

「日医標準レセプトソフト」

ORCA Project

システムパッケージリリース情報 【Version 4.6.0】

2011 年 5 月 24 日

社団法人日本医師会

1 パッケージアップグレード

1-1 注意事項

(1) バックアップについて

このパッケージへアップグレードするとデータベースの変更が行われます。

アップグレードを行う前には必ずデータベースのバックアップを行い、エラーが発生していないことを確認してください。

アップグレード中にエラーが発生した場合に、アップグレード前のバックアップが正常に行われていない場合は修復不可能となります。

また、グループ診療のためデータベース変更では、システム管理に記録されている医療機関 ID（アップグレード処理日に有効なもの）と異なる医療機関 ID を持つ行を無条件に削除します。よって、アップグレードは正常に終了しても、システム管理に想定を超えた内容の登録があった場合、すべて行削除される可能性もあります。この場合もバックアップが正常に行われていない場合は修復不可能となります。

このような状態にあっては、サポートセンターへ問い合わせをされても、サポートの対象とは認めませんので悪しからずご了承ください。

(2) 処理時間について

パッケージアップグレードによりテーブルの構造変更を行います。

変更を行うテーブル数も多いうえに対象のテーブルに格納されている行数が多い場合は、マシンの性能（スペック）にもよりますがかなり処理時間がかかると予想されますのでバージョンアップの計画を立てて行うようにしてください。

お昼の休憩時間等にパッケージアップグレードを行うということは避けていただいた方が無難です。業務に支障をきたさない時間帯に余裕をもって行ってください。

(3) バージョン 4.5.0 以前からアップグレードする場合について

各バージョンのシステムパッケージリリース情報も併せて参照してください。

本資料では、バージョン 4.6.0 における情報に限り説明を行っています。

1-2 パッケージアップグレードの方法

ここでは、アップグレードを対象に説明をします。クリーンインストールする場合は、公式ホームページの「日医標準レセプトソフトのダウンロード・インストール」をご参照ください。

(1) データベースのバックアップをします。

パッケージをアップグレードする場合は、操作に先立って必ずデータベースのバックアップを行ってください。

バックアップの方法

ターミナルプログラム(gnome-terminal, kterm など)から以下のコマンドを入力します。

```
$ sudo -u orca pg_dump -0 orca > バックアップファイル名
```

圧縮する場合

```
$ sudo -u orca pg_dump -0 orca | gzip > バックアップファイル名.gz
```

(2) スキーマチェックを行います。

現在のデータベースの状態をチェックします。

最新のチェックファイルによりチェック処理を行うため以下の方法により処理を行います。

ターミナルプログラム(gnome-terminal, kterm など)から以下のコマンドを入力します。

```
$ wget http://ftp.orca.med.or.jp/pub/etc/jma-receipt-dbscmchk.tgz
```

```
$ tar xvfz jma-receipt-dbscmchk.tgz
```

```
$ cd jma-receipt-dbscmchk
```

```
$ sudo bash jma-receipt-dbscmchk.sh
```

処理が終了したらメッセージが表示されますが、不整合が発生した場合、作業ディレクトリ

(jma-receipt-dbscmchk/)に jma-receipt-dbscmchk.log というファイルが作成されます。

ログファイルの内容を確認してください。

整合性に問題なしと出ればスキーマは問題ないです。

(3) apt の確認をします。

[lucid の場合]

/etc/apt/sources.list.d/jma-receipt-lucid(nn).list のエントリに以下の記述があるか確認をしてください。なければ追加あるいは変更してください。

```
deb http://ftp.orca.med.or.jp/pub/ubuntu lucid4.6 jma
```

[hardy の場合]

/etc/apt/sources.list.d/jma-receipt-hardy(nn).list のエントリに以下の記述があるか確認をしてください。なければ追加あるいは変更してください。

```
deb http://ftp.orca.med.or.jp/pub/ubuntu hardy4.6 jma
```

(4) パッケージをダウンロードします。

ターミナルプログラム(gnome-terminal, kterm など)から以下のコマンドを入力します。

```
$ sudo aptitude update  
$ sudo aptitude dist-upgrade -dy
```

ダウンロード中に中断されたなどエラーが発生していないことを確認します。エラーが発生し終了した場合は再度ダウンロードを行います。

(5) アップグレードします。

ターミナルプログラム(gnome-terminal, kterm など)から以下のコマンドを入力します。

```
$ sudo aptitude dist-upgrade
```

再起動について

設定によりますが、「日レセ」パッケージのアップグレードをすると、アプリケーションは自動で再起動します。ですが、サブプロセスが終了できずにそのまま残る場合があります。

パッケージアップグレード後、再起動をしていただくときれいな状態でアプリケーションが起動できます。

(6) プログラム更新処理をします。

パッケージをアップグレードした後は、プログラムの緊急修正情報があるか確認を行ってください。

緊急修正情報がある場合は、必ず業務メニュー画面より「プログラム更新」を行ってください。

1-3 データベース構造変更の確認について

業務メニュー画面の「92 マスタ更新」を選択するとデータベースの構造バージョンを確認できます。

正常にアップグレード(インストール)が完了すると「構造 ver(自)」及び「構造 ver(ORCA)」の表示は次のようになります。

「S-040600-1-20110522-1」 平成 23 年 5 月 24 日現在

また、/var/log/jma-receipt/orca-db-install-4.6.0.log というファイルが作成されますので “ERROR” という文字が含まれていないか確認します。

```
$ grep ERROR /var/log/jma-receipt/orca-db-install-4.6.0.log
```

さらに、スキーマチェックを行います。

```
$ cd jma-receipt-dbscmchk  
$ sudo bash jma-receipt-dbscmchk.sh
```

不整合が発生していないことを確認します。

1-4 マスタ更新データの提供について

パッケージをアップグレードしただけでは各種マスタは更新されません。必ず業務メニュー画面より「92 マスタ更新」を行ってください。

業務メニュー画面の「92 マスタ更新」を選択しマスタ更新管理一覧画面より「更新」ボタン(F12 キー)を選択すると更新処理を開始します。

提供されているマスタ更新データの最新情報は公式ホームページあるいは業務メニュー画面の新着情報をご確認ください。

マスタ更新処理プログラムは、`/var/lib/jma-receipt/master/01/orca-mstlog/` にログを書き出します。拡張子が ERR というファイルが空(0 バイト)でない場合は、対となるファイル名の拡張子 PRF というファイルの内容を確認します。(`/var/lib/jma-receipt/master/01/orca-mstlog/` の 01 は医療機関識別番号です。)

2 Version4.5.0 に対する修正プログラム提供内容

Version4.5.0 システムに適用させるため提供した修正プログラムについては、すべて当該バージョンに含まれています。提供したプログラムの修正内容については、公式ホームページあるいは業務メニュー画面の新着情報をご確認ください。

3 Version4.6.0 での留意すべき変更内容

3-1 OpenCOBOL 1.0 対応

OpenCOBOL (COBOL コンパイラ) のバージョン 1.0 を採用しています。

バージョン 1.0 に対応するため、現時点で判明している変更点などについて説明します。

3-1-1 OpenCOBOL パッケージ

(1) 現時点でのバージョンは以下となります。

Lucid の場合

```
open-cobol 1.0-u2jma6
libcob1-dev 1.0-u2jma6
libcob1 1.0-u2jma6
```

Hardy の場合

```
open-cobol 1.0-u1jma6
libcob1-dev 1.0-u1jma6
libcob1 1.0-u1jma6
```

Lucid では、open-cobol(1.0-4)パッケージが提供されていますが、ORCA プロジェクトから提供している上記バージョンのパッケージを使用してください。

(2) パッチ

ORCA プロジェクトから提供する open-cobol は、以下のパッチを適用しています。

- ・ COPY 文の REPLACING 指定で " //仮原文-1// BY //仮原文-2// " が使用できるようにした
- ・ 最適化オプションは、指定しない場合に -O0 とするようにした
- ・ ファイル節 (FILE SECTION) のレコード記述項で定義されたレコード領域に対して INITIALIZE 命令が処理されるようにした

3-1-2 0.8.9 から 1.0 の変更点

1.0 は標準的な COBOL 規約の動作となります。

0.8.9 は標準的な COBOL 規約とは異なる動作をするものがあり、0.8.9 で正しく動作していたプログラムでも 1.0 では異なる動作となることがあるため、プログラムを修正する必要が発生する場合があります。

(1) コンパイル時のチェックがより厳格となりました。

(2) 連絡節 (LINKAGE SECTION) の連絡レコードのレコード長は、呼び出し側プログラムの CALL 文の USING 句で指定した領域の長さと合わせる必要があります。

連絡節の連絡レコードのレコード長が CALL 文の USING 句で指定した領域の長さより大きい場合は、呼び出し側プログラムへ復帰した時に、呼び出し側プログラムのどこかの領域を超過した部分の内容で上書きしてしまいます。

0.8.9 では、呼び出し側プログラムの USING 句で指定した領域部分について上書きされていた。

(3) INITIALIZE 文の動作が変更となりました。

0.8.9 の INITIALIZE 文では、FILLER 句 (FILLER を記述しない場合も含む) で指定したデータ領域も初期化の対象としていました。

1.0 の INITIALIZE 文では、FILLER 句 (FILLER を記述しない場合も含む) で指定したデータ領域は初期化の対象となりません。

- (4) OPEN 文の EXTEND 指定での動作が変更となりました。
0.8.9 では、指定したファイルが存在しない場合、ファイルを作成していました。
1.0 では、ファイルを作成せずエラーとなります。
- (5) 索引ファイルに対してファイルアクセス句 (ACCESS MODE) に順呼び出し (SEQUENTIAL) を指定した場合のレコード出力時の動作が変更となりました。
0.8.9 では、RECORD KEY に指定した項目の内容に関係なく出力できていました。
1.0 では、RECORD KEY に指定した項目の内容が昇順となるように出力しなければエラーとなります。
- (6) COPY 文の REPLACING の動作が変更となりました。
CPBUNBEN.INC に 03 BUNBEN-BUNBEN-MONEY と定義があり
COPY "CPBUNBEN.INC" REPLACNG //BUNBEN-// BY //WBUNBEN-//. とした場合
0.8.9 では、03 WBUNBEN-WBUNBEN-MONEY と変換されます。
1.0 では、03 WBUNBEN-BUNBEN-MONEY と変換されます。
COPY 文を
COPY "CPBUNBEN.INC" REPLACING == BUNBEN-== BY ==WBUNBEN-==. とすると、いずれのコンパイラでも 03 WBUNBEN-BUNBEN-MONEY と変換され正常コンパイルができました。
- (7) Shift_JIS コードのファイルを扱う場合は注意が必要です。
ファイルの改行コードが CR(0x0D)+LF(0x0A) の場合で
ORGANIZATION IS LINE SEQUENTIAL
と定義してレコードを読み込んだ場合
0.8.9 では、CR(0x0D)が残って読み込まれます。
1.0 では、改行コードは消去された内容で読み込まれます。

3-1-3 コンパイルオプション

コンパイル時に指定するオプションは以下となります。COPY 句ライブラリの指定は-I で指定します。

```
-m -dynamic -fixed -fimplicit-init -std=cobol85 -Wall
```

この内容は、日レセのインストールにより作成される

/etc/jma-receipt/jma-receipt.env

ファイル内に記述されている環境変数 COBOLFLAGS で確認できます。

site-upgrade.sh スクリプトでは、この環境変数を使用してコンパイルしています。

3-2 データベーステーブルスキーマ変更

3-2-1 変更テーブル一覧

住所	TBL_ADRS	項目追加、データ型変更、桁数変更
分娩管理	TBL_BUNBEN	項目追加、データ型変更、桁数変更
病名	TBL_BYOMEI	項目追加、データ型変更、項目名変更
オンライン返戻明細	TBL_HENREI_BODY	項目追加
オンライン返戻管理	TBL_HENREI_KANRI	項目追加
保険組合せ	TBL_HKNCOMBI	データ型変更
入力セット	TBL_INPUTSET	項目追加
受診履歴	TBL_JYURRK	データ型変更
受診履歴（プレビュー）	TBL_JYURRK_PRV	データ型変更
地方公費請求書	TBL_KOHSKY	項目追加
入院履歴照会	TBL_NRRKSRH	データ型変更
入院診療行為照会	TBL_NSRYSRH	項目追加、データ型変更
オンラインレセ電明細	TBL_ONRECE_BODY	項目追加
患者病名	TBL_PTBYOMEI	項目追加、データ型変更
患者コメント	TBL_PTCOM	項目追加、データ型変更
患者コメント（プレビュー）	TBL_PTCOM_PRV	項目追加、データ型変更
患者保険情報	TBL_PTHKNINF	データ型変更
患者情報	TBL_PTINF	データ型変更
患者情報（プレビュー）	TBL_PTINF_PRV	データ型変更
患者公費情報	TBL_PTKOHINF	データ型変更
患者入院履歴	TBL_PTYUINRRK	データ型変更
患者情報照会	TBL_PTSRH	項目追加
レセプトコメント	TBL_RECECOM	項目追加、データ型変更、桁数変更
レセプト電算	TBL_RECEDEN	項目追加
受診履歴照会	TBL_RRKS RH	データ型変更
請求管理他	TBL_SEIKYU_ETC	項目追加
請求管理公費	TBL_SEIKYU_KOH	項目追加
請求管理基本	TBL_SEIKYU_MAIN	項目追加
診療行為照会	TBL_SRYSRH	項目追加
主科	TBL_SYUKA	項目追加
収納明細	TBL_SYUMEI	データ型変更
収納明細（プレビュー）	TBL_SYUMEI_PRV	データ型変更
収納基本	TBL_SYUNOU_MAIN	項目追加
収納（プレビュー）	TBL_SYUNOU_PRV	項目追加、データ型変更
収納合計	TBL_SYUTOTAL	項目追加、データ型変更
収納合計（プレビュー）	TBL_SYUTOTAL_PRV	項目追加、データ型変更
点数	TBL_TENSU	データ型変更
点数付加	TBL_TENSUPLUS	項目追加、データ型変更
年金履歴	TBL_TNKRRK	データ型変更
低所得者履歴	TBL_TSYRRK	データ型変更
ワーク診療行為	TBL_WKSRYACT	項目追加、桁数変更

3-2-2 追加テーブル一覧

電子点数表 1	TBL_ETENSU_1
電子点数表 2	TBL_ETENSU_2
電子点数表 2 追加	TBL_ETENSU_2_jma
電子点数表 2 除外	TBL_ETENSU_2_off
電子点数表 2 検体	TBL_ETENSU_2_sample
電子点数表 3 - 1	TBL_ETENSU_3_1
電子点数表 3 - 2	TBL_ETENSU_3_2
電子点数表 3 - 3	TBL_ETENSU_3_3
電子点数表 3 - 4	TBL_ETENSU_3_4
電子点数表 4	TBL_ETENSU_4
電子点数表 5	TBL_ETENSU_5
感染症判定	TBL_INFECTION_JUDGMENT
請求管理照会	TBL_SKYSRH
同義語	TBL_SYNONYM

3-2-3 テーブルスキーマ変更における周辺システムとの連携について

カラムの追加やカラムのデータ型を変更しているテーブルがいくつかあります。

日レセと連携を行う周辺システムをご使用の場合は、アップグレードする前に十分に動作確認等を行ってください。

特に ODBC 経由等でデータベースへアクセスされている場合はご注意ください。

ワーク診療行為テーブルを変更しています。周辺システム連携では、最も使用されるテーブルの一つですので特にご注意ください。

3-3 補足コメントの追加による傷病名の記載について

患者病名テーブル(tbl_ptbyomei)に補足コメントの項目を追加しました。

前バージョンまでは傷病名の一部として入力していた補足コメントですが、別項目として入力することが可能となりました。

これにより、傷病名の記載を必要とする帳票等について、補足コメントを含めた傷病名で印字する必要があるものについては、別項目として記録された傷病名と補足コメントを「傷病名+「(」+補足コメント+「)」」と編集する必要があります。

本体では、補足コメントを含めた内容で印字する帳票については、傷病名編集関数「ORCSMKBYOMEI」を使用して編集しています。(レセプト、会計カード、カルテ)

カスタマイズされている帳票等で傷病名の記載で補足コメントが必要な場合は修正を行ってください。

(使用例)

* サブプロ用領域

*

COPY "CPORCSMKBYOMEI.INC".

```
INITIALIZE                                ORCSMKBYOMEIAREA
MOVE   PTBYO-UTAGAI FLG                 TO   SMKBYO-UTAGAI FLG
MOVE   PTBYO-BYOMEI                     TO   SMKBYO-BYOMEI
MOVE   PTBYO-HOSOKU-COMMENT              TO   SMKBYO-HOSOKU-COMMENT
CALL   "ORCSMKBYOMEI"                    USING ORCSMKBYOMEIAREA
MOVE   SMKBYO-BYOMEI                     TO   ○ ○
```

3-4 使用可能なクライアントについて

バージョン 4.6.0 に対し使用可能なクライアントは、バージョン 1.4.6 以降の glclient、glclient2 及び monsiaj ver 20110304 以降のみです。

バージョン 4.5.0 以前に対し使用していた glclient、glclient2 及び monsiaj ver 20110106 以前は接続できません。

3-5 請求管理テーブルへの項目追加について

請求管理テーブル(tbl_seiky_main, tbl_seiky_etc, tbl_seiky_koh)のプライマリキーに新設カラムの公費 ID(kohid_key)を追加しました。

精神入院(20)、児童保護(53)、心神喪失(30)については公費単独レセプトとなりますが、他の公費と適用開始日が同一の場合にそれらを識別する情報が不足していました。

今回それを識別するための情報を追加し、公費毎の単独レセプト作成を可能としました。

(例:精神入院 20 と生保 12)

精神入院(20)、児童保護(53)、心神喪失(30)の公費単独レセプトの場合のみ、公費 ID(kohid_key)に該当する公費の kohid を設定します。(デフォルトは 0)

併せて、公費請求テーブル(tbl_kohsky)、レセ電テーブル(tbl_receden)についても同様に公費 ID(kohid_key)を追加しました。

このため上記の各テーブルを使用する帳票等について、参照方法の変更が必要となる場合がありますのでご注意ください。

3-6 収納基本テーブルへの項目追加について

収納基本テーブル(tbl_syunou_main)に保険適用区分(HKNTEKKBN)を追加しました。

保険適用区分：医療保険の患者負担額を項目「公費一部負担金」に収容している場合は"1"を設定する。
それ以外の場合は空白を設定する。

主保険＋長期の組合せの請求書兼領収書について、現行「公費一部負担金」欄に表示している患者の医療費負担額を「保険分負担金額」欄に表示するように変更しました。

カスタマイズされている帳票等で同様の変更が必要な場合は修正を行ってください。

3-7 労災保険、自賠責保険でのユーザーコードの金額入力対応について

診療種別区分が“ 8 0 9 その他(労災・自賠責入力) ”の剤の自費金額は、診療行為テーブル(tbl_sryact) の自費金額(SRY- JIHIMONEY) に登録されています。

労災保険、自賠責保険でその他項目の金額をユーザーコード毎に編集している帳票等では修正が必要となります。

入院の診療費明細書、カルテ 3 号紙(入院・外来)、会計カード

外来の診療費明細書はサブルーチンプログラム(ORCHC04SUB)で対応をしていますので、これを使用している場合は修正の必要はありません。使用していない場合は修正する必要がありますのでご注意ください。

3-8 レセプト病名記載順対応について

レセプト病名記載順(転帰分による記載の設定)対応で、レセプトの病名の記載順の設定が可能となりました。

レセプト点検などでレセプトと照らし合せて確認をするような帳票等については、レセプトと同じ記載順にする方が便利と思われます。記載順を同様にする場合は修正を行ってください。

3-9 お薬手帳カスタマイズ用パラメーターの変更について

お薬手帳印刷(ORCHC62)のパラメーターに伝票番号を追加しました。

診療行為業務の請求確認画面から出力した場合と、前回処方画面の再印刷ボタン(F9)から出力した場合に該当する伝票番号をパラメーターとして編集します。

*伝票番号(主の伝票番号)

03 ORCHC62-DENPNUM PIC 9(07).

主たる伝票番号を設定する。処方内容の対象でなくても編集を行う。

*伝票番号(処方内容の対象伝票番号)

03 ORCHC62-GRP-DENPNUM-G.

05 ORCHC62-GRP-DENPNUM PIC 9(07) OCCURS 10.

処方内容の対象となった伝票番号を編集する。

診療科(ドクター)が1つの時は、ORCHC62-DENPNUM と ORCHC62-GRP-DENPNUM(1)は同じ伝票番号となります。

3-10 ユーザーコメント入力数値小数部対応について

処方せん等に記載する服用方法に数量を編集可能としました。

また、数量編集可能なユーザコメントについても入力数量に小数部も編集可能としました。

(1) 用法コード(001xxxxxx)に“挿入位置”と“コメント区分”の項目を追加しました。

挿入位置

用法名(漢字名称)にコメント(数量)を挿入する位置(文字桁位置)を設定します。

挿入するコメント(数量)は最大5つまで設定可能となります。

点数マスタの施設基準コード6～10の項目を流用し設定値を管理します。

施設基準コード6 : 1番目のコメント挿入位置

施設基準コード7 : 2番目のコメント挿入位置

施設基準コード8 : 3番目のコメント挿入位置

施設基準コード9 : 4番目のコメント挿入位置

施設基準コード10 : 5番目のコメント挿入位置

例

用法名(漢字名称) “朝:g 昼:g 夕:g”

↑ ↑ ↑
3 7 11

挿入位置1 : 3

挿入位置2 : 7

挿入位置3 : 11

診療行為入力で入力されたコメント(数量)を 0.15 0.25 0.35 とすると

編集後の内容は “朝:0.15g 昼:0.25g 夕:0.35g” となります。

コメント区分

処方せん二次元バーコード(QRコード)での服用方法の記録では、用法レコードへ記録する服用方法が識別できなくてはなりません。剤内に複数の薬品があり、それぞれが不均等である場合などは、正しい服用方法が記録できなくなるためこの区分を設定します。

0 : 服用方法 (デフォルト)

1 : 服用方法コメント有

2 : コメント

二次元バーコード(QRコード)で用法レコードに設定する服用方法は、コメント区分が“0 服用方法”、“1 服用方法コメント有”のコードとなります。

点数マスタの検体検査コメントを流用し設定値を管理します。

例

6xxxxxxx 4	A薬品	
001999301 1 1 2	【朝1錠 昼1錠 夕2錠】	← コメント区分 = 2
6xxxxxxx 5	B薬品	
001999301 1.5 1.5 2	【朝1.5錠 昼1.5錠 夕2錠】	← コメント区分 = 2
001000301	【1日3回毎食後に】	← コメント区分 = 0 (剤の服用方法)

(2) 診療行為入力

入力コード蘭の入力文字数は最大半角 54 文字 (54 バイト) とします。

入力できる 1 コメント (数量) は、最大整数部 5 桁、小数部 5 桁までの組合せで最大 8 文字まで入力できます。

名称は最大 37 文字表示ですので、編集後の文字数が 37 文字以上となる場合は警告メッセージを表示します。

処方せんの処方蘭の 1 行の文字長は全角 40 文字である。

剤の先頭行の先頭には “*” を編集する。

用法名には、先頭と末尾に墨付括弧を付ける。

1 行文字数 (40) - 編集文字 [*, 【, 】] (3) = 37 文字

用法コードの漢字名称と入力されたコメント (数量) から編集された内容は、患者コメントテーブルの入力コメントに格納します。

入力コメントに格納する内容は、先頭と末尾に付けた墨付括弧を削除した内容となります。

服用方法コード	コメント (数量) * 日数
9 1 40 1 3	= 54
: 空白 1 文字	

(3) 処方せんの編集

服用方法の編集

用法コードから点数マスタを読み、コメント区分に合わせた編集を行います。

0 : 服用方法 の場合

“【” + 点数マスタの漢字名称 + “】”

1 : 服用方法コメント有 と 2 : コメント の場合

“【” + 患者コメントの入力コメント + “】”

二次元バーコード (QRコード) の編集

剤内の最下行の明細から最初に現れるコメント区分が 0 又は 1 の用法コードを剤全体に係わる服用方法として記録する。

(4) お薬情報 (薬剤情報提供書) の編集

用法コードのコメント区分が 0 又は 1 のコードを服用方法とします。

剤内の最下行の明細から最初に現れる用法コードを服用方法とします。ただし、用法コードに服用時点が設定されているコードを優先します。つまり、最後の用法コードに服用時点の設定がなく、そのより前に用法コードがあり服用時点の設定のある場合、このコードが服用方法となります。

用法コードのコメント区分が 2 のコードは、直前の薬剤に対する服用コメントコードとします。前に薬剤がない場合は編集の対象外とします。

服用方法のコメントとなりますので、プログラムオプションのコメント設定とは無関係とし、必ず編集します。

(5) ユーザーコメント (0084xxxxx) にカラム位置の設定があり桁数が 0 (ゼロ) で登録されている場合、

用法コードと同様にカラム位置から数量の挿入編集を行います。入力できる数量も用法コードの仕様と同様とします。

処方せん、お薬手帳、お薬情報 (薬剤情報提供書)、入院処方せんなど、用法コードの名称を表示している帳票をカスタマイズされている場合で同様の変更が必要な場合は修正を行ってください。

3-11 住所マスタの一括置換えについて

現行のポスタルガイドのままの編集を見直し、町域名から小字や番地の括弧部分を分離した住所マスタとしましたので住所テーブル(tbl_adrs)を一括置換しました。

また、京都府については通り名による表記が一般的ということから、通り名の編集有無について機能追加しましたので、ポスタルガイドにある町域名から通り名を切り出して住所マスタに反映させました。

さらに、ポスタルガイドに記載されていない通り名について少しとなりますがマスタ化を試みました。追加した郵便番号を別紙にまとめてありますので、ご参照いただきご意見をいただければと思います。

この新住所マスタの一括置換えにより、今までユーザー自身で追加、修正した内容はすべてなくなります。また、システム管理マスタで設定する周辺住所については、新住所マスタと管理キーに矛盾が生じる場合がありますので、正しい住所が呼び出されるか確認をお願いします。

正式版のリリースまでにさらに内容を調整する場合がありますのでご注意ください。

(1) 京都府の通り名の編集

システム管理の1017患者登録機能情報設定画面の「京都府通り名編集区分」で設定します。

初期値は「1 編集する」とするので通り名付きで編集します。

通り名編集をしない場合は、「0 編集しない」に設定してください。

(2) 住所マスタテーブルからの編集

「京都府通り名編集区分」を「1 編集する」と設定した場合、患者登録画面で京都府の郵便番号を入力すると、通り名が登録されている場合には、“京都府” + 市区町村名 + 通り名 + 町域名の編集を行い、住所欄に表示します。

住所マスタテーブルのカラム名では、

PREFNAME + CITYNAME + TOWNNAME2 + TOWNNAME

となります。