

Paper 300-2013

Healthcare Claims Processing with Base SAS® through Denormalization of ANSI 837 Format

Victor Shigaev, Centers for Disease Control and Prevention; Roberto Valverde, Centers for Disease Control and Prevention

ABSTRACT

Sometimes dealing with healthcare claims can be messy. As a result of HIPAA, all health insurance claims must be submitted to insurance payers using the ANSI X12 837 messaging standard. This standard creates a compact hierarchical file for quick transmission between trading partners but because of the really complex nested structure of the data this standard is not always easy to read in and be analyzed. The paper will give a brief introduction to the X12 837 messaging standard, provide users a simple way to divide raw claims data by claims through de-normalization, and a way to use SAS as a main tool to process and analyze the claims data.

INTRODUCTION

The Health Insurance Portability and Accountability Act (HIPAA) requires that all health insurance payers comply with Electronic Data interchange (EDI). As a result, all claims for services, supplies, equipment, and health care must be in the American National Standards Institute (ANSI) X12 837 messaging standard for payment by insurance payers.

There are 4 different 837 claim/encounter standards:

1. Institutional Claims (837i) – Hospitals, Nursing Homes, Ambulatory surgery centers.
2. Dental claims (837d) – Dental practices.
3. Professional claims (837p) – Physician practices.
4. Health care data reporting (837r) – used for reporting purposes.

There are some differences between the 4 different standards but all must comply with the hierarchical organizational structure defined by the 837 format which results in an ASCII text file filled with repeating segments and loops¹:

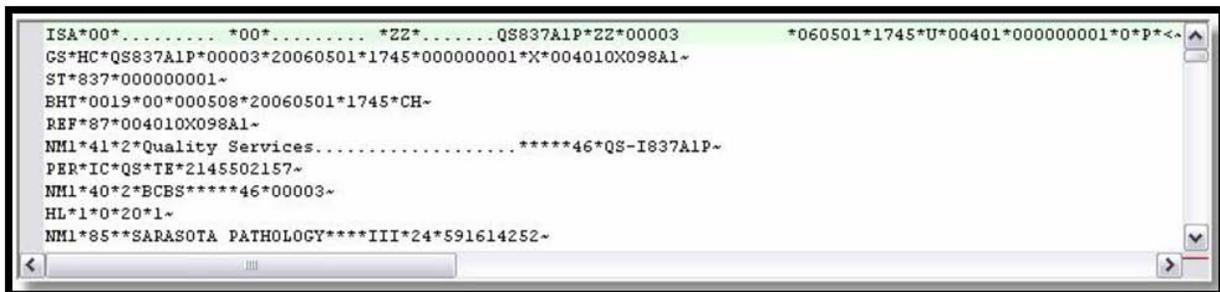
- Housekeeping information about the 837 Transaction Set reported.
 - HEADER Loop
 - Repeats 1 time
- The submitter of the information is identified with necessary contact information.
 - Loop ID - 1000
 - Repeats 1 time
- The receiver of the information is identified with necessary contact information.
 - Loop ID - 1000
 - Repeats 1 time
- The provider of health care services is identified.
 - Loop ID - 2000

- Repeats more than 1 time to allow submissions with information from multiple providers
- Each provider is labeled to enable identifying the relationship between patients receiving care from that provider later in the structure.
- For each provider of health care services the patients receiving care are identified.
 - Loop ID - 2010
 - Repeats more than 1 time to allow submissions with information from multiple patients
 - A relationship is established between the provider of the services and the patient receiving services from that provider.

The result is a text document that can look like this:

```
ST*837*0021*005010X222A1~BHT*0019*00*244579*20061015*1023*CH
~NM1*41*2*PREMIER BILLING SERVICE*****46*TGJ23~PER*IC*JERRY*
TE*3055552222*EX*231~NM1*40*2*KEY INSURANCE COMPANY*****46*6
6783JJT~HL*1**20*1~PRV*BI*PXC*203BF0100Y~NM1*85*2*BEN KILDAR
E SERVICE*****XX*9876543210~N3*234 SEAWAY ST~N4*MIAMI*FL*331
11~REF*EI*587654321~NM1*87*2~N3*2345 OCEAN BLVD~N4*MAIMI*FL*
33111~HL*2*1*22*1~SBR*P**2222~SJ*****CI~NM1*IL*1*SMITH*JANE
****MI*JS00111223333~DMG*D8*19430501*F~NM1*PR*2*KEY INSURANC
E COMPANY*****PI*999996666~REF*G2*KA6663~HL*3*2*23*0~PAT*19~
NM1*QC*1*SMITH*TED~N3*236 N MAIN ST~N4*MIAMI*FL*33413~DMG*D8
*19730501*M~CLM*26463774*100***11:B:1*Y*A*Y*I~REF*D9*1731234
5600006351~HI*BK:0340*BF:V7389~LX*1~SV1*HC:99213*40*UN*1***1
~DTP*472*D8*20061003~LX*2~SV1*HC:87070*15*UN*1***1~DTP*472*D
8*20061003~LX*3~SV1*HC:99214*35*UN*1***2~DTP*472*D8*20061010
~LX*4~SV1*HC:86663*10*UN*1***2~DTP*472*D8*20061010~SE*42*002
1~
```

Figure 1. Example of EDI X12 file



```
ISA*00*.....*00*.....*ZZ*.....QS837A1P*ZZ*00003.....*060501*1745*U*00401*000000001*0*P*~
CS*HC*QS837A1P*00003*20060501*1745*000000001*X*004010X098A1~
ST*837*000000001~
BHT*0019*00*000508*20060501*1745*CH~
REF*87*004010X098A1~
NM1*41*2*Quality Services.....*****46*QS-I837A1P~
PER*IC*QS*TE*2145502157~
NM1*40*2*BCBS*****46*00003~
HL*1*0*20*1~
NM1*85**SARASOTA PATHOLOGY***III*24*591614252~
```

Figure 2. Example of typical EDI X12 file²

Data files may come in as big files with claims that span a period of days or numerous smaller files with claims for just that day.

In most cases the claims are pretty straight forward with the patient being the same as the insurance subscriber or the subscriber having only 1 claim per dependent which we will refer to as a simple claim. Other times the 837 Implementation Guide (www.wpc-edi.com) allows for the following scenarios per transaction:

- Multiple patients per subscriber.
- A claim under the subscriber and a claim under each one of the patients for that same subscriber.
- Multiple claims under the subscriber and multiple claims under each one of the patients for that subscriber.
- A combination of all of the above.

This results in a complex nesting structure which we will refer to as a combined claim. These cases make it a little more difficult to visualize since due to the hierarchical layout the subscriber information maybe way above the patient information and there may be more than one patient/dependent associated with that subscriber. In the cases of these combined claims it is often better to de-normalize or flatten the claims so the subscriber information can be seen alongside the patient information.

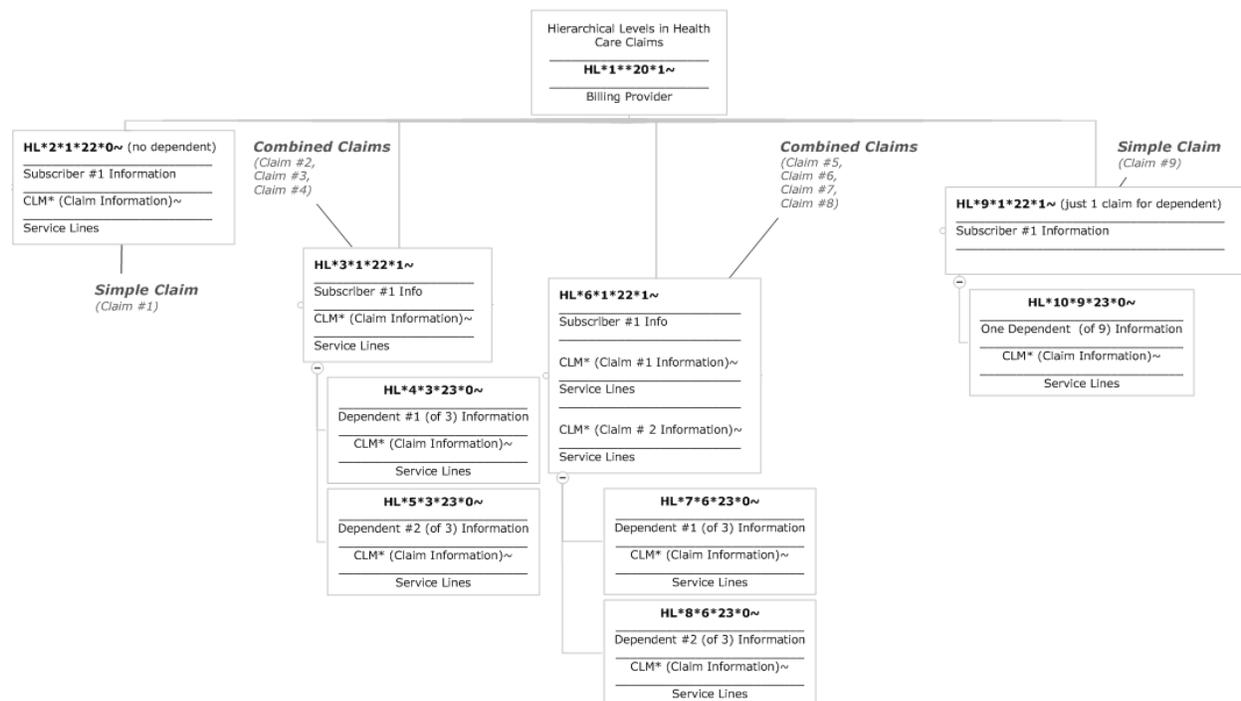


Figure 3. Complex nesting structure of combined claims

Through this paper we hope to provide a simple solution to reading in ANSI x12 837 claims data and to de-normalize the claims data so all information (subscriber and patient) per claim is located in same area. For our example we will be using the typical EDI X12 file (see figure 2)

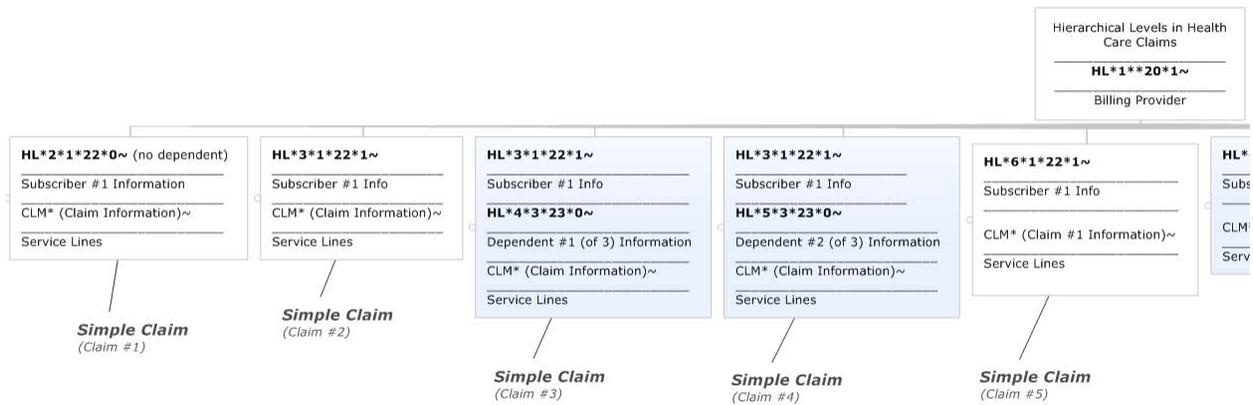


Figure 4. Denormalized combined claim

DENORMALIZING A COMBINED CLAIM

STEP 1: READING FROM THE 837 TEXT FILES INTO SAS

Using the infile statement the raw claims data is read into SAS creating a variable called var1 that holds the lines of data (1 record per line).

```
filename f1 "\\LOCAL PATH\data.txt";
libname my "\\YOUR PROJECT\";

data dat0;
line + 1;          *CREATES A LINE NUMBER FOR EACH LINE;
length myinfile $ 300 from_file $30 var1 $256; *INITIALIZING VARIABLES;
infile f1 pad missover filename = myinfile;    *READS IN FILE;
input @1 var1 $256.;
from_file = scan(myinfile, -1, "\"); /*CREATES A VARIABLE THAT STORES
SOURCE OF DATA*/

run;
```

Figure 5. Reads in the raw claims data.

STEP 2: IDENTIFYING THE LINES IN THE DATA

The lines are identified by the first characters in each segment. The "HL" lines give information about the nested structure of the following data. In the case of simple claims, the subscriber and patient are the same so no de-normalization is needed. The "HL" lines are scanned to find the instances of combined claims. The data in these types of claims are identified into information that contains subscriber information and patient information to be used for the de-normalization process. An internal claim ID is also created to be used later to merge in the subscriber data.

```
data dat1;
retain IND2 CLM22;
length line 8 IND $ 3 IND2 $3;
set dat0;

/* THE FOLLOWING SORTS OUT THE LINES INTO SUBSCRIBER AND PATIENT INFO*/
if substr (var1, 1, 3) = "HL*" then IND = scan(var1, 4, "*", "m");
else if substr (var1, 1, 3) = "CLM" then IND = "CLM";
if IND ne '' then IND2 = IND;

/* INITIATING AN INTERNAL CLAIM ID NUMBER*/
```

```

if IND = '22' then CLM22 + 1;
run;

```

Figure 6. Identifies the lines in the claims data.

STEP 3: IDENTIFYING WHERE TO INSERT THE SUBSCRIBER INFORMATION FOR COMBINED CLAIMS

Information about where the subscriber information needs to be inserted is needed so a look up table is created that contains the line where the subscriber information should go. Using the indicators we created in the previous step we can identify in between which line numbers we need the subscriber information.

```

data dat2 (keep=Insert_22_after);
set dat1 (where=(IND ne ''));
length IND_LAG $ 3 LINE_LAG 8;
IND_LAG = lag(IND);
LINE_LAG = lag(LINE);
if (IND = "CLM" and IND_LAG = "CLM") or (IND = "23" and IND_LAG ne "22")
then Insert_22_after = LINE - 1;
if Insert_22_after ne .;
run;

```

Figure 7. Creates a table with line numbers for inserting the subscriber information (claim 22 level information) for combined claims.

STEP 4: GETTING SUBSCRIBER INFORMATION TO BE MERGED

Now that we have where the subscriber information needs to go, we need to get the actual subscriber information. First thing we do is to use the table we created in step 3 and create a reference table by merging on the information about whose subscriber information is associated with the insertion lines based on the internal claim number created in step 2. Then we use PROC SQL to get the subscriber information needed from the SAS data file created in step 2 and merge it onto the reference table created in this step.

```

data dat3 (keep = CLM22 LINE);
merge dat1(in=a)                                /*TABLE WITH CLAIM INFO*/
dat2(in=b rename=Insert_22_after=LINE); /*TABLE WITH LINE INFO*/
by LINE;
if a;
if a & b;
run;

```

Figure 8. Creates a reference table between the claim information and the line number for insertion.

```

proc sql;
create table dat3_22 as
select CLM22, LINE as LINE22, var1
from dat1
where IND2 = '22'
order by 1,2;
quit;

```

Figure 9. Merges subscriber information to be inserted onto the combined claims.

STEP 5: MERGE ON SUBSCRIBER INFORMATION TO ALL COMBINED CLAIMS

Using PROC SQL we combine the subscriber information with the reference table created in the previous step. PROC SQL will create a Cartesian product that will replicate the subscriber information the associated lines.

```

proc sql;
create table dat4 as
select a.CLM22, LINE, LINE22, var1
from dat3 a, dat3_22 b
where a.CLM22 = b.CLM22;
quit;

```

Figure 10. Replicates subscriber information as Cartesian product onto all related dependencies in combined claim.

STEP 6: COMBINE MERGED SUBSCRIBER INFORMATION WITH THE ORIGINAL FILE TO CREATE DE-NORMALIZED STRUCTURE AND SORT BY ORIGINAL LINE NUMBER AND INTERNAL CLAIM ID NUMBER

Now that we have all the subscriber information and line numbers that we need in a data set, we combine it with the original data set and sort by line number. The result is a file that has all the combined claims denormalized with all the subscriber information in the same section as the corresponding patient claim.

```

data dat5;
set dat1 dat4;
run;

proc sort data = dat5 out = my.data_denorm;
by LINE LINE22;
run;

```

Figure 11. The subscriber information is appended to the original file and sorted by the original line number.

STEP 7: ASSIGN INTERNAL CLAIM ID AND CREATE SEGMENT FLAGS

Finally we create a dataset with just the information we want (var1) and create a new data set. A new claim ID is created in order to count the number claims and segment flags, "HL" (hierarchical level) and "22" (subscriber level), are created in order to make the file easier to view.

```

data dat_denorm_final (keep = var1);
set my.data_denorm;
run;

data dat1_f;
retain claim0 0;
length claim0 8 pref $ 3 flag_HL $ 4 ;
set dat_denorm_final ;

/* CREATES SEGMENT FLAGS FOR EASIER VIEWING */
pref = substr (var1, 1, 4);
if pref = "HL*" then flag_HL = scan(var1, 4, "*");
if flag_HL = '22' then claim0 + 1;
run;

```

Figure 12. A new claim ID is created and segment flags are added for easier viewing.

CONCLUSION

The ASCII X12 837 standard creates a compact file for easy transmission between trading partners using a hierarchical structure. De-normalizing gives you the option to use base SAS as an instrument to process this type of data for analytical purposes without having to purchase expensive software tools.

REFERENCES

1. Retrieved November 14, 2012 from <http://www.phdsc.org/standards/x12/tutorial.asp>.
2. Retrieved November 14, 2012 from "Beginners Guide to EDI X12 (including HIPAA)" <http://www.xtranslator.com/prod/beginguidex12.pdf>

ACKNOWLEDGMENTS

The authors would like to thank Tatiana Nieves for her contributions to this paper and to Bob Davis for helping guiding us through the world of combined claims.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. In addition, other solutions for duplicate claims or reading in raw claims data like that in Figure 1 can be obtained by contacting the authors.

Victor Shigaev
Centers for Disease Control and Prevention
3311 Toledo Rd
Hyattsville, MD 20782
301-458-4170
vshigaev@cdc.gov

Roberto Valverde
Centers for Disease Control and Prevention
3311 Toledo Rd
Hyattsville, MD 20782
301-458-4274
rvalverde@cdc.gov

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.