

This release contains bug fixes, new modules, and enhancements to existing functionality. Beginning with release 8.8, we do not attach User Manual documents because of their size. There are three ways to receive updated document segments:

1. If your mailbox can receive large size documents (2MB or higher per document), you can request them directly via email through InSyst Support
2. We will make the documents available via our FTP site. **Please contact the Echo Oakland office for the account and password for your county.**
3. You can request the document be sent to you in CD format through InSyst Support.

4. The User's manual, the Reports manual and the CSI chapter from the Operations manual are on the FTP website. We apologize but the manuals may not be updated with new features or changes. The release notes will contain the most recent information on new features or changes.

Significant changes and new modules in this release:

1. Medi-Medi Exempt by selected "place of service" codes, taxonomy, Rehab services.
2. Medicare and Insurance billing exemption for "90 days no response".
3. Insurance claim by RU group

Critical tasks BEFORE/AFTER applying this release:

Before installing this release, you must have successfully applied release 9.1A.

Please retrieve new versions of needed Data Junction files from our FTP site or contact David Allen to receive them by email.

We have released new versions of the 837I and 837P maps for Medi-Cal as well as two new schema files necessary to properly process taxonomy codes and to handle the "90 days no response" situation. Note that file naming conventions for these maps have changed.

The new maps being released are named MediCal837I_DMH_VDJ1001A.djs, MediCal837P_ADJ_VDJ1001A.djs, and MediCal837P_DMH_VDJ1001A.djs.

The new (replacement) schemas are:

HIPAA_Health_Care_Claim_Institutional.X096.A1.X12.4010.837.1.1.ds.xml and
HIPAA_Health_Care_Claim_Professional.X098.A1.X12.4010.837.1.0.tds.ds.xml.

SCREEN CHANGES

Procedure maintenance screen – Added new field “Rehab Service”, a Y/N flag to indicate whether this 3-digit InSyst procedure code is for a Rehab service or not. When the flag is set to “Y”(Yes), services using this procedure code are qualified to be Medi-Medi exempt on Medi-Cal billing.

Additional usage notes:

1. Once the flag is set to “Y”, the effect will be retroactively applied to all Provider Balances as well as future Provider Balances created for this procedure.
 2. There is no need to add such a flag to PB. The flag in PB is for display only and is based on the value already set up via the Procedure maintenance screen.
-
-

Staff maintenance screen – Added two new taxonomy codes: “FFS” and “Medicare”. The original taxonomy is now labeled “Medi-Cal”. As of this release, although additional taxonomy codes were added to the screen, the service posting and Medi-Cal claiming still uses the original “TAXONOMY_CODE” field. The FFS and Medicare taxonomy code fields are for users to populate the values, but not used for any other purpose as of this release.

Claim maintenance Screen – For Void or “Implicit Void” claims, the screen has not been displaying the negative amount properly (users saw “#####” instead). This problem has been fixed.

Void and Replace Maintenance – Added a new rule so that the service being entered for “Replace with New” must be a Medi-Cal billable service (by checking the FRC).

Multi Service Entry Screen – A deadlock in the database may cause an erroneous message on the screen. The error handling has been corrected.

Medi-Cal Eligibility Screen - Changed validation of late reason code from a hard-coded list to table driven (SPECIAL_REASON_MASTER). We have also enhanced the SOC clearance process so that the clearance amount can not be greater than the cost of service. The screen will also take into account a prior SOC clearance amount on the same service. When two services are on the same day, after the first service’s cost of service has been exhausted for SOC purposes, the screen will locate the second service for SOC clearance.

Payment Entry Screen – Added new denial code “K” for “90 days no response”. If County staff wants to use this exempt reason, they should enter \$0 payment amount and “K” must be used as the denial code. The 837 will be automatically populated with State required data for this exemption. Notes: Users do not have to use this feature until the State has implemented this billing feature.

POSTING RELATED, BMENU, AND MISC. ITEMS

Medi-Cal

The following have been implemented in this release for CA Medi-Cal claim exemptions.

Medi-Medi and Medi-Ins exempts for “90 days no response” from OHC. To do this, pull up the claim in the payment entry screen. Enter zero dollar payment and a denial code “K”.

Medi-Medi exempts for:

- i) Place of service for Mobile, School, Telephone, and Community Services
- ii) Marriage Family Therapist (MFT) with taxonomy code 106H00000X
- iii) H2017 Rehabilitation Services

As of 9.2, the order of selecting a single Taxonomy Code (TC) in InSyst for Medi-Cal billing is as followed:

1. Send staff's TC if FFS (including the case of MFT); else
2. Send Staff's TC if MFT (medi-medi exempt); else
3. Send crosswalk's TC if not blank; else
4. Send staff's TC if crosswalk's TC is blank; else
5. Send <blank> TC if staff's TC is invalid (This will cause a claim to be denied.)

Medicare

Medicare HCFA/CMS submitter information now comes from the ANSI_STRUCTURE table. It was previously populated and used data from fields in the COUNTY_MASTER table. This standardizes the setup in one place and supports the group functionality of multiple submitter IDs.

Fixed Taxonomy error in DJ map

Insurance

Insurance claim by RU group is available in this release. The set up and usage of this feature is very similar to Medi-Cal/Medicare claim by RU group. Please see separate document on how to use this feature.

General

Service Posting has been enhanced so that, going forward, Medi-Cal services that are exempted by HCPC codes will be grouped under one claim form while the non-exempt services are grouped under another. Technically, that means the County will be able to suppress non-exempt claims more precisely while releasing the exempt claims.

New DJ Maps for Medi-Cal 835 processing (MediCal835_ADP_V09001E.djs and MediCal835_DMH_V09001E.djs) were released late in the InSyst 9.1 cycle (October 29,

2010). These maps fixed an issue where a wrong CIN number was loaded in the Pseudo EOB file due to the fact that the State skips certain data segments, including the CIN, when a claim is denied. If you have not already updated to these maps, they are available from the FTP site or by email from David Allen.

REPORTS

MH1980 CSV file output – New fields are added to this output. They are: GUID, Reporting Unit, Void/Replace flag (indicating Original claim, Void, Replace, or Implicit Void), and Primary Aid Code

STATE REPORTS

None

DATABASE Changes

*Note that the source for all metadata changes is recorded in files named **\$metadata*.*** in the **release directory**. If you have a data warehouse that requires the technical details of metadata changes you can review these files for the information.*

```
drop trigger INSURANCE_CLAIM_group_ru_ins;
drop trigger INSURANCE_CLAIM_group_ru_upd;
drop trigger INSURANCE_CLAIM_group_ru_del;
drop trigger INSURANCE_CLAIM_group_upd;
drop trigger INSURANCE_CLAIM_group_del;
drop table INSURANCE_CLAIM_group_ru;
drop table INSURANCE_CLAIM_group;
```

```
create table INSURANCE_CLAIM_group (
    group_label          claim_group_label    constraint
INSG_NOT_NULL_GROUP_LABEL not null not deferrable,
    group_description    claim_group_description constraint
INSG_NOT_NULL_GROUP_DESCRIPTION not null not deferrable,
    claim_type           claim_group_type     constraint
INSG_NOT_NULL_CLAIM_TYPE not null not deferrable,
    record_stamp         record_stamp
);
```

```
create unique index ins_claim_group_pk on INSURANCE_CLAIM_group (group_label);
create unique index ins_claim_group_stamp_index on INSURANCE_CLAIM_group
(record_stamp);
```

```
create table INSURANCE_CLAIM_group_ru (
```

```

        group_label          claim_group_label  constraint
INSGR_NOT_NULL_GROUP_LABEL not null not deferrable,
        reporting_unit      reporting_unit     constraint INSGR_NOT_NULL_RU
not null not deferrable,
        record_stamp        record_stamp,
        effective_yyyyymm   effective_yyyyymm  constraint
INSGR_NOT_NULL_EFFECTIVE_YYYYMM not null not deferrable,
        expiration_yyyyymm  expiration_yyyyymm default 0
);

create unique index ins_claim_group_ru_pk on INSURANCE_CLAIM_group_ru
(group_label, reporting_unit, effective_yyyyymm);
create unique index ins_claim_group_ru_stamp_index on INSURANCE_CLAIM_group_ru
(record_stamp);

create trigger INSURANCE_CLAIM_group_upd
after update of group_label on INSURANCE_CLAIM_group
referencing old as old1
when exists ( select *
              from INSURANCE_CLAIM_group_ru
              where group_label = old1.group_label )
(error)
for each row
;

create trigger INSURANCE_CLAIM_group_del
before delete on INSURANCE_CLAIM_group
when exists ( select *
              from INSURANCE_CLAIM_group_ru
              where group_label = INSURANCE_CLAIM_group.group_label )
(error)
for each row
;

insert into INSURANCE_CLAIM_group values ('NORM_INS','Normal
Insurance','INSURANCE',current_timestamp);

create trigger INSURANCE_CLAIM_group_ru_ins
after insert on INSURANCE_CLAIM_group_ru
when exists (
select *
from INSURANCE_CLAIM_group_ru mcg2
where mcg2.reporting_unit =
INSURANCE_CLAIM_group_ru.reporting_unit
and mcg2.record_stamp <>INSURANCE_CLAIM_group_ru.record_stamp
and ( mcg2.effective_yyyyymm
<=INSURANCE_CLAIM_group_ru.expiration_yyyyymm
or INSURANCE_CLAIM_group_ru.expiration_yyyyymm is null
or INSURANCE_CLAIM_group_ru.expiration_yyyyymm = 0 )
and ( mcg2.expiration_yyyyymm
>=INSURANCE_CLAIM_group_ru.effective_yyyyymm
or mcg2.expiration_yyyyymm is null
or mcg2.expiration_yyyyymm = 0 )
) or
( INSURANCE_CLAIM_group_ru.expiration_yyyyymm <> 0
and INSURANCE_CLAIM_group_ru.expiration_yyyyymm <
INSURANCE_CLAIM_group_ru.effective_yyyyymm

```

```

) or
  not exists (
  select *
  from INSURANCE_CLAIM_group mcg3
  where mcg3.group_label =
INSURANCE_CLAIM_group_ru.group_label
  )
  (error)
  for each row;

create trigger INSURANCE_CLAIM_group_ru_upd
  after update on INSURANCE_CLAIM_group_ru
  referencing old as old1 new as new1
  when exists (
  select *
  from INSURANCE_CLAIM_group_ru mcg4
  where mcg4.reporting_unit = new1.reporting_unit
    and mcg4.record_stamp <> new1.record_stamp
    and ( mcg4.effective_yyyyymm <= new1.expiration_yyyyymm
      or new1.expiration_yyyyymm is null
      or new1.expiration_yyyyymm = 0 )
    and ( mcg4.expiration_yyyyymm >= new1.effective_yyyyymm
      or mcg4.expiration_yyyyymm is null
      or mcg4.expiration_yyyyymm = 0 )
  ) or
  ( new1.expiration_yyyyymm <> 0
    and new1.expiration_yyyyymm < new1.effective_yyyyymm
  ) or
  not exists (
  select *
  from INSURANCE_CLAIM_group mcg5
  where mcg5.group_label = new1.group_label
  )
  (error)
  for each row;

show table INSURANCE_CLAIM_group;
show table INSURANCE_CLAIM_group_ru;

create domain REHAB_SERVICE_FLAG CHAR(1) query header is 'Rehab' /
'Service';
alter table PROCEDURES add column REHAB_SERVICE_FLAG REHAB_SERVICE_FLAG;

show domain REHAB_SERVICE_FLAG;
show table PROCEDURES;

alter domain TAXONOMY_CODE query header is 'TAXONOMY';

alter table STAFF_MASTER alter column TAXONOMY_CODE query header is
'TAXONOMY';
alter table STAFF_MASTER add column TAXONOMY_CODE_FFS TAXONOMY_CODE query
header is 'FFS'/'TAXONOMY';
alter table STAFF_MASTER add column TAXONOMY_CODE_MEDICARE TAXONOMY_CODE
query header is 'Medicare'/'TAXONOMY';

create index VR_RECORD_STAMP_INDEX on VOID_REPLACE (RECORD_STAMP)

```

```

        type is SORTED node size 1052 percent fill 90 disable
compression store in VOID_REPLACE_IDX (THRESHOLDS ARE ( 82, 97, 97 ));
create index    VR_RECORD_STAMP_INDEX          on VOID_REPLACE (RECORD_STAMP)
        type is SORTED node size 1052 percent fill 90 disable
compression store in SMALL_INDEX_AREA (THRESHOLDS ARE ( 82, 97, 97 ));
create index    VR_RECORD_STAMP_INDEX          on VOID_REPLACE (RECORD_STAMP) type is
SORTED disable compression;

-- CLAIM_LINE_STAMP may have been originally intended to store data from original
claim_line, but was never populated
drop    index    VR_CLAIM_LINE_INDEX;
alter   table    VOID_REPLACE                drop column CLAIM_LINE_STAMP;
alter   domain   VCR_STATUS                  edit string is 'ZZ9';

-- VOID_REPLACE_CLAIM_LINE_STAMP will store either explicit-VOID or REPLACE
claim_line_stamp
-- VOID_CLAIM_LINE_STAMP will store either VOID or Implicit-VOID's
claim_line_stamp, so for a VOID both VOID_REPLACE_CLAIM_LINE_STAMP &
VOID_CLAIM_LINE_STAMP will be the same

alter   table    VOID_REPLACE                add column VOID_CLAIM_LINE_STAMP
CLAIM_LINE_STAMP          query header 'Void/Implicit-Void' / 'Claim_Line Stamp';

create index    VR_VOID_CLAIM_LINE_INDEX      on VOID_REPLACE
(VOID_CLAIM_LINE_STAMP)
        type is SORTED node size 1052 percent fill 90 disable compression
store in VOID_REPLACE_IDX (THRESHOLDS ARE ( 82, 97, 97 ));
create index    VR_VOID_CLAIM_LINE_INDEX      on VOID_REPLACE
(VOID_CLAIM_LINE_STAMP)
        type is SORTED node size 1052 percent fill 90 disable compression
store in SMALL_INDEX_AREA (THRESHOLDS ARE ( 82, 97, 97 ));

create index    VR_VOID_CLAIM_LINE_INDEX      on VOID_REPLACE
(VOID_CLAIM_LINE_STAMP) type is SORTED disable compression;

drop    index    VR_CSI_INDEX;
create index    VR_CSI_INDEX                  on VOID_REPLACE (LAST_CSI_REPORTING_DATE,
VCR_ACTION, SERVICE_STAMP, VCR_STATUS) type is SORTED percent fill 90 disable
compression store in VOID_REPLACE_IDX (THRESHOLDS ARE (80,97,97 ));

create index    VR_CSI_INDEX                  on VOID_REPLACE (LAST_CSI_REPORTING_DATE,
VCR_ACTION, SERVICE_STAMP, VCR_STATUS) type is SORTED percent fill 90 disable
compression store in SMALL_INDEX_AREA (THRESHOLDS ARE (80,97,97 ));

create index    VR_CSI_INDEX                  on VOID_REPLACE (LAST_CSI_REPORTING_DATE,
VCR_ACTION, SERVICE_STAMP, VCR_STATUS) type is SORTED percent fill 90 disable
compression;

```