLOAD K ZTSK D HOME^%ZIS G Q U IO D PRINT Q K DGPGM,DGVAR,POP,IBADDM .*** ; ;I \$D(XRT0) S:\$D(XRTN) XRTN="IBCF2TP" D T1^%ZOSV ;stop rt clock Q PRINT ; print test pattern .*** ; ;I \$D(XRT0) S:\$D(XRTN) XRTN="IBCF2TP" D T1^%ZOSV ;stop rt clock ;S XRTL=\$ZU(0),XRTN="IBCF2TP-2" D T0^%ZOSV ;start rt clock N X S X="" CARRIER ; S IBADDM=+\$P(\$G(^IBE(350.9,1,1)),U,27) W:\$E(IOST,1,2)@"C-" @IOF W !,?IBADDM,"INSURANCE CARRIER NAME",!?IBADDM,"CARRIER ADDRESS LINE 1",!?IBADDM,"CARRIER ADDRESS LINE 2",!?IBADDM,"CARRIER ADDRESS LINE 3",!?IBADDM,"CARRIER CITY, STATE ZIP" LINE2 W !!!,X,?7,X,?14,X,?23,X,?30,X,?38,X,?44,X,?49,"SUBSCRIBER ID#" LINE4 W !!,"PATIENT NAME",?30,"MM DD YY",?41,X,?46,X,?49,"INSURED'S NAME" LINE6 W !!,"PATIENT ADDRESS STREET",?32,X,?37,X,?41,X,?46,X,?49,"INSURED'S ADDRESS STREET" LINE8 W !!,"PATIENT ADDRESS CITY",?25,"ST",?49,"INSURED'S ADDRESS CITY",?73,"ST" LINE10 W !!,"PT ZIP CODE",?14,"999",?18,"999-9999",?34,X,?40,X,?46,X,?49,"INS ZIP CODE",?64,"999",?68,"999-9999" LINE12 W !!,"OTHER INSURED'S NAME",?49,"INSURED'S POLICY GROUP" LINE14 W !!,"OTHER POLICY NUMBER",?34,X,?40,X,?53,"MM DD YY",?67,X,?74,X LINE16 W !!,?34,X,?40,X,?44,"ST", LINE18 W !!,?34,X,?40,X,?49,"XX",?52,"PROPERTY/CAS #" LINE20 W !!,"OTHER'S INSURANCE PLAN",?51,X,?56,X LINE26 W !!!!!,?1,"MM DD YY",?15,"999",?28,"999",?36,"MM DD YY",?53,"MM DD YY",?67,"MM DD YY" LINE28 W !!,"XX",?4,"REFERRING PHYSICIAN",?27,"PHYSICIAN ID",?53,"MM DD YY",?67,"MM DD YY" LINE30 W !!,?51,X,?56,X,?61,"9999.99",?70,"9999.99" LINE31 W !,?40,"X" LINE32 W !,?2,"X99999",?15,"X99999",?28,"X99999",?40,"X99999" LINE34 W !,?2,"X99999",?15,"X99999",?28,"X99999",?40,"X99999" | |

| | |
|--|----------------|
| Routine Name | IBCF2TP |
| LINE35 W !,?2,"X99999",?15,"X99999",?28,"X99999",?40,"X99999" LINE38 ;lines 38,40,42,44,46,48 all the same W !! F I=1:1:6 W !,"MM DD YY",?9,"MM DD YY",?18,X,?21,X,?25,"CPT",?32,"MODIF",?41,"DIAG",?50,"9999.99",?58,X,?61,X,?64,X,?67,X,?70,"BC/BS #" LINE50 W !,"FEDERAL TAX ID",?18,X,?22,"PAT ACCT#",?37,X,?42,X,?52,"9999.99",?62,"9999.99",?71,"9999.99" LINE52 W !,?22,"VAMC",?49,"AGENT CASHIER (999) 999-9999" LINE53 W !,?22,"STREET ADDRESS",?49,"STREET ADDRESS" LINE54 W !,?22,"CITY, STATE ZIP",?49,"CITY, STATE ZIP" I \$D(ZTQUEUED) S ZTREQ="@ " Q D ^%ZISC .*** ; ;I \$D(XRT0) S:\$D(XRTN) XRTN="IBCF2TP" D T1^%ZOSV ;stop rt clock Q | |

6.2.2.7.3. Unique Record(s) for CMS – 1500

IB DATA ELEMENT : N-OTH CLAIM QUAL (HCFA 1500)

| Field Name(s) | Current Value | New Value |
|----------------------|---------------|---|
| .01 Name | | N-OTH CLAIM QUAL (HCFA 1500) |
| .02 Security Level | | N |
| .03 Type of Element | | E |
| .04 Element Category | | I |
| .05 Base File | | 399 |
| 1 Extract Code | | S IBXDATA=\$P((\$G(^DGCR(399,IBIFN, "U4")),U,2) S IBXDATA=\$S(IBXDATA'=""."Y4",1."") |
| 3 Description | | This is the Qualifier for the Other Claim ID. It will always be "Y4" |

IB DATA ELEMENT : N-OTHER CLAIM ID (HCFA 1500)

| Field Name(s) | Current Value | New Value |
|---------------------|---------------|------------------------------|
| .01 Name | | N-OTHER CLAIM ID (HCFA 1500) |
| .02 Security Level | | N |
| .03 Type of Element | | E |

| Field Name(s) | Current Value | New Value |
|----------------------|---------------|--|
| .04 Element Category | | I |
| .05 Base File | | 399 |
| 1 Extract Code | | S IBXDATA=\$P((\$G(^DGCR(399,IBFN,"U4")),U,2) |
| 3 Description | | This Claim ID is designated by NUCC. It is the Property and Casualty Number. |

IB DATA ELEMENT : N-CLAIM CODES (HCFA 1500)

| Field Name(s) | Current Value | New Value |
|----------------------|---------------|---------------------------|
| .01 Name | | N-CLAIM CODES (HCFA 1500) |
| .02 Security Level | | N |
| .03 Type of Element | | E |
| .04 Element Category | | I |
| .05 Base File | | 399 |
| 1 Extract Code | | |
| 3 Description | | |

IB DATA ELEMENT : N-REFER PROV QUAL (HCFA 1500)

| Field Name(s) | Current Value | New Value |
|----------------------|---------------|---|
| .01 Name | | N-REFER PROV QUAL (HCFA 1500) |
| .02 Security Level | | N |
| .03 Type of Element | | E |
| .04 Element Category | | I |
| .05 Base File | | 399 |
| 1 Extract Code | | S IBXDATA=\$\$PRQUAL^NBCEF83(IBXIEN) |
| 3 Description | | Check for a Referring Doctor. If there is one set the Qualifier to "DN". If not a referring doctor, check for Supervising Doctor. If there is a supervising doctor set the Qualifier to "DQ". |

IB DATA ELEMENT : N-CURR ILLNESS DATE QUAL (HCFA 1500)

| Field Name(s) | Current Value | New Value |
|----------------------|---------------|---|
| .01 Name | | N-CURR ILLNESS DATE QUAL (HCFA 1500) |
| .02 Security Level | | N |
| .03 Type of Element | | E |
| .04 Element Category | | I |
| .05 Base File | | 399 |
| 1 Extract Code | | N IBZ S IBXDATA=\$P(\$\$QUAL^IBCF22(IBXIE N,14,.IBXSAVE,.IBZ),U,2) |
| 3 Description | | Qualifier for box 14 |

IB DATA ELEMENT : N-OTHER DATE QUAL (HCFA 1500)

| Field Name(s) | Current Value | New Value |
|----------------------|---------------|---|
| .01 Name | | N-OTHER DATE QUAL (HCFA 1500) |
| .02 Security Level | | N |
| .03 Type of Element | | E |
| .04 Element Category | | I |
| .05 Base File | | 399 |
| 1 Extract Code | | N IBZ S IBXDATA=\$P(\$\$QUAL^IBCF22(IBXIE N,15,.IBXSAVE,.IBZ),U,2) |
| 3 Description | | Qualifier for box 15 |

IB DATA ELEMENT : N-CURR ILLNESS DATE (HCFA 1500)

| Field Name(s) | Current Value | New Value |
|----------------------|---------------|---|
| .01 Name | | N-CURRENT DATE (HCFA 1500) |
| .02 Security Level | | N |
| .03 Type of Element | | E |
| .04 Element Category | | I |
| .05 Base File | | 399 |
| 1 Extract Code | | N IBZ S IBXDATA=\$P(\$\$QUAL^IBCF22(IBXIE N,14,.IBXSAVE,.IBZ),U,1) I \$D(IBZ) M IBXSAVE("DATE")=IBZ |
| 3 Description | | This is either LMP or Onset of Illness |

| Field Name(s) | Current Value | New Value |
|---------------|---------------|-----------|
| | | date. |

IB DATA ELEMENT : N-OTHER DATE (HCFA 1500)

| Field Name(s) | Current Value | New Value |
|----------------------|---------------|--|
| .01 Name | | N-OTHER DATE (HCFA 1500) |
| .02 Security Level | | N |
| .03 Type of Element | | E |
| .04 Element Category | | I |
| .05 Base File | | 399 |
| 1 Extract Code | | N IBZ S IBXDATA=\$P(\$\$QUAL^IBCF22(IBXIE N,15,.IBXSAVE,.IBZ),U,1) I \$D(IBZ) M IBXSAVE("DATE")=IBZ |
| 3 Description | | This is accident date |

IB DATA ELEMENT : N-ADD CLAIM INFO 1 (HCFA 1500)

| Field Name(s) | Current Value | New Value |
|----------------------|---------------|--------------------------------|
| .01 Name | | N-ADD CLAIM INFO 1 (HCFA 1500) |
| .02 Security Level | | N |
| .03 Type of Element | | E |
| .04 Element Category | | I |
| .05 Base File | | 399 |
| 1 Extract Code | | GET MORE INFO |
| 3 Description | | This is new NUCC information |

IB DATA ELEMENT : N-ADD CLAIM INFO 2 (HCFA 1500)

| Field Name(s) | Current Value | New Value |
|----------------------|---------------|--------------------------------|
| .01 Name | | N-ADD CLAIM INFO 2 (HCFA 1500) |
| .02 Security Level | | N |
| .03 Type of Element | | E |
| .04 Element Category | | I |
| .05 Base File | | 399 |
| 1 Extract Code | | GET MORE INFO |

| Field Name(s) | Current Value | New Value |
|---------------|---------------|------------------------------|
| 3 Description | | This is new NUCC information |

IB DATA ELEMENT : N-DIAGNOSIS IND

| Field Name(s) | Current Value | New Value |
|----------------------|---------------|---|
| .01 Name | | |
| .02 Security Level | | N |
| .03 Type of Element | | E |
| .04 Element Category | | I |
| .05 Base File | | 399 |
| 1 Extract Code | | Check if ICD 9 or ICD10 codes are on the claim. |
| 3 Description | | Determine if diagnosis codes are ICD9 or ICD10. |

IB DATA ELEMENT : N-DIAGNOSIS CODE 5 (HCFA 1500)

| Field Name(s) | Current Value | New Value |
|----------------------|---------------|--|
| .01 Name | | N-DIAGNOSIS CODE 5 (HCFA 1500) |
| .02 Security Level | | N |
| .03 Type of Element | | E |
| .04 Element Category | | I |
| .05 Base File | | 399 |
| 1 Extract Code | | D:\$D(IBXSAVE("DX")) F^IBCEF("N-DIAGNOSES") S IBXDATA=\$G(IBXSAVE("DX",5)) |
| 3 Description | | Data element relies on previously extracted data in array IBXSAVE("DX") from the HCFA DIAGNOSES. If IBXSAVE("DX") does not exist, extract the field that sets it. The second in the sequence of diagnoses for bill IBXIEN for the HCFA 1500. This is the internal diagnosis code (ICD9). |

IB DATA ELEMENT : N-DIAGNOSIS CODE 6 (HCFA 1500)

| Field Name(s) | Current Value | New Value |
|---------------|---------------|-----------|
|---------------|---------------|-----------|

| Field Name(s) | Current Value | New Value |
|----------------------|---------------|--|
| .01 Name | | N-DIAGNOSIS CODE 6 (HCFA 1500) |
| .02 Security Level | | N |
| .03 Type of Element | | E |
| .04 Element Category | | I |
| .05 Base File | | 399 |
| 1 Extract Code | | D:'\$D(IBXSAVE("DX")) F^IBCEF("N-DIAGNOSES") S IBXDATA=\$G(IBXSAVE("DX",6)) |
| 3 Description | | Data element relies on previously extracted data in array IBXSAVE("DX") from the HCFA DIAGNOSES. If IBXSAVE("DX") does not exist, extract the field that sets it. The second in the sequence of diagnoses for bill IBXIEN for the HCFA 1500. This is the internal diagnosis code (ICD9). |

IB DATA ELEMENT : N-DIAGNOSIS CODE 7 (HCFA 1500)

| Field Name(s) | Current Value | New Value |
|----------------------|---------------|--|
| .01 Name | | N-DIAGNOSIS CODE 7 (HCFA 1500) |
| .02 Security Level | | N |
| .03 Type of Element | | E |
| .04 Element Category | | I |
| .05 Base File | | 399 |
| 1 Extract Code | | D:'\$D(IBXSAVE("DX")) F^IBCEF("N-DIAGNOSES") S IBXDATA=\$G(IBXSAVE("DX",7)) |
| 3 Description | | Data element relies on previously extracted data in array IBXSAVE("DX") from the HCFA DIAGNOSES. If IBXSAVE("DX") does not exist, extract the field that sets it. The second in the sequence of diagnoses for bill IBXIEN for the HCFA 1500. This is the internal diagnosis code (ICD9). |

IB DATA ELEMENT : N-DIAGNOSIS CODE 8 (HCFA 1500)

| Field Name(s) | Current Value | New Value |
|----------------------|---------------|--|
| .01 Name | | N-DIAGNOSIS CODE 8 (HCFA 1500) |
| .02 Security Level | | N |
| .03 Type of Element | | E |
| .04 Element Category | | I |
| .05 Base File | | 399 |
| 1 Extract Code | | D:'\$D(IBXSAVE("DX")) F^IBCEF("N-DIAGNOSES") S IBXDATA=\$G(IBXSAVE("DX",8)) |
| 3 Description | | Data element relies on previously extracted data in array IBXSAVE("DX") from the HCFA DIAGNOSES. If IBXSAVE("DX") does not exist, extract the field that sets it. The second in the sequence of diagnoses for bill IBXIEN for the HCFA 1500. This is the internal diagnosis code (ICD9). |

IB DATA ELEMENT : N-DIAGNOSIS CODE 9 (HCFA 1500)

| Field Name(s) | Current Value | New Value |
|----------------------|---------------|--|
| .01 Name | | N-DIAGNOSIS CODE 9 (HCFA 1500) |
| .02 Security Level | | N |
| .03 Type of Element | | E |
| .04 Element Category | | I |
| .05 Base File | | 399 |
| 1 Extract Code | | D:'\$D(IBXSAVE("DX")) F^IBCEF("N-DIAGNOSES") S IBXDATA=\$G(IBXSAVE("DX",9)) |
| 3 Description | | Data element relies on previously extracted data in array IBXSAVE("DX") from the HCFA DIAGNOSES. If IBXSAVE("DX") does not exist, extract the field that sets it. The second in the sequence of diagnoses for bill IBXIEN for the HCFA 1500. This is the internal diagnosis code (ICD9). |

IB DATA ELEMENT : N-DIAGNOSIS CODE 10 (HCFA 1500)

| Field Name(s) | Current Value | New Value |
|----------------------|---------------|--|
| .01 Name | | N-DIAGNOSIS CODE 10(HCFA 1500) |
| .02 Security Level | | N |
| .03 Type of Element | | E |
| .04 Element Category | | I |
| .05 Base File | | 399 |
| 1 Extract Code | | D:'\$D(IBXSAVE("DX")) F^IBCEF("N-DIAGNOSES") S IBXDATA=\$G(IBXSAVE("DX",10)) |
| 3 Description | | Data element relies on previously extracted data in array IBXSAVE("DX") from the HCFA DIAGNOSES. If IBXSAVE("DX") does not exist, extract the field that sets it. The second in the sequence of diagnoses for bill IBXIEN for the HCFA 1500. This is the internal diagnosis code (ICD9). |

IB DATA ELEMENT : N-DIAGNOSIS CODE 11 (HCFA 1500)

| Field Name(s) | Current Value | New Value |
|----------------------|---------------|--|
| .01 Name | | N-DIAGNOSIS CODE 11(HCFA 1500) |
| .02 Security Level | | N |
| .03 Type of Element | | E |
| .04 Element Category | | I |
| .05 Base File | | 399 |
| 1 Extract Code | | D:'\$D(IBXSAVE("DX")) F^IBCEF("N-DIAGNOSES") S IBXDATA=\$G(IBXSAVE("DX",11)) |
| 3 Description | | Data element relies on previously extracted data in array IBXSAVE("DX") from the HCFA DIAGNOSES. If IBXSAVE("DX") does not exist, extract the field that sets it. The second in the sequence of diagnoses for bill IBXIEN for the HCFA 1500. This is the internal diagnosis code (ICD9). |

IB DATA ELEMENT : N-DIAGNOSIS CODE 12 (HCFA 1500)

| Field Name(s) | Current Value | New Value |
|----------------------|---------------|--|
| .01 Name | | N-DIAGNOSIS CODE 12(HCFA 1500) |
| .02 Security Level | | N |
| .03 Type of Element | | E |
| .04 Element Category | | I |
| .05 Base File | | 399 |
| 1 Extract Code | | D:\$D(IBXSAVE("DX")) F^IBCEF("N-DIAGNOSES") S IBXDATA=\$G(IBXSAVE("DX",12)) |
| 3 Description | | Data element relies on previously extracted data in array IBXSAVE("DX") from the HCFA DIAGNOSES. If IBXSAVE("DX") does not exist, extract the field that sets it. The second in the sequence of diagnoses for bill IBXIEN for the HCFA 1500. This is the internal diagnosis code (ICD9). |

IB FORM SKELETON DEFINITION: CLAIM QUAL (BX 11b)

| Field Name(s) | Current Value | New Value |
|------------------------------|---------------|--|
| .01 Bill Form | | CMS 1500 |
| .02 Security Level | | N |
| .03 Associate Form | | Pointer to IB Form Skeleton (364.6) CLAIM QUAL (BX 11b) |
| .04 Page or Sequence | | 1 |
| .05 First Line Number | | 23 |
| .08 Starting Column or Piece | | 50 |
| .09 Length | | 2 |
| .1 Short Description | | CLAIM QUAL (BX 11b) |

IB FORM SKELETON DEFINITION: INSURED EMPLOYR NAME (BX-11B)

| Field Name(s) | Current Value | New Value |
|---------------|---------------|-----------|
| .01 Bill Form | | CMS 1500 |

| Field Name(s) | Current Value | New Value |
|------------------------------|---|---|
| .02 Security Level | | N |
| .03 Associate Form | Pointer to IB Form Skeleton (364.6) INSURED EMPLOYR NAME (BX-11B) | Pointer to IB Form Skeleton (364.6) INSURED EMPLOYR NAME (BX-11B) |
| .04 Page or Sequence | 1 | 1 |
| .05 First Line Number | 23 | 23 |
| .08 Starting Column or Piece | 50 | 54 |
| .09 Length | 29 | 20 |
| .1 Short Description | INSURED EMPLOYR NAME (BX-11B) | CLAIM ID (BX 11b) |

IB FORM SKELETON DEFINITION: REFERRING PROV QUAL (BX 17)

| Field Name(s) | Current Value | New Value |
|------------------------------|---------------|---|
| .01 Bill Form | | CMS 1500 |
| .02 Security Level | | N |
| .03 Associate Form | | Pointer to IB Form Skeleton (364.6) REFERRING PROV QUAL (BX 17) |
| .04 Page or Sequence | | 1 |
| .05 First Line Number | | 35 |
| .08 Starting Column or Piece | | 2 |
| .09 Length | | 2 |
| .1 Short Description | | REFERRING PROV QUAL (BX 17) |

IB FORM SKELETON DEFINITION: CURRENT DATE QUAL (BX-14)

| Field Name(s) | Current Value | New Value |
|-----------------------|---------------|---|
| .01 Bill Form | | CMS 1500 |
| .02 Security Level | | N |
| .03 Associate Form | | Pointer to IB Form Skeleton (364.6) CURRENT DATE QUAL (BX 14) |
| .04 Page or Sequence | | 1 |
| .05 First Line Number | | 33 |

| Field Name(s) | Current Value | New Value |
|------------------------------|---------------|---------------------------|
| .08 Starting Column or Piece | | 16 |
| .09 Length | | 1 |
| .1 Short Description | | CURRENT DATE QUAL (BX 14) |

IB FORM SKELETON DEFINITION: OTHER DATE QUAL (BX 15)

| Field Name(s) | Current Value | New Value |
|------------------------------|---------------|--|
| .01 Bill Form | | CMS 1500 |
| .02 Security Level | | N |
| .03 Associate Form | | Pointer to IB Form Skeleton (364.6) OTHER DATE QUAL (BX 15) |
| .04 Page or Sequence | | 1 |
| .05 First Line Number | | 33 |
| .08 Starting Column or Piece | | 31 |
| .09 Length | | 3 |
| .1 Short Description | | OTHER DATE QUAL (BX 15) |

IB FORM SKELETON DEFINITION: REFERRING PROV NAME (BX 17)

| Field Name(s) | Current Value | New Value |
|------------------------------|---|---|
| .01 Bill Form | CMS 1500 | CMS 1500 |
| .02 Security Level | N | N |
| .03 Associate Form | Pointer to IB Form Skeleton (364.6) REFERRING PROV NAME (BX 17) | Pointer to IB Form Skeleton (364.6) REFERRING PROV NAME (BX 17) |
| .04 Page or Sequence | 1 | 1 |
| .05 First Line Number | 35 | 35 |
| .08 Starting Column or Piece | 1 | 4 |
| .09 Length | 26 | 26 |
| .1 Short Description | REFERRING PROV NAME (BX 17) | REFERRING PROV NAME (BX 17) |

IB FORM SKELETON DEFINITION: DIAGNOSIS CODE 2 (BX-21.2)

| Field Name(s) | Current Value | New Value |
|------------------------------|---------------|-----------|
| .05 First Line Number | 41 | 39 |
| .08 Starting Column or Piece | 3 | 14 |

IB FORM SKELETON DEFINITION: DIAGNOSIS CODE 3 (BX-21.3)

| Field Name(s) | Current Value | New Value |
|------------------------------|---------------|-----------|
| .05 First Line Number | 39 | 39 |
| .08 Starting Column or Piece | 30 | 27 |

IB FORM SKELETON DEFINITION: DIAGNOSIS CODE 4 (BX-21.4)

| Field Name(s) | Current Value | New Value |
|------------------------------|---------------|-----------|
| .05 First Line Number | 41 | 39 |
| .08 Starting Column or Piece | 30 | 40 |

IB FORM SKELETON DEFINITION: DIAGNOSIS IND

| Field Name(s) | Current Value | New Value |
|------------------------------|---------------|--|
| .01 Bill Form | | CMS 1500 |
| .02 Security Level | | N |
| .03 Associate Form | | Pointer to IB Form Skeleton (364.6) DIAGNOSIS IND |
| .04 Page or Sequence | | 1 |
| .05 First Line Number | | 38 |
| .08 Starting Column or Piece | | 40 |
| .09 Length | | 3 |
| .1 Short Description | | DIAGNOSIS IND |

IB FORM SKELETON DEFINITION: DIAGNOSIS CODE 5 (BX-21.5)

| Field Name(s) | Current Value | New Value |
|--------------------|---------------|-----------|
| .01 Bill Form | | CMS 1500 |
| .02 Security Level | | N |

| Field Name(s) | Current Value | New Value |
|------------------------------|---------------|---|
| .03 Associate Form | | Pointer to IB Form Skeleton (364.6) DIAGNOSIS CODE 5 (BX-21.5) |
| .04 Page or Sequence | | 1 |
| .05 First Line Number | | 40 |
| .08 Starting Column or Piece | | 3 |
| .09 Length | | 8 |
| .1 Short Description | | DIAGNOSIS CNT (BX-21.5) |

IB FORM SKELETON DEFINITION: DIAGNOSIS CODE 6 (BX-21.6)

| Field Name(s) | Current Value | New Value |
|------------------------------|---------------|---|
| .01 Bill Form | | CMS 1500 |
| .02 Security Level | | N |
| .03 Associate Form | | Pointer to IB Form Skeleton (364.6) DIAGNOSIS CODE 6 (BX-21.6) |
| .04 Page or Sequence | | 1 |
| .05 First Line Number | | 40 |
| .08 Starting Column or Piece | | 14 |
| .09 Length | | 8 |
| .1 Short Description | | DIAGNOSIS CODE 6 (BX-21.6) |

IB FORM SKELETON DEFINITION: DIAGNOSIS CODE 7 (BX-21.7)

| Field Name(s) | Current Value | New Value |
|------------------------------|---------------|---|
| .01 Bill Form | | CMS 1500 |
| .02 Security Level | | N |
| .03 Associate Form | | Pointer to IB Form Skeleton (364.6) DIAGNOSIS CODE 7 (BX-21.7) |
| .04 Page or Sequence | | 1 |
| .05 First Line Number | | 40 |
| .08 Starting Column or Piece | | 27 |
| .09 Length | | 8 |
| .1 Short Description | | DIAGNOSIS CODE 7 (BX-21.7) |

IB FORM SKELETON DEFINITION: DIAGNOSIS CODE 8 (BX-21.8)

| Field Name(s) | Current Value | New Value |
|------------------------------|---------------|---|
| .01 Bill Form | | CMS 1500 |
| .02 Security Level | | N |
| .03 Associate Form | | Pointer to IB Form Skeleton (364.6) DIAGNOSIS CODE 8 (BX-21.8) |
| .04 Page or Sequence | | 1 |
| .05 First Line Number | | 40 |
| .08 Starting Column or Piece | | 40 |
| .09 Length | | 8 |
| .1 Short Description | | DIAGNOSIS CODE 8 (BX-21.8) |

IB FORM SKELETON DEFINITION: DIAGNOSIS CODE 9 (BX-21.9)

| Field Name(s) | Current Value | New Value |
|------------------------------|---------------|---|
| .01 Bill Form | | CMS 1500 |
| .02 Security Level | | N |
| .03 Associate Form | | Pointer to IB Form Skeleton (364.6) DIAGNOSIS CODE 9 (BX-21.9) |
| .04 Page or Sequence | | 1 |
| .05 First Line Number | | 41 |
| .08 Starting Column or Piece | | 3 |
| .09 Length | | 8 |
| .1 Short Description | | DIAGNOSIS CODE 9 (BX-21.9) |

IB FORM SKELETON DEFINITION: DIAGNOSIS CODE 10 (BX-21.10)

| Field Name(s) | Current Value | New Value |
|----------------------|---------------|---|
| .01 Bill Form | | CMS 1500 |
| .02 Security Level | | N |
| .03 Associate Form | | Pointer to IB Form Skeleton (364.6) DIAGNOSIS CODE 10 (BX-21.10) |
| .04 Page or Sequence | | 1 |

| Field Name(s) | Current Value | New Value |
|------------------------------|---------------|------------------------------|
| .05 First Line Number | | 41 |
| .08 Starting Column or Piece | | 14 |
| .09 Length | | 8 |
| .1 Short Description | | DIAGNOSIS CODE 10 (BX-21.10) |

IB FORM SKELETON DEFINITION: DIAGNOSIS CODE 11 (BX-21.11)

| Field Name(s) | Current Value | New Value |
|------------------------------|---------------|---|
| .01 Bill Form | | CMS 1500 |
| .02 Security Level | | N |
| .03 Associate Form | | Pointer to IB Form Skeleton (364.6) DIAGNOSIS CODE 11 (BX-21.11) |
| .04 Page or Sequence | | 1 |
| .05 First Line Number | | 41 |
| .08 Starting Column or Piece | | 27 |
| .09 Length | | 8 |
| .1 Short Description | | DIAGNOSIS CODE 11 (BX-21.11) |

IB FORM SKELETON DEFINITION: DIAGNOSIS CODE 12 (BX-21.12)

| Field Name(s) | Current Value | New Value |
|------------------------------|---------------|---|
| .01 Bill Form | | CMS 1500 |
| .02 Security Level | | N |
| .03 Associate Form | | Pointer to IB Form Skeleton (364.6) DIAGNOSIS CODE 12 (BX-21.12) |
| .04 Page or Sequence | | 1 |
| .05 First Line Number | | 41 |
| .08 Starting Column or Piece | | 40 |
| .09 Length | | 8 |
| .1 Short Description | | DIAGNOSIS CODE 12 (BX-21.12) |

IB FORM FIELD DEFINITION : CLAIM QUAL (BX-11B)

| Field Name(s) | Current Value | New Value |
|--------------------------|-------------------------------------|---|
| .01 Form Field Reference | Pointer to IB Form Skeleton (364.6) | Pointer to IB Form Skeleton (364.6) CLAIM QUAL (BX-11B) |
| .02 Security Level | | N |
| .03 Data Element | Pointer to IB Data Element (364.5) | Pointer to IB Data Element (364.5) N-OTH CLAIM QUAL (HCFA 1500) |
| .07 Pad Character | | N |
| 1 Format Code | | |
| 2 Format Description | | |

IB FORM FIELD DEFINITION : INSURED EMPLOYR NAME (BX-11B)

| Field Name(s) | Current Value | New Value |
|--------------------------|---|---|
| .01 Form Field Reference | Pointer to IB Form Skeleton (364.6) INSURED EMPLOYR NAME (BX-11B) | Pointer to IB Form Skeleton (364.6) INSURED EMPLOYR NAME (BX-11B) |
| .02 Security Level | N | N |
| .03 Data Element | Pointer to IB Data Element (364.5) N-CURR INSURED EMPLOYER INFO | Pointer to IB Data Element (364.5) N-OTHER CLAIM ID (HCFA 1500) |
| .07 Pad Character | N | N |
| 1 Format Code | S IBXDATA=\$P(IBXDATA,U) | S IBXDATA=\$P(IBXDATA,U) |
| 2 Format Description | Employer name is the first '^' piece of field's value. | This is the Property/Casualty number. |

IB FORM FIELD DEFINITION : BOX 10D CMS-1500

| Field Name(s) | Current Value | New Value |
|--------------------------|--|--|
| .01 Form Field Reference | Pointer to IB Form Skeleton (364.6) BOX 10D CMS-1500 | Pointer to IB Form Skeleton (364.6) BOX 10D CMS-1500 |
| .02 Security Level | | N |
| .03 Data Element | Pointer to IB Data Element (364.5) N-GET FROM PREVIOUS EXTRACT | Pointer to IB Data Element (364.5) N-CLAIM CODES (HCFA 1500) |
| .07 Pad Character | N | N |
| 1 Format Code | | |

| Field Name(s) | Current Value | New Value |
|----------------------|--|-----------|
| 2 Format Description | CMS-1500, BOX 10d. RESERVED FOR LOCAL USE. Not currently used. | |

IB FORM FIELD DEFINITION : CURR DATE QUAL (BX-14.1)

| Field Name(s) | Current Value | New Value |
|--------------------------|---------------|--|
| .01 Form Field Reference | | Pointer to IB Form Skeleton (364.6) CURR DATE QUAL (BX-14.1) |
| .02 Security Level | | N |
| .03 Data Element | | Pointer to IB Data Element (364.5) N- CURRENT DATE QUAL (HCFA 1500) |
| .07 Pad Character | | N |
| 1 Format Code | | S IBXDATA=\$P(IBXDATA,U,2) |
| 2 Format Description | | Qualifier for the date |

IB FORM FIELD DEFINITION : OTHER DATE QUAL (BX-15.1)

| Field Name(s) | Current Value | New Value |
|--------------------------|---------------|--|
| .01 Form Field Reference | | Pointer to IB Form Skeleton (364.6) OTHER DATE QUAL (BX-15.1) |
| .02 Security Level | | N |
| .03 Data Element | | Pointer to IB Data Element (364.5) n- |
| .07 Pad Character | | N |
| 1 Format Code | | S IBXDATA=\$P(IBXDATA,U,2) |
| 2 Format Description | | Date Qualifier |

IB FORM FIELD DEFINITION : DATE OF CURR ILLNESS (BX-14)

| Field Name(s) | Current Value | New Value |
|--------------------------|--|---|
| .01 Form Field Reference | Pointer to IB Form Skeleton (364.6) DATE OF CURR ILLNESS (BX-14) | Pointer to IB Form Skeleton (364.6) DATE OF CURR ILLNESS (BX-14) |
| .02 Security Level | | N |

| Field Name(s) | Current Value | New Value |
|----------------------|---|---|
| .03 Data Element | Pointer to IB Data Element (364.5) N-EVENT DATE | Pointer to IB Data Element (364.5) N-CURRENT DATE (HCFA 1500) |
| .07 Pad Character | N | N |
| 1 Format Code | S IBXDATA=\$\$DATE^IBCF2(IBXDATA,1) | S IBXDATA=\$P(IBXDATA,U),IBXDATA=\$\$DATE^IBCF2(IBXDATA,1) |
| 2 Format Description | | Box 14 date |

IB FORM FIELD DEFINITION : DATE OF SIMLAR ILLNESS (BX-15)

| Field Name(s) | Current Value | New Value |
|--------------------------|--|--|
| .01 Form Field Reference | Pointer to IB Form Skeleton (364.6) DATE OF SIMLAR ILLNESS (BX-15) | Pointer to IB Form Skeleton (364.6) DATE OF SIMLAR ILLNESS (BX-15) |
| .02 Security Level | N | N |
| .03 Data Element | Pointer to IB Data Element (364.5) N-SIMILAR ILLNESS DATE | Pointer to IB Data Element (364.5) N-OTHER DATE (HCFA 1500) |
| .07 Pad Character | N | N |
| 1 Format Code | S IBXDATA=\$\$DATE^IBCF2(IBXDATA,1) | S IBXDATA=\$P(IBXDATA,U),IBXDATA=\$\$DATE^IBCF2(IBXDATA,1) |
| 2 Format Description | | Box 15 date |

IB FORM FIELD DEFINITION : PT MARITAL STAT=SINGL (BX-8/A)

| Field Name(s) | Current Value | New Value |
|--------------------|---|---|
| .03 Data Element | Pointer to IB Data Element (364.5) N-PATIENT MARITAL STATUS | Pointer to IB Data Element (364.5) N-GET FROM PREVIOUS EXTRACT |
| 1 Format Code | S IBXSAVE("MSTAT")=\$\$MARITAL^IBCEFG1(+IBXDATA),IBXDATA=\$\$(IBXSAVE("MSTAT")="I":"X",1:"") | S IBXDATA="" |
| 3 Format Code Desc | Save the value of marital status in IBXSAVE variable for later use. If the value is 'I' (NEVER MARRIED), this field's value is 'X'. | This field is no longer used. Save the value of marital status in IBXSAVE variable for later use. If the value is 'I' (NEVER MARRIED), this field's value is 'X'. |

IB FORM FIELD DEFINITION : PT MARITAL STAT=MARRD (BX-8/B)

| Field Name(s) | Current Value | New Value |
|--------------------|--|--|
| .03 Data Element | Pointer to IB Data Element (364.5) N-GET FROM PREVIOUS EXTRACT | Pointer to IB Data Element (364.5) N-GET FROM PREVIOUS EXTRACT |
| 1 Format Code | I \$G(IBXSAVE("MSTAT"))'=""', "M"[I BXSAVE("MSTAT") S IBXDATA="X" | S IBXDATA="" |
| 3 Format Code Desc | This field relies on the existence of variable IBXSAVE("MSTAT") previously extracted. If its value is 'M' (MARRIED), this field's value will be 'X'. | This field is no longer used. This field relies on the existence of variable IBXSAVE("MSTAT") previously extracted. If its value is 'M' (MARRIED), this field's value will be 'X'. |

IB FORM FIELD DEFINITION : PT MARITAL STAT=OTHER (BX-8/C)

| Field Name(s) | Current Value | New Value |
|--------------------|--|---|
| .03 Data Element | Pointer to IB Data Element (364.5) N-GET FROM PREVIOUS EXTRACT | Pointer to IB Data Element (364.5) N-GET FROM PREVIOUS EXTRACT |
| 1 Format Code | S IBXDATA=\$S(\$G(IBXSAVE("MSTAT"))=""', "M"[IBXSAVE("MSTAT"):"X", 1:"") K IBXSAVE("MSTAT") | S IBXDATA="" |
| 3 Format Code Desc | This field relies on the existence of variable IBXSAVE("MSTAT") previously extracted. If its value is not a 'M', or 'I', assume it must be other and this field's value will be 'X'. | This field is no longer used. This field relies on the existence of variable IBXSAVE("MSTAT") previously extracted. If its value is not a 'M', or 'I', assume it must be other and this field's value will be 'X'. |

IB FORM FIELD DEFINITION : PT EMPL STAT=EMPLOYD (BX-8/2A)

| Field Name(s) | Current Value | New Value |
|------------------|--|--|
| .03 Data Element | Pointer to IB Data Element (364.5) N-PATIENT EMPLOYMENT STATUS | Pointer to IB Data Element (364.5) N-GET FROM PREVIOUS EXTRACT |
| 1 Format Code | S IBXDATA=\$S(1,2,4,6,"["+"_"+IBX | S IBXDATA="" |

| Field Name(s) | Current Value | New Value |
|--------------------|---|--|
| | DATA_,""): "X",1:"") | |
| 3 Format Code Desc | If the value of this field is '1' (EMPLOYED FULL TIME), '2' (EMPLOYED PART TIME), '4' (SELF EMPLOYED), or '6' (FULL TIME MILITARY), assume employed and output this field as 'X'. | This field is no longer used. If the value of this field is '1' (EMPLOYED FULL TIME), '2' (EMPLOYED PART TIME), '4' (SELF EMPLOYED), or '6' (FULL TIME MILITARY), assume employed and output this field as 'X'. |

IB FORM FIELD DEFINITION : OTH INSURED DOB (BX-9B/1)

| Field Name(s) | Current Value | New Value |
|--------------------|--|--|
| .03 Data Element | Pointer to IB Data Element (364.5) N-CURR INSURED PT RELATION | Pointer to IB Data Element (364.5) N-GET FROM PREVIOUS EXTRACT |
| 1 Format Code | S IBXDATA="" I \$O(IBXDATA("")) N Z S Z=\$G(IBXDATA(1)) K IBXDATA S IBXSAVE("OIDEM")=Z,IBXDATA =\$\$DATE^IBCF2(\$P(Z,U),1) | S IBXDATA="" |
| 3 Format code Desc | Save the other insured's demographic data in IBXSAVE for later use. Format the birthdate as MM DD YYYY. | This field is no longer used. Save the other insured's demographic data in IBXSAVE for later use. Format the birthdate as MM DD YYYY. |

IB FORM FIELD DEFINITION : OTH INS SEX=MALE (BX-9B/2A)

| Field Name(s) | Current Value | New Value |
|--------------------|---|--|
| .03 Data Element | Pointer to IB Data Element (364.5) N-GET FROM PREVIOUS EXTRACT | Pointer to IB Data Element (364.5) N-GET FROM PREVIOUS EXTRACT |
| 1 Format Code | I \$P(\$G(IBXSAVE("OIDEM")),U,2) ="M" S IBXDATA="X" | S IBXDATA="" |
| 3 Format code Desc | This field relies on the existence of the IBXSAVE("OIDEM") variable previously extracted. The second '^' piece contains the sex and if this is an 'M', this field's value is 'X'. | This field is no longer used. This field relies on the existence of the IBXSAVE("OIDEM") variable previously extracted. The second '^' piece contains the sex and if this is an 'M', this field's value is 'X'. |

IB FORM FIELD DEFINITION : OTH INS SEX=FEMALE (BX-9B/2B)

| Field Name(s) | Current Value | New Value |
|--------------------|---|---|
| .03 Data Element | Pointer to IB Data Element (364.5) N-GET FROM PREVIOUS EXTRACT | Pointer to IB Data Element (364.5) N- GET FROM PREVIOUS EXTRACT |
| 1 Format Code | N Z S Z=\$P(\$G(IBXSAVE("OIDEM")),U,2) S:Z'="M"&(Z'="") IBXDATA="X" K IBXSAVE("OIDEM") | S IBXDATA="" |
| 3 Format code Desc | This field relies on the existence of the IBXSAVE("OIDEM") variable previously extracted. The second '^' piece contains the sex and if this is not a null and is not an 'M', this field's value is 'X'. | This field is no longer used. This field relies on the existence of the IBXSAVE("OIDEM") variable previously extracted. The second '^' piece contains the sex and if this is not a null and is not an 'M', this field's value is 'X'. |

IB FORM FIELD DEFINITION: TOT CHRG-PRIOR PYMNTS (BX-30)

| Field Name(s) | Current Value | New Value |
|--------------------|---|---|
| .03 Data Element | Pointer to IB Data Element (364.5) N-HCFA 1500 BALANCE DUE BOX | Pointer to IB Data Element (364.5) N- GET FROM PREVIOUS EXTRACT |
| 1 Format Code | S IBXSAVE("BDUE")=IBXDATA K IBXDATA I \$O(^TMP("IBXDATA",\$J,IBXREC,""), -1)'>1 S IBXDATA=\$\$DOL^IBCEF77(IBXSA VE("BDUE"),8) K IBXSAV E("BDUE"),IBXSAVE("PTOT"),IBXS AVE("TOT") | S IBXSAVE("BDUE")=IBXDATA K IBXDATA I \$O(^TMP("IBXDATA",\$J,IBXREC,""),- 1)'>1 S IBXDATA=\$\$DOL^IBCEF77(IBXSAV E("BDUE"),8) K IBXSAV E("BDUE"),IBXSAVE("PTOT"),IBXSA VE("TOT") S IBXDATA="" |
| 3 Format code Desc | | This field is no longer used. |

6.2.2.8. System Feature: Health Care Claim Transactions (837)

This section covers the design for the Functional Requirements listed in Section 2.6.8 of the RSD.

6.2.2.8.1. Functional Requirements: Health Care Claim Transactions (837)

| SRS REQ ID | REQ Title | Comments / Notes |
|---------------|---|------------------|
| 2.6.8.1 | 12 Diagnoses (DXs) - Professional Claim | |
| 2.6.8.2 | Service Line Charge Amount | |

| SRS REQ ID | REQ Title | Comments / Notes |
|------------|---|------------------|
| 2.6.8.3 | Service Line Non-Covered Charge Amount | |
| 2.6.8.4 | Transmit Workman's Compensation Claims – Institutional | |
| 2.6.8.5 | Transmit Workman's Compensation Claims – Professional | |
| 2.6.8.6 | Functional Requirement: Assignment Code – Institutional | |
| 2.6.8.7 | Functional Requirement: Assignment Code – Professional | |
| 2.6.8.8 | Functional Requirement: Diagnoses Pointers – Professional | |

6.2.2.8.2. Routines (Entry Points)

| | | |
|---------------------------------|--|--|
| Routine Name | IBCEF2 | |
| Enhancement Category | <input type="checkbox"/> New <input checked="" type="checkbox"/> Modify <input type="checkbox"/> Delete <input type="checkbox"/> No Change | |
| Requirement Traceability Matrix | 2.6.8.1 | |
| Related Options | FORMATTER AND EXTRACTOR SPECIFIC BILL FUNCTIONS FORMATTER/EXTRACT BILL FUNCTIONS | |
| Related Routines | Routines “Called By” | Routines “Called” |
| | IBCB11, ICB BB, ICB BB1, ICB BB8, IBCCCB, IBCE837A, IBCECSA1, IBCEF11, IBCEF2, IBCEF21, IBCEF22, IBCEF7, IBCEF71, IBCEF72, IBCEF76, IBCEF78, IBCEF80, IBCEFP, IBCEMCA3, IBCEMCL, IBCEOB2, IBCEPTC0, IBCERP2, IBCERP6, IBCEU1, IBCEXTR1, IBCF23, IBCNSP1, IBCRBC, IBCRBC1, IBCRBC11, IBCRBG, IBCSC3, IBCU41, IJTLA1, IBJTLB1, IBRFN4, IBY115NV, IBYPNTEG | \$\$COBN^IBCEF, \$\$FT^IBCEF, \$\$INPAT^IBCEF, \$\$FINDINS^IBCEF1, \$\$COID^IBCEF21, \$ESGHPNL^IBCEF21, \$\$ESGHPST^IBCEF21, DMDT^IBCEF21, DISDT^IBCEF21, HOS^IBCEF22, \$\$MRASEC^IBCEF4, \$\$TXMT^IBCEF4, SETGBL^IBCEFG, \$\$DOLLAR^IBCEFG1, \$\$MCRONBIL^IBEFUNC, \$\$MCRWNR^IBEFUNC, \$\$WNRBILL^IBEFUNC, INP^VADPT , \$\$UP^XLFSTR |
| Data Dictionary (DD) References | BILL/CLAIMS File (#399) RATE TYPE File (#399.3) IB BILL/CLAIMS DIAGNOSIS File (#362.3) PATIENT MOVEMENT File (#405) | |
| Related Protocols | | |
| Related Integration Control | N/A | |

| | |
|--|---|
| Routine Name | IBCEF2 |
| Registrations (ICRs) | |
| Data Passing | <input type="checkbox"/> Input <input type="checkbox"/> Output Reference <input type="checkbox"/> Both <input type="checkbox"/> Global Reference <input type="checkbox"/> Local |
| Input Attribute Name and Definition | |
| Output Attribute Name and Definition | |
| Current Logic | |
| <p>ID1(LN,DX,CT) ;Special entryptoint for diagnoses to 'save' the fact ; a dx code is an e-code. ; LN is last entry # output, returned as the entry # (IBXLINE) to assign to this entry ; DX = the actual Dx code array(RECORD ID). Pass by reference, DX returned null if ; dx was not output ; CT = the ct on the 'DC' entry. pass by reference, returned null if ; the end of the valid dx codes has been reached N IBINS,VAL,CNT,DXIEN,DXQ,EDX,I,POA S IBINS=(\$\$FT^IBCEF(IBXIEN)=3) S VAL="DC"_CT S VAL=\$E(VAL_" ",1,4) S EDX=(\$E(\$G(DX))="E") ; TRUE if e-code DX S I=\$S(EDX:3,1:2) S:EDX DXQ=\$S(+\$G(^TMP("DCX",\$J,2))>0:"BF",1:"BK") ; first non e-code DX is principal (qualifier "BK"), the rest have qualifier "BF" I IBINS D .I CT>28 S CT="" Q ; Max of 28 codes for institutional/UB .S DXIEN=\$P(DX(CT),U,2) Q:DXIEN="" .S POA=\$P(\$G(^IBA(362.3,DXIEN,0)),U,4) I POA="",\$\$INPAT^IBCEF(IBXIEN) S POA=1 ; POA indicator defaults to "1", if not present on inpatient claim .S:EDX DXQ="BN" ; e-code DX qualifier .Q I 'IBINS S:EDX DXQ="BF" S POA="" ; on CMS-1500 e-code DX qualifiers are "BF" and there's no POA I 'IBINS,CT>8 S ^TMP("IBXSAVE",\$J,"DX",IBXIEN)=\$G(^TMP("IBXSAVE",\$J,"DX",IBXIEN))+1,^TMP("IBXSAVE",\$J,"DX",IB XIEN,\$P(DX(+^TMP("IBXSAVE",\$J,"DX",IBXIEN)),U,2))=\$G(^TMP("IBXSAVE",\$J,"DX",IBXIEN)) S DX="" Q I CT="",DX="" D .; populate ^TMP("DCX") scratch global .S ^TMP("DCX",\$J,1)=CT,CNT=\$G(^TMP("DCX",\$J,I))+1,^TMP("DCX",\$J,I)=CNT .S (^TMP("DCX",\$J,I,CNT),^TMP("DCX",\$J,1,CT))=DX_U_DXQ_U_POA .S LN=LN+1 D ID(LN,VAL) S ^TMP("IBXSAVE",\$J,"DX",IBXIEN,\$P(DX(LN),U,2))=LN,^TMP("IBXSAVE",\$J,"DX",IBXIEN)=CT,CT=CT+1 .Q Q</p> | |

| Routine Name | IBCEF2 |
|---|--------|
| Modified Logic (Changes are in bold) | |
| ID1(LN,DX,CT) ;Special entryptoint for diagnoses to 'save' the fact ; a dx code is an e-code. ; LN is last entry # output, returned as the entry # (IBXLINE) to assign to this entry ; DX = the actual Dx code array(RECORD ID). Pass by reference, DX returned null if ; dx was not output ; CT = the ct on the 'DC' entry. pass by reference, returned null if ; the end of the valid dx codes has been reached N IBINS,VAL,CNT,DXIEN,DXQ,EDX,I,POA S IBINS=(\$\$FT^IBCEF(IBXIEN)=3) S VAL="DC"_CT S VAL=\$E(VAL_" ",1,4) S EDX=(\$E(\$G(DX))="E") ; TRUE if e-code DX S I=\$S(EDX:3,1:2) S:EDX DXQ=\$S(+\$G(^TMP("DCX",\$J,2))>0:"BF",1:"BK") ; first non e-code DX is principal (qualifier "BK"), the rest have qualifier "BF" I IBINS D .I CT>28 S CT="" Q ; Max of 28 codes for institutional/UB .S DXIEN=\$P(DX(CT),U,2) Q:DXIEN="" .S POA=\$P(\$G(^IBA(362.3,DXIEN,0)),U,4) I POA="",\$\$INPAT^IBCEF(IBXIEN) S POA=1 ; POA indicator defaults to "1", if not present on inpatient claim .S:EDX DXQ="BN" ; e-code DX qualifier .Q I 'IBINS S:EDX DXQ="BF" S POA="" ; on CMS-1500 e-code DX qualifiers are "BF" and there's no POA I 'IBINS,CT> 12 S ^TMP("IBXSAVE",\$J,"DX",IBXIEN)=\$G(^TMP("IBXSAVE",\$J,"DX",IBXIEN))+1,^TMP("IBXSAVE",\$J,"DX",IB XIEN,\$P(DX(+^TMP("IBXSAVE",\$J,"DX",IBXIEN)),U,2))=\$G(^TMP("IBXSAVE",\$J,"DX",IBXIEN)) S DX="" Q I CT="",DX="" D .; populate ^TMP("DCX") scratch global .S ^TMP("DCX",\$J,1)=CT,CNT=\$G(^TMP("DCX",\$J,I))+1,^TMP("DCX",\$J,I)=CNT .S (^TMP("DCX",\$J,I,CNT),^TMP("DCX",\$J,1,CT))=DX_U_DXQ_U_POA .S LN=LN+1 D ID(LN,VAL) S ^TMP("IBXSAVE",\$J,"DX",IBXIEN,\$P(DX(LN),U,2))=LN,^TMP("IBXSAVE",\$J,"DX",IBXIEN)=CT,CT=CT+1 .Q Q | |

6.2.2.8.3. Unique Record(s) for 837

RATE TYPE File (#399.3) Workers' Comp.

| Field Name(s) | Current Value | New Value |
|---------------------|---------------|-------------|
| ELECTRONIC TRANSMIT | NOT ALLOWED | ALLOWED (1) |

IB FORM SKELETON DEFINITION : SEQ 185 - Service Line Charge Amt

| Field Name(s) | Current Value | New Value |
|--------------------------|---------------------|---------------------|
| BILL FORM | IB 837 TRANSMISSION | IB 837 TRANSMISSION |
| STARTING COLUMN OR PIECE | 9 | 9 |
| LENGTH | 9 | 18 |

IB FORM SKELETON DEFINITION : SEQ 185 - Service Line Non-covered Charge Amt

| Field Name(s) | Current Value | New Value |
|--------------------------|---------------------|---------------------|
| BILL FORM | IB 837 TRANSMISSION | IB 837 TRANSMISSION |
| STARTING COLUMN OR PIECE | 12 | 12 |
| LENGTH | 9 | 18 |

IB FORM FIELD DEFINITION : SEQ 51 – Medicare Assignment Code

| Field Name(s) | Current Value | New Value |
|---------------------------------|---|---------------------|
| Requirement Traceability Matrix | 2.6.8.6, 2.6.8.7 | |
| BILL FORM | IB 837 TRANSMISSION | IB 837 TRANSMISSION |
| PIECE | 5 | 5 |
| FORMAT CODE | K IBXDATA S IBIFN=IBXIEN N IBZZ S IBZZ=\$\$MCRONBIL^IBEFUNC(I BIFN,"",1) S IBXDATA=\$\$(IBZZ:"A",IBZZ=0:" C",1:"") | S IBXDATA="A" |

IB FORM SKELETON DEFINITION : SEQ 180 – Diagnosis Code Pointer(1)

| Field Name(s) | Current Value | New Value |
|---------------------------------|---------------------|---------------------|
| Requirement Traceability Matrix | 2.6.8.8 | |
| BILL FORM | IB 837 TRANSMISSION | IB 837 TRANSMISSION |
| PIECE | 11 | 11 |
| LENGTH | 1 | 2 |

IB FORM SKELETON DEFINITION : SEQ 180 – Diagnosis Code Pointer(2)

| Field Name(s) | Current Value | New Value |
|---------------------------------|---------------|-----------|
| Requirement Traceability Matrix | 2.6.8.8 | |

| Field Name(s) | Current Value | New Value |
|---------------|---------------------|---------------------|
| BILL FORM | IB 837 TRANSMISSION | IB 837 TRANSMISSION |
| PIECE | 12 | 12 |
| LENGTH | 1 | 2 |

IB FORM SKELETON DEFINITION : SEQ 180 – Diagnosis Code Pointer(3)

| Field Name(s) | Current Value | New Value |
|---------------------------------|---------------------|---------------------|
| Requirement Traceability Matrix | 2.6.8.8 | |
| BILL FORM | IB 837 TRANSMISSION | IB 837 TRANSMISSION |
| PIECE | 13 | 13 |
| LENGTH | 1 | 2 |

IB FORM SKELETON DEFINITION : SEQ 180 – Diagnosis Code Pointer(1)

| Field Name(s) | Current Value | New Value |
|---------------------------------|---------------------|---------------------|
| Requirement Traceability Matrix | 2.6.8.8 | |
| BILL FORM | IB 837 TRANSMISSION | IB 837 TRANSMISSION |
| PIECE | 14 | 14 |
| LENGTH | 1 | 2 |

6.2.2.9. System Feature: Miscellaneous Existing Requirements

This section covers the design for the Functional Requirements listed in Section 2.6.9 of the RSD.

6.2.2.9.1. Functional Requirements: Miscellaneous Existing Requirements

| SRS REQ ID | REQ Title | Comments / Notes |
|------------|---|------------------|
| 2.6.9.1 | Correct – FEAT765 Functional Requirement: Transmit Revenue/Procedure Codes with Zero Charge Amounts | |
| 2.6.9.2 | Delete – FEAT602 Functional Requirement: Transmit Service Facility Contact Data | |

6.2.2.9.2. Routines (Entry Points)

No routines to modify.

6.2.2.9.3. Unique Record ID

IB FORM FIELD DEFINITION : SEQ 180 - Line Item Charge Amt

| Field Name(s) | Current Value | New Value |
|---------------------------------|---|--|
| Requirement Traceability Matrix | 2.6.9.1 | |
| BILL FORM | IB 837 TRANSMISSION | IB 837 TRANSMISSION |
| STARTING COLUMN OR PIECE | 5 | 5 |
| FORMAT CODE | K IBXDATA S IBXNOREQ=\$\$NFT^IBCEF1(2,IB XIEN) N Z S Z=0 F S Z=\$O(IBXSAVE("OUTPT",Z)) Q:'Z S:\$P(IBXSAVE("OUTPT",Z),U,8) '="" IBXDA TA(Z)=\$\$DOLLAR^IBCEFG1(\$P(IBXSAVE("OUTPT",Z),U,8)*\$P(IB XSAVE("OUTPT",Z),U,9)) | K IBXDATA S IBXNOREQ=\$\$NFT^IBCEF1(2, IBXIEN) N Z S Z=0 F S Z=\$O(IBXSAVE("OUTPT",Z)) Q:'Z S:\$P(IBXSAVE("OUTPT",Z),U, 8)'="" IBXDA TA(Z)=\$\$DOLLAR^IBCEFG1(\$ P(IBXSAVE("OUTPT",Z),U,8)*\$ P(IBXSAVE("OUTPT",Z),U,9)) S:\$P(IBXSAVE("OUTPT",Z),U, 8)'="" IBXDATA(Z)="0.00" |

IB FORM FIELD DEFINITION : SEQ 185 - Service Line Charge Amt

| Field Name(s) | Current Value | New Value |
|---------------------------------|---|---|
| Requirement Traceability Matrix | 2.6.9.1 | |
| BILL FORM | IB 837 TRANSMISSION | IB 837 TRANSMISSION |
| PIECE | 9 | 9 |
| FORMAT CODE | K IBXDATA N Z S Z=0 F S Z=\$O(IBXSAVE("INPT",Z)) Q:'Z S:\$P(IBXSAVE("INPT",Z),U,5)'="" IBXDATA(Z)=\$\$DOLLAR^IBCEF G1(\$P(IBXSAVE("I NPT",Z),U,5)) | K IBXDATA N Z S Z=0 F S Z=\$O(IBXSAVE("INPT",Z)) Q:'Z S:\$P(IBXSAVE("INPT",Z),U,5) '="" IBXDATA(Z)=\$\$DOLLAR^IBCE FG1(\$P(IBXSAVE("I NPT",Z),U,5)) S:\$P(IBXSAVE("INPT",Z),U,5)= "" IBXDATA(Z)="0.00" |

IB FORM FIELD DEFINITION : SEQ 57 - Facility contact name qualifier

| Field Name(s) | Current Value | New Value |
|---------------------------------|---------------------|---------------------|
| Requirement Traceability Matrix | 2.6.9.2 | |
| BILL FORM | IB 837 TRANSMISSION | IB 837 TRANSMISSION |

| Field Name(s) | Current Value | New Value |
|---------------|---|--------------|
| PIECE | 17 | 17 |
| FORMAT CODE | S IBXDATA="IC" I \$P(\$G(IBXSAVE("LAB/FAC",IBXI EN,"C",1,"CONTACT")),U,1)=" S IBXDATA="" | S IBXDATA="" |

IB FORM FIELD DEFINITION : SEQ 57 - Facility Contact name

| Field Name(s) | Current Value | New Value |
|---------------------------------|---|---------------------|
| Requirement Traceability Matrix | 2.6.9.2 | |
| BILL FORM | IB 837 TRANSMISSION | IB 837 TRANSMISSION |
| PIECE | 18 | 18 |
| FORMAT CODE | S IBXDATA=\$P(\$G(IBXSAVE("LAB /FAC",IBXIEN,"C",1,"CONTACT") ,U,1) | S IBXDATA="" |

IB FORM FIELD DEFINITION : SEQ 57 - Facility phone number qualifier

| Field Name(s) | Current Value | New Value |
|---------------------------------|---|---------------------|
| Requirement Traceability Matrix | 2.6.9.2 | |
| BILL FORM | IB 837 TRANSMISSION | IB 837 TRANSMISSION |
| PIECE | 19 | 19 |
| FORMAT CODE | S IBXDATA="TE" I \$P(\$G(IBXSAVE("LAB/FAC",IBXI EN,"C",1,"CONTACT")),U,2)=" S IBXDATA="" | S IBXDATA="" |

IB FORM FIELD DEFINITION : SEQ 57 - Facility phone number

| Field Name(s) | Current Value | New Value |
|---------------------------------|---|---------------------|
| Requirement Traceability Matrix | 2.6.9.2 | |
| BILL FORM | IB 837 TRANSMISSION | IB 837 TRANSMISSION |
| PIECE | 20 | 20 |
| FORMAT CODE | S IBXDATA=\$\$NOPUNCT^IBCEF(\$P(\$G(IBXSAVE("LAB/FAC",IBXI EN,"C",1,"CONTACT")),U,2),1) | S IBXDATA="" |

IB FORM FIELD DEFINITION : SEQ 57 - Facility phone extension qualifier

| Field Name(s) | Current Value | New Value |
|---------------------------------|---|---------------------|
| Requirement Traceability Matrix | 2.6.9.2 | |
| BILL FORM | IB 837 TRANSMISSION | IB 837 TRANSMISSION |
| PIECE | 21 | 21 |
| FORMAT CODE | S IBXDATA="EX" I \$P(\$G(IBXSAVE("LAB/FAC",IBXI EN,"C",1,"CONTACT")),U,3)=" S IBXDATA="" | S IBXDATA="" |

IB FORM FIELD DEFINITION : SEQ 57 - Facility phone extension

| Field Name(s) | Current Value | New Value |
|---------------------------------|---|---------------------|
| Requirement Traceability Matrix | 2.6.9.2 | |
| BILL FORM | IB 837 TRANSMISSION | IB 837 TRANSMISSION |
| PIECE | 22 | 22 |
| FORMAT CODE | S IBXDATA=\$P(\$G(IBXSAVE("LAB /FAC",IBXIEN,"C",1,"CONTACT") ,U,3) | S IBXDATA="" |

6.2.2.10. Database Repository

This section is not applicable to this SDD.

6.2.2.11. System Features

This section is not applicable to this SDD.

6.2.2.12. Design Element Tables

This section is not applicable to this SDD.

7. External Interface Design

This section is not applicable to this SDD.

7.1. Interface Architecture

This section is not applicable to this SDD.

7.2. Interface Detailed Design

This section is not applicable to this SDD.

8. Human-Machine Interface

This section is not applicable to this SDD.

8.1. Interface Design Rules

This section is not applicable to this SDD.

8.2. Inputs

This section is not applicable to this SDD.

8.3. Outputs

This section is not applicable to this SDD.

8.4. Navigation Hierarchy

This section is not applicable to this SDD.

8.4.1. Screen [x.1]

This section is not applicable to this SDD.

8.4.2. Screen [x.2]

This section is not applicable to this SDD.

8.4.3. Screen [x.3]

This section is not applicable to this SDD.

9. System Integrity Controls

This section is not applicable to this SDD.

10. Approval Signatures

The signature below is an acknowledgement that the signatory understands the purpose and content of this document.



Re DELIVERABLE
Provider Operating Ri

Signed: _____ 3/6/2014_____

Integrated Project Team Chair

Date



RE DELIVERABLE
Provider Operating Ri

Signed: _____ 2/20/2014_____

Business Sponsor

Date



Re DELIVERABLE
Provider Operating Ri

Signed: _____ 3/6/2014_____

IT Program Manager



Re DELIVERABLE
Provider Operating Ri

Signed: _____ 3/6/2014_____

Project Manager



RE DELIVERABLE
Provider Operating R

Signed: _____ 3/7/2014_____

Enterprise Architecture

Date



RE SDD eBilling
Claims Compliance eC

Signed: _____ SDE_____ 3/10/2014_____

Service Delivery and Engineering

Date

A. Additional Information

A.1. RTM

[Requirements Traceability Matrix \(RTM\)](#)

A.2. Packaging and Installation

The following documents will be delivered as part of this project:

- [Current Workflow Analysis Document](#)
- [Requirements Specification Document \(RSD\)](#)
- [Requirements Traceability Matrix \(RTM\)](#)
- Software Design Document (SDD)
- [Interface Control Document \(ICD\)](#)
- [Entity Relationship Diagrams \(ERD\)](#)
- Release Notes/Installation Guide
- Technical/Security Manual
- EDI User Guide

A.3. Design Metrics

This section is not applicable to this SDD.

A.4. Acronym List and Glossary

This section of the SDD contains tables for the Acronyms and Definitions that are used with this document.

A-4.1 Acronyms

| Term | Definition |
|------|---|
| AITC | Austin Information Technology Center (formerly located in Austin, TX) |
| AR | Accounts Receivable software version 4.5 |
| COB | Coordination of Benefits |
| CBO | Chief Business Office |
| CSA | Claim Status Awaiting Resolution |
| CMS | Center for Medicare and Medicaid Services |

| | |
|-----------|---|
| DMI | Data Management Interface |
| EDI | Electronic Data Interchange |
| EIN | Employer's Identification Number |
| EOB | Explanation of Benefits |
| FSC | Financial Services Center – Austin, Texas |
| HIPAA | Health Insurance Portability and Accountability Act of 1996 |
| ICN | Individual Control Number |
| IB | Integrated Billing software version 2.0 |
| MRA | Medicare-equivalent Remittance Advice |
| MRW | MRA Management Work List |
| M (MUMPS) | Massachusetts General Hospital Utility Multi-Programming System |
| Non-MRA | Translates to non-Medicare |
| PS | Product Services |
| RX | Prescription (Outpatient Medication) |
| TPJI | Third Party Joint Inquiry |
| VAMC | Veterans Administration Medical Center |
| VistA | Veterans Health Information Systems and Technology Architecture |
| VPE | View/Print EDI Bill Extract Data |
| WNR | Will Not Reimburse |

A-4.2 Definitions

| Term | Definition |
|----------|--|
| 837 | Transaction set for Health Care Claim, used to send a claim to a trading partner |
| 835 | Transaction set for Health Care Claim Payment Advice (or remittance advice). This is returned from the insurer to the billing facility. Generally this is referred to as an Explanation of Benefits (EOB or MRA) |
| CMS-1500 | Preprinted forms to which professional third-party claims can be printed |
| Emdeon | The clearinghouse which handles both VA claims printing and the transmission of claims to electronic payers |
| EOB | This is the return file (835) from non-Medicare payers that provides data pertaining to the claim adjudication and the amounts paid by the payer |

| Term | Definition |
|-------------------------|--|
| MRA Request claim | This is the initial claim request to Medicare that is submitted for the purpose of obtaining MRA notice only |
| MRA | This is the return file (835) from Medicare that provides data on allowable amounts. MRA reports are normally required for creation of secondary claims |
| MRA Secondary Claim | This secondary claim is a result of the primary claim being an MRA Request claim |
| Non-MRA Secondary Claim | This secondary claim is a result of the primary claim being to any insurer other than Medicare WNR |
| Payer | An insurance company, fiscal intermediary, government agency, other agency, or individual responsible for the payment of health care claims |
| Translator | A software package owned and residing at the Austin Services Center that allows reformatting data in internal VA formats to EDI formats and Vice Versa. This includes the ability to simultaneously handle multiple versions of EDI. The FSC translator also provides for non ASC X12 formats. |
| UB04 | Preprinted forms to which institutional third-party claims can be printed |
| User | The person or persons who operate or interact directly with VistA. |

A-4.3 References

| Name | Location | Date |
|---|------------|-----------|
| ASC X12N/005010X221 Health Care Claim Payment/Advice (835) – Technical Report Type 3 | [REDACTED] | May 2006 |
| ASC X12N/005010X222 Health Care Claim - Professional (837) – Technical Report Type 3 | [REDACTED] | May 2006 |
| ASC X12N/005010X223 Health Care Claim - Institutional (837) – Technical Report Type 3 | [REDACTED] | May 2006 |
| National Uniform Claim Committee – 1500 Claim Form Reference Manual Version 9.0 | [REDACTED] | July 2013 |