HEADWATERS
ECONOMICS

County Business Patterns Data Compilation and Disclosure Estimation Process

This report describes processing steps implemented to prepare the U.S. Department of Commerce, Census Bureau, County Business Patterns (CBP) mid-March employment data for use by the EPS-HDT application. The steps include downloading the data from the CBP web site, calculating estimates of employment when data are not disclosed by the Census Bureau, importing the reported and estimated data into a standard format used by the EPS-HDT application, and archiving previous years' data.

The purpose of this report is to: (1) Describe the process for downloading and organizing CBP data, (2) Describe the estimation method and the resulting accuracy, and (3) Provide a record for annual updates.

Contact:
Ray Rasker
406-570-7044
ray@headwaterseconomics.org

6/8/2010

Table of Contents

Introduction

This report describes processing steps implemented to prepare the U.S. Department of Commerce, Census Bureau, County Business Patterns (CBP) mid-March employment data for use by the EPS-HDT application. The steps include downloading the data from the CBP web site, calculating estimates of employment when data are not disclosed by the Census Bureau, importing the reported and estimated data into a standard format used by the EPS-HDT application, and archiving previous years' data.

The purpose of this report is to: (1) Describe the process for downloading and organizing CBP data, (2) Describe the estimation method and the resulting accuracy, and (3) Provide a record for annual updates.

Estimation of Non-Disclosed CBP Employment Numbers

When CBP does not provide data for the number of mid-March employees by industry, other CBP reported data (numbers of employees and companies by size class) are used to generate an estimate. The approach involves summing employment across all CBP designated size classes (firms with 1-4 employees, firms with 5-9 employees, firms with 10-19 employees, etc.). In all cases, when total mid-March employment for an industry is not available, some of the size class information is unavailable as well.

When the number of employees within a size class is not reported, we estimate the number by multiplying the average number of employees within the size class by the number of establishments within the size class. We use a hierarchical method (employing state, then national-level data) to calculate the average number of employees for size classes with missing data.

For example, for a particular county and industry (NAICS code) and year, if the number of employees within a size class is not reported, we calculate the average number of employees within the size class at the state-level. This is done by dividing the number of employees within the same industry and size class by the number of firms within the same industry and size class. Rarely, when the state-level data is unavailable, this same calculation is performed on national-level data per industry and size class. Finally, if the number of employees for a particular industry and size class is not available at the county, state, or national-level, we set the average number of employees equal to a value within the range of the size class (the mid-point is used for smaller classes and the minimum is used for larger size classes to lower our incidence of overestimating employment numbers).

Once we have obtained either the reported, if available, or estimated employees per size class, these values are summed. The final value to be used for the total mid-March employment is adjusted to fall within the CBP employment flag reported for records where data is not reported. The flag establishes a range within which the actual value would fall. If our estimate is less than or greater than the min/max values set by the flag, the estimate is adjusted up or down accordingly.

Due to the extent of size class data that CBP makes available, the estimation method yields high accuracy. We conducted an accuracy assessment for all county/industry/year combinations (2,968,100 records) where employment numbers were reported. The median difference between the reported number of employees and the estimated number was 1.04. The median percent error was 0.91. Fifty-one percent of the time the estimate was within 10 employees of the reported value, 83 percent of the time the estimate was within 50 employees of the reported value, and 91 percent of the time the estimate was within 100 employees of the reported value. Accuracy was lowest in industries with strong seasonal fluctuations (particularly NAICS 114: Fishing, Hunting, and Trapping) in which the CBP reported mid-March data tended to be lower than the estimate.

Details of the estimation method are provided in Appendix B. Tables describing the accuracy assessment of estimates are shown in Appendix M.

Tables used in the CBP application

There are six tables used in this application. The first five are in the database called "CBP_YEAR". Once the current year data has been processed, this database should be renamed to eg "CBP2007" and stored in the appropriate CBP archive folder. If it becomes necessary to add a new naics code to the final table, it is a simple process to append it from these annual databases.

1. tblCBPAllCurrentYR – holds the annual data from the 3 files provided by CBP. These three files are appended into a single table for each year.

    a. Cbp07us.txt
    b. Cbp07st.txt
    c. Cbp07co.txt

2. tbltEMPmmYEAR – holds the annual data with estimates for all NAICS codes. This will be created for each new year of data.

3. tbltEMPmmNAICS – holds all years data for selected NAICS codes. Data is appended to this table from the "tbltEMPmmYEAR" table for each new year of data.

4. tblNaics – list of all NAICS codes with descriptions as well as a filed to identify which naics codes are included in the CBP application.

5. tblCBPFlags – list of employment flags used in CBP data with minimum and maximum values.

6. tblcbpnaics – the final table that holds a record for all fips by naics by year required in the EPS-HDT application. This table is stored in a separate database.

Field names

The following standardized field names will be used in the final data table tblcbpnaics:

FIELD	FORMAT	COMMENT
fips	mediumint	Always 0 for US, shown as combined value of state & county FIPS (e.g. MT state FIPS 30 + Gallatin County FIPS 031 yields fips 30000 for the state of Montana and and fips 30031 for Gallatin County. PRIMARY KEY
id1	tinytext	NAICS, Linecode, or other where necessary.. (where id1 = naics, line, or other TBD) PRIMARY KEY
id2	tinytext	Owner or other where necessary. (where id2 = owner, or other TBD) PRIMARY KEY
yr	year	4-digit year. PRIMARY KEY
val	float	Value from cell. DEFAULT NULL
discl	tinytext	Disclosure code, if any. DEFAULT NULL

Data Append Process

Step 1: Import the current year data from the CBP web download site:

http://www.census.gov/econ/cbp/download/index.htm

The first step for each year is to check all of the fields in each CBP file to make sure that the field names are the same as in the table "tblCBPAllCurrentYR"(Appendix A). There are files with the county, state and U.S. record layout for each year. This only needs to be done if there is a change in the CBP data structure. There was no problem appending 1998-2007 data to the table "tblCBPAllCurrentYR". All three text files (County, State and US) are appended to this table.

There are three files to download:

Complete County file – a zipped file with comma delimited data
Complete State file – a zipped file with comma delimited data
Complete U.S. file – a text with comma delimited data

When importing the US data there seems to be an issue with the "ap" value in the first record. It can be manually entered to complete the data set although it is not used in this application so the import error table can be deleted and it is not necessary to enter the value. If there is a need to have this data, the text file can be opened and the value manually entered.

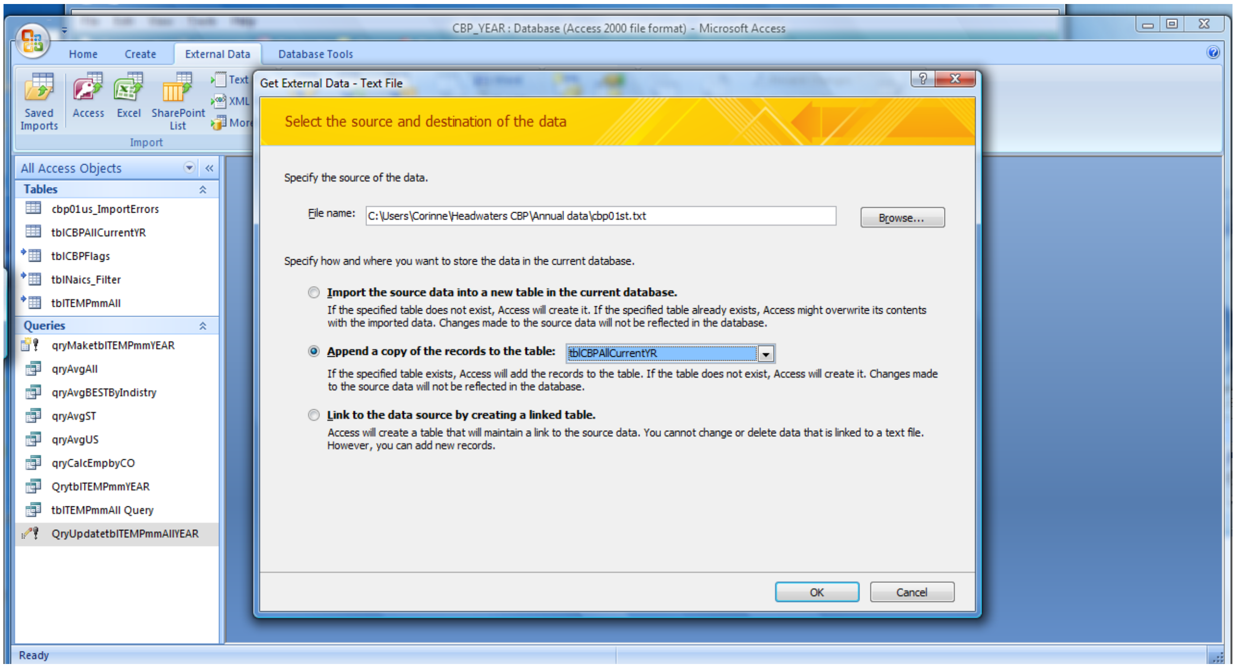
Open the CBP_YEAR database.

Use the ACCESS append text file wizard. Check the box to use the first line as field names. The FIPS codes for the U.S., state and county will be set to default to 0 so there are no null values. Append all three text files to the table "tblCBPAllCurrentYR". The three text files for 2007 are:

Cbp07us.txt
Cbp07st.txt
Cbp07co.txt – this file can take 5-10 minutes to append.

The following snapshots are from the ACCESS2007 import wizard although the database files are in ACCESS 97-2003 format.

Make sure to append the data to the table called “tblCBPAllCurrent”.



Get External Data - Text File

Select the source and destination of the data

Specify the source of the data.

File name: C:\Users\Corinne\Headwaters CBP\Annual data\cbp01st.txt [Browse...]

Specify how and where you want to store the data in the current database.

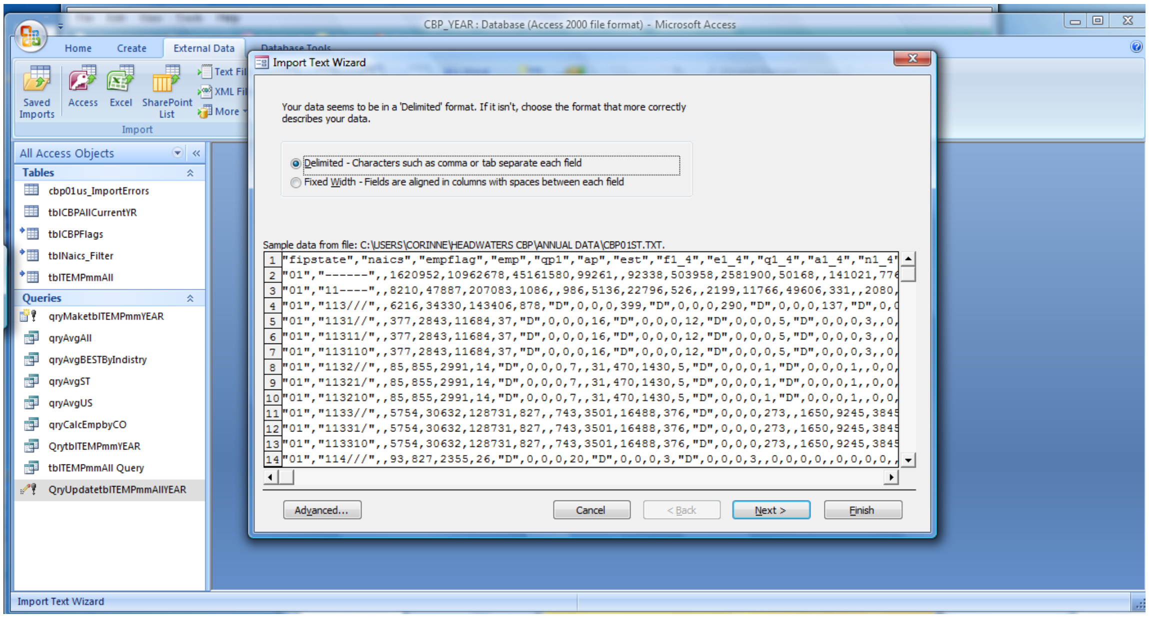
○ **Import the source data into a new table in the current database.**
If the specified table does not exist, Access will create it. If the specified table already exists, Access might overwrite its contents with the imported data. Changes made to the source data will not be reflected in the database.

● **Append a copy of the records to the table:** tblCBPAllCurrentYR
If the specified table exists, Access will add the records to the table. If the table does not exist, Access will create it. Changes made to the source data will not be reflected in the database.

○ **Link to the data source by creating a linked table.**
Access will create a table that will maintain a link to the source data. You cannot change or delete data that is linked to a text file. However, you can add new records.

[OK] [Cancel]

The data is comma delimited.



Import Text Wizard

Your data seems to be in a 'Delimited' format. If it isn't, choose the format that more correctly describes your data.

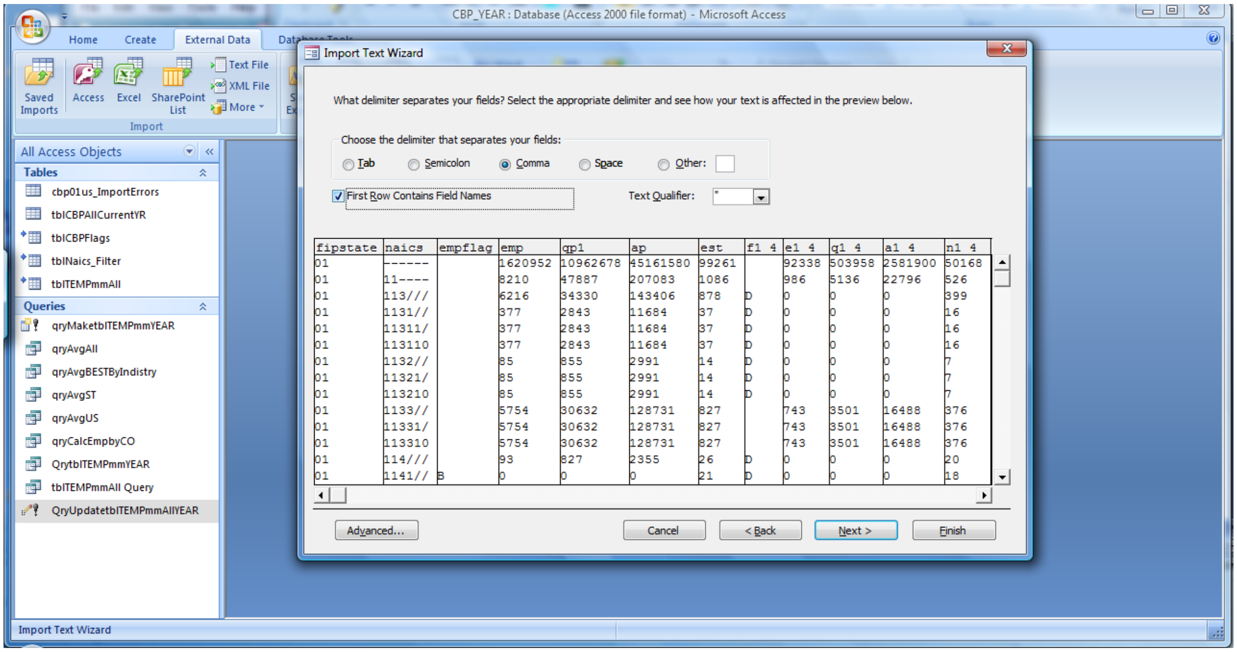
● Delimited - Characters such as comma or tab separate each field
○ Fixed Width - Fields are aligned in columns with spaces between each field

Sample data from file: C:\USERS\CORINNE\HEADWATERS CBP\ANNUAL DATA\CBP01ST.TXT.

```
"fipstate","naics","empflag","emp","qp1","ap","est","f1_4","e1_4","q1_4","a1_4","n1_4"
"01","------",,1620952,10962678,45161580,99261,,92338,503958,2581900,50168,,141021,776
"01","11----",,8210,47887,207083,1086,,986,5136,22796,526,,2199,11766,49606,331,,2080,
"01","113///",,6216,34330,143406,878,"D",0,0,0,399,"D",0,0,0,290,"D",0,0,0,137,"D",0,0
"01","1131//",,377,2843,11684,37,"D",0,0,0,16,"D",0,0,0,12,"D",0,0,0,5,"D",0,0,0,3,,0,
"01","11311/",,377,2843,11684,37,"D",0,0,0,16,"D",0,0,0,12,"D",0,0,0,5,"D",0,0,0,3,,0,
"01","113110",,377,2843,11684,37,"D",0,0,0,16,"D",0,0,0,12,"D",0,0,0,5,"D",0,0,0,3,,0,
"01","1132//",,85,855,2991,14,"D",0,0,0,7,,31,470,1430,5,"D",0,0,0,1,"D",0,0,0,1,,0,0,
"01","11321/",,85,855,2991,14,"D",0,0,0,7,,31,470,1430,5,"D",0,0,0,1,"D",0,0,0,1,,0,0,
"01","113210",,85,855,2991,14,"D",0,0,0,7,,31,470,1430,5,"D",0,0,0,1,"D",0,0,0,1,,0,0,
"01","1133//",,5754,30632,128731,827,,743,3501,16488,376,"D",0,0,0,273,,1650,9245,3845
"01","11331/",,5754,30632,128731,827,,743,3501,16488,376,"D",0,0,0,273,,1650,9245,3845
"01","113310",,5754,30632,128731,827,,743,3501,16488,376,"D",0,0,0,273,,1650,9245,3845
"01","114///",,93,827,2355,26,"D",0,0,0,20,"D",0,0,0,3,"D",0,0,0,3,,0,0,0,0,,0,0,0,0,,
```

[Advanced...] [Cancel] [< Back] [Next >] [Finish]

Make sure the text qualifier is set to the quote symbol in the following window.



Import Text Wizard

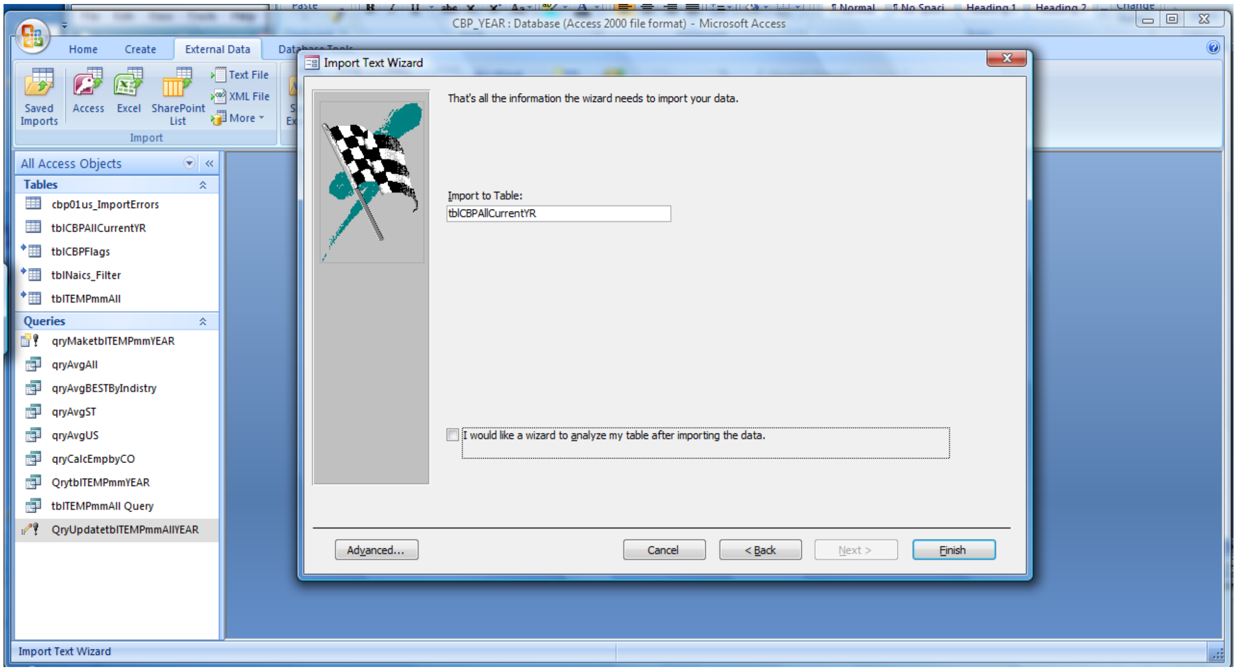
What delimiter separates your fields? Select the appropriate delimiter and see how your text is affected in the preview below.

Choose the delimiter that separates your fields:

○ Tab ○ Semicolon ● Comma ○ Space ○ Other: □

☑ First Row Contains Field Names   Text Qualifier: "

fipstate	naics	empflag	emp	qp1	ap	est	f1 4	e1 4	q1 4	a1 4	n1 4
01	------		1620952	10962678	45161580	99261		92338	503958	2581900	50168
01	11----		8210	47887	207083	1086		986	5136	22796	526
01	113///		6216	34330	143406	878	D	0	0	0	399
01	1131//		377	2843	11684	37	D	0	0	0	16
01	11311/		377	2843	11684	37	D	0	0	0	16
01	113110		377	2843	11684	37	D	0	0	0	16
01	1132//		85	855	2991	14	D	0	0	0	7
01	11321/		85	855	2991	14	D	0	0	0	7
01	113210		85	855	2991	14	D	0	0	0	7
01	1133//		5754	30632	128731	827		743	3501	16488	376
01	11331/		5754	30632	128731	827		743	3501	16488	376
01	113310		5754	30632	128731	827		743	3501	16488	376
01	114///		93	827	2355	26	D	0	0	0	20
01	1141//	B	0	0	0	21	D	0	0	0	18



Import Text Wizard

That's all the information the wizard needs to import your data.

Import to Table: tblCBPAllCurrentYR

□ I would like a wizard to analyze my table after importing the data.

The append process is the same for all three files – US, state and county data. The state and county data must be unzipped prior to import. The cbp01us_ImportErrors table that is created when the US data is imported can be deleted.

Step 2: Append reported and estimated employment data for all NAICS codes

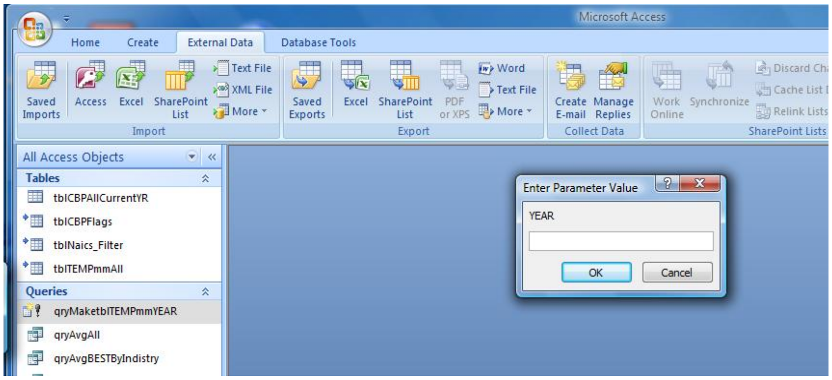
After all of the US, state and county data have been appended, the append query “qryAppendtblTEMPmmYEAR” (Appendix G) must be run. This query appends a record for every NAICS code in the “tblNAICS_Filter” (Appendix I) table located in the CBP_reference_data.mdb database to the table “tblTEMPmmYEAR” (Appendix F) which includes all of the data that will be used in the EPS-HDT application. There are about 597000 records in the created table. It can take 5-10 minutes to run this query. The append query selects fields from the “qryCalcEmpbyCO” query (Appendix B).

Two additional queries are run to append metro/non-metro data at the state and national levels. These queries select data from “tbltEMPmmYEAR” grouped by whether they are metro or non-metro and summed for the state level (“qrymetroST” Appendix K) or the national level (“qrymetroUS” Appendix L). The fipscty code is set to -998 for metro data and -997 for non-metro data at the national level. At the state level, the fips code is set to - fipsST & 998 or 997 (eg. -51998).

The fields in the new table are:

Name	Type	Size
fipsSTCty	Long Integer	7
fipstate	Long Integer	4
fipscty	Long Integer	4
naics	Text	8
EYr	Text	255
Discl	Long Integer	4
empflag	Text	255
emp	Double	8
STemp	Double	8
USemp	Double	8
AValue	Double	8
CalcEmp	Double	8

You must enter the year and then select OK in order to append the data.



Enter Parameter Value

YEAR

Step 3: Append data to the table used in the EPS-HDT application

The last query inserts the final data required by the EPS-HDT application into the table “tblcbpNAICS”. This table contains data for the selected NAICS codes for all years. Run the append query “qryAppendcbpNAICS” (Appendix I). Data is selected from the table “tbltEMPmmYEAR” (Appendix G) which is filtered by the NAICS codes included in the table “tblNaics_Filter” (Appendix J). For 1998-2007 there are 6,359,723 records.

Step 4: Append new NAICS code data to the table used in the EPS-HDT application

If there is a need to add additional NAICS codes at later date this query can be executed after changing the data in the table “tblNaics” where Naics_Filter <> 0. Change the value in the NAICS_filter field (set it to a number that is not currently used so that only the desired records are selected. The append query will have to be executed for each year since the data for each year is kept in separate databases. Use the link manager to easily re-link to the appropriate annual table.

Step 5: Archive the current year database

Rename CBP_YEAR.mdb to the current year e.g. “CBP2007.mdb” and move it to the folder where all of the backup data is stored.

The final step is to copy the CBP_YEAR – empty file and rename it CBP_YEAR.mdb so you are ready to append the next year’s data files.

Appendices

Appendix A: Annual data imported from CBP

This table contains a large number of fields so that all data provided by CBP at the national, state and county levels can be imported into the same table. There are fields that may only be relevant to one or two of those data files but all data can be imported into this table. Additional fields can be added if there are changes in the CBP data structure.

Table: tblCBPAllCurrentYR

Columns

Name	Type	Size
ID	Long Integer	4
uscode	Long Integer	4
fipstate	Long Integer	4
fipscty	Long Integer	4
naics	Text	10
empflag	Text	5
emp_nf	Text	5
emp	Double	8
qp1_nf	Text	5
qp1	Double	8
ap_nf	Text	5
ap	Double	8
est	Double	8
f1_4	Text	10
e1_4nf	Text	5
e1_4	Long Integer	4
q1_4nf	Text	5
q1_4	Long Integer	4
a1_4nf	Text	5
a1_4	Long Integer	4
n1_4	Long Integer	4
f5_9	Text	10
e5_9nf	Text	5
e5_9	Long Integer	4
q5_9nf	Text	5
q5_9	Long Integer	4
a5_9nf	Text	5
a5_9	Long Integer	4
n5_9	Long Integer	4
f10_19	Text	5
e10_19nf	Text	5
e10_19	Long Integer	4
q10_19nf	Text	5
q10_19	Long Integer	4
a10_19nf	Text	5
a10_19	Long Integer	4
n10_19	Long Integer	4
f20_49	Text	5
e20_49nf	Text	5
e20_49	Long Integer	4
q20_49nf	Text	5
q20_49	Long Integer	4
a20_49nf	Text	5
a20_49	Long Integer	4

Table: tblCBPAllCurrentYR continued…

n20_49	Long Integer	4
f50_99	Text	5
e50_99nf	Text	5
e50_99	Long Integer	4
q50_99nf	Text	5
q50_99	Long Integer	4
a50_99nf	Text	5
a50_99	Long Integer	4
n50_99	Long Integer	4
f100_249	Text	5
e100_249nf	Text	5
e100_249	Long Integer	4
q100_249nf	Text	5
q100_249	Long Integer	4
a100_249nf	Text	5
a100_249	Long Integer	4
n100_249	Long Integer	4
f250_499	Text	5
e250_499nf	Text	5
e250_499	Long Integer	4
q250_499nf	Text	5
q250_499	Long Integer	4
a250_499nf	Text	5
a250_499	Long Integer	4
n250_499	Long Integer	4
f500_999	Text	5
e500_999nf	Text	5
e500_999	Long Integer	4
q500_999nf	Text	5
q500_999	Long Integer	4
a500_999nf	Text	5
a500_999	Long Integer	4
n500_999	Long Integer	4
f1000	Text	5
e1000nf	Text	5
e1000	Long Integer	4
q1000nf	Text	5
q1000	Long Integer	4
a1000nf	Text	5
a1000	Long Integer	4
n1000	Long Integer	4
censtate	Text	255
n1000_1	Long Integer	4
n1000_2	Long Integer	4
n1000_3	Long Integer	4
n1000_4	Long Integer	4
cencty	Text	255

Appendix B: qryCalcEmpbyCty

This query includes all of the calculations required to create the data for the EPS-HDT application plus several fields that can be used to check the calculations, if desired. An estimate is calculated (“calcemp” field) and that estimate is then adjusted to fit the minimum-maximum designated by the reported employment flag (“empflag” field). It is based on data from three sources –

1. qryEmpAll or tblCBPAllCurrentYR (Appendix A)– a list of the imported data records with fields used to calculate estimates for total employment when it is not reported or else the table to which all of the CBP files was appended.
2. qryAcgBESTByIndustry (Appendix C)– this query compares the state and national level employment estimates by industry and sets up a single estimated value for each size class. This best estimate value is used to calculate an estimated value when employment data is not reported at the county level. This query is described in Appendix C. The two queries required to set up the comparison are described in Appendices D and E.
3. tblCBPFlags (Appendix D)– this file provides the minimum and maximum values for each employment flag (“empFlag” field) reported in the county data file for records where the total employment data (“emp” field) is not reported.

The fields highlighted in hot pink are included in the final table. Those highlighted in yellow are used to make calculations required to get the final estimates. The text highlighted in blue provides an explanation of the calculated field that follows the highlighted text.

```
SELECT Val(Format([qryEmpAll].[fipstate],"00") & Format([qryEmpAll].[fipscty],"000")) AS fips,

        qryEmpAll.fipstate,
        qryEmpAll.fipscty,
        qryEmpAll.naics,
        [Enter Year] AS Yr,
        tblCBPFlags.Min,
        tblCBPFlags.Max,
        qryAvgBESTByIndistry.STemp,
        qryAvgBESTByIndistry.USemp,
        qryEmpAll.empflag,
        qryEmpAll.emp,
        qryEmpAll.est,
```

The calculated estimate of employment (“CalcEmp”) is the sum of the best value for each establishment class size below. When a value is reported in the “e*” field, that value is always used. When that value is zero, the best estimate is used (state first and national second).

```
        ([Best1_4]+[best5_9]+[best10_19]+[Best20_49]+[Best50_99]
                +[Best100_249]+[Best250_499]+[Best500_999]+[Bestn10001_4]) AS CalcEmp,
```

A disclosure flag (“Discl”) is calculated based on whether the estimate was forced to meet the flag min/max. If the value was reported the disclosure flag is set to 0. If the value is an estimate and was adjusted to meet the min/max value it is set to 2. If the estimate fell within the flag range the disclosure flag is set to 1.

```
IIf([emp]>0,0,IIf([Calcemp]<[min] Or [Calcemp]>[Max],2,1)) AS Discl,
```

The final value to be used for the total mid-March employment is adjusted to fall within the employment flag ("EmpFlag" field) reported for records where data is not reported. The flag establishes a range within which the actual value would fall. If the "CalcEmp" value is less than or greater than the min/max values set by the flag, the estimate is adjusted up or down accordingly. The final value is called AValue. When the estimate is less than 1 but greater than 0 and the minimum value is set at 0-4, a value of 1 is used since there cannot be a partial employee in the establishment.

```
IIf([Emp]>0,[Emp],IIf([CalcEmp]<1,IIf([Min]=0,1,[Min]),IIf([CalcEmp]<[Min],[Min],
        IIf([CalcEmp]>[Max],[Max],[CalcEmp])))) AS Val,

qryEmpAll.n1_4,
qryEmpAll.e1_4,
IIf([e1_4]>0,[e1_4],[Bestavg1_4]*[n1_4]) AS Best1_4,
qryEmpAll.n5_9,
qryEmpAll.e5_9,
IIf([e5_9]>0,[e5_9],[Bestavg5_9]*[n5_9]) AS Best5_9,
qryEmpAll.n10_19,
qryEmpAll.e10_19,
IIf([e10_19]>0,[e10_19],[Bestavg10_19]*[n10_19]) AS Best10_19,
qryEmpAll.n20_49,
qryEmpAll.e20_49,
IIf([e20_49]>0,[e20_49],[Bestavg20_49]*[n20_49]) AS Best20_49,
qryEmpAll.n50_99,
qryEmpAll.e50_99,
IIf([e50_99]>0,[e50_99],[Bestavg50_99]*[n50_99]) AS Best50_99,
qryEmpAll.n100_249,
qryEmpAll.e100_249,
IIf([e100_249]>0,[e100_249],[Bestavg100_249]*[n100_249]) AS Best100_249,
qryEmpAll.n250_499,
qryEmpAll.e250_499,
IIf([e250_499]>0,[e250_499],[Bestavg250_499]*[n250_499]) AS Best250_499,
qryEmpAll.n500_999,
qryEmpAll.e500_999,
IIf([e500_999]>0,[e500_999],[Bestavg500_999]*[n500_999]) AS Best500_999,
qryEmpAll.n1000, qryEmpAll.e1000,
```

When an employment value is reported (field "e1000") for the class size of 1000 or greater, that reported value is always used. If the reported value equals 0, the best estimate calculated below as "bestn1000" is used.

```
IIf(nz([e1000])>0,[e1000],[bestn1000]) AS Bestn10001_4,
```

Bestn1000 is the sum of each class size times the reported number of establishments in that class size.

```
((nz([n1000_1])*[bestn1000_1])+(nz([n1000_2])*[Bestn1000_2])+(nz([n1000_3])*[bestn1000_3])
        +(nz([n1000_4])*[bestn1000_4])) AS bestn1000,
```

The following fields calculate the best estimate for each class size in the greater than 1000 size. At the county level this group is broken into 4 class sizes. This additional information is used to refine the estimates to fall within the size class reported. If the estimated value is less than the minimum size, the adjusted value will be the minimum. If the estimated value is greater than the maximum size, the value will be adjusted down to the maximum.

```
qryEmpAll.n1000_1,
qryAvgBESTByIndistry.Bestavg1000_1499,
IIf([Bestavg1000_1499]<1000,1000,IIf([Bestavg1000_1499]>1499,1499,[Bestavg1000_1499]))
         AS Bestn1000_1,
IIf(nz([n1000_1])>0,[Bestn1000_1]*nz([n1000_1]),0) AS Best1000_1499,
qryEmpAll.n1000_2,
qryAvgBESTByIndistry.Bestavg1500_2499,
IIf([Bestavg1500_2499]<1500,1500,IIf([Bestavg1500_2499]>2499,2499,[Bestavg1500_2499]))
        AS Bestn1000_2,
qryEmpAll.n1000_3,
qryAvgBESTByIndistry.Bestavg2500_4999,
IIf([Bestavg2500_4999]<2500,2500,IIf([Bestavg2500_4999]>4999,4999,[Bestavg2500_4999]))
        AS Bestn1000_3,
qryEmpAll.n1000_4,
qryAvgBESTByIndistry.Bestavg5000,
IIf([Bestavg5000]<5000,5000,[Bestavg5000]) AS Bestn1000_4
FROM (qryEmpAll LEFT JOIN tblCBPFlags ON qryEmpAll.empflag = tblCBPFlags.strFlag)
         LEFT JOIN qryAvgBESTByIndistry ON (qryEmpAll.fipstate = qryAvgBESTByIndistry.fipstate)
         AND (qryEmpAll.naics = qryAvgBESTByIndistry.naics);
```

Appendix C: qryAvgBestByIndustry

This query compares the national estimated averages (“qryAvgUS” described in Appendix E) with the state estimated averages (“qryAvgST” described in Appendix F) by industry. The state estimate is used if it is reported. When there is no reported data at the state level, the national estimate is used. A best estimate is made for each of the following establishment size classes:

N1_4	1-4 Employee Size Class
N5_9	5-9 Employee Size Class
N10_19	10-19 Employee Size Class
N20_49	20-49 Employee Size Class
N50_99	50-99 Employee Size Class
N100_249	100-249 Employee Size Class
N250_499	250-499 Employee Size Class
N500_999	500-999 Employee Size Class

The following are only reported at the county level:

N1000_1	Employment Size Class: 1,000-1,499 Employees
N1000_2	Employment Size Class: 1,500-2,499 Employees
N1000_3	Employment Size Class: 2,500-4,999 Employees
N1000_4	Employment Size Class: 5,000 or More Employees

This query links the results of two queries qryAvgUS and qryAvgST to identify the best available estimate for each establishment size class. The state by industry estimate is the first choice. When a state estimate is not available, the national estimate is used. The field names highlighted in yellow are used in calculating the final estimates for county data that was not reported.

```
SELECT qryAvgST.naics,
qryAvgST.fipstate,
qryAvgST.STemp,
qryAvgUS.USemp,
```

When the state value is 0 the US value is used.

```
IIf(nz([STemp],0)=0,[USemp],[STemp]) AS STUS_emp,
qryAvgST.STest, qryAvgUS.USest,
qryAvgST.STn1_4,
qryAvgUS.USn1_4,
```

The best available value – state first and then US – is selected for each size class.

```
IIf([STavge1_4]>0,[STavge1_4],[USavge1_4]) AS Bestavg1_4,
IIf([STavge5_9]>0,[STavge5_9],[USavge5_9]) AS Bestavg5_9,
IIf([STavge10_19]>0,[STavge10_19],[USavge10_19]) AS Bestavg10_19,
IIf([STavge20_49]>0,[STavge20_49],[USavge20_49]) AS Bestavg20_49,
IIf([STavge50_99]>0,[STavge50_99],[USavge50_99]) AS Bestavg50_99,
IIf([STavge100_249]>0,[STavge100_249],[USavge100_249]) AS Bestavg100_249,
IIf([STavge250_499]>0,[STavge250_499],[USavge250_499]) AS Bestavg250_499,
IIf([STavge500_999]>0,[STavge500_999],[USavge500_999]) AS Bestavg500_999,
IIf([STavge1000_1499]>0,[STavge1000_1499],[USavge1000_1499]) AS Bestavg1000_1499,
IIf([STavge1500_2499]>0,[STavge1500_2499],[USavge1500_2499]) AS Bestavg1500_2499,
IIf([STavge2500_4999]>0,[STavge2500_4999],[USavge2500_4999]) AS Bestavg2500_4999,
IIf([STavge5000]>0,[STavge5000],[USavge5000]) AS Bestavg5000
FROM qryAvgST RIGHT JOIN qryAvgUS ON qryAvgST.naics = qryAvgUS.naics
ORDER BY qryAvgST.naics,
qryAvgST.fipstate;
```

Appendix D: tblCBPFlags

This table is a list of the CBP flags used in the CBP data with the minimum and maximum values assigned to each. The reported "empflag" provides a range for the total employment that is used to adjust estimates when employment data is not reported. The "strflag" field links to the "empflag" field in other tables.

Columns

Name	Type	Size
ID	Long Integer	4
strFlag	Text	255
EmplSize	Text	255
Midpoint	Long Integer	4
Min	Long Integer	4
Max	Long Integer	4

Appendix E: qryAvgST

This query calculates an average for each size class in each industry at the national level. This average will be the last best value used to estimate county employment when that data is not reported. Average values are calculated by dividing the reported employment value by the number of establishments in the size class when there is a reported employment value. When there is no reported value the average for the size class is used for the smaller size classes (1-4 uses 3, 5-9 uses 7, 10-19 uses 14.5, and 20-49 uses 34.5) and the minimum size class value is used for the larger classes. For the class with 1000 or more employees another division is added because the county level data reports the number of establishments in 4 additional levels (1000-1499, 1500-2499, 2500-4999 and >5000). For these categories the (n1000/e1000) average is used when data is reported and the minimum value of each class is used when no data is available. The field names highlighted in yellow are used in the query “qryAvgBESTByIndustry”.

```
SELECT tblCBPAllCurrentYR.naics,
tblCBPAllCurrentYR.uscode,
tblCBPAllCurrentYR.fipstate,
tblCBPAllCurrentYR.fipscty,
tblCBPAllCurrentYR.empflag,
tblCBPAllCurrentYR.emp_nf,
tblCBPAllCurrentYR.emp AS USemp,
tblCBPAllCurrentYR.est AS USest,
tblCBPAllCurrentYR.n1_4 AS USn1_4,
tblCBPAllCurrentYR.e1_4 AS USe1_4,
```

The US value to be used for each size class is calculated in the following IIF statements. The first four class sizes use the midpoint of the class while the rest use the minimum value of the class size to minimize the chance of over-estimation.

```
IIf([e1_4]>0,([e1_4]/[n1_4]),3) AS USavge1_4,
IIf([e5_9]>0,([e5_9]/[n5_9]),7) AS USavge5_9,
IIf([e10_19]>0,([e10_19]/[n10_19]),14.5) AS USavge10_19,
IIf([e20_49]>0,([e20_49]/[n20_49]),34.5) AS USavge20_49,
IIf([e50_99]>0,([e50_99]/[n50_99]),50) AS USavge50_99,
IIf([e100_249]>0,([e100_249]/[n100_249]),100) AS USavge100_249,
IIf([e250_499]>0,([e250_499]/[n250_499]),250) AS USavge250_499,
IIf([e500_999]>0,([e500_999]/[n500_999]),500) AS USavge500_999,
tblCBPAllCurrentYR.e1000,
tblCBPAllCurrentYR.n1000,
```

The following classes use the minimum class size of the data reported in the county data. The minimum class size is used as the default.

```
IIf([e1000]>0,([e1000]/[n1000]),1000) AS USavge1000_1499,
IIf([e1000]>0,([e1000]/[n1000]),1500) AS USavge1500_2499,
IIf([e1000]>0,([e1000]/[n1000]),2500) AS USavge2500_4999,
IIf([e1000]>0,([e1000]/[n1000]),5000) AS USavge5000
FROM tblCBPAllCurrentYR
```

```
WHERE (((tblCBPAllCurrentYR.naics) Is Not Null)
AND ((tblCBPAllCurrentYR.fipstate)=0)
AND ((tblCBPAllCurrentYR.fipscty)=0));
```

Appendix F: qryAvgST

This query provides an average calculation for each size class in each industry in each state. When no data is available, a zero is used since the national data will be the next best estimate. The same average is used for all four subdivisions of the 100 or greater size class since the state level does not break down the one size class.

```
SELECT tblCBPAllCurrentYR.naics,
tblCBPAllCurrentYR.uscode,
tblCBPAllCurrentYR.fipstate,
tblCBPAllCurrentYR.fipscty,
tblCBPAllCurrentYR.empflag,
tblCBPAllCurrentYR.emp_nf,
tblCBPAllCurrentYR.emp AS STemp,
tblCBPAllCurrentYR.est AS STest,
tblCBPAllCurrentYR.n1_4 AS STn1_4,
tblCBPAllCurrentYR.e1_4 AS STe1_4,
IIf([e1_4]>0,([e1_4]/[n1_4]),0) AS STavge1_4,
tblCBPAllCurrentYR.e5_9,
tblCBPAllCurrentYR.n5_9,
```

If there is no value at the state level, the US level value is used.

```
IIf([e5_9]>0,([e5_9]/[n5_9]),0) AS STavge5_9,
IIf([e10_19]>0,([e10_19]/[n10_19]),0) AS STavge10_19,
IIf([e20_49]>0,([e20_49]/[n20_49]),0) AS STavge20_49,
IIf([e50_99]>0,([e50_99]/[n50_99]),0) AS STavge50_99,
IIf([e100_249]>0,([e100_249]/[n100_249]),0) AS STavge100_249,
IIf([e250_499]>0,([e250_499]/[n250_499]),0) AS STavge250_499,
IIf([e500_999]>0,([e500_999]/[n500_999]),0) AS STavge500_999,
tblCBPAllCurrentYR.e1000,
tblCBPAllCurrentYR.n1000,
IIf([e1000]>0,([e1000]/[n1000]),0) AS STavge1000_1499,
IIf([e1000]>0,([e1000]/[n1000]),0) AS STavge1500_2499,
IIf([e1000]>0,([e1000]/[n1000]),0) AS STavge2500_4999,
IIf([e1000]>0,([e1000]/[n1000]),0) AS STavge5000
FROM tblCBPAllCurrentYR
WHERE (((tblCBPAllCurrentYR.naics) Is Not Null)
  AND ((tblCBPAllCurrentYR.uscode)=0) AND ((tblCBPAllCurrentYR.fipscty)=0));
```

Appendix G: tblTEMPmmYEAR

This table contains the data that will be added to the CBP all years database plus a few extra fields to allow cross checking of some of the data calculations.

Columns

Name	Type	Size
fipstate	Integer	2
fipscty	Integer	2
naics	Text	10
EYr	Integer	2
USemp	Double	8
STemp	Double	8
emp	Double	8
empflag	Text	5
Min	Long Integer	4
Max	Long Integer	4
Discl	Text	2
CalcEmp	Double	8
Val	Double	8

Appendix H: qryAppendtbltEMPmmYEAR

This query selects data from the query “qryCalcEmpbyCty” to insert into the table “tbltEMPmmYEAR”. The combined fips state and county value is added to this table.

```
INSERT INTO tbltEMPmmYEAR
( fipsSTCty,
fipstate,
fipscty,
naics,
EYr,
USemp,
STemp,
emp,
empflag,
Min,
Max,
Discl,
CalcEmp,
Val )
SELECT Format([fipstate],"00") & Format([fipscty],"000") AS fipsSTCty,
qryCalcEmpbyCty.fipstate,
qryCalcEmpbyCty.fipscty,
qryCalcEmpbyCty.naics,
qryCalcEmpbyCty.EYr,
qryCalcEmpbyCty.USemp,
qryCalcEmpbyCty.STemp,
qryCalcEmpbyCty.emp,
qryCalcEmpbyCty.empflag,
qryCalcEmpbyCty.Min,
qryCalcEmpbyCty.Max,
qryCalcEmpbyCty.Discl,
qryCalcEmpbyCty.CalcEmp,
qryCalcEmpbyCty.val
FROM qryCalcEmpbyCty;
```

Appendix I: qryAppendtEMPmmNAICS

This query appends the current year data with estimated values and a disclosure flag into the all year table "tbltEMPmmNAICS". The data is taken from the table "tbltEMPmmYEAR" which is filtered by the list of NAICS codes included in the table "tblNaics_Filter". This dramatically decreases the number if records in the final table.

```
INSERT INTO tbltEMPmmNAICS
( fipstate,
fipscty,
FipSTCTY,
naics,
Yr,
Val,
Discl )
SELECT tbltEMPmmYEAR.fipstate,
tbltEMPmmYEAR.fipscty,
tbltEMPmmYEAR.FipSTCTY,
tbltEMPmmYEAR.naics,
tbltEMPmmYEAR.Yr,
tbltEMPmmYEAR.Val,
tbltEMPmmYEAR.Discl
FROM tbltEMPmmYEAR INNER JOIN tblNaics_Filter
            ON tbltEMPmmYEAR.naics=tblNaics_Filter.naics;
```

Appendix J: tblNaics

This table contains a complete list of all past and present NAICS codes. The naics_filter field is used to identify the naics codes that are included in the final cbpnaics table. It is a linked reference table.

Columns

Name	Type	Size
naics	Text	255
naicstxt	Text	255
naics_filter	Number	byte

Appendix K: qrymetroST

The query to append state level metro/non-metro data is “qrymetroST”.

```
INSERT INTO tblTEMPmmYEAR ( naics, EYr, fipstate, fips, msa, AValue, Discl )
SELECT tblTEMPmmYEAR.naics,
tblTEMPmmYEAR.EYr,
tblTEMPmmYEAR.fipstate,
```

Assign -998 to metro data. Data assigned to statewide (999) was grouped with metro data.

```
 (IIf([msa]="metro",Val("-" & Format([tbltEMPmmYEAR].[fipstate],"00") & "998"),
```

Assign -997 to non-metro data.

```
IIf([msa]="non-metro",Val("-" & Format([tbltEMPmmYEAR].[fipstate],"00") & "997")))) AS fips,
tblTEMPmmYEAR.msa,
Sum(tblTEMPmmYEAR.AValue) AS SumOfAValue,
Max(tblTEMPmmYEAR.Discl) AS MaxOfDiscl
FROM tblTEMPmmYEAR
GROUP BY tblTEMPmmYEAR.naics,
tblTEMPmmYEAR.EYr,
tblTEMPmmYEAR.fipstate,
(IIf([msa]="metro",Val("-" & Format([tbltEMPmmYEAR].[fipstate],"00") & "998"),
IIf([msa]="non-metro",Val("-" & Format([tbltEMPmmYEAR].[fipstate],"00") & "997")))), tblTEMPmmYEAR.msa
HAVING (((tblTEMPmmYEAR.msa) ="metro" Or (tblTEMPmmYEAR.msa) ="non-metro"));
```

Appendix L: qrymetroUS

The query to append national level metro/non-metro data is “qrymetroUS”.

```
INSERT INTO tblTEMPmmYEAR ( naics, EYr, fips, msa, AValue, Discl )
SELECT tblTEMPmmYEAR.naics,
tblTEMPmmYEAR.EYr,
```

Assign -998 to metro data. Data assigned to statewide (999) was grouped with metro data.

```
(IIf([msa]="metro",Val("-" & "998"),
```

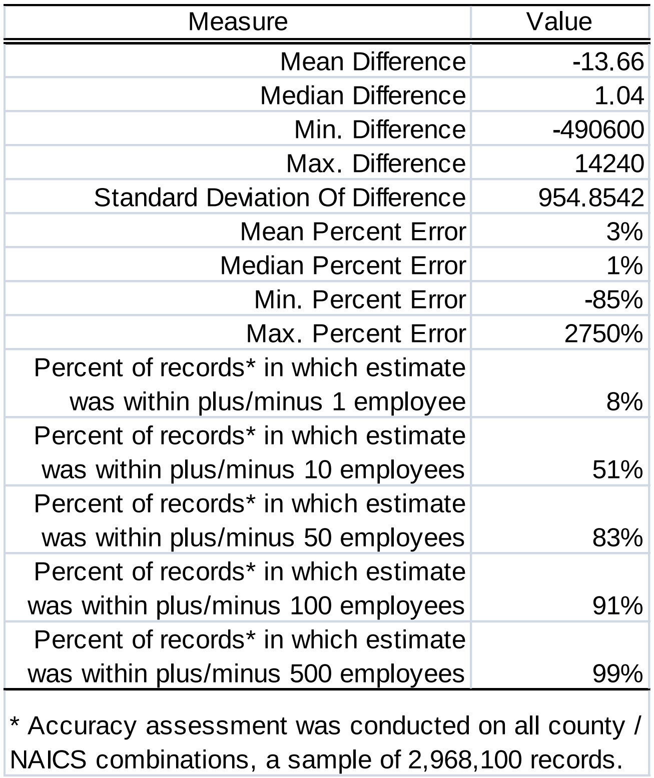
Assign -997 to non-metro data.

```
IIf([msa]="non-metro",Val("-" & "997")))) AS fips,
tblTEMPmmYEAR.msa,
Sum(tblTEMPmmYEAR.AValue) AS SumOfAValue,
Max(tblTEMPmmYEAR.Discl) AS MaxOfDiscl
FROM tblTEMPmmYEAR
GROUP BY tblTEMPmmYEAR.naics,
tblTEMPmmYEAR.EYr,
(IIf([msa]="metro",Val("-" & "998"),IIf([msa]="non-metro",Val("-" & "997")))),
 tblTEMPmmYEAR.msa
HAVING (((tblTEMPmmYEAR.msa)="metro" Or (tblTEMPmmYEAR.msa)="non-metro"));
```

Appendix M: Estimation Accuracy

The following tables show a comparison of reported employment with estimated employment.

Table 1. Summary statistics describing the accuracy of the employment estimates.



Measure	Value
Mean Difference	-13.66
Median Difference	1.04
Min. Difference	-490600
Max. Difference	14240
Standard Deviation Of Difference	954.8542
Mean Percent Error	3%
Median Percent Error	1%
Min. Percent Error	-85%
Max. Percent Error	2750%
Percent of records* in which estimate was within plus/minus 1 employee	8%
Percent of records* in which estimate was within plus/minus 10 employees	51%
Percent of records* in which estimate was within plus/minus 50 employees	83%
Percent of records* in which estimate was within plus/minus 100 employees	91%
Percent of records* in which estimate was within plus/minus 500 employees	99%

* Accuracy assessment was conducted on all county / NAICS combinations, a sample of 2,968,100 records.

Table 2. Outliers in 2007: Percent Error > 100 and Difference > plus/minus 50

State	County	NAICS	NAICS Description	Reported	Estimated	Difference	Percent Error
AK	Juneau City and Borough	114	Fishing, Hunting and Trapping	2	57	55	2750%
AK	Kodiak Island Borough	114	Fishing, Hunting and Trapping	60	202	142	237%
AK	Valdez-Cordova Census Area	114	Fishing, Hunting and Trapping	23	90	67	291%
AK	Wrangell-Petersburg Census Area	114	Fishing, Hunting and Trapping	51	124	73	142%
NJ	Ocean County	114	Fishing, Hunting and Trapping	52	105	53	102%
AK	Juneau City and Borough	1141	Fishing	2	57	55	2750%
AK	Kodiak Island Borough	1141	Fishing	60	199	139	231%
AK	Valdez-Cordova Census Area	1141	Fishing	23	90	67	291%
AK	Wrangell-Petersburg Census Area	1141	Fishing	51	124	73	142%
NJ	Ocean County	1141	Fishing	52	105	53	102%
AK	Juneau City and Borough	11411	Fishing	2	57	55	2750%
AK	Kodiak Island Borough	11411	Fishing	60	199	139	231%
AK	Valdez-Cordova Census Area	11411	Fishing	23	90	67	291%
AK	Wrangell-Petersburg Census Area	11411	Fishing	51	124	73	142%
NJ	Ocean County	11411	Fishing	52	105	53	102%
CA	Los Angeles County	7115	Independent Artists, Writers, and Performers	14086	28325	14239	101%
CO	Arapahoe County	7115	Independent Artists, Writers, and Performers	42	93	51	121%
CO	Boulder County	7115	Independent Artists, Writers, and Performers	77	169	92	119%
CO	Douglas County	7115	Independent Artists, Writers, and Performers	32	87	55	173%
CO	El Paso County	7115	Independent Artists, Writers, and Performers	62	160	98	158%
GA	DeKalb County	7115	Independent Artists, Writers, and Performers	123	263	140	113%
GA	Gwinnett County	7115	Independent Artists, Writers, and Performers	77	155	78	101%
IL	DuPage County	7115	Independent Artists, Writers, and Performers	97	205	108	111%
NJ	Essex County	7115	Independent Artists, Writers, and Performers	127	270	143	113%
NC	Mecklenburg County	7115	Independent Artists, Writers, and Performers	65	142	77	119%
CA	Los Angeles County	99	Unclassified	433	1382	949	219%
CA	Orange County	99	Unclassified	135	453	318	236%
CA	Riverside County	99	Unclassified	89	237	148	166%
CA	Sacramento County	99	Unclassified	46	123	77	167%
CA	San Bernardino County	99	Unclassified	46	138	92	200%
CA	San Francisco County	99	Unclassified	59	156	97	164%
CA	San Mateo County	99	Unclassified	32	84	52	162%
DE	New Castle County	99	Unclassified	15	81	66	440%
FL	Broward County	99	Unclassified	89	235	146	164%
FL	Hillsborough County	99	Unclassified	30	90	60	201%
FL	Miami-Dade County	99	Unclassified	98	293	195	199%
FL	Palm Beach County	99	Unclassified	69	170	101	146%
MA	Essex County	99	Unclassified	65	133	68	104%
MA	Suffolk County	99	Unclassified	65	136	71	109%
MN	Hennepin County	99	Unclassified	47	130	83	176%
MO	Adair County	99	Unclassified	69	207	138	200%
MO	Cape Girardeau County	99	Unclassified	67	208	141	210%
MO	Jackson County	99	Unclassified	26	120	94	362%
NY	Queens County	99	Unclassified	77	219	142	185%
NY	Suffolk County	99	Unclassified	50	150	100	200%
OH	Cuyahoga County	99	Unclassified	31	111	80	259%
OK	Oklahoma County	99	Unclassified	47	101	54	114%