

# Guide to Reproducing the Tables and Figures from the Analysis of Hydraulic Fracturing Fluid Data from the FracFocus Chemical Disclosure Registry 1.0

## Background

The data underlying the tables and figures in the *Analysis of Hydraulic Fracturing Fluid Data from the FracFocus Chemical Disclosure Registry 1.0* (EPA/601/R14/003; referred to as FracFocus 1.0 Data Analysis) can be recreated using queries in the Microsoft Access 2013 database developed from FracFocus 1.0 disclosures. For 12 of the tables from the FracFocus 1.0 Data Analysis (see list below), the only action required to reproduce the tables and figures from the report is to conduct an “Access query” by double-clicking the name of the corresponding query in the project database (e.g., “Table 1 – FracFocus Disclosure Count”). For the 26 remaining tables and figures presented in the FracFocus 1.0 Data Analysis (see list below), additional calculations are needed using the statistical program R (R Core Team, 2013) in order to reproduce the results in the FracFocus 1.0 Data Analysis.

Reproduction of tables and figures that include data analyzed in R (e.g., Tables ES-1 and ES-2) requires R to connect to pre-built queries in Access and then summarize query results and calculate summary statistics. R may be downloaded at <http://cran.us.r-project.org/>. Reproduction of tables that use R code will require the following packages in R in addition to the provided code: RODBC, data.table, reshape2, plyr, ggplot2, gridExtra, gtable, scales. Additionally, Geographic Information System (GIS) software is needed to reproduce the five maps.

The following list indicates the method needed to reproduce each table or figure in the FracFocus 1.0 Data Analysis and the page on which the R code can be found. If additional analysis in R is required, the query that provides the raw data is identified in the column, “Corresponding Query in the EPA Project Database.” The R code used to generate specific tables is provided in this guide beginning on page 4.

Table/Figure from FracFocus 1.0 Data Analysis	Corresponding Query in the EPA Project Database	How To Reproduce	Page Number for R Code
Table ES-1	Table ES-1 – Summary	Raw data in Access query + R Code	4
Table ES-2	Table ES-2 – Most Frequently Reported Ingredients	Raw data in Access query + R Code	5
Table 1	Table 1 - Geog Distrib Parsed Disclosures	Access query	
Table 2	Table 2 - FracFocus Disclosure Count	Access query	
Table 4	Table 4 - Chem CBI By State	Access query	
Table 5	Table 5 - FracFocus Filters	Access query	
Table 6	Table 6 - Disclosures Submitted	Access query	
Table 7	Table 7 - Chemicals Per Disclosure By State	Raw data in Access query + R Code	6
Table 8	Table 8 - Twenty Frequent Chemicals Oil	Raw data in Access query + R Code	7
Table 9	Table 9 - Twenty Frequent Chemicals Gas	Raw data in Access query + R Code	9

<b>Table/Figure from FracFocus 1.0 Data Analysis</b>	<b>Corresponding Query in the EPA Project Database</b>	<b>How To Reproduce</b>	<b>Page Number for R Code</b>
Table 10	Table 10 - Frequent Chemical Purposes	Raw data in Access query + R Code	11
Table 11	Table 11 - Counties Included Regional Diversity	Access query	
Table 12	Table 12 - Twenty Chemicals Selected Counties	Raw data in Access query + R Code	12
Table 13	Table 13 - Nonaqueous Base Fluids	Raw data in Access query + R Code	13
Table 14	Table 14 - Usage Nonaqueous Ingredients By State	Access query	
Table 15	Table 15 - Water Usage By State	Raw data in Access query + R Code	15
Table 16	Table 16 - Water Usage 90th Percentile Counties	Raw data in Access query + R Code	16
Table 17	Table 17a - Disclosures By Water Source Table 17b - Disclosures By Water Source Table 17c - Disclosures By Water Source	Raw data in Access query + R Code	17
Table 18	Table 18a - Median Concentrations By Source Table 18b - Median Concentrations By Source	Raw data in Access query + R Code	19
Table 19	Table 19 - Ten Frequent Proppants	Raw data in Access query + R Code	21
Table B-1	Table B-1 - Chemical families CBI	Access query	
Table B-2	Table B-2 - Most Frequently Reported CBI Purposes	Raw data in Access query + R Code	23
Appendix C	qryChemAnalysis	Raw data in Access query + R Code	24
Table D-1	Table D-1a - Disclosures By Operator By State Table D-1 - Disclosures By Operator By State	Access query	
Table E-1	Table E-1 - Reporting regulations	Access query	
Table F-1	Table F-1 – Additive Purposes	Access query	
Table G-1	Table G-1 - Twenty Chemicals Andrews	Raw data in Access query + R Code	27
Table G-2	Table G-2 - Twenty Chemicals Bradford	Raw data in Access query + R Code	29
Table G-3	Table G-3 - Twenty Chemicals Dunn	Raw data in Access query + R Code	31
Table G-4	Table G-4 - Twenty Chemicals Garfield	Raw data in Access query + R Code	33
Table G-5	Table G-5 - Twenty Chemicals Kern	Raw data in Access query + R Code	35
Table H-1	Table H-1 - County Water Usage	Raw data in Access query + R Code	
Figure 2	Maps	Raw data in Access query + R Code + GIS	38
Figure 3	Maps	Raw data in Access query + R Code + GIS	39
Figure 4	Figure 4 - Monthly Distribution	Access query	
Figure 5	Maps	Raw data in Access query + R Code + GIS	40

---

<b>Table/Figure from FracFocus 1.0 Data Analysis</b>	<b>Corresponding Query in the EPA Project Database</b>	<b>How To Reproduce</b>	<b>Page Number for R Code</b>
Figure 6	Maps	Raw data in Access query + R Code + GIS	41
Figure 7	Maps	Raw data in Access query + R Code + GIS	42

## R Code to Reproduce Table ES-1

**Table ES-1 includes state-specific information on the number of unique disclosures with a fracture date between January 1, 2011, and February 28, 2013; total water volumes reported per disclosure; and the number of unique additive ingredients reported per disclosure**

```
library(RODBC)
library(data.table)
library(reshape2)
conn<-odbcConnectAccess2007("FracFocus_03-2015.accdb")

#####
# State, number of disclosures

ES<-data.table(sqlQuery(conn,"SELECT * FROM [Table ES-1 - Summary]"))
ES<-rbind(ES,data.table(State='Entire Dataset',Disclosures=sum(ES$Disclosures)))

#####
# Table 15 - Water volume per disclosure, 5th and 95th
df<-data.table(sqlQuery(conn,"SELECT * FROM [Table 15 - Water Usage By State]"))
state.out<-

df[,list(Water.Median=median(as.numeric(VolumeFFQA),na.rm=TRUE),Water.5th=quantile(VolumeFFQA
,0.05,na.rm=TRUE),Water.95th=quantile(VolumeFFQA,0.95,na.rm=TRUE)),by=State]
state.out.total<-df[,list(State=as.character("Entire Dataset"),
Water.Median=median(VolumeFFQA,na.rm=TRUE),
Water.5th=quantile(VolumeFFQA,0.05,na.rm=TRUE),Water.95th=quantile(VolumeFFQA,
0.95,na.rm=TRUE)),]
tbl15<-rbind(state.out,state.out.total)

#####
# Table 7 - Chemicals per disclosure
df<-data.table(sqlQuery(conn,"SELECT * FROM [Table 7 - Chemicals Per Disclosure By State]"))

df.summary<-
df[,list(Chemicals.Median=median(as.numeric(NumChemicals)),Chemicals.5th=quantile(NumChemicals,
0.05), Chemicals.95th=quantile(NumChemicals,0.95)),by=State]
df.summary[State=="z_State Uncertain",State:="State Uncertain"]
df.total<-
df[,list(Chemicals.Median=median(as.numeric(NumChemicals)),Chemicals.5th=quantile(NumChemicals,
0.05), Chemicals.95th=quantile(NumChemicals,0.95))]
df.total[,State:="Entire Dataset"]
tbl7<-rbind(df.summary,df.total,use.names=TRUE)

setkey(ES,State)
setkey(tbl7,State)
setkey(tbl15,State)
write.csv(ES[tbl15[tbl7]],'Table ES-1.csv',row.names=FALSE)
```

## R Code to Reproduce Table ES-2

**Table ES-2 includes the most frequently reported additive ingredients in disclosures associated with oil wells and in disclosures associated with gas wells**

```
library(RODBC)
library(data.table)
library(reshape2)
conn<-odbcConnectAccess2007("FracFocus_03-2015.accdb")

ES<-data.table(sqlQuery(conn,"SELECT * FROM [Table ES-2 - Most Frequently Reported Chemicals]"))

ES_Oil<-ES[TypeFFQA=='Oil']
ES_Gas<-ES[TypeFFQA=='Gas']

processData <- function(df) {
  wellrecords<-length(unique(df$WellId))
  ingredrecords<-length(df$IngredientId)
  setkey(df,ChemicalName,Cas,WellId)

  df.well<-
df[,list(SumFluidConcentration=sum(FluidConcentration)),by=c("ChemicalName","Cas","WellId")]
  setkey(df.well,ChemicalName,Cas)
  df.rollup<-
df.well[,list(NumberOfWells=length(WellId),PercentOfWells=length(WellId)/wellrecords,
MedianPercentOfFluid=median(SumFluidConcentration),
PercentileFluid_5th=quantile(SumFluidConcentration,0.05),
PercentileFluid_95th=quantile(SumFluidConcentration,0.95)),by=c("ChemicalName","Cas")]
  df.rollup[,c("NumberOfWellsPct","MedianPercentOfFluid"):=list(paste(NumberOfWells,"
(",round(PercentOfWells*100,4),"%"),sep=""),paste(MedianPercentOfFluid,"%",sep=""))]]
  df.rollup[Cas!="",freqrank:=rank(-NumberOfWells,ties.method="min")]
  df.rollup<-df.rollup[order(freqrank)]
  df.rollup<-df.rollup[freqrank<=12 | is.na(freqrank),]

  return(df.rollup[,list(freqrank,ChemicalName,NumberOfWellsPct, MedianPercentOfFluid)])
}

ES_Oil <- processData(ES_Oil)
ES_Oil[,Type:='Oil']
ES_Oil[,freqrank:=NULL]
setcolorder(ES_Oil,c('Type',names(ES_Oil)[names(ES_Oil)!='Type']))

ES_Gas <- processData(ES_Gas)
ES_Gas[,Type:='Gas']
ES_Gas[,freqrank:=NULL]
setcolorder(ES_Gas,c('Type',names(ES_Gas)[names(ES_Gas)!='Type']))

write.csv(cbind(ES_Oil,ES_Gas),'Table ES-2.csv',row.names=FALSE)
```

## R Code to Reproduce Table 7

**Table 7 includes the number of unique additive ingredients per disclosure, summarized by state**

```
library(RODBC)
library(data.table)
conn<-odbcConnectAccess2007("FracFocus_03-2015.accdb")
df<-data.table(sqlQuery(conn,"SELECT * FROM [Table 7 - Chemicals Per Disclosure By State]"))

df.summary<-
df[,list(Disclosures=length(WellId),Median.Chemicals=quantile(NumChemicals,0.5),Percentile.5th.Chemicals=quantile(NumChemicals,0.05),Percentile.95th.Chemicals=quantile(NumChemicals,0.95)),by=State]
df.summary[State=="z_State Uncertain",State:="State Uncertain"]
df.total<-
df[,list(Disclosures=length(WellId),Median.Chemicals=quantile(NumChemicals,0.5),Percentile.5th.Chemicals=quantile(NumChemicals,0.05),Percentile.95th.Chemicals=quantile(NumChemicals,0.95))]
df.total[,State:="Entire Dataset"]
df.summary<-rbind(df.summary,df.total,use.names=TRUE)
write.csv(df.summary,"Table 7.csv",row.names=FALSE)
```

## R Code to Reproduce Table 8

**Table 8 includes the twenty most frequently reported additive ingredients in oil disclosures, ranked by frequency of occurrence**

```
library(RODBC)
library(data.table)
conn<-odbcConnectAccess2007("FracFocus_03-2015.accdb")
df<-data.table(sqlQuery(conn,"SELECT * FROM [Table 8 - Twenty Frequent Chemicals Oil]"))
wellrecords<-length(unique(df$WellId))
ingredrecords<-length(df$IngredientId)
summarydf<-data.frame(WellRecordsUsed=wellrecords, IngredientRecordsUsed=ingredrecords)

setkey(df,ChemicalName,Cas,WellId)
df.well<-
df[,list(SumFluidConcentration=sum(FluidConcentration)),by=c("ChemicalName","Cas","WellId")]

setkey(df.well,ChemicalName,Cas)

df.rollup1<-df.well[,list(NumberOfWells=length(WellId),PercentOfWells=length(WellId)/wellrecords,
MedianPercentOfFluid=median(SumFluidConcentration),
PercentileFluid_5th=quantile(SumFluidConcentration,0.05),
PercentileFluid_95th=quantile(SumFluidConcentration,0.95)),by=c("ChemicalName","Cas")]

setkey(df,ChemicalName,Cas)

df.rollup2<-
df[,list(NumberOfRecords=length(IngredientId),PercentOfRecords=length(IngredientId)/ingredrecords,
MedianPercentAdditive=median(AdditiveConcentration),
PercentileAdditive_5th=quantile(AdditiveConcentration,0.05),
PercentileAdditive_95th=quantile(AdditiveConcentration,0.95)),by=c("ChemicalName","Cas")]

setkey(df.rollup1,ChemicalName,Cas)
setkey(df.rollup2,ChemicalName,Cas)
df.rollupfinal<-df.rollup1[df.rollup2]
df.rollupfinal[,NumberOfWellsPct:=paste(NumberOfWells,"
(",round(PercentOfWells*100,4),"%","sep=")]
df.rollupfinal[,NumberOfRecordsPct:=paste(NumberOfRecords," (",
round(PercentOfRecords*100,4),"%","sep=")]

df.rollupfinal[Cas!="",freqrank:=rank(-NumberOfWells,ties.method="min")]

df.rollupfinal[,c("Cas","MedianPercentOfFluid","PercentileFluid_5th",
"PercentileFluid_95th","MedianPercentAdditive","PercentileAdditive_5th","PercentileAdditive_95th"):=
list(paste("","Cas,sep="),
paste(MedianPercentOfFluid,"%",sep="),
paste(PercentileFluid_5th,"%",sep="),
paste(PercentileFluid_95th,"%",sep="),
```

```
paste(MedianPercentAdditive,"%",sep=""),
paste(PercentileAdditive_5th,"%",sep=""),
paste(PercentileAdditive_95th,"%",sep=""))]]

df.rollupfinal<-df.rollupfinal[order(freqrank)]
df.rollupfinal<-df.rollupfinal[freqrank<=20 | is.na(freqrank),]

df.out1<-df.rollupfinal[,list(ChemicalName,Cas,NumberOfWellsPct,
MedianPercentOfFluid,PercentileFluid_5th,PercentileFluid_95th,
NumberOfRecordsPct,MedianPercentAdditive,PercentileAdditive_5th, PercentileAdditive_95th)]
df.out1[ChemicalName=="",ChemicalName:="Total"]

write.csv(df.out1,"Table 8.csv",row.names=FALSE)
```



## R Code to Reproduce Table 9

**Table 9 includes the twenty most frequently reported additive ingredients in gas disclosures, ranked by frequency of occurrence**

```
library(RODBC)
library(data.table)
conn<-odbcConnectAccess2007("FracFocus_03-2015.accdb")
df<-data.table(sqlQuery(conn,"SELECT * FROM [Table 9 - Twenty Frequent Chemicals Gas]"))
wellrecords<-length(unique(df$WellId))
ingredrecords<-length(df$IngredientId)
summarydf<-data.frame(WellRecordsUsed=wellrecords, IngredientRecordsUsed=ingredrecords)

setkey(df,ChemicalName,Cas,WellId)
df.well<-
df[,list(SumFluidConcentration=sum(FluidConcentration)),by=c("ChemicalName","Cas","WellId")]

setkey(df.well,ChemicalName,Cas)

df.rollup1<-df.well[,list(NumberOfWells=length(WellId),PercentOfWells=length(WellId)/wellrecords,
MedianPercentOfFluid=median(SumFluidConcentration),
PercentileFluid_5th=quantile(SumFluidConcentration,0.05),
PercentileFluid_95th=quantile(SumFluidConcentration,0.95)),by=c("ChemicalName","Cas")]

setkey(df,ChemicalName,Cas)

df.rollup2<-
df[,list(NumberOfRecords=length(IngredientId),PercentOfRecords=length(IngredientId)/ingredrecords,
MedianPercentAdditive=median(AdditiveConcentration),
PercentileAdditive_5th=quantile(AdditiveConcentration,0.05),
PercentileAdditive_95th=quantile(AdditiveConcentration,0.95)),by=c("ChemicalName","Cas")]

setkey(df.rollup1,ChemicalName,Cas)
setkey(df.rollup2,ChemicalName,Cas)
df.rollupfinal<-df.rollup1[df.rollup2]
df.rollupfinal[,NumberOfWellsPct:=paste(NumberOfWells,"
(",round(PercentOfWells*100,4),"%","sep=")]
df.rollupfinal[,NumberOfRecordsPct:=paste(NumberOfRecords," (",
round(PercentOfRecords*100,4),"%","sep=")]

df.rollupfinal[Cas!="",freqrank:=rank(-NumberOfWells,ties.method="min")]

df.rollupfinal[,c("Cas","MedianPercentOfFluid","PercentileFluid_5th",
"PercentileFluid_95th","MedianPercentAdditive","PercentileAdditive_5th","PercentileAdditive_95th"):=
list(paste("","Cas,sep="),
paste(MedianPercentOfFluid,"%",sep="),
paste(PercentileFluid_5th,"%",sep="),
paste(PercentileFluid_95th,"%",sep="),
```

```
paste(MedianPercentAdditive,"%",sep=""),
paste(PercentileAdditive_5th,"%",sep=""),
paste(PercentileAdditive_95th,"%",sep=""))]]

df.rollupfinal<-df.rollupfinal[order(freqrank)]
df.rollupfinal<-df.rollupfinal[freqrank<=20 | is.na(freqrank),]

df.out1<-df.rollupfinal[,list(ChemicalName,Cas,NumberOfWellsPct,
MedianPercentOfFluid,PercentileFluid_5th,PercentileFluid_95th,
NumberOfRecordsPct,MedianPercentAdditive,PercentileAdditive_5th, PercentileAdditive_95th)]
df.out1[ChemicalName=="",ChemicalName:="Total"]

write.csv(df.out1,"Table 9.csv",row.names=FALSE)
```

## R Code to Reproduce Table 10

**Table 10 includes the frequently reported additive ingredients and commonly listed purposes for additives that contain the ingredients**

```
library(RODBC)
library(data.table)
conn<-odbcConnectAccess2007("FracFocus_03-2015.accdb")
df<-data.table(sqlQuery(conn,"SELECT * FROM [Table 10 - Frequent Chemical Purposes]"))

df.well<-df[,length(unique(WellId)),by=c("ChemicalName","Cas")]
df.well[,freqrank:=rank(-V1,ties.method="min")]
df.well<-df.well[order(ChemicalName)]
# CAS numbers in Tables 14 and 15
df.well<-df.well[Cas %in% c('7647-01-0','67-56-1','64742-47-8','67-63-0','7732-18-5','64-17-5','107-19-7','111-30-8','107-21-1','77-92-9','1310-73-2','7727-54-0','14808-60-7','10222-01-2','7647-14-5','9000-30-0','64-19-7','111-76-2','91-20-3','64742-94-5','1310-58-3','9003-35-4'),]
frequentChems<-df.well$Cas
df2<-df[Cas %in% frequentChems]
df2_chems<-df2[!is.na(Category),length(IngredientId),by=list(ChemicalName,Cas,Category)]
df2_chems<-df2_chems[order(ChemicalName, -V1)]
df2_chems[,Category2:=paste(Category," (",V1,")",sep="")]
df2_chems<-df2_chems[,paste(Category2,collapse=" ",by=list(ChemicalName,Cas))

write.csv(df2_chems,"Table 10.csv",row.names=FALSE)
```

## R Code to Reproduce Table 12

**Table 12 includes a comparison of twenty most frequently reported chemicals among selected counties**

```
library(RODBC)
library(data.table)
library(reshape2)
conn<-odbcConnectAccess2007("FracFocus_03-2015.accdb")
df<-data.table(sqlQuery(conn,"SELECT * FROM [Table 12 - Twenty Chemicals Selected Counties]"))

df.national<-df[,length(unique(WellId)),by=c("ChemicalName","Cas")]
df.national[,freqrnk:=rank(-V1,ties.method="min")]
df.national<-df.national[freqrnk<=20,]

df.selected<-df[(County=="Andrews" & State=="Texas") | (County=="Bradford" &
State=="Pennsylvania") | (County=="Dunn" & State=="North Dakota") | (County=="Garfield" &
State=="Colorado") | (County=="Kern" & State=="California"),]

df2<-df.selected[,length(unique(WellId)),by=list(State,County,ChemicalName,Cas)]
df2[,freqrnk:=rank(-V1,ties.method="min"),by=list(State,County)]
df2<-df2[freqrnk<=20,]

counties<-c("Andrews","Dunn","Kern","Bradford","Garfield")
df_final<-
data.table(data.frame(County1=c(rep(counties,each=5),rep("National",times=5)),County2=c(rep(countie
s,times=6)),Percent=as.numeric(NA)))
for(i in 1:length(counties)) {
  for(j in 1:length(counties)) {
    if(counties[i]==counties[j]) {
      pct <- 1
    } else {
      pct <- sum(df2[County==counties[i]]$Cas %in%
df2[County==counties[j]]$Cas)/mean(c(length(df2[County==counties[i]]$Cas),length(df2[County==count
ies[j]]$Cas)))
    }
    df_final[County1==counties[i] & County2==counties[j],Percent:=pct]
  }
  pct<-sum(df2[County==counties[i]]$Cas %in%
df.national$Cas)/mean(c(length(df2[County==counties[i]]$Cas),length(df.national$Cas)))
  df_final[County1=="National" & County2==counties[i],Percent:=pct]
}

df_final<-dcast(melt(df_final,measure="Percent"),formula=County1~County2,sum)
write.csv(df_final,"Table 12.csv",row.names=FALSE)
```

## R Code to Reproduce Table 13

**Table 13 includes the non-aqueous ingredients reported in base fluids**

```
library(RODBC)
library(data.table)
conn<-odbcConnectAccess2007("FracFocus_03-2015.accdb")
df<-data.table(sqlQuery(conn,"SELECT * FROM [Table 13 - Nonaqueous Base Fluids]"))
wellrecords<-length(unique(df$WellId))
ingredrecords<-length(df$IngredientId)
summarydf<-data.frame(WellRecordsUsed=wellrecords,
IngredientRecordsUsed=ingredrecords)

setkey(df,ChemicalName,Cas,WellId)
df.well<-
df[,list(SumFluidConcentration=sum(FluidConcentration)),by=c("ChemicalName","Cas","WellId")]
setkey(df.well,ChemicalName,Cas)

df.rollup1<-df.well[,list(NumberOfWells=length(WellId),PercentOfWells=length(WellId)/wellrecords,
MedianPercentOfFluid=median(SumFluidConcentration),
PercentileFluid_5th=quantile(SumFluidConcentration,0.05),
PercentileFluid_95th=quantile(SumFluidConcentration,0.95)),by=c("ChemicalName","Cas")]

setkey(df,ChemicalName,Cas)
df.rollup2<-
df[,list(NumberOfRecords=length(IngredientId),PercentOfRecords=length(IngredientId)/ingredrecords,
MedianPercentAdditive=median(AdditiveConcentration),
PercentileAdditive_5th=quantile(AdditiveConcentration,0.05),
PercentileAdditive_95th=quantile(AdditiveConcentration,0.95)),by=c("ChemicalName","Cas")]

setkey(df.rollup1,ChemicalName,Cas)
setkey(df.rollup2,ChemicalName,Cas)
df.rollupfinal<-df.rollup1[df.rollup2]
df.rollupfinal[,NumberOfWellsPct:=paste(NumberOfWells," (",
round(PercentOfWells*100,4),"%)",sep="")]
df.rollupfinal[,NumberOfRecordsPct:=paste(NumberOfRecords," (",
round(PercentOfRecords*100,4),"%)",sep="")]
df.rollupfinal[Cas!="",freqrnk:=rank(-NumberOfWells,ties.method="max")]
df.rollupfinal[Cas!="",fluidrnk:=rank(-MedianPercentOfFluid,
ties.method="max")]

df.rollupfinal[,c("Cas","MedianPercentOfFluid","PercentileFluid_5th",
"PercentileFluid_95th","MedianPercentAdditive","PercentileAdditive_5th","PercentileAdditive_95th"):=
list(paste("","Cas",sep=""),
paste(MedianPercentOfFluid,"%",sep=""),
paste(PercentileFluid_5th,"%",sep=""),
paste(PercentileFluid_95th,"%",sep=""),
paste(MedianPercentAdditive,"%",sep=""),
```

```
paste(PercentileAdditive_5th,"%",sep=""),
paste(PercentileAdditive_95th,"%",sep=""))]]
df.rollupfinal<-df.rollupfinal[order(freqrank)]

df.out1<-df.rollupfinal[,list(freqrank,ChemicalName,Cas,NumberOfWellsPct,
MedianPercentOfFluid,PercentileFluid_5th,PercentileFluid_95th,
NumberOfRecordsPct,MedianPercentAdditive,PercentileAdditive_5th, PercentileAdditive_95th)]

write.csv(df.out1,"Table 13.csv",row.names=FALSE)
```

## R Code to Reproduce Table 15

**Table 15 includes total water volumes, summarized by state**

```
library(RODBC)
library(data.table)
conn<-odbcConnectAccess2007("FracFocus_03-2015.accdb")
df<-data.table(sqlQuery(conn,"SELECT * FROM [Table 15 - Water Usage By State]"))
state.out<-
df[,list(NumDisclosures_ValidVolume=length(WellId),Water.Cumulative=sum(VolumeFFQA,na.rm=TRUE)
,Volume.Median=quantile(VolumeFFQA,0.5,na.rm=TRUE),Water.5th=quantile(VolumeFFQA,0.05,na.rm=
TRUE),Water.95th=quantile(VolumeFFQA,0.95,na.rm=TRUE)),by=State]

state.out.total<-df[,list(State=as.character("Entire Dataset"),
NumDisclosures_ValidVolume=length(WellId),Water.Cumulative=sum(VolumeFFQA,na.rm=TRUE),Volum
e.Median=quantile(VolumeFFQA,0.5,
na.rm=TRUE),Water.5th=quantile(VolumeFFQA,0.05,na.rm=TRUE),Water.95th=quantile(VolumeFFQA,
0.95,na.rm=TRUE)),]

state.out[,volrank:=rank(-Water.Cumulative)]
state.out[State=="State Uncertain",volrank:=9999]
state.out<-state.out[order(volrank)]
state.out[,volrank:=NULL]

state.out<-rbind(state.out,state.out.total)

write.csv(state.out,"Table 15.csv",row.names=FALSE)
```

## R Code to Reproduce Table 16

**Table 15 includes total water volumes for selected counties in approximately the 90th percentile of disclosures**

```
library(RODBC)
library(data.table)
conn<-odbcConnectAccess2007("FracFocus_03-2015.accdb")
df<-data.table(sqlQuery(conn,"SELECT * FROM [Table 16 - Water Usage 90th Percentile Counties]"))
selectCounties<-df[((State=='Arkansas' & County=='White') | (State=='Arkansas' & County=='Conway') |
(State=='Arkansas' & County=='Clebune') | (State=='New Mexico' & County=='Lea') | (State=='North
Dakota' & County=='Dunn') | (State=='Pennsylvania' & County=='Susquehanna') | (State=='Pennsylvania'
& County=='Tioga') | (State=='Texas' & County=='Wheeler') | (State=='Texas' & County=='Johnson') |
(State=='Texas' & County=='Wise') | (State=='Texas' & County=='DeWitt') | (State=='Texas' &
County=='Irion') | (State=='Texas' & County=='Denton') | (State=='Texas' & County=='Reeves') |
(State=='Texas' & County=='Howard') | (State=='Texas' & County=='Gaines') | (State=='Texas' &
County=='Mitchell') | (State=='Texas' & County=='Milam') | (State=='Wyoming' &
County=='Sweetwater')))]

part1<-
selectCounties[,list(NumDisclosures_ValidVolume=length(WellId),PctOil=sum(ifelse(TypeFFQA=="Oil",1,
0))/length(WellId),PctGas=sum(ifelse(TypeFFQA=="Gas",1,0))/length(WellId),Water.Cumulative=sum(Vo
lumeFFQA,na.rm=TRUE),Volume.Median=median(VolumeFFQA,na.rm=TRUE),Water.5th=quantile(Volu
meFFQA,0.05,na.rm=TRUE),Water.95th=quantile(VolumeFFQA,0.95,na.rm=TRUE)),by=list(State,County)]

part2<-selectCounties[,list(State="All 90th Percentile
Counties",County="",NumDisclosures_ValidVolume=length(WellId),PctOil=sum(ifelse(TypeFFQA=="Oil",
1,0))/length(WellId),PctGas=sum(ifelse(TypeFFQA=="Gas",1,0))/length(WellId),Water.Cumulative=sum(
VolumeFFQA,na.rm=TRUE),Volume.Median=median(VolumeFFQA,na.rm=TRUE),Water.5th=quantile(Vol
umeFFQA,0.05,na.rm=TRUE),Water.95th=quantile(VolumeFFQA,0.95,na.rm=TRUE)),]

part3<-df[,list(State="Entire
Dataset",County="",NumDisclosures_ValidVolume=length(WellId),PctOil=sum(ifelse(TypeFFQA=="Oil",
1,0))/length(WellId),PctGas=sum(ifelse(TypeFFQA=="Gas",1,0))/length(WellId),Water.Cumulative=sum(Vo
lumeFFQA,na.rm=TRUE),Volume.Median=median(VolumeFFQA,na.rm=TRUE),Water.5th=quantile(Volu
meFFQA,0.05,na.rm=TRUE),Water.95th=quantile(VolumeFFQA,0.95,na.rm=TRUE)),]

part1<-part1[order(-Water.Cumulative)]
county.out<-rbind(part1,part2,part3)

write.csv(county.out,"Table 16.csv",row.names=FALSE)
```



## R Code to Reproduce Table 17

Table 17 includes the number of disclosures having terms suggestive of water sources, summarized by state. Three queries in the database produce data that is analyzed in R:

- Table 17a identifies disclosures with terms related to water sources.
- Table 17b produces the numbers of disclosures that meet quality assurance criteria. The data is used for calculations of percentages of disclosures that report a water source term.
- Table 17c identifies disclosures with source water terms that include only “water” without further specifics.

```
library(RODBC)
library(data.table)
library(reshape2)
conn<-odbcConnectAccess2007("FracFocus_03-2015.accdb")
df<-data.table(sqlQuery(conn,"SELECT * FROM [Table 17a - Disclosures By Water Source]"))
df[,StateFFQA:=as.character(StateFFQA)]
df<-df[order(WellId,Source)]
df[,CombinedSources:=paste(unique(Source),collapse="/"),by=WellId]
df[,State:=ifelse(AllStateOK==TRUE,StateFFQA,"z_State Uncertain")]

df_well_counts<-df[,list(Disclosures=length(unique(WellId))),by=list(State,CombinedSources)]

df_state_totals<-df[,list(Disclosures=length(unique(WellId))),by=State]
df_state_totals[,CombinedSources:="z_Total"]
df_well_counts<-rbind(df_well_counts,df_state_totals,use.names=TRUE)

df_source_totals<-df[,list(Disclosures=length(unique(WellId))),by=CombinedSources]
df_source_totals[,State:="zz_Total"]
df_well_counts<-rbind(df_well_counts,df_source_totals,use.names=TRUE)

df_total<-
data.frame(State="zz_Total",CombinedSources="z_Total",Disclosures=length(unique(df$WellId)))
df_well_counts<-rbind(df_well_counts,df_total)

df_valid_volume_disclosures<-data.table(sqlQuery(conn,"SELECT * FROM [Table 17b - Disclosures By Water Source]"))
df_valid_volume_disclosures<-
rbind(df_valid_volume_disclosures,data.frame(State="zz_Total",Disclosures=sum(df_valid_volume_disclosures$Disclosures)))
df_valid_volume_disclosures[,CombinedSources:="zz_DisclosuresWithValidConc"]
df_well_counts<-rbind(df_well_counts,df_valid_volume_disclosures,use.names=TRUE)

df_nonspecific_water<-data.table(sqlQuery(conn,"SELECT * FROM [Table 17c - Disclosures By Water Source]"))
df_nonspecific_water[,StateFFQA:=as.character(StateFFQA)]
df_nonspecific_water[,State:=ifelse(AllStateOK==TRUE,StateFFQA,"z_State Uncertain")]
df_nonspecific_water<-df_nonspecific_water[,list(Disclosures=length(unique(WellId))),by=State]
```

---

```

df_nonspecific_water<-
rbind(df_nonspecific_water,data.frame(State="zz_Total",Disclosures=sum(df_nonspecific_water$Disclosures)))
df_nonspecific_water[,CombinedSources:="zzzz_WaterUnspecified"]
df_well_counts<-rbind(df_well_counts,df_nonspecific_water,use.names=TRUE)

states<-unique(df_well_counts$State)
df_blankrow<-
data.table(data.frame(State=states,CombinedSources="zzz_PercentageSourceIdentified",Disclosures=as.numeric(NA)))
for(i in 1:length(states)) {
  identify_source<-df_well_counts[State==states[i] & CombinedSources=="z_Total"]$Disclosures
  valid_conc<-df_well_counts[State==states[i] &
CombinedSources=="zz_DisclosuresWithValidConc"]$Disclosures
  df_blankrow[State==states[i],Disclosures:=round(ifelse(length(identify_source)>0,identify_source,0)/valid_conc,4)]
}

df_well_counts[,Disclosures:=as.numeric(Disclosures)]
df_well_counts<-rbind(df_well_counts,df_blankrow,use.names=TRUE)
df_well_counts<-
dcast(melt(df_well_counts,measure="Disclosures"),formula=CombinedSources~State,sum)
df_well_counts<-data.table(df_well_counts)
df_well_counts[CombinedSources=="z_Total",CombinedSources:="_Total"]
df_well_counts[CombinedSources=="zz_DisclosuresWithValidConc",CombinedSources:="DisclosuresWithValidConc"]
df_well_counts[CombinedSources=="zzz_PercentageSourceIdentified",CombinedSources:="PercentageSourceIdentified"]
df_well_counts[CombinedSources=="zzzz_WaterUnspecified",CombinedSources:="WaterUnspecified"]
setnames(df_well_counts,c("z_State Uncertain","zz_Total"),c("State Uncertain","Total"))
df_well_counts[,c("CombinedSources","Sort1"):=list(as.character(CombinedSources),ifelse(CombinedSources %in% c('fresh','lease water','surface'),'1_fresh',ifelse(CombinedSources %in% c('produced','produced/recycled','recycled'),'2_reused',ifelse(!(CombinedSources %in% c('DisclosuresWithValidConc','PercentageSourceIdentified','_Total','WaterUnspecified')),'3_mixedother','4_bottom'))))]
df_well_counts<-df_well_counts[order(Sort1,CombinedSources)]
df_well_counts[,Sort1:=NULL]
tmp<-names(df_well_counts)[!(names(df_well_counts) %in% c('State Uncertain','CombinedSources','Total'))]
setcolorder(df_well_counts,c('CombinedSources',tmp[order(tmp)],'State Uncertain','Total'))
write.csv(df_well_counts,"Table 17.csv",row.names=FALSE)

```

---

## R Code to Reproduce Table 18

Table 18 includes the median of the maximum fluid concentrations of water by source, summarized by state. Two queries in the database produce data that is analyzed in R:

- Table 18a produces fracturing fluid concentrations for water source terms
- Table 18b produces fracturing fluid concentration for “water” without other source terms

```
library(RODBC)
library(data.table)
library(reshape2)
conn<-odbcConnectAccess2007("FracFocus_03-2015.accdb")
df<-data.table(sqlQuery(conn,"SELECT * FROM [Table 18a - Median Concentrations By Source]"))
df[,StateFFQA:=as.character(StateFFQA)]
df<-df[order(WellId,Source)]
df[,CombinedSources:=paste(unique(Source),collapse="/"),by=WellId]
df[,State:=ifelse(AllStateOK==TRUE,StateFFQA,"z_State Uncertain")]

df2<-df[,sum(FluidConcentration,na.rm=TRUE),by=list(WellId,Source, State,CombinedSources)]
df2_state_source<-df2[,round(median(V1,na.rm=TRUE),0),by=list(State,CombinedSources,Source)]
df2_state_source<-
df2_state_source[,paste(V1[toupper(substr(Source,1,7))!="NOMINAL"],collapse="/"),by=list(State,
CombinedSources)]
df2_state<-
df2[toupper(substr(Source,1,7))!="NOMINAL",round(median(V1,na.rm=TRUE),0),by=list(State)]
df2_state[,CombinedSources:="z_MedianAllSpecific"]
df2_source<-df2[,round(median(V1,na.rm=TRUE),0),by=list(CombinedSources,Source)]
df2_source<-
df2_source[,paste(V1[toupper(substr(Source,1,7))!="NOMINAL"],collapse="/"),by=list(CombinedSources
)]
df2_source[,State:="zz_Total"]
df2_state_source<-
rbind(df2_state_source,df2_state,df2_source,data.frame(State="zz_Total",CombinedSources="z_Media
nAllSpecific",V1=round(median(df2[toupper(substr(Source,1,7))!="NOMINAL"]$V1,na.rm=TRUE),0)),use.
names=TRUE)

df_nonspecific_water<-data.table(sqlQuery(conn,"SELECT * FROM [Table 18b - Median Concentrations
By Source]"))
df_nonspecific_water[,StateFFQA:=as.character(StateFFQA)]
df_nonspecific_water[,State:=ifelse(AllStateOK==TRUE,StateFFQA,"z_State Uncertain")]
df3<-df_nonspecific_water[,sum(FluidConcentration,na.rm=TRUE),by=list(WellId,Source,State)]
df3_water<-df3[,round(median(V1,na.rm=TRUE),0),by=list(State,Source)]
df3_water<-
rbind(df3_water,data.frame(State="zz_Total",Source="Water",V1=round(median(df3$V1,na.rm=TRUE),
0),stringsAsFactors=FALSE),use.names=TRUE)
df3_water[,Source:="zz_MedianNonSpecific"]
setnames(df3_water,"Source","CombinedSources")

df2_state_source<-rbind(df2_state_source,df3_water,use.names=TRUE)
```

```
df2_state_source<-
dcast(melt(df2_state_source,measure="V1"),formula=CombinedSources~State,function(x)
paste(x,collapse=""))
df2_state_source<-data.table(df2_state_source)

df2_state_source[CombinedSources=="z_MedianAllSpecific",CombinedSources:="MedianAllSpecific"]
df2_state_source[CombinedSources=="zz_MedianNonSpecific",CombinedSources:="MedianNonSpecific"]

setnames(df2_state_source,c("z_State Uncertain","zz_Total"),c("State Uncertain","Total"))

df2_state_source[,c("CombinedSources","Sort1"):=list(as.character(CombinedSources),ifelse(CombinedSources %in% c('fresh','lease water','surface'),'1_fresh',ifelse(CombinedSources %in% c('produced','produced/recycled','recycled'),'2_reused',ifelse(!(CombinedSources %in% c('MedianAllSpecific','MedianNonSpecific')),'3_mixedother','4_bottom')))))]
df2_state_source<-df2_state_source[order(Sort1,CombinedSources)]
df2_state_source[,Sort1:=NULL]
tmp<-names(df2_state_source)[!(names(df2_state_source) %in% c('State Uncertain','CombinedSources','Total'))]
setcolorder(df2_state_source,c('CombinedSources',tmp[order(tmp)],'State Uncertain','Total'))

write.csv(df2_state_source,"Table 18.csv",row.names=FALSE)
```

## R Code to Reproduce Table 19

**Table 19 includes the ten most frequently reported proppant ingredients, ranked by frequency of occurrence**

```
library(RODBC)
library(data.table)
conn<-odbcConnectAccess2007("FracFocus_03-2015.accdb")
df<-data.table(sqlQuery(conn,"SELECT * FROM [Table 19 - Ten Frequent Proppants]"))
wellrecords<-length(unique(df$WellId))
ingredrecords<-length(df$IngredientId)
summarydf<-data.frame(WellRecordsUsed=wellrecords, IngredientRecordsUsed=ingredrecords)

setkey(df,ChemicalName,Cas,WellId)
df.well<-
df[,list(SumFluidConcentration=sum(FluidConcentration)),by=c("ChemicalName","Cas","WellId")]
freqprops<-df.well[,length(WellId),by=Cas]
freqprops[Cas!="",freqrank:=rank(-SumFluidConcentration,ties.method="min")]
freqprops<-freqprops[freqrank<=10,]
freqprops<-as.character(freqprops$Cas)

setkey(df.well,ChemicalName,Cas)

df.rollup1<-df.well[,list(NumberOfWells=length(WellId),PercentOfWells=length(WellId)/wellrecords,
MedianPercentOfFluid=median(SumFluidConcentration),
PercentileFluid_5th=quantile(SumFluidConcentration,0.05),
PercentileFluid_95th=quantile(SumFluidConcentration,0.95)),by=c("ChemicalName","Cas")]

setkey(df,ChemicalName,Cas)

df.rollup2<-
df[,list(NumberOfRecords=length(IngredientId),PercentOfRecords=length(IngredientId)/ingredrecords,
MedianPercentAdditive=median(AdditiveConcentration),
PercentileAdditive_5th=quantile(AdditiveConcentration,0.05),
PercentileAdditive_95th=quantile(AdditiveConcentration,0.95)),by=c("ChemicalName","Cas")]

setkey(df.rollup1,ChemicalName,Cas)
setkey(df.rollup2,ChemicalName,Cas)

df.rollupfinal<-df.rollup1[df.rollup2]
df.rollupfinal[,NumberOfWellsPct:=paste(NumberOfWells,"
(",round(PercentOfWells*100,4),"%)",sep="")]
df.rollupfinal[,NumberOfRecordsPct:=paste(NumberOfRecords," (",
round(PercentOfRecords*100,4),"%)",sep="")]

df.rollupfinal[Cas!="",freqrank:=rank(-NumberOfWells,ties.method="min")]
```

```
df.rollupfinal[,c("Cas", "MedianPercentOfFluid", "PercentileFluid_5th",
"PercentileFluid_95th", "MedianPercentAdditive", "PercentileAdditive_5th", "PercentileAdditive_95th"):=
  list(paste("", Cas, sep=""),
paste(MedianPercentOfFluid, "%", sep=""),
paste(PercentileFluid_5th, "%", sep=""),
paste(PercentileFluid_95th, "%", sep=""),
paste(MedianPercentAdditive, "%", sep=""),
paste(PercentileAdditive_5th, "%", sep=""),
paste(PercentileAdditive_95th, "%", sep="")))]

df.rollupfinal<-df.rollupfinal[order(freqrank)]
df.rollupfinal<-df.rollupfinal[freqrank<=10 | is.na(freqrank),]

df.out1<-df.rollupfinal[,list(ChemicalName, Cas, NumberOfWellsPct,
MedianPercentOfFluid, PercentileFluid_5th, PercentileFluid_95th,
NumberOfRecordsPct, MedianPercentAdditive, PercentileAdditive_5th, PercentileAdditive_95th)]
df.out1[ChemicalName=="", ChemicalName:="Total"]

write.csv(df.out1, "Table 19.csv", row.names=FALSE)
```

## R Code to Reproduce Table B-2

**Table B-2 includes the most frequently reported chemical families among CBI ingredients and their most commonly listed purposes**

```
library(RODBC)
library(data.table)
conn<-odbcConnectAccess2007("FracFocus_03-2015.accdb")
df<-data.table(sqlQuery(conn,"SELECT * FROM [Table B-2 - Most Frequently Reported CBI Purposes]"))
df2<-df[order(FamilyName, -Ingredients)]
df2<-df2[!is.na(Category)]
df2[,CategoryCounts:=paste(Category, '(',Ingredients, ')',sep="")]
df_final<-df2[,list(Categories=paste(CategoryCounts,collapse=",")),by=FamilyName]
write.csv(df_final,"AllFamilies_Categories.csv",row.names=FALSE)
top20families<-df2[,sum(Disclosures),by=FamilyName][order(-V1)][1:20,as.character(FamilyName)]
df_final<-df_final[FamilyName %in% top20families,]
write.csv(df_final,"Table B-2.csv",row.names=FALSE)
```

## R Code to Reproduce Appendix C

**Appendix C includes histograms of hydraulic fracturing fluid concentrations for most frequently reported additive ingredients**

```
library(RODBC)
library(plyr)
library(ggplot2)
library(gridExtra)
library(gtable)
library(scales)

#### Import Access Files
conn<-odbcConnectAccess2007("FracFocus_03-2015.accdb")
df<-sqlQuery(conn,"SELECT * FROM [qryChemAnalysis]")

# create histograms
histGraph <- function(df, cas.num) {
  #Subset data
  df.subset <- df[which(df$Cas == cas.num),]
  chemname <- as.vector(df.subset$ChemicalName[1])

  #Sum Fluid Concentration
  df.sum <- ddply(df.subset, .(WellId, TypeFFQA),
    summarize, SumFluidConcentration=sum(FluidConcentration))

  #set upper limit for limit graph
  upper.limit <- as.vector(quantile(df.sum$SumFluidConcentration, c(.95)))

  # Create a dataframe with all, oil, and gas data
  df.total <- df.sum
  df.total$TypeFFQA <- "All disclosures"
  df.all <- rbind(df.total, df.sum)

  # to use for legend
  df.all$Median <- df.all$TypeFFQA

  #create oil and gas indices
  oil.index <- which(df.all$TypeFFQA == "Oil")
  gas.index <- which(df.all$TypeFFQA == "Gas")
  total.index <- which(df.all$TypeFFQA == "All disclosures")

  #Add the median to the dataframe to use in the aes function
  df.median <- data.frame(Median = c("All disclosures", "Oil", "Gas"),
    Value=c(median(df.all$SumFluidConcentration[total.index]),
      median(df.all$SumFluidConcentration[oil.index]),
      median(df.all$SumFluidConcentration[gas.index])))
```



---

```

# Base Plot
p0 = ggplot(df.all, aes(x=SumFluidConcentration, fill = Median),show_guide=F)
p0 = p0 + geom_histogram(show_guide=F) + geom_histogram(color = l("black"), show_guide=F)
p0 = p0 + geom_vline(data=df.median, aes(xintercept= Value, fill=Median), linetype=1 ,color="black",
                      size=1, show_guide=F)
p0 = p0 + facet_wrap(~ Median, ncol=1)
p0 = p0 + ylab(NULL)
p0 = p0 + xlab("Sum of Fluid Concentration (%)")
p0 = p0 + theme(plot.title=element_text(size=10), axis.title.x=element_text(size=10))

# Upper Limit on X-Axis
p2 = p0 + ggtitle("Upper Limit (95th Percentile\nof Total) on the X-Axis")
p2 = p2 + scale_x_continuous(limits=c(0, upper.limit))

# Log Scale
p3 = p0 + ggtitle("Log Scale on X-Axis\n")
p3 = p3 + scale_x_log10(labels= trans_format("log10", function(x) 10^x))

# create plot to extract legend from
df.legend = data.frame(Legend = "Median", Value = median(df.sum$SumFluidConcentration))
p.legend = ggplot(df.all, aes(x=SumFluidConcentration),show_guide=F)
p.legend = p.legend + geom_histogram(show_guide=F) + geom_histogram(color = l("black"),
show_guide=F)
p.legend = p.legend + geom_vline(data=df.legend, aes(xintercept= Value, fill=Legend), linetype=1
,color="black",
                      size=1, show_guide=T)
legend = gtable_filter(ggplot_gtable(ggplot_build(p.legend)), "guide-box")

# label for global y-axis
label = textGrob("Frequency", rot = 90, vjust = 0.5)

# Arrange graphs on one page
png(file = paste(chemname, "png", sep="."))
grid.arrange(label,
              arrangeGrob(p2 + theme(legend.position="none"),
                           p3 + theme(legend.position="none"),
                           ncol = 2,
                           main = textGrob(chemname, vjust = 1, gp = gpar(fontface = "bold", cex = 1.25))
                           ),
              legend,
              widths=unit.c(unit(2, "lines"), unit(1, "npc") - unit(2, "lines") - legend$width, legend$width),
nrow=1)
dev.off()
}

### Run Plots

```

---

```
number.wells <- ddply(df, .(ChemicalName, Cas),
  summarize,NumberOfWells=length(unique(WellId)),NumberOfIngredients=length(IngredientId))
number.wells$freqrank <- rank(-number.wells$NumberOfWells,ties.method="min")
number.wells <- number.wells[order(number.wells$freqrank),]

# top 20 most frequent chemicals
top20cas <- number.wells$Cas[1:20]

# histogram loop
for (i in 1:length(top20cas)) {
  histGraph(df, top20cas[i])
}
```

## R Code to Reproduce Table G-1

**Table G-1 includes the twenty most frequently reported additive ingredients in Andrews County, Texas, ranked by frequency of occurrence**

```
library(RODBC)
library(data.table)
conn<-odbcConnectAccess2007("FracFocus_03-2015.accdb")
df<-data.table(sqlQuery(conn,"SELECT * FROM [Table G-1 - Twenty Chemicals Andrews]"))
wellrecords<-length(unique(df$WellId))
ingredrecords<-length(df$IngredientId)
summarydf<-data.frame(WellRecordsUsed=wellrecords, IngredientRecordsUsed=ingredrecords)

setkey(df,ChemicalName,Cas,WellId)
df.well<-
df[,list(SumFluidConcentration=sum(FluidConcentration)),by=c("ChemicalName","Cas","WellId")]

setkey(df.well,ChemicalName,Cas)

df.rollup1<-df.well[,list(NumberOfWells=length(WellId),PercentOfWells=length(WellId)/wellrecords,
MedianPercentOfFluid=median(SumFluidConcentration),
PercentileFluid_5th=quantile(SumFluidConcentration,0.05),
PercentileFluid_95th=quantile(SumFluidConcentration,0.95)),by=c("ChemicalName","Cas")]

setkey(df,ChemicalName,Cas)

df.rollup2<-
df[,list(NumberOfRecords=length(IngredientId),PercentOfRecords=length(IngredientId)/ingredrecords,
MedianPercentAdditive=median(AdditiveConcentration),
PercentileAdditive_5th=quantile(AdditiveConcentration,0.05),
PercentileAdditive_95th=quantile(AdditiveConcentration,0.95)),by=c("ChemicalName","Cas")]

setkey(df.rollup1,ChemicalName,Cas)
setkey(df.rollup2,ChemicalName,Cas)
df.rollupfinal<-df.rollup1[df.rollup2]
df.rollupfinal[,NumberOfWellsPct:=paste(NumberOfWells,"
(",round(PercentOfWells*100,4),"%","sep="")]
df.rollupfinal[,NumberOfRecordsPct:=paste(NumberOfRecords," (",
round(PercentOfRecords*100,4),"%","sep=")]

df.rollupfinal[Cas!="",freqrank:=rank(-NumberOfWells,ties.method="min")]

df.rollupfinal[,c("Cas","MedianPercentOfFluid","PercentileFluid_5th",
"PercentileFluid_95th","MedianPercentAdditive","PercentileAdditive_5th","PercentileAdditive_95th"):=
list(paste("","Cas,sep=""),
paste(MedianPercentOfFluid,"%",sep=""),
paste(PercentileFluid_5th,"%",sep=""),
paste(PercentileFluid_95th,"%",sep="),
```

```
paste(MedianPercentAdditive,"%",sep=""),
paste(PercentileAdditive_5th,"%",sep=""),
paste(PercentileAdditive_95th,"%",sep=""))]]

df.rollupfinal<-df.rollupfinal[order(freqrank)]
df.rollupfinal<-df.rollupfinal[freqrank<=20 | is.na(freqrank),]

df.out1<-df.rollupfinal[,list(ChemicalName,Cas,NumberOfWellsPct,
MedianPercentOfFluid,PercentileFluid_5th,PercentileFluid_95th,
NumberOfRecordsPct,MedianPercentAdditive,PercentileAdditive_5th, PercentileAdditive_95th)]
df.out1[ChemicalName=="",ChemicalName:="Total"]

write.csv(df.out1,"Table G-1.csv",row.names=FALSE)
```

## R Code to Reproduce Table G-2

**Table G-2 includes the twenty most frequently reported additive ingredients in Bradford County, Pennsylvania, ranked by frequency of occurrence**

```
library(RODBC)
library(data.table)
conn<-odbcConnectAccess2007("FracFocus_03-2015.accdb")
df<-data.table(sqlQuery(conn,"SELECT * FROM [Table D-2 - Twenty Chemicals Bradford]"))
wellrecords<-length(unique(df$WellId))
ingredrecords<-length(df$IngredientId)
summarydf<-data.frame(WellRecordsUsed=wellrecords, IngredientRecordsUsed=ingredrecords)

setkey(df,ChemicalName,Cas,WellId)
df.well<-
df[,list(SumFluidConcentration=sum(FluidConcentration)),by=c("ChemicalName","Cas","WellId")]

setkey(df.well,ChemicalName,Cas)

df.rollup1<-df.well[,list(NumberOfWells=length(WellId),PercentOfWells=length(WellId)/wellrecords,
MedianPercentOfFluid=median(SumFluidConcentration),
PercentileFluid_5th=quantile(SumFluidConcentration,0.05),
PercentileFluid_95th=quantile(SumFluidConcentration,0.95)),by=c("ChemicalName","Cas")]

setkey(df,ChemicalName,Cas)

df.rollup2<-
df[,list(NumberOfRecords=length(IngredientId),PercentOfRecords=length(IngredientId)/ingredrecords,
MedianPercentAdditive=median(AdditiveConcentration),
PercentileAdditive_5th=quantile(AdditiveConcentration,0.05),
PercentileAdditive_95th=quantile(AdditiveConcentration,0.95)),by=c("ChemicalName","Cas")]

setkey(df.rollup1,ChemicalName,Cas)
setkey(df.rollup2,ChemicalName,Cas)
df.rollupfinal<-df.rollup1[df.rollup2]
df.rollupfinal[,NumberOfWellsPct:=paste(NumberOfWells,"
(",round(PercentOfWells*100,4),"%","sep=")]
df.rollupfinal[,NumberOfRecordsPct:=paste(NumberOfRecords," (",
round(PercentOfRecords*100,4),"%","sep=")]

df.rollupfinal[Cas!="",freqrank:=rank(-NumberOfWells,ties.method="min")]

df.rollupfinal[,c("Cas","MedianPercentOfFluid","PercentileFluid_5th",
"PercentileFluid_95th","MedianPercentAdditive","PercentileAdditive_5th","PercentileAdditive_95th"):=
list(paste("","Cas,sep="),
paste(MedianPercentOfFluid,"%",sep="),
paste(PercentileFluid_5th,"%",sep="),
paste(PercentileFluid_95th,"%",sep=),
```

```
paste(MedianPercentAdditive,"%",sep=""),
paste(PercentileAdditive_5th,"%",sep=""),
paste(PercentileAdditive_95th,"%",sep=""))]]

df.rollupfinal<-df.rollupfinal[order(freqrank)]
df.rollupfinal<-df.rollupfinal[freqrank<=20 | is.na(freqrank),]

df.out1<-df.rollupfinal[,list(ChemicalName,Cas,NumberOfWellsPct,
MedianPercentOfFluid,PercentileFluid_5th,PercentileFluid_95th,
NumberOfRecordsPct,MedianPercentAdditive,PercentileAdditive_5th, PercentileAdditive_95th)]
df.out1[ChemicalName=="",ChemicalName:="Total"]

write.csv(df.out1,"Table G-2.csv",row.names=FALSE)
```

## R Code to Reproduce Table G-3

**Table G-3 includes the twenty-one most frequently reported additive ingredients in Dunn County, North Dakota, ranked by frequency of occurrence**

```
library(RODBC)
library(data.table)
conn<-odbcConnectAccess2007("FracFocus_03-2015.accdb")
df<-data.table(sqlQuery(conn,"SELECT * FROM [Table D-3 - Twenty Chemicals Dunn]"))
wellrecords<-length(unique(df$WellId))
ingredrecords<-length(df$IngredientId)
summarydf<-data.frame(WellRecordsUsed=wellrecords, IngredientRecordsUsed=ingredrecords)

setkey(df,ChemicalName,Cas,WellId)
df.well<-
df[,list(SumFluidConcentration=sum(FluidConcentration)),by=c("ChemicalName","Cas","WellId")]

setkey(df.well,ChemicalName,Cas)

df.rollup1<-df.well[,list(NumberOfWells=length(WellId),PercentOfWells=length(WellId)/wellrecords,
MedianPercentOfFluid=median(SumFluidConcentration),
PercentileFluid_5th=quantile(SumFluidConcentration,0.05),
PercentileFluid_95th=quantile(SumFluidConcentration,0.95)),by=c("ChemicalName","Cas")]

setkey(df,ChemicalName,Cas)

df.rollup2<-
df[,list(NumberOfRecords=length(IngredientId),PercentOfRecords=length(IngredientId)/ingredrecords,
MedianPercentAdditive=median(AdditiveConcentration),
PercentileAdditive_5th=quantile(AdditiveConcentration,0.05),
PercentileAdditive_95th=quantile(AdditiveConcentration,0.95)),by=c("ChemicalName","Cas")]

setkey(df.rollup1,ChemicalName,Cas)
setkey(df.rollup2,ChemicalName,Cas)
df.rollupfinal<-df.rollup1[df.rollup2]
df.rollupfinal[,NumberOfWellsPct:=paste(NumberOfWells,"
(",round(PercentOfWells*100,4),"%","sep=")]
df.rollupfinal[,NumberOfRecordsPct:=paste(NumberOfRecords," (",
round(PercentOfRecords*100,4),"%","sep=")]

df.rollupfinal[Cas!="",freqrank:=rank(-NumberOfWells,ties.method="min")]

df.rollupfinal[,c("Cas","MedianPercentOfFluid","PercentileFluid_5th",
"PercentileFluid_95th","MedianPercentAdditive","PercentileAdditive_5th","PercentileAdditive_95th"):=
list(paste("","Cas,sep="),
paste(MedianPercentOfFluid,"%",sep="),
paste(PercentileFluid_5th,"%",sep="),
paste(PercentileFluid_95th,"%",sep=),
```

```
paste(MedianPercentAdditive,"%",sep=""),
paste(PercentileAdditive_5th,"%",sep=""),
paste(PercentileAdditive_95th,"%",sep=""))]]

df.rollupfinal<-df.rollupfinal[order(freqrank)]
df.rollupfinal<-df.rollupfinal[freqrank<=20 | is.na(freqrank),]

df.out1<-df.rollupfinal[,list(ChemicalName,Cas,NumberOfWellsPct,
MedianPercentOfFluid,PercentileFluid_5th,PercentileFluid_95th,
NumberOfRecordsPct,MedianPercentAdditive,PercentileAdditive_5th, PercentileAdditive_95th)]
df.out1[ChemicalName=="",ChemicalName:="Total"]

write.csv(df.out1,"Table G-3.csv",row.names=FALSE)
```



## R Code to Reproduce Table G-4

**Table G-4 includes the twenty most frequently reported additive ingredients in Garfield County, Colorado, ranked by frequency of occurrence**

```
library(RODBC)
library(data.table)
conn<-odbcConnectAccess2007("FracFocus_03-2015.accdb")
df<-data.table(sqlQuery(conn,"SELECT * FROM [Table D-4 - Twenty Chemicals Garfield]"))
wellrecords<-length(unique(df$WellId))
ingredrecords<-length(df$IngredientId)
summarydf<-data.frame(WellRecordsUsed=wellrecords, IngredientRecordsUsed=ingredrecords)

setkey(df,ChemicalName,Cas,WellId)
df.well<-
df[,list(SumFluidConcentration=sum(FluidConcentration)),by=c("ChemicalName","Cas","WellId")]

setkey(df.well,ChemicalName,Cas)

df.rollup1<-df.well[,list(NumberOfWells=length(WellId),PercentOfWells=length(WellId)/wellrecords,
MedianPercentOfFluid=median(SumFluidConcentration),
PercentileFluid_5th=quantile(SumFluidConcentration,0.05),
PercentileFluid_95th=quantile(SumFluidConcentration,0.95)),by=c("ChemicalName","Cas")]

setkey(df,ChemicalName,Cas)

df.rollup2<-
df[,list(NumberOfRecords=length(IngredientId),PercentOfRecords=length(IngredientId)/ingredrecords,
MedianPercentAdditive=median(AdditiveConcentration),
PercentileAdditive_5th=quantile(AdditiveConcentration,0.05),
PercentileAdditive_95th=quantile(AdditiveConcentration,0.95)),by=c("ChemicalName","Cas")]

setkey(df.rollup1,ChemicalName,Cas)
setkey(df.rollup2,ChemicalName,Cas)
df.rollupfinal<-df.rollup1[df.rollup2]
df.rollupfinal[,NumberOfWellsPct:=paste(NumberOfWells,"
(",round(PercentOfWells*100,4),"%","sep=")]
df.rollupfinal[,NumberOfRecordsPct:=paste(NumberOfRecords," (",
round(PercentOfRecords*100,4),"%","sep=")]

df.rollupfinal[Cas!="",freqrank:=rank(-NumberOfWells,ties.method="min")]

df.rollupfinal[,c("Cas","MedianPercentOfFluid","PercentileFluid_5th",
"PercentileFluid_95th","MedianPercentAdditive","PercentileAdditive_5th","PercentileAdditive_95th"):=
list(paste("","Cas,sep="),
paste(MedianPercentOfFluid,"%",sep="),
paste(PercentileFluid_5th,"%",sep="),
paste(PercentileFluid_95th,"%",sep="),
```

```
paste(MedianPercentAdditive,"%",sep=""),
paste(PercentileAdditive_5th,"%",sep=""),
paste(PercentileAdditive_95th,"%",sep=""))]]

df.rollupfinal<-df.rollupfinal[order(freqrank)]
df.rollupfinal<-df.rollupfinal[freqrank<=20 | is.na(freqrank),]

df.out1<-df.rollupfinal[,list(ChemicalName,Cas,NumberOfWellsPct,
MedianPercentOfFluid,PercentileFluid_5th,PercentileFluid_95th,
NumberOfRecordsPct,MedianPercentAdditive,PercentileAdditive_5th, PercentileAdditive_95th)]
df.out1[ChemicalName=="",ChemicalName:="Total"]

write.csv(df.out1,"Table G-4.csv",row.names=FALSE)
```

## R Code to Reproduce Table G-5

**Table G-5 includes the twenty most frequently reported additive ingredients in Kern County, California, ranked by frequency of occurrence**

```
library(RODBC)
library(data.table)
conn<-odbcConnectAccess2007("FracFocus_03-2015.accdb")
df<-data.table(sqlQuery(conn,"SELECT * FROM [Table D-5 - Twenty Chemicals Kern]"))
wellrecords<-length(unique(df$WellId))
ingredrecords<-length(df$IngredientId)
summarydf<-data.frame(WellRecordsUsed=wellrecords, IngredientRecordsUsed=ingredrecords)

setkey(df,ChemicalName,Cas,WellId)
df.well<-
df[,list(SumFluidConcentration=sum(FluidConcentration)),by=c("ChemicalName","Cas","WellId")]

setkey(df.well,ChemicalName,Cas)

df.rollup1<-df.well[,list(NumberOfWells=length(WellId),PercentOfWells=length(WellId)/wellrecords,
MedianPercentOfFluid=median(SumFluidConcentration),
PercentileFluid_5th=quantile(SumFluidConcentration,0.05),
PercentileFluid_95th=quantile(SumFluidConcentration,0.95)),by=c("ChemicalName","Cas")]

setkey(df,ChemicalName,Cas)

df.rollup2<-
df[,list(NumberOfRecords=length(IngredientId),PercentOfRecords=length(IngredientId)/ingredrecords,
MedianPercentAdditive=median(AdditiveConcentration),
PercentileAdditive_5th=quantile(AdditiveConcentration,0.05),
PercentileAdditive_95th=quantile(AdditiveConcentration,0.95)),by=c("ChemicalName","Cas")]

setkey(df.rollup1,ChemicalName,Cas)
setkey(df.rollup2,ChemicalName,Cas)
df.rollupfinal<-df.rollup1[df.rollup2]
df.rollupfinal[,NumberOfWellsPct:=paste(NumberOfWells,"
(",round(PercentOfWells*100,4),"%","sep="")]
df.rollupfinal[,NumberOfRecordsPct:=paste(NumberOfRecords," (",
round(PercentOfRecords*100,4),"%","sep=")]

df.rollupfinal[Cas!="",freqrank:=rank(-NumberOfWells,ties.method="min")]

df.rollupfinal[,c("Cas","MedianPercentOfFluid","PercentileFluid_5th",
"PercentileFluid_95th","MedianPercentAdditive","PercentileAdditive_5th","PercentileAdditive_95th"):=
list(paste("","Cas,sep=""),
paste(MedianPercentOfFluid,"%",sep=""),
paste(PercentileFluid_5th,"%",sep=""),
paste(PercentileFluid_95th,"%",sep="),
```

```
paste(MedianPercentAdditive,"%",sep=""),
paste(PercentileAdditive_5th,"%",sep=""),
paste(PercentileAdditive_95th,"%",sep=""))]]

df.rollupfinal<-df.rollupfinal[order(freqrank)]
df.rollupfinal<-df.rollupfinal[freqrank<=20 | is.na(freqrank),]

df.out1<-df.rollupfinal[,list(ChemicalName,Cas,NumberOfWellsPct,
MedianPercentOfFluid,PercentileFluid_5th,PercentileFluid_95th,
NumberOfRecordsPct,MedianPercentAdditive,PercentileAdditive_5th, PercentileAdditive_95th)]
df.out1[ChemicalName=="",ChemicalName:="Total"]

write.csv(df.out1,"Table G-5.csv",row.names=FALSE)
```

## R Code to Reproduce Table H-1

**Table H-1 includes total water volumes, summarized by county**

```
library(RODBC)
library(data.table)
conn<-odbcConnectAccess2007("FracFocus_03-2015.accdb")
df<-data.table(sqlQuery(conn,"SELECT * FROM [Table A-1 - County Water Usage]"))

part1<-df[,list(NumDisclosures_ValidVolume=length(WellId),
               Water.Cumulative=sum(VolumeFFQA,na.rm=TRUE),
               Volume.Median=median(VolumeFFQA,na.rm=TRUE),
               Water.5th=quantile(VolumeFFQA,0.05),
               Water.95th=quantile(VolumeFFQA,0.95)),
          by=list(State,County)]
part2<-df[,list(State="Entire Dataset",
               County="",
               NumDisclosures_ValidVolume=length(WellId),
               Water.Cumulative=sum(VolumeFFQA,na.rm=TRUE),
               Volume.Median=median(VolumeFFQA,na.rm=TRUE),
               Water.5th=quantile(VolumeFFQA,0.05),
               Water.95th=quantile(VolumeFFQA,0.95)),]

part1<-part1[order(-NumDisclosures_ValidVolume)]
county.out<-rbind(part1,part2)

write.csv(county.out,"Table H-1.csv",row.names=FALSE)
```

## R Code to Reproduce Figure 2

Figure 2 includes the geographic distribution of disclosures in the project database

Note that the symbology of the Figure2 column in the output is as follows:

-2: All of the specified county's wells have an unconfirmed location.

-1: The specified county has no wells in the FracFocus database.

**Values > 0:** The number of confirmed location wells in the specified county.

```
library(RODBC)
library(data.table)
conn<-odbcConnectAccess2007("FracFocus_03-2015.accdb")
df<-data.table(sqlQuery(conn,"SELECT * FROM
[Maps]",as.is=c(TRUE,TRUE,TRUE,TRUE,FALSE,FALSE,FALSE,FALSE)))
county.out<-df[,list(Confirmed=sum(ifelse(!is.na(WellId) & (AllCountyOK==TRUE | (AllCountyOK==FALSE
& StateName=='Alaska')),1,0)),Unconfirmed=sum(ifelse(!is.na(WellId) & AllCountyOK==FALSE &
StateName!='Alaska',1,0))),by=list(FIPS,CountyName,StateName)]
county.out[,Figure2:=ifelse(Confirmed>0,Confirmed,ifelse(Confirmed==0 & Unconfirmed==0,-1,-2))]
county.out[,c("Confirmed","Unconfirmed"):=list(NULL,NULL)]
write.csv(county.out,"Figure 2.csv",row.names=FALSE)
```

## R Code to Reproduce Figure 3

**Figure 3 includes the geographic distribution of disclosures by production type**

Note that the symbology of the Figure4 column in the output is as follows:

**-2:** All of the specified county's wells have an unconfirmed location.

**-1:** The specified county has no wells in the FracFocus database.

**Values from 0-1:** The percentage of the specified county's confirmed wells that are gas wells.

```
library(RODBC)
library(data.table)
conn<-odbcConnectAccess2007("FracFocus_03-2015.accdb")
df<-data.table(sqlQuery(conn,"SELECT * FROM
[Maps]",as.is=c(TRUE,TRUE,TRUE,TRUE,FALSE,FALSE,FALSE,FALSE)))
county.out<-df[,list(Confirmed_Gas=sum(ifelse(!is.na(WellId) & (AllCountyOK==TRUE |
(AllCountyOK==FALSE & StateName=='Alaska')) &
TypeFFQA=="Gas",1,0)),Confirmed_Oil=sum(ifelse(!is.na(WellId) & (AllCountyOK==TRUE |
(AllCountyOK==FALSE & StateName=='Alaska')) &
TypeFFQA=="Oil",1,0)),Unconfirmed=sum(ifelse(!is.na(WellId) & AllCountyOK==FALSE &
StateName!='Alaska',1,0))),by=list(FIPS,CountyName,StateName)]
county.out[,Figure4:=ifelse(Confirmed_Gas>0 |
Confirmed_Oil>0,Confirmed_Gas/(Confirmed_Gas+Confirmed_Oil),ifelse(Confirmed_Gas==0 &
Confirmed_Oil==0 & Unconfirmed==0,-1,-2))]
county.out[,c("Confirmed_Gas","Confirmed_Oil","Unconfirmed"):=list(NULL,NULL,NULL)]
write.csv(county.out,"Figure 3.csv",row.names=FALSE)
```

## R Code to Reproduce Figure 5

**Figure 5 includes the cumulative total water volumes, summarized by county**

Note that the symbology of the Figure 5 column in the output is as follows:

**-3:** All of the specified county's confirmed location wells have invalid or missing volume data.

**-2:** All of the specified county's wells have an unconfirmed location.

**-1:** The specified county has no wells in the FracFocus database.

**Values > 0:** The cumulative (valid) volume for the specified county's confirmed location wells.

```
library(RODBC)
library(data.table)
conn<-odbcConnectAccess2007("FracFocus_03-2015.accdb")
df<-data.table(sqlQuery(conn,"SELECT * FROM
[Maps]",as.is=c(TRUE,TRUE,TRUE,TRUE,FALSE,FALSE,FALSE,FALSE)))
county.out<-df[,list(Confirmed_Location=sum(ifelse(!is.na(WellId) & (AllCountyOK==TRUE |
(AllCountyOK==FALSE & StateName=='Alaska')),1,0)),Unconfirmed_Location=sum(ifelse(!is.na(WellId) &
AllCountyOK==FALSE & StateName!='Alaska',1,0)),Confirmed_Volume=sum(ifelse(!is.na(WellId) &
(AllCountyOK==TRUE | (AllCountyOK==FALSE & StateName=='Alaska')) &
!is.na(Volume),1,0)),Cumulative_Volume=sum(ifelse(!is.na(WellId) & (AllCountyOK==TRUE |
(AllCountyOK==FALSE & StateName=='Alaska')) &
!is.na(Volume),Volume,0))),by=list(FIPS,CountyName,StateName)]
county.out[,Figure5:=ifelse(Confirmed_Location==0 & Unconfirmed_Location==0,-
1,ifelse(Confirmed_Location==0 & Unconfirmed_Location > 0,-2,ifelse(Confirmed_Location>0 &
Confirmed_Volume==0,-3,Cumulative_Volume)))]
county.out[,c("Confirmed_Location","Unconfirmed_Location","Confirmed_Volume","Cumulative_Volum
e"):=list(NULL,NULL,NULL,NULL)]
write.csv(county.out,"Figure 5.csv",row.names=FALSE)
```



## R Code to Reproduce Figure 6

**Figure 6 includes the median total water volumes per disclosure, summarized by county**

Note that the symbology of the Figure6 column in the output is as follows:

-3: All of the specified county's confirmed location wells have invalid or missing volume data.

-2: All of the specified county's wells have an unconfirmed location.

-1: The specified county has no wells in the FracFocus database.

**Values > 0:** The median (valid) volume for the specified county's confirmed location wells.

```
library(RODBC)
library(data.table)
conn<-odbcConnectAccess2007("FracFocus_03-2015.accdb")
df<-data.table(sqlQuery(conn,"SELECT * FROM
[Maps]",as.is=c(TRUE,TRUE,TRUE,TRUE,FALSE,FALSE,FALSE,FALSE)))
county.out<-df[,list(Confirmed_Location=sum(ifelse(!is.na(WellId) & (AllCountyOK==TRUE |
(AllCountyOK==FALSE & StateName=='Alaska')),1,0)), Unconfirmed_Location=sum(ifelse(!is.na(WellId) &
AllCountyOK==FALSE & StateName!='Alaska',1,0)),Confirmed_Volume=sum(ifelse(!is.na(WellId) &
(AllCountyOK==TRUE | (AllCountyOK==FALSE & StateName=='Alaska')) &
!is.na(Volume),1,0)),Median_Volume=median(ifelse(!is.na(WellId) & (AllCountyOK==TRUE |
(AllCountyOK==FALSE & StateName=='Alaska')) &
!is.na(Volume),Volume,as.double(NA)),na.rm=TRUE)),by=list(FIPS,CountyName,StateName)]
county.out[,Figure6:=ifelse(Confirmed_Location==0 & Unconfirmed_Location==0,-
1,ifelse(Confirmed_Location==0 & Unconfirmed_Location > 0,-2,ifelse(Confirmed_Location > 0 &
Confirmed_Volume==0,-3,Median_Volume)))]
county.out[,c("Confirmed_Location","Unconfirmed_Location","Confirmed_Volume","Median_Volume"):
=list(NULL,NULL,NULL,NULL)]
write.csv(county.out,"Figure 6.csv",row.names=FALSE)
```

## R Code to Reproduce Figure 7

**Figure 7 includes the variability in reported total water volumes per disclosure, as measured by the difference between the 5th and 95th percentiles**

Note that the symbology of the Figure7 column in the output is as follows:

**-3:** All of the specified county's confirmed location wells have invalid or missing volume data.

**-2:** All of the specified county's wells have an unconfirmed location.

**-1:** The specified county has no wells in the FracFocus database.

**Values > 0:** The median (valid) volume for the specified county's confirmed location wells.

```
library(RODBC)
library(data.table)
conn<-odbcConnectAccess2007("FracFocus_03-2015.accdb")
df<-data.table(sqlQuery(conn,"SELECT * FROM
[Maps]",as.is=c(TRUE,TRUE,TRUE,TRUE,FALSE,FALSE,FALSE,FALSE)))
county.out<-df[,list(Confirmed_Location=sum(ifelse(!is.na(WellId) & (AllCountyOK==TRUE |
(AllCountyOK==FALSE & StateName=='Alaska')),1,0)), Unconfirmed_Location=sum(ifelse(!is.na(WellId) &
AllCountyOK==FALSE & StateName!='Alaska',1,0)),Confirmed_Volume=sum(ifelse(!is.na(WellId) &
(AllCountyOK==TRUE | (AllCountyOK==FALSE & StateName=='Alaska'))) &
!is.na(Volume),1,0)),Range=quantile(ifelse(!is.na(WellId) & (AllCountyOK==TRUE | (AllCountyOK==FALSE
& StateName=='Alaska'))) & !is.na(Volume),Volume,as.double(NA)),0.95,na.rm=TRUE)-
quantile(ifelse(!is.na(WellId) & (AllCountyOK==TRUE | (AllCountyOK==FALSE & StateName=='Alaska'))) &
!is.na(Volume),Volume,as.double(NA)),0.05,na.rm=TRUE)),by=list(FIPS,CountyName,StateName)]
county.out[,Figure7:=ifelse(Confirmed_Location==0 & Unconfirmed_Location==0,-
1,ifelse(Confirmed_Location==0 & Unconfirmed_Location > 0,-2,ifelse(Confirmed_Location > 0 &
Confirmed_Volume==0,-3,Range)))]
county.out[,c("Confirmed_Location","Unconfirmed_Location","Confirmed_Volume","Range"):=list(NULL
,NULL,NULL,NULL)]
write.csv(county.out,"Figure 7.csv",row.names=FALSE)
```