Department of Veterans Affairs

Copays on Hold – Medicare/No Insurance

System Design Document



IB\*2.0\*554

February 2016

Version 0.06

Revision History

| Date | Version | Description | Author | Reviewer | Issue Date | Formal Review |
| --- | --- | --- | --- | --- | --- | --- |
| 07/07/2015 | 0.01 | Sections 1-4 for AERB Review Request | CPAC PMOSS |  |  |  |
| 07/24/2015 | 0.02 | Initial Draft | xxxx | xxxx | 07/24/2015 | 08/11/2015 |
| 08/12/2015 | 0.03 | Updates per Formal Review | xxxx | xxxx | 08/13/2015 | 08/24/2015 |
| 08/31/2015 | 0.04 | Additional updates from Formal Review | xxxxx | xxxxx | 09/03/2015 |  |
| 01/04/2016 | 0.05 | Added code | xxxxx | xxxxxx | 01/08/2016 |  |
| 2/1/2014 | 0.06 | Updates per Formal Review | xxxxx | xxxx | 02/11/2016 |  |

| Activity | New Capability (1) | Feature Enhancement (2) |
| --- | --- | --- |
| **Field Deployment (A)** | Yes | Yes |
| **Cloud/Web Deployment (B)** | No | No |
| **Mobile Application (C)** | No | No |

Table of Contents

[1. Introduction 7](#_Toc442889513)

[1.1 Scope 7](#_Toc442889514)

[1.2 User Profiles 8](#_Toc442889515)

[2 Background 9](#_Toc442889516)

[2.1 Overview of the System 9](#_Toc442889517)

[2.2 Overview of the Business Process 9](#_Toc442889518)

[2.3 Overview of the Significant Requirements 11](#_Toc442889519)

[2.4 Overview of Functional Requirements 11](#_Toc442889520)

[2.4.1 Overview of Functional Workload/Performance Requirements 17](#_Toc442889521)

[2.4.2 Overview of Operational Requirements 17](#_Toc442889522)

[2.4.3 Overview of Technical Requirements 17](#_Toc442889523)

[2.4.4 Overview of Security or Privacy Requirements 17](#_Toc442889524)

[2.4.5 Overview of System Criticality and High Availability Requirements 17](#_Toc442889525)

[2.4.6 Single Sign-on Requirements 17](#_Toc442889526)

[2.4.7 Requirement for Use of Enterprise Portals 17](#_Toc442889527)

[2.4.8 Special Device Requirements 18](#_Toc442889528)

[3 Conceptual Design 18](#_Toc442889529)

[3.1 Conceptual Application Design 18](#_Toc442889530)

[3.1.1 Application Context 18](#_Toc442889531)

[3.1.2 High-Level Application Design 18](#_Toc442889532)

[3.1.3 Application Locations 18](#_Toc442889533)

[3.2 Conceptual Data Design 18](#_Toc442889534)

[3.2.1 Project Conceptual Data Model 18](#_Toc442889535)

[3.2.2 Database Information 20](#_Toc442889536)

[3.2.3 User Interface Data Mapping 20](#_Toc442889537)

[3.3 Conceptual Infrastructure Design 21](#_Toc442889538)

[3.3.1 System Criticality and High Availability 21](#_Toc442889539)

[3.3.2 Special Technology 21](#_Toc442889540)

[3.3.3 Technology Locations 21](#_Toc442889541)

[3.3.4 Conceptual Infrastructure Diagram 21](#_Toc442889542)

[3.3.4.1 Location of Environments and External Interfaces 21](#_Toc442889543)

[3.3.4.2 Conceptual Production String Diagram 21](#_Toc442889544)

[4 System Architecture 21](#_Toc442889545)

[4.1 Hardware Architecture 21](#_Toc442889546)

[4.2 Software Architecture 21](#_Toc442889547)

[4.3 Network Architecture 23](#_Toc442889548)

[4.4 SOA/ESS 23](#_Toc442889549)

[4.5 Enterprise Architecture 23](#_Toc442889550)

[5 Data Design 24](#_Toc442889551)

[5.1 Database Management System (DBMS) Files 24](#_Toc442889552)

[5.2 Non-DBMS Files 24](#_Toc442889553)

[5.3 Data View 24](#_Toc442889554)

[6 Detailed Design 24](#_Toc442889555)

[6.1 Hardware Detailed Design 24](#_Toc442889556)

[6.2 Software Detailed Design 25](#_Toc442889557)

[6.2.1 Conceptual Design 25](#_Toc442889558)

[6.2.1.1 Product Perspective 25](#_Toc442889559)

[6.2.1.1.1 User Interfaces 25](#_Toc442889560)

[6.2.1.1.2 Hardware Interfaces 25](#_Toc442889561)

[6.2.1.1.3 Software Interfaces 25](#_Toc442889562)

[6.2.1.1.4 Communications Interfaces 25](#_Toc442889563)

[6.2.1.1.5 Memory Constraints 25](#_Toc442889564)

[6.2.1.1.6 Special Operations 26](#_Toc442889565)

[6.2.1.2 Product Features 26](#_Toc442889566)

[6.2.1.3 User Characteristics 26](#_Toc442889567)

[6.2.1.4 Dependencies and Constraints 26](#_Toc442889568)

[6.2.2 Specific Requirements 26](#_Toc442889569)

[6.2.2.1 Database Repository 26](#_Toc442889570)

[6.2.2.2 System Features 26](#_Toc442889571)

[6.2.2.3 Design Element Tables 26](#_Toc442889572)

[6.2.2.3.1 Routines (Entry Points) 26](#_Toc442889573)

[6.2.2.3.2 Templates 73](#_Toc442889574)

[6.2.2.3.3 Data Entries Affected by the Design 75](#_Toc442889575)

[6.2.2.3.4 Unique Record(s) 75](#_Toc442889576)

[6.2.2.3.5 File or Global Size Changes 75](#_Toc442889577)

[6.2.2.3.6 Mail Groups 75](#_Toc442889578)

[6.2.2.3.7 Security Keys 75](#_Toc442889579)

[6.2.2.3.8 Options 75](#_Toc442889580)

[6.2.2.3.9 Protocols 76](#_Toc442889581)

[6.2.2.3.10 RPC 82](#_Toc442889582)

[6.2.2.3.11 Constants Defined in Interface 82](#_Toc442889583)

[6.2.2.3.12 Variables Defined in Interface 82](#_Toc442889584)

[6.2.2.3.13 Types Defined in Interface 82](#_Toc442889585)

[6.2.2.3.14 GUI 82](#_Toc442889586)

[6.2.2.3.15 GUI Classes 83](#_Toc442889587)

[6.2.2.3.16 Current Form 83](#_Toc442889588)

[6.2.2.3.17 Modified Form 83](#_Toc442889589)

[6.2.2.3.18 Components on Form 83](#_Toc442889590)

[6.2.2.3.19 Events 83](#_Toc442889591)

[6.2.2.3.20 Methods 83](#_Toc442889592)

[6.2.2.3.21 Special References 83](#_Toc442889593)

[6.2.2.3.22 Class Events 83](#_Toc442889594)

[6.2.2.3.23 Class Methods 83](#_Toc442889595)

[6.2.2.3.24 Class Properties 83](#_Toc442889596)

[6.2.2.3.25 Uses Clause 83](#_Toc442889597)

[6.2.2.3.26 Forms 83](#_Toc442889598)

[6.2.2.3.27 Functions 83](#_Toc442889599)

[6.2.2.3.28 Dialog 83](#_Toc442889600)

[6.2.2.3.29 Help Frame 83](#_Toc442889601)

[6.2.2.3.30 HL7 Application Parameter 84](#_Toc442889602)

[6.2.2.3.31 HL7 Logical Link 84](#_Toc442889603)

[6.2.2.3.32 COTS Interface 84](#_Toc442889604)

[6.3 Network Detailed Design 84](#_Toc442889605)

[6.4 Security and Privacy 84](#_Toc442889606)

[6.4.1 Security 84](#_Toc442889607)

[6.4.2 Privacy 84](#_Toc442889608)

[6.5 SOA/ESS Detailed Design 84](#_Toc442889609)

[7 External System Interface Design 84](#_Toc442889610)

[8 Human-Machine Interface 84](#_Toc442889611)

[Attachment A – Approval Signatures 85](#_Toc442889612)

[Attachment B – Acronyms 86](#_Toc442889613)

[Attachment C – Additional Information 88](#_Toc442889614)

[A.1 Identification of Technology and Standards 88](#_Toc442889615)

[A.2 Constraining Policies, Directives and Procedures 88](#_Toc442889616)

[A.3 RTM 88](#_Toc442889617)

[A.4 Packaging and Installation 88](#_Toc442889618)

[A.5 Design Metrics 88](#_Toc442889619)

[Attachment D – Change Log 89](#_Toc442889620)

List of Tables

[Table 1: Scope Inclusions 7](#_Toc442889306)

[Table 2: Primary and Secondary Users 9](#_Toc442889307)

[Table 3: Functional Requirements 12](#_Toc442889308)

[Table 4 Database Information 19](#_Toc442889309)

[Table 5: (Grouping): Routines 26](#_Toc442889310)

[Table 6: Templates 74](#_Toc442889311)

[Table 7: Options 75](#_Toc442889312)

[Table 8: Protocols 76](#_Toc442889313)

**List of Figures**

[Figure 1: Application Context 18](#_Toc442889314)

[Figure 2: Data Model 20](#_Toc442889315)

[Figure 3 Netwrok Architecture 23](#_Toc442889316)

[Figure 4 Enterprise Architecture 24](#_Toc442889317)

# Introduction

The Copays on Hold – Medicare/No Insurance project is one of five enhancements comprising the Electronic Data Interchange (EDI) Consolidated Patient Account Center (CPAC) Revenue Enhancements project. This project will modify the Veterans Health Information Systems and Technology Architecture (VistA) Integrated Billing (IB) software application to automatically release copay charges for patients based on certain conditions. This will result in more efficient use of Department of Veterans Affairs (VA) resources by discontinuing the manual process of releasing first party copayments. It will also ensure that business processes are in adherence with the Veterans Health Administration (VHA) Directive 2012-005 Application of Third-Party Reimbursement to Veteran Copayments ([http:/xxxxxxxxxxxxx/vhapublications/ViewPublication.asp?pub\_ID=2480](http://xxxxxxxxxxxxx/vhapublications/ViewPublication.asp?pub_ID=2480)) that copays are held only for billable insurance reimbursements.

## Scope

The Copays on Hold Medicare/No Insurance Business Requirements Document (BRD) is available via Technical Services Project Repository (TSPR) here: <http://xxxxxxxxxxxxxxxxxx/warboard/anotebk.asp?proj=1785&Type=Active>

This document addresses the software design that will satisfy the technical requirements analysis, design, development, testing, release support, and documentation for the Copays on Hold – Medicare/No Insurance project.

Table 1: Scope Inclusions

| Includes |
| --- |
| Release copay charges for a patient during Buffer File processing when both of the following conditions are met: all outstanding Buffer File entries assigned to the patient have been processed, and the patient has no billable insurance in effect for the date the service/fill date was provided. |
| Automatically release copay charges for a patient when insurance file coverage limitations for the date of service/fill date indicate that there is no coverage for that particular type of service. This applies to any insurance active for the date of service/fill date being evaluated. |
| Prohibit copays from being placed on hold if the patient has TRICARE and no other billable insurance in effect for the date of service/fill date being evaluated. |
| Prohibit copays from being placed on hold if the patient has Civilian Health and Medical Program of the VA (CHAMPVA) and no other billable insurance in effect for the date of service/fill date being evaluated. |
| Provide the capability to produce the VistA Held Charges report containing the following fields: Site/Division, Patient Name, Patient ID, Insurance, Coverage Limitations, Coverage Effective Dates, Event Type, Date of Service or Fill Date, RX Number (when applicable), Amount on Hold, Date Copay Billed, Third Party Claim(s) numbers, Date Third Party Claims Billed, Billed Charges for Third Party Claims, Days on Hold. |
| Provide the capability to generate the VistA Held Charges report for a Date of Service/Fill Date as defined by the user. |
| Provide the capability to generate the VistA Held Charges report for a range of Dates of Service/Fill Dates as defined by the user. |
| Provide the capability to generate the VistA Held Charges report for all Dates of Service/Fill Dates. |
| Provide the capability to generate the VistA Held Charges report for a Site (Division within Site) as defined by the user. |
| Provide the capability to generate the VistA Held Charges report for more than one Site (Division within Site) as defined by the user. |
| Provide the capability for the user to generate the VistA Held Charges report for all Sites (Division within Site). |
| Provide the capability to generate the VistA Held Charges report for a Patient as defined by the user. |
| Provide the capability to generate the VistA Held Charges report for more than one Patient as defined by the user. |
| Provide the capability for the user to generate the VistA Held Charges report for all Patients. |
| Provide the capability for the user to choose whether insurance information is included on the VistA Held Charges report. Insurance information includes Insurance, Coverage Limitations, and Coverage Effective Dates. |
| Provide the capability for the user to choose whether Previously Released Copay charges are included on the VistA Held Charges report. |
| Generate the VistA Held Charges report in List Manager format. |
| Provide a List Manager action to export the VistA Held Charges report in Excel format. |
| Provide a List Manager action to release held co-pay charges for a selected line item on the VistA Held Charges report. |
| Limit the ability to release held co-pay charges from the VistA Held Charges report List Manager screen by requiring the user to possess the IB AUTHORIZE security key. |
| Group data by patient on the VistA Held Charges report, with each copayment appearing as a separate line item within each patient. |
| Include only charges having an ‘on hold’ status on the VistA Held Charges report. |
| Generate the Patient ID on the VistA Held Charges report as the first letter of the last name/last four digits of the patient’s Social Security Number (SSN). |
| Include only the insurance information that was active at the time of the episode of care relating to the held copay charges on the VistA Held Charges Report. |

## User Profiles

The intended user base of the proposed modifications to the VistA IB system are Billers, First Party Accounts management, Health Resource Center (HRC) First Party Call Center, and Revenue Supervisors. These users are proficient with the existing VistA IB systems but will undergo a technical training curriculum to ensure they are aware of how to use the new system enhancements.

Additional specifics for this project are elaborated in Table 2:

Table 2: Primary and Secondary Users

| Name | Description | Responsibilities |
| --- | --- | --- |
| Primary Users | Facility Revenue Staff | Responsible for submitting claims for health services rendered for reimbursement. |
| Primary Users | First Party Accounts Management | Responsible for managing first party copay charges and matching associated third party claims for offset. |
| Secondary Users | HRC First Party Call Center | Responsible for responding to health benefits eligibility and billing inquiries from Veterans. |
| Secondary Users | Revenue Supervisors | Responsible for overseeing the first party copay process. |

# Background

The Copays on Hold project details the needed system functionality to comply with the legislative mandate for compliance with national standardized operating rules for unique, standard CPACs.

## Overview of the System

CPAC will improve efficiency and accountability in revenue operations. VA health care depends primarily on annual congressional appropriations. To expand the funding available for Veteran healthcare services, VA supplements the funds from congress by billing health insurance companies for non-service connected care provided by VA Medical Centers (VAMCs). The CPAC program works with VAMCs to generate this revenue from insurance companies. Veterans can use their insurance benefits to help offset copays that VA may charge for non-service connected care. Providing health insurance information also benefits all Veterans by generating more funds for VAMCs to provide healthcare services.

## Overview of the Business Process

The VistA IB software application automates the creation of third party billing forms allowing for the entry, editing, authorizing, printing, and canceling of bills. It also provides reports to identify billable episodes of care, patient and insurance inquiries, and statistical data. Part of this functionality involves processing first party bills for medications or services. First party bills are created when a patient incurs charges based on the patient’s priority group and/or income level for medical services rendered at a VA health care facility. The charges incurred are generally copayments for patients who are treated as an inpatient, outpatient, or in receipt of prescriptions. If the Veteran does not have third party billable insurance, copay charges are released to the patient; and he/she receives a monthly statement showing the copay amounts due. If the Veteran does have billable insurance, copay charges are automatically put on hold by the system (within the 350 IB Action File) for up to 90 days. This holding period allows time for VA to submit bills to the third party insurance carrier as well as time for the carrier to submit payment(s) back to VA. Revenue personnel actively review Veteran copayments on hold (“held charges”) to ensure third party payments are applied to offset the amount due by the patient as well as release charges where the services are not covered by their third party insurance. These core reviews will not be impacted by this project’s enhancements.

Veterans and Active Duty personnel who present for care under their TRICARE benefit are not subject to the Veteran copayments referenced in this document. Those beneficiaries are assessed a cost share based on third party payments collected and individual TRICARE policy coverage.

The Buffer File was created to provide a temporary holding file for newly identified health insurance policies until such time that the policy information could be verified and placed into the Patient Insurance (PI) File for billing. When the Buffer File was designed in the 1990s, all copays were automatically placed on hold when any entry was made in the Buffer File or if the patient had active insurance in the VistA PI file. Since the buffer was only used to collect billable insurance and/or Medicare for those Veterans with insurance, the hold copay feature was not a problem. Sites could manage the copays on hold for those limited number of entries in the Buffer File that may have erroneously been created. This became a problem when the Insurance Capture Buffer (ICB) was released to the field with instructions to actually create buffer entries for ‘No Insurance’ and ‘Patient Refused’. Collecting and verifying all Medicare information from the Buffer File, regardless of whether the patient has any billable insurance or not, compounded the problem. This has resulted in increased man hours to manually release copay charges from lengthy ‘Held Charges Reports’. If the report is not worked on a regular basis, the system will auto-release the charges in 90 days. VHA has mandated that copays are held only for billable insurance reimbursements. Additionally, Veterans can become confused due to the long wait from the time service was rendered to the time they receive a bill. This has increased the volume of calls that are being made by the Veteran to HRC and CPACs for explanations associated with the delay.

To remedy this situation, the VistA IB software application will be updated to automatically release copay charges for patients when all of the following criteria have been determined/met:

* All outstanding Buffer File entries currently assigned to the patient have been processed.
* Patient has no billable third party insurance for the copay date.
* If the patient only has TRICARE or CHAMPVA in the PI File and no other billable insurance (identified by the type of plan as well as plan coverage limitations).

The VistA Held Charges Report will also be enhanced to provide a better tool to examine, triage, and disposition held copay charges.

A high-level view of the current (as-is) and future (to-be) third-party billing precertification processes/workflows are available via TSPR here: <http://xxxxxxxxxxxxxxx/warboard/anotebk.asp?proj=1785&Type=Active>

Until documents are approved and posted to TSPR, they will be made available upon request from the project team.

## Overview of the Significant Requirements

The Copays on Hold – Medicare/No Insurance Requirements Specification Document (RSD), Requirements Traceability Matrix (RTM), and BRD are available via TSPR here:

<http://xxxxxxxx/warboard/anotebk.asp?proj=1785&Type=Active>

Until documents are approved and posted to TSPR, they will be made available upon request from the project team.

The significant functional requirements associated with this effort are to provide the ability for the VistA IB software application to automatically release copay charges for patients, based on the following conditions:

* Automatically release copay charges when all outstanding Buffer File entries assigned to the patient have been processed and the patient has no billable third-party insurance subject to the copay date.
* Do not place the copays on hold if the patient only has TRICARE or CHAMPVA coverage and no other billable insurance subject to the copay date.

Additionally, the VistA Held Charges Report will be reworked extensively and converted to a List Manager format. The List Manager format will allow the user to interact with the report to facilitate the examination of data. List Manager actions will be created that will provide the user with the capability to release the copay charges directly from the List Manager report screen, and also the capability to export the report to Excel format. New report filtering selections will be provided to further customize the report generation.

## Overview of Functional Requirements

Provide the ability for the VistA IB software application to automatically release copay charges for patients, based on the following conditions:

* Automatically release copay charges when all outstanding Buffer File entries assigned to the patient have been processed and the patient has no billable third-party insurance subject to the copay date.
* Do not place the copays on hold if the patient only has TRICARE or CHAMPVA coverage and no other billable insurance subject to the copay date.

Table 3: Functional Requirements

| Business Need (BN) | Owner Req. (OWNR)  Number | OWNR | RSD Requirement | Priority |
| --- | --- | --- | --- | --- |
| BN 2: Provide the ability for the VistA IB software application to automatically release copay charges for patients, when appropriate. | 2.1 | Provide the ability for the VistA IB software application to automatically release copay charges when processing the Buffer File and all of the following criteria have been determined/met:  All outstanding Buffer File entries currently assigned to the Veteran have been processed;  Veteran has no billable third party insurance for the copay date, as well as plan coverage limitations. | **COH-2.1-01**  The system shall automatically release copay charges for a Veteran during Buffer File processing when both of the following conditions are met: all outstanding Buffer File entries assigned to the Veteran have been processed, and the Veteran has no billable insurance in effect for the date the service/fill date was provided. | High |
| BN 2: Provide the ability for the VistA IB software application to automatically release copay charges for patients, when appropriate. | 2.2 | Provide the ability to not place copays on hold if the Veteran only has TRICARE or CHAMPVA insurance within the VistA insurance file and no other billable insurance (identified by the type of plan as well as plan coverage limitations). | **COH-2.2-01**  The system shall prohibit copays from being placed on hold if the Veteran has TRICARE insurance within the VistA insurance file and no other billable insurance in effect for the date of service/fill date being evaluated.  **COH-2.2-02**  The system shall prohibit copays from being placed on hold if the Veteran has CHAMPVA insurance within the VistA insurance file and no other billable insurance in effect for the date of service/fill date being evaluated. | High |
| BN 2: Provide the ability for the VistA IB software application to automatically release copay charges for patients, when appropriate. | 2.3 | Provide the ability to automatically release copayment charges when insurance file coverage limitations for any insurance active for that date of service/fill date indicate “NO” coverage for that particular type of service. | **COH-2.3-01**  The system shall automatically release copay charges for a Veteran when insurance file coverage limitations for the date of service/fill date indicate that there is no coverage for that particular type of service. This applies to any insurance active for the date of service/fill date being evaluated. | High |
| BN 3: Provide enhancements to the VistA Held Charges Report to address current CPAC processes and use of the report. | 3.1 | System shall include a List Manager User Interface with available VistA fields currently required for effective review of held copayment charges:   1. Site/Division 2. Patient Name 3. Patient ID 4. Insurance 5. Coverage limitations 6. Coverage effective dates 7. Event type 8. Date of service or fill date 9. RX number when applicable 10. Amount on hold 11. Date copay billed 12. Third party claim(s) numbers associated with patient and date of service 13. Current Status of associated third party claims identified in Field #12. 14. Date third party claims billed 15. Billed charges for third party claims 16. Days On Hold | **COH-3.1-01**  The system shall provide the capability to produce the VistA Held Charges report containing the following fields:   1. Site/Division 2. Patient Name 3. Patient ID 4. Insurance 5. Coverage limitations 6. Coverage effective dates 7. Event type 8. Date of service or fill date 9. RX number when applicable 10. Amount on hold 11. Date copay billed 12. Third party claim(s) numbers associated with patient and date of service 13. Current Status of associated third party claims 14. Date third party claims billed 15. Billed charges for third party claims 16. Days on Hold | High |
| BN 3: Provide enhancements to the VistA Held Charges Report to address current CPAC processes and use of the report. | 3.2 | System shall provide report parameters that allow customization of data:   1. Timeframe (based on date of service/fill date) 2. Site/Division 3. Patient name and last 4 SSN 4. Include insurance information (y/n) 5. Include previously released copay charges (y/n)   Excel ready print to file | **COH-3.2-01**  The system shall provide the capability to generate the VistA Held Charges report for a Date of Service/Fill Date as defined by the user.  **COH-3.2-02**  The system shall provide the capability to generate the VistA Held Charges report for a range of Dates of Service/Fill Dates as defined by the user.  **COH-3.2-03**  The system shall provide the capability to generate the VistA Held Charges report for all Dates of Service/Fill Dates.  **COH-3.2-04**  The system shall provide the capability to generate the VistA Held Charges report for a Site (Division within Site) as defined by the user.  **COH-3.2-05**  The system shall provide the capability to generate the VistA Held Charges report for more than one Site (Division within Site) as defined by the user.  **COH-3.2-06**  The system shall provide the capability for the user to generate the VistA Held Charges report for all Sites (including Division within Site).  **COH-3.2-07**  The system shall provide the capability to generate the VistA Held Charges report for a Veteran as defined by the user.  **COH-3.2-08**  The system shall provide the capability to generate the VistA Held Charges report for more than one Veteran as defined by the user.  **COH-3.2-09**  The system shall provide the capability for the user to generate the VistA Held Charges report for all Veterans.  **COH-3.2-10**  The system shall provide the capability for the user to choose whether insurance information is included on the VistA Held Charges report. Insurance information includes Insurance, Coverage Limitations, and Coverage Effective Dates.  **COH-3.2-11**  The system shall provide the capability for the user to choose whether Previously Released Copay charges are included on the VistA Held Charges report.  **COH-3.2-12**  The system shall generate the VistA Held Charges report in List Manager format.  **COH-3.2-13**  The system shall provide a List Manager action to export the VistA Held Charges report in Excel format.  **COH-3.2-14**  The system shall provide a List Manager action to release held co-pay charges for a selected line item on the VistA Held Charges report.  **COH-3.2-15**  The system shall limit the ability to release held co-pay charges from the VistA Held Charges report List Manager screen by requiring the user to possess the IB AUTHORIZE security key. | High |
| BN 3: Provide enhancements to the VistA Held Charges Report to address current CPAC processes and use of the report. | 3.3 | System shall perform an internal sort so file output will group copayment charge data by patient eliminating duplicate entries of patient name and insurance information. | **COH-3.3-01**  The system shall group data by patient on the VistA Held Charges report, with each copayment appearing as a separate line item within each patient.  (Note: presently the report groups data by payments, causing the same patient to appear in multiple locations on the report.)  **COH-3.3-02**  The system shall include only charges having on “on hold” status on the VistA Held Charges report.  **COH-3.3-03**  The system shall generate the Patient ID on the VistA Held Charges report as the first letter of the last name/last four digits of the Veteran’s SSN. For example, a Veteran whose last name begins with the letter ‘B’ and whose SSN ends with 1234 will appear on the report as B1234. | High |
| BN 3: Provide enhancements to the VistA Held Charges Report to address current CPAC processes and use of the report. | 3.4 | System shall only pull PI information relevant to the oldest date of held charges for a patient (date of service). | **COH-3.4-01**  The system shall include only the insurance information (as listed in requirement COH-3.2-10) that was active at the time of the episode of care relating to the held copay charges on the VistA Held Charges Report. | High |

### Overview of Functional Workload/Performance Requirements

Not applicable – there are no functional workload/performance requirements associated with this enhancement project.

### Overview of Operational Requirements

Not applicable – there are no changes to existing operational requirements.

### Overview of Technical Requirements

Not applicable – there are no technical requirements that drive the conceptual design.

### Overview of Security or Privacy Requirements

All VA and VHA security or privacy requirements will be adhered to. Cross-cutting security requirements are contained in the VA Enterprise Requirements Repository (ERR). Efforts that involve the collection and maintenance of individually identifiable information must be covered by a Privacy Act system of records notice.

### Overview of System Criticality and High Availability Requirements

Not applicable – there are no changes to existing system criticality or high availability requirements.

### Single Sign-on Requirements

Not applicable – there are no changes to the existing single sign-on requirements.

### Requirement for Use of Enterprise Portals

Not applicable – there are no changes to the requirements for use of enterprise portals.

### Special Device Requirements

Not applicable – no special devices are required.

# Conceptual Design

This project is composed of a VistA IB patch and there are no conceptual design considerations. The actual software detailed design is presented in Section 6.2.2 Specific Requirements.

## Conceptual Application Design

### Application Context

VistA is deployed universally across VHA at more than 1,500 sites of care, including each Veterans Affairs Medical Center (VAMC), Community Based Outpatient Clinic (CBOC) and Community Living Center (CLC), as well as at nearly 300 VA Vet Centers. The Fee Basis and Integrated Billing application modules are located within the nationally released VistA application. This will not change as a result of this project.

Figure 1: Application Context

### High-Level Application Design

Not applicable – this project does not require any changes to the high-level conceptual application design.

### Application Locations

Not applicable – this project does not require any changes to the application locations.

## Conceptual Data Design

### Project Conceptual Data Model

The database being utilized for this project is VistA. Within the VistA database, two software modules will be utilized – Fee Basis and Integrated Billing.

Table 4 Database Information

|  |  |  |  |
| --- | --- | --- | --- |
| Database Name | Description | Type | Steward |
| VistA | Veterans Health Information Systems and Technology Architecture | I | VA |

The VistA Fee Basis module, and associated Mumps database, provides for more efficient and accurate operation of the fee for service program with reduction of paperwork, savings in staff hours, minimization of errors, and by allowing medical facilities to have greater control over disbursement of fee medical, pharmacy, and travel monies.

The VistA Integrated Billing module, and associated Mumps database, allows for the capture, maintenance, and storage of insurance data including policy information and related benefits. It provides the ability to electronically transmit bills to payers with the data required by HIPAA. It also provides the ability to receive and store electronic 835 Health Care Claim Payment/Advice. It includes the ability to create printed bills which can then be mailed to payers who are unable to accept an electronic claim. A Claims Tracking feature is available to assist utilization review staff in tracking episodes of care, completing pre-certifications, completing continued stay reviews, and processing appeals and denials. An Automated Biller module provides a process that automatically creates bills for billable events which can then be finished and authorized by the billing staff. The IB software also provides many reporting features that support the billing staff by providing statistics, tracking and historical information.



Figure 2: Data Model

For a more detailed visual of the VistA data model, refer to the Entity Relationship Diagram (ERD) available via TSPR here:

<http://xxxxxxxxxxxxxxxxxx/warboard/anotebk.asp?proj=1785&Type=Active>

Until the ERD is approved and posted to TSPR, it will be made available upon request from the project team.

### Database Information

### User Interface Data Mapping

VistA Fee Basis receives data from the Fee Basis Claims System (FBCS), a GUI system that sends authorization and claim data to the VistA Fee Basis system. These fields are mapped in VistA, via Remote Procedure Calls (RPCs). There will be no changes to VistA Fee Basis interface mapping for this project.

The VistA Integrated Billing package electronically transmit bills to payers with the data required by HIPAA. It also provides the ability to receive and store electronic 835 Health Care Claim Payment/Advice. There will be no changes to VistA Integrated Billing interface mapping for this project.

## Conceptual Infrastructure Design

### System Criticality and High Availability

No changes are expected to system criticality and high availability, as a result of this project. Reliability and availability requirements for VistA are managed by regional data centers.

### Special Technology

VistA has historically been built on a client-server architecture, which ties together workstations and personal computers with user interfaces at VA facilities. No special technologies will be modified as a result of this project.

### Technology Locations

All software changes for this project will be completed within nationally released VistA software code. Mumps routines (programs) will be modified and added to the Fee Basis and Integrated Billing software.

### Conceptual Infrastructure Diagram

#### Location of Environments and External Interfaces

Please see Figure 1 and Figure 2. Modifications for this project will be completed within the bounds of the VistA system, only.

#### Conceptual Production String Diagram

No conceptual production string diagram exists for the VistA Fee Basis and Integrated Billing modules. The standard VistA configuration will be utilized.

# System Architecture

This project is composed of a VistA patch which does not require any changes to the VistA architecture or hardware architecture. However, brief descriptions of these entities are provided below.

## Hardware Architecture

The IB application is part of the VistA suite of applications. The VistA Monograph states:

“VistA is deployed universally across VHA at more than 1,500 sites of care, including each VAMC, Community Based Outpatient Clinic (CBOC) and Community Living Center (CLC), as well as at nearly 300 VA Vet Centers. These locations will have a variety of hardware platforms but all meet a minimal standard required to support VistA.”

## Software Architecture

The IB system is composed of multiple VistA applications. They also interact with many other VistA standard modules, such as Mailman, Kernel, Health Level 7 (HL7), and FileMan. The entire VistA application consists of 200+ modules. The VistA monograph includes this description for VistA:

“VistA is an integrated Electronic Health Record (EHR) information technology system with application packages that share a common data store and common internal services. The data store and VistA kernel are implemented in the Massachusetts General Hospital Utility Multi-Programming System (MUMPS) computer language, and the Computerized Patient Record System (CPRS) Graphical User Interface (GUI) is implemented in Delphi. Application clients use a highly-efficient proprietary protocol to access data. VistA is highly configurable and customizable, and in addition to appropriate connectivity amongst VistA modules, VistA supports the integration of best-of-breed applications at multiple levels, including MUMPS Application Programming Interface (API), Remote Procedure Call (RPC), Medical Domain Web Services (MDWS), HL7, and data exchange via Blue Button or eHealth Exchanges. VistA comprises nearly 200 distinct applications/modules, 15,000 routines, and millions of lines of computer code.

The backbone of VHA’s clinical and administrative information technology capability, VistA has historically been built on a client-server architecture, which ties together workstations and personal computers with GUIs at VA facilities. The CPRS GUI is as highly customizable and runs on workstations, laptops, tablets (including iPads) and smart phones. VistA interoperates with numerous Commercial Off-the-Shelf (COTS) software applications and with selected information technology systems of other federal agencies and, increasingly, health information exchange networks. At the time of publication of this edition of the Monograph, comprehensive proposed enhancements to VistA were in the initial stages; referred to as “VistA Evolution” these enhancements are will reflect development and architecture enhancements to allow greater interaction with data and greater efficiency for the VistA system.”

Note that the changes required by this project are based in the MUMPS computer language and do not impact CPRS.

## Network Architecture

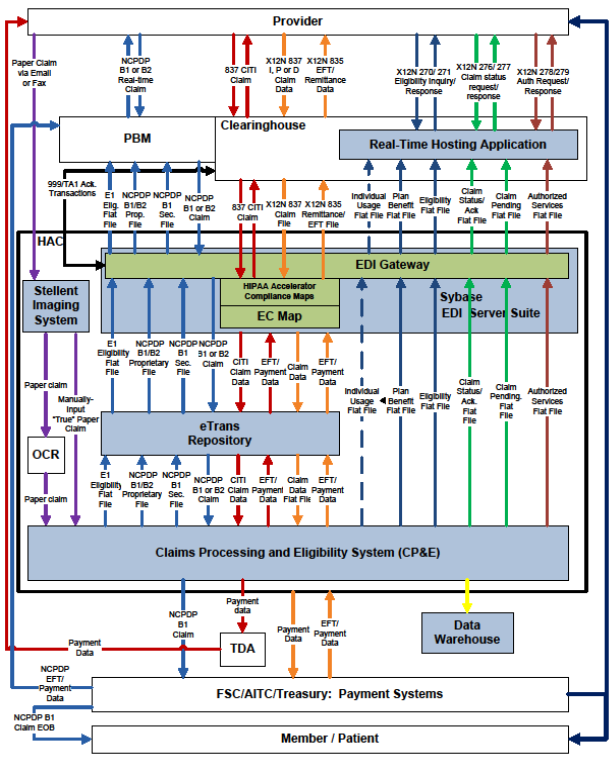


Figure 3 Netwrok Architecture

## SOA/ESS

Not applicable – these applications do not use Service Oriented Architecture (SOA)/Enterprise Shared Services (ESS).

## Enterprise Architecture

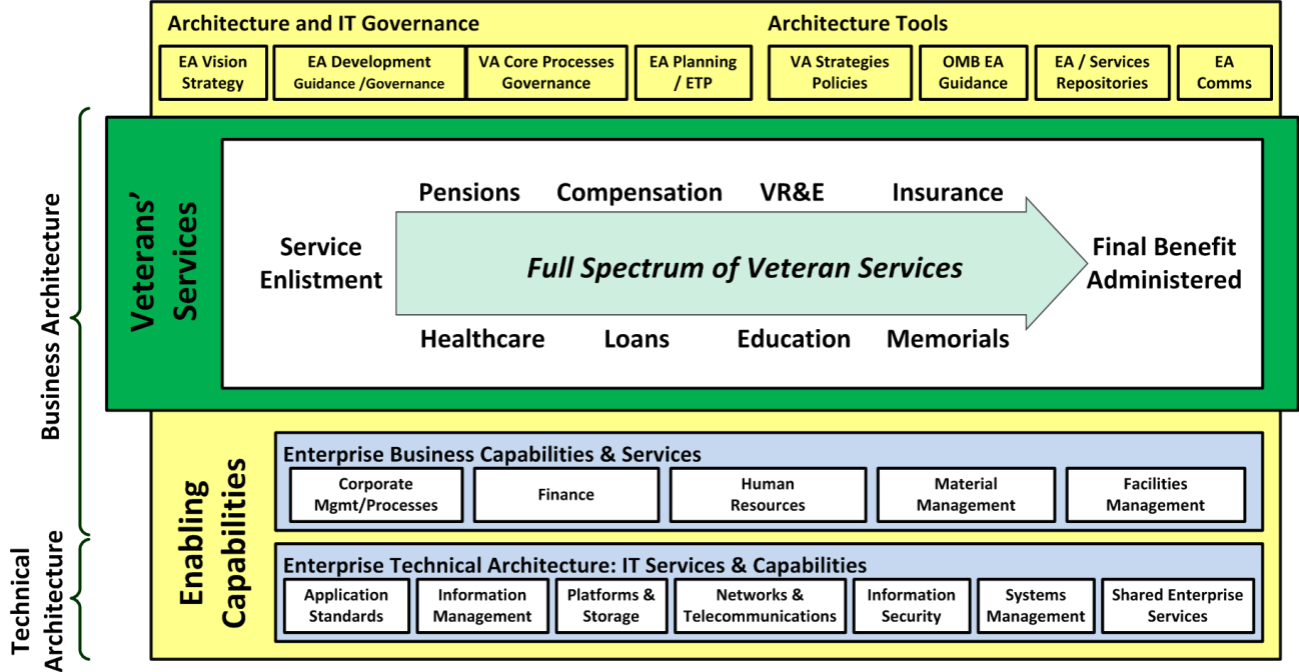


Figure 4 Enterprise Architecture

# Data Design

## Database Management System (DBMS) Files

This is a VistA INTEGRATED BILLING patch. The database management system (DBMS) in use is VA FileMan.

The VistA INTEGRATED BILLING package uses the standard VA FileMan data dictionaries (DDs) and files.

## Non-DBMS Files

There are no non-DBMS Files in use.

## Data View

This project is a VistA INTEGRATED BILLIGN patch. The ERD is one of the deliverables with this project.

# Detailed Design

The detailed design will be updated as design and development continues; this section will be updated and included with later versions of this SDD. Blank sections have been kept for version control.

## Hardware Detailed Design

This project does not require any changes to the existing hardware. These enhancements may require an additional amount of storage for the new files described above. Further analysis will be done as a part of the Systems Engineering and Design Review.

## Software Detailed Design

### Conceptual Design

#### Product Perspective

The VistA eIV system is a software package where third-party insurance claims are entered and managed that resides in IB.

##### User Interfaces

User Interface with all IB application software is performed through existing and new VistA screens. VistA is a character based application accessible through terminal emulator software resident on networked computers.

The IB application is a character-based system using VA FileMan as its database manager and Cache/M as the programming language. This software application is part of the VistA suite of applications, which include additional clinical and financial applications. There are no GUIs.

##### Hardware Interfaces

Communications between VistA sites and HIPAA EDI Services pass through the Local Area Networks (LANs) and across the Department of Veterans Affairs (VA) intranet. The Services communicate over the Austin Information Technology Center (AITC) LAN and through the private frame relay to the communications clearinghouse for this project. The clearinghouse communicates with the trading partner processors that are continually designated using previously negotiated contractual terms and conditions.

##### Software Interfaces

The following software must be installed prior to the release of this product.

* Kernel V. 8.0
* MailMan V. 8.0
* VA FileMan V. 22.0
* Health Level Seven (HL7) V. 1.6
* Integrated Billing V. 2.0
* InterSystems Cache Cube 2011.1.2

##### Communications Interfaces

For IB, the primary communication is the submission of insurance eligibility inquiries and health care claim transactions from the VistA sites to third-party payers via the FSC AITC and a clearinghouse and then receiving a response. Refer to Sec. 3.2.2 for sharing between VistA Databases.

##### Memory Constraints

There are no memory constraints associated with this project.

##### Special Operations

There are no special operations associated with this project.

#### Product Features

The product shall provide the following functionality:

* Add a List Manager Held Charges Report to enable the users to process any held charges for a patient.
* The List Manager will include the actions to allow the user to adequately process the Held Charges on the Report. These will include Patient Insurance, Claim Tracking, TPJI, and Account Profile.
* Update other held charges routines to help in processing held charges.

#### User Characteristics

In general, the resources that shall work with IB software need to be knowledgeable in the area of Copays on Hold.

#### Dependencies and Constraints

* VA Standards and Conventions for MUMPS development
* VA Technical Reference Model (TRM)

### Specific Requirements

Please see Table 3.

#### Database Repository

This project does not include a change to the existing logical database design.

#### System Features

The RTM designator on Table 4: (Grouping): Routines identifies the specific functional requirements as presented in the Requirements Specification Document (RSD) for this project, with regards to each routine change.

#### Design Element Tables

##### Routines (Entry Points)

Table 5: (Grouping): Routines

| Routines | Activities | | | |
| --- | --- | --- | --- | --- |
| **Routine Name** | IBCNBAR | | | |
| **Enhancement Category** | New | Modify | Delete | No Change |
| **RTM** | COH-2.1-01 | | | |
| **Related Options** |  | | | |

| Related Routines | Routines “Called By” | Routines “Called” |
| --- | --- | --- |
|  | ^IBCNBAA  ^IBCNBLA1 | FILE^DIE, $$GET1^DIQ, ^DIR, GRP^IBCNBCD, INS^IBCNBCD, POLICY^IBCNBCD, DELDATA^IBCNBED, STATUS^IBCNBEE, $$HOLD^IBCNBLL, GRP^IBCNBMI, INS^IBCNBMI, PAT^IBCNBMI, POLICY^IBCNBMI, $$NEWGRP^IBCNBMN, $$NEWINS^IBCNBMN, $$NEWPOL^IBCNBMN, $$BUFFER^IBCNBU1, WAIT^IBCNBUH, EBFILE^IBCNEHL1, UPDIREC^IBCNEHL3, UPDPOL^IBCNICB, $$INSURED^IBCNS1, ^IBCNSEVT, AFTER^IBCNSEVT, COVERED^IBCNSM31, $$DUPCO^IBCNSOK1, $$DUPPOL^IBCNSOK1, POL^IBCNSU41, $$PTHLD^IBOHCR, $$UPDATE^IVMLINS4 |

| Routines | Activities | | | | |
| --- | --- | --- | --- | --- | --- |
| **Data Dictionary (DD) References** | ^IBA(355.33,  ^IBCN(365 | | | | |
| **Related Protocols** |  | | | | |
| **Related Integration Control Registrations (ICRs)** |  | | | | |
| **Data Passing** | Input | Output Reference | Both | Global Reference | Local |
| **Input Attribute Name and Definition** | Name: NONE  Definition: NONE | | | | |
| **Output Attribute Name and Definition** | Name: NONE  Definition: NONE | | | | |

| Current Logic |
| --- |
| ; if source is eIV, update insurance record field in transmission queue (365.1/.13)  I $P(^IBA(355.33,IBBUFDA,0),U,3)=5 D UPDIREC^IBCNEHL3($O(^IBCN(365,"AF",IBBUFDA,"")),IBPOLDA)  ; update buffer file entry so only stub remains and status is changed  D STATUS^IBCNBEE(IBBUFDA,"A",IBNEWINS,IBNEWGRP,IBNEWPOL) ; update buffer entry's status to accepted  D DELDATA^IBCNBED(IBBUFDA) ; delete buffer's insurance/patient data;  S IBCDFN=IBPOLDA S:+IBSOURCE=3 IVMINSUP=1 D AFTER^IBCNSEVT,^IBCNSEVT ; insurance event driver  ;  ACCPTQ Q  REJPROC ;Entry point for REJECAPI^IBCNICB (Patch 413)  ;  N DFN S DFN=+$G(^IBA(355.33,+IBBUFDA,60))  S RESULT="-1^PATIENT IEN MISSING FROM BUFFER ENTRY" Q:'$G(DFN)  I +$P($G(^IBA(355.33,+IBBUFDA,0)),U,3)=3 D IVM(0,IBBUFDA,$G(IVMREPTR),$G(IBSUPRES))  ;  S RESULT=0  D STATUS^IBCNBEE(+IBBUFDA,"R",0,0,0),DELDATA^IBCNBED(+IBBUFDA) W:$G(IBSUPRES)'>0 " ... done."  Q |

| Modified Logic (Changes are highlighted) |
| --- |
| CLEANUP ; general updates and checks done whenever insurance is added/edited and clean up buffer file  N IBSOURCE S IBSOURCE=$P($G(^IBA(355.33,IBBUFDA,0)),U,3)  N RELHLD S RELHLD=0  ;  ;Don't do PAT^IBCNBMI for ICB ACCEPAPI^IBCNICB interface  I $G(IBSUPRES)'>0,+IBPOLDA D PAT^IBCNBMI(DFN,IBPOLDA) ; update DOB&SSN of Pat Ins from Pat file  D POL^IBCNSU41(DFN) ; update Tricare sponsor data  D COVERED^IBCNSM31(DFN) ; update 'Covered by Insurance' field (2,.3192  I +IBSOURCE=3 D IVM(1,IBBUFDA,$G(IVMREPTR),$G(IBSUPRES)) ; update/notify IVM  ;Suppress Write in $$DUPCO^IBCNSOK1 if called from ICB Interface  I +IBINSDA,+IBPOLDA S IBX=$$DUPCO^IBCNSOK1(DFN,IBINSDA,IBPOLDA,$S($G(IBSUPRES)>0:0,1:1)) ; warning if duplicate policy added for patient  S RESULT(0)="0"\_$S($G(IBX):"^Warning - Duplicate or inconsistent insurance data",1:"")  ;  ;Suppress Write in $$DUPPOL^IBCNSOK1 if called from ICB Interface  I +IBGRPDA S IBX=$$DUPPOL^IBCNSOK1(IBGRPDA,$S($G(IBSUPRES)>0:0,1:1)) ; warning if duplicate plan was added  S:IBX RESULT(0)=RESULT(0)\_"^Warning - Duplicate or inconsistent policy data"  ;  ;Suppress Write in $$PTHLD^IBOHCR if called from ICB Interface  I +IBNEWPOL I +$$PTHLD^IBOHCR(DFN,1,$S($G(IBSUPRES)>0:0,1:1)) D  . W:$G(IBSUPRES)'>0 !!,"Patient's bills On Hold date updated due to new insurance."  . S RESULT(0)=RESULT(0)\_"^Patient's bills On Hold date updated due to new insurance"  ;  I $$HOLD^IBCNBLL(DFN) D  . W:$G(IBSUPRES)'>0 !!,"There are bills On Hold for this patient."  . S RESULT(0)=RESULT(0)\_"^There are bills On Hold for this patient"  ;  ;Suppress DIR call functionality for ICB ACCEPAPI^IBCNICB interface  D:$G(IBSUPRES)'>0  . W !! S DIR(0)="FO",DIR("A")="Press 'V' to view the changes or Return to continue" D ^DIR  . I Y="V"!(Y="v") W !! D INS^IBCNBCD(IBBUFDA,IBINSDA),WAIT^IBCNBUH,GRP^IBCNBCD(IBBUFDA,IBGRPDA),WAIT^IBCNBUH,POLICY^IBCNBCD(IBBUFDA,IBPOLDA),WAIT^IBCNBUH  ;  ; if source is eIV, update insurance record field in transmission queue (365.1/.13)  I $P(^IBA(355.33,IBBUFDA,0),U,3)=5 D UPDIREC^IBCNEHL3($O(^IBCN(365,"AF",IBBUFDA,"")),IBPOLDA)  ; update buffer file entry so only stub remains and status is changed  D STATUS^IBCNBEE(IBBUFDA,"A",IBNEWINS,IBNEWGRP,IBNEWPOL) ; update buffer entry's status to accepted  D DELDATA^IBCNBED(IBBUFDA) ; delete buffer's insurance/patient data  ;  ; IB\*2.0\*554  S RELHLD=$$RELHLD(DFN)  I +DFN,RELHLD D  . ;Suppress Write in $$PTHLD^IBOHCR if called from ICB Interface  . I +$$PTHLD^IBOHCR(DFN,2,$S($G(IBSUPRES)>0:0,1:1)) D  . . I $G(IBSUPRES)'>0 W !!,"Patient has no other active Insurance.",!,"All patient bills On Hold waiting for Insurance to be released." D WAIT^IBCNBUH  . . S RESULT=RESULT\_"^Patient has no other active Insurance. All patient bills On Hold waiting for Insurance to be released."  ;  S IBCDFN=IBPOLDA S:+IBSOURCE=3 IVMINSUP=1 D AFTER^IBCNSEVT,^IBCNSEVT ; insurance event driver  ;  ACCPTQ Q  REJPROC ;Entry point for REJECAPI^IBCNICB (Patch 413)  ;  N DFN S DFN=+$G(^IBA(355.33,+IBBUFDA,60))  N RELHLD S RELHLD=0  S RESULT="-1^PATIENT IEN MISSING FROM BUFFER ENTRY" Q:'$G(DFN)  I +$P($G(^IBA(355.33,+IBBUFDA,0)),U,3)=3 D IVM(0,IBBUFDA,$G(IVMREPTR),$G(IBSUPRES))  ;  S RESULT=0  D STATUS^IBCNBEE(+IBBUFDA,"R",0,0,0),DELDATA^IBCNBED(+IBBUFDA) W:$G(IBSUPRES)'>0 " ... done."  ;  ; IB\*2.0\*554  S RELHLD=$$RELHLD(DFN)  I +DFN,RELHLD D  . ;Suppress Write in $$PTHLD^IBOHCR if called from ICB Interface  . I +$$PTHLD^IBOHCR(DFN,2,$S($G(IBSUPRES)>0:0,1:1)) D  . . I $G(IBSUPRES)'>0 W !!,"Patient has no other active Insurance.",!,"All patient bills On Hold waiting for Insurance to be released." D WAIT^IBCNBUH  . . S RESULT=RESULT\_"^Patient has no other active Insurance. All patient bills On Hold waiting for Insurance to be released."  ;  Q  ;IB\*2.0\*554  RELHLD(DFN) ;Check if need to release copay on hold  N INS,BUF,COV,RELHLD  S (RELHLD,INS,BUF,COV)=0  S INS=$$INSURED^IBCNS1(DFN)  S BUF=$$BUFFER^IBCNBU1(DFN)  I INS S COV=$$IBCOV(DFN)  ;  I INS,'IBCOV,'BUF S RELHLD=1  I 'INS,'BUF S RELHLD=1  Q RELHLD  ;  IBCOV(DFN) ;Check if coverage for date  N IBN,IBCOV  S (IBN,IBCOV)=0  F S IBN=$O(^IB("AH",DFN,IBN)) Q:IBN="" D  . S IBX=$G(^IB(IBN,0))  . S X5=+$P(IBX,U,5) Q:X5'=8  . S IBFR=$P(IBX,U,14)  . S X3=$P(IBX,U,3)  . S IBVDT=$S(IBFR'="":IBFR,1:DT),IBAT=$P(^IBE(350.1,X3,0),U,11)  . S IBCAT=$S(IBAT<4:"INPATIENT",IBAT=4:"OUTPATIENT",IBAT=5:"PHARMACY",IBAT=8:"OUTPATIENT",IBAT=9:"INPATIENT",1:"")  . I IBCAT'="" S IBCOV=$$PTCOV^IBCNSU3(DFN,IBVDT,IBCAT)  Q IBCOV |

| Routines | Activities | | | |
| --- | --- | --- | --- | --- |
| **Routine Name** | IBRUTL | | | |
| **Enhancement Category** | New | Modify | Delete | No Change |
| **RTM** | COH-2.2-01, COH-2.2-02 | | | |
| **Related Options** |  | | | |

| Related Routines | Routines “Called By” | Routines “Called” |
| --- | --- | --- |
|  | IBCCCB0, IBJTLA1  IBJTLB1, IBNCPUT1  IBR, IBRREL | ^DIE, $$BUFFER^IBCNBU1, ^IBCNS, $$PTCOV^IBCNSU3, $$HOLDECME^IBNCPUT1, UP3^IBR, ^IBRBUL, $$FMDIFF^XLFDT |

| Routines | Activities | | | | |
| --- | --- | --- | --- | --- | --- |
| **Data Dictionary (DD) References** | ^DGCR(399, ^DIC(36, ^IB(, ^IBA(362.4, ^IBE(350.1, ^IBE(350.9 | | | | |
| **Related Protocols** |  | | | | |
| **Related Integration Control Registrations (ICRs)** |  | | | | |
| **Data Passing** | Input | Output Reference | Both | Global Reference | Local |
| **Input Attribute Name and Definition** | Name: NONE  Definition: NONE | | | | |
| **Output Attribute Name and Definition** | Name: NONE  Definition: NONE | | | | |

| Current Logic |
| --- |
| HOLD(X,IBN,IBDUZ,IBSEQNO) ; Place IB Action on hold?  ; Input: X -- Zeroth node of IB Action  ; IBN -- ien of IB Action  ; IBDUZ -- User ID  ; IBSEQNO -- 1 (New Action), 3 (Update Action)  ; Returns: 1 -- Yes, 0 -- No  ;  N DFN,IBCOV,IBINDT,IBOUTP,HOLD,IBHOLDP,IBDUZ,I  N IBVDT,IBAT,IBCAT,IBALTC,TRI,CHP,J,XX  ;  S HOLD=0  S IBHOLDP=$P($G(^IBE(350.9,1,1)),"^",20) ; Site parameter - HOLD MT BILLS W/INSURANCE  S DFN=+$P(X,"^",2)  ;  ;check if ECME RX copay needs to be placed on HOLD  I $$HOLDECME^IBNCPUT1(X)=0 G HOLDQ  ;  I $P(X,"^",5)=8 G HOLDQ ; action is already on hold  I '$P($G(^IBE(350.1,+$P(X,"^",3),0)),"^",10) G HOLDQ ; action can't be placed on hold  ;  ; - see if patient has insurance on Charge 'To' Date (otherwise Event date)  ; - includes check of plan coverage limitation  S IBINDT=+$P($G(^IB(+$G(IBN),0)),U,15)  I 'IBINDT S IBINDT=+$P($G(^IB(+$P(X,"^",16),0)),"^",17) I 'IBINDT S IBINDT=DT  S IBOUTP=1  D ^IBCNS  ;  S IBVDT=$S(IBINDT'="":IBINDT,1:DT),IBAT=$P(^IBE(350.1,(+$P(X,U,3)),0),U,11)  S IBCAT=$S(IBAT<4:"INPATIENT",IBAT=4:"OUTPATIENT",IBAT=5:"PHARMACY",IBAT=8:"OUTPATIENT",IBAT=9:"INPATIENT",1:"")  S IBCOV="" I IBCAT'="" S IBCOV=$$PTCOV^IBCNSU3(DFN,IBVDT,IBCAT),HOLD=IBCOV  I 'IBCOV,+$$BUFFER^IBCNBU1(DFN) S (IBCOV,HOLD)=1 ; if patient has a buffer entry place charge on hold  ;  ;  ; - generate bulletin if patient has insurance, bulletin not suppressed  I IBCOV,'$P($G(^IBE(350.9,1,0)),"^",15),'$$ECME(IBN) D ^IBRBUL  ;  ; - update action to 'Hold' if parameter is set and vet has insurance  I IBHOLDP,IBCOV S DIE="^IB(",DA=IBN,DR=".05////8" D ^DIE,UP3^IBR:IBSEQNO=3 K DA,DIE,DR  ;  HOLDQ Q +$G(HOLD) |

| Modified Logic (Changes are highlighted) |
| --- |
| HOLD(X,IBN,IBDUZ,IBSEQNO) ; Place IB Action on hold?  ; Input: X -- Zeroth node of IB Action  ; IBN -- ien of IB Action  ; IBDUZ -- User ID  ; IBSEQNO -- 1 (New Action), 3 (Update Action)  ; Returns: 1 -- Yes, 0 -- No  ;  N DFN,IBCOV,IBINDT,IBOUTP,HOLD,IBHOLDP,IBDUZ,I  N IBVDT,IBAT,IBCAT,IBALTC,TRI,CHP,J,XX  ;  S HOLD=0  S IBHOLDP=$P($G(^IBE(350.9,1,1)),"^",20) ; Site parameter - HOLD MT BILLS W/INSURANCE  S DFN=+$P(X,"^",2)  ;  ;check if ECME RX copay needs to be placed on HOLD  I $$HOLDECME^IBNCPUT1(X)=0 G HOLDQ  ;  I $P(X,"^",5)=8 G HOLDQ ; action is already on hold  I '$P($G(^IBE(350.1,+$P(X,"^",3),0)),"^",10) G HOLDQ ; action can't be placed on hold  ;  ; - see if patient has insurance on Charge 'To' Date (otherwise Event date)  ; - includes check of plan coverage limitation  S IBINDT=+$P($G(^IB(+$G(IBN),0)),U,15)  I 'IBINDT S IBINDT=+$P($G(^IB(+$P(X,"^",16),0)),"^",17) I 'IBINDT S IBINDT=DT  S IBOUTP=1  D ^IBCNS  ;  ; IB\*2.0\*544 - baa  ;When placing charges On Hold added check so if only insurance for the  ;patient is Tricare or CHAMPVA the charge is Not placed On Hold.  ;Previously if the patient had any active insurance the charge was placed On Hold.  I IBINS D  . N XX  . S (J,TRI,CHP)=0  . S XX=0 F S XX=$O(IBDD(XX)) Q:XX="" D  .. S ZZ=IBDD(XX),INS=+ZZ\_","  .. S TYPNAM=$$GET1^DIQ(36,INS,.13,"E"),J=J+1  .. S:TYPNAM["TRICARE" TRI=1 S:TYPNAM["CHAMPVA" CHP=1  ;  I $G(J)=1,TRI G HOLDQ ; don't place on hold if only has TRICARE  I $G(J)=1,CHP G HOLDQ ; don't place on hold if only has CHAMPVA  ; IB\*2.0\*544 – baa  ;  S IBVDT=$S(IBINDT'="":IBINDT,1:DT),IBAT=$P(^IBE(350.1,(+$P(X,U,3)),0),U,11)  S IBCAT=$S(IBAT<4:"INPATIENT",IBAT=4:"OUTPATIENT",IBAT=5:"PHARMACY",IBAT=8:"OUTPATIENT",IBAT=9:"INPATIENT",1:"")  S IBCOV="" I IBCAT'="" S IBCOV=$$PTCOV^IBCNSU3(DFN,IBVDT,IBCAT),HOLD=IBCOV  I 'IBCOV,+$$BUFFER^IBCNBU1(DFN) S (IBCOV,HOLD)=1 ; if patient has a buffer entry place charge on hold  ;  ;  ; - generate bulletin if patient has insurance, bulletin not suppressed  I IBCOV,'$P($G(^IBE(350.9,1,0)),"^",15),'$$ECME(IBN) D ^IBRBUL  ;  ; - update action to 'Hold' if parameter is set and vet has insurance  I IBHOLDP,IBCOV S DIE="^IB(",DA=IBN,DR=".05////8" D ^DIE,UP3^IBR:IBSEQNO=3 K DA,DIE,DR  ;  HOLDQ Q +$G(HOLD)  ; |

| Routines | Activities | | | |
| --- | --- | --- | --- | --- |
| **Routine Name** | IBOHLS | | | |
| **Enhancement Category** | New | Modify | Delete | No Change |
| **RTM** | COH-3.1-01 - COH-3.1-15 | | | |
| **Related Options** | IB OUTPUT HELD CHARGES LM | | | |

| Related Routines | Routines “Called By” | Routines “Called” |
| --- | --- | --- |
|  | IBOHLS2 | ^%DT, ^%ZISC, ^DIC, $$GET1^DIQ, ^DIR, SORT^IBOHLS1, EN^VALM, CLEAR^VALM1 |

| Routines | Activities | | | | |
| --- | --- | --- | --- | --- | --- |
| **Data Dictionary (DD) References** | ^TMP($J | | | | |
| **Related Protocols** | IB EXPAND HELD CHARGES  IB EXPORT TO FILE  IB HELD CHARGES ACCOUNT PROFILE  IB HELD CHARGES CLAIM TRACKING  IB HELD CHARGES PATIENT INSURANCE  IB HELD CHARGES TPJI  IB PRINT HELD CHARGES RPT  IB RELEASE HELD CHARGES | | | | |
| **Related Integration Control Registrations (ICRs)** |  | | | | |
| **Data Passing** | Input | Output Reference | Both | Global Reference | Local |
| **Input Attribute Name and Definition** | Name: NONE  Definition: NONE | | | | |
| **Output Attribute Name and Definition** | Name: NONE  Definition: NONE | | | | |

| Current Logic |
| --- |
| NEW ROUTINE |

| Modified Logic (Changes are highlighted) |
| --- |
| IBOHLS ;ALB/JWS,BAA - IB HELD CHARGES LIST MANAGER ; 08-SEP-2015  ;;2.0;INTEGRATED BILLING;\*\*554\*\*;21-MAR-94;Build 3  ;;Per VA Directive 6402, this routine should not be modified.  ;;  ;  EN ; -- main entry point for HELD CHARGES LIST  ; add code to do filters here  N FILTERS  I '$$FILTER(.FILTERS) Q  ;  ; code to do sort  D SORT  ;  D EN^VALM("IBOH HELD CHARGES LIST")  D ^%ZISC  Q  ;  HDR ; -- header code  ;  N BDATE,EDATE,INSTS,PATS,IINS,OLDH  N VAL  S BDATE=$P(FILTERS(0),U,1),EDATE=$P(FILTERS(0),U,2)  S INSTS=$P(FILTERS(0),U,3),PATS=$P(FILTERS(0),U,4)  S IINS=FILTERS(3)  ;  S VALM("TITLE")=" Held Charges"  Q  ;  INIT ; -- init variables and list array  ; input - none  ; output ^TMP($J,"IBOHLS")  N BDATE,EDATE,INSTS,PATS,IINS,OLDH  S BDATE=$P(FILTERS(0),U,1),EDATE=$P(FILTERS(0),U,2)  S INSTS=$P(FILTERS(0),U,3),PATS=$P(FILTERS(0),U,4)  S IINS=FILTERS(3)  D BLD  Q  ;  SORT ; get the data  N BDATE,EDATE,INSTS,PATS,IINS,OLDH  S BDATE=$P(FILTERS(0),U,1),EDATE=$P(FILTERS(0),U,2)  S INSTS=$P(FILTERS(0),U,3),PATS=$P(FILTERS(0),U,4)  S IINS=FILTERS(3),CNT=0  S ^TMP($J,"IBOHLSF")=FILTERS(0)  K ^TMP($J,"IBOHLS")  K ^TMP($J,"IBHOLD")  K ^TMP($J,"IBOHLS INS")  ;  D SORT^IBOHLS1  Q  ;  EXPAND ; -- expand code  D FULL^VALM1  N I,J,IBXX,VALMY,ECNT,PNAME,DFN,IBHLD0,REC  D EN^VALM2($G(XQORNOD(0)))  I $D(VALMY) S IBXX=0 F S IBXX=$O(VALMY(IBXX)) Q:'IBXX D ;W !,"Entry ",X,"Selected" D  . K ^TMP($J,"IBOHLE")  . S REC=$G(^TMP($J,"IBOHLSX",IBXX))  . S DFN=$P(REC,U,1),PNAME=$P(REC,U,2),ECNT=$P(REC,U,3)  . S IBIEN=$P(^TMP($J,"IBOHLS",PNAME,ECNT,"IBND"),U,3)  . Q:IBIEN=""  . S ^TMP($J,"IBOHLSE")=DFN\_U\_ECNT\_U\_PNAME\_U\_IBIEN  . D EN^VALM("IBOH HELD CHARGES EXPAND")  . Q  D BLD  S VALMBCK="R"  Q  ;  BLD ; build data to display  ; build display  K ^TMP($J,"IBOHLSX")  K ^TMP("VALMAR",$J)  N FIRST,VCNT,CNT,NAME,BCNT,RNB,RX  S VALMCNT=0  S (CNT,VCNT)=0,NAME=""  F S NAME=$O(^TMP($J,"IBOHLS",NAME)) Q:NAME="" D  . S FIRST=1  . F S CNT=$O(^TMP($J,"IBOHLS",NAME,CNT)) Q:CNT="" D  .. S VCNT=VCNT+1  .. S LINE=$$SETL("",VCNT,"",1,4) ;line#  .. S XX=^TMP($J,"IBOHLS",NAME,CNT)  .. S DFN=$P(^TMP($J,"IBOHLS",NAME,CNT,"IBND"),U,1)  .. S NAME=$P(XX,U)  .. S LINE=$$SETL(LINE,NAME,"",5,22)  .. S LINE=$$SETL(LINE,$P(XX,U,2),"",28,6)  .. S LINE=$$SETL(LINE,$P(XX,U,3),"",37,6)  .. S LINE=$$SETL(LINE,$$FMTE^XLFDT($P(XX,U,4),"2DZ"),"",44,8)  .. S LINE=$$SETL(LINE,$$FMTE^XLFDT($P(XX,U,5),"2DZ"),"",54,8)  .. S LINE=$$SETL(LINE,$P(XX,U,6),"",64,5)  .. S LINE=$$SETL(LINE,$J($P(XX,U,7),8,2),"",72,8)  .. S VALMCNT=VALMCNT+1  .. D SET^VALM10(VALMCNT,LINE,VCNT)  .. S ^TMP($J,"IBOHLSX",VCNT)=DFN\_U\_NAME\_U\_CNT  .. I $D(^TMP($J,"IBOHLS",NAME,CNT,1)) D  ... S RX=^TMP($J,"IBOHLS",NAME,CNT,1),RX="Rx#:"\_RX  ... S LINE=$$SETL("",RX,"",37,20)  ... S VALMCNT=VALMCNT+1  ... D SET^VALM10(VALMCNT,LINE,VCNT)  .. I $D(^TMP($J,"IBOHLS",NAME,CNT,2)) D  ... S BCNT=0 F S BCNT=$O(^TMP($J,"IBOHLS",NAME,CNT,2,BCNT)) Q:BCNT="" D  .... S XX=^TMP($J,"IBOHLS",NAME,CNT,2,BCNT)  .... S LINE=$$SETL("","Bill: ","",6,6)  .... S LINE=$$SETL(LINE,$P(XX,U),"",14,10)  .... S LINE=$$SETL(LINE,$P(XX,U,2),"",26,10)  .... S LINE=$$SETL(LINE,$$FMTE^XLFDT($P(XX,U,3),"2DZ"),"",38,10)  .... S LINE=$$SETL(LINE,$$FMTE^XLFDT($P(XX,U,4),"2DZ"),"",50,10)  .... S LINE=$$SETL(LINE,$P(XX,U,5),"",62,4)  .... S LINE=$$SETL(LINE,$J($P(XX,U,6),8,2),"",68,10)  .... S VALMCNT=VALMCNT+1  .... D SET^VALM10(VALMCNT,LINE,VCNT)  .... S RNB=$P(XX,U,7)  .... I RNB'="" D  ..... S LINE=$$SETL("","RNB: ","",6,6)  ..... S LINE=$$SETL(LINE,RNB,"",14,60)  ..... S VALMCNT=VALMCNT+1  ..... D SET^VALM10(VALMCNT,LINE,VCNT)  .. I $D(^TMP($J,"IBOHLS INS",NAME)),FIRST D ; IF DISPLAYING INSURANCE INFORMATION  ... S FIRST=0  ... N ZZ,ZZ1,ZZ2  ... S LINE=$$SETL("","Insurance","",6,9)  ... S LINE=$$SETL(LINE,"Subscriber","",24,10)  ... S LINE=$$SETL(LINE,"Group","",42,5)  ... S LINE=$$SETL(LINE,"Eff Dt","",58,6)  ... S LINE=$$SETL(LINE,"Exp Dt","",70,6)  ... S VALMCNT=VALMCNT+1  ... D SET^VALM10(VALMCNT,LINE,VCNT)  ... S VALMCNT=VALMCNT+1  ... S $P(ZZ2,"-",78)=""  ... S LINE=$$SETL("",ZZ2,"",6,78)  ... D SET^VALM10(VALMCNT,LINE,VCNT)  ... S ZZ=0 F S ZZ=$O(^TMP($J,"IBOHLS INS",NAME,ZZ)) Q:ZZ="" D  .... S ZZ1=^TMP($J,"IBOHLS INS",NAME,ZZ)  .... S LINE=$$SETL("",$P(ZZ1,U),"",6,15)  .... S LINE=$$SETL(LINE,$P(ZZ1,U,2),"",24,15)  .... S LINE=$$SETL(LINE,$P(ZZ1,U,3),"",42,14)  .... S LINE=$$SETL(LINE,$$FMTE^XLFDT($P(ZZ1,U,4),"2DZ"),"",58,10)  .... S LINE=$$SETL(LINE,$$FMTE^XLFDT($P(ZZ1,U,5),"2DZ"),"",70,10)  .... S VALMCNT=VALMCNT+1  .... D SET^VALM10(VALMCNT,LINE,VCNT)  .... I '$O(^TMP($J,"IBOHLS INS",NAME,ZZ,0)) Q  .... S LINE=$$SETL("","Plan Coverage Eff. Date Covered? Limit Comments","",10,60)  .... S VALMCNT=VALMCNT+1  .... D SET^VALM10(VALMCNT,LINE,VCNT)  .... S ZZ2=0 F S ZZ2=$O(^TMP($J,"IBOHLS INS",NAME,ZZ,ZZ2)) Q:ZZ2="" D  ..... S ZZ1=^TMP($J,"IBOHLS INS",NAME,ZZ,ZZ2)  ..... S LINE=$$SETL("",$P(ZZ1,U),"",10,15)  ..... S LINE=$$SETL(LINE,$P(ZZ1,U,2),"",27,8)  ..... S LINE=$$SETL(LINE,$P(ZZ1,U,3),"",40,12)  ..... S LINE=$$SETL(LINE,$P(ZZ1,U,4),"",55,25)  ..... S VALMCNT=VALMCNT+1  ..... D SET^VALM10(VALMCNT,LINE,VCNT)  Q  ;  SETL(LINE,DATA,LABEL,COL,LNG) ; Creates a line of data to be set into the body  ; of the worklist  ; Input: LINE - Current line being created  ; DATA - Information to be added to the end of the current line  ; LABEL - Label to describe the information being added  ; COL - Column position in line to add information add  ; LNG - Maximum length of data information to include on the line  ; Returns: Line updated with added information  S LINE=LINE\_$J("",(COL-$L(LABEL)-$L(LINE)))\_LABEL\_$E(DATA,1,LNG)  Q LINE  ;  HELP ; -- help code  S X="?" D DISP^XQORM1 W !!  Q  ;  EXIT ; -- exit code  K ^TMP($J,"IBOHLSF")  K ^TMP($J,"IBOHLS")  K ^TMP($J,"IBHOLD")  K ^TMP($J,"IBOHLS INS")  ;  D CLEAR^VALM1,CLEAN^VALM10  D ^%ZISC  Q  ;  FILTER(FILTERS) ; filter display  ; Sets an array of filters to determine which entris to include in display  ; Input: None  ; Output:  ; Returns: 0 if the user entered '^' or timed out, 1 otherwise  ; FILTERS(0) = from date ^ to date ^ 0 (all) 1 (selected) institutions ^ 0 (all) 1 (selected) patients  ; FILTERS(1) = inst ien ^ inst ien ^ etc...  ; FILTERS(2) = pat ien ^ pat ien ^ etc...  ; FILTERS(3) = 0 (NO) 1 (YES) to include insurance information  ; FILTERS(4) = 0 (NO) 1 (YES) to include previously released copay charges  N DIR,DIROUT,DIRUT,DTOUT,DUOUT,X,XX,Y,IBDATES  K FILTERS  ; get date range  S IBDATES="Date of Service",IBDATES=$$FMDATES(IBDATES) I IBDATES=0 Q 0  S FILTERS(0)=IBDATES  ;  ; Site (Division) filter  S DIR(0)="S",DIR("A")="Select(A)ll or (S)elected Division(s):",DIR("B")="All"  S DIR("?",1)="Enter 'A' to not filter by Division."  S DIR("?")="Enter 'S' to view entries for selected Division(s)."  S $P(DIR(0),U,2)="A:All Divisions;S:Selected Divisions"  W ! D ^DIR K DIR  I $G(DIRUT) Q 0  S X=$$UP^XLFSTR(X)  S $P(FILTERS(0),U,3)=$S(Y="A":0,1:1)  ;  ; Set Division filter  I $P(FILTERS(0),U,3)=1 D  .D ASKDIV(.FILTERS)  .I  ;  ; Patient filter  S DIR(0)="S",DIR("A")="Select(A)ll or (S)elected Patient(s):",DIR("B")="All"  S DIR("?",1)="Enter 'A' to not filter by Patient."  S DIR("?")="Enter 'S' to view entries for selected Patients."  S $P(DIR(0),U,2)="A:All Patients;S:Selected Patients"  W ! D ^DIR K DIR  I $G(DIRUT) Q 0  S X=$$UP^XLFSTR(X)  S $P(FILTERS(0),U,4)=$S(Y="A":0,1:1)  ; Set Patient / Veteran filter  I $P(FILTERS(0),U,4)=1 D ASKPAT(.FILTERS)  ;  S DIR(0)="Y",DIR("A")="Include Insurance information on the Held Charges list",DIR("B")="NO"  S DIR("?",1)=" Enter: 'Y' - to include patient insurance information on the Held Charges list"  S DIR("?",2)=" 'N' - to exclude patient insurance information on the Held Charges list"  S DIR("?",3)=" '^' - to exit this option"  D ^DIR K DIR  I $G(DIRUT) Q 0  S FILTERS(3)=+Y  D SHOWFILT(.FILTERS)  I X="^" Q 0  Q 1  ;  FMDATES(PROMPT) ; ask for date range  N %DT,X,Y,DT1,DT2,IB0,IB1,IB2  S DIR(0)="S",DIR("A")="Select(A)ll or (S)elected Date(s):",DIR("B")="All"  S DIR("?",1)="Enter 'A' to view all Dates."  S DIR("?")="Enter 'S' to view entries for selected Dates."  S $P(DIR(0),U,2)="A:All Dates;S:Selected Dates"  W ! D ^DIR K DIR  I X="^" Q 0  I $G(DIRUT) Q 0  I $E(Y)="A" S DT1=0\_U\_DT G FMDQ  S DT1="",IB1="Start with date entered: ",IB2="Go to date entered: "  I $G(PROMPT)'="" S IB1="Start with "\_PROMPT\_": ",IB2="Go to "\_PROMPT\_": "  S %DT="AEX",%DT("A")=IB1 D ^%DT K %DT I Y<0!($P(Y,".",1)'?7N) G FMDQ  S (%DT(0),DT2)=$P(Y,".",1) I DT2'>DT S %DT("B")="Today"  S %DT="AEX",%DT("A")=IB2 D ^%DT K %DT I Y<0!($P(Y,".",1)'?7N) G FMDQ  S DT1=DT2\_U\_$P(Y,".",1)  FMDQ Q DT1  ;  ASKDIV(FILTERS) ; Sets a list of Divisions to be displayed in the Held Charges Worklist  ; Input: FILTERS - Current Array of filter settings  ; Output: FILTERS - Updated Array of filter settings  N DIC,DIR,DIRUT,DIVS,DUOUT,FIRST,IBIENS,IBIENS2,IEN,N,X,XX,Y  S DIC=4,DIC(0)="AEM",FIRST=1  F D Q:+IEN<1  . D ONEDIV(.DIC,.IEN,.FIRST) ; One Division prompt  . Q:+IEN<1  . S IBIENS($P(IEN,U,2))=$P(IEN,U,1)  . S IBIENS2($P(IEN,U,1))=$P(IEN,U,2)  I '$D(IBIENS) S $P(FILTERS(0),U,3)=0 Q  ;  ; Set the filter node responses in alphabetical order  S XX=""  F D Q:XX=""  . S XX=$O(IBIENS(XX))  . Q:XX=""  . S N=IBIENS(XX)  . S FILTERS(1,N)=""  Q  ;  ONEDIV(DIC,IEN,FIRST) ; Prompts the user for a Division  ; Input: DIC - Variable/Array of settings needed for ^DIC call  ; FIRST - Set to 1 initially and then 0 for subsequent calls  ; Output: FIRST - Set to 0  ; IEN - IEN of the selected Division  ; null of no selection was made  S DIC("A")=$S(FIRST:"Select a Division: ",1:"Select Another Division: ")  D ^DIC  S FIRST=0,IEN=Y\_U\_X  Q  ;  ASKPAT(FILTERS) ; Sets a list of patients  ; the HCSR Worklist  ; Input: FILTERS - Current Array of filter settings  ; Output: FILTERS - Updated Array of filter settings  N CLINS,DIC,DIR,DIRUT,DIVS,DUOUT,FIRST,IBIENS,IBIENS2,IEN,N,NM,NODE,WARDS,X,XX,Y  S DIC(0)="AEQMN",DIC="^DPT(",FIRST=1  F D Q:+IEN<1  . D ONEPAT(.DIC,.IEN,.FIRST) ; One patient  . Q:+IEN<1  . S IBIENS($P(IEN,U,2))=$P(IEN,U,1)  . S IBIENS2($P(IEN,U,1))=$P(IEN,U,2)  I '$D(IBIENS) S $P(FILTERS(0),U,4)=0 Q  ;  ; Set the filter node responses in alphabetical order  S XX=""  F D Q:XX=""  . S XX=$O(IBIENS(XX))  . Q:XX=""  . S N=IBIENS(XX)  . S FILTERS(2,N)=""  . ;S FILTERS(2)=$S($G(FILTERS(2))'="":FILTERS(2)\_U\_N,1:N)  Q  ;  ONEPAT(DIC,IEN,FIRST) ; Prompts the user for a clinic or ward  ; Input: DIC - Variable/Array of settings needed for ^DIC call  ; FIRST - Set to 1 initially and then 0 for subsequent calls  ; Output: FIRST - Set to 0  ; IEN - IEN of the selected Ward or clinic Entry  ; null of no selection was made  N DPTNOFZY S DPTNOFZY=1 ;Suppress PATIENT file fuzzy lookups  S DIC("A")=$S(FIRST:"Select Patient: ",1:"Select Another Patient: ")  D ^DIC  S FIRST=0,IEN=Y  S DFN=+Y  Q  ;  SHOWFILT(FILTERS) ;EP  ; Displays the currently selected filter selections for the  ; Held Charges ListManager display  ; Input: FILTERS() - Array of filter settings. See FILTERS for a detailed  ; explanation of the FILTERS array  ; Output: Current Filter settings are displayed  ;  N DIR,DIROUT,DIRUT,DTOUT,DUOUT,IEN,IX,LEN,XX  S STDT=$P(FILTERS(0),U)  W !!!,"Show From Date: ",$S(STDT=0:"First",1:$$FMTE^XLFDT(STDT,"2DZ"))  W !," Thru Date: ",$$FMTE^XLFDT($P(FILTERS(0),U,2),"2DZ")  W !,"Show All Divisions or Selected Divisions: "  W $S($P(FILTERS(0),U,3)=0:"All",1:"Selected")  ;  ; Division list (if any)  I ($P(FILTERS(0),U,3)=1) D  . S LINE="Divisions to Display: "  . S IEN=0,PFLG=0  . F S IEN=$O(FILTERS(1,IEN)) Q:IEN="" D  . . S XX=$$GET1^DIQ(4,IEN\_",",.01)  . . S LINE=LINE\_$S(LINE="Divisions to Display: ":"",1:", ")\_XX  . W !,$$WRAP(.LINE,.PFLG,80)  . F I=0:0 Q:'PFLG W !,?22,$$WRAP(.LINE,.PFLG,58)  ;  W !,"Show All Patients or Selected Patients: "  W $S($P(FILTERS(0),U,4)=0:"All",1:"Selected")  ; Patient Inclusion list (if any)  I ($P(FILTERS(0),U,4)=1) D  . S LINE="Patients to Display: "  . S IEN=0,PFLG=0  . F S IEN=$O(FILTERS(2,IEN)) Q:IEN="" D  . . S XX=$$GET1^DIQ(2,IEN\_",",.01)  . . S LINE=LINE\_$S(LINE="Patients to Display: ":"",1:", ")\_XX  . W !,$$WRAP(.LINE,.PFLG,80)  . F I=0:0 Q:'PFLG W !,?21,$$WRAP(.LINE,.PFLG,60)  ;  W !,"Include Insurance information on the Held Charges list? ",$S(FILTERS(3)=1:"Yes",1:"No")  K DIR  D PAUSE^VALM1  Q  ;  WRAP(STR,FLG,CL) ;  ; STR - STRING TO BE WRAPPED PASSED IN BE REFERENCE SO IT CONTAINS THE REMAING PORTION OF STRING  ; FLG - FLAG TO INDICATE WRAPPING NEEDS TO OCCUR  ; CL - COLUMN LENGTH  ;  ; NO WRAPPING REQUIRED  I $L(STR)'>CL S FLG=0 Q STR  S FLG=1  N A,B,C  ; POSITION AFTER COLUMN WIDTH BREAK IS A SPACE  I $E(STR,CL+1)=" " S B=$E(STR,1,CL),STR=$E(STR,CL+2,999) Q B  S A=$E(STR,1,CL)  ; NO SPACES WITHIN COLUMN WITH, JUST BREAK AT COLUMN WIDTH  I $L(A," ")=1 S STR=$E(STR,CL+1,999) Q A  ; BREAK ON LAST SEMICOLON PIECE WITHIN COLUMN WIDTH  S C=$L(A," ")  S B=$P(A," ",1,C-1)  S STR=$P(A," ",C)\_$E(STR,CL+1,999)  Q B |

| Routines | Activities | | | |
| --- | --- | --- | --- | --- |
| **Routine Name** | IBOHLS1 | | | |
| **Enhancement Category** | New | Modify | Delete | No Change |
| **RTM** | COH-3.1-01 - COH-3.1-15 | | | |
| **Related Options** | IB OUTPUT HELD CHARGES LM | | | |

| Related Routines | Routines “Called By” | Routines “Called” |
| --- | --- | --- |
|  | IBOHLS | C^%DTC, $$GET1^DIQ, ALL^IBCNS1, $$DAT1^IBOUTL, $$FILE^IBRXUTL, $$SUBFILE^IBRXUTL, DEM^VADPT, $$FMDIFF^XLFDT |

| Routines | Activities | | | | |
| --- | --- | --- | --- | --- | --- |
| **Data Dictionary (DD) References** | ^DGCR(399, ^DIC(36, ^IB(, ^IB(“AC”, ^IB(“AH”, ^IBA(355.32, ^IBA(362,4, ^IBE(350.1, ^IBE(355.31, ^IBT(356, ^TMP($J | | | | |
| **Related Protocols** | IB EXPAND HELD CHARGES  IB EXPORT TO FILE  IB HELD CHARGES ACCOUNT PROFILE  IB HELD CHARGES CLAIM TRACKING  IB HELD CHARGES PATIENT INSURANCE  IB HELD CHARGES TPJI  IB PRINT HELD CHARGES RPT  IB RELEASE HELD CHARGES | | | | |
| **Related Integration Control Registrations (ICRs)** |  | | | | |
| **Data Passing** | Input | Output Reference | Both | Global Reference | Local |
| **Input Attribute Name and Definition** | Name: NONE  Definition: NONE | | | | |
| **Output Attribute Name and Definition** | Name: NONE  Definition: NONE | | | | |

| Current Logic |
| --- |
| NONE |

| Modified Logic (Changes are highlighted) |
| --- |
| IBOHLS1 ;ALB/BAA - IB HELD CHARGES LIST MANAGER ; 08-SEP-2015  ;;2.0;INTEGRATED BILLING;\*\*554\*\*;21-MAR-94;Build 3  ;;Per VA Directive 6402, this routine should not be modified.  ;;  ;  SORT ; get the data  N CNT  S CNT=0  ; compile data to display here  I 'PATS D  . S IBN=0 F S IBN=$O(^IB("AC",8,IBN)) Q:'IBN D CHRGS(IBN,PATS)  I PATS D  . S DFN=0 F S DFN=$O(FILTERS(2,DFN)) Q:'DFN D  .. S IBN=0 F S IBN=$O(^IB("AH",DFN,IBN)) Q:'IBN D CHRGS(IBN,PATS)  Q  ;  CHRGS(IBN,PATS) ; charges on hold  N IBFR,IBTO,HDAYS,IBND,HINST,DFN,HST,IBACT,IBCHG,ID,SS,SSLE,SSLS,NAME  S IBND=$G(^IB(IBN,0)) Q:'IBND  S IBACT=+IBND  S DFN=$P(IBND,U,2)  D PAT  S HST=$P(IBND,U,5)  I HST'=8 Q  S IBFR=$P(IBND,U,14),IBTO=$P(IBND,U,15)  I IBFR<BDATE!(IBFR>EDATE) Q  S HINST=$P(IBND,U,13)  I INSTS,'$D(FILTERS(1,HINST)) Q  S HDAYS=$$FMDIFF^XLFDT(DT,IBFR,1)  S IBCHG=$P(IBND,U,7)  D BILLS  Q  ;  PAT ; patient name  N VAERR,VADM D DEM^VADPT I VAERR K VADM  S NAME=$G(VADM(1)) S:NAME="" NAME=" "  S SS=$P($G(VADM(2)),U,1),SSLE=$L(SS),SSLS=6 I $E(SS,SSLE)="P" S SSLS=5  S ID=$E(NAME,1)\_$E(SS,SSLS,SSLE)  Q  ;  BILLS ; find bills for charges on hold  N IBT,IBATYPE,IBCHRG,IBTP  S IBATYPE=$S($P($G(^IBE(350.1,+$P(IBND,U,3),0)),U)["OPT":"O",$P($G(^IBE(350.1,+IBND,U,3,0)),U)["PSO":"RX",1:"I")  S CNT=CNT+1  S IBTP=$P(IBND,"^",3),IBTP=$P($G(^IBE(350.1,IBTP,0)),"^",1),IBTP=$S(IBTP["PSO NSC":"RXNSC",IBTP["PSO SC":"RX SC",1:$E(IBTP,4,7))  S ^TMP($J,"IBOHLS",NAME,CNT)=NAME\_U\_ID\_U\_IBTP\_U\_IBFR\_U\_IBTO\_U\_HDAYS\_U\_IBCHG  S ^TMP($J,"IBOHLS",NAME,CNT,"IBND")=DFN\_U\_NAME\_U\_IBN\_U\_IBFR\_U\_IBTO  I IBATYPE="I" D INP  I IBATYPE="O" D OTP  E D RX  I IINS,$D(^TMP($J,"IBOHLS",NAME,CNT)),'$D(^TMP($J,"IBOHLS INS",NAME)) D GETINS  Q  ;  INP ; inpatient bills  N IBEV,IBBILL,IBT,X,X1,X2,IBEND,IBOK,IBBCHG,IBBILL0,IBBILLU1,BILL,BCNT,BLTRK,RNB,STATUS  N HLDDT,AUDT,IBTYPE  S IBTYPE=$P(IBND,"^",3),IBTYPE=$P($G(^IBE(350.1,IBTYPE,0)),"^",1),IBTYPE=$S(IBTYPE["PSO NSC":"RXNSC",IBTYPE["PSO SC":"RX SC",1:$E(IBTYPE,4,7))  S IBEV=$P(IBND,U,16) Q:'IBEV ; parent event  S IBEV=($P($G(^IB(IBEV,0)),U,17)\1) Q:'IBEV ; date of parent event  S X1=IBEV,X2=1 D C^%DTC S IBEND=X  S BCNT=0  S IBT=(IBEV-.0001) F S IBT=$O(^DGCR(399,"D",IBT)) Q:'IBT!(IBT'<IBEND) S IBBILL=0 F S IBBILL=$O(^DGCR(399,"D",IBT,IBBILL)) Q:IBBILL="" D  . S IBBILL0=$G(^DGCR(399,IBBILL,0))  . S BILL=$P(IBBILL0,U,1)  . S AUDT=$$GET1^DIQ(399,IBBILL,10,"I")  . S HLDDT=$S(AUDT'="":$$FMDIFF^XLFDT(DT,AUDT,1),1:HDAYS)  . S IBBCHG=$$GET1^DIQ(430,IBBILL,77)  . S STATUS=$$GET1^DIQ(430,IBBILL,8,"O")  . S (BLTRK,RNB)=""  . S BLTRK=$O(^IBT(356,"E",IBBILL,BLTRK))  . I BLTRK'="" S RNB=$$GET1^DIQ(356,BLTRK,.19,"O")  . D INPTCK  . I IBOK D  .. ;BILL#AR STATUS^DATE BILLED^AUTH DATE^HLD DAYS^CHARGE^RNB^BILL TRK #  .. S BCNT=BCNT+1  .. S ^TMP($J,"IBOHLS",NAME,CNT,2,BCNT)=BILL\_U\_STATUS\_U\_IBT\_U\_AUDT\_U\_HLDDT\_U\_IBBCHG\_U\_RNB\_U\_BLTRK ;BILL#^AR STATUS^DATE BILLED^CHARGE  .. S ^TMP($J,"IBOHLS",NAME,CNT,2,BCNT,"B")=IBBILL\_U\_BLTRK\_U\_RNB  Q  ;  INPTCK ; does bill belong to charge? returns IBOK=0 if no  N IBBILLU  S IBBILLU=$G(^DGCR(399,IBBILL,"U"))  S IBBILL=$P(IBBILL0,U,1)  S IBOK=1  CK1 ; for same patient?  I DFN=$P(IBBILL0,U,2)  S IBOK=$T  Q:'IBOK  CK2 ; same type- inp or opt?  N B S B=$S(+$P(IBBILL0,U,5)<3:"I",1:"O")  I B=IBATYPE S IBOK=1  S IBOK=$T  Q:'IBOK  CK3 ; overlap in date range?  N F,T  S F=+IBBILLU,T=$P(IBBILLU,U,2)  I (IBTO<F)!(IBFR>T)  S IBOK='$T  Q:'IBOK  CK4 ; insurance bill?  I $P(IBBILL0,U,11)="i"  S IBOK=$T  Q  ;  OTP ; outpatient bills  N X,IBV,IBBILL,IBOK,IBBILL0,IBBCHG,IBBILLU1,IBBILLU,BILL,BCNT  S BCNT=0  S IBV=(IBFR\1)-.0001 F S IBV=$O(^DGCR(399,"AOPV",DFN,IBV)) Q:'IBV!(IBV>IBTO) S IBBILL=0 D  .F S IBBILL=$O(^DGCR(399,"AOPV",DFN,IBV,IBBILL)) Q:('IBBILL) D  .. S IBBILL0=$G(^DGCR(399,IBBILL,0)),IBBILLU=$G(^DGCR(399,IBBILL,"U")) D CK4 Q:'IBOK  .. S BILL=$P(IBBILL0,U,1)  .. S AUDT=$$GET1^DIQ(399,IBBILL,10,"I")  .. S HLDDT=$S(AUDT'="":$$FMDIFF^XLFDT(DT,AUDT,1),1:HDAYS)  .. S IBBCHG=$$GET1^DIQ(430,IBBILL,77)  .. S STATUS=$$GET1^DIQ(430,IBBILL,8,"O")  .. S (BLTRK,RNB)=""  .. S BLTRK=$O(^IBT(356,"E",IBBILL,BLTRK))  .. I BLTRK'="" S RNB=$$GET1^DIQ(356,BLTRK,.19,"O")  .. S BILL=$P(IBBILL0,U,1),BCNT=BCNT+1  .. S IBBILLU1=$G(^DGCR(399,IBBILL,"U1")),IBBCHG=$P(IBBILLU1,U,1)  .. S ^TMP($J,"IBOHLS",NAME,CNT,2,BCNT)=BILL\_U\_STATUS\_U\_IBV\_U\_AUDT\_U\_HLDDT\_U\_IBBCHG\_U\_RNB\_U\_BLTRK ;BILL#^AR STATUS^DATE BILLED^AUTH DATE^DAYS ON HOLD^CHARGE^RNB^BILL TRK NO  .. S ^TMP($J,"IBOHLS",NAME,CNT,2,BCNT,"B")=IBBILL\_U\_BLTRK\_U\_RNB  Q  ;  RX ; rx refill billsget insurance information  N XX,IBINS,IBX,ICNT,INSCO,SUBID,PLNID,EFFDT,EFDTCK,EXPDT,EXDTCK,LEDT,SUBNAM  N PLNCOV,PEFDT,PCOVD,PCOM,PCNT,COVFN,GRP,CKDT,IBCNT,IBINS0,IBINS7,LIM,INSTYP  S (PLNCOV,PEFDT,PCOVD,PCOM)=""  S ICNT=0  D ALL^IBCNS1(DFN,"IBINS")  S XX=0  F S XX=$O(IBINS(XX)) Q:'XX D  . S IBINS0=IBINS(XX,0)  . S IBINS7=$G(IBINS(XX,7))  . S PLNID=$P(IBINS0,U,18),GRP=$P(IBINS0,U,3)  . I $P($G(^IBA(355.3,PLNID,0)),"^",11) Q ;plan is inactive  . S INSCO=$P(^DIC(36,+IBINS0,0),U,1),REIMB=$P(INSCO,U,2)  . I $P(INSCO,U,5) Q ;insurance company inactive  . S SUBID=$P(IBINS7,U,2)  . S SUBNAM=$P(IBINS7,U,1)  . S EXDTCK=+$P(IBINS0,U,4)  . S EFDTCK=+$P(IBINS0,U,8)  . I EXDTCK,EXDTCK<IBFR Q ; if insurance expired before the from date of copay quit  . I EFDTCK,EFDTCK>IBTO Q ; if insurance not in effect for period quit  . S EFFDT=$$DAT1^IBOUTL(EFDTCK)  . S EXPDT=$$DAT1^IBOUTL(EXDTCK)  . S ICNT=ICNT+1  . ;ins co^sub id^plan id^effective dt^expiration date  . S ^TMP($J,"IBOHLS",NAME,CNT,3,ICNT)=IBINS0\_U\_PLNID  . S ^TMP($J,"IBOHLS INS",NAME,ICNT)=INSCO\_U\_SUBNAM\_U\_GRP\_U\_EFFDT\_U\_EXPDT  . ;plan coverage^effective date^covered?^limit comments  . S LIM=0,PCNT=0  . F S LIM=$O(^IBE(355.31,LIM)) Q:'LIM D  .. S PLNCOV=$P($G(^IBE(355.31,LIM,0)),U),IBCNT=0,PEFDT=""  .. S PCOVD="",LEDT="",PCOM=""  .. F S LEDT=$O(^IBA(355.32,"APCD",PLNID,LIM,LEDT)) Q:$S(LEDT="":IBCNT,1:0) D Q:LEDT=""  ... S COVFN=+$O(^IBA(355.32,"APCD",PLNID,LIM,+LEDT,"")),PCOVD=$G(^IBA(355.32,+COVFN,0))  ... S PEFDT=$$DAT1^IBOUTL($P(LEDT,"-",2))  ... I PCOVD="" S PCOVD="BY DEFAULT" D SETINS(PLNCOV,PEFDT,PCOVD,PCOM,IBCNT) Q  ... S IBCNT=IBCNT+1,PCOM=""  ... I PCOVD'="" S CVD=$P(PCOVD,U,4),PCOVD=$S(CVD:$S(CVD<2:"YES",CVD=2:"CONDITIONAL",1:"UNKNOWN"),1:"NO")  ... I '$O(^IBA(355.32,COVFN,2,0)) D SETINS(PLNCOV,PEFDT,PCOVD,PCOM,IBCNT) Q  ... S (IBS,IB0)=0 F S IB0=$O(^IBA(355.32,COVFN,2,IB0)) Q:'IB0 D  .... S PCOM=""  .... S PCOM=^IBA(355.32,COVFN,2,IB0,0)  .... I IBS=0 D SETINS(PLNCOV,PEFDT,PCOVD,PCOM,IBCNT)  .... I IBS>0 D SETCOV(PCOM)  .... S IBS=IBS+1  Q  ;  SETINS(PLNCOV,PEFDT,PCOVD,PCOM,IBCNT) ; SET GLOBAL ENTRY  S PCNT=PCNT+1  I IBCNT>1 S PLNCOV=""  S ^TMP($J,"IBOHLS",NAME,CNT,3,ICNT,PCNT)=""  S ^TMP($J,"IBOHLS INS",NAME,ICNT,0)=IBINS0\_U\_PLNID  S ^TMP($J,"IBOHLS INS",NAME,ICNT,PCNT)=PLNCOV\_U\_PEFDT\_U\_PCOVD\_U\_PCOM  Q  ;  SETCOV(PCOM) ; SET COVERAGE WHEN MULTIPLE  S PCNT=PCNT+1  S ^TMP($J,"IBOHLS INS",NAME,ICNT,PCNT)=""\_U\_""\_U\_""\_U\_PCOM  Q |

| Routines | Activities | | | |
| --- | --- | --- | --- | --- |
| **Routine Name** | IBOHLS2 | | | |
| **Enhancement Category** | New | Modify | Delete | No Change |
| **RTM** | COH-3.1-01 - COH-3.1-15 | | | |
| **Related Options** | IB OUTPUT HELD CHARGES LM | | | |

| Related Routines | Routines “Called By” | Routines “Called” |
| --- | --- | --- |
|  |  | ^%ZIS, HOME^%ZIS, ^%ZISC, ^%ZTLOAD, $$S^%ZTLOAD, ^DIR,  BOD^IBOHLS, REL^IBOHRL, EN^VALM, CLEAR^VALM1, FULL^VALM1, PAUSE^VALM1, EN^VALM2, $$FMTE^XLFDT, $$NOW^XLFDT |

| Routines | Activities | | | | |
| --- | --- | --- | --- | --- | --- |
| **Data Dictionary (DD) References** | ^TMP($J | | | | |
| **Related Protocols** | IB EXPAND HELD CHARGES  IB EXPORT TO FILE  IB HELD CHARGES ACCOUNT PROFILE  IB HELD CHARGES CLAIM TRACKING  IB HELD CHARGES PATIENT INSURANCE  IB HELD CHARGES TPJI  IB PRINT HELD CHARGES RPT  IB RELEASE HELD CHARGES | | | | |
| **Related Integration Control Registrations (ICRs)** |  | | | | |
| **Data Passing** | Input | Output Reference | Both | Global Reference | Local |
| **Input Attribute Name and Definition** | Name: NONE  Definition: NONE | | | | |
| **Output Attribute Name and Definition** | Name: NONE  Definition: NONE | | | | |

| Current Logic |
| --- |
| NEW ROUTINE |

| Modified Logic (Changes are highlighted) |
| --- |
| IBOHLS2 ;ALB/BAA - IB HELD CHARGES LIST MANAGER ; 08-SEP-2015  ;;2.0;INTEGRATED BILLING;\*\*554\*\*;21-MAR-94;Build 3  ;;Per VA Directive 6402, this routine should not be modified.  ;;  ;  REL ; release selected copay charges  D FULL^VALM1  N I,J,IBXX,VALMY,IBND,DATA,NAME,CNT,DFN,IBCHRGS,RELCPY  S RELCPY=""  ;  ;  D EN^VALM2($G(XQORNOD(0)))  I $D(VALMY) S IBXX=0 F S IBXX=$O(VALMY(IBXX)) Q:'IBXX D  . S DATA=$G(^TMP($J,"IBOHLSX",IBXX))  . S DFN=$P(DATA,U,1)  . S NAME=$P(DATA,U,2)  . S CNT=$P(DATA,U,3)  . S DATA=^TMP($J,"IBOHLS",NAME,CNT,"IBND")  . S IBND=$P(DATA,U,3)  . S IBCHRGS=^TMP($J,"IBOHLS",NAME,CNT)  . Q:IBND="" D RELHLD(DFN,IBND,IBCHRGS)  . I RELCPY=1 K ^TMP($J,"IBOHLS",NAME,CNT)  ;  D BLD^IBOHLS  S VALMBCK="R"  Q  ;  RELHLD(DFN,IBN,IB0) ; queue copay for release  K IBR60  K ^TMP($J,"IBHOLD")  I '$$KCHK^XUSRB("IB AUTHORIZE") D G RELHLDQ  . W !?5,"The necessary key is IB AUTHORIZE. Please see your manager." ;  . D PAUSE^VALM1  ;  W !," Copay for "\_$P(IB0,U,1)\_" - "\_$P(IB0,U,2)\_" for the amount of $"\_$P(IB0,U,7)\_" will be released."  ;  S DIR(0)="Y",DIR("A")="Are you sure you want to Release this Copay",DIR("B")="NO"  S DIR("?",1)=" Enter: 'Y' - to Release the Copay"  S DIR("?",2)=" 'N' - to NOT Relese the Copay"  S DIR("?",3)=" '^' - to exit this option"  D ^DIR K DIR  I Y'=1 D Q  . S RELCPY=0  . W !," Release of Copay for "\_$P(IB0,U,1)\_" - "\_$P(IB0,U,2)\_" canceled."  . D PAUSE^VALM1  ;  S ^TMP($J,"IBHOLD",DFN,IBN)=""  ;  D REL^IBOHRL ; Release charges  ;  W !," Copay for "\_$P(IB0,U,1)\_" - "\_$P(IB0,U,2)\_" for the amount of $"\_$P(IB0,U,7)\_" has been queued for released."  S RELCPY=1  D PAUSE^VALM1  K ^TMP($J,"IBHOLD")  RELHLDQ Q ;  ;  RPT(RTN,FILTERS) ; print the information  N BDATE,EDATE  S BDATE=$P(FILTERS(0),U,1),EDATE=$P(FILTERS(0),U,2)  D DEVICE("PR")  D PAUSE^VALM1  D BLD^IBOHLS  S VALMBCK="R" Q  Q  ;  DEVICE(TYPE) ; Ask user to select device  ;  N %ZIS,CRT,MAXCNT,POP  S %ZIS="QM" D ^%ZIS G:POP ENQ  I $D(IO("Q")) D G ENQ  .S ZTDESC="VistA Held Charges Report"  .I TYPE="PR" S ZTRTN="PRINT^IBOHLS2"  .I TYPE="EF" S ZTRTN="EXCEL^IBOHLS2"  .F I="^TMP($J,""IBHOLS"",","FILTERS" S ZTSAVE(I)=""  .D ^%ZTLOAD K IO("Q") D HOME^%ZIS  .W !!,$S($D(ZTSK):"This job has been queued as task #"\_ZTSK\_".",1:"Unable to queue this job.")  .K ZTSK,IO("Q")  ;  ; print report  I IOST["C-" S MAXCNT=IOSL-3,CRT=1  E S MAXCNT=IOSL,CRT=0  ;  I TYPE="PR" U IO D PRINT("IBOHLS",BDATE,EDATE,MAXCNT)  I TYPE="EF" U IO D EXCEL("IBOHLS",BDATE,EDATE,MAXCNT)  ;  D ^%ZISC  ;  I $D(ZTQUEUED) S ZTREQ="@"  K OUT  ;  ENQ Q  ;  PRINT(RTN,BDATE,EDATE,MAX) ; -- print the current data  N REC,CNT,LCNT,RX,IBQUIT,FIRST,XX,NAME,LINE,ZZ,ZZ1,ZZ2,PGC  S LCNT=0,PGC=0,IBQUIT=0  D CLEAR^VALM1  U IO  D HEADER  S NAME="" F S NAME=$O(^TMP($J,"IBOHLS",NAME)) Q:NAME="" D  . S FIRST=1  . S CNT=0 F S CNT=$O(^TMP($J,"IBOHLS",NAME,CNT)) Q:CNT="" D  .. D:$Y>MAX HEADER Q:IBQUIT  .. S LINE=$$SETL("","","",1,2) ;line#  .. ;PATNAME^PATID^TYPE^FROM/FILL DATE^TO/RLS DATE^#DAYS ON HOLD^CHARGE  .. S REC=^TMP($J,"IBOHLS",NAME,CNT)  .. S LINE=$$SETL(LINE,$P(REC,U),"",4,22)  .. S LINE=$$SETL(LINE,$P(REC,U,2),"",26,6)  .. S LINE=$$SETL(LINE,$P(REC,U,3),"",36,6)  .. S LINE=$$SETL(LINE,$$FMTE^XLFDT($P(REC,U,4),"2DZ"),"",46,8)  .. S LINE=$$SETL(LINE,$$FMTE^XLFDT($P(REC,U,5),"2DZ"),"",56,8)  .. S LINE=$$SETL(LINE,$P(REC,U,6),"",63,5)  .. S LINE=$$SETL(LINE,$J($P(REC,U,7),8,2),"",71,8)  .. S LCNT=LCNT+1  .. S OUT(LCNT)=LINE  .. I $D(^TMP($J,"IBOHLS",NAME,CNT,1)) S RX=^(1),RX="Rx#:"\_RX D  ... ;RX VALUE  ... S LINE=$$SETL("",RX,"",37,20)  ... S LCNT=LCNT+1  ... S OUT(LCNT)=LINE  .. I $D(^TMP($J,"IBOHLS",NAME,CNT,2)) D  ... ;BILL#AR STATUS^DATE BILLED^CHARGE  ... S BCNT=0 F S BCNT=$O(^TMP($J,"IBOHLS",NAME,CNT,2,BCNT)) Q:BCNT="" D  .... S REC=^TMP($J,"IBOHLS",NAME,CNT,2,BCNT)  .... S LINE=$$SETL("","Bill: ","",6,18)  .... S LINE=$$SETL(LINE,$P(REC,U),"",15,10)  .... S LINE=$$SETL(LINE,$P(REC,U,2),"",26,10)  .... S LINE=$$SETL(LINE,$$FMTE^XLFDT($P(REC,U,3),"2DZ"),"",38,8)  .... S LINE=$$SETL(LINE,$J($P(REC,U,4),8,2),"",48,11)  .... S LCNT=LCNT+1  .... S OUT(LCNT)=LINE  .... S RNB=$P(REC,U,7)  .... I RNB'="" D  ..... S LINE=$$SETL("","RNB: ","",6,6)  ..... S LINE=$$SETL(LINE,RNB,"",14,60)  .. I $D(^TMP($J,"IBOHLS",NAME,CNT,3)),FIRST D ; IF DISPLAYING INSURANCE INFORMATION  ... N ZZ,ZZ1,ZZ2  ... ;ins co^sub id^plan id^effective dt^expiration  ... S FIRST=0  ... S LINE=$$SETL("","Insurance","",8,9)  ... S LINE=$$SETL(LINE,"Subscriber","",28,10)  ... S LINE=$$SETL(LINE,"Group","",44,5)  ... S LINE=$$SETL(LINE,"Eff Dt","",54,6)  ... S LINE=$$SETL(LINE,"Exp Dt","",66,6)  ... S LCNT=LCNT+1  ... S OUT(LCNT)=LINE  ... S LCNT=LCNT+1  ... S $P(ZZ2,"-",68)=""  ... S LINE=$$SETL("",ZZ2,"",8,68)  ... S OUT(LCNT)=LINE  ... S ZZ=0 F S ZZ=$O(^TMP($J,"IBOHLS INS",NAME,ZZ)) Q:ZZ="" D  .... ;plan coverage^effective date^covered?^limit  .... S ZZ1=^TMP($J,"IBOHLS INS",NAME,ZZ)  .... S LINE=$$SETL("",$P(ZZ1,U),"",8,15)  .... S LINE=$$SETL(LINE,$P(ZZ1,U,2),"",28,10)  .... S LINE=$$SETL(LINE,$P(ZZ1,U,3),"",44,6)  .... S LINE=$$SETL(LINE,$$FMTE^XLFDT($P(ZZ1,U,4),"2DZ"),"",54,8)  .... S LINE=$$SETL(LINE,$$FMTE^XLFDT($P(ZZ1,U,5),"2DZ"),"",66,8)  .... S LCNT=LCNT+1  .... S OUT(LCNT)=LINE  .... S LINE=$$SETL("","Plan Coverage Effective Date Covered? Limit Comments","",10,60)  .... S LCNT=LCNT+1  .... S OUT(LCNT)=LINE  .... S ZZ2=0 F S ZZ2=$O(^TMP($J,"IBOHLS INS",NAME,ZZ,ZZ2)) Q:ZZ2="" D  ..... S ZZ1=^TMP($J,"IBOHLS INS",NAME,ZZ,ZZ2)  ..... S LINE=$$SETL("",$P(ZZ1,U),"",10,15)  ..... S LINE=$$SETL(LINE,$$FMTE^XLFDT($P(ZZ2,U,2),"2DZ"),"",28,8)  ..... S LINE=$$SETL(LINE,$P(ZZ1,U,3),"",46,10)  ..... S LINE=$$SETL(LINE,$P(ZZ1,U,4),"",59,20)  ..... S LCNT=LCNT+1  ..... S OUT(LCNT)=LINE  ;  S XX=0  F S XX=$O(OUT(XX)) Q:XX="" D:$Y>MAX HEADER Q:IBQUIT W !,OUT(XX)  W !!,?5,"END OF REPORT"  Q  ;  HEADER ; -- print header  N DIR,X,Y,DTOUT,DUOUT,OFFSET,HDR,DASHES,DASHES2,LIN,IBPXT  S IBPXT=0  ;  I CRT,PGC>0,'$D(ZTQUEUED) D I IBPXT G HEADERX  . I MAX<51 F LIN=1:1:(MAX-$Y) W !  . S DIR(0)="E" D ^DIR K DIR  . I $D(DTOUT)!$D(DUOUT) S IBPXT=1 Q  I $D(ZTQUEUED),$$S^%ZTLOAD() S (ZTSTOP,IBPXT)=1 G HEADERX  S PGC=PGC+1  W @IOF,!,?1,"VistA Held Charges Report"  S HDR=$$FMTE^XLFDT($$NOW^XLFDT,1)\_" Page: "\_PGC  S OFFSET=80-$L(HDR)  W ?OFFSET,HDR  S HDR=$$FMTE^XLFDT(BDATE,"5Z")\_" - "\_$$FMTE^XLFDT(EDATE,"5Z")  S OFFSET=80-$L(HDR)\2  W !,?OFFSET,HDR  W !,?2,"Patient Name ID Type Fr/Fl Dt To/Rls Dt Days Amount"  W !,?2,"-------------------------------------------------------------------------------"  HEADERX ; EXIT  Q  ;  EXPORT(RTN,FILTERS) ; -- print excel spreadsheet.  N REC,CNT,RX,IBQUIT,BDATE,EDATE,NAME  S LCNT=0,PGC=0,IBQUIT=0  S BDATE=$P(FILTERS(0),U,1),EDATE=$P(FILTERS(0),U,2)  D ^%ZISC  D DEVICE("EF")  ;  D BLD^IBOHLS  D PAUSE  S VALMBCK="R"  Q  ;  EXCEL(RTN,BDATE,EDATE,MAX) ; print the data in excel format  D CLEAR^VALM1  U IO  N LINE,LCNT,PCE,REC,OUT,NAME,XX,BCNT,CNT,NXT,ZZ,ZZ1,ZZ2,OUT  D EXHDR  S LCNT=0  S NAME="" F S NAME=$O(^TMP($J,"IBOHLS",NAME)) Q:NAME="" D COUNT  ;  S XX=0  F S XX=$O(OUT(XX)) Q:XX="" W !,OUT(XX)  ;  W !,"END OF REPORT"  D PAUSE  Q  ;  COUNT ; format output  S FIRST=1,CNT=0  F S CNT=$O(^TMP($J,"IBOHLS",NAME,CNT)) Q:CNT="" D  . S LCNT=LCNT+1  . S REC=^TMP($J,"IBOHLS",NAME,CNT)  . ;PATNAME^PATID^TYPE^Fr/Fl D^To/Rls^#Days On Hold^CHARGE $ Fr/Fl Dt and To/Rls  . S $P(REC,U,4)=$$FMTE^XLFDT($P(REC,U,4),"2DZ")  . S $P(REC,U,5)=$$FMTE^XLFDT($P(REC,U,5),"2DZ")  . S OUT(LCNT)=REC  . ;RX VALUE  . I $D(^TMP($J,"IBOHLS",NAME,CNT,1)) S $P(OUT(LCNT),U,8)=^TMP($J,"IBOHLS",NAME,CNT,1)  . I $D(^TMP($J,"IBOHLS",NAME,CNT,2)) D  .. S LINE="",XX=0  .. S BCNT=0 F S BCNT=$O(^TMP($J,"IBOHLS",NAME,CNT,2,BCNT)) Q:BCNT="" D  ... ;BILL#^AR STATUS^DATE BILLED^CHARGE  ... S REC=^TMP($J,"IBOHLS",NAME,CNT,2,BCNT)  ... S $P(REC,U,3)=$$FMTE^XLFDT($P(REC,U,3),"2DZ")  ... S REC=$TR(REC,U,"|")  ... S XX=XX+1  ... S $P(LINE,";",XX)=REC  .. S $P(OUT(LCNT),U,9)=LINE  . I $D(^TMP($J,"IBOHLS INS",NAME)),FIRST D ; IF DISPLAYING INSURANCE INFORMATION  .. S FIRST=0  .. S ZZ=0 F S ZZ=$O(^TMP($J,"IBOHLS INS",NAME,ZZ)) Q:ZZ="" D  ... ;ins co^sub id^plan id^effective dt^expiration  ... S ZZ1=^TMP($J,"IBOHLS INS",NAME,ZZ)  ... S $P(ZZ1,U,4)=$$FMTE^XLFDT($P(ZZ1,U,4),"2DZ")  ... S $P(ZZ1,U,5)=$$FMTE^XLFDT($P(ZZ1,U,5),"2DZ")  ... S ZZ1=$TR(ZZ1,U,"|")  ... S $P(OUT(LCNT),U,10)=ZZ1  ... S PCE=10,XX=0  ... S ZZ2=0 F S ZZ2=$O(^TMP($J,"IBOHLS INS",NAME,ZZ,ZZ2)) Q:ZZ2="" D  .... ;plan coverage^effective date^covered?^limit  .... S REC=^TMP($J,"IBOHLS INS",NAME,ZZ,ZZ2)  .... S $P(REC,U,2)=$$FMTE^XLFDT($P(REC,U,2),"2DZ")  .... S REC=$TR(REC,U,"|"),XX=XX+1  .... S $P(OUT(LCNT),U,PCE+XX)=REC  ... S NXT=$O(^TMP($J,"IBOHLS INS",NAME,ZZ)) I NXT'="" S LCNT=LCNT+1 S OUT(LCNT)=$P(OUT(LCNT-1),U,1,9)  Q  ;  EXHDR ; -- excel header  S HDR="Patient Name"\_U\_"ID"\_U\_"Type"\_U\_"Fr/Fl Dt"\_U\_"To/Rls Dt"\_U\_"Days"\_U\_"Amount"\_U\_"RX"\_U\_"BILL"\_U\_"Insurance"\_U\_"Coverage"  W !,HDR  Q  ;  PAUSE ;pause at end of screen if beeing displayed on a terminal  Q:$E(IOST,1,2)'["C-" N DIR,DUOUT,DTOUT,DIRUT W !  S DIR(0)="E" D ^DIR K DIR  I $D(DUOUT)!($D(DIRUT)) S IBQUIT=1  Q  ;  PATINS ; view patient insurance  D FULL^VALM1  N I,J,IBXX,VALMY,ECNT,DFN,GOPAT  D EN^VALM2($G(XQORNOD(0)))  I $D(VALMY) S IBXX=0 F S IBXX=$O(VALMY(IBXX)) Q:'IBXX D  . S REC=$G(^TMP($J,"IBOHLSX",IBXX))  . S ^TMP($J,"PATINS")=$P(REC,U,1),GOTPAT=1  . D EN^VALM("IBCNS INSURANCE MANAGEMENT")  D BLD^IBOHLS  S VALMBCK="R"  Q  ;  CLMTRK ; look at claims tracking  D FULL^VALM1  N I,J,IBXX,VALMY,ECNT,NAME,GOTPAT,RC,IBFR,IBTO  D EN^VALM2($G(XQORNOD(0)))  K ^TMP($J,"IBOHLS CLMTRK")  I $D(VALMY) S IBXX=0 F S IBXX=$O(VALMY(IBXX)) Q:'IBXX D  . S RC=$G(^TMP($J,"IBOHLSX",IBXX))  . S DFN=$P(RC,U,1),NAME=$P(RC,U,2),ECNT=$P(RC,U,3),GOTPAT=1  . S RC=^TMP($J,"IBOHLSF")  . S IBFR=$P(RC,U,1),IBTO=$P(RC,U,2)  . S ^TMP($J,"IBCLMTRK")=DFN\_U\_IBFR\_U\_IBTO  .D EN^VALM("IBT CLAIMS TRACKING EDITOR")  K ^TMP($J,"IBOHLS CLMTRK")  D BLD^IBOHLS  S VALMBCK="R"  Q  ;  PATCLM ; look at claims INFO  D FULL^VALM1  N IBXX,VALMY,ECNT,PNAME,RC,XX,IBIFN  D EN^VALM2($G(XQORNOD(0)))  I $D(VALMY) S IBXX=0 F S IBXX=$O(VALMY(IBXX)) Q:'IBXX D  . S RC=$G(^TMP($J,"IBOHLSX",IBXX))  . S DFN=$P(RC,U,1),NAME=$P(RC,U,2),ECNT=$P(RC,U,3)  . S XX=0 F S XX=$O(^TMP($J,"IBOHLS",NAME,ECNT,2,XX)) Q:XX="" D  .. S RC=^TMP($J,"IBOHLS",NAME,ECNT,2,XX,"B"),IBIFN=$P(RC,U,1)  .. D EN^VALM("IBJT CLAIM INFO")  D BLD^IBOHLS  S VALMBCK="R"  Q  ;  PATACP ; look at ACCOUNT PROFILE  D FULL^VALM1  N IBXX,VALMY,ECNT,NAME,RC,DFN,CPY  D EN^VALM2($G(XQORNOD(0)))  D CLEAR^VALM1  I $D(VALMY) S IBXX=0 F S IBXX=$O(VALMY(IBXX)) Q:'IBXX D  . S RC=$G(^TMP($J,"IBOHLSX",IBXX))  . S DFN=$P(RC,U,1),NAME=$P(RC,U,2),ECNT=$P(RC,U,3)  . N DIC,X,Y,DEBT,PRCADB,DA,PRCA,COUNT,OUT,SEL,BILL,BAT,TRAN,DR,DXS,DTOUT,DIROUT,DIRUT,DUOUT  . N DPTNOFZY,DPTNOFZK S (DPTNOFZY,DPTNOFZK)=1  . S COUNT=0,CPY=1  . S PRCATY="ALL",X=NAME  . S X=$$UPPER^VALM1(X)  . S Y=$S($O(^PRCA(430,"B",X,0)):$O(^(0)),$O(^PRCA(430,"D",X,0)):$O(^(0)),1:-1)  . I Y>0 S DEBT=$P($G(^PRCA(430,Y,0)),"^",9) I DEBT S PRCADB=$P($G(^RCD(340,DEBT,0)),"^"),^DISV(DUZ,"^PRCA(430,")=Y,$P(DEBT,"^",2)=$$NAM^RCFN01(DEBT) D COMP^PRCAAPR,EN1^PRCAATR(Y) Q  . S DIC="^RCD(340,",DIC(0)="E" D ^DIC  . I Y<0 W !,"No entries found for "\_NAME Q  . S ^DISV(DUZ,"^RCD(340,")=+Y,PRCADB=$P(Y,"^",2),DEBT=+Y\_"^"\_$P(@("^"\_$P(PRCADB,";",2)\_+PRCADB\_",0)"),"^")  . D COMP^PRCAAPR,HDR^PRCAAPR1,HDR2^PRCAAPR1,DIS^PRCAAPR1  . D PAUSE^VALM1  D BLD^IBOHLS  S VALMBCK="R"  Q  ;  SETL(LINE,DATA,LABEL,COL,LNG) ; Creates a line of data to be set into the body  ; of the worklist  ; Input: LINE - Current line being created  ; DATA - Information to be added to the end of the current line  ; LABEL - Label to describe the information being added  ; COL - Column position in line to add information add  ; LNG - Maximum length of data information to include on the line  ; Returns: Line updated with added information  S LINE=LINE\_$J("",(COL-$L(LABEL)-$L(LINE)))\_LABEL\_$E(DATA,1,LNG)  Q LINE |

| Routines | Activities | | | |
| --- | --- | --- | --- | --- |
| **Routine Name** | IBOHLSE | | | |
| **Enhancement Category** | New | Modify | Delete | No Change |
| **RTM** | COH-3.1-01 - COH-3.1-15 | | | |
| **Related Options** | IB OUTPUT HELD CHARGES LM | | | |

| Related Routines | Routines “Called By” | Routines “Called” |
| --- | --- | --- |
|  |  | ^%ZISC, $$GET1^DIQ, EN^VALM, FULL^VALM1, SET^VALM10, $$FMTE^XLFDT, DISP^XQORM1 |

| Routines | Activities | | | | |
| --- | --- | --- | --- | --- | --- |
| **Data Dictionary (DD) References** | ^IB(  ^TMP($J | | | | |
| **Related Protocols** | IB EXPAND HELD CHARGES  IB EXPORT TO FILE  IB HELD CHARGES ACCOUNT PROFILE  IB HELD CHARGES CLAIM TRACKING  IB HELD CHARGES PATIENT INSURANCE  IB HELD CHARGES TPJI  IB PRINT HELD CHARGES RPT  IB RELEASE HELD CHARGES | | | | |
| **Related Integration Control Registrations (ICRs)** |  | | | | |
| **Data Passing** | Input | Output Reference | Both | Global Reference | Local |
| **Input Attribute Name and Definition** | Name: NONE  Definition: NONE | | | | |
| **Output Attribute Name and Definition** | Name: NONE  Definition: NONE | | | | |

| Current Logic |
| --- |
| NONE |

| Modified Logic (Changes are highlighted) |
| --- |
| IBOHLSE ;ALB/BAA - IB HELD CHARGES LIST MANAGER ; 08-SEP-2015  ;;2.0;INTEGRATED BILLING;\*\*554\*\*;21-MAR-94;Build 3  ;;Per VA Directive 6402, this routine should not be modified.  ;;  ;  EN ; -- main entry point for HELD CHARGES EXPAND  ; add code to do filters here  ;  D EN^VALM("IB HELD CHARGES EXPAND")  Q  ;  HDR ; -- header code  ;  S VALM("TITLE")=" Expanded Held Charges"  S XQORM("B")="QUIT"  Q  ;  INIT ; -- init variables and list array  ; input - none  ; output ^TMP($J,"IBOHLSE")  N DFN,CNT,NAME,IBIEN  S REC=^TMP($J,"IBOHLSE")  S DFN=$P(REC,U,1),CNT=$P(REC,U,2)  S NAME=$P(REC,U,3),IBIEN=$P(REC,U,4)  D BLD  Q  ;  BLD ; build data to display  N CNT,SS,LINE,XX,RX,GMT,IB0,IB1,PATIEN,XX  S (CNT,VALMCNT)=0  S IB0=^IB(IBIEN,0)  S IB1=^IB(IBIEN,1)  S SS=$$GET1^DIQ(2,DFN,.09)  ;  D FULL^VALM1  D SET^VALM10(1,"PATIENT : "\_NAME\_" - "\_SS,"")  ;  D SET^VALM10(2,"TYPE : "\_$$GET1^DIQ(350.1,$P(IB0,U,3),.01),"")  ;  D SET^VALM10(3,"RESULTING FROM : "\_$P(IB0,U,4),"")  ;  S LINE="",LINE=$$SETL(LINE,$$GET1^DIQ(350.21,$P(IB0,U,5),.01),"STATUS : ",1,38)  S LINE=$$SETL(LINE,$$FMTE^XLFDT($P(IB0,U,17),"2DZ"),"EVENT DATE : ",40,38)  D SET^VALM10(4,LINE,"")  ;  S LINE="",LINE=$$SETL(LINE,$P(IB0,U,9),"PARENT : ",1,38)  S LINE=$$SETL(LINE,$P(IB0,U,7),"CHARGE : ",40,38)  D SET^VALM10(5,LINE,"")  ;  D SET^VALM10(6,"INSTITUTION : "\_$$GET1^DIQ(4,$P(IB0,U,13),.01),"")  ;  D SET^VALM10(7,"BILLED : "\_$$FMTE^XLFDT($P(IB0,U,14),"2DZ")\_" - "\_$$FMTE^XLFDT($P(IB0,U,15),"2DZ"),"")  ;  S LINE="",LINE=$$SETL(LINE,$$GET1^DIQ(4,$P(IB0,U,13),.01),"AR BILL # : ",1,38)  S LINE=$$SETL(LINE,$$FMTE^XLFDT($P(IB0,U,18),"2DZ"),"LAST BILLED : ",40,38)  D SET^VALM10(8,LINE,"")  ;  D SET^VALM10(9,"IB COPAY TRANS # : "\_$$GET1^DIQ(354.71,$P(IB0,U,19),.01),"")  D SET^VALM10(10,"CLINIC STOP : "\_$$GET1^DIQ(352.5,$P(IB0,U,20),.01),"")  ;  S GMT=$P(IB0,U,21)  D SET^VALM10(11,"GMT RELATED : "\_$S(GMT=1:"Yes",1:"No"),"")  D SET^VALM10(12,"PFSS ACCT REF : "\_$$GET1^DIQ(375,$P(IB0,U,22),.01),"")  ;  S EAWHO=$$GET1^DIQ(200,$P(IB1,U,1),.01)  S EADAT=$$FMTE^XLFDT($P(IB1,U,2),"2DZ")  D SET^VALM10(13,"ENTRY ADDED : "\_EAWHO\_" "\_EADAT,"")  ;  S LUWHO=$$GET1^DIQ(200,$P(IB1,U,3),.01)  S LUDAT=$$FMTE^XLFDT($P(IB1,U,4),"2DZ")  D SET^VALM10(14,"LAST UPDATED : "\_LUWHO\_" "\_LUDAT,"")  ;  D SET^VALM10(15,"CHAMPVA ADM DATE : "\_$$FMTE^XLFDT($P(IB1,U,5),"2DZ"),"")  D SET^VALM10(16,"ON HOLD DATE : "\_$$FMTE^XLFDT($P(IB1,U,6),"2DZ"),"")  D SET^VALM10(17,"HOLD-REVIEW DATE : "\_$$FMTE^XLFDT($P(IB1,U,7),"2DZ"),"")  S VALMBCK="R",VALMBG=1,VALMCNT=16  Q  ;  HELP ; -- help code  N X  S X="?" D DISP^XQORM1 W !!  Q  ;  EXIT ; -- exit code  ;K ^TMP($J)  D ^%ZISC  I  S VALMBCK="R" Q  Q  ;  SETL(LINE,DATA,LABEL,COL,LNG) ; Creates a line of data to be set into the body  ; of the worklist  ; Input: LINE - Current line being created  ; DATA - Information to be added to the end of the current line  ; LABEL - Label to describe the information being added  ; COL - Column position in line to add information add  ; LNG - Maximum length of data information to include on the line  ; Returns: Line updated with added information  ;S LINE=LINE\_$J("",(COL-$L(LABEL)-$L(LINE)))\_LABEL\_$E(DATA,1,LNG)  S NEW=LABEL\_$E(DATA,1,LNG)  S $E(LINE,COL)=NEW  Q LINE |

| Routines | Activities | | | |
| --- | --- | --- | --- | --- |
| **Routine Name** | IBCNSM | | | |
| **Enhancement Category** | New | Modify | Delete | No Change |
| **RTM** | COH-3.1-01 - COH-3.1-15 | | | |
| **Related Options** |  | | | |

| Related Routines | Routines “Called By” | Routines “Called” |
| --- | --- | --- |
|  |  | ^DIC, DT^DICRW, ^DIR, $$BUFFER^IBCNBU1, $$GRP^IBCNS, ALL^IBCNS1, $$PLCOV^IBCNSU3, POL^IBCNSU41, $$DAT1^IBOUTL, PID^VADPT, EN^VALM, $$SETFLD^VALM1, FULL^VALM1, CLEAN^VALM10, $$FMTE^XLFDT |

| Routines | Activities | | | | |
| --- | --- | --- | --- | --- | --- |
| **Data Dictionary (DD) References** | ^DIC(36, ^DPT(, ^IBA(354, ^IBA(355.3, ^IBE(355.1, ^TMP(“IBNSM”, ^TMP(“IBNSMDX”, ^VA(200 | | | | |
| **Related Protocols** |  | | | | |
| **Related Integration Control Registrations (ICRs)** |  | | | | |
| **Data Passing** | Input | Output Reference | Both | Global Reference | Local |
| **Input Attribute Name and Definition** | Name: NONE  Definition: NONE | | | | |
| **Output Attribute Name and Definition** | Name: NONE  Definition: NONE | | | | |

| Current Logic |
| --- |
| ;  INIT ; -- set up initial variables  S U="^",VALMCNT=0,VALMBG=1  K ^TMP("IBNSM",$J),^TMP("IBNSMDX",$J)  ;K I,X,SDBEG,SDEND,SDB,XQORNOD,SDFN,SDCLN,DA,DR,DIE,DNM,DQ  S DIR(0)="350.9,4.02",DIR("A")="Select Patient Name or Insurance Co."  D ^DIR K DIR I $D(DIRUT) S VALMQUIT="" G INITQ  S IBY=Y  I IBY["DPT(" S IBTYP="P",DFN=+IBY D BLD  I IBY["DIC(" S IBTYP="I",IBCNS=+IBY D EN^VALM("IBCNS INSURANCE COMPANY") S VALMQUIT=""  ;  INITQ Q  ; |

| Modified Logic (Changes are highlighted) |
| --- |
| ;  INIT ; -- set up initial variables  S U="^",VALMCNT=0,VALMBG=1  K ^TMP("IBNSM",$J),^TMP("IBNSMDX",$J)  ;K I,X,SDBEG,SDEND,SDB,XQORNOD,SDFN,SDCLN,DA,DR,DIE,DNM,DQ  I $G(GOTPAT)=1 D G INITQ ; \*554 - baa ; allow entry from another list manager  . N IBTYP,DFN  . S DFN=^TMP($J,"PATINS")  . S IBTYP="P"  . D BLD  S DIR(0)="350.9,4.02",DIR("A")="Select Patient Name or Insurance Co."  D ^DIR K DIR I $D(DIRUT) S VALMQUIT="" G INITQ  S IBY=Y  I IBY["DPT(" S IBTYP="P",DFN=+IBY D BLD  I IBY["DIC(" S IBTYP="I",IBCNS=+IBY D EN^VALM("IBCNS INSURANCE COMPANY") S VALMQUIT=""  ;  INITQ Q  ; |

| Routines | Activities | | | |
| --- | --- | --- | --- | --- |
| **Routine Name** | IBTRE | | | |
| **Enhancement Category** | New | Modify | Delete | No Change |
| **RTM** | COH-3.1-01 - COH-3.1-15 | | | |
| **Related Options** |  | | | |

| Related Routines | Routines “Called By” | Routines “Called” |
| --- | --- | --- |
|  |  |  |

| Routines | Activities | | | | |
| --- | --- | --- | --- | --- | --- |
| **Data Dictionary (DD) References** |  | | | | |
| **Related Protocols** |  | | | | |
| **Related Integration Control Registrations (ICRs)** |  | | | | |
| **Data Passing** | Input | Output Reference | Both | Global Reference | Local |
| **Input Attribute Name and Definition** | Name: NONE  Definition: NONE | | | | |
| **Output Attribute Name and Definition** | Name: NONE  Definition: NONE | | | | |

| Current Logic |
| --- |
| ;  INIT ; -- init variables and list array  S U="^",VALMCNT=0,VALMBG=1  K ^TMP("IBTRE",$J),^TMP("IBTREDX",$J)  K I,X,XQORNOD,DA,DR,DIE,DNM,DQ,IBTEDT,IBTBDT  D PAT^IBCNSM I $D(VALMQUIT) G INITQ  S IBTBDT=$$FMADD^XLFDT(DT,-365) ; default start date 1 year in past  S IBTEDT=$$FMADD^XLFDT(DT,14) ;default end date is 7 days in future.  D BLD  INITQ Q  ; |

| Modified Logic (Changes are highlighted) |
| --- |
| ;  INIT ; -- init variables and list array  S U="^",VALMCNT=0,VALMBG=1  K ^TMP("IBTRE",$J),^TMP("IBTREDX",$J)  K I,X,XQORNOD,DA,DR,DIE,DNM,DQ,IBTEDT,IBTBDT  I $G(GOTPAT)=1 D G INITQ ; \*554 - baa ; allow entry from another list manager  . N REC,DFN  . S REC=^TMP($J,"IBCLMTRK")  . S DFN=$P(REC,U,1),IBTBDT=$$FMADD^XLFDT($P(REC,U,2)),IBTEDT=$$FMADD^XLFDT($P(REC,U,3))  . D BLD  D PAT^IBCNSM I $D(VALMQUIT) G INITQ  S IBTBDT=$$FMADD^XLFDT(DT,-365) ; default start date 1 year in past  S IBTEDT=$$FMADD^XLFDT(DT,14) ;default end date is 7 days in future.  D BLD  INITQ Q  ; |

| Routines | Activities | | | |
| --- | --- | --- | --- | --- |
| **Routine Name** | IBAMTC | | | |
| **Enhancement Category** | New | Modify | Delete | No Change |
| **RTM** | COH-2.3-01 | | | |
| **Related Options** |  | | | |

| Related Routines | Routines “Called By” | Routines “Called” |
| --- | --- | --- |
|  | IBAMTD | ^%DTC, C^%DTC, H^%DTC, NOW^%DTC, YMD^%DTC, KILL^%ZTLOAD, $$BILST^DGMTUB, $$MVT^DGPMOBS, CVEXMAIL^IBACV, IFCVEXP^IBACV, $$ISLTCADM^IBAECN1, NJ^IBAECN1, ^IBAERR1, PURGE^IBAERR3, BULL^IBAMTC1, MAIN^IBAMTC2, ADM^IBAMTI, REHLD^IBAMTV3, RELPR^IBAMTV3, EN^IBARXEL, NIGHT^IBARXMA, ^IBATEI1, SITE^IBAUTL, SERV^IBAUTL2, CLOCK^IBAUTL3, CLOCKCL^IBAUTL3, EVFIND^IBAUTL3, EVUPD^IBAUTL3, ^IBAUTL4, $$SECT^IBAUTL5, $$STD^IBAUTL5, FILER^IBAUTL5, LAST^IBAUTL5, LAST^IBAUTL5, $$SWSTAT^IBBAPI, ^IBCD, EN^IBCE, SENDEII^IBCNFSND, PUT^IBCNHUT2, BJ^IBJDE, EN^IBOHRL, $$FY^IBOUTL, NIGHTLY^IBTRKR, ELIG^VADPT, IN5^VADPT, $$FMADD^XLFDT |

| Routines | Activities | | | | |
| --- | --- | --- | --- | --- | --- |
| **Data Dictionary (DD) References** | ^DGPM(, ^DPT(“CN”, ^IB(, ^IBE(351, ^IBE(351.2, ^TMP($J | | | | |
| **Related Protocols** |  | | | | |
| **Related Integration Control Registrations (ICRs)** |  | | | | |
| **Data Passing** | Input | Output Reference | Both | Global Reference | Local |
| **Input Attribute Name and Definition** | Name: NONE  Definition: NONE | | | | |
| **Output Attribute Name and Definition** | Name: NONE  Definition: NONE | | | | |

| Current Logic |
| --- |
| D NIGHTLY^IBTRKR ; claims tracking nightly update  ;  D ^IBCD ; automated biller  ;  D RELPR^IBAMTV3 ; auto-release patient charges on hold at least 60 days  ;  D EN^IBOHRL ; auto-release patient charges on hold longer than 90 days |

| Modified Logic (Changes are highlighted) |
| --- |
| D NIGHTLY^IBTRKR ; claims tracking nightly update  ;  D ^IBCD ; automated biller  ;  ; IB\*2.0\*554  D RELHLD^IBAMTV3 ; auto-release patient charges on hold if no coverage for date  ;  D RELPR^IBAMTV3 ; auto-release patient charges on hold at least 60 days  ;  D EN^IBOHRL ; auto-release patient charges on hold longer than 90 days |

| Routines | Activities | | | |
| --- | --- | --- | --- | --- |
| **Routine Name** | IBAMTV3 | | | |
| **Enhancement Category** | New | Modify | Delete | No Change |
| **RTM** | COH-2.3-01 | | | |
| **Related Options** |  | | | |

| Related Routines | Routines “Called By” | Routines “Called” |
| --- | --- | --- |
|  | IBAMTC | ^%DTC, NOW^%DTC, $$LST^DGMTU, $$BUFFER^IBCNBU1, ^IBCNS, $$INSURED^IBCNS1, ssPTCOV^IBCNSU3, $$CHECK^IBCNSU3, $$CHECK^IBECEAU, $$PT^IBEFUNC, MAIL^IBOHRL, REL^IBOHRL, $$DAT1^IBOUTL, $$DAT2^IBOUTL, EN^VALM, $$SETSTR^VALM1, FULL^VALM1, CLEAN^VALM10, DISP^XQORM1 |

| Routines | Activities | | | | |
| --- | --- | --- | --- | --- | --- |
| **Data Dictionary (DD) References** | ^IB(, ^IB(“AC”, ^IB(“AJ”, ^IBE(350.1, ^TMP(“IBAMTV3”, ^TMP($J | | | | |
| **Related Protocols** |  | | | | |
| **Related Integration Control Registrations (ICRs)** |  | | | | |
| **Data Passing** | Input | Output Reference | Both | Global Reference | Local |
| **Input Attribute Name and Definition** | Name:  Definition: | | | | |
| **Output Attribute Name and Definition** | Name:  Definition: | | | | |

| Current Logic |
| --- |
| RELPR ; Release charges on hold at least 60 days old.  K ^TMP($J,"IBHOLD") D NOW^%DTC S TDY=%  S IBN=0 F S IBN=$O(^IB("AC",21,IBN)) Q:'IBN D  .S DFN=+$P($G(^IB(IBN,0)),U,2),X2=+$P($G(^IB(IBN,1)),U,4) Q:'DFN!('X2)  .S X1=TDY D ^%DTC Q:X<60 S ^TMP($J,"IBHOLD",DFN,IBN)=""  ;  I '$D(^TMP($J,"IBHOLD")) G RELQ  S IBR60=1 D REL^IBOHRL ; Release charges  S IBSTJB=$$DAT2^IBOUTL(TDY) D MAIL^IBOHRL ; Send bulletin  ;  RELQ K DFN,IBDUZ,IBEND,IBN,IBDIFROM,IBNOS,IBNUM,IBRCOUNT,IBR60,IBSEQNO,IBSTJB  K IBT,TDY,XMDUZ,XMGRP,XMSUB,XMTEXT,XMY,X,X1,X2,%,^TMP($J,"IBHOLD")  Q |

| Modified Logic (Changes are highlighted) |
| --- |
| RELPR ; Release charges on hold at least 60 days old.  K ^TMP($J,"IBHOLD") D NOW^%DTC S TDY=%  S IBN=0 F S IBN=$O(^IB("AC",21,IBN)) Q:'IBN D  .S DFN=+$P($G(^IB(IBN,0)),U,2),X2=+$P($G(^IB(IBN,1)),U,4) Q:'DFN!('X2)  .S X1=TDY D ^%DTC Q:X<60 S ^TMP($J,"IBHOLD",DFN,IBN)=""  ;  I '$D(^TMP($J,"IBHOLD")) G RELQ  S IBR60=1 D REL^IBOHRL ; Release charges  S IBSTJB=$$DAT2^IBOUTL(TDY) D MAIL^IBOHRL ; Send bulletin  ;  RELQ K DFN,IBDUZ,IBEND,IBN,IBDIFROM,IBNOS,IBNUM,IBRCOUNT,IBR60,IBSEQNO,IBSTJB  K IBT,TDY,XMDUZ,XMGRP,XMSUB,XMTEXT,XMY,X,X1,X2,%,^TMP($J,"IBHOLD")  Q  ;  ;  ; IB\*2.0\*554  RELHLD ; Release copay charges. Automatically release copay charges  ; for a patient when insurance file coverage limitations for the  ; date of service/fill date indicate that there is no coverage  ; for that particular type of service.  ;  N XX,IBINS,IBX,DFN,X2,IBFR,IBTO,IBINDT,IBINS,IBOUTP  N IBAT,IBCAT,IBCOV,IBVDT  ;  S IBN=0 F S IBN=$O(^IB("AC",8,IBN)) Q:'IBN D  . S IBX=$G(^IB(IBN,0))  . S DFN=+$P(IBX,U,2)  . S X3=$P(IBX,U,3)  . S X5=+$P(IBX,U,5) Q:X5'=8  . S IBFR=$P(IBX,U,14),IBTO=$P(IBX,U,15)  . S IBINDT=IBFR  . S IBINS=0  . S IBOUTP=1  . S RELHLD=$$RELHLD^IBCNBAR(DFN)  . I RELHLD S ^TMP($J,"IBHOLD",DFN,IBN)=""  ;  I '$D(^TMP($J,"IBHOLD")) G RELHQ  D REL^IBOHRL ; Release charges  ;  RELHQ K ^TMP($J,"IBHOLD")  Q  ; |

##### Templates

Table 6: Templates

| Templates | Description | | | |
| --- | --- | --- | --- | --- |
| **Template Name** | IBOH HELD CHARGES LIST | | | |
| **Enhancement Category** | New | Modify | Delete | No Change |
| **RSD** | COH-3.1-01 - COH-3.1-15 | | | |
| **Template Type** | Sort | Input | Print | Other |
| **Related Options** | IB OUTPUT HELD CHARGES LM | | | |

| **Related Routines** | **Routines “Called By”** | **Routines “Called”** |
| --- | --- | --- |
|  |  | HDR^IBOHLS  EXPAND^IBOHLS  HELP^IBOHLS  EXIT^IBOHLS  INIT^IBOHLS |

| Routines | Description |
| --- | --- |
| **DD References** |  |
| **Global References** |  |

| Templates | Description | | | |
| --- | --- | --- | --- | --- |
| **Template Name** | IBOH HELD CHARGES EXPAND | | | |
| **Enhancement Category** | New | Modify | Delete | No Change |
| **RSD** | COH-3.1-01 - COH-3.1-15 | | | |
| **Template Type** | Sort | Input | Print | Other |
| **Related Options** | IB OUTPUT HELD CHARGES LM | | | |

| **Related Routines** | **Routines “Called By”** | **Routines “Called”** |
| --- | --- | --- |
|  |  | HDR^IBOHLE  INIT^IBOHLE  EXIT^IBOHLE  HELP^IBOHLE |

| Routines | Description |
| --- | --- |
| **DD References** |  |
| **Global References** |  |

##### Data Entries Affected by the Design

No Data Entries affected by the design.

##### Unique Record(s)

No Unique Record ID affected by the design.

##### File or Global Size Changes

No File or Global changes.

##### Mail Groups

No Mail Groups changes.

##### Security Keys

No Security Key changes.

##### Options

Table 7: Options

| Options | Activities | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Option Name** | IB OUTPUT HELD CHARGES LM | | | | | | | | | | |
| **Enhancement Category** | New | Modify | | | | Delete | | | No Change | | |
| **Associated Menu Options that will invoke this reference** |  | | | | | | | | | | |
| **Data Passing** | Input | | Output | | Both | | | Global Reference | | | Local Reference |
| **Menu Text Description** | This is the new Held Charges Report List Manager. This will enable the users to work the held charges more efficiently. | | | | | | | | | | |
| **Option Type** | Edit | | | Print | | | Menu | | | Inquire | |
| Action | | | Run Routine | | | Other | | |  | |
| **Associated Routine** | EN^IBOHLS | | | | | | | | | | |
| **Option Definition** |  | | | | | | | | | | |

| Current Entry Action Logic |
| --- |
| NONE |

| Modified Entry Action Logic (Changes are highlighted) |
| --- |
| NONE |

| Current Exit Action Logic |
| --- |
| NONE |

| Modified Exit Action Logic (Changes are highlighted) |
| --- |
| NONE |

##### Protocols

Table 8: Protocols

| Protocols | Activities |
| --- | --- |
| **Protocol Name** | IBOH EXPAND HELD CHARGES |
| **Enhancement Category** | New  Modify  Delete  No Change |
| **Associated Protocols** | IBOH HELD CHARGES WORKLIST MENU |
| **Data Passing** | Input  Output  Both  Global Reference  Local Reference |
| **Item Text Description** | This options expands the selected entry. |
| **Protocol Type** | Action  Menu  Protocol  Menu  Limited Protocol  Extended Action  Dialog  Other |
| **Associated Routine** | IBOHLS |

| Current Entry Action Logic |
| --- |
| NONE |

| Modified Entry Action Logic (Changes are in bold) |
| --- |
| D EXPAND^IBOHLS |

| Current Exit Action Logic |
| --- |
| NONE |

| Modified Exit Action Logic (Changes are in bold) |
| --- |
| NONE |

##### 

| **Protocols** | **Activities** |
| --- | --- |
| **Protocol Name** | IBOH EXPORT TO FILE |
| **Enhancement Category** | New  Modify  Delete  No Change |
| **Associated Protocols** | IBOH HELD CHARGES WORKLIST MENU |
| **Data Passing** | Input  Output  Both  Global Reference  Local Reference |
| **Item Text Description** | This option exports the selected entry to an excel file. |
| **Protocol Type** | Action  Menu  Protocol  Menu  Limited Protocol  Extended Action  Dialog  Other |
| **Associated Routine** | IBOHLS2 |

| Current Entry Action Logic |
| --- |
| NONE |

| Modified Entry Action Logic (Changes are in bold) |
| --- |
| D EXPORT^IBOHLS2("IBHOLS",.FILTERS) |

| Current Exit Action Logic |
| --- |
| NONE |

| Modified Exit Action Logic (Changes are in bold) |
| --- |
| NONE |

| Protocols | Activities |
| --- | --- |
| **Protocol Name** | IBOH RELEASE HELD CHARGES |
| **Enhancement Category** | New  Modify  Delete  No Change |
| **Associated Protocols** | IBOH HELD CHARGES WORKLIST MENU |
| **Data Passing** | Input  Output  Both  Global Reference  Local Reference |
| **Item Text Description** | This option Releases the selected entry. |
| **Protocol Type** | Action  Menu  Protocol  Menu  Limited Protocol  Extended Action  Dialog  Other |
| **Associated Routine** | IBOHLS2 |

| Current Entry Action Logic |
| --- |
| NONE |

| Modified Entry Action Logic (Changes are in bold) |
| --- |
| D REL^IBOHLS2 |

| Current Exit Action Logic |
| --- |
| NONE |

| Modified Exit Action Logic (Changes are in bold) |
| --- |
| NONE |

| Protocols | Activities |
| --- | --- |
| **Protocol Name** | IBOH HELD CHARGES CLAIM TRACKING |
| **Enhancement Category** | New  Modify  Delete  No Change |
| **Associated Protocols** | IBOH HELD CHARGES WORKLIST MENU |
| **Data Passing** | Input  Output  Both  Global Reference  Local Reference |
| **Item Text Description** | This option displays the claim tracking information for the selected entry. |
| **Protocol Type** | Action  Menu  Protocol  Menu  Limited Protocol  Extended Action  Dialog  Other |
| **Associated Routine** | IBOHLS2 |

| Current Entry Action Logic |
| --- |
| NONE |

| Modified Entry Action Logic (Changes are in bold) |
| --- |
| D CLMTRK^IBOHLS2 |

| Current Exit Action Logic |
| --- |
| NONE |

| Modified Exit Action Logic (Changes are in bold) |
| --- |
| NONE |

| Protocols | Activities |
| --- | --- |
| **Protocol Name** | IBOH HELD CHARGES PATIENT INSURANCE |
| **Enhancement Category** | New  Modify  Delete  No Change |
| **Associated Protocols** | IBOH HELD CHARGES WORKLIST MENU |
| **Data Passing** | Input  Output  Both  Global Reference  Local Reference |
| **Item Text Description** | This option displays the patient insurance screen for  the selected entry. |
| **Protocol Type** | Action  Menu  Protocol  Menu  Limited Protocol  Extended Action  Dialog  Other |
| **Associated Routine** | IBOHLS2 |

| Current Entry Action Logic |
| --- |
| NONE |

| Modified Entry Action Logic (Changes are in bold) |
| --- |
| D PATINS^IBOHLS2 |

| Current Exit Action Logic |
| --- |
| NONE |

| Modified Exit Action Logic (Changes are in bold) |
| --- |
| NONE |

| Protocols | Activities |
| --- | --- |
| **Protocol Name** | IBOH HELD CHARGES TPJI |
| **Enhancement Category** | New  Modify  Delete  No Change |
| **Associated Protocols** | IBOH HELD CHARGES WORKLIST MENU |
| **Data Passing** | Input  Output  Both  Global Reference  Local Reference |
| **Item Text Description** | This options displays the patient claim information  screen for the selected entry. |
| **Protocol Type** | Action  Menu  Protocol  Menu  Limited Protocol  Extended Action  Dialog  Other |
| **Associated Routine** | IBOHLS2 |

| Current Entry Action Logic |
| --- |
| NONE |

| Modified Entry Action Logic (Changes are in bold) |
| --- |
| D PATCLM^IBOHLS2 |

| Current Exit Action Logic |
| --- |
| NONE |

| Modified Exit Action Logic (Changes are in bold) |
| --- |
| NONE |

| Protocols | Activities |
| --- | --- |
| **Protocol Name** | IBOH HELD CHARGES WORKLIST MENU |
| **Enhancement Category** | New  Modify  Delete  No Change |
| **Associated Protocols** | IBOH HELD CHARGES LIST |
| **Data Passing** | Input  Output  Both  Global Reference  Local Reference |
| **Item Text Description** | This is the menu option for the Held Charges list manager. |
| **Protocol Type** | Action  Menu  Protocol  Menu  Limited Protocol  Extended Action  Dialog  Other |
| **Associated Routine** |  |

| Current Entry Action Logic |
| --- |
| NONE |

| Modified Entry Action Logic (Changes are in bold) |
| --- |
| NONE |

| Current Exit Action Logic |
| --- |
| NONE |

| Modified Exit Action Logic (Changes are in bold) |
| --- |
| NONE |

| Protocols | Activities |
| --- | --- |
| **Protocol Name** | IBOH PRINT HELD CHARGES RPT |
| **Enhancement Category** | New  Modify  Delete  No Change |
| **Associated Protocols** | IBOH HELD CHARGES WORKLIST MENU |
| **Data Passing** | Input  Output  Both  Global Reference  Local Reference |
| **Item Text Description** | This option prints the held charges report. |
| **Protocol Type** | Action  Menu  Protocol  Menu  Limited Protocol  Extended Action  Dialog  Other |
| **Associated Routine** | IBOHLS2 |

| Current Entry Action Logic |
| --- |
| NONE |

| Modified Entry Action Logic (Changes are in bold) |
| --- |
| D RPT^IBOHLS2("IBOHLS",.FILTERS) |

| Current Exit Action Logic |
| --- |
| NONE |

| Modified Exit Action Logic (Changes are in bold) |
| --- |
| NONE |

| Protocols | Activities |
| --- | --- |
| **Protocol Name** | IBOH HELD CHARGES ACCOUNT PROFILE |
| **Enhancement Category** | New  Modify  Delete  No Change |
| **Associated Protocols** | IBOH HELD CHARGES WORKLIST MENU |
| **Data Passing** | Input  Output  Both  Global Reference  Local Reference |
| **Item Text Description** |  |
| **Protocol Type** | Action  Menu  Protocol  Menu  Limited Protocol  Extended Action  Dialog  Other |
| **Associated Routine** | IBOHLS2 |

| Current Entry Action Logic |
| --- |
| NONE |

| Modified Entry Action Logic (Changes are in bold) |
| --- |
| D PATACP^IBOHLS2 |

| Current Exit Action Logic |
| --- |
| NONE |

| Modified Exit Action Logic (Changes are in bold) |
| --- |
| NONE |

##### RPC

No RPC changes.

##### Constants Defined in Interface

No Interface changes.

##### Variables Defined in Interface

No Interface changes.

##### Types Defined in Interface

No Interface changes.

##### GUI

No GUI changes.

##### GUI Classes

##### Current Form

No Form changes

##### Modified Form

No Forms Modified.

##### Components on Form

No Components changed.

##### Events

No Events Changed.

##### Methods

No Methods changed.

##### Special References

No Special References changed.

##### Class Events

No Class Events changed.

##### Class Methods

No Class Methods changed.

##### Class Properties

No Class Properties changed.

##### Uses Clause

No Uses Clause changed.

##### Forms

No Forms changed.

##### Functions

No Functions changed.

##### Dialog

No Dialog changes.

##### Help Frame

No Help Frame changes.

##### HL7 Application Parameter

No HL7 application Parameter changes.

##### HL7 Logical Link

No HL7 Logical Link changes.

##### COTS Interface

No COTS Interface changes.

## Network Detailed Design

There are no network topology changes related to this project.

## Security and Privacy

This project does not add any additional security or privacy design considerations.

### Security

Security and Privacy are controlled by the VistA application and the underlying operating system. This application does not mandate any additional security or privacy requirements.

### Privacy

Security and Privacy are controlled by the VistA application and the underlying operating system. This application does not mandate any additional security or privacy requirements.

## SOA/ESS Detailed Design

Not applicable – reference Section 4.4.

# External System Interface Design

Not applicable – there is no external system involved with this project.

# Human-Machine Interface

Not applicable – this project does not change the human-machine interface, which is done via the VistA user options.

# Attachment A – Approval Signatures

The signature below is an acknowledgement that the signatory understands the purpose and content of this document. By signing this document, you agree to this as the SDD to begin work on the project described within, and commitment of the necessary resources.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_­­­­­­­­­­­­\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

xxxxxx

Office of Information & Technology (OI&T) Project Manager (PjM), Date

Integrated Project Team (IPT) Co-Chair

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_­­­­­­­­­­­­\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

xxxxx

Business Sponsor Date

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_­­­­­­­­­­­­\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

xxxxx

VHA PjM, IPT Co-Chair Date

# Attachment B – Acronyms

| Term | Definition |
| --- | --- |
| AITC | Austin Information Technology Center |
| API | Application Programming Interface |
| BRD | Business Requirements Document |
| BN | Business Need |
| CBOC | Community Based Outpatient Clinic |
| CHAMPVA | Civilian Health and Medical Program of the VA |
| CLC | Community Living Center |
| COTS | Commercial Off-the-Shelf |
| CPAC | Consolidated Patient Account Center |
| CPRS | Computerized Patient Record System |
| DBMS | Database Management System |
| DD | Data Dictionary |
| EDI | Electronic Data Interchange |
| EHR | Electronic Health Record |
| ERD | Entity Relationship Diagram |
| ERR | Enterprise Requirements Repository |
| ESS | Enterprise Shared Services |
| GUI | Graphical User Interface |
| HL7 | Health Level 7 |
| HRC | Health Resource Center |
| IB | Integrated Billing |
| ICB | Insurance Capture Buffer |
| ICR | Integration Control Registration |
| IPT | Integrated Project Team |
| LAN | Local Area Network |
| MDWS | Medical Domain Web Services |
| MUMPS | Massachusetts General Hospital Utility Multi-Programming System |
| OI&T | Office of Information & Technology |
| OWNR | Owner Requirement |
| PI | Patient Insurance |
| PjM | Project Manager |
| RPC | Remote Procedure Call |
| RSD | Requirements Specification Document |
| RTM | Requirements Traceability Matrix |
| SDD | System Design Document |
| SOA | Service Oriented Architecture |
| SSN | Social Security Number |
| TRM | Technical Reference Model |
| TSPR | Technical Services Project Repository |
| VA | Department of Veterans Affairs |
| VAMC | VA Medical Center |
| VHA | Veterans Health Administration |
| VistA | Veterans Health Information Systems and Technology Architecture |

# Attachment C – Additional Information

This section and its sub-sections are to be determined. Blank sections have been kept for version control.

## A.1 Identification of Technology and Standards

## A.2 Constraining Policies, Directives and Procedures

## A.3 RTM

The Copays on Hold – Medicare/No Insurance RTM is available via TSPR here:

[http:/xxxxxxxxxxxxxxxxx/warboard/anotebk.asp?proj=1785&Type=Active](http://xxxxxxxxxxxxxxxxx/warboard/anotebk.asp?proj=1785&Type=Active)

Until the RTM is approved and posted to TSPR, it will be made available upon request from the project team.

## A.4 Packaging and Installation

## A.5 Design Metrics

# Attachment D – Change Log

| Version | Change | Notes | Status/Resolution |
| --- | --- | --- | --- |
| 0.02 | Replace embedded diagrams with links to master/latest file location (TSPR) globally | Avoids blurriness and/or version control issues | Completed in v0.02 tracked changes returned to vendor |
| 0.02 | Update acronyms globally |  |  |
| 0.02 | Section 1.2: updated primary users |  |  |
| 0.02 | Section 2.2: “To remedy…specific patient.” | Update language accordingly |  |
| 0.02 | Section 2.2 (previously 3.1): address current state (CSWA) comments |  |  |
| 0.02 | Section 2.3.1, bullet 1: “Automatically…copay date.” | This is a critical “and”. Also if this is executed then the CPAC work around processes for Camp Lejeune and Clay Hunt will be rendered non-functional for the assessment of the services for copayment charges. | N/A |
| 0.02 | Section 2.3.1: add in table of functional requirements | To maintain consistency with other CPAC SDDs |  |
| 0.02 | 508d entire document | Use this template moving forward | Completed in v0.02 tracked changes returned to vendor |
| 0.03 | Globally: Verbiage updated to match across all 5 SDDs |  |  |
| 0.03 | Globally: Ensure acronyms are defined at first mention; update Acronyms table accordingly |  |  |
| 0.03 | Globally: 508 images – add alternative text |  |  |
| 0.03 | Section 2.2: Updated overview to reflect FSWA verbiage |  |  |
| 0.03 | Section 2.3: Subsections re-inserted; were lost since last version after formal review |  |  |
| 0.03 | Section 2.3.1: Add in table of functional requirements | To maintain consistency with other CPAC SDDs |  |
| 0.03 | Section 3.1.1: Comments on CSWA and FSWA | Diagrams are now referenced in Section 2.2; do not embed to control version history – just point to document location |  |
| 0.03 | Section 3.2.4 and subsections: Don’t exist in other SDDs | Needed or need to be added in other SDDs? |  |

Template Revision History

| Date | Version | Description | Author |
| --- | --- | --- | --- |
| June 2015 | 2.10 | Changed Heading 1 default setting to eliminate page break before | Process Management |
| May 2015 | 2.9 | Edited for Section 508 conformance and remediated with Common Look Office tool | Process Management |
| February 2015 | 2.8 | Incorporates revisions from PMAS Reform Lockdown; namely removing requirements for information that can be obtained from other PMAS authoritative sources. | Andrew Slawter, Office of Technology Strategies |
| September 2014 | 2.7 | Adds Enterprise Shared Services terms and requires AERB Compliance Certificate attachment. | Process Management |
| August 2014 | 2.6 | Signature block update authorized by AERB CR\_018934 | Process Management |
| March 2014 | 2.5 | Section 508 repairs to new version approved by AERB Chair approved | Process Management |
| August 2013 | 2.3 | Replaced the Service Architecture sub-section with new sub-sections for consumed and provided services. Also applied miscellaneous feedback from VA team. | ASD Enterprise Shared Services (ESS) Work Group |
| June 2013 | 1.3 | Upgraded to MS Office 2007-2010 format | Process Management |
| June 2013 | 1.2 | Address inconsistencies in Section 3, Conceptual Design, Correct headings | Process Management |
| March 2013 | 1.1 | Formatted to documentation standards and edited for Section 508 conformance | Process Management |
| January 2013 | 1.0 | Initial Document | PMAS Business Office |

See TOGAF® 9.1, Part III: ADM Guidelines & Techniques, Gap Analysis on TOGAF website at <http://pubs.opengroup.org/architecture/togaf9-doc/arch/chap27.html>