Comprehensive Detailed System Design

Kentucky MMIS Project

Cabinet for Health and Family Services
Kentucky Medicaid Office

March 10, 2008

Cabinet for Health and Family Services Kentucky Medicaid Office	
Role:	Name:
Author	Scott Lowry
Reviewer	Patti George
EDS Management	Glenn Jennings, Account Manager
Client	Commissioner Elizabeth A. Johnson Deputy Commissioner Carrie Banahan Claims Division Director Rhonda Poston Executive Director/CIO Lorna S. Jones Chief Technical Officer Sandeep Kapoor
DELIVERABLE TITLE: System Wide Detailed System Design	DATE SUBMITTED: March 10, 2008
LOCATION and FILE NAME: KY_MMIS_Comprehensive_DSD__FINAL_(03-10-2008).doc	AUTHORING TOOL: MSWord2003

Document Change Log

Version	Changed Date	Changed By	Reason
1.0	9/13/2007	Scott Lowry	Creation of new document – 1st draft.
2.0	9/19/2007	Scott Lowry	Revisions made according to internal review comments – 2nd Draft.
3.0		Scott Lowry	Revisions made according to Commonwealth comments – 3rd Draft.
4.0	12/10/2007	Scott Lowry	Revisions made according to Walkthrough Comment Log – FINAL.

Table of Contents

1 Introduction Comprehensive DSD 1

2 System Wide Subsystem Detailed System Design 2

2.1 System Wide Subsystem Introduction 2
2.2 System Flow 3
2.2.1 interChange System Architecture 3
2.2.2 Infrastructure 4
2.2.3 Backup System Overview 5
2.3 Processes 15
2.3.1 Correspondence, Letter, and Notice Management Process Model 15
2.3.1.1 Process Description 15
2.4 Letter Generator Data Model 18
2.4.1 External System Interfaces 18
2.5 System Wide Subsystem Job Scripts 19
2.5.1 Autosys job - Autosys 19
2.5.2 watch_file - Autosys File Watcher 19
2.5.3 AIMJD010 - Change cycle date parm 19
2.5.4 AIMJW010 - Change cycle date parm 19
2.5.5 SYSJD130 - Clean the logs directory of files that are over one day old 20
2.5.6 SYSJD110 - Clean the prod override directory of members that are over three days old20
2.5.7 AIMJD999 - Flag cycle date when all jobs are complete 20
2.5.8 AIMJM010 - Maintain Monthly Cycle Date 21
2.5.9 AIMJQ010 - Maintain Quarterly Cycle Date 21
2.5.10 CHKJDOVR - Produces the $PRODDIR/override activity report 21
2.5.11 SYSJSOAP_START - SYSJSOAP_START 23
2.5.12 SYSJD120 - Send the email report for the production overrides. 23
2.5.13 SYSJDCHSE - System Check of Real-Time Processes 23
2.6 Programs 25
2.6.1 COLD Report Storage and Indexing -- COLD Report Storage and Indexing 25
2.6.2 SoapSrvrMgr.sh -- Start, stop the SOAP servers based on the input parameter 26
2.6.3 aimp010d -- Update parms table cycle start date 27
2.6.4 aimp010m -- aimp010m 28
2.6.5 aimp010q -- aimp010q 29
2.6.6 aimp010w -- aimp010w 30
2.6.7 aimp999d -- aimp999d 31
2.6.8 archive_logs.sh -- Move log file archives 32
2.6.9 autorep -- autorep 33
2.6.10 biztalk -- Biz Talk 35
2.6.11 clean_pid_files -- Cleanup SOAP work files 36
2.6.12 clmphash -- Build Hash Index 37
2.6.13 copy2crld -- CRLD copy 39
2.6.14 copy2routedir -- Copy Reports to Router 40
2.6.15 cycle_email.sh -- get the email addresses for the cycle monitors 41
2.6.16 dbdelete -- Delete rows from a database table 42
2.6.17 dbunload -- Utility to unload database tables based on a query 44
2.6.18 dumppos -- Dump POS TXNS 46
2.6.19 eoj_msg.csh -- Format the end of job message 47
2.6.20 execpgm.csh -- Execute programs from batch jobs 48
2.6.21 exit -- UNIX shell exit comment 49
2.6.22 ftp -- File transfer 50
2.6.23 gdg -- Generated Data Group 51
2.6.24 grant.revoke.privs.ks -- grant.revoke.privs.ks 52
2.6.25 jil -- Autosys 53

2.6.26 job_begin.src -- Common Job Script initialization 54
2.6.27 jsbeg_msg.csh -- Formats the beginning of JobStep message 55
2.6.28 letter_requests -- letter_requests 56
2.6.29 letter_templates -- letter_templates 57
2.6.30 libclmhash.so -- Hash and Parse Utilities 58
2.6.31 libded.so -- look ups for the DED 59
2.6.32 liblbms.so -- LBMS utility functions 60
2.6.33 libmemhash.so -- Memory hash functions 61
2.6.34 libmrpf.so -- Report writing functions library 62
2.6.35 libnatural.so -- Support functions for natural 63
2.6.36 libSOAPsave.so -- Save files through SOAP 64
2.6.37 libsysltr.so -- Create entry in letter request table 65
2.6.38 libxml.so -- XML Processing 66
2.6.39 libxml_nomt.so -- XML Processing 67
2.6.40 libxmlmrge.so -- Query Unload and Merge 68
2.6.41 massdrup.sh -- massdrup.sh 69
2.6.42 oraload.sh -- Oracle Table Load-Unload Utility 70
2.6.43 otsortd -- Sort - UNIX 73
2.6.44 override.src -- Processes Job Script Overrides 75
2.6.45 ps2comm.lib -- ps2comm link library 76
2.6.46 route_rpt.ksh -- Route report 77
2.6.47 sendevent -- Schedule a job in Autosys 78
2.6.48 set_ftplogin -- Set-up netrc file 79
2.6.49 SOAPsrvr -- Soap Server transaction processor 80
2.6.50 sqlldr -- Oracle Utility 81
2.6.51 watcher -- Watcher 82
2.6.52 xml_db -- XML-2-Database Transfer Utility 83
2.6.53 xmlpmrge -- Do merge base on hierarchical query 85
2.6.54 xmlpunld -- Generate a hierarchical xml data stream from a query tree 87
2.6.55 xmlpunld.sh -- xmlpunld.sh 89
2.7 Internal Web Pages 90
2.7.1 PAGE: System Wide Base Page 90
2.7.2 PANEL: System Wide Page Header 92
2.7.3 PANEL: Site Settings 95
2.7.4 PANEL: Help 99
2.7.5 PANEL: Audit Trail 101
2.8 Reports 106
2.8.1 Audit Trail Report 106
2.8.2 SYS-RPTFTR -- Report Footer Standard 111
2.8.3 SYS-RPTHDR -- Report Header Standard 113
2.9 Letters 116
2.10 Internal and External Interfaces 117
2.11 Requirement Matrix and Cross-reference 117
2.12 Change Orders 129
2.12.1 Target Operational Date Change - 43 129
2.12.2 Use Captive for Data Capture - 44 130
2.12.3 RSD & GSD combined into one GSD - 45 131
2.12.4 Create KY Operating Procedures - 212 132
2.12.5 Remove PES Requirement - 402 134
2.12.6 Eliminate agreed upon Req. - 403 135
2.12.7 Biztalk at Cold Harbor - 1072 136
2.12.8 OnBase Storage - 1073 137
2.12.9 Audit Reports - 1251 138
2.12.10 KY LG URL - 1958 139
2.12.11 State specific setup - 1967 140
2.12.12 Replace New KY MMIS claims mgr - 1977 141

2.12.13 Remove func.ops. Requirement - 1978 142
2.12.14 FirstDecision Schema for DSS - 1980 143
2.12.15 fix schema generation - 1982 144
2.12.16 Letters in COLD - 2096 146
2.12.17 ReportHeader - 2129 147
2.12.18 Kentucky Banner - 2153 148
2.12.19 promote Core fixes to Kentucky - 2165 149
2.12.20 Report Routing - 2185 150
2.12.21 Waiver Mailing - 2188 151
2.12.22 Epocrates Mailing - 2189 152
2.13 Data Element Dictionary 153
2.13.1 T_AUDIT_PURGE 153
2.13.2 T_AUDIT_TBL_EXCPT 153
2.13.3 T_BATCH_RESTART 153
2.13.4 T_CODE_RESOLUTION 154
2.13.5 T_DATE_AGING 155
2.13.6 T_GIS_CODES 155
2.13.7 T_GROUP 156
2.13.8 T_HIST_PART 156
2.13.9 T_HOLIDAY_DATES 157
2.13.10 T_HS_DIR_PART 157
2.13.11 T_LETTER 158
2.13.12 T_LETTER_GROUP 158
2.13.13 T_LETTER_REQUEST 158
2.13.14 T_LETTER_TEMPLATE 159
2.13.15 T_LTR_RQST_TEMPLAT 160
2.13.16 T_LTR_RQST_TRK 160
2.13.17 T_QUERY 161
2.13.18 T_RTE_RPT_CTL 162
2.13.19 T_STATE_ADDR 163
2.13.20 T_SYSTEM_KEYS 164
2.13.21 T_SYSTEM_PARMS 165
2.13.22 T_SYS_ERR_MSG 165

3 System Wide Subsystem Appendix A – Letter Generator v2 Training Documentation 167

4 System Wide subsystem Appendix B – interChange User Interface Standards 168

5 System Wide subsystem Appendix C – Additional System Documentation 169

6 Appendix of Subsystem Detailed System Designs 170

1 Introduction Comprehensive DSD

This document provides the Comprehensive Detailed System Design (DSD) for the Kentucky Medicaid specific interChange implementation. The Comprehensive DSD includes system wide DSD material that does not fit within other subsystems. In FINAL form, an Appendix is included that provides the DSD for each of the following subsystems which have already been presented as a separate DSD:

- Benefit Administration;
- Claims;
- Contact Tracking Management System (CTMS);
- DSS/DW (Decision Support System/Data Warehouse) ;
- EPSDT (Early and Periodic Screening, Diagnosis, and Treatment) ;
- External Data Sharing and Exchange;
- Financial;
- maxMC
- Member Management;
- MAR (Management and Administrative Reporting) ;
- Provider;
- Quality Assurance and Audits;
- Reference;
- Security Management;
- Service/Prior Authorization;
- SUR (Surveillance and Utilization Review; and,
- TPL (Third Party Liability).

The DSD validates the design process and approach to solution, and identifies those items as specified in the Kentucky MMIS (KY MMIS) Request for Proposal (RFP) and as specified by Kentucky Medicaid.

2 System Wide Subsystem Detailed System Design

2.1 System Wide Subsystem Introduction

This section provides information on systems and utilities that do not fit within individual subsystems. In addition to a variety of utilities, this document includes detailed documentation on the letter generator, the interChange .NET application, Audit Trail Panels and Reports, Help Panels and the source code repositories.

The letter generator is a core EDS product and is configured for the Commonwealth of Kentucky using letter templates which are documented in the applicable subsystem DSD's. The technical details of the letter generator itself are provided in a developer and operation guide for the Letter Generator version 2.0 for use by EDS developers. This document was produced by the EDS CORE (base) interChange team and is not formatted according to Kentucky documentation standards. This document is incorporated as Appendix A.

The interchange .NET application is documented as pages and panels throughout this document and all other DSD's. Included here is the developer's document that discusses many of the internal details of the application. This document is incorporated as Appendix B.

The Audit Trail Panels and Reports provide information on the change history of records within interchange. These programs are controlled by xml configuration files which provide the information on what tables to present to the user and in what format. A representative sample of an audit panel and audit report is provided in this document.

The help panels, like the Audit panels, utilize content contained in XML files that can be edited by business analysts.

Source code is centrally managed with repositories for unique platforms. Included in this document are the various repositories and their locations.

There are four appendices associated with the system wide subsystem.

- Appendix A Letter Generator v2 Design Documentation
  this is a developer and operation guide for the Letter Generator version 2.0 for use by EDS developers. This document was produced by the EDS CORE (base) interChange team and is not formatted according to Kentucky documentation standards.

- Appendix B Letter Generator V2 Training Documentation

- This document provides detailed instruction for operations and maintenance of the Letter Generator sub-system. This document was produced by the EDS CORE (base) interChange team and is not formatted according to Kentucky documentation standards.

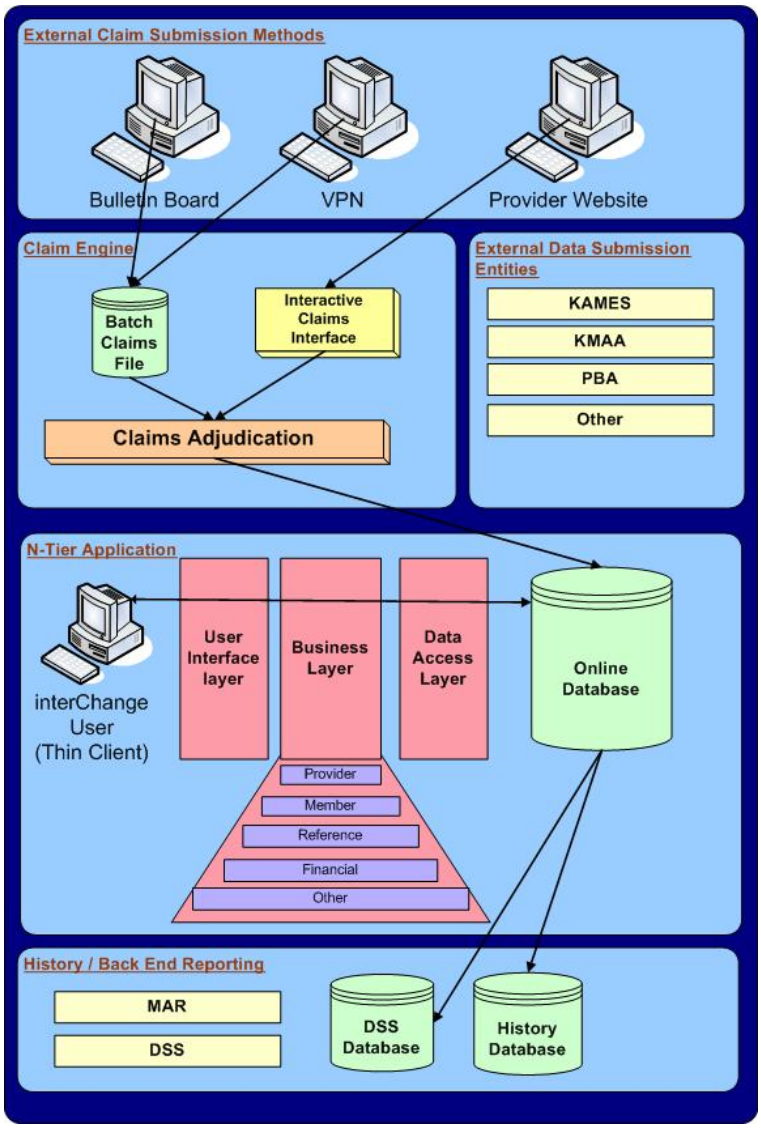
- Appendix C interChange .NET developers guide
  this contains to major sections. The first section is an introduction to the technologies used, the second is a description of how to use the EDS framework classes.

- Appendix D interChange User Interface Standards
  this presents the interChange .NET User Interface Standards Revision 1.3, February 2, 2006. This document was produced by the EDS CORE interChange team and is not formatted according to Kentucky documentation standards

2.2 System Flow

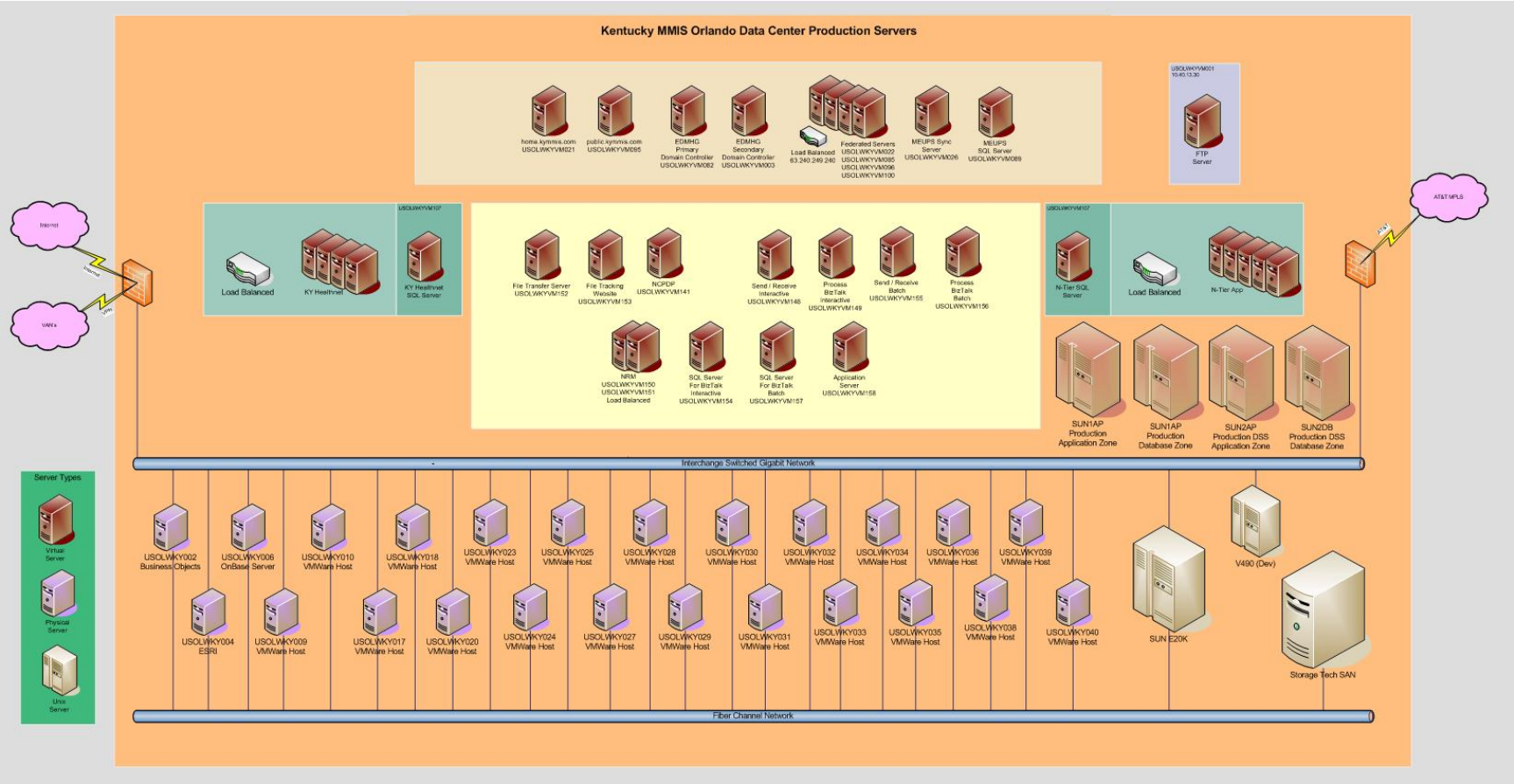
2.2.1 interChange System Architecture

The system is logically divided into four primary components: Claims engine, User Interface, Batch, and the History and Back End Reporting. The Claims Engine receives interactive transactions from external sources, adjudicates the claim, and returns the appropriate response to the external source. The User Interface is an N-tier application providing segregated and loosely coupled presentation, business logic, and data logic layers. The User Interface provides access to the online subsystem functions through a thin client and/or web browser. The Batch component is used to maintain and report on data in the online database. The History and Back End reporting component analyzes, reports, and supports the management of the activities that have occurred in the front end systems. The system interfaces with a variety of data sources including external data submission entities which influence processing within the system. The external data submission entities are organizations that supply information to the Medicaid Management Information System (MMIS).



2.2.2 Infrastructure

The following diagram depicts the infrastructure of the KY MMIS server in Orlando. There are two primary components of this diagram. The first component is the physical servers which are shown between the two blue network backbones. The majority of these servers are virtualized using VMWare or Solaris 10 domains and Zones.. The servers depicted in the top part of the diagram are virtual servers. They have been grouped on the diagram by functionality. The only virtual servers shown on the diagram are production servers.



2.2.3 Backup System Overview

The backup system in the ODC is a leveraged resource and will therefore serve all Medicaid accounts. The system is managed by the Veritas NetBackup software. Each machine requiring backup capability are configured as NetBackup media servers which has access to a set of drives in a virtual tape library.

The virtual tape library (VTL) consists of a disk array(s) and a VTL Appliance that sits in front of it, causing the disk array to appear to any server on the SAN as a tape library instead of a disk array. It looks and behaves like a tape library in every way, with tapes, slots, drives, compression, etc. Each account will have their own virtual tape library utilizing their allocated SATA storage. Each account will be allocated 2 times their primary storage for SATA.

Selected backups sent to the virtual tape library are copied to tapes in the physical library on a pre-defined schedule. Once on physical tapes, they will be removed and stored at an offsite facility. The physical tapes will be returned and re-cycled on a defined schedule. The schedule for copying of virtual tapes to physical tapes, the offsite of tapes and their return is defined by the policies outlined in a later section of this document.

2.2.3.1 Source Code repositories

There are three source code repositories for interchange applications. The use of multiple repositories is driven by technology needs.

.NET Source Code repository

The source code for the interchange .NET application, KY Healthnet, MEUPS, FTS, NRM, MAR, SUR and MEUPS reside on USOLWKYVM092 and are under the control of a source code management system called Vault.

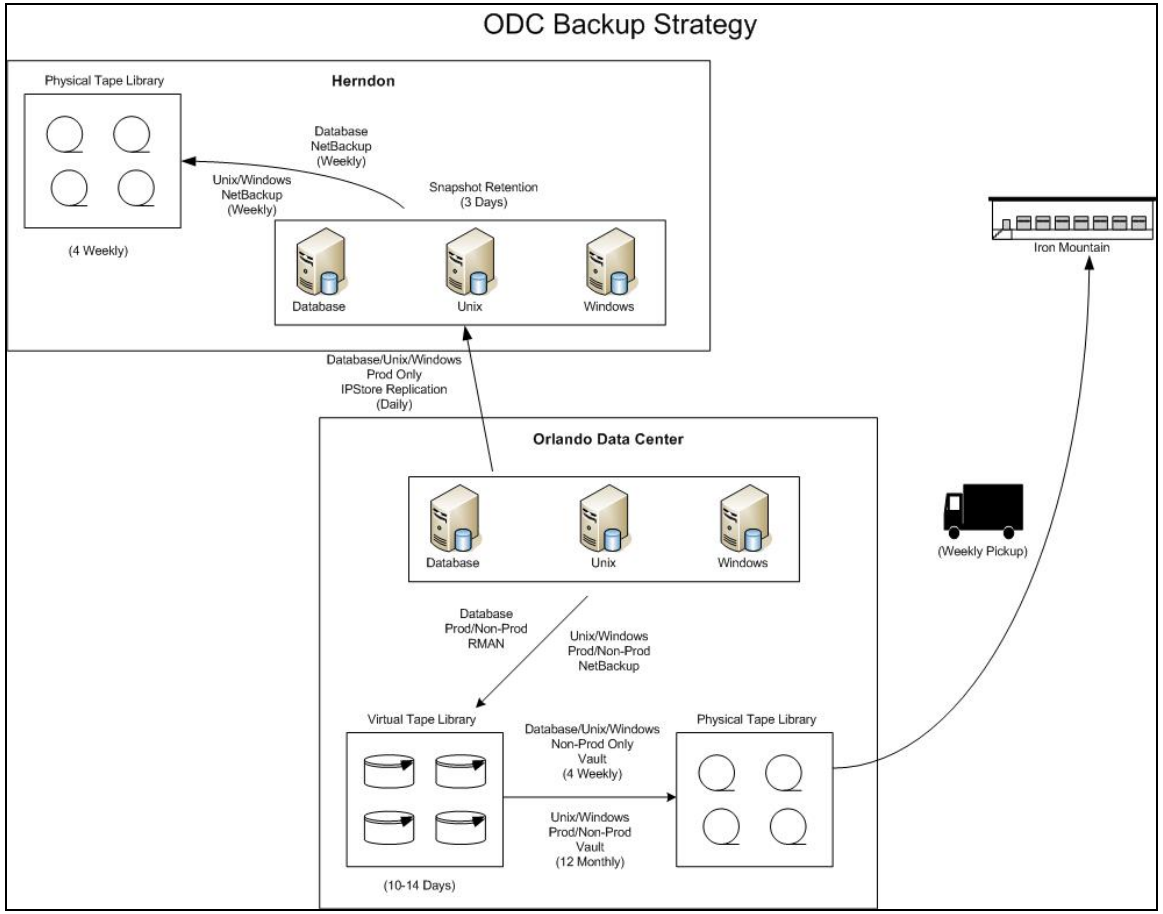
The source code for the C/Unix programs on the SUN e25K are on the DSKYSUN0AP zone under control of sccs (source code control system). EDS has put wrappers around SCCS to limit its functionality for developers. These wrappers are called VCTL (Version Control).

The source code for the DSS Business Objects applications resides on USOLWKY002 and is under control of the Business Objects application.

2.2.3.2 Backup Strategy

The backup strategy is designed to minimize recovery time and tape handling costs. Onsite recovery time is minimized by storing recent onsite backups on a virtual tape library. The virtual tape library resides on disk storage. This allows for shorter database and file restore times than would be possible using a traditional physical tape library. In addition, disaster recovery backups for all critical systems are performed using SAN replication to an offsite location. This allows for quick start-up of systems in the event of a disaster because the data is already on disk storage and does not need to be restored from tape.

The following diagram illustrates the overall backup strategy.



Onsite backups are performed at the Orlando Data Center (ODC). Database, unix filesystems and windows folders are written to the Virtual Tape Library (VTL). Backups are retained on the VTL for up to two (2) weeks. Daily ODC SAN snapshots are taken to replicate production data to the disaster recovery site in Herndon. Daily snapshots are taken of all production databases, unix filesystems and windows folders. At least three (3) snapshots will be retained on storage in Herndon. Once a week, a full tape backup will be performed in Herndon to copy one snapshot to tape for longer storage. All database, unix filesystems and windows folders included in the snapshot are backed up. The Herndon tape backups will be retained in the tape library for four (4) weeks at which time the tapes will be recycle.

In the ODC, weekly backups of non-production data is copied to physical tapes for offsite storage. All non-production databases, unix filesystems and windows folders are sent offsite to an Iron Mountain facility once per week. In addition, full backups of all unix filesystems and windows folders are taken once per month and sent offsite. The monthly backups include production and non-production data. Weekly backups are retained for 4 weeks offsite. Monthly backups are retained for 12 months offsite. All offsite tapes are transported and stored in locked containers. In the event that a tape needs to be recalled, the entire container will be back onsite.

This strategy provides for up to two (2) weeks of onsite recovery and up to three (3) days of production disaster recovery checkpoints. It also provides for weekly disaster recovery from tape of all non-production test systems. The strategy also provides four (4) weekly backups of all data and twelve (12) monthly backups of flat file data for research purposes, if necessary.

The backup strategy is also designed to minimize the demands placed on the backup system yet still meet the recovery needs. As with all hardware, the backup system has fixed resources and although it has been designed to be efficient, provide high throughput and to scale as needed, there is a breaking point where it can become overloaded. As a result, the backup strategy must be designed to make efficient use of these leveraged resources.

The following techniques will be used to minimize impact on the backup system:

- Daily database backups will employ an incremental backup strategy instead of performing resource intensive full backups. Databases backed up daily will perform a weekly full backup and daily incremental. Databases backed up weekly will continue performing full backups to simplify the tape retention and rotation policy.
- All Unix filesystem and Window folder backups will employ an incremental backup strategy consisting of a weekly full and daily incremental backups.
- Tape vaulting will be utilized to copy virtual tapes to physical media for offsite purposes. This replaces backups taken specifically to offsite tapes. This eliminates the burden the additional backups place on the system.
- Backups will be staggered throughout the week in order to distribute the load on the backup system. Unix and Window full backups will be performed by virtual server and spread out throughout the week.
- SQL Server database backups are written to disk. The disk location is either on the physical server or SAN attached. The database backups are backed up to virtual tape using the Windows folder backups.

The offsite tapes will be stored in the Orlando area to minimize transportation costs. In the event of a possible wide area disaster (i.e. a hurricane), the vendor may be asked to transport the tapes to a remote facility until the danger has passed.

2.2.3.3 Backup Policies

The backup policies are designed to meet the recovery objectives. The policies define where the backup data resides and how long the data is retained within the onsite backup system. The policies do not address the offsite replication to the disaster recovery site. This is discussed in the SAN replication section.

The following chart outlines the backup policies. There will be a set of policies for each site where xx is the state code for the site (e.g. KY-DB-Daily).

Policy	Backup Type	VTL Retention	PTL Retention	Offsite Retention
xx-AL-Daily-Offsite	Oracle Arclogs	14 Days	N/A	4 Weeks
xx-DB-Daily-Onsite	Oracle DB	14 Days	N/A	N/A
xx-DB-Weekly-Offsite	Oracle DB	2 Weeks	N/A	4 Weeks
xx-DB-Weekly-Onsite	Oracle DB	2 Weeks	N/A	N/A
xx-FF-Daily-Onsite	Unix/Windows	14 Days	N/A	N/A
xx-FF-Weekly-Offsite	Unix/Windows	2 Weeks	N/A	4 Weeks
xx-FF-Weekly-Onsite	Unix/Windows	2 Weeks	N/A	N/A
xx-FF-Monthly-Offsite	Unix/Windows	1 Month	N/A	12 Months

xx-AL-Daily-Offsite – This policy is for all database archive log backups. Includes both production and non-production databases. Archive logs are backed up to virtual tapes (SATA) assigned to this policy as needed throughout the day/night. The retention on this policy should be greater to or equal to the database backup policies. This ensures that database backup sets are always recoverable. The virtual tapes are copied to physical tapes as time permits and sent to offsite storage on a weekly basis. Physical tapes are returned from offsite storage and recycled after four (4) weeks.

xx-DB-Daily-Onsite – This policy is intended for daily database backups that remain onsite. It is primarily for cumulative incremental backups when using an incremental strategy. It may include full daily backups for those databases where it doesn't make sense to employ an incremental strategy. Both production and non-production database backups use this policy. Backups are retained on the virtual tapes (SATA) to ensure the backup set is always onsite and available on secondary storage. Virtual tapes are recycled and reused after 14 days.

xx-DB-Weekly-Offsite – This policy is intended for weekly database backups that need to be sent offsite. It is primarily for weekly full backups of non-production databases. Full backups for non-production databases in an incremental backup strategy are backed up here. As are full backups for non-production databases only backed up once a week. Backups are written to virtual tapes (SATA) and retained for two (2) weeks. The virtual tapes are copied to physical tapes as time permits and sent to offsite storage on a weekly basis. Physical tapes are returned from offsite storage and recycled after four (4) weeks.

xx-DB-Weekly-Onsite – This policy is intended for weekly database backups that remain onsite. It is primarily for weekly full backups of production databases. Full backups for production databases in an incremental backup strategy are backed up here. As are full backups for production databases only backed up once a week. Backups are written to virtual tapes (SATA) and retained for two (2) weeks. Virtual tapes are recycled and reused after 2 weeks.

xx-FF-Daily-Onsite – This policy is intended for daily Unix and Windows backups that remain onsite. It is primarily for incremental backups when using an incremental strategy. Both production and non-production backups use this policy. Backups are retained on the virtual tapes (SATA) to ensure the backup set is always onsite and available on secondary storage. Virtual tapes are recycled and reused after 14 days.

xx-FF-Weekly-Offsite – This policy is intended for weekly Unix and Windows backups that need to be sent offsite. It is primarily for weekly full backups of non-production files. Backups are written to virtual tapes (SATA) and retained for two (2) weeks. The virtual tapes are copied to physical tapes as time permits and sent to offsite storage on a weekly basis. Physical tapes are returned from offsite storage and recycled after four (4) weeks.

xx-FF-Weekly-Onsite – This policy is intended for weekly Unix and Windows backups that remain onsite. It is primarily for weekly full backups of production files. Backups are written to virtual tapes (SATA) and retained for two (2) weeks. Virtual tapes are recycled and reused after 2 weeks.

xx-FF-Monthly-Offsite – This policy is intended for monthly Unix and Windows backups to be sent offsite. It includes production and non-production files. Backups are written to virtual tapes (SATA) and retained for one (1) month. The virtual tapes are copied to physical tapes as time permits and sent to offsite storage on a monthly basis. Physical tapes are returned from offsite storage and recycled after twelve (12) weeks.

2.2.3.4 Backup Staggering

Many sites will be utilizing the leveraged backup system. To make efficient use of available resources, it is prudent to spread out the backup load. Each account will be responsible for distributing their backup load across a weekly time period. Staggering backups this way will help ensure smooth operation of the virtual tape libraries.

Backups will be scheduled to minimize the overlap of database, unix and windows backups. Database backups should be scheduled so that no more than one backup per virtual server is running at any given time. With large databases, it may be necessary to run these with little to no overlap on other virtual servers. Unix and window backups will be scheduled by virtual machine instead of physical server. Backups for virtual machines will be run on different nights to minimize runtime and to spread the work load.

If balancing backup load balancing by account is unsuccessful then we may need to look a scheduling scheme which assigns backup windows to each site. Due to the complexity of managing such a schedule, this will only be considered if necessary.

2.2.3.5 Remote SAN Replication

In the event that the ODC becomes non-operational due to some sort of disaster, processing will be switched to the disaster recovery facility in Herndon, VA. The production systems will be kept on warm standby at the Herndon facility using SAN-level replication.

The SAN replication will occur at the IPStore controller level using the TimeMark software. Replication occurs at the block-level. Only changed blocks are replicated. Blocks are compressed to minimize network traffic. The network traffic is encrypted to ensure a secure transmission. The application and database servers are not active during the replication.

The replication is performed by disk group. Files with interdependencies should be synchronized at the same time and therefore need to be placed in the same disk group. Otherwise, there is no guarantee they will be synchronized. This means all database files should be kept in the same disk group to ensure a consistent snapshot occurs. Databases in archivelog mode will be quiesed by the TimeMark for the snapshot. The quiesce will throw the database in backup mode until the snapshot is taken. It only takes a few seconds to establish the snapshot at which point the database is taken out of backup mode.

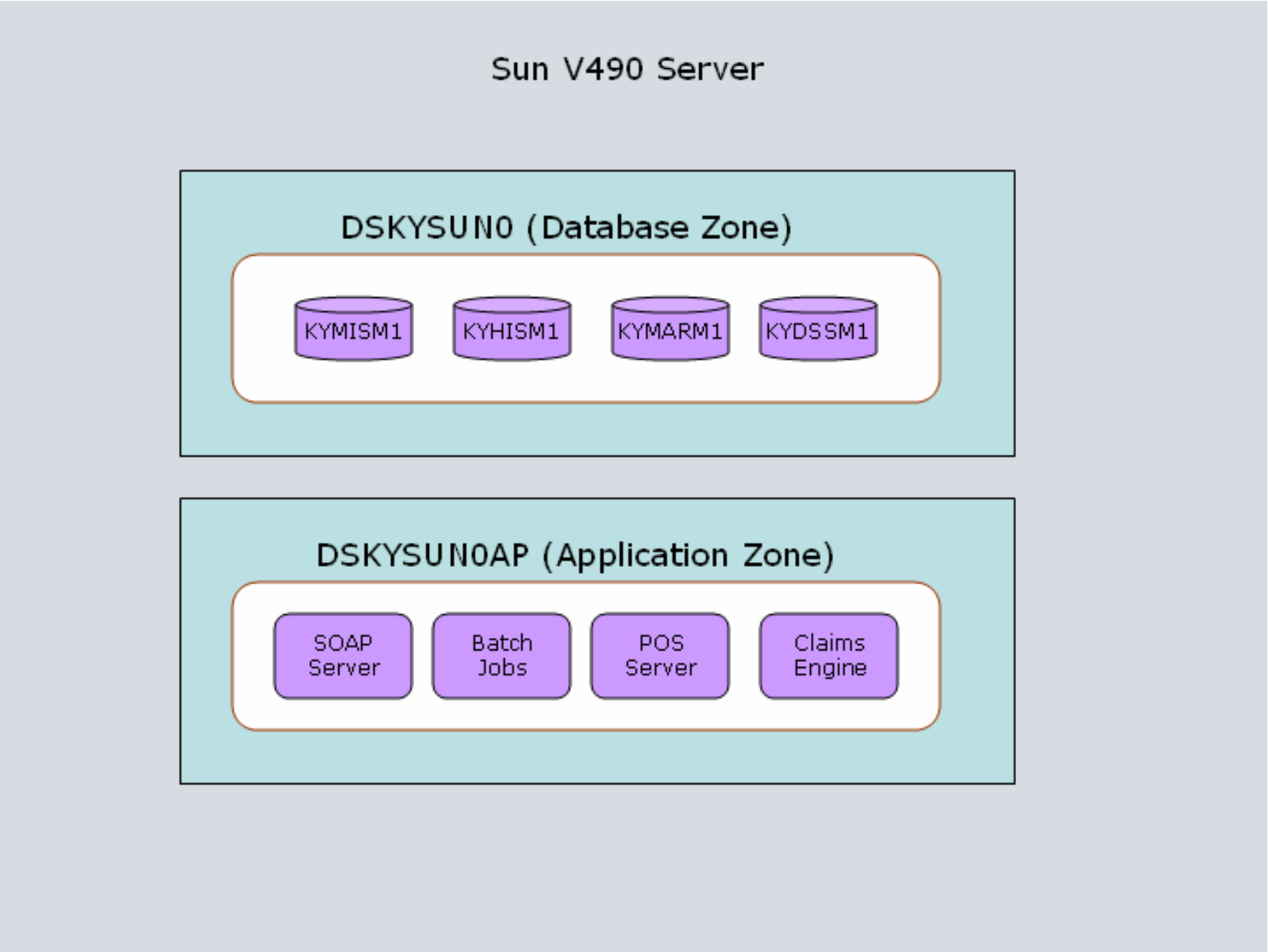
Replication snapshots will be scheduled at least once per day. The snapshots will occur based on application recovery points. This will ensure there is a consistent application restart time. At least three (3) snapshots will be retained on disk at the Herndon site. This will allow for multiple recovery points in case the latest is not usable for any reason. In addition, a weekly tape backup of the latest snapshot will be taken at the Herndon site. This will provide the availability of additional snapshots in case they are needed. It will also provide a backup on tape if something should happen to the Herdon SAN. The tape containing the backups will be stored in the tape library. They will not be sent offsite.

2.2.3.6 SUN V490

The SUN V490 server (depicted in the diagram on the next page) is the DDI model office server. This server is logically split into two separate “Zones”. Zones provide a new isolation primitive for the Solaris OS, which results in secure, flexible, scalable and lightweight virtualized OS services which behave like different Solaris instances. Together with the existing Solaris Resource management framework, Solaris Zones form the basis of Solaris Containers. Each zone is a "Virtual Machine". The SUN V490 server contains two zones: DSKYSUN0 and DSKYSUN0AP.

The DSKYSUN0 zone is the database zone. This zone manages the model office databases for OnBase (KYONBT1), ESRI (KYSDET1) and interChange (KYMISM1) including online, history, MAR, and DSS and SUR.

The DSKYSUN0AP zone is the application zone. This zone manages all interChange executables for both batch and interactive processing.


Sun V490 Server
DSKYSUN0 (Database Zone)
KYMISM1
KYHISM1
KYMARM1
KYDSSM1
DSKYSUN0AP (Application Zone)
SOAP Server
Batch Jobs
POS Server
Claims Engine

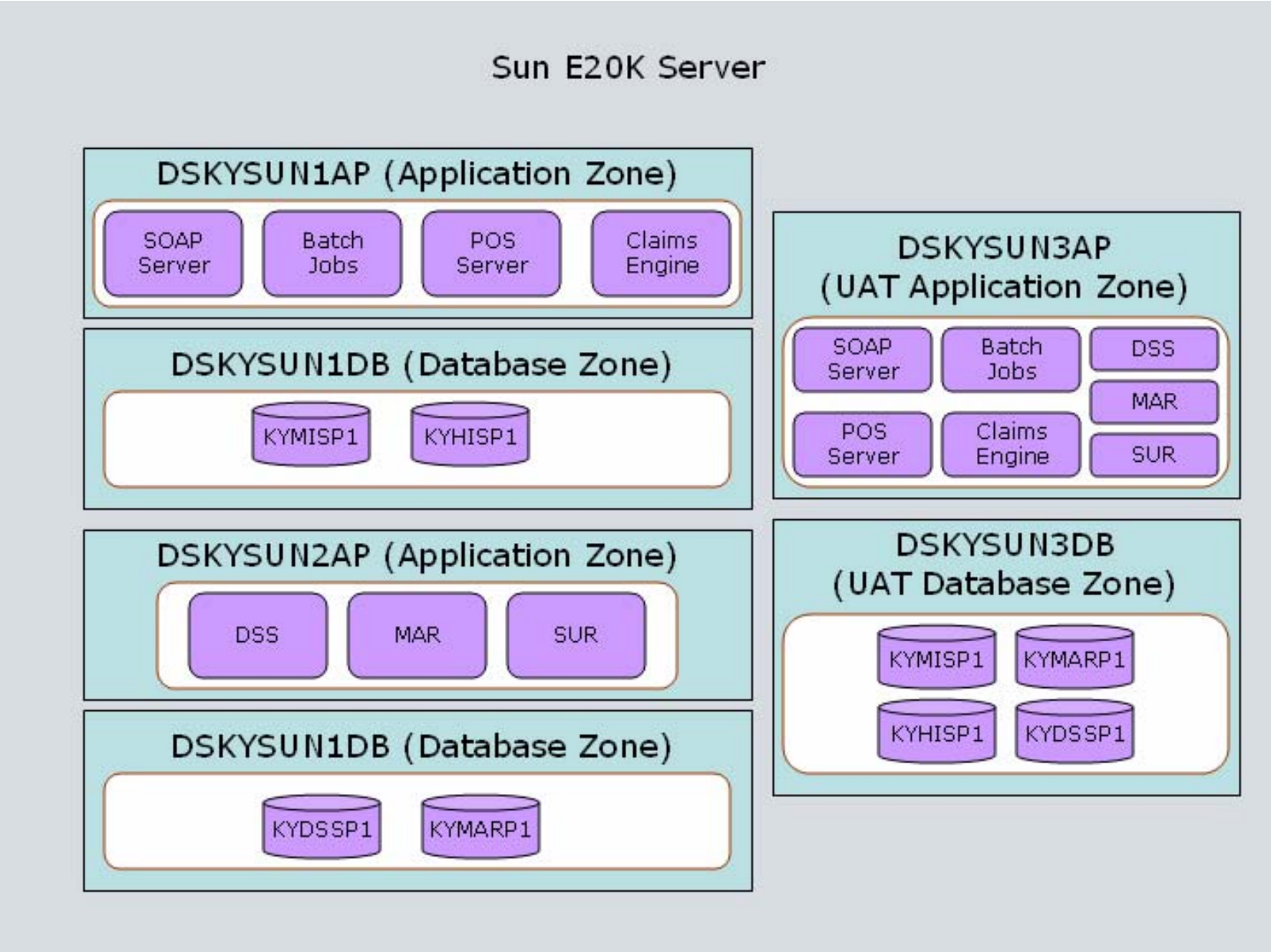
2.2.3.7 SUN-E20K

The SUN E20K server (depicted in the diagram on the next page) is the DDI production server. This server is logically split into six separate “Zones”. Generally, these zones are production, UAT and DSS. Each of these divisions is further split into application and database zones.

The production interChange zones are DSKYSUN1AP for the applications and DSKYSUN1DB for the databases. The application zone (DSKYSUN1AP) runs all of the interChange C/UNIX executables for the batch and interactive processes. This includes the SOAP server processes that receive interactive requests from the Internet website (provider portal). The database zone (DSKYSUN1DB) runs the online and history databases.

The DSS zones are DSKYSUN2AP for the DSS applications and DSKYSUN2DB for the DSS database.

The UAT zones combine online, history, and dss zones. UAT is separated into an application zone (DSKYSUN3AP) and a database zone (DSKYSUN3DB).


Sun E20K Server
DSKYSUN1AP (Application Zone)
SOAP Server
Batch Jobs
POS Server
Claims Engine
DSKYSUN1DB (Database Zone)
KYMISP1
KYHISP1
DSKYSUN2AP (Application Zone)
DSS
MAR
SUR
DSKYSUN1DB (Database Zone)
KYDSSP1
KYMARP1
DSKYSUN3AP (UAT Application Zone)
SOAP Server
Batch Jobs
DSS
MAR
POS Server
Claims Engine
SUR
DSKYSUN3DB (UAT Database Zone)
KYMISP1
KYMARP1
KYHISP1
KYDSSP1

2.3 Processes

2.3.1 Correspondence, Letter, and Notice Management Process Model

2.3.1.1 Process Description

This process generates official notices, letters, and correspondence of all types. Correspondence letters and/or notices are generated by the KY MMIS when a letter generating event occurs. Each subsystem has the definition of its letters and when they would be generated. Other correspondence letters and/or notices are generated when directed by a KY MMIS user.

2.3.1.2 interChange Process Components

The correspondence, letter, and notice management process is supported by a combination of KY MMIS pages/panels, an integrated letter generator solution within interChange, and batch processes. Details on the letter generator can be found in Appendix A & B.

2.3.1.3 KY MMIS Pages/Panels

The KY MMIS pages/panels are intended for limited use by systems support personnel maintaining the letter generation functions of the KY MMIS. End-user letter functions are performed within each functional area by customized panels or batch processes.

The "Reports and Letters" Panel within a subsystem displays the subsystem letters. Click on a letter to open a search panel. Enter the search criteria to display a data list. The row on the data list gives the details of the persons receiving the letter. Clicking on any row opens the Letter Request page of Letter Generator and the letter is generated automatically.

2.3.1.4 Letter Generator

The letter generator defines letter templates and the data from the interChange data model that is required to populate the letter templates. The EDS support staff and select KY MMIS power users have access to this application.

The Letter Query Maintenance Panel is used to define the database query that retrieves the data from the interChange database used to create the letter. This panel is accessed from the Send/Compose hyperlink located on the Main Web Menu. The data is retrieved as an XML document and merged into the letter template. The XML document created by the Letter Query is the dynamic portion of the letter template.

The Letter Template Maintenance panel defines the static text for the letter and is accessed from the Main Web Menu using the Templates hyperlink. The letter template is defined using HTML. XSD statements insert the dynamic data from XML document query. The merging of the Letter Query's XML document (dynamic data) with the Letter Template's HTML (static data) is completed either in an interactive online mode which is displayed in Microsoft Word, or it is completed in a batch mode to be routed for print and archival.

Letter Type:

The Letter Generator is designed to operate as a fully automated system on web-based interaction and batch generation modes. The letter type is determined based on which method would be used to generate the letter - Web interface or batch generation. The type of the letter can be specified while creating the template. There is a drop-down menu "Batch Generation". If the option "Yes" is chosen the letter is generated in batch. Both generation methods operate independently of each other. The web interface is available 24 hours per day, provided the MMIS database is available. The batch generator executes at predefined times throughout the day, or can also execute continuously as requests for letter correspondence are generated.

Each generation method executes predefined data queries based on each letter's design and populated identification values against the MMIS database. The data then is merged into the letter template to produce the personalized letter correspondence to the end party.

2.3.1.5 Batch Processes

Batch letters can be requested from batch processes or online for batch generation. To request a letter, the process or panel enters the request for the letter on the Letter Request panel. The request includes key information for the Letter Query including the date the letter should be generated and the name of the Letter Template. These letters are generated during the batch cycles. Letters can be defined for generation in the interChange batch cycle in postscript format and/or letters can be generated on the mail merge server in MS Word document format. The letters are grouped in separate print files by letter type.

2.3.1.6 Archival and Online Retrieval

The letters output from the batch processes are copied to COLD/OnBase for online retrieval, keyed by the defined business key (member ID, SSN, provider number, etc.).

2.3.1.7 Correspondence, Letter And Notice Management Objects

The table below contains the objects associated with Correspondence, Letter, and Notice Management for System Wide subsystem.

Technical Name	Title	Type
SYSJDLTR	Batch Letter Generator	Batch Job
lgrp.aspx	Letter Group Maintenance	Panel
LetterQuery.asp	Letter Query Maintenance	Panel
LetterRequest.aspx	Letter Request	Panel
ltmaintedit.aspx	Letter Template Maintenance	Panel
LetterTracking.aspx	Letter tracking	Panel
ltmaint.aspx	Letters Menu	Panel
LetterPromo.aspx	Promotion of letter to different environment	Panel

2.3.1.8 Data Warehouse Objects

No Data Warehouse objects are associated with the System Wide subsystem.

2.3.1.9 Database Utilities

The Database Utilities are a set of common utilities that allow the programmer to manage large amounts of data in the database tables.

Technical Name	Title	Type
dbdelete	Delete Rows From a Database Table	Program
dbunload	Unload Database Tables Based on a Query	Program
oraload.sh	Oracle Table Load-Unload Utility	Program

2.3.1.10 Performance Utilities

The System Wide subsystem includes three utilities to enhance the performance of applications.

The oraload utility speeds up the movement of data from one database to another. It uses advanced techniques for the loading and unloading of data.

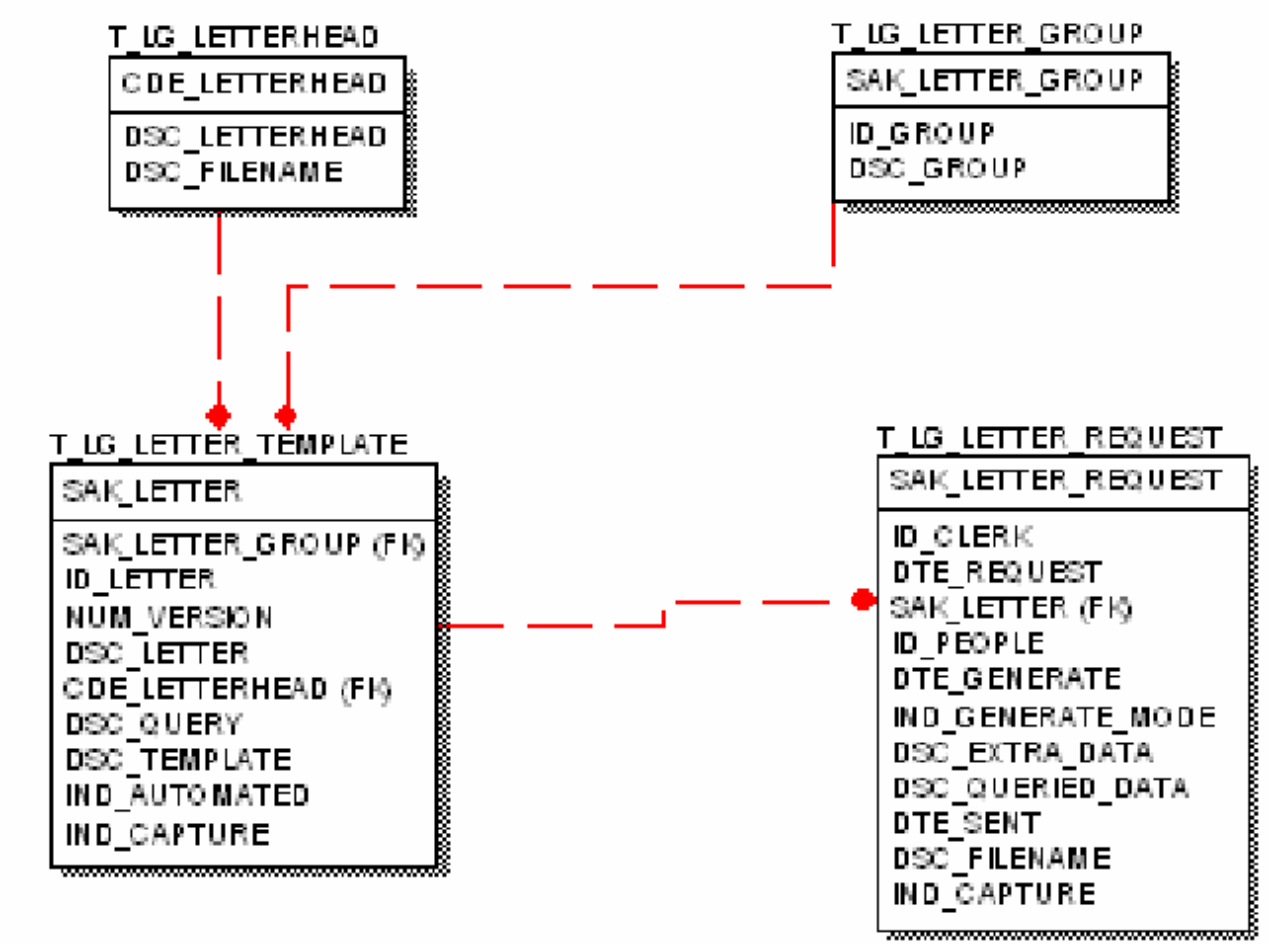
The clmphash utility indexes data that can be mapped into memory during the execution of an application.

The dbdelete utility deletes rows of data from a table.

The following table lists the common utilities supported by the System Wide subsystem.

Technical Name	Title	Type
clmphash	Build Hash Index	Program
dbdelete	Delete Rows From a Database Table	Program
dbunload	Unload Database Tables Based on a Query	Program
oraload.sh	Oracle Table Load-Unload Utility	Program

2.4 Letter Generator Data Model



2.4.1 External System Interfaces

System Wide subsystem interfaces are detailed in the KY MMIS Interoperability Plan on the Project Workbook Project Deliverables page. Access it as follows:

- Click “Program Management” link on left side of PWB home page.
- Click “Project Deliverables” link.
- Click “DSD Refresh” under “Deliverable Documents- DDI”

The Interoperability Plan will be located in this directory.

2.5 System Wide Subsystem Job Scripts

2.5.1 Autosys job - Autosys

2.5.1.1 Detailed Job Script Information

Unix Script:	Autosys job
Description:	This job script executes inside a parent autosys process. Information on autosys jobs and schedules can be found at ddipwb.kymmis.com/KYXIXDDI/subsystem/cycle.

2.5.2 watch_file - Autosys File Watcher

2.5.2.1 Detailed Job Script Information

Unix Script:	watch_file
Description:	This job script executes the file watcher process of autosys that looks for a file based on parameters specified in the autosys job.

2.5.3 AIMJD010 - Change cycle date parm

2.5.3.1 Detailed Job Script Information

Unix Script:	AIMJD010
Description:	This job script updates the system parm table with the date of the start of the cycle.
Job Step:	js 010-aimp010d
Description:	This job step executes the AIMP010D.
Input/Output Files: None	

2.5.4 AIMJW010 - Change cycle date parm

2.5.4.1 Detailed Job Script Information

Unix Script:	AIMJW010	
Description:	This job script executes weekly aim job updates of the system parms table with the range of dates for the weekly cycle.	
Job Step:		js 010-aimp010w
Description:		This job step executes AIMP010W.
Input/Output Files:		
naterror.aim01001.log	O	asd

2.5.5 SYSJD130 - Clean the logs directory of files that are over one day old

2.5.5.1 Detailed Job Script Information

Unix Script:	SYSJD130
Description:	This job script executes the maintenance of the specified directory by moving the members that are older than one day to the archive directory. This SYSJD130 job executes daily just after midnight to handle the maintenance a specified directory by moving the files in that directory that are older than one day to the archive directory. When files in the archive directory are greater than seven days old they are moved to the Archive folder. The primary purpose is to clean up log directories.
Job Step:	js 010-archive_logs.sh
Description:	This job step executes the clean up of the specified directory.
Input/Output Files: None	

2.5.6 SYSJD110 - Clean the prod override directory of members that are over three days old

2.5.6.1 Detailed Job Script Information

Unix Script:	SYSJD110
Description:	This job script executes the clean up of the prod override directory of members that are over three days old.
Job Step:	js 010-override_maint.sh
Description:	This job step executes the clean up of the specified override directory.
Input/Output Files: None	
Input/Output Files: None	

2.5.7 AIMJD999 - Flag cycle date when all jobs are complete

2.5.7.1 Detailed Job Script Information

Unix Script:	AIMJD999	
Description:	This job script executes the table to indicate that the daily cycle has finished. Job script js10 executes a program that flags the cycle as being complete. Abend Instructions: js10 - Simply restart job at js10.	
Job Step:		js 010-aimp999d
Description:		This job step executes the AIMP999D.
Input/Output Files:		
naterror.aim01001.log	O	asd

2.5.8 AIMJM010 - Maintain Monthly Cycle Date

2.5.8.1 Detailed Job Script Information

Unix Script:	AIMJM010	
Description:	This job script executes the monthly aim job updates of the system parms table with the range of dates for the monthly cycle.	
Job Step:		js 010-aimp010m
Description:		This job step executes the AIMP010M.
Input/Output Files:		
naterror.aim01001.log	O	asd

2.5.9 AIMJQ010 - Maintain Quarterly Cycle Date

2.5.9.1 Detailed Job Script Information

Unix Script:	AIMJQ010	
Description:	This job script executes the quarterly aim job updates of the system parms table with the range of dates for the quarterly cycle.	
Job Step:		js 010-aimp010q
Description:		This job step executes the AIMP010Q.
Input/Output Files:		
naterror.aim01001.log	O	asd

2.5.10 CHKJDOVR - Produces the $PRODDIR/override activity report

2.5.10.1 Detailed Job Script Information

Unix Script:	CHKJDOVR	
Description:	This job script creates a report of activity in the override directories for the previous day.	
Job Step:		js 010-rcp
Description:		This job step copies the change.log file from sun1 to sun2.
Input/Output Files:		
change.log	I	This file contains the change log file.
change.log.dscfsun0	O	This file contains the change log sun1 file.
Job Step:		js 020-rcp
Description:		This job step copies the change.log file from sun3 to sun2.
Input/Output Files:		

change.log	I	This file contains the change log file.
change.log.sun3	O	This file contains the change log sun3 file.
Job Step:		js 030-cp
Description:		This job step copies the change.log file on sun2 to match names for sun1 and sun3 file.
Input/Output Files:		
change.log	I	This file contains the change log file.
change.log.sun2	O	This file contains the change log sun2 file.
Job Step:		js 040-echo
Description:		This job step sends previous date in the following format: Mon Sep 2 hh:mm:ss EST 1996.
Input/Output Files:		
prevdate	I	This file contains the previous date.
currdt.$cdate	O	This file contains the current file.
Job Step:		js 050-grep
Description:		This job step isolates the name of the day, the month and the number of the day.
Input/Output Files:		
currdt.$cdate	I	This file contains the current file.
currdte.$cdate	O	This file contains the current date file.
Job Step:		js 060-egrep
Description:		This job step greps for the specified date in each of the change.log files.
Input/Output Files:		
change.log.dscfsun0	I	This file contains the change log sun1 file.
change.log.sun2	I	This file contains the change log sun2 file.
change.log.sun3	I	This file contains the change log sun3 file.
currdte.$cdate	I	This file contains the current date file.
currora.$cdate	O	This file contains the current date file.
Job Step:		js 070-egrep

Description:		This job step checks for processing errors exit (-1) endif js70 - greps for the specified date in each of the change.log files.
Input/Output Files:		
currora.$cdate	I	This file contains the current date file.
currorb.$cdate	O	This file contains the current date file.
Job Step:		js 080-mail
Description:		This job step sends a Pathway Mail of the cycle stats.
Input/Output Files:		
currorb.$cdate	I	This file contains the current date file.
pathway mail	O	This file contains the Send Pathway Mail.

2.5.11 SYSJSOAP_START - SYSJSOAP_START

2.5.11.1 Detailed Job Script Information

Unix Script:	SYSJSOAP_START
Description:	This job script starts the SOAP server each morning at 6am.
Input/Output Files:	None

2.5.12 SYSJD120 - Send the email report for the production overrides.

2.5.12.1 Detailed Job Script Information

Unix Script:	SYSJD120
Description:	This job script creates an e-mail to send to the primary cycle monitor and leaders with current information about the override directory status.
Job Step:	js 010-mail
Description:	This job step mails the message.
Input/Output Files:	None

2.5.13 SYSJDCHSE - System Check of Real-Time Processes

2.5.13.1 Detailed Job Script Information

Unix Script:	SYSJDCHSE
Description:	This job script checks to see if a specific set of real-time processes are up and running. A parm is passed to the job to indicate which real-time process to check. This includes checks for DAEMON, POS, SOAP and NCPDP.
Job Step:	js 010-sysjdchse.ksh

Description:	This job step checks the servers.
Input/Output Files:	None

2.6 Programs

2.6.1 COLD Report Storage and Indexing -- COLD Report Storage and Indexing

Technical Name:	COLD Report Storage and Indexing
Program Title:	COLD Report Storage and Indexing.
Programming Language:	UNIX Script
Description:	Reports that must be stored and indexed on COLD.

2.6.1.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.2 SoapSrvrMgr.sh -- Start, stop the SOAP servers based on the input parameter.

Technical Name:	SoapSrvrMgr.sh
Program Title:	Start or stop the SOAP servers based on the input parameter.
Programming Language:	UNIX Script
Description:	This script starts, stops, or bounces (stops and restarts) the SOAP servers based on the input parameter. Based on the ID executing this script, the environment is established and the servers are started or stopped, or bounced.

2.6.2.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.3 aimp010d -- Update parms table cycle start date

Technical Name:	aimp010d
Program Title:	Update parms table cycle start date
Programming Language:	C
Description:	To update the system parms table with the date of the start cycle. **Input Parameters:** PROGRAM **Exit Values:** EXIT_SUCCESS - Normal termination EXIT_FAILURE - Abnormal termination because of open, read, allocation, or input errors. **Input Files:** None **Output Files:** None **Input Tables:** T_SYSTEM_PARMS **Output Tables:** T_SYSTEM_PARMS **Sort Criteria:** None **Switches:** None **Link Procedure:** libnatural, libxml, and libgutils **Special Logic Notes:** None

2.6.3.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.4 aimp010m -- aimp010m

Technical Name:	aimp010m
Program Title:	aimp010m
Programming Language:	C
Description:	To update the system parms table with the date of the start of the monthly cycle. **Input Parameters:** None **Exit Values:** EXIT_SUCCESS - Normal termination EXIT_FAILURE - Abnormal termination because of open, read, allocation, or input errors. **Input Files:** None **Output Files:** None **Input Tables:** T_SYSTEM_PARMS **Output Tables:** T_SYSTEM_PARMS **Sort Criteria:** None **Switches:** None **Link Procedure:** libnatural, libxml, and libgutils **Special Logic Notes:** None

2.6.4.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.5 aimp010q -- aimp010q

Technical Name:	aimp010q
Program Title:	aimp010q
Programming Language:	C
Description:	To update the system parms table with the date of the start of the quarterly cycle. **Input Parameters:** None **Exit Values:** EXIT_SUCCESS - Normal termination EXIT_FAILURE - Abnormal termination because of open, read, allocation, or input errors. **Input Files:** None **Output Files:** None **Input Tables:** T_SYSTEM_PARMS **Output Tables:** T_SYSTEM_PARMS **Sort Criteria:** None **Switches:** None **Link Procedure:** libnatural, libxml, and libgutils **Special Logic Notes:** None

2.6.5.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.6 aimp010w -- aimp010w

Technical Name:	aimp010w
Program Title:	aimp010w
Programming Language:	C
Description:	To update the system parms table with the date of the start of the weekly cycle. **Input Parameters:** None **Exit Values:** EXIT_SUCCESS - Normal termination EXIT_FAILURE - Abnormal termination because of open, read, allocation, or input errors. **Input Files:** None **Output Files:** naterror.aim01001.log **Input Tables:** T_SYSTEM_PARMS **Output Tables:** T_SYSTEM_PARMS **Sort Criteria:** None **Switches:** None **Link Procedure:** libnatural, libxml, and libgutils **Special Logic Notes:** None

2.6.6.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.7 aimp999d -- aimp999d

Technical Name:	aimp999d
Program Title:	aimp999d
Programming Language:	C
Description:	To update the system parms table to indicate that the cycle is complete. **Input Parameters:** T_SYSTEM_PARMS **Exit Values:** EXIT_SUCCESS - Normal termination EXIT_FAILURE - Abnormal termination because of open, read, allocation, or input errors. **Input Files:** EPSJA None **Output Files:** EPSJA naterror.aim01001.log **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None **Link Procedure:** libnatural, libxml, and libgutils **Special Logic Notes:** None

2.6.7.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.8 archive_logs.sh -- Move log file archives

Technical Name:	archive_logs.sh
Program Title:	Move log file archives.
Programming Language:	UNIX Script
Description:	This script handles the maintenance of any directory and move files that are not directories to an archive directory after they are over one day old and leave them in archive for seven days. Archive files are removed from the system when they are over seven days old.

2.6.8.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.9 autorep -- autorep

Technical Name:	autorep
Program Title:	autorep
Programming Language:	UNIX Script
Description:	Command to report information about AutoSys jobs and boxes. **Input Parameters:** -J job_name - Indicates that a Job Report is desired. job_name specifies the name of the job (including boxes) on which to report. Any valid AutoSys job name may be specified. To report on all jobs, specify ALL. The % character may be used in the job_name as a wildcard (for example; %box% selects all jobs containing the string box). -M machine_name - Indicates that a Machine Report is desired, which lists the machine s Max Load, Current Load, and Factor. machine_name specifies the machine on which to report. This may be a virtual machine, a real machine, or ALL for all machines; the machine must be defined to AutoSys. -G global_name - Indicates that a global variable report is desired; listing the variable name, value, and last modification ate. global_nam. **Exit Values:** 0 - Normal termination. Other - Abnormal termination because of open, read, allocation, or input errors. **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None **Link Procedure:**

	None **Special Logic Notes:** None

2.6.9.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.10 biztalk -- Biz Talk

Technical Name:	biztalk
Program Title:	Biz Talk.
Programming Language:	Other.
Description:	This is the utility used to format data between Kentucky Member Management and interChange Member. BizTalk Server 2006 is Microsoft's premiere server for building solutions for business process and integration. EDS will install and maintain BizTalk in both the OIT ColdHarbor and EDS Orlando Data centers. The OIT ColdHarbor installation will serve as a data translation point between multiple organizations. The EDS Orlando Installation will be used to translate the output of the EDIFECS, Inc. HIPAA compliance product into the XML format needed by interChange.

2.6.10.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.11 clean_pid_files -- Cleanup SOAP work files

Technical Name:	clean_pid_files
Program Title:	Cleanup SOAP work files.
Programming Language:	Perl.
Description:	Removes SOAP work files in /tmpsort/SOAPsrvr.

2.6.11.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.12 clmphash -- Build Hash Index

Technical Name:	clmphash
Program Title:	Build Hash Index
Programming Language:	C
Description:	This is an executable module which creates a hash index on a set of data files. A parm_file is used to document the data files to be hashed and the index file to be built. Please read the document Memory Mapping Routines for more detailed documentation on this utility. **Input Parameters:** None **Exit Values:** EXIT_SUCCESS - Normal termination EXIT_FAILURE - Abnormal termination because of open, read, allocation, or input errors. **Input Files:** MARJQ990 mr_pr_svc_loc.dat mrq99002.ctl t_clm_pgm_xref.dat t_diagnosis.dat t_diagnosis.out t_modifier.dat t_modifier.out t_pmp_svc_loc.dat t_pr_prov.dat t_pr_type.dat t_pr_type.out t_proc.dat t_proc.out t_re_base.dat t_re_pmp_assign.dat SRGJW338 eombidx.ctl SRGJW348 eombidx.ctl SRGJW350 srg3idx.ctl SRGJW370 srg5idx.ctl SRGJW330 srg1idx.ctl **Output Files:** MARJQ990

	mr_pr_svc_loc.idx t_clm_pgm_xref.idx t_diagnosis.idx t_modifier.idx t_pmp_svc_loc.idx t_pr_prov.idx t_pr_type.idx t_proc.idx t_re_base.idx t_re_pmp_assign.idx **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None **Link Procedure:** None **Special Logic Notes:** None

2.6.12.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.13 copy2crld -- CRLD copy

Technical Name:	copy2crld
Program Title:	CRLD copy.
Programming Language:	UNIX Script
Description:	This is a script to copy report files to crld dir. **Input Parameters:** The copy2crld requires 2 arguments: (1) the source report filename (2) the report name **Exit Values:** 0 - Normal termination Other - Abnormal termination because of open, read, allocation, or input errors. **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None **Link Procedure:** None **Special Logic Notes:** None

2.6.13.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.14 copy2routedir -- Copy Reports to Router

Technical Name:	copy2routedir
Program Title:	Copy Reports to Router.
Programming Language:	UNIX Script
Description:	This is a script to copy report files and route to the printer or COLD. **Input Parameters:** Name of report file to be copied **Exit Values:** 0 - Normal termination Other - Abnormal termination because of open, read, allocation, or input errors. **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None **Link Procedure:** None **Link Procedure:** None **Special Logic Notes:** The script sleeps 60 seconds if a lock file exists.

2.6.14.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.15 cycle_email.sh -- get the email addresses for the cycle monitors

Technical Name:	cycle_email.sh
Program Title:	Get the email addresses for the cycle monitors.
Programming Language:	UNIX Script
Description:	This script gets the email addresses for the cycle monitor, cycle leaders, and cycle others from the /opt/utils/ops/data/oncall.beeper.list file header and returns the email addresses to the calling script in the following environmental variables: CYCLE_MONITOR CYCLE_LEADERS CYCLE_OTHERS To use this script you need to put the following command in your csh script; "source cycle_email.sh".

2.6.15.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.16 dbdelete -- Delete rows from a database table

Technical Name:	dbdelete
Program Title:	Delete rows from a database table.
Programming Language:	C
Description:	This is a generic utility that removes rows from a database table based on a list of sak values. The program uses an arrayed delete algorithm to optimize performance of the utility. **Program Usage:** dbdelete [-s] table_name sak_file ignore_not_found(y or n) -s: silence **Input Parameters:** table_name - The table name sak_file - The File ID ignore_not_found (y or n) - If should ignore not found (y or n) **Exit Values:** EXIT_SUCCESS - Normal termination EXIT_FAILURE - Abnormal termination because of open, read, allocation, or input errors. **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None **Link Procedure:** None **Special Logic Notes:** None

2.6.16.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.17 dbunload -- Utility to unload database tables based on a query

Technical Name:	dbunload
Program Title:	Utility to unload database tables based on a query.
Programming Language:	C
Description:	This is a generic utility that can be used to execute a SQL query against the database and place the result into a data file. This program is typically used as part of the oraload.sh scripts. Occasionally it is used as a standalone program. **Program Usage:** dbunload -a arraysize [-t] [-c] [-n] [-v] [-b] [-f table_defn_file] table_name output_dest_file -t - add a record terminator (newline) -c - concat (append) to output file -n - handle NULL values correctly (adds 1 byte per column per row) -v - handle VARCHAR2 columns correctly (adds 2 bytes per column per row) -b - boundary alignment (not implemented yet) -y - unload in ASCII characters (convert all numbers) (not implemented yet) (the table_defn_file defaults to stdout if not specified) (If table_name is "-" the select statement is read from stdin) **Input Parameters:** -t - add a record terminator (newline) -c - concat (append) to output file -n - handle NULL values correctly (adds 1 byte per column per row) -v - handle VARCHAR2 columns correctly (adds 2 bytes per column per row) -b - boundary alignment (not implemented yet) -y - unload in ASCII characters (convert all numbers) (not implemented yet) The table_defn_file defaults to stdout if not specified If table_name is "-" the select statement is read from stdin Output_dest_file is the name of the output file **Exit Values:** 0 - Normal termination Other - Abnormal termination because of open, read, allocation, or input errors. **Input Files:** None **Output Files:** None

Technical Name:	dbunload
	Input Tables: None **Output Tables:** None **Sort Criteria:** None **Switches:** None **Link Procedure:** None **Special Logic Notes:** None

2.6.17.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.18 dumppos -- Dump POS TXNS

Technical Name:	dumppos
Program Title:	Dump POS txns.
Programming Language:	C
Description:	This program is used to dump POS transactions from the poschild.log files for debugging purposes.

2.6.18.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.19 eoj_msg.csh -- Format the end of job message

Technical Name:	eoj_msg.csh
Program Title:	Format the end of job message.
Programming Language:	UNIX Script
Description:	Formats the end of job message.

2.6.19.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.20 execpgm.csh -- Execute programs from batch jobs

Technical Name:	execpgm.csh
Program Title:	Execute programs from batch jobs.
Programming Language:	UNIX Script
Description:	This shell is called from the AIM batch job scripts to execute commands.

2.6.20.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.21 exit -- UNIX shell exit comment

Technical Name:	exit
Program Title:	UNIX shell exit comment.
Programming Language:	UNIX Command
Description:	Exit job. **Input Parameters:** n - Value to be returned (optional) **Exit Values:** Value specified by n Exit status of the last command executed if n is omitted. **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None **Link Procedure:** None **Special Logic Notes:** None

2.6.21.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.22 ftp -- File transfer

Technical Name:	ftp
Program Title:	File transfer.
Programming Language:	UNIX Command
Description:	File transfer.

2.6.22.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.23 gdg -- Generated Data Group

Technical Name:	gdg
Program Title:	Generated Data Group.
Programming Language:	C
Description:	Generates a data group of a designated number of members that is maintained for a particular file name. **Input Parameters:** GdgFile **Exit Values:** 0 - Normal termination Other - Abnormal termination because of open, read, allocation, or input errors. **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None **Link Procedure:** None **Special Logic Notes:** None

2.6.23.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.24 grant.revoke.privs.ks -- grant.revoke.privs.ks

Technical Name:	grant.revoke.privs.ks
Program Title:	grant.revoke.privs.ks
Programming Language:	UNIX Script
Description:	System utility to grant or revoke privileges to a table.

2.6.24.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.25 jil -- Autosys

Technical Name:	jil
Program Title:	Autosys.
Programming Language:	Other (Autosys Job Information Language)
Description:	This is the Autosys job scheduling utility.

2.6.25.1 Change Orders

ID	NAME	DESCRIPTION
73	Configure Autosys for Member	There is a need to configure Autosys for use with the Commonwealth batch interfaces.

2.6.26 job_begin.src -- Common Job Script initialization

Technical Name:	job_begin.src
Program Title:	Common Job Script initialization.
Programming Language:	UNIX Script
Description:	Common Job Script initialization. Called at the beginning of each job script. Options are: -o [override filename] -r [restart step]

2.6.26.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.27 jsbeg_msg.csh -- Formats the beginning of JobStep message

Technical Name:	jsbeg_msg.csh
Program Title:	Formats the beginning of JobStep message.
Programming Language:	UNIX Script
Description:	Formats the beginning of JobStep message.

2.6.27.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.28 letter_requests -- letter_requests

Technical Name:	letter_requests
Program Title:	letter_requests
Programming Language:	C
Description:	This program links the on-line letter request to the associated letter template.

2.6.28.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.29 letter_templates -- letter_templates

Technical Name:	letter_templates
Program Title:	letter_templates
Programming Language:	C
Description:	This utility takes the XML data and inserts it into the requested template.

2.6.29.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.30 libclmhash.so -- Hash and Parse Utilities

Technical Name:	libclmhash.so
Program Title:	Hash and Parse Utilities.
Programming Language:	C
Description:	The executable that contains all of the logic necessary to create a hash index on a set of data files. libclmhash.so.mak contains: clmshash.c clmspars.c dummyClaimCheck.c

2.6.30.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.31 libded.so -- look ups for the DED

Technical Name:	libded.so
Program Title:	Look ups for the DED.
Programming Language:	C
Description:	This is a routine for code value look up for the DED listing.

2.6.31.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.32 liblbms.so -- LBMS utility functions

Technical Name:	liblbms.so
Program Title:	LBMS utility functions.
Programming Language:	C
Description:	These are the standard utilities developed to aid in the conversion from LBMS into C code.

2.6.32.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.33 libmemhash.so -- Memory hash functions

Technical Name:	libmemhash.so
Program Title:	Memory hash functions.
Programming Language:	C
Description:	Routines for memory hash manipulation.

2.6.33.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.34 libmrpf.so -- Report writing functions library

Technical Name:	libmrpf.so
Program Title:	Report writing functions library.
Programming Language:	C
Description:	This library contains the functions that standardize the creation of formatted reports.

2.6.34.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.35 libnatural.so -- Support functions for natural

Technical Name:	libnatural.so
Program Title:	Support functions for natural.
Programming Language:	C
Description:	Support functions for programs converted from natural.

2.6.35.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.36 libSOAPsave.so -- Save files through SOAP

Technical Name:	libSOAPsave.so
Program Title:	Save files through SOAP.
Programming Language:	C
Description:	Routines for saving objects through the SOAP server.

2.6.36.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.37 libsysltr.so -- Create entry in letter request table

Technical Name:	libsysltr.so
Program Title:	Create entry in letter request table.
Programming Language:	C
Description:	This routine inserts an entry into the letter request table.

2.6.37.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.38 libxml.so -- XML Processing

Technical Name:	libxml.so
Program Title:	XML Processing.
Programming Language:	C
Description:	The executable that contains all of the logic necessary to process an XML input file. libxml.so.mak contains: XMLDOM.c XPathGrammer.y XPath.c XSL.c XMLSchema.c XMLSchemaParse.c XMLSchemaCreate.c XMLSchemaValidate.c XMLSchemaLoad.c XMLSchemaUnload.c XMLSchemaGenerate.c XMLLib.c XMLC.c XMLSqlUnload.c ERWin.c Soap.c HTTPSocket.c dhash.c list.c value.c DBOracle.sc hmspconn.sc db_oci.c log.c SQL.c SQL.y SQL.l tools.c

2.6.38.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.39 libxml_nomt.so -- XML Processing

Technical Name:	libxml_nomt.so
Program Title:	XML Processing.
Programming Language:	C
Description:	The executable that contains all of the logic necessary to process an XML input file. (No -mt in DEBUG) libxml_nomt.so.mak contains: XMLDOM.c XPathGrammer.y XPath.c XSL.c XMLSchema.c XMLSchemaParse.c XMLSchemaCreate.c XMLSchemaValidate.c XMLSchemaLoad.c XMLSchemaUnload.c XMLSchemaGenerate.c XMLLib.c XMLC.c XMLSqlUnload.c ERWin.c Soap.c HTTPSocket.c dhash.c list.c value.c DBOracle.sc hmspconn.sc db_oci.c SQL.c SQL.y SQL.l UNIXDOTNET.c tools.c

2.6.39.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.40 libxmlmrge.so -- Query Unload and Merge

Technical Name:	libxmlmrge.so
Program Title:	Query Unload and Merge.
Programming Language:	C
Description:	The executable that contains all of the logic necessary to unload and merge a hierarchical query. libxmlmrge.so.mak contains: xmlsmrge.c

2.6.40.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.41 massdrup.sh -- massdrup.sh

Technical Name:	massdrup.sh
Program Title:	massdrup.sh
Programming Language:	UNIX Script
Description:	Mass update. **Input Parameters:** -i input directory -o output directory -s control file listing sak files **Exit Values:** 0 - Normal termination Other - Abnormal termination because of open, read, allocation, or input errors. **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None **Link Procedure:** None **Special Logic Notes:** None

2.6.41.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.42 oraload.sh -- Oracle Table Load-Unload Utility

Technical Name:	oraload.sh
Program Title:	Oracle Table Load-Unload Utility.
Programming Language:	UNIX Script
Description:	This is a UNIX utility used to load and unload Oracle database tables. This utility makes use of the dbunload and sqlldr programs to push data in and out of the database. **Program Usage:** oraload.sh -a array_size -bceint -m -s sysin_control_file -f ctl_fyl_loc -h hash_ctl_cards -a array size -b do an export and then an import -c only perform conventional loads -d only display how the control file is split up -e export only -f already generated sqlload ctl file location -h control cards to use when building the hash indexes -i import only -m indicates to execute the loads/unloads in parallel groups -p handle null columns properly -o the schema of the tables being exported/imported -s control file listing tables to be loaded/unloaded -t do not truncate any tables -u unrecoverable option -v use schema name as subdir on $DATADIR NOTE: The following environment variables must be set prior to running this script: FROM_PSWD --> specify the database to unload from in the format "user/id@database" TO_PSWD --> specify the database to load into in the format "user/id@database" **Input Parameters:** -a array size -b do an export and then an import -c only perform conventional loads -d only display how the control file is split up -e export only -f already generated sqlload .ctl file location -h control cards to use when building the hash indexes -i import only -m indicates to execute the loads/unloads in parallel groups -p handle null columns properly -o the schema of the tables being exported/imported

Technical Name:	oraload.sh
	-s control file listing tables to be loaded/unloaded -t do not truncate any tables -u unrecoverable option -v use schema name as subdir on $DATADIR **Exit Values:** 0 - Normal Termination. Other - Abnormal termination because of open, read, allocation, or input errors. **Input Files:** Control files. **Output Files:** Log files. **Input Tables:** Those listed in control files. **Output Tables:** Those listed in control files. **Sort Criteria:** None **Switches:** None **Link Procedure:** None NOTE: The following environment variables must be set prior to running this script: FROM_PSWD --> specify the database to unload from in the format "user/id@database" TO_PSWD --> specify the database to load into in the format "user/id@database" Paths to log files and control files.

2.6.42.1 Change Orders

ID	NAME	DESCRIPTION
1581	Myers & Stauffer Extract	The Myers & Stauffer extracts include full exports of Claims, Member, Provider and Prior Authorization information from the DSS. These exports are sent to Myers & Stauffer LC.

ID	NAME	DESCRIPTION
1582	PWC Extract	The PricewaterhouseCoopers (PwC) extracts include claims and eligibility information from the DSS. These extracts are sent to PricewaterhouseCoopers, LLP.
1583	Navigant Extract	The Navigant extracts include claim information from the DSS. These extracts are sent to Navigant Consulting, Inc.
1642	IPRO Extract	The Island Peer Review Organization (IPRO) extract includes five extracts from the DSS: Encounter, Provider, Member, CPT code, and Drug code information. These extracts are sent to Island Peer Review Organization, Inc.
2070	eKASPER Interface	The eKASPER interface is a bi-directional interface between eKASPER and the DSS. The extract file from the DSS includes Medicaid members and applicable identifier information. The interface file from eKASPER to the DSS includes prescription information for controlled substance dispensed to the Medicaid members requested.
2074	LPSM Extract	The Lead Poisoning Surveillance Module (LPSM) extract includes lead screening data from the DSS. This extract is sent to Cabinet of Health and Family Services (CHFS).
2121	Artemetrx Extract	The Artemetrx extract includes three extracts from the DSS: Eligibility, Medical and Pharmacy Claims. These extracts are sent to Artemetrx, LLC.

2.6.43 otsortd -- Sort - UNIX

Technical Name:	otsortd
Program Title:	Sort – UNIX.
Programming Language:	UNIX Command
Description:	This is a UNIX sort utility that is used to sort and unduplicate files in the UNIX system. The otsortd is Sort/Merge/Select Utility written in C for Unix platforms from Opttech.com. It can process files of any size. It takes Input, Output, and Control file as parameters. Any number of input files can be specified. Based on the Control file it Sorts/Merges the Input file(s) and stores the results in Output file. This utility is used in number jobs to sort input files, output files and intermediate files as necessary. It is also used from the command line on an ad-hoc basis. **Input Parameters:** file1.dat - input file file2.dat - output file file2.ctl - control file **Exit Values:** 0 - Normal termination Other - Abnormal termination because of open, read, allocation, or input errors. **Input Files:** User specified **Output Files:** User Specified **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None **Link Procedure:** None **Special Logical Notes:** None

2.6.43.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.44 override.src -- Processes Job Script Overrides

Technical Name:	override.src
Program Title:	Processes Job Script Overrides.
Programming Language:	UNIX Script
Description:	Find all global and step specific overrides and source them. Typically performed before each execpgm in a job script.

2.6.44.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.45 ps2comm.lib -- ps2comm link library

Technical Name:	ps2comm.lib
Program Title:	ps2comm link library
Programming Language:	C
Description:	This is a link library of common routines used in interChange. The ps2comm.lib is the library that contains the functions called by the Eligibility Transaction Processor (ETP). The Eligibility Transaction Processor edits the incoming transactions from the KAMES, PA62 and SDX systems and applies the updates to the Members demographic, eligibility and case information. The ETP is used for daily and reconciliation updates.

2.6.45.1 Change Orders

ID	NAME	DESCRIPTION
1095	Verify HIC Function	There is a need for a common subroutine to verify the HIC Number. This function validates the HIC number received on the Member update transactions from KAMES, PA62, and SDX.

2.6.46 route_rpt.ksh -- Route report

Technical Name:	route_rpt.ksh
Program Title:	Route report.
Programming Language:	UNIX Script
Description:	This script loops through the $RPTDIR/route_rpt directory to get reports to be routed and then route the report according to the instructions in the sysin.

2.6.46.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.47 sendevent -- Schedule a job in Autosys

Technical Name:	sendevent
Program Title:	Schedule a job in Autosys.
Programming Language:	Other
Description:	Program that allows jobs to schedule a job in the Autosys.

2.6.47.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.48 set_ftplogin -- Set-up netrc file

Technical Name:	set_ftplogin
Program Title:	Set-up netrc file.
Programming Language:	UNIX Script
Description:	Copy the master.netrc file to netrc for ftp process. **Input Parameters:** None **Exit values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None **Link Procedures:** None

2.6.48.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.49 SOAPsrvr -- Soap Server transaction processor

Technical Name:	SOAPsrvr
Program Title:	Soap Server transaction processor.
Programming Language:	C
Description:	The SOAP server is an enhanced socket listener program that expects packets to be wrapped in http headers similar to a formal SOAP protocol. Upon receipt of a SOAP request, the program examines the request and determines which interchange program to call. The soapsrv is called by the AVR applications and KYHealthNet. Transaction processed includes eligibility inquiries, prior authorizations and claim submissions. A similar program (possrvr) that handles pure socket connections is documented in the EDI subsystem.

2.6.49.1 Change Orders

ID	NAME	DESCRIPTION
74	Configure Member for KAMES	There is a need to configure interChange Member processing for use with KAMES and other Commonwealth interfaces.

2.6.50 sqlldr -- Oracle Utility

Technical Name:	sqlldr
Program Title:	Oracle Utility.
Programming Language:	Oracle Utility.
Description:	This is the Oracle utility used to load information to an Oracle table.

2.6.50.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.51 watcher -- Watcher

Technical Name:	watcher
Program Title:	Watcher.
Programming Language:	Other
Description:	Autosys function that watches for a file.

2.6.51.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.52 xml_db -- XML-2-Database Transfer Utility

Technical Name:	xml_db
Program Title:	XML-2-Database Transfer Utility.
Programming Language:	C
Description:	The link between XML formatted data and the Oracle database. There are a variety of commands for this tool listed below. Generating Schema: There is a schema needed before loading or unloading XML. You can generate a schema for any database table as long as it is in ERWin. Determine what ERWin subject area your table is in. In the following example a file called 'procs.xsd' is created. xml_db -g t_proc -subjectArea="Reference" Unloading: You can specify "where-clause" conditions when unloading by adding "key='value'" options. The key is an element name (like proc for T_PROC) and the value is a literal SQL where-clause. xml_db -u procs proc='proc.sak_procedure > 1000' maxFee='dte_end between 19990101 and 19990701'. You may specify a maximum "depth" with the option -d. Use the -w3c option to get w3c-compliant dates in the format YYYY-MM-DD. Loading: This executes INSERT or UPDATE to load an XML tree into the database. If you pass a filename, xml_db does not execute the SQL statements but, instead, spool them to the output file DDL. This executes CREATE TABLE or ALTER TABLE based on the data-definition in an XML Schema. Header: This creates a C header file based on the data-definition in an XML Schema. Check Schema: This verifies that all dB:tables and dB:fields listed in the Schema are present in the database. It reports missing fields too. All these commands make database connections and use AIM_PSWD as defined in your environment or in your aim.cfg. **Input Parameters:** g - Generate Schema Definition from Database Meta-Information u - Unload XML Tree from Database l - Load XML Tree into Database d - Data Definition Language (DDL) **Exit Values:** EXIT_SUCCESS - Normal termination. EXIT_FAILURE - Abnormal termination because of open, read, allocation, or input errors. **Input Files:** None

	Output Files: None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None **Link Procedure:** libxml and libgutils

2.6.52.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.53 xmlpmrge -- Do merge base on hierarchical query

Technical Name:	xmlpmrge
Program Title:	Do merge base on hierarchical query.
Programming Language:	C
Description:	A SQL unloader takes a hierarchical query and executes a merge style operation on all of the child nodes. **Input Parameters:** a - alterTags = 1 q - loadDBName = optarg d - debug_level = atoi(optarg) f - formattedReport = TRUE i - Create and Load the input document o - Open the output file s - saveAtNode = optarg c - commitPoint = atoi(optarg) m - maxRows = atoi(optarg) p - set_displayProgress(50000) p - set_displayProgress(atoi(optarg)) u - set_useAsName(1) k - queryDoc->documentElement, name, value l - Open the shared library with the user hooks **Exit Values:** 0 - Normal termination Other - Abnormal termination because of open, read, allocation, or input errors. **Input Files:** SYSJDLTR syspbltr.xml **Output Files:** SYSJDLTR letters.xml **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None

	Link Procedure: libxml and libxmlmrge

2.6.53.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.54 xmlpunld -- Generate a hierarchical xml data stream from a query tree

Technical Name:	xmlpunld
Program Title:	Generate a hierarchical xml data stream from a query tree.
Programming Language:	C
Description:	This program generates a hierarchical xml data stream base on a query tree. **Input Parameters:** Option = "I" - Input xml document Option = "o" - Output xml document Option = "k" - Key name value pair Option = "m" - File size Option = "w" - Where clause Option = "a" - Set alter_to_schema_flagI Option = "u" - Set use_as_name_flag **Exit Values:** EXIT_SUCCESS - Normal termination EXIT_FAILURE - Abnormal termination because of open, read, allocation, or input errors. **Input Files:** REFJR300 coveredBenefitsP.tree.xml TPLJW100 tpw10001.ctl TPLJW101 tpw10101.ctl TPLJW102 tpw10201.ctl CLD38001.ctl CLD38002.ctl CLD38003.ctl CLD38004.ctl CLD38005.ctl CLD38006.ctl CLD38007.ctl CLD38008.ctl **Output Files:** TPLJW100 tpw10001.tmp TPLJW101 tpw10101.tmp TPLJW102 tpw10201.tmp pszOutputDoc **Input Tables:** None **Output Tables:** None **Sort Criteria:**

	None **Switches:** None **Link Procedure:** liblxml, liblgutils, and liblmalloc

2.6.54.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.6.55 xmlpunld.sh -- xmlpunld.sh

Technical Name:	xmlpunld.sh
Program Title:	xmlpunld.sh
Programming Language:	C
Description:	This utility is called in a job to perform an XML unload. It utilizes control cards to select the data.

2.6.55.1 Change Orders

ID	NAME	DESCRIPTION
None		

2.7 Internal Web Pages

The interChange .NET application uses visual system objects called pages, panels and controls. The document is set up to display first the Page, then all associated Panels. If a panel is accessible through more than one page, it displays multiple times in the document.

2.7.1 PAGE: System Wide Base Page

This page is a base page in System Wide subsystem that has the main menu, header, and footer. The main menu of this base page is affected by security. Only the menu items available to a user will be displayed on this page.

2.7.1.1 Description

This page is a base page in System Wide subsystem that has the main menu, header, and footer.

2.7.1.2 Technical Name

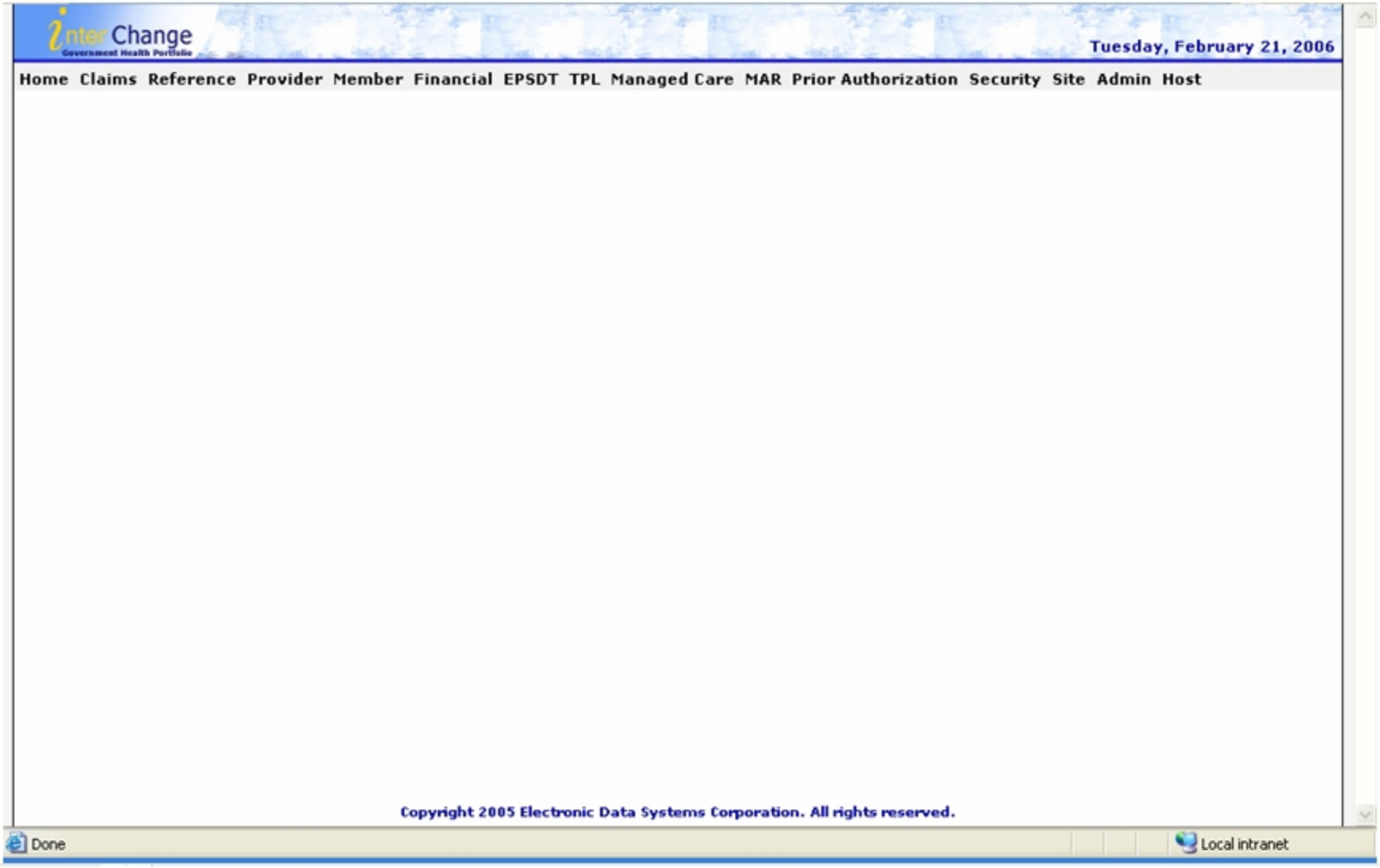
SystemWide.BasePage

2.7.1.3 Web Page Name

Internal Web

For readability, the page layout displays on the next page.

2.7.1.4 System Wide Base Page Layout



2.7.2 PANEL: System Wide Page Header

2.7.2.1 Description
Header Panel for the Page.

2.7.2.2 Technical Name
SystemWide.PageHeader

2.7.2.3 Panel Name
PageHeader

2.7.2.4 System Wide Page Header Layout

2.7.2.5 Extra Features
None

2.7.2.6 Field Descriptions

Field	Description	Field Type	Data Type	Length	DB Table	DB Attributes
time stamp	Current date	Field	Date (CCYYMMDD)	50		
site map	Link to Site Map	Hyperlink	N/A	0		
Admin	Link to login as administrator and setup site settings	Menu Item	N/A	0		
Claims	Link to Claims Subsystem	Menu Item	N/A	0		

Field	Description	Field Type	Data Type	Length	DB Table	DB Attributes
CTMS	Link to CTMS subsystem	Menu Item	N/A	0		
EPSDT	Link to EPSDT Subsystem	Menu Item	N/A	0		
Financial	Link to Financial Subsystem	Menu Item	N/A	0		
Home	Landing Page	Menu Item	N/A	0		
Host	Link to login as a host and setup host settings	Menu Item	N/A	0		
MAR	Link to MAR Subsystem	Menu Item	N/A	0		
Managed Care	Link to Managed Care Subsystem	Menu Item	N/A	0		
Member	Link to Recipient Subsystem	Menu Item	N/A	0		
Prior Authorization	Link to Prior Authorization Subsystem	Menu Item	N/A	0		

Field	Description	Field Type	Data Type	Length	DB Table	DB Attributes
Provider	Link to Provider Subsystem	Menu Item	N/A	0		
Reference	Link to Reference Subsystem	Menu Item	N/A	0		
Security	Link to Security settings	Menu Item	N/A	0		
Site	Link to change personal settings	Menu Item	N/A	0		
TPL	Link to TPL Subsystem	Menu Item	N/A	0		

2.7.2.7 Field Edits

Field	Field Type	Error Code	Error Message	To Correct
No field edits found for this window.				

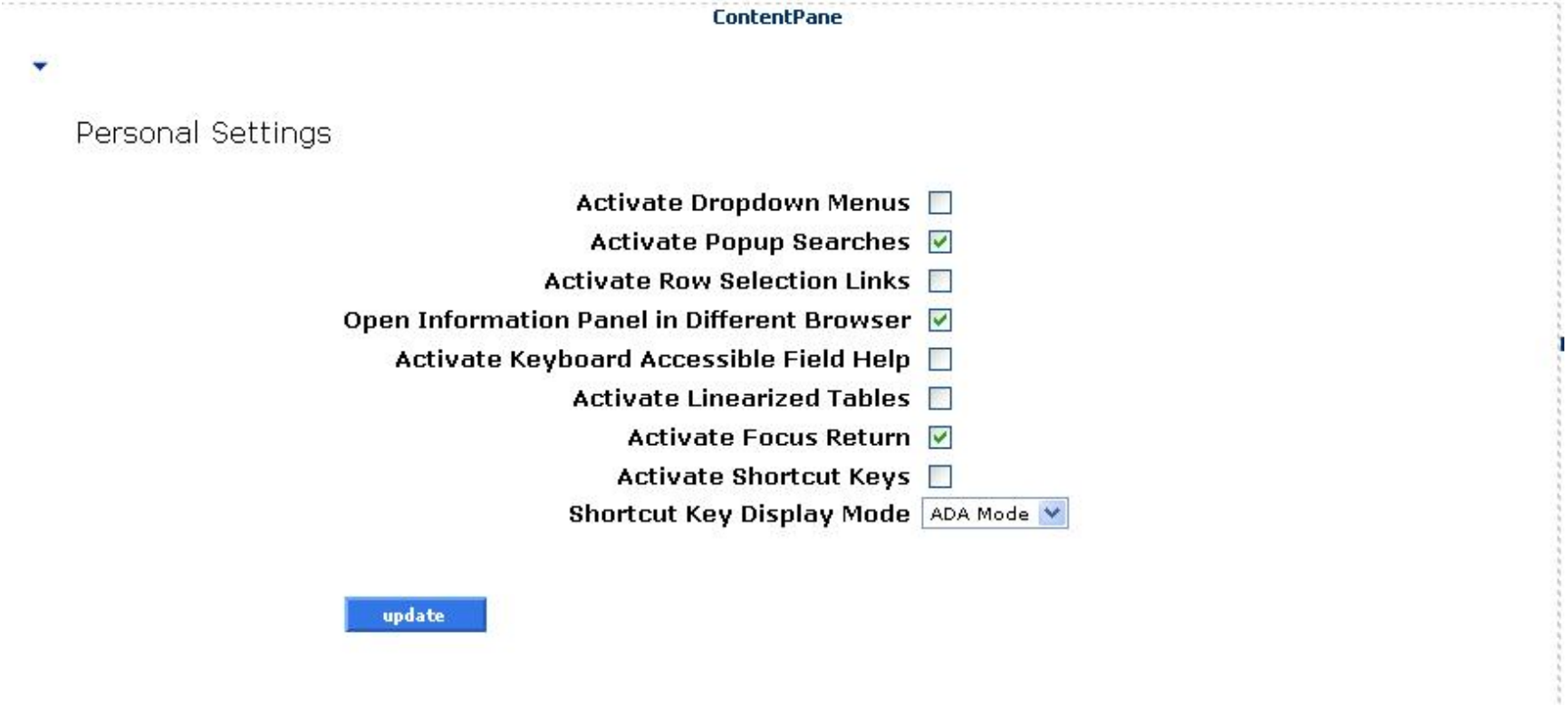
2.7.3 PANEL: Site Settings

2.7.3.1 Description
Site configuration settings for Dot Net Nuke.

2.7.3.2 Technical Name
SiteSettings

2.7.3.3 Panel Name
SiteSettings

2.7.3.4 Site Settings Layout



2.7.3.5 Extra Features

None

2.7.3.6 Field Descriptions

Field	Description	Field Type	Data Type	Length	DB Table	DB Attributes
Activate Dropdown Menus	When selected, submenus will drop down from main menus.	Checkbox	Boolean	1		
Activate Popup Searches	Not applicable to interChange	Checkbox	Boolean	1		
Activate Row Selection Links	Not applicable to interChange	Checkbox	Boolean	1		
Open Information Panel in Different Browser	This setting applies to the search results panels. When a row is selected from the search results, should the system open a new browser or not.	Checkbox	Boolean	1		
Activate Keyboard Accessible Field Help	Not applicable to interchange	Checkbox	Boolean	1		
Activate Linearlized Tables	When selected tables are shown in ADA compliant	Checkbox	Boolean	1		

Field	Description	Field Type	Data Type	Length	DB Table	DB Attributes
	(Linearlized) format for all users.					
Activate Focus Return	This setting returns focus to the last panel with focus instead of the top of the page where the information panel is located.	Checkbox	Boolean	1		
Activate Shortcut Keys	This setting enables and disables shortcut keys. Shortcut keys were part of an early release of interchange but have fallen out of use for all but ADA users.	Checkbox	Boolean	1		
Shortcut Key Display Mode	Shortcuts representation to the user can be “none” which means they don’t show up. Underlined, which means the letter for the shortcut key is underlined. ADA Mode, which means it is not visible, but is	Menu Item	Numeric	1		

Field	Description	Field Type	Data Type	Length	DB Table	DB Attributes
	in alt tags for readers such as JAWS.					

2.7.3.7 Field Edits

Field	Field Type	Error Code	Error Message	To Correct
No field edits found for this window.				

2.7.3.8 Associated Requirements

ID
None

2.7.3.9 CO / Defects

ID	TYPE	NAME	DESCRIPTION	Current Status
None				

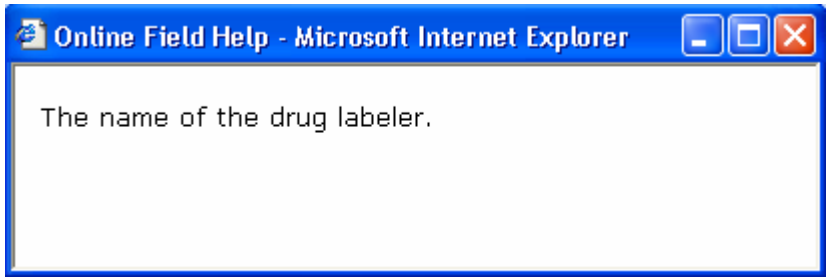
2.7.4 PANEL: Help

Description
This panel appears when field or panel help is selected in the interChange .NET application. Each label that appears on a data panel can have help text associated with it. Field help text comes from the fieldHelp attribute on the Property element in the Business Entities config.xml file.

Technical Name
Help

Panel Name
Help

Site Settings Layout



Extra Features
None

Field Descriptions

Field	Description	Field Type	Data Type	Length	DB Table	DB Attributes
Help Text	Help Text	Checkbox	Boolean	1		

Field Edits

Field	Field Type	Error Code	Error Message	To Correct
No field edits found for this window.				

Associated Requirements

ID
None

CO / Defects

ID	TYPE	NAME	DESCRIPTION	Current Status
None				

2.7.5 PANEL: Audit Trail

2.7.5.1 Description

This panel is a general purpose panel that utilizes configuration data from the data panels to adjust its display for the associated content. The example shows is a member base information panel and its associated audit panel. The base information panel was included on the page display below to provide context for the audit panel. NPI data was redacted.

2.7.5.2 Technical Name

SystemWide.AuditTrail

2.7.5.3 Panel Name

Audit Trail

2.7.5.4 System Wide Page Header Layout

Base Information

Member ID	[redacted]	Last Name*	[redacted]	Foster Care Ind	No
SSN*	[redacted]	First Name*	[redacted]	Trust Ind	No
Date of Birth*	06/05/1961	MI	C	DJJ Ind	No
Date of Death		Suffix		Investigation Ind	
Gender Code*	Male	Home Street Address 1*	[redacted]	Language	UNDETERMINED
Race Code	White	Street Address 2		Duplicate	Yes
Multi-Race Code*	04 White	Street Address 3		Twin	No
Ethnicity	01 Non-Hispanic	City*	PRINCETON	Header Inst Status	
Phone		State*	KY		
Phone Type*	No Phone	Zip Code*	42445		
Add Phone		Zip Ext	[redacted]		
Add Phone Type*	No Phone	Home County*	Caldwell		

» Base Information Audit History

Columns

☐ Specialty Health	☐ Member ID	☐ Last Name
☐ First Name	☐ MI	☐ Suffix
☐ Home Street Address 1	☐ Address 2	☐ Address 3
☐ City	☐ Zip Code	☐ Zip + 4
☐ SSN	☐ Date of Birth	☐ Death Date
☐ Ethnicity	☐ Marital Status	☐ Conuty Office of Residence
☐ Primary Language	☐ Correspondence Language	☐ Source Code
☐ Phone	☐ Add Phone	☑ Member Status
☐ Home County	☐ Race Code	☐ Phone Indicator
☑ Add Phone Ind	☐ County Office of Service	☐ Gender
☐ Citizen	☐ Living Arrangement	☐ State
☐ Date Added	☑ Date Updated	☑ Address Source
☑ User Name	☑ System Date	☑ Action Code

Start Date
End Date
Show All ☐
Records 20

search
deselect all
select all

Audit Results

Member Status	Add Phone Ind	Date Updated	Address Source	User Name	System Date	Action Code
				KYBAT	7/29/2007 6:14:57 PM	I
Y				KYBAT	7/29/2007 6:14:57 PM	U
	8	07/25/2007	KAMES	gyrbox@eds.kyxix.edsmhg.com	8/22/2007 11:35:54 AM	U

2.7.5.5 Extra Features

None

2.7.5.6 Field Descriptions

Field	Description	Field Type	Data Type	Length	DB Table	DB Attributes
time stamp	Current date	Field	Date (CCYYMMDD)	50		
site map	Link to Site Map	Hyperlink	N/A	0		
Admin	Link to login as administrator and setup site settings	Menu Item	N/A	0		
Claims	Link to Claims Subsystem	Menu Item	N/A	0		
CTMS	Link to CTMS subsystem	Menu Item	N/A	0		
EPSDT	Link to EPSDT Subsystem	Menu Item	N/A	0		
Financial	Link to Financial Subsystem	Menu Item	N/A	0		
Home	Landing Page	Menu Item	N/A	0		
Host	Link to login as a host and setup host settings	Menu Item	N/A	0		

Field	Description	Field Type	Data Type	Length	DB Table	DB Attributes
MAR	Link to MAR Subsystem	Menu Item	N/A	0		
Managed Care	Link to Managed Care Subsystem	Menu Item	N/A	0		
Member	Link to Recipient Subsystem	Menu Item	N/A	0		
Prior Authorization	Link to Prior Authorization Subsystem	Menu Item	N/A	0		
Provider	Link to Provider Subsystem	Menu Item	N/A	0		
Reference	Link to Reference Subsystem	Menu Item	N/A	0		
Security	Link to Security settings	Menu Item	N/A	0		
Site	Link to change personal settings	Menu Item	N/A	0		
TPL	Link to TPL Subsystem	Menu Item	N/A	0		

2.7.5.7 Field Edits

Field	Field Type	Error Code	Error Message	To Correct
No field edits found for this window.				

2.7.5.8 Associated Requirements

ID
None

2.7.5.9 CO / Defects

ID	TYPE	NAME	DESCRIPTION	Current Status
None				

2.8 Reports

Some information in this section is represented in table format. In order to fit information on the page, some data field information may wrap to the next line.

2.8.1 Audit Trail Report

A report is required to audit all changes to all fields that are maintained by the user in each subsystem. The report allows tracking of all previous values, which made the change, and the date of the change.

Each subsystem has its own report. The report consists of each field in each of the panels. The current value of the field displays. Previous field values including the date of the change and the user making the change, display underneath the current value.

Reporting Service generates the predefined reports when requested by the user clicking on the "Audit" button.

2.8.1.1 Technical Name

Audit Trail Report

2.8.1.2 Audit Trail Report Layout

```
Report  : E-120-D                    COMMONWEALTH OF KENTUCKY          Run Date:      04/28/2006
Process : MIS                 MEDICAD MANANGEMENT INFORMATION SYSTEM   Run Time:      3:09:15 PM
Location: KY                            Member Audit Trail             Page    :               1
```

ID: 1512021

Created on: 8/13/2004 6:16:15 PM

First Name:

Last Name: GATES

Previous Value	Responsible Party	Date Of Change
GATES	DZXKLC	2/7/2006 9:26:03 AM

Middle Name:

Address 1:

Report : E-120-D COMMONWEALTH OF KENTUCKY Run Date: 04/28/2006
Process : MIS MEDICAD MANANGEMENT INFORMATION SYSTEM Run Time: 3:09:15 PM
Location: KY Member Audit Trail Page : 2

Level of Care

Provider: BAILY MEDICAL SUPPLIES

Previous Value	Responsible Party	Date Of Change
SMITH JOHN D	DSCFNETA	4/14/2005 5:40:08 PM
RANDOLPH PHILIPS MEMORY HOSP	DSCFNETA	4/14/2005 5:12:16 PM
FRANCIS MEMORIAL CLINIC	DSCFNETA	4/14/2005 5:00:01 PM
FRANCIS MEMORIAL CLINIC	DSCFNETA	4/14/2005 5:00:01 PM
FRANCIS MEMORIAL CLINIC	DSCFNETA	4/14/2005 5:00:01 PM

Level of Care: Personal Care

Previous Value	Responsible Party	Date Of Change
ICF/MR	DSCFNETA	4/14/2005 5:44:47 PM
ICF/MRDD	DSCFNETA	4/14/2005 5:43:55 PM
Personal Care	DSCFNETA	4/14/2005 5:19:42 PM
ICF III	DSCFNETA	4/14/2005 4:54:44 PM
ICF II	DSCFNETA	4/14/2005 4:53:59 PM
ICF I	DSCFNETA	4/14/2005 4:53:31 PM
ICF/MR	DSCFNETA	4/14/2005 4:52:18 PM

Effective Date: 20030101

End Date: 22991231

Provider: BAILY MEDICAL SUPPLIES

Level of Care: ICF/MR

Effective Date: 20020101

End Date: 20021231

*** END OF REPORT ***

2.8.1.3 Field Descriptions

Field	Description	Length	Data Type	DB tables	DB Attributes
ID	The system assigned internal key for a unique member.	20	NUMBER	T_RE_BASE	SAK_ID
Created On	Date the member record was created.	25	DATETIME	A_T_RE_BASE OR A_T_RE_LOC	DTE_DSYSDATE
First Name	The first name of a member.	12	CHAR	T_RE_BASE	NAM_FIRST
Last Name	The last name of a member.	20	CHAR	T_RE_BASE	NAM_LAST
Middle Name	The middle initial of the member.	1	CHAR	T_RE_BASE	NAME_MID_INIT
Address 1	The first line of the member's street address.	30	Char	T_RE_BASE	ADR_STREET_1
Provider	This is the name associated with an organization or person.	50	CHAR	T_PR_NAM	NAME
Level of Care	Identifies the level of care for a member.	100	CHAR	T_RE_LOC_CODE	DSC_LOC
Effective Date	The date that the corresponding provider is authorized to bill for the level of care.	8	NUMBER	T_RE_LOC	DTE_EFFECTIVE
End Date	The date that the corresponding provider is no longer authorized to bill for the recipient's level of care.	8	NUMBER	T_RE_LOC	DTE_END

Field	Description	Length	Data Type	DB tables	DB Attributes
Responsible Party	ID of the user the made update, insert, or delete.	30	CHAR	A_T_RE_BASE OR A_T_RE_LOC	NAM_USER
Date Of Change	Date and time that the user made that change to the record.	25	DATETIME	A_T_RE_BASE.or A_T_RE_LOC	DTE_ SYSDATE
Previous Value	Previous value of field being audited			A_T_RE_BASE or A_RE_LOC	Corresponding field in either A_T_RE_BASE or A_T_RE_LOC to the field above it on report

2.8.1.4 Associated Programs

Program	Description
No associated Programs found.	

2.8.1.5 Associated Requirements

ID
No associated Requirements found.

2.8.1.6 Change Orders

ID	Name	Description
None.		

2.8.2 SYS-RPTFTR -- Report Footer Standard

This is the standard footer used on all interChange MMIS reports. Only one of the two footers is printed, depending on whether or not the report contains data.

2.8.2.1 Technical Name

SYS-RPTFTR

2.8.2.2 SYS-RPTFTR -- Report Footer Standard Layout

```
… report content

…

                    *** END OF REPORT ***

                 *** NO DATA THIS RUN ***
```

- This layout shows the standard report footers. Only one footer will be displayed. If no data is reported, then the footer '*** NO DATA THIS RUN ***' is displayed. If data is reported, then the footer '*** END OF REPORT ***' is displayed. The footers are centered on the report.

2.8.2.3 Field Descriptions

Field	Description	Length	Data Type	DB Table	DB Attributes
No field descriptions needed.					

2.8.2.4 Associated Programs

Program	Description
None	

2.8.2.5 Associated Requirements

ID
None

2.8.2.6 Change Orders

ID	Name	Description
None		

2.8.3 SYS-RPTHDR -- Report Header Standard

This is the standard header layout used on interChange MMIS reports, as defined by the SRS reports manager.

2.8.3.1 Technical Name

SYS-RPTHDR

2.8.3.2 SYS-RPTHDR -- Report Header Standard Layout

```
Report  : XXX-XXXX-X                    interChange MMIS                    Run Date: MM/DD/CCYY
Process : XXXXXXXX          MEDICAID MANAGEMENT INFORMATION SYSTEM          Run Time:   HH:MM:SS
Location: XXXXXXXX                          (TITLE)                            Page:      9999
```

- This layout shows the required elements and relative positions of those elements.
- The line length is ordinarily intended to be 132 characters.
- Each center element should be centered on its line.

Left Margin:
Report Field—Contains the tracking number assigned to the report.
Process Field—Contains the job name started the program that created the report.
Location Field—Contains the program name that created the report.

Center:
interChange MMIS—Standard name on all reports
Title Field—Contains a brief written title/description of the report.
Report Period Field—Should be included on all reports. Daily & Weekly reports should be in the mm/dd/ccyy format, Monthly reports should be in mm/ccyy format, Quarterly should be in mm/dd/ccyy – mm/dd/ccyy format, annual reports should be in ccyy format.

Right Margin:
Run Date Field—Contains that date on which the report was created in mm/dd/ccyy format.
Run Time Field—Contains the time in which the report was completed in hh:mm:ss format.
Page Field—Contains the page number of that page in the report.

2.8.3.3 Field Descriptions

Field	Description	Length	Data Type	DB Table	DB Attributes
As Of	The date (mm/dd/ccyy) that corresponds to the data in the report in mm/dd/ccyy format.	18	Number		
Location	The name of the program that created the report.	8	Character		
Page	The page number of that page in the report and the number of pages in the whole report (length variable to accommodate total pages).	21	Character		
Process	The job script name that starts the program that created the report.	8	Character		
Report	The tracking number assigned to the report.	10	Character		
Report Period	Should be included on all reports.	39	Character		
Run Date	That date on which the report was created in mm/dd/ccyy format.	21	Character		
Run Time	The time in which the report was completed in hh:mm:ss format.	21	Character		
Standard name line one	The system identification.	34	Character		
Standard name line two	Reporting System.	16	Character		
Title	A brief written title/description of the report (length is variable).	100	Character		

2.8.3.4 Associated Programs

Program	Description
None	

2.8.3.5 Associated Requirements

ID
None

2.8.3.6 Change Orders

ID	Name	Description
None		

2.9 Letters

The System Wide subsystem does not directly produce any letters.

2.10 Internal and External Interfaces

The System Wide subsystem does not directly have any internal or external interfaces.

2.11 Requirement Matrix and Cross-reference

This section provides a crosswalk of each functional requirement included in the New KY MMIS Requirements Checklist shown in the RFP Appendix to MMIS design components. This is accomplished by listing each requirement with its current status, associated change orders and mapped system objects.

Subsystem	Requirement	Status	Mapped Objects	Change Orders
System Wide	20	Open	clm0018d clmpdr02 clmpmucl clmsStopLoss clmsWrite clmsphrmReversal	
System Wide	20.000	Informational		
System Wide	20.010	Informational		
System Wide	20.020	Informational	T_HIPAA_LANGUAGE T_HIPAA_MC_ASSIGN T_PA_ITEM_DTL_XREF T_PA_PAUTH adjHistOnly adjQuery adjRecpLink adjRegular adjVoid clmDelete clmpdcr2 clmsAudit clmsLoadHist clmsPaPauth clmsPricing clmsWrite clmsdntl clmsdntlDtl clmsdntlWrite clmsphrm clmsphrmDtl clmsphrmWrite clmsphys clmsphysProc	

Subsystem	Requirement	Status	Mapped Objects	Change Orders
			clmsphysWrite clmsub92 clmsub92Hdr clmsub92IPPricing clmsub92Proc clmsub92Write edir001w.sc Clerk Account Maintenance	
System Wide	20.060	Informational		
System Wide	20.070	Informational		
System Wide	20.080	Informational		
System Wide	20.090	Informational		
System Wide	30.050.001V	Change Order Identified		
System Wide	30.050.002.002.5A			
System Wide	30.060	EDS Met Task		
System Wide	30.060.001	Informational		
System Wide	30.060.001.001	EDS Met Task		
System Wide	30.060.001.002	EDS Met Task		
System Wide	30.060.001.003	EDS Met Task		
System Wide	30.060.001.004	Informational		
System Wide	30.060.001.005	EDS Met Task		
System Wide	30.060.001.006	EDS Met Task		

Subsystem	Requirement	Status	Mapped Objects	Change Orders
System Wide	30.060.001.007	EDS Met Task		
System Wide	30.060.001.008	EDS Met Task		
System Wide	30.060.002	Informational		
System Wide	30.060.002.001	EDS Met Task		
System Wide	30.060.002.002	EDS Met Task		
System Wide	30.060.002.003	EDS Met Task		
System Wide	30.060.002.004	EDS Met Task		
System Wide	30.060.002.005	EDS Met Task		
System Wide	30.060.002.006	EDS Met Task		
System Wide	30.060.002.007	EDS Met Task		
System Wide	30.060.002.008	EDS Met Task		
System Wide	30.060.002.009	EDS Met Task		
System Wide	30.060.002.010	EDS Met Task		
System Wide	30.060.002.011	EDS Met Task		
System Wide	30.060.003	EDS Met Task		

Subsystem	Requirement	Status	Mapped Objects	Change Orders
System Wide	30.060.003.001	EDS Met Task		
System Wide	30.060.003.002	EDS Met Task		
System Wide	30.060.003.003	EDS Met Task		
System Wide	30.060.003.004	EDS Met Task		
System Wide	30.060.004	EDS Met Task		
System Wide	30.060.004.001	EDS Met Task		
System Wide	30.060.004.002	EDS Met Task		
System Wide	30.060.004.003	EDS Met Task		
System Wide	30.060.004.004	EDS Met Task		
System Wide	30.060.004.005	EDS Met Task		
System Wide	30.090.007.004.1	Change Order Identified		1410 - 820 Outbound Process 1411 - 834 Outbound Process 1423 - 270 Inbound Process 1424 - 271 Outbound Process 1425 - 276 Inbound Process 1426 - 277 Outbound Process 1427 - 277U Outbound Process 1480 - 835 Outbound Process 1770 - VAN Activity

Subsystem	Requirement	Status	Mapped Objects	Change Orders
				Reports 1773 - 997 Outbound Process 1774 - 834 Inbound Process 1775 - 277U Inbound Process 1791 - File Transfer Service Provider 1792 - File Transfer Service Member 1800 - File Transfer Service Mgd Care 1825 - File Transfer Service Reference 2112 - 277U Outbound Process 2113 - Encounters - Identification 2155 - 270 Batch Process 2156 - 276 Batch Process
System Wide	40	Informational		
System Wide	40.000	Informational		
System Wide	40.010	Informational		
System Wide	40.015	Informational		
System Wide	40.016	Informational		
System Wide	40.020	Informational		
System Wide	40.025	Informational		
System Wide	40.030	Informational		
System Wide	40.040	Informational		
System Wide	40.050	Informational		
System Wide	40.055	Informational		

Subsystem	Requirement	Status	Mapped Objects	Change Orders
System Wide	40.060	Informational		
System Wide	40.065	Informational		
System Wide	40.070	Informational		
System Wide	40.075	Informational		
System Wide	40.075.010	Informational		
System Wide	40.075.020	Informational		
System Wide	40.075.020.001	Operations		
System Wide	40.075.020.002	Operations		
System Wide	40.075.020.003	Operations		
System Wide	40.075.020.004	Operations		
System Wide	40.075.020.005	Operations		
System Wide	40.075.020.006	Operations		
System Wide	40.075.030	Informational		
System Wide	40.075.040	Informational		
System Wide	40.075.050	Informational		
System Wide	40.075.060	Informational		
System Wide	40.075.060.001	Informational		
System Wide	40.075.060.002	Informational		

Subsystem	Requirement	Status	Mapped Objects	Change Orders
System Wide	40.075.060.003	Informational		
System Wide	40.075.060.004	Informational		
System Wide	40.075.060.005	Informational		
System Wide	40.075.060.006	Informational		
System Wide	40.075.060.007	Informational		
System Wide	40.075.060.008	Informational		
System Wide	40.075.060.009	Informational		
System Wide	40.075.060.010	Informational		
System Wide	40.075.060.013	Informational		
System Wide	40.075.060.014	Informational		
System Wide	40.075.070	Informational		
System Wide	40.075.070.001	Informational		
System Wide	40.075.070.002	Informational		
System Wide	40.075.070.003	Informational		
System Wide	40.075.070.004	Informational		

Subsystem	Requirement	Status	Mapped Objects	Change Orders
System Wide	40.075.070.005	Informational		
System Wide	40.075.070.006	Informational		
System Wide	40.075.070.007	Operations		
System Wide	40.075.070.008	Informational		
System Wide	40.075.070.009	Informational		
System Wide	40.075.070.010	Informational		
System Wide	40.075.070.011	Operations		1967 - state specific setup
System Wide	40.075.070.012	Operations		
System Wide	40.075.070.013	Operations		
System Wide	40.075.070.014	Informational		
System Wide	40.075.070.015	Operations		
System Wide	40.075.070.016	Informational		
System Wide	40.075.070.017	Operations		
System Wide	40.075.070.018	Informational		
System Wide	40.080	Informational		

Subsystem	Requirement	Status	Mapped Objects	Change Orders
System Wide	40.085	Informational		
System Wide	40.090	Informational		
System Wide	40.095	Operations		
System Wide	40.100	Operations		
System Wide	40.105	Operations		
System Wide	40.110	Informational		
System Wide	40.115	Informational		
System Wide	40.120	Informational		
System Wide	40.125	Informational		
System Wide	40.130	Informational		
System Wide	40.135	Informational		
System Wide	40.140	Informational		
System Wide	40.145	Informational		
System Wide	40.150	Informational		
System Wide	40.160	Informational		
System Wide	40.170	Informational		
System Wide	40.180	Informational		
System Wide	40.190	Informational		
System Wide	40.200	Informational		
System Wide	40.210	Informational		
System Wide	40.220	Informational		
System Wide	40.230	Informational		

Subsystem	Requirement	Status	Mapped Objects	Change Orders
System Wide	40.240	Informational		
System Wide	40.250	Informational		
System Wide	40.260	Informational		
System Wide	40.270	Informational		
System Wide	40.280	Operations		
System Wide	50	Informational		
System Wide	50.000	Informational		
System Wide	50.010	Informational		
System Wide	50.020	Informational		
System Wide	50.030	Informational		
System Wide	50.040	Informational		
System Wide	50.050	Informational		
System Wide	50.060	Informational		
System Wide	50.070	Informational		
System Wide	50.080	Informational		
System Wide	50.090	Informational		
System Wide	50.100	Informational		
System Wide	50.110	Informational		
System Wide	50.120	Informational		
System Wide	50.130	Informational		
System Wide	50.140	Informational		
System Wide	50.150	Informational		

Subsystem	Requirement	Status	Mapped Objects	Change Orders
System Wide	50.160	Informational		
System Wide	50.170	Informational		
System Wide	50.180	Informational		
System Wide	50.190	Informational		
System Wide	50.210	Informational		
System Wide	50.220	Informational		
System Wide	60	Informational		
System Wide	60.000	Informational		
System Wide	60.010	Informational		
System Wide	60.020	Informational		
System Wide	60.030	Informational		
System Wide	60.040	Informational		
System Wide	60.050	Informational		
System Wide	60.060	Informational		
System Wide	60.070	Informational		
System Wide	60.080	Informational		
System Wide	60.090	Informational		
System Wide	60.100	Informational		
System Wide	70	Informational		
System Wide	70.000	Informational		
System Wide	70.010	Informational		
System Wide	70.020	Informational		

Subsystem	Requirement	Status	Mapped Objects	Change Orders
System Wide	70.030	Informational		
System Wide	70.040	Informational		
System Wide	70.050	Informational		
System Wide	70.060	Informational		
System Wide	80	Informational		
System Wide	80.000	Informational		
System Wide	80.010	Informational		
System Wide	80.020	Informational		

2.12 Change Orders

2.12.1 Target Operational Date Change - 43

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
43	Change Order		System Wide			

2.12.1.1 Desired Solution

Change the target operational date for New KY Medicaid Management Information System (MMIS) (DDI) from 10/01/2006 to 11/01/2006 based on the actual signing date for the contract.

2.12.1.2 Business Impact

None.

2.12.1.3 Technical Specifications

None.

2.12.1.4 Clarifications

No associated clarifications found.

2.12.1.5 Associated Requirements

Requirement ID	Type
None	

2.12.1.6 Associated System Objects

Technical Name	Object Type	Title
No associated system objects found.		

2.12.1.7 Change Order Status

Status	Date
Change Order Written	06/13/2005
Closed	02/10/2006

2.12.2 Use Captive for Data Capture - 44

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
44	Change Order		System Wide			

2.12.2.1 Desired Solution
Use Captiva as the screen capture tool for New KY MMIS (DDI) instead of TIS.

2.12.2.2 Business Impact
None.

2.12.2.3 Technical Specifications
None.

2.12.2.4 Clarifications
No associated clarifications found.

2.12.2.5 Associated Requirements

Requirement ID	Type
None	

2.12.2.6 Associated System Objects

Technical Name	Object Type	Title
No associated system objects found.		

2.12.2.7 Change Order Status

Status	Date
Change Order Written	06/13/2005
Closed	02/10/2006

2.12.3 RSD & GSD combined into one GSD - 45

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
45	Change Order		System Wide			

2.12.3.1 Desired Solution
Combine the Requirements Specification Document (RSD) content and the General Specification Document (GSD) content into one GSD document for the New KY MMIS (DDI) project.

2.12.3.2 Business Impact
None.

2.12.3.3 Technical Specifications
None.

2.12.3.4 Clarifications
No associated clarifications found.

2.12.3.5 Associated Requirements

Requirement ID	Type
None	

2.12.3.6 Associated System Objects

Technical Name	Object Type	Title
No associated system objects found.		

2.12.3.7 Change Order Status

Status	Date
Change Order Written	06/13/2005
Closed	02/10/2006

2.12.4 Create KY Operating Procedures - 212

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
212	Task	S	System Wide			1

2.12.4.1 Desired Solution

Create Kentucky interChange Operating Procedures.

RFP Text:

Section 30.050 - New KY MMIS Operations, Maintenance, and Modifications Phase

The Contractor shall be responsible for operating, maintaining and modifying the New KY MMIS according to the Center for Medicare and Medicaid Services (CMS) State Medicaid Manual; all Federal mandates; and all Commonwealth requirements, statues, and regulations.

The Contractor's Scope of Work shall include performance of the operational functions. Each function is presented in terms of:

- Commonwealth or its designee responsibilities; and,
- Contractor responsibilities.

In addition to these responsibilities, the performance expectations for operations are listed in Section 30 within this RFP. The contractor shall perform all functions necessary to operate a flexible, successful, complete, and CMS certifiable KY MMIS.

2.12.4.2 Business Impact

This task will be used to identify and track the steps necessary to create and maintain the operating procedures necessary to maintain the MMIS.

2.12.4.3 Technical Specifications

See the section in the work plan called **KY interChange Operating Procedures**. All work for setting up and documenting the KY operational procedures will be handled under this work plan section.

2.12.4.4 Clarifications

The need for this task was identified during the Benefits Administration RV/JAD session on 6/29/2005.

2.12.4.5 Associated Requirements

Requirement ID	Type
None	

2.12.4.6 Associated System Objects

Technical Name	Object Type	Title
No associated system objects found.		

2.12.4.7 Change Order Status

Status	Date
Issue Identified	07/07/2005
Change Order Written	08/09/2005

2.12.5 Remove PES Requirement - 402

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
402	Change Order		System Wide			

2.12.5.1 Desired Solution
Remove thick-client requirements (PES) and use a thin-client approach instead.

2.12.5.2 Business Impact
None.

2.12.5.3 Technical Specifications
None.

2.12.5.4 Clarifications
No associated clarifications found.

2.12.5.5 Associated Requirements

Requirement ID	Type
None	

2.12.5.6 Associated System Objects

Technical Name	Object Type	Title
No associated system objects found.		

2.12.5.7 Change Order Status

Status	Date
Change Order Written	08/05/2005
Closed	02/10/2006

2.12.6 Eliminate agreed upon Req. - 403

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
403	Change Order		System Wide			

2.12.6.1 Desired Solution
This change order handles the elimination of agreed upon Claims and Provider Data Maintenance Requirements.

2.12.6.2 Business Impact
None.

2.12.6.3 Technical Specifications
None.

2.12.6.4 Clarifications
No associated clarifications found.

2.12.6.5 Associated Requirements

Requirement ID	Type
None	

2.12.6.6 Associated System Objects

Technical Name	Object Type	Title
No associated system objects found.		

2.12.6.7 Change Order Status

Status	Date
Change Order Written	08/05/2005
Closed	02/10/2006

2.12.7 Biztalk at Cold Harbor - 1072

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
1072	Change Order		System Wide			

2.12.7.1 Desired Solution
EDS Support of Medicaid BizTalk environment at Cold Harbor.

2.12.7.2 Business Impact
None.

2.12.7.3 Technical Specifications
None.

2.12.7.4 Clarifications
No associated clarifications found.

2.12.7.5 Associated Requirements

Requirement ID	Type
None	

2.12.7.6 Associated System Objects

Technical Name	Object Type	Title
No associated system objects found.		

2.12.7.7 Change Order Status

Status	Date
Change Order Written	09/16/2005
Closed	02/10/2006

2.12.8 OnBase Storage - 1073

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
1073	Change Order		System Wide			

2.12.8.1 Desired Solution
OnBase storage of all Claims images.

2.12.8.2 Business Impact
None.

2.12.8.3 Technical Specifications
None.

2.12.8.4 Clarifications
No associated clarifications found.

2.12.8.5 Associated Requirements

Requirement ID	Type
None	

2.12.8.6 Associated System Objects

Technical Name	Object Type	Title
No associated system objects found.		

2.12.8.7 Change Order Status

Status	Date
Change Order Written	09/16/2005
Closed	02/10/2006

2.12.9 Audit Reports - 1251

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
1251	Change Order		System Wide			

2.12.9.1 Desired Solution
Create Audit Reports.

2.12.9.2 Business Impact
None.

2.12.9.3 Technical Specifications
Each subsystem will have its own report. The report will consists of each field in each of the panels. The current value of the field is displayed. Previous field values including the date of the change and the user making the change are displayed underneath the current value. Reporting Service will be used to generate the reports.

2.12.9.4 Clarifications
Subsystem Audit reports
BPEL, Clm, CT, DR, EPSDT, Financial, MC, PA, Prov, Recip, Ref, SP, SUR, SURCT, SW, TPL

For each of the above areas, a report will be generated showing the fields in the main panel and related subpanels. The current value of each field will be displayed along with all previous values, user who changed the value and date of change

2.12.9.5 Associated Requirements

Requirement ID	Type
None	

2.12.9.6 Associated System Objects

Technical Name	Object Type	Title
No associated system objects found.		

2.12.9.7 Change Order Status

Status	Date
Issue Identified	10/06/2005
Change Order Written	03/20/2006

2.12.10 KY LG URL - 1958

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
1958	Change Order		System Wide			

2.12.10.1 Desired Solution
To change the Config Util file to make it point to the correct URL for KY Letter Generator.

2.12.10.2 Business Impact
None.

2.12.10.3 Technical Specifications
Make changes in the config util file for KY to make it point to the KY Letter Generator URL

2.12.10.4 Clarifications
No associated clarifications found.

2.12.10.5 Associated Requirements

Requirement ID	Type
None	

2.12.10.6 Associated System Objects

Technical Name	Object Type	Title
No associated system objects found.		

2.12.10.7 Change Order Status

Status	Date
Change Order Written	01/16/2006
Construction in Process	02/10/2006

2.12.11 State specific setup - 1967

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
1967	Change Order		System Wide			

2.12.11.1 Desired Solution

Many labels relating to Medicaid and health care vary from state to state. For example, in Kentucky, recipients are supposed to be referred to as members. In other states, the label of a member could be client, beneficiary, etc.

The interChange application has a list of such labels that must be defined for each state, known as the state localization variable list.

This change order requires that the state localization variable list values be defined for the Commonwealth of Kentucky.

It also requires that the necessary changes be made to the source code development environment so that Kentucky-specific functionality will be used by the application instead of generic, Core functionality.

2.12.11.2 Business Impact

This change is needed so that a system-wide, consistent method exists of overriding the Core system functionality to meet the requirements of the Kentucky MMIS.

2.12.11.3 Technical Specifications

The state localization variable list will be kept in a standard Microsoft .net resource file, which will be embedded in the state localization DLL. The details are in the attached design documentation.

2.12.11.4 Clarifications

No associated clarifications found.

2.12.11.5 Associated Requirements

Requirement ID	Type
None	

2.12.11.6 Associated System Objects

Technical Name	Object Type	Title
No associated system objects found.		

2.12.11.7 Change Order Status

Status	Date
Ready for MO Impl	01/19/2006

2.12.12 Replace New KY MMIS claims mgr - 1977

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
1977	Change Order		System Wide			

2.12.12.1 Desired Solution
Replace New KY MMIS claims manager position with one additional Business Analyst and one additional Provider Representative for the duration of the contract.

2.12.12.2 Business Impact
None.

2.12.12.3 Technical Specifications
None.

2.12.12.4 Clarifications
Betsy Elrod - This was added as a placeholder change order for Scope changes for the account per Derek Simons on 1/24/2006.

2.12.12.5 Associated Requirements

Requirement ID	Type
None	

2.12.12.6 Associated System Objects

Technical Name	Object Type	Title
No associated system objects found.		

2.12.12.7 Change Order Status

Status	Date
Change Order Written	01/24/2006
Closed	02/10/2006

2.12.13 Remove func.ops. Requirement - 1978

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
1978	Change Order		System Wide			

2.12.13.1 Desired Solution
Remove MMIS functional operations requirements.

2.12.13.2 Business Impact
None.

2.12.13.3 Technical Specifications
None.

2.12.13.4 Clarifications
Betsy Elrod - This was added as a placeholder change order for Scope changes for the account per Derek Simons on 1/24/2006.

2.12.13.5 Associated Requirements

Requirement ID	Type
None	

2.12.13.6 Associated System Objects

Technical Name	Object Type	Title
No associated system objects found.		

2.12.13.7 Change Order Status

Status	Date
Change Order Written	01/24/2006
Closed	02/10/2006

2.12.14 FirstDecision Schema for DSS - 1980

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
1980	Change Order		System Wide			

2.12.14.1 Desired Solution
Duplicate the FirstDecision schema on MMIS DSS hardware and establish a load process for all FirstDecision data. Develop universes and reports for the First Health data.

2.12.14.2 Business Impact
None.

2.12.14.3 Technical Specifications
None.

2.12.14.4 Clarifications
Betsy Elrod - This was added as a placeholder change order for Scope changes for the account per Derek Simons on 1/24/2006.

2.12.14.5 Associated Requirements

Requirement ID	Type
None	

2.12.14.6 Associated System Objects

Technical Name	Object Type	Title
No associated system objects found.		

2.12.14.7 Change Order Status

Status	Date
Change Order Written	01/24/2006

2.12.15 fix schema generation - 1982

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
1982	Change Order		System Wide			1

2.12.15.1 Desired Solution
Fix error with generating an XML schema.

Error one - table columns defined as datetime got the integer data type instead of the date data type.
Error two - a table in the Erwin logical model got the data type in the column instead of a column name. This caused the xml_db with the -g option (generate) to core.

Doco is on Core CO 10104. Detail is repeated here.

2.12.15.2 Business Impact
Unable to generate table schema's with the xml_db utility.

2.12.15.3 Technical Specifications
Updates in XMLSchemaGenerate.c
In reviewing the query to get the data type with the new version of ERwin (4.1.4) the datetime data type is returned as DATE. This was not being supported in XMLSchemaGenerate.c and the value was defaulting to integer.

Updates in ERWin.c
Logic to use either the logical or physical column name.

2.12.15.4 Clarifications
No associated clarifications found.

2.12.15.5 Associated Requirements

Requirement ID	Type
None	

2.12.15.6 Associated System Objects

Technical Name	Object Type	Title
No associated system objects found.		

2.12.15.7 Change Order Status

Status	Date
Change Order Written	01/25/2006
Ready for MO Impl	01/25/2006
MO Implementation	03/03/2006

Status	Date
Ready for UAT Impl	03/03/2006

2.12.16 Letters in COLD - 2096

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
2096	Change Order		System Wide			

2.12.16.1 Desired Solution
To send all the letters generated by Letter Generator to COLD.

2.12.16.2 Business Impact
None.

2.12.16.3 Technical Specifications
None.

2.12.16.4 Clarifications
No associated clarifications found.

2.12.16.5 Associated Requirements

Requirement ID	Type
None	

2.12.16.6 Associated System Objects

Technical Name	Object Type	Title
No associated system objects found.		

2.12.16.7 Change Order Status

Status	Date
Change Order Written	02/20/2006

2.12.17 ReportHeader - 2129

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
2129	Change Order		System Wide			

2.12.17.1 Desired Solution
Report header for 4th and 5th line.

2.12.17.2 Business Impact
None.

2.12.17.3 Technical Specifications
None.

2.12.17.4 Clarifications
No associated clarifications found.

2.12.17.5 Associated Requirements

Requirement ID	Type
None	

2.12.17.6 Associated System Objects

Technical Name	Object Type	Title
No associated system objects found.		

2.12.17.7 Change Order Status

Status	Date
Change Order Written	03/02/2006
Closed	03/22/2006

2.12.18 Kentucky Banner - 2153

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
2153	Change Order		System Wide			

2.12.18.1 Desired Solution
Create Kentucky Banner for Home Page for .NET.

2.12.18.2 Business Impact
None.

2.12.18.3 Technical Specifications
Create Kentucky Banner for home page for .NET Loading a Kentucky Banner out to Model Office. This is a sample to be reviewed, may have additional color changes, etc. after review. This included a KY Skin folder, updated portal.ascx, and jpegs. Change in conjunction with this was in iC_MMIS.Template to bring in the KY Skin (done under CO 1967).

2.12.18.4 Clarifications
No associated clarifications found.

2.12.18.5 Associated Requirements

Requirement ID	Type
None	

2.12.18.6 Associated System Objects

Technical Name	Object Type	Title
No associated system objects found.		

2.12.18.7 Change Order Status

Status	Date
Issue Identified	03/10/2006
Construction in Process	03/10/2006
Ready for MO Impl	03/30/2006

2.12.19 promote Core fixes to Kentucky - 2165

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
2165	Change Order		System Wide			

2.12.19.1 Desired Solution
Promote the fixes made in the system-wide code for the Core portal into the Kentucky application. Please see supplemental documentation for details.

2.12.19.2 Business Impact
None.

2.12.19.3 Technical Specifications
None.

2.12.19.4 Clarifications
No associated clarifications found.

2.12.19.5 Associated Requirements

Requirement ID	Type
None	

2.12.19.6 Associated System Objects

Technical Name	Object Type	Title
No associated system objects found.		

2.12.19.7 Change Order Status

Status	Date
Ready for MO Impl	03/16/2006

2.12.20 Report Routing - 2185

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
2185	Change Order		System Wide			

2.12.20.1 Desired Solution
Migrate the PowerBuilder and Unix shell scripts from TennCare to Core to support batch report distribution. The user adds report routing and distribution information via the online panels. The batch process reads this information and routes reports accordingly.

2.12.20.2 Business Impact
Batch report distribution is currently being asked for by Wisconsin and Kentucky. It is a System Wide subsystem function that supports the routing and distribution of reports. It is new functionality from TennCare that is being added into Core.

2.12.20.3 Technical Specifications
Migrate the TennCare PowerBuilder report distribution functionality, including the data model, to Core. The PowerBuilder code and data model definitions are attached to the CO. Also migrate the C-shell from TennCare to Core. Modifications will be necessary to the shell script to use lp instead of lpafp to spool reports to the printer.

2.12.20.4 Clarifications
Promoting associated Core COs to KY - Model release #21 3/30/2006 [Mark Veness]

2.12.20.5 Associated Requirements

Requirement ID	Type
None	

2.12.20.6 Associated System Objects

Technical Name	Object Type	Title
No associated system objects found.		

2.12.20.7 Change Order Status

Status	Date
Assigned - SE	03/22/2006

2.12.21 Waiver Mailing - 2188

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
2188	Change Order		System Wide			

2.12.21.1 Desired Solution
Give EDS permission to invoice for the 700,000 envelopes required for the waiver mailing.

2.12.21.2 Business Impact
None.

2.12.21.3 Technical Specifications
None.

2.12.21.4 Clarifications
This was added as a placeholder change order for Scope changes for the account per Damien Pierre on 3/23/06.

2.12.21.5 Associated Requirements

Requirement ID	Type
None	

2.12.21.6 Associated System Objects

Technical Name	Object Type	Title
No associated system objects found.		

2.12.21.7 Change Order Status

Status	Date
Change Order Written	03/23/2006

2.12.22 Epocrates Mailing - 2189

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
2189	Change Order		System Wide			

2.12.22.1 Desired Solution
Give EDS permission to invoice for the envelopes that were ordered/used for the Epocrates mailing.

2.12.22.2 Business Impact
None.

2.12.22.3 Technical Specifications
None.

2.12.22.4 Clarifications
This was added as a placeholder change order for Scope changes for the account per Damien Pierre on 3/23/06.

2.12.22.5 Associated Requirements

Requirement ID	Type
None	

2.12.22.6 Associated System Objects

Technical Name	Object Type	Title
No associated system objects found.		

2.12.22.7 Change Order Status

Status	Date
Change Order Written	03/23/2006

2.13 Data Element Dictionary

2.13.1 T_AUDIT_PURGE

This table lists all audit tables, each owner, and retention period.

Column Name	Description	Type	Length	Precision	Primary Key
OWNER	This is the owner of the audit table. Valid values are AIM and AIM00.	CHAR	8	0	N
TABLE_NAME	Name of the audit table.	CHAR	30	0	N
RET_PERIOD	Retention period for rows on the audit tables. See entity description for details.	NUMBER	9	0	N

2.13.2 T_AUDIT_TBL_EXCPT

This table lists other database tables that have their audit trail rows on a table with a name other than its own name prefixed with 'A_'.

Column Name	Description	Type	Length	Precision	Primary Key
NAM_TBL_ACTUAL	The name of a database table that has audit trail rows on a table with a name other than 'A_' + its own table name.	CHAR	20	0	N
NAM_TBL_AUDIT	The name of a database table that stores audit trail rows for a table with a name other than its own table name without the 'A_' prefix.	CHAR	20	0	N

2.13.3 T_BATCH_RESTART

This table keeps track of the number of transactions that have been successfully processed and committed during this batch cycle for a particular module set. The module set represents an entire process. The commit frequency is used by a process to determine how often to commit database changes.

Column Name	Description	Type	Length	Precision	Primary Key
NAM_PROGRAM	Name of the module set performing restart logic.	CHAR	8	0	N

Column Name	Description	Type	Length	Precision	Primary Key
QTY_TXN_COUNT	Number of transactions committed for current cycle.	NUMBER	9	0	N
QTY_COMMIT_FREQ	Tells process how often to commit. This value may be changed to increase or decrease the commit frequency for a particular process.	NUMBER	9	0	N

2.13.4 T_CODE_RESOLUTION

Use this table to resolve system assigned keys (SAKs) to the user's online value. It is used in conjunction with the audit trail system to automate the viewing of what a SAK value is defining. i.e., SAK_RECIP is key to the member tables but the users need to know the Member ID (MID) or MEDICAID ID.

Column Name	Description	Type	Length	Precision	Primary Key
NAM_TABLE	The name of the Physical Database Table.	VARCHAR2	32	0	N
NAM_CODE	This is the key field used to retrieve a system key.	CHAR	18	0	N
NAM_USER_VALUE	This is the key field used to retrieve a user value for a system key.	CHAR	18	0	N
NAM_DESC	This is the description of the system key field.	CHAR	18	0	N
CDE_CODE_TYPE	Used to identify data type of the NAM CODE.	CHAR	1	0	N
CDE_RETURN_VALUE	Used to identify data type of the NAM USER VALUE.	CHAR	1	0	N

2.13.5 T_DATE_AGING

Use the Date table to simplify date aging and date math.

Column Name	Description	Type	Length	Precision	Primary Key
DATE_JUL	The true JULIAN date, days since 12/31 4713 b.c.	NUMBER	9	0	N
DATE_INT	Standard integer date i.e. 19941230	NUMBER	9	0	N
DATE_IND	This contains the first letter of the day of the week or 'H' for Holidays.	CHAR	1	0	N
DATE_YTD	This contains the date in year to date, day of year format i.e., YYDDD 94310.	NUMBER	9	0	N

2.13.6 T_GIS_CODES

This table contains the return/accuracy codes from the software that calculates the longitude and latitude based on a given address.

Column Name	Description	Type	Length	Precision	Primary Key
CDE_GIS_QUALITY	This is the return code value from the GIS software (MapInfo).	NUMBER	4	0	Y
NUM_GIS_QUALITY	This is the accuracy code number that we will display to the user. The values will range from 1 to 5.	NUMBER	4	0	N
DSC_SHORT_GIS_QUALITY	Short description of the accuracy of the calculated longitude/latitude.	CHAR	40	0	N
DSC_LONG_GIS_QUALITY	Long description of the accuracy of the calculated longitude/latitude.	VARCHAR2	4000	0	N

2.13.7 T_GROUP

This entity groups several letters under a single topic. This can be used to limit the number of letters seen by an organizational unit.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_GROUP	A unique identifier for the letter group.	NUMBER	9	0	Y
ID_GROUP	The external identifier by which this group is known.	CHAR	20	0	N
DSC_GROUP	A description of the group that indicates the type of letters that are contained within this group.	VARCHAR2	4000	0	N

2.13.8 T_HIST_PART

Each row in this table identifies a schema partition of the Big History claim data.

Column Name	Description	Type	Length	Precision	Primary Key
NUM_PARTITION	A value used in ordering multiple occurrences.	NUMBER	4	0	Y
NAM_SCHEMA	This is the name of the schema to which a partition belongs. The use of a schema allows multiple physical database tables (and the columns on those tables) within the same database instance to have the same name.	CHAR	5	0	N
DTE_HIST_FROM	The from or low date in the range of History partition.	NUMBER	8	0	N
DTE_HIST_TO	The to or high date in the range of History partition.	NUMBER	8	0	N

2.13.9 T_HOLIDAY_DATES

This entity contains the holiday dates and descriptions that fall on weekdays, along with the date of the next business day that follows these holidays.

Column Name	Description	Type	Length	Precision	Primary Key
DTE_HOLIDAY	This field holds the dates for holidays that fall on weekdays.	NUMBER	8	0	N
DTE_NEXT_BUSINESS	This field holds the next business date after a holiday that falls on a weekday.	NUMBER	8	0	N
DSC_HOLIDAY	This field holds the descriptions for holidays that fall on weekdays.	CHAR	30	0	N

2.13.10 T_HS_DIR_PART

Each row in this table identifies a schema partition of the Big History Claim Directory data.

Column Name	Description	Type	Length	Precision	Primary Key
NUM_PARTITION	This is a value used in ordering multiple occurrences.	NUMBER	4	0	Y
NAM_SCHEMA	This is the name of the schema to which a partition belongs. The use of a schema allows multiple physical database tables (and the columns on those tables) within the same database instance to have the same name.	CHAR	5	0	N
DTE_HIST_FROM	The from or low date in the range of History partition.	NUMBER	8	0	N
DTE_HIST_TO	The to or high date in the range of History partition.	NUMBER	8	0	N

2.13.11 T_LETTER

This table contains letters available to be generated in a batch or on-request mode are identified and stored in the table.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_LETTER	This is a unique identifier for the letter.	NUMBER	9	0	Y
NUM_VERSION	This is the version number of the letter.	NUMBER	9	0	N
ID_LETTER	This is the external identifier for this letter.	CHAR	20	0	N
DSC_LETTER	This is a description of the letter and its purpose.	VARCHAR2	4000	0	N

2.13.12 T_LETTER_GROUP

This entity tracks the letters that are a member of a specific group.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_LETTER	This is a unique identifier for the letter.	NUMBER	9	0	Y
SAK_GROUP	This is a unique identifier for the letter group.	NUMBER	9	0	Y

2.13.13 T_LETTER_REQUEST

This table contains the details of all the letter requests submitted.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_LETTER_REQUEST	This is a unique identifier for the letter request.	NUMBER	9	0	Y
ID_CLERK	This is the clerk that requested the letter.	CHAR	8	0	N
DTE_REQUEST	This is the date the letter was requested.	NUMBER	8	0	N

2.13.14 T_LETTER_TEMPLATE

This table contains the details of the letter templates in the database.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_LETTER	This is a unique identifier for the letter	NUMBER	9	0	Y
SAK_LETTER_TEMPLATE	This is a unique identifier for the letter template.	NUMBER	9	0	Y
SAK_QUERY	This is a unique identifier for the query	NUMBER	9	0	N
DSC_TEMPLATE	This is the html markup of the letter that will be generated.	CLOB	0	0	N
IND_AUTOMATED	This is an indicator that determines whether the letter template has been validated and whether it is ready for production use.	CHAR	1	0	N
IND_LANDSCAPE	This indicates whether the letter should be printed in a landscape format. N = portrait, Y = landscape	CHAR	1	0	N
IND_DUPLEX	This indicates whether to double side the letter output. N = single sided, Y = double sided.	CHAR	1	0	N
IND_LETTERHEAD	This indicates the type of letter head to be created when the letter is printed.	CHAR	1	0	N

2.13.15 T_LTR_RQST_TEMPLAT

This table contains the details of the letter requests to be generated per template.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_LETTER	This is a unique identifier for the letter.	NUMBER	9	0	Y
SAK_LETTER_TEMPLATE	This is a unique identifier for the letter template.	NUMBER	9	0	Y
SAK_LETTER_REQUEST	This is a unique identifier for the letter request.	NUMBER	9	0	Y
ID_PEOPLE	This is the identifier of the person or organization you want the letter to go to.	CHAR	256	0	N
DTE_GENERATE	This is the date you want the letter to be printed if the mode is batch.	NUMBER	8	0	N
IND_GENERATE_MODE	This is the mode in which to print the letter (Batch or Online).	CHAR	1	0	N
DSC_EXTRA_DATA	This is extra data that will be included in the letter. This field will typically be used in a batch mode rather than online.	VARCHAR2	4000	0	N

2.13.16 T_LTR_RQST_TRK

This table contains the details of the letters generated.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_LETTER	This is a unique identifier for the letter.	NUMBER	9	0	Y
SAK_LETTER_TEMP	This is a unique identifier for the letter	NUMBER	9	0	Y

Column Name	Description	Type	Length	Precision	Primary Key
LATE	template.				
SAK_LETTER_REQUEST	This is a unique identifier for the letter request.	NUMBER	9	0	Y
NUM_VERSION_QUERY	This is the version number of the query that was used to generate this letter.	NUMBER	9	0	N
NUM_VERSION_LETTER	This is the version number of the letter that was created.	NUMBER	9	0	N
IND_MEDIA_TYPE	This is the media type that the letter was sent on. Valid values will be F-Fax, P-Paper	CHAR	1	0	N
DTE_SENT	This is the date the letter was created and printed.	NUMBER	8	0	N

2.13.17 T_QUERY

A query is run against the database to gather information for a letter template. This entity stores the query and a description of the type of data it gathers.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_QUERY	This is a unique identifier for the query.	NUMBER	9	0	Y
NUM_VERSION_QUERY	This is the version number of the query.	NUMBER	9	0	N
ID_QUERY	This is the external identifier by which a user will refer to this query.	CHAR	20	0	N
DSC_QUERY	This is a description of the query and the data that it will return.	VARCHAR2	4000	0	N

Column Name	Description	Type	Length	Precision	Primary Key
DSC_XML_QUERY	This is the SQL query structured in a "xmlpunld" notation.	CLOB	0	0	N

2.13.18 T_RTE_RPT_CTL

This table is used in the interChange (iCE) development and test environment to emulate the Computer Output to Laser Disk (COLD) facility.

Column Name	Description	Type	Length	Precision	Primary Key
RPT_FILE_NAME	This is the UNIX file name of the report.	VARCHAR	30	0	Y
ENV	This is the environment in which the report is produced (mod/prod etc.)	CHAR	3	0	N
OUT_MEDIA	This is the output media type - crld, print, etc.	CHAR	4	0	N
COLD_NAME	This is the 8 character name from the report banner line.	CHAR	10	0	N
COLD_OPT	This is any option to be passed to the COLD solution software.	CHAR	10	0	N
PRINT_VAR	This is any print variable needed.	CHAR	10	0	N
PRINT_CTL_FILE	This is the name of the print control file, if needed.	VARCHAR	30	0	N
PRT_NUM_CPY	This is the number of copies to be printed.	CHAR	2	0	N
EMAIL_ADDR	This is the email address of the individual or group to be notified when this report is ready.	VARCHAR	40	0	N

2.13.19 T_STATE_ADDR

This table contains the address information for the state health care agency.

Column Name	Description	Type	Length	Precision	Primary Key
NAM_PROGRAM	This identifies a module set or program.	CHAR	8	0	Y
DTE_EFFECTIVE	This is the effective date for the parm to begin.	NUMBER	8	0	Y
DTE_END	This is the date that the parm is no longer in effect.	NUMBER	8	0	N
DSC_30	This is the description of the table entry.	VARCHAR2	30	0	N
NAM	This is the address line containing the state name.	VARCHAR2	50	0	N
ADR_STREET_1	This is the first address line of the address.	VARCHAR2	30	0	N
ADR_STREET_2	This is the second address line of the address.	VARCHAR2	30	0	N
ADR_CITY	This is the Mailing address city.	VARCHAR2	15	0	N
ADR_STATE	This is the Mailing address state.	CHAR	2	0	N
ADR_ZIP_CODE	This is the field used to store the five digit address zip code.	CHAR	5	0	N
ADR_ZIP_CODE_4	This is the field used to store the address zip code plus four.	CHAR	4	0	N

Column Name	Description	Type	Length	Precision	Primary Key
NUM_PHONE	This is the Phone number.	CHAR	15	0	N
NUM_FAX	This is the State Fax Number.	CHAR	15	0	N
NAM_CONTACT	This is the name of the contact person or department.	VARCHAR2	40	0	N

2.13.20 T_SYSTEM_KEYS

This contains the last system assigned keys used for each entity that contains a SAK.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_NAME	This is the name of the system assigned key.	CHAR	18	0	Y
SAK	This is the system assigned internal key used to uniquely identify an entity. Its value is set to the last SAK used.	NUMBER	9	0	N
DSC_SYSTEM_KEY	This is the description of the entity the system assigned key defines.	CHAR	20	0	N
FIELD_NAME	This is the exact column name of the system assigned key in the kernel table where it originates.	CHAR	30	0	N
TABLE_NAME	This is the exact table name where the system assigned key originates. This is often called the "kernel" or "entity" table.	CHAR	30	0	N
SCHEMA_NAME	Contains the name of the Schema of the table where SAK	VARCHAR2	50	0	N

Column Name	Description	Type	Length	Precision	Primary Key
	originates.				

2.13.21 T_SYSTEM_PARMS

This table maintains parms that are needed for system programs to process.

Column Name	Description	Type	Length	Precision	Primary Key
NAM_PROGRAM	This identifies a module set or program.	CHAR	8	0	N
DTE_EFFECTIVE	This is the effective date for the parm to begin.	NUMBER	8	0	N
DTE_END	This is the date that the parm is no longer in effect.	NUMBER	8	0	N
IND	This identifies whether the parm entry is active.	CHAR	1	0	N
DATE_PARM_1	This is the first date used for parm processing.	NUMBER	8	0	N
DTE_PARM_2	This is the second date used for parm processing.	NUMBER	8	0	N
TIME_SYSTEM	This is the time the table data was entered..	NUMBER	8	0	N
TME_ASSIGNED	This is the time the values were assigned to the table.	NUMBER	8	0	N
DSC_30	This contains specific values to be used as system parms.	CHAR	30	0	N

2.13.22 T_SYS_ERR_MSG

This table contains the system error messages used during processing and inquiry.

Column Name	Description	Type	Length	Precision	Primary Key
CDE_MESSAGE	Contains the code number for each error message	NUMBER	9	0	Y

Column Name	Description	Type	Length	Precision	Primary Key
DSC_MESSAGE	Contains the description of the error message	CHAR	50	0	N

3 System Wide Subsystem Appendix A – Letter Generator v2 Training Documentation

The Letter Generator document provides supplemental information for the System Wide subsystem DSD. You may access the documentation via the following links:

https://ddipwb.kymmis.com/KYXIXDDI/Subsystem/utils/FolderList.asp?Folder=../System%20Wide/LG20_Documentation

https://ddipwb.kymmis.com/KYXIXDDI/Subsystem/System%20Wide/LG2.0-Dev&Ops.Guide.0current.pdf

4 System Wide subsystem Appendix B – interChange User Interface Standards

The interChange .NET User Interface Standards Revision 1.3 was produced by the EDS CORE interChange team and is not formatted according to Kentucky documentation standards.

You may access the documentation via the following link:

https://ddipwb.kymmis.com/KYXIXDDI/Subsystem/System%20Wide/InterchangeNET_UserInterfaceStandards.doc

5 System Wide subsystem Appendix C – Additional System Documentation

Additional system documentation is accessible on Project Workbook.

Access the PWB and click the "Tech Design" link on the main menu. Select "System Wide" from the subsystem list on the left side of the Tech Design landing page.

Logical program flow documentation is accessible via the following links near the bottom of the left side of the page:

"Source Code Visio Diagrams" link accesses pages with a bold heading for every program in the system with links under the heading for the associated diagram.

"Source Code Details" link will provide information such as functions, structures. It also includes another path to the Visio diagrams as well as links to the actual Source Code.

Program logic for the .NET framework is accessible via the "DotNet Patterns" link near the bottom of the left side of the page.

6 Appendix of Subsystem Detailed System Designs

This section includes all submitted and approved DSDs. DSDs are presented in the following sequence. Use the PDF bookmarks to navigate to a particular document:

- Security Management;
- Benefit Administration;
- Claims;
- Contact Tracking Management System (CTMS);
- maxMC
- DSS/DW (Decision Support System/Data Warehouse) ;
- EPSDT (Early and Periodic Screening, Diagnosis, and Treatment) ;
- External Data Sharing and Exchange;
- Financial;
- Member Management;
- MAR (Management and Administrative Reporting) ;
- Provider;
- Quality Assurance and Audits;
- Reference;
- Service/Prior Authorization;
- SUR (Surveillance and Utilization Review; and,
- TPL (Third Party Liability).

Benefit Administration Detailed System Design

Kentucky MMIS Project

Cabinet for Health and Family Services
Department for Medicaid Services

March 7, 2008

Cabinet for Health and Family Services Department for Medicaid Services	
Role:	Name:
Author	Amy Sanders
Reviewer	Patti George
EDS Management	Glenn Jennings, Account Manager
Client	Commissioner Elizabeth A. Johnson Deputy Commissioner Carrie Banahan Claims Division Director Rhonda Poston Executive Director/CIO Lorna S. Jones Chief Technical Officer Sandeep Kapoor
DELIVERABLE TITLE: Benefit Administration Detailed System Design	DATE SUBMITTED: March 7, 2008
FILE NAME: KY_MMIS_Benefit_Administration_DSD_REFRESH_v2_FINAL_(03-07-2008).doc	AUTHORING TOOL: MSWord2003

Document Change Log

Version	Changed Date	Changed By	Reason
1.0	10/1/2007	Troy Schexnayder	Updated document with internal revisions
2.0	10/29/2007	Troy Schexnayder	Revisions made according to Commonwealth comments

Table of Contents

1 Introduction 1

2 Benefit Administration Detailed System Design 2

2.1 System Flow 2
2.1.1 interChange System Architecture 2
2.1.2 Benefit Administration System Flow 3
2.2 Subsystem Description 4
2.2.1 Introduction to Benefit Administration 4
2.3 Benefit Administration Processes 5
2.3.1 Process Overview Description 5
2.3.2 Benefit Package Maintenance 7
2.3.3 Service Limitation Maintenance 14
2.3.4 Reference Systems Maintenance 19
2.3.5 Fee-for-Service Payment Rates Maintenance 20
2.4 Data Model 25
2.4.1 Benefit Administration Logical Data Model 25
2.4.2 Benefit Administration Physical Data Model 26
2.4.3 External System Interfaces 27
2.5 Job Scripts 28
2.6 Programs 28
2.7 Internal Web Pages 28
2.8 Reports 28
2.9 Letters 28
2.10 Internal and External Interfaces 28
2.11 Requirement Matrix and Cross Reference 29
2.12 Change Orders 38
2.12.1 Benefit Administration Criteria - 156 38
2.12.2 Create KY Operating Procedures – 212 40
2.12.3 Copy Benefit Plan Functionality - 554 43
2.12.4 Add Limit Time Unit Options - 558 44
2.12.5 Implement Core BP Enhancements - 626 46
2.12.6 ADD index on T_RU_COMPARISON - 7205 48
2.12.7 I3391 - 8021 49
2.12.8 I3420 - 8486 51
2.12.9 I3423 - 8487 53
2.12.10 I3424 - 8488 55
2.13 Data Element Dictionary 57
2.13.1 T_ASNG_HIERARCHY 57
2.13.2 T_ASNG_NODE 59
2.13.3 T_BENEFIT_PLAN_TYPE 59
2.13.4 T_BENEFIT_TYPE 60
2.13.5 T_BNFT_ADJ_FACTOR 60
2.13.6 T_BNFT_ADJ_FACTOR_TYPE 61
2.13.7 T_BNFT_GROUP_STD 62
2.13.8 T_CDE_AID 63
2.13.9 T_CLAIM_TYPE 63
2.13.10 T_CLASSIFICATION_BNFT 63
2.13.11 T_CLM_BNFT_ROLE 64
2.13.12 T_CONDITION 64
2.13.13 T_COUNTY 65
2.13.14 T_DIAGNOSIS 66
2.13.15 T_DRG 66
2.13.16 T_DRUG 67

2.13.17 T_FIN_PAYER_COB 74
2.13.18 T_FINANCIAL_PAYER 75
2.13.19 T_GROUP_STD_BNFT 76
2.13.20 T_HIERARCHY_NODE 77
2.13.21 T_MODIFIER 78
2.13.22 T_MODIFIER_TYPE 79
2.13.23 T_OCCURRENCE 79
2.13.24 T_PATIENT_STAT 80
2.13.25 T_PAYER_HIERARCHY 80
2.13.26 T_PAYER_NODE 82
2.13.27 T_PGM_CT_XREF 82
2.13.28 T_PGM_DEPENDENCY 83
2.13.29 T_PGM_EXCLUSION 83
2.13.30 T_PGM_HIERARCHY 84
2.13.31 T_PLACE_OF_SERVICE 85
2.13.32 T_PRCNG_TYPE_SPEC 86
2.13.33 T_PRICING_IND 86
2.13.34 T_PROC 88
2.13.35 T_PROC_ICD9 89
2.13.36 T_PR_ENROLL_PGM 90
2.13.37 T_PR_PHP_ELIG 91
2.13.38 T_PR_SPEC_CDE 92
2.13.39 T_PR_TYPE_CDE 93
2.13.40 T_PUB_HLTH_PGM 94
2.13.41 T_RATE_TYPE 97
2.13.42 T_REVENUE_CODE 97
2.13.43 T_RE_ELIG 98
2.13.44 T_RE_PMP_ASSIGN 99
2.13.45 T_RU_ACTION 102
2.13.46 T_RU_ARRAY 102
2.13.47 T_RU_COMPARISON 103
2.13.48 T_RU_CONDITION 104
2.13.49 T_RU_DECISION 105
2.13.50 T_RU_DECISION_VAR 106
2.13.51 T_RU_DISCRETE_SET 107
2.13.52 T_RU_DOMAIN 107
2.13.53 T_RU_RELATION 108
2.13.54 T_RU_RELATION_XREF 109
2.13.55 T_RU_RULE 111
2.13.56 T_RU_RULE_ACTION 111
2.13.57 T_RU_TRANSACTION 111
2.13.58 T_RU_TUPLE_SET 112
2.13.59 T_RU_VARIABLE 113
2.13.60 T_TYPE_OF_BILL 114

1 Introduction

This document provides the Benefits Administration Detailed System Design (DSD). The DSD validates the design process and approach to solution, and identifies those items as specified in the Kentucky MMIS (KY MMIS) Request for Proposal (RFP) and as specified by Kentucky Medicaid.

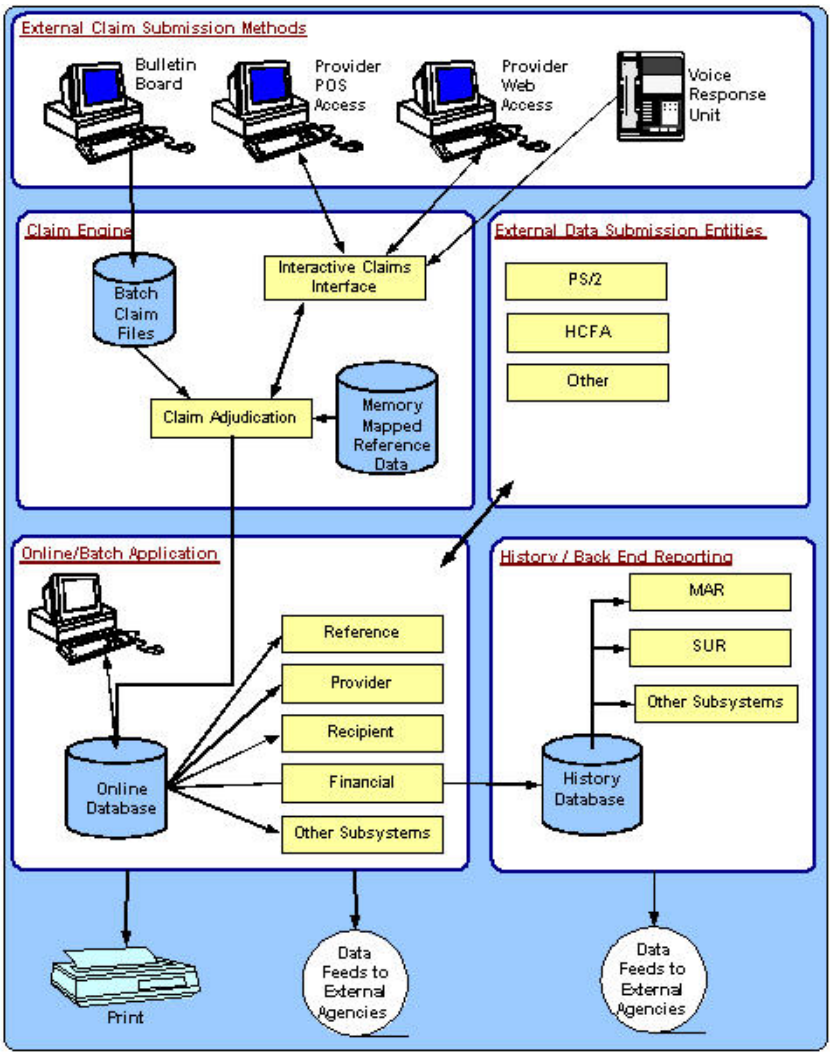
Please note: Benefit Administration owns no system objects (jobs, programs, panel, etc.). All system objects related to the Benefits Administration function are owned and maintained within the Reference Subsystem area and are delivered as part of the Reference Detailed System Design (DSD) document.

2 Benefit Administration Detailed System Design

2.1 System Flow

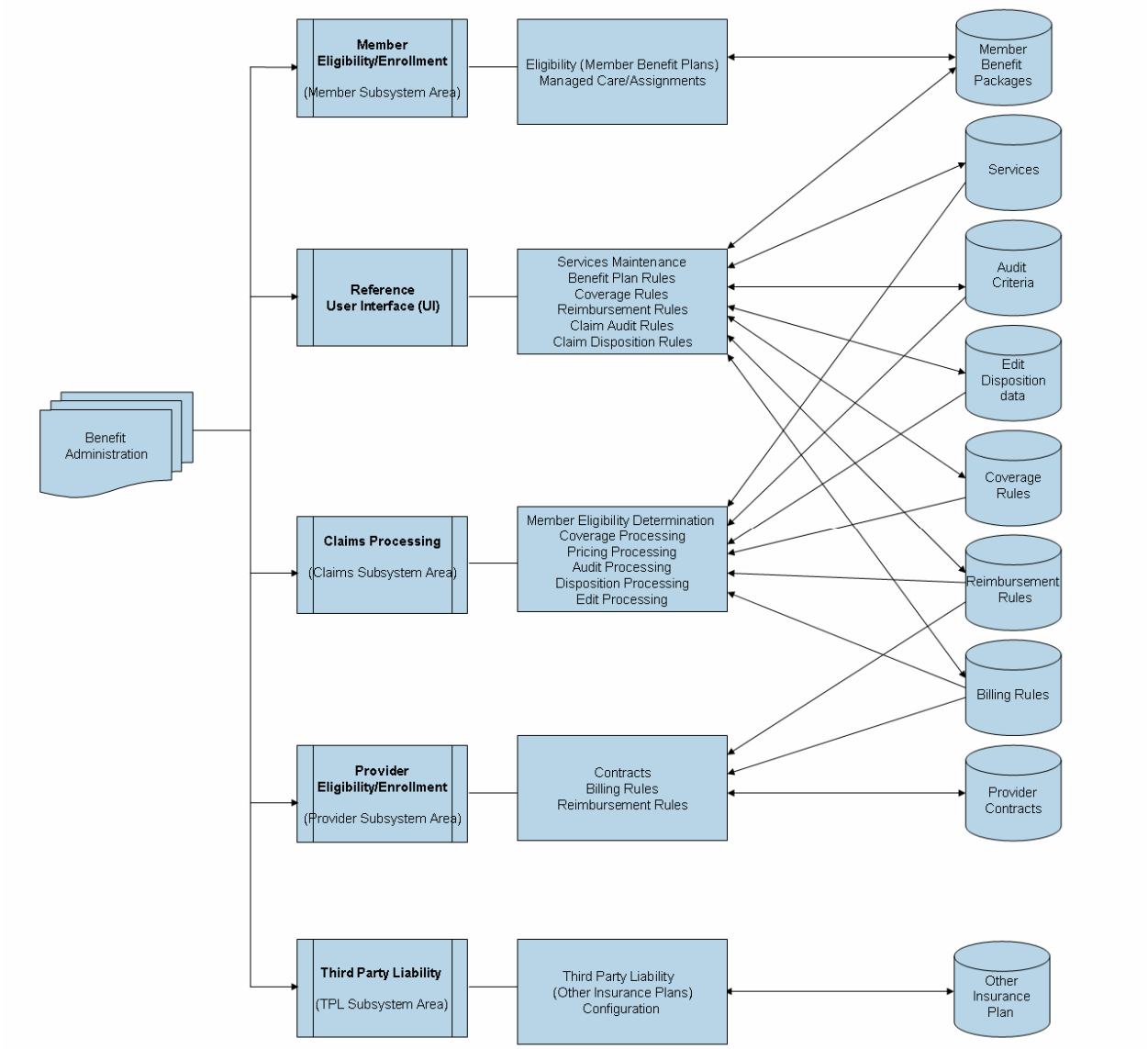
2.1.1 interChange System Architecture

The system is logically divided into four primary components: Claims engine, User Interface, Batch, and the History and Back End Reporting. The Claims engine is responsible for receiving interactive transactions from external sources, adjudicating them, and returning the appropriate response. The User Interface is an N-tier application providing segregated and loosely coupled presentation, business logic, and data logic layers. The user interface provides access to the online subsystem functions through a thin client, the web browser. The Batch component is responsible for maintaining and reporting on data contained within the online database. The History and Back End reporting component is responsible for analyzing, reporting, and supporting the management of the activities that have occurred in the two front end systems. The system interfaces with a variety of data sources which influence processing within the system. The External data submission entities are organizations that supply information to the Medicaid Management Information System (MMIS).



2.1.2 Benefit Administration System Flow

The following diagram represents Benefit Administration input/output processing in a production context (job, jils, directories, and data).



2.2 Subsystem Description

2.2.1 Introduction to Benefit Administration

The primary purpose of the Benefits Administration function is to provide the flexibility to develop and maintain benefit packages required to allow the MMIS to operate as a multi-payer system.

The interChange Benefit Administration functionality incorporates capabilities which allows for multi-payer plans and claim payment based on benefit/member eligibility across multi-payers including Medicaid, Department of Mental Health, Private Insurers, and multiple funding sources within one payer.

The following documents will be used to help identify and define the benefit packages necessary to support Commonwealth policy:

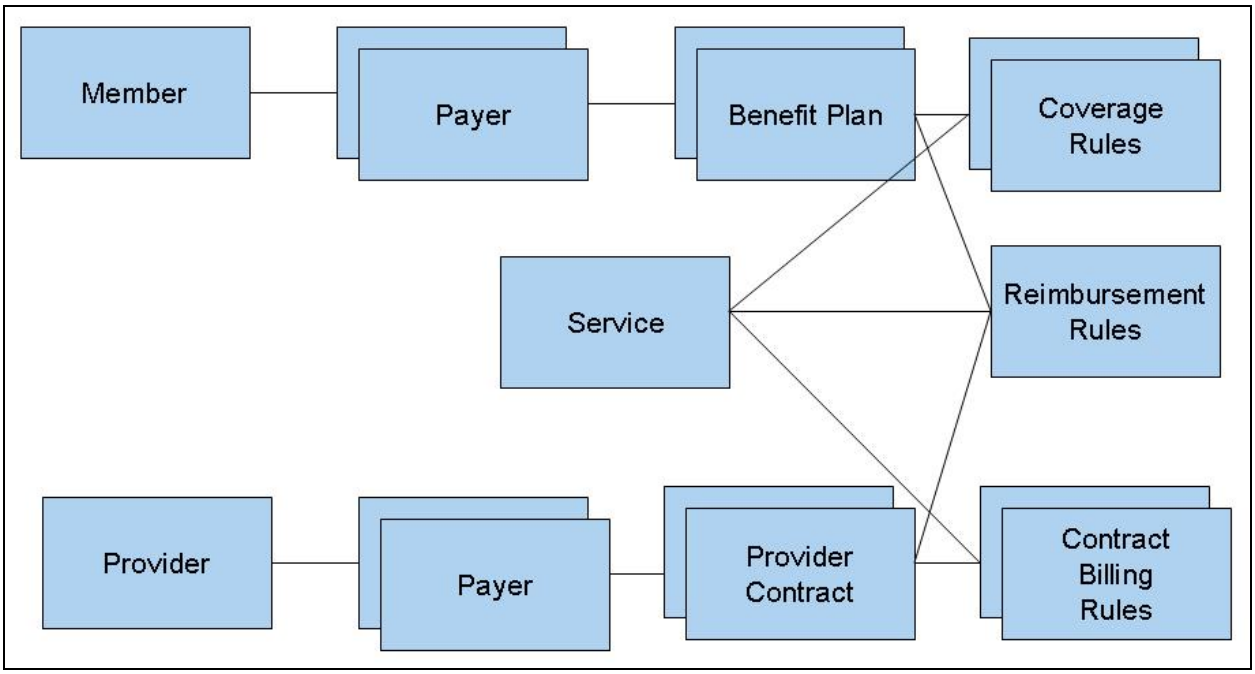
- Benefit Administration Overview consists of a high-level presentation on the basics for Benefit Administration. It includes information on Payers, Benefit Plans, Benefit Coverage Rules, Provider Contracts, Contract Billing Rules, and Contract Reimbursement Rules;
- Medical Assistance (MA) Status Code – Benefit Plan (BP) Crosswalk consists of all the legacy Program and Status codes and the corresponding interChange Benefit plan;
- Benefit Plan Combinations and Exclusions documents relationships between Benefit Plans, such as excluded combinations, allowed combinations, dependencies, etc.; and,
- Benefit Administration Related Claim Edits documents a list of currently identified edits related to Provider Contract Rules, Coverage Rules, Benefit Plan Rules, and Reimbursement Rules. Audits are not included within this list.

2.3 Benefit Administration Processes

2.3.1 Process Overview Description

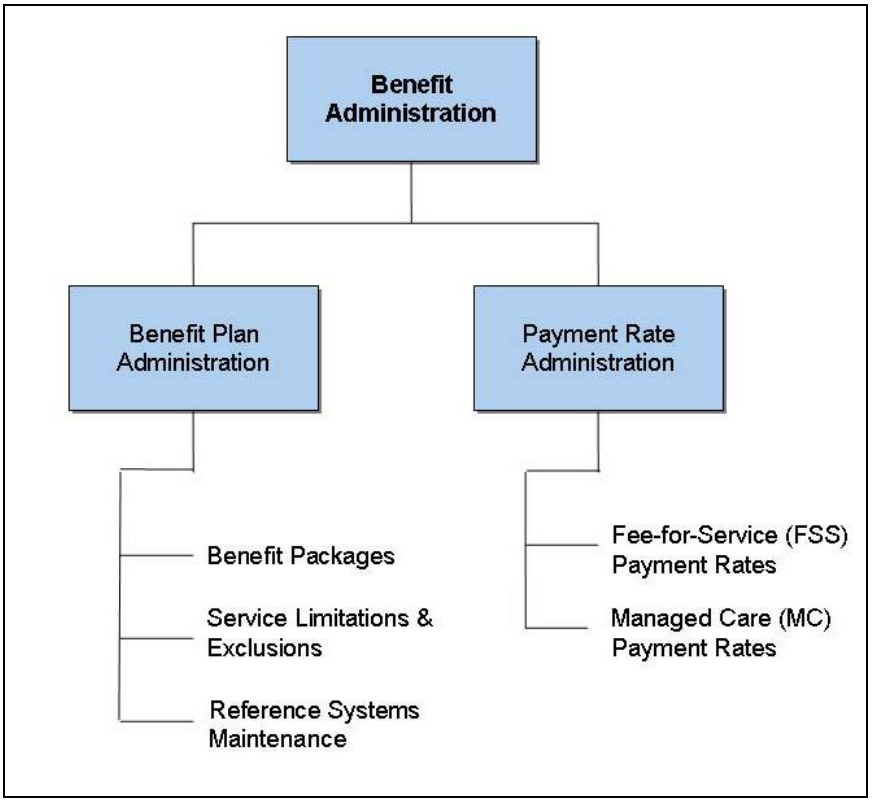
The primary purpose of the Benefits Administration function is to record and maintain information about the services covered under various benefit packages (e.g., Medicaid, Waiver, and others) including scope of services and any associated reimbursement rate information.

The following diagram displays the various components of the Benefit Administration functions:



2.3.1.1 interChange Narrative

The Benefits Administration processes are used to define the benefits that a member can receive and for which providers may be reimbursed. Services (procedure codes, revenue codes, diagnosis codes, and drugs) are grouped into logical units called benefit plans. Within a single benefit plan, authorized users can configure how the individual services will be covered. The various member populations within the Commonwealth are then given eligibility to one or more of these benefit plans, and their claims are adjudicated based on these user-configured rules.



The main processing areas used to support the Benefits Administration functionality are:

- Benefit Plan Administration
  - Benefit Packages;
  - Service Limitations and Exclusions; and,
  - Reference Systems Maintenance.
- Payment Rate Administration
  - Fee-for-Service (FSS) Payment Rates; and,
  - Managed Care (MC) Payment Rates.

(Note: Managed Care processes will be documented in the Managed Care Detailed System Design (DSD) document.)

2.3.2 Benefit Package Maintenance

2.3.2.1 Process Description

The Benefit Package maintenance process defines the various member benefit packages necessary to support the Kentucky Department for Medicaid Services' (DMS) policy. Within a benefit plan, authorized users can configure how the individual services will be covered. Additionally, those users can determine the funding source that will pay for the services covered by the benefit plan. Between benefit plans, users can configure the order in which plans will cover a service and how the plans will interact with each other. The various member populations within the Commonwealth are then given eligibility to one or more of these benefit plans, and their claims are adjudicated based on these user-configured rules.

The Benefit Plan Maintenance function includes the following benefit administration related data:

- Benefit Plans;
- Coverage Rules;
- Reimbursement Rules; and,
- Billing Rules.

2.3.2.2 Benefit Plans

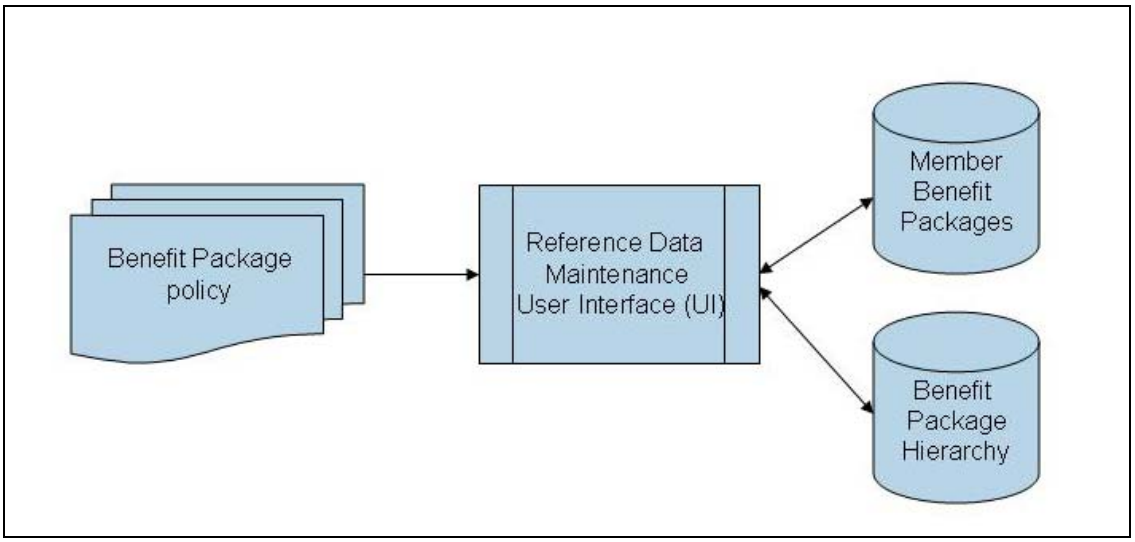
A benefit plan is a classification of benefits a member can receive. For example, TXIX is a Benefit Plan.

The following pages and panels are used to add/maintain Benefit Plan information:

Title	Narrative
Member Plan	Use the Member Plan panel to maintain the list health care plans in the system. There are five parts to the panel, which include: • Dependent Plan Data; • Excluded Plan Data; • Classification Mapping Data; • Benefit Coverage; and, • Benefit Plan Group Type. The dependent plans must exist together on member eligibility with the selected plan. The Excluded plans can not exist with the selected plan. Navigation Path: [Reference - Benefit Administration] - [Member Plan] - [Member Plan]
Benefit Plan Hierarchy.	This window is used to display and maintain benefit plan hierarchy relationships. The benefit plan hierarchy defines the order of processing for members who are enrolled in multiple benefit plans. Navigation Path: [Reference - Benefit Administration] - [Benefit Plan Hierarchy] - [Benefit Plan Hierarchy]

2.3.2.2.1 Process flow

The process flow diagrams below provide a visual representation of the maintenance of Benefit Package information. The Benefit Package information will be maintained through the online User Interface (UI).



2.3.2.3 Coverage Rules

Coverage rules define coverage restrictions for services within a Benefit Plan. For example a service may only be covered for specific age ranges.

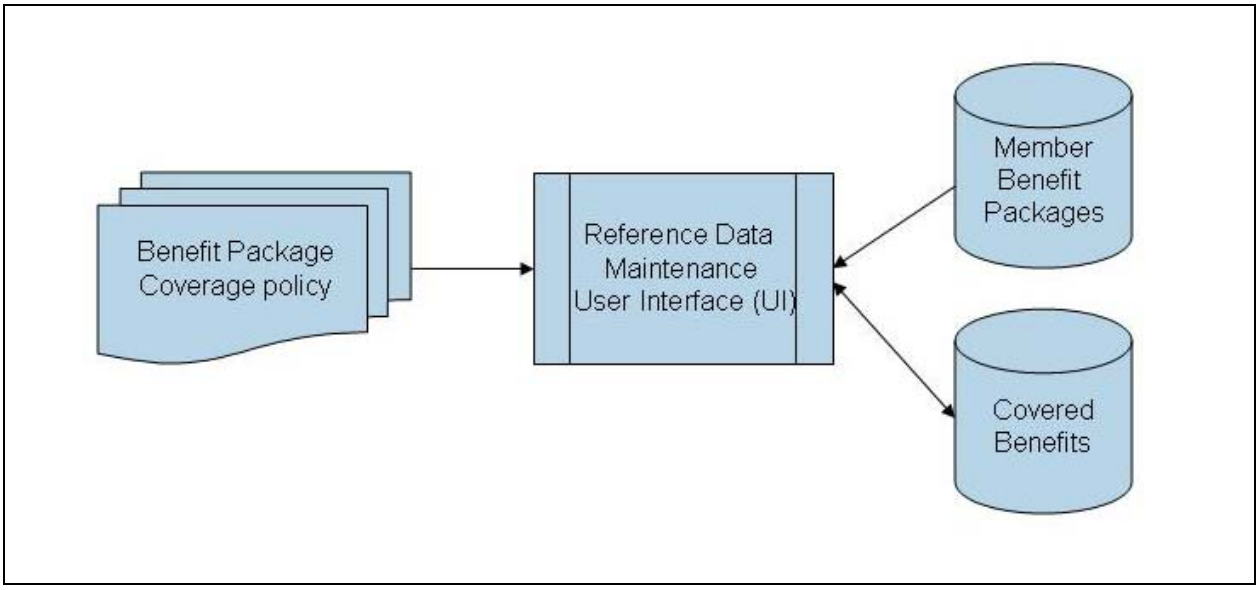
The following panels are used to add/maintain coverage rule information:

Title	Narrative
Benefit Plan Coverage Rules-All Services	Use the Benefit Plan Coverage Rules-All Services panel to view the member plan coverage information for a specific procedure, diagnosis, drug, DRG, and/or revenue code. Navigation Path: [Reference - Diagnosis] - [(select row from search results)] - [Benefit Plan Coverage Rules] OR Navigation Path: [Reference - Revenue] - [(select row from search results)] - [Benefit Plan Coverage Rules] OR Navigation Path: [Reference - Drug] - [(select row from search results)] - [Benefit Plan Coverage Rules] OR Navigation Path: [Reference - Procedure] - [select HCPCS radio button]- [(select row from search results)] - [Benefit Plan Coverage Rules] OR Navigation Path: [Reference - Procedure] - [select ICD-9-CM radio button]- [(select row from search results)] - [Benefit Plan Coverage Rules] OR Navigation Path: [Reference - DRG] - [(select row from search results)] - [Benefit Plan Coverage Rules]

Title	Narrative
Member Plan	Use the Member Plan panel to maintain the list health care plans in the system. There are five parts to the panel, which include: • Dependent Plan Data; • Excluded Plan Data; • Classification Mapping Data; • Benefit Coverage; and, • Benefit Plan Group Type. The dependent plans must exist together on member eligibility with the selected plan. The Excluded plans can not exist with the selected plan. Navigation Path: [Reference - Benefit Administration] - [Member Plan] - [Member Plan]

2.3.2.3.1 Process flow

The process flow diagrams below provide a visual representation of the maintenance of Coverage Rules information. The Benefit Package information will be maintained through the online User Interface (UI).



2.3.2.4 Reimbursement Rules

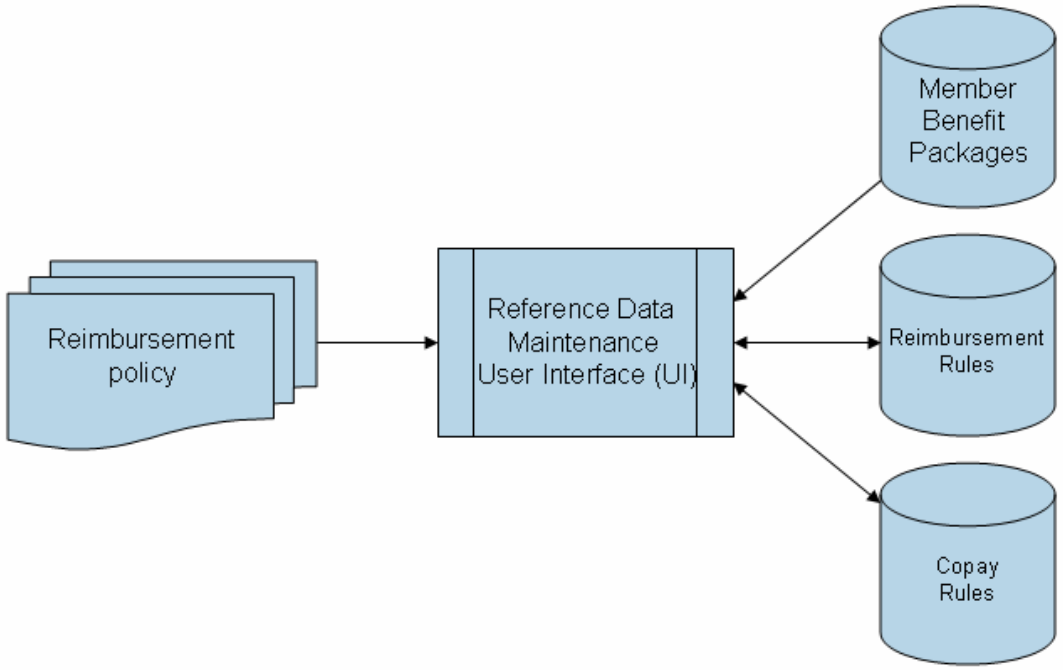
Reimbursement rules are rules for which a Provider can be reimbursed. For example, reimburse the Provider for a specific Benefit Plan and Provider Contract. The reimbursement rules also specify the pricing methodology to be used for the provider's reimbursement.

The following panels are used to add/maintain reimbursement rule information:

Title	Narrative
Copay	Use the Copay panel to maintain the rules by which copay can be calculated during claims processing. Navigation Path: [Reference - Benefit Administration] - [Copay]
Reimbursement Agreement	Use the Reimbursement panel to maintain the rules for which a Provider can be reimbursed for a service. Navigation Path: [Reference - Benefit Administration] -[Reimbursement Agreement] - [Reimbursement Agreement]
Reimbursement Agreement-All Services	Use the Reimbursement Rules-All Services panel to view the benefit plan reimbursement information for various service codes supported by the MMIS including: procedures, diagnosis, drug or revenue codes. The reimbursement rules are used to determine the pricing methodology to use when adjudicating a claim. Navigation Path: [Reference - Diagnosis] - [(select row from search results)] - [Reimbursement Rules] OR Navigation Path: [Reference - Drug] - [(select row from search results)] - [Reimbursement Rules] OR Navigation Path: [Reference - Procedure] - [select HCPCS radio button]- [(select row from search results)] - [Reimbursement Rules] OR Navigation Path: [Reference - Procedure] - [select ICD-9-CM radio button]- [(select row from search results)] - [Reimbursement Rules] OR Navigation Path: [Reference - Revenue] - [(select row from search results)] - [Reimbursement Rules] OR Navigation Path: [Reference - DRG] - [(select row from search results)] - [Reimbursement Rules]

2.3.2.4.1 Process flow

The process flow diagrams below provide a visual representation of the maintenance of Reimbursement Rules information. The Reimbursement rules information will be maintained through the online User Interface (UI).



2.3.2.5 Billing Rules

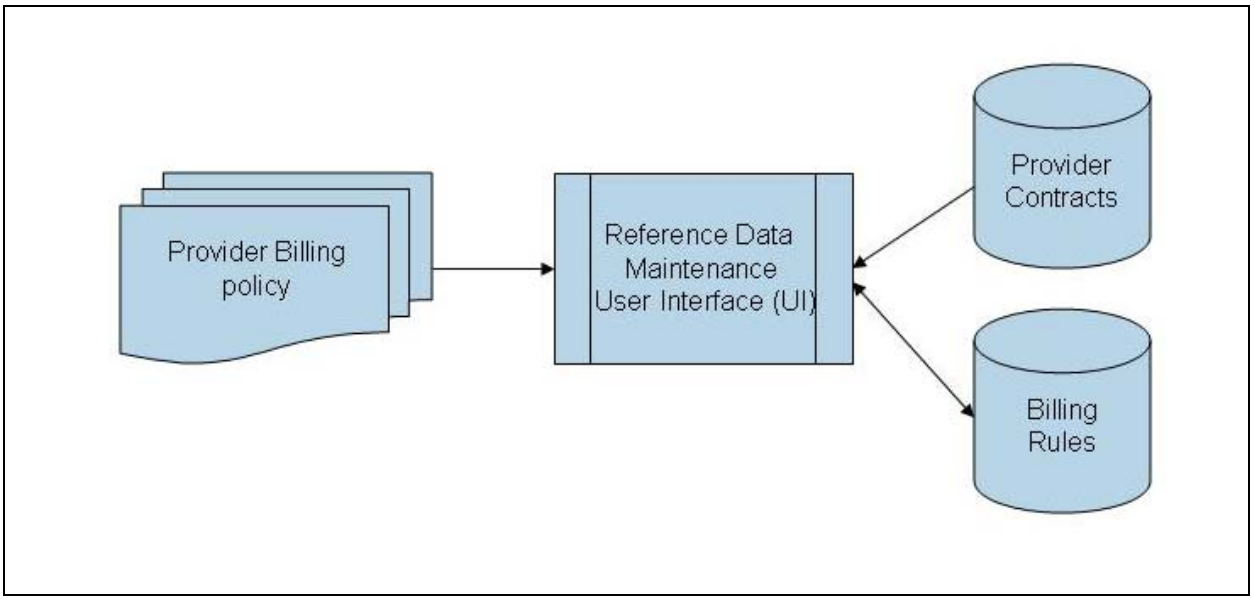
Billing Rules are rules for a Provider within a contract. For example, a Provider can only bill certain claim types.

The following panels are used to add/maintain Billing rule information:

Title	Narrative
Provider Contract	Use the Provider Contract panel to maintain the list of Provider Contracts in the system and to classify the services a Provider may bill. Navigation Path: [Reference - Benefit Administration] - [Provider Contract] - [Provider Contract]
Contract Billing Rules-All Services	Use the Contract Billing Rules-All Services panel to display provider contracts applicable to the benefit (Diagnosis, DRG, Drug, ICD9, HCPCS, and Revenue Code). Navigation Path: [Reference - Diagnosis] - [(select row from search results)] - [Contract Billing Rules] OR Navigation Path: [Reference - Procedure] - [select HCPCS radio button]- [(select row from search results)] - [Contract Billing Rules] OR Navigation Path: [Reference - Procedure] - [select ICD-9-CM radio button]- [(select row from search results)] - [Contract Billing Rules] OR Navigation Path: [Reference - Revenue] - [(select row from search results)] - [Contract Billing Rules] OR Navigation Path: [Reference - Drug] - [(select row from search results)] - [Contract Billing Rules] OR Navigation Path: [Reference - DRG] - [(select row from search results)] - [Contract Billing Rules]

2.3.2.5.1 Process flow

The process flow diagrams below provide a visual representation of the maintenance of Billing Rules information. The provider billing rules information will be maintained through the online User Interface (UI).



2.3.3 Service Limitation Maintenance

2.3.3.1 Process Description

The Benefit Package maintenance process defines the various member benefit packages necessary to support the Kentucky Department for Medicaid Services' (DMS) policy. Within a benefit plan, authorized users can configure how the individual services will be covered. Additionally, those users can determine the funding source that will pay for the services covered by the benefit plan. Between benefit plans, users can configure the order in which plans will cover a service and how the plans will interact with each other. The various member populations within the Commonwealth are then given eligibility to one or more of these benefit plans, and their claims are adjudicated based on these user-configured rules.

The Service Limitations and Exclusions utilize the following functional areas:

- Coverage Rules;
- Billing Rules; and,
- Edits and Audits.

2.3.3.2 Coverage Rules
See the Coverage Rules information under the Benefit Package Maintenance section of the Benefits Administration Process Model documentation above.

2.3.3.3 Billing Rules
See the Billing Rules information under the Benefit Package Maintenance section of the Benefits Administration Process Model documentation above.

2.3.3.4 Edits and Audits
Edit and Audit criteria is used to enforce Kentucky Department for Medicaid Services' (DMS) policy when adjudicating claims and encounters. The interChange system allows authorized users to configure how services are to be administered within specific benefit plans, including limitations on services authorized under each benefit package and services included or excluded for each benefit package.

The following panels are the main Edit/Audit panels related to enforcing Benefit Administration functions. A complete list of Edit/Audit related pages and panels will be delivered with the Reference Detailed System Design (DSD) document.

Title	Narrative
Error Disposition - Disposition Criteria	Detail disposition information concerning a specific edit/audit. The correct disposition detail to use for a claim will be determined based on the claim type, level of care, outcome (full failure or cutback), provider specialty, date of receipt, and date of service of the claim. If there is not an exact match on claim type or provider specialty, the defaults of 0 and 00 are used. Navigation Path: [Reference - Error Disposition Search] - [(Add button) OR (select row from search results)] - [Disposition Criteria]
Audit Criteria -Error Disposition	The AuditCritPanel contains general information concerning a specific audit. This information includes whether diagnosis codes, provider specialties, member levels of care, member ages, and member programs are part of the audit criteria. Navigation Path: [Reference - Error Disposition Search] - [(Add button) OR (select row from search results)] - [(Expand Error Disposition on left by clicking +)] - [Audit Criteria] - [Audit Criteria Base]
Audit Program-Error Disposition	The State and EDS will use the Audit Health Program/Benefit Plan Restrictions panel to identify which health coverage programs/benefit plans are to be included in or excluded from the audit criteria. Navigation Path: [Reference - Error Disposition Search] - [(Add button) OR (select row from search results)] - [(Expand Error Disposition on left by clicking +)] - [Audit Restriction] - [Health Program(only available if Program on Audit Criteria Base is "Include" or "Exclude"]

Title	Narrative
Audit Parameters(Limitation)-Error Disposition	This panel contains the limitation audit parameters, including the time span audited, the number of allowed units and/or dollars, and whether the audit is restricted to particular places of service. Navigation Path: [Reference - Error Disposition Search] - [(Add button) OR (select row from search results)] - [(Expand Error Disposition on left by clicking +)] - [Audit Criteria] - [Audit Parameters (only available if the Audit Type is Limit)]
Audit Parameters(Contra-Indicated)-Error Disposition	This panel contains information about auditing performed for services that contradict services that have already been paid for a member. Also known as relationship auditing criteria. Navigation Path: [Reference - Error Disposition Search] - [(Add button) OR (select row from search results)] - [(Expand Error Disposition on left by clicking +)] - [Audit Criteria] - [Audit Parameters (only available if the Audit Type is Contra-Indicated)]
Audit Procedure (Contra)-Error Disposition	The State and EDS will use the Audit Procedure Restriction - Contra panel to view, add, or update procedure codes and their modifiers or revenue codes that should be included when performing Contra-Indicated audits. Navigation Path: [Reference - Error Disposition Search] - [(Add button) OR (select row from search results)] - [(Expand Error Disposition on left by clicking +)] - [Audit Restriction] - [Procedure (only available if Procedure on Audit Criteria Base is "Include" or "Exclude" and Audit Type is Contra-indicated]
Form Edits	Use the Form Edits page to maintain the rules for predefined set of decisions such including Dental Claim Form Header edits, Detail Claim Form Detail edits, Physician Claim Form Header edits etc. Navigation Path: [Reference - Benefit Administration] -[Form Edits] - [Form Edits]

2.3.3.4.1 Additional Edit/Audit information

An edit is defined as a verification of claim data on the submitted claim. These edits consists of checks for required presence, format, consistency, reasonableness, and allowable values. The way a claim dispositions when it sets an edit in interChange system is developed and maintained through a GUI window application that allow a user to include the criteria that determines what disposition the claim will take when it encounters an edit during claim processing. The criteria to determine what disposition the claim will take when it sets an edit during claim processing include: claim region, claim type, member service program, status of the edit (active, inactive), provider type, provider specialty, claim effective date of service, claim effective date of receipt, and the location where the claim will suspend to if the disposition of the edit is set to suspend.

An audit is defined as a verification of claim data against other claims in history. These audits consist of checks for duplicate services, service limits being exceeded, and billing more than once for a once-in-a-lifetime service. The process to determine the way a claim dispositions when it sets an audit) is developed and maintained using the same GUI interface as used in building edits. In addition, in interChange Error Status Code processing function there are windows that allow a user to develop the audit criteria based on claim data elements including: member service program, claim type, provider type, provider specialty, member age, procedure codes, diagnosis codes, revenue codes, type of bill, NDC, therapeutic class, Generic Price Indicators, GCN, GCN sequence number, and the HICL (hierarchal ingredient code list) sequence number. The design allows the user to build lists of data elements as the audit criteria. The system allows the data element definition to be either positive or negative. This is accomplished through the use of the inclusion or exclusion indicator on the audit definition.

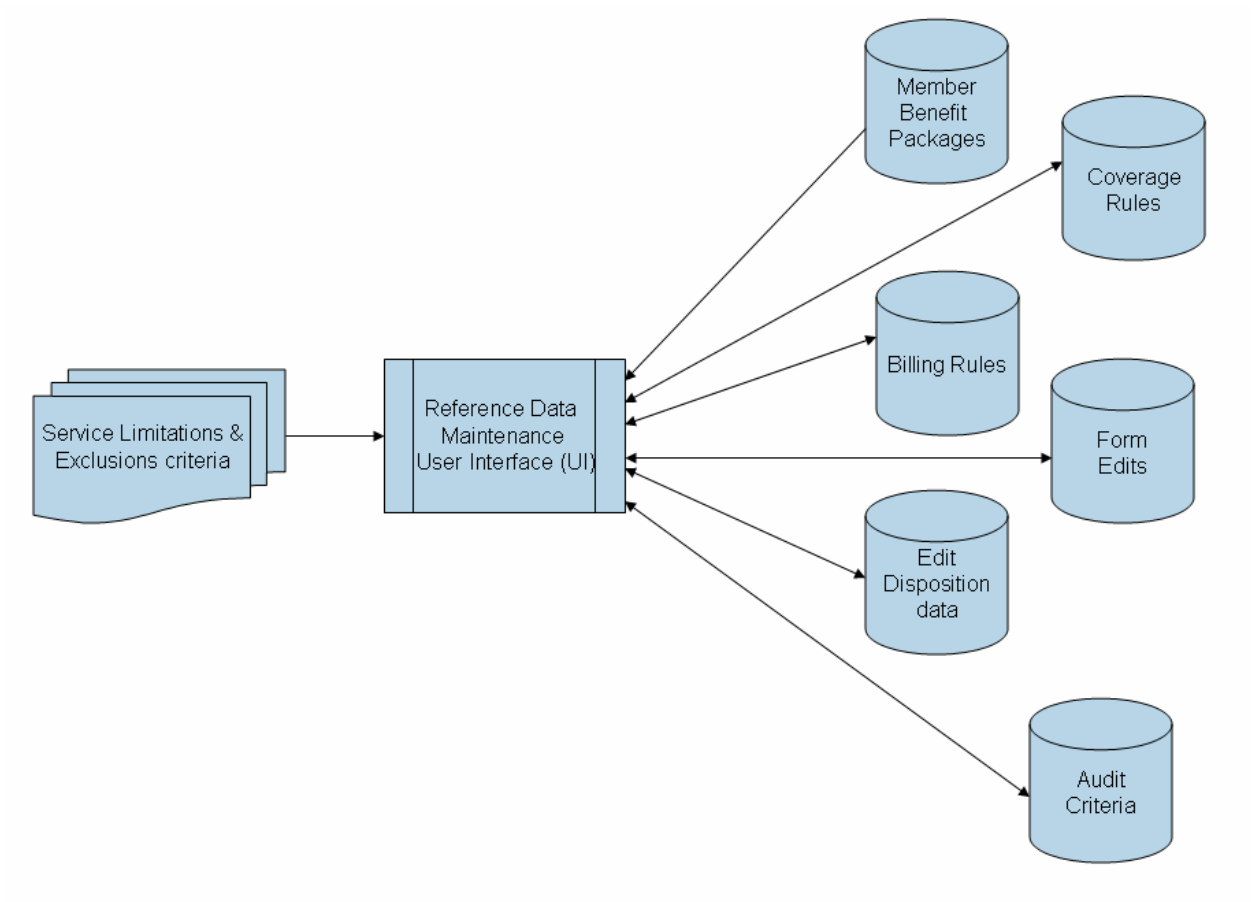
The system supports the audit types listed below:

- Duplicate Checking – This type of audit will compare current claim detail or header information (depending on the claim type) with claims information on history. Duplicate audit will post against a claim if the current claim information matches history either exactly or partially.
- Limitation – This type of audit is used to set dollar or unit limitations on services (procedure code, revenue code, or NDC) based on defined criteria.
- Contra-Indicated – This type of audit is used to limit services based on relationships between procedure codes on history and the one on the current detail of the claim. If a history procedure is indicated on the audit as a history type of procedure, and the procedure on the current detail is indicated on the audit as a current procedure, then the relationship is met and the audit will be set.
- Negative Contra-Indicated – This type of audit is used to limit services based on relationships between procedure codes on history and the one on the current detail of the claim. If a history procedure is indicated on the audit as a history type of procedure, and the procedure on the current detail is NOT indicated on the audit as a current procedure, then the relationship is not met and the audit will post against the claim.
- Umbrella – This type of audit is used to limit pre-operative and post-operative care for surgical procedures. The surgical procedures that apply to a specific umbrella audit are identified by the number of days in the global surgery field on the Medicare fee schedule. The pre-operative and post-operative care procedures are defined on the audit criteria. If either a surgery or a pre-operative/post-operative care procedure code is billed on the current claim detail, and the other type of procedure is on a history claim the audit will post against the claim.
- Bundling – This type of audit is used to identify current claims that are being billed with procedure codes that are considered to be components of a global procedure when the global procedure has already been paid on history.
- Unbundling – This type of audit is used to identify current claims that are being billed with procedure codes that are global procedures when a component procedure of that global procedure has already been paid on history.
- Cross Claim Type – Conflict audit compare dates between services on different claims types to verify that there are no conflicting services for overlapping dates.

- Step Therapy – This audit determines that all criteria are met for a drug listed as part of the step therapy program. Each drug in the program is assigned to one of five step therapy levels. Each level has criteria for number of prescriptions, number of days, and continuous coverage. In order to step up to a higher-level drug, the member's drug history must have fulfilled the step therapy level's criteria requirements.

2.3.3.4.2 Process flow

The process flow diagrams below provide a visual representation of the maintenance of Service Limitations and Exclusions information. The Service Limitation and Exclusion rules will be maintained through the online User Interface (UI).



2.3.4 Reference Systems Maintenance

2.3.4.1 Process Description

The Reference Systems Maintenance process maintains the various code tables and criteria tables necessary to support the Kentucky Department for Medicaid Services' (DMS) policy.

The Reference Systems Maintenance function includes the following benefit administration related data:

- Service Codes;
- Coverage Rules;
- Billing Rules;
- Reimbursement Rules;
- Edits and Audits; and,
- Pricing Data.

2.3.4.2 Service Codes

The Reference System Maintenance area maintains the following service codes:

- Procedure codes (HCPCS, CPT-4, and ADA);
- Diagnosis codes;
- National Drug Codes (NDC);
- Revenue codes;
- Surgical and Ancillary Procedure codes (ICD-9 Procedures);
- Diagnosis Related Group (DRG) codes; and,
- Modifiers.

Other miscellaneous Benefit Plan Administration related codes sets maintained in the reference system area include:

- Benefit Plans;
- Place of Service (POS);
- Explanation of Benefits (EOB);
- Type of Bill (TOB);
- Occurrence codes;
- Value codes;
- Pricing Indicators;
- Claim Types; and,

- Condition codes.

Details about the above code sets will be delivered as part of the Reference Data Maintenance Detailed System Design (DSD) document.

2.3.4.3 Coverage Rules
See the Coverage Rules information under the Benefit Package Maintenance section of the Benefits Administration Process Model documentation.

2.3.4.4 Billing Rules
See the Billing Rules information under the Benefit Package Maintenance section of the Benefits Administration Process Model documentation.

2.3.4.5 Reimbursement Rules
See the Reimbursement Rules information under the Benefit Package Maintenance section of the Benefits Administration Process Model documentation.

2.3.4.6 Edits and Audits
See the Edits and Audits information under the Service Limitation Maintenance section of the Benefits Administration Process Model documentation.

2.3.4.7 Pricing Data
See the Pricing Data information under the Fee-for-Service section of the Benefits Administration Process Model documentation.

2.3.5 Fee-for-Service Payment Rates Maintenance

2.3.5.1 Process Description
The Fee-for-Service Payment Rates maintenance process defines the various rules and pricing data necessary to support the Kentucky Department for Medicaid Services' (DMS) provider reimbursement policy for fee-for-service (FFS) processing. The reimbursement rules describe how a service (procedure, drug, DRG, etc.) should be priced and the pricing data to be used.

The Fee-for-Service Payments rates include the following related data:

- Reimbursement Rules; and,
- Pricing Data.

2.3.5.1.1 Reimbursement Rules

See the Reimbursement Rules information under the Benefit Package Maintenance section of the Benefits Administration Process Model documentation above.

2.3.5.1.2 Pricing Data

Pricing data is used to enforce Kentucky Department for Medicaid Services' (DMS) policy when determining provider reimbursement for fee-for-service claims. The interChange system allows authorized users to configure how services are to be reimbursed within specific benefit plans, including utilizing different pricing methods and rates for a covered service.

The following panels are the main pricing panels related to the Benefit Administration functions. A complete list of pricing related pages and panels will be delivered with the Reference Detailed System Design (DSD) document.

Title	Narrative
Pricing	Pricing indicator specifies the payment methodology that should be applied or was applied when determining the payment to be made to a provider for providing a service (procedure). Navigation Path: [Reference - Related Data] - [Codes] - [Pricing]
Rates - DRG	This panel is used to update, delete and add rates for the Diagnosis Related Group (DRG) codes. Navigation Path: [Reference - DRG - Search] - [(select row from search results)] - [Rates]
Max Fee	The Commonwealth and EDS will use the Max Fee panel to view, add, or update the max fee allowed amount and relative value units for a procedure code or procedure code/modifier combination. Navigation Path: [Reference - Procedure] - [(select HCPCS]) - [(select row from search results)] - [(select Max Fee)]
Codes-Rate Type Code Maintenance	This panel is used to maintain the list of valid rate type codes used in Claims Pricing. Navigation Path: [Reference - Related Data] - [Codes] - [Rate Type]
RBRVS Maintenance	The Commonwealth and EDS will use the RBRVS panel to maintain information related to RBRVS pricing for a specific procedure code. Resource Based Relative Value Scale (RBRVS) pricing methodology information for a procedure. The main source for this data will be the Medicare Fee Schedule Data Base (MFSDB). However, system users may also provide RBRVS pricing information. Navigation Path: [Reference - Procedure] - [(select row from search results)] - [RBRVS]

Title	Narrative
Modifier Information	This panel is used to view Modifier Information. [Reference - Modifier] - [Search] - [Click on any row provided on the list]
Modifier Rate Data	This panel is used to maintain modifier rate data. Processing modifiers are used to change or determine the allowed amount assigned to a procedure. The procedure allowed amount may be calculated or modified with the use of the percentage attribute, amount attribute or quantity attribute. Navigation Path: [Reference - Modifier] - [(select row from search results)] - [Base Information] – [Rate]
Revenue Flat Fee	The State and EDS will use the Revenue Flat Fee panel to view or update specific revenue codes that have a different reimbursement rate based on emergency status. By clicking on the add button the user can add a new date segment or modify the existing one. Navigation Path: [Reference - Revenue] - [(select row from search results)] - [Flat Fee]
Provider Customary Charge	The Provider Customary Charge panel is used to update Usual Customary Charge (UCC) rates. Navigation Path: [Provider - Search] - [select row from search results] - [Customary Charge]

2.3.5.1.3 Additional Pricing information

The Benefits Administration and Reference Data Maintenance processes support the claims pricing function needed for accurate pricing of the various claim types listed below, which are processed within the interChange system:

- Pharmacy Pricing – The system prices both compound and pharmacy (non-compound) drug claims. Both types of pharmacy claims are priced using the appropriate per unit rate for the NDC and date of service billed. The rate is multiplied by the number of units billed to determine the allowed charge. A dispensing fee is added to the allowed price calculated for the claim. This allowed charge is then typically compared to the billed amount, and the lower of the two amounts is determined as the claim's allowed charge. Deductions that may apply to these claims include third party insurance payments, and Medical Assistance Program copay amounts. These amounts are deducted from the claim's allowed charge to determine the amount to be paid on the claim.

- Family Planning Pricing – Family Planning claims are priced for eligible family planning services. Only certain approved procedure codes are eligible for reimbursement. Eligible services are priced using the pricing method established for professional claims, which is described below. Deductions for third party insurance payments and copays may apply.

- Professional Claim Pricing – Professional claims cover a wide range of services. These include transportation services, funeral services, dental services, physician and nurse practitioner services, and lab services, to name a few. Professional claims are billed using procedure codes and procedure code modifiers. Pricing calculates the allowed amount for each service detail of the claim by multiplying the appropriate rate by the number of units allowed. In addition to the rate and number of units, the calculation of the detail allowed amount takes into consideration the member's benefit plan, the rate type associated with the service program, and the processing and pricing modifiers billed on the claim. After calculation of the initial detail allowed charge, pricing and processing modifiers are applied to increase or decrease the detail allowed charge based on an amount or a percentage, as defined within the user-maintained modifier windows.

- Long Term Care Claim Pricing – Long Term Care (LTC) claims are submitted by LTC facilities and reflect the full realm of services provided to the members who live within each facility. The LTC pricing methodology supports the payment of a provider-specific per diem amount for each covered day of facility care. Additionally, therapeutic leave days are paid up to a maximum number of days per calendar year, and hospital bed hold days are paid up to a maximum per period of hospitalization, as established. The claim allowed amount is calculated by multiplying the facility and therapeutic leave days times the facility's per diem rate plus the hospital leave days times 1/3 of the facility's per diem (full per diem if ICF/MR) and the amount calculated for the MA share of the Medicare coinsurance days. Patient payment responsibility amounts are deducted from the sum of the per diem allowed amounts for the facility and therapeutic leave days, plus MA coinsurance share, but not the hospital leave days. Reductions to patient payment amounts include health insurance premiums; expenses applied to pharmacy services and other medical services, and must be submitted on the claim. The per diem rates on file are provider-specific. Third party payments are applied to all but the hospital leave days, when applicable.

- Inpatient Claim Pricing – Inpatient claims are priced using either a provider-specific per diem method, or the DRG pricing method. Pricing at a per diem rate applies to certain types of Rehab and Psych hospitals. Most other inpatient claims are priced based on the DRG pricing method, with some special case and exception logic applied. Pricing for Cost Outliers and Day Outliers are supported.

- Outpatient Claim Pricing – The majority of outpatient claims are priced based on the procedure code pricing method.

- Encounter Claim Pricing – Encounter claims are priced using fee-for-service pricing methods, when they can be determined. These claims store the calculated price on the claim but no payment is made to the provider. When the system is unable to establish a price for an encounter claim, the encounter is priced at zero.

- Crossover Claim Pricing – For Crossover claims, the fee-for-service allowed amount for the procedure and modifier code are calculated at the claim detail level. Subsequently, comparisons and calculations for coinsurance, deductible, and Medicare-approved amounts are made to determine a payable amount for a claim that does not exceed the lower of the co-insurance and deductible billed, or Medical Assistance fee-for service allowed amount. Deductions may apply.

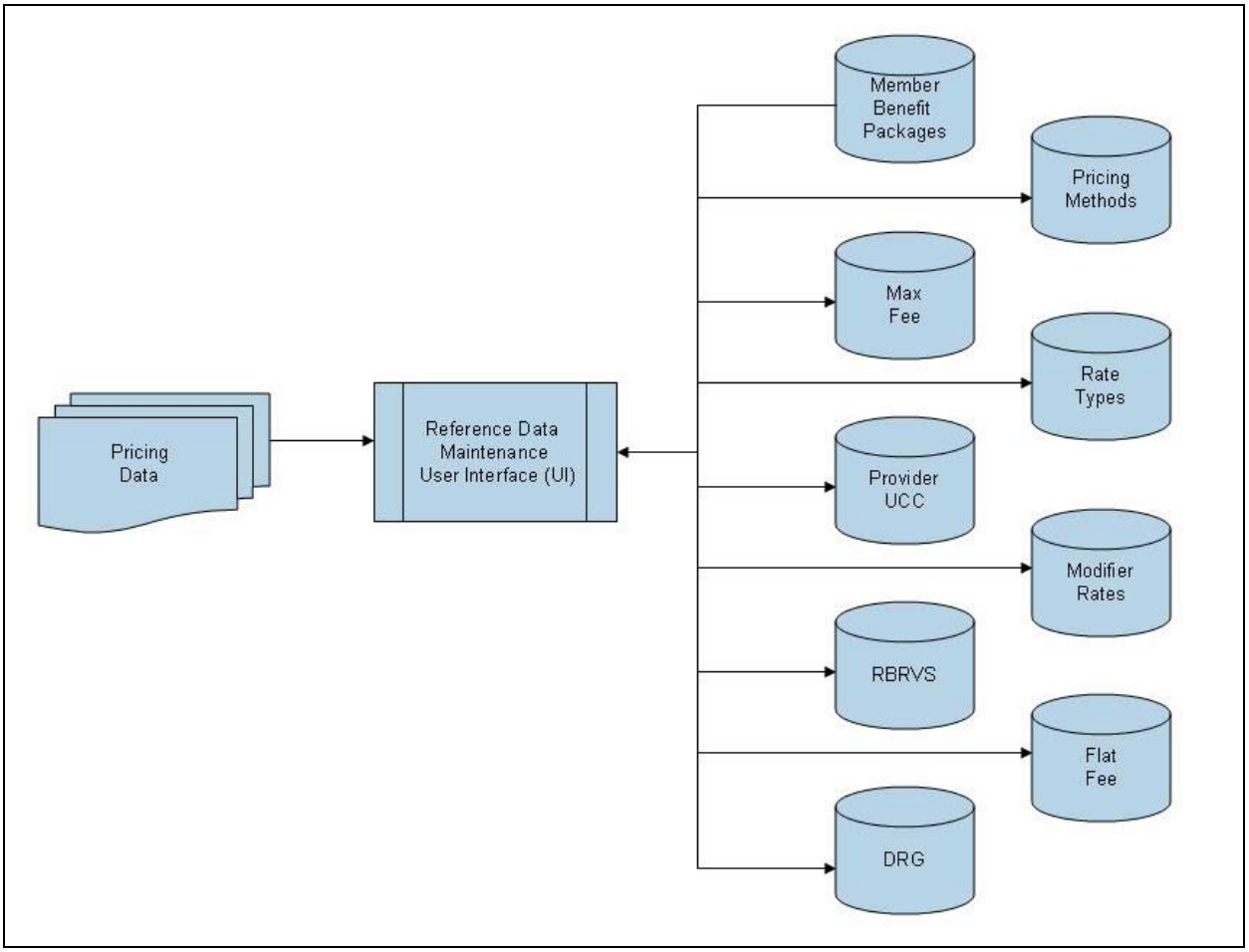
Certain deductions may be applied to the computed Medical Assistance-allowed amount, depending on the claim and member criteria. These deductions include:

- Copay – This deduction is applied to claims, when applicable, and is based on defined policy.
- Patient Pay / Patient Liability – The patient payment/patient liability amount for a member is stored, and is applied during claims pricing based on the amount submitted on the claim. If the claim includes a patient payment amount, this amount is subtracted in the calculation of the payment amount.
- TPL – Third Party Liability is applied based on the amount submitted on the claim. A TPL amount on the claim is subtracted in the calculation of the payment amount.

Modifications and additions to the pricing methodologies needed to support Kentucky Department for Medicaid Services' (DMS) fee-for-service pricing policy will be identified and documented during Claims pricing workgroups. Required changes identified during the claims pricing workgroups will be documented in Change Orders (CO) and assigned to the appropriate subsystem area(s).

2.3.5.1.4 Process flow

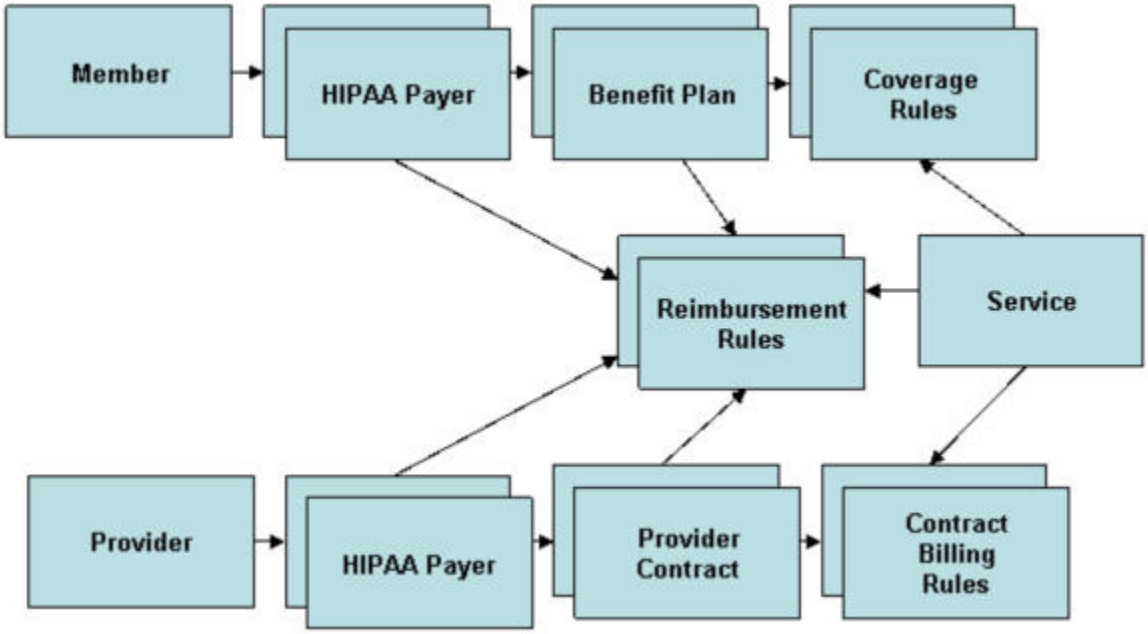
The process flow diagrams below provide a visual representation of the maintenance of pricing information. The pricing data will be maintained through the online User Interface (UI).



2.4 Data Model

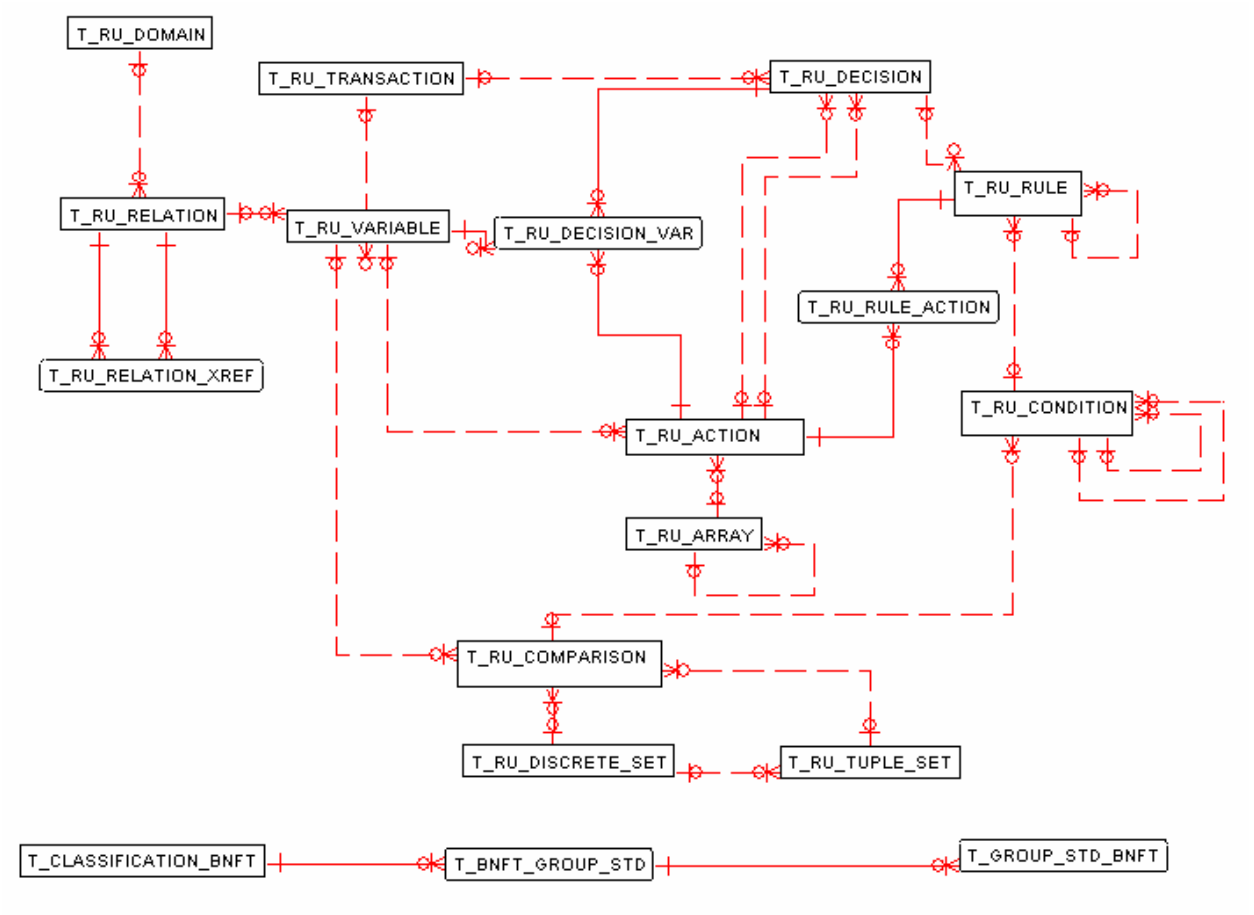
The following data model gives a view of the primary entities within the Benefits Administration Subsystem.

2.4.1 Benefit Administration Logical Data Model



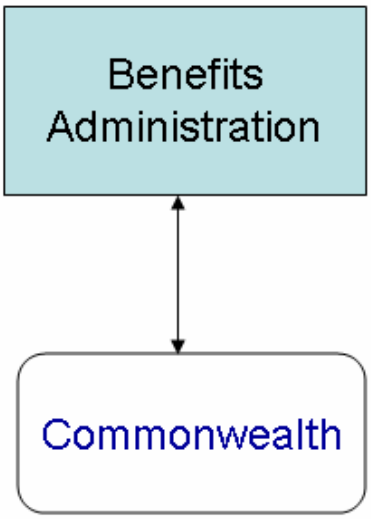
- Benefit Plan: A classification of benefits a member can receive.
- Contract Billing Rules: Billable rules for a provider within a contract. For example, a provider can only bill certain claim types;
- Reimbursement Rules: Rules for which a provider can be reimbursed. For example, reimburse the provider for a specific Benefit Plan and Provider Contract;
- Coverage Rules: Coverage parameters for services within a Benefit Plan. For example, a service may only be covered for a specific age range;
- HIPAA Payer: Entity responsible for the administration of a Benefit Plan and reimbursement of providers;
- Provider Contract: A classification of services a provider can bill; and,
- Service: Procedure, diagnosis, National Drug Code (NDC), Revenue codes, and Diagnosis Related Group (DRG) codes.

2.4.2 Benefit Administration Physical Data Model



2.4.3 External System Interfaces

The following context diagram gives a view of the entities with which this subsystem interfaces.



2.5 Job Scripts

The Benefits Administration functional area has no associated job scripts.

2.6 Programs

The Benefits Administration functional area has no associated programs.

2.7 Internal Web Pages

The Benefits Administration functional area does not directly maintain Pages or Panels. See the Reference DSD for pages/panels that support this subsystem.

2.8 Reports

Benefits Administration reporting needs are met by the Reference and the Decision Support subsystem areas.

2.9 Letters

The Benefits Administration functional area does not produce or send any letters.

2.10 Internal and External Interfaces

The Benefits Administration functional area has no Inputs/Outputs, or External file layouts.

Note: Any Benefits-related data interfaces required by KMAA are handled by the Reference Subsystem area.

2.11 Requirement Matrix and Cross Reference

This section provides a crosswalk of each functional requirement included in the KY MMIS Requirements Checklist shown in the RFP Appendix to MMIS design components. This is accomplished by listing each requirement with its current status, associated change orders and mapped system objects.

Requirement	Status	Mapped Objects	Change Orders
30.030.002	No mapping required		
30.030.002.001	No mapping required		
30.030.002.001.1	No mapping required		
30.030.002.001.10	No mapping required		
30.030.002.001.11	No mapping required		
30.030.002.001.12	No mapping required		
30.030.002.001.13	No mapping required		
30.030.002.001.14	No mapping required		
30.030.002.001.15	No mapping required		
30.030.002.001.16	No mapping required		
30.030.002.001.17	No mapping required		
30.030.002.001.18	No mapping required		
30.030.002.001.2	No mapping required		
30.030.002.001.3	No mapping required		
30.030.002.001.4	No mapping required		

Requirement	Status	Mapped Objects	Change Orders
30.030.002.001.5	No mapping required		
30.030.002.001.6	No mapping required		
30.030.002.001.7	No mapping required		
30.030.002.001.8	No mapping required		
30.030.002.001.9	No mapping required		
30.030.002.002	No mapping required		
30.030.002.002.1	No mapping required		
30.030.002.002.2	No mapping required		
30.030.002.002.3	No mapping required		
30.030.002.002.4	No mapping required		
30.030.002.002.5	No mapping required		
30.030.002.002.6	No mapping required		
30.030.002.002.7	No mapping required		
30.050.001B	Change Order Identified		
30.050.002	Informational		
30.050.002.001	Informational		
30.050.002.001.1	Commonwealth		
30.050.002.001.10	Commonwealth		
30.050.002.001.11	Commonwealth		

Requirement	Status	Mapped Objects	Change Orders
30.050.002.001.12	Commonwealth		
30.050.002.001.13	Commonwealth		
30.050.002.001.14	Commonwealth		
30.050.002.001.15	Commonwealth		
30.050.002.001.16	Commonwealth		
30.050.002.001.17	Commonwealth		
30.050.002.001.18	Commonwealth		
30.050.002.001.2	Commonwealth		
30.050.002.001.3	Commonwealth		
30.050.002.001.4	Commonwealth		
30.050.002.001.5	Commonwealth		
30.050.002.001.6	Commonwealth		
30.050.002.001.7	Commonwealth		
30.050.002.001.8	Commonwealth		
30.050.002.001.9	Commonwealth		
30.050.002.002	Informational		
30.050.002.002.1	RV Sign-Off Task		212 - Create KY Operating Procedures
30.050.002.002.2	RV Sign-Off Task		212 - Create KY Operating Procedures
30.050.002.002.3	RV Sign-Off Task		212 - Create KY Operating Procedures
30.050.002.002.4	RV Sign-Off Task		212 - Create KY Operating Procedures
30.050.002.002.5	Transferred		
30.050.002.002.6	RV Sign-Off Task		212 - Create KY Operating Procedures
30.090.002	Informational		
30.090.002.001	Informational		
30.090.002.001.1	RV Sign-Off Task	Error Disposition Information	554 - Copy Benefit Plan

Requirement	Status	Mapped Objects	Change Orders
		Page Fin.RelatedData.ascx - Other Prov.ProviderRelatedDataCodes.ascx Reference ICD-9-CM Procedure Information Reference Revenue Information Page Reference.DiagInfoPg Reference.DrugInfoPg Reference.HCPCSProcedure Information Reference.Related Data Other Provider Contract Rules-All Services Ref.HealthPgmHierarchyPanel.ascx Ref.Reimbrursement Agreement	functionality 212 - Create KY Operating Procedures
30.090.002.002	Informational		
30.090.002.002.1	EDS Met Task	T_CLM_COVERED_ROLE T_CLM_PAYABLE_ROLE T_COVERED_BENEFIT T_COV_BNFT_BNFT T_COV_BNFT_BP T_COV_BNFT_CONDITION T_COV_BNFT_OCCURR T_COV_BNFT_TYPE_SPEC T_COV_CND_AXIS_RANGE T_COV_MOD_AXIS_RANGE T_COV_OCC_AXIS_RANGE T_PAYABLE_BENEFIT T_PAYER_CONTRACT_BNFT T_PAYER_PLAN_BENEFIT T_PAY_BNFT_BNFT T_PAY_BNFT_CONDITION T_PAY_BNFT_CONTRACT T_PAY_BNFT_OCCURR T_PAY_BNFT_PLAN T_PAY_BNFT_TOB T_PAY_BNFT_TYPE_SPEC T_PAY_CND_AXIS_RANGE T_PAY_MOD_AXIS_RANGE	212 - Create KY Operating Procedures 626 - Implement Core BP Enhancements

Requirement	Status	Mapped Objects	Change Orders
		T_PAY_OCC_AXIS_RANGE Error Disposition Information Page Fin.RelatedData.ascx - Other Prov.ProviderInformationProviderMaintenanceServiceLocation.ascx Prov.ProviderRelatedDataCodes.ascx Member.Memberinformation Reference ICD-9-CM Procedure Information Reference Related Data Codes Reference Revenue Information Page Reference.DiagInfoPg Reference.DrugInfoPg Reference.HCPCSProcedure Information Benefit Plan Coverage Rules-All Services BenefitCoverageEditPanel.ascx Fin.PayerPanel.ascx Prov.EnrollmentProgramCode.ascx Provider Contract Rules-All Services Ref.AsngHierarchyPanel.ascx Ref.AuditCritBasePanel.ascx Ref.CoveredBenefitPanel.ascx Ref.DispositionCriteriaPanel.ascx Ref.FlatFeePanel.ascx Ref.HealthPgmHierarchyPanel.ascx Ref.HealthProgramPanel.ascx Ref.OIProgramPanel.ascx Ref.PayerHierarchyPanel.ascx Ref.ProviderCoveredProgramPanel.ascx Ref.Reimbrursement Agreement Ref.Reimbursement	

Requirement	Status	Mapped Objects	Change Orders
		Agreement Ref.ReimbursementAgreementEditPanel.ascx	
30.090.002.002.10	RV Sign-Off Task	Error Disposition Information Page Fin.RelatedData.ascx - Other Prov.ProviderInformationProviderMaintenanceServiceLocation.ascx Prov.ProviderRelatedDataCodes.ascx Member.Memberinformation Reference ICD-9-CM Procedure Information Reference Related Data Codes Reference Revenue Information Page Reference.DiagInfoPg Reference.DrugInfoPg Reference.HCPCSProcedure Information Benefit Plan Coverage Rules-All Services Fin.PayerPanel.ascx Prov.EnrollmentProgramCode.ascx Provider Contract Rules-All Services Ref.AuditCritBasePanel.ascx Ref.DispositionCriteriaPanel.ascx Ref.HealthPgmHierarchyPanel.ascx Ref.HealthProgramPanel.ascx Ref.PayerHierarchyPanel.ascx Ref.Reimbrursement Agreement	156 - Benefit Administration Criteria
30.090.002.002.11	RV Sign Off	Error Disposition - Audit Criteria Page Error Disposition - Audit Restriction Error Disposition Information Page	

Requirement	Status	Mapped Objects	Change Orders
30.090.002.002.12	EDS Met CO	Error Disposition - Audit Criteria Page Error Disposition - Audit Restriction Error Disposition Information Page	558 - Add Limit Time Unit Options
30.090.002.002.13	RV Sign Off	Error Disposition - Audit Criteria Page Error Disposition - Audit Restriction Error Disposition Information Page	
30.090.002.002.14	RV Sign Off	Error Disposition Information Page Reference Revenue Information Page Reference.DiagInfoPg Reference.DrugInfoPg Reference.HCPCSProcedure Information Reference.Related Data Other	
30.090.002.002.15	RV Sign-Off Task	Reference.Related Data Other Ref.HealthPgmHierarchyPanel.ascx Ref.HealthProgramPanel.ascx	156 - Benefit Administration Criteria
30.090.002.002.16	RV Sign Off	Error Disposition - Audit Criteria Page Error Disposition - Audit Restriction Error Disposition Information Page	156 - Benefit Administration Criteria
30.090.002.002.2	RV Sign-Off Task	Benefit Plan COB Benefit Plan Coverage Rules-All Services Provider Contract Rules-All Services Ref.HealthPgmHierarchyPanel.ascx Ref.ProviderCoveredProgramPanel.ascx clmsProgram	156 - Benefit Administration Criteria

Requirement	Status	Mapped Objects	Change Orders
30.090.002.002.3	EDS Met		
30.090.002.002.5	RV Sign-Off Task		156 - Benefit Administration Criteria
30.090.002.002.6	EDS Met Task	Error Disposition - Audit Criteria Page Error Disposition - Audit Restriction Error Disposition Information Page Reference Revenue Information Page Reference.DiagInfoPg Reference.HCPCSProcedure Information Reference.Related Data Other Benefit Plan Coverage Rules-All Services Ref.AuditCritBasePanel.ascx Ref.DispositionCriteriaEobDataPanel.ascx Ref.DispositionCriteriaPanel.ascx Ref.HealthProgram Ref.HealthProgramPanel.ascx	
30.090.002.002.7	RV Sign Off	Error Disposition Information Page Reference Revenue Information Page Reference.DiagInfoPg Reference.HCPCSProcedure Information Reference.Related Data Other	
30.090.002.002.8	Transferred		
30.090.002.002.9	RV Sign-Off Task		156 - Benefit Administration Criteria
30.090.002.003	Informational		
30.090.002.003.1	EDS Met		
30.090.002.003.2	RV Sign-Off CO		

Requirement	Status	Mapped Objects	Change Orders
30.090.002.003.3	EDS Met	REF-0200-M REF-0202-M REF-0204-M REF-0206-M REF-0207-M REF-0208-M REF-0209-M	
30.090.002.004	Informational		
30.090.002.004.1	EDS Met Task	Error Disposition Information Page Reference.Related Data Other Benefit Plan Coverage Rules-All Services Provider Contract Rules-All Services Ref.HealthPgmHierarchyPanel.ascx Ref.Reimbrursement Agreement	212 - Create KY Operating Procedures
30.100.002	No mapping required		
30.100.002.001	No mapping required		
30.110.002	Informational		
30.110.002.001	RV Sign-Off Task		
30.110.002.002	RV Sign-Off Task		

2.12 Change Orders

Note: Change Orders with a status of “Cancelled” at the time this document was prepared are not included in this document.

2.12.1 Benefit Administration Criteria - 156

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
156	Task		Benefit Administration			1

2.12.1.1 Desired Solution

Define Benefit Administration rules necessary to support Kentucky Medicaid policy. The criteria that needs to be defined includes:

- Member enrollment programs;
- Provider contracts;
- Payer information;
- Covered Services;
- Service limitations; and,
- Other rules needed for claims processing

Note: Some of the rules/criteria needed to support the Benefits Package function will be defined during the Claims RV/JAD sessions when discussing claims editing, auditing, and pricing.

2.12.1.2 Business Impact

None.

2.12.1.3 Technical Specifications

Policy Documentation will be maintained on the Kentucky Project Workbook (PWB) under the following link:

https://ddipwb.kymmis.com/KYXIXDDI/Subsystem/Benefit%20Administration/default.asp

2.12.1.4 Clarifications

None.

2.12.1.5 Associated Requirements

Requirement ID	Type
30.090.002.002.10	RFP Requirement
30.090.002.002.15	RFP Requirement
30.090.002.002.16	RFP Requirement
30.090.002.002.2	RFP Requirement

Requirement ID	Type
30.090.002.002.5	RFP Requirement
30.090.002.002.9	RFP Requirement
30.090.007.002.16A	RFP Split Requirement

2.12.1.6 Associated System Objects

Technical Name	Object Type	Title
T_COVERAGE_XREF	Database Table	

2.12.1.7 Change Order Status

Status	Date
Define/Analyze in Progress	06/24/2005
Completed	09/01/2007

2.12.2 Create KY Operating Procedures – 212

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
212	Task		System Wide			1

2.12.2.1 Desired Solution

Create Kentucky interChange Operating Procedures.

RFP Text:

Section 30.050 - New KY MMIS Operations, Maintenance, and Modifications Phase

The Contractor shall be responsible for operating, maintaining and modifying the New KY MMIS according to the CMS State Medicaid Manual; all Federal mandates; and all Commonwealth requirements, statues, and regulations.

The Contractor's Scope of Work shall include performance of the operational functions. Each function is presented in terms of:

- Commonwealth or its designee responsibilities, and
- Contractor responsibilities

In addition to these responsibilities, the performance expectations for operations are listed in Section 30 within this RFP. The contractor shall perform all functions necessary to operate a flexible, successful, complete, and CMS certifiable KY MMIS.

2.12.2.2 Business Impact

This task will be used to identify and track the steps necessary to create and maintain the operating procedures necessary to maintain the MMIS.

2.12.2.3 Technical Specifications

See the section in the work plan called KY interChange Operating Procedures. All work for setting up and documenting the KY operational procedures will be handled under this work plan section.

2.12.2.4 Clarifications

None.

2.12.2.5 Associated Requirements

Requirement ID	Type
30.050.002.1	RFP Requirement
30.050.002.002.2	RFP Requirement
30.050.002.002.3	RFP Requirement
30.050.002.002.4	RFP Requirement
30.050.002.002.6	RFP Requirement

Requirement ID	Type
30.050.004.002.1	RFP Requirement
30.050.004.002.10	RFP Requirement
30.050.004.002.19	TPL Resource HIPP Maintenance
30.050.004.002.20	TPL Resource HIPP Maintenance
30.050.004.002.4	Training and Manuals
30.050.004.002.5	Training and Manuals
30.050.005.002.1	RFP Requirement
30.050.006.002	RFP Requirement
30.050.006.002.1	RFP Requirement
30.050.006.002.2	RFP Requirement
30.050.006.002.4	RFP Requirement
30.050.006.002.5	RFP Requirement
30.050.006.002.6	RFP Requirement
30.050.006.002.7	RFP Requirement
30.050.006.002.8	RFP Requirement
30.090.002.001.1	RFP Requirement
30.090.002.001.1A	RFP Split Requirement
30.090.002.002.1	RFP Requirement
30.090.002.004.1	RFP Requirement
30.090.004.001.7	TPL Resource HIPP Maintenance
30.090.006.001.1	RFP Requirement
30.090.006.002.2	RFP Requirement

Requirement ID	Type
30.090.017.001.2	RFP Requirement
30.110.006.002	RFP Requirement
30.110.006.005	RFP Requirement

2.12.2.6 Associated System Objects

Technical Name	Object Type	Title
None		

2.12.2.7 Change Order Status

Status	Date
Issue Identified	07/07/2005
Change Order Written	08/09/2005

2.12.3 Copy Benefit Plan Functionality - 554

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
554	Task		Reference			1

2.12.3.1 Desired Solution
Provide system functionality to copy an existing benefit plan rule(s) to a new benefit plan and then allow modifications to the new benefit plan.

2.12.3.2 Business Impact
Allow authorized user to copy a Benefit Plan and make modifications in order to simplify the process of added a new benefit plan.

2.12.3.3 Technical Specifications
Core interChange is developing a process to copy a benefit plan and/or benefit plan rule.

Implement Core interChange functionality to the new Kentucky MMIS.

2.12.3.4 Clarifications
None.

2.12.3.5 Associated Requirements

Requirement ID	Type
30.090.002.001.1	RFP Requirement

2.12.3.6 Associated System Objects

Technical Name	Object Type	Title
None		

2.12.3.7 Change Order Status

Status	Date
Issue Identified	08/21/2005
Change Order Written	01/05/2006
Deferred	12/17/2006

2.12.4 Add Limit Time Unit Options - 558

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
558	Change Order		Reference			1

2.12.4.1 Desired Solution

Add additional time unit options for limit audits.

1. Add Calendar Week as an option on Limit Audits.
2. Add Calendar Year as an option on Limit Audits.

Note:
Additional time unit options may be identified during Claim Auditing workgroups.

2.12.4.2 Technical Specifications

Add the following Time Unit options to Limit Audit Parameter panel field Time Unit.

1. Calendar Week; and,
2. Calendar Year.

Note:
Additional time unit options may be identified during Claim Auditing workgroups. Core is adding a code table to hold the Audit time limits parameters. See Core CO 8567 for the data model changes needed to support this change.

2.12.4.3 Business Impact

Add Time Unit parameters necessary to support Commonwealth auditing policy to Limit Audit Parameter panel.

2.12.4.4 Clarifications

01/26/2006 - The matching CO in Core is 9686.

12/02/2005 - Work on this CO needs to follow implementation of Core CO 8567.

08/21/2005 - Additional information/requirements will be identified during claims edit/audit work groups and during policy review.

2.12.4.5 Associated Requirements

Requirement ID	Type
30.090.006.002.10	RFP Requirement
30.090.006.002.10	RFP Requirement

2.12.4.6 Associated System Objects

Technical Name	Object Type	Title
None		

2.12.4.7 Change Order Status

Status	Date
Issue Identified	08/21/2005
Technical Design In Progress (obsolete)	12/07/2005
SE Assigned	12/07/2005
Ready for Tech Walkthrough	12/08/2005
Construction in Progress	01/26/2006
Ready for Construction Wthru	02/13/2006
Ready for Model Office	02/15/2006
Model Office Implemented	04/06/2006
UAT Implemented	11/01/2006
Prod Implemented	06/14/2007

2.12.5 Implement Core BP Enhancements - 626

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
626	Task		Reference			1

2.12.5.1 Desired Solution

Implement and document Enhanced Benefit Administration User Interface (UI).

Core interChange is enhancing the Benefit Administration functionality to allow more user driven criteria.

An overview of the Benefit Plan functionality being developed can be found under the Benefit Administration page on the Project Workbook. The overview document is titled Benefit Administration Overview.

2.12.5.2 Business Impact

Enhance ability to user configure Benefit Plans.

2.12.5.3 Technical Specifications

Implement enhanced Benefit Administration user interface pages and panels.

Update all related user documentation.

This Change Order includes the following interChange functionality:

Benefit Plan Service Classification Benefit Plan Coverage Rules Reimbursement Agreement Rules Provider Contract Rules Member Plan maintenance functions Financial Payer maintenance functions

2.12.5.4 Clarifications

At the time of the Reference RV/JAD sessions, full Benefit Plan functionality was not available for demonstration. This CO has been written to track the effort needed to bring into Kentucky all Benefit Plan functionality available in the Core interChange system. This includes the Benefit Plan Coverage Rules, Reimbursement Rules and Provider Contract Rules being developed as part of interChange Core functionality.

Full Core interChange Benefit Plan related functionality will be made available to the new Kentucky MMIS.

2.12.5.5 Associated Requirements

Requirement ID	Type
30.090.002.002.1	RFP Requirement
30.090.006.002.3	RFP Requirement
30.090.006.002.4	RFP Requirement

2.12.5.6 Associated System Objects

Technical Name	Object Type	Title
Provider Contract Rules-All Services	Panel	Contract Billing Rules
Ref.AsngHierarchyPanel.ascx	Panel	Benefit Administration-Assignment Plan Hierarchy
Ref.CoveredBenefitPanel.ascx	Panel	Benefit Plan Coverage Rules-All Services
Ref.HealthPgmHierarchyPanel.ascx	Panel	Benefit Administration-Benefit Plan Hierarchy
Ref.HealthProgramPanel.ascx	Panel	Benefit Administration-Member Plan
Ref.PayerHierarchyPanel.ascx	Panel	Benefit Administration-Financial Payer Hierarchy
Ref.Reimbrursement Agreement	Panel	Contract Reimbursement Rules

2.12.5.7 Change Order Status

Status	Date
Issue Identified	08/26/2005
Change Order Written	01/15/2006
Ready for Model Office	08/07/2006
UAT Implemented	04/23/2007

2.12.6 ADD index on T_RU_COMPARISON - 7205

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
7205	Change Order		Benefit Administration			

2.12.6.1 Desired Solution
During troubleshooting of a claim processing problem, it was found that a secondary index on T_RU_COMPARISON needed to be added.

2.12.6.2 Business Impact
Required for claim processing performance

2.12.6.3 Technical Specifications
Create an index on T_RU_COMPARISON on cde_value_comp, cde_variable.

2.12.6.4 Clarifications
Email 08/20/07 from Troy Schexnayder: "CO 7205 - This was completed by Reference CO 7105."

2.12.6.5 Associated Requirements

Requirement ID	Type
No associated Requirements found.	

2.12.6.6 Associated System Objects

Technical Name	Object Type	Title
None		

2.12.6.7 Change Order Status

Status	Date
Change Order Written	04/06/2007
Completed	09/01/2007

2.12.7 I3391 - 8021

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
8021	Change Order		Benefit Administration			2

2.12.7.1 Desired Solution
To allow repair codes to bypass the prior authorization requirement if less than $300

2.12.7.2 Business Impact
None.

2.12.7.3 Technical Specifications
None.

2.12.7.4 Clarifications
Codes E1340, L4205, L4210, L7500, L7510, L7520 an L8049 should bypass the PA requirement if the amount billed is less than $300.00. this is a change effective with the regulation 907 KAR 1:479 which was adopted on 1/5/07.

Audits 641 and 327 would be affected.

Email sent to Jimmie Estill on 6/27 to schedule walkthrough. Per John Hoffman, during the CO meeting held today, Jimmie is on vacation this week. Walkthrough can be scheduled for next week when Jimmie is back in the office.

Waiting to hear from Jimmie Estill regarding a time and date to go over testing documentation.

Completed walk through; approval received to move into production.

Walkthrough on 7/5/2007 with Jimmie Estill and Patricia Biggs. Approved to move into production; moved into production on 7/10/2007. Edit pages need updated with new criteria; new edit manual being reviewed by DMS (approval not received to date for new edit manual).

Edit pages need updated with new criteria; new edit manual being reviewed by DMS (approval not received to date for new edit manual).

Edit pages need updated with new criteria; new edit manual being reviewed by DMS (approval not received to date for new edit manual).

No change. Waiting DMS approval of new edit manual. CO requires edit page update.

No change. Waiting DMS approval of new edit manual. CO requires edit page update.

No change. Waiting DMS approval of new edit manual. CO requires edit page update.

No change. Waiting DMS approval of new edit manual. CO requires edit page update.

John Hoffmann has approved updating a draft of the appropriate edit page to send with sign off.

Sign-off memo submitted to the Commonwealth 09/05/07.

Waiting for Memo from DMS regarding agreement that CO 8021 is complete.

Waiting for Memo from DMS regarding agreement that CO 8021 is complete.

Sign off letter received from DMS.

2.12.7.5 Associated Requirements

Requirement ID	Type
No associated Requirements found.	

2.12.7.6 Associated System Objects

Technical Name	Object Type	Title
None		

2.12.7.7 Change Order Status

Status	Date
Change Order Written	05/17/2007
Assigned BA	05/17/2007
Assigned BA	06/19/2007
MO Testing Complete	06/29/2007
Prod Implemented	07/10/2007
Sign-Off Requested	09/05/2007
Completed	10/10/2007

2.12.8 I3420 - 8486

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
8486	Change Order		Benefit Administration			1

2.12.8.1 Desired Solution
To add 3 HCPCS codes

2.12.8.2 Business Impact
None.

2.12.8.3 Technical Specifications
None.

2.12.8.4 Clarifications
Effective July 1, 2007 CMS has added the following three HCPCS codes that are to be added to the system. Provider type 90

- K0553 Combination oral/nasal mask used with continuous positive airway pressure device, each;
- K0554 oral cushion for combination oral/nasal mask, replacement only; and,
- K0555 Nasal pillows for combination Oral/Nasal mask, replacement only.

Codes above require prior authorization. Max fees: K0553 $250.00; K0554 $50.00; K0555 $50.00.

Model Office testing completed. Walkthrough scheduled with Karen Sayles and Patricia Biggs this afternoon.

Rules have not been moved into Production to date. Waiting approval on copay CO before synching EBP and production. Sign-off letter will be sent once production has been updated.

Rules moved into production on 7/10/2007. Sign off being requested.

Sign-off request forwarded to the Commonwealth 07/12/07

Memo received from DMS today - CO is considered complete.

2.12.8.5 Associated Requirements

Requirement ID	Type
No associated Requirements found.	

2.12.8.6 Associated System Objects

Technical Name	Object Type	Title
None		

2.12.8.7 Change Order Status

Status	Date
Change Order Written	06/22/2007
Assigned BA	06/22/2007
MO Testing Complete	06/29/2007
Prod Implemented	07/10/2007
Sign-Off Requested	07/12/2007
Completed	07/18/2007

2.12.9 I3423 - 8487

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
8487	Change Order		Benefit Administration			1

2.12.9.1 Desired Solution
Allow payment of claims for PET scans due to cancer diagnosis between 9/15/06 and 1/03/07.

2.12.9.2 Business Impact
None.

2.12.9.3 Technical Specifications
None.

2.12.9.4 Clarifications
In conjunction with CO #1571, CPT codes in the 70000 series should be exempted from a prior authorization requirement when cross-referenced to a diagnosis indicating malignancy or transplant effective for dates of service from 9/15/06 through 1/03/07.

Providers were notified by general mailing on September 12, 2006 of a prior authorization exemption for all radiology services provided to a member with a malignancy or transplant diagnosis.

Meeting scheduled on July 6 with Jan Thornton and Charles Douglas to go over CO 8487.

Clarification received on memo and it's related CO 1571. Model office updates and testing started.

Extension requested on 7/11/07 and approved for date 7/20.

Extension to 8/10/2007 denied. Issue encountered in MO testing.

No change in status. Encountered problem in MO testing.

Encountered problem in MO testing; issue needs to be reviewed by a SE.

Will contact Mark Veness and go over issue encountered in MO.

Issue identified in MO was corrected by Phil Stokes for Physician contract. MO testing in progress.

Walkthrough today at DMS to review test claims. During review of test claims it was determined that all codes noted on Legacy Edit 538 and/or on Medical Review need to be updated with effective date of service 09/15/2006. Based on COs 1571 and 1632 some of the rules have effective date of 01/04/2007. Also, these two COs addressed updating rules only on the PHYS (PT 64/65) and NPRCT (PT 78). It was determined today, that all contracts need (should) be consistent. So rules on all applicable contracts will be updated once addendum to CO 8487 is received.

Awaiting addendum. All radiology codes on legacy edit 538 need to be back dated to 09/15/2006 with all appropriate diagnosis criteria; all applicable provider contracts.

Test cases for CO 8487 addendum sent to DMS this morning (9/13) for review. If test cases are approved rules will be updated on provider contracts for PTs 01, 31, 35, 36, 39, 74, 80, and 86.

Email sent requesting an extension. The addendum will require updating approximately 870 rules.

Waiting on addendum testing approval. Requested extension date will be based on date approval is received on addendum testing. Rules have been updated on provider contracts PHYS and NPRCT, PET Scan codes only and moved into production.

Waiting on approval of addendum testing.

Waiting on approval of addendum testing. Once approval is received, rules will be updated.

Waiting on approval of addendum testing. Once approval is received, rules will be updated.

Email received by Jan Thornton on 10/22/2007 stating that testing on the addendum had been approved.

2.12.9.5 Associated Requirements

Requirement ID	Type
No associated Requirements found.	

2.12.9.6 Associated System Objects

Technical Name	Object Type	Title
None		

2.12.9.7 Change Order Status

Status	Date
Change Order Written	06/22/2007
Assigned BA	06/22/2007
MO Testing in Progress	07/10/2007
Extension Requested	07/20/2007
Extension Approved	07/20/2007
MO Testing in Progress	07/26/2007
Model Office Implemented	08/27/2007
UAT Implemented	08/27/2007
Extension Requested	09/13/2007
Prod Implemented	09/20/2007

2.12.10 I3424 - 8488

Identifier	Type	Level	Subsystem	Computed	Estimated	Priority
8488	Change Order		Benefit Administration			1

2.12.10.1 Desired Solution
Add new CPT and HCPCS codes to Reference Subsystem in ICE.

2.12.10.2 Business Impact
None.

2.12.10.3 Technical Specifications
None.

2.12.10.4 Clarifications
Please add the following HCPCS codes to the Procedure file for Provider types "64" & "65" effective 5/01/07:

- Q4083 Hyalgan or Supartz $130.50 (outpt. only);
- Q4084 Synivsc $211.80 (outpt. Only);
- Q4085 Euflexxa $131.21 (outpt. only); and,
- Q4086 Orthovisc $225.00 (outpt. only).

Delete HCPCS code J7317 from the procedure file for provider types "64" & "65", effective 5/31/07.

Add provider type "80" to the provider panel for CPT code 17111, effective 1/01/07.

In order to pay supplies and services according to existing policies, Medicaid systems must comply with national medical coding changes.

Email to Kelli with question - CO states to delete J7317 effective 5/31/07 - J7317 has a current end date of 12/31/06 -JP

Response received from Kelli on 7/6/07, documentation added to PWB - JP

Walkthrough requested. - JP

Walkthrough requested 8/10/07 - JP

Waiting on CW to walkthrough test results. JP

Testing approved - DMS - Gayle Nickles, Candy Johnson, Kelli O'Brien 8/17/07.

Sign-off memo submitted to the Commonwealth 08/21/07.

Sign-off memo submitted to the Commonwealth 08/21/07 - No update.

Received sign-off memo 9/6/07

2.12.10.5 Associated Requirements

Requirement ID	Type
No associated Requirements found.	

2.12.10.6 Associated System Objects

Technical Name	Object Type	Title
None		

2.12.10.7 Change Order Status

Status	Date
Change Order Written	06/22/2007
Assigned BA	06/22/2007
MO Testing in Progress	06/29/2007
MO Testing Complete	07/10/2007
Testing Approved - DMS	08/17/2007
Sign-Off Requested	08/21/2007
Completed	09/06/2007

2.13 Data Element Dictionary

Because of the length of some column names and the use of underscore characters, some names in the tables below may wrap to the next line.

Please Note: The primary key of an entity (table) is defined by the data modeler that developed the subsystem data model. It is sometimes the business key to the table, but often, the primary key is an internally defined attribute such as a System Assigned Key. The primary key is typically the attribute or set of attributes that defines a unique row on a table.

2.13.1 T_ASNG_HIERARCHY

An assignment hierarchy is an ordered set of assignment plans for which a member may be assigned concurrently. For example, a beneficiary may be assigned to a PCP and to a Health Community Based Services agency at overlapping periods of time. A hierarchy tread is used to determine which assignment to select at the time of processing

Column Name	Description	Type	Length	Precision	Primary Key
SAK_THREAD_ASSIGNMENT	System assigned key used to uniquely identify a hierarchy thread. The hierarchy thread is used to identify the order of processing of Financial Payers, Benefit Plans, or Assignment Plans for beneficiaries who are enrolled in multiple entities of any of these types..	NUMBER	9	0	Y
SAK_FIN_PAYER	The system assigned key that identifies a unique payer within interChange.	NUMBER	9	0	N
DTE_EFFECTIVE	Date the hierarchy thread becomes valid for use in claims processing.	NUMBER	8	0	N
DTE_END	The last date the thread is valid for use in claims processing. The thread may still be used when the date of service on the claim is within the effective dates of the thread. To stop the use of a particular thread altogether the DTE_INACTIVE date must be used.	NUMBER	8	0	N

Column Name	Description	Type	Length	Precision	Primary Key
NUM_PGM_1_32	Bit flags for T_PUB_HLTH_PGM.NUM_HIER_ID_ASGN 1 to 32. Used as part of the primary key and editing to determine assignment plans order of processing within a Financial Payer. If NUM_HIER_ID_ASGN column of T_PUB_HLTH_PGM IS BETWEEN 1 AND 32 for a benefit plan on the thread, then corresponding bit in this column will be set to 1 otherwise the bit will be set to 0.	NUMBER	10	0	N
NUM_PGM_33_64	Bit flags for T_PUB_HLTH_PGM.NUM_HIER_ID_ASGN 33 to 64. Used as part of the primary key and editing to determine assignment plans order of processing within a Financial Payer. If NUM_HIER_ID_ASGN column of T_PUB_HLTH_PGM IS BETWEEN 33 AND 64 for a benefit plan on the thread, then corresponding bit in this column will be set to 1 otherwise the bit will be set to 0.	NUMBER	10	0	N
NUM_PGM_65_96	Bit flags for T_PUB_HLTH_PGM.NUM_HIER_ID_ASGN 65 to 96. Used as part of the primary key and editing to determine assignment plans order of processing within a Financial Payer. If NUM_HIER_ID_ASGN column of T_PUB_HLTH_PGM IS BETWEEN 65 AND 96 for a benefit plan on the thread, then corresponding bit in this column will be set to 1 otherwise the bit will be set to 0.	NUMBER	10	0	N
DTE_INACTIVE	This is the date when the usage of this thread was stopped. From this date onwards the thread will not be used regardless of the date of service billed on a claim.	NUMBER	8	0	N

2.13.2 T_ASNG_NODE

A given thread may have as many participants as required by business policy. This table is used to identify the participating entities as well as the processing sequence within the thread. The entity types in a thread are assignment plans.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_THREAD_ASSIGNMENT	System assigned key used to uniquely identify a hierarchy thread. The hierarchy thread is used to identify the order of processing of Financial Payers, Benefit Plans, or Assignment Plans for beneficiaries who are enrolled in multiple entities of any of these types..	NUMBER	9	0	Y
SAK_ASGN_PLAN	System assigned internal key for a medical assistance program.	NUMBER	9	0	N
NUM_SEQUENCE_THREAD	This column determines the processing order for the entities participating in the thread. For example, if the thread is of type 'PAYR', then this sequence number determines which Financial Payer will be processed first, second, and so on.	NUMBER	4	0	N

2.13.3 T_BENEFIT_PLAN_TYPE

A benefit plan type is associated with a group of benefit plan codes. This is used to identify benefit plans for use in certain processing methodologies.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_BENEFIT_PLAN_TYPE	System assigned key for a unique Benefit Plan group that represents a single or collection of Benefit Plan codes.	NUMBER	9	0	Y
DSC_50	Describes the benefit plan type.	CHAR	50	0	N
DSC_DEFINITION	Definition of where and/ or how this benefit plan group is used.	VARCHAR2	4000	0	N

2.13.4 T_BENEFIT_TYPE

The service type is used is used to identify the coding scheme for a service. Valid schemes include procedure (HCPCS & CPT-4), ICD-9-CM procedures, ICD-9-CM diagnosis, Revenue Code and Drug codes (NDCs).

Column Name	Description	Type	Length	Precision	Primary Key
CDE_BENEFIT_TYPE	The benefit type code is used to identify the coding scheme for a service. Valid schemes include procedure (HCPCS & CPT-4), ICD-9-CM procedures, ICD-9-CM diagnosis, Revenue Code and Drug codes (NDCs).	CHAR	3	0	Y
DSC_BENEFIT_TYPE	Describes the type of benefit.	VARCHAR2	18	0	N

2.13.5 T_BNFT_ADJ_FACTOR

This table is used to store most of the information required for benefit adjustments to current pricing methodology in some states. Information on this table may or may not be used for claim auditing. This table will or will not be populated with miscellaneous information based on states needs.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_BENEFIT_ADJ_FACTOR	System assigned key to uniquely identify a valid benefit adjustment type.	NUMBER	9	0	Y
SAK_BENEFIT_ADJ_FACTOR_TYPE	System assigned key to uniquely identify a valid benefit adjustment type.	NUMBER	9	0	Y
DTE_EFFECTIVE	The date of service on which the benefit adjustment factor stops being valid.	NUMBER	8	0	N
DTE_END	The date of service on which the benefit adjustment factor stops being valid.	NUMBER	8	0	N
DTE_INACTIVE	This is the date when this segment becomes inactive and will not be used in for claims received on and after it.	NUMBER	8	0	N

Column Name	Description	Type	Length	Precision	Primary Key
PCT_RATE_ADJ	The percentage that will be used to adjust an allowed amount already calculated using the pricing methodologies that applies to the claim/detail.	NUMBER	5	3	N
AMT_RATE_ADJ	The amount that will be used to adjust an allowed amount already calculated using the pricing methodology that applies to the claim/detail.	NUMBER	9	2	N
CDE_ALWD_CALCULATION	This indicator is necessary to identify if the benefit adjustment factor will be applied before or after the allowed amount was finalized. Some Benefit adjustment factors can be applied before a comparison between the billed amount and the allowed amount is done. Values "B" Before the calculated allowed amount is compared to the billed amount. Value "A" after the calculated amount is finalized	CHAR	1	0	N

2.13.6 T_BNFT_ADJ_FACTOR_TYPE

This table is used to store information about the type and the description of the class of products or services to which the adjustment factor applies. Information on this table may or may not be used for claim auditing. This table will or will not be populated with miscellaneous information based on states needs.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_BENEFIT_ADJ_FACTOR_TYPE	System assigned key to uniquely identify a valid benefit adjustment type.	NUMBER	9	0	Y
CDE_BNFT_ADJ_TYPE	The code identifying the class of services/products to which the adjustment factor applies.	CHAR	11	0	N
DSC_BNFT_ADJ_TYPE	The description of the Benefit Adjustment Factor	VARCHAR2	250	0	N

2.13.7 T_BNFT_GROUP_STD

This table represents hierarchies of groups of benefits for the purpose of defining conditions at levels higher than the individual benefits. This structure will allow end users to set up benefit and assignment plans easier and faster than defining conditions at the individual benefit level. Each type of benefit such as procedure code, NDC, and the rest can be grouped into sets of benefits. These groups may then be grouped into a larger set of benefits. An example of this grouping technique is the procedure codes CPT4 manual. This manual defines a hierarchy of groups of procedures. The end user may then define conditions at any point of the hierarchy. The conditions are then inherited by the lower levels.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_BNFT_GROUP	System assigned key for a unique benefit group.	NUMBER	9	0	Y
CDE_BENEFIT_TYPE	The benefit type code is used to identify the coding scheme for a service. Valid schemes include procedure (HCPCS & CPT-4), ICD-9-CM procedures, ICD-9-CM diagnosis, Revenue Code and Drug codes (NDCs). Other allowed values may identify GROUPS of benefits such as Procedure groups and so on.	CHAR	3	0	Y
SAK_CLASS_HIERARCHY	Unique row identifier for this table.	NUMBER	9	0	Y
SAK_PARENT_GROUP	System assigned key for a unique parent group.	NUMBER	9	0	N
DSC_GROUP	This is the description of the group originating from the standard code sets. For example, the source for procedure code group descriptions will be the CPT4 and HCPCS standard manuals.	VARCHAR2	4000	0	N
NAM_GROUP	This is the description of the group that the codes are in within the CPT or HCPC group.	VARCHAR2	100	0	N

2.13.8 T_CDE_AID

Contains the valid aid categories and descriptions.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_CDE_AID	System assigned key to uniquely identify a valid aid category.	NUMBER	9	0	Y
CDE_AID_CATEGORY	Identifies the type of aid for which a member is eligible.	CHAR	2	0	N
DSC_AID_CATEGORY	Describes the type of aid for which a member is eligible.	CHAR	50	0	N
IND_DIAG_IN_EXCLUD	Not currently used.	CHAR	1	0	N
IND_SPEC_IN_EXCLUD	Not currently used.	CHAR	1	0	N
IND_EMERGENCY	Not currently used.	CHAR	1	0	N

2.13.9 T_CLAIM_TYPE

List of the valid types of claims that can be submitted and processed in the MMIS.

Column Name	Description	Type	Length	Precision	Primary Key
CDE_CLM_TYPE	Value for the type of claim that can be processed in the MMIS system.	CHAR	1	0	Y
DSC_CLM_TYP	Description of the value assigned to a specific claim type.	CHAR	50	0	N

2.13.10 T_CLASSIFICATION_BNFT

These are the group classifications for the benefits. Each classification structures the grouping into hierarchies. For example, drugs (NDC) are likely to have more than one classification each represented by a hierarchy of groups; one hierarchy could be used to specify general coverage conditions such as age and gender; a different hierarchy of groups may be more appropriate for Preferred drug conditions.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_CLASS_HIERARCHY	Unique row identifier for this table.	NUMBER	9	0	Y
CDE_CLASS_BNFT_HIER	A short hand code to identify the hierarchy.	CHAR	4	0	N
DSC_CLASS_HIERARCHY	A description of what the hierarchy is and to use it.	VARCHAR2	20	0	N

2.13.11 T_CLM_BNFT_ROLE

This is a list of all possible roles a benefit (procedure, diagnosis) might represent on a claim. For example, a diagnosis may be primary (principal on the 837 I), secondary, and so on. Other values may designate "any detail diagnosis" or "any header diagnosis", and so on.

Column Name	Description	Type	Length	Precision	Primary Key
CDE_BENEFIT_ROLE	This identifies the role of the particular benefit (procedure, diagnosis id) on the claim. Values include PRIM (primary diagnosis), ADMT (admitting diagnosis), PRIN (principal procedure), ADTL (all at detail), AHDR (all at header) and so on.	CHAR	4	0	Y
DSC_BENEFIT_ROLE	A description of the role. The description should make a reference to the field(s) on the claim that contain the value for the benefit in question. For example, on the 837 Inst the Admitting diagnosis is in the first HI segment.	VARCHAR2	100	0	N

2.13.12 T_CONDITION

Contains the valid codes used to identify conditions relating to a UB claim that may affect payer processing.

Column Name	Description	Type	Length	Precision	Primary Key
CDE_COND	Code used to identify conditions relating to a UB claim that may affect payer processing.	CHAR	2	0	Y
DSC_COND	Description of a conditions relating to a UB claim that may affect payer processing.	CHAR	40	0	N

2.13.13 T_COUNTY

This table represents the geographical/political counties in a given state. Only one address per county code is used.

Column Name	Description	Type	Length	Precision	Primary Key
CDE_COUNTY	The county code used to identify a geographical/ political area in the state.	VARCHAR2	10	0	Y
CDE_LOCALITY	This attribute indicates the locality code for a county. Valid values include 01 - Metropolitan, 02 - Urban, 03 - Rural.	CHAR	2	0	N
DSC_COUNTY	Twelve character field containing the name of a specific county.	CHAR	12	0	N
ADR_STREET_1	The first line of the address to use in correspondence with county.	CHAR	30	0	N
ADR_STREET_2	The second line of the address to use in correspondence with county.	CHAR	30	0	N
ADR_CITY	The name of the city to use in correspondence with county.	CHAR	15	0	N
ADR_STATE	This is the state code for the county office address.	CHAR	2	0	N
ADR_ZIP_CODE	The five byte zip code to use in correspondence with county.	CHAR	5	0	N
ADR_ZIP_CODE_4	The four-digit zip code that follows the five-digit zip code.	CHAR	4	0	N
CDE_URBAN_RURAL	Indicates whether or not a county is Rural (R) or Urban (U). This is used for payment of capitation under the Managed Care RBMC program.	CHAR	1	0	N

Column Name	Description	Type	Length	Precision	Primary Key
CDE_IDCARD_CNTL	This control number is assigned by West Lake, Ohio for each county office to determine which return address to print on the envelope for the member id card. Not used by OK MMIS.	CHAR	5	0	N
CDE_REGION_GEO	A user defined code that identifies a geographical region.	CHAR	4	0	N

2.13.14 T_DIAGNOSIS

A condition that requires medical attention.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_DIAG	System assigned key for the diagnosis.	NUMBER	9	0	Y
CDE_DIAG	A code for the condition requiring medical attention.	CHAR	7	0	N
DSC_25	The short nomenclature for a medical condition.	CHAR	40	0	N
DSC_LONG	The long nomenclature for a medical condition.	CHAR	250	0	N
DSC_LAY	Description of the code in layman terms.	VARCHAR2	100	0	N

2.13.15 T_DRG

Diagnosis Related Groups are used as the standard classification system for dealing with inpatient hospital data. Developed for evaluating resource use and utilization patterns. DRGs are used nationwide as the basis for the analysis and prospective payment of hospital care.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_DRG	System assigned key for a unique DRG.	NUMBER	9	0	Y
CDE_DRG	This is a three byte code field used to identify a DRG grouping. The DRG code and description are obtained from HCIA.	CHAR	4	0	N

Column Name	Description	Type	Length	Precision	Primary Key
DSC_DRG	This is a forty byte character field used to describe a DRG grouping. The DRG code and description are obtained from HCIA.	VARCHAR2	132	0	N

2.13.16 T_DRUG

The necessary data that defines a drug. This includes the drug's NDC code, manufacturer, strength, dosage form, package size, and so on. The majority of this data is supplied by our drug vendor, First Data Bank. Some drugs and their data are supplied by the State for State-unique drugs. Also, some data updates are received from First Data Bank through the National ProDUR system.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_DRUG	System assigned key for a unique drug	NUMBER	9	0	Y
CDE_NDC	National Drug Code is comprised of a five-byte numeric labeler code, four-byte numeric product code and a two-byte numeric package code. Used to uniquely identify a drug, its labeler and package size of a product for pricing and service/prior authorization.	CHAR	11	0	N
SAK_DRUG_MANUF	This is the system assigned key used to uniquely identify the drug manufacturer.	NUMBER	9	0	N
SAK_DRUG_GEN	The system assigned key for a unique generic drug.	NUMBER	9	0	N
SAK_DRUG_RBT_LBLR	System assigned key that is four bytes long	NUMBER	9	0	N
CDE_UPDATE_TYPE	Indicates if drug record should by updated by Drug Update process. 'Y' = Yes (Update), 'N' = No (Bypass).	CHAR	1	0	N
CDE_DRUG_CATGRY	The Drug Category Code indicates that a drug product belongs to a category that is commonly treated as an exception in third party plans.	CHAR	1	0	N

Column Name	Description	Type	Length	Precision	Primary Key
CDE_DEA	The Drug Enforcement Administration (DEA) code denotes the degree of potential abuse and Federal control of a drug. This code is subject to change by Federal regulation.	CHAR	1	0	N
CDE_DRUG_CLASS	Classifies a drug by its availability to the consumer according to federal specifications. Values may change even after the NDC has become obsolete. Valid Values are: O = Over-the-Counter. A prescription is not required per the product labeling. F = Prescription required per the product labeling.	CHAR	1	0	N
CDE_DRUG_FORM	The Drug Form Code indicates the basic drug measurement unit for performing price calculations. The current codes are: EA - (tables, kits and so on) ML - (liquids) GM - (solids).	CHAR	2	0	N
CDE_DRUG_SOURCE	The Drug Source Code differentiates single source from multiple source drugs. The current codes are: 1 - Multiple sources & 2 - Single source. Note that this field does not distinguish between the "innovator" products and its substitutes.	CHAR	1	0	N
CDE_THERA_CLS_FDA	The HCFA FDA Therapeutic Equivalency Code (HCFA_FDA) indicates that although the drugs may have a different therapeutic classification, the FDA considers them therapeutically equivalent. The HCFA_FDA is provided on the Health Care Financing Administration's quarterly tape.	CHAR	2	0	N

Column Name	Description	Type	Length	Precision	Primary Key
CDE_THERA_CLS_AHFS	Identifies the pharmacologic therapeutic category of the drug product according to the American Hospital Formulary Service (AHFS) classification system.	CHAR	10	0	N
CDE_NDC_FORMAT	The NDC Format Indicator (NDCFI) is used to identify the original 10-character format of the NDC and the type of code, such as NDC, UPC, and HRI. See the "National Drug Code Configuration Editorial Policies" and "National Drug Code Format Indicator Editorial Policies" in the NDDF documentation for further discussion.	CHAR	1	0	N
NUM_TOP_VOL_RANK	Indicates the top volume products' rankings as determined by syndicated market research studies. Volume rankings are updated annually based on the total number of prescriptions dispensed (new and refills) through retail pharmacy outlets. All forms and package sizes of the product are factored into the aggregate rank; therefore, each applicable NDC carries the same rank number. For products ranked below the top 200 (FDB) or top 999 (Micromedex), this field is 0.	NUMBER	4	0	N
DSC_NDC	Contains the name that appears on the package label provided by the manufacturer. This column is populated for all products, brand and generic.	VARCHAR2	35	0	N
DSC_DRUG_PACK	The Package Description Mnemonic is the abbreviated description of the package.	CHAR	10	0	N

Column Name	Description	Type	Length	Precision	Primary Key
DTE_NDDF_ADD	The date on which the drug record was added to the National Drug Data File (NDDF) master file	NUMBER	8	0	N
DTE_NDDF_UPDT	Date of update (DUPDC) of NDC Record (NDCTBL) on the weekly NDDF update file.	NUMBER	8	0	N
DTE_LST_CHG_AWP	The date on which the drug's current blue book average wholesale unit or package price was changed on the NDDF master file	NUMBER	8	0	N
DTE_TERM_HCFA	This date is supplied on the CMS quarterly tape. The date is actually supplied to CMS from the drug manufacture/distributor. The date represents the shelf life expiration date of the last batch produced.	NUMBER	8	0	N
DTE_DRUG_OBSOLETE	The date on which the drug product was no longer available in the market place as per the manufacturer's notification, or the best estimate of that date.	NUMBER	8	0	N
IND_DRUG_GENERIC	The Generic Product Indicator (GPI) will distinguish a product as either a generic drug product or as the more expensive branded drug products.	CHAR	1	0	N
IND_DRUG_INNOV	This field identifies the original innovator product for a particular generic code number. Values are: (0)=Default - non innovator drug; (1)=Innovator - held original patent. It is possible to have more than one product to appear to be the innovator.	CHAR	1	0	N

Column Name	Description	Type	Length	Precision	Primary Key
IND_DRUG_STD_PACK	Identifies the package size and associated price vectors to be used when pricing the "standard package size" which is defined as 100's for non-unit dose, non-prepack tablets and capsules and 473 or 480 ml for liquids. Current codes are: 0 -All & 1-Std.	CHAR	1	0	N
IND_SYSTEMIC	This indicates if the drug is systemic, which means that it can "seep" into the body, be absorbed, and affect the entire body. Values are: "S" - Systemic, "N" - Non-systemic, "O" - Other.	CHAR	1	0	N
IND_UNIT_DOSE	Marks a drug as packaged in unit doses. Unit dose is defined by FDB as all products labeled as Unit Dose by the mfr. This indicator does not apply to injectable products, suppositories, or powder packets. Current codes are: 1=Unit Dose & 0=All other.	CHAR	1	0	N
IND_UNIT_OF_USE	The Unit of Use (UU) field denotes those packages which are supplied with appropriate labeling and (usually) child resistant closures and are appropriate to dispense as a unit. Valid values are: 1 = Unit of Use, 0 = All Other Products	CHAR	1	0	N
QTY_DRUG_PACK_SZ	This field contains the metric quantity used to derive a unit price. It is the usual labeled quantity from which the pharmacist dispenses, such as 100 tablets, 1000 capsules, 20 ml vial, and so on.	NUMBER	11	3	N
QTY_SHELF_PACK	The number of saleable units packed together and sold as a unit	NUMBER	9	0	N

Column Name	Description	Type	Length	Precision	Primary Key
QTY_SHIPPER	The number of saleable units (packages) in the case available from the manufacturer	NUMBER	9	0	N
QTY_DRUG_CASE_SZ	This field is used as a divisor. The labeler's catalog case price is divided by the Case Size to calculate the single package price.	NUMBER	9	0	N
CDE_STATUS1	This field is used to identify whether the drug has been replaced or reused. The valid values are 'R' for replace and 'U' for reuse.	CHAR	1	0	N
CDE_DRUG_STATUS	This field is used to identify the status of a drug. A drug will be marked as either Active (A) or Inactive (I). This information is used to allow the user to review additions and updates to a drug before they are activated for use by the MMIS.	CHAR	1	0	N
DSC_LN	The Label Name contains a combination of the drug name appearing on the package label, the strength description, and the dosage form description for a specified product.	VARCHAR2	40	0	N
IND_HCFA_UNIT	Indicates a product's unit of measure, as supplied on the CMS quarterly update. These units represent the standard measurements used by CMS for determining rebate quantities.	CHAR	3	0	N

Column Name	Description	Type	Length	Precision	Primary Key
NUM_HCFA_PS	CMS Units Per Package Size indicates the number of units per package as supplied on the Centers for Medicare and Medicaid Services' (CMS) quarterly update. This column must be used in conjunction with the CMS Unit Type Indicator to determine the appropriate number of units. Information in this column is provided by CMS and may vary from the First DataBank Package Size and Drug Form Code.	NUMBER	11	3	N
DTE_HCFA_APPC	CMS Approval date contains the date the FDA approves a product for sale in the marketplace, for products introduced after 9/30/90 (the date the Rebate program started). Products introduced prior to this date will contain 19900930 in this column in lieu of the actual market entry date. This data is supplied on the CMS quarterly update. The date is supplied to CMS by the drug labeler.	NUMBER	8	0	N
DTE_HCFA_MRKC	CMS Market Entry Date contains the date a manufacturer releases the drug product to the marketplace, for products introduced after 9/30/90 (the date the Rebate program started). Products introduced prior to this date will contain 19900930 in this column in lieu of the actual market entry date. This data is supplied on the CMS quarterly update. The date is provided to CMS by the drug labeler.	NUMBER	8	0	N

Column Name	Description	Type	Length	Precision	Primary Key
CDE_OBC3	Orange Book Code (three-Byte version) identifies the equivalency ratings assigned to approved prescription products according to the FDA's Approved Drug Products with Therapeutic Equivalence Evaluations (Orange Book).	CHAR	3	0	N
CDE_UPDATE_SOURCE	Indicates source of drug update data. F = FDB or M = Micromedex.	CHAR	1	0	N
IND_GNI	A one-character alphanumeric column that specifies whether a product is a brand named product, a generically named product, or an alternative product, using the product name as the criteria.	CHAR	1	0	N
IND_MAINT	A one-character alphanumeric column that identifies a drug as a maintenance drug. Blank = Not a maintenance drug 1 = Maintenance drug	CHAR	1	0	N

2.13.17 T_FIN_PAYER_COB

This represents pairs of Financial Payers that Coordinate payment of Benefits. These pairs are matched against pairs from threads in the PAYER hierarchy table (T_PAYER_HIERARCHY). The order of the SAK_FIN_PAYER is significant. It does not follow that if Payer 1 coordinates with Payer 2 that Payer 2 coordinates with Payer 1. For example a hierarchy thread may have Payer 1 followed by Payer 2. Another thread may have Payer 3 followed by Payer 2 and then Payer 1. Payer 1 may coordinate with Payer 2 in the first thread, but Payer 2 may not coordinate with Payer 1 in the second thread. If order is not relevant then two rows must be inserted in this table, one for each combination for the specific thread.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_FIN_COB_GRP	This is a unique identifier for a row on this table.	NUMBER	9	0	Y
SAK_FIN_PAYER	The system assigned key that identifies a unique payer within interChange.	NUMBER	9	0	N

Column Name	Description	Type	Length	Precision	Primary Key
SAK_FIN_PAYER_CHILD	The system assigned key that identifies a unique payer within interChange.	NUMBER	9	0	N
DTE_EFFECTIVE	The date the financial payer COB took effect	NUMBER	0	0	N
DTE_END	The date the financial payer COB is no longer in effect	NUMBER	0	0	N
DTE_INACTIVE	This is the date this row becomes inactive. For new claims the ICN is compared to this date. If the ICN date is greater than or equal to this date, the row will not be used to process the claim.	NUMBER	8	0	N

2.13.18 T_FINANCIAL_PAYER

The Financial payer will generally represent a unique organization responsible for the underwriting of the claims transactions to be paid.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_FIN_PAYER	The system assigned key that identifies a unique payer within interChange.	NUMBER	9	0	Y
SAK_FIN_CYCLE	The system assigned key that identifies a unique financial cycle within interChange.	NUMBER	9	0	N
NUM_HIER_ID_PAYER	This is another system assigned identifier that is used to identify a hierarchy thread that contains this Financial Payer.	NUMBER	9	0	N
CDE_FIN_PAYER	An optional business code value used to identify the payer. This code may be used for reporting or identifying payers on feeds to external systems.	CHAR	4	0	N
DSC_FIN_PAYER_SHORT	A short business description of the payer.	VARCHAR2	30	0	N
DSC_FIN_PAYER	A business description of the payer that will be used for reporting and windows.	VARCHAR2	1000	0	N

Column Name	Description	Type	Length	Precision	Primary Key
IND_RECOUP_OTH_PAY	Indicates whether this payer is allowed to recoup funds from other payers claims payments or only recoup funds from the same payers claim payments.	CHAR	1	0	N
IND_OTH_PAY_RECOUP	Indicates whether this payer allows other payers to recoup funds from this payers claim payments or if this payer does not allow this.	CHAR	1	0	N
IND_BUDGET	Indicates if this payer limits the amount of claims paid based on a budget.	CHAR	1	0	N
DTE_EFFECTIVE	The date the claim type to program restriction becomes effective for use in claims editing.	NUMBER	8	0	N
DTE_END	The last date the claim type to program restriction is effective for use in claims editing.	NUMBER	8	0	N
DTE_INACTIVE	This is the date this row becomes inactive. For new claims the system date is compared to this date. If the system date is greater than or equal to this date, the row will not be used to process the claim.	DATE	8	0	N

2.13.19 T_GROUP_STD_BNFT

This is the "leaf" node of the benefit hierarchies. The bottom group in the T_BNFT_GROUP_STD points to this table and relates that leaf group to a specific benefit.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_BNFT_GROUP	System assigned key for a unique benefit group.	NUMBER	9	0	Y

Column Name	Description	Type	Length	Precision	Primary Key
CDE_BENEFIT_TYPE	The benefit type code is used to identify the coding scheme for a service. Valid schemes include procedure (HCPCS & CPT-4), ICD-9-CM procedures, ICD-9-CM diagnosis, Revenue Code and Drug codes (NDCs). Other allowed values may identify GROUPS of benefits such as Procedure groups and so on..	CHAR	3	0	Y
SAK_BENEFIT	System assigned key for a unique drug	NUMBER	9	0	Y
SAK_CLASS_HIERARCHY	Unique row identifier for this table.	NUMBER	9	0	Y

2.13.20 T_HIERARCHY_NODE

A given thread may have as many participants as required by business policy. This table is used to identify the participating entities as well as the processing sequence within the thread. The entity types in a thread are benefit plans.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_THREAD	System assigned key used to uniquely identify a hierarchy thread. The hierarchy thread is used to identify the order of processing of Financial Payers, Benefit Plans, or Assignment Plans for beneficiaries who are enrolled in multiple entities of any of these types..	NUMBER	9	0	Y
SAK_PUB_HLTH	System assigned internal key for a member plan.	NUMBER	9	0	N
NUM_SEQUENCE_THREAD	This column determines the processing order for the entities participating in the thread. For example, if the thread is of type 'PAYR', then this sequence number determines which Financial Payer will be processed first, second, and so on.	NUMBER	4	0	N

2.13.21 T_MODIFIER

A modifier provides the means by which the reporting physician or provider can indicate that a service or procedure that has been performed has been altered by some specific circumstance but not changed in its definition or code. The judicious application of modifiers obviates the necessity for separate procedure listings that may describe the modifying circumstance. Modifiers may be used to indicate to the member of a report that: 1. A service or procedure has both a professional and technical component. 2. A service or procedure was performed by more than one physician and/or in more than one location. 3. A service or procedure has been increased or reduced. 4. Only part of a service was performed. 5. An adjunctive service was performed. 6. A bilateral procedure was performed. 7. A service or procedure was provided more than once. 8. Unusual events occurred. The attribute CDE indicates the type of modifier as follows: Code Description Affect on Claim. D Denial Will cause a detail to deny. 1 Pricing Used to "look up" the allowed amount for a procedure. 2 Processing Will change the allowed amount by a specified percentage or dollar amount or change allowed units by a specified quantity. 3 Informational Provides additional information about the procedure performed. 4 Review Indicates that the detail should be suspended for manual review. M Max Payment Indicates the maximum payment allowed for a procedure billed with modifier of this type. The effective date is used to specify when the modifier and it's type is effective for claims processing.

Column Name	Description	Type	Length	Precision	Primary Key
CDE_PROC_MOD	The modifier code used to further describe a procedure.	CHAR	2	0	Y
DTE_EFFECTIVE	The date that the modifier and its type are to become effective for claims processing.	NUMBER	8	0	N
DTE_END	The date that the modifier and its type are no longer in effect for claims processing.	NUMBER	8	0	N
DSC_MODIFIER	Short text that describes the modifier.	VARCHAR2	40	0	N
DSC_LONG	Long text that describes the modifier.	VARCHAR2	4000	0	N
CDE	Describes the usage/type of the modifier. Valid values are Pricing (1), Processing (2), Information (3), and Suspend for Manual Review (4).	CHAR	1	0	N
SAK_MODIFIER	This is a unique identifier for a row on this table.	NUMBER	9	0	N

Column Name	Description	Type	Length	Precision	Primary Key
CDE_CATEGORY	Indicates whether the modifier is a HCPCS or Ambulance Modifier. Valid values are A(Ambulance) and H(HCPCS)	CHAR	1	0	N
DTE_CMS_ADD	Date that CMS added the Modifier	NUMBER	8	0	N
DTE_CMS_TERM	Date that CMS terminated the Modifier.	NUMBER	8	0	N

2.13.22 T_MODIFIER_TYPE

A modifier is assigned a modifier type. The modifier types are used to describe the way a modifier will be used in claims processing. A modifier can only be assigned to one type.

Column Name	Description	Type	Length	Precision	Primary Key
CDE	Code used to identify type of modifier.	CHAR	1	0	Y
DSC_25	Text description for a modifier type.	CHAR	25	0	N

2.13.23 T_OCCURRENCE

Valid values and descriptions of codes which define a significant event relating to a particular UB claim that may affect payer processing.

Column Name	Description	Type	Length	Precision	Primary Key
CDE_OCCURRENCE	Code which defines a significant event relating to a particular UB claim that may affect payer processing.	CHAR	2	0	Y
DSC_OCCURRENCE	"The text description of an occurrence code."	CHAR	50	0	N

2.13.24 T_PATIENT_STAT

Indicates all valid patient status values and their descriptions.

Column Name	Description	Type	Length	Precision	Primary Key
CDE_PATIENT_STATUS	Indicates the status of the member as of the ending service date of the period covered on a UB claim.	CHAR	2	0	Y
DSC_PATIENT_STATUS	Description of the status of the member as of the ending service date of the period covered on a UB claim.	CHAR	80	0	N

2.13.25 T_PAYER_HIERARCHY

The PAYER hierarchy contains the threads used to control the order of claim adjudication at the PAYER level. Payer threads are ordered sets of Payers that may cover a member concurrently.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_THREAD_PAYER	System assigned key used to uniquely identify a hierarchy thread. The hierarchy thread is used to identify the order of processing of Financial Payers for member beneficiaries who are concurrently enrolled under multiple payers.	NUMBER	9	0	Y
DTE_EFFECTIVE	Date the hierarchy thread becomes valid for use in claims processing.	NUMBER	8	0	N
DTE_END	The last date the thread is valid for use in claims processing. The thread may still be used when the date of service on the claim is within the effective dates of the thread. To stop the use of a particular thread altogether the DTE_INACTIVE date must be used.	NUMBER	8	0	N
DTE_INACTIVE	This is the date when the usage of this thread was stopped. From this date onwards the thread will not be used regardless of the date of service billed on a claim.	NUMBER	8	0	N

Column Name	Description	Type	Length	Precision	Primary Key
NUM_PGM_1_32	Bit flags for Financial Payer NUM_HIER_ID 1 - 32. Used as part of the primary key and editing to determine order of processing for a Financial Payer. If NUM_HIER_ID column of T_FINANCIAL_PAYER IS BETWEEN 1 AND 32 for a Payer on the thread, then corresponding bit in this column will be set to 1 otherwise the bit will be set to 0.	NUMBER	10	0	N
NUM_PGM_33_64	Bit flags for Financial Payer NUM_HIER_ID 1 - 32. Used as part of the primary key and editing to determine order of processing for a Financial Payer. If NUM_HIER_ID column of T_FINANCIAL_PAYER IS BETWEEN 33 AND 64 for a Payer on the thread, then corresponding bit in this column will be set to 1 otherwise the bit will be set to 0.	NUMBER	10	0	N
NUM_PGM_65_96	Bit flags for Financial Payer NUM_HIER_ID 1 - 32. Used as part of the primary key and editing to determine order of processing for a Financial Payer. If NUM_HIER_ID column of T_FINANCIAL_PAYER IS BETWEEN 65 AND 96 for a Payer on the thread, then corresponding bit in this column will be set to 1 otherwise the bit will be set to 0.	NUMBER	10	0	N

2.13.26 T_PAYER_NODE

A given thread may have as many participants as required by business policy. This table is used to identify the participating entities as well as the processing sequence within the thread. The entity types in a thread are Payers.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_THREAD_PAYER	System assigned key used to uniquely identify a hierarchy thread. The hierarchy thread is used to identify the order of processing of Financial Payers for member beneficiaries who are concurrently enrolled under multiple payers.	NUMBER	9	0	Y
SAK_FIN_PAYER	The system assigned key that identifies a unique payer within interChange.	NUMBER	9	0	Y
NUM_SEQUENCE_THREAD	This column determines the processing order for the entities participating in the thread. For example, if the thread is of type 'PAYR', then this sequence number determines which Financial Payer will be processed first, second, and so on.	NUMBER	4	0	N

2.13.27 T_PGM_CT_XREF

This entity is used in claims editing to determine if a claim type is billable for a specific program. The IND_CT_EDITING attribute on the T_PUB_HLTH_PGM tables indicates what type of editing is to be performed. If the indicator is set to 'N' (non), no claim type to program editing is performed. If the indicator is set to 'I' (include), only the claim types listed are billable for the specified program. If the indicator is set to 'E' (exclude), the claim types listed are not billable for the specified program.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_PUB_HLTH	System assigned internal key for a member plan.	NUMBER	9	0	Y
CDE_CLM_TYPE	Value for the type of claim that can be processed in the MMIS system.	CHAR	1	0	Y
DTE_EFFECTIVE	The date the claim type to program restriction becomes effective for use in claims editing.	NUMBER	8	0	N

Column Name	Description	Type	Length	Precision	Primary Key
DTE_END	The last date the claim type to program restriction is effective for use in claims editing.	NUMBER	8	0	N

2.13.28 T_PGM_DEPENDENCY

This entity is used to list programs that are dependent and must exist together on member eligibility.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_PUB_HLTH	System assigned internal key for a medical assistance program.	NUMBER	9	0	Y
SAK_CHILD_PGM	System assigned internal key for a medical assistance program.	NUMBER	9	0	Y

2.13.29 T_PGM_EXCLUSION

This entity is used to list programs that cannot exist together on member eligibility.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_PUB_HLTH	System assigned internal key for a medical assistance program.	NUMBER	9	0	Y
SAK_EXCLUDED_PGM	System assigned internal key for a medical assistance program.	NUMBER	9	0	Y
IND_MC_EXCL	This column is used to indicate which exclusion table rows are for Managed Care exclusions specifically.	CHAR	1	0	N

2.13.30 T_PGM_HIERARCHY

The program hierarchy contains the threads used to control the order of claim adjudication at the Benefit Plan level. The top level is controlled with Payer threads. Payer threads are ordered sets of Payers that may cover a member concurrently. These benefit plan threads are ordered sets of benefit plans that may cover a member concurrently.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_THREAD	System assigned key used to uniquely identify a hierarchy thread. The hierarchy thread is used to identify the order of processing of Financial Payers, Benefit Plans, or Assignment Plans for beneficiaries who are enrolled in multiple entities of any of these types.	NUMBER	9	0	Y
SAK_FIN_PAYER	The system assigned key that identifies a unique payer within interChange.	NUMBER	9	0	N
DTE_EFFECTIVE	Date the hierarchy thread becomes valid for use in claims processing.	NUMBER	8	0	N
DTE_END	The last date the thread is valid for use in claims processing. The thread may still be used when the date of service on the claim is within the effective dates of the thread. To stop the use of a particular thread altogether the DTE_INACTIVE date must be used.	NUMBER	8	0	N
DTE_INACTIVE	This is the date when the usage of this thread was stopped. From this date onwards the thread will not be used regardless of the date of service billed on a claim.	NUMBER	8	0	N

Column Name	Description	Type	Length	Precision	Primary Key
NUM_PGM_1_32	Bit flags for T_PUB_HLTH_PGM.NUM_HIER_ID_BNFT 1 to 32. Used as part of the primary key and editing to determine assignment plans order of processing within a Financial Payer. If NUM_HIER_ID_BNFT column of T_PUB_HLTH_PGM IS BETWEEN 1 AND 32 for a benefit plan on the thread, then corresponding bit in this column will be set to 1 otherwise the bit will be set to 0.	NUMBER	10	0	N
NUM_PGM_33_64	Bit flags for T_PUB_HLTH_PGM.NUM_HIER_ID_BNFT 33 to 64. Used as part of the primary key and editing to determine assignment plans order of processing within a Financial Payer. If NUM_HIER_ID_BNFT column of T_PUB_HLTH_PGM IS BETWEEN 33 AND 64 for a benefit plan on the thread, then corresponding bit in this column will be set to 1 otherwise the bit will be set to 0.	NUMBER	10	0	N
NUM_PGM_65_96	Bit flags for T_PUB_HLTH_PGM.NUM_HIER_ID_BNFT 65 to 96. Used as part of the primary key and editing to determine assignment plans order of processing within a Financial Payer. If NUM_HIER_ID_BNFT column of T_PUB_HLTH_PGM IS BETWEEN 65 AND 96 for a benefit plan on the thread, then corresponding bit in this column will be set to 1 otherwise the bit will be set to 0.	NUMBER	10	0	N

2.13.31 T_PLACE_OF_SERVICE

Location where medical services were provided.

Column Name	Description	Type	Length	Precision	Primary Key
CDE_POS	Place of medical assistance service code.	CHAR	2	0	Y
DSC_POS	Description of place where medical assistance service is performed.	CHAR	50	0	N
IND_PROC_IN_EXCLUD	Indicates whether the associated list of procedure codes is valid or invalid for the place of service.	CHAR	1	0	N
IND_DIAG_IN_EXCLUD	Indicates whether the associated list of diagnosis codes is valid or invalid for the place of service.	CHAR	1	0	N

2.13.32 T_PRCNG_TYPE_SPEC

This is a list of provider type/specialty combinations that restricts rows of the T_PRICING_METHOD table. A value of 00 or 000 for type or specialty represents no restriction on type or specialty, respectively. For example, if type = 01 and specialty = 000 then the restriction applies only to providers with type = 01 regardless of the specialty.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_REIMB_AGRMNT	Unique identifier for rows on this table. This is also the unique index for the audit trail table and will not change.	NUMBER	9	0	Y
CDE_PROV_TYPE	Type that a provider is licensed for.	CHAR	2	0	Y
CDE_PROV_SPEC	A code representing the specialized area of practice for a provider.	CHAR	3	0	Y
CDE_PR_ROLE	This code represents the provider role on the claim. For example, a value of BILL REPRESENTS THE BILLING PROVIDER ON THE CLAIM.	CHAR	4	0	Y

2.13.33 T_PRICING_IND

Pricing indicator specifies the payment methodology that should be applied or was applied when determining the payment to be made to a provider for providing a service (procedure).

Column Name	Description	Type	Length	Precision	Primary Key
IND_PRICING	Pricing indicator which dictates the method by which a procedure must be priced or indicates how a claim detail was priced.	CHAR	6	0	Y
DSC_25	Text description of the pricing indicator.	CHAR	25	0	N
IND	Indicates if the pricing indicator value is applicable for HCPC procedures. A value of "Y" signifies the indicator is valid for HCPC procedures and "N" says it is not.	CHAR	1	0	N

2.13.34 T_PROC

CPT4 or HCPCS procedure and its descriptions. Procedures are assigned codes that uniquely identify the service performed.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_PROCEDURE	System assigned key used to uniquely identify a procedure.	NUMBER	9	0	Y
CDE_PROC	Code used to identify a medical, dental, or DME procedure.	CHAR	6	0	N
DSC_PROCEDURE	A short medical description of a specific, singular medical or dental service which is performed for the express purpose of identification or treatment of the patient's condition.	CHAR	40	0	N
DSC_LONG	A long medical description of a specific, singular medical or dental service which is performed for the express purpose of identification or treatment of the patient's condition.	VARCHAR2	4000	0	N
DSC_EOMB	Explanation of Medical Benefits description used on Member EOMB.	CHAR	30	0	N
DTE_HCFA_ADD	Defined by CMS as the date the HCPCS code was added to the CMS common procedure coding system.	NUMBER	8	0	N
DTE_HCFA_TERM	Defined by CMS as the last date for which a procedure code may be used by Medicare providers.	NUMBER	8	0	N
CDE_MC_SVC_CLASS	The service class indicates the type of services provided.	CHAR	2	0	N
DSC_LAY	Description of the code in layman terms.	VARCHAR2	100	0	N

Column Name	Description	Type	Length	Precision	Primary Key
CDE_CMS_TOS	The carrier assigned CMS type of service which describes the particular kind(s) of service represented by the procedure code	CHAR	1	0	N
CDE_MCAR_COVRG	A code denoting Medicare coverage status	CHAR	1	0	N

2.13.35 T_PROC_ICD9

Surgical or diagnostic procedure and its descriptions. Procedures are assigned codes that are used to identify the service performed.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_PROC_ICD9	System assigned key used to uniquely identify an ICD-9-CM procedure.	NUMBER	9	0	Y
CDE_PROC_ICD9	Code which indicates a specific, surgical or diagnostic procedure which is performed for the express purpose of identification or treatment of the patient's condition.	CHAR	4	0	N
DSC_PROCEDURE	A short medical description of a specific, surgical or diagnostic procedure which is performed for the express purpose of identification or treatment of the patient's condition.	CHAR	40	0	N
DSC_LONG	A long medical description of a specific, surgical or diagnostic procedure which is performed for the express purpose of identification or treatment of the patient's condition.	CHAR	250	0	N

Column Name	Description	Type	Length	Precision	Primary Key
CDE_CLASS	Describes the operative or non operative class of a procedure. 1 = Class 1, surgery, 2 = Class 2, significant procedure, 3 = Class 3 significant procedure, 4 = Class 4, other	CHAR	1	0	N
CDE_MC_SVC_CLASS	The service class indicates the type of services provided.	CHAR	2	0	N
DSC_LAY	Description of the code in layman terms.	VARCHAR2	100	0	N

2.13.36 T_PR_ENROLL_PGM

This table contains the list of programs that a provider can be enrolled in.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_PROV_PGM	This is the system assigned key to the provider contract under which providers may be certified..	NUMBER	9	0	Y
SAK_FIN_PAYER	The system assigned key that identifies a unique payer within interChange.	NUMBER	9	0	N
CDE_PROV_PGM	Identifies the medical assistance programs that a provider can enroll in.	CHAR	5	0	N
DSC_PROV_PGM	This is the short description to the provider enrollment program.	CHAR	20	0	N
DSC_PROV_PGM_LONG	This is the long description to the provider enrollment program.	VARCHAR2	4000	0	N

Column Name	Description	Type	Length	Precision	Primary Key
IND_CT_EDITING	The IND_CT_EDITING attribute indicates what type of claim type to program editing is to be performed. If the indicator is set to 'N' (non), no claim type to program editing is performed. If the indicator is set to 'I' (include), only the claim types listed are billable for the specified program. If the indicator is set to 'E' (exclude), the claim types listed are not billable for the specified program.	CHAR	1	0	N
DTE_EFFECTIVE	The date the claim type to program restriction becomes effective for use in claims editing.	NUMBER	8	0	N
DTE_END	The last date the claim type to program restriction is effective for use in claims editing.	NUMBER	8	0	N
DTE_INACTIVE	This is the date when the usage of this thread was stopped. From this date onwards the thread will not be used regardless of the date of service billed on a claim.	DATE	8	0	N

2.13.37 T_PR_PHP_ELIG

The Provider Program Eligibility table is a cross-reference between the providers and the programs for which they are eligible.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_PROV	System assigned key that uniquely identifies the provider enrollment tracking.	NUMBER	9	0	Y
CDE_SERVICE_LOC	Suffix added to the provider number to identify the various locations that a provider does business.	CHAR	1	0	Y
SAK_SHORT	This is the system-assigned internal key that is two-bytes long. It is used to uniquely identify a row without using updateable attributes.	NUMBER	4	0	Y

Column Name	Description	Type	Length	Precision	Primary Key
SAK_PROV_PGM	This is the system assigned key to the provider contract under which providers may be certified..	NUMBER	9	0	Y
SAK_FIN_PAYER	The system assigned key that identifies a unique payer within interChange.	NUMBER	9	0	N
DTE_EFFECTIVE	This is the effective date when the maximum recoupment amount takes effect.	NUMBER	8	0	N
DTE_END	This is the end date when the maximum recoupment amount ends.	NUMBER	8	0	N
DTE_INACTIVE	This is the date when the usage of this thread was stopped. From this date onwards the thread will not be used regardless of the date of service billed on a claim.	DATE	8	0	N
CDE_ENROLL_STATUS	This is the letter assigned to the enrollment status description to uniquely identify it. Examples of valid values are: R=Retired, D=Deceased, M=Return Mail, I=Term by IFSSA, H=Term by CMS, B=Term by HPB, and A=Active.	CHAR	1	0	N

2.13.38 T_PR_SPEC_CDE

Valid provider specialties and their PA and Procedure inclusions or exclusions and whether this specialty is eligible as a Primary Medical Provider in managed care.

Column Name	Description	Type	Length	Precision	Primary Key
CDE_PROV_SPEC	A code representing the specialized area of practice for a provider.	CHAR	3	0	Y
DSC_PROV_SPEC	Description associated with a provider specialty code.	CHAR	50	0	N
IND_PROC_IN_EXCLUD	Indicates whether the associated list of procedure codes is valid or invalid for the specialty.	CHAR	1	0	N

2.13.39 T_PR_TYPE_CDE

Provider type code used to identify the major grouping of service(s) or occupation(s) of health care providers.

Column Name	Description	Type	Length	Precision	Primary Key
CDE_PROV_TYPE	Type that a provider is licensed for.	CHAR	2	0	Y
DSC_PROV_TYPE	Description associated with a provider type code.	CHAR	50	0	N
IND_LVL_CARE	Indicates whether the corresponding provider type requires a level of care segment or not.	CHAR	1	0	N
IND_PEER_GROUP	Indicates whether the corresponding provider type requires a peer group or not.	CHAR	1	0	N
IND_LICENSE_REQD	Indicates whether the corresponding provider type requires a license.	CHAR	1	0	N
IND_GROUP	When used on this table, this attribute indicates whether a provider type is only valid for a group. A value of 'Y' indicates that a member of a group may not be assigned the corresponding provider type.	CHAR	1	0	N
IND_COPAY	This field indicates whether this provider type is set up for copay or not. The valid values are 'Y' (Yes) and 'N' (No).	CHAR	1	0	N

2.13.40 T_PUB_HLTH_PGM

This entity represents the Benefit or Assignment plans defined within a Financial Payer supported by the system. A beneficiary/member will be enrolled in one or more of these plans based on the criteria established by the Financial Payer. The Benefit Plan lists the services for which the beneficiary/member is eligible to receive. A specific service may be subject to different rules and restrictions under different Benefit Plans. An Assignment plan is a generalization of the traditional Lockin concept. A beneficiary/member may be assigned to a provider or provider organization in order to receive certain services. These services must be provided or, in some cases, referred by the assigned provider. The services may be reimbursed on a fee for service or capitation basis. The provider may be compensated for acting as a gate keeper in some instances.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_PUB_HLTH	System assigned internal key for a member plan.	NUMBER	9	0	Y
SAK_FIN_PAYER	The system assigned key that identifies a unique payer within interChange.	NUMBER	9	0	N
CDE_PGM_HEALTH	Identifies the medical assistance program that is supported in the system.	CHAR	5	0	N
DSC_PGM_HEALTH	Describes the medical assistance program.	VARCHAR2	50	0	N
DSC_PGM_DEFINITION	Text definition of program describing who is eligible and what types of services are provided.	VARCHAR2	4000	0	N
CDE_TYPE_PLAN	This is the type of plan. There are two types of Plans. For Benefit Plans, the value in this column is 'BNFT'. For Assignment Plans, the value in this column is 'ASGN'.	CHAR	4	0	N
IND_RECIP_ONLY	Yes/No indicator used to identify programs that are used for member enrollment only. No services are covered by the program.	CHAR	1	0	N

Column Name	Description	Type	Length	Precision	Primary Key
IND_MAJOR_PGM	Yes/No indicator used to identify a program is a major program. A major program is a program that can stand alone. The program needs no other program. A major program can not be combined with any other major program.	CHAR	1	0	N
IND_STAND_ALONE	Yes/No indicator used to identify a program is a stand alone program. A stand alone program can only stand by itself. No other program can exist.	CHAR	1	0	N
IND_CT_EDITING	The IND_CT_EDITING attribute indicates what type of claim type to program editing is to be performed. If the indicator is set to 'N' (non), no claim type to program editing is performed. If the indicator is set to 'I' (include), only the claim types listed are billable for the specified program. If the indicator is set to 'E' (exclude), the claim types listed are not billable for the specified program.	CHAR	1	0	N
IND_COPAY	Yes/No indicator used to identify programs that qualify for copay calculations during claims payment determination.	CHAR	1	0	N
IND_DUAL	Yes/No indicator used to identify a program is a Dual program. A dual program is a program that can either stand alone or with another major/dual program.	CHAR	1	0	N

Column Name	Description	Type	Length	Precision	Primary Key
IND_TPL_ACTION	This indicator identifies the action taken for TPL editing based on the OI Plan. Note, we are allowing nulls at this time because panels outside of TPL Matrix will be inserting into this table and will not enter anything into this column. Later (new CO), we will change the other panels and change this column to not null.	CHAR	1	0	N
NUM_HIER_ID_BNFT	This is another system assigned identifier that is used to identify a hierarchy thread that contains this benefit plan within a Payer. This column contains only positive values when the CDE_TYPE_PLAN is equal to 'BNFT'. Otherwise the value of this column is zero. The value is maintained as an ascending sequence of numbers starting with 1. There is a sequence for each payer. So Payer 1 may have values 1, 2, 3; and Payer 2 might have values 1,2, 3, 4.	NUMBER	9	0	N
NUM_HIER_ID_ASGN	This is another system assigned identifier that is used to identify a hierarchy thread that contains this assignment plan. This column contains only positive values when the CDE_TYPE_PLAN is equal to 'ASGN'. Otherwise the value of this column is zero. The value is maintained as an ascending sequence of numbers starting with 1. There is a sequence for each payer. So Payer 1 may have values 1, 2, 3; and Payer 2 might have values 1,2, 3, 4.	NUMBER	9	0	N
DTE_EFFECTIVE	The date the claim type to program restriction becomes effective for use in claims editing.	NUMBER	8	0	N

Column Name	Description	Type	Length	Precision	Primary Key
DTE_END	The last date the claim type to program restriction is effective for use in claims editing.	NUMBER	8	0	N
DTE_INACTIVE	This is the date this row becomes inactive. For new claims the system date is compared to this date. If the system date is greater than or equal to this date, the row will not be used to process the claim.	DATE	8	0	N
DTE_ACTIVE	This is the date this row becomes active. For new claims the system date is compared to this date. If the system date is less than this date, the row will not be used to process the claim.	DATE	8	0	N

2.13.41 T_RATE_TYPE

The Rate Type describes the rate to use in determining provider reimbursement. The rate type is used to allow different rates to be applied for various benefit plans.

Column Name	Description	Type	Length	Precision	Primary Key
CDE_RATE_TYPE	Code used to identify the rate type to use in determining provider reimbursement.	CHAR	3	0	Y
DSC_RATE_TYPE	Description of rate type.	VARCHAR2	18	0	N

2.13.42 T_REVENUE_CODE

A three-digit code which identifies a specific accommodation or ancillary service.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_REVENUE	System assigned key used to uniquely identify a revenue code.	NUMBER	9	0	Y
CDE_REVENUE	This identifies a specific accommodation or ancillary service. Revenue codes are determined by CMS.	NUMBER	4	0	N
DSC_REV_CODE	This describes a specific accommodation or ancillary service.	CHAR	70	0	N

Column Name	Description	Type	Length	Precision	Primary Key
DTE_EFFECTIVE	This is the date of service that the revenue code became effective.	NUMBER	8	0	N
DTE_END	This is the last date of service that the revenue code is effective.	NUMBER	8	0	N
NUM_RATE_ID	This field represents a grouping of revenue codes. Each revenue code has a rate ID number and more than one revenue code will have the same number. Revenue codes are grouped this way for use in different pricing methodologies.	NUMBER	4	0	N
CDE_MC_SVC_CLASS	The service class indicates the type of services provided.	CHAR	2	0	N
IND_PROC_IN_EXCLUD	This field indicates whether the associated drug, revenue code, or procedure range is included or excluded from the billing restrictions for a provider.	CHAR	1	0	N
IND_SPEC_IN_EXCLUD	This field indicates whether the associated provider specialty is included or excluded from the billing restrictions for a revenue code.	CHAR	1	0	N

2.13.43 T_RE_ELIG

Member Program Eligibility contains the member's program eligibility segments. An eligibility segment specifies a period of time that the member is eligible for a Medical Assistance program coverage. Each eligibility segment is maintained by an effective date and end date. For further information about how the program code and aid categories are determined and how they relate, refer to the Member User Manual.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_RECIP	The system assigned internal key for a unique member.	NUMBER	9	0	Y
SAK_PGM_ELIG	Used along with the SAK RECIP to uniquely identify an eligibility segment for a particular member.	NUMBER	4	0	Y
SAK_PUB_HLTH	System assigned internal key for a member plan.	NUMBER	9	0	N
SAK_FIN_PAYER	The system assigned key that identifies a unique payer within interChange.	NUMBER	9	0	N
DTE_EFFECTIVE	The date that the member becomes eligible for the corresponding Medical Assistance program.	NUMBER	8	0	N
DTE_END	The date that the member is no longer eligible for the corresponding Medical Assistance program.	NUMBER	8	0	N
SAK_ELIG_STOP	System assigned key to uniquely identify an eligibility stop reason code and it's description.	NUMBER	9	0	N
CDE_STATUS1	The status code for the program eligibility segment. A blank means the segment is active, a 'G' means it is Guaranteed Eligibility, and an 'H' means that the segment is history and no longer valid.	CHAR	1	0	N

2.13.44 T_RE_PMP_ASSIGN

This describes a member's Primary Medical Provider (PMP) assignment. This information is required for a member to be linked to a PMP and to participate in a managed care program.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_RE_PMP_ASSIGN	This is a unique identifier for the PMP assignment.	NUMBER	9	0	Y
SAK_RECIP	The system assigned internal key for a unique member.	NUMBER	9	0	N
SAK_PUB_HLTH	System assigned internal key for a member plan.	NUMBER	9	0	N

Column Name	Description	Type	Length	Precision	Primary Key
SAK_PMP_SER_LOC	The unique system assigned key to identify a Primary Medical Provider's service location.	NUMBER	9	0	N
DTE_EFFECTIVE	This is the date that the assignment became effective.	NUMBER	8	0	N
DTE_END	This is the date that the assignment is no longer effective.	NUMBER	8	0	N
CDE_STATUS1	Specifies if the assignment is valid (space) or if the assignment has been historied ('H').	CHAR	1	0	N
IND_PRIMARY	Indicates if this is the primary assignment for the given program. In order to allow multiple assignments to the same program, (that is the member is assigned to an OB/GYN and assignment to a Family Practitioner at the same time) a single assignment MUST be designated as the primary assignment so that enrollment counts for a program are not exaggerated, and so on. The assignment that gets marked as primary is determined by the focus of the PMP.	CHAR	1	0	N
CDE_MC_REGION	This is the code that identifies a region (or Managed Care coverage area) of the state. A region is made up of different geographical areas such as zip codes, counties, or the entire state.	CHAR	5	0	N
SAK_PROV_MBR	System assigned key that uniquely identifies the provider enrollment tracking.	NUMBER	9	0	N
CDE_SERVICE_LOC_MBR	The service location of the provider that the member will be seeing for their actual care.	CHAR	1	0	N

Column Name	Description	Type	Length	Precision	Primary Key
CDE_RSN_MC_START	This indicates the reason a member was assigned to a specific PMP; for example, newly eligible or an approved change. This also includes the reasons a members relationship with a PMP was terminated; for example, death or an approved change.	CHAR	2	0	N
CDE_RSN_MC_STOP	This indicates the reason a member was assigned to a specific PMP; for example, newly eligible or an approved change. This also includes the reasons a members relationship with a PMP was terminated; for example, death or an approved change.	CHAR	2	0	N
SAK_MC_ENT_ADD	Unique identifier for the Managed Care entity (interChange eligibility, enrollment broker, business staff, system processes, and so on). This column identifies the entity that created the PMP assignment.	NUMBER	9	0	N
DTE_ADDED	Date that the PMP assignment was created/inserted.	NUMBER	8	0	N
SAK_MC_ENT_MBR	Unique identifier for the Managed Care entity (PS/2, Enrollment Broker, OHCA, system processes, and so on). This column identifies the entity that added the values for the SAK_PROV_MBR and CDE_SERVICE_LOC_MBR fields.	NUMBER	9	0	N
DTE_PROV_MBR_ADD	This is the date that the MCO's PCP (the group member) was associated with the member.	NUMBER	8	0	N
SAK_MC_ENT_CHANGE	Unique identifier for the Managed Care entity (PS/2, Enrollment Broker, OHCA, system processes, and so on). This column identifies the entity that last updated the PMP assignment.	NUMBER	9	0	N
DTE_CHANGED	This is the date that the PMP assignment was last updated.	NUMBER	8	0	N

Column Name	Description	Type	Length	Precision	Primary Key
SAK_MC_ENT_TERM	Unique identifier for the Managed Care entity (PS/2, Enrollment Broker, OHCA, system processes, and so on). This column identifies the entity that end dated the PMP assignment.	NUMBER	9	0	N
DTE_TERMED	This is the actual date that the PMP assignment was end dated. It is not necessarily the date that the PMP assignment is no longer effective (the end_date).	NUMBER	8	0	N

2.13.45 T_RU_ACTION

An action is executed by calling the setter function of the variable and passing in the output value. This assumes an action variable has one and only one argument. SAK_ACTION CDE_VARIABLE CDE_VALUE IND_VALUE_TYPE 10 EDIT 1245 S 11 EDIT 1246 S 12 EDIT 1247 S 13 EDIT 1248 S 14 EDIT 1249 S . 27 PA true S

Column Name	Description	Type	Length	Precision	Primary Key
SAK_ACTION	This is the primary key for this table.	NUMBER	9	0	Y
CDE_VARIABLE	Points to the Transaction variable that is the direct object of the Action.	CHAR	4	0	N
CDE_VALUE	If IND_VALUE_TYPE is "A", then this column is a FK to T_RU_ARRAY. Otherwise it's an arbitrary string value.	VARCHAR2	50	0	N
CDE_VALUE_TYPE	This value is either "S" for Single Value, or "A" for ARRAY.	CHAR	1	0	N

2.13.46 T_RU_ARRAY

This is the array that contains the values for relation type variables such as modifier and modifier type.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_VALUE	Identifier for an array. IMPORTANT: There can be multiple rows with the same PK. Each row is another value of the array..	NUMBER	8	0	Y
SAK_VALUE_CHILD	If the value is -1 then CDE_VALUE contains the value for this array position..	NUMBER	9	0	N
CDE_VALUE_CHILD	If SAK_VALUE_CHILD is -1 this is the value of the array entry. Otherwise it is " ".	VARCHAR2	50	0	N
NUM_ORDER	This is the position within the array of this entry which may be an array in itself.	NUMBER	4	0	N

2.13.47 T_RU_COMPARISON

Comparisons are the leaf nodes of the Rule's Condition tree. A Rule variable is matched against a value.

A singly occurring variable (example, Procedure or Benefit Group)) that matches a single value will be matched against the value in CDE_VALUE_SET using the operator in CDE_COMPARISON_OP. If the comparison is of type R (range) then two values separated by a "-" will be in CDE_VALUE.

A singly occurring variable (example, place of service) that may be matched against a discreet list of values will have the list of values identified by CDE_VALUE_SET and located in the table T_RU_DISCREET_SET.

A tuple variable (a variable that may occur multiple times in the transaction such as procedure modifier) will have its tuple set identified by CDE_VALUE_SET and located in the table T_RU_TUPLE_SET.

SAK_COMP CDE_VARIABLE IND CDE_VALUE IND_VALUE_TYPE 1 PROC EQ 123 S; 2 DOS BW 20010924 - 20050101 R; 3 AGE BW 0 - 45 R; 4 GRP EQ 8 S; 5 CT BW 1 S

Column Name	Description	Type	Length	Precision	Primary Key
SAK_COMPARISON	Primary key.	NUMBER	9	0	Y
SAK_RULE	Primary-Key for this table	NUMBER	9	0	N

Column Name	Description	Type	Length	Precision	Primary Key
CDE_VARIABLE	Primary key for this table. The short identifier for the variable. It must be unique for the transaction that owns it.	CHAR	4	0	N
CDE_COMPARISON_OP	Possible values are "EQ", "NE", "GT", "LT", "GE", "LE", "BW", and "IN". Others may be added as needed.	CHAR	2	0	N
CDE_VALUE_COMP	If IND_VALUE_TYPE is "D", then this column is a FK to T_RU_DISCRETE_SET. If the IND is "T", then this column is a FK to T_RU_TUPLE_SET. Otherwise it's an arbitrary string value that's interpreted differently for each domain. A RangedValue (for the "BW" operation) will have two domain values separated by " - ".	VARCHAR2	50	0	N
CDE_VALUE_TYPE	If IND_VALUE_TYPE is "D", then this column is a FK to T_RU_DISCRETE_SET. If the IND is "T", then this column is a FK to T_RU_TUPLE_SET. Otherwise it's an arbitrary string value that's interpreted differently for each domain. A RangedValue (for the "BW" operation) will have two domain values separated by " - ".	CHAR	1	0	N

2.13.48 T_RU_CONDITION

A Condition may be one of two things: 1. Two CONDITIONs identified by SAK_CONDITION_LEFT and SAK_CONDITION_RIGHT AND combined with the logical connective in column CDE_LOGICAL_CONNECTIVE, or: 2. A COMPARISON pointed to by column SAK_COMPARISON. SAK_COND IND SAK_COMP SAK_COND_LEFT SAK_COND_RIGHT 1 AND -1 2 5 2 AND -1 3 4 3 1 -1 -1 4 2 -1 -1 5 3 -1 -1 6 AND -1 7 10 7 AND -1 8 9 8 4 -1 -1 9 5 -1 -1 10 NOT -1 11 -1 11 6 -1 -1

Column Name	Description	Type	Length	Precision	Primary Key
SAK_CONDITION	FK to T_RU_ACTION. It's optional. This will be greater then -1 if some action (for example setting an edit) must occur when the required variables of a rule match a transaction, but this variable does not match the transaction.	NUMBER	9	0	Y
SAK_RULE	Primary-Key for this table	NUMBER	9	0	N
SAK_CONDITION_LEFT	FK to T_RU_CONDITION. If IND_LOGICAL_OP = " ", then this is -1.Otherwise it's > 0.	NUMBER	9	0	N
SAK_CONDITION_RIGHT	FK to T_RU_CONDITION. If IND_LOGICAL_OP = " ", then this is -1.Otherwise it's > 0.	NUMBER	9	0	N
SAK_COMPARISON	Primary Key for row on T_RU_COMPARISON.	NUMBER	9	0	N
CDE_LOGICAL_CONNECTIVE	Define the relationship between the left and right conditions. Allowed values are "AND", "OR", "NOT", and " ". The value " " indicates that SAK_COMPARISON > 0.	CHAR	3	0	N

2.13.49 T_RU_DECISION

Any decision point in the transaction can be based on rules. The critical part of a decision within an instance of a transaction is that only 1 rule applies. That means rules within a decision cannot overlap such that a potential transaction exists where the conditions of both rules would match it. This concept of non-overlapping rules is a critical component of rules-based transaction processing. For example, if the transaction is Claim Submission then an instance of such transaction is a claim submitted by a provider. During the processing of that claim there are many decision points - example, is a service on the claim covered? There can only be one rule that can provide the answer. SAK CDE DSC 1 BP Member Plan - Procedure 1 BD Member Plan - Diagnosis 1 PP Provider Contract - Procedure 1 PD Provider Contract - Diagnosis 1 RP Reimbursement Agreement - Procedure 1 RD Reimbursement Agreement – Diagnosis

Column Name	Description	Type	Length	Precision	Primary Key
CDE_DECISION	Primary-Key. A short identifier for the decision. It must be unique within a transaction. Examples, BP (Member Benefit Plan - Procedure), BD (Member Benefit Plan - Diagnosis).	CHAR	4	0	Y
CDE_TRANSACTION	This is the unique identifier for this table. Example, CLMS (Claim Submission).	CHAR	4	0	N
DSC_DECISION	A descriptive name of the decision. It should be unique for a transaction. Example, is the provider allowed to bill the service on the claim?, is the Procedure covered under the Member Benefit Plan?	VARCHAR2	80	0	N
SAK_ACTION_FAILS	FK to T_RU_ACTION. It's optional. This will be greater then -1 if some action (for example setting an edit such as procedure not covered for benefit plan) must occur when all REQUIRED variables of a rule do not match the transaction.	NUMBER	9	0	N
SAK_ACTION_PASS	This is the primary key for this table.	NUMBER	9	0	N

2.13.50 T_RU_DECISION_VAR

The collection of variables used by a decision. This means rules for that decision can be built with comparisons to these variables. Required variables must appear in all rules of a decision, and thus we can index by required variables (for example, Date of Service and Procedure are required for "Member Plan - Procedure"). CDE_DECISION CDE_VARIABLE IND_REQUIRED SAK_ACTION BP PROC Y -1 BP DOS Y -1 BP AGE N 10 BP QTY N 11 BP MOD N 12 BD DIAG Y -1 BD DOS Y -1 BD AGE N 13 BD QTY N 14 PP PROC Y -1 PP DOS Y -1 PP QTY N 15 PP MOD N 16 PD PDH Y -1 PD DOS Y -1 PD QTY N 17 RP PROC Y -1 RP DOS Y -1 RP QTY N 18 RP MOD N 19 RD PDH Y -1 RD DOS Y -1 RD QTY N 20

Column Name	Description	Type	Length	Precision	Primary Key
CDE_DECISION	Primary-Key. A short identifier for the decision. It must be unique within a transaction. Examples, BP (Member Benefit Plan - Procedure), BD (Member Benefit Plan - Diagnosis).	CHAR	4	0	Y
CDE_VARIABLE	Primary key for this table. The short identifier for the variable. It must be unique for the transaction that owns it.	CHAR	4	0	Y
SAK_ACTION_NOMATCH	This is the primary key for this table.	NUMBER	9	0	Y
IND_REQUIRED	Either "Y" or "N". "Y" means that all rules of this type must have that variable in their condition, which also means we can hash by that variable.	CHAR	1	0	N

2.13.51 T_RU_DISCRETE_SET

A discrete set is a collection of values. Each value in the set is a separate row on this table, all with the same SAK_VALUE.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_VALUE	Primary Key. IMPORTANT: There can be multiple rows with the same PK. Each row is another value in the set.	NUMBER	9	0	Y
CDE_VALUE	This is string of characters. Sometimes it will represent a SAK of a data base row such as the SAK of a procedure code.	VARCHAR2	50	0	N

2.13.52 T_RU_DOMAIN

Domain is equivalent to a data-type. Integer and Date are very common data-types, but MMIS has its own data-types defined as code-tables. Each variable on the transaction has a domain data-type. Examples: Integer DATE Boolean Place of Service Procedure code Member gender.

Column Name	Description	Type	Length	Precision	Primary Key
CDE_DOMAIN	The short identifier of the domain, used as a primary key. Examples, INT (integer), DATE, BOOL (boolean), POS (place of service).	CHAR	4	0	Y
DSC_DOMAIN	A descriptive name for the domain. This may appear in the UI. Examples, Integer, Place of Service.	VARCHAR2	40	0	N
NAM_DOMAIN_PLURAL	A plural name of the domain that will be used in the UI to label a listbox of the values in the domain. Examples, Integers, Booleans, Places of Service,	VARCHAR2	40	0	N
NAM_TABLE	If this domain can define its list of values in a code table then this is set to the name of the code table. Otherwise, it is " ".	VARCHAR2	40	0	N
NAM_COLUMN_CDE	If the value in NAM_TABLE is not " " then this is the column name of the CODE value on the source table for the domain values. Otherwise, its " ". Examples, CDE_POS, CDE_CLM_TYPE.	VARCHAR2	40	0	N
NAM_COLUMN_DSC	If the value in NAM_SOURCE_DOMAIN_TBL is not " ", then this is the column name of the DESCRIPTION value on the source table for the domain values. Otherwise, it is " ". Example, DSC_CLM_TYPE, DSC_MODIFIER.	VARCHAR2	40	0	N

2.13.53 T_RU_RELATION

A relation is an ordered collection of named attributes, where each attribute is a subset of a domain or sub-relation. Database tables, views, and query result-sets are examples of a relation. So are our C structures, or a parameter list to a function. Our rule variables must be based on relations, since getter and setter functions of any programming language can have arbitrarily complex data. While domains hide their internal complexities, relations expose them. So an instance of a relation would be a row (or "tuple"). And an instance of a domain would be an encapsulated object whose internal data-structure we know nothing about. BTW, if CDE_DOMAIN is populated, this row is not really a relation, only a placeholder for a domain. Row examples: SAK DOMAIN MULTIVALUE 1 INT N 2 INT N 3 GND N 4 CT N 5 POS N 6 N 7 N 22 Y 23 MOD N 24 MODT N 25 OCC N

Column Name	Description	Type	Length	Precision	Primary Key
SAK_RELATION	The unique identifier for a row on this table.	NUMBER	9	0	Y
CDE_DOMAIN	This is an optional column. It is equal to " " if the relation has at least two attributes. Otherwise this "Relation" has only one attribute in which case it represents a domain to which it points.	CHAR	4	0	N
IND_MULTIVALUE	Set to "Y" if this variable holds not just one instance of the domain for the TRANSACTION VARIABLE, but an ordered collection of domain instances. Else set to "N". Although the relationship of this table is one to one with T_TRANSACTION_VARIABLE it must be split off from T_TRANSACTION_VARIABLE because of its complex structure.	CHAR	1	0	N

2.13.54 T_RU_RELATION_XREF

This is the common bill of materials structure (parts are made of parts that are made of parts and so on) that relate attributes. Each row represents an attribute of the "Relation" identified by SAK_RELATION. The attributes are identified by SAK_RELATION_CHILD. But SAK_RELATION_CHILD may itself be a Relation. This allows structures of any complexity. Row examples - refer to examples in T_RU_RELATION: SAK NUM_ORDER SAK_RELATION_CHILD 6 0 23 6 1 23 6 2 23 7 0 25 7 1 25 7 2 25 7 3 25 22 0 23 22 1 24 NOTE: Here is what some of the variables would look like in the Claim engine. int get_Age() { . return .; } List *get_Modifiers() { List *list = newList(); list_push(list, mod1); list_push(list, mod2); list_push(list, mod3); list_push(list, mod4); return list; } void set_ModType(List *tuples) { List *tuple; foreach (tuple, tuples) { char *modifierCode = list_item(tuple, 0); char *modifierType = list_item(tuple, 1); . } }

Column Name	Description	Type	Length	Precision	Primary Key
SAK_RELATION	This value groups the attributes of the relation identified by SAK_RELATION.	NUMBER	9	0	Y
SAK_RELATION_CHILD	This identifies each attribute of the RELATION to which SAK_RELATION for a row on this table. this FK points to either a DOMAIN or to another RELATION with more than one attribute.	NUMBER	9	0	Y

Column Name	Description	Type	Length	Precision	Primary Key
NUM_ORDER	This represents the attribute position within the relation..	NUMBER	4	0	N

2.13.55 T_RU_RULE

A rule is made up of a condition tree (that is, logical ANDs, ORs and comparisons) and a collection of actions. If the condition matches a transaction, all the actions are executed for that transaction. Rules also belong to a particular decision and for quick access decisions must mark which variables are required in their rules. All required variables are in the left sub-condition of the root condition, which must be a logical AND. SAK_RULE CDE_DECISION SAK_CONDITION CDE_RULE_EXCEPTION 1 BP 1 -1 2 BP 2 -1 Note: The 1st rule is "If the procedure is 99201, and date-of-service is between 9/24/2001 and 1/1/2005, and the member's age is between 0 and 45; then the procedure is covered". The 2nd rule's condition is "If the group is Anesthesia, and date-of-service is between 7/28/1999 and 12/31/2299, and the claim type is not A or C, then PA is required".

Column Name	Description	Type	Length	Precision	Primary Key
SAK_RULE	Primary-Key for this table	NUMBER	9	0	Y
CDE_DECISION	Primary-Key. A short identifier for the decision. It must be unique within a transaction. Examples, BP (Member Benefit Plan - Procedure), BD (Member Benefit Plan - Diagnosis).	CHAR	4	0	N
SAK_RULE_EXCEPTION	FK to T_RU_RULE. Optional. If it's greater than -1, then it points to a rule that is excluded from the decision making process. This is used to modify or override a rule inherited form a group higher in the classification.	NUMBER	9	0	N
SAK_CONDITION	It it's -1, there's no condition which means the rule never matches and cannot be applied. This is used when excluding a parent rule.	NUMBER	9	0	N

2.13.56 T_RU_RULE_ACTION

A rule can have multiple actions, and are executed when the rule matches a transaction. SAK_RULE SAK_ACTION SAK_CONDITION 1 27 -1

Column Name	Description	Type	Length	Precision	Primary Key
SAK_RULE	Primary-Key for this table	NUMBER	9	0	Y
SAK_ACTION	This is the primary key for this table.	NUMBER	9	0	Y

2.13.57 T_RU_TRANSACTION

Transaction processing is pairs of requests from and responses to an outside entity. The request is the input-data of a transaction, and the response is the output-data. Both the request

and response are structured as a collection of variables. One example transaction in MMIS is "Claim Submission".

Column Name	Description	Type	Length	Precision	Primary Key
CDE_TRANSACTION	This is the unique identifier for this table. Example, CLMS (Claim Submission).	CHAR	4	0	Y
DSC_TRANSACTION	This is a descriptive name for the transaction. Example, Claim submission.	VARCHAR2	4000	0	N

2.13.58 T_RU_TUPLE_SET

A tuple-set value is a collection of rows on this table, all with the same SAK_VALUE. Each row is a requirement of the tuple-set, and all the requirements must match the variable in the comparison, so in that sense they are like logical-ANDs. However, since each requirement may match only part of the variable, the part that matches cannot be reused by another requirement. The whole comparison is true if all minimums of each requirement are met, and the whole variable is been matched up to requirements.

Column Name	Description	Type	Length	Precision	Primary Key
SAK_VALUE	Primary key.	NUMBER	9	0	Y
SAK_VALUE_DISCRETE	Primary Key. IMPORTANT: There can be multiple rows with the same PK. Each row is another value in the set.	NUMBER	9	0	Y
NUM_MIN_AXIS	The minimum number of items in the variable array that this requirement must match.	NUMBER	9	0	N
NUM_MAX_AXIS	The MAXIMUM number of items in the variable array that this requirement must match.	NUMBER	9	0	N

2.13.59 T_RU_VARIABLE

A transaction is a collection of variables. Each variable has a name, which is unique within the transaction. And each variable has a domain data-type. For example, a claim has variable called "Date of Service" whose domain is "Date". Variables can be derived; like Member's Age which is based on the member ID from the input-transaction, then is used to look-up the member's date of birth on-file, and then subtracted from the current date to get age. And while a transaction request is essentially a collection of input variables, the transaction response is also a collection output (or "action") variables. Row examples: SAK CDE DSC I SAK_RELATION NAM_FUNCTION NAM_LIBRARY 1 AGE Age I 1 get_Age libclmcomm.so 1 MOD Modifier I 6 get_Modifiers libclmcomm.so 1 PROC ProcedureI 14 get_Procedure libclmcomm.so 1 EDIT Set Edit O 18 set_ClaimEdit libclmcomm.so 1 PA PA RequiredO 19 set_PARequired libclmcomm.so 1 RATE Rate TypeO 21 set_RateType libclmcomm.so 1 MODT Modifier TypeO 22 set_ModType libclmcomm.so

Column Name	Description	Type	Length	Precision	Primary Key
CDE_VARIABLE	Primary key for this table. The short identifier for the variable. It must be unique for the transaction that owns it.	CHAR	4	0	Y
CDE_TRANSACTION	This is the unique identifier for this table. Example, CLMS (Claim Submission).	CHAR	4	0	N
SAK_RELATION	The unique identifier for a row on this table.	NUMBER	9	0	N
DSC_VARIABLE	A descriptive name of the variable. This may be used in the UI to label the variable.	VARCHAR2	40	0	N
CDE_INPUT_OUTPUT	"I" for input, "O" for output. Input variables can be used in a rule's condition. Output variables can be used in rule's action.	CHAR	1	0	N
NAM_FUNCTION	If this variable is input-data on the transaction request, this C function should be a getter that takes no arguments and returns a value of the domain type. If the variable is output action, the function should be a setter which takes one argument of the domain type and returns nothing.	VARCHAR2	50	0	N

Column Name	Description	Type	Length	Precision	Primary Key
NAM_LIBRARY	The library that contains the function identified in NAM_TRAN_FUNCTION.	VARCHAR2	50	0	N

2.13.60 T_TYPE_OF_BILL

This table contains the valid types of bills and their description.

Column Name	Description	Type	Length	Precision	Primary Key
CDE_TYPE_OF_BILL	Code which indicates a specific type of facility.	CHAR	3	0	Y
DSC_TYPE_OF_BILL	Describes the specific type of facility.	CHAR	20	0	N
IND_PROC_IN_EXCLUD	Code which indicates whether the associated list of procedure codes is valid or invalid for the type of bill.	CHAR	1	0	N

Claims Detailed System Design

Kentucky MMIS Project

Cabinet for Health and Family Services
Department for Medicaid Services

March 7, 2008

Cabinet for Health and Family Services Department for Medicaid Services	
Role:	Name:
Author	Amy Sanders
Reviewer	Patti George
EDS Management	Glenn Jennings, Account Manager
Client	Commissioner Elizabeth A Johnson Commissioner Carrie Banahan Claims Division Director Rhonda Poston Executive Director/CIO Lorna S. Jones Chief Technical Officer Sandeep Kapoor
DELIVERABLE TITLE: Claims Detailed System Design	DATE SUBMITTED: March 7, 2008
FILE NAME: KY_MMIS_Claims_DSD_REFRESH_v5.1_FINAL_(03-07-2008).doc	AUTHORING TOOL: MSWord2003

Document Change Log

Version	Changed Date	Changed By	Reason
1.0	09/07/2007	Jennifer Brown, Bob Carter, Howard Warren	Creation of new document.
2.0	9/16/2007	Jennifer Brown, Bob Carter, Howard Warren	Revisions made according to internal review comments.
3.0	11/21/2007	Jennifer Brown, Bob Carter, Howard Warren	Revisions made according to Commonwealth comments.
4.0	11/27/2007	Jennifer Brown, Bob Carter, Howard Warren	Updated with the latest information - REFRESH.
5.0	12/11/2007	Jennifer Brown, Bob Carter, Howard Warren	Global walkthrough updates
5.1	1/28/2008 1/31/2008	Cathy Hill	Made revisions for DHS per Patti George

Table of Contents

1 Introduction 1

2 Detailed System Design 2

2.1 System Flow 2
2.1.1 interChange System Architecture 2
2.1.2 Claims System Flow 3
2.2 Subsystem Description 9
2.2.1 Introduction to Claims 9
2.2.2 Claims Entry 10
2.2.3 Claims Receipt and Control 14
2.2.4 Edit and Audit 17
2.2.5 Claims Pricing 20
2.2.6 Claims Resolution 22
2.2.7 Point of Sale (POS)/Prospective Drug Utilization Review (ProDUR) 23
2.2.8 Claims Disposition 27
2.2.9 Claims Reporting 28
2.2.10 Claim Check 29
2.3 Claims Processes 30
2.3.1 Claims and Encounters 30
2.3.2 Claims Processing 31
2.3.3 Receive Claim 34
2.3.4 Validate Claim 36
2.3.5 Process Claim Payment 40
2.3.6 Encounters Processing 42
2.3.7 Receive Encounter 43
2.3.8 Validate Encounter 45
2.3.9 Return Encounter Errors 49
2.4 Data Model 50
2.4.1 External System Interfaces 50
2.5 Job Scripts 51
2.5.1 CLMJD011 - Clerk ID Recycle Claims Report 51
2.5.2 CLMJD012 - EOB Listing Report 52
2.5.3 CLMJD015 - ClaimCheck Potential History To Adjust Report 53
2.5.4 CLMJD018 - Claims Processing Daily Summary Reports 54
2.5.5 CLMJD020 - Aged Detail Suspense Report 55
2.5.6 CLMJD021 - Suspense File Analysis By Error Code Report 56
2.5.7 CLMJD023 - Claims Error Report (FFS) 57
2.5.8 CLMJD027 - Transaction Register Report 58
2.5.9 CLMJD051 - Aged Detail Adjustment Report 59
2.5.10 CLMJD054 - Error Recycle Parameters Report 60
2.5.11 CLMJD060 - Suspense File Analysis by Claim Type Report 61
2.5.12 CLMJD065 - Aged Suspended Claims By Location Report 62
2.5.13 CLMJD075 - Daily upload and update of attachment for electronic claims 63
2.5.14 CLMJD099 - Captiva Received Rejected 65
2.5.15 CLMJD1000 - Allows clmd1000 to be run 66
2.5.16 CLMJD10D - On-Line Daily Adjustment Report 69
2.5.17 CLMJD110 - Online Mass Adjustment 70
2.5.18 CLMJD123 - Claims Error Report (Encounters) 71
2.5.19 CLMJD130 - Claims Submission Statistics - Daily Report 72
2.5.20 CLMJD132 - Encounters Submission Statistics - Daily 73
2.5.21 CLMJD140 - Encounter Batch Summary 74
2.5.22 CLMJD210 - Claims for Provider Mass Rate Adjustments 75
2.5.23 CLMJD211 - Claims for Provider Mass Rate Adjustments 76

2.5.24 CLMJD25D - Services To Be Clerically Denied Report 77
2.5.25 CLMJD26D - Claims Paid Due To Force Report 78
2.5.26 CLMJD300 - Daemon process to kick off Sendclms 79
2.5.27 CLMJD350 - Move claims from NEWDAY to the input directory 80
2.5.28 CLMJD351 - Start the claim client program 81
2.5.29 CLMJD422 - Combined Claim Error Report 82
2.5.30 CLMJD500 - Top Ten Claims Approved For Payment 83
2.5.31 CLMJD956 - Void Adjustments 84
2.5.32 CLMJDA01 - Create audit driver tables 85
2.5.33 CLMJDA99 - Remove adjusted claims 86
2.5.34 CLMJDER1 - Daily Error Summary Extract 87
2.5.35 CLMJDER2 - Daily Error Summary By Claim Type Report 88
2.5.36 CLMJDFCST - Produce listing of all scheduled jobs for a given range of dates 90
2.5.37 CLMJDI10 - Encounters Threshold Errors Extracts 91
2.5.38 CLMJDI20 - This job creates and send an extact of Dental Claims Creates Dental Claim Extract For McKesson 92
2.5.39 CLMJDMADJ - Mass Adjustments 93
2.5.40 CLMJDM_PAY_VOID - Calls adjustment process for claims associated with Mass Adjustment requests 94
2.5.41 CLMJDM_PROV_RATE - Call the adjustment routine for claims Mass Adjustment requests.95
2.5.42 CLMJDNDY - Move previous day claims to re-run 96
2.5.43 CLMJDR01 - Copy the recycled claims to newday 97
2.5.44 CLMJDR02 - Batch job to recycle suspended claims 98
2.5.45 CLMJM031 - Suspense File Analysis By Claim Type Report 99
2.5.46 CLMJM046 - Potential Narcotic Drug Abuse Report 100
2.5.47 CLMJM050 - Retro Active Rate Adjustment Report 101
2.5.48 CLMJM130 - Claims Submission Statistics - Monthly Report 102
2.5.49 CLMJM132 - Encounters Submission Statistics - Monthly 103
2.5.50 CLMJMA00 - Adjustment Request Update 104
2.5.51 CLMJMACR - Mass Adjustment Claim Select 105
2.5.52 CLMJO132 - Encounters Submission Statistics - On Request 106
2.5.53 CLMJQ058 - ClaimCheck Error Report 107
2.5.54 CLMJQ991 - Abortions Procedure Report 108
2.5.55 CLMJSTOP - Shut Down the Claims Engine. 109
2.5.56 CLMJSTRT - Starts the Claims Engine 110
2.5.57 CLMJW064 - Suspense Claims by Claim Type / Region Report 111
2.5.58 CLMJW130 - Claims Submission Statistics - Weekly Report 112
2.5.59 CLMJW860 - Pended Claims Aging 113
2.5.60 CLRJD001 - Update daily claims report parameters 114
2.5.61 CRLJD001 - Creates Claim Link/Unlink Adjustments 115
2.5.62 ENCJDNDY - Move previous day claims to rerun - Encounters 116
2.5.63 SYSJDRDIST - Rdist the exe's to CORE 117
2.6 Programs 118
2.6.1 ClaimDrvr.c -- Main Claim Driver Module 118
2.6.2 PosSrvrMgr.sh -- This script will start, stop the Socket Listener based on the input parameter 119
2.6.3 RulesEngine -- Claims Rules Processor 120
2.6.4 Soap -- Soap 121
2.6.5 adj837 -- Adjust 837 Claim And 837 Voids(Reversals) 122
2.6.6 adjComm -- Adjustment Functions 124
2.6.7 adjDelete -- Adjustment Delete Functions 126
2.6.8 adjHistOnly -- Adjustment TPL History Only (adjHistOnly) 128
2.6.9 adjInit -- Adjustment Environment Initialization 130
2.6.10 adjMain -- Adjustments Processing 132
2.6.11 adjQuery -- Adjustment Query Functions 134

2.6.12 adjRecpLink -- Adjustment Recipient Link Functions 136
2.6.13 adjRecycle -- Adjustment Recycle Extract Functions 138
2.6.14 adjRegular -- Adjustment Regular Functions 140
2.6.15 adjVoid -- Void And Reversal Functions 142
2.6.16 aimGlobal -- Utility Functions 144
2.6.17 aim_pswd -- Get AIM_PSWD 146
2.6.18 claimList.c -- claimList 147
2.6.19 claimList.h -- claimList Header 151
2.6.20 claimList.xsd -- claimList Schema 152
2.6.21 claimcheck.csh -- ClaimCheck Runtime Environment Variable Set Script 153
2.6.22 clm0010d -- Online Adjustment Daily Report 154
2.6.23 clm0011d -- Clerk ID Recycle Claims Report 157
2.6.24 clm0012d -- EOB Listing 160
2.6.25 clm0015d_cc -- Claim Check Potential History To Adjust 162
2.6.26 clm0018d -- Claims Processing Daily Summary 165
2.6.27 clm0020d -- Aged Detail Suspense Report 168
2.6.28 clm0021d -- Suspense File Analysis By Error Code 171
2.6.29 clm0023d -- Claims Error Report (FFS) 174
2.6.30 clm0025d -- Services To Be Clerically Denied Report 177
2.6.31 clm0026d -- Claims Paid Due To Force Report 180
2.6.32 clm0027d -- Transaction Register 183
2.6.33 clm0031m -- Suspense File Analysis By Claim Type 186
2.6.34 clm0050m -- Retro Active Rate Adjustment Report 188
2.6.35 clm0051d -- Aged Detail Adjustment Report 190
2.6.36 clm0054d -- Error Recycle Parameters 193
2.6.37 clm0058q -- ClaimCheck Error 196
2.6.38 clm0060d -- Suspense File Analysis By Claim Type 199
2.6.39 clm0065d -- Aged Suspended Claims By Location Report 201
2.6.40 clm0099d -- Captiva Received Rejected 203
2.6.41 clm0110d -- Online Mass Adjustment Report 205
2.6.42 clm0123d -- Claims Error Report (Encounters) 207
2.6.43 clm0130 -- Claims Submission Statistics 210
2.6.44 clm0132 -- Encounters Submission Statistics 214
2.6.45 clm0140d -- Encounter Batch Summary 217
2.6.46 clm140d -- Encounters Batch Summary 219
2.6.47 clm15del -- ClmChk PHTAR Delete Data 220
2.6.48 clm5000d -- Top Ten Claims Approved For Payment 222
2.6.49 clm58del -- ClaimCheck Delete Savings Data 224
2.6.50 clm8600w -- Pended Claims Aging Report 226
2.6.51 clm9914q -- Abortions Procedure Report 228
2.6.52 clmBenefitPlanGroup.sc -- Reads Table T_BENEFIT_PLAN_GROUP 231
2.6.53 clmPosGroup -- Reads Table T_PLACE_OF_SERVICE_GROUP 233
2.6.54 clm_manual_mass_adj -- Manual Mass Adjustment Request 234
2.6.55 clmdaily -- Claim Daily Report Jobs 236
2.6.56 clmp046m -- Potential Schedule 2 Narcotic Drug Abuse 238
2.6.57 clmp101s.sc -- Auto Adjustment Selection 240
2.6.58 clmp210d -- Claims For Provider Mass Rate Adjustments 242
2.6.59 clmp422 -- Combined Claim Error 245
2.6.60 clmp956d -- Void Adjustments 249
2.6.61 clmpd075 -- Attachment Indicator Update 251
2.6.62 clmpda01 -- Remove Adjusted Claims 253
2.6.63 clmpdi10 -- Encounters Threshold Errors Extract 255
2.6.64 clmpdi15 -- Encounters Errors 277 Formatter 258
2.6.65 clmpdi20 -- McKesson ClaimCheck Dental Extract 260
2.6.66 clmpdr01 -- Create claims XML for input to claims engine 262
2.6.67 clmpdr02 -- Batch Program To Recycle Suspended Claims 264

2.6.68 clmper01 -- Daily Error Summary By Claim Type 266
2.6.69 clmpma00 -- Update Adjustment Request Status 269
2.6.70 clmpma03 -- Void Adjustments 271
2.6.71 clmpma_mass_adj -- Call the adjustment routine for claims mass adjustment requests273
2.6.72 clmpmacr -- Mass Adjustment Claim Select 275
2.6.73 clmpsplt -- FFS Claims Splitter 278
2.6.74 clmpw064 -- Suspended Claims By Claim Type / Region 280
2.6.75 clmsAssignment -- Managed Care Claim Related Edits 282
2.6.76 clmsAudit -- General Claims Auditing 284
2.6.77 clmsAutoPa -- Pharmacy Auto PA Module For DUR+ 289
2.6.78 clmsBenefitPlanGroup -- Reads Table T_BENEFIT_PLAN_GROUP 291
2.6.79 clmsDiag -- Claim Diagnosis 293
2.6.80 clmsDiagGroup -- Claims Diagnosis Groups 295
2.6.81 clmsDisp -- Claim Disposition 297
2.6.82 clmsDtl -- General Claim Detail 299
2.6.83 clmsHdr -- General Claim Header 302
2.6.84 clmsIcd9Group -- Claims ICD9 Procedure Groups 304
2.6.85 clmsIcnAssign -- ICN Assignment 306
2.6.86 clmsLoadHist -- Load Member Claim History 307
2.6.87 clmsMisc -- Miscellaneous General Claims 309
2.6.88 clmsModGroup -- Modifier Group 311
2.6.89 clmsMultiPayer -- Claim MultiPayer COB Pricing 313
2.6.90 clmsPRTypeSpecGroup.sc --
Claims Provider Type and Specialty Group Processing 315
2.6.91 clmsPaPauth -- General Claim PA 317
2.6.92 clmsPrProvBilling -- Claims Billing Provider 320
2.6.93 clmsPricing -- Claims Pricing 322
2.6.94 clmsPricing_st -- KY Specific Claims Pricing 331
2.6.95 clmsProcGroup -- Claims Procedure Groups 333
2.6.96 clmsProgram -- Process claims by member benefit plan 335
2.6.97 clmsPurePricing -- clmPurePricing 337
2.6.98 clmsReBase -- Claims Member 339
2.6.99 clmsRevenueGroup -- Claims Revenue Groups 341
2.6.100 clmsRulesCaller -- Claims Rules Interface 343
2.6.101 clmsRulesWrapper -- Rules Engine Setter/Getter Interface 344
2.6.102 clmsSpendDown -- SpendDown 346
2.6.103 clmsTpl -- Claims TPL 349
2.6.104 clmsTplBillings -- Get TPL Info 351
2.6.105 clmsWrite -- Claims Write 353
2.6.106 clmsXover -- Crossover Edits 356
2.6.107 clmsakicn -- Claim Sak And ICN Assignment 358
2.6.108 clmsccInt -- ClaimCheck Integration Program 360
2.6.109 clmsdntl -- Dental Main Driver 363
2.6.110 clmsdntlAudit -- Dental Auditing 365
2.6.111 clmsdntlDtl -- Dental Detail 367
2.6.112 clmsdntlHdr -- Dental Header 369
2.6.113 clmsdntlPricing -- Dental Pricing 371
2.6.114 clmsdntlWrite -- Dental Database Update 373
2.6.115 clmspars -- Parsing Engine 376
2.6.116 clmsphrm -- Pharmacy Main Driver 377
2.6.117 clmsphrmAudit -- Pharmacy Audits 379
2.6.118 clmsphrmCob -- Pharmacy COB 381
2.6.119 clmsphrmDUR -- Pharmacy ProDUR 382
2.6.120 clmsphrmDrug -- Pharmacy Drug 384
2.6.121 clmsphrmDtl -- Pharmacy Detail 386
2.6.122 clmsphrmHdr -- Phamacy Header 388

2.6.123 clmsphrmPricing -- Pharmacy Pricing 390
2.6.124 clmsphrmReversal -- Pharmacy Reversal 392
2.6.125 clmsphrmWrite -- Pharmacy Database Update 394
2.6.126 clmsphys -- Physician Main Driver 396
2.6.127 clmsphysAudit -- Physician Auditing 399
2.6.128 clmsphysDiag -- Physician Diagnosis 401
2.6.129 clmsphysDtl -- Physician Detail 403
2.6.130 clmsphysDtl_st -- KY Physician Detail 405
2.6.131 clmsphysHdr -- Physician Header 407
2.6.132 clmsphysPricing -- Physician Pricing 409
2.6.133 clmsphysPricing_st -- KY PHYS Pricing Module 412
2.6.134 clmsphysProc -- Physician Procedure 414
2.6.135 clmsphysProvEdit -- Physician Provider Edits 416
2.6.136 clmsphysRecipEdit -- Physician Recipient Edits 417
2.6.137 clmsphysWrite -- Physician Database Update 418
2.6.138 clmsphysWrite_st -- Physician database update for state specific data elements 420
2.6.139 clmstats -- Claim Statistics 421
2.6.140 clmsub92 -- UB92 Main Driver 422
2.6.141 clmsub92Audit -- UB92 Auditing 425
2.6.142 clmsub92CondX -- UB92 Condition Codes 427
2.6.143 clmsub92DRGPricing -- UB92 DRG Pricing 428
2.6.144 clmsub92DiagX -- UB92 Diagnosis 430
2.6.145 clmsub92Dtl -- UB92 Detail 432
2.6.146 clmsub92HHPricing -- UB92 Home Health Pricing 434
2.6.147 clmsub92Hdr -- UB92 Header 436
2.6.148 clmsub92IPPricing -- UB92 Inpatient Pricing 439
2.6.149 clmsub92Icd9X -- UB92 ICD9 Procedures 441
2.6.150 clmsub92LTCPricing -- UB92 Long Term Care Pricing 443
2.6.151 clmsub92OPPricing -- UB92 Outpatient Pricing 445
2.6.152 clmsub92Occur -- UB92 Occurrence Codes 447
2.6.153 clmsub92Payer -- UB92 Payer Information 449
2.6.154 clmsub92Pricing -- UB92 Pricing 451
2.6.155 clmsub92Pricing_st -- KY UB92 Pricing Module 454
2.6.156 clmsub92Proc -- UB92 Procedures 456
2.6.157 clmsub92TobGroup -- UB92 Type Of Bill Groups 458
2.6.158 clmsub92Value -- UB92 Value Codes 460
2.6.159 clmsub92Write -- UB92 Database Update 462
2.6.160 crlpd001 -- Creates Claim Link/Unlink Adjustments 464
2.6.161 dummyClaimCheck -- dummyClaimCheck 466
2.6.162 encsplit -- Encounter Claims Splitter 467
2.6.163 fcst.sh -- Produce list of scheduled jobs 469
2.6.164 libadjcomm.so -- Adjustment Common Library 470
2.6.165 libadjxml.so -- Common Adjustment Functions 471
2.6.166 libclmbtch.so -- Tuxedo Interface 472
2.6.167 libclmcomm.so -- Claim Engine Common Library 473
2.6.168 libclmcomm_st.so -- KY Claim Engine Common Library 482
2.6.169 libclmdntl.so -- Dental Claim Library 484
2.6.170 libclmdrvr.so -- Claim Driver 487
2.6.171 libclmphrm.so -- Pharmacy Claim Library 488
2.6.172 libclmphys.so -- Physician Claim Library 490
2.6.173 libclmphys_st.so -- KY PHYS Claim Library 494
2.6.174 libclmsys.so -- Database Connection Utility 495
2.6.175 libclmub92.so -- UB92 Claim Library 496
2.6.176 libclmub92_st.so -- KY UB92 Claim Library 501
2.6.177 libclmwrite.so -- Claims Engine Database Updater 503
2.6.178 libclmwrite_st.so -- Claims engine database update - for state specific tables 507

2.6.179 libsoapClaim.so -- Soap Claim Shared Library 508
2.6.180 list -- Linked List Functions 509
2.6.181 log -- Error Logging 510
2.6.182 mailfcst.sh -- Mail forecast reports to cycle monitor team 511
2.6.183 rdist.customer.ksh -- rdist.customer.ksh 512
2.6.184 sendclmd -- Reads a configuration file to start or shutdown claim engine 513
2.6.185 sendclms -- Claim Driver 515
2.6.186 soapClaim -- soapClaim 517
2.6.187 tools -- Common Functions 518
2.6.188 value -- Dynamic Type-Casting 519
2.6.189 webdrvr -- Web Driver 520
2.6.190 xmlsmrge -- SQL Unload And Merge 522
2.7 Internal Web Pages 523
2.7.1 PAGE: Claims Adjustment Information Page 523
2.7.2 PANEL: Claim Adjustment Navigation 525
2.7.3 PANEL: Adjustment Request 527
2.7.4 PANEL: Net Verification 535
2.7.5 PANEL: Adjustment Mini-Search 540
2.7.6 PANEL: Adjustment Information 542
2.7.7 PAGE: Claims Adjustment Search Page 545
2.7.8 PANEL: Adjustment Search 547
2.7.9 PANEL: Adjustment Search Results 550
2.7.10 PAGE: Claims Assignment Page 553
2.7.11 PANEL: Scheduling Criteria 554
2.7.12 PANEL: QA Review Criteria 560
2.7.13 PANEL: Schedule Claims 563
2.7.14 PANEL: Claims Assignment 567
2.7.15 PANEL: Assignments Navigation 573
2.7.16 PAGE: Claims Data Correction Page 575
2.7.17 PANEL: Diagnosis 577
2.7.18 PANEL: EOB 580
2.7.19 PANEL: Occurrence 586
2.7.20 PANEL: Value 589
2.7.21 PANEL: Dental Claim Correction 592
2.7.22 PANEL: Physician Claim Detail 601
2.7.23 PANEL: Detail Error 610
2.7.24 PANEL: Detail Health Program 615
2.7.25 PANEL: Medicare Information - Physician 622
2.7.26 PANEL: Adjustment Information - Daughter or Mother 627
2.7.27 PANEL: Pharmacy Claim Detail Navigation 630
2.7.28 PANEL: Submitted Data - UB92 633
2.7.29 PANEL: CAS Inquiry 635
2.7.30 PANEL: Misc Information 638
2.7.31 PANEL: Check 642
2.7.32 PANEL: Submitted Data - Dental 645
2.7.33 PANEL: Pharmacy Data Correction (this panel is not currently used) 647
2.7.34 PANEL: Decision Rules 658
2.7.35 PANEL: Claim Member Coinsurance 661
2.7.36 PANEL: Attachment 663
2.7.37 PANEL: Detail EOB 666
2.7.38 PANEL: Physician Claim Navigation 670
2.7.39 PANEL: Dental Claim Navigation 674
2.7.40 PANEL: Additional Claim Information 678
2.7.41 PANEL: Location 681
2.7.42 PANEL: Pharmacy Claim Detail 684
2.7.43 PANEL: Display TCN 689

2.7.44 PANEL: ICD-9-CM 691
2.7.45 PANEL: Related History 694
2.7.46 PANEL: Detail Quadrant Cavity 697
2.7.47 PANEL: Dental Claim Detail 700
2.7.48 PANEL: UB92 Claim Navigation 708
2.7.49 PANEL: Prior Authorization 712
2.7.50 PANEL: Dental Claim Detail Navigation 715
2.7.51 PANEL: Audit History Panel 717
2.7.52 PANEL: Claim Member Copay 721
2.7.53 PANEL: Condition 723
2.7.54 PANEL: Health Program 725
2.7.55 PANEL: DRG 731
2.7.56 PANEL: Expecting Date 734
2.7.57 PANEL: Physician Claim Detail Navigation 736
2.7.58 PANEL: Detail Related History 738
2.7.59 PANEL: Pharmacy Claim Navigation 741
2.7.60 PANEL: Detail Submitted Data - Dental 744
2.7.61 PANEL: Detail Submitted Data - UB92 747
2.7.62 PANEL: MCO Data 751
2.7.63 PANEL: Claims Suspense List 755
2.7.64 PANEL: UB92 Claim Detail 757
2.7.65 PANEL: UB92 Claim Correction 764
2.7.66 PANEL: Physician Claim Correction 776
2.7.67 PANEL: Medicare Information - UB92 787
2.7.68 PANEL: Cash Disposition 791
2.7.69 PANEL: Error 794
2.7.70 PANEL: Payer 801
2.7.71 PANEL: Submitted Data - Physician 804
2.7.72 PANEL: Data Correction Note 806
2.7.73 PANEL: Detail Submitted Data - Physician 810
2.7.74 PANEL: UB92 Claim Detail Navigation 814
2.7.75 PAGE: Claims Encounter Page 817
2.7.76 PANEL Encounter Data Navigation 818
2.7.77 PANEL: Encounter Batch Summary Search Results 820
2.7.78 PANEL: Encounter Errors 824
2.7.79 PANEL: EDI Encounter Batch Status Search 827
2.7.80 PANEL: Encounter Batch Summary Search 829
2.7.81 PANEL: EDI Encounter Batch Status Search Results 832
2.7.82 PANEL: Encounter Claims Related Data 834
2.7.83 PAGE: Claims Information Page 836
2.7.84 PANEL: Diagnosis 838
2.7.85 PANEL: EOB 840
2.7.86 PANEL: Value 844
2.7.87 PANEL: Physician Claim Detail 846
2.7.88 PANEL: Detail Error 855
2.7.89 PANEL: Detail Health Program 859
2.7.90 PANEL: Medicare Information - Physician 865
2.7.91 PANEL: Adjustment Information - Daughter or Mother 870
2.7.92 PANEL: Pharmacy Claim Detail Navigation 872
2.7.93 PANEL: Submitted Data - UB92 875
2.7.94 PANEL: CAS Inquiry 877
2.7.95 PANEL: Misc Information 880
2.7.96 PANEL: Check 883
2.7.97 PANEL: Submitted Data - Dental 886
2.7.98 PANEL: Decision Rules 888
2.7.99 PANEL: Claim Member Coinsurance 891

2.7.100 PANEL: Attachment ... 893
2.7.101 PANEL: Detail EOB ... 895
2.7.102 PANEL: Physician Claim Navigation ... 899
2.7.103 PANEL: Dental Claim Navigation ... 903
2.7.104 PANEL: NCPDP Reject Codes ... 907
2.7.105 PANEL: Additional Claim Information ... 909
2.7.106 PANEL: Location ... 912
2.7.107 PANEL: Pharmacy Claim Detail ... 915
2.7.108 PANEL: Display TCN ... 919
2.7.109 PANEL: ICD-9-CM ... 921
2.7.110 PANEL: Related History ... 924
2.7.111 PANEL: Detail Quadrant Cavity ... 926
2.7.112 PANEL: Dental Claim Detail ... 928
2.7.113 PANEL: UB92 Claim Navigation ... 936
2.7.114 PANEL: Prior Authorization ... 940
2.7.115 PANEL: Dental Claim Detail Navigation ... 943
2.7.116 PANEL: Audit History Panel ... 946
2.7.117 PANEL: Member Copay ... 950
2.7.118 PANEL: Condition ... 953
2.7.119 PANEL: Health Program ... 955
2.7.120 PANEL: Dental Claim ... 961
2.7.121 PANEL: DRG ... 969
2.7.122 PANEL: Expecting Date ... 972
2.7.123 PANEL: Physician Claim Detail Navigation ... 974
2.7.124 PANEL: Detail Related History ... 977
2.7.125 PANEL: Pharmacy Claim Navigation ... 980
2.7.126 PANEL: Detail Submitted Data - Dental ... 983
2.7.127 PANEL: Detail Submitted Data - UB92 ... 986
2.7.128 PANEL: MCO Data ... 990
2.7.129 PANEL: Physician Claim ... 994
2.7.130 PANEL: Pharmacy Claim ... 1003
2.7.131 PANEL: UB92 Claim Detail ... 1012
2.7.132 PANEL: Medicare Information - UB92 ... 1019
2.7.133 PANEL: Cash Disposition ... 1023
2.7.134 PANEL: Error ... 1026
2.7.135 PANEL: Payer ... 1033
2.7.136 PANEL: Submitted Data - Physician ... 1036
2.7.137 PANEL: Claims Mini Search ... 1038
2.7.138 PANEL: UB92 Claim ... 1040
2.7.139 PANEL: Detail Submitted Data - Physician ... 1050
2.7.140 PANEL: UB92 Claim Detail Navigation ... 1054
2.7.141 PAGE: Claims Mass Adjustment Information Page ... 1057
2.7.142 PANEL: Mass Claims Adjustment Navigation ... 1059
2.7.143 PANEL: Mass Adjustment Information ... 1061
2.7.144 PANEL: Mass Adjustment Entry ... 1064
2.7.145 PANEL: Mass Net Verification ... 1069
2.7.146 PANEL: Mass Adjustment Criteria ... 1075
2.7.147 PANEL: Adjustment Mini-Search ... 1084
2.7.148 PAGE: Claims Related Data Page ... 1086
2.7.149 PANEL:Edit Recycle ... 1087
2.7.150 PANEL: Related Data -Other ... 1091
2.7.151 PANEL: Encounter Threshold ... 1093
2.7.152 PAGE: Claims Search Page ... 1096
2.7.153 PANEL: Claims Search ... 1098
2.7.154 PANEL: Claims Search Results ... 1106
2.8 Reports ... 1111

2.8.1 CLM-0010-D -- Online Adjustment Daily Report 1111
2.8.2 CLM-0011-D -- Clerk ID Recycle Claims Report 1115
2.8.3 CLM-0012-D -- EOB Listing 1118
2.8.4 CLM-0015-D -- ClaimCheck Potential History to Adjust 1121
2.8.5 CLM-0016-D -- Claims Processing Daily Summary 1126
2.8.6 CLM-0017-D -- Original Claims Processing Daily Summary 1136
2.8.7 CLM-0018-D -- Adjustment Credit Claims Processing Daily Summary 1145
2.8.8 CLM-0020-D -- Aged Detail Suspense Report 1154
2.8.9 CLM-0021-D -- Suspense File Analysis by Error Code 1158
2.8.10 CLM-0022-D -- Combined Claim Error 1162
2.8.11 CLM-0023-D -- Claims Error Report FFS 1181
2.8.12 CLM-0024-D -- Daily Error Summary By Claim Type 1190
2.8.13 CLM-0025-D -- Services to be Clerically Denied Report 1196
2.8.14 CLM-0026-D -- Claims Paid Due To Force Report 1201
2.8.15 CLM-0027-D -- Transaction Register 1206
2.8.16 CLM-0031-M -- Suspense File Analysis of Current and Previous Fiscal Year 1212
2.8.17 CLM-0046-M -- Potential Schedule 2 Narcotic Drug Abuse 1217
2.8.18 CLM-0050-M -- Retro Active Rate Adjustment Report 1222
2.8.19 CLM-0051-D -- Aged Detail Adjustment Report 1229
2.8.20 CLM-0054-D -- Error Recycle Parameters 1232
2.8.21 CLM-0055-D -- Daily Error Summary By Claim Type Original Claims 1235
2.8.22 CLM-0056-D -- Daily Error Summary By Claim Type Adjustments 1241
2.8.23 CLM-0058-Q -- ClaimCheck Error Report 1247
2.8.24 CLM-0060-D -- Suspense File Analysis of Current Days 1252
2.8.25 CLM-0064-W -- Suspended Claims by Claim Type - Region 1263
2.8.26 CLM-0065-D -- Aged Suspended Claims By Location Report 1266
2.8.27 CLM-0099-D -- Captiva Received Rejected 1269
2.8.28 CLM-0110-D -- Online Mass Adjustment Report 1272
2.8.29 CLM-0123-D -- Claims Error Report Encounters 1276
2.8.30 CLM-0130-D -- Claims Submission Statistics - Daily 1282
2.8.31 CLM-0130-M -- Claims Submission Statistics - Monthly 1291
2.8.32 CLM-0130-W -- Claims Submission Statistics - Weekly 1300
2.8.33 CLM-0132-D -- Encounters Submission Statistics - Daily 1309
2.8.34 CLM-0132-M -- Encounters Submission Statistics - Monthly 1316
2.8.35 CLM-0132-O -- Encounters Submission Statistics - On Request 1323
2.8.36 CLM-0140-D -- Encounter Batch Summary 1330
2.8.37 CLM-5000-D -- Top Ten Claims Approved For Payment 1333
2.8.38 CLM-8600-W -- Pended Claims Aging 1339
2.8.39 CLM-9914-Q -- Abortions Procedure Report 1343
2.9 Letters 1347
2.10 Claims Internal and External Interfaces 1348
2.10.1 Entities 1348
2.10.2 Input Files 1348
2.10.3 Output Files 1349
2.10.4 Input-Output Files 1349
2.10.5 File Layouts 1349
2.10.6 Output McKesson ClaimCheck Dental File 1350
2.11 Requirement Matrix and Cross Reference 1360
2.12 Change Orders 1412
2.12.1 Display Encounter Data Elements - 159 1412
2.12.2 Allow individual claim deselect - 161 1414
2.12.3 Change "Recipient" - online - 162 1416
2.12.4 RPTS - Change "Recipient" - 163 1418
2.12.5 Claim Search- Changes - 195 1420
2.12.6 Claim Search Results - Changes - 196 1422
2.12.7 Encounters Threshold Errors - 197 1424

2.12.8 Enable image access - 242 1427
2.12.9 RPTS - Remove HCA from Heading - 429 1429
2.12.10 RPTS - CHG Exceptions/Audit-Edit - 565 1431
2.12.11 RPTS - Change EDI- to CLM- - 665 1433
2.12.12 RPTS - CLM0012D-CHG HDG to EOB - 785 1436
2.12.13 RPTS - CLM0022D-CHG Report Name - 1033 1438
2.12.14 RPTS - CLM0024D-CHG RESC Heading - 1065 1440
2.12.15 RPTS - Add Standard Footer - 1070 1442
2.12.16 RPTS- Remove Reports From System - 1074 1445
2.12.17 RPTS - CLM0064W- Add Heading - 1075 1447
2.12.18 RPTS - CLM0054D-Separate Report - 1082 1449
2.12.19 RPTS - Set Claim Age Parm = 31 - 1084 1451
2.12.20 RPTS-Encounter Statistic-CLM0132 - 1089 1453
2.12.21 RPTS-CLM0031M-CHG RPT Timeframe - 1091 1458
2.12.22 RPTS- CLM-0010-D Add MassAdjRqNr - 1109 1460
2.12.23 RPTS - CLM-5000-D Create Report - 1110 1461
2.12.24 CC - integration module - 1121 1463
2.12.25 RPTS - CLM-8600-W Create Report - 1122 1465
2.12.26 ccInt implement cc v8.5 changes - 1135 1467
2.12.27 ccInt process outpatient claims - 1136 1474
2.12.28 ccInt - modify history pull - 1137 1476
2.12.29 RPTS - CLM-9914-Q Create Report - 1141 1479
2.12.30 Mass Adjustment Criteria ranges - 1228 1482
2.12.31 CORE 10358Add billed/paid MADJ - 1229 1484
2.12.32 Adj Rqst panel add reason desc - 1230 1486
2.12.33 Adj rqst panel require prov id - 1239 1487
2.12.34 Mass Net Verf add totals - 1240 1489
2.12.35 Related history - add error code - 1241 1491
2.12.36 CORE 10258 MSS_VD_RTR_MCaRe CLMS - 1243 1493
2.12.37 Adjustment - lock criteria - 1253 1495
2.12.38 CORE 10280 MFY MSSADJ CLM SELCTN - 1262 1497
2.12.39 DB Model - Encounters - 1268 1499
2.12.40 DB Model - FFS - 1270 1501
2.12.41 Add/Modify FFS Data Elements - 1271 1503
2.12.42 Configure Edit Criteria - 1272 1505
2.12.43 RPTS - Exclude Encounter Claims - 1282 1506
2.12.44 Add Encounter Page - 1283 1511
2.12.45 Add Encounter Batch Summary - 1284 1512
2.12.46 Add Encounter Error Panel - 1285 1513
2.12.47 Add Encounter Claim Related Data - 1286 1514
2.12.48 Add Encounter EDI Batch Search - 1287 1516
2.12.49 Add CAS Inquiry Panel - 1288 1518
2.12.50 Add NCPDP Reject Codes - 1289 1520
2.12.51 Configure Limit Audits - 1323 1521
2.12.52 Configure contraindicated audits - 1324 1523
2.12.53 Configure conflict audits - 1325 1525
2.12.54 Duplicate audit criteria - 1326 1527
2.12.55 Audit criteria - other - 1327 1530
2.12.56 PA Logic Changes - 1328 1532
2.12.57 Encounters - Splitter - 1329 1534
2.12.58 Change Cross Over Panels - 1347 1536
2.12.59 Verify mass rate adjustments - 1358 1539
2.12.60 CORE 10281Mass Rate Verification - 1359 1541
2.12.61 Claims Pricing - DRGs - 1362 1543
2.12.62 Add Display Of Patient Liability - 1375 1544
2.12.63 Add Fields Display - Panels - 1376 1546

2.12.64 Change Fields - Panels - 1390 1548
2.12.65 Health Program Panels - 1391 1550
2.12.66 KY Prior Auth Pricing - 1416 1552
2.12.67 DRG Pricing - 1417 1554
2.12.68 Patient Liability - 1418 1556
2.12.69 Core 8814 Benefit Adj Factor 5 - 1474 1559
2.12.70 UI for Claim Rules Matched - 1475 1560
2.12.71 Core 8816 Benefit Adj Factor 7 - 1476 1562
2.12.72 Core 8938 Expand Cnty Cde Claims - 1579 1565
2.12.73 Core 8990 Pricing Ind Expansion - 1645 1567
2.12.74 Core 8991 Pricing Ind Expansion - 1646 1569
2.12.75 Core 8993 Pricing Ind Expansion - 1648 1571
2.12.76 Core 8184 Add POS T_REIMB_AGREEM - 1651 1573
2.12.77 Core 10357 2408 DRG 1 - 1674 1575
2.12.78 Core 10356 2402 DRG 3 - 1676 1577
2.12.79 Core 10289 1555 - DRG 6 - 1679 1579
2.12.80 Core 10302 1463 - DRG 7 - 1680 1581
2.12.81 Core 10355 1466 - DRG 8 - 1681 1583
2.12.82 Core 10311 1077 - DRG 10 - 1683 1585
2.12.83 Core 10307 1448 - DRG 12 - 1685 1587
2.12.84 Core 10301 1447 - DRG 13 - 1686 1589
2.12.85 Xover Edits - 1694 1591
2.12.86 Xover Pricing - 1695 1593
2.12.87 Report Writer - Standard Footer - 1732 1594
2.12.88 Data Correction of Adjustments - 1778 1597
2.12.89 Add num_dtl t_clm_patliab_x app - 1795 1599
2.12.90 App chgs t_*_ub92_xover - 1796 1601
2.12.91 App chgs t_*_phys_xover - 1797 1603
2.12.92 App chgs t_pharm_hdr_keys - 1798 1605
2.12.93 Claims Search Results Panel - 1822 1607
2.12.94 Core 9704 T_REF_UCC 3 - 1922 1609
2.12.95 Core 9740 Reimb Rule Parms 3 - 1941 1611
2.12.96 DM indexes & default updates - 1963 1613
2.12.97 McKesson ClaimCheck Dental -File - 1976 1615
2.12.98 RPTS - Correct Source Code - 2012 1621
2.12.99 CORE 10380 NEW T_ADJMS_PAYBILL - 2013 1625
2.12.100 Diag s/d processing - core 9983 - 2014 1627
2.12.101 Limit audit by quadrnt-Core 9907 - 2015 1629
2.12.102 Equal auditing units - Core 9902 - 2016 1631
2.12.103 Audt proc dbChg impact-Core 9825 - 2017 1633
2.12.104 Contact limit audit - Core 9804 - 2018 1634
2.12.105 Calendar week limits - Core 9803 - 2019 1636
2.12.106 Conflict audit dates - Core 9802 - 2020 1638
2.12.107 Bypass on referral - Core 10373 - 2028 1640
2.12.108 CLM0022D -Invalid sak_case field - 2044 1642
2.12.109 Add Standard Date Heading Line - 2045 1644
2.12.110 Timely Filing Edits - CORE 6452 - 2046 1649
2.12.111 TPL Exhaust Ind - CORE 9712 - 2058 1651
2.12.112 Encounter Threshold - new panel - 2064 1652
2.12.113 CLM0130-Add Non-Adjudicated CLMS - 2071 1654
2.12.114 CLM9914&CLM5000 - Implement NPI - 2078 1657
2.12.115 PCTD205 -Delete Obsolete Reports - 2079 1659
2.12.116 Add KY Edit 054 - iC 3300 - 2080 1661
2.12.117 KY Dup Audits - 2 History Tables - 2082 1663
2.12.118 Re-Code State Specific Changes - 2083 1664
2.12.119 Add KY Edit 056 - iC 3301 - 2084 1667

2.12.120 Add KY Edit 067 - iC 3302 - 2090 1669
2.12.121 Add KY Edit 070 - iC 3303 - 2091 1671
2.12.122 Dup audits - dental - 2101 1673
2.12.123 Dup audits - ub92 - 2102 1675
2.12.124 CORE Dup audits - new tables - 2103 1676
2.12.125 CORE Dup audits - ub92 write - 2104 1678
2.12.126 Encounters - Pricing - 2106 1680
2.12.127 Encounters - Determination - 2107 1681
2.12.128 Encounters - Disposition - 2108 1683
2.12.129 Encounters - End Batch Process - 2109 1684
2.12.130 Encounters - Accept/Reject - 2110 1685
2.12.131 Encounters - Directories - 2114 1687
2.12.132 CORE Dup audits - phys write - 2115 1689
2.12.133 Dup audits - phys - 2116 1691
2.12.134 Dup audits - ub/phys read - 2126 1693
2.12.135 Multiple Surgery - CE - 2136 1694
2.12.136 KY - Add Hotlinks for PA#, CCN - 2164 1696
2.12.137 PA - Modify calls to find PA - 2167 1697
2.12.138 AVR - XML Transactions - 2174 1700
2.12.139 PA - use admit date for C/T I - 2181 1703
2.12.140 Allow Update of Coinsurance Days - 2194 1705
2.12.141 Send Enc Panel Data to Excel - 2196 1707
2.12.142 CLM0031M- Remove - 2198 1708
2.12.143 CLM0031M&CLM0060D-New Rpt Title - 2199 1709
2.12.144 CLM5000-Remove COS Field - 2208 1711
2.12.145 Enable ECS claim attachment - 2213 1712
2.12.146 Check panel - add totals - 2214 1713
2.12.147 Provider License Batch changes - 2219 1715
2.12.148 Add MCO ICN to Enc Related Clms - 2227 1717
2.12.149 CLM0023D - Add Encounter Section - 2232 1718
2.12.150 Add Non-Covered Charges - 2238 1721
2.12.151 Decision Form Rules Call - 2249 1723
2.12.152 Decision Form Rule Getter-EE 163 - 2250 1725
2.12.153 Multiple Surgery - DM - 2268 1727
2.12.154 KY Pricing No proc found def % - 2276 1729
2.12.155 KY Pricing - MAXFLT - 2277 1732
2.12.156 KY Pricing PT 21 Federal Match - 2278 1734
2.12.157 KY Phys Eval & Management - 2279 1735
2.12.158 KY School Based Group Pricing - 2280 1737
2.12.159 Add KY Edit 271 - iC 3305 - 2283 1740
2.12.160 KY DME Pricing E1399 - 2290 1742
2.12.161 KY DME Pricing PA Purchases - 2291 1744
2.12.162 KY Adult/Child Case Pricing - 2292 1747
2.12.163 KY CE Add Member County - 2294 1750
2.12.164 CLM0140D-Encounter Batch Summary - 2295 1752
2.12.165 KY Member County Pricing Method - 2297 1754
2.12.166 KY Non DRG Inpatient Pricing - 2298 1756
2.12.167 KY UB92 Lab Rate Defaults - 2299 1758
2.12.168 KY Pct of Charge Pricing Method - 2300 1760
2.12.169 KY Hospice Uses LTC Rates - 2301 1762
2.12.170 KY UB92 Rev Code Rates - 2302 1764
2.12.171 KY UB92 Single Unit Flat Rate - 2303 1766
2.12.172 KY Rev Code Pricing Xwalk - 2304 1768
2.12.173 KY Price Pct by Member Prov DOS - 2305 1770
2.12.174 KY Rev Proc Fee Pricing - 2306 1771
2.12.175 KY Bundled Rate Pricing - 2307 1773

2.12.176 KY Zero Pay Inclusive Rev Codes - 2308 1775
2.12.177 KY Max Flat 2 Pricing - 2309 1777
2.12.178 KY Pricing Child Special Hlth - 2337 1779
2.12.179 KY Pricing Impact Plus - 2338 1780
2.12.180 KY Pricing Support Comm Living - 2340 1782
2.12.181 Fix record length in clmsidx.ctl - 2359 1785
2.12.182 Update xsd for T_LIMIT_PARM - 2361 1786
2.12.183 Create new panel for Recipient B - 2365 1787
2.12.184 KY Pricing Primary Care / Rural - 2395 1788
2.12.185 McKesson ClaimCheck Dental-Job - 2396 1790
2.12.186 KY Rev Units Flat Pricing - 2423 1793
2.12.187 KY Pct of Per Diem Pricing - 2429 1795
2.12.188 KY UCC Flat Rate Pricing - 2446 1797
2.12.189 Add KY Edit 271 (103) - iC 3306 - 2448 1800
2.12.190 Add KY Edit 026 - iC 3308 - 2450 1802
2.12.191 Add KY Edit 036 - iC 3309 - 2451 1804
2.12.192 Add KY Edit 051 - iC 3361 - 2452 1806
2.12.193 Add KY Edit 052 - iC 3311 - 2453 1808
2.12.194 Add KY Edit 073 - iC 3312 - 2454 1810
2.12.195 Add KY Edit 124 - iC 3380 - 2457 1812
2.12.196 Add KY Edit 123 - iC 3372 - 2458 1814
2.12.197 Add KY edit 122 - iC 3317 - 2459 1816
2.12.198 Add KY Edit 117 - iC 3318 - 2460 1818
2.12.199 Add KY Edit 412 - iC 3320 - 2462 1820
2.12.200 Add KY Edit 354 - iC 3321 - 2463 1822
2.12.201 Add KY Edit 101 - iC 3322 - 2472 1824
2.12.202 Add KY Edit 377 - iC 3323 - 2474 1826
2.12.203 KY OP ASC Pricing - 2476 1828
2.12.204 KY Transplant Pricing - 2486 1830
2.12.205 Modify iC edit 368 (KY edit 083) - 2502 1832
2.12.206 Modify iC edit 365 (KY edit 084) - 2503 1834
2.12.207 KY Supports Comm Living - Config - 2535 1836
2.12.208 KY Edit 282 - iC Edit 2500 - 2539 1837
2.12.209 KY Edit 282 - iC Edit 2501 - 2540 1839
2.12.210 KY Edit 299 - iC Edit 2602 - 2541 1841
2.12.211 KY Specific - iC Edit 1008 - 2543 1843
2.12.212 Add KY Edit 188 - iC 3307 - 2544 1844
2.12.213 Add KY Edit 921 - iC 3326 - 2559 1846
2.12.214 Add KY Edit 918 - iC 3327 - 2560 1848
2.12.215 Add KY Header E915 - iC 3329 - 2562 1850
2.12.216 Add KY Detail E915 - iC 3330 - 2563 1852
2.12.217 Add KY Edit 912 - iC 3331 - 2567 1854
2.12.218 Add KY Edit 835 - iC 3332 - 2568 1856
2.12.219 Add KY Edit 923 - iC 3333 - 2569 1857
2.12.220 Modify iC edit 2502 for KY E283 - 2571 1859
2.12.221 Add Ky Edit 942 - iC 3334 - 2606 1861
2.12.222 Add KY Edit 981 - iC 3335 - 2609 1863
2.12.223 Update UB92 Submitted Data - 2610 1865
2.12.224 Encounter Panel table change - 2611 1867
2.12.225 Modify iC to match KY Edit 180 - 2613 1868
2.12.226 Modify iC to Match KY Edit 181 - 2614 1870
2.12.227 Modify iC to Match KY Edit 182 - 2615 1871
2.12.228 Modify iC to Match KY Edit 183 - 2616 1872
2.12.229 KY Edit 198 - iC Edit 2057 - 2617 1873
2.12.230 CLM0140D-Replace T_UPLOAD Table - 2622 1875
2.12.231 Add KY Edit 989 - iC 3337 - 2626 1876

2.12.232 DB changes to include Level of C - 2694 1878
2.12.233 Update T_UB92_DTL_EXT_KEY Table - 2701 1880
2.12.234 Modify Prescb Prov on Pharm Hdr - 2711 1882
2.12.235 Update T_UB92_EXT_KEY.XSD - 2713 1884
2.12.236 Add Ky Edit 996 - iC 3339 - 2741 1886
2.12.237 Add Ky Edit 977 - iC 3340 - 2742 1888
2.12.238 Add Ky Edit 930 Header- iC 3341 - 2764 1889
2.12.239 Add Ky E930 (Detail) - iC 3342 - 2767 1891
2.12.240 Add KY Edit 953 - iC 3343 - 2775 1892
2.12.241 Add KY Edit 078 - 3344 - 2776 1894
2.12.242 Add KY Edit 011 - iC 3345 - 2777 1896
2.12.243 Modify iC to Match KY Edit 146 - 2779 1898
2.12.244 Modify iC to Match KY Edit 151 - 2780 1900
2.12.245 Modify iC to Match KY Edit 152 - 2786 1902
2.12.246 Add Ky Edit 972 - iC 3346 - 2794 1903
2.12.247 Add Ky Edit 978 - iC 3347 - 2823 1905
2.12.248 Add Ky Edit 986 - iC 3348 - 2824 1907
2.12.249 Add Ky Edit 987 - iC 3349 - 2834 1908
2.12.250 Add KY Edit 284 - iC 3351 - 2919 1910
2.12.251 Modify iC to Match KY Edit 283 - 2920 1912
2.12.252 KY CE Add Relationships - 2990 1914
2.12.253 KY CE Bundled Pricing Variable - 2996 1916
2.12.254 KY CE Add OOS Ind Variable - 2999 1918
2.12.255 Add KY Edit 074 - iC 3352 - 3002 1920
2.12.256 Add KY Edit 007 - iC 3353 - 3003 1922
2.12.257 Modify iC to Match KY Edit 007 - 3004 1924
2.12.258 Add KY Edit 063 - iC 3354 - 3005 1926
2.12.259 Add KY Edit 096 - iC 3355 - 3006 1928
2.12.260 Modify iC edit 226 - 3009 1930
2.12.261 KY Remove Old Mult Surg ASCs - 3011 1932
2.12.262 KY PA Edits DME Bypass - 3017 1933
2.12.263 Add KY Edit 279 - iC 3356 - 3025 1935
2.12.264 Add KY Edit 280 - iC 3357 - 3026 1937
2.12.265 Add Ky Edit 319 - iC 3358 - 3028 1939
2.12.266 Add Partition Key To ADJ XREF Un - 3029 1941
2.12.267 Dummy CO 4 unlinked binaries/XSD - 3074 1943
2.12.268 Add Ky Edit 961 - iC 3359 - 3101 1945
2.12.269 KYAMEND Support PA mass updates - 3102 1947
2.12.270 Benefit Limits - Visit/contact I - 3131 1949
2.12.271 KYAMEND new PA fields - 3137 1951
2.12.272 Add Ky Edit 310 - iC 3360 - 3149 1952
2.12.273 Add Ky Edit 908 - iC 3363 - 3155 1954
2.12.274 Update KY DNN file - 3175 1956
2.12.275 Add Hyperlinks for ICN / Check - 3199 1957
2.12.276 KY DRG Claims Panel Updates - 3246 1959
2.12.277 DM - add columns to T_CLM_MISC - 3272 1960
2.12.278 KY Pricing Lesser Of Bill/Allow - 3338 1962
2.12.279 Claim Misc Info Config Change - 3347 1963
2.12.280 Require full ICN to limit search - 3362 1965
2.12.281 Re-write Decision Rules panel - 3363 1966
2.12.282 LTC DME Pricing Cutbacks - 3394 1968
2.12.283 Add Rules Setter Xover Pricing - 3415 1970
2.12.284 Crossover Pricing - 3418 1971
2.12.285 Binary update test-mod - 3427 1973
2.12.286 Xover and DRG Fund Code Asg - 3464 1979
2.12.287 Redesign Claim Detail Panels - 3481 1980

2.12.288 No navigation to return to Adj's - 3506 1982
2.12.289 UB Dtl Keys - Add subNonCovAmt - 3529 1984
2.12.290 Save Submitted Non Covered Amoun - 3530 1986
2.12.291 Add Submitted Non Cov Amount UI - 3531 1988
2.12.292 Change Provider ID to Number - 3541 1989
2.12.293 Post Hdr-Dtl - 3552 1991
2.12.294 Reimb Modifier Pricing Type - CE - 3553 1992
2.12.295 2652 Spenddown by Benefit Plan - 3557 1994
2.12.296 7472 Cost Share Amounts on Clm - 3558 1995
2.12.297 Enable image access - 3559 1996
2.12.298 DM - Update Dental Header Tbls - 3560 1997
2.12.299 UI - Dntl - Accident Date/Cause - 3563 1999
2.12.300 UF - E593 - Can't Adj a Susp Cla - 3565 2000
2.12.301 Add error eobs to adjustments - 3566 2001
2.13 Data Element Dictionary 2002
2.13.1 T_ADJMS_AGE_SEX 2002
2.13.2 T_ADJMS_AID_CAT 2002
2.13.3 T_ADJMS_CT 2003
2.13.4 T_ADJMS_DATES 2004
2.13.5 T_ADJMS_DIAG 2006
2.13.6 T_ADJMS_DRG 2006
2.13.7 T_ADJMS_DRUG 2007
2.13.8 T_ADJMS_ERROR 2008
2.13.9 T_ADJMS_LOC 2009
2.13.10 T_ADJMS_MEDAST 2009
2.13.11 T_ADJMS_MODIF 2010
2.13.12 T_ADJMS_PROV 2011
2.13.13 T_ADJMS_PROVTYP 2012
2.13.14 T_ADJMS_PRSPEC 2013
2.13.15 T_ADJMS_PRSVC 2014
2.13.16 T_ADJMS_RECIP 2014
2.13.17 T_ADJMS_REGION 2015
2.13.18 T_ADJMS_REVENUE 2016
2.13.19 T_ADJMS_SERV 2016
2.13.20 T_ADJ_DENT_XREF 2017
2.13.21 T_ADJ_MASS_CLAIM 2018
2.13.22 T_ADJ_MASS_RQST 2021
2.13.23 T_ADJ_PHRM_XREF 2022
2.13.24 T_ADJ_PHYS_XREF 2023
2.13.25 T_ADJ_RET_RQST_ERR 2024
2.13.26 T_ADJ_REV_RQST 2024
2.13.27 T_ADJ_RULES 2025
2.13.28 T_ADJ_UB92_XREF 2026
2.13.29 T_ADJ_VOID_RQST 2027
2.13.30 T_ATTACHMENT_XREF 2028
2.13.31 T_BATCH_ERR_SUM 2028
2.13.32 T_CCF_ERROR_DATA 2029
2.13.33 T_CC_CR_SAVE 2030
2.13.34 T_CC_HIST_ADJUST 2031
2.13.35 T_CDE_AID 2033
2.13.36 T_CHCK_CLM_XREF 2034
2.13.37 T_CLAIM_ERROR 2035
2.13.38 T_CLAIM_LOCAT 2037
2.13.39 T_CLAIM_TYPE 2037
2.13.40 T_CLM_ADR_N3_N4 2038
2.13.41 T_CLM_AMT 2038

2.13.42 T_CLM_AMT_VARIANCE 2039
2.13.43 T_CLM_BATCH_RNG 2040
2.13.44 T_CLM_BATCH_XREF 2041
2.13.45 T_CLM_CAS 2041
2.13.46 T_CLM_CN1 2042
2.13.47 T_CLM_COMP_DAYS 2043
2.13.48 T_CLM_CR1 2043
2.13.49 T_CLM_CR2 2044
2.13.50 T_CLM_CR3 2045
2.13.51 T_CLM_CR3_TEST 2046
2.13.52 T_CLM_CR5 2046
2.13.53 T_CLM_CR6 2048
2.13.54 T_CLM_CRC 2050
2.13.55 T_CLM_DIAG_XREF 2051
2.13.56 T_CLM_DN2 2051
2.13.57 T_CLM_DTL_TTH_SFC 2052
2.13.58 T_CLM_DTP 2052
2.13.59 T_CLM_ENTITY 2053
2.13.60 T_CLM_ENT_NM_ADR 2054
2.13.61 T_CLM_EOB_XREF 2055
2.13.62 T_CLM_HDR_CCF 2057
2.13.63 T_CLM_HDR_DELIVERY 2057
2.13.64 T_CLM_HH_CARE_PLAN 2057
2.13.65 T_CLM_HSD 2058
2.13.66 T_CLM_K3 2059
2.13.67 T_CLM_MEA 2059
2.13.68 T_CLM_MIA 2060
2.13.69 T_CLM_MISC 2064
2.13.70 T_CLM_MOA 2066
2.13.71 T_CLM_MODIFIER 2067
2.13.72 T_CLM_NAME_NM1 2068
2.13.73 T_CLM_NTE 2068
2.13.74 T_CLM_OTH_PYR_DTL 2069
2.13.75 T_CLM_PATLIAB_X 2070
2.13.76 T_CLM_PAYER_ENTITY 2071
2.13.77 T_CLM_PGM_XREF 2072
2.13.78 T_CLM_PHRM_PAPER_IMG 2076
2.13.79 T_CLM_PHRM_ 2077
2.13.80 T_CLM_PWK 2078
2.13.81 T_CLM_PYR_ENTNMADR 2079
2.13.82 T_CLM_PYR_MOD 2080
2.13.83 T_CLM_QTY 2081
2.13.84 T_CLM_REF 2081
2.13.85 T_CLM_RU_MATCHED 2082
2.13.86 T_CLM_SBR 2083
2.13.87 T_CLM_STATUS 2084
2.13.88 T_CLM_SUM_ERR_X 2085
2.13.89 T_COS_DTL_XREF 2085
2.13.90 T_CT_TXN_XREF 2086
2.13.91 T_DCOR_QLTY_CTL 2087
2.13.92 _DCOR_SCHEDULE 2087
2.13.93 T_DENTAL_DTL_KEYS 2088
2.13.94 T_DENTAL_HDR_KEYS 2091
2.13.95 T_DENY_DNTL_DTL 2094
2.13.96 T_DENY_DNTL_HDR 2098
2.13.97 T_DENY_PHRM_DTL 2104

2.13.98 T_DENY_PHRM_HDR 2104
2.13.99 T_DENY_PHYS_DTL 2110
2.13.100 T_DENY_PHYS_HDR 2115
2.13.101 T_DENY_UB92_DTL 2121
2.13.102 T_DENY_UB92_HDR 2124
2.13.103 T_DNTL_OTH_PYR_HDR 2131
2.13.104 T_EDI_INTERCHANGE 2132
2.13.105 T_ENC_BATCH_PROCESS 2134
2.13.106 T_FINAL_PHYS_XOVER 2135
2.13.107 T_FINAL_UB92_XOVER 2136
2.13.108 T_FIN_FUND_CODE 2137
2.13.109 T_FIN_PROCESS_EOMB 2138
2.13.110 T_HIST_DIRECTORY 2138
2.13.111 T_HIST_PART 2140
2.13.112 T_HS_DIR_PART 2140
2.13.113 T_INST_OTH_PYR_HDR 2141
2.13.114 T_LOCATION_NOTE 2142
2.13.115 T_MCARE_DEDUCTIBLE 2143
2.13.116 T_MEDPOL_UB92 2144
2.13.117 T_MEDPOL_UB92_DIAG 2146
2.13.118 T_MPHX_DENTAL_DTL 2147
2.13.119 T_MPHX_INPT_HDR 2150
2.13.120 T_MPHX_INPT_HDR2 2151
2.13.121 T_MPHX_MISC 2152
2.13.122 T_MPHX_PHRM_DTL 2153
2.13.123 T_MPHX_PHYS_DTL 2155
2.13.124 T_MPHX_PH_DIAG 2158
2.13.125 T_MPHX_XOVER 2158
2.13.126 T_PARTY_IDENTIFIER 2159
2.13.127 T_PA_ITEM_DTL_XREF 2160
2.13.128 T_PBM_PHARM_EXTRACT 2161
2.13.129 T_PD_DNTL_DTL 2161
2.13.130 T_PD_DNTL_HDR 2164
2.13.131 T_PD_PHARM_DTL 2170
2.13.132 T_PD_PHARM_HDR 2171
2.13.133 T_PD_PHYS_DTL 2177
2.13.134 T_PD_PHYS_HDR 2182
2.13.135 T_PD_UB92_DTL 2188
2.13.136 T_PD_UB92_HDR 2191
2.13.137 T_PHRM_COB 2198
2.13.138 T_PHRM_COB_XREF 2198
2.13.139 T_PHRM_DTL_KEYS 2199
2.13.140 T_PHRM_HDR_KEYS 2199
2.13.141 T_PHRM_HDR_OPT 2200
2.13.142 T_PHYS_DEXT_KEY 2202
2.13.143 T_PHYS_HDR_KEY 2206
2.13.144 T_PRICING_XOVER 2210
2.13.145 T_PROF_OTH_PYR_HDR 2210
2.13.146 T_SAK_ICN_DAILY 2211
2.13.147 T_SUSP_ADJ_XREF 2212
2.13.148 T_SUSP_DENTAL_DTL 2212
2.13.149 T_SUSP_DENTAL_HDR 2216
2.13.150 T_SUSP_PHRM_DTL 2221
2.13.151 T_SUSP_PHRM_HDR 2222
2.13.152 T_SUSP_PHYS_DTL 2227
2.13.153 T_SUSP_PHYS_HDR 2232

2.13.154 T_SUSP_PHYS_XOVER 2237
2.13.155 T_SUSP_RLHX 2238
2.13.156 T_SUSP_UB92_DTL 2239
2.13.157 T_SUSP_UB92_HDR 2241
2.13.158 T_SUSP_UB92_XOVER 2247
2.13.159 T_TCN_XREF 2248
2.13.160 T_UB92_DTL_EXT_KEY 2248
2.13.161 T_UB92_DTL_OUTPAT_EDIT 2251
2.13.162 T_UB92_DTL_OUTPAT_GROUP 2251
2.13.163 T_UB92_DTL_OUTPAT_PRICE 2255
2.13.164 T_UB92_HDR_COND_X 2255
2.13.165 T_UB92_HDR_DIAG_X 2256
2.13.166 T_UB92_HDR_EXT_KEY 2257
2.13.167 T_UB92_HDR_ICD9CM 2260
2.13.168 T_UB92_HDR_INP 2261
2.13.169 T_UB92_HDR_OCC 2263
2.13.170 T_UB92_HDR_OUTPAT 2264
2.13.171 T_UB92_HDR_OUTPAT_EDIT 2266
2.13.172 T_UB92_HDR_PAYER 2267
2.13.173 T_UB92_HDR_VALUE 2267
2.13.174 T_UB92_NH_HDR 2269
2.13.175 T_UB92_TREATMENT 2269

3 Buy-In Data Mainenance Detail System Design 2270
3.1 Buy-In Data Maintenace Subsystem Flow Diagram 2270
3.2 Subsystem Description 2270
3.2.1 Introduction to Buy-In Data Maintenace 2270
3.3 Buy-In Data Maintenance Processes 2271
3.3.1 Buy-In Through PCG Process Model 2271
3.3.2 EDB Process Model 2273
3.3.3 Member Extract to PCG Process Model 2274
3.4 Buy-In Data Maintenace Data Model 2276
3.4.1 Database Entity Relationships 2276
3.4.2 External System Interfaces 2276
3.5 Job Scripts 2277
3.5.1 BIAJM001 - Print Buy-In Part A Billing report 2277
3.5.2 BIAJM002 - Print the Buy-In Part A List of Transactions Sent to Baltimore (Sending) report 2278
3.5.3 BIAJM004 - Buy-In Part A Exception Error By Transaction Code 2279
3.5.4 BIAJM010 - Buy-In Part A Expenditure report 2280
3.5.5 BIAJM011 - Print Buy-In Part A State Totals by Member Population report 2281
3.5.6 BIAJM012 - Print Buy-In Part A Conditional report 2282
3.5.7 BIAJM013 - Buy-In Part A Premium Totals 2283
3.5.8 BIBJM001 - Print Buy-In Part B Billing report 2284
3.5.9 BIBJM002 - Print the Buy-In Part B List of Transactions Sent to Baltimore (Sending) report. 2285
3.5.10 BIBJM004 - Buy-In Part B Exception Error By Transaction Code 2286
3.5.11 BIBJM010 - Buy-In Part B Expenditure report 2287
3.5.12 BIBJM011 - Buy-In Part B State Totals By Member Population 2288
3.5.13 BIBJM013 - Buy-In Part B Premium Totals 2289
3.5.14 BUYENDJM - Buy-in Month End Parm Job Script 2290
3.5.15 BUYJD603 - Daily Extract for KMAA 2291
3.5.16 BUYJM001 - Print Possible Medicare Eligible report 2292
3.5.17 BUYJM152 - EDB Missing Sequence report 2293
3.5.18 BUYJM401 - Buy-In Billing Process Error report 2294
3.5.19 BUYJM403 - Buy-In Original ID Change report 2295

3.5.20 BUYJM404 - Buy-In Death report 2296
3.5.21 BUYJM405 - Buy-In Accretion report 2297
3.5.22 BUYJM406 - Buy-In Transactions to CMS Summary report 2298
3.5.23 BUYJM407 - Buy-In Manual Transactions Submitted report 2299
3.5.24 BUYJM408 - Buy-In Matching Transactions report 2300
3.5.25 BUYJM409 - Buy-In Recipients Not Billed report 2301
3.5.26 BUYJM410 - Buy-In Unmatched Recipients Billed report 2302
3.5.27 BUYJM411 - Buy-In Misc Debit/Credit report 2303
3.5.28 BUYJM431 - MMA Outbound reports 2304
3.5.29 BUYJM508 - EDB-HIC Cross Reference report 2305
3.5.30 BUYJM523 - EDB Processing reports 2306
3.5.31 BUYJM600 - PCG Buy-In Reconciliation File 2308
3.5.32 BUYJM603 - Process PCG Input Files 2309
3.5.33 BUYJM605 - Buy-In PCG Billing File Update 2310
3.5.34 BUYJM_MMAENROLL - Part D Current Enrollments 2314
3.5.35 BUYJM_MMAMERGE - Part D Outbound Processing 2315
3.5.36 BUYJM_MMARECON - Part D Adjustments to Previous Enrollments 2316
3.5.37 BUYJM_MMARETRO - Part D Retroactive Enrollments 2317
3.5.38 ELGJD111 - Part D Daily Reconciliation 2318
3.5.39 ELGJM150 - Process EDB file 2319
3.5.40 ELGJM153 - EDB Extract File 2320
3.5.41 ELGJM650 - Part D Inbound Processing 2321
3.5.42 ELGJM651 - Part D MMA Inbound report 2323
3.5.43 ELGJM654 - Update Clawback Information 2324
3.5.44 ELGJM655 - Print the Clawback report 2325
3.6 Programs 2326
3.6.1 tpl0021m -- Buy-In Part B State Totals by Beneficiary Population 2326
3.6.2 bia1001m -- Print Buy-In Part A Billing report 2327
3.6.3 bia1002m -- Print the Buy-In Part A List of Transactions Sent to Baltimore (Sending) report 2329
3.6.4 bia1004m -- Buy-In Part A Exception Error By Transaction Code 2331
3.6.5 bia1010m -- Buy-In Part A Expenditure report 2335
3.6.6 bia1011m -- Print Buy-In Part A State Totals by Member Population report 2337
3.6.7 bia1012m -- Buy-In Part A Conditional report services utilization report 2338
3.6.8 bia1013m -- Buy-In Part A Premium Totals 2339
3.6.9 bib2001m -- Print Buy-In Part B Billing report 2341
3.6.10 bib2002m -- Buy-In Part B List of Transactions Sent to Baltimore (Sending) 2343
3.6.11 bib2004m -- Buy-In Part B Exception Error By Transaction Code 2345
3.6.12 bib2010m -- Buy-In Part B Expenditure report 2347
3.6.13 bib2011m -- Print Buy-In Part B State Totals by Member Population report 2349
3.6.14 bib2013m -- Buy-In Part B Premium Totals 2350
3.6.15 buympm -- Buy-in mismatch program 2352
3.6.16 buypd603 -- Buy-In Extract for KMAA 2353
3.6.17 buypm401 -- Billing Process Error report 2354
3.6.18 buypm403 -- Original ID Change report 2355
3.6.19 buypm404 -- Buy-In Death report 2357
3.6.20 buypm405 -- Buy-In Accrete report 2358
3.6.21 buypm406 -- Transaction to CMS Summary report 2359
3.6.22 buypm407 -- Manual Transactions Submitted report 2360
3.6.23 buypm408 -- Matching Transactions report 2361
3.6.24 buypm409 -- Recipients Not Billed report 2362
3.6.25 buypm410 -- Unmatched Recipients Billed report 2363
3.6.26 buypm411 -- Misc Debit/Credit report 2364
3.6.27 buypm412 -- Trans Code Financial Summary report 2365
3.6.28 buypm431 -- MMA Outbound reports 2366
3.6.29 buypm523 -- EDB Entitlement Processing reports 2367

3.6.30 buypm600 -- PCG Reconciliation File Extract 2369
3.6.31 buypm603 -- Buy-In PCG Input File update 2370
3.6.32 buypm604 -- Pseudo-41 2371
3.6.33 buypm605 -- Buy-In PCG Billing File Update 2372
3.6.34 buypm606 -- Buy-In Transaction 23 filter program 2374
3.6.35 buypmMMAEnroll -- Part D Current Month Enrollments 2375
3.6.36 buypmMMAMerge -- Part D Outbound File Generator 2376
3.6.37 buypmMMARecon -- Part D Retroactive Changes to Enrollments 2377
3.6.38 buypmMMARetro -- Part D Retroactive Enrollments 2378
3.6.39 buypmalt -- Alternate Demographics Patch 2379
3.6.40 buypxlat -- Buy-In Data Translator 2380
3.6.41 buyrpm -- Buy-In Receiving Batch Program 2381
3.6.42 buysapm -- Buy-In State Actions Generation Program 2383
3.6.43 buyspm -- Buy-In Sending Batch Program 2385
3.6.44 elgpd05m -- Part D Daily Recon 2387
3.6.45 elgpm150 -- Process EDB file 2388
3.6.46 elgpm152 -- Missing Sequence Number report 2390
3.6.47 elgpm153 -- EDB Extract file 2392
3.6.48 elgpm650 -- Part D MMA Processing 2394
3.6.49 elgpm651 -- Part D MMA Inbound report 2395
3.6.50 elgpm652 -- Part D MMA Outbound report 2396
3.6.51 elgpm654 -- Load Clawback amounts 2397
3.6.52 elgpm655 -- Print Clawback Summary report 2398
3.6.53 elgpm660 -- Part D Entitlement Date Mismatch report 2399
3.6.54 elgpm670 -- Part D PDP Errors report 2400
3.7 Internal Web Pages 2401
3.8 Reports 2402
3.8.1 BIA-1001-M -- Buy-In Part A Billing (Receiving) 2402
3.8.2 BIA-1002-M -- Buy-In Part A Input Transactions Within interChange 2410
3.8.3 BIA-1004-M -- Buy-In Part A Exception Error By Transaction Code 2414
3.8.4 BIA-1010-M -- Buy-In Part A Expenditure Report 2419
3.8.5 BIA-1011-M -- Buy-In Part A State Totals by Member Population 2422
3.8.6 BIA-1012-M -- Buy-In Part A Conditional 2424
3.8.7 BIA-1013-M -- Buy-In Part A Premium Totals 2427
3.8.8 BIB-2001-M -- Buy-In Part B Billing (Receiving) 2432
3.8.9 BIB-2002-M -- Buy-In Part B Input Transactions Within interChange 2442
3.8.10 BIB-2004-M -- Buy-In Part B Exception Error By Transaction Code 2447
3.8.11 BIB-2010-M -- Buy-In Part B Expenditure Report 2452
3.8.12 BIB-2011-M -- Buy-In Part B State Totals by Member Population 2455
3.8.13 BIB-2013-M -- Buy-In Part B Premium Totals 2458
3.8.14 BUY-3001-M -- Possible Medicare Eligible's 2462
3.8.15 BUY-4001-M -- Buy-In Billing Process Error 2465
3.8.16 BUY-4003-M -- Buy-In Original ID Change 2468
3.8.17 BUY-4004-M -- Buy-In Death 2471
3.8.18 BUY-4005-M -- Buy-In Accrete 2474
3.8.19 BUY-4006-M -- Buy-In Transaction To CMS Summary 2477
3.8.20 BUY-4007-M -- Buy-In Manual Transactions Submitted 2479
3.8.21 BUY-4008-M -- Buy-In Matching Transactions 2483
3.8.22 BUY-4009-M -- Buy-In Members Not Billed 2485
3.8.23 BUY-4010-M -- Buy-In Unmatched Members Billed 2488
3.8.24 BUY-4011-M -- Buy-In Misc Debit Credit 2491
3.8.25 BUY-4310-M -- MMA Enrollment File 2494
3.8.26 BUY-4311-M -- LIS Enrollment File 2497
3.8.27 BUY-5003-M -- EDB Entitlement Errors 2501
3.8.28 BUY-5004-M -- EDB Entitlement Update 2504
3.8.29 BUY-5005-M -- EDB Entitlement Shutdown 2507

3.8.30 BUY-5006-M -- EDB Entitlement Add 2509
3.8.31 BUY-5007-M -- EDB Entitlement Bypass 2512
3.8.32 BUY-5008-M -- EDB-HIC Cross Reference 2515
3.8.33 BUY-5009-M -- EDB Entitlement Denied 2517
3.8.34 Buy-In Report Header -- Buy-In Report Standard Header 2520
3.8.35 ELG-0152-M -- EDB Missing Sequence Number 2522
3.8.36 ELG-0650-M -- Part D MMA File Error 2524
3.8.37 ELG-0651-M -- Part D MMA Inbound 2526
3.8.38 ELG-0652-M -- Part D MMA Outbound 2528
3.8.39 ELG-0655-M -- Part D Liability 2530
3.8.40 ELG-0660-M -- Part D Reconciliation 2533
3.8.41 ELG-0670-M -- Part D Missing PDP Assignments 2535
3.9 Letters 2537
3.10 Internal and External Interfaces 2538
3.10.1 External Files 2538
3.10.2 Input Files 2538
3.10.3 Output Files 2538
3.10.4 File Layouts 2538
3.10.5 Buy-In New CMS Layouts 2539
3.10.6 CMS EDB File Layout 2565
3.10.7 EDS DB File Layout 2576
3.10.8 KMAA Extract File 2580
3.10.9 Part D MMA Enrollment File (Old Version) 2581
3.10.10 Part D MMA Enrollment File 2587
3.10.11 Part D MMA Enrollment Response File (Old Version) 2598
3.10.12 PCG Modified CMS Layouts 2673
3.10.13 Record Layout – EDB Finder File 2674
3.11 Requirement Matrix and Cross Reference 2676
3.12 Change Orders 2680
3.12.1 Report - 'Recipient' to 'Member' - 183 2680
3.12.2 Panel - 'Recipient' to 'Member' - 184 2681
3.12.3 Process EDB Enrollment Database - 226 2683
3.12.4 New Table - T_RE_EDB - 427 2685
3.12.5 Create EDB Panels - 428 2686
3.12.6 Use Standard Report Headings - 458 2688
3.12.7 Modify BIA-1001-M Report - 459 2689
3.12.8 Modify BIA-1002-M Report - 460 2690
3.12.9 Modify BIA-1004-M Report - 461 2692
3.12.10 Disable BIA-1009-M BIB-2009-M - 462 2694
3.12.11 Modify BIA-1011-M Report - 463 2695
3.12.12 Modify BIA-1013-M Report - 464 2697
3.12.13 Modify BIB-2001-M Report - 465 2699
3.12.14 Modify BIB-2002-M Report - 466 2701
3.12.15 Modify BIB-2004-M Report - 467 2703
3.12.16 Modify BIB-2011-M Report - 468 2705
3.12.17 Modify BIB-2013-M Report - 469 2707
3.12.18 Billing Process Error Report - 476 2709
3.12.19 Original ID Change Report - 478 2711
3.12.20 Buy-In Death Report - 479 2713
3.12.21 Retroactive Buyin Accrete Report - 480 2715
3.12.22 Transaction to CMS Summary Rpt - 481 2717
3.12.23 Manual Transactions Submitted - 482 2719
3.12.24 Matching Transactions Report - 483 2721
3.12.25 Recipients Not Billed Report - 484 2723
3.12.26 Unmatched Recipients Billed - 485 2725
3.12.27 Misc Debit/Credit Transactions - 486 2727

3.12.28 Create New CMS Datamodel - 488 2729
3.12.29 Buy-In Alternate Demographics - 489 2731
3.12.30 EDB-HIC Cross-Reference Report - 506 2733
3.12.31 EDB Finder File Extract - 1077 2735
3.12.32 Buy-In A Combined Panel - 1078 2737
3.12.33 Buy-In B Combined Panel - 1079 2739
3.12.34 PCG Buy-In Update - 1080 2741
3.12.35 EDB Missing Sequence Report - 1156 2743
3.12.36 PCG Buy-In Recon Interface - 1183 2745
3.12.37 CMS Batch Receiving System - 1419 2747
3.12.38 CMS State Action System - 1420 2749
3.12.39 CMS Batch Sending System - 1421 2751
3.12.40 EDB Processing Reports - 1464 2752
3.12.41 Buy-In Extract for KMAA - 2029 2754
3.12.42 Pseudo-41 Transactions - 2073 2756
3.12.43 Modify BIA-1010-M BIB-2010-M - 2144 2758
3.12.44 Update data model - 2258 2760
3.12.45 Modify reports - 2259 2762
3.12.46 Part D - Standard Report Text - 2468 2764
3.12.47 Part D - Convert Core Panels - 2469 2766
3.12.48 Part D - Convert Batch System - 2470 2768
3.12.49 Modify Buy-In Batch Scripts - 2570 2770
3.12.50 New Table T_RE_CLAWBACK - 2638 2772
3.12.51 Load Clawback Data - 2639 2773
3.12.52 New report ELG-0655-M - 2641 2775
3.12.53 Modify MMA Panel - 2642 2776
3.12.54 Repla Tbl: T_RE_AID_ELIG_STATE - 3064 2778
3.12.55 New Table T_BUY_PARTD_LIS - 3312 2779
3.12.56 New Panel - Part D LIS - 3313 2780
3.12.57 Merge Part D LIS - 3314 2782
3.12.58 Modify EDB Finder File - 3387 2784
3.12.59 Modify Buy-In Menu - 3472 2786
3.12.60 REL2-DCR1178-CO1178-Add PRO recs - 4418 2788
3.12.61 REL2-Forward PCG files to CMS - 5654 2789
3.12.62 Send reports to PCG - 5656 2790
3.12.63 REL2-New Index on T_RE_EDB - 6075 2791
3.12.64 Process EDB File from HMS - 6872 2793
3.12.65 Modify EDB Reports for HMS data - 6873 2796
3.12.66 Modify EDB Panel to indicate HMS - 6874 2798
3.12.67 Modify MMA History Panel - 7321 2800
3.12.68 Revise Part D Dual Status Code - 7689 2802
3.12.69 Update Clawback Amts - 8213 2804
3.12.70 I3429 - 8530 2805
3.12.71 Part D - Add code 99 - 8773 2807
3.12.72 Add Legacy edits to EDB input - 8777 2809
3.12.73 I3482 - 8974 2811
3.12.74 Fulll EDB finder file - 9036 2812
3.13 Data Element Dictionary 2813
3.13.1 T_BUYA_BILL 2813
3.13.2 T_BUYA_BILL_INFO 2815
3.13.3 T_BUYA_DEMO_CHANGE 2816
3.13.4 T_BUYA_ERROR 2817
3.13.5 T_BUYA_EXCEPT 2818
3.13.6 T_BUYA_MISMATCH 2820
3.13.7 T_BUYA_PERD 2824
3.13.8 T_BUYA_PREM 2824

3.13.9 T_BUYB_BILL 2825
3.13.10 T_BUYB_BILL_INFO 2828
3.13.11 T_BUYB_DEMO_CHANGE 2829
3.13.12 T_BUYB_ERROR 2830
3.13.13 T_BUYB_EXCEPT 2831
3.13.14 T_BUYB_MISMATCH 2833
3.13.15 T_BUYB_PERD 2837
3.13.16 T_BUYB_PREM 2837
3.13.17 T_BUY_PARMS 2838
3.13.18 T_BUY_PARTD_LIS 2839
3.13.19 T_CDE_BUY_BILL 2840
3.13.20 T_CDE_BUY_ELIG 2842
3.13.21 T_CDE_BUY_PREM 2842
3.13.22 T_RE_BASE 2842

1 Introduction

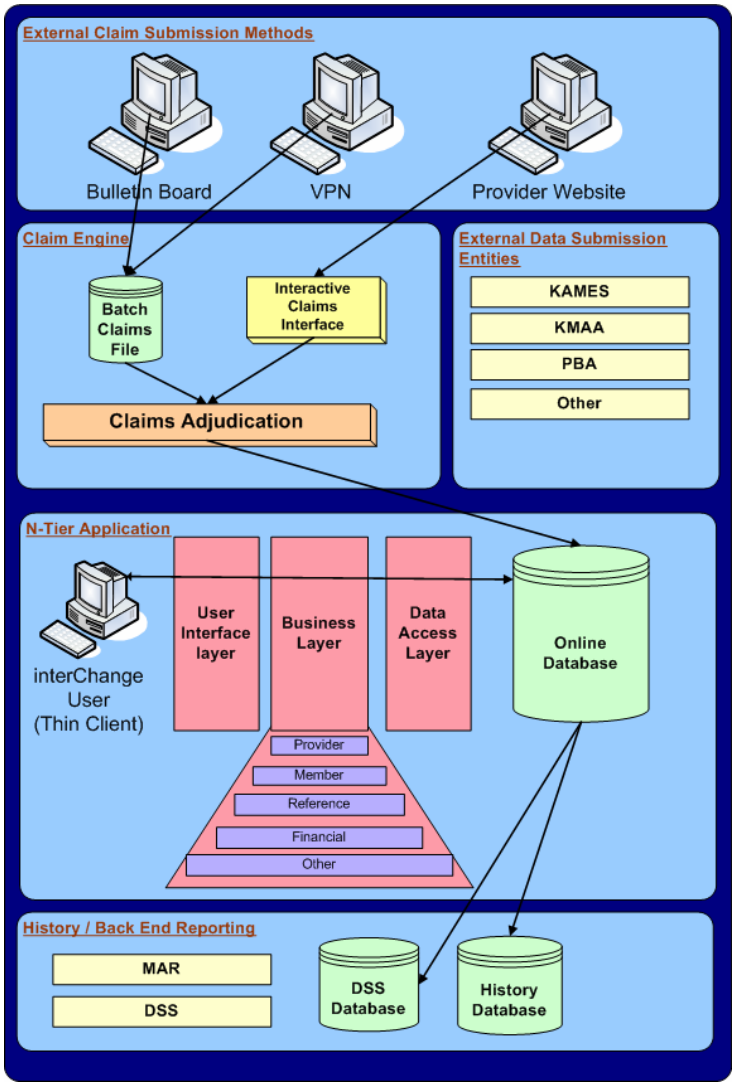
This document provides the Claims Detailed System Design (DSD). This DSD combines two functional areas: Claims and Buy-In Data Maintenance. The DSD validates the design process and approach to solution, and identifies those items as specified in the Kentucky MMIS (KY MMIS) Request for Proposal (RFP) and as specified by Kentucky Medicaid.

2 Detailed System Design

2.1 System Flow

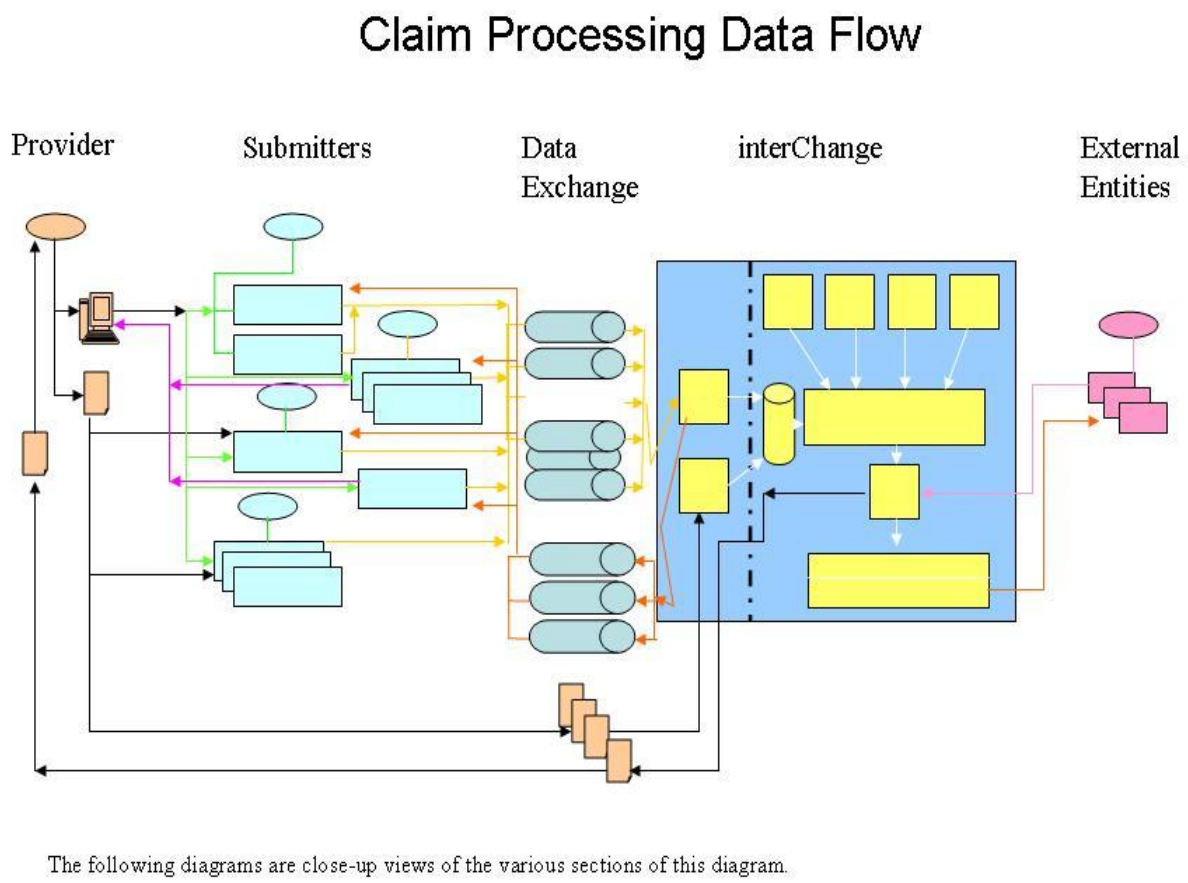
2.1.1 interChange System Architecture

The system is logically divided into four primary components: Claims engine, User Interface, Batch, and the History and Back End Reporting. The Claims engine is responsible for receiving interactive transactions from external sources, adjudicating them, and returning the appropriate response. The User Interface is an N-tier application providing segregated and loosely coupled presentation, business logic, and data logic layers. The user interface provides access to the online subsystem functions through a thin client, the web browser. The Batch component is responsible for maintaining and reporting on data contained within the online database. The History and Back End reporting component is responsible for analyzing, reporting, and supporting the management of the activities that have occurred in the two front end systems. The system interfaces with a variety of data sources which influence processing within the system. The External data submission entities are organizations that supply information to the Medicaid Management Information System (MMIS).

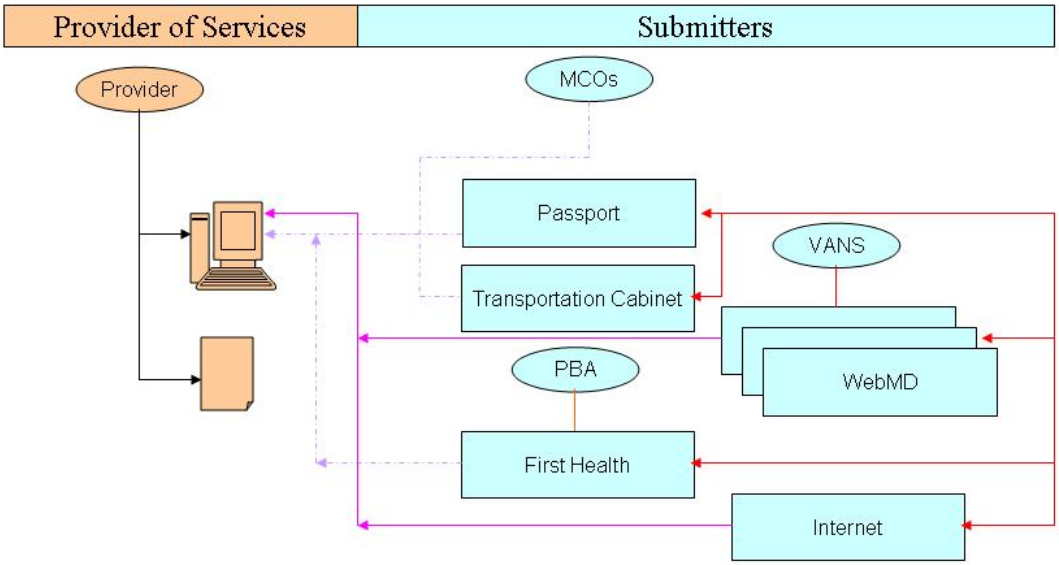


2.1.2 Claims System Flow

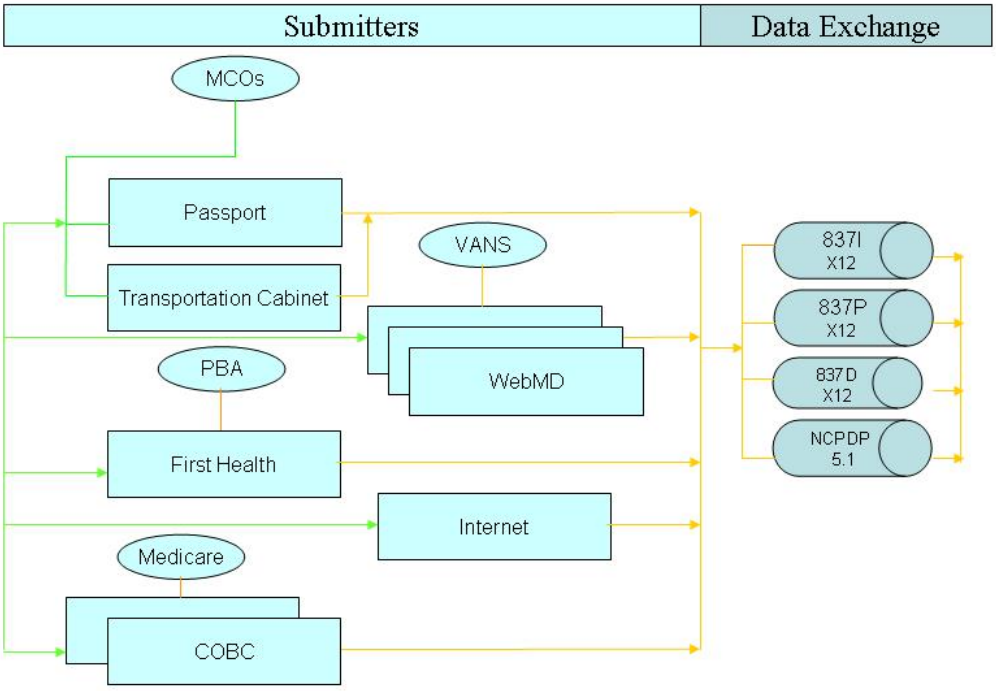
The diagram below shows the main flow of data into the Claims subsystem and processes that produce Claims outputs.

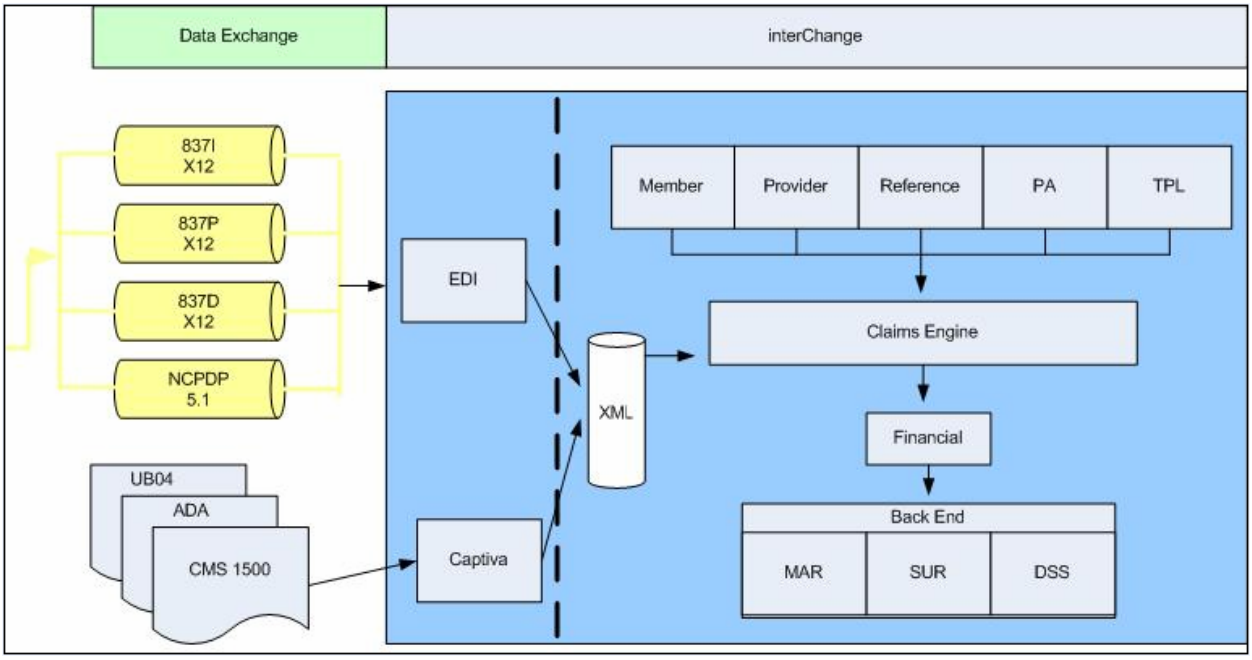


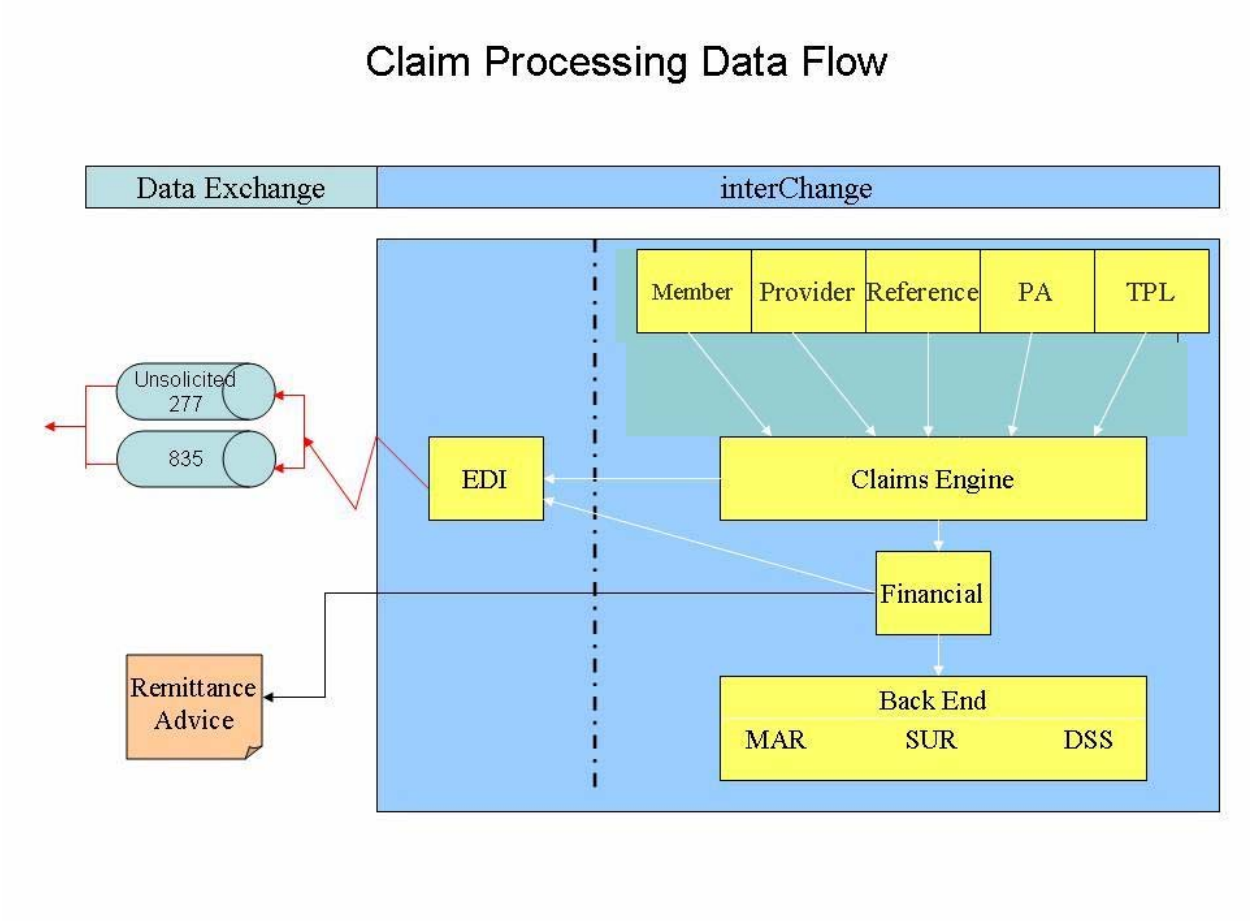
Claim Processing Data Flow

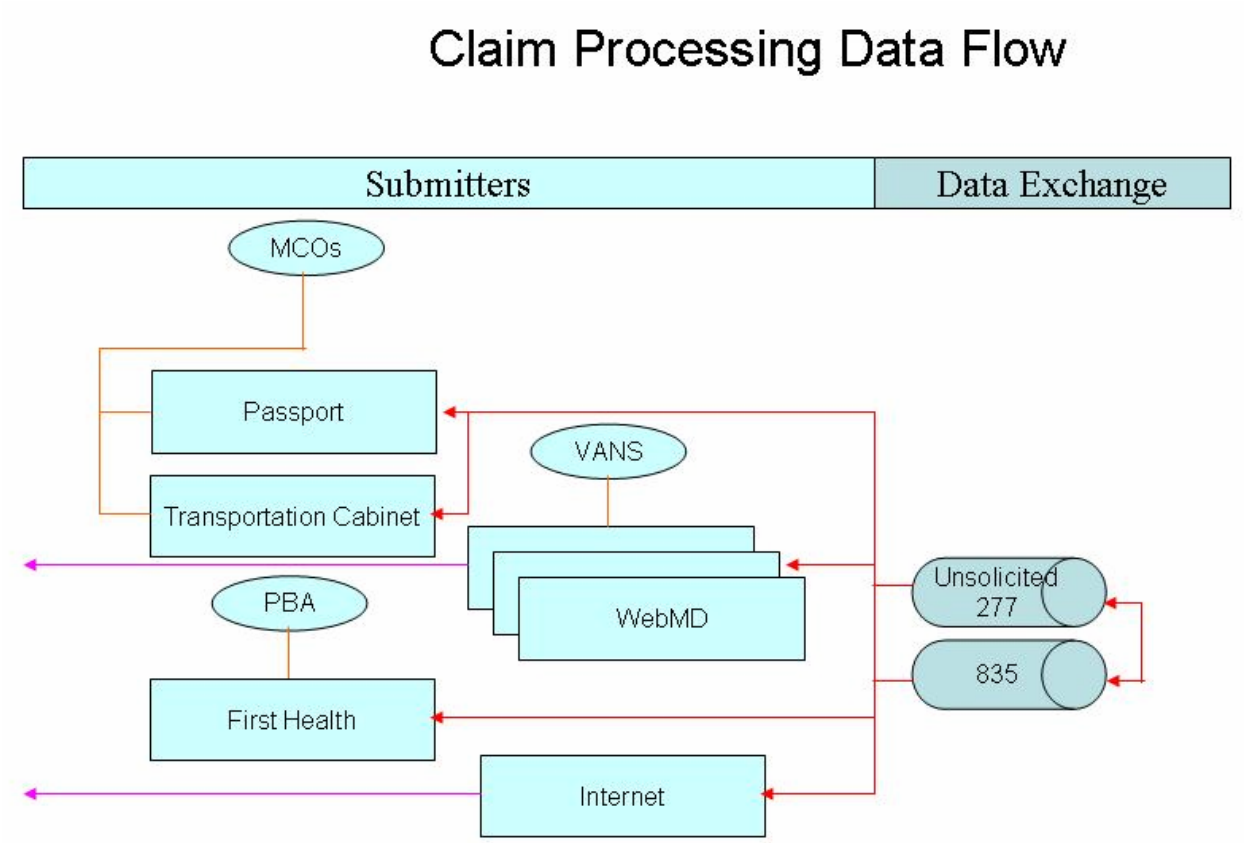


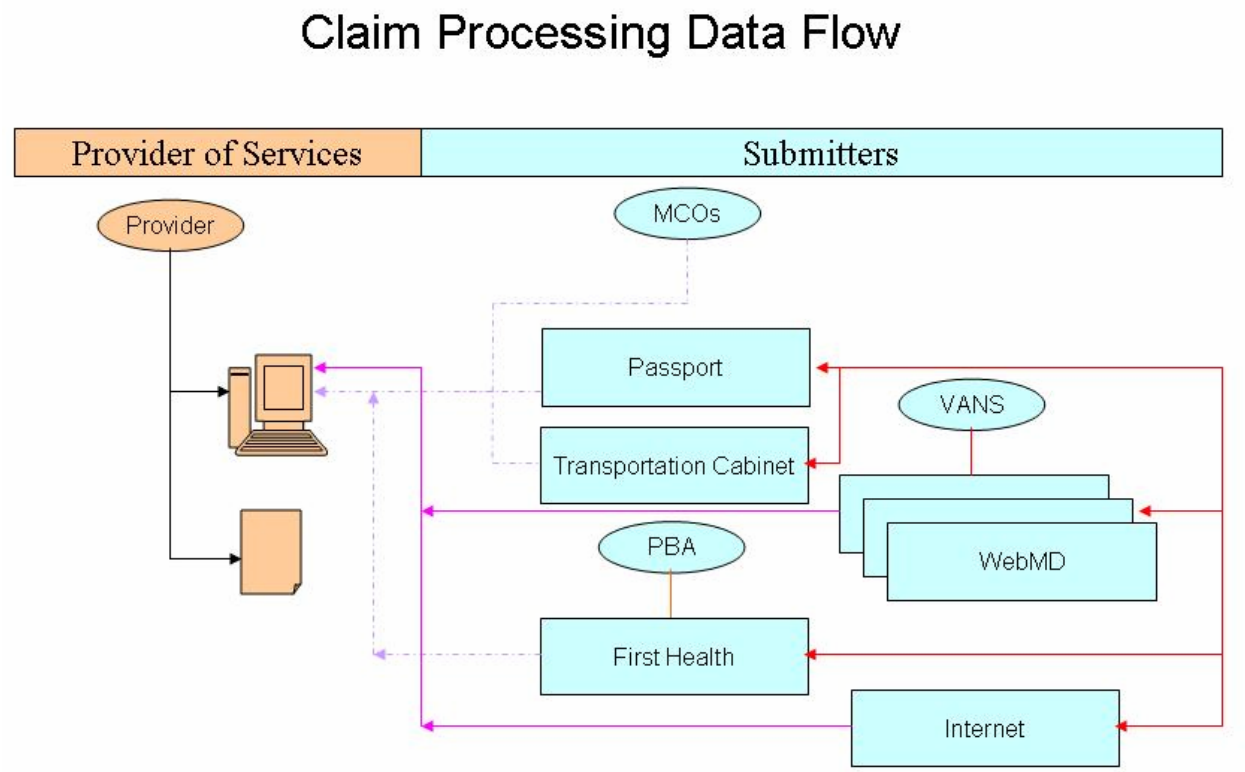
Claim Processing Data Flow



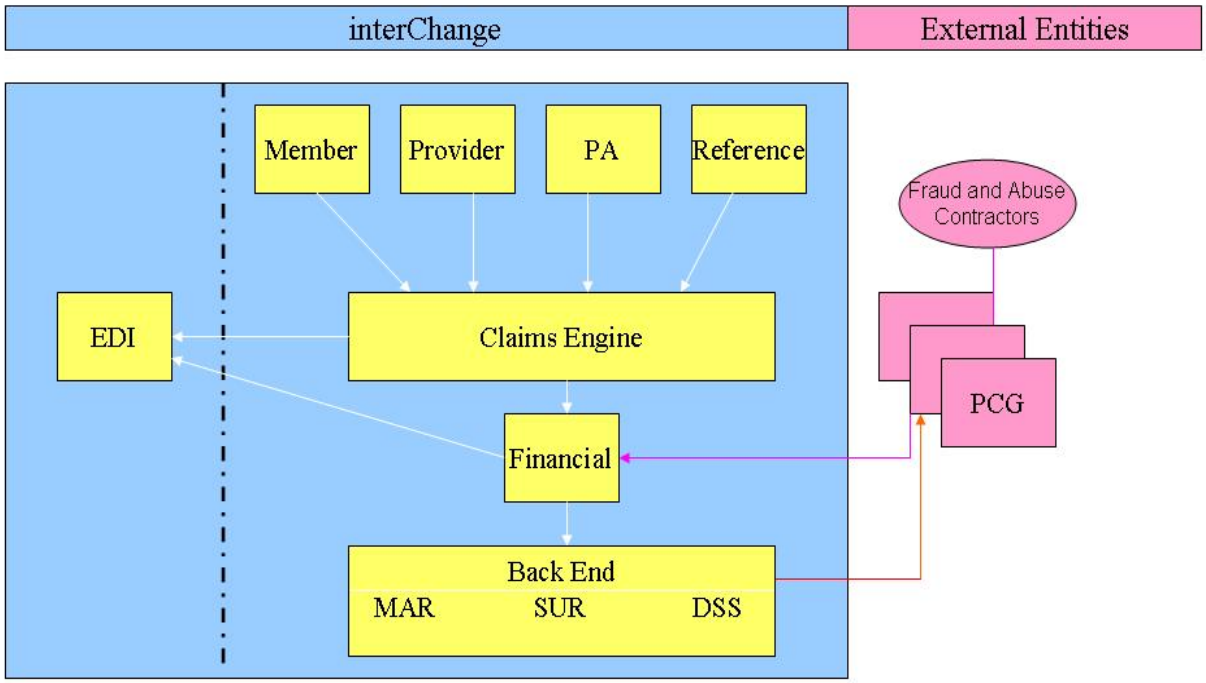



Claim Processing Data Flow
Data Exchange
interChange
Member
Provider
Reference
PA
TPL
Unsolicited 277
835
EDI
Claims Engine
Financial
Remittance Advice
Back End
MAR
SUR
DSS


Claim Processing Data Flow
Submitters
Data Exchange
MCOs
Passport
Transportation Cabinet
VANS
WebMD
PBA
First Health
Internet
Unsolicited 277
835


Claim Processing Data Flow
Provider of Services
Submitters
Provider
MCOs
Passport
Transportation Cabinet
VANS
WebMD
PBA
First Health
Internet

Claim Processing Data Flow



2.2 Subsystem Description

2.2.1 Introduction to Claims

Claims and Encounter processing functions ensure claims for eligible members, received from enrolled providers for covered services, are accurately processed and adjudicated in accordance with State and Federal requirements. The Claims processing function encompasses the tracking and processing of claims transactions up through adjudication. Data from Reference, Provider, Member, Service/Prior Authorization, Third-Party Liability (TPL), and Claims History is used in processing claims. The Encounter processing function encompasses the receipt, data validation, and processing of encounters. The data used in claims processing is also used for encounter validation.

The interChange system provides a user-driven and maintainable claims processing environment. Data elements owned by interChange are captured and stored with a single key name in one physical location within the database, and are accessed by all the other interChange processes. A majority of the claims processing, such as edit and auditing, claims pricing, claim disposition, and claim adjudication, are completed in their logical entirety for each claim processed. These activities occur concurrently, with many claims processed simultaneously with continuous claims processing.

The MMIS Claims Processing function includes:

- CLAIMS ENTRY ensures the accuracy, reasonableness and integrity of interChange entered data;
- CLAIMS RECEIPT AND CONTROL ensures all claim records are captured at the earliest possible time and in an accurate manner;
- EDIT/AUDIT PROCESSING ensures that claim records are processed in accordance with state policy;
- CLAIMS PRICING calculates the payment amount for each service according to the rules and limitations applicable to each claim type, service, and provider type;
- CLAIMS RESOLUTION is to support the correction of suspended claims;
- POINT OF SALE (POS)/PROSPECTIVE DRUG UTILIZATION REVIEW(PRODUR) provides for the on-line processing of pharmacy claims submitted in real-time by pharmacists and prevents the dispensing of inappropriate drugs through direct intervention. The Commonwealth of Kentucky will not be using this functionality. They will be using a Pharmacy Benefit Manager/Administrator (PBA);
- ADJUSTMENT PROCESSING supports the adjustment of adjudicated claims;
- CLAIMS DISPOSITION controls the outcome of each edit and audit posted to a claim based on user-defined criteria;
- CLAIMS REPORTING supports user review and management of claim processing functions; and,
- CLAIM CHECK provides enhanced claim editing capability using third party correct coding software.

2.2.2 Claims Entry

The Claims Entry functional area provides the processes for receiving, collecting, tracking, and reporting of claims data submitted to the inter*C*hange. It functions as the interface to the inter*C*hange, and facilitates the consistent and accurate presentation of claim data to the inter*C*hange for processing to adjudication. Claims Entry encompasses both hard copy paper and electronic claim submission via multiple media types.

External entities such as providers, service bureaus, counties, vendors, Value Added Networks (VANs), and other DMS-approved billing entities submit claims using the media most convenient to them.

The processing within the Claims Entry functional area includes acceptance of standardized claim formats, performance of high level editing on selected data elements for validity, consistency and format, addition of data elements to claims as required for Kentucky claims processing, and outputting files for acceptance by the inter*C*hange.

The tracking and reporting of received claims facilitates the identification of all claims received and passed to the inter*C*hange for processing, and facilitates timely and accurate claims adjudication. The permanent storage of all claims received through the Claims Entry functional area is accomplished using imaging technology that creates a permanent “snapshot” record of claims data as it is received by the entry function.

Claims Entry processes support objectives to receive claims and associated attachments through multiple standardized formats and multiple media. Diversity in the data entry functions is supported to facilitate the many programs administered, as well as to facilitate provider automation in claims submission activities. Claim data and entry statistics are produced and available for tracking, analysis, and review via Claims Entry-generated reports that are available through the OnBase storage and retrieval component of the inter*C*hange.

2.2.2.1 Paper Claims Entry

Paper claims are submitted on the following claim forms:

- CMS 1500;
- ADA 2006;
- ADS-1 Proprietary format; and,
- UB04.

The Paper Claims Entry function is based on data capture applications and services. The system utilizes high speed scanners, optical character recognition (OCR) processors, and specialized reject and repair workstations to complete the capture of data from approved paper claim documents.

Claim documents and attachments are scanned to capture the data from defined claim fields, and then proceed through the workflow services.

Accuracy is enhanced using field editing techniques to temporarily stop a claim from processing for manual validation or intervention. Extract files from the inter*C*hange are created and loaded into the paper claims entry application, against which entered data is validated. The definition of

valid values and other business rules for each specific field is established to define the systematic and manual processes within the data capture workflow.

Tracking claims throughout each processing step in the workflow, reporting on processing statistics, claim entry processing status, balancing, and claims sent to the interChange for processing, are an integral part of the paper claims entry processes. Upon completion of the data capture process, the paper claim data is formatted into a common claim format for processing by the interChange.

Claim images captured through this process are saved to the storage and retrieval system for access by interChange system users. An image of every claim and the associated attachments received is captured to provide a permanent retrieval record of all paper claims received.

2.2.2.2 Electronic Claims Entry

Electronic submission media includes:

- Magnetic (tape, diskette, and compact disc);
- Direct submission from a computer through a modem;
- FTP (file transfer protocol);
- POSNet; and
- Internet.

Electronic fee-for-service and encounter claims and adjustments are accepted in the HIPAA compliant X12 837 4010A:

- 837 Dental (837D);
- 837 Professional (837P);
- 837 Institutional (837I); and
- NCPDP 5.1 Pharmacy.

The Electronic Claims Entry function allows providers and other billing entities to submit electronic claims and encounter data using standardized communication protocols and transaction sets. To allow for maximum flexibility, the interChange accepts electronically submitted claims and encounter data via magnetic tape, diskette, CD-ROM, point-of-service (POS), via file transfer protocol (FTP), POSNet, and through the Internet.

The Health Insurance Portability and Accountability Act (HIPAA) ASC X.12 837 4010A electronic formats are an integral part of the Claims Entry function.

The interChange accepts and receives electronic media claims data through the EDS Business Exchange Services. The EDS Business Exchange Services is utilized to perform translation services, high level editing on specified fields, provide acknowledgement transactions to submitters, and to provide reports that document the acceptance and rejection of claims and files.

The EDS Business Exchange Services is accessed directly by approved billing entities or by EDS for electronic claims submitted directly to EDS (such as magnetic tape media claims).

Upon receipt of a claim file, the EDS Business Exchange Services sends an electronic acknowledgement or receipt, logs the claim data received for tracking and reporting within the interChange, and performs established high level edits. Following the completion of the edit function, reports are generated and sent back to the submitter with details of the submitted data.

The submitted claim data captured by the EDS Business Exchange Services is formatted into XML-structured content and forwarded to the interChange for processing. Electronic claims received by EDS are logged and transmitted to the EDS Business Exchange Services via file transfer protocol (FTP).

The same processes above are executed with the returned acknowledgements and reports files sent to EDS via FTP. Additionally, a daily extract file of all transactions received by the EDS Business Exchange Services is sent to interChange for tracking, reporting, and balancing within the interChange.

The interactive transmission and response form of electronic claims submission enables providers to enter individual claims using the Internet or POS device. This process submits a claim, processes it to final disposition, and returns an electronic response on that claim in a matter of seconds. This form of billing is primarily used in pharmacy claims processing, where it incorporates the Prospective Drug Utilization Review (ProDUR) into the claims process.

The specifics of this processing are discussed in detail in subsequent functional area Detailed System Design (DSD) documents, and are not addressed within the Claims Entry DSD.

All claims, regardless of media, are translated into a common file structure for the interChange system. The file structure used is Extensible Markup Language (XML) format. The XML file structure allows complex data structures, such as claims, to be communicated in a common format between different computer systems. For electronic claims, the EDS Business Exchange Services translator accepts ASC X12 HIPAA-compliant claims, and formats them into the XML file structures recognized by the interChange system. Paper claims are acquired through the Captiva data capture suite of products, and formatted into the XML data structures.

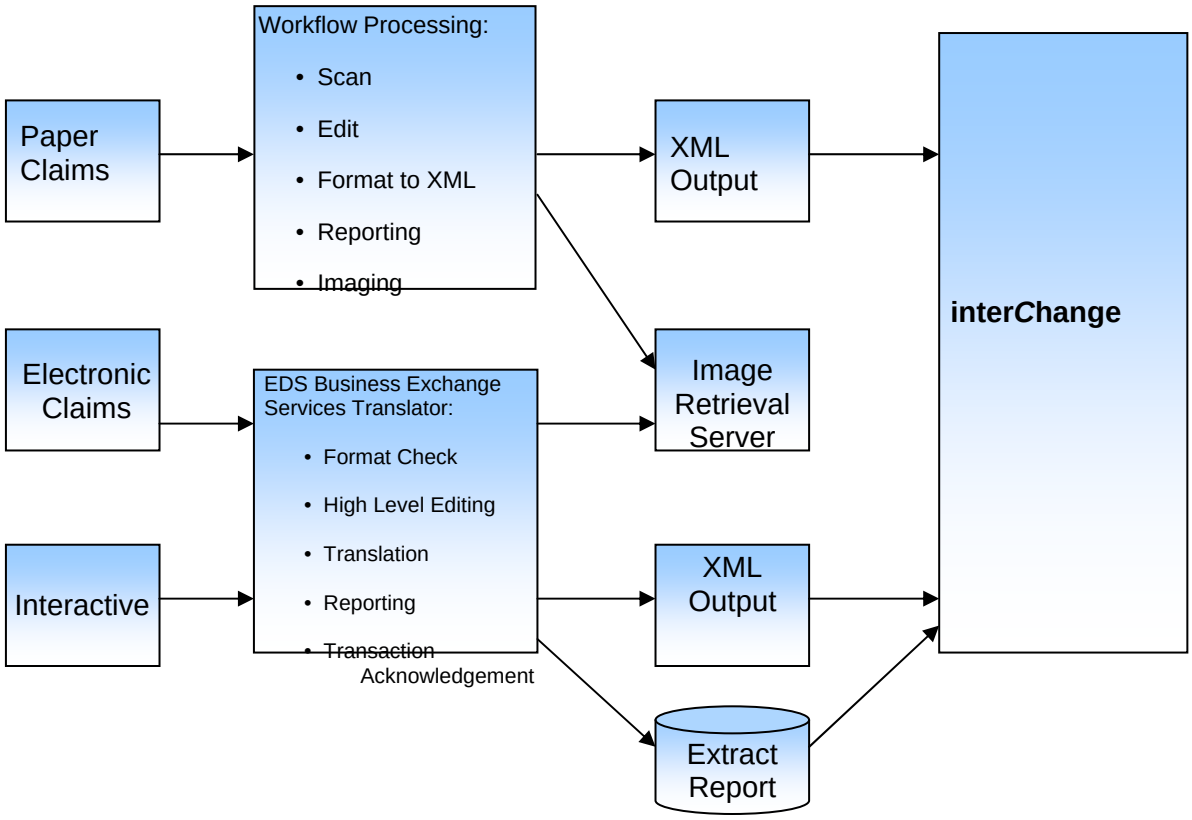
Pharmacy claims are adjudicated by the PBA. The PBA transmits these adjudicated claims as batch NCPDP transactions to interChange every other week as part of the PBA financial cycle. The NCPDP transactions are translated into the XML data structure. Claims submitted through the internet are transmitted to the interChange system directly in the required XML format.
These processes support the entry of fee-for-service claims, encounter claims, and claim adjustments and associated attachments.

Submitted claim data is electronically captured and imaged for permanent storage and retrieval purposes.

Every claim or encounter is received in a common format and processed through the claims adjudication system. All claims are assigned a unique internal control number (ICN) that identifies the claim throughout the Claims Entry and interChange processing activities. All electronic claims and encounter data are stored in a storage and retrieval sustem within interChange.

Data Model / Hierarchy Chart

Claims Entry Process Model



The Claims Entry functional area does not have its own data model, as it is really an interface into the interChange for external entities.

Claim transactions from the external community enters the interChange through the appropriate service for the media (paper, all forms of electronic 837 and NCPDP, and internet), which performs data translation, data editing, statistical reporting, and transaction routing to the appropriate interChange services.

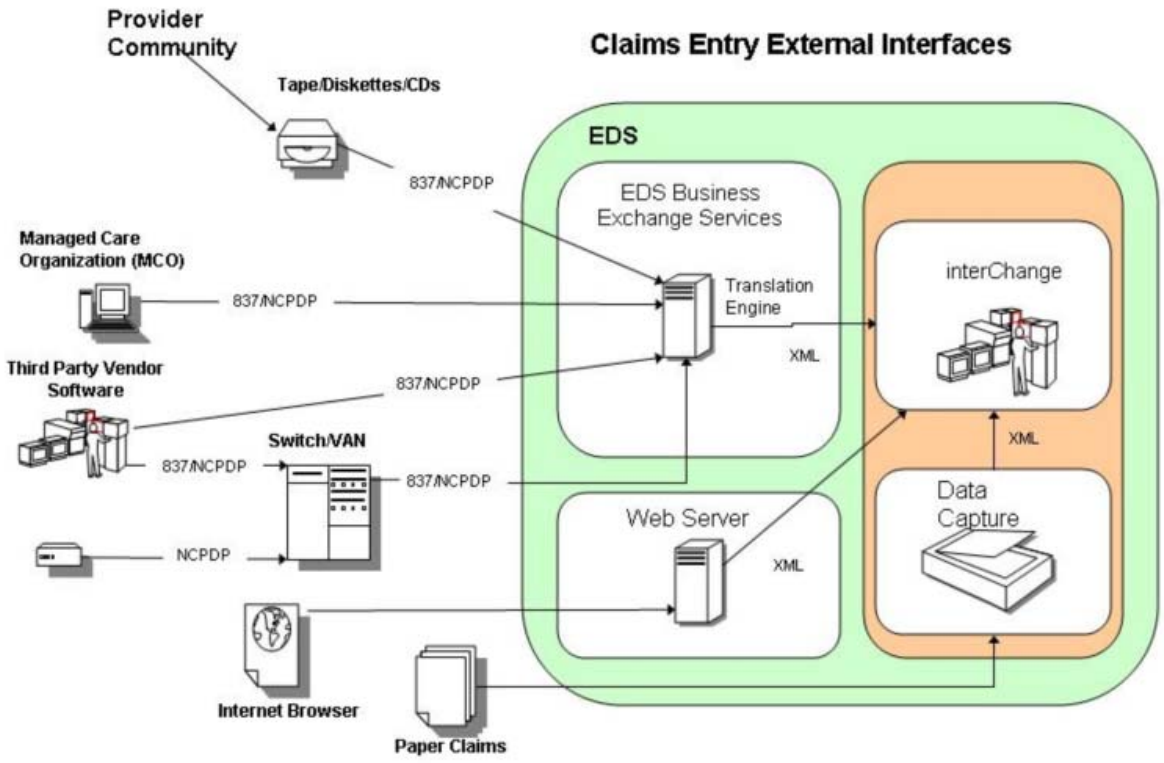
The interChange service accepts the transactions, perform the necessary processing, and update the interChange database, as necessary.

The following process model depicts at a high level, the Claims Entry interface with the interChange.

2.2.2.3 External Interfaces

The following is a context diagram that provides an overview of the external entities with which the Claims Entry Functional Area interfaces.

The diagram identifies the files and their format that are received and sent via the Claims Entry functional area.



2.2.3 Claims Receipt and Control

The Claims Receipt and Control function provides receipt and control of the following claim transactions and supporting documentation:

- New day fee-for-service and encounter claims;
- Claim Adjustments;
- Claim Attachments; and
- Prior authorizations (PA).

This functional area provides the control processes for tracking claims from their receipt into the system, through their addition to claims history when adjudicated. This functional area provides the following abilities:

- Assigns a unique tracking number to a claim within 24 hours of receipt.
- Identifies the media on which a claim was submitted.
- Identifies the status and current location of all claims.
- Associates paper claims with paper attachments.
- Associates electronic claims with paper attachments.

- Identifies missing claims. This is in reference to the Claims Captiva Received Rejected Report (CLM-0099-D), which compares the output from the paper claims data capture system to the input in the system. When this process identifies missing Internal Control Numbers (ICN), they are displayed on this report.
- Identifies the number of claims received and processed, and the media through which they were received.

Member-specific fee-for-service claims and adjustments are received from providers and processed for payment or denial. Encounter claims and adjustments are received from Managed Care Organizations (MCOs) and other approved encounter submission entities, and processed to adjudication for approval or rejection. Claim attachments are received, uniquely identified and tracked, and used as supporting documentation in the adjudication process for claims and claim adjustments. Attachments are received for both paper and electronically submitted claims.

As claims are accepted and entered into the interChange system they are tracked using various online pages and panels. These pages and panels allow users to identify the location and status of each record. Claim and encounter specific adjustments are received and processed for fee-for-service claims and encounter transactions that should be reprocessed due to a rate change, error, or some other reason.

2.2.3.1 Internal Control Number (ICN)

The Internal Control Number (ICN) assigned to each claim is unique from all other ICNs within the system, and is the identifying means through which each claim is processed, tracked, and reported. The ICN is comprised of multiple components that identify critical information about the claim for defined claims processing. The first two characters of the ICN represent the region code that identifies the source/region of the claim submission. The region code is used to identify paper claims, electronic claims, claim adjustments, pharmacy claims, converted claims, internet claims, and special handling claims. Converted claims are those claims that have been converted from the current Kentucky MMIS into the new interChange. These claims have been converted from the current format into the new system claim formats.

The ICN also represents the date the claim was received in the year and Julian date portion of the field. This facilitates control reporting of claims for the receipt date, as well as other downstream processing, such as edit disposition. The following table describes the ICN format:

2.2.3.2 ICN Format

Region Code RR	Year and Julian Date YYJJJ	Batch Number BBB	Claim Sequence SSS
99	99999	999	999

Although batch ranges can be assigned by claim type and region, the batch numbers are typically not restricted to allow the greatest possible number of possible ICN values per region and Julian date.

The claim type indication is integral to claims processing, and designates the interChange - recognized claim format within which a claim or encounter belongs. The interChange recognizes the following four claim formats and their associated claim types:

2.2.3.3 Institutional:

- Inpatient;
- Outpatient;
- Long Term Care;
- Inpatient Crossover;
- Outpatient Crossover; and
- Home Health.

2.2.3.4 Professional:

- Medical, including Maternity Care, MA Transportation, Family Planning and Waiver Services; and,
- Part B Crossovers.

2.2.3.5 Dental:

- Dental.

2.2.3.6 Pharmacy:

- Pharmacy and
- Compound.

2.2.3.7 Claim Type to Claim Type Code

The following table provides the Claim Type to Claim Type Code cross-reference:

Claim Type	Claim Type Description	Claim Type Code
Inpatient Crossover	Medicare Part A Inpatient Crossover claims	A
Professional Crossover	Medicare Part B Professional Crossover claims	B
Outpatient Crossover	Medicare Part B Outpatient Crossover claims	C
Dental	Dental Service Claims and Encounters	D
Home Health	Home Health Claims and Encounters	H
Inpatient	Institutional Inpatient Service Claims and Encounters	I
Long Term Care	Long Term Care Service Claims	L
Outpatient	Institutional Outpatient Service Claims and Encounters	O

Claim Type	Claim Type Description	Claim Type Code
Professional	Professional Medical Service Claims and Encounters	M
Pharmacy	Pharmacy Service Claims and Encounters	P
Compound Pharmacy	Pharmacy Compound Service Claims and Encounters	Q

As claims are accepted and entered into the interChange they are tracked through various online panels. These panels allow users to identify the location and status of each claim at any point in processing.

The interChange produces several reports that track claims and encounters received and processed. Reports are produced to track the number of claims, encounters, and adjustments received, and to track the type of media from which claim submissions are received.

2.2.4 Edit and Audit

Edit and audit processing in the interChange ensures that claim records are edited and audited in accordance with the claim processing policy. An Error Status Code is assigned to each edit and audit defined within interChange. The interChange Edit and Audit Processing function provides the following capabilities:

- Dynamic disposition of claims for edits and audits based on user-defined and maintained table-driven instructions;
- Development of distinct audit types based on inclusion and exclusion of different claim elements;
- Exemption of providers from edits and audits for a specified time period;
- Return of HIPAA-compliant adjustment reason and remarks codes to the provider, for claims setting edits and audits;
- Setting of edits and audits at either the header or detail level of the claim, depending on the claim type and criteria;
- Establishment of rules to indicate if the edit or audit can be overridden or denied;
- Setting of edits and audits for reporting that do not affect the processing outcome of the claim; and,
- Identification of the most common errors set on claims.

The edit function verifies the accuracy, validity, required presence, format, consistency, allowable values, and integrity of data submitted. This is done by comparing claim data against reference data, provider data, valid value lists, and member data. The audit function compares the data of a claim in process with other claim data in paid claims history to determine the appropriateness of the service reflected on the claim in relation to other services received by the

member. Verification for duplicate services, service limitations, and service conflicts are performed within the audit function.

The InterChange Edit/Audit processing function is provided through panels that are designed to allow a user to develop audit criteria based on the following claim data elements*:

- Member service program;
- Claim type;
- Provider type;
- Provider specialty;
- Member age;
- Procédure codes;
- Diagnoses codes;
- Revenue codes;
- Type of bill;
- NDC;
- Therapeutic class;
- Generic Price Indicators;
- GCN, and GCN sequence number; and
- HICL (hierarchal ingredient code list) sequence number.

NOTE: This is not an all-inclusive list of audit criteria.

This design permits lists of data elements to be built as the audit criteria. The system allows the data element definition to be either positive or negative, which is accomplished through the use of an inclusion or exclusion indicator on the audit definition.

Edits and audits are executed based on a claim's type. In addition to claim type specific edits, global edits (applicable to all claim types) are also executed. The Error Status Code assigned to edits and audits are posted to claims to indicate that a defined condition that can alter the expected outcome has occurred in the processing of the claim. The disposition status is determined based on table-driven instructions, which are accessed and maintained via online panels. These criteria include:

- ICN Region;
- Claim Type;
- Date of Service;
- Date of Receipt;
- Provider Type;

- Service Program;
- Provider ID; and
- Provider Specialty.

Error Status Codes are defined to set at either the claim header or the claim detail level. Disposition statuses associated with Error Status Codes include pay and list, deny, or suspend. Additionally, the suspense or system location is associated with an ESC to facilitate claim tracking and reporting.

All audits, except duplicate checking go through the Audit Maintenance panels. These panels allow new audits to be created without requiring a request for programming changes to the system.

The interChange system supports the following audit types:

- Duplicate Checking – This type of audit is used to minimize fraud by identifying claims for services that appear to have already been received and paid. This audit checks the claim header or detail, as appropriate for the claim type;
- Limitation – This type of audit is used to enforce policies on how often, or how much, can be spent for specific services over a period of time. Limitation audits can be set using a dollar amount, or by service unit limits using procedure code, revenue code, or NDC criteria;
- Contra-Indicated – This type of audit is used to enforce policies on the services that cannot occur if a specific service has already been performed. These audits are set when the service on a claim in process is illogical or in conflict with another service within the member's history. An example of this kind of audit is billing for a tooth restoration after the tooth has already been extracted;
- Negative Contra-Indicated – This type of audit is used to enforce policies regarding services that should occur together. These audits are set when the service on the claim in process requires a service to exist in history before it is paid;
- Umbrella – This type of audit is used to enforce policies regarding pre-operative and post-operative care. This audit is set when a claim is submitted for follow up care services with costs that are included within the payment of the surgery service from claims history;
- Bundling – This type of audit is used to identify billed services that are considered to be components of a global procedure that has already been paid;
- Unbundling – This type of audit is used to ensure that if a component of a global procedure has been paid in the past, the global procedure is not allowed;
- Conflict – This type of audit is used to identify conflicts between user specified criteria such as claim type, provider type/specialty, and member level of care, member age or other criteria. For example, if the user decides to identify LTC claims with dates of service that overlap with a claim for the same member for hospice, a conflict audit can be defined in the interChange to identify those claims; and,

- Step Therapy – This type of audit is used to enforce policies regarding the sequence in which a set of drugs can be administered to a member. Each drug in the program is assigned to one of five step therapy levels. Each level has criteria for the number of prescriptions, number of days, and continuous coverage. In order to step up to a higher-level drug, the member's drug history must have fulfilled the step therapy level's criteria requirements.

The inter*C*hange Edit/Audit function generates multiple daily or weekly reports that monitor claims processing activities. These reports demonstrate the occurrences of edits and audits among claim types, providers and provider types, claim ICN region codes, and claim source and media.

The Edit and Audit workgroup will define the criteria for each edit and audit implemented within the inter*C*hange, and has identified the disposition criteria.

2.2.5 Claims Pricing

The claims pricing function determines the pricing methodology used to calculate the amount paid for services rendered to eligible members. This function includes the calculation of the allowed amount and application of payment deductions and cost sharing requirements, based on member and claim criteria.

The allowable amount and the payment amount for each service billed are calculated according to the rules and limitations applicable to each claim type, provider type, service location, and benefit plan.

The pricing function determines the applicability of payment reductions such as co-pay, other insurance amounts, Medicare payment amounts, and patient payment/liability amounts, and applies these reductions to the calculated claim allowed amount to determine the claim payment amount, in accordance with policy.

Date-specific rates are maintained within the Reference and Provider rate tables in the inter*C*hange system to support multiple pricing methodologies.

The claims pricing function supports accurate pricing of the various claim types listed below, which are processed within the inter*C*hange system:

- Pharmacy Pricing – The system prices both compound and pharmacy (non-compound) drug claims. Both types of pharmacy claims are priced using the appropriate per unit rate for the NDC and date of service billed. The rate is multiplied by the number of units billed to determine the allowed charge. A dispensing fee is added to the allowed price calculated for the claim. This allowed charge is then typically compared to the billed amount, and the lower of the two amounts is determined as the claim's allowed charge. Deductions that may apply to these claims include third party insurance payments, and Medical Assistance Program co-pay amounts. These amounts are deducted from the claim's allowed charge to determine the amount to be paid on the claim.

- Family Planning Pricing – Family Planning claims are priced for eligible family planning services. Only certain approved procedure codes are eligible for reimbursement. Eligible services are priced using the pricing method established for professional claims, which is described below. Deductions for third party insurance payments and co-pays may apply.

- Professional Claim Pricing – Professional claims cover a wide range of services. These include transportation services, dental services, physician and nurse practitioner services, and lab services, to name a few. Professional claims are billed using procedure codes and procedure code modifiers. Pricing calculates the allowed amount for each service detail of the claim by multiplying the appropriate rate by the number of units allowed. In addition to the rate and number of units, the calculation of the detail allowed amount takes into consideration the member's benefit plan, the rate type associated with the service program, and the processing and pricing modifiers billed on the claim. After calculation of the initial detail allowed charge, pricing and processing modifiers are applied to increase or decrease the detail allowed charge based on an amount or a percentage, as defined within the user-maintained modifier panels.

- Long Term Care Claim Pricing – Long Term Care (LTC) claims are submitted by LTC facilities and reflect the full realm of services provided to the members who live within each facility. The LTC pricing methodology supports the payment of a provider-specific per diem amount for each covered day of facility care. Additionally, therapeutic leave days are paid up to a maximum number of days per calendar year, and hospital bed hold days are paid up to a maximum per period of hospitalization, as established. Patient liability responsibility amounts are deducted from the sum of the per diem allowed amounts for the facility, therapeutic leave days and hospital leave days. Reductions to patient liability amounts include health insurance premiums and other medical services, and must be submitted on the claim. The per diem rates on file are provider-specific. Third party payments are applied, as applicable.

- Inpatient Claim Pricing – Inpatient claims are priced using either a provider-specific per diem method, or the DRG pricing method. Pricing at a per diem rate applies to certain types of Rehab, Critical Access and Psych hospitals. Most other inpatient claims are priced based on the DRG pricing method, with some special case and exception logic applied. Pricing for Cost Outliers and Day Outliers are supported.

- Outpatient Claim Pricing – The majority of outpatient claims are priced based on the procedure code pricing method.

- Encounter Claim Pricing – Encounter claims are priced using fee-for-service pricing methods, when they can be determined. These claims store the calculated price on the claim but no payment is made to the provider. When the system is unable to establish a price for an encounter claim, the encounter is priced at zero.

- Crossover Claim Pricing – For Crossover claims, the fee-for-service allowed amount for the procedure and modifier code are calculated at the claim detail level. Subsequently, comparisons and calculations for coinsurance, deductible, and Medicare-approved amounts are made to determine a payable amount for a claim that does not exceed the lower of the co-insurance and deductible billed, or Medical Assistance fee-for service allowed amount. Deductions may apply.

Certain deductions may be applied to the computed Medical Assistance-allowed amount, depending on the claim and member criteria. These deductions include:

- Co-pay – This deduction is applied to claims, when applicable, and is based on defined policy;

- Patient Liability – The patient liability amount for a member is stored, and is applied during claims pricing based on the amount submitted on the claim. If the claim includes

a patient payment amount, this amount is subtracted in the calculation of the payment amount; and,

- TPL – Third Party Liability is applied based on the amount submitted on the claim. A TPL amount on the claim is subtracted in the calculation of the payment amount.

2.2.6 Claims Resolution

The Claims Resolution function in interChange ensures that suspended claim records are resolved in a timely manner, in accordance with policies and procedures. The Claims Edit/Audit function described above determines the errors to post to claims, which subsequently determines the disposition of claims.

Claims with errors that have a suspend status are placed in suspended claims processing locations for review and data correction. Data corrections managers, resolution users, and other authorized personnel have access to the suspended claims to perform data corrections online, through the data correction panels.

These panels provide the user with suspended claim information that facilitates the correction of erroneous data, or the manual disposition of the claim or claim detail, based on procedures established and documented in the Resolution Manual for the Commonwealth.

The Claims Resolution function in interChange provides the following abilities:

- Online resolution of suspended claims;
- Facilitation for the assignment and reassignment of suspended claims to a resolution user, or from one resolution user to another;
- Organization of manually suspended claims by moving them into separate user- defined suspense locations (sub-locations);
- Association of claim errors with the suspended claim;
- Access to claim-related data through electronic, pages and panels based techniques;
- Application of individual resolutions to claim (override or deny errors set);
- Application of global resolution to multiple suspended claims;
- Addition of HIPAA-compliant adjustment reason and remarks codes to claims;
- Access to historical claim-related data for duplicate and limitation audits; and,
- Complete reprocessing of suspended claims.

Pages and panels within interChange have been defined to support automated organization and management of suspended claims for resolution.

Managers use these pages and panels to define the criteria for automatically categorizing and assigning suspended claims to a specific resolution user, or, to allow claims to remain unassigned until a broader team determines priorities for their resolution.

Pages and panels are supported which allow a user to route multiple suspended claims that meet specified criteria to another claim location for review or data correction by a specific resolution user.

This functionality supports claim resolutions by users with in-depth knowledge of specific conditions or problems.

In inter*C*hange, resolution users employ the online panels to view all claims assigned to them for resolution. Suspended claim listings for each system-identified resolution user are provided to facilitate the data correction process.

Users view the suspended claims assigned to them, and select those to be resolved from the suspended claim list. When a suspended claim is selected from the list, the data corrections panels appropriate for that claim are opened so the claim data can be reviewed and the appropriate corrections applied.

Upon the completion of data corrections, the claim is resubmitted into the system for reprocessing. The functionality of the data corrections panels includes the display of all errors associated with a claim, and the associated disposition of each error.

The resolutions user is able to inquire on historical claim data related to the suspended claim, when applicable. Information that includes the historical claim ICN, claim detail number, provider, member, and reference information to assist in the resolution of the suspended claim is available from the data correction panels. Through the claim resolution function, an override may be applied to an edit, or a claim denial forced on an individual or global basis, in accordance with the Edit/Audit Resolution procedures established.

On occasion the resolution user may be required to add additional information to a suspended claim. The inter*C*hange allows the addition of HIPAA-compliant adjustment reason and remarks codes to communicate additional information about a suspended claim to the submitting provider. This information appears on the provider's remittance advice or 835 payment transaction.

The claim resolution function supports the capture of data corrections to claims so they may be completely reprocessed through the system. This includes the complete re-editing, auditing, and pricing of all claim data. When data corrections for a claim have been applied, the resolutions user changes the claim status to "resubmit" to mark the claim for reprocessing within system.

The inter*C*hange claim resolution processing generates operational reports both daily and weekly to demonstrate the number of occurrences of edits and audits by claim type, provider ID and type, and claim region codes (to distinguish fee-for-service from encounter claims). Reports are also generated that demonstrate resolution user activity related to the application of edit/audit force actions (override, deny) for the resolution of suspended claims. Other reports generated display the age of suspended claims in a suspense location, by claim type, for new and resubmitted claims. Reporting provides the ability to track and monitor claims and claim-related activity, from receipt through adjudication.

2.2.7 Point of Sale (POS)/Prospective Drug Utilization Review (ProDUR)

The Point of Sale (POS) function provides for the interactive processing of pharmacy claims submitted in real-time by pharmacists through networks provided by contracted switch vendors. The Commonwealth of Kentucky will not be using this functionality. They will be using a Pharmacy Benefit Manager/Administrator (PBA).

Functions of the POS system include pharmacy claim submission and reversal. In addition, new functionality provides for the interactive submission, inquiry and reversal of Pharmacy Prior Authorization (PA) requests.

The Prospective DUR (ProDUR) system incorporates direct access to the MMIS for the purpose of performing prepayment review of drug therapy before prescriptions are filled. The ProDUR processing will include screening for the following alerts:

- Drug/Pregnancy;
- Drug High Dose;
- Drug Low Dose;
- Drug/Disease;
- Drug/Age-Pediatric;
- Drug/Age-Geriatric;
- Drug/Drug;
- Therapeutic Duplication;
- Ingredient Duplication;
- Duration of Therapy;
- Underutilization; and
- Overutilization.

The ProDUR alerts will be prioritized and configured as either overrideable/non-overrideable according to State/Commonwealth Policy and the recommendations of the State/Commonwealth DUR board. Claims that set an overrideable ProDUR alert will require the pharmacist to resubmit the claim along with the appropriate intervention and outcome codes to override the alert and allow the claim to be paid.

2.2.7.1 Claim Adjustments

The inter*C*hange supports claim adjustments for previously adjudicated claims. There are many reasons to modify an original claim, including correction of an erroneous billing, repricing of a claim as the result of a rate change, or to reflect monies recovered against claims through the TPL recovery process.

The claim adjustment process includes creation of a new claim with the corrected or altered data, and a unique ICN region code that identifies the claim as an adjustment. At any time during claim processing activities, the actions taken on a claim can be determined.

The claims adjustment function provides the following capabilities;

- User-friendly interface to enter single and mass claim adjustment requests;
- Real-time claim adjustment retrieval and data correction for individual claim adjustments;
- Mass claim adjustment selection that identifies for adjustment processing all claims that meet user-selected claim criteria; and,
- Verification of claim adjustment results prior to committing claim adjustments to history.

Adjustments that are created through the screens or through the mass adjustment process are all recorded in the MMIS history and reported through out the MMIS. All the subsystems such

as Claims, Financial, and MAR contain special activities to support the adjustment process. These activities ensure accurate recording of claim alterations that come in the form of an adjustment.

There are some general rules which apply to all adjustments:

Only Paid claims can be adjusted;

Paid claims come from either the week-to-date or history database partitions (AIM00 & AIM01); and,

Adjustments pull information from the original Paid claim and store the adjustment information in the AIM00 Suspense tables (Voids go directly to AIM00 Denied).

The system only allows adjustments to the most recent version of a claim, and prevents adjustments to claims that are not yet adjudicated, including adjustments that are in process.

Special adjustment ICN regions have been defined within the inter*C*hange to identify the adjustment type when adjustments are entered. These regions allow someone viewing an ICN to quickly identify an adjustment, and whether it is an adjustment to an encounter claim or fee-for-service claim. Informational reason codes can be assigned to adjustment and mass adjustment claim requests to indicate the reason for the claim adjustment.

Internal review and approval guidelines have been established for mass claim adjustments. Mass claim adjustment requests are entered through an online adjustment request panel. This panel accommodates entry of the new adjustment ICN, the ICN of the adjudicated claim to be adjusted, and the determination of how the adjustment is to be processed, whether it is a full reprocess, detail level reprocess, or a history-only update.

The indication of whether the adjustment is to be verified before or after adjustment processing is also indicated to allow users to review the impact of the adjustment. Claims with mass adjustments process through the claims engine while applying changes that are captured through the adjustment, and then suspend for internal review and approval. An individual other than the initiator is required to approve the request, prior to final processing through the financial functional area for payment.

History-only adjustments are supported. An example of a history-only adjustment is the application of TPL collection results applied at the claim level. The receipt of payments from private insurance companies for medical assistance claims are applied to the claims for which payment is applicable. An RA is not produced as a result of an insurance collection, but the claim record is updated to reflect the private insurance payment.

Many other functional areas, such as Financial, Provider, and MAR, contain special activities to support the claims adjustment process. These activities ensure accurate recording of claim alterations in the form of adjustments.

2.2.7.2 Individual Claim Adjustments

Providers may initiate individual claim adjustments to correct an erroneous billing by submitting an adjustment claim to the inter*C*hange. Electronically submitted adjustments are processed systematically.

Those submitted on paper claim forms are entered into the system through the adjustment screens and processed through the claims engine as described above. As adjustments result in

the creation and processing of claims, the provider's remittance advice (RA) and 835 transactions contain the original claim data with the adjusted claim.

The RA demonstrates what happened to the provider's claims, financial transactions, and claim adjustment activities. Provider-initiated adjustments process completely through the system after initial entry and data correction.

Individual claim adjustments may also be initiated by the Commonwealth. These adjustments are entered into the system through the same screens used for provider-initiated adjustments. Processing is the same as through the claims engine.

2.2.7.3 Mass Claim Adjustments

There are situations when policy information has been altered retroactively, and therefore impacts the payment of claims in history with the potential need to reprocess a large number of paid claims.

The inter*C*hange has a mass adjustment function that allows adjustment users to identify all claims from history that are affected by a policy change, and make changes to either individual claims, or all of the affected claims.

The selected claims are reprocessed through the claims engine and the identified changes are applied to these adjustment ("daughter") claims.

These claims suspend for internal review and approval before final processing through the financial functional area.

With this mass adjustment capability, corrections and changes can be made to rates, and other claims data, and applied retroactively to claims in history.

The criteria for selecting claims for mass adjustment processing include:

- Dates of Service;
- Region Code;
- Claim Type;
- Provider ID;
- Provider Type & Specialty;
- Diagnoses Code;
- NDC;
- Gender;
- Age;
- DRG;
- Revenue Code;
- Service Program;
- Procedure Code & Modifier; and

- Member ID No.

2.2.7.4 Automatic Retro Rate Adjustments

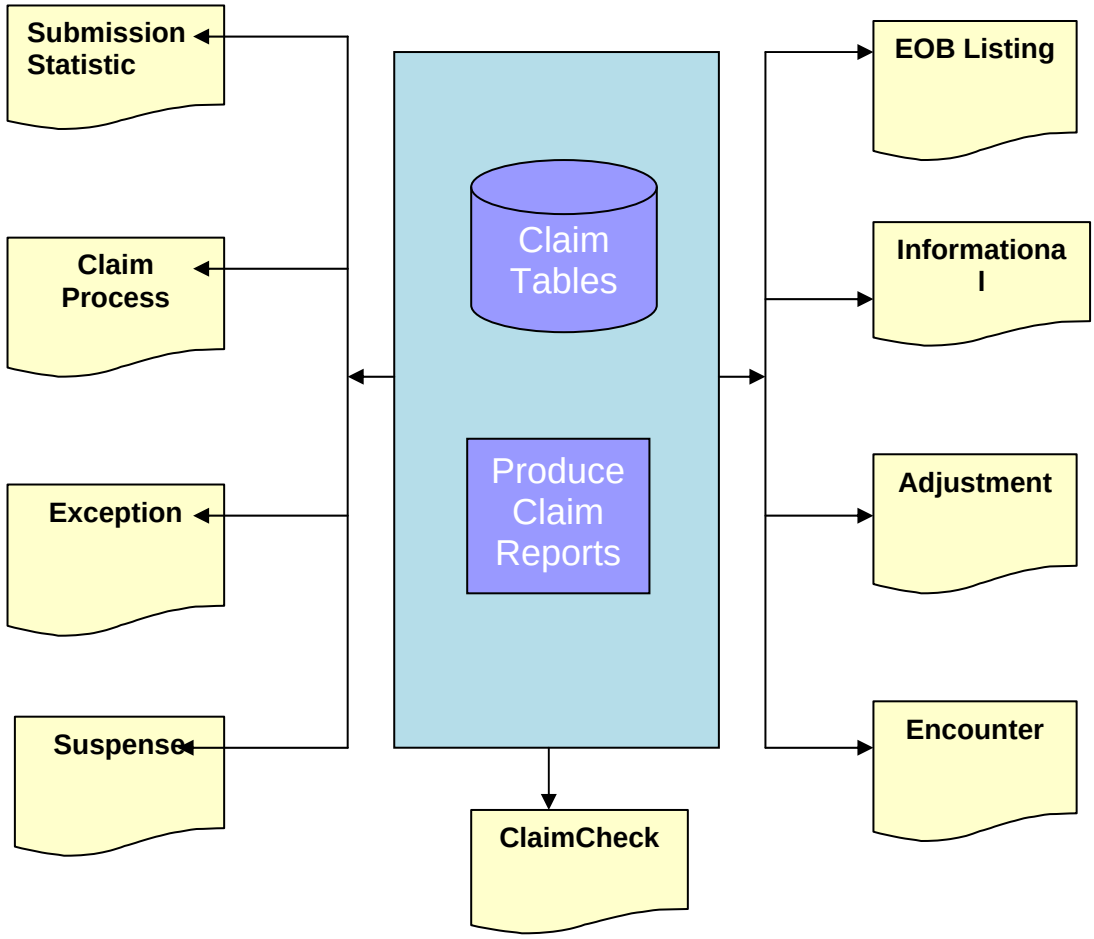
Retroactive rate changes to provider-specific per diem rates have been designed within the interChange to automatically create mass adjustment requests and select all claims from paid claims history that may be affected by a retroactive rate change. Claims are selected, and adjustment records created, for processing. These adjustment claims later suspend for internal review and DMS approval, prior to final processing through the financial functional area.

2.2.8 Claims Disposition

The Claims Disposition process does several functions. The primary functions include:

- Determining the disposition of an edit/audit;
- Determining the disposition of a header and detail(s);
- Determining the disposition of a claim;
- Determining the EOB associated to a claim from an edit/audit posted; and,
- Determining the location of a claim.

2.2.9 Claims Reporting



2.2.9.1 Diagram Abstract

The process flow diagram provides a visual representation of the claims functional area reports:

- Submission Statistic Reports are CLM-0130-D, CLM-0130-M and CLM-0130-W;
- Claims Process Reports are CLM-0016-D, CLM-0017-D and CLM-0027-D;
- Exception Reports are CLM-0022-D, CLM-0023-D, CLM-0024-D, CLM-0025-D, CLM-0026-D, CLM-0055-D and CLM-0056-D;
- Suspense Reports are CLM-0011-D, CLM-0020-D, CLM-0021-D, CLM-0031-M, CLM-0054-D, CLM-0060-D, CLM-0064-W and CLM-0065-D;
- EOB Listing Report is CLM-0012-D;
- Informational Reports are CLM-0046-M, CLM-0099D, CLM-5000-D, CLM-8600-W and CLM-9914-Q;
- Adjustment Reports are CLM-0010-D, CLM-0018-D, CLM-0050-M, CLM-0051-D, and CLM-0110-D;

- Encounter Reports are CLM-0123-D,CLM-0132-D, CLM-0132-M, CLM-0132-O; and CLM-0140 and,
- ClaimCheck Reports are CLM-0015-D and CLM-0058-Q.

All reports will be stored and accessible in OnBase. The report data will be extracted from the claim subsystem. The detailed layout of each report can be found in the report section of the DSD.

2.2.10 Claim Check

The Commonwealth of Kentucky will be using McKesson's ClaimCheck product for the enhanced editing solution for claims processing. Although ClaimCheck is a Commercial Off-The-Shelf (COTS) software product, it requires a customized integration with the MMIS.

The key considerations of the integration module involve the placement of the call to ClaimCheck, the conditions for making or bypassing the call to ClaimCheck for a given claim, the selection of history data to be sent and the method for handling recommendations returned from ClaimCheck.

2.2.10.1 Placement of the call to ClaimCheck

ClaimCheck will be called after edits, audits and pricing but before final pricing (application of other insurance, spend down, co pay and patient liability).

Placement of the call at this point will facilitate accurate calculation of cost savings.

2.2.10.2 Conditions for bypassing ClaimCheck

Only claim types O (outpatient) and M (physician) will be sent to ClaimCheck. Encounters will not be sent but adjustments will be sent to ClaimCheck.

Procedure codes excepted from ClaimCheck in the Commonwealth's current system will continue to be excluded.

McKesson will perform ClaimCheck on claim type D (dental). interChange will interface with McKesson sending paid dental claims. McKesson will provide enhanced editing reports to DMS for dental claims. Based on DMS analysis of these reports, DMS will request configuration of additional dental audits on interChange.

2.2.10.3 Smart History Pull

Only paid history details will be passed to ClaimCheck. Claims for the same provider, member and date of service will be selected.

Claims for the same provider and member with different dates of service will only be selected under the following conditions:

- The claim service being audited is a surgical procedure and the history service is an evaluation and management code occurring within 90 days before or after the surgical code's date of service;
- The claim service being audited is an evaluation and management procedure and the history service is a surgical procedure code occurring within 90 days before or after the E & M code's date of service; and,
- The history procedure is flagged as a once in a lifetime procedure.

2.3 Claims Processes

2.3.1 Claims and Encounters

2.3.1.1 Process Description

Claims and encounter records reflect services provided to the Commonwealth of Kentucky clients by health care providers, managed care contractors and the pharmacy benefit manager. Claim and Encounter processes receive, edit, pay fee-for-service (FFS) claims, and store health care service encounters. The Department of Medicaid Services (DMS) receives claims in both paper and electronic file formats. Encounters are received in electronic file transmissions from Managed Care Organizations (MCOs). Claims are received in electronic file transmissions, adjudicated using edit logic routines verifying the eligibility of the member, eligibility of the rendering provider, and eligibility of the service provided, to determine whether the claim is to be paid or denied. Encounters are subjected to edit logic routines verifying the eligibility of the member, provider, and service. Encounters are priced but do not make a payment. The encounter is stored in order to report on the services provided and received. interChange handles Pharmacy Benefit Administrator (PBA) pharmacy claims as encounters. The processes include:

- Receiving, editing, auditing, pricing, adjudicating to a pay, suspend or deny status, and storing fee for-service claim records;
- Receiving, editing, auditing, pricing and storing encounter records. Encounter records are accepted if edits thresholds are within tolerance;
- Generating claim payments for fee for service claims;
- Generating Remittance Advice(s) to explain the detail of paid and denied claims; and,
- Generating Unsolicited 277 transactions to return Encounter records that fail threshold edits.

2.3.1.2 Process Reference List

Connected via	Connected to	Source	Destination
Provider Claim	Provider (External Entity)		X
Claim Payment Information and Remittance Advice (RA)	Provider (External Entity)	X	
Medicare Intermediary Claims	Medicare Claims Processor (External Entity)		X
Passport Encounter	Managed Care Contractor (External Entity)		X
Transportation Cabinet Encounter	Managed Care Contractor (External Entity)		X
PBA Pharmacy Claim	Pharmacy Benefit Administrator (External Entity)		X

Connected via	Connected to	Source	Destination
Rejected Claim	Provider (External Entity)	X	
Rejected Medicare Intermediary Claim	Medicare Claims Processor (External Entity)	X	
Rejected Encounter	Managed Care Contractor (External Entity)	X	

2.3.2 Claims Processing

2.3.2.1 Process Description

Claims are received from Fee for Service providers either directly or through intermediaries. They are validated and, if error free, are paid according to Commonwealth established fee schedules. Data elements needed for adjudication including other insurance, co-pay, patient liability, deductible, and co-insurance payment information are received on claims records and stored. Following allowed amount calculation, any required co-pay, deductible, co-insurance, or other cost-sharing are reflected in the claim payment amount based on the benefit schedule. The claims are stored in history for later reporting and post-payment auditing.

2.3.2.2 Process Reference List

Connected via	Connected to	Source	Destination
Medicare Intermediary Claim	Medicare Claims Processor (External Entity)		X
Member Eligibility data	Member Data (Internal Entity)		X
Service/Prior Authorization data	KMAA (External Entity) through Prior Auth (Internal Entity)		X
Provider Adjustment Request	Provider of Services (External Entity)		X
Provider Claim	Provider of Services (External Entity)		X
Provider enrollment data	Provider Data (Internal Entity)		X
Reference data	Reference Data (Internal Entity)		X
Claim Payment and Remittance Advice	Provider (External Entity)	X	
Claim Payment Calculation and Remittance Advice (RA)	Provider of Services (External Entity)	X	

Connected via	Connected to	Source	Destination
Electronic Remittance Advice	Provider (External Entity)	X	
Medicare Intermediary Managed Care Claim	Passport (External Entity)	X	
Payment Authorization Transactions	Commonwealth MAR System(External Entity)	X	
Service/Prior Authorization utilization	KMAA (External Entity) through Prior Auth (Internal Entity)	X	
Rejected Claim	Provider of Services (External Entity)	X	
Rejected Medicare Intermediary Claim	Medicare Claims Processor (External Entity)	X	

2.3.2.3 Claims Processing

These are the reports and pages associated with Claims Processing for the Claims subsystem.

Technical Name	Title	Type
Clm.AdjustmentInformationPage.ascx	Claims Adjustment Information Page	Page
Clm.AdjustmentSearchPage.ascx	Claims Adjustment Search Page	Page
Clm.Assignments.acsx	Claims Assignment Page	Page
Clm.DataCorrectionPage.ascx	Claims Data Correction Page	Page
Clm.ClaimInformationPage.ascx	Claims Information Page	Page
Clm.MassClaimAdjustmentBasePage.acsx	Claims Mass Adjustment Information Page	Page
Clm.RelatedDataPage.ascx	Claims Related Data Page	Page
Clm.ClaimSearchPage.ascx	Claims Search Page	Page
CLM-0010-D	Online Adjustment Daily Report	Report
CLM-0011-D	Clerk ID Recycle Claims Report	Report
CLM-0012-D	EOB Listing	Report

Technical Name	Title	Type
CLM-0015-D	ClaimCheck Potential History to Adjust	Report
CLM-0016-D	Claims Processing Daily Summary	Report
CLM-0017-D	Original Claims Processing Daily Summary	Report
CLM-0018-D	Adjustment Credit Claims Processing Daily Summary	Report
CLM-0020-D	Aged Detail Suspense Report	Report
CLM-0021-D	Suspense Analysis by Error Code	Report
CLM-0022-D	Combined Claim Error	Report
CLM-0023-D	Claim Error Report	Report
CLM-0024-D	Daily Error Summary By Claim Type	Report
CLM-0025-D	Services to be Clerically Denied	Report
CLM-0026-D	Claims Paid Due To Force Report	Report
CLM-0027-D	Transaction Register	Report
CLM-0031-M	Suspense File Analysis by Claim Type 2	Report
CLM-0046-M	Potential Schedule 2 Narcotic Drug Abuse	Report
CLM-0050-M	Retro Active Rate Adjustment Report	Report
CLM-0051-D	Aged Detail Adjustment	Report
CLM-0054-D	Error Recycle Parameters	Report
CLM-0055-D	Daily Error Summary By Claim Type Original Claims	Report
CLM-0056-D	Daily Error Summary By Claim Type Adjustments	Report

Technical Name	Title	Type
CLM-0058-Q	ClaimCheck Error	Report
CLM-0060-D	Suspense File Analysis By Claim Type 3	Report
CLM-0064-W	Suspended Claims by Claim Type - Region	Report
CLM-0065-D	Aged Suspended Report	Report
CLM-0130-D	Claims Submission Statistics - Daily	Report
CLM-0130-M	Claims Submission Statistics - Monthly	Report
CLM-0130-W	Claims Submission Statistics - Weekly	Report
CLM-5000-D	Top Ten Claims Approved For Payment	Report
CLM-8600-W	Pended Claims Aging	Report
CLM-9914-Q	Abortions Procedure Report	Report

2.3.3 Receive Claim

2.3.3.1 Process Description

Claims are received from a variety of sources, including electronic, primarily via the internet and electronic data interChange (EDI) as well as paper formats. Claims are also received from Medicare intermediaries (crossover claims). Crossover claims for members enrolled in a managed care plan are sent to the managed care organization (MCO) directly by Medicare Coordination of Benefits (COB) intermediaries. Passport is the only MCO currently receiving crossover claims for processing. In turn, these claims are sent as encounters to the Medicaid Management Information System (MMIS).

System objects for the Receive Claim process are documented on the EDI and Internet subsystems.

2.3.3.2 Process Reference List

Connected via	Connected to	Source	Destination
Provider Adjustment Request	Provider of Services (External Entity)		X
Provider Claim	Provider of Services (External Entity)		X

Connected via	Connected to	Source	Destination
Medicare Intermediary Claim	Medicare Claims Processor (External Entity)		X
Medicare Intermediary Managed Care Claim	Passport (External Entity)	X	
Valid Format Claim	Adjudicate Claim (Process)	X	
Rejected Claim	Provider of Services (External Entity)	X	
Rejected Medicare Intermediary Claim	Medicare Claims Processor (External Entity)	X	

2.3.3.3 interChange Process Narrative

Electronic claims and adjustments are received in the HIPAA standard 837 claims format depending on the type of claim submitted (837 Dental, 837 Professional or 837 Institutional). Paper claims are received on standard claim forms, UB04 for institutional claims, CMS-1500 for professional claims and American Dental Association (ADA) 2006 form for dental claims. All electronic claims are translated by the EDI subsystem into an XML format and placed in a queue for the claims engine to process. Captiva captures paper claims via optical character recognition (OCR). If the claim does not pass the OCR process, they will be entered manually by the data entry operator. All paper claims are converted to XML format by the data capture subsystem. When ready the paper claims are moved to a queue and are processed by the claims engine. Paper adjustments are imaged by Captiva and assigned an adjustment ICN (internal control number). Users access the Claims Adjustment Information page to associate the imaged requests to an original claim.

Mass Adjustment claims are also internally generated based on user specified mass adjustment criteria and as a result of provider rate adjustments. Claims generated as a result of mass adjustment requests are adjusted claims.

2.3.3.4 Receive Claim

Objects associated with Receiving Claims for the Claims subsystem.

Technical Name	Title	Type
CLM-0130-D	Claims Submission Statistics - Daily	Report
CLM-0130-M	Claims Submission Statistics - Monthly	Report
CLM-0130-W	Claims Submission Statistics - Weekly	Report
EDI-0100-D	Interactive Activity by VAN - Daily	Report
EDI-0100-M	Interactive Activity by VAN - Monthly	Report
EDI-0100-W	Interactive Activity by VAN - Weekly	Report
EDI-0110-D	Interactive Activity Time of Day by VAN - Daily	Report

Technical Name	Title	Type
EDI-0120-D	Interactive Transaction Timing by VAN - Daily	Report
EDI-0120-M	Interactive Transaction Timing by VAN - Monthly	Report
EDI-0120-W	Interactive Transaction Timing by VAN - Weekly	Report
EDI-0200-D	Interactive Activity for WEB DDE - Daily	Report
EDI-0200-M	Interactive Activity for WEB DDE - Monthly	Report
EDI-0200-W	Interactive Activity for WEB DDE - Weekly	Report
EDI-0210-D	Interactive Activity Time of Day for WEB DDE - Daily	Report
EDI-9000-M	Billing Agent Provider Xref Report	Report
Clm.AdjustmentInformation Page.ascx	Claims Adjustment Information Page	Page
Clm.MassClaimAdjustment BasePage.acsx	Claims Mass Adjustment Information Page	Page

2.3.4 Validate Claim

2.3.4.1 Process Description

Claims are adjudicated following Commonwealth of Kentucky policy and Federal guidelines. Claims that suspend during processing are available for review and are held until removed by correction by the submitter or by recycling.

System objects for Validate Claim are documented in the Claims subsystem.

2.3.4.2 Process Reference List

Connected via	Connected to	Source	Destination
Member Eligibility data	Member Data (Internal Entity)		X
Service/Prior Authorization data	KMAA (External Entity) through Prior Auth (Internal Entity)		X
Provider Enrollment data	Provider Data (KMAA)		X
Reference data	Reference Data (Internal Entity)		X
Valid Format Claim	Receive Claim (Process)		X
Adjudicated Claim	Process Claim Payment (Process)	X	
Service/Prior Authorization	KMAA (External Entity) through Prior	X	

Connected via	Connected to	Source	Destination
utilization	Auth (Internal Entity)		
SHPS			

2.3.4.3 interChange Process Narrative

Claims are processed independently through validation routines that verify the accuracy of procedure coding, edit for dates of service, verify member eligibility, verify provider enrollment status and certification (where required), and validate a variety of other data elements that vary by type of claim submitted. In addition, each service is evaluated to ensure that it is payable under the benefit plan(s) in which the member identified on the claim is enrolled, and that it is payable under the reimbursement rules associated with the provider type and specialty of the provider of services. Policy edits and audits enforce Kentucky Medicaid policy and Federal guidelines regarding service limitations, duplicate billing, billing for contraindicated services, billing for conflicting services between the submitted claim and other claims in history, and between service lines on the submitted claim.

Claims that pass validation checks are priced to determine the allowed charge based on standard pricing methodologies according to Kentucky Medicaid guidelines. After all edits and audits have been performed and the claim has been determined to be eligible for payment, final pricing is performed to determine the amount to be paid by applying co-pay, patient liability, spend down amounts, other insurance payments and Service/Prior Authorization limits to the allowed charge according to Kentucky Medicaid policy.

In addition to the validation checks and audits performed within interChange, the interChange system is integrated with McKesson's ClaimCheck enhanced editing software. This software ensures that claims adhere to accepted billing practices and recommends bundling of service lines where appropriate.

2.3.4.4 Process Control

Claims released to claims engine are processed through validation edits, pricing, audits, enhanced editing and final pricing. One of three outcomes is possible: the claim is ready for payment; payment is denied; the claim is suspended to a specific location for review and correction.

2.3.4.5 Validation

Validation is a significant function of the claims engine. Claims are validated through the use of integrity edits and audits. Edits and audits are supported through a large number of programs that are linked together as one executable called the claims engine. Execution of the edits and audits are controlled by driver programs within the claims engine based on controls established using claims and reference pages and panels.

The "Error Disposition Criteria" panel allows authorized users to control the disposition status of edits and audits. The disposition is controlled by a variety of fields including: claim type, region, provider specialty, benefit plan, and many others. For claim processing, the valid disposition status values are: pay, deny, list and pay, suspend, or super suspend. A claim may encounter multiple errors at the claim header level or on the same service line that have different disposition status values. A status code of super suspend always causes the claim to super suspend even when another error on the same claim service line has a disposition of deny. A deny status takes precedence over suspend or pay. A suspend status takes precedence over

pay. For example, if a claim has set errors with a disposition status of deny as well as an edit whose disposition status is super suspend, the claim would suspend for review. But, if a claim has set errors with a disposition status of deny as well as an error whose disposition status is suspend, the claim would deny.

Each error is associated to an Explanation of Benefit (EOB). The EOB code appears on the Remittance Advice (RA) and is associated to the claim. The RA is described in the Financial subsystem on both the Kentucky project workbook and GSD/DSD.

Reference pages such as: “Reference ICD-9-CM Procedure Information”, “Reference HCPCS Procedure Information”, “Reference Modifier Information”, “Reference Revenue Information”, “Reference Diagnosis Information”, “Reference Related Data Codes” are all used in controlling data integrity and applying policy edits.

The “Error Disposition - Audit Criteria” page is activated when an error is defined as an audit. The page contains a number of panels used to set parameters for a specific audit. Limitation audits, contra audits, and umbrella audits each have a separate audit parameters panel to capture required audit criteria.

Member and provider edits check the integrity of the IDs submitted on the claim. Additional information is obtained based on the date of service and member ID and Provider Number for policy editing. For example, the member information third party liability (TPL) is checked to determine if some other payer should have been billed prior to submitting the claim to Medicaid.

There are numerous relationship edits performed on the claim header and details. Some of them are generic and are therefore encapsulated in a common routine that contains functions called for all claim types. An example of an edit that is applied to all claims is the timely filling edit. Claims must be received from Providers within one year of service being rendered. All of the edits are described under the Claims Edit/Audit Processing section of the project workbook and DSD. Other edits are specific to a claim form such as dental editing. These functions are located in claim form specific source code.

2.3.4.6 Validate Claim

These are objects associated with Validating Claims for the Claims subsystem.

Technical Name	Title	Type
Clm.ClaimInformationPage.ascx	Claims Information Page	Page
Clm.ClaimSearchPage.ascx	Claims Search Page	Page
Clm.DataCorrectionPage.ascx	Claims Data Correction Page	Page
Clm.RelatedDataPage.ascx	Claims Related Data Page	Page
Error Disposition - Audit Criteria Page	Error Disposition - Audit Criteria Page	Page
Error Disposition - Audit Restriction	Error Disposition - Audit Restriction Page	Page
Error Disposition Information Page	Error Disposition Information Page	Page
Error Disposition Search page	Error Disposition Search page	Page
Reference DRG Information Page	Reference DRG Information Page	Page

Technical Name	Title	Type
Reference DRG Search Page	Reference DRG Search Page	Page
Reference Drug Search	Reference Drug Search Page	Page
Reference ICD-9-CM Procedure Information	Reference ICD-9-CM Procedure Information	Page
Reference Related Data Codes	Reference Related Data Codes	Page
Reference Related Data Xref	Reference Related Data Xref	Page
Reference Revenue Information Page	Reference Revenue Information	Page
Reference Revenue Search	Reference Revenue Search	Page
Reference.DiagInfoPg	Reference Diagnosis Information Page	Page
Reference.DiagSearchPg	Reference Diagnosis Search Page	Page
Reference.DrugInfoPg	Reference Drug Information Page	Page
Reference.HCPCSProcedureInformation	Reference HCPCS Procedure Information Page	Page
Reference.ICD9ProcSearchPg	Reference ICD-9-CM Procedure Search Page	Page
Reference.ModifierInfoPage	Reference Modifier Information Page	Page
Reference.ModifierSearchPage	Reference Modifier Search Page	Page
Reference.ProcSearchPg	Reference HCPCS Procedure Search Page	Page
Reference.Related Data Other	Reference Related Data Other	Page
CLM-0010-D	Online Adjustment Daily Report	Report
CLM-0011-D	Clerk ID Recycle Claims Report	Report
CLM-0012-D	EOB Listing	Report
CLM-0015-D	ClaimCheck Potential History to Adjust	Report
CLM-0016-D	Claims Processing Daily Summary	Report
CLM-0017-D	Original Claims Processing Daily Summary	Report
CLM-0018-D	Adjustment Credit Claims Processing Daily Summary	Report
CLM-0020-D	Aged Detail Suspense Report	Report

Technical Name	Title	Type
CLM-0021-D	Suspense Analysis by Error Code	Report
CLM-0022-D	Combined Claim Error	Report
CLM-0023-D	Claim Error Report	Report
CLM-0024-D	Daily Error Summary By Claim Type	Report
CLM-0025-D	Services to be Clerically Denied	Report
CLM-0026-D	Claims Paid Due To Force Report	Report
CLM-0027-D	Transaction Register	Report
CLM-0031-M	Suspense File Analysis by Claim Type 2	Report
CLM-0051-D	Aged Detail Adjustment	Report
CLM-0054-D	Error Recycle Parameters	Report
CLM-0055-D	Daily Error Summary By Claim Type Original Claims	Report
CLM-0056-D	Daily Error Summary By Claim Type Adjustments	Report
CLM-0058-Q	ClaimCheck Error	Report
CLM-0060-D	Suspense File Analysis By Claim Type 3	Report
CLM-0064-W	Suspended Claims by Claim Type - Region	Report
CLM-0065-D	Aged Suspended Report	Report
CLM-5000-D	Top Ten Claims Approved For Payment	Report
CLM-8600-W	Pended Claims Aging	Report

2.3.5 Process Claim Payment

2.3.5.1 Process Description

Claims that pass validation, (i.e. ‘clean claims’) adjudicate to the payment step. In the payment process the Financial subsystem summarizes claims payments together with expenditures and recoupment amounts applied to open accounts receivables to determine a payment amount for each provider with activity in the payment cycle. The calculated payment amounts are transmitted to the Commonwealth’s Statewide Accounting System MARS for payment via check or Electronic Funds Transfer (EFT). “Paid, denied and suspended claims are reported on the Remittance Advice sent to the provider to explain the processing results. The explanation of benefits (EOB) assigned during claims processing is the primary means for communicating processing outcomes to providers on the Remittance Advice. Providers may also choose to receive an electronic remittance advice (ERA). EOB codes are not allowed on the 835

remittance advice format, so each EOB is mapped to a HIPAA adjustment reason code for reporting on the ERA.

System objects for Processing Claim Payments are documented in the Financial subsystem.

2.3.5.2 Process Reference List

Connected via	Connected to	Source	Destination
Adjudicated Claim	Validate Claim (Process)		X
Claim Payment and Remittance Advice	Provider (External Entity)	X	
Electronic Remittance Advice	Provider (External Entity)	X	
Payment Authorization Transactions	Commonwealth MAR System(External Entity)	X	

2.3.5.3 interChange Process Narrative

The interChange claims engine uses a variety of payment methods for calculating payments for a claim. The Claims Pricing function calculates the payment amount for each service according to the reimbursement, billing rules, and limitations applicable to each claim type, benefit plan, service, and provider type and/or specialty. This process takes into consideration factors such as the Medicaid allowed amount, billed amount, Third Party Liability (TPL) payments, Medicare payments, and any cost-sharing requirements.

The interChange financial payment cycle pays claims on a weekly basis unless a special cycle is requested. The approved-to-pay and denied claims are combined with other financial transactions for the week and a total payment amount is calculated per provider. The detailed transactions are formatted into a remittance advice sent to the provider.

The payment requests are passed to the Commonwealth's Medicaid Accounting and Reporting System (MAR) which issues the payments. All claims finalized during the financial payment cycle are moved from week-to-date to finalized history.

2.3.5.4 Process Claim Payment

These are reports and pages associated with Processing Claim Payments for the Claims subsystem.

Technical Name	Title	Type
Financial Payment Hold Information	Financial Payment Hold Information	Page
Financial Payment Information	Financial Payment Information	Page
Financial Payment Search	Financial Payment Search	Page
Financial Schedule Information	Financial Schedule Information	Page

Technical Name	Title	Type
Financial Withholding Information	Financial Withholding Information	Page
Financial Withholding Search	Financial Withholding Search	Page
Remittance Reports	Remittance Reports	Reports

2.3.6 Encounters Processing

2.3.6.1 Process Description

Encounters are received regularly from Passport, the Transportation Cabinet, and the Pharmacy Benefit Manger based on their payment cycles. They are validated, priced, and stored in the week-to-date, then moved to finalized history for later reporting and auditing. Encounter claims describe services provided to Kentucky Medicaid members by Managed Care Organizations, including co-pay, deductible, and co-insurance payment information. Data received on encounter records is validated and stored for service utilization analysis and reporting.

2.3.6.2 Process Reference List

Connected via	Connected to	Source	Destination
CMS1500 Encounter	Managed Care Contractor (External Entity)		X
Dental Encounter	Managed Care Contractor (External Entity)		X
Pharmacy Encounter	Managed Care Contractor (External Entity)		X
UB04 Encounter	Managed Care Contractor (External Entity)		X
PBA Pharmacy Claim	Pharmacy Benefit Administrator (External Entity)		X
Log Encounter Error	Encounter Batch Control (Data Store)	X	
Store Encounter	Encounter (Data Store)	X	
Store Encounter Batch Control	Encounter Batch Control (Data Store)	X	
Return Encounter Errors	Encounter Batch Control (Data Store)	X	

2.3.6.3 Encounters Processing

These are the reports and pages associated with Encounters Processing for the Claims subsystem.

Technical Name	Title	Type
Clm.EncounterDataPage.ascx	Claims Encounter Page	Page
Clm.ClaimInformationPage.ascx	Claims Information Page	Page
Clm.ClaimSearchPage.ascx	Claims Search Page	Page
CLM-0132-D	Encounter Submission Statistics – Daily	Report
CLM-0132-M	Encounter Submission Statistics – Monthly	Report
CLM-0132-O	Encounter Submission Statistics – On Request	Report

2.3.7 Receive Encounter

2.3.7.1 Process Description

Encounter data is received regularly in electronic transmission from the managed care contractors and the pharmacy benefit administrator using national HIPAA 837 and NCPDP 5.1 format.

Additional system objects for receiving encounters are documented in the EDI subsystem.

2.3.7.2 Process Description List

Connected via	Connected to	Source	Destination
CMS Encounter	Managed Care Contractor (External Entity)		X
Dental Encounter	Managed Care Contractor (External Entity)		X
UB04 Encounter	Managed Care Contractor (External Entity)		X
Pharmacy Encounter	Managed Care Contractor (External Entity)		X
PBA Pharmacy Claim	Pharmacy Benefit Administrator (External Entity)		X
Encounters Received	Validate Encounter (Process)	X	
Store Encounter	Encounter (Data Store)	X	
Store Encounter Batch Control	Encounter Batch Control (Data Store)	X	

2.3.7.3 interChange Process Narrative

Encounter data is received on a weekly to monthly basis for original submissions and on request for resubmissions in electronic transmissions from MCOs and the PBA using national standard formats (HIPAA 837 and NCPDP). Transmissions are received via a variety of media including: IBM cartridges, File Transfer Protocol (FTP), Network Data Mover (NDM), 3.5 Diskette, Compact Disk, Web portal, and additional transmission procedures as authorized by

DMS. The Electronic Data Interchange subsystem (EDI) translates received transactions from NCPDP and X12 standard formats into XML format The file is then picked up by a process called the "splitter" and is processed by the claims engine.

The claims engine edits encounters with similar edits as fee-for-service claims. Encounters have a claim status and an encounter status. Encounters that have been paid by the MCO have a claim status of paid. If no edits set for the claim, it also has an encounter status of paid. Edits that set for encounters are dispositioned as "pay and list" or "threshold". A claim has an encounter claim status of threshold if it has failed a threshold edit. A claim has an encounter claim status of informational if it has failed a pay and list edit. If more than 5% of the claims in the batch are threshold status, the entire batch rejects. If the batch does not reject but has threshold status claims, the threshold status claims are not moved to final history but the other claims are. The MCO must resubmit any claim that is in threshold status. They must resubmit the entire batch if there's more threshold claims than the allowed threshold percentage. Additional data about the encounter that is unique to encounters is displayed in the MCO Data panel. This panel shows the MCO claim number, MCO paid date, MCO paid amount, encounter status and the MMIS Allowed Amount. Encounters are priced by the MMIS if the claim status is paid and the encounter status is paid or informational. "Priced" means that the allowed amount is calculated as if the encounter was a fee-for-service claim. The allowed amount is not calculated if the encounter is not complete enough to be priced. The calculated allowed amount is displayed as the MMIS Allowed Amount on the MCO Data panel.

2.3.7.4 Process Control

Transmissions are only accepted and captured for processing from authorized submitters who can be found on the ”EDI Trading Partner Maintenance panel”. The “EDI Inbound X12 Balancing Report” and the “EDI NCPDP Balancing Daily Report” are used to keep track of the number of accepted transactions by the EDI translators by submitter as well as the number rejected. The accepted transmissions are passed to the claims engine for processing.

2.3.7.5 Receive Encounter

These are objects associated with Receiving Encounters for the Claims subsystem.

Technical Name	Title	Type
EDI Trading Partner Maintenance	EDI Trading Partner Maintenance	Panel
Clm.EDI Encounter Batch Search Clm.EDI Encounter Batch Search Results	EDI Encounter Batch Search EDI Encounter Batch Search Results	Panel Panel
CLM-0132-D	Encounter Submission Statistics – Daily	Report
CLM-0132-M	Encounter Submission Statistics – Monthly	Report
CLM-0132-O	Encounter Submission Statistics – On Request	Report

2.3.8 Validate Encounter

2.3.8.1 Process Description

Encounter data is processed as a batch through validation routines that include the accuracy of procedure coding, editing for dates of service, member eligibility, and receipt date by the MCO. If the error threshold is exceeded, the entire batch is rejected. Individual encounters that fail threshold errors are also rejected. Notifications are sent back to the MCO with enough information to support corrections. In this case, the encounters are stored for a DMS-specified number of days for reporting and analysis. After the DMS-specified number of days elapses, these encounters are deleted from the database. In turn they are replaced when the corrected Encounters are received.

System objects for Validate Encounter are documented on the Claims and Reference subsystems.

2.3.8.2 Process Reference List

Connected via	Connected to	Source	Destination
Encounters Received	Receive Encounter (Process)		X
Encounter Batch Validated	Return Encounter Errors (Process)	X	
Log Encounter Error	Encounter Batch Control (Data Store)	X	
Update Encounter	Encounter (Data Store)	X	

2.3.8.3 interChange Process Narrative

Encounter data is processed as a batch through validation routines that include, the accuracy of procedure coding, editing for dates of service, member eligibility, and receipt date by the MCO. Encounters that pass the validation are then priced using Fee-for-Service methodologies where applicable. If the error threshold is exceeded, the entire batch is rejected. Individual encounters that fail threshold errors are also rejected. Notifications are sent back to the MCO with enough information to support corrections. Regardless of the outcome, all encounters are stored and available for analysis and reporting. Accepted and Rejected batches of encounters are recorded and identified in the “Encounter Batch Summary Search/Results panels”. interChange deletes the rejected batches and individual encounters with threshold errors after a DMS-specified number of days. interChange expects replacement records with corrected data from MCOs.

2.3.8.4 Process Control

Batches passed to the claims engine are processed through the validation edits and outcome is displayed through a series of panels. The “Encounter Batch Summary Search/Results panel” summarizes the outcome of the validation process for each batch of encounters. The total number of encounters, percent of encounters with errors, the number of voids, the amount billed, and amount paid provide the high level information on each batch of encounters.

Information regarding specific errors on the batch is obtained using the “Encounter Error panel”, the “Encounter Claims Related Data panel”, and various claim form inquiry panels. The error summary provides a big picture view of the encounters with errors submitted by the MCO for a

specific submission. DMS analysts use the information to focus on the most significant problems an MCO has in their submission or claims processing. Using the panels, analysts have a complete package of information that can be used to the quality of encounters data. MCOs use the returned information to correct their batches and resubmit the encounters.

2.3.8.5 Validation

Validation is a significant function of the claims engine. Encounters are validated through the use of integrity edits and audits. Edits and audits are supported through a large number of programs that are linked together as one executable called the claims engine. Execution of the edits and audits are controlled by the programs, but are based on controls established using claims and reference pages and panels.

"Error Disposition panels" control how edits and audits are dispositioned. For encounter processing the disposition is: Threshold and Informational. The Threshold errors are accumulated and used in determining when a batch should be accepted or rejected. The Informational errors are used to monitor the quality of claims processing by MCOs. Reference panels such as: "Modifier Maintenance", "HCPCS Procedure Group Maintenance", "Revenue Maintenance", "Codes -Occurrence", "Diagnosis Maintenance", and "Diagnosis Restrictions Maintenance", are all used in controlling integrity and policy edits.

The "Audit Criteria panel" is activated when an error is described as an audit. The panel has many sub-panels used to code specific audits. The claims engine uses the criteria described in the set of panels to perform limitation audits, contra audits, negative contra audits, and umbrella audits.

Member and provider edits check the integrity of the IDs submitted on the encounter. Additional information is obtained based on the date of service, member ID, and Provider Number for policy editing. For example, through the member information third party liability (TPL) is checked to test if some other insurer should have been billed first.

There are numerous relationship edits performed on the claim header and details. Some of them are generic and are therefore encapsulated in a common routine that contains functions called for all claim types. All of the edits are described under the Claims Edit/Audit Processing section of the project workbook and DSD. Others edits are specific to a claim form such as dental editing. These functions are located in claim form-specific source code.

Not all edits are activated for encounters. Many of the edits available are turned off due to the specific policy enforced by the edit. Many policy edits are not enforced for encounters due to conflicting policy or differing approaches taken by the MCOs.

Encounter claims that are free from threshold errors and not part of a rejected batch are priced, written to the database, and moved to finalized history. Encounters are used in duplicate claims checking as well as for reporting and auditing.

Encounters are priced using fee-for-service (FFS) methodologies where applicable and the interChange calculated allowed amount is stored in addition to the amount allowed by the MCO. DMS uses this information to perform analysis of encounter data.

2.3.8.6 Validate Encounter

These are objects associated with Validating Encounters for the Claims subsystem.

Technical Name	Title	Type

Technical Name	Title	Type
Clm.Encounter Batch Search Clm.Encounter Batch Search Results	Encounter Batch Summary Search Encounter Batch Summary Search Results	Panel Panel
Clm.Encounter Errors	Encounter Error	Panel
Clm.Encounter Claims Related Data	Encounter Claim Related Data	Panel
Clm.Encounter Data Navigation	Encounters Data navigation	Panel
Clm.ClaimEobHdrPanel.ascx	EOB	Panel
Clm.ClaimErrorHdrPanel.ascx	Errors	Panel
Clm.RelatedHistoryPanel.ascx	Related History	Panel
Clm.AdjustmentInfoPanel.acsx	Adjustment Information – Daughter or Mother	Panel
Clm.PaymentInfoHdrPanel.ascx	Health Program	Panel
Clm.ClaimDiagnosisPanel.ascx	Diagnosis	Panel
Clm.CAS Inquiry	CAS Inquiry	Panel
Clm.MCO Data	MCO Data	Panel
Clm.DentalClaimPanel.ascx	Dental Claim	Panel
Clm.DentalDetailPanel.ascx	Dental Claim Detail	Panel
Clm.ToothCavityPanel.ascx	Detail Quadrant Cavity	Panel
Clm.PhysicianClaimPanel.ascx	Physician Claim	Panel
Clm.PhysicianDetailPanel.ascx	Physician Claim Detail	Panel
Clm.ClaimDeliveryPanel.ascx	Expecting Date of Delivery	Panel
Clm.PharmacyClaimPanel.ascx	Pharmacy Claim	Panel
Clm.PharmacyDetailPanel.ascx	Pharmacy Claim Detail	Panel

Technical Name	Title	Type
Clm.NCPDP Reject Codes	NCPDP Reject Codes	Panel
Clm.UB92ClaimPanel.ascx	UB92 Claim	Panel
Clm.UB92DetailPanel.ascx	UB92 Claim Detail	Panel
Clm.ClaimConditionPanel.ascx	Condition	Panel
Clm.ClaimICD9Panel.ascx	ICD-9-CM	Panel
Clm.ClaimOccurrencePanel.ascx	Occurrence	Panel
Clm.ClaimPayerPanel.ascx	Payer	Panel
Clm.ClaimsValuePanel.ascx	Value	Panel
HCPCS Procedure Maintenance	HCPCS Procedure Maintenance	Panel
HCPCS Procedure Group Maintenance	HCPCS Procedure Group Maintenance	Panel
HCPCS Procedure Restriction	HCPCS Procedure Restrictions	Panel
HCPCS Procedure - Tooth Quadrant Restriction Maintenance	Tooth Quadrant	Panel
HCPCS Procedure - Tooth Restriction Maintenance	Tooth Restriction Maintenance	Panel
HCPCS Procedure - Revenue Restriction Inquiry	Revenue Restrictions	Panel
ICD-9-CM Procedure Maintenance	ICD9-CM Procedure Maintenance	Panel
ICD-9-CM Procedure Restrictions Maintenance	ICD9-CM Procedure Restrictions Maintenance	Panel
Diagnosis Maintenance	Diagnosis Maintenance	Panel
Diagnosis Restrictions Maintenance	Diagnosis Restrictions Maintenance	Panel
AuditCritPanel	Audit Criteria	Panel

Technical Name	Title	Type
EOB	Other - EOB	Panel
Error Disposition Criteria	Error Disposition Criteria	Panel
Error Disposition Line Item - EOB	Error Disposition Line Item - EOB	Panel
Error Disposition Line Item - Region	Error Disposition Line Item - Region	Panel
Error Disposition Line Item Provider Type - Spec Maintenance	Error Disposition Line Item – Provider Type-Spec Maintenance	Panel

2.3.9 Return Encounter Errors

2.3.9.1 Process Description

Encounter claims that fail threshold errors are returned to the MCO for correction and resubmission.

2.3.9.2 Process Reference List

Connected via	Connected to	Source	Destination
Encounter Batch Validated	Validate Encounter (Process)		X
Encounter Return	Return Encounter Errors (Data Store)	X	

2.3.9.3 interChange Process Narrative

Encounter claims that fail threshold errors are returned to the MCO for correction and resubmission. interChange extracts encounter claims that failed threshold errors, an unsolicited HIPAA 277 transaction is created and sent to the MCO. These transactions contain general error information. The error information supplied on the 277 transaction is cross-referenced to interChange edit and audit error codes.

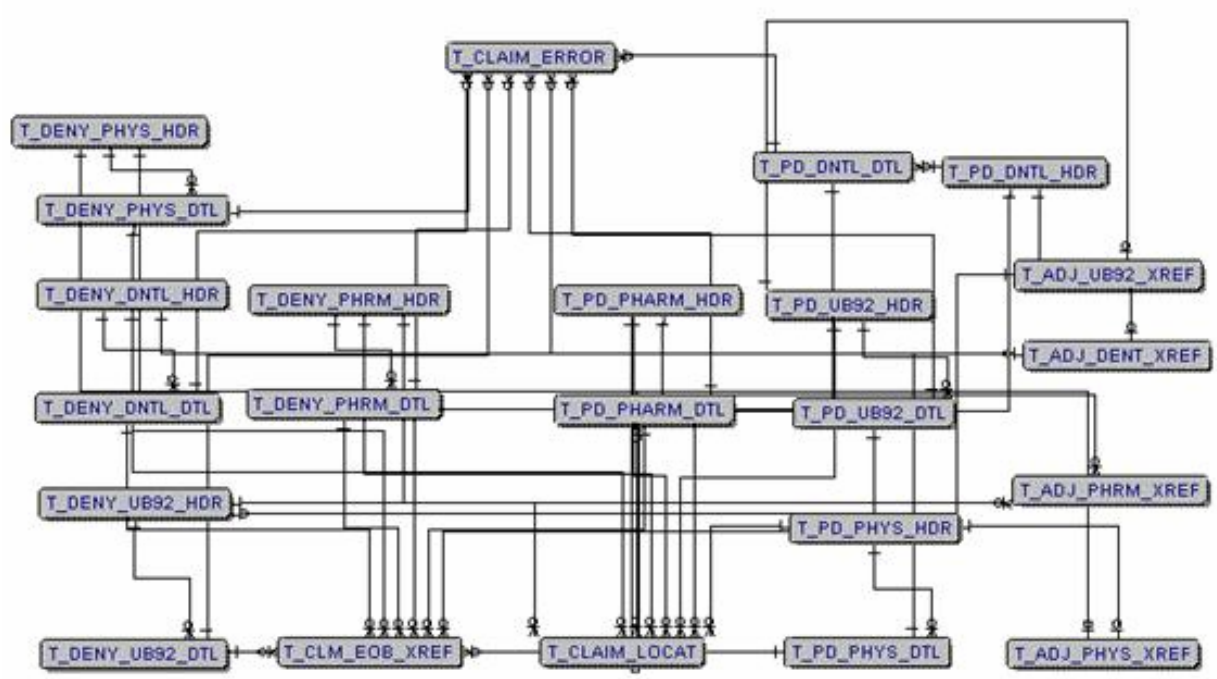
2.3.9.4 Return Encounter Errors

These are objects associated with Returning Encounter Errors.

Technical Name	Title	Type
Clm.Encounter Batch Search	Encounter Batch Summary Search	Panel
Clm.Encounter Batch Search Results	Encounter Batch Summary Search Results	Panel

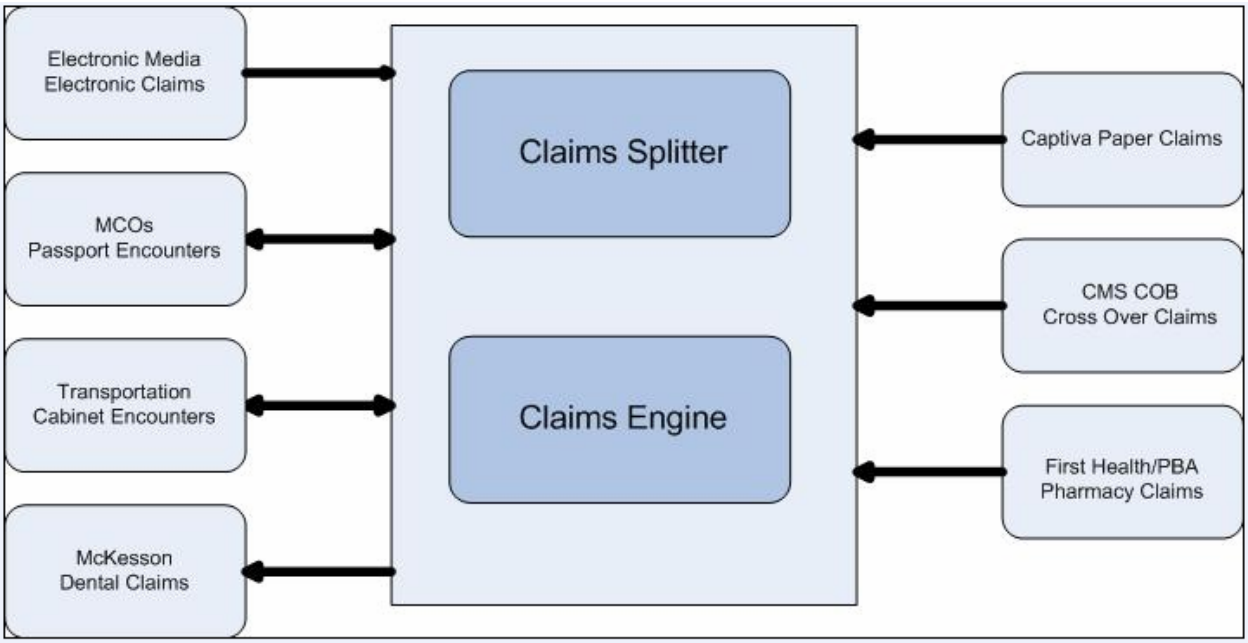
2.4 Data Model

The following excerpt from the data model gives a view of the entities within this subsystem.



2.4.1 External System Interfaces

The following context diagrams gives a view of the entities with which this subsystem interfaces.



2.5 Job Scripts

2.5.1 CLMJD011 - Clerk ID Recycle Claims Report

2.5.1.1 Detailed Job Script Information

Unix Script:	CLMJD011
Description:	This job produces a report that lists the number of errors each clerk ID has worked in the suspense table. Produces report: CLM-0011-D.

Job Step:		js 010-clm0011d
Description:		This job step creates the Clerk ID Recycle Claims Report.
Input/Output Files:		
clm01101.rpt	O	Clerk ID Recycle Claims Report
Job Step:		js 030-copy2routedir
Description:		This job step copies the report to OnBase.
Input/Output Files:		
clm01101.rpt	I	Clerk ID Recycle Claims Report

2.5.2 CLMJD012 - EOB Listing Report

2.5.2.1 Detailed Job Script Information

Unix Script:	CLMJD012	
Description:	The purpose of this report is to list each EOB code and description. Produces Report: CLM-0012-D	
Job Step:		js 010-clm0012d
Description:		This job step creates the EOB Listing Report.
Input/Output Files:		
clm01201.rpt	O	EOB Listing
Job Step:		js 030-copy2routedir
Description:		This job step copies the report to OnBase.
Input/Output Files:		
clm01201.rpt	I	EOB Listing

2.5.3 CLMJD015 - ClaimCheck Potential History To Adjust Report

2.5.3.1 Detailed Job Script Information

Unix Script:	CLMJD015
Description:	This job produces a report that contains the claim detail lines audited by ClaimCheck, which can require adjustment if from paid history. Produces report: CLM-0015-D.

Job Step:		js 010-clm0015d_cc
Description:		This job step creates the ClaimCheck Potential History to Adjust report.
Input/Output Files:		
clm01501.rpt	O	ClaimCheck Potential History To Adjust
Job Step:		js 020-copy2routedir
Description:		This job step copies the reports to OnBase.
Input/Output Files:		
clm01501.rpt	I	ClaimCheck Potential History To Adjust
Job Step:		js 030-clm15del
Description:		This job step deletes old data from the table which contains claims on history that ClaimCheck recommends denial of, as well as the details that were used by ClaimCheck to make this decision. The system keeps only 1 month of data (30 days old from aim cycle date).
Input/Output Files:		

2.5.4 CLMJD018 - Claims Processing Daily Summary Reports

2.5.4.1 Detailed Job Script Information

Unix Script:	CLMJD018
Description:	The purpose of this report is to list summary information by claim type and medical provider type. The following reports are generated from this process: CLM0016D - Claims Processing Daily Summary; CLM0017D - Original Claims Processing Daily Summary; and CLM0018D - Adjustment/Credit Claims Processing Daily Summary.

Job Step:		js 010-clm0018d
Description:		This job step creates the Processing Daily Summary Reports.
Input/Output Files:		
clm01601.rpt	O	Claims Processing Daily Summary
clm01701.rpt	O	Original Claims Processing Daily Summary
clm01801.rpt	O	Adjustment Credit Claims Processing Daily Summary
Job Step:		js 020-copy2routedir
Description:		This job step copies the reports to OnBase.
Input/Output Files:		
clm01601.rpt	I	Claims Processing Daily Summary
clm01701.rpt	I	Original Claims Processing Daily Summary
clm01801.rpt	I	Adjustment Credit Claims Processing Daily Summary

2.5.5 CLMJD020 - Aged Detail Suspense Report

2.5.5.1 Detailed Job Script Information

Unix Script:	CLMJD020
Description:	This job produces a report which is a detail listings by claim type of claims that have been in the suspense file longer than a certain number of days specified by parameter. Produces Report: CLM-0020-D

Job Step:		js 010-clm0020d
Description:		This job step produces the Aged Detail Suspense Report.
Input/Output Files:		
clm02001.rpt	O	Aged Detail Suspense Report
Job Step:		js 030-copy2routedir
Description:		This jobs step copies the report to OnBase.
Input/Output Files:		
clm02001.rpt	I	Aged Detail Suspense Report

2.5.6 CLMJD021 - Suspense File Analysis By Error Code Report

2.5.6.1 Detailed Job Script Information

Unix Script:	CLMJD021
Description:	This job creates a report that is a list of the occurrences of each error code in the suspense file. Produces Report: CLM-0021-D

Job Step:		js 010-clm0021d
Description:		This job step creates the Suspense File Analysis by Error Code report.
Input/Output Files:		
clm02101.rpt	O	Suspense File Analysis by Error Code
Job Step:		js 030-copy2routedir
Description:		This job step copies the report to OnBase.
Input/Output Files:		
clm02101.rpt	I	Suspense File Analysis by Error Code

2.5.7 CLMJD023 - Claims Error Report (FFS)

2.5.7.1 Detailed Job Script Information

Unix Script:	CLMJD023
Description:	This job creates a report which is used by the Claims Resolution Unit as an inventory of claims to be resolved. The detailed error reports by claim type are used by clerks to resolve claims. Produces Report: CLM-0023-D.

Job Step:		js 010-clm0023d
Description:		This job step creates the Claims Error Report (FFS).
Input/Output Files:		
clm02301.rpt	O	Claims Error Report (FFS)
Job Step:		js 030-copy2routedir
Description:		This job step copies the report to OnBase.
Input/Output Files:		
clm02301.rpt	I	Claims Error Report (FFS)

2.5.8 CLMJD027 - Transaction Register Report

2.5.8.1 Detailed Job Script Information

Unix Script:	CLMJD027
Description:	This job creates a report which is to identify every claim processed in a daily claim adjudication cycle. Each claim is identified by ICN, Member ID Number, Provider Number, Transaction Code, Dates of Service, and Payment Amount. Additionally, this report displays any error codes posted to the claim throughout the daily claim adjudication cycle. Produces Report: CLM-0027-D

Job Step:		js 010-clm0027d
Description:		This job step creates the Transaction Register report.
Input/Output Files:		
clm02701.rpt	O	Transaction Register
Job Step:		js 030-copy2routedir
Description:		This job step copies the report to OnBase.
Input/Output Files:		
clm02701.rpt	I	Transaction Register

2.5.9 CLMJD051 - Aged Detail Adjustment Report

2.5.9.1 Detailed Job Script Information

Unix Script:	CLMJD051	
Description:	This job produces a report that list adjustments that have aged. Produces Report: CLM-0051-D	
Job Step:		js 010-clm0051d
Description:		This job step creates the Aged Detail Adjustment Report.
Input/Output Files:		
clm05101.rpt	O	Aged Detail Adjustment Report
Job Step:		js 030-copy2routedir
Description:		This job step copies the report to OnBase.
Input/Output Files:		
clm05101.rpt	I	Aged Detail Adjustment Report

2.5.10 CLMJD054 - Error Recycle Parameters Report

2.5.10.1 Detailed Job Script Information

Unix Script:	CLMJD054
Description:	The purpose of this report is to list each recycle request currently submitted by the user. This report is by CDE_ESC. Produces Report: CLM-0054-D

Job Step:		js 010-clm0054d
Description:		This job step creates the Error Recycle Parameters report.
Input/Output Files:		
clm0054d.rpt	O	Error Recycle Parameters
Job Step:		js 030-copy2routedir
Description:		This job step copies the report to OnBase.
Input/Output Files:		
clm0054d.rpt	I	Error Recycle Parameters

2.5.11 CLMJD060 - Suspense File Analysis by Claim Type Report

2.5.11.1 Detailed Job Script Information

Unix Script:	CLMJD060
Description:	This job creates a report that lists the number and dollar value totals of the various claim types that are currently in the suspense file. It also includes the same totals for claims that are 1 to 30 days old, 31 to 60 days old, 61 to 90 days old and over 90 days old, as well as totals for each of the above. Produces Report: CLM-0060-D

Job Step:		js 010-clm0060d
Description:		This job step creates the Suspense File Analysis By Claim Type 3 report.
Input/Output Files:		
clm06001.rpt	O	Suspense File Analysis By Claim Type
Job Step:		js 030-copy2routedir
Description:		This job step copies the report to OnBase.
Input/Output Files:		
clm06001.rpt	I	Suspense File Analysis By Claim Type

2.5.12 CLMJD065 - Aged Suspended Claims By Location Report

2.5.12.1 Detailed Job Script Information

Unix Script:	CLMJD065
Description:	This job creates a report which lists the totals for aged claims that are 1 to 30 days old, 31 to 60 days old, 61 to 90 days old, and over 90 days old. It also lists the totals for each location, each aged claims period for all the locations, and the grand total. Produces Report: CLM-0065-D

Job Step:		js 010-clm0065d
Description:		This job step creates the Aged Suspended Claims By Location Report.
Input/Output Files:		
clm06501.rpt	O	Aged Suspended Claims By Location Report
Job Step:		js 030-copy2routedir
Description:		This job step copies the report to OnBase.
Input/Output Files:		
clm06501.rpt	I	Aged Suspended Claims By Location Report

2.5.13 CLMJD075 - Daily upload and update of attachment for electronic claims

2.5.13.1 Detailed Job Script Information

Unix Script:	CLMJD075	
Description:	This script concatenates all the input attachment info coming from cold server and feeds it to the program CLMPD075 which updates the attachment xref table with the attachment indicator and resubmits the claim once the attachment has arrived. **NOTE:** Kentucky currently does not allow electronic claims with attachments. This job is on "ICE" for Kentucky and is not to be run.	
Job Step:		js 010-cat
Description:		This job step concatenates the input attachment files from the cold server for that day.
Input/Output Files:		
ACN.txt	I	This has all the txt files coming in from the RRI server for that day with the attachment info.
Attachments.dat	I	This file contains the attachment files for the attachments that arrived before the claim in.
clm07510.ctl	I	This system removes the newline character for the input attachment files.
attachmentr.dat	O	This file contains the input data from the day's attachment files and the previous day's gdg.
Job Step:		js 020-clmpd075
Description:		This job step executes the C program clmpd075 which updates the attachment xref table.
Input/Output Files:		
attachmentr.dat	I	This file contains the input data from the day's attachment files and the previous day's gdg.
Attachments.dat	O	This file contains the attachment files for the attachments that arrived before the claim in.
Job Step:		js 030-rm
Description:		This job step removes the txt files that come in daily and the concatenated file output that comes out of js10.
Input/Output Files:		
ACN.txt	I	This has all the txt files coming in from the RRI server for that day with the attachment info.

attachmentr.dat	I	This file contains the input data from the day's attachment files and the previous day's gdg.
Job Step:		js 040-rm
Description:		This job step removes the attached complete file.
Input/Output Files:		
Attachcomplete.dat	I	This file indicates the successful completion of the batch job that transmits attachment data from COLD server to SUN box.

2.5.14 CLMJD099 - Captiva Received Rejected

2.5.14.1 Detailed Job Script Information

Unix Script:	CLMJD099
Description:	This job creates report CLM-0099-D which compares the output from the paper claims data capture system (Captiva) to the input in the system. The report identifies any missing Internal Control Numbers (ICNs) and displays them on this report.

Job Step:		js 010-cp
Description:		This step copies the input file to a gdg file.
Input/Output Files:		
ICNList.txt	I	Temporary file containg list of ICN's which came into the system from Captiva.
ICNList.dat	O	File containing list of ICN's which came into the system from Captiva.
Job Step:		js 020-clm0099d
Description:		This step reads the ICN list and generates the report.
Input/Output Files:		
ICNList.dat	I	File containing list of ICN's which came into the system from Captiva.
clm0099D.rpt	O	Captiva Received Rejected Report
Job Step:		js 040-copy2routedir
Description:		This step sends the report to COLD.
Input/Output Files:		
clm0099D.rpt	I	Captiva Received Rejected Report

2.5.15 CLMJD1000 - Allows clmd1000 to be run

2.5.15.1 Detailed Job Script Information

Unix Script:	CLMJD1000
Description:	js10 executes program clmd1000 to resubmit review claims where the clerk id on review has designated to all claims released. There are no other job steps in the job. All updates are on the data base so there are no input or output files

Job Step:		js 010 - sqlplus
Description:		Job step 10 updates T_DSS_PARMS with a monthly paid date range.
Input/Output Files:		
dsm75001.sql	I	This file updates T_DSS_PARMS with a monthly paid date range.
pcg.date_parm_$today.log	O	The log file for T_DSS_PARMS update process.
Job Step:		js 020 - sed
Description:		Job step 20 replaces the date parameters in the sqls with actual dates taken from T_DSS_PARMS.
Input/Output Files:		
dsm75002.shl	I	This file contains the physician claims sql code with date parameters
dsm75003.shl	I	This file contains the UB claims sql code with date parameters.
dsm75004.shl	I	This file contains the Dental claims sql code with date parameters.
dsm75005.shl	I	This file contains the Pharmacy claims sql code with dates.
dsm75002.sql	O	This file contains the physician claims sql code with dates.
dsm75003.sql	O	This file contains the UB claims sql code with dates.
dsm75004.sql	O	This file contains the dental claims sql code with dates.
dsm75005.sql	O	This file contains the Pharmacy claims sql code with date parameters.

Job Step:		js 030 - sqlplus
Description:		Job step 30 processes the Physcian claims sql code.
Input/Output Files:		
dsm75002.sql	I	This file contains the physician claims sql code with dates.
pcg.ub92_”$BEG_DATE”_”END_DATE”.txt	O	This file contains the UB claims extract output file.
pcg.ub92_$today.log	O	The log file for the UB claim extract process.
Job Step:		js 040 - sqlplus
Description:		Job step 40 processes the UB claims sql code.
Input/Output Files:		
dsm75003.sql	I	This file contains the UB claims sql code with dates.
pcg.ub92_”$BEG_DATE”_”END_DATE”.txt	O	This file contains the UB claims extract output file.
pcg.ub92_$today.log	O	The log file for the UB claim extract process.
Job Step:		js 050 - sqlplus
Description:		Job step 50 processes the Dental claims sql code.
Input/Output Files:		
dsm75003.sql	I	This file contains the UB claims sql code with dates.
pcg.ub92_”$BEG_DATE”_”END_DATE”.txt	O	This file contains the UB claims extract output file.
pcg.ub92_$today.log	O	The log file for the UB claim extract process.
Job Step:		js 060 - sqlplus
Description:		Job step 60 processes the Pharmacy claims sql code.
Input/Output Files:		
dsm75005.sql	I	This file contains the Pharmacy claims sql code with date parameters.
pcg.pharm_”$BEG_DATE”_”END_DATE”.txt	O	This file contains the UB claims extract output file.
pcg.pharmacy_$today.log	O	The log file for the UB claim extract process.

Job Step:		js 070 - mailx
Description:		
Input/Output Files:		

2.5.16 CLMJD10D - On-Line Daily Adjustment Report

2.5.16.1 Detailed Job Script Information

Unix Script:	CLMJD10D
Description:	This job creates a report which is used to support the on-line updates that have been made to the credit/adjustment data entry. Produces Report: CLM-0010-D

Job Step:		js 010-clm0010d
Description:		This job step creates the Online Adjustment Daily Report.
Input/Output Files:		
clm0010D.rpt	O	Online Adjustment Daily Report
Job Step:		js 030-copy2routedir
Description:		This job step copies the report to OnBase.
Input/Output Files:		
clm0010D.rpt	I	Online Adjustment Daily Report

2.5.17 CLMJD110 - Online Mass Adjustment

2.5.17.1 Detailed Job Script Information

Unix Script:	CLMJD110
Description:	This job creates a report which is used to support the online updates that have been made to adjustment data entry. Produces Report: CLM-0110-D.

Job Step:		js 010-clm0110d
Description:		This job step creates the Online Mass Adjustment Report.
Input/Output Files:		
clm0110d.rpt	O	Online Mass Adjustment Report
Job Step:		js 030-copy2routedir
Description:		This job step copies the report to OnBase.
Input/Output Files:		
clm0110d.rpt	I	Online Mass Adjustment Report

2.5.18 CLMJD123 - Claims Error Report (Encounters)

2.5.18.1 Detailed Job Script Information

Unix Script:	CLMJD123
Description:	This job creates a report which is used by the Claims Resolution Unit as an inventory of encounters to be resolved. The detailed error reports by claim type are used by clerks to resolve encounters. Produces Report: CLM-0123-D.

Job Step:		js 010-clm0123d
Description:		This job step creates the Claims Error Report (Encounters).
Input/Output Files:		
clm12301.rpt	O	Claims Error Report (Encounters)
Job Step:		js 030-copy2routedir
Description:		This job step copies the report to OnBase.
Input/Output Files:		
clm12301.rpt	I	Claims Error Report (Encounters)

2.5.19 CLMJD130 - Claims Submission Statistics - Daily Report

2.5.19.1 Detailed Job Script Information

Unix Script:	CLMJD130
Description:	This job creates a reports that provides a daily summary of claim counts sent to the claims engine by submission type (NCPDP, WEB DDE, ASC x12). It reports information at the summary level, and at the submitter level. Produces Report: CLM-0130-D

Job Step:		js 010-clm0130
Description:		This jobs step creates the Claims Submission Statistics - Daily.
Input/Output Files:		
clm0130d.rpt	O	Claims Submission Statistics - Daily
Job Step:		js 030-copy2routedir
Description:		This job step copies the report to OnBase.
Input/Output Files:		
clm0130d.rpt	O	Claims Submission Statistics - Daily

2.5.20 CLMJD132 - Encounters Submission Statistics - Daily

2.5.20.1 Detailed Job Script Information

Unix Script:	CLMJD132
Description:	This job creates a report that provides a daily Encounters Submission Statistics summary of claim counts sent to the claims engine by submission type (NCPDP, ASC x12). Produces Report: CLM-0132-D

Job Step:		js 010-clm0132
Description:		This job step creates the Encounters Submission Statistics - Daily
Input/Output Files:		
clm0132d.rpt	O	Encounters Submission Statistics - Daily
Job Step:		js 030-copy2routedir
Description:		This job step copies the report to ONBase.
Input/Output Files:		
clm0132d.rpt	I	Encounters Submission Statistics - Daily

2.5.21 CLMJD140 - Encounter Batch Summary

2.5.21.1 Detailed Job Script Information

Unix Script:	CLMJD140
Description:	This job creates a report which tells how the Managed Care Organization's entire batch file of encounter claims processed. Produces Report: CLM-0140-D

Job Step:		js 010-clm0140d
Description:		This job step creates the Encounter Batch Summary Report.
Input/Output Files:		
clm0140d.rpt	O	Encounter Batch Summary Report
Job Step:		js 030-copy2routedir
Description:		This job step copies the report to OnBase.
Input/Output Files:		
clm0140d.rpt	I	Encounter Batch Summary Report

2.5.22 CLMJD210 - Claims for Provider Mass Rate Adjustments

2.5.22.1 Detailed Job Script Information

Unix Script:	CLMJD210	
Description:	Identifies claims associated with Provider Mass Rate Adjustment requests.	
Job Step:		js 010-clmp210d
Description:		The job step executes clmp210d.
Input/Output Files:		
database - in	I	This program uses database tables as it only input
database - out	O	This program uses database tables as its only output.

2.5.23 CLMJD211 - Claims for Provider Mass Rate Adjustments

2.5.23.1 Detailed Job Script Information

Unix Script:	CLMJD211	
Description:	Identifies claims associated with Provider Mass Rate Adjustment requests.	
Job Step:		js 010-clmp211d
Description:		The job step executes clmp211d.
Input/Output Files:		
database - in	I	This program uses database tables as its only input
database - out	O	This program uses database tables as its only output.

2.5.24 CLMJD25D - Services To Be Clerically Denied Report

2.5.24.1 Detailed Job Script Information

Unix Script:	CLMJD25D
Description:	This job creates a report which lists the claims that had services denied by a clerk, thereby enabling supervisory personnel to identify those clerks who appear to resolve an inordinate number of suspended claims by denying error codes. Produces Report: CLM-0025-D

Job Step:		js 010-clm0025d
Description:		This job step creates the Services to be Clerically Denied Report.
Input/Output Files:		
clm0025d.rpt	O	Services to be Clerically Denied Report
Job Step:		js 030-copy2routedir
Description:		This job step copies the report to OnBase.
Input/Output Files:		
clm0025d.rpt	I	Services to be Clerically Denied Report

2.5.25 CLMJD26D - Claims Paid Due To Force Report

2.5.25.1 Detailed Job Script Information

Unix Script:	CLMJD26D
Description:	This job creates a report which lists those claims that have errors that were clerically forced or overridden, thereby enabling supervisory personnel to identify those clerks who appear to resolve an inordinate number of suspended claims by forcing error codes. Produces Report: CLM-0026-D

Job Step:		js 010-clm0026d
Description:		This job step creates the Claims to be Paid Due To Force Report.
Input/Output Files:		
clm0026d.rpt	O	Claims to be Paid Due To Force Report
Job Step:		js 030-copy2routedir
Description:		This job step copies the report to OnBase.
Input/Output Files:		
clm0026d.rpt	I	Claims to be Paid Due To Force Report

2.5.26 CLMJD300 - Daemon process to kick off Sendclms

2.5.26.1 Detailed Job Script Information

Unix Script:	CLMJD300
Description:	This job boots the sendclm daemon. The sendclms job (CLMJD300) actually processes the claims. As claim files are processed, they are written to the backup directory for the current date - $PRODDIR/data/claims/input/ccyymmdd-bkp. If a claim errors off (causes an abend), it is placed in the $PRODDIR/data/claims/input/ccyymmdd-bad directory. To review the logs associated with bad claims, look in the $LOGDIR directory for files named the following: stdout.. and stderr..

Job Step:		js 010-sendclmd
Description:		This job step boots the sendclmd daemon.
Input/Output Files:		
Not available	I	Not available

2.5.27 CLMJD350 - Move claims from NEWDAY to the input directory

2.5.27.1 Detailed Job Script Information

Unix Script:	CLMJD350
Description:	This job executes the import/export utility to copy tables from the claims engine database to the on-line database. In addition to moving the claim files FROM the newday directory TO the input/Today directory, the splitter job (CLMJD350) also does the following: - splits the large files into smaller files that contain 300 or fewer claims (we can change this in the Jobscript by specifying the -m input parameter to clmpsplt program) - assigns claim saks to paper claims - assigns claim saks and ICNs to electronic and tape claims - renames the original file name by adding a suffix which indicates how many claims are in the file as well as the claim type. Example : 56850085.100111910A.837I.0001.0050.B_xml - split file.

Job Step:	js 010-clmpsplt
Description:	This job step executes clmpsplt. It exports tables from the claim engine database.

Input/Output Files:

Not available	I	Not available

2.5.28 CLMJD351 - Start the claim client program

2.5.28.1 Detailed Job Script Information

Unix Script:	CLMJD351
Description:	This job executes the claim client program that sends NECS claim records to the appropriate servers.

Job Step:	js 010-sendclms
Description:	Start a claim engine instance
Input/Output Files:	

2.5.29 CLMJD422 - Combined Claim Error Report

2.5.29.1 Detailed Job Script Information

Unix Script:	CLMJD422
Description:	This job creates a report which is used by the Pending Claims Unit to correct suspended claims. Produces Report: CLM-0022-D

Job Step:		js 010-clmp422
Description:		This job step creates the Combined Claim Error report.
Input/Output Files:		
cld42201.dat	O	Combined Claim Error
Job Step:		js 020-copy2routedir
Description:		This job step copies the report to OnBase.
Input/Output Files:		
cld42201.dat	I	Combined Claim Error

2.5.30 CLMJD500 - Top Ten Claims Approved For Payment

2.5.30.1 Detailed Job Script Information

Unix Script:	CLMJD500
Description:	This job creates a report which lists the ten claims with the highest reimbursement amounts approved for payment for each claim type for the adjudication cycle. Produces Report: CLM-5000-D

Job Step:		js 010-clm5000d
Description:		This job step creates the top ten report
Input/Output Files:		
clm5000d.rpt	O	Top Ten Claims Approved For Payment report
Job Step:		js 020-copy2routedir
Description:		This job step copies the report to OnBase.
Input/Output Files:		
clm5000d.rpt	I	Top Ten Claims Approved For Payment report

2.5.31 CLMJD956 - Void Adjustments

2.5.31.1 Detailed Job Script Information

Unix Script:	CLMJD956	
Description:	This job creates the void automatic adjustments	
Job Step:		js 010-clmp956d
Description:		Void Adjustments
Input/Output Files:		
Database-In	I	This file uses database tables as its only input.
Database-Out	O	This file uses database tables as its only output.

2.5.32 CLMJDA01 - Create audit driver tables

2.5.32.1 Detailed Job Script Information

Unix Script:	CLMJDA01
Description:	This job executes the export utility to create the t_audit_ndc, t_audit_proc, and t_audit_rev database tables for claims engine audit processing.

Job Step:		js 010-aim_pswd
Description:		get aim_pswd
Input/Output Files:		
Job Step:		js 020-sqldba
Description:		create t_audit_ndc
Input/Output Files:		
database - in	I	This program uses database tables as it only input
database - out	O	This program uses database tables as its only output.
Job Step:		js 030-sqldba
Description:		create t_audit_proc
Input/Output Files:		
database - in	I	This program uses database tables as it only input
database - out	O	This program uses database tables as its only output.
Job Step:		js 040-sqldba
Description:		create t_audit_rev
Input/Output Files:		
database - in	I	This program uses database tables as it only input
database - out	O	This program uses database tables as its only output.

2.5.33 CLMJDA99 - Remove adjusted claims

2.5.33.1 Detailed Job Script Information

Unix Script:	CLMJDA99	
Description:	Calls adjustment process to delete claims marked for deletion.	
Job Step:		js 010-clmpda01
Description:		The job step executes clmpda01.
Input/Output Files:		
database - in	I	This program uses database tables as it only input
database - out	O	This program uses database tables as its only output.

2.5.34 CLMJDER1 - Daily Error Summary Extract

2.5.34.1 Detailed Job Script Information

Unix Script:	CLMJDER1
Description:	The purpose of this job is to create a database extract. This extract includes a header, and detail records. The gdg created from the extracts is used by CLMJDER2 to generate Daily Error Summary reports.

Job Step:		js 010-oraload.sh
Description:		This job step Export Inverse Code Log Table from the On-line database.
Input/Output Files:		
clm0001a.dat	I	This file contains the daily errors extract.
clm0001a.sql	I	This file contains SQL control cards to extract daily system parameters date.
clm0001b.dat	I	This file contains the daily errors extract.
clm0001b.sql	I	This file contains the daily errors extract SQL.
clm0001d.ctl	I	This file contains the oraload control card to identify SQL cards.
Job Step:		js 020-cat
Description:		This job step combines the header and detail records.
Input/Output Files:		
clm0001a.dat	I	This file contains the daily errors extract.
clm0001b.dat	I	This file contains the daily errors extract.
clmexcex.dat	O	This file contains concatenated daily errors extract.

2.5.35 CLMJDER2 - Daily Error Summary By Claim Type Report

2.5.35.1 Detailed Job Script Information

Unix Script:	CLMJDER2
Description:	This job creates reports which show how many times each error occurred by claim type for the last 5 cycles for all claims, original claims and adjustments. Produces Reports: CLM-0024-D(All), CLM-0055-D(Original), and CLM-0056-D(Adjustments).

Job Step:		js 010-otsortd
Description:		This job step sorts the last five error extract files.
Input/Output Files:		
clm01029.ctl	I	This file contains the error extract sort criteria.
clmexcex.dat	I	This file contains concatenated daily errors extract.
clm01029.srt	O	This file contains the last five error extracts.
Job Step:		js 020-clmper01
Description:		This job step reads the sorted error extract files and produces reports: • CLM-0024-D • CLM-0055-D • CLM-0056-D
Input/Output Files:		
clm01029.srt	I	This file contains the last five error extracts.
clm0024D.rpt	O	Daily Error Summary By Claim Type
clm0055D.rpt	O	Daily Error Summary By Claim Type Original Claims
clm0056D.rpt	O	Daily Error Summary By Claim Type Adjustments
Job Step:		js 040-copy2routedir
Description:		This job step copies the CLM-0024-D report to OnBase.
Input/Output Files:		
clm0024D.rpt	B	Daily Error Summary By Claim Type
Job Step:		js 060-copy2routedir

Description:		This job step copies the CLM-0055-D report to OnBase.
Input/Output Files:		
clm0055D.rpt	B	Daily Error Summary By Claim Type Original Claims
Job Step:		js 080-copy2routedir
Description:		This job step copies the CLM-0056-D report to OnBase.
Input/Output Files:		
clm0056D.rpt	B	Daily Error Summary By Claim Type Adjustments

2.5.36 CLMJDFCST - Produce listing of all scheduled jobs for a given range of dates

2.5.36.1 Detailed Job Script Information

Unix Script:	CLMJDFCST
Description:	This produces and print a listing of all jobs scheduled to run between 14:00 on the current day and 13:59 on the next day. On Friday, this prints a listing of all jobs scheduled to run between 14:00 on Friday and 13:59 on Monday.

Job Step:		js 010-fcst.sh
Description:		Produce the forecast listing
Input/Output Files:		
cldfcst01.rpt	O	Cycle job forecast report
Job Step:		js 030-mailfcst.sh
Description:		Mails the forecast listing
Input/Output Files:		
cldfcst01.rpt	I	Cycle job forecast report

2.5.37 CLMJDI10 - Encounters Threshold Errors Extracts

2.5.37.1 Detailed Job Script Information

Unix Script:	CLMJDI10	
Description:	This job creates an extract for the threshold errors which are identified on the encounter claims. The extract is then input to the EDI process to create the 277 transaction which is sent back to MCOs.	
Job Step:		js 010-clmpdi10
Description:		This job step executes program that extracts encounter claims satisfying selection criteria
Input/Output Files:		
clmpdi101.dat	O	encounter claim list
clmpdi101_err01.dat	O	encounter claim error file
Job Step:		js 020-otsortd
Description:		Sort extracts
Input/Output Files:		
clm00i10.ctl	I	Control file for sorting clmpdi101.dat file
clmpdi101.dat	I	encounter claim list
clmpdi102.dat	O	Sorted encounter lists file.
Job Step:		js 030-clmpdi15
Description:		This job step executes program that formats 277 transactions for MCO based on 277U transactions specifications
Input/Output Files:		
clmpdi102.dat	I	Sorted encounter lists file.
clmpdi103.dat	O	This is the unmapped unsolicited 277 file which is the input file into the map.
Job Step:		js 040-cp
Description:		Copy the 277U encounter claims file to the claims folders /export/ftp/sysstat/claims/outbound/277 where sysstat = dev or test or mod or prod for further EDI processing
Input/Output Files:		

2.5.38 CLMJDI20 - This job creates and send an extact of Dental Claims Creates Dental Claim Extract For McKesson

2.5.38.1 Detailed Job Script Information

Unix Script:	CLMJDI20
Description:	This job creates and sends an extact of Dental Claims to the FTP outbound directory.

Job Step:		js 010-clmpdi20
Description:		This step creates a dental extract file and sends the file to the outbound directory.
Input/Output Files:		
CGMPRD.CCDEK.FTP.TRANSIN	O	This is an extract of Dental Claims which is sent to the outbound directory. The file will be transferred to McKesson for ClaimCheck processing.

2.5.39 CLMJDMADJ - Mass Adjustments

2.5.39.1 Detailed Job Script Information

Unix Script:	CLMJDMADJ
Description:	Calls adjustment process for claims associated with Mass Adjustment requests. The job step executes clmpma_mass_adj which calls the adjustment process for all Claims Mass Adjustment requests on the T_ADJ_MASS_RQST table. (The adjustment process selects all claims for the request from the T_ADJ_MASS_CLAIM table.)

Job Step:	js 010-clmpma_mass_adj
Description:	The job step executes clmpma_mass_adj
Input/Output Files:	

2.5.40 CLMJDM_PAY_VOID - Calls adjustment process for claims associated with Mass Adjustment requests

2.5.40.1 Detailed Job Script Information

Unix Script:	CLMJDM_PAY_VOID
Description:	Calls adjustment process for claims associated with Mass Adjustment requests.

Job Step:	js 010-clmpma03
Description:	The job step executes clmpma03
Input/Output Files:	

2.5.41 CLMJDM_PROV_RATE - Call the adjustment routine for claims Mass Adjustment requests.

2.5.41.1 Detailed Job Script Information

Unix Script:	CLMJDM_PROV_RATE
Description:	Calls adjustment process for claims associated with Provider Rate Mass Adjustment requests.
Job Step:	js 010-clmpma_mass_adj
Description:	The job step executes clmpma_mass_adj - clmpma_mass_adj 4
Input/Output Files:	
Job Step:	js 020-clmpma_mass_adj
Description:	The job step executes clmpma_mass_adj - clmpma_mass_adj 12
Input/Output Files:	

2.5.42 CLMJDNDY - Move previous day claims to re-run

2.5.42.1 Detailed Job Script Information

Unix Script:	CLMJDNDY
Description:	The nightly Move Previous Day Claims to Re-run job executes numerous Unix shell commands to clean the encounters processing directories, as follows: • Moves any un-processed claims files (*.xml, *.?_xml, and *.rst files) from the current and previous months input directories (excluding current days input directory) into the current day's input directory. • Moves any *.xml files from *-hold directories (excluding current day's *-hold directory) into the newday directory and then deletes the *-hold directory. For directories with names like input/ccyymmdd* (for example input/20030601, input/20030601- bad, input/20030601-bkp) but excluding today's date, it performs as follows: • Moves any files from *-bad directories into the newday-bad directory. • Creates a tape archive (TAR) and compresses each daily backup (*-bkp) directory and then deletes the corresponding input/ccyymmdd* directories. For example, a TAR is created for directory /20030601-bkp and is compressed to create 20030601-bkp.tar.Z file, and then all /20030601* directories are deleted. Finally, it checks the newday-bad directory. If it contains any files, then it: • Executes UNIX shell script cycle_monitor.sh to determine the cycle monitor's user ID • Formats an E-mail listing of the files in the newday-bad directory and includes directions to investigate the bad files. • Sends E-mail to the user ID if the cycle monitor groups E-mail address.

Job Step:		js 010-UNIX
Description:		Execute Unix shell commands to clean-up claims processing directories.
Input/Output Files:		
Not available	I	Not available

2.5.43 CLMJDR01 - Copy the recycled claims to newday

2.5.43.1 Detailed Job Script Information

Unix Script:	CLMJDR01
Description:	This job executes the claims recycle program to create a claims xml file from suspended claims. The file is generated by looking for status code = 'R' in T_HIST_DIRECTORY table. The recycled file is sent to the recycle directory for recycling process.

Job Step:	js 010-aim_pswd
Description:	This job step sets up the job environment(Get AIM_PSWD).
Input/Output Files:	
Job Step:	js 020-clmpdr01
Description:	This job step executes a program to create claims xml for input to claims engine.
Input/Output Files:	

claims_recycle.mmddyy.hhmmss.n.xml	O	Recycled claim XML file.

2.5.44 CLMJDR02 - Batch job to recycle suspended claims

2.5.44.1 Detailed Job Script Information

Unix Script:	CLMJDR02
Description:	This job is to recycle Suspended claims.

Job Step:	js 010-clmpdr02
Description:	Run clmpdr02 program
Input/Output Files:	

2.5.45 CLMJM031 - Suspense File Analysis By Claim Type Report

2.5.45.1 Detailed Job Script Information

Unix Script:	CLMJM031
Description:	This job creates a report which list summary information by claim type and by medical provider type for a claim adjudication cycle. This report includes adjustments as well. Produces Report: CLM-0031-M

Job Step:		js 010-clm0031m
Description:		This job step creates the Suspense File Analysis by Claim Type report.
Input/Output Files:		
clm03101.rpt	O	Suspense File Analysis by Claim Type
Job Step:		js 030-copy2routedir
Description:		This job step copies the report to OnBase.
Input/Output Files:		
clm03101.rpt	I	Suspense File Analysis by Claim Type

2.5.46 CLMJM046 - Potential Narcotic Drug Abuse Report

2.5.46.1 Detailed Job Script Information

Unix Script:	CLMJM046
Description:	This job creates a report that summarizes prescribing information relating to narcotic drugs. Produces Report: CLM-0046-M

Job Step:		js 010-clmp046m
Description:		This job step creates the Potential Narcotic Drug Abuse report.
Input/Output Files:		
clm04601.rpt	O	Potential Narcotic Drug Abuse
Job Step:		js 030-copy2routedir
Description:		This job step copies the report to OnBase.
Input/Output Files:		
clm04601.rpt	I	Potential Narcotic Drug Abuse

2.5.47 CLMJM050 - Retro Active Rate Adjustment Report

2.5.47.1 Detailed Job Script Information

Unix Script:	CLMJM050
Description:	This job creates a report that shows all error codes of all claims for the previous daily claims Cycle. Produces Report: CLM-0050-M

Job Step:		js 010-clm0050m
Description:		This step creates the Retro Active Rate Adjustment Report.
Input/Output Files:		
clm05001.rpt	O	Retro Active Rate Adjustment Report
Job Step:		js 030-copy2routedir
Description:		This job step copies the report to OnBase.
Input/Output Files:		
clm05001.rpt	I	Retro Active Rate Adjustment Report

2.5.48 CLMJM130 - Claims Submission Statistics - Monthly Report

2.5.48.1 Detailed Job Script Information

Unix Script:	CLMJM130
Description:	This job creates a monthly report which provides a count of suspended, paid and denied claims by submission type and claim type and then by sender. Produces Report: CLM-00130-M

Job Step:		js 010-clm0130
Description:		This job step creates the Claims Submission Statistics - Monthly report.
Input/Output Files:		
clm0130m.rpt	O	Claims Submission Statistics - Monthly
Job Step:		js 030-copy2routedir
Description:		This job step copies the report to OnBase.
Input/Output Files:		
clm0130m.rpt	I	Claims Submission Statistics - Monthly

2.5.49 CLMJM132 - Encounters Submission Statistics - Monthly

2.5.49.1 Detailed Job Script Information

Unix Script:	CLMJM132
Description:	This job creates a report that provides a Monthly Encounters Submission Statistics summary of claim counts sent to the claims engine by submission type (NCPDP, ASC x12). Produces Report: CLM-0132-M

Job Step:		js 010-clm0132
Description:		This job step creates the Encounters Submission Statistics - Monthly
Input/Output Files:		
clm0132m.rpt	O	Encounters Submission Statistics - Monthly
Job Step:		js 030-copy2routedir
Description:		This job step copies the report to OnBase.
Input/Output Files:		
clm0132m.rpt	I	Encounters Submission Statistics - Monthly

2.5.50 CLMJMA00 - Adjustment Request Update

2.5.50.1 Detailed Job Script Information

Unix Script:	CLMJMA00
Description:	This job updates the adjustment requests status to in process or finalized depending on status of all the children (adjustments) that are a part of the request.

Job Step:	js 010-clmpma00
Description:	Update Adjustment Request Status
Input/Output Files:	

2.5.51 CLMJMACR - Mass Adjustment Claim Select

2.5.51.1 Detailed Job Script Information

Unix Script:	CLMJMACR
Description:	Calls adjustment process to select claims for mass adjustment. The job step executes clmpmacr which retrieves mass requests from the T_ADJ_MASS_RQST table and inserts the associated claims into the T_ADJ_MASS_CLAIM table. It uses the filter criteria on the T_ADJMS_* tables to identify the claims. (Other tables are accessed to cross reference filter fields with the claims.)

Job Step:	js 010-clmpmacr
Description:	The job step executes clmpmacr.
Input/Output Files:	

2.5.52 CLMJO132 - Encounters Submission Statistics - On Request

2.5.52.1 Detailed Job Script Information

Unix Script:	CLMJO132
Description:	This job creates a reports that provides On Request Encounters Submission Statistics summary of claim counts sent to the claims engine by submission type (NCPDP, ASC x12).

Job Step:		js 010-clm0132
Description:		This job step creates the Encounters Submission Statistics - On Request
Input/Output Files:		
clm0132o.rpt	O	Encounters Submission Statistics - On Request
Job Step:		js 030-copy2routedir
Description:		This job step copies the report to OnBase.
Input/Output Files:		
clm0132o.rpt	I	Encounters Submission Statistics - On Request

2.5.53 CLMJQ058 - ClaimCheck Error Report

2.5.53.1 Detailed Job Script Information

Unix Script:	CLMJQ058
Description:	This job creates a savings report that indicates savings generated from clinical products errors, excluding denied claims that are to be rebilled. This report is generated quarterly. Produces Report: CLM-0058-Q

Job Step:		js 010-clm0058q
Description:		This job step creates the ClaimCheck Error Report.
Input/Output Files:		
clq05801.rpt	O	ClaimCheck Error Report
Job Step:		js 030-copy2routedir
Description:		This jobs step copies the report to OnBase.
Input/Output Files:		
clq05801.rpt	I	ClaimCheck Error Report
Job Step:		js 040-clm58del
Description:		This job step deletes some ClaimCheck savings data.
Input/Output Files:		

2.5.54 CLMJQ991 - Abortions Procedure Report

2.5.54.1 Detailed Job Script Information

Unix Script:	CLMJQ991
Description:	This job creates an abortions procedure report that shows paid abortion claims for the quarter. This report is generated quarterly. Produces Report: CLM-9914-Q

Job Step:		js 010-clm9914q
Description:		This job step creates the Abortions Procedure Report.
Input/Output Files:		
clm9914q.rpt	O	Quarterly Abortions Procedure Report
Job Step:		js 020-copy2routedir
Description:		This job step copies the report to OnBase.
Input/Output Files:		
clm9914q.rpt	I	Quarterly Abortions Procedure Report

2.5.55 CLMJSTOP - Shut Down the Claims Engine.

2.5.55.1 Detailed Job Script Information

Unix Script:	CLMJSTOP
Description:	This job shuts down the claims engine to prevent any changes to claims that is processed in the Financial cycle.

Job Step:	js 010-SoapSrvrMgr.sh
Description:	Stop the Soap Server
Input/Output Files:	
Job Step:	js 020-PosSrvrMgr.sh
Description:	Stop the POS Server
Input/Output Files:	
Job Step:	js 030-NcpdpSrvrMgr.sh
Description:	Stop the Ncpdp Server
Input/Output Files:	

2.5.56 CLMJSTRT - Starts the Claims Engine

2.5.56.1 Detailed Job Script Information

Unix Script:	CLMJSTRT
Description:	This job starts the SOAP server as well as the POS servers for 270/276, PE, PA and NCPDP. This is run after the financial history load has completed and after the daily claims refresh completes.

Job Step:	js 010-SoapSrvrMgr.sh
Description:	Start the Soap Server
Input/Output Files:	
Job Step:	js 020-PosSrvrMgr.sh
Description:	Start the POS Server for 270/276
Input/Output Files:	
Job Step:	js 030-PosSrvrMgr.sh
Description:	Start the PE Server
Input/Output Files:	
Job Step:	js 040-PosSrvrMgr.sh
Description:	Start the PA Server
Input/Output Files:	
Job Step:	js 050 PosSrvrMgr.sh
Description:	Start the NCPDP Server
Input/Output Files:	

2.5.57 CLMJW064 - Suspense Claims by Claim Type / Region Report

2.5.57.1 Detailed Job Script Information

Unix Script:	CLMJW064
Description:	This job creates a weekly report that reports the number of suspended claims by region and claim type. Produces Report: CLM-0064-W

Job Step:		js 010-clmpw064
Description:		This job step creates the Suspended Claims By Claim Type/Region.
Input/Output Files:		
clm0064.rpt	O	Suspended Claims by Claim Type/Region
Job Step:		js 020-sort
Description:		This job step sort the suspended claim extract file.
Input/Output Files:		
clm0064.rpt	I	Suspended Claims by Claim Type/Region
Job Step:		js 030-copy2routedir
Description:		This job step copies the report to OnBase.
Input/Output Files:		
clm0064.rpt	I	Suspended Claims by Claim Type/Region

2.5.58 CLMJW130 - Claims Submission Statistics - Weekly Report

2.5.58.1 Detailed Job Script Information

Unix Script:	CLMJW130
Description:	This job creates a report that provides a weekly summary of claim counts sent to the claims engine by submission type (WEB DDE, ASC x12). The information is reported at the summary level, and at the submitter level. Produces Report: CLM-0130-W

Job Step:		js 010-clm0130
Description:		This job step creates the Claims Submission Statistic-Weekly report.
Input/Output Files:		
clm0130w.rpt	O	Claims Submission Statistics - Weekly
Job Step:		js 030-copy2routedir
Description:		This job step copies the report to OnBase.
Input/Output Files:		
clm0130w.rpt	I	Claims Submission Statistics - Weekly

2.5.59 CLMJW860 - Pended Claims Aging

2.5.59.1 Detailed Job Script Information

Unix Script:	CLMJW860
Description:	This job creates a report that provides a total of pended claims documents by claim type and number of days in which they have been in a pended status. The submitted charge for the date range and a cumulative submitted charge are also shown on this report . Produces Report: CLM-8600-W

Job Step:		js 010-clm8600w
Description:		This job step creates the Pended Claims Aging Report
Input/Output Files:		
clm8600w.rpt	I	This file contains the output of report CLM-8600-W, the Pending Claims Aging Report
Job Step:		js 020-copy2routedir
Description:		This job step sends the Pended Claims Aging Report file to OnBase
Input/Output Files:		
clm8600w.rpt	O	This file contains the output of report CLM-8600-W, the Pending Claims Aging Report

2.5.60 CLRJD001 - Update daily claims report parameters

2.5.60.1 Detailed Job Script Information

Unix Script:	CLRJD001
Description:	The date parameters 1 is changed to the previous date and date parameters 2 is changed to the current date. The parameters for CLMDAILY are used claims daily reports.

Job Step:		js 010-clmdaily
Description:		Execute clmdaily
Input/Output Files:		
Not available	I	Not available

2.5.61 CRLJD001 - Creates Claim Link/Unlink Adjustments

2.5.61.1 Detailed Job Script Information

Unix Script:	CRLJD001
Description:	Creates Claim Link/Unlink Adjustments. This job executes crlpd001 which reads link transactions from the T_RECIP_LINK_XREF table and unlink transactions from the T_RE_CLM_DELINK table and creates adjustments for them on the T_ADJ_MASS_RQST and T_ADJ_MASS_CLAIM tables using claim information from the T_HIST_DIRECTORY table.
Job Step:	js 010-crlpd001
Description:	The job step executes crlpd001 which reads link transactions from the T_RECIP_LINK_XREF table and unlink transactions from the T_RE_CLM_DELINK table and creates adjustments for them on the T_ADJ_MASS_RQST and T_ADJ_MASS_CLAIM tables using claim information from the T_HIST_DIRECTORY table.
Input/Output Files:	

2.5.62 ENCJDNDY - Move previous day claims to rerun - Encounters

2.5.62.1 Detailed Job Script Information

Unix Script:	ENCJDNDY
Description:	The nightly Move Previous Day Claims to Re-run job executes numerous Unix shell commands to clean the encounters processing directories, as follows: • Moves any un-processed claims files (*.xml, *.?_xml, and *.rst files) from the current and previous months input directories (excluding current days input directory) into the current day's input directory. • Moves any *.xml files from *-hold directories (excluding current day's *-hold directory) into the newday directory and then deletes the *-hold directory. For directories with names like input/ccyymmdd* (for example input/20030601, input/20030601- bad, input/20030601-bkp) but excluding today's date, it performs as follows: • Moves any files from *-bad directories into the newday-bad directory. • Creates a tape archive (TAR) and compresses each daily backup (*-bkp) directory and then deletes the corresponding input/ccyymmdd* directories. For example, a TAR is created for directory /20030601-bkp and is compressed to create 20030601-bkp.tar.Z file, and then all /20030601* directories are deleted. Finally, it checks the newday-bad directory. If it contains any files, then it: • Executes UNIX shell script cycle_monitor.sh to determine the cycle monitor's user ID • Formats an E-mail listing of the files in the newday-bad directory and includes directions to investigate the bad files. • Sends E-mail to the user ID if the cycle monitor's user ID is an EDSNet ID (a user ID starting with dstn), otherwise it displays the contents of the E-mail on the system console.
Job Step:	js 010-UNIX
Description:	Execute Unix shell commands to clean-up claims processing directories.
Input/Output Files:	

2.5.63 SYSJDRDIST - Rdist the exe's to CORE

2.5.63.1 Detailed Job Script Information

Unix Script:	SYSJDRDIST
Description:	Rdist the exe's to CORE.

Job Step:	js 010-rdist.customer.ksh
Description:	This script redistributes the application code for the source master to the production servers.
Input/Output Files:	

2.6 Programs

2.6.1 ClaimDrvr.c -- Main Claim Driver Module

Technical Name:	ClaimDrvr.c
Program Title:	Main Claim Driver Module
Programming Language:	C
Description:	This is the main claim driver module that validates the processing of claims. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.1.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.2 PosSrvrMgr.sh -- This script will start, stop the Socket Listener based on the input parameter.

Technical Name:	PosSrvrMgr.sh
Program Title:	This script will start, stop the Socket Listener based on the input parameter.
Programming Language:	UNIX Script
Description:	This script will start, stop the POS Server based on the input parameter. PosSrvrMgr.sh is a UNIX script and not a program. No program documentation exists for this UNIX script.

2.6.2.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.3 RulesEngine -- Claims Rules Processor

Technical Name:	RulesEngine
Program Title:	Claims Rules Processor
Programming Language:	C
Description:	This is the special purpose rules engine for claims decision making.

2.6.3.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.4 Soap -- Soap

Technical Name:	Soap
Program Title:	Soap
Programming Language:	C
Description:	Determines service name based on XML node value, "" ==> claim service. Using config file, soapsrvr.nomt.conf, function to invoke is determined using service name, "claim". Typically this equates to the "soapProcess Claim" function. Transaction identifier is assigned at this point. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** • T_TRANSACTION_TYPE **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.4.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.5 adj837 -- Adjust 837 Claim And 837 Voids(Reversals)

Technical Name:	adj837
Program Title:	Adjust 837 Claim And 837 Voids(Reversals)
Programming Language:	C
Description:	This module contains functions for 837 Claim Adjustment, 837 voids and reversals. These functions are called from the WEB or EBX claims, from soapClaim.c engine. The adjustment uses the claimList.xsd schema to retrieve and load records to the database. This program will call adjMain.sc to perform the Voids (Reversals) and for Adjust 837 this program will create a new ICN and sak_claim then call (physMain, dntlMain, ub92Main, phrmMain). This will finish the 837 process. These two functions perform member, provider and ICN edits before calling the void function or creating an adjustment to be processed by the claims engine. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_SUSP_ADJ_XREF • T_ADJ_DENT_XREF • T_ADJ_PHRM_XREF • T_ADJ_PHYS_XREF • T_ADJ_UB92_XREF • T_PR_MCARE_BILL • T_HIST_DIRECTORY

	• T_TCN_XREF • T_RE_BASE • T_ADJ_MASS_CLAIM • T_ERROR_DISP **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** -lgutils, -lsocket, -lnsl, -lxml, -lxmlmrge, -lclmdrvr, -lclmbtch, -lclmcomm, -lclmphys, -lclmdntl, -lclmub92, -lclmphrm, -lclmwrite, -lclmhash **Special Logic Notes:** None

2.6.5.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.6 adjComm -- Adjustment Functions

Technical Name:	adjComm
Program Title:	Adjustment Functions
Programming Language:	C
Description:	This module contains routines used throughout the adjustment functions (adjDelete.sc, adjHistOnly.sc, adjInit.sc, adjMain.sc, adjQuery.sc, adjRecpLink.sc, adjRecycle.sc, adjRegular.sc, adjVoid.sc). **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_ADJ_MASS_RQST • T_ADJ_MASS_CLAIM • T_HIST_DIRECTORY • T_EOB • T_PD_DNTL_HDR • T_PD_PHARM_HDR • T_PD_PHYS_HDR • T_PD_UB92_HDR **Output Tables:** None **Sort Criteria:**

	None **Switches:** None at this time. **Link Procedure:** -lgutils, -lsocket, -lnsl, -lxml, -lxmlmrge, -lclmdrvr, -lclmbtch, -lclmcomm, -lclmphys, -lclmdntl, -lclmub92, -lclmphrm, -lclmwrite, -lclmhash **Special Logic Notes:** None

2.6.6.1 Change Orders

ID	NAME	DESCRIPTION
1795	Add num_dtl t_clm_patliab_x app	Modify the core objects that access t_clm_patliab_x to use detail number. Existing sql is based on patient liability applied at header level so the detail number to use when modifying existing objects is zero (0).
1797	App chgs t_*_phys_xover	Modify objects that access the t_susp_phys_xover, t_deny_phys_xover, t_pd_phys_xover tables to populate the num_dtl , also modify associated xsd's. Existing sql assumes values on these tables are header values, so num_dtl should be zero (0).

2.6.7 adjDelete -- Adjustment Delete Functions

Technical Name:	adjDelete
Program Title:	Adjustment Delete Functions
Programming Language:	C
Description:	The Adjustment Net Verification panels allow the user to mark a claim adjustment for deletion. This module contains functions related to deleting adjustment claims. The claims engine deletes from suspense before processing a claim. Once the adjustment has been deleted the Adjustment Net Verification panels will show a Deleted status for the deleted claim adjustment. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_HIST_DIRECTORY • T_SUSP_ADJ_XREF • T_ADJ_DENT_XREF • T_ADJ_PHRM_XREF • T_ADJ_PHYS_XREF • T_ADJ_UB92_XREF • T_CASH_RCPT_XREF **Output Tables:**

	• T_HIST_DIRECTORY • T_SUSP_ADJ_XREF • T_ADJ_DENT_XREF • T_ADJ_PHRM_XREF • T_ADJ_PHYS_XREF • T_ADJ_UB92_XREF • T_CASH_RCPT_DISP • T_CASH_RCPT_XREF • T_ADJ_MASS_CLAIM **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** -lgutils, -lsocket, -lnsl, -lxml, -lxmlmrge, -lclmdrvr, -lclmbtch, -lclmcomm, -lclmphys, -lclmdntl, -lclmub92, -lclmphrm, -lclmwrite, -lclmhash **Special Logic Notes:** None

2.6.7.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.8 adjHistOnly -- Adjustment TPL History Only (adjHistOnly)

Technical Name:	adjHistOnly
Program Title:	Adjustment TPL History Only (adjHistOnly)
Programming Language:	C
Description:	The Adjustment Request panel will allow the user to mark an adjustment as History Only, the user will be able to enter the TPL recovered amount. This module contains functions used for the history only function. This function creates a daughter, processes TPL, and writes to the claim tables. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_CLM_PGM_XREF • T_ADJ_MASS_CLAIM • T_RE_BASE • T_CASH_RECEIP • T_CASH_RCPT_DISP **Output Tables:** • T_CASH_RCPT_DISP • T_CASH_RCPT_XREF **Sort Criteria:** None

	Switches: None at this time. **Link Procedure:** -lgutils, -lsocket, -lnsl, -lxml, -lxmlmrge, -lclmdrvr, -lclmbtch, -lclmcomm, -lclmphys, -lclmdntl, -lclmub92, -lclmphrm, -lclmwrite, -lclmhash **Special Logic Notes:** None

2.6.8.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.9 adjInit -- Adjustment Environment Initialization

Technical Name:	adjInit
Program Title:	Adjustment Environment Initialization
Programming Language:	C
Description:	This module contains functions to set up the environment variables, process T_ADJ_RULES table loading the table rules for each form to create environment variables for cde_form ENVR and the claimDef pointer. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_ADJ_RULES **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** -lgutils, -lsocket, -lnsl, -lxml, -lxmlmrge, -lclmdrvr, -lclmbtch, -lclmcomm, -lclmphys, -lclmdntl, -lclmub92, -lclmphrm, -lclmwrite, -lclmhash **Special Logic Notes:**

	None

2.6.9.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.10 adjMain -- Adjustments Processing

Technical Name:	adjMain
Program Title:	Adjustments Processing
Programming Language:	C
Description:	The adjustments program processes adjustments by using the claimList.xsd schema to retrieve and load records to the database. The claimList schema relates all claim tables in parent to child relationships. When Adjustments unloads data related to a claim, all the data is loaded into an XML document. From there, the adjustment functions act on the data in the XML document (updating attributes to XML elements, removing XML elements or adding XML elements). When the data manipulation is complete for a particular adjustment function, either the document is loaded to the database or the document is written to an output file. For single adjustments, one claim is extracted. For mass adjustments many claims are pulled in one unload function, looped through and loaded to the suspense tables. The adjustment functions work from the panels or in a batch mode. The libadjxml.so shared library is registered as a stored procedure accessable to the panels. It is because of the panel access that adjustments uses the T_ADJ_RULES to establish its environment variables. This table also controls the translation functions of the single and mass adjustments. Although this program is called most often to execute single and mass adjustments functions, other functions such as history only or void and reversals follow the same basic steps: unload using a sak claim, change the data according to the functions needs, and load the data back out. Input Parameters: • with_context - not used • rqst - the number of the mass request in the table T_ADJ_MASS_CLAIM • sak - the request, which identifies the function to call • seq - used to pass the Internal Control Number (ICN) sequence for regular adjustments; other functions use this to pass sak claims • test - allows batch programs to simulate function calls without

	committing to the database • msg - stores and accumulates return m

2.6.10.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.11 adjQuery -- Adjustment Query Functions

Technical Name:	adjQuery
Program Title:	Adjustment Query Functions
Programming Language:	C
Description:	This module contains functions related to creating a query document. When executed the data will be loaded data into the claimList schema to create an XML document to process. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** -lgutils, -lsocket, -lnsl, -lxml, -lxmlmrge, -lclmdrvr, -lclmbtch, -lclmcomm, -lclmphys, -lclmdntl, -lclmub92, -lclmphrm, -lclmwrite, -lclmhash **Special Logic Notes:** None

2.6.11.1 Change Orders

ID	NAME	DESCRIPTION
1795	Add num_dtl t_clm_patliab_x app	Modify the core objects that access t_clm_patliab_x to use detail number. Existing sql is based on patient liability applied at header level so the detail number to use when modifying existing objects is zero (0).
1796	App chgs t_*_ub92_xover	Modify objects that insert to the t_susp_ub92_xover, t_deny_ub92_xover, t_pd_ub92_xover tables to populate the amt_paid_mcare and amt_alwd_mcare columns being added, also modify associated xsd's. Evaluate objects that select data from these tables to determine if any modification is required.
1797	App chgs t_*_phys_xover	Modify objects that access the t_susp_phys_xover, t_deny_phys_xover, t_pd_phys_xover tables to populate the num_dtl , also modify associated xsd's. Existing sql assumes values on these tables are header values, so num_dtl should be zero (0).

2.6.12 adjRecpLink -- Adjustment Recipient Link Functions

Technical Name:	adjRecpLink
Program Title:	Adjustment Recipient Link Functions
Programming Language:	C
Description:	This module contains functions used for the member link function. This function creates a daughter for each linked claim, updates the member data on the daughter claim, and writes it to the paid tables. Load and execute queryDoc (delink or link), verify member. For each claim, bypass if previously adjusted, assign a new sak and ICN, delete current locations and create new locations. For each claim inactivate the mother, create an xref record, update the daughter with new member data, update elig element at header or detail. Remove invalid claims from document and load document into database with remaining claims. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_RE_CLM_DELINK • T_RECIP_LINK_XREF • T_RE_BASE • T_RE_ELIG • T_RE_AID_ELIG **Output Tables:** None

	Sort Criteria: None **Switches:** None at this time. **Link Procedure:** -lgutils, -lsocket, -lnsl, -lxml, -lxmlmrge, -lclmdrvr, -lclmbtch, -lclmcomm, -lclmphys, -lclmdntl, -lclmub92, -lclmphrm, -lclmwrite, -lclmhash **Special Logic Notes:** None

2.6.12.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.13 adjRecycle -- Adjustment Recycle Extract Functions

Technical Name:	adjRecycle
Program Title:	Adjustment Recycle Extract Functions
Programming Language:	C
Description:	This module contains functions used for the claim recycle extract function. The two main routines are called as hooks from xmlSqlMergeElements as part of clmpdr01.sc. testr.sc also creates the extract file in a local directory without updating the database. The Data Correction panel provides the resolution user with the information needed to resolve the suspended claim these claims are marked with an `R' Save With Resubmit. The claims marked with `R' resubmit will be process here. Adjustments marked for resubmit will be extracted. At the end of adjRecycle, write the closing tag. Rename XML document file from recycle directory to newday. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_SUSP_ADJ_XREF • T_ADJ_MASS_CLAIM • T_CLAIM_LOCAT **Output Tables:** • T_HIST_DIRECTORY **Sort Criteria:** None

	Switches: None at this time. **Link Procedure:** -lgutils, -lsocket, -lnsl, -lxml, -lxmlmrge, -lclmdrvr, -lclmbtch, -lclmcomm, -lclmphys, -lclmdntl, -lclmub92, -lclmphrm, -lclmwrite, -lclmhash **Special Logic Notes:** None

2.6.13.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.14 adjRegular -- Adjustment Regular Functions

Technical Name:	adjRegular
Program Title:	Adjustment Regular Functions
Programming Language:	C
Description:	This module contains functions for single and mass adjustments. These functions translate elements based on the elements rules defined on the t_adj_rules table. Regular_adjustments load and execute queryDoc translate to suspense based on adjustment rules. Assign ICN, load to database, then set final status. If fail, set error status and rollback. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_PA_ITEM_DTL_XREF **Output Tables:** • T_PA_ITEM_DTL_XREF • T_ADJ_MASS_CLAIM **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** -lgutils, -lsocket, -lnsl, -lxml, -lxmlmrge, -lclmdrvr, -lclmbtch, -

	lclmcomm, -lclmphys, -lclmdntl, -lclmub92, -lclmphrm, -lclmwrite, -lclmhash **Special Logic Notes:** None

2.6.14.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.15 adjVoid -- Void And Reversal Functions

Technical Name:	adjVoid
Program Title:	Void And Reversal Functions
Programming Language:	C
Description:	This module contains functions for voids and reversals. These functions are called from the panels, claims engine and batch programs. Check if pharmacy reversal, populate history globals, load and execute queryDoc. Check if claims have been previously adjusted, for claim form invalidate mother claim, load xref records, assign ICN, and zero out Financial and pa records. Remove locations and create 97 and 66 locations, prepare values for and load daughter to database. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_ADJ_MASS_CLAIM **Output Tables:** • T_ADJ_MASS_CLAIM **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** -lgutils, -lsocket, -lnsl, -lxml, -lxmlmrge, -lclmdrvr, -lclmbtch, -lclmcomm, -lclmphys, -lclmdntl, -lclmub92, -lclmphrm, -lclmwrite, -

	lclmhash **Special Logic Notes:** None

2.6.15.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.16 aimGlobal -- Utility Functions

Technical Name:	aimGlobal
Program Title:	Utility Functions
Programming Language:	C
Description:	Utility functions Utility functions for the claims engine reside here. These functions include date validations, number manipulations, and character field validity. Functions to copy field's values from the input XML to storage areas within the claim engine reside here. Stack functions for navigating the limit audit history memory area are stored here. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** -lclmcomm **Special Logic Notes:** None

2.6.16.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.17 aim_pswd -- Get AIM_PSWD

Technical Name:	aim_pswd
Program Title:	Get AIM_PSWD
Programming Language:	UNIX Command
Description:	This command stream uses setenv, grep and awk to get the AIM_PSWD from the system environment This is a UNIX command only, not a program. No program documentation exists for this UNIX script.

2.6.17.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.18 claimList.c -- claimList

Technical Name:	claimList.c
Program Title:	claimList
Programming Language:	C
Description:	Claim schema for the xml input file. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input tables** None **Output tables** • t_hist_directory • t_pd_phys_hdr • t_phys_dext_key • t_clm_amt • t_clm_cr2 • t_clm_oth_pyr_dtl • t_clm_modifier • t_clm_cas • t_clm_pyr_mod • t_clm_qty • t_clm_entity • t_clm_ref

	• t_party_identifier • t_clm_ent_nm_adr • t_clm_adr_n3_n4 • t_clm_name_nm1 • t_clm_cr5 • t_clm_cn1 • t_clm_cr1 • t_clm_cr3 • t_clm_k3 • t_clm_hsd • T_CLM_PWK • T_CLM_NTE • T_CLM_CRC • T_CLM_DTP • T_CLM_MEA • t_claim_error • t_clm_eob_xref • t_clm_pgm_xref • t_susp_rlhx • t_pa_item_dtl_xref • t_mphx_phys_dtl • t_deny_mphx_phys • t_mphx_ph_diag • t_pd_phys_dtl • t_adj_phys_xref • t_susp_adj_xref • t_phys_hdr_key • t_clm_hh_care_plan

	• t_clm_sbr • t_clm_moa • t_clm_payer_entity • t_clm_pyr_entnmadr • t_prof_oth_pyr_hdr • t_edi_interchange • t_final_phys_xover • t_clm_diag_xref • t_claim_locat • t_clm_hdr_delivery • t_clm_patliab_x • t_susp_phys_hdr • t_susp_phys_dtl • t_susp_phys_xover • t_deny_phys_hdr • t_deny_phys_dtl • t_pd_ub92_hdr • t_ub92_dtl_elig • t_ub92_dtl_ext_key • t_medpol_ub92 • t_pd_ub92_dtl • t_adj_ub92_xref • t_ub92_hdr_elig • t_ub92_hdr_cond_x • t_ub92_hdr_occ • t_ub92_hdr_ext_key • t_clm_cr6 • t_clm_mia

	• t_inst_oth_pyr_hdr

2.6.18.1 Change Orders

ID	NAME	DESCRIPTION
1796	App chgs t_*_ub92_xover	Modify objects that insert to the t_susp_ub92_xover, t_deny_ub92_xover, t_pd_ub92_xover tables to populate the amt_paid_mcare and amt_alwd_mcare columns being added, also modify associated xsd's. Evaluate objects that select data from these tables to determine if any modification is required.
1797	App chgs t_*_phys_xover	Modify objects that access the t_susp_phys_xover, t_deny_phys_xover, t_pd_phys_xover tables to populate the num_dtl , also modify associated xsd's. Existing sql assumes values on these tables are header values, so num_dtl should be zero (0).
1798	App chgs t_pharm_hdr_keys	Modify objects that insert into the t_pharm_hdr_keys table to populate the sak_edi_interchange column being added, also modify associated xsd's. Evaluate objects that select data from these tables to determine if any modification is required.

2.6.19 claimList.h -- claimList Header

Technical Name:	claimList.h
Program Title:	claimList Header
Programming Language:	C
Description:	The claimList header file.

2.6.19.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.20 claimList.xsd -- claimList Schema

Technical Name:	claimList.xsd
Program Title:	claimList Schema
Programming Language:	Other
Description:	The claim list schema defines the relationship between claim tables and xml node names. It is used by the claims engine to control the processing and insertion of claim data to the database. This is a schema only, not a program. No program documentation exists for this system object.

2.6.20.1 Change Orders

ID	NAME	DESCRIPTION
1795	Add num_dtl t_clm_patliab_x app	Modify the core objects that access t_clm_patliab_x to use detail number. Existing sql is based on patient liability applied at header level so the detail number to use when modifying existing objects is zero (0).
1796	App chgs t_*_ub92_xover	Modify objects that insert to the t_susp_ub92_xover, t_deny_ub92_xover, t_pd_ub92_xover tables to populate the amt_paid_mcare and amt_alwd_mcare columns being added, also modify associated xsd's. Evaluate objects that select data from these tables to determine if any modification is required.
1797	App chgs t_*_phys_xover	Modify objects that access the t_susp_phys_xover, t_deny_phys_xover, t_pd_phys_xover tables to populate the num_dtl , also modify associated xsd's. Existing sql assumes values on these tables are header values, so num_dtl should be zero (0).
1798	App chgs t_pharm_hdr_keys	Modify objects that insert into the t_pharm_hdr_keys table to populate the sak_edi_interchange column being added, also modify associated xsd's. Evaluate objects that select data from these tables to determine if any modification is required.

2.6.21 claimcheck.csh -- ClaimCheck Runtime Environment Variable Set Script

Technical Name:	claimcheck.csh
Program Title:	ClaimCheck Runtime Environment Variable Set Script
Programming Language:	UNIX Script
Description:	This script is used in the runtime (claims cycle) environment and is necessary for setting ClaimCheck environment variables. This must be sourced in the claims cycle job or other jobs that process a claim through the claims engine. claimcheck.csh is a UNIX script and not a program. No program documentation exist for this UNIX script

2.6.21.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.22 clm0010d -- Online Adjustment Daily Report

Technical Name:	clm0010d
Program Title:	Online Adjustment Daily Report
Programming Language:	C
Description:	The Online Adjustment Daily Report program creates the Online Adjustment Daily Report CLM- 0010-D, which lists the online updates made to the credit adjustment data. Input Parameters: • The name of the output file • date_parm_1 • dte_parm_2 • time_system • tme_assigned Exit Values: • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors Input Files: None Output Files: clm0010D.rpt Input Tables: • T_SYSTEM_PARMS • T_ADJ_MASS_CLAIM • T_PR_PROV • T_RE_BASE • T_EOB • T_ADJ_PHYS_XREF

	• T_ADJ_DENT_XREF • T_ADJ_PHRM_XREF • T_ADJ_UB92_XREF • T_SUSP_ADJ_XREF • T_CLAIM_LOCAT Output Tables: None Sort Criteria: None Switches: None Link Procedure: Link with hmspconn. Special Logic Notes: None

2.6.22.1 Change Orders

ID	NAME	DESCRIPTION
1109	RPTS- CLM-0010-D Add MassAdjRqNr	Add the mass adjustment request number on the Daily Claim Adjustment Detail List
163	RPTS - Change "Recipient"	Change every instance of "Recipient", "Current ID" or "Medicaid ID" to "Member ID" on all claim reports.
2045	Add Standard Date Heading Line	It is critical that the headers are clarified and coded to exact specification. There are three potential page widths, 150 columns, 132 columns and 80 columns. Most reports are 132 columns. 80 and 150 column reports should be the exception. There are three primary header lines followed by a fourth optional line. The optional line is the date range line, this line should have just the begin and end date WITHOUT A LABEL PREFIX. If a report is a daily report it will have just one date. The EXACT specifications with column positions are below in the supplemental

ID	NAME	DESCRIPTION
		documentation. If a report name, run date or report date is off by even one character position, the report will be rejected by the OnBase import process and will need to be reworked. I have provided examples of reports of all three widths for both daily reports and those with a date range. Please note there are two tabs in the attachment.
2083	Re-Code State Specific Changes	Add the method of incorporating state specific coding requirements within an existing core module to report programs.

2.6.23 clm0011d -- Clerk ID Recycle Claims Report

Technical Name:	clm0011d
Program Title:	Clerk ID Recycle Claims Report
Programming Language:	C
Description:	This program generates report CLM-0011-D. This report lists the number of errors each clerk ID has worked in the Suspense Table. **Input Parameters:** • date_parm_1 • dte_parm_2 • time_system • time_assigned When program name (nam_program) is equal to 'CLMDAREP'. **Exit Values:** • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** clm01101.rpt **Input Tables:** • T_SYSTEM_PARMS • T_CLAIM_LOCAT • AIM01.T_CLAIM_LOCAT • T_PD_DNTL_HDR • T_PD_PHARM_HDR • T_PD_PHYS_HDR • T_PD_UB92_HDR

	• T_DENY_DNTL_HDR • T_DENY_PHARM_HDR • T_DENY_PHYS_HDR • T_DENY_UB92_HDR • T_PD_DNTL_DTL • T_PD_PHARM_DTL • T_PD_PHYS_DTL • T_PD_UB92_DTL • T_DENY_DNTL_DTL • T_DENY_PHARM_DTL • T_DENY_PHYS_DTL • T_DENY_UB92_DTL • T_SUSP_DNTL_DTL • T_SUSP_PHARM_DTL • T_SUSP_PHYS_DTL • T_SUSP_UB92_DTL • T_CLAIM_ERROR • T_CLERK (Temporary table) • T_RECYCLED_CLAIM(Temporary table) • T_MANUAL_PRICED_EDITS(Temporary table) • T_FORCED_EDITS(Temporary table) • T_DENIED_EDITS(Temporary table) • T_OTHER_EDITS(Temporary table) **Output Tables:** • T_CLERK (Temporary table) • T_RECYCLED_CLAIM(Temporary table) • T_MANUAL_PRICED_EDITS(Temporary table) • T_FORCED_EDITS(Temporary table)

	• T_DENIED_EDITS(Temporary table) • T_OTHER_EDITS(Temporary table) **Sort Criteria:** Clerk ID **Switches:** None at this time. **Link Procedure:** None

2.6.23.1 Change Orders

ID	NAME	DESCRIPTION
2045	Add Standard Date Heading Line	It is critical that the headers are clarified and coded to exact specification. There are three potential page widths, 150 columns, 132 columns and 80 columns. Most reports are 132 columns. 80 and 150 column reports should be the exception. There are three primary header lines followed by a fourth optional line. The optional line is the date range line, this line should have just the begin and end date WITHOUT A LABEL PREFIX. If a report is a daily report it will have just one date. The EXACT specifications with column positions are below in the supplemental documentation. If a report name, run date or report date is off by even one character position, the report will be rejected by the OnBase import process and will need to be reworked. I have provided examples of reports of all three widths for both daily reports and those with a date range. Please note there are two tabs in the attachment.

2.6.24 clm0012d -- EOB Listing

Technical Name:	clm0012d
Program Title:	EOB Listing
Programming Language:	C
Description:	This program generates report CLM-0012-D. This report lists each EOB code and its description. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** • clm01201.rpt **Input Tables:** • T_EOB **Output Tables:** None **Sort Criteria:** • EOB Code. **Switches:** None at this time. **Link Procedure:** Link with libmrpf. **Special Logic Notes:** There are no special logic notes for this program.

2.6.24.1 Change Orders

ID	NAME	DESCRIPTION
1070	RPTS - Add Standard Footer	Add the standard footer to report.
2045	Add Standard Date Heading Line	It is critical that the headers are clarified and coded to exact specification. There are three potential page widths, 150 columns, 132 columns and 80 columns. Most reports are 132 columns. 80 and 150 column reports should be the exception. There are three primary header lines followed by a fourth optional line. The optional line is the date range line, this line should have just the begin and end date WITHOUT A LABEL PREFIX. If a report is a daily report it will have just one date. The EXACT specifications with column positions is below in the supplemental documentation. If a report name, run date or report date is off by even one character position, the report will be rejected by the OnBase import process and will need to be reworked. I have provided examples of reports of all three widths for both daily reports and those with a date range. Please note there are two tabs in the attachment.
785	RPTS - CLM0012D-CHG HDG to EOB	This report lists each EOB code and its description. Change the column heading of 'EDIT/AUDIT CODE' to be EOB.

2.6.25 clm0015d_cc -- Claim Check Potential History To Adjust

Technical Name:	clm0015d_cc
Program Title:	Claim Check Potential History To Adjust
Programming Language:	C
Description:	This program creates report CLM-0015-D. This report contains claim detail lines audited by claimcheck, which can require adjustment if from paid history. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** clm01501.rpt **Input Tables:** • T_CC_HIST_ADJUST • T_SYSTEM_PARMS • T_PD_DNTL_HDR • T_PD_PHARM_HDR • T_PD_PHYS_HDR • T_PD_UB92_HDR • T_DENY_DNTL_HDR • T_DENY_PHRM_HDR • T_DENY_PHYS_HDR • T_DENY_UB92_HDR **Output Tables:**

	None **Sort Criteria:** • System assigned key for claim's internal control number. • Date on which services were first performed for a member. **Switches:** None at this time. **Link Procedure:** Link with hmspconn **Special Logic Notes:** • When the SAK_CLAIM value changes need to print the report page and reset for the next Claim. • History/Current indicator is : ▪ P - Processed claim is the first row of the report. ▪ A - Affected claims rows go in the History Claim Lines Potentially Affected section. The 'A' is hard coded to a value of 'H' on the report because it's considered a History Line. ▪ C - Current claims rows go in the Other Current & History Claim lines section of the report. ▪ H - History claims rows go in the Other Current & History Claim lines section of the report. </

2.6.25.1 Change Orders

ID	NAME	DESCRIPTION
1070	RPTS - Add Standard Footer	Add the standard footer to report. This KY CO is duplicated in the Core PWB. See Core CO 12077 - all documentation will be updated on this CO.
163	RPTS - Change "Recipient"	Change every instance of "Recipient", "Current ID" or "Medicaid ID" to "Member ID" on all claim reports.
1732	Report Writer - Standard Footer	The report writer program has a routine which creates the standard footers for reports. Some of the claims reports have the standard footers hard coded in the programs rather than letting report writer create them. For maintainability all claim

ID	NAME	DESCRIPTION
		report programs should use report writer to create the standard footers.
2012	RPTS - Correct Source Code	In preparation for the reports work group sessions it was determined that some of the core reports were not working correctly. The Kentucky decision was to pull these reports from other states. Please see the list of reports and the state which will provide the affected reports in the Technical Specifications section.
2045	Add Standard Date Heading Line	It is critical that the headers are clarified and coded to exact specification. There are three potential page widths, 150 columns, 132 columns and 80 columns. Most reports are 132 columns. 80 and 150 column reports should be the exception. There are three primary header lines followed by a fourth optional line. The optional line is the date range line, this line should have just the begin and end date WITHOUT A LABEL PREFIX. If a report is a daily report it will have just one date. The EXACT specifications with column positions are below in the supplemental documentation. If a report name, run date or report date is off by even one character position, the report will be rejected by the OnBase import process and will need to be reworked. I have provided examples of reports of all three widths for both daily reports and those with a date range. Please note there are two tabs in the attachment.
2083	Re-Code State Specific Changes	Add the method of incorporating state specific coding requirements within an existing core module to report programs.
3938	CLM-0015-D: Claimcheck Potential	The process of the report on the PWB (CLMJD015) does not match with the OnBase report (CLMJDC015). JOB CLMJD015 is the correct process for this report.
565	RPTS - CHG Exceptions/Audit-Edit	Terminology need to be consistent through the reports when referring to "exceptions/errors/edit-audit number" The field should be changed from "Exception, Edit-Audit" to "Error code" and "Error description" and from "Edit" to "Error".

2.6.26 clm0018d -- Claims Processing Daily Summary

Technical Name:	clm0018d
Program Title:	Claims Processing Daily Summary
Programming Language:	C
Description:	The Adjustment Credit Claims Processing Daily Summary program creates the following reports listing summary information by claim type and medical provider type for claims daily cycle, weekly cycle or claims from history: • CLM-0016-* - Claims Processing Summary (summarizes data from CLM-0017-* and CLM- 0018-*) • CLM-0017-D - Original Claims Processing Summary (Daily cycle only) • CLM-0018-D - Adjustment/Credit Claims Processing Summary (Daily cycle only) Where * = D for daily, W for weekly or H for histo **Input Parameters:** • Command line parameter of D, W or H (for daily cycle, weekly cycle or claims from history respectively) is required • For claims from history, the payment issue date can be specified as an optional command line parameter. If is not specified, then Dte_Parm_2 retrieved from T_SYSTEM_PARMS where nam_program is equal to FINCYCLE is used as the payment issue date • Date_Parm_1, Dte_Parm_2, Time_System, Tme_Assigned are retrieved from T_SYSTEM_PARMS where nam_program is equal to CLMDAREP (daily cycle), or CLMBAL_W (weekly cycle) or CLMBAL_H (claims from history) **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to invalid input parameters, error opening output report file or database error **Input Files:** None

	Output Files: CLMJD018: • clm01601.rpt • clm01701.rpt • clm01801.rpt CLMJH018: • clm01601h.rpt CLMJW018: • clm01601w.rpt **Input Tables:** • DUAL • T_CLAIM_LOCAT • T_CLM_PGM_XREF • T_DENY_DNTL_HDR • T_DENY_PHRM_HDR • T_DENY_PHYS_HDR • T_DENY_UB92_HDR • T_PD_DNTL_HDR • T_PD_PHARM_HDR • T_PD_PHYS_HDR • T_PD_UB92_HDR • T_SUSP_DENTAL_HDR • T_SUSP_PHRM_HDR • T_SUSP_PHYS_HDR • T_SUSP_UB92_HDR • T_SYSTEM_PARMS **Output Tables:** None **Sort Criteria:** No

2.6.26.1 Change Orders

ID	NAME	DESCRIPTION
1070	RPTS - Add Standard Footer	Add the standard footer to report. This KY CO is duplicated in the Core PWB. See Core CO 12077 - all documentation will be updated on this CO.
2012	RPTS - Correct Source Code	In preparation for the reports work group sessions it was determined that some of the core reports were not working correctly. The Kentucky decision was to pull these reports from other states. Please see the list of reports and the state which will provide the affected reports in the Technical Specifications section.
4251	CLM-0018D: Adj Credit Claims	CLM0018D: 1. This is a daily report. The second page of the layout of the PWB has a from -thru date range. From - Thru date range is not present on the OnBase . 2. The title of the report on the first page and the second pages is "Adjustment Credit Claims Processing Daily Summary" and on PWB the second page is "Adj/ Credit Claims Processing Daily Summary by Medical Provider".

2.6.27 clm0020d -- Aged Detail Suspense Report

Technical Name:	clm0020d
Program Title:	Aged Detail Suspense Report
Programming Language:	C
Description:	This program generates CLM-0020-D. This report is a detail listings by claim type of claims that have been in the suspense file longer than a certain number of days specified by parameter. **Input Parameters:** • Number of days aged **Exit Values:** • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** clm02001.rpt **Input Tables:** • T_CLAIM_LOCAT • T_CLAIM_TYPE • T_CLAIM_ERROR • T_ERROR_DISP • T_RE_BASE • T_PR_PROV • T_SUSP_DNTL_HDR • T_SUSP_PHARM_HDR • T_SUSP_PHYS_HDR • T_SUSP_UB92_HDR

	• SUSP_CLAIM (Temporary table) • CURR_CLAIM (Temporary table) **Output Tables:** • SUSP_CLAIM (Temporary table) • CURR_CLAIM (Temporary table) **Sort Criteria:** • Claim Type • Claim Number • Error code • Claim age(Days) **Switches:** None at this time. **Link Procedure:** Link with lgutils and libmrpf. **Special Logic Notes:** • New page if the Claim type has changed. • The program creates and uses two temporary tables SUSP_CLAIM, CURR_CLAIM from a Union of all the tables listed under input tables above. The report is created by doing a selects on these temporary tables. The tables are dropped at the end of the program.

2.6.27.1 Change Orders

ID	NAME	DESCRIPTION
1070	RPTS - Add Standard Footer	Add the standard footer to report. This KY CO is duplicated in the Core PWB. See Core CO 12077 - all documentation will be updated on this CO.
163	RPTS - Change "Recipient"	Change every instance of "Recipient", "Current ID" or "Medicaid ID" to "Member ID" on all claim reports.

ID	NAME	DESCRIPTION
2045	Add Standard Date Heading Line	It is critical that the headers are clarified and coded to exact specification. There are three potential page widths, 150 columns, 132 columns and 80 columns. Most reports are 132 columns. 80 and 150 column reports should be the exception. There are three primary header lines followed by a fourth optional line. The optional line is the date range line, this line should have just the begin and end date WITHOUT A LABEL PREFIX. If a report is a daily report it will have just one date. The EXACT specifications with column positions are below in the supplemental documentation. If a report name, run date or report date is off by even one character position, the report will be rejected by the OnBase import process and will need to be reworked. I have provided examples of reports of all three widths for both daily reports and those with a date range. Please note there are two tabs in the attachment.
2083	Re-Code State Specific Changes	Add the method of incorporating state specific coding requirements within an existing core module to report programs.
429	RPTS - Remove HCA from Heading	Take "HCA" out of report title.
565	RPTS - CHG Exceptions/Audit-Edit	Terminology need to be consistent through the reports when referring to "exceptions/errors/edit-audit number" The field should be changed from "Exception, Edit-Audit" to "Error code" and "Error description" and from "Edit" to "Error".

2.6.28 clm0021d -- Suspense File Analysis By Error Code

Technical Name:	clm0021d
Program Title:	Suspense File Analysis By Error Code
Programming Language:	C
Description:	This program generates CLM-0021-D (SUSPENSE FILE ANALYSIS BY ERROR CODE) . This report provides a count by error code of claims that were suspended during the previous daily claims cycle. This report also provides the cumulative total by error code. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** clm02101.rpt **Input Tables:** • T_CLAIM_LOCAT • T_CLAIM_TYPE • T_CLAIM_ERROR • T_ERROR_DISP • T_SUSP_DNTL_HDR • T_SUSP_PHRM_HDR • T_SUSP_PHYS_HDR • T_SUSP_UB92_HDR • T_SYSTEM_PARMS **Output Tables:** None

	Sort Criteria: • Exception Code • Claim Type **Switches:** None at this time. **Link Procedure:** Link with lgutils and libmrpf. **Special Logic Notes:** Location code not in ('98', '97', '66').

2.6.28.1 Change Orders

ID	NAME	DESCRIPTION
2012	RPTS - Correct Source Code	In preparation for the reports work group sessions it was determined that some of the core reports were not working correctly. The Kentucky decision was to pull these reports from other states. Please see the list of reports and the state which will provide the affected reports in the Technical Specifications section.
2045	Add Standard Date Heading Line	It is critical that the headers are clarified and coded to exact specification. There are three potential page widths, 150 columns, 132 columns and 80 columns. Most reports are 132 columns. 80 and 150 column reports should be the exception. There are three primary header lines followed by a fourth optional line. The optional line is the date range line, this line should have just the begin and end date WITHOUT A LABEL PREFIX. If a report is a daily report it will have just one date. The EXACT specifications with column positions are below in the supplemental documentation. If a report name, run date or report date is off by even one character position, the report will be rejected by the OnBase import process and will need to be reworked. I have provided examples of reports of all three widths for both daily reports and those with a date range. Please note there are two tabs in the attachment.

ID	NAME	DESCRIPTION
2083	Re-Code State Specific Changes	Add the method of incorporating state specific coding requirements within an existing core module to report programs.
565	RPTS - CHG Exceptions/Audit-Edit	Terminology need to be consistent through the reports when referring to "exceptions/errors/edit-audit number" The field should be changed from "Exception, Edit-Audit" to "Error code" and "Error description" and from "Edit" to "Error".

2.6.29 clm0023d -- Claims Error Report (FFS)

Technical Name:	clm0023d
Program Title:	Claims Error Report (FFS)
Programming Language:	C
Description:	This program generates report CLM-0023-D. This report is used by the Claims Resolution Unit as an inventory of claims to be resolved. Detailed error reports by claim type are used by clerks to resolve claims. . **Input Parameters:** Parameters retrieved from T_SYSTEM_PARMS where name of the program is 'AIMCYCLE' • date_parm_1 • dte_parm_2 **Exit Values:** • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** clm02301.rpt **Input Tables:** • T_DENY_DNTL_HDR • T_DENY_PHRM_HDR • T_DENY_PHYS_HDR • T_DENY_UB92_HDR • T_PD_DNTL_HDR • T_PD_PHARM_HDR • T_PD_PHYS_HDR • T_PD_UB92_HDR

	• T_SUSP_DNTL_HDR • T_SUSP_PHARM_HDR • T_SUSP_PHYS_HDR • T_SUSP_UB92_HDR • T_SYSTEM_PARMS • T_PD_UB92_DTL • T_DENY_UB92_DTL • T_SUSP_UB92_DTL • T_RE_BASE • T_ERROR_DISP • T_CLAIM_ERROR • T_TEMP_PRINTTYPEL (Temporary Table) • T_ERROR_DISP_LINE • T_CLM_PGM_XREF **Output Tables:** None **Sort Criteria:** • Location • ICN **Switches:** None at this time. **Link Procedure:** Link with lmrpf. **Special Logic Notes:** Detailed Exception Reports by claim type are used by clerks to resolve claims.

2.6.29.1 Change Orders

ID	NAME	DESCRIPTION
163	RPTS - Change "Recipient"	Change every instance of "Recipient", "Current ID" or "Medicaid ID" to "Member ID" on all claim

ID	NAME	DESCRIPTION
		reports.
2045	Add Standard Date Heading Line	It is critical that the headers are clarified and coded to exact specification. There are three potential page widths, 150 columns, 132 columns and 80 columns. Most reports are 132 columns. 80 and 150 column reports should be the exception. There are three primary header lines followed by a fourth optional line. The optional line is the date range line, this line should have just the begin and end date WITHOUT A LABEL PREFIX. If a report is a daily report it will have just one date. The EXACT specifications with column positions are below in the supplemental documentation. If a report name, run date or report date is off by even one character position, the report will be rejected by the OnBase import process and will need to be reworked. I have provided examples of reports of all three widths for both daily reports and those with a date range. Please note there are two tabs in the attachment.
2083	Re-Code State Specific Changes	Add the method of incorporating state specific coding requirements within an existing core module to report programs.
565	RPTS - CHG Exceptions/Audit-Edit	Terminology need to be consistent through the reports when referring to "exceptions/errors/edit-audit number" The field should be changed from "Exception, Edit-Audit" to "Error code" and "Error description" and from "Edit" to "Error".

2.6.30 clm0025d -- Services To Be Clerically Denied Report

Technical Name:	clm0025d
Program Title:	Services To Be Clerically Denied Report
Programming Language:	C
Description:	This program generates CLM-0025-D. This report lists the claims that had services denied by a clerk, thereby enabling supervisory personnel to identify those clerks who appear to resolve an inordinate number of suspended claims by denying error codes. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Input Tables:** • T_SYSTEM_PARMS • T_CLAIM_ERROR • T_PD_PHYS_HDR • T_SUSP_PHYS_HDR • T_DENY_PHYS_HDR • T_PD_DNTL_HDR • T_SUSP_DENTAL_HDR • T_DENY_DNTL_HDR • T_PD_PHARM_HDR • T_SUSP_PHRM_HDR • T_DENY_PHRM_HDR • T_PD_UB92_HDR

	• T_PD_UB92_DTL • T_SUSP_UB92_HDR • T_SUSP_UB92_DTL • T_DENY_UB92_HDR • T_DENY_UB92_DTL • T_PR_TYPE • T_PR_PROV • T_RE_BASE • T_CLAIM_ERROR • T_ERROR_DISP **Output Tables:** None **Sort Criteria:** • ICN • DETAIL NUMBER **Switches:** None at this time. **Link Procedure** Link with libmrpf and libeutil. **Special Logic Notes:** There are no special logic notes for this program.

2.6.30.1 Change Orders

ID	NAME	DESCRIPTION
163	RPTS - Change "Recipient"	Change every instance of "Recipient", "Current ID" or "Medicaid ID" to "Member ID" on all claim reports.
2045	Add Standard Date Heading Line	It is critical that the headers are clarified and coded to exact specification. There are three potential page widths, 150 columns, 132 columns and 80 columns. Most reports are 132 columns. 80 and 150 column reports should be the exception.

ID	NAME	DESCRIPTION
		There are three primary header lines followed by a fourth optional line. The optional line is the date range line, this line should have just the begin and end date WITHOUT A LABEL PREFIX. If a report is a daily report it will have just one date. The EXACT specifications with column positions are below in the supplemental documentation. If a report name, run date or report date is off by even one character position, the report will be rejected by the OnBase import process and will need to be reworked. I have provided examples of reports of all three widths for both daily reports and those with a date range. Please note there are two tabs in the attachment.
2083	Re-Code State Specific Changes	Add the method of incorporating state specific coding requirements within an existing core module to report programs.
7345	Calculation error CLM-0025-D	The report is not calculating the total claim count in the Clerk ID Recap area correctly.
8500	CLMJD025 - Provider Type Abend	CLMJD025 is abending sporadically in the production daily cycle. It abends with a sql error -- ORA-01405: fetched column value is NULL.

2.6.31 clm0026d -- Claims Paid Due To Force Report

Technical Name:	clm0026d
Program Title:	Claims Paid Due To Force Report
Programming Language:	C
Description:	The Claims Paid Due to Force Report program creates report CLM-0026-D, listing claims having errors that were clerically forced or overridden, thereby enabling supervisory personnel to identify clerks who appear to resolve an inordinate number of suspended claims by forcing errors codes. **Input parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** Clm0026D.rpt **Input Tables:** • T_SYSTEM_PARMS • T_CLAIM_ERROR • T_HIST_DIRECTORY • T_PD_PHYS_HDR • T_PD_DNTL_HDR • T_PD_PHARM_HDR • T_PD_UB92_HDR • T_PR_TYPE • T_PR_PROV • T_RE_BASE

	• T_CLAIM_ERROR • T_ERROR_DISP **Output Tables:** None **Sort Criteria:** • ICN • DETAIL NUMBER **Switches:** None at this time. **Link Procedures:** Link with libmrpf and libeutil. **Special Logic Notes:** There are no special logic notes for this program.

2.6.31.1 Change Orders

ID	NAME	DESCRIPTION
163	RPTS - Change "Recipient"	Change every instance of "Recipient", "Current ID" or "Medicaid ID" to "Member ID" on all claim reports.
2045	Add Standard Date Heading Line	It is critical that the headers are clarified and coded to exact specification. There are three potential page widths, 150 columns, 132 columns and 80 columns. Most reports are 132 columns. 80 and 150 column reports should be the exception. There are three primary header lines followed by a fourth optional line. The optional line is the date range line, this line should have just the begin and end date WITHOUT A LABEL PREFIX. If a report is a daily report it will have just one date. The EXACT specifications with column positions are below in the supplemental documentation. If a report name, run date or report date is off by even one character position, the report will be rejected by the OnBase import process and will need to be reworked. I have provided examples of reports of all three widths for both daily reports and those with a date range. Please note there are two tabs in the attachment.

ID	NAME	DESCRIPTION
2083	Re-Code State Specific Changes	Add the method of incorporating state specific coding requirements within an existing core module to report programs.

2.6.32 clm0027d -- Transaction Register

Technical Name:	clm0027d
Program Title:	Transaction Register
Programming Language:	C
Description:	This program generates CLM-0027-D. The purpose of this report is to identify every claim processed in a daily claim adjudication cycle. Each claim is identified by ICN, Member ID Number, Provider Number, Transaction Code, Dates of Service, and Payment Amount. Additionally, this report will display any error codes posted to the claim throughout the daily claim adjudication cycle. **Input Parameters:** Parameters retrieved from T_SYSTEM_PARMS where name of the program is 'AIMCYCLE' • date_parm_1 • dte_parm_2 **Exit Values:** • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** clm02701.rpt **Input Tables:** • T_SYSTEM_PARMS • T_CLAIM_LOCAT • T_CLAIM_ERROR • T_ERROR_DISP • T_RE_BASE • T_PR_PROV

	• T_PD_PHYS_HDR • T_DENY_PHYS_HDR • T_PD_DNTL_HDR • T_DENY_DNTL_HDR • T_PD_UB92_HDR • T_DENY_UB92_HDR • T_PD_PHARM_HDR • T_DENY_PHRM_HDR • T_SUSP_PHYS_HDR • T_SUSP_DENTAL_HDR • T_SUSP_PHRM_HDR • T_SUSP_UB92_HDR **Output Tables:** None **Sort Criteria:** • ICN **Switches:** None at this time. **Link Procedure:** Link with lmrpf. **Special Logic Notes:** None

2.6.32.1 Change Orders

ID	NAME	DESCRIPTION
163	RPTS - Change "Recipient"	Change every instance of "Recipient", "Current ID" or "Medicaid ID" to "Member ID" on all claim reports.
2045	Add Standard Date Heading Line	It is critical that the headers are clarified and coded to exact specification. There are three potential page widths, 150 columns, 132 columns and 80 columns. Most reports are 132 columns. 80 and

ID	NAME	DESCRIPTION
		150 column reports should be the exception. There are three primary header lines followed by a fourth optional line. The optional line is the date range line, this line should have just the begin and end date WITHOUT A LABEL PREFIX. If a report is a daily report it will have just one date. The EXACT specifications with column positions are below in the supplemental documentation. If a report name, run date or report date is off by even one character position, the report will be rejected by the OnBase import process and will need to be reworked. I have provided examples of reports of all three widths for both daily reports and those with a date range. Please note there are two tabs in the attachment.
2083	Re-Code State Specific Changes	Add the method of incorporating state specific coding requirements within an existing core module to report programs.
565	RPTS - CHG Exceptions/Audit-Edit	Terminology need to be consistent through the reports when referring to "exceptions/errors/edit-audit number" The field should be changed from "Exception, Edit-Audit" to "Error code" and "Error description" and from "Edit" to "Error".

2.6.33 clm0031m -- Suspense File Analysis By Claim Type

Technical Name:	clm0031m
Program Title:	Suspense File Analysis By Claim Type
Programming Language:	C
Description:	This program generates CLM-0031-M. This report will print a suspense file analysis by claim type of current and last fiscal years. The second page of the report will be a break out by Provider type with CMS 1500. **Input Parameters:** The name of the report file. Exit Values: EXIT_SUCCESS - normal termination. EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors Input Files: None Output Files: clm03101.rpt Input Tables: • T_SYSTEM_PARMS • T_MR_PROV_SUSP Output Tables: None Sort Criteria: • CDE_CLM_TYPE for the claim type section • CDE_PROV_TYPE for the provider type section Switches: None Link Procedure: Link with lmrpf. Special Logic Notes: None

2.6.33.1 Change Orders

ID	NAME	DESCRIPTION
1070	RPTS - Add Standard Footer	Add the standard footer to report. This KY CO is duplicated in the Core PWB. See Core CO 12077 - all documentation will be updated on this CO.
2045	Add Standard Date Heading Line	It is critical that the headers are clarified and coded to exact specification. There are three potential page widths, 150 columns, 132 columns and 80 columns. Most reports are 132 columns. 80 and 150 column reports should be the exception. There are three primary header lines followed by a fourth optional line. The optional line is the date range line, this line should have just the begin and end date WITHOUT A LABEL PREFIX. If a report is a daily report it will have just one date. The EXACT specifications with column positions are below in the supplemental documentation. If a report name, run date or report date is off by even one character position, the report will be rejected by the OnBase import process and will need to be reworked. I have provided examples of reports of all three widths for both daily reports and those with a date range. Please note there are two tabs in the attachment.

2.6.34 clm0050m -- Retro Active Rate Adjustment Report

Technical Name:	clm0050m
Program Title:	Retro Active Rate Adjustment Report
Programming Language:	C
Description:	This program generates CLM-0050-M. This report lists the Retro Active Rate Adjustments by member. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - Normal termination. • EXIT_FAILURE - Abnormal termination because of open, read, allocation, or input errors **Input Files:** None **Output Files:** clm05001.rpt **Input Tables:** • T_SYSTEM_PARMS • T_PD_PHYS_HDR • T_PR_PROV • T_RE_BASE • T_PD_PHARM_HDR • T_PD_DNTL_HDR • T_PD_UB92_HDR • T_CLM_PATLIAB_X • T_ADJ_PHYS_XREF • T_EOB • T_CLAIM_LOCAT **Output Tables:**

	None **Sort Criteria:** • ID_MEDICAID **Switches:** None at this time **Link Procedure:** libmrpf, and libgutil. **Special Logic Notes:** None

2.6.34.1 Change Orders

ID	NAME	DESCRIPTION
163	RPTS - Change "Recipient"	Change every instance of "Recipient", "Current ID" or "Medicaid ID" to "Member ID" on all claim reports.
1795	Add num_dtl t_clm_patliab_x app	Modify the core objects that access t_clm_patliab_x to use detail number. Existing sql is based on patient liability applied at header level so the detail number to use when modifying existing objects is zero (0).
2083	Re-Code State Specific Changes	Add the method of incorporating state specific coding requirements within an existing core module to report programs.
3882	CLM-0050-M: Retroactive Rate Adj	This is a monthly report and the date heading mm/dd/ccyy - mm/dd/ccyy is missing on the report . This From - Thru date heading is present on the PWB. The member id and member name are on separate lines on OnBase but on same row on PWB.

2.6.35 clm0051d -- Aged Detail Adjustment Report

Technical Name:	clm0051d
Program Title:	Aged Detail Adjustment Report
Programming Language:	C
Description:	This program generates CLM-0051-D report (AGED DETAIL ADJUSTMENT REPORT) to display adjustments that have aged. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** clm05101.rpt **Input Tables:** • T_CLAIM_LOCAT • T_SUSP_DENTAL_HDR • T_SUSP_PHRM_HDR • T_SUSP_PHYS_HDR • T_SUSP_UB92_HDR • T_CLAIM_TYPE • T_CLAIM_ERROR • T_ERROR_DISP • T_RE_BASE • T_PR_PROV • SUSP_CLAIM(Temporary table) • CURR_CLAIM(Temporary table)

	Output Tables: • SUSP_CLAIM(Temporary table) • CURR_CLAIM(Temporary table) **Sort Criteria:** • Claim Type • Claim Age (Days) • ICN • Edits **Switches:** None at this time. **Link Procedure:** Link with lmrpf and lgutils. **Special Logic Notes:** • Number of days aged is set to 30. • Start new page for the report if the claim type has changed. • The program creates and uses temporary tables SUSP_CLAIM and CURR_CLAIM from a Union of all the tables listed under input tables above. The report is created by doing a selects on these temporary tables. The tables are dropped at the end of the program.

2.6.35.1 Change Orders

ID	NAME	DESCRIPTION
163	RPTS - Change "Recipient"	Change every instance of "Recipient", "Current ID" or "Medicaid ID" to "Member ID" on all claim reports.
2045	Add Standard Date Heading Line	It is critical that the headers are clarified and coded to exact specification. There are three potential page widths, 150 columns, 132 columns and 80 columns. Most reports are 132 columns. 80 and 150 column reports should be the exception. There are three primary header lines followed by a

ID	NAME	DESCRIPTION
		fourth optional line. The optional line is the date range line, this line should have just the begin and end date WITHOUT A LABEL PREFIX. If a report is a daily report it will have just one date. The EXACT specifications with column positions are below in the supplemental documentation. If a report name, run date or report date is off by even one character position, the report will be rejected by the OnBase import process and will need to be reworked. I have provided examples of reports of all three widths for both daily reports and those with a date range. Please note there are two tabs in the attachment.
2083	Re-Code State Specific Changes	Add the method of incorporating state specific coding requirements within an existing core module to report programs.
429	RPTS - Remove HCA from Heading	Take "HCA" out of report title.
565	RPTS - CHG Exceptions/Audit-Edit	Terminology need to be consistent through the reports when referring to "exceptions/errors/edit-audit number" The field should be changed from "Exception, Edit-Audit" to "Error code" and "Error description" and from "Edit" to "Error".

2.6.36 clm0054d -- Error Recycle Parameters

Technical Name:	clm0054d
Program Title:	Error Recycle Parameters
Programming Language:	C
Description:	The program generates CLM-0054-D. This report lists each suspense transaction input to the system. It also prints appropriate error messages. This report is by CDE_ESC. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** clm0054d.rpt **Input Tables:** • T_EDIT_RECYCLE • T_ERROR_DISP **Output Tables:** None **Sort Criteria:** • Edit/Audit Number • Claim Type **Switches:** None at this time. **Link Procedure:** Link with lmrpf and lgutils. **Special Logic Notes:**

	None

2.6.36.1 Change Orders

ID	NAME	DESCRIPTION
1082	RPTS - CLM0054D-Separate Report	The Commonwealth would like this report to be split into two sections - one specific for edits and one specific for audits.
2045	Add Standard Date Heading Line	It is critical that the headers are clarified and coded to exact specification. There are three potential page widths, 150 columns, 132 columns and 80 columns. Most reports are 132 columns. 80 and 150 column reports should be the exception. There are three primary header lines followed by a fourth optional line. The optional line is the date range line, this line should have just the begin and end date WITHOUT A LABEL PREFIX. If a report is a daily report it will have just one date. The EXACT specifications with column positions are below in the supplemental documentation. If a report name, run date or report date is off by even one character position, the report will be rejected by the OnBase import process and will need to be reworked. I have provided examples of reports of all three widths for both daily reports and those with a date range. Please note there are two tabs in the attachment.
2083	Re-Code State Specific Changes	Add the method of incorporating state specific coding requirements within an existing core module to report programs.
4253	CLM0054D: Error recycle parameter	The column headings and the data under each column are not aligned as per the PWB. For example The column Error code has error code numbers listed below and the data is not displayed under the heading.
565	RPTS - CHG Exceptions/Audit-Edit	Terminology need to be consistent through the reports when referring to "exceptions/errors/edit-audit number" The field should be changed from "Exception, Edit-Audit" to "Error code" and "Error

ID	NAME	DESCRIPTION
		description" and from "Edit" to "Error".

2.6.37 clm0058q -- ClaimCheck Error

Technical Name:	clm0058q
Program Title:	ClaimCheck Error
Programming Language:	C
Description:	This program generates CLM-0058-Q. the savings report that indicates savings generated from clinical products errors, excluding denied claims that are to be re-billed. This report will be generated quarterly. Not all edits will be included on this report. Only edits that is non-duplicated between MMIS errors and ClaimCheck. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** clq05801.rpt **Input Tables:** • T_ERROR_DISP • T_CC_INDICATORS • T_SYSTEM_PARMS • DUAL(Dummy table used to view SYSDATE) • T_SUSP_PHYS_HDR • T_PD_PHYS_HDR • T_DENY_PHYS_HDR • T_CC_CR_SAVE • T_HIST_DIRECTORY • T_DENY_MPHX_PHYS

	Output Tables: None **Sort Criteria:** • CDE_ESC for the error code type section. **Switches:** None **Link Procedure:** Link with lmrpf **Special Logic Notes:** None

2.6.37.1 Change Orders

ID	NAME	DESCRIPTION
1070	RPTS - Add Standard Footer	Add the standard footer to report. This KY CO is duplicated in the Core PWB. See Core CO 12077 - all documentation will be updated on this CO.
2045	Add Standard Date Heading Line	It is critical that the headers are clarified and coded to exact specification. There are three potential page widths, 150 columns, 132 columns and 80 columns. Most reports are 132 columns. 80 and 150 column reports should be the exception. There are three primary header lines followed by a fourth optional line. The optional line is the date range line, this line should have just the begin and end date WITHOUT A LABEL PREFIX. If a report is a daily report it will have just one date. The EXACT specifications with column positions are below in the supplemental documentation. If a report name, run date or report date is off by even one character position, the report will be rejected by the OnBase import process and will need to be reworked. I have provided examples of reports of all three widths for both daily reports and those with a date range. Please note there are two tabs in the attachment.
2083	Re-Code State Specific Changes	Add the method of incorporating state specific

ID	NAME	DESCRIPTION
		coding requirements within an existing core module to report programs.
4035	Add indClaimCheck-UB92 Claims	Add the indClaimCheck field for UB claims.
565	RPTS - CHG Exceptions/Audit-Edit	Terminology need to be consistent through the reports when referring to "exceptions/errors/edit-audit number" The field should be changed from "Exception, Edit-Audit" to "Error code" and "Error description" and from "Edit" to "Error".

2.6.38 clm0060d -- Suspense File Analysis By Claim Type

Technical Name:	clm0060d
Program Title:	Suspense File Analysis By Claim Type
Programming Language:	C
Description:	The Suspense File Analysis by Claim Type program creates the report Suspense File Analysis By Claim Type 3 CLM-0060-D, which lists the number and dollar value totals of the various claim types currently in the suspense file. Claims numbers and totals are accumulated for the categories of current day, 1 to 30 days old, 31 to 60 days old, 61 to 90 days old, and over 90 days old. The average claims per day are presented for each claim type and all claim types. Note the content of the report is from the Suspense Tables as updated from the Suspense File. Adjustments are excluded from this report. The program is best viewed on a 150 column window. **Input Parameters:** The name of the output file From T_SYSTEM_PARMS: dte_parm_2 with the condition that NAM_PROGRAM = CLMDAREP **Exit Values:** • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** clm06001.rpt **Input Tables:** • T_SYSTEM_PARMS • T_SUSP_PHYS_HDR • T_SUSP_DENTAL_HDR • T_SUSP_PHRM_HDR • T_SUSP_UB92_HDR **Output Tables:** None

	Sort Criteria: None **Switches:** None at this time. **Link Procedure:** Link with hmspconn and dsindate. **Special Logic Notes:** None

2.6.38.1 Change Orders

ID	NAME	DESCRIPTION
1070	RPTS - Add Standard Footer	Add the standard footer to report. This KY CO is duplicated in the Core PWB. See Core CO 12077 - all documentation will be updated on this CO.
2045	Add Standard Date Heading Line	It is critical that the headers are clarified and coded to exact specification. There are three potential page widths, 150 columns, 132 columns and 80 columns. Most reports are 132 columns. 80 and 150 column reports should be the exception. There are three primary header lines followed by a fourth optional line. The optional line is the date range line, this line should have just the begin and end date WITHOUT A LABEL PREFIX. If a report is a daily report it will have just one date. The EXACT specifications with column positions are below in the supplemental documentation. If a report name, run date or report date is off by even one character position, the report will be rejected by the OnBase import process and will need to be reworked. I have provided examples of reports of all three widths for both daily reports and those with a date range. Please note there are two tabs in the attachment.

2.6.39 clm0065d -- Aged Suspended Claims By Location Report

Technical Name:	clm0065d
Program Title:	Aged Suspended Claims By Location Report
Programming Language:	C
Description:	This program creates report CLM-0065-D which lists the totals for aged claims that are 1 to 30 days old, 31 to 60 days old, 61 to 90 days old, and over 90 days old. It also lists the totals for each location, each aged claims period for all the locations, and the grand total. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** clm06501.rpt **Input Tables:** • T_CLAIM_LOCAT • T_SYSTEM_PARMS • DUAL (Dummy table used to view SYSDATE) • T_SUSP_UB92_HDR • T_SUSP_PHYS_HDR • T_SUSP_DENTAL_HDR • T_SUSP_PHRM_HDR **Output Tables:** None **Sort Criteria:**

	• CDE_LOCATION for the claim location section. • NUM_ICN for the icn number section. **Switches:** None **Link Procedure:** Link with lmrpf **Special Logic Notes:** None

2.6.39.1 Change Orders

ID	NAME	DESCRIPTION
1070	RPTS - Add Standard Footer	Add the standard footer to report. This KY CO is duplicated in the Core PWB. See Core CO 12077 - all documentation will be updated on this CO.
2045	Add Standard Date Heading Line	It is critical that the headers are clarified and coded to exact specification. There are three potential page widths, 150 columns, 132 columns and 80 columns. Most reports are 132 columns. 80 and 150 column reports should be the exception. There are three primary header lines followed by a fourth optional line. The optional line is the date range line, this line should have just the begin and end date WITHOUT A LABEL PREFIX. If a report is a daily report it will have just one date. The EXACT specifications with column positions are below in the supplemental documentation. If a report name, run date or report date is off by even one character position, the report will be rejected by the OnBase import process and will need to be reworked. I have provided examples of reports of all three widths for both daily reports and those with a date range. Please note there are two tabs in the attachment.

2.6.40 clm0099d -- Captiva Received Rejected

Technical Name:	clm0099d
Program Title:	Captiva Received Rejected
Programming Language:	C
Description:	This Program Generates CLM-0099-D. This report compares the output from the paper claims data capture system (Captiva) to the input in the system. The report identifies any missing Internal Control Numbers (ICNs) and displays them on this report. **Input Parameters:** • Frequency(Daily) • Start Date (YYYYMMDD) **Exit Values:** • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** • Files with extension TXT on $DATADIR/claims/newicn **Output Files:** • clm0099d.rpt **Input Tables:** • T_CLAIM_LOCAT • T_SYSTEM_PARMS **Output Tables:** None **Sort Criteria:** None

	Switches: None **Link Procedure:** None **Special Logic Notes:** None

2.6.40.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.41 clm0110d -- Online Mass Adjustment Report

Technical Name:	clm0110d
Program Title:	Online Mass Adjustment Report
Programming Language:	C
Description:	The Online Mass Adjustment Report program creates the Daily Report CLM- 0110-D, which lists the mass adjustment online updates that have been made. Input Parameters: • The name of the output file • date_parm_1 • dte_parm_2 • time_system Exit Values: • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors Input Files: None Output Files: clm0110D.rpt Input Tables: • T_SYSTEM_PARMS • T_ADJ_MASS_CLAIM • T_RE_BASE • T_EOB • T_ADJ_PHYS_XREF • T_ADJ_DENT_XREF • T_ADJ_PHRM_XREF

	• T_ADJ_UB92_XREF • T_SUSP_ADJ_XREF • T_CLAIM_LOCAT Output Tables: None Sort Criteria: None Switches: None Link Procedure: Link with hmspconn. Special Logic Notes: None

2.6.41.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.42 clm0123d -- Claims Error Report (Encounters)

Technical Name:	clm0123d
Program Title:	Claims Error Report (Encounters)
Programming Language:	C
Description:	This program generates report CLM-0123-D. This report is used by the Claims Resolution Unit as an inventory of encounters to be resolved. Detailed error reports by claim type are used by clerks to resolve encounters. . **Input Parameters:** Parameters retrieved from T_SYSTEM_PARMS where name of the program is 'AIMCYCLE' • date_parm_1 • dte_parm_2 **Exit Values:** • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** • clm12301.rpt **Input Tables:** • T_DENY_DNTL_HDR • T_DENY_PHRM_HDR • T_DENY_PHYS_HDR • T_DENY_UB92_HDR • T_PD_DNTL_HDR • T_PD_PHARM_HDR • T_PD_PHYS_HDR • T_PD_UB92_HDR • T_SUSP_DNTL_HDR

	• T_SUSP_PHARM_HDR • T_SUSP_PHYS_HDR • T_SUSP_UB92_HDR • T_SYSTEM_PARMS • T_PD_UB92_DTL • T_DENY_UB92_DTL • T_SUSP_UB92_DTL • T_RE_BASE • T_ERROR_DISP • T_CLAIM_ERROR • T_TEMP_PRINTTYPEL (Temporary Table) • T_ERROR_DISP_LINE • T_CLM_PGM_XREF **Output Tables:** None **Sort Criteria:** • Location • ICN **Switches:** None **Link Procedure:** Link with lmrpf. **Special Logic Notes:** Detailed Exception Reports by claim type are used by clerks to resolve claims.

2.6.42.1 Change Orders

ID	NAME	DESCRIPTION
2232	CLM0023D - Add Encounter Section	There was a request from the Commonwealth (Tammy Bullock) that encounters should be included on the CLM-0023-D(Claims Error Report). This report should be divided into two sections. One section should report on the FFS claims and the other section should report on the encounter claims.

2.6.43 clm0130 -- Claims Submission Statistics

Technical Name:	clm0130
Program Title:	Claims Submission Statistics
Programming Language:	C
Description:	This program generates CLM-0130-D, CLM-0130-W and CLM-0130-M. These reports provide a daily summary of claim counts sent to the claims engine by submission type (NCPDP, WEB DDE, and ASC x12). Reports at the summary level, and at the submitter level. . The frequency parameter for this program can be 'd' for daily, 'w' for weekly and 'm' for monthly. **Input Parameters:** • Frequency(Weekly, Daily, Monthly) • Start Date (YYYYMMDD) **Exit Values:** • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** • CLM0130D : None • CLM0130W :None • CLM0130M :None **Output Files:** • clm0130D.rpt • clm0130W.rpt • clm0130M.rpt **Input Tables:** • T_BATCH

	• T_CLAIM_LOCAT • T_CLAIM_TYPE • T_DENTAL_HDR_KEYS • T_HIST_DIRECTORY • T_PHRM_HDR_KEYS • T_PHYS_HDR_KEY • T_UB92_HDR_EXT_KEY • T_TRANSACTION **Output Tables:** None **Sort Criteria:** • T_BATCH.CDE_SUBMIT • T_CLAIM_TYPE.DSC_CLM_TYP • T_HIST_DIRECTORY.CDE_CLM_STATUS Submitter • T_BATCH.SENDER • T_BATCH.CDE_SUBMIT • T_CLAIM_TYPE.DSC_CLM_TYP • T_HIST_DIRECTORY.CDE_CLM_STATUS **Switches:** None **Link Procedure:** Link with Link with libclmbtch, libclmdrvr, liblbms, libclmhash, libxml, libmrpf, and libeutil. **Special Logic Notes:** The report will create a: • (D)aily • (W)eekly • (M)onthly

	Based on Frequency and Start Date, the program will determine the date range, and create the report for that range.

2.6.43.1 Change Orders

ID	NAME	DESCRIPTION
2012	RPTS - Correct Source Code	In preparation for the reports work group sessions it was determined that some of the core reports were not working correctly. The Kentucky decision was to pull these reports from other states. Please see the list of reports and the state which will provide the affected reports in the Technical Specifications section.
2045	Add Standard Date Heading Line	It is critical that the headers are clarified and coded to exact specification. There are three potential page widths, 150 columns, 132 columns and 80 columns. Most reports are 132 columns. 80 and 150 column reports should be the exception. There are three primary header lines followed by a fourth optional line. The optional line is the date range line, this line should have just the begin and end date WITHOUT A LABEL PREFIX. If a report is a daily report it will have just one date. The EXACT specifications with column positions are below in the supplemental documentation. If a report name, run date or report date is off by even one character position, the report will be rejected by the OnBase import process and will need to be reworked. I have provided examples of reports of all three widths for both daily reports and those with a date range. Please note there are two tabs in the attachment.
2071	CLM0130-Add Non-Adjudicated CLMS	A Non-adjudicated claims count should be added to the CLM-0130 reports.
665	RPTS - Change EDI- to CLM-	The submission statistics reports are produced from claim tables and not EDI tables. Claims update the information for these reports. The report names, program names, and jobs names should be changed to CLM.

ID	NAME	DESCRIPTION
7083	CLM-0130-D has incorrect data	Reports CLM-0130-D, CLM-0130-W, and CLM-0130-M data is not matching up between the summary and submitter levels. It seems to be leaving out certain claim types at the submitter level (i.e. Dental claims). The numbers should match between the two levels. There are some submission types missing from the report. Per the PWB, this is to report a daily summary of claim counts sent to the claims engine by submission type (NCPDP, WEB DDE, ASC x12 and RRI). What is meant by "Non-Standard"? There is data in the summary level that is not broken down by the submitter level on the report (i.e. WEB DDE, 837D, etc...).

2.6.44 clm0132 -- Encounters Submission Statistics

Technical Name:	clm0132
Program Title:	Encounters Submission Statistics
Programming Language:	C
Description:	The program creates the daily, monthly, and weekly Encounters Submission Statistics Reports which lists summary counts of Electronic Data interChange (EDI) encounters sent to the claims engine by submission type and by submitter. Paper and RRI encounters are not processed by this program. **Input Parameters:** Command line parameter of D, W or M for daily, weekly, or monthly cycle respectively. The report dates are determined from the system date based on the cycle type. For the daily cycle, both the from and to dates are the system date. For the weekly cycle, the to date is the system date and the from date is the system date minus six days. For the monthly cycle, the from date is the first of the month containing the system date, and the to date is the last day of that same month. **Exit Values:** • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** • CLM0132D : None • CLM0132W :None • CLM0132M :None **Output Files:** • clm0132D.rpt • clm0132W.rpt • clm0132M.rpt **Input Tables:** • DUAL (Dummy table used to view SYSDATE)

	• T_BATCH • T_CLAIM_LOCAT • T_CLAIM_TYPE • T_DENTAL_HDR_KEYS • T_DENY_DNTL_HDR • T_DENY_PHRM_HDR • T_DENY_PHYS_HDR • T_DENY_UB92_HDR • T_HIST_DIRECTORY • T_PD_DNTL_HDR • T_PD_PHARM_HDR • T_PD_PHYS_HDR • T_PD_UB92_HDR • T_PHRM_HDR_KEYS • T_PHYS_HDR_KEY • T_UB92_HDR_EXT_KEY **Output Tables:** None **Sort Criteria:** The first section of the report • T_BATCH.CDE_SUBMIT • T_CLAIM_TYPE.DSC_CLM_TYP • T_HIST_DIRECTORY.CDE_CLM_STATUS • T_DENY_DNTL_HDR.CDE_CLAIM_FREQUENCY The second section of the report • T_PD_PHARM_HDR.ID_SUBMITTER</L

2.6.44.1 Change Orders

ID	NAME	DESCRIPTION
2012	RPTS - Correct Source Code	In preparation for the reports work group sessions it was determined that some of the core reports were not working correctly. The Kentucky decision was to pull these reports from other states. Please see the list of reports and the state which will provide the affected reports in the Technical Specifications section.
2045	Add Standard Date Heading Line	It is critical that the headers are clarified and coded to exact specification. There are three potential page widths, 150 columns, 132 columns and 80 columns. Most reports are 132 columns. 80 and 150 column reports should be the exception. There are three primary header lines followed by a fourth optional line. The optional line is the date range line, this line should have just the begin and end date WITHOUT A LABEL PREFIX. If a report is a daily report it will have just one date. The EXACT specifications with column positions are below in the supplemental documentation. If a report name, run date or report date is off by even one character position, the report will be rejected by the OnBase import process and will need to be reworked. I have provided examples of reports of all three widths for both daily reports and those with a date range. Please note there are two tabs in the attachment.
665	RPTS - Change EDI- to CLM-	The submission statistics reports are produced from claim tables and not EDI tables. Claims update the information for these reports. The report names, program names, and jobs names should be changed to CLM.

2.6.45 clm0140d -- Encounter Batch Summary

Technical Name:	clm0140d
Program Title:	Encounter Batch Summary
Programming Language:	C
Description:	The Encounter Batch Summary program creates the Encounter Batch Summary Report CLM-0140-D, which reports how the Managed Care Organization's entire batch file of encounter claims processed. Input Parameters: • The name of the output file • date_parm_1 • dte_parm_2 Exit Values: • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors Input Files: None Output Files: clm0140D.rpt Input Tables: • T_ENC_BATCH_PROCESS • T_FILE_TRACK • T_DOCUMENT_TRACK • T_TP • T_CDE_TP_TYPE Output Tables: None Sort Criteria:

	None Switches: None Link Procedure: None Special Logic Notes: None

2.6.45.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.46 clm140d -- Encounters Batch Summary

Technical Name:	clm140d
Program Title:	Encounters Batch Summary
Programming Language:	C
Description:	Create a daily claims report showing how the entire batch from a MCO was processed.

2.6.46.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.47 clm15del -- ClmChk PHTAR Delete Data

Technical Name:	clm15del
Program Title:	ClmChk PHTAR Delete Data
Programming Language:	C
Description:	This program is associated with clm0015d: Potential History To Adjust Report (PHTAR). This program will delete the Claim Check PHTAR Data used for the report. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_SYSTEM_PARMS • T_CC_HIST_ADJUST **Output Tables:** • T_CC_HIST_ADJUST **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** Link with libmrpf **Special Logic Notes:** • Set number of records to purge at a time 'DEL_REC_COUNT' to

	10000 rows. • Keep only one month of data in the PHTAR table - t_cc_hist_adjust .

2.6.47.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.48 clm5000d -- Top Ten Claims Approved For Payment

Technical Name:	clm5000d
Program Title:	Top Ten Claims Approved For Payment
Programming Language:	C
Description:	This program generates CLM-5000-D, which lists the top ten claims approved for payment for each claim type for the adjudication cycle **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to error opening output file or input file(s) **Input Files:** None **Output Files:** clm5000d.rpt **Input Tables:** • T_SYSTEM_PARMS • T_CLAIM_LOCAT • T_CLAIM_TYPE • T_PD_PHYS_HDR • T_PD_DNTL_HDR • T_PD_UB92_HDR • T_PD_PHARM_HDR • T_RE_BASE • T_PR_TYPE • T_PR_PROV **Output Tables:** None

	Sort Criteria: None **Switches:** None **Link Procedure:** Link with lmrpf **Special Logic Notes:** None

2.6.48.1 Change Orders

ID	NAME	DESCRIPTION
1110	RPTS - CLM-5000-D Create Report	EDS will create a process to generate Top Ten Claims Approved For Payment report.
4326	CLM:5000D-Top 10 Claims Approved	CLM:5000D is a daily report. The provider id data is not displayed on the report on OnBase. CO 2078 was implemented to display the NPI provider id for this report.
7058	CLM-5000-D Header Date	On the CLM-5000-D the date within the header is not centered.

2.6.49 clm58del -- ClaimCheck Delete Savings Data

Technical Name:	clm58del
Program Title:	ClaimCheck Delete Savings Data
Programming Language:	C
Description:	This program archives old data from the ClaimCheck Savings Table. The amount of data to keep is stored on the system parms data and it is the number of months to keep. This program is associated with the clm0058q: ClaimCheck Savings Report. This program will delete the ClaimCheck Savings Data used for the Savings Report. Input Parameters: None Exit Values: • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_SYSTEM_PARMS • T_HIST_DIRECTORY • T_CC_CR_SAVE **Output Tables:** • T_CC_CR_SAVE **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:**

	Link with libmrpf **Special Logic Notes:** It will delete "chunks" of data at a time. For example, 50000 rows to delete, it will delete 5000 rows at a time. The number of rows to purge is DEL_REC_COUNT.

2.6.49.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.50 clm8600w -- Pended Claims Aging Report

Technical Name:	clm8600w
Program Title:	Pended Claims Aging Report
Programming Language:	C
Description:	This program generates CLM-8600-W, which provides a total of pended claims documents by claim type and number of days in which they have been in a pended status. The submitted charge for the date range and a cumulative submitted charge are shown on this report. Aging on the iC report creates break age range dates in 30-day buckets. The last age group that is reported is for suspended claims older than 210 days. A final claim type group of 'ALL' concludes the report. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to error opening output file or input file(s) **Input Files:** None **Output Files:** clm8600w.rpt **Input Tables:** • T_SYSTEM_PARMS • T_SUSP_PHYS_HDR • T_SUSP_DENTAL_HDR • T_SUSP_UB92_HDR • T_SUSP_PHRM_HDR **Output Tables:** None **Sort Criteria:** None

	Switches: None **Link Procedure:** Link with lmrpf **Special Logic Notes:** None

2.6.50.1 Change Orders

ID	NAME	DESCRIPTION
1122	RPTS - CLM-8600-W Create Report	Create the Pended Claims Aging Report in interChange.
4282	CLM-8600W: PENDED CLAIMS AGING	CLM-8600-W is a weekly report and the From - Thru date range is not displayed on the Onbase report . PWB has a date range.
7162	CLM-8600-W has incorrect totals	Date span appears to be incorrect and the cumulative figures for claim type "all" appears to be duplicating.

2.6.51 clm9914q -- Abortions Procedure Report

Technical Name:	clm9914q
Program Title:	Abortions Procedure Report
Programming Language:	C
Description:	This program generates CLM-9914-Q, Abortions Procedure Report. This report will be generated quarterly. The claims displayed on this report will be selected by a procedure code grouping, SAK_PROC_TYPE=1114. Procedures may be added or removed over time. • The initial CPT abortion procedure codes are 59840, 59841, 59850, 59851, 59855, 59856, 59100, 59852, 59857, and 59866. • The SAK_PROC_TYPE code to use for grouping abortion codes in T_PROC_TYPE, T_PROC_Group and T_PROC_CODE_GROUP will be 1114. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** clm9914q.rpt **Input Tables:** • DUAL(Dummy table used to view SYSDATE) • T_SYSTEM_PARMS • T_PROC • T_PROC_CODE_GROUP • AIM01.T_PD_PHYS_HDR • AIM01.T_PD_PHYS_DTL • AIM01.T_PD_UB92_HDR • AIM01.T_PD_UB92_DTL

	• T_RE_BASE • T_PR_PROV **Output Tables:** None **Sort Criteria:** • AIM01.T_PD_XXXX_HDR.DTE_FINAL **Switches:** None **Link Procedure:** Link with lmrpf **Special Logic Notes:** None

2.6.51.1 Change Orders

ID	NAME	DESCRIPTION
1141	RPTS - CLM-9914-Q Create Report	EDS will create a process to generate Abortions Procedure Report.
2045	Add Standard Date Heading Line	It is critical that the headers are clarified and coded to exact specification. There are three potential page widths, 150 columns, 132 columns and 80 columns. Most reports are 132 columns. 80 and 150 column reports should be the exception. There are three primary header lines followed by a fourth optional line. The optional line is the date range line, this line should have just the begin and end date WITHOUT A LABEL PREFIX. If a report is a daily report it will have just one date. The EXACT specifications with column positions are below in the supplemental documentation. If a report name, run date or report date is off by even one character position, the report will be rejected by the OnBase import process and will need to be reworked. I have provided examples of reports of all three widths for both daily reports and those with a date range. Please note there are two tabs in the attachment.

ID	NAME	DESCRIPTION
3882	CLM-0050-M: Retroactive Rate Adj	This is a monthly report and the date heading mm/dd/ccyy - mm/dd/ccyy is missing on the report . This From - Thru date heading is present on the PWB. The member id and member name are on separate lines on OnBase but on same row on PWB.

2.6.52 clmBenefitPlanGroup.sc -- Reads Table T_BENEFIT_PLAN_GROUP

Technical Name:	clmBenefitPlanGroup.sc
Program Title:	Reads Table T_BENEFIT_PLAN_GROUP
Programming Language:	C
Description:	Claims benefit plan group processing. This module is responsible for reading table T_BENEFIT_PLAN_GROUP to determine if given benefit plan is included in a defined grouping of benefit plans. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_BENEFIT_PLAN_GROUP • T_BENEFIT_PLAN_TYPE **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.52.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.53 clmPosGroup -- Reads Table T_PLACE_OF_SERVICE_GROUP

Technical Name:	clmPosGroup
Program Title:	Reads Table T_PLACE_OF_SERVICE_GROUP
Programming Language:	C
Description:	This module is responsible for reading table T_PLACE_OF_SERVICE_GROUP to find if a given place of service code is in a group. Copied from Kansas. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.53.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.54 clm_manual_mass_adj -- Manual Mass Adjustment Request

Technical Name:	clm_manual_mass_adj
Program Title:	Manual Mass Adjustment Request
Programming Language:	C
Description:	This program will create a mass adjustment for a group of claims manually loaded to a temporary table of claim system assigned keys (sak_claim key values). The temporary table is manually created by an SE based on a special request. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_ADJ_MANUAL_MASS (temporary table) • T_EOB **Output Tables:** • T_ADJ_MASS_CLAIM • T_ADJ_MASS_RQST **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None

	Special Logic Notes: None

2.6.54.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.55 clmdaily -- Claim Daily Report Jobs

Technical Name:	clmdaily
Program Title:	Claim Daily Report Jobs
Programming Language:	C
Description:	The program updates the T_SYSTEM_PARMS table for claims daily report jobs. It runs Monday through Friday. The job updates the date parm 1 to the previous day and the date parm 2 to the current date for the program name CLMDAREP The parm values for CLMDAREP are used for all the claims daily reports. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** • T_SYSTEM_PARMS **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.55.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.56 clmp046m -- Potential Schedule 2 Narcotic Drug Abuse

Technical Name:	clmp046m
Program Title:	Potential Schedule 2 Narcotic Drug Abuse
Programming Language:	C
Description:	This program generates CLM-0046-M. The purpose of this report is to summarize prescribing information relating to narcotic drugs. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - Normal termination. • EXIT_FAILURE - Abnormal termination because of open, read, allocation, or input errors **Input Files:** None **Output Files:** clm04601.rpt **Input Tables:** • T_SYSTEM_PARMS • DUAL (Dummy table used to view SYSDATE) • T_PD_PHARM_HDR • T_PD_PHARM_DTL • T_DENY_PHRM_HDR • TMP_PHARM_HDR • T_PD_PHARM_DTLI • T_DRUG • T_PR_PROV • T_RE_BASE • T_PR_LOC_NM_ADR • T_PR_NAM

	Output Tables: • TMP_PHARM_HDR **Sort Criteria:** • ID_MEDICAID **Switches:** None at this time **Link Procedure:** Link with libfinccommon and libmrpf. **Special Logic Notes:** None

2.6.56.1 Change Orders

ID	NAME	DESCRIPTION
163	RPTS - Change "Recipient"	Change every instance of "Recipient", "Current ID" or "Medicaid ID" to "Member ID" on all claim reports.
2083	Re-Code State Specific Changes	Add the method of incorporating state specific coding requirements within an existing core module to report programs.
3882	CLM-0050-M: Retroactive Rate Adj	This is a monthly report and the date heading mm/dd/ccyy - mm/dd/ccyy is missing on the report . This From - Thru date heading is present on the PWB. The member id and member name are on separate lines on OnBase but on same row on PWB.
4047	CLM0046M-Date Heading& Fix Abend	Report CLM-0046-M is not coded to print the standard date heading line. There is also a problem with how the report resolved the state specific headings.

2.6.57 clmp101s.sc -- Auto Adjustment Selection

Technical Name:	clmp101s.sc
Program Title:	Auto Adjustment Selection
Programming Language:	C
Description:	Automatically feeds the adjustment process. Selects adjustments based on a members date of death. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • A_T_RE_PAT_LIAB • A_T_RE_SPEND_LIAB • AIM00.T_PD_PHYS_HDR • AIM00.T_PD_UB92_HDRr • DUAL • T_ADJ_DENT_XREF • T_ADJ_MASS_CLAIM • T_ADJ_PHRM_XREF • T_ADJ_PHYS_XREF • T_ADJ_UB92_XREF • T_EOB

	• T_RE_BASE • T_RE_PAT_LIAB • T_RE_SPEND_LIAB • T_SUSP_ADJ_XREF • T_SYSTEM_PARMS **Output Tables:** • T_ADJ_MASS_CLAIM • T_ADJ_MASS_RQST **Sort Criteria:** None **Switches:** None at this time **Link Procedures:** Link with lmrpf, lnatural, and lgutils **Special Logic Notes:** None

2.6.57.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.58 clmp210d -- Claims For Provider Mass Rate Adjustments

Technical Name:	clmp210d
Program Title:	Claims For Provider Mass Rate Adjustments
Programming Language:	C
Description:	This module contains the routines used to identify the claims associated with Provider Mass Rate Adjustment requests. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_ADJMS_LOC • T_ADJMS_PROVTYP • T_ADJMS_DATES • T_ADJMS_PRSPEC • T_PR_UB_LOC_RATE • T_PR_LOC_RATE • T_PR_TYPE • T_PR_SPEC • T_PR_SVC_LOC • T_HIST_DIRECTORY • T_ADJ_MASS_CLAIM

	• T_ADJ_DENT_XREF • T_ADJ_PHRM_XREF • T_ADJ_PHYS_XREF • T_ADJ_UB92_XREF • T_SUSP_ADJ_XREF • T_ADJ_MASS_RQST **Output Tables:** • T_ADJ_MASS_RQST • T_ADJ_MASS_CLAIM **Sort Criteria:** None **Switches:** None at this time **Link Procedure:** Link with -lxml **Special Logic Notes:** The following are needed to make sure LIKEs work properly. When looking for an exact match, compared fields of "LIKE" must be considered the same length. • EXEC SQL VAR REQUEST_PARMS.CDE_LOC IS STRING(4) • EXEC SQL VAR REQUEST_PARMS.CDE_PROV_SPEC IS STRING(4) Mass rate adjustment requests have a SAK_RQSTt of 4. Requests that haven't had claims identified have cde_status1 = 'S'.

2.6.58.1 Change Orders

ID	NAME	DESCRIPTION
1359	CORE 10281Mass Rate Verification	Modify the Mass Rate Adjustment claims identification process to assign a new adjustment status value to be used for the initial status of provider rate adjustments on the mass adjustment request table. The new status will be used to allow a user to verify and remove any rate

ID	NAME	DESCRIPTION
		adjustment requests.

2.6.59 clmp422 -- Combined Claim Error

Technical Name:	clmp422
Program Title:	Combined Claim Error
Programming Language:	C
Description:	This program generates the Combined Claim Error(CLM-0022-D) report by extracting t_claim_error related tables. This report is used by the Pending Claims Unit to correct suspended claims. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - Normal termination. • EXIT_FAILURE - Abnormal termination because of open, read, allocation, or input errors **Input Files:** None **Output Files:** cld42201.dat **Input Tables:** • T_SYSTEM_PARMS • T_ADJ_DENT_XREF • T_ADJ_PHRM_XREF • T_ADJ_PHYS_XREF • T_ADJ_UB92_XREF • T_CLAIM_ERROR • T_CLAIM_LOCAT • T_CLM_DIAG_XREF • T_CLM_MODIFIER • T_CLM_PGM_XREF • T_DRUG

	• T_ERR_DISP_LINE • T_ERROR_DISP • T_FIN_FUND_CODE • T_HIST_DIRECTORY • T_PR_PROV • T_PROC • T_RE_BASE • T_SUSP_DENTAL_DTL • T_SUSP_DENTAL_HDR • T_SUSP_PHRM_DTL • T_SUSP_PHRM_HDR • T_SUSP_PHYS_DTL • T_SUSP_PHYS_HDR • T_SUSP_RLHX • T_SUSP_UB92_DTL • T_SUSP_UB92_HDR • T_UB92_HDR_DIAG_X • T_UB92_HDR_ICD9CM • T_UB92_HDR_OCC • T_UB92_DTL_EXT_KEY • T_UB92_HDR_PAYER **Output Tables:** None **Sort Criteria:** • SAK_CLAIM **Switches:** None at this time. **Link Procedure:** Link with libmrpf, libclmp422, libtools, libthread, libnsl, libsocket, libdl, libgutils, and libxml.

	Special Logic Notes: None

2.6.59.1 Change Orders

ID	NAME	DESCRIPTION
1033	RPTS - CLM0022D-CHG Report Name	CLM-0022-D provides detail information on the claims with errors. It provides statement cover period information , statement of services information , ICN, recipient id, LOC provide plus LOC, prov spec, prov type, first and last date of service, net claim charge, payment amount, error code, claim disposition and error forced indicator. The name of this report should be changed to Combined Claim Error report.
163	RPTS - Change "Recipient"	Change every instance of "Recipient", "Current ID" or "Medicaid ID" to "Member ID" on all claim reports.
2044	CLM0022D -Invalid sak_case field	This report program is referencing sak_case on T_RE_BASE. This field was removed from this table in Core CO 7888. Core has added a defect to correct the report (CLM-0022-D) which is referencing the sak_case in the Core code. Because we are unsure of the implementation of the Core defect this CO was created to make sure that this change gets applied (or not missed) in the Kentucky code.
2045	Add Standard Date Heading Line	It is critical that the headers are clarified and coded to exact specification. There are three potential page widths, 150 columns, 132 columns and 80 columns. Most reports are 132 columns. 80 and 150 column reports should be the exception. There are three primary header lines followed by a fourth optional line. The optional line is the date range line, this line should have just the begin and end date WITHOUT A LABEL PREFIX. If a report is a daily report it will have just one date. The EXACT specifications with column positions are below in the supplemental documentation. If a report name, run date or

ID	NAME	DESCRIPTION
		report date is off by even one character position, the report will be rejected by the OnBase import process and will need to be reworked. I have provided examples of reports of all three widths for both daily reports and those with a date range. Please note there are two tabs in the attachment.
2083	Re-Code State Specific Changes	Add the method of incorporating state specific coding requirements within an existing core module to report programs.
3882	CLM-0050-M: Retroactive Rate Adj	This is a monthly report and the date heading mm/dd/ccyy - mm/dd/ccyy is missing on the report . This From - Thru date heading is present on the PWB. The member id and member name are on separate lines on OnBase but on same row on PWB.
565	RPTS - CHG Exceptions/Audit-Edit	Terminology need to be consistent through the reports when referring to "exceptions/errors/edit-audit number" The field should be changed from "Exception, Edit-Audit" to "Error code" and "Error description" and from "Edit" to "Error".

2.6.60 clmp956d -- Void Adjustments

Technical Name:	clmp956d
Program Title:	Void Adjustments
Programming Language:	C
Description:	Void Adjustments for Voided Checks **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • DUAL • T_ADJ_MASS_CLAIM • T_ADJ_VOID_RQST • T_CASH_RECEIPT • T_ADJ_DENT_XREF • T_ADJ_PHRM_XREF • T_ADJ_PHYS_XREF • T_ADJ_UB92_XREF • T_SUSP_ADJ_XREF **Output Tables:** • T_ADJ_MASS_CLAIM • T_ADJ_VOID_RQST

	Sort Criteria: None **Switches:** None at this time. **Link Procedure:** Link with -lmrpf -lnatural -lgutils **Special Logic Notes:** None

2.6.60.1 Change Orders

ID	NAME	DESCRIPTION
1243	CORE 10258 MSS_VD_RTR_MCaRe CLMS	When retroactive Medicare Part A or Part B coverage is added for a member, the TPL subsystem will be identifying claims that have been paid by Medicaid that would have been denied if the Medicare coverage had been on file at the time the claim was processed. Providers will be notified and will be given a certain number of days to provide a refund. When a refund is made, the associated claims will be voided using the existing process (performing a region 57 (check related void) adjustment). If the provider does not send a refund within the specified time period, the TPL subsystem will initiate a 'Mass Void' request. Claims will modify the process used for processing check void transactions to include TPL initiated retroactive Medicare voids as well. This process is driven off the t_adj_void_rqst table. A separate change order will be written and assigned to TPL to identify the affected claims and write them to the t_adj_void_rqst table. The claims process will need to be modified to identify those requests initiated by the TPL process and assign them to region 56 (non check related void) when adding rows to the t_adj_mass_claim table.

2.6.61 clmpd075 -- Attachment Indicator Update

Technical Name:	clmpd075
Program Title:	Attachment Indicator Update
Programming Language:	C
Description:	This program updates the attachment indicator in the attachment xref table to 'Y' once the attachment has arrived for the claim. It also changes the status of the claim to resubmit('R') based on their corresponding claim type. This is a daily update program to check for the status of the attachment associated with the claim. **NOTE:** Kentucky currently does not allow electronic claims with attachments. This program is currently not executed. Job CLMJD075 which runs this program is on "ICE". It will not be run. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** attachmentr.dat **Output Files:** Attachments.dat **Input Tables:** • T_ATTACHMENT_XREF • T_RE_BASE • T_PR_PROV **Output Tables:** • T_HIST_DIRECTORY • T_SUSP_PHYS_HDR • T_SUSP_UB92_HDR

	• T_SUSP_DENTAL_HDR • T_ATTACHMENT_XREF **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** Link with lclmbtch, lclmdrvr, llbms, lclmhash, lnatural and lgutils **Special Logic Notes:** None

2.6.61.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.62 clmpda01 -- Remove Adjusted Claims

Technical Name:	clmpda01
Program Title:	Remove Adjusted Claims
Programming Language:	C
Description:	This module contains the routines used to call the adjustment routine for the claims flagged for deletion. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_ADJ_MASS_CLAIM • T_SUSP_ADJ_XREF • T_ADJ_DENT_XREF • AIM01.T_ADJ_DENT_XREF • T_ADJ_PHRM_XREF • AIM01.T_ADJ_PHRM_XREF • T_ADJ_PHYS_XREF • AIM01.T_ADJ_PHYS_XREF • T_ADJ_UB92_XREF • AIM01.T_ADJ_UB92_XREF **Output Tables:** • T_ADJ_MASS_CLAIM

	Sort Criteria: None **Switches:** None at this time. **Link Procedure:** Link with -lxml -lgutils -ladjxml **Special Logic Notes:** The program calls adjMain.sc module to control initialization and adjustment request for the majority of the adjustment functions.

2.6.62.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.63 clmpdi10 -- Encounters Threshold Errors Extract

Technical Name:	clmpdi10
Program Title:	Encounters Threshold Errors Extract
Programming Language:	C
Description:	This program creates an extract for the threshold errors which are identified on the encounter claims. The extract will then be input to the EDI process that creates the 277 transactions used to notify submitters of encounter claims containing threshold errors. The process has 2 major programs: 1. clmpdi10.sc : Encounter Errors Extract 2. clmpdi15.sc : Encounter Errors 277 Formatter The T_CLAIM_LOCAT table is used to make the decision on whether the claim fits into the time frame of the current. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** • Encounter claims with threshold errors extract **Input Tables:** • T_CLAIM_LOCAT • T_SYSTEM_PARMS • T_PD_PHYS_HDR • T_DENY_PHYS_HDR • T_PD_UB92_HDR

	• T_DENY_UB92_HDR • T_PD_DNTL_HDR • T_DENY_DNTL_HDR • T_PD_PHARM_HDR • T_DENY_PHRM_HDR • T_DENY_DNTL_HDR • T_DENY_DNTL_HDR • T_UB92_HDR_EXT_KEY • T_DENTAL_HDR_KEYS • T_PHRM_HDR_KEYS • T_PHYS_HDR_KEY • T_RE_BASE • T_PIN • T_WEB_RECEIVER • T_TRANSACTION_TYPE • T_CLM_ENTITY • T_PARTY_IDENTIFIER • T_CLM_ENT_NM_ADR **Output Tables:** None **Sort Criteria:** • sort_receiver_id[35] • sort_provider_id[16] • sort_recipient_id[12] • filler1[2] **Switches:** None **Link Procedure:** None

	Special Logic Notes: None

2.6.63.1 Change Orders

ID	NAME	DESCRIPTION
197	Encounters Threshold Errors	Create an extract for the threshold errors which were identified on the encounter claims. Claims should send this information to the EDI subsystem. EDI will then format the 277 transaction which will be sent back to MCOs. We will be receiving encounters from Passport and the Transportation cabinet. This KY CO is duplicated in the Core PWB. See Core CO 11760 - all documentation will be updated on this CO.
8352	277 File generation for passport	The 277 interface has to include the duplicate ICN number that is triggering the DUP. The codes for errors are going to be: F1 PAID F2 THRESHOLD F5 DENIED A cleanup is also needed to get rid of EOB's that were arriving as extra info. Also as part of this defect we are fixing a condition where a EOB 5001 is being triggered incorrectly (DUPLICATE CLAIM)

2.6.64 clmpdi15 -- Encounters Errors 277 Formatter

Technical Name:	clmpdi15
Program Title:	Encounters Errors 277 Formatter
Programming Language:	C
Description:	This program reads an input file of sorted claims that need to be reported on the unsolicited 277 file. It reads all the necessary data for each claim and produces an output claims file. This file will be read by the Sybase mapping tool to create the actual files that will be sent to the providers. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** • Sorted Sak List of claims to be reported. **Output Files:** • 277 transaction formatted file **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None **Link Procedure:** None **Special Logic Notes:** None

2.6.64.1 Change Orders

ID	NAME	DESCRIPTION
197	Encounters Threshold Errors	Create an extract for the threshold errors which were identified on the encounter claims. Claims should send this information to the EDI subsystem. EDI will then format the 277 transaction which will be sent back to MCOs. We will be receiving encounters from Passport and the Transportation cabinet. This KY CO is duplicated in the Core PWB. See Core CO 11760 - all documentation will be updated on this CO.
8352	277 File generation for passport	The 277 interface has to include the duplicate ICN number that is triggering the DUP. The codes for errors are going to be: F1 PAID F2 THRESHOLD F5 DENIED A cleanup is also needed to get rid of EOB's that were arriving as extra info. Also as part of this defect we are fixing a condition where a EOB 5001 is being triggered incorrectly (DUPLICATE CLAIM)

2.6.65 clmpdi20 -- McKesson ClaimCheck Dental Extract

Technical Name:	clmpdi20
Program Title:	McKesson ClaimCheck Dental Extract
Programming Language:	C
Description:	This program will gather the information needed to generate the McKesson ClaimCheck Dental Extract. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_SYSTEM_PARMS • T_CLAIM_LOCAT • T_PD_DNTL_HDR • T_RE_BASE • T_PR_IDENTIFIER • T_PR_ID_TYPE • T_MPHX_DENTAL_DTL • T_PROC • T_PR_PROV **Output Tables:** None

	Sort Criteria: • NUM_ICN **Switches:** None **Link Procedure:** None **Special Logic Notes:** None

2.6.65.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.66 clmpdr01 -- Create claims XML for input to claims engine

Technical Name:	clmpdr01
Program Title:	Create claims XML for input to claims engine
Programming Language:	C
Description:	This program extracts claims into XML format based on the sql statement in T_ADJ_RULES. **Input Parameters:** RECYCLESQL - value of CDE_PARENT used to retrieve the SQL statement template from the table T_ADJ_RULES **Exit Values:** • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_ADJ_RULES • T_SYSTEM_KEYS **Output Tables:** • T_ADJ_RULES **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** The program may be linked with dummyClaimCheck to create a test module. **Special Logic Notes:** None

2.6.66.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.67 clmpdr02 -- Batch Program To Recycle Suspended Claims

Technical Name:	clmpdr02
Program Title:	Batch Program To Recycle Suspended Claims
Programming Language:	C
Description:	This module is responsible for modifying the claim status from suspended (S) to resubmit (R) in the applicable suspended claims table. Claims resubmit request is submitted using panel 'Edit Recycle Maintenance' which provided user interface to the 'T_EDIT_RECYCLE' database table. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_EDIT_RECYCLE • T_CLAIM_LOCAT • T_CLAIM_ERROR • T_HIST_DIRECTORY **Output Tables:** • T_EDIT_RECYCLE • T_CLAIM_ERROR • T_SUSP_DENTAL_HDR • T_SUSP_PHRM_HDR

	• T_SUSP_PHYS_HDR • T_SUSP_UB92_HDR • T_HIST_DIRECTORY **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** -lxml, -lgutils, -lclmcomm , -lclmphys, -lclmub92, -lclmdntl, -lclmphrm **Special Logic Notes:** None

2.6.67.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.68 clmper01 -- Daily Error Summary By Claim Type

Technical Name:	clmper01
Program Title:	Daily Error Summary By Claim Type
Programming Language:	C
Description:	This program generates CLM-0024-D, CLM-0055-D and CLM-0056-D. These reports show how many times each error occurred by claim type for the last 5 days for all claims, original claims and adjustments. If the initial extract file does not exist, or has no header record, the program will get the initial date from T_SYSTEM_PARMS. If the header exists, the date from the current file will be used. **Input Parameters:** Initial date from T_SYSTEM_PARMS if the initial extract file does not exist, or if has no header record. **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** • clm01029.srt **Output Files:** • clm0055D.rpt • clm0056D.rpt • clm0024D.rpt **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time **Link Procedures:** Links with lmrpf

	Special Logic Notes: If the header exists, the date from the current file is used.

2.6.68.1 Change Orders

ID	NAME	DESCRIPTION
1065	RPTS - CLM0024D-CHG RESC Heading	The column header RESC should be changed to say RECS.
1070	RPTS - Add Standard Footer	Add the standard footer to report. This KY CO is duplicated in the Core PWB. See Core CO 12077 - all documentation will be updated on this CO.
2045	Add Standard Date Heading Line	It is critical that the headers are clarified and coded to exact specification. There are three potential page widths, 150 columns, 132 columns and 80 columns. Most reports are 132 columns. 80 and 150 column reports should be the exception. There are three primary header lines followed by a fourth optional line. The optional line is the date range line, this line should have just the begin and end date WITHOUT A LABEL PREFIX. If a report is a daily report it will have just one date. The EXACT specifications with column positions is below in the supplemental documentation. If a report name, run date or report date is off by even one character position, the report will be rejected by the OnBase import process and will need to be reworked. I have provided examples of reports of all three widths for both daily reports and those with a date range. Please note there are two tabs in the attachment.
2083	Re-Code State Specific Changes	Add the method of incorporating state specific coding requirements within an existing core module to report programs.
3880	CLM-0056-D: Daily error summary	Location of the report on the PWB reads "CLMPER01 "and Onbase the location reads "CLM0056D". The title of the report " Daily Error Summary By Claim Type Adjustments" is all caps on the PWB .
3940	CLM-0055D-D: Daily Error Summary	1. Location of the report on Onbase (CLMPER01) does not match with PWB. 2. Heading of the report and "Totals by claims type" is not caps 3. No data is displayed on "Cycle 1 OCC field" 4. On page 7 of the report "Totals by Claim Type" the data is truncated after the last col.

ID	NAME	DESCRIPTION
4019	CLM-0024D: Daily Error Summary	1. The location of the report on PWB does not match with Onbase .The correct location is CLMPER01. 2. The heading "Claim type "and the desc are all on one report line. This does not match with Onbase. 3. There is no line of asterisks after the headers on PWB . There is a line of asterisks below the headers. 4. The title of the report , totals by claim type are all caps on PWB and does not match with Onbase. 5. On PWB it reads as Grand total and on Onbase it reads as " Grand Totals" 6. The data for OCC CYCLE 1 is missing .
565	RPTS - CHG Exceptions/Audit-Edit	Terminology need to be consistent through the reports when referring to "exceptions/errors/edit-audit number" The field should be changed from "Exception, Edit-Audit" to "Error code" and "Error description" and from "Edit" to "Error".
8283	Report Printing - 24D, 55D, 56D	When a hard copy of the report is printed in Kentucky characters are being lost off the right hand side of the page.

2.6.69 clmpma00 -- Update Adjustment Request Status

Technical Name:	clmpma00
Program Title:	Update Adjustment Request Status
Programming Language:	C
Description:	Select all active request from T_ADJ_MASS_RQST that have a cde_status1 of I (claims identified) or P (claims processing). Then updates T_ADJ_MASS_RQST cde_status1 to an F (finalized) or P (processed) depending on the individual adjustment status that are a part of the request. Also updates the fields for NUM_COUNT, NUM_COUNT_ADJ, NUM_COUNT_DEL for each request on T_ADJ_MASS_RQST which is not "finalized". **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_ADJ_MASS_RQST • T_ADJ_MASS_CLAIM • T_ADJ_DENT_XREF • T_ADJ_PHRM_XREF • T_ADJ_PHYS_XREF • T_ADJ_UB92_XREF **Output Tables:** • T_ADJ_MASS_RQST **Sort Criteria:**

	None **Switches:** None at this time. **Link Procedure:** Link with lxml, lgutils, lclmcomm, lclmphys, lclmub92, lclmdntl, lclmphrm, lclmdrvr, lclmbtch, ladjcomm and leutil **Special Logic Notes:** Compiling with -DDEBUG results in debug prints statements on stdout.

2.6.69.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.70 clmpma03 -- Void Adjustments

Technical Name:	clmpma03
Program Title:	Void Adjustments
Programming Language:	C
Description:	This module contains the routines used to call the adjustment routine for the claims associated with Void Adjustment requests. **Input Parameters:** None. **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_ADJ_MASS_CLAIM • T_ADJ_MASS_RQST **Output Tables:** • T_ADJ_MASS_CLAIM • T_ADJ_MASS_RQST **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** -lxml, -lgutils, -ladjxml **Special Logic Notes:**

	None

2.6.70.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.71 clmpma_mass_adj -- Call the adjustment routine for claims mass adjustment requests

Technical Name:	clmpma_mass_adj
Program Title:	Call the adjustment routine for claims mass adjustment requests
Programming Language:	C
Description:	Call the adjustment routine for claims Mass Adjustment requests **Input Parameters:** For claims associated with Mass Adjustment requests, sakRqst = 1 For claims associated with Provider Rate Mass Adjustment requests, • sakRqst = 4, OR • sakRqst = 12 **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_ADJ_MASS_RQST • T_ADJ_MASS_CLAIM **Output Tables:** • T_ADJ_MASS_RQST **Sort Criteria:** None **Switches:** None at this time.

	Link Procedure: -lxml, -lgutils, -ladjxml, -leutil **Special Logic Notes:** The program calls adjMain.sc module to control initialization and adjustment request for the majority of the adjustment functions. 0 is passed for claim sequence # because it isn't used for this sakRqst. All claims for the request will be adjusted. adjMain(0, rqst, sakRqst, 0, test_flag, msg).

2.6.71.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.72 clmpmacr -- Mass Adjustment Claim Select

Technical Name:	clmpmacr
Program Title:	Mass Adjustment Claim Select
Programming Language:	C
Description:	This module contains the routines used to select claims for mass adjustment requests. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_ADJMS_PRSVC • T_ADJMS_RECIP • T_ADJMS_ERROR • T_ADJMS_DRUG • T_ADJMS_MODIF • T_CLAIM_ERROR • T_PD_PHARM_DTL • T_CLM_MODIFIER • T_ADJMS_DATES • T_ADJMS_PROVTY • T_ADJMS_PRSPEC

	• T_ADJMS_DIAG • T_UB92_HDR_DIAG_x • T_CLM_DIAG_XREF • T_ADJMS_REVENUE • T_PD_UD92_DTL • T_ADJMS_MEDAST • T_CLM_PGM_XREF • T_RE_BASE • T_ADJMS_AID_CAT • T_HIST_DIRECTORY **Output Tables:** • T_ADJ_MASS_RQST • T_ADJ_MASS_CLAIM **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** -lxml, -ladjxml, -lmemhash **Special Logic Notes:** None

2.6.72.1 Change Orders

ID	NAME	DESCRIPTION
1262	CORE 10280 MFY MSSADJ CLM SELCTN	Add an indicator (check box) to the Mass Adjustment Criteria panel to allow selection of claims where the header net billed amount equals the total reimbursement amount for the claim. Modify the mass adjustment claim selection process to apply the check of billed amount equal paid amount when the criteria is selected for a request.

ID	NAME	DESCRIPTION
2013	CORE 10380 NEW T_ADJMS_PAYBILL	Create a new reference table that will be used to hold parameters of the Mass Adjustment Criteria Panel (CO 1229) and then used by the Mass Adjustment Claim Select program (CO 1262).

2.6.73 clmpsplt -- FFS Claims Splitter

Technical Name:	clmpsplt
Program Title:	FFS Claims Splitter
Programming Language:	C
Description:	This module will split a large file of claims into smaller chunks based on its claim type. **Input Parameters:** -m maxClaimPerFile -i inputDir -o outputDir **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** The claimList files in xml format **Output Files:** xml files in smaller chunks based on its claim type **Input Tables:** **Output Tables:** • T_BATCH **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.73.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.74 clmpw064 -- Suspended Claims By Claim Type / Region

Technical Name:	clmpw064
Program Title:	Suspended Claims By Claim Type / Region
Programming Language:	C
Description:	This program generates CLM-0064-W. This is a weekly report that will report the number of suspended claims by region and claim type. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** clm0064.rpt **Input Tables:** • T_REGION • T_SUSP_PHYS_HDR • T_SUSP_UB92_HDR • T_SUSP_PHRM_HDR • T_SUSP_DENTAL_HDR **Output Tables:** None **Sort Criteria:** None **Switches:** None **Link Procedure:** -lmrpf

	Special Logic Notes: None

2.6.74.1 Change Orders

ID	NAME	DESCRIPTION
1070	RPTS - Add Standard Footer	Add the standard footer to report. This KY CO is duplicated in the Core PWB. See Core CO 12077 - all documentation will be updated on this CO.
1075	RPTS - CLM0064W- Add Heading	This report currently displays the region and the region description but does not display a heading for these fields. Add a heading for the region and the region description to the report.
2045	Add Standard Date Heading Line	It is critical that the headers are clarified and coded to exact specification. There are three potential page widths, 150 columns, 132 columns and 80 columns. Most reports are 132 columns. 80 and 150 column reports should be the exception. There are three primary header lines followed by a fourth optional line. The optional line is the date range line, this line should have just the begin and end date WITHOUT A LABEL PREFIX. If a report is a daily report it will have just one date. The EXACT specifications with column positions is below in the supplemental documentation. If a report name, run date or report date is off by even one character position, the report will be rejected by the OnBase import process and will need to be reworked. I have provided examples of reports of all three widths for both daily reports and those with a date range. Please note there are two tabs in the attachment.

2.6.75 clmsAssignment -- Managed Care Claim Related Edits

Technical Name:	clmsAssignment
Program Title:	Managed Care Claim Related Edits
Programming Language:	C
Description:	This module is responsible for managed care and lockin claim related edits. It will process each assignment plan to which a member belongs based on the claim dates of service and calls the rules engine to determine whether a given service is subject to or exempt from the requirements of the assignment plan. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_RE_ELIG • T_RE_ASSIGNMENT • T_COVERED_BENEFIT • T_COV_BNFT_POS • T_COV_BNFT_MOD • T_COV_BNFT_PA_POS • T_PUB_HLTH_PGM • T_PGM_CT_XREF • T_COV_CT_XREF • T_RE_PMP_ASSIGN

	• T_PMP_SVC_LOC **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.75.1 Change Orders

ID	NAME	DESCRIPTION
3668	CORE 13861 - EBP Dx Changes	CORE Change Order 8738 required several different diagnosis for coverage, billing and reimbursement rules. Not all were implemented and definitions of the roles were not clear. Definitions of the diagnosis types/roles: Header Primary: First position of the diagnosis at the header. Header Secondary: Second position of the diagnosis at the header. Admit Diagnosis: Diagnosis indicated as the Admit diagnosis. Emergency Diagnosis: Diagnosis indicated as the Emergency diagnosis. Header Other: Any diagnosis at the header that is not the Header Primary, Header Secondary, Admit or Emergency diagnosis. Header Any: Any of the Header, Secondary, Admit, Emergency, Other diagnosis. Detail Primary: Diagnosis pointed to from the first detail diagnosis pointer. Detail Secondary: Diagnosis pointed to from the second detail diagnosis pointer. Detail Other: Diagnosis pointed to from other than the first and second detail diagnosis pointers. Detail Any : Any of the diagnosis pointed to by the detail.

2.6.76 clmsAudit -- General Claims Auditing

Technical Name:	clmsAudit
Program Title:	General Claims Auditing
Programming Language:	C
Description:	This module is responsible for auditing information about the claim detail. General claims auditing functions. Includes audit criteria, umbrella audit criteria, limit audit criteria, procedure audit criteria, contra audit criteria, PA, ProDUR generic drug and ProDUR therapeutic drug history info, medical policy history info, and an application provided duplicate check function. Prior to calling clmAudit (one of the entry points), the current claim and history claims should be set. Use the set_mh_* and set_mc_* routines to accomplish this. Also includes step therapy for pharmacy. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** • t_mphx_dental_dtl • t_mphx_phrm_dtl • t_mphx_phys_dtl • t_medpol_ub92 • t_contra_proc • t_contra_parm • t_limit_parm • t_umbrella_parm • t_audit_crit

	• t_re_loc • t_audit_ndc • t_audit_proc • t_revenue_code • t_audit_rev • t_audit_st • t_audit_st_mem • t_audit_st_level • t_audit_cl_type_x • t_audit_claim_form **Output Tables:** • t_mphx_dental_dtl • t_mphx_phrm_dtl • t_mphx_phys_dtl • t_medpol_ub92 • t_pa_item_dtl_xref **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** -lclmcomm **Special Logic Notes:** None

2.6.76.1 Change Orders

ID	NAME	DESCRIPTION
1136	ccInt process outpatient claims	Modify the base integration module as needed to process outpatient claims 1. Modify claim check integration module calls to include outpatient claim processing

ID	NAME	DESCRIPTION
		2. Modify integration history pull to retrieve outpatient claim history 3. Modify outpatient data structures to maintain submitted and replacement procedure codes
1325	Configure conflict audits	Configure existing conflict audits in interChange. 15 existing audits have been identified as conflict audits to be implemented in interChange. Of these, only two audits involve a large number of services. The following items will need to be addressed to configure all conflict audits as they are currently defined: Audit that prevents EPSDT providers from billing for services on the same day as other provider types may require change to how provider type is currently used in conflict audits Some audits have provider specific exclusions Audit that checks the number of surfaces billed per tooth may require change to how surface is currently used in conflict audits Procedure code restrictions that can not be performed by current limitation or contra indicated audits
1795	Add num_dtl t_clm_patliab_x app	Modify the core objects that access t_clm_patliab_x to use detail number. Existing sql is based on patient liability applied at header level so the detail number to use when modifying existing objects is zero (0).
2014	Diag s/d processing - core 9983	A new same/different indicator and a new diagnosis type indicator are being added to the audit criteria table. A new function needs to be added in the claims audit program to process the indicator.
2015	Limit audit by quadrnt-Core 9907	A new quadrant matching field is being added to the audit criteria table. A new function needs to be added in the claims audit program to process the code. There can be up to 5 "quadrant" values specified per service line. The code values will be used in processing as follows: If the value is "C" - (complete match) all quadrant values must match between the current and history claims. If the value is "I" - (inclusive match) all quadrant values on the current claim line must be found on the service in history. If the value is "P" - (partial match) at least one of the

ID	NAME	DESCRIPTION
		quadrant values must match between the current and history claim. If the value is "E" - (exclusive) no quadrant values can match between the current and history claim.
2016	Equal auditing units - Core 9902	Modify the clmsAudit program to implement the Kentucky processing that allows codes measured in different unit increments to be combined on an equal basis in limitations. This change will associate a conversion factor to each procedure code included in the audit. The conversion factor is used to allow codes with a different basis of measurement to be compared on an equal basis. For example, if two codes are both counted toward a limit of 5 hours of therapy per month and one is billed in 15 minute units, the other in 1 hour units, the conversion factor of .25 can be applied to the code billed in 15 minute units to count toward the hours limit.
2018	Contact limit audit - Core 9804	Kentucky has current audits that limit services by the number of 'visits' or 'contacts' rather than the units of service or dollar limitations currently supported. A visit is considered to be the entire set of services rendered by a provider to a recipient on a single date of service. To support auditing of visits a new value will be added to the group of allowed values for unit type on the limitation audit parameters table. Processing of the new value will require that all details meeting the criteria defined for the limitation will have additional processing performed against them to count the number of unique dates of service. If the from and to dates of service on a single detail span more than one day the dates spanned in the from/to range will determine the number of 'visits' applied to the limit.
2019	Calendar week limits - Core 9803	In order to support current KY audit processing, a new value of 'calendar week' will be added to the values for limit audit time units. Corresponding processing needs to be added in the claims audit program to process calendar week limits.
2020	Conflict audit dates - Core 9802	Add the ability to process an optional date relationship between current and history claim based on values added to (new) table T_CONFLICT_PARM. If no parameters are entered for a conflict the current overlapping date check will be performed.

ID	NAME	DESCRIPTION
7940	Primary Care/RHC on ADA Form	Effective 2007/04/01, primary care centers (PT 31) and rural health clinics (PT35) are required to submit dental services on ADA dental claim forms. The system needs to be updated to accept, edit, audit, and price claims for these provider type on dental claim forms.
8634	Conflict & Neg Contra Audit prob	Claim 2007179102902 is a Part B Nursing Facility claims that is failing 3 suspect duplicate audits (5614, 5615, 5619) against a non-crossover Nursing Facility claim. A Part B claim on the same dates as a non-crossover Nursing Facility claim is OK so it shouldn't be failing any of these suspect dupe audits. This claim is also failing negative contra audit 5254 when it should not be. Also, all of these audits are failing at the detail and I think they should fail at the header for these claim types (A & L). See additional info below:

2.6.77 clmsAutoPa -- Pharmacy Auto PA Module For DUR+

Technical Name:	clmsAutoPa
Program Title:	Pharmacy Auto PA Module For DUR+
Programming Language:	C
Description:	This module is responsible for performing all auto PA checks and creating PAs when applicable. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_DRUG • T_DRUG_GROUP • T_AUTO_PA • T_GCN_SEQNO_GROUP • T_GENERIC_DRUG • T_THERAPEUTIC • T_PA_PAUTH • T_PA_LINE_ITEM • T_GCN_SEQNO_GROUP • T_AUTO_PA_AGE • T_AUTO_PA_PRIM_DIAG

	• T_AUTO_PA_SECD_DIAG • T_AUTO_PA_STEP_THPY_1 • T_AUTO_PA_STEP_THPY_2 • T_AUTO_PA_COMOR_DIAG • T_AUTO_PA_TXNMY • T_AUTO_PA_GRNDFTR • T_PR_TAXONOMY • T_PR_TAXONOMY_CDE • T_DIAGNOSIS • T_DIAG_GROUP • T_MPHX_DSE_PROF • T_MPHX_PHRM_DTL • T_THERAPEUTIC • T_THERAPEUTIC_GROUP **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.77.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.78 clmsBenefitPlanGroup -- Reads Table T_BENEFIT_PLAN_GROUP

Technical Name:	clmsBenefitPlanGroup
Program Title:	Reads Table T_BENEFIT_PLAN_GROUP
Programming Language:	C
Description:	This module is responsible for reading table T_BENEFIT_PLAN_GROUP to find if a given benefit plan is in a group. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_BENEFIT_PLAN_TYPE • T_BENEFIT_PLAN_GROUP **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.78.1 Change Orders

ID	NAME	DESCRIPTION
1137	ccInt - modify history pull	Modify the history pull filtering process to pull data according to Kentucky requirements: 1. Do not send denied details to ClaimCheck 2. Limit the history passed for evaluation and management vs. surgical procedures to 90 days before/after the procedure being audited 3. Limit procedures beyond the 90 day limit to those procedures identified as lifetime procedure 4. Do not send Encounter Claims to ClaimCheck
7267	Coins shouldn't apply for bypass	The bypass rules for copays should also apply to coinsurance. I was trying to process a claim with V22 diag and the system took the coinsurance. I suspect the bypass rules coded in the copay configuration are not used in the coinsurance logic. This includes PT, claim type, diagnosis and benefit plans for bypassing the coinsurance logic.

2.6.79 clmsDiag -- Claim Diagnosis

Technical Name:	clmsDiag
Program Title:	Claim Diagnosis
Programming Language:	C
Description:	This program validates diagnosis codes against the T_DIAGNOSIS table. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_DIAGNOSIS **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.79.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.80 clmsDiagGroup -- Claims Diagnosis Groups

Technical Name:	clmsDiagGroup
Program Title:	Claims Diagnosis Groups
Programming Language:	C
Description:	This module is responsible for reading table T_DIAG_GROUP to find if the given diagnosis is in a group. Diagnosis type saks are hard coded for each group (I.e. prenatal care is sak 5, pregnancy is sak 6, etc.). **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** • t_diag_code_group **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** -lclmcomm **Special Logic Notes:** None

2.6.80.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.81 clmsDisp -- Claim Disposition

Technical Name:	clmsDisp
Program Title:	Claim Disposition
Programming Language:	C
Description:	This program is responsible for the disposition of a claim. Assigns a disposition to a line item/claim based on the errors found. **Input Parameters:** None. **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables** • T_ERROR_DISP • T_ERR_DISP_LINE • T_REGION_DISP • T_LOCATION • T_EOB_ERR_XREF • T_EOB **Output tables** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** -lclmcomm

	Special Logic Notes: None

2.6.81.1 Change Orders

ID	NAME	DESCRIPTION
1121	CC - integration module	Install the ClaimCheck integration module currently running in the Kentucky installation of interChange. Modify existing claims engine modules to ensure that calls to the integration module are placed appropriately. ClaimCheck processing will be performed after edits, pricing and audits have been performed, but before final pricing (application of tpl, spenddown, copay) occurs.

2.6.82 clmsDtl -- General Claim Detail

Technical Name:	clmsDtl
Program Title:	General Claim Detail
Programming Language:	C
Description:	This module is responsible for holding generic claim detail information (things like amount billed, quantity units billed, claim status, amount paid, etc.) and edits common to all claim types (things like provider's specialty is valid to billed procedure, diagnosis code is valid for procedure, etc.). Routines to set the detail information (set_cd_xxxx where xxxx is a field name like cdeClmStatus), get the detail information (get_cd_xxxx), do a check (is_cd_xxxx where xxxx might be a value like Susp), or do a validation (xref_cd_xxxx where xxxx might be dteSvcDteIcn to validate that date of service is NOT before the ICN date). **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables** • t_proc • t_modifier • t_modifier_rstn • t_proc_limits • t_clia_cert • T_DIAGNOSIS • t_proc_spec_lim • t_process_mod

	• t_proc_modifier • t_re_elig • t_fin_fund_code • t_pr_php_elig • t_pub_hlth_pgm • t_pgm_ct_xref • t_revenue_code • t_spec_mod • t_pr_state_share • t_pr_taxonomy_cde • t_pr_type_spec **Output Tables** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** -lclmcomm **Special Logic Notes:** None

2.6.82.1 Change Orders

ID	NAME	DESCRIPTION
2167	PA - Modify calls to find PA	Background: Functions in the base system that match claim services requiring authorization to a PA (service authorization) currently do not use the prior authorization number submitted on the claim as part of the matching criteria. Certain services billed by Kentucky providers that currently require authorization are billed using non-specific codes and are priced from the PA. It is

ID	NAME	DESCRIPTION
		possible for a given member to require multiple services during the same time period that will be billed under the same procedure code. In these cases, the only way to ensure that the claim is priced appropriately is to match the PA number submitted on the claim to the service authorization or suspend the claim for manual pricing. Proposed solution: Modify is_allpa_available function in clmsPaPauth.sc to find a matching PA number (t_pa_pauth vs. PA number supplied on the claim). Look up matches in t_clm_ref table and process using the criteria below: 1. Check for the presence of a clmRef node on the claim and if present 2. Process through each clmRef node checking for the QLF_REFERENCE_ID for a value of 'G1' (prior authorization) and if present 3. Check the CDE_REF_ID for a value that matches one of the PA numbers found in processing the PA cursor and ifa match is found 4. Check the NUM_DTL to verify that it is zero or matches the detail number passed in the call to is_allpa_available.
2298	KY Non DRG Inpatient Pricing	Claims submitted by psychiatric hospitals, as well as many inpatient services are not priced using the DRG pricing method. There are several different methods used based on the providers out of state status, disproportionate share status, and the age of the member at the time the services are rendered. Changes need to be made to inpatient pricing to perform these additional pricing methods, as well as to be able to identify when to use the various pricing methods.
2777	Add KY Edit 011 - iC 3345	Add KY specific edit in interChange to match KY policy for existing edit 011.
3149	Add Ky Edit 310 - iC 3360	Add KY specific edit in interChange to match KY policy for existing edit 310 where a new admissions not payable because of provider non-compliance.
3843	Copay Processing - CE	Deduct copay based on rule parameters when processing a claim.
4035	Add indClaimCheck-UB92 Claims	Add the indClaimCheck field for UB claims.

2.6.83 clmsHdr -- General Claim Header

Technical Name:	clmsHdr
Program Title:	General Claim Header
Programming Language:	C
Description:	Claims header processing. This module is responsible for holding claim header information (things like claim sak, member first and last name, etc.) and edits common to all claim types. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables** • t_prov_clm_xref • t_pr_state_share • t_pr_state_agency • t_fmap_pct • t_pr_taxonomy_cde • t_pr_type_spec • t_re_base • t_pub_hlth_pgm • t_attachment_xref • t_place_of_service • t_re_pat_liab • t_pr_taxonomy_cde • t_taxonomy_cl_type_x

	Output Tables • t_attachment_xref **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** -lclmcomm **Special Logic Notes:** None

2.6.83.1 Change Orders

ID	NAME	DESCRIPTION
2298	KY Non DRG Inpatient Pricing	Claims submitted by psychiatric hospitals, as well as many inpatient services are not priced using the DRG pricing method. There are several different methods used based on the providers out of state status, disproportionate share status, and the age of the member at the time the services are rendered. Changes need to be made to inpatient pricing to perform these additional pricing methods, as well as to be able to identify when to use the various pricing methods.
6936	OPASC Pricing Issues	Claims for critical access providers that contain a operating room revenue code (360) are inappropriately being cut back and zero paid. Additionally, dental codes billed with revenue code 360 should be excluded from all final pricing logic and be paid the OP percentage.
7337	Price KCHIP State Share	School based providers are only reimbursed the federal match amount for claims submitted to Medicaid. The federal match percentage is different for KCHIP member and regular Medicaid members. Changes were implemented to assign the fund code outside of the claims engine in KY which resulted in the state share pricing methodology no longer working. The claims engine needs to be able to determine the correct (KCHIP or non-KCHIP) fund code to be used to price these claims correctly.

2.6.84 clmsIcd9Group -- Claims ICD9 Procedure Groups

Technical Name:	clmsIcd9Group
Program Title:	Claims ICD9 Procedure Groups
Programming Language:	C
Description:	Claims ICD9 procedure processing. This module is responsible for reading table T_ICD9_GROUP to find if given icd9 procedure is in a group. Procedure code group saks are hardcoded (I.e. abortion procedures use sak 4, surgical procedures use sak 1001, etc.). **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** • t_proc_icd9_code_group **Output Tables:** • none **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** -lclmcomm **Special Logic Notes:** None

2.6.84.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.85 clmsIcnAssign -- ICN Assignment

Technical Name:	clmsIcnAssign
Program Title:	ICN Assignment
Programming Language:	C
Description:	This program assigns the ICN to a claim. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.85.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.86 clmsLoadHist -- Load Member Claim History

Technical Name:	clmsLoadHist
Program Title:	Load Member Claim History
Programming Language:	C
Description:	Load member history. This module is responsible for loading medpol information to memory. There are separate routines for each claim type (I.e. load_ub92History, load_dntlHistory, etc.). **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** • T_MPHX_PHYS_DTL • T_MEDPOL_UB92 • T_PROC • T_MPHX_DENTAL_DTL • T_MPHX_PHRM_DTL • T_DRUG • T_GENERIC_DRUG **Output Tables:** • None **Sort Criteria:** None **Switches:** None at this time.

	Link Procedure: -lclmcomm **Special Logic Notes:** None

2.6.86.1 Change Orders

ID	NAME	DESCRIPTION
1136	ccInt process outpatient claims	Modify the base integration module as needed to process outpatient claims 1. Modify claim check integration module calls to include outpatient claim processing 2. Modify integration history pull to retrieve outpatient claim history 3. Modify outpatient data structures to maintain submitted and replacement procedure codes
8277	Claims Engine RTI	Task to be used for the promotion of claims engine changes that do not affect any of the outputs of claims processing and have no effect on the overall error status, payment or disposition of claims but are needed to the throughput of claims, prevent claims from processing to the 'bad' files or to otherwise optimize the performance of claims processing.

2.6.87 clmsMisc -- Miscellaneous General Claims

Technical Name:	clmsMisc
Program Title:	Miscellaneous General Claims
Programming Language:	C
Description:	This program is responsible for miscellaneous claims processing. This module has miscellaneous functions, such as finding a provider's specialty, creating a claim response, and ITF functions like walkExpectedTree (claims only). Adjustments uses the checkCashReceipt function to create financial disposition records, which writes to T_CASH_RCPT_DISP, T_CASH_RCPT_XREF. The xref_set_adjXref function is used to create adjustment cross references records on T_SUSP_ADJ_XREF, T_ADJ_DENT_XREF, T_ADJ_PHRM_XREF, T_ADJ_PHYS_XREF, T_ADJ_UB92_XREF. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input tables** • t_clm_pgm_xref • t_pr_spec • t_error_disp • t_eob • test_case_released • t_cash_receipt • t_cash_rcpt_disp • t_susp_adj_xref • t_adj_mass_claim

	Output tables • t_adj_mass_claim • test_case_released **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** -lclmcomm **Special Logic Notes:** None

2.6.87.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.88 clmsModGroup -- Modifier Group

Technical Name:	clmsModGroup
Program Title:	Modifier Group
Programming Language:	C
Description:	This module is responsible for reading table T_MOD_GROUP to find if a given modifier is in a group. Modifiers are grouped using hardcoded saks (I.e. transportation modifiers use sak 1000, anesthesia modifiers use sak 1003). **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_MOD_GROUP **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** -lmemhash **Special Logic Notes:** None

2.6.88.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.89 clmsMultiPayer -- Claim MultiPayer COB Pricing

Technical Name:	clmsMultiPayer
Program Title:	Claim MultiPayer COB Pricing
Programming Language:	C
Description:	This Module contains the functions to do the final Pricing, Spenddown,TPL and COB among Payers and Benefit Plans. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.89.1 Change Orders

ID	NAME	DESCRIPTION

ID	NAME	DESCRIPTION
1416	KY Prior Auth Pricing	Pricing using prior authorizations needs to be modified based on the manner in which the PAs are entered into the system and stored.
1417	DRG Pricing	CO to handle necessary updates to implement KY Pricing policy to DRG pricing methodology.
8359	pat liab - billing and case mgr	Phase I Pat liab is being deducted from ABI waiver when the billing provider is not the case manager. Phase II Pat liab deductions overall fixes.
8537	Error code 5030 - Fin error rpt	Claims are being flushed from the financial cycle with error 5030 due to the claim header paid amount not matching the amount paid on the claim program xref. There are several situations causing this to occur: 1. Physician crossover claims are not deducting the TPL amount at the detail level. Since these claims are detail paid, the TPL amount should be deducted at the detail. 2. Some UB claims are paid at the detail level. In these cases, the TPL and patient liability amounts are not being deducted correctly on the claim program xref table. 3. Some physician (CT M) and dental claims do not have a header paid amount but do have paid amounts on the health plan panel.
8721	LandW reso fix	find and fix where the ind_rpt on t_claim_error is being set to 'Y' or 'N'. The only valid values are ' ', 'L' or 'W'. Where clmsDisp.sc is currently setting a 'Y', it should be set to 'L'. Where clmsDisp.sc is currently setting a 'N', it should be set to a space. According to Bob Carter: Yes, it is the cde_print_type that populates ind_rpt on t_claim_error. Also there are a couple of instances in clmsDisp.sc that hard code a value of 'Y' or 'N'. Yes, if we want to use 'L' and 'W' we would need to have those values on in cde_print_type on t_err_disp_line or we could translate the values in clmsDisp.

2.6.90 clmsPRTypeSpecGroup.sc -- Claims Provider Type and Specialty Group Processing

Technical Name:	clmsPRTypeSpecGroup.sc
Program Title:	Claims Provider Type and Specialty Group Processing
Programming Language:	C
Description:	This module is responsible for reading table T_TYPE_SPEC_GROUP to find if given provider type - provider specialty combination is in a group. The selection process includes the analysis of the Include/Exclude indicators in each record, and the wild card selections PT/00 -> All provider specialties for the provider type. 00/PS -> All provider types for provider specialty. If a match of PT/PS and wild cads with an Include flag is found, then the process will search for exclude flags. If another provider type and specialty is found, then the search will assume the existing record is excluded, and all other combinations are to be included. Parameters: None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_TYPE_SPEC_CODE • T_TYPE_SPEC_GROUP **Output Tables:** • none **Sort Criteria:** None **Switches:**

	None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.90.1 Change Orders

ID	NAME	DESCRIPTION
3017	KY PA Edits DME Bypass	When billed by a DME provider, procedure code E1399 should not fail prior authorization edits under the following conditions: - Detail dates of service (DOS) from 7/1/2002 to 1/2/2003 (inclusive) - The billed amount of the detail is less than or equal to $300.00 - There is no PA number on the claim
6883	Claims not pricing correctly	Claims not pricing correctly for DME-unable to test copay/coins A PT90 claim was entered without a PA number, there was a PA in the system for that member/prov/DOS. The system processes that claim against that PA. However for DME (PT 90) and EPSDT services (EPSDT Related Services - PT 45) the claim should not pay if the PA is not on the claim. Please include Beth Jennings also in any discussion. Data file is ../MOclaims/Sandeep_test/Coins/PT90.xml
7940	Primary Care/RHC on ADA Form	Effective 2007/04/01, primary care centers (PT 31) and rural health clinics (PT35) are required to submit dental services on ADA dental claim forms. The system needs to be updated to accept, edit, audit, and price claims for these provider type on dental claim forms.
8173	Coins - only accumulate - outpt	Only accumulate coinsurance for the emergency room service (proc code 99281).

2.6.91 clmsPaPauth -- General Claim PA

Technical Name:	clmsPaPauth
Program Title:	General Claim PA
Programming Language:	C
Description:	Claims PA processing. General claims service/prior authorization functions. Includes routines for saving claim detail PA values, getting claim detail PA values (I.e. get_pa_revenue, get_pa_drug, etc.), and various edits (I.e. xref_pa_fundCode to check if pa fund code matches claims). **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** • T_PA_ASSIGN_CODE • T_FIN_FUND_CODE • T_PA_PAUTH • T_PA_ESC • T_PA_ITEM_DTL_XREF • T_PA_LINE_ITEM **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:**

	-lmemhash **Special Logic Notes:** None

2.6.91.1 Change Orders

ID	NAME	DESCRIPTION
1416	KY Prior Auth Pricing	Pricing using prior authorizations needs to be modified based on the manner in which the PAs are entered into the system and stored.
2167	PA - Modify calls to find PA	Background: Functions in the base system that match claim services requiring authorization to a PA (service authorization) currently do not use the prior authorization number submitted on the claim as part of the matching criteria. Certain services billed by Kentucky providers that currently require authorization are billed using non-specific codes and are priced from the PA. It is possible for a given member to require multiple services during the same time period that will be billed under the same procedure code. In these cases, the only way to ensure that the claim is priced appropriately is to match the PA number submitted on the claim to the service authorization or suspend the claim for manual pricing. Proposed solution: Modify is_allpa_available function in clmsPaPauth.sc to find a matching PA number (t_pa_pauth vs. PA number supplied on the claim). Look up matches in t_clm_ref table and process using the criteria below: 1. Check for the presence of a clmRef node on the claim and if present 2. Process through each clmRef node checking for the QLF_REFERENCE_ID for a value of 'G1' (prior authorization) and if present 3. Check the CDE_REF_ID for a value that matches one of the PA numbers found in processing the PA cursor and ifa match is found 4. Check the NUM_DTL to verify that it is zero or matches the detail number passed in the call

ID	NAME	DESCRIPTION
		to is_allpa_available.
2181	PA - use admit date for C/T I	Modify code that checks the service dates against the PA effective and end dates to use admit date for inpatient claims when testing the effective date. For all inpatient claims, check for an accommodation code. If there is an accommodation code, then use dteAdmission in place of dteFirst, then check for DRG pricing. Where the claim is priced using DRG pricing, use dteAdmission in place of dteLast when checking whether the PA matches the claim.
3017	KY PA Edits DME Bypass	When billed by a DME provider, procedure code E1399 should not fail prior authorization edits under the following conditions: - Detail dates of service (DOS) from 7/1/2002 to 1/2/2003 (inclusive) - The billed amount of the detail is less than or equal to $300.00 - There is no PA number on the claim
3102	KYAMEND Support PA mass updates	There is a requirement for mass update of prior authorizations. This change order supports mass updates of PA rates and service codes. A table (t_pa_mass_upd) will be created to hold multiple service codes and rates with effective begin and end dates associated with a PA line item. Claims processing will include this table in determining if a valid PA exists for a claim to process against.
6883	Claims not pricing correctly	Claims not pricing correctly for DME-unable to test copay/coins A PT90 claim was entered without a PA number, there was a PA in the system for that member/prov/DOS. The system processes that claim against that PA. However for DME (PT 90) and EPSDT services (EPSDT Related Services - PT 45) the claim should not pay if the PA is not on the claim. Please include Beth Jennings also in any discussion. Data file is ../MOclaims/Sandeep_test/Coins/PT90.xml

2.6.92 clmsPrProvBilling -- Claims Billing Provider

Technical Name:	clmsPrProvBilling
Program Title:	Claims Billing Provider
Programming Language:	C
Description:	Claims billing provider processing. This module is responsible for holding billing provider information (things such as service location code, county, etc.), reading related tables (read_pr_svcLocBilled for reading t_pr_svc_loc, for example), and for certain edits (xref_pb_provRend for determining that the rendering provider is a member of the billing provider's group if billing provider is a group provider). **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** • t_pr_identifier • t_pr_prov • t_pr_type • t_pr_spec • t_pr_grp_mbr • t_mco_pr_network • t_pr_disp_fee • t_pr_mcare_bill • t_pr_prev_prov • t_pr_svc_loc

	• t_re_elig • t_re_pmp_assign • t_pmp_svc_loc • t_error_disp • t_type_spec_excl **Output Tables:** None **Sort Criteria:** None **Switches:** None **Link Procedure:** libclmcomm **Special Logic Notes:** None

2.6.92.1 Change Orders

ID	NAME	DESCRIPTION
2999	KY CE Add OOS Ind Variable	Some institutional claims are priced based on the type of out-of-state indicator assigned to the billing provider on the claim. The billing provider's bed size and disproportionate share status are also used to determine the pricing method. These attributes need to be available to assign the appropriate pricing methods in the reimbursement rules.

2.6.93 clmsPricing -- Claims Pricing

Technical Name:	clmsPricing
Program Title:	Claims Pricing
Programming Language:	C
Description:	This module is responsible for general claims pricing routines. It includes many routines for accessing pricing data, as well as numerous routines used to price a claim. **Input Parameters:** Detail and Header information related to member programs, performed procedures, or revenue **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** • T_REF_UCC • T_PREVAILING • T_MAXFEE • T_REVENUE_FLAT_FEE • T_PR_UB_LOC_RATE • T_PROC_ASC • T_COPAY • T_RBRVS • T_GPCI • T_PROC_CODE_CONV_FACTOR • T_CONV_FACTOR • T_CLM_AMT_VARIANCE

	• T_XOVER_RATE • T_EOB **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** libclmcomm **Special Logic Notes:** None

2.6.93.1 Change Orders

ID	NAME	DESCRIPTION
1416	KY Prior Auth Pricing	Pricing using prior authorizations needs to be modified based on the manner in which the PAs are entered into the system and stored.
1417	DRG Pricing	CO to handle necessary updates to implement KY Pricing policy to DRG pricing methodology.
1476	Core 8816 Benefit Adj Factor 7	See Core CO 8816 all documentation will be updated on this CO Provide the infrastructure to allow modifications to all pricing methodologies that have a variance or incentive from the regular pricing methodology.
1645	Core 8990 Pricing Ind Expansion	See Core CO 8990 - all documentation will be updated on this CO To be able to provide an easier identification of pricing methodology that was utilized to price a claim or detail it is necessary to make an expansion of the pricing indicator field. The current process utilizes a 1 digit alphanumeric field. The modifications requested impact panels, tables and programs that currently utilize the pricing indicator. The new field size will be expanded from 1 digit to 6 alphanumeric characters.

ID	NAME	DESCRIPTION
1646	Core 8991 Pricing Ind Expansion	See Core CO 8991 - all documentation will be updated on this CO To be able to provide an easier identification of pricing methodology that was utilized to price a claim or detail it is necessary to make an expansion of the pricing indicator field. The current process utilizes a 1 digit alphanumeric field. The modifications requested impact panels, tables and programs that currently utilize the pricing indicator. The new field size will be expanded from 1 digit to 6 alphanumeric characters.
1648	Core 8993 Pricing Ind Expansion	See Core CO 8993 - all documentation will be updated on this CO To be able to provide an easier identification of pricing methodology that was utilized to price a claim or detail it is necessary to make an expansion of the pricing indicator field. The current process utilizes a 1 digit alphanumeric field. The modifications requested impact panels, tables and programs that currently utilize the pricing indicator. The new field size will be expanded from 1 digit to 6 alphanumeric characters.
1695	Xover Pricing	Change Cross over Pricing to include necessary fields to support KY pricing policy fro both institutional and professional cross over claims. Medicare information for Physician cross over's will be at the header and detail level. Related COs: 1694, 1347 KY CO 1694 KY CO 1347
1922	Core 9704 T_REF_UCC 3	See Core CO 9704 all documentation will be updated on this CO Add rate type to table t_ref_ucc to support the reimbursement rules assignment.
2136	Multiple Surgery - CE	Pricing method needed for multiple surgeries (same date of service).
2194	Allow Update of Coinsurance Days	On the KY version of the Medicaid Information Panel for UB Xover Claims, the Coinsurance Days field is currently not updatable. The user needs to be able to enter or update the value in this field. The Coinsurance Days field is coming from the

ID	NAME	DESCRIPTION
		t_clm_quantity table. If a record with 'CD' in the qualifier field does not exist for this claim, then a record needs to be added. If a record does exist, then it needs to be updated. This is currently outside of the current framework, so a System Wide CO may need to be written to address this issue. (See Tech Specs for an alternate plan if this does not work.)
2277	KY Pricing - MAXFLT	In some current KY programs, some procedure codes (benefits) price on a flat fee (not multiplied by the units on the claim).
2278	KY Pricing PT 21 Federal Match	The existing state share processing in CORE should meet the needs of the School Corporations that only receive the federal match from the Medicaid program. Since KCHIP normally has a higher federal matching rate, the fund code needs to be used on the federal match table to indicate if the percentage is for KCHIP or non-KCHIP claims.
2279	KY Phys Eval & Management	In the KY physician program evaluation and management procedures are reimbursed one per year (12 month period). Specific procedures, if billed within a 12-month period, will be paid at a reduced rate (which is usually the max fee amount for a lower coded service).
2290	KY DME Pricing E1399	When billed by a DME provider, procedure code E1399 is manually priced under the following conditions: For detail dates of service (DOS) from 7/1/2002 to 1/2/2003 (inclusive) - The billed amount of the detail is less than or equal to $300.00 - There is no PA number on the claim FYI - The Manual pricing formula is: Price amount is the net cost plus .20% or the detailed billed amount, whichever is the lesser.
2294	KY CE Add Member County	Non Emergency transportation claims from 1/1/2000 through 4/30/2003 are billed on paper and use the member's county code, detail procedure code, and the billing provider's specialty to determine the method to be used to price each detail.
2297	KY Member County Pricing Method	Some claims in Kentucky are priced using rates based on the county in which the member

ID	NAME	DESCRIPTION
		receiving the service resides. The revenue code on the claim will indicate if this pricing method should be used. Currently, hospice claims determine rates in this manner.
2299	KY UB Lab Rate Defaults	When institutional claims are submitted with lab services, a different default method is used to determine the payment amount if there is no rate on file for the lab service. A provider specific reimbursement percentage is used, rather than the non-provider specific percentage used on physician claims. See CO 2306 - all documentation will be updated on this CO.
2300	KY Pct of Charge Pricing Method	Some institutional claims are priced using a provider specific percentage of billed charges for specific revenue codes.
2302	KY UB Rev Code Rates	Some institutional claims are priced using a provider specific per unit rate for specific revenue codes.
2303	KY UB Single Unit Flat Rate	Some institutional claims are priced using a flat unit rate for specific revenue codes. Sometimes, the flat rate is all that will be paid for a service. At other times, the flat rate times the units of service on the detail will be paid.
2305	KY Price Pct by Member Prov DOS	There are situations when the Commonwealth chooses to price institutional claims by a percentage of billed charges. The percentage is assigned based on the combination of member id, billing provider id, and claim service dates. This is often used for out of state organ transplant claims.
2306	KY Rev Proc Fee Pricing	Some institutional claims are priced using a max fee rate assigned to the procedure code that is billed with the revenue code on the detail.
2307	KY Bundled Rate Pricing	Some institutional claims are priced using a provider bundled rate for specific revenue codes. When a bundled rate revenue code is billed on a claim, all non-bundled rate revenue codes are zero paid. Additionally, some professional claims will price using a provider specific bundled rate for certain procedure codes. Once one bundled code on a claim has been priced, all remaining services,

ID	NAME	DESCRIPTION
		including ones that are bundled should be zero paid.
2308	KY Zero Pay Inclusive Rev Codes	Many institutional claims are priced using per diem rates that are inclusive of other services billed on a claim. When inclusive payments are made on a claim, the revenue codes that are not priced using the per diem rates should be zero paid.
2309	KY Max Flat 2 Pricing	Create new pricing methods to support the pricing methods currently used. This KY CO is duplicated in the Core PWB. See Core CO 12480 - all documentation will be updated on this CO.
2337	KY Pricing Child Special Hlth	There are some professional services that are priced using a provider specific percent of charges or unit rate.
2338	KY Pricing Impact Plus	Impact Plus services are priced using a rate set up on a prior authorization. When the provider submits a claim with a per unit charge that is less than the prior approved amount, the system should only pay the amount billed by the provider.
2395	KY Pricing Primary Care / Rural	Primary care and rural health providers are reimbursed using provider specific per diem rates. Some rural health providers are reimbursed using a provider specific percent of charges. The per diem rate is only paid once per claim and is only paid if specific services are billed.
2423	KY Rev Units Flat Pricing	Some outpatient claims can be priced using a revenue code per unit rate that will always be the allowed amount, even if the provider submits a lower charge for the services.
2446	KY UCC Flat Rate Pricing	Create new pricing methods to support the pricing methods currently used. This KY CO is duplicated in the Core PWB. See Core CO 12482 - all documentation will be updated on this CO.
2476	KY OP ASC Pricing	Outpatient claims submitted with revenue code 360 are paid the ASC rate on file for the procedure code billed with the revenue code. These are considered flat rate services and other services submitted on the claim that are not included in the list of flat rate services will be denied.

ID	NAME	DESCRIPTION
2486	KY Transplant Pricing	Inpatient claims related to specific types of transplants are manually or systematically manually priced, rather than using the DRG payment method.
2535	KY Supports Comm Living - Config	Create new pricing method to support current pricing method used for Supports for Community Living.
2990	KY CE Add Relationships	In order to price KY claims correctly, users need to be able to establish reimbursement rules using combinations of the following attributes: revenue code, DRG codes, ICD9 procedure code, and HCPCS procedure codes.
2996	KY CE Bundled Pricing Variable	Some institutional and professional claims are priced using a provider bundled rate for specific revenue and/or procedure codes. When a bundled rate revenue/procedure code is billed on a claim, all non-bundled rate codes are zero paid.
3011	KY Remove Old Mult Surg ASCs	The multiple surgery pricing logic in the CORE system needs to be removed from KY code. It is being replaced with new multiple surgery pricing logic.
3338	KY Pricing Lesser Of Bill/Allow	Some of the pricing methods utilized by the Commonwealth pay the calculated allowed amount, even if it exceeds the billed amount on the claim. The claims pricing finalization process needs to be able to pay these claims at the full allowed amount, rather than cutting back to the lesser of the claim billed or allowed amount.
3394	LTC DME Pricing Cutbacks	DME services billed by Nursing Facilities on LTC claims are priced at a different rate if multiple services are submitted.
3415	Add Rules Setter Xover Pricing	Many crossover claims are priced using methods that compare the regular Medicaid allowable with the Medicare paid amount. As a result, crossover claims will need to first be priced as if they were "straight" Medicaid claims. Once this has been done, a special crossover pricing method will need to be applied to determine the final claim payment amount. In order for the claims engine to know which special crossover pricing method to use, a new field needs to be added to the reimbursement rules panel to allow users to enter the crossover pricing method that applies to the rule for

ID	NAME	DESCRIPTION
		crossover claims.
3418	Crossover Pricing	Many crossover claims are priced using methods that compare the regular Medicaid allowable with the Medicare paid amount. As a result, crossover claims will need to first be priced as if they were "straight" Medicaid claims. Once this has been done, a special crossover pricing method will need to be applied to determine the final claim payment amount. In order for the claims engine to know which special crossover pricing method to use, a new field needs to be added to the reimbursement rules panel to allow users to enter the crossover pricing method that applies to the rule for crossover claims.
3464	Xover and DRG Fund Code Asg	Fund codes are not currently being assigned correctly for DRG inpatient claims. Additionally, the changes to crossover pricing methods required by the Commonwealth are going to change the way that the fund code is assigned on some crossover claims.
3466	NF Ancillary Bypass by PRO	There are some ancillary services that are not covered in nursing facilities if the services are for a member classified as Brain Injury (PRO H), Ventilator (PRO V), or Infectious Disease (PRO C).
3553	Reimb Modifier Pricing Type - CE	Reimbursement Agreement rules are used in the claim-engine to decide how a service is priced. Currently, these rules can be restricted to specific modifier codes. And modifier codes have a default pricing type. States have a need to create reimbursement rules that vary the pricing type a modifier uses. The claim-engine must handle reimbursement rules with modifier pricing types.
3843	Copay Processing - CE	Deduct copay based on rule parameters when processing a claim.
5381	KY Default Pricing II	Many KY programs operate in the following manner. If no procedure rate is found, they take the billed amount and multiple it by a percentage for the Medicaid reimbursement. See additional information under Tech Specs.
6153	Incorr pricing for PT29,45 w/PA	Provider Type 29 and 45 claims that use the Prior Auth record to price are not pricing correctly

ID	NAME	DESCRIPTION
		Provider type 29 (Impact Plus) and 45 (EPSDT Special Services) claims use the authorized amount from the prior authorization record to price. These are not pricing correctly. See attached model office claim examples (ICNs 2007027003586 & 2007027003280). Although just a guess, it looks like authorized amount is being divided twice by the authorized units.
6936	OPASC Pricing Issues	Claims for critical access providers that contain a operating room revenue code (360) are inappropriately being cut back and zero paid. Additionally, dental codes billed with revenue code 360 should be excluded from all final pricing logic and be paid the OP percentage.
7267	Coins shouldn't apply for bypass	The bypass rules for copays should also apply to coinsurance. I was trying to process a claim with V22 diag and the system took the coinsurance. I suspect the bypass rules coded in the copay configuration are not used in the coinsurance logic. This includes PT, claim type, diagnosis and benefit plans for bypassing the coinsurance logic.
7513	Additive copay not working	The EOB shows that a there was a additive copay, but the copay panel does not show any copay. It shows the same on a different icn (see attached).
7940	Primary Care/RHC on ADA Form	Effective 2007/04/01, primary care centers (PT 31) and rural health clinics (PT35) are required to submit dental services on ADA dental claim forms. The system needs to be updated to accept, edit, audit, and price claims for these provider type on dental claim forms.
8173	Coins - only accumulate - outpt	Only accumulate coinsurance for the emergency room service (proc code 99281).

2.6.94 clmsPricing_st -- KY Specific Claims Pricing

Technical Name:	clmsPricing_st
Program Title:	KY Specific Claims Pricing
Programming Language:	C
Description:	Contains Kentucky specific call-out functions related to general claims pricing routines. It includes many routines for accessing pricing data, as well as numerous routines used to price a claim. **Input Parameters:** Detail and Header information related to member programs, performed procedures, or revenue **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** • T_REF_UCC • T_PREVAILING • T_MAXFEE • T_REVENUE_FLAT_FEE • T_PR_UB_LOC_RATE • T_PROC_ASC • T_COPAY • T_RBRVS • T_GPCI • T_PROC_CODE_CONV_FACTOR • T_CONV_FACTOR • T_CLM_AMT_VARIANCE

	• T_XOVER_RATE • T_EOB **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** libclmcomm **Special Logic Notes:** None

2.6.94.1 Change Orders

ID	NAME	DESCRIPTION
2300	KY Pct of Charge Pricing Method	Some institutional claims are priced using a provider specific percentage of billed charges for specific revenue codes.
2302	KY UB Rev Code Rates	Some institutional claims are priced using a provider specific per unit rate for specific revenue codes.
2303	KY UB Single Unit Flat Rate	Some institutional claims are priced using a flat unit rate for specific revenue codes. Sometimes, the flat rate is all that will be paid for a service. At other times, the flat rate times the units of service on the detail will be paid.

2.6.95 clmsProcGroup -- Claims Procedure Groups

Technical Name:	clmsProcGroup
Program Title:	Claims Procedure Groups
Programming Language:	C
Description:	Claims procedure group processing. This module is responsible for reading table T_PROC_GROUP to find if given procedure is in a group. Saks for grouping procedures are hardcoded (I.e. medical supply procedures use sak 15, durable equipment use sak 16, etc.). **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** • T_PROC_CODE_GROUP • T_PROC_TYPE **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** -lmemhash **Special Logic Notes:** None

2.6.95.1 Change Orders

ID	NAME	DESCRIPTION
2777	Add KY Edit 011 - iC 3345	Add KY specific edit in interChange to match KY policy for existing edit 011.
3017	KY PA Edits DME Bypass	When billed by a DME provider, procedure code E1399 should not fail prior authorization edits under the following conditions: - Detail dates of service (DOS) from 7/1/2002 to 1/2/2003 (inclusive) - The billed amount of the detail is less than or equal to $300.00 - There is no PA number on the claim
3843	Copay Processing - CE	Deduct copay based on rule parameters when processing a claim.
4367	ASC Pricing for Physician	Add ASC pricing logic for Physician claims.
6936	OPASC Pricing Issues	Claims for critical access providers that contain a operating room revenue code (360) are inappropriately being cut back and zero paid. Additionally, dental codes billed with revenue code 360 should be excluded from all final pricing logic and be paid the OP percentage.

2.6.96 clmsProgram -- Process claims by member benefit plan

Technical Name:	clmsProgram
Program Title:	Process claims by member benefit plan
Programming Language:	C
Description:	This module is responsible for editing, pricing, and auditing claims based on the hierarchy of the member's eligible benefit plan. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.96.1 Change Orders

ID	NAME	DESCRIPTION
2291	KY DME Pricing PA Purchases	For KY, some DME purchases require prior authorization (a PA indicator is found on the

ID	NAME	DESCRIPTION
		procedure master file). In the current KY system, the rate for purchases that not rentals (RR modifier) are on the PDD file - MCAID-1. Rentals are also priced in the same manner as DME purchases that require PA. Rentals are identified by the use of the RR modifier. In the current KY system, the rate for rentals come from the PR (Procedure Rate) file (Outpatient - pricing specialty - 9).
2298	KY Non DRG Inpatient Pricing	Claims submitted by psychiatric hospitals, as well as many inpatient services are not priced using the DRG pricing method. There are several different methods used based on the providers out of state status, disproportionate share status, and the age of the member at the time the services are rendered. Changes need to be made to inpatient pricing to perform these additional pricing methods, as well as to be able to identify when to use the various pricing methods.
4035	Add indClaimCheck-UB92 Claims	Add the indClaimCheck field for UB claims.

2.6.97 clmsPurePricing -- clmPurePricing

Technical Name:	clmsPurePricing
Program Title:	clmPurePricing
Programming Language:	C
Description:	This program is called by the SOAP server to get pricing value based on pricing method. This program is for the WEB only. Input ATTR: sakProcedure (required) sakPubHlth (required) mod (optional) dos (required) pricingInd (optional) Exit Value: NOT_DOLLAR - normal termination FATAL - abnormal termination Output ATTR: value (pricing amount) **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.97.1 Change Orders

ID	NAME	DESCRIPTION
1416	KY Prior Auth Pricing	Pricing using prior authorizations needs to be modified based on the manner in which the PAs are entered into the system and stored.
1417	DRG Pricing	CO to handle necessary updates to implement KY Pricing policy to DRG pricing methodology.
1476	Core 8816 Benefit Adj Factor 7	See Core CO 8816 all documentation will be updated on this CO Provide the infrastructure to allow modifications to all pricing methodologies that have a variance or incentive from the regular pricing methodology.

2.6.98 clmsReBase -- Claims Member

Technical Name:	clmsReBase
Program Title:	Claims Member
Programming Language:	C
Description:	Claims member processing. This module is responsible for holding member information (I.e. Medicaid id, first name, last name, etc.) and reading member related tables (read_old_pcn to read old member ids). This module is responsible for holding member information and read member related tables. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** • t_re_old_pcn • t_re_base • t_re_name_xref • t_re_lockin_info • t_re_lockin_period • t_re_loc • t_pgm_hierarchy • t_re_elig • t_pub_hlth_pgm • t_recip_review

	• t_re_pmp_assign • t_pmp_svc_loc • t_pr_loc_nm_adr • t_pr_nam • t_pr_adr • t_re_base **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** libclmcomm **Special Logic Notes:** None

2.6.98.1 Change Orders

ID	NAME	DESCRIPTION
2340	KY Pricing Support Comm Living	Price claims for the Supports for Community Living based on the provider, member and claim data. This is a new pricing methodology that is KY specific.

2.6.99 clmsRevenueGroup -- Claims Revenue Groups

Technical Name:	clmsRevenueGroup
Program Title:	Claims Revenue Groups
Programming Language:	C
Description:	Claims revenue group processing. This module is responsible for reading table T_REV_GROUP to find if given revenue is in a group. Saks for revenue types are hardcoded (I.e. emergency is sak 1, surgery sak is 2, etc.). **Input Parameters:** • cdeRev - revenue code • cdeRevType - revenue group type • dteService - date of service **Exit Values:** • EXIT_SUCCESS - if found • EXIT_FAILURE - if not found **Input Files:** None **Output Files:** None **Input Tables:** • t_rev_code_group **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** libclmcomm **Special Logic Notes:**

	None

2.6.99.1 Change Orders

ID	NAME	DESCRIPTION
2606	Add Ky Edit 942 - iC 3334	Add KY specific edit in interChange to match KY policy for existing edit 942 where the revenue code 129 is not valid with any other accommodation revenue code.
2777	Add KY Edit 011 - iC 3345	Add KY specific edit in interChange to match KY policy for existing edit 011.

2.6.100 clmsRulesCaller -- Claims Rules Interface

Technical Name:	clmsRulesCaller
Program Title:	Claims Rules Interface
Programming Language:	C
Description:	This module contains functions that drive the claims rules engine decision making process.

2.6.100.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.101 clmsRulesWrapper -- Rules Engine Setter/Getter Interface

Technical Name:	clmsRulesWrapper
Program Title:	Rules Engine Setter/Getter Interface
Programming Language:	C
Description:	This program contains functions that get the claims data used to make comparisons to the rules criteria for the decision being processed.: None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • none **Output Tables:** • none **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.101.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.102 clmsSpendDown -- SpendDown

Technical Name:	clmsSpendDown
Program Title:	SpendDown
Programming Language:	C
Description:	This module check if long/short term spenddown has been met. It includes functions to retrieve patient liability data, audit patient liability, and insert liability data. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** • T_RE_SPEND_LIAB • T_RE_PAT_LIAB • T_CLM_PATLIAB_X • T_RE_CASE_XREF • T_RE_SPEND_PROV **Output Tables:** None **Sort Criteria:** None **Switches:** None **Link Procedure:** None **Special Logic Notes:** None

	Module Type And Name: • Share Library: libclmwrite.so

2.6.102.1 Change Orders

ID	NAME	DESCRIPTION
1418	Patient Liability	Patient Liability Deduction? Institutional Claims (psychiatric Hospital, Nursing Facility, PRTF) The patient liability amount listed on the member patient liability table is a monthly amount. If the claim is billed for a full month (i. e., 1/1/96 through 1/31/96), the entire patient liability amount is subtracted. If the claim is billed for less than a one month period, the patient liability amount must be prorated. To determine the prorated amount, multiply the Patient Liability amount by the number of covered days and divide the product (of Patient Liability x Covered Days) by the number of days in the month being billed. NOTE: Patient liability is not deducted from Psychiatric Hospital claims for days a member is in Psychiatric Resident Treatment Facility (PRTF) bed reserve status. Also, patient liability is not deducted on Mental Hospital claims if the member is QMB.
1795	Add num_dtl t_clm_patliab_x app	Modify the core objects that access t_clm_patliab_x to use detail number. Existing sql is based on patient liability applied at header level so the detail number to use when modifying existing objects is zero (0).
8359	pat liab - billing and case mgr	Phase I Pat liab is being deducted from ABI waiver when the billing provider is not the case manager. Phase II Pat liab deductions overall fixes.
8728	Pat liab deduction exclusions	SCL, HCB, and Adult Day Care CDO providers should be excluded from the patient liability deduction. This logic is in the claims engine but appears to not be functioning correctly. See claim 2207170003417. These providers have specific contracts that indicate they are CDO.

ID	NAME	DESCRIPTION
9155	Hospice Pat Liab Claims Bad	Hospice claims are currently going to the bad file when attempting to deduct patient liability from claims. They are going bad because the claims engine is attempting to insert duplicate rows into the data base causing a fatal data base error. As a result, these claims are not written to the data base and are not paid to the providers.

2.6.103 clmsTpl -- Claims TPL

Technical Name:	clmsTpl
Program Title:	Claims TPL
Programming Language:	C
Description:	Claims TPL processing. This module is responsible for getting TPL information. It also contains TPL edits. (This program is called by both the claims engine and the TPL billing programs. If changes are made then both subsystems may be impacted.) Input Parameters: None Exit Values: None Input Files: None Output Files: None Input Tables: • T_TPL_RESOURCE • T_COVERAGE_XREF • T_RE_MEDICARE_A • T_RE_MEDICARE_B • T_MEDB_NONCOVERED • T_TPL_MATRIX_XREF **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time.

	Link Procedure: libclmcomm **Special Logic Notes:** None

2.6.103.1 Change Orders

ID	NAME	DESCRIPTION
3668	CORE 13861 - EBP Dx Changes	CORE Change Order 8738 required several different diagnosis for coverage, billing and reimbursement rules. Not all were implemented and definitions of the roles were not clear. Definitions of the diagnosis types/roles: Header Primary: First position of the diagnosis at the header. Header Secondary: Second position of the diagnosis at the header. Admit Diagnosis: Diagnosis indicated as the Admit diagnosis. Emergency Diagnosis: Diagnosis indicated as the Emergency diagnosis. Header Other: Any diagnosis at the header that is not the Header Primary, Header Secondary, Admit or Emergency diagnosis. Header Any: Any of the Header, Secondary, Admit, Emergency, Other diagnosis. Detail Primary: Diagnosis pointed to from the first detail diagnosis pointer. Detail Secondary: Diagnosis pointed to from the second detail diagnosis pointer. Detail Other: Diagnosis pointed to from other than the first and second detail diagnosis pointers. Detail Any : Any of the diagnosis pointed to by the detail.

2.6.104 clmsTplBillings -- Get TPL Info

Technical Name:	clmsTplBillings
Program Title:	Get TPL Info
Programming Language:	C
Description:	This module is responsible for getting TPL information. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_RE_MEDICARE_B • T_RE_MEDICARE_A • T_TPL_RESOURCE • T_COVERAGE_XREF • T_MEDB_NONCOVERED • T_TPL_MATRIX_XREF **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:**

	None **Special Logic Notes:** None

2.6.104.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.105 clmsWrite -- Claims Write

Technical Name:	clmsWrite
Program Title:	Claims Write
Programming Language:	C
Description:	This module is responsible for claims general writing to tables. This module is responsible for inserting common claim information to database: T_CLAIM_ERROR, T_CLAIM_LOCAT, T_HIST_DIRECTORY, T_SUSP_RLHX , T_PA_ITEM_DTL_XREF , T_CLM_PGM_XREF, and T_ADJ_[PHRM|DENT|PHYS|UB92]_XREF Input Parameters: None Exit Values: None Input Files: None Output Files: None Input Tables: None Output Tables: • T_CLAIM_ERROR • T_CLAIM_LOCAT • T_HIST_DIRECTORY • T_SUSP_RLHX • T_PA_ITEM_DTL_XREF • T_CLM_PGM_XREF Other functions will write to other tables **Sort Criteria:** None

	Switches: None at this time. **Link Procedure:** libclmcomm **Special Logic Notes:** None

2.6.105.1 Change Orders

ID	NAME	DESCRIPTION
1121	CC - integration module	Install the ClaimCheck integration module currently running in the Kentucky installation of interChange. Modify existing claims engine modules to ensure that calls to the integration module are placed appropriately. ClaimCheck processing will be performed after edits, pricing and audits have been performed, but before final pricing (application of tpl, spenddown, copay) occurs.
3843	Copay Processing - CE	Deduct copay based on rule parameters when processing a claim.
7084	Dtl not writing T_CLM_PGM_XREF	The DSS loads of claim data (denied claims) in the Parallel environment were not writing the first detail of the claim to the T_CLM_PGM_XREF table. Additionally, many claims do not contain the claim type on the t_clm_pgm_xref table. See attached email under supplemental documentation for more information.
8806	Int adjustment failing valid EOB	All internet adjustments are failing for a valid EOB is required. This information is not being populated from the internet provider initiated adjustments. There is no User ID or adjustment EOB being populated to the claims. An EOB code can be created and it can be used for internet adjustments only. If it needs to be a default value, then this can be done on the batch side. Perhaps a default Analyst ID can also be plugged in the id_clerk field as well.
8964	Amt wrong on T_CLM-PGM_XREF	Table T_CLM_PGM_XREF, column AMT_ENCOUNTER has incorrect amounts. AMT_ALWD is correct but AMT_ENCOUNTER is wrong. AMT_ENCOUNTER should be the amount that the MCO paid. I don't know if

ID	NAME	DESCRIPTION
		AMT_ENCOUNTER is getting overlaid when the claim is priced or if it's never had the right amount in it. UAT Example: ICN 700114001012, SAK 400311365 T_CLM_PGM_XREF NUM_DTL AMT_ENCOUNTER AMT_ALWD 1 12.03 12.03 2 49.04 49.04 3 49.04 49.04 Since amt_alwd is what we priced, amt_encounter should be what the MCO Paid It should be ? NUM_DTL AMT_ENCOUNTER AMT_ALWD 1 10.12 12.03 2 41.23 49.04 3 41.23 49.04

2.6.106 clmsXover -- Crossover Edits

Technical Name:	clmsXover
Program Title:	Crossover Edits
Programming Language:	C
Description:	Cross over claim edits. This module is responsible for holding crossover claim information (like amtDeductible, amtCoinsurance, etc.) and edits (like M'care timely filing). **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** • T_RE_MEDICARE_A • T_RE_MEDICARE_B • T_MCARE_DEDUCTIBLE • T_XOVER_COIN_RATES • T_CLM_PATLIAB_X **Output Tables:** • T_CLM_PATLIAB_X **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** libclmcomm **Special Logic Notes:**

	None

2.6.106.1 Change Orders

ID	NAME	DESCRIPTION
1795	Add num_dtl t_clm_patliab_x app	Modify the core objects that access t_clm_patliab_x to use detail number. Existing sql is based on patient liability applied at header level so the detail number to use when modifying existing objects is zero (0).
2459	Add KY edit 122 - iC 3317	Add KY specific edit in interChange to match KY policy for existing edit 122 for services not approved by Medicare
2567	Add KY Edit 912 - iC 3331	Add KY specific edit in interChange to match KY policy for existing edit 912 where the Medicare paid amount greater than $250.00.
5503	Medicare CO--No Comp XIX Alld Am	ASC ICN--1007005001023--This claim has a deductible of 100.00 and coins-amount of 10.00. The 1st, 2nd, and 4th details have a billed amount of --$25.00 and the 3rd is for 50.00. This claim should have applied 25.00 to the first and 2nd details and the last 50 of the deductible to the 3rd detail. The claim is in paid status with a paid amount of 0.00. More examples of ICNs--1007022001256 1007022001259 1007022001704

2.6.107 clmsakicn -- Claim Sak And ICN Assignment

Technical Name:	clmsakicn
Program Title:	Claim Sak And ICN Assignment
Programming Language:	C
Description:	This module assigns the claim sak and ICN to claims. **Input Parameters:** None **Exit values:** • EXIT_SUCCESS - normal termination. • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time **Link Procedures:** libclmdrvr.so.mak. This is the make file that compiles and links clmsakicn.sc into the libclmdrvr.so library. **Special Logic Notes:** clmsakicn.sc is a subprogram that contains several functions that can be called by other programs.

2.6.107.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.108 clmsccInt -- ClaimCheck Integration Program

Technical Name:	clmsccInt
Program Title:	ClaimCheck Integration Program
Programming Language:	C
Description:	This program drives the integration of ClaimCheck with the claims engine. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.108.1 Change Orders

ID	NAME	DESCRIPTION
1121	CC - integration module	Install the ClaimCheck integration module currently running in the Kentucky installation of interChange.

ID	NAME	DESCRIPTION
		Modify existing claims engine modules to ensure that calls to the integration module are placed appropriately. ClaimCheck processing will be performed after edits, pricing and audits have been performed, but before final pricing (application of tpl, spenddown, copay) occurs.
1135	ccInt implement cc v8.5 changes	Modify the ClaimCheck integration module to meet ClaimCheck v8.5 data requirements: 1. Modify limit checking to allow 100 details to be sent on input, 125 to be evaluated on output 2. Modify input structure to accommodate changes enabling ICD10 3. Modify input structure to accommodate field size changes 4. Modify input structure to accommodate removed data elements 5. Modify output structure accommodate field size changes 6. Modify output structure to accommodate new data elements Add logic to accommodate new claim check edits.
1136	ccInt process outpatient claims	Modify the base integration module as needed to process outpatient claims 1. Modify claim check integration module calls to include outpatient claim processing 2. Modify integration history pull to retrieve outpatient claim history 3. Modify outpatient data structures to maintain submitted and replacement procedure codes
1137	ccInt - modify history pull	Modify the history pull filtering process to pull data according to Kentucky requirements: 1. Do not send denied details to ClaimCheck 2. Limit the history passed for evaluation and management vs. surgical procedures to 90 days before/after the procedure being audited 3. Limit procedures beyond the 90 day limit to those procedures identified as lifetime procedure 4. Do not send Encounter Claims to ClaimCheck
1139	Develop ClaimCheck OPS procedure	Develop procedures for the operation of the enhanced claim editing function within Interchange.

ID	NAME	DESCRIPTION
4035	Add indClaimCheck-UB92 Claims	Add the indClaimCheck field for UB claims.
8310	Claims: Update claim_pass_switch	Provide for the Claims Audit Log File to be turned off, so that claims processing can take place without error.

2.6.109 clmsdntl -- Dental Main Driver

Technical Name:	clmsdntl
Program Title:	Dental Main Driver
Programming Language:	C
Description:	Dental mainline processing. This module contains public service entry points DntlwithCommit and DntlnoCommit. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open,read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** Following tables are used in some functions. See clmsdntlHdr, clmsdntlDtl, clmsdntlAudit modules for tables> **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** -lxml -lgutils -lclmcomm -lclmwrite **Special Logic Notes:** None

2.6.109.1 Change Orders

ID	NAME	DESCRIPTION
2167	PA - Modify calls to find PA	Background: Functions in the base system that match claim services requiring authorization to a PA (service authorization) currently do not use the prior authorization number submitted on the claim as part of the matching criteria. Certain services billed by Kentucky providers that currently require authorization are billed using non-specific codes and are priced from the PA. It is possible for a given member to require multiple services during the same time period that will be billed under the same procedure code. In these cases, the only way to ensure that the claim is priced appropriately is to match the PA number submitted on the claim to the service authorization or suspend the claim for manual pricing. Proposed solution: Modify is_allpa_available function in clmsPaPauth.sc to find a matching PA number (t_pa_pauth vs. PA number supplied on the claim). Look up matches in t_clm_ref table and process using the criteria below: 1. Check for the presence of a clmRef node on the claim and if present 2. Process through each clmRef node checking for the QLF_REFERENCE_ID for a value of 'G1' (prior authorization) and if present 3. Check the CDE_REF_ID for a value that matches one of the PA numbers found in processing the PA cursor and ifa match is found 4. Check the NUM_DTL to verify that it is zero or matches the detail number passed in the call to is_allpa_available.
3843	Copay Processing - CE	Deduct copay based on rule parameters when processing a claim.
7940	Primary Care/RHC on ADA Form	Effective 2007/04/01, primary care centers (PT 31) and rural health clinics (PT35) are required to submit dental services on ADA dental claim forms. The system needs to be updated to accept, edit, audit, and price claims for these provider type on dental claim forms.

2.6.110 clmsdntlAudit -- Dental Auditing

Technical Name:	clmsdntlAudit
Program Title:	Dental Auditing
Programming Language:	C
Description:	This module is responsible for Dental specific audits. This program performs three functions: 1) Formats the dental medpol structure for the current detail. 2) Selects and format dental history for the current member. 3) Executes the common audit program to perform the individual audits. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** -libclmdntl **Special Logic Notes:** None

2.6.110.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.111 clmsdntlDtl -- Dental Detail

Technical Name:	clmsdntlDtl
Program Title:	Dental Detail
Programming Language:	C
Description:	Dental detail processing. This module is responsible for holding dental claim detail information (like tooth number, tooth surface, performing provider number, etc.), routines to set values (like set_dd_xxxx where xxxx might be indDiagTreat for setting detail diagnosis treatment indicator) and get values (like get_dd_xxxx where xxxx might be cdeToothSurface for getting the detail tooth surface code), and edits (like xref_dd_xxxx where xxxx might be procToothSurface to check if procedure code requires tooth surface). **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** • T_PLACE_OF_SERVICE • T_PROC_TOOTH • T_TOOTH_QUADRANT • T_TOOTH • T_TOOTH_SURFACE • T_PR_RST_SVC **Output Tables:** None **Sort Criteria:** None **Switches:**

	None at this time. **Link Procedure:** -libclmdntl **Special Logic Notes:** None

2.6.111.1 Change Orders

ID	NAME	DESCRIPTION
3668	CORE 13861 - EBP Dx Changes	CORE Change Order 8738 required several different diagnosis for coverage, billing and reimbursement rules. Not all were implemented and definitions of the roles were not clear. Definitions of the diagnosis types/roles: Header Primary: First position of the diagnosis at the header. Header Secondary: Second position of the diagnosis at the header. Admit Diagnosis: Diagnosis indicated as the Admit diagnosis. Emergency Diagnosis: Diagnosis indicated as the Emergency diagnosis. Header Other: Any diagnosis at the header that is not the Header Primary, Header Secondary, Admit or Emergency diagnosis. Header Any: Any of the Header, Secondary, Admit, Emergency, Other diagnosis. Detail Primary: Diagnosis pointed to from the first detail diagnosis pointer. Detail Secondary: Diagnosis pointed to from the second detail diagnosis pointer. Detail Other: Diagnosis pointed to from other than the first and second detail diagnosis pointers. Detail Any : Any of the diagnosis pointed to by the detail.

2.6.112 clmsdntlHdr -- Dental Header

Technical Name:	clmsdntlHdr
Program Title:	Dental Header
Programming Language:	C
Description:	Dental header processing. This module is responsible for holding dental header information (like cdeAccident, cdeEmergency, etc.), routines to set values (I.e. set_dh_xxxx where xxxx might be cdeClmSubmitRsn to set claim Submission reason code) and edits (like xref_dh_xxxx where xxxx might be netChargeBalance to make sure net billed = amount billed - tpl amount). **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** • T_ACCIDENT_TYPE • T_PLACE_OF_SERVICE **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** libclmdntl **Special Logic Notes:** None

2.6.112.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.113 clmsdntlPricing -- Dental Pricing

Technical Name:	clmsdntlPricing
Program Title:	Dental Pricing
Programming Language:	C
Description:	Dental pricing. This module is responsible for dental pricing. Contains a single entry point, PRIC_DD_PRICING. **Input Parameters:** The claim detail number for which pricing is requested. **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** • T_REF_UCC • T_PREVAILING • T_MAXFEE **Output Tables:** None **Sort Criteria:** None **Location** Contained within the libclmcomm.so library **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.113.1 Change Orders

ID	NAME	DESCRIPTION
1476	Core 8816 Benefit Adj Factor 7	See Core CO 8816 all documentation will be updated on this CO Provide the infrastructure to allow modifications to all pricing methodologies that have a variance or incentive from the regular pricing methodology.
7940	Primary Care/RHC on ADA Form	Effective 2007/04/01, primary care centers (PT 31) and rural health clinics (PT35) are required to submit dental services on ADA dental claim forms. The system needs to be updated to accept, edit, audit, and price claims for these provider type on dental claim forms.

2.6.114 clmsdntlWrite -- Dental Database Update

Technical Name:	clmsdntlWrite
Program Title:	Dental Database Update
Programming Language:	C
Description:	Dental writing to tables. This module is responsible for inserting the dental claim into database. It uses a schema to convert the claim in XML format to a C structure and xsdDatabaseLoad to write to the data base. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** • T_PA_PAUTH • T_SUSP_DENTAL_HDR • T_DENY_DNTL_HDR • T_PD_DNTL_HDR • T_SUSP_DENTAL_DTL • T_DENY_DNTL_DTL • T_PD_DNTL_DTL • T_DENTAL_HDR_KEYS • T_DENTAL_DTL_KEYS • T_MPHX_DENTAL_DTL **Sort Criteria:**

	None **Switches:** None at this time. **Link Procedure:** -lxml -lgutils -lclmcomm -lclmwrite **Special Logic Notes:** None

2.6.114.1 Change Orders

ID	NAME	DESCRIPTION
3843	Copay Processing - CE	Deduct copay based on rule parameters when processing a claim.
6042	Update Dental Mepol Tbl	Currently, data written to the dental medpol table (T_MPHX_DENTAL_DTL) if the claim has both a billing and performing provider. The dental medpol table gets the performing provider sak on both the prov_billing (sak of the billing provider) and the sak_prov_perf (sak of the performing provider). ICN - 2007024003929 - No performing provider - prov_billing & sak_prov_perf gets the billing provider sak. select SAK_CLAIM, NUM_DTL, SAK_RECIP, PROV_BILLING, SAK_PROV_PERF, SAK_PROV_LOC_BILL, SAK_PROV_LOC_PERF from t_mphx_dental_dtl where sak_claim = 400333338; SAK_CLAIM NUM_DTL SAK_RECIP PROV_BILLING SAK_PROV_PERF SAK_PROV_LOC_BILL SAK_PROV_LOC_PERF ---------- ---------- ---------- ---------- ---------- ---------- ---------- 400333338 1 1120521 500013859 500013859 -1 26363 400333338 2 1120521 500013859 500013859 -1 26363 400333338 3 1120521 500013859 500013859 -1 26363 3 rows selected ICN - 2007024001600 - Billing and performing provider. prov_billing & sak_prov_perf gets the performing provider sak. select SAK_CLAIM, NUM_DTL, SAK_RECIP, PROV_BILLING, SAK_PROV_PERF, SAK_PROV_LOC_BILL, SAK_PROV_LOC_PERF

ID	NAME	DESCRIPTION
		from t_mphx_dental_dtl where sak_claim = 400330346; SAK_CLAIM NUM_DTL SAK_RECIP PROV_BILLING SAK_PROV_PERF SAK_PROV_LOC_BILL ---------- ---------- ---------- ---------- ---------- ---------- -- SAK_PROV_LOC_PERF ---------- 400330346 1 1123993 500045106 500045106 -1 62666 1 row selected.
8277	Claims Engine RTI	Task to be used for the promotion of claims engine changes that do not affect any of the outputs of claims processing and have no effect on the overall error status, payment or disposition of claims but are needed to inprove the throughput of claims, prevent claims from processing to the 'bad' files or to otherwise optimize the performance of claims processing.
8745	Referring provider ID	Referring provider ID numbers are not being moved from the input fields (party identifier) to the correct fields on the claim header/keys tables.
9239	Prov Ids Not on DB	The physician and dental rendering provider ids are not being stored in the claim header when claims are written to the data base tables. Additionally, the first and second other provider ids are not being written to the data base for UB claims. This needs to be corrected to store this data. Existing claims that are missing this data also need to be updated with a one time fix to history.

2.6.115 clmspars -- Parsing Engine

Technical Name:	clmspars
Program Title:	Parsing Engine
Programming Language:	C
Description:	This is the main parsing engine used to break a line into its keyword/value pair. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.115.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.116 clmsphrm -- Pharmacy Main Driver

Technical Name:	clmsphrm
Program Title:	Pharmacy Main Driver
Programming Language:	C
Description:	This module contains the main pharmacy driver routines (PhrmnoCommit, PhrmBatchnoCommit, PhrmwithCommit, PhrmBatchwithCommit, RevnoCommit, RevBatchnoCommit, RevwithCommit, and RevBatchwithCommit) for drug claim processing. These routines also contain various edits. **Input Parameters:** None **Exit Values:** None **Input Files:** • xml format file **Output Files:** • output in xml format **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Commom Library:** libclmphrm.so **Link Procedure:** None **Special Logic Notes:** None

2.6.116.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.117 clmsphrmAudit -- Pharmacy Audits

Technical Name:	clmsphrmAudit
Program Title:	Pharmacy Audits
Programming Language:	C
Description:	This module is responsible for Pharmacy specific audits. This program performs three functions: 1) Formats the pharmacy mepol structure for the current detail 2) Selects and format pharmacyl history for the current member 3) Executes the common audit program to perform the individual audits. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Commom Library:** libclmphrm.so **Link Procedure:** None **Special Logic Notes:** None

2.6.117.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.118 clmsphrmCob -- Pharmacy COB

Technical Name:	clmsphrmCob
Program Title:	Pharmacy COB
Programming Language:	C
Description:	This module is responsible for COB specific to pharmacy. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.118.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.119 clmsphrmDUR -- Pharmacy ProDUR

Technical Name:	clmsphrmDUR
Program Title:	Pharmacy ProDUR
Programming Language:	C
Description:	This program Contains routines to store and retrieve drug utilization review information like drug therapeutic class, dispense date, etc. Routines to store values are named set_xx_xxxx where xxxx might be cancelClaimInd while routines to retrieve values are named get_xx_xxxx where xxxx might be alertDispInfo. Routines to generate reject and accept responses are included as well as captured responses. There are also edit routines named xref_xx_xxxx where xxxx might be proDurHighDose. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.119.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.120 clmsphrmDrug -- Pharmacy Drug

Technical Name:	clmsphrmDrug
Program Title:	Pharmacy Drug
Programming Language:	C
Description:	Pharmacy drug detail information (things like cdeNdc, cdeDrugClass, etc.) and routines to set values (set_rx_xxxx where xxxx might be amtMAC) and get values (get_rx_xxxx where xxxx might be numDrugGcnSeq). There are no real edits in here. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Commom Library:** libclmphrm.so **Link Procedure:** None **Special Logic Notes:** None

2.6.120.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.121 clmsphrmDtl -- Pharmacy Detail

Technical Name:	clmsphrmDtl
Program Title:	Pharmacy Detail
Programming Language:	C
Description:	This program contains routines to store and retrieve drug detail information (cdeNdc, qtyDisp, rePMP - member primary medical provider, etc.). Routines to store information are named set_xd_xxxx where xxxx might be cdeNdc. Edit logic is sometimes buried within these routines. Other routines to edit claims are named xref_xd_xxxx where xxxx might be dteDispDrugDteTermHcfa which ensures that the drug dispense date doesn't come after CMS has expired the drug. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Commom Library:** libclmphrm.so **Link Procedure:** None **Special Logic Notes:**

	None

2.6.121.1 Change Orders

ID	NAME	DESCRIPTION
3668	CORE 13861 - EBP Dx Changes	CORE Change Order 8738 required several different diagnosis for coverage, billing and reimbursement rules. Not all were implemented and definitions of the roles were not clear. Definitions of the diagnosis types/roles: Header Primary: First position of the diagnosis at the header. Header Secondary: Second position of the diagnosis at the header. Admit Diagnosis: Diagnosis indicated as the Admit diagnosis. Emergency Diagnosis: Diagnosis indicated as the Emergency diagnosis. Header Other: Any diagnosis at the header that is not the Header Primary, Header Secondary, Admit or Emergency diagnosis. Header Any: Any of the Header, Secondary, Admit, Emergency, Other diagnosis. Detail Primary: Diagnosis pointed to from the first detail diagnosis pointer. Detail Secondary: Diagnosis pointed to from the second detail diagnosis pointer. Detail Other: Diagnosis pointed to from other than the first and second detail diagnosis pointers. Detail Any : Any of the diagnosis pointed to by the detail.

2.6.122 clmsphrmHdr -- Phamacy Header

Technical Name:	clmsphrmHdr
Program Title:	Pharmacy Header
Programming Language:	C
Description:	This program contains routines to store and retrieve drug header information like idProvPrescrb, indEmergency, indPregnancy, etc. as well as edit routines. Routines to store values are named set_xh_xxxx where xxxx might be cdeClmType. Routines to do edits are named xref_xh_xxxx where xxxx might be numDtls which ensures that the number of details does not exceed the appropriate limit. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Commom Library:** libclmphrm.so **Link Procedure:** None **Special Logic Notes:** None

2.6.122.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.123 clmsphrmPricing -- Pharmacy Pricing

Technical Name:	clmsphrmPricing
Program Title:	Pharmacy Pricing
Programming Language:	C
Description:	This module is responsible for pricing routines specific to pharmacy details. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Commom Library:** libclmphrm.so **Link Procedure:** None **Special Logic Notes:** None

2.6.123.1 Change Orders

ID	NAME	DESCRIPTION
1476	Core 8816 Benefit Adj Factor 7	See Core CO 8816 all documentation will be updated on this CO Provide the infrastructure to allow modifications to all pricing methodologies that have a variance or incentive from the regular pricing methodology.
3843	Copay Processing - CE	Deduct copay based on rule parameters when processing a claim.

2.6.124 clmsphrmReversal -- Pharmacy Reversal

Technical Name:	clmsphrmReversal
Program Title:	Pharmacy Reversal
Programming Language:	C
Description:	This module contains routines to store and retrieve drug reversal information (things like txnStatus, NCPDP version, etc.). Routines to store values are named set_xi_xxxx where xxxx might be txnStatus and routines to retrieve values are named get_xi_xxxx where xxxx might be rejectCode. Other routines exist to add accept and reject information to the reversal XML document. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.124.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.125 clmsphrmWrite -- Pharmacy Database Update

Technical Name:	clmsphrmWrite
Program Title:	Pharmacy Database Update
Programming Language:	C
Description:	This program writes pharmacy claim data to the database, including medpol tables. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** • T_PA_PAUTH • T_SUSP_PHRM_HDR • T_DENY_PHRM_HDR • T_PD_PHARM_HDR • T_PHRM_HDR_KEYS • T_PHRM_DTL_KEYS • T_MPHX_PHRM_DTL **Sort Criteria:** None **Switches:** None at this time. **Commom Library:** libclmphrm.so

	Link Procedure: None **Special Logic Notes:** None

2.6.125.1 Change Orders

ID	NAME	DESCRIPTION
3137	KYAMEND new PA fields	Add new PA fields to clmsphrmWrite.on inserts into T_PA_PAUTH_STATE and T_PA_LINE_ITEM.
3843	Copay Processing - CE	Deduct copay based on rule parameters when processing a claim.
4478	KYAmend - Claims	Modify clmsphrmWrite to default new field on inset into T_PA_LINE_ITEM.

2.6.126 clmsphys -- Physician Main Driver

Technical Name:	clmsphys
Program Title:	Physician Main Driver
Programming Language:	C
Description:	This module contains driver routines (PhysnoCommit and PhyswithCommit) for physician claim processing. Some physician claims edits are also included. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** -lxml -lgutils -lclmcomm -lclmwrite **Special Logic Notes:** None

2.6.126.1 Change Orders

ID	NAME	DESCRIPTION
2090	Add KY Edit 067 - iC 3302	Add KY specific edit in interChange to match KY policy for existing edit 067.
2136	Multiple Surgery - CE	Pricing method needed for multiple surgeries (same date of service).
2167	PA - Modify calls to find PA	Background: Functions in the base system that match claim services requiring authorization to a PA (service authorization) currently do not use the prior authorization number submitted on the claim as part of the matching criteria. Certain services billed by Kentucky providers that currently require authorization are billed using non-specific codes and are priced from the PA. It is possible for a given member to require multiple services during the same time period that will be billed under the same procedure code. In these cases, the only way to ensure that the claim is priced appropriately is to match the PA number submitted on the claim to the service authorization or suspend the claim for manual pricing. Proposed solution: Modify is_allpa_available function in clmsPaPauth.sc to find a matching PA number (t_pa_pauth vs. PA number supplied on the claim). Look up matches in t_clm_ref table and process using the criteria below: 1. Check for the presence of a clmRef node on the claim and if present 2. Process through each clmRef node checking for the QLF_REFERENCE_ID for a value of 'G1' (prior authorization) and if present 3. Check the CDE_REF_ID for a value that matches one of the PA numbers found in processing the PA cursor and ifa match is found 4. Check the NUM_DTL to verify that it is zero or matches the detail number passed in the call to is_allpa_available.
2451	Add KY Edit 036 - iC 3309	Add KY specific edit in interChange to match KY policy for existing edit 036.
2459	Add KY edit 122 - iC 3317	Add KY specific edit in interChange to match KY policy for existing edit 122 for services not approved by Medicare

ID	NAME	DESCRIPTION
2567	Add KY Edit 912 - iC 3331	Add KY specific edit in interChange to match KY policy for existing edit 912 where the Medicare paid amount greater than $250.00.
2777	Add KY Edit 011 - iC 3345	Add KY specific edit in interChange to match KY policy for existing edit 011.
4367	ASC Pricing for Physician	Add ASC pricing logic for Physician claims.
8537	Error code 5030 - Fin error rpt	Claims are being flushed from the financial cycle with error 5030 due to the claim header paid amount not matching the amount paid on the claim program xref. There are several situations causing this to occur: 1. Physician crossover claims are not deducting the TPL amount at the detail level. Since these claims are detail paid, the TPL amount should be deducted at the detail. 2. Some UB claims are paid at the detail level. In these cases, the TPL and patient liability amounts are not being deducted correctly on the claim program xref table. 3. Some physician (CT M) and dental claims do not have a header paid amount but do have paid amounts on the health plan panel.

2.6.127 clmsphysAudit -- Physician Auditing

Technical Name:	clmsphysAudit
Program Title:	Physician Auditing
Programming Language:	C
Description:	This module is responsible for Physician auditing. This program performs three functions: 1) Formats the physician mepol structure for the current detail 2) Selects and format physician history for the current member 3) Executes the common audit program to perform the individual audits. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_DENY_MPHX_PHYS • T_PROC **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** -lxml -lgutils -lclmcomm -lclmwrite

	Special Logic Notes: None

2.6.127.1 Change Orders

ID	NAME	DESCRIPTION
8397	Time of Pick Up field	Claims failing error for time of Pick up missing/invalid
8632	Duplicate DME Payments	For non-crossovers, we appear to have a problem: We are checking same provider, same procedure code, same or overlapping DOS. Then, if both the current and history claims have either an RR modifier or not we check to see if the procedure is E1399 and if the prior auth numbers match. Then, and only then, will we post the dupe audit. Additionally, we are not posting the audit for E1399 when the pa numbers are the same.

2.6.128 clmsphysDiag -- Physician Diagnosis

Technical Name:	clmsphysDiag
Program Title:	Physician Diagnosis
Programming Language:	C
Description:	This module is responsible for physician claim diagnosis code related edits. Edits are name xref_md_xxxx where xxxx might be diagSex. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_DIAG_LIMIT **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** -lxml -lgutils -lclmcomm -lclmwrite **Special Logic Notes:** None

2.6.128.1 Change Orders

ID	NAME	DESCRIPTION
3668	CORE 13861 - EBP Dx Changes	CORE Change Order 8738 required several different diagnosis for coverage, billing and reimbursement rules. Not all were implemented and definitions of the roles were not clear. Definitions of the diagnosis types/roles: Header Primary: First position of the diagnosis at the header. Header Secondary: Second position of the diagnosis at the header. Admit Diagnosis: Diagnosis indicated as the Admit diagnosis. Emergency Diagnosis: Diagnosis indicated as the Emergency diagnosis. Header Other: Any diagnosis at the header that is not the Header Primary, Header Secondary, Admit or Emergency diagnosis. Header Any: Any of the Header, Secondary, Admit, Emergency, Other diagnosis. Detail Primary: Diagnosis pointed to from the first detail diagnosis pointer. Detail Secondary: Diagnosis pointed to from the second detail diagnosis pointer. Detail Other: Diagnosis pointed to from other than the first and second detail diagnosis pointers. Detail Any : Any of the diagnosis pointed to by the detail.

2.6.129 clmsphysDtl -- Physician Detail

Technical Name:	clmsphysDtl
Program Title:	Physician Detail
Programming Language:	C
Description:	This program is responsible for holding physician claim detail information (like cdeEmergency, indEmergency, diagTreat, etc.), routines to set values (like set_md_xxxx where xxxx might be indClaimCheck) and edits (like xref_md_xxxx where xxxx might be filingLimit for checking if date of service exceeds filing limits). **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_PLACE_OF_SERVICE • T_CLM_COMP_DAYS **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** -lxml -lgutils -lclmcomm -lclmwrite **Special Logic Notes:**

	None

2.6.129.1 Change Orders

ID	NAME	DESCRIPTION
2090	Add KY Edit 067 - iC 3302	Add KY specific edit in interChange to match KY policy for existing edit 067.
2280	KY School Based Group Pricing	The system needs to determine the allowable cost per person when a group session is submitted on a claim. The allowed amount will be a per person rate. This is a new pricing methodology that is KY specific.
2292	KY Adult/Child Case Pricing	For KY programs Adult & Children Targeted Case from dates of service (DOS) from 1/1/2003 to current date price by taking the lesser of the Average Unit Cost or the rate from the PDD file. PDD - MCAID-1 rate for Children Targeted Case (PT 28). PDD - MCAID-2 rate for Adult Targeted Case (PT 27). In either case, the rate is not multiplied by units on the claim and the amount is not compared to the claims billed amount. On 12/31/2002 and all dates prior, for Adult & Children Targeted Case price by taking the lesser of the rate from the PDD file or the claim detail billed amount. PDD - MCAID-1 rate for Children Targeted Case (PT 28). PDD - MCAID-2 rate for Adult Targeted Case (PT 27). In either case, the rate is not multiplied by units on the claim. See CO 2277 - pricing by MAXFLT pricing methodology.
8397	Time of Pick Up field	Claims failing error for time of Pick up missing/invalid
8453	Valid values for PT 30 edit 3304	Edit 3304 fails if employee ID is missing or invalid. The valid employee ID values for provider type 30 (Community Mental Health) are 0001 - 9999 rather than 1111 - 9999, as is currently coded in the edit.

2.6.130 clmsphysDtl_st -- KY Physician Detail

Technical Name:	clmsphysDtl_st
Program Title:	KY Physician Detail
Programming Language:	C
Description:	Contains Kentucky specific call-out functions related to physician claim detail information and edits. **Input Parameters:** This program is responsible for holding physician claim detail information specific to KY requirements. **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** libclmuphys_st **Special Logic Notes:** None

2.6.130.1 Change Orders

ID	NAME	DESCRIPTION
2250	Decision Form Rule Getter-EE 163	KY specific CO

ID	NAME	DESCRIPTION
		Add getter for miscellaneous code Employer ID when the qualifier is = EI

2.6.131 clmsphysHdr -- Physician Header

Technical Name:	clmsphysHdr
Program Title:	Physician Header
Programming Language:	C
Description:	This program is responsible for holding physician claim header information (things like cdeDiag, indAttachment, numPatAcct, etc.), routines to store the values (like set_mh_xxxx where xxxx might be amtNetBilled) and edits (like xref_mh_xxxx where xxxx might be cdeDiag to validate a given diagnosis code). **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_ACCIDENT_TYPE • T_DIAGNOSIS • T_PLACE_OF_SERVICE **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** -lxml -lgutils -lclmcomm -lclmwrite

	Special Logic Notes: None

2.6.131.1 Change Orders

ID	NAME	DESCRIPTION
3668	CORE 13861 - EBP Dx Changes	CORE Change Order 8738 required several different diagnosis for coverage, billing and reimbursement rules. Not all were implemented and definitions of the roles were not clear. Definitions of the diagnosis types/roles: Header Primary: First position of the diagnosis at the header. Header Secondary: Second position of the diagnosis at the header. Admit Diagnosis: Diagnosis indicated as the Admit diagnosis. Emergency Diagnosis: Diagnosis indicated as the Emergency diagnosis. Header Other: Any diagnosis at the header that is not the Header Primary, Header Secondary, Admit or Emergency diagnosis. Header Any: Any of the Header, Secondary, Admit, Emergency, Other diagnosis. Detail Primary: Diagnosis pointed to from the first detail diagnosis pointer. Detail Secondary: Diagnosis pointed to from the second detail diagnosis pointer. Detail Other: Diagnosis pointed to from other than the first and second detail diagnosis pointers. Detail Any : Any of the diagnosis pointed to by the detail.

2.6.132 clmsphysPricing -- Physician Pricing

Technical Name:	clmsphysPricing
Program Title:	Physician Pricing
Programming Language:	C
Description:	This module is responsible for physician claim pricing. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.132.1 Change Orders

ID	NAME	DESCRIPTION
1476	Core 8816 Benefit Adj Factor 7	See Core CO 8816 all documentation will be updated on this CO Provide the infrastructure to allow modifications to all

ID	NAME	DESCRIPTION
		pricing methodologies that have a variance or incentive from the regular pricing methodology.
2277	KY Pricing - MAXFLT	In some current KY programs, some procedure codes (benefits) price on a flat fee (not multiplied by the units on the claim).
2280	KY School Based Group Pricing	The system needs to determine the allowable cost per person when a group session is submitted on a claim. The allowed amount will be a per person rate. This is a new pricing methodology that is KY specific.
2291	KY DME Pricing PA Purchases	For KY, some DME purchases require prior authorization (a PA indicator is found on the procedure master file). In the current KY system, the rate for purchases that not rentals (RR modifier) are on the PDD file - MCAID-1. Rentals are also priced in the same manner as DME purchases that require PA. Rentals are identified by the use of the RR modifier. In the current KY system, the rate for rentals come from the PR (Procedure Rate) file (Outpatient - pricing specialty - 9).
2292	KY Adult/Child Case Pricing	For KY programs Adult & Children Targeted Case from dates of service (DOS) from 1/1/2003 to current date price by taking the lesser of the Average Unit Cost or the rate from the PDD file. PDD - MCAID-1 rate for Children Targeted Case (PT 28). PDD - MCAID-2 rate for Adult Targeted Case (PT 27). In either case, the rate is not multiplied by units on the claim and the amount is not compared to the claims billed amount. On 12/31/2002 and all dates prior, for Adult & Children Targeted Case price by taking the lesser of the rate from the PDD file or the claim detail billed amount. PDD - MCAID-1 rate for Children Targeted Case (PT 28). PDD - MCAID-2 rate for Adult Targeted Case (PT 27). In either case, the rate is not multiplied by units on the claim. See CO 2277 - pricing by MAXFLT pricing methodology.

ID	NAME	DESCRIPTION
2446	KY UCC Flat Rate Pricing	Create new pricing methods to support the pricing methods currently used. This KY CO is duplicated in the Core PWB. See Core CO 12482 - all documentation will be updated on this CO.
3843	Copay Processing - CE	Deduct copay based on rule parameters when processing a claim.
4367	ASC Pricing for Physician	Add ASC pricing logic for Physician claims.
6882	Claims not pricing correctly	Claims not pricing correctly for DME-unable to test copay/coins The claim has 1 detail lines, the claim has a PA with 500 aut amt with 2 units. Claim billed 1 unit but the paid amount was 125 which seems like is autamt/autunit/autunit.The xml is in UAT ttg under automated txn\moclaims\sandeep_test\coins\PT90.xml
7940	Primary Care/RHC on ADA Form	Effective 2007/04/01, primary care centers (PT 31) and rural health clinics (PT35) are required to submit dental services on ADA dental claim forms. The system needs to be updated to accept, edit, audit, and price claims for these provider type on dental claim forms.

2.6.133 clmsphysPricing_st -- KY PHYS Pricing Module

Technical Name:	clmsphysPricing_st
Program Title:	KY PHYS Pricing Module
Programming Language:	C
Description:	Contains Kentucky specific call-out functions related to general Physician claims pricing. **Input Parameters:** This module is responsible for physician claim pricing for KY specific pricing requirements. **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** libclmuphys_st **Special Logic Notes:** None

2.6.133.1 Change Orders

ID	NAME	DESCRIPTION
2280	KY School Based Group Pricing	The system needs to determine the allowable cost

ID	NAME	DESCRIPTION
		per person when a group session is submitted on a claim. The allowed amount will be a per person rate. This is a new pricing methodology that is KY specific.
2292	KY Adult/Child Case Pricing	For KY programs Adult & Children Targeted Case from dates of service (DOS) from 1/1/2003 to current date price by taking the lesser of the Average Unit Cost or the rate from the PDD file. PDD - MCAID-1 rate for Children Targeted Case (PT 28). PDD - MCAID-2 rate for Adult Targeted Case (PT 27). In either case, the rate is not multiplied by units on the claim and the amount is not compared to the claims billed amount. On 12/31/2002 and all dates prior, for Adult & Children Targeted Case price by taking the lesser of the rate from the PDD file or the claim detail billed amount. PDD - MCAID-1 rate for Children Targeted Case (PT 28). PDD - MCAID-2 rate for Adult Targeted Case (PT 27). In either case, the rate is not multiplied by units on the claim. See CO 2277 - pricing by MAXFLT pricing methodology.

2.6.134 clmsphysProc -- Physician Procedure

Technical Name:	clmsphysProc
Program Title:	Physician Procedure
Programming Language:	C
Description:	This module is responsible for holding physician claim procedure code and modifier related edits. Edits are named xref_md_xxxx where xxxx might be deliveryLimit to check if date of service exceed Delivery Service filing limits. **Input Parameters:** None **Exit Values:** None. Not a program. **Input Files:** None **Output Files:** None **Input Tables:** • T_RE_LOC • T_RE_LOC_DIAG_RNG • T_RE_LOC_PROC_RNG **Output Tables:** None **Sort Criteria:** None **Switches:** None **Link Procedure:** None. Not a program. **Special Logic Notes:** None

2.6.134.1 Change Orders

ID	NAME	DESCRIPTION
3668	CORE 13861 - EBP Dx Changes	CORE Change Order 8738 required several different diagnosis for coverage, billing and reimbursement rules. Not all were implemented and definitions of the roles were not clear. Definitions of the diagnosis types/roles: **Header Primary:** First position of the diagnosis at the header. **Header Secondary:** Second position of the diagnosis at the header. **Admit Diagnosis:** Diagnosis indicated as the Admit diagnosis. **Emergency Diagnosis:** Diagnosis indicated as the Emergency diagnosis. **Header Other:** Any diagnosis at the header that is not the Header Primary, Header Secondary, Admit or Emergency diagnosis. **Header Any:** Any of the Header, Secondary, Admit, Emergency, Other diagnosis. **Detail Primary:** Diagnosis pointed to from the first detail diagnosis pointer. **Detail Secondary:** Diagnosis pointed to from the second detail diagnosis pointer. **Detail Other:** Diagnosis pointed to from other than the first and second detail diagnosis pointers. **Detail Any** : Any of the diagnosis pointed to by the detail.

2.6.135 clmsphysProvEdit -- Physician Provider Edits

Technical Name:	clmsphysProvEdit
Program Title:	Physician Provider Edits
Programming Language:	C
Description:	This module is responsible for holding physician claim provider related edits. Edits are named xref_md_xxxx where xxxx might be provRendLockin which ensures that the member is not locked in to a particular provider or claim type. **Input Parameters:** None **Exit Values:** None. Not a program. **Input Files:** None **Output Files:** None **Input Tables:** • T_PR_RST_SVC **Output Tables:** None **Sort Criteria:** None **Switches:** None **Link Procedure:** None. Not a program. **Special Logic Notes:** None

2.6.135.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.136 clmsphysRecipEdit -- Physician Recipient Edits

Technical Name:	clmsphysRecipEdit
Program Title:	Physician Recipient Edits
Programming Language:	C
Description:	This module is responsible for physician claim member related edits. **Input Parameters:** None **Exit Values:** None. Not a program. **Input Files:** None **Output Files:** None **Input Tables:** • T_RE_LOC **Output Tables:** None **Sort Criteria:** None **Switches:** None **Link Procedure:** None. Not a program. **Special Logic Notes:** None

2.6.136.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.137 clmsphysWrite -- Physician Database Update

Technical Name:	clmsphysWrite
Program Title:	Physician Database Update
Programming Language:	C
Description:	This module is responsible for inserting the CMS claim into the database. **Input Parameters:** None **Exit Values:** None. **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None **Link Procedure:** None. Not a program. **Special Logic Notes:** None

2.6.137.1 Change Orders

ID	NAME	DESCRIPTION
2115	CORE Dup audits - phys write	This is a duplicate for CORE CO 10783. Update core to write state specific physician history data to T_MPHX_MISC.

ID	NAME	DESCRIPTION
3843	Copay Processing - CE	Deduct copay based on rule parameters when processing a claim.
8277	Claims Engine RTI	Task to be used for the promotion of claims engine changes that do not affect any of the outputs of claims processing and have no effect on the overall error status, payment or disposition of claims but are needed to inprove the throughput of claims, prevent claims from processing to the 'bad' files or to otherwise optimize the performance of claims processing.
8745	Referring provider ID	Referring provider ID numbers are not being moved from the input fields (party identifier) to the correct fields on the claim header/keys tables.
9239	Prov Ids Not on DB	The physician and dental rendering provider ids are not being stored in the claim header when claims are written to the data base tables. Additionally, the first and second other provider ids are not being written to the data base for UB claims. This needs to be corrected to store this data. Existing claims that are missing this data also need to be updated with a one time fix to history.

2.6.138 clmsphysWrite_st -- Physician database update for state specific data elements

Technical Name:	clmsphysWrite_st
Program Title:	Physician database update for state specific data elements
Programming Language:	C
Description:	Functions required to insert claim data to tables containing data elements specific to KY processing. **Input Parameters:** This module is responsible for inserting the CMS claim into the database for KY specific data requirements. **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** libclmphys_st **Special Logic Notes:** None

2.6.138.1 Change Orders

ID	NAME	DESCRIPTION
8397	Time of Pick Up field	Claims failing error for time of Pick up missing/invalid

2.6.139 clmstats -- Claim Statistics

Technical Name:	clmstats
Program Title:	Claim Statistics
Programming Language:	C
Description:	This program saves all of the statistics for the specified transaction. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.139.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.140 clmsub92 -- UB92 Main Driver

Technical Name:	clmsub92
Program Title:	UB92 Main Driver
Programming Language:	C
Description:	UB92 mainline processing. UB claim driver routines. Driver routines include UB92noCommit and UB92withCommit. **Input Parameters:** Pointer to XML file containing claim information. **Exit Values:** None **Input Files:** XML document portraying the ub92/837I based claim **Output Files:** Processed claim in XML format. **Output Tables:** None This module is contained in the libclmub92.so library **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.140.1 Change Orders

ID	NAME	DESCRIPTION
1	file	DUMMY This is a dummy change order. Do not change.
1136	ccInt process outpatient claims	Modify the base integration module as needed to process outpatient claims

ID	NAME	DESCRIPTION
		1. Modify claim check integration module calls to include outpatient claim processing 2. Modify integration history pull to retrieve outpatient claim history 3. Modify outpatient data structures to maintain submitted and replacement procedure codes
2080	Add KY Edit 054 - iC 3300	Add KY specific edit in interChange to match KY policy for existing edit 054.
2084	Add KY Edit 056 - iC 3301	Add KY specific edit in interChange to match KY policy for existing edit 056.
2167	PA - Modify calls to find PA	Background: Functions in the base system that match claim services requiring authorization to a PA (service authorization) currently do not use the prior authorization number submitted on the claim as part of the matching criteria. Certain services billed by Kentucky providers that currently require authorization are billed using non-specific codes and are priced from the PA. It is possible for a given member to require multiple services during the same time period that will be billed under the same procedure code. In these cases, the only way to ensure that the claim is priced appropriately is to match the PA number submitted on the claim to the service authorization or suspend the claim for manual pricing. Proposed solution: Modify is_allpa_available function in clmsPaPauth.sc to find a matching PA number (t_pa_pauth vs. PA number supplied on the claim). Look up matches in t_clm_ref table and process using the criteria below: 1. Check for the presence of a clmRef node on the claim and if present 2. Process through each clmRef node checking for the QLF_REFERENCE_ID for a value of 'G1' (prior authorization) and if present 3. Check the CDE_REF_ID for a value that matches one of the PA numbers found in processing the PA cursor and ifa match is found 4. Check the NUM_DTL to verify that it is zero or matches the detail number passed in the call to is_allpa_available.
2283	Add KY Edit 271 - iC 3305	Add KY specific edit in interChange to match KY policy for existing edits 103 and 271 (Legacy edit recycles for edit 271) Edit 103/271 - Member applied income not current for date of service.
2450	Add KY Edit 026 - iC 3308	Add KY specific edit in interChange to match KY policy for

ID	NAME	DESCRIPTION
		existing edit 026.
2452	Add KY Edit 051 - iC 3361	Add KY specific edit in interChange to match KY policy for existing edit 051 for claim type L, provider types 11 and 12, and claim type A, provider types 01, 02, 11, and 12.
2458	Add KY Edit 123 - iC 3372	Add KY specific edit in interChange to match KY policy for existing edit 123 for claim may not span a member's 1st birthday.
2502	Modify iC edit 368 (KY edit 083)	Modify interchange edit 368 to match KY policy for existing edit 083. See exclusion criteria under technical specifications.
2503	Modify iC edit 365 (KY edit 084)	Modify interchange edit 365 to match KY policy for existing edit 084. See exclusion criteria under technical specifications.
2606	Add Ky Edit 942 - iC 3334	Add KY specific edit in interChange to match KY policy for existing edit 942 where the revenue code 129 is not valid with any other accommodation revenue code.
2777	Add KY Edit 011 - iC 3345	Add KY specific edit in interChange to match KY policy for existing edit 011.
3149	Add Ky Edit 310 - iC 3360	Add KY specific edit in interChange to match KY policy for existing edit 310 where a new admissions not payable because of provider non-compliance.
4035	Add indClaimCheck-UB92 Claims	Add the indClaimCheck field for UB claims.

2.6.141 clmsub92Audit -- UB92 Auditing

Technical Name:	clmsub92Audit
Program Title:	UB92 Auditing
Programming Language:	C
Description:	This module is responsible for UB auditing. This program performs three functions: 1) Formats the UB medpol structure for the current detail 2) Selects and format UB history for the current member 3) Executes the common audit program to perform the individual audits. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None **Link Procedure:** None **Special Logic Notes:** None

2.6.141.1 Change Orders

ID	NAME	DESCRIPTION
4035	Add indClaimCheck-UB92 Claims	Add the indClaimCheck field for UB claims.
8955	Inst Crossover Dupe Audit	Institutional crossover claims are duping against details on the same claim. Example ICN: 1107186000425.

2.6.142 clmsub92CondX -- UB92 Condition Codes

Technical Name:	clmsub92CondX
Program Title:	UB92 Condition Codes
Programming Language:	C
Description:	UB92 condition codes processing. This module is responsible for processing UB claim header condition code information and edits. **Input Parameters:** None **Exit Values:** None. Not a program. **Input Files:** None **Output Files:** None **Input Tables:** • T_CONDITION **Output Tables:** None **Sort Criteria:** None **Switches:** None **Link Procedure:** None. Not a program. **Special Logic Notes:** None

2.6.142.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.143 clmsub92DRGPricing -- UB92 DRG Pricing

Technical Name:	clmsub92DRGPricing
Program Title:	UB92 DRG Pricing
Programming Language:	C
Description:	UB92 DRG pricing program. This program supports DRG pricing method for Inpatient claims. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.143.1 Change Orders

ID	NAME	DESCRIPTION
1417	DRG Pricing	CO to handle necessary updates to implement KY Pricing policy to DRG pricing methodology.

ID	NAME	DESCRIPTION
1476	Core 8816 Benefit Adj Factor 7	See Core CO 8816 all documentation will be updated on this CO Provide the infrastructure to allow modifications to all pricing methodologies that have a variance or incentive from the regular pricing methodology.
1674	Core 10357 WI 2408 DRG 1	See Core CO 10357 all documentation will be updated on this CO See WI CO 2408 During the DRG grouping process the DRG grouper returns additional information that needs to be stored and displayed.
1676	Core 10356 WI 2402 DRG 3	See Core CO 10356 all documentation will be updated on this CO See WI CO 2402 The Kentucky DRG pricing methodology requires that some DRG codes be replaced with more specific DRG codes. This is required to price neonatal and transplant DRG codes with the rates established by Kentucky policy.

2.6.144 clmsub92DiagX -- UB92 Diagnosis

Technical Name:	clmsub92DiagX
Program Title:	UB92 Diagnosis
Programming Language:	C
Description:	UB92 diagnosis code processing. This module is responsible for holding UB claim header diagnosis code information (like cdeDiag, indEmergency, etc.) and diagnosis code related functions to store (set_ug_xxxx where xxxx might be cdeDiag), retrieve (get_ug_xxxx where xxxx might be sakDiagPrimary), or an edit (xref_ub_xxxx where xxxx might be primarydiag to validate that a primary and admit diagnosis code has been specified). **Input Parameters:** None **Exit Values:** None. Not a program. **Input Files:** None **Output Files:** None **Input Tables:** • T_DIAGNOSIS **Output Tables:** None **Sort Criteria:** None **Switches:** None **Link Procedure:** None. Not a program. **Special Logic Notes:** None

2.6.144.1 Change Orders

ID	NAME	DESCRIPTION
3668	CORE 13861 - EBP Dx Changes	CORE Change Order 8738 required several different diagnosis for coverage, billing and reimbursement rules. Not all were implemented and definitions of the roles were not clear. Definitions of the diagnosis types/roles: **Header Primary:** First position of the diagnosis at the header. **Header Secondary:** Second position of the diagnosis at the header. **Admit Diagnosis:** Diagnosis indicated as the Admit diagnosis. **Emergency Diagnosis:** Diagnosis indicated as the Emergency diagnosis. **Header Other:** Any diagnosis at the header that is not the Header Primary, Header Secondary, Admit or Emergency diagnosis. **Header Any:** Any of the Header, Secondary, Admit, Emergency, Other diagnosis. **Detail Primary:** Diagnosis pointed to from the first detail diagnosis pointer. **Detail Secondary:** Diagnosis pointed to from the second detail diagnosis pointer. **Detail Other:** Diagnosis pointed to from other than the first and second detail diagnosis pointers. **Detail Any** : Any of the diagnosis pointed to by the detail.
6936	OPASC Pricing Issues	Claims for critical access providers that contain a operating room revenue code (360) are inappropriately being cut back and zero paid. Additionally, dental codes billed with revenue code 360 should be excluded from all final pricing logic and be paid the OP percentage.

2.6.145 clmsub92Dtl -- UB92 Detail

Technical Name:	clmsub92Dtl
Program Title:	UB92 Detail
Programming Language:	C
Description:	UB92 detail processing. This module is responsible for holding UB detail information (things like amount billed, quantity units billed, claim status, amount paid, etc.) and edits (things like valid revenue code, provider type is valid for claim type, etc.). Routines to set the detail information (set_ud_xxxx where xxxx is a field name like idProvAttending), get the detail information (get_ud_xxxx), or do a validation (xref_ud_xxxx where xxxx might be cdeRevenueProvSpecialty to validate that the provider specialty is valid for the revenue code used). **Input Parameters:** None **Exit Values:** None. Not a program. **Input Files:** None **Output Files:** None **Input Tables:** • T_REVENUE_CODE • T_PR_HB_LIC • T_MC_MSTR_CERT_CODE • T_COV_DIAG_XREF • T_COV_BILL_XREF • T_COV_REV_XREF • T_COV_TYPE_XREF • T_COV_SPEC_XREF • T_REV_SPEC_RSTN

	• T_PR_SPEC • T_PR_RST_SVC • T_REV_PROC_XREF • T_TOB_PROC_XREF • T_RE_BASE • T_ATTACHMENT_XREF • T_RE_ELIG • T_RE_PMP_ASSIGN • T_PMP_SVC_LOC • T_PR_PROV • T_PR_PREV_PROV **Output Tables:** None **Sort Criteria:** None **Switches:** None **Link Procedure:** None. Not a program. **Special Logic Notes:** None

2.6.145.1 Change Orders

ID	NAME	DESCRIPTION
2606	Add Ky Edit 942 - iC 3334	Add KY specific edit in interChange to match KY policy for existing edit 942 where the revenue code 129 is not valid with any other accommodation revenue code.
4035	Add indClaimCheck-UB92 Claims	Add the indClaimCheck field for UB claims.

2.6.146 clmsub92HHPricing -- UB92 Home Health Pricing

Technical Name:	clmsub92HHPricing
Program Title:	UB92 Home Health Pricing
Programming Language:	C
Description:	UB92 home health pricing. This module is responsible for pricing UB Home Health claims. **Input Parameters:** Detail and header information related to member programs, performed procedures, and modifiers **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None This module is contained in the libclmub92.so library **Sort Criteria:** None **Switches:** None **Link Procedure:** None **Special Logic Notes:** None

2.6.146.1 Change Orders

ID	NAME	DESCRIPTION

ID	NAME	DESCRIPTION
1476	Core 8816 Benefit Adj Factor 7	See Core CO 8816 all documentation will be updated on this CO Provide the infrastructure to allow modifications to all pricing methodologies that have a variance or incentive from the regular pricing methodology.
2300	KY Pct of Charge Pricing Method	Some institutional claims are priced using a provider specific percentage of billed charges for specific revenue codes.
2301	KY Hospice Uses LTC Rates	Respite care services billed on hospice claims are priced using a percentage of the per diem rate for the facility that the member is placed in. The LTC facility ID is submitted on the claim that contains the respite care services in a local use field.
2302	KY UB92 Rev Code Rates	Some institutional claims are priced using a provider specific per unit rate for specific revenue codes.
2303	KY UB92 Single Unit Flat Rate	Some institutional claims are priced using a flat unit rate for specific revenue codes. Sometimes, the flat rate is all that will be paid for a service. At other times, the flat rate times the units of service on the detail will be paid.
2304	KY Rev Code Pricing Xwalk	Hospice claims submitted with services for inpatient respite care and general inpatient care are priced using the provider procedure rate on file up to a certain number of units. In the case of inpatient respite care, the last unit billed on a detail is priced using the routine home care rate. If more than four units are billed, all units after the fourth unit are priced using the routine home care rate. The only exception to this additional unit pricing is if the patient status is 41 and the claim header TDOS is equal to the detail TDOS. In this case, all units price using the inpatient respite care rate. In the case of general inpatient care, the last unit billed on a detail is priced using the routine home care rate. The exceptions to this additional unit pricing are: 1) If the patient status is 41 and the claim date of discharge is equal to the detail TDOS. In this case, all units price using the inpatient respite care rate. 2) If the patient status is 30 and the last digit of the type of bill is a 2 or 3. In this case, all units price using the inpatient respite care rate.

2.6.147 clmsub92Hdr -- UB92 Header

Technical Name:	clmsub92Hdr
Program Title:	UB92 Header
Programming Language:	C
Description:	UB92 header processing. This module is responsible for holding UB claim header information (like cdeTypeOfBill, amtPaidPatn, etc.), routines to store values (like set_uh_xxxx where xxxx might be cdeProvLoc), routines to retrieve values (like get_uh_xxxx where xxxx might be amtTpl), and edits (like xref_uh_xxxx where xxxx might be dteLastSvcDteIcn to verify that the last date of service is before the date in the claim's ICN). **Input Parameters:** None **Exit Values:** None. Not a program. **Input Files:** None **Output Files:** None **Input Tables:** • T_TOB_CT_XREF • T_PR_PROV • T_PR_PREV_PROV • T_PR_HB_LIC • T_PATIENT_STAT • T_ADMIT_SOURCE • T_ADMIT_TYPE • T_MDC • T_DIAG_LIMIT • T_PROC_ICD9_DIA

	• T_PR_EDIT_EXEMPT • T_ERROR_DISP • T_RE_LOC • T_RE_LOC_TOB_XF **Output Tables:** None **Sort Criteria:** None **Switches:** None **Link Procedure:** None. Not a program. **Special Logic Notes:** None

2.6.147.1 Change Orders

ID	NAME	DESCRIPTION
1674	Core 10357 WI 2408 DRG 1	See Core CO 10357 all documentation will be updated on this CO See WI CO 2408 During the DRG grouping process the DRG grouper returns additional information that needs to be stored and displayed.
2080	Add KY Edit 054 - iC 3300	Add KY specific edit in interChange to match KY policy for existing edit 054.
2084	Add KY Edit 056 - iC 3301	Add KY specific edit in interChange to match KY policy for existing edit 056.
2298	KY Non DRG Inpatient Pricing	Claims submitted by psychiatric hospitals, as well as many inpatient services are not priced using the DRG pricing method. There are several different methods used based on the providers out of state status, disproportionate share status, and the age of the member at the time the services are rendered. Changes need to be made to inpatient pricing to perform these additional pricing methods, as well as to be able to identify when to use the various pricing methods.

ID	NAME	DESCRIPTION
2452	Add KY Edit 051 - iC 3361	Add KY specific edit in interChange to match KY policy for existing edit 051 for claim type L, provider types 11 and 12, and claim type A, provider types 01, 02, 11, and 12.
2458	Add KY Edit 123 - iC 3372	Add KY specific edit in interChange to match KY policy for existing edit 123 for claim may not span a member's 1st birthday.
2502	Modify iC edit 368 (KY edit 083)	Modify interchange edit 368 to match KY policy for existing edit 083. See exclusion criteria under technical specifications.
2503	Modify iC edit 365 (KY edit 084)	Modify interchange edit 365 to match KY policy for existing edit 084. See exclusion criteria under technical specifications.
3149	Add Ky Edit 310 - iC 3360	Add KY specific edit in interChange to match KY policy for existing edit 310 where a new admissions not payable because of provider non-compliance.
3668	CORE 13861 - EBP Dx Changes	CORE Change Order 8738 required several different diagnosis for coverage, billing and reimbursement rules. Not all were implemented and definitions of the roles were not clear. Definitions of the diagnosis types/roles: **Header Primary:** First position of the diagnosis at the header. **Header Secondary:** Second position of the diagnosis at the header. **Admit Diagnosis:** Diagnosis indicated as the Admit diagnosis. **Emergency Diagnosis:** Diagnosis indicated as the Emergency diagnosis. **Header Other:** Any diagnosis at the header that is not the Header Primary, Header Secondary, Admit or Emergency diagnosis. **Header Any:** Any of the Header, Secondary, Admit, Emergency, Other diagnosis. **Detail Primary:** Diagnosis pointed to from the first detail diagnosis pointer. **Detail Secondary:** Diagnosis pointed to from the second detail diagnosis pointer. **Detail Other:** Diagnosis pointed to from other than the first and second detail diagnosis pointers. **Detail Any** : Any of the diagnosis pointed to by the detail.

2.6.148 clmsub92IPPricing -- UB92 Inpatient Pricing

Technical Name:	clmsub92IPPricing
Program Title:	UB92 Inpatient Pricing
Programming Language:	C
Description:	UB92 inpatient pricing program. Main Inpatient program that contains routines to access DRG or per diem pricing methods for Inpatient claims. **Input Parameters:** Detail and header information related to member programs, performed procedures, revenue codes, and modifiers **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None (see clmsPricing) **Output Tables:** None This module is contained in the libclmub92.so library **Sort Criteria:** None **Switches:** None **Link Procedure:** None **Special Logic Notes:** None

2.6.148.1 Change Orders

ID	NAME	DESCRIPTION
1416	KY Prior Auth Pricing	Pricing using prior authorizations needs to be modified based on the manner in which the PAs are entered into the system and stored.
1476	Core 8816 Benefit Adj Factor 7	See Core CO 8816 all documentation will be updated on this CO Provide the infrastructure to allow modifications to all pricing methodologies that have a variance or incentive from the regular pricing methodology.
2298	KY Non DRG Inpatient Pricing	Claims submitted by psychiatric hospitals, as well as many inpatient services are not priced using the DRG pricing method. There are several different methods used based on the providers out of state status, disproportionate share status, and the age of the member at the time the services are rendered. Changes need to be made to inpatient pricing to perform these additional pricing methods, as well as to be able to identify when to use the various pricing methods.
2300	KY Pct of Charge Pricing Method	Some institutional claims are priced using a provider specific percentage of billed charges for specific revenue codes.
2302	KY UB92 Rev Code Rates	Some institutional claims are priced using a provider specific per unit rate for specific revenue codes.
9128	PAPCTB Pricing Incorrect	The PAPCTB pricing method is supposed to pay the provider contract rate on file times the amount billed on the claim (less any non-covered charges). Currently, this amount is being compared to $75,000 and paid at the lesser of the two amounts. This should not be part of the pricing method. This pricing method is not being performed when edit 4381 (no reimbursement rule) is posted.

2.6.149 clmsub92Icd9X -- UB92 ICD9 Procedures

Technical Name:	clmsub92Icd9X
Program Title:	UB92 ICD9 Procedures
Programming Language:	C
Description:	UB ICD9 procedure code processing. This module is responsible for holding UB claim header ICD9 procedure code information (like cdeIcd9, cdeQualifier, qtyAgeMin, etc.) and related functions to store (set_ui_xxxx where xxxx might be cdeIcd9) and retrieve (like get_ui_xxxx where xxxx might be numIcd9s) that information. **Input Parameters:** Detail and header information related to member programs, performed procedures, and qualifiers. **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** • T_PROC_ICD9 • T_PROC_ICD9_LIM **Output Tables:** None This module is contained in the libclmub92.so library **Sort Criteria:** None **Switches:** None **Link Procedure:** None **Special Logic Notes:**

	None

2.6.149.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.150 clmsub92LTCPricing -- UB92 Long Term Care Pricing

Technical Name:	clmsub92LTCPricing
Program Title:	UB92 Long Term Care Pricing
Programming Language:	C
Description:	UB long term care pricing This module is responsible for pricing Long Term Care claims. **Input Parameters:** Detail and header information related to member programs, dates of accommodation, and revenue codes associated with the accommodation. **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** • T_LTC_LV_RATE • T_PR_LOC_RATE • T_PR_RECP_LOC_X **Output Tables:** None This module is contained in the libclmub92.so library **Sort Criteria:** None **Switches:** None **Link Procedure:** None **Special Logic Notes:** None

2.6.150.1 Change Orders

ID	NAME	DESCRIPTION
1416	KY Prior Auth Pricing	Pricing using prior authorizations needs to be modified based on the manner in which the PAs are entered into the system and stored.
1476	Core 8816 Benefit Adj Factor 7	See Core CO 8816 all documentation will be updated on this CO Provide the infrastructure to allow modifications to all pricing methodologies that have a variance or incentive from the regular pricing methodology.
2300	KY Pct of Charge Pricing Method	Some institutional claims are priced using a provider specific percentage of billed charges for specific revenue codes.
2302	KY UB92 Rev Code Rates	Some institutional claims are priced using a provider specific per unit rate for specific revenue codes.
2429	KY Pct of Per Diem Pricing	Leave days submitted on nursing facility claims are paid at a provider specific percentage of the provider's per diem rate on file. There are also some inpatient claims that price using a percentage of the provider's per diem rate.

2.6.151 clmsub92OPPricing -- UB92 Outpatient Pricing

Technical Name:	clmsub92OPPricing
Program Title:	UB92 Outpatient Pricing
Programming Language:	C
Description:	UB outpatient pricing. This module is responsible for pricing an outpatient claim. Outpatient claims are priced by revenue code flat rate, ASC rate, and HCPCS max fee (procedure code * units on the claim). **Input Parameters:** Detail and header information related to member programs, performed procedures, and modifiers **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** • T_PROC_ASC **Output Tables:** None This module is contained in the libclmub92.so library **Sort Criteria:** None **Switches:** None **Link Procedure:** None **Special Logic Notes:** None

2.6.151.1 Change Orders

ID	NAME	DESCRIPTION
1476	Core 8816 Benefit Adj Factor 7	See Core CO 8816 all documentation will be updated on this CO Provide the infrastructure to allow modifications to all pricing methodologies that have a variance or incentive from the regular pricing methodology.
2277	KY Pricing - MAXFLT	In some current KY programs, some procedure codes (benefits) price on a flat fee (not multiplied by the units on the claim).
2300	KY Pct of Charge Pricing Method	Some institutional claims are priced using a provider specific percentage of billed charges for specific revenue codes.
2302	KY UB92 Rev Code Rates	Some institutional claims are priced using a provider specific per unit rate for specific revenue codes.
6936	OPASC Pricing Issues	Claims for critical access providers that contain a operating room revenue code (360) are inappropriately being cut back and zero paid. Additionally, dental codes billed with revenue code 360 should be excluded from all final pricing logic and be paid the OP percentage.
7641	PAYMENT FOR 360 REVENUE CODES	FOR REVENUE CODE 360 THE CLAIMS SHOULD ONLY PAY THE HIGHEST LINE ITEM 360 REVENUE CODE EXCEPT FOR DENTAL PROCEDURE CODES.
8031	835 balancing - PHYS claim	Balancing issue (835) with some Physician claims in the UAT environment.

2.6.152 clmsub92Occur -- UB92 Occurrence Codes

Technical Name:	clmsub92Occur
Program Title:	UB92 Occurrence Codes
Programming Language:	C
Description:	UB occurrence code processing. This module is responsible for holding UB claim occurrence code information and occurrence code related functions to store (set_uo_cdeOccurrence), retrieve (get_uo_spanOcc), or an edit (check for missing or invalid dates, etc). **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.152.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.153 clmsub92Payer -- UB92 Payer Information

Technical Name:	clmsub92Payer
Program Title:	UB92 Payer Information
Programming Language:	C
Description:	UB payer processing. This module is responsible for holding UB claim header payer information (like cdePayer, amtPriorPayment, etc.) and related functions. These related functions include routines to store (set_up_xxxx where xxxx might be cdePayer) and retrieve (get_up_xxxx where xxxx might be numPayer) payer information. **Input Parameters:** Header information related to payer identification and amount collected/due. **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None This module is contained in the libclmub92.so library **Sort Criteria:** None **Switches:** None **Link Procedure:** None **Special Logic Notes:** None

2.6.153.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.154 clmsub92Pricing -- UB92 Pricing

Technical Name:	clmsub92Pricing
Program Title:	UB92 Pricing
Programming Language:	C
Description:	UB general and miscellaneous pricing routines. **Input Parameters:** Detail and header information related to member programs, performed procedures, modifiers, revenue codes, and copayment, etc. **Exit Values:** None **Input Files:** None **Output Files:** None **Output Tables:** None This module is contained in the libclmub92.so library **Sort Criteria:** None **Switches:** None **Link Procedure:** None **Special Logic Notes:** None

2.6.154.1 Change Orders

ID	NAME	DESCRIPTION
1416	KY Prior Auth Pricing	Pricing using prior authorizations needs to be modified based on the manner in which the PAs are entered

ID	NAME	DESCRIPTION
		into the system and stored.
1476	Core 8816 Benefit Adj Factor 7	See Core CO 8816 all documentation will be updated on this CO Provide the infrastructure to allow modifications to all pricing methodologies that have a variance or incentive from the regular pricing methodology.
2298	KY Non DRG Inpatient Pricing	Claims submitted by psychiatric hospitals, as well as many inpatient services are not priced using the DRG pricing method. There are several different methods used based on the providers out of state status, disproportionate share status, and the age of the member at the time the services are rendered. Changes need to be made to inpatient pricing to perform these additional pricing methods, as well as to be able to identify when to use the various pricing methods.
2300	KY Pct of Charge Pricing Method	Some institutional claims are priced using a provider specific percentage of billed charges for specific revenue codes.
2302	KY UB92 Rev Code Rates	Some institutional claims are priced using a provider specific per unit rate for specific revenue codes.
2429	KY Pct of Per Diem Pricing	Leave days submitted on nursing facility claims are paid at a provider specific percentage of the provider's per diem rate on file. There are also some inpatient claims that price using a percentage of the provider's per diem rate.
3843	Copay Processing - CE	Deduct copay based on rule parameters when processing a claim.
6029	IP Clm Pays Incrrct Per Diem	ICN 2007015001962--paid the incorrect per diem--the claims paid 357.13 and it should have paid 324.66.
6936	OPASC Pricing Issues	Claims for critical access providers that contain a operating room revenue code (360) are inappropriately being cut back and zero paid. Additionally, dental codes billed with revenue code 360 should be excluded from all final pricing logic and be paid the OP percentage.
7641	PAYMENT FOR 360 REVENUE CODES	FOR REVENUE CODE 360 THE CLAIMS SHOULD ONLY PAY THE HIGHEST LINE ITEM 360 REVENUE CODE EXCEPT FOR DENTAL PROCEDURE CODES.
9110	Fix Prov Contract Rate Pricing	The PAPCTB pricing method is not including the claim type as part of the selection criteria from the provider

ID	NAME	DESCRIPTION
		contract rate table. As a result, the selection is returning multiple rows and causing claims to go to the bad file.
9128	PAPCTB Pricing Incorrect	The PAPCTB pricing method is supposed to pay the provider contract rate on file times the amount billed on the claim (less any non-covered charges). Currently, this amount is being compared to $75,000 and paid at the lesser of the two amounts. This should not be part of the pricing method. This pricing method is not being performed when edit 4381 (no reimbursement rule) is posted.

2.6.155 clmsub92Pricing_st -- KY UB92 Pricing Module

Technical Name:	clmsub92Pricing_st
Program Title:	KY UB92 Pricing Module
Programming Language:	C
Description:	Contains Kentucky specific call-out functions related to general UB claims pricing. **Input Parameters:** Detail and header information related to member program, service codes, modifiers, and so on. Used for KY specific UB claim pricing. **Exit Values:** • EXIT_SUCCESS - if found • EXIT_FAILURE - if not found **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** libclmub92_st **Special Logic Notes:** None

2.6.155.1 Change Orders

ID	NAME	DESCRIPTION
2300	KY Pct of Charge Pricing Method	Some institutional claims are priced using a provider specific percentage of billed charges for specific revenue codes.
2302	KY UB92 Rev Code Rates	Some institutional claims are priced using a provider specific per unit rate for specific revenue codes.
2429	KY Pct of Per Diem Pricing	Leave days submitted on nursing facility claims are paid at a provider specific percentage of the provider's per diem rate on file. There are also some inpatient claims that price using a percentage of the provider's per diem rate.
7940	Primary Care/RHC on ADA Form	Effective 2007/04/01, primary care centers (PT 31) and rural health clinics (PT35) are required to submit dental services on ADA dental claim forms. The system needs to be updated to accept, edit, audit, and price claims for these provider type on dental claim forms.

2.6.156 clmsub92Proc -- UB92 Procedures

Technical Name:	clmsub92Proc
Program Title:	UB92 Procedures
Programming Language:	C
Description:	UB procedure code processing. This module is responsible for performing edits on UB HCPC procedure codes. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.156.1 Change Orders

ID	NAME	DESCRIPTION
3668	CORE 13861 - EBP Dx Changes	CORE Change Order 8738 required several different diagnosis for coverage, billing and reimbursement rules. Not all were implemented and definitions of the roles were

ID	NAME	DESCRIPTION
		not clear. Definitions of the diagnosis types/roles: **Header Primary:** First position of the diagnosis at the header. **Header Secondary:** Second position of the diagnosis at the header. **Admit Diagnosis:** Diagnosis indicated as the Admit diagnosis. **Emergency Diagnosis:** Diagnosis indicated as the Emergency diagnosis. **Header Other:** Any diagnosis at the header that is not the Header Primary, Header Secondary, Admit or Emergency diagnosis. **Header Any:** Any of the Header, Secondary, Admit, Emergency, Other diagnosis. **Detail Primary:** Diagnosis pointed to from the first detail diagnosis pointer. **Detail Secondary:** Diagnosis pointed to from the second detail diagnosis pointer. **Detail Other:** Diagnosis pointed to from other than the first and second detail diagnosis pointers. **Detail Any** : Any of the diagnosis pointed to by the detail.

2.6.157 clmsub92TobGroup -- UB92 Type Of Bill Groups

Technical Name:	clmsub92TobGroup
Program Title:	UB92 Type Of Bill Groups
Programming Language:	C
Description:	UB type of bill group processing. This module is responsible for reading T_TOB_GROUP to find if given type of bill is in a group. Routines are named is_tg_xxxx where xxxx might be HHOutpatient which is hardcoded to be in sak type of bill group 1001. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** • T_TOB_GROUP **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.157.1 Change Orders

ID	NAME	DESCRIPTION

ID	NAME	DESCRIPTION
2458	Add KY Edit 123 - iC 3372	Add KY specific edit in interChange to match KY policy for existing edit 123 for claim may not span a member's 1st birthday.

2.6.158 clmsub92Value -- UB92 Value Codes

Technical Name:	clmsub92Value
Program Title:	UB92 Value Codes
Programming Language:	C
Description:	UB value code processing. This module is responsible for holding UB claim header value code/amount information (like cdeValue, amtValue, etc.), routines to store values (set_uv_xxxx where xxxx might be cdeValue) and retrieve values (get_uv_xxxx where xxxx might be numValue). **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** • T_VALUE **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.158.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.159 clmsub92Write -- UB92 Database Update

Technical Name:	clmsub92Write
Program Title:	UB92 Database Update
Programming Language:	C
Description:	UB write to tables. This module is responsible for inserting the UB claim into database. It uses a schema to convert the claim in XML format to a C structure and xsdDatabaseLoad to write to the data base. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** This program will always populate the paid 'C' structure regardless of what type of claim (paid, denied, or suspended) is being processed. The paid, denied, and suspended xsd schemas have been changed to be identical. The only difference is that some fields in the schemas might not have a db field attribute, so they should not be written to the db table.

2.6.159.1 Change Orders

ID	NAME	DESCRIPTION
1674	Core 10357 WI 2408 DRG 1	See Core CO 10357 all documentation will be updated on this CO See WI CO 2408 During the DRG grouping process the DRG grouper returns additional information that needs to be stored and displayed.
2104	CORE Dup audits - ub92 write	This is a duplicate of CORE CO #10686 Update claims history - write XOVER history data to new table T_MPHX_XOVER. This CO is a child of CORE CO#10604 CO#10604 is a child of KY CO#1326 See "KY Audit: " sections in Supplemental Documentation "KY_duplicate_reqs_ub92_dental.doc".
3530	Save Submitted Non Covered Amoun	In order to be HIPAA compliant, the non-covered amount submitted on each claim detail needs to be stored in a field separate from the non-covered amount which can be updated during claims processing, based on criteria established by the Commonwealth.
3843	Copay Processing - CE	Deduct copay based on rule parameters when processing a claim.
4035	Add indClaimCheck-UB92 Claims	Add the indClaimCheck field for UB claims.
8277	Claims Engine RTI	Task to be used for the promotion of claims engine changes that do not affect any of the outputs of claims processing and have no effect on the overall error status, payment or disposition of claims but are needed to inprove the throughput of claims, prevent claims from processing to the 'bad' files or to otherwise optimize the performance of claims processing.
8745	Referring provider ID	Referring provider ID numbers are not being moved from the input fields (party identifier) to the correct fields on the claim header/keys tables.
9239	Prov Ids Not on DB	The physician and dental rendering provider ids are not being stored in the claim header when claims are written to the data base tables. Additionally, the first and second other provider ids are not being written to the data base for UB claims. This needs to be corrected to store this data. Existing claims that are missing this data also need to be updated with a one time fix to history.

2.6.160 crlpd001 -- Creates Claim Link/Unlink Adjustments

Technical Name:	crlpd001
Program Title:	Creates Claim Link/Unlink Adjustments
Programming Language:	C
Description:	This module creates Claim Link/Unlink Adjustments This program reads link transactions from the T_RECIP_LINK_XREF table and unlink transactions from the T_RE_CLM_DELINK table and creates adjustments for them on the T_ADJ_MASS_RQST and T_ADJ_MASS_CLAIM tables using claim information from the T_HIST_DIRECTORY table. **Input Parameters:** None **Exit Values:** • EXIT_SUCCESS - Normal termination • EXIT_FAILURE - Abnormal termination because of open, read, allocation, or input errors **Input Files:** None **Output Files:** None **Input Tables:** • T_RECIP_LINK_XREF • T_RE_CLM_DELINK • T_HIST_DIRECTORY **Output Tables:** • T_RECIP_LINK_XREF • T_RE_CLM_DELINK • T_ADJ_MASS_RQST • T_ADJ_MASS_CLAIM

	Sort Criteria: • T_RECIP_LINK_XREF.SAK_LINK **Switches:** None at this time **Link procedures:** libxml, libadjxml, and libeutil **Special Logic Notes:** None

2.6.160.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.161 dummyClaimCheck -- dummyClaimCheck

Technical Name:	dummyClaimCheck
Program Title:	dummyClaimCheck
Programming Language:	C
Description:	This is a dummy function that is called only in unit testing when linking in claim check is not desired. **Input Parameters:** None **Exit Values:** None **Input Files:** None **Output Files:** None **Input Tables:** None **Output Tables:** None **Sort Criteria:** None **Switches:** None at this time. **Link Procedure:** None **Special Logic Notes:** None

2.6.161.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.162 encsplit -- Encounter Claims Splitter

Technical Name:	encsplit
Program Title:	Encounter Claims Splitter
Programming Language:	C
Description:	This module will split a large file of encounter claims into smaller files based on: • claim type • Member Id This module will create files of ~1000 encounter claims. The module is Memory Mapped. **Input Parameters:** • m - the maximum number of claims per file • i - the path to the input directory • o - the path to the output **Exit Values:** • EXIT_SUCCESS - normal termination • EXIT_FAILURE - abnormal termination due to open, read, allocation, or input errors **Input Files:** Files in the input directory **Output Files:** Files in the output directory **Input Tables:** • T_ENC_BATCH_PROCESS **Output Tables:** • T_ENC_BATCH_PROCESS **Sort Criteria:** None **Switches:** None at this time.

	Link procedure: hmspconn **Special Logic Notes:** None

2.6.162.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.163 fcst.sh -- Produce list of scheduled jobs

Technical Name:	fcst.sh
Program Title:	Produce list of scheduled jobs
Programming Language:	UNIX Script
Description:	Product List of Scheduled Jobs. fcst.sh is a UNIX script and not a program. No program documentation exist for this UNIX script.

2.6.163.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.164 libadjcomm.so -- Adjustment Common Library

Technical Name:	libadjcomm.so
Program Title:	Adjustment Common Library
Programming Language:	C
Description:	The executable that contains the logic for adjustments and voids. libadjcomm.so is an executable and not a program. No program documentation exist for this executable.

2.6.164.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.165 libadjxml.so -- Common Adjustment Functions

Technical Name:	libadjxml.so
Program Title:	Common Adjustment Functions
Programming Language:	C
Description:	This module controls initialization and adjustment request for the majority of the adjustment functions. libadjxml.so is an executable and not a program. No program documentation exist for this executable.

2.6.165.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.166 libclmbtch.so -- Tuxedo Interface

Technical Name:	libclmbtch.so
Program Title:	Tuxedo Interface
Programming Language:	C
Description:	These routines are used as the interface between the claims tuxedo services and the medpol tuxedo routines which will keep the medpol tables across multiple nodes in synch. libclmbtch.so is an executable and not a program. No program documentation exist for this executable.

2.6.166.1 Change Orders

ID	NAME	DESCRIPTION
No associated Change Orders found		

2.6.167 libclmcomm.so -- Claim Engine Common Library

Technical Name:	libclmcomm.so
Program Title:	Claim Engine Common Library
Programming Language:	C
Description:	The executable that contains the edit and audit functions which are common to all claim types, such as provider eligibility and member eligibility edits. The programs also interface with Claimcheck to audit claims. Claim disposition is also done in this library. libclmcomm.so is an executable and not a program. No program documentation exist for this executable.

2.6.167.1 Change Orders

ID	NAME	DESCRIPTION
1476	Core 8816 Benefit Adj Factor 7	See Core CO 8816 all documentation will be updated on this CO Provide the infrastructure to allow modifications to all pricing methodologies that have a variance or incentive from the regular pricing methodology.
1651	Core 8184 Add POS T_REIMB_AGREEM	Add the place of service (POS) & type of bill (TOB) parameter to the reimbursement rules table. This will allow support for pricing by the place of service.
1922	Core 9704 T_REF_UCC 3	See Core CO 9704 all documentation will be updated on this CO Add rate type to table t_ref_ucc to support the reimbursement rules assignment.
2136	Multiple Surgery - CE	Pricing method needed for multiple surgeries (same date of service).
2276	KY Pricing No proc found def %	Many KY programs operate in the following manner. If no procedure rate is found, they take the billed amount and multiple it by a percentage for the Medicaid reimbursement.
2277	KY Pricing - MAXFLT	In some current KY programs, some procedure codes (benefits) price on a flat fee (not multiplied by the units on the claim).
2291	KY DME Pricing PA Purchases	For KY, some DME purchases require prior

ID	NAME	DESCRIPTION
		authorization (a PA indicator is found on the procedure master file). In the current KY system, the rate for purchases that not rentals (RR modifier) are on the PDD file - MCAID-1. Rentals are also priced in the same manner as DME purchases that require PA. Rentals are identified by the use of the RR modifier. In the current KY system, the rate for rentals come from the PR (Procedure Rate) file (Outpatient - pricing specialty - 9).
2300	KY Pct of Charge Pricing Method	Some institutional claims are priced using a provider specific percentage of billed charges for specific revenue codes.
2302	KY UB92 Rev Code Rates	Some institutional claims are priced using a provider specific per unit rate for specific revenue codes.
2303	KY UB92 Single Unit Flat Rate	Some institutional claims are priced using a flat unit rate for specific revenue codes. Sometimes, the flat rate is all that will be paid for a service. At other times, the flat rate times the units of service on the detail will be paid.
2340	KY Pricing Support Comm Living	Price claims for the Supports for Community Living based on the provider, member and claim data. This is a new pricing methodology that is KY specific.
2446	KY UCC Flat Rate Pricing	Create new pricing methods to support the pricing methods currently used. This KY CO is duplicated in the Core PWB. See Core CO 12482 - all documentation will be updated on this CO.
2459	Add KY edit 122 - iC 3317	Add KY specific edit in interChange to match KY policy for existing edit 122 for services not approved by Medicare
2567	Add KY Edit 912 - iC 3331	Add KY specific edit in interChange to match KY policy for existing edit 912 where the Medicare paid amount greater than $250.00.
2606	Add Ky Edit 942 - iC 3334	Add KY specific edit in interChange to match KY policy for existing edit 942 where the revenue code 129 is not valid with any other accommodation revenue code.
2777	Add KY Edit 011 - iC 3345	Add KY specific edit in interChange to match KY policy for existing edit 011.
3017	KY PA Edits DME Bypass	When billed by a DME provider, procedure code E1399 should not fail prior authorization edits under

ID	NAME	DESCRIPTION
		the following conditions: - Detail dates of service (DOS) from 7/1/2002 to 1/2/2003 (inclusive) - The billed amount of the detail is less than or equal to $300.00 - There is no PA number on the claim
3668	CORE 13861 - EBP Dx Changes	CORE Change Order 8738 required several different diagnosis for coverage, billing and reimbursement rules. Not all were implemented and definitions of the roles were not clear. Definitions of the diagnosis types/roles: **Header Primary:** First position of the diagnosis at the header. **Header Secondary:** Second position of the diagnosis at the header. **Admit Diagnosis:** Diagnosis indicated as the Admit diagnosis. **Emergency Diagnosis:** Diagnosis indicated as the Emergency diagnosis. **Header Other:** Any diagnosis at the header that is not the Header Primary, Header Secondary, Admit or Emergency diagnosis. **Header Any:** Any of the Header, Secondary, Admit, Emergency, Other diagnosis. **Detail Primary:** Diagnosis pointed to from the first detail diagnosis pointer. **Detail Secondary:** Diagnosis pointed to from the second detail diagnosis pointer. **Detail Other:** Diagnosis pointed to from other than the first and second detail diagnosis pointers. **Detail Any** : Any of the diagnosis pointed to by the detail.
3843	Copay Processing - CE	Deduct copay based on rule parameters when processing a claim.
4360	UB92 Reimbursement Rule Priority	UB claims contain multiple types of services that may be used for pricing. Outpatient, Home Health, and LTC claims contain both procedure codes and revenue codes. In KY, the procedure code is never used for setting up reimbursement rules on these claim types. Inpatient claims use ICD9 procedure codes, DRG codes, and revenue codes. This is the order in which reimbursement rules should be evaluated and once one has been set, there is not need to continue looking for pricing methods on