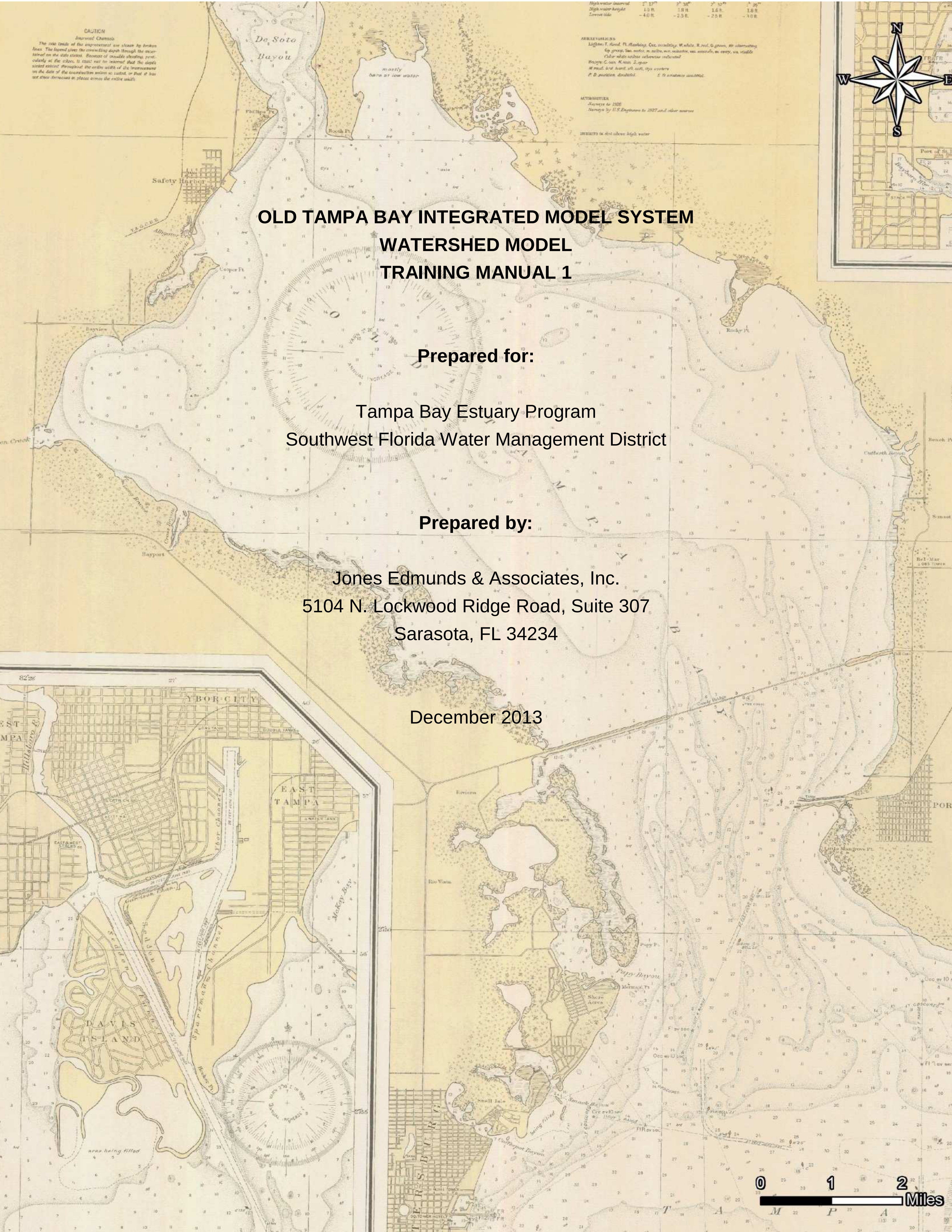
APPENDIX A1

WATERSHED MODEL
TRAINING MANUAL



OLD TAMPA BAY INTEGRATED MODEL SYSTEM
WATERSHED MODEL
TRAINING MANUAL 1

Prepared for:

Tampa Bay Estuary Program
Southwest Florida Water Management District

Prepared by:

Jones Edmunds & Associates, Inc.
5104 N. Lockwood Ridge Road, Suite 307
Sarasota, FL 34234

December 2013

INTRODUCTION

This training manual has been prepared to support the first training session for the Old Tampa Bay Integrated Model System Watershed Model. The objectives of the first training are:

- Provide instruction to the user group on the Watershed Model structure and how to set up and run the watershed model for the OTB Integrated Model System
- Provide instructions post processing steps to produce graphics and statistics

Based on these objectives, the manual provides specific descriptions and instructions for the time frame for the OTB Integrated Model System (2000-2009) and is not a general user's manual. The first part of the training manual steps through setting up the model files, running the model and post processing information for the hydrology portion of the model. The second part of the training manual steps through the water quality model component development and post processing.

The instructions for the hydrology are presented in three primary steps, these are;

- Step 1: Setting up the Watershed Model
- Step 2: Running the Watershed Model
- Step 3: Watershed Model post-processing for calibration graphics and statistics

STEP 1: SETTING UP THE WATERSHED MODEL

Some of the coding that integrates the three components requires a specific file placement, naming, and setup. Below are the preliminary steps to:

- Configuring the Software Environment
- Installing the Model Files
- Understanding the Modeling Components

A. CONFIGURING THE SOFTWARE ENVIRONMENT

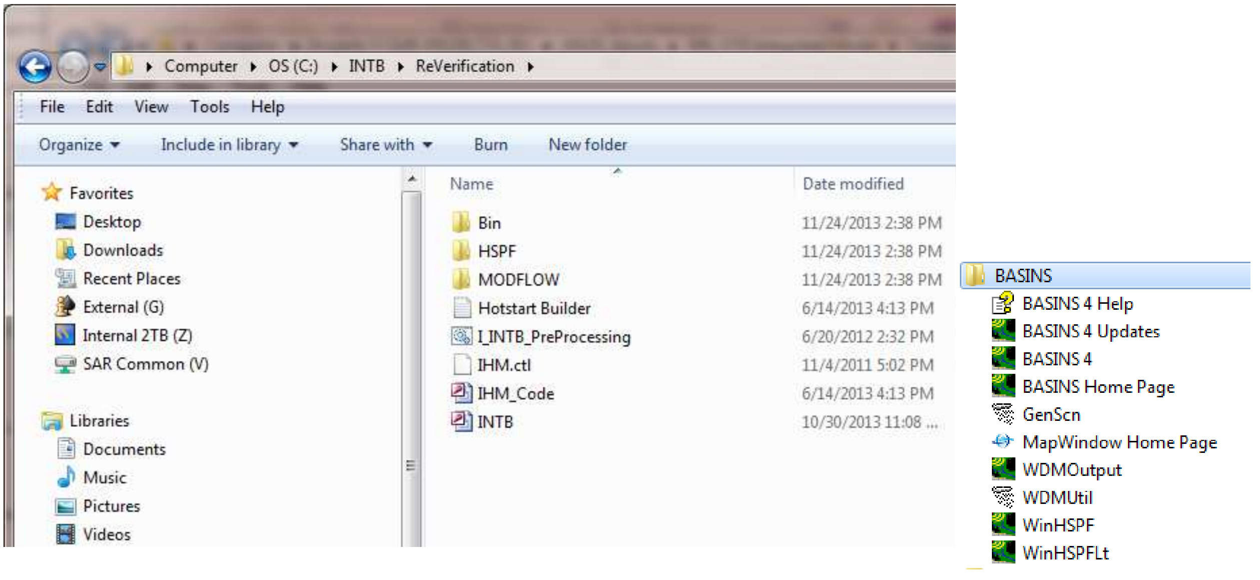
Before installing any model files set up the software environments as follows:

1. Install Microsoft Components
   a. .NET Framework v4.5
   b. SQL Server 2012 Express LocalDB
2. Confirm Microsoft Access 2003 or later (32-bit version if 2010 or later)
3. Install EPA BASINS 4.0 or later
4. Install Golden Software's Grapher v10.0 or later. Using machine administrator login, register Grapher.

B. INSTALLING THE MODEL FILES

1. Model Files Directory

To minimize model run time and minimize errors due to file path length limitations, start by creating a working directory on the local drive (C:\INTB). In the electronic training materials is a folder 4_IHM_Install\ReVerification. Copy the ReVerification folder only to the local drive (C:\INTB).



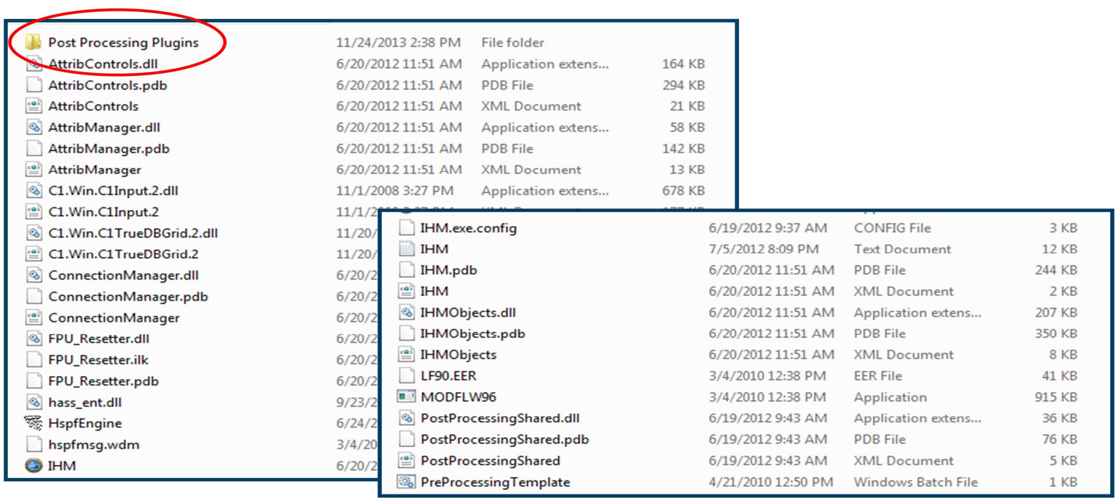
The necessary folders\files included in the folder are:

❖ **Bin folder:** contains executable files for the three model components, necessary processing files, and the IHM interface executable file to start the model run

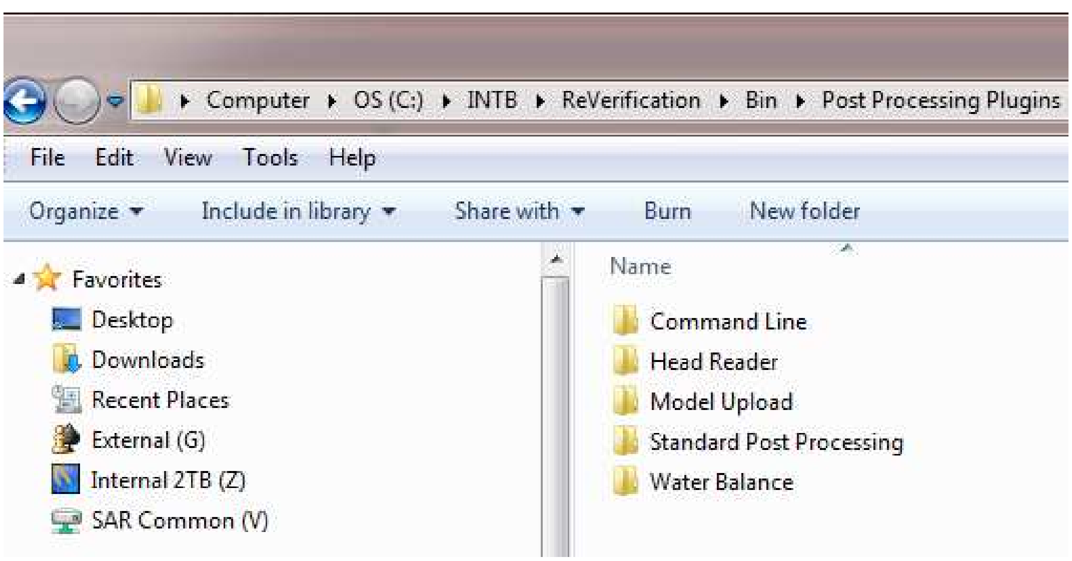
- ❖ **HSPF folder:** contains four user control input files (.uci) and six water data management (.wdm) files
- ❖ **MODFLOW folder:**contains files necessary to run the MODFLOW application interfaced with IHM
- ❖ **I_INTB_PreProcessing.bat:** automatically initializes predetermined processes needed to start the model
- ❖ **IHM.ctl:**control file
- ❖ **IHM_Code.mdb:** contains integration code for the interface between the 3 model components. This file is not accessible to the user.
- ❖ **INTB.mdb:** contains modeling parameters, reference tables, look up tables, and observed data for reverification

<u>2. Post Processing Plugins:</u>

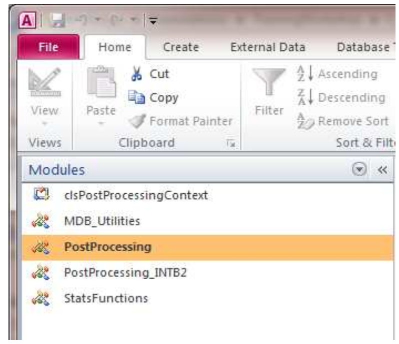
a. Navigate to the Bin Folder and open the Post Processing Plugin folder.



b. Each folder contains a .exe file (Standard Post Processing contains 2 .exe files) that need to be opened the prior to the first model run on a system.



c. Double click and open the following .exe files, no action necessary, just close the file.

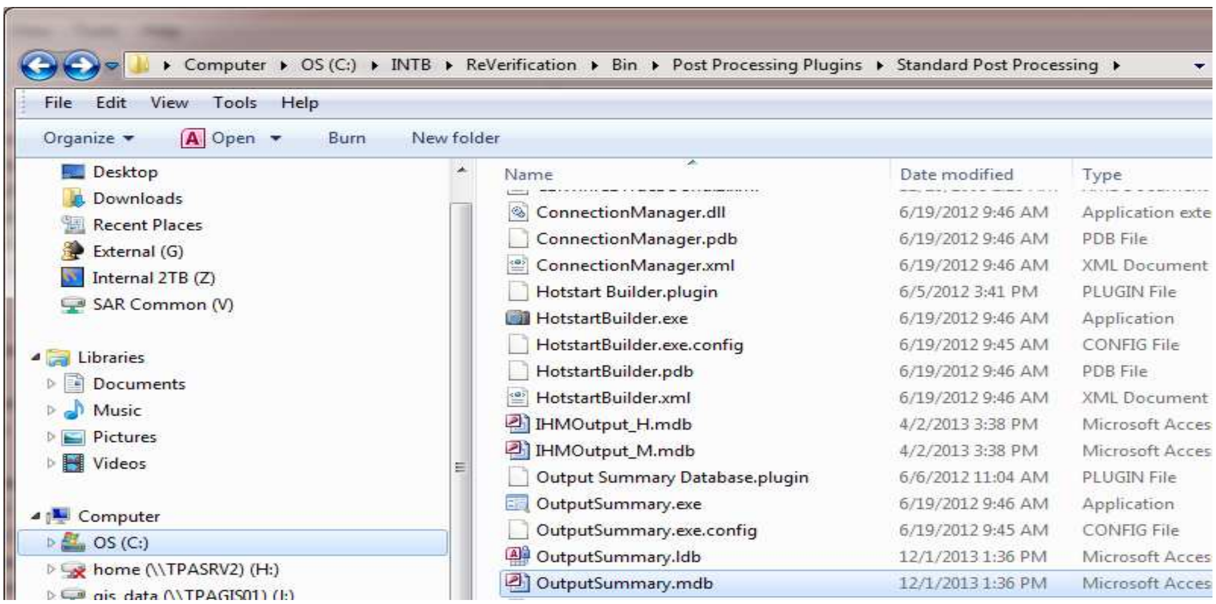


is

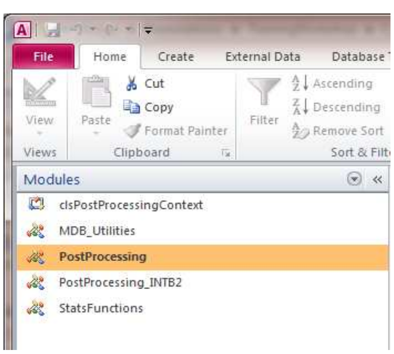
- Command Line
- Head Reader Plugin
- Model Upload
- HotStart Builder
- Output Summary
- Water Balance

3. Default Debugging:

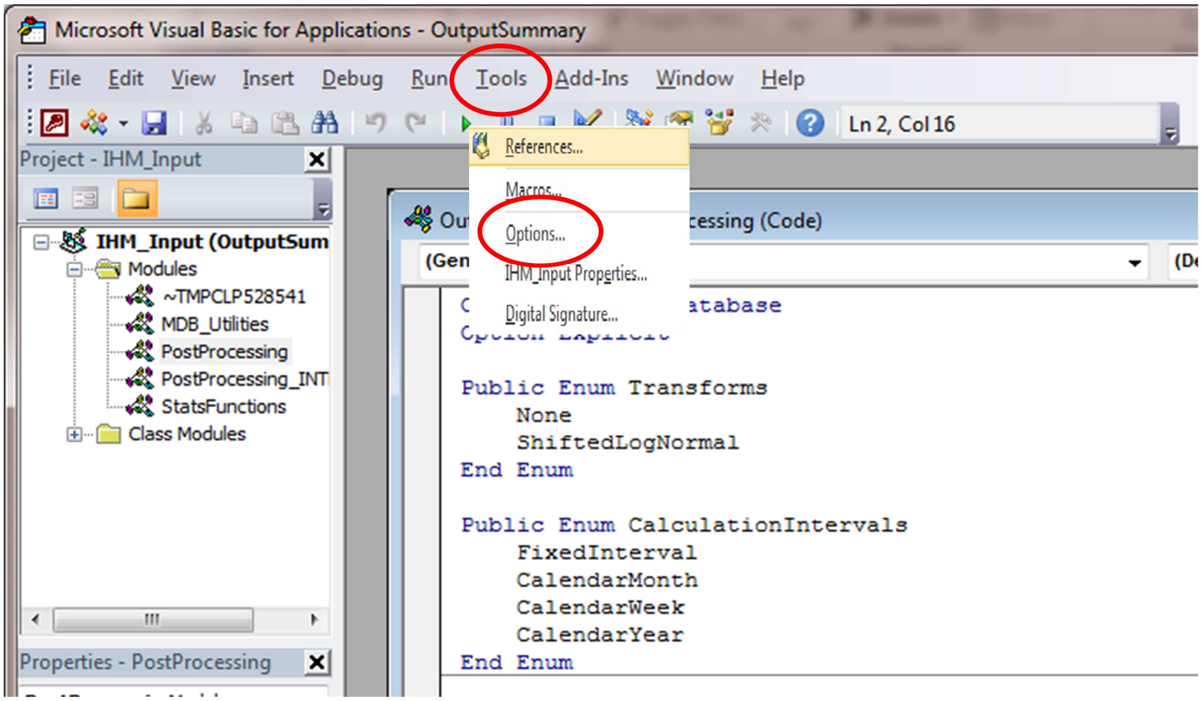
a. In the ReVerification run folder, open "Bin\Post Processing Plugins\Standard Post Processing\OutputSummary.mdb".



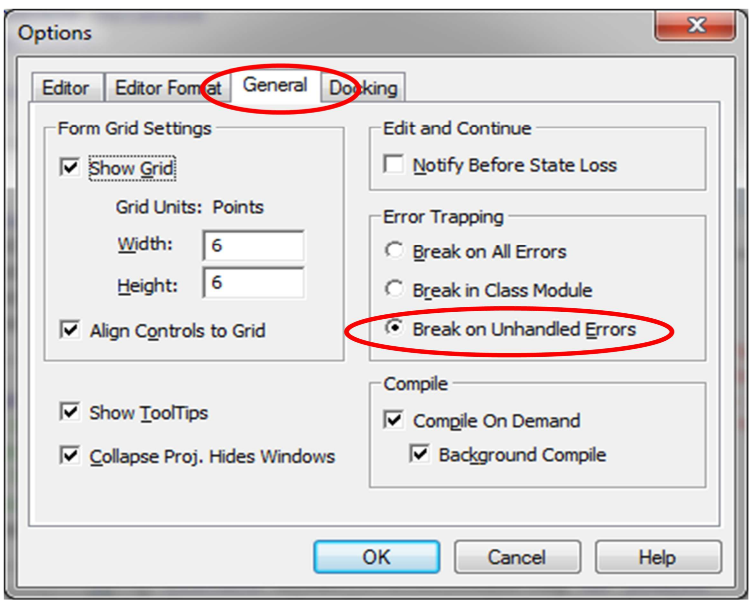
b. Double click on Post Processing to open any of the code scripts.



c. On the menu bar, go to Tools\Options

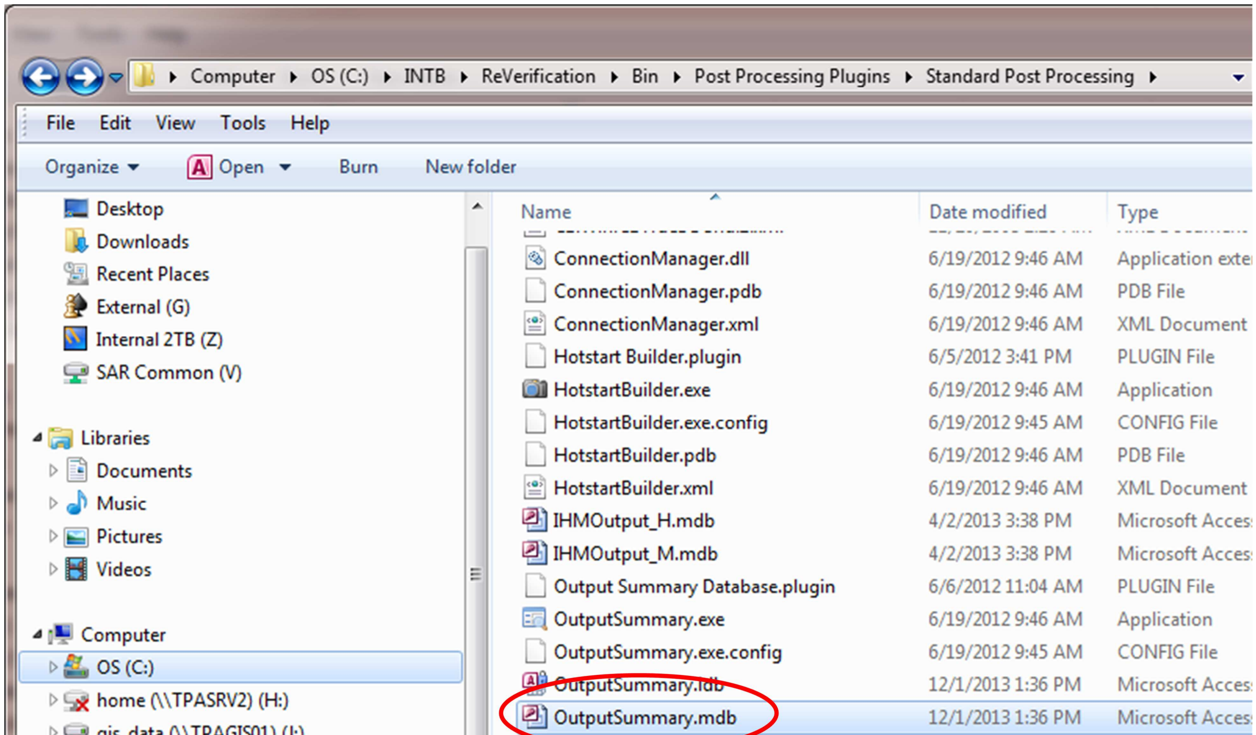


d. Move to the “General” tab. Check “Break on Unhandled Errors” and then dismiss the Options dialog.

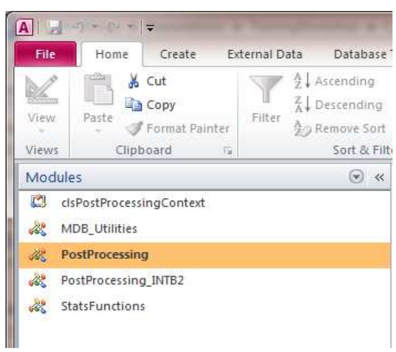


<u>4. Grapher Library</u>

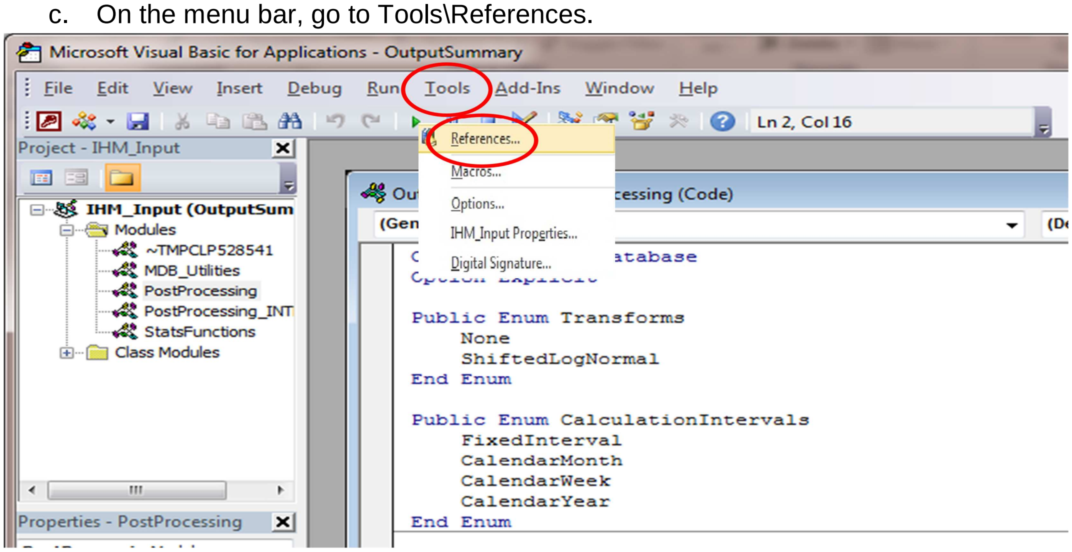
a. In the root of the ReVerification run folder, open “Bin\Post Processing Plugins\Standard Post Processing\OutputSummary.mdb”.



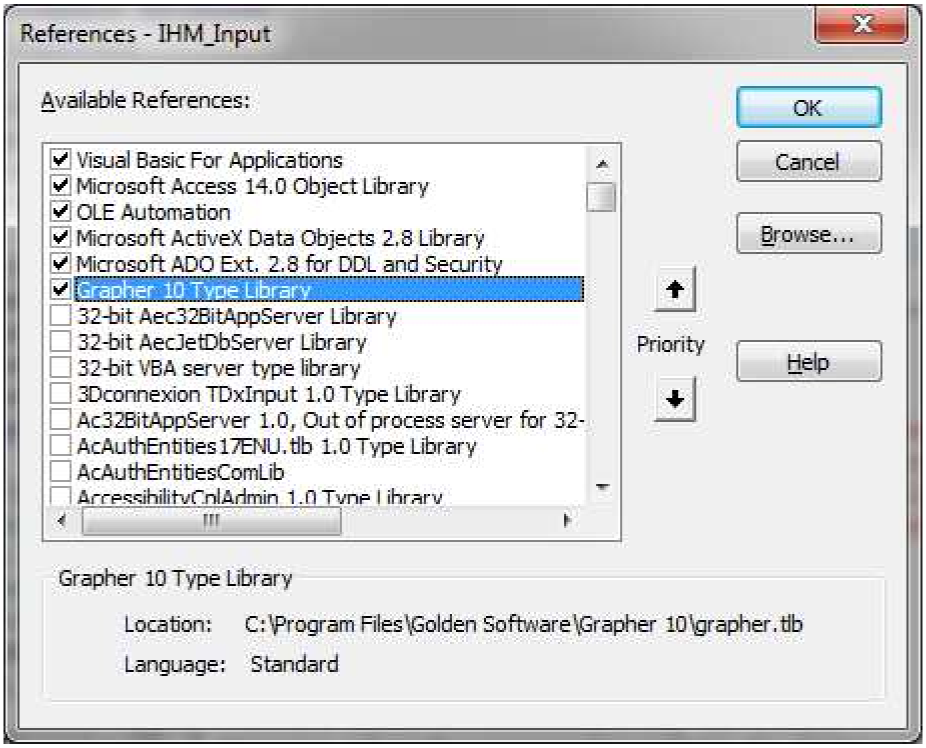
b. Double click on Post Processing to open any of the code scripts.



c. On the menu bar, go to Tools\References.

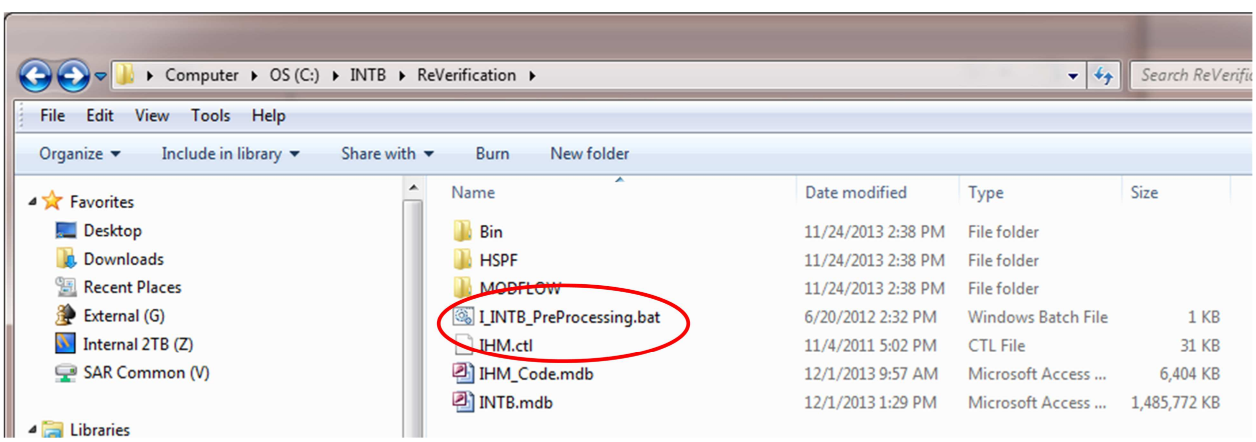


d. Check the Grapher library that is consistent with the install version of Grapher. You may find that a different version of Grapher is checkmarked with "MISSING" printed next to it. Uncheck this version of the Grapher library first. Then, scroll through the list of references to find the installed version of Grapher and checkmark it. Exit the database when finished.



5. Batch File

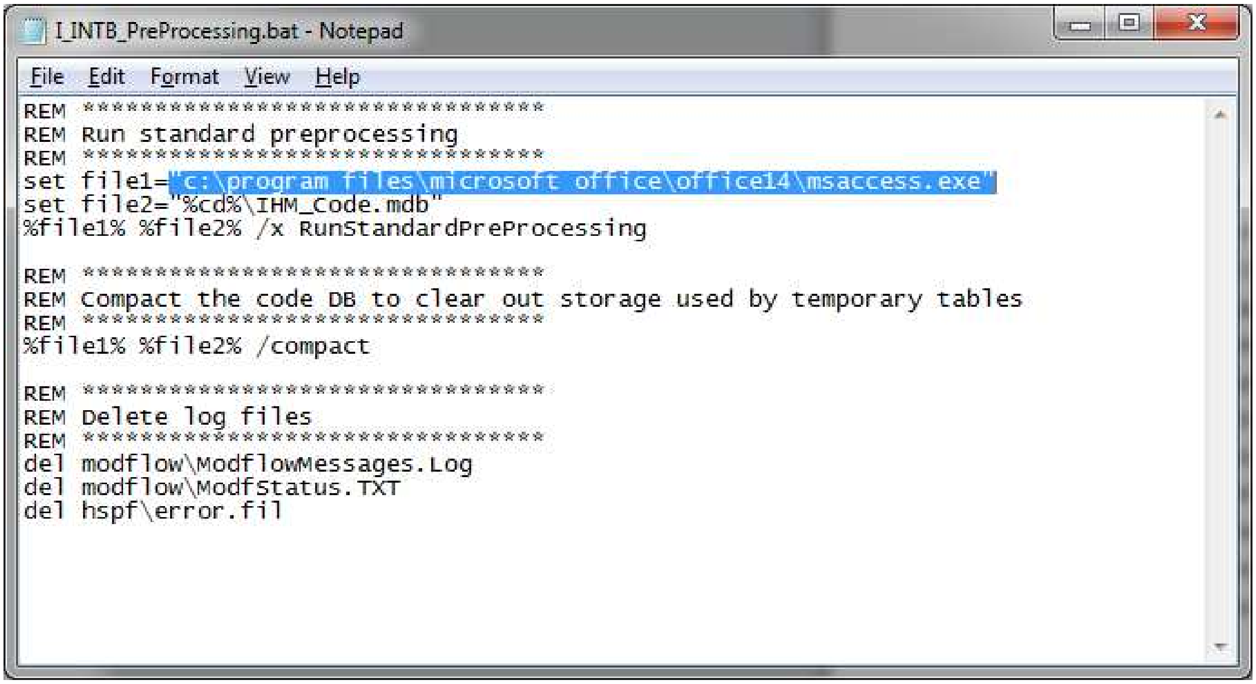
a. Navigate to the installation file set up folder and open the .bat file in a text editor (ie Notebook).



b. Set the file 1 equal to the program file location for Microsoft Access executable file.

Note: .mdb is the file extension for earlier versions of Access
.accdb is the file extension for newer versions of Access
Use .mbd for this model

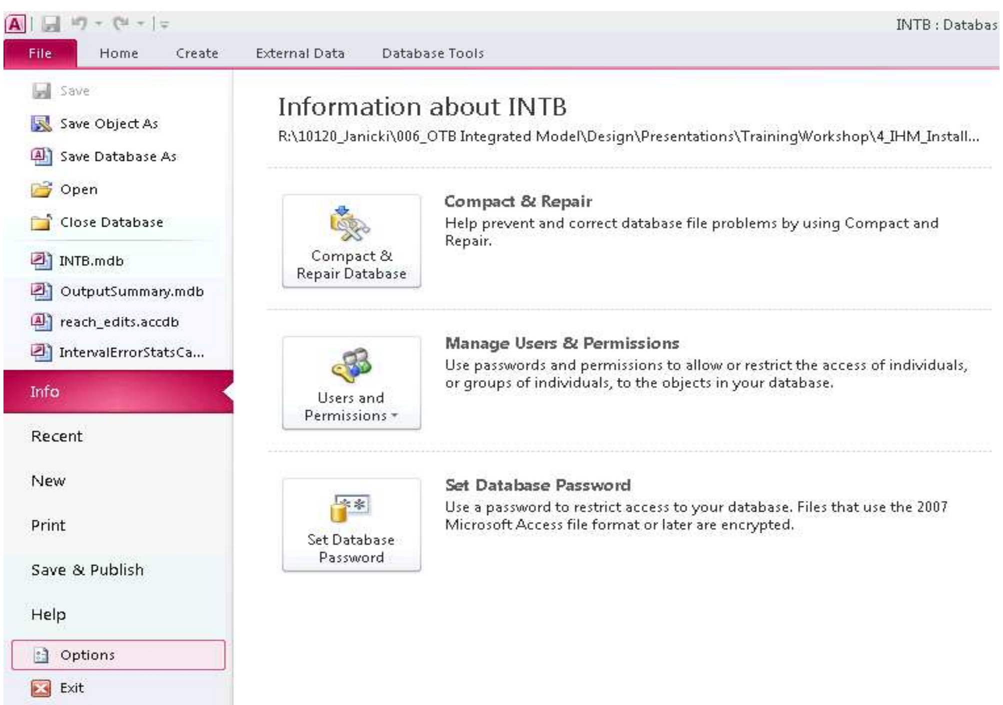
Note: Microsoft Office Suite must be 32-bit, not 64-bit



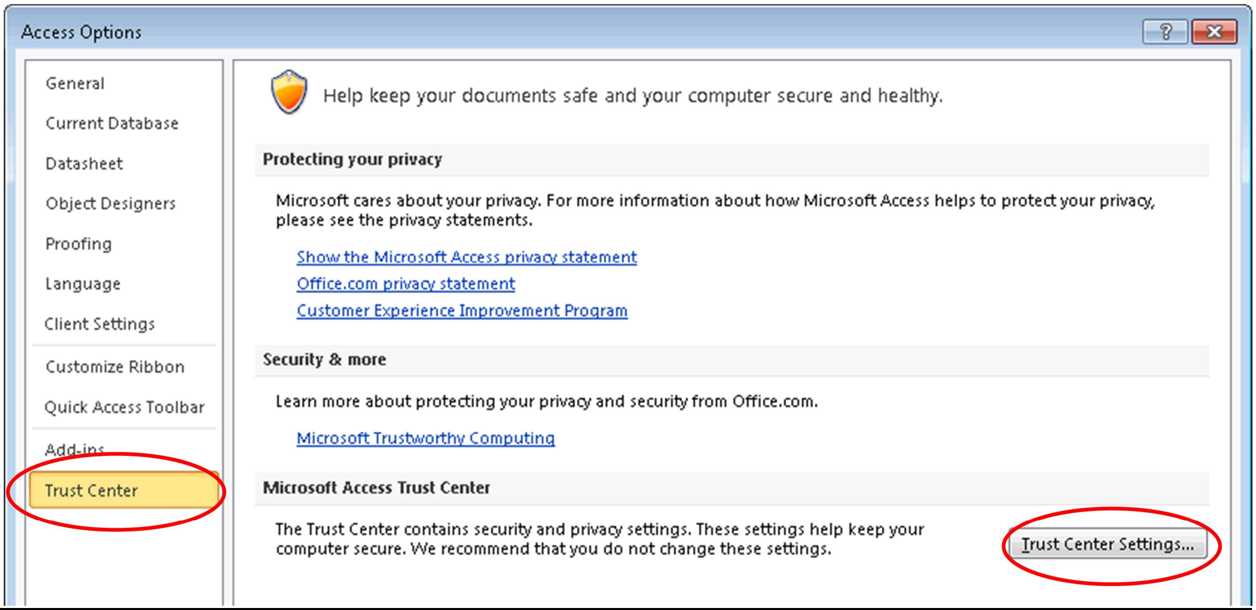
6. Modeling Database

Prior to executing the simulation for the first time, Trust Security Settings need to be set.

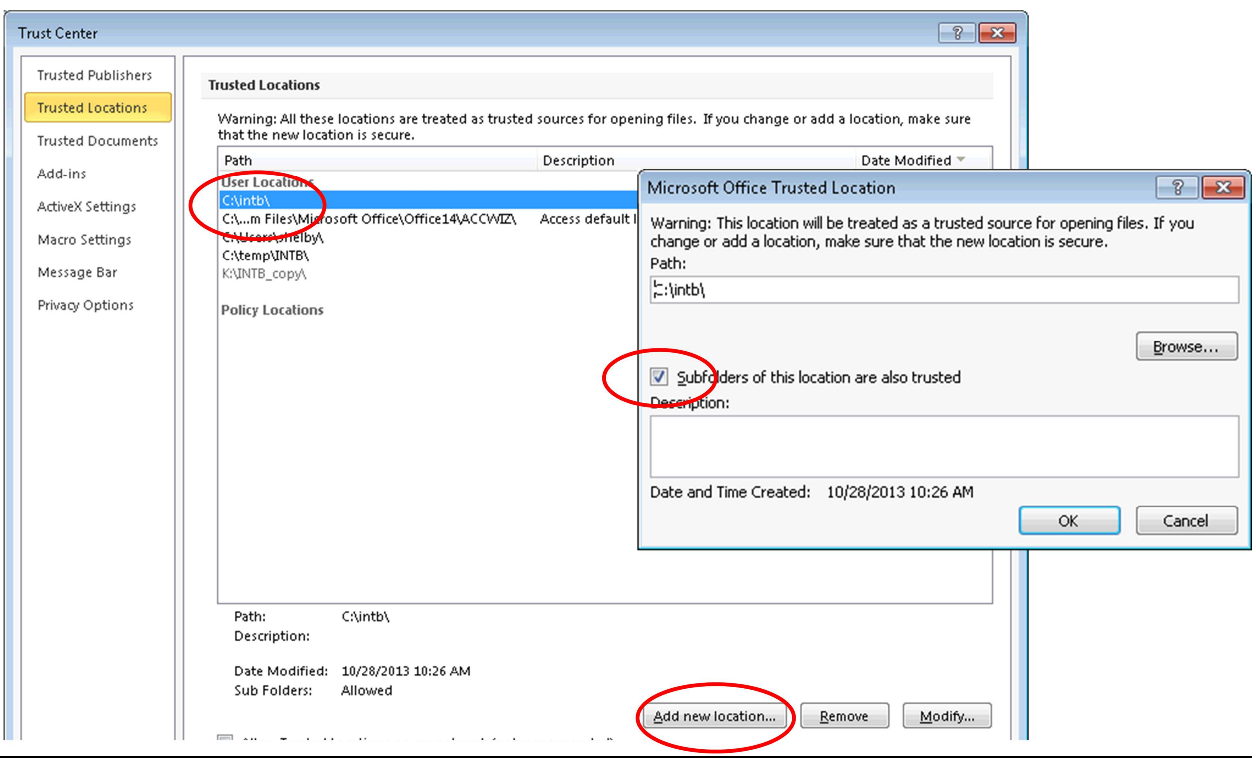
a. Open the INTB.mdb database and click the file button in the top left corner and click Options.



b. Click on Trust Center then Trust Center Settings.



c. Click on Trusted Locations and add a new location. Once added check the Subfolders of this location area also trusted.

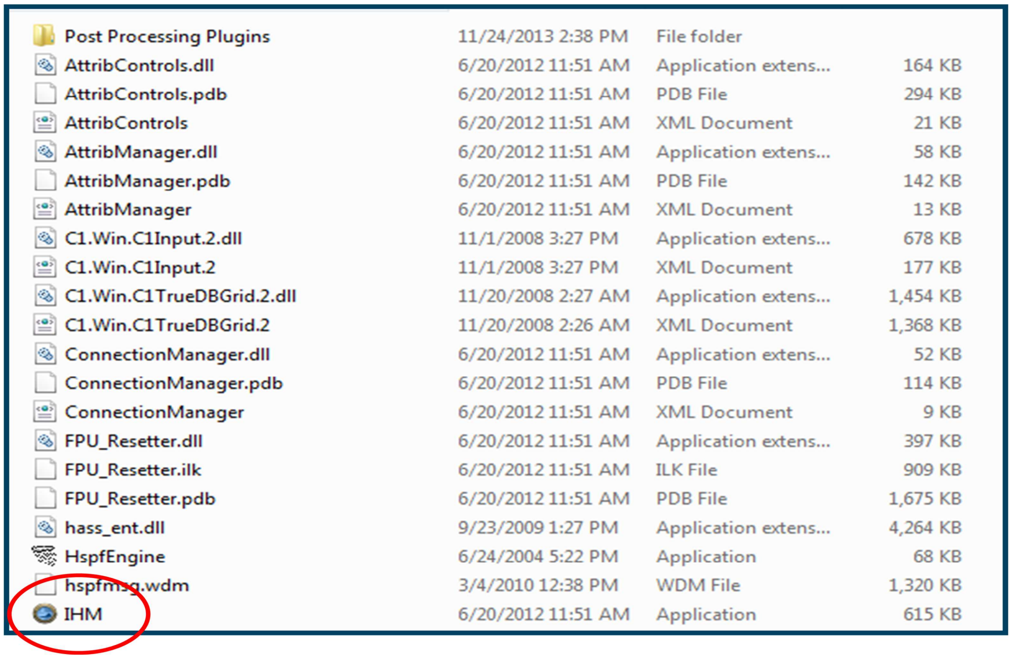


Note: *MS Access 2007 or 2010*: Add the working directory as a trusted location in the Access Trust Center: File\Options\Trust Center, click on the “Trust Center Settings” button. Be sure to checkmark the “Subfolders of the location are also trusted” box on the Trusted Location configuration form.
MS Access 2003: Set security to low under the Access menu: Tools\Macros\Security.

STEP 2: RUNNING THE MODEL

A. GRAPHICAL INTERFACE

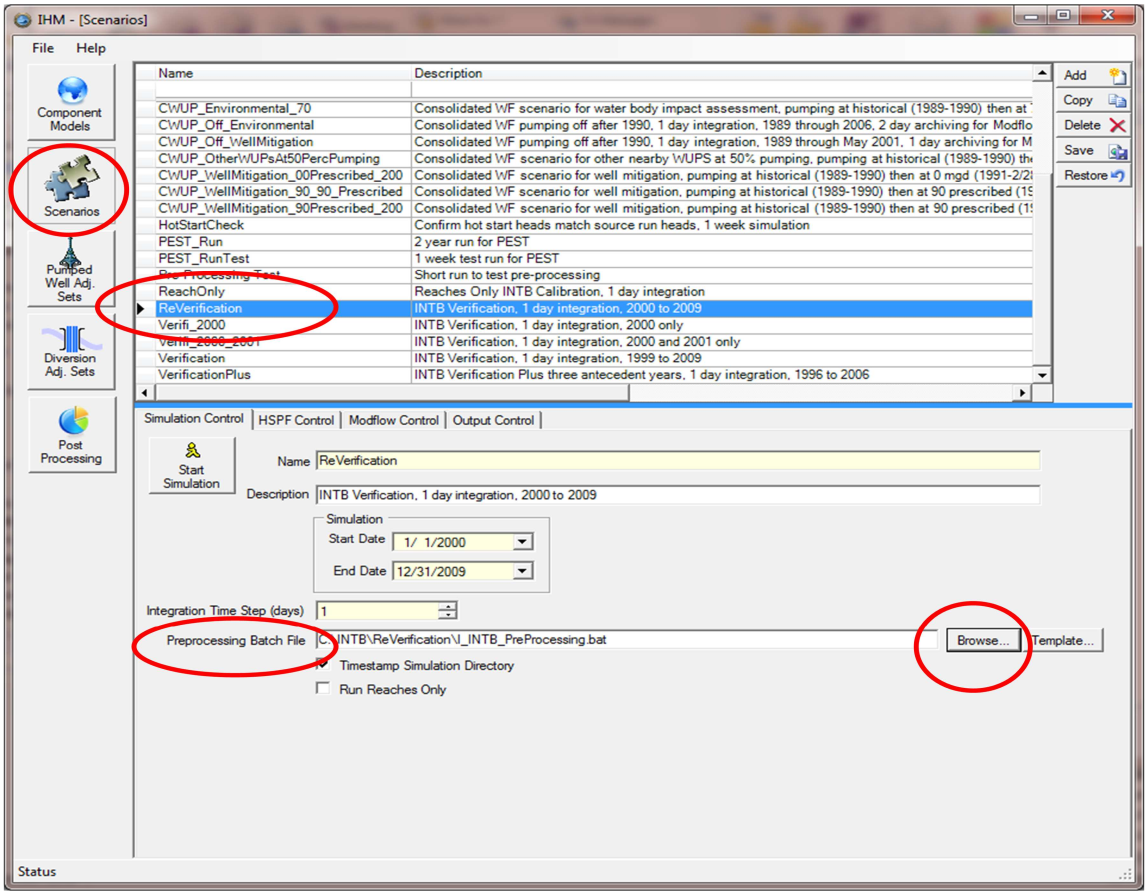
1. Navigate to the INTB model scenario working directory (typically in the C: drive). Open the “bin” folder and double click on “IHM.exe” to open the IHM interface.



Post Processing Plugins	11/24/2013 2:38 PM	File folder	
AttribControls.dll	6/20/2012 11:51 AM	Application extens...	164 KB
AttribControls.pdb	6/20/2012 11:51 AM	PDB File	294 KB
AttribControls	6/20/2012 11:51 AM	XML Document	21 KB
AttribManager.dll	6/20/2012 11:51 AM	Application extens...	58 KB
AttribManager.pdb	6/20/2012 11:51 AM	PDB File	142 KB
AttribManager	6/20/2012 11:51 AM	XML Document	13 KB
C1.Win.C1Input.2.dll	11/1/2008 3:27 PM	Application extens...	678 KB
C1.Win.C1Input.2	11/1/2008 3:27 PM	XML Document	177 KB
C1.Win.C1TrueDBGrid.2.dll	11/20/2008 2:27 AM	Application extens...	1,454 KB
C1.Win.C1TrueDBGrid.2	11/20/2008 2:26 AM	XML Document	1,368 KB
ConnectionManager.dll	6/20/2012 11:51 AM	Application extens...	52 KB
ConnectionManager.pdb	6/20/2012 11:51 AM	PDB File	114 KB
ConnectionManager	6/20/2012 11:51 AM	XML Document	9 KB
FPU_Resetter.dll	6/20/2012 11:51 AM	Application extens...	397 KB
FPU_Resetter.ilk	6/20/2012 11:51 AM	ILK File	909 KB
FPU_Resetter.pdb	6/20/2012 11:51 AM	PDB File	1,675 KB
hass_ent.dll	9/23/2009 1:27 PM	Application extens...	4,264 KB
HspfEngine	6/24/2004 5:22 PM	Application	68 KB
hspfmsg.wdm	3/4/2010 12:38 PM	WDM File	1,320 KB
IHM	6/20/2012 11:51 AM	Application	615 KB

B. LOADING THE SIMULATION

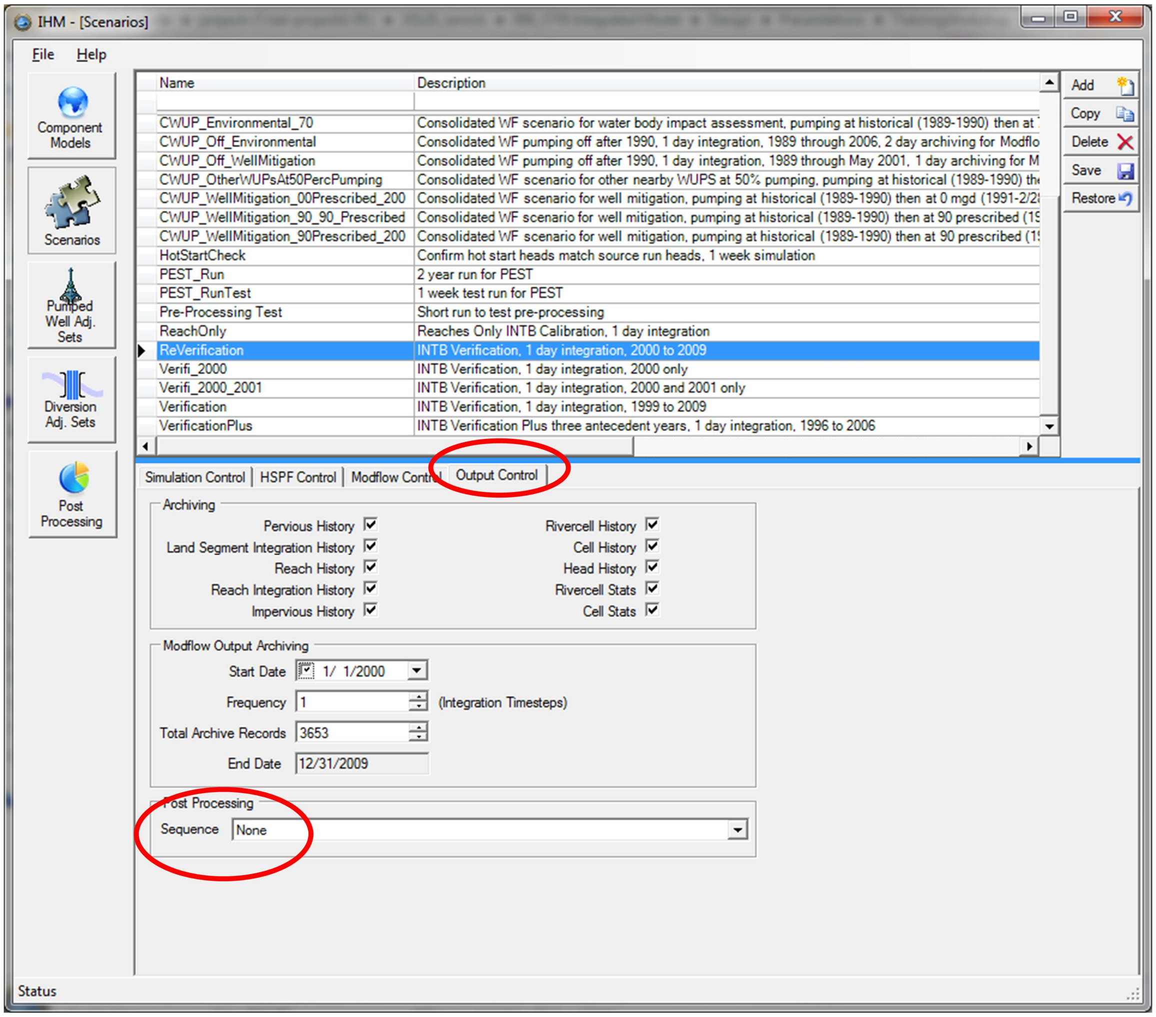
1. At the menu bar, use File\Open to open the modeling database. On the left hand side of the interface click on the “Scenarios” button. Within the box at the top half of the interface, select the name “ReVerification” from the list of possible simulation scenarios.



2. On the bottom half of the interface, click on the Simulation Control Tab and browse to the preprocessing batch file shown above.

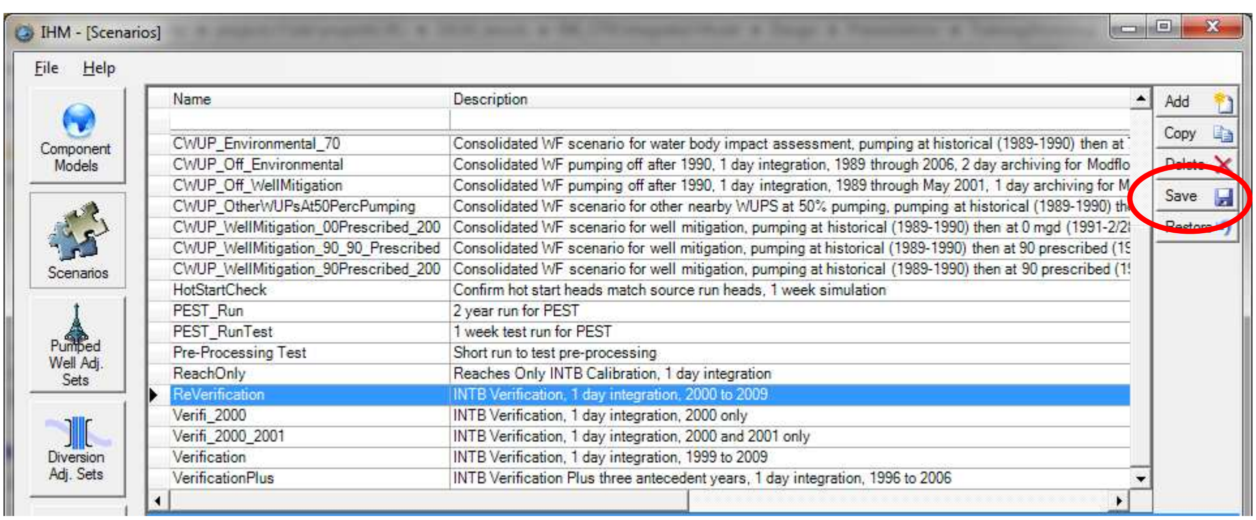
C. Output Control Settings

1. On the bottom half of the interface, click on the Output Control Tab and set the Sequence to None for the first simulation.

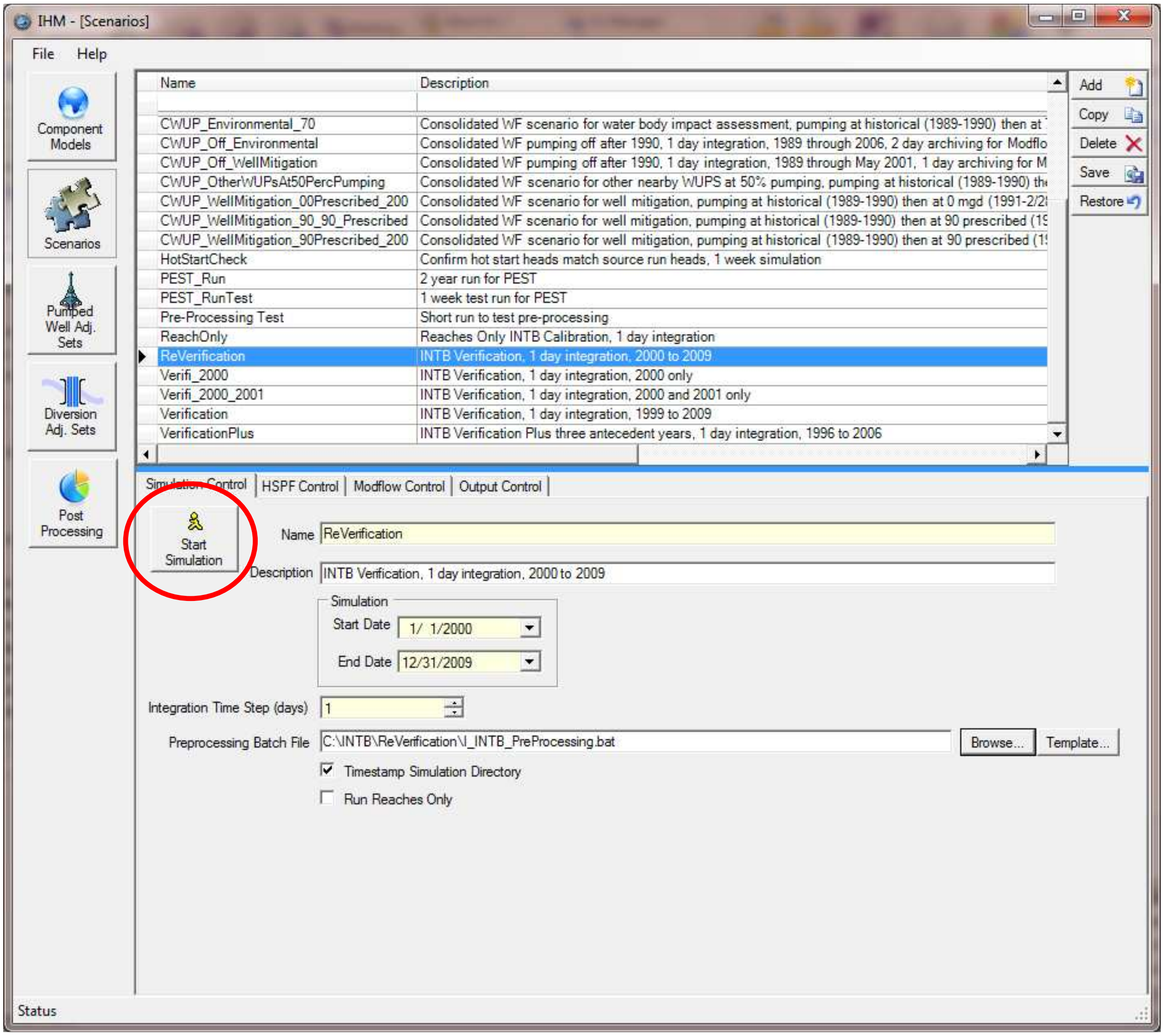


C. EXECUTE THE MODEL

1. Save the simulation scenario.



2. On the "Simulation Control" tab at the bottom half of the interface, click on "Start Simulation" which will begin the simulation and open a log screen.



Note: The simulation will create a number of temporary and output files in the working directory including a log file anda series of output files and databases. The temporary and output files can occupy a very large amount of disk space depending on the input data set and the configuration options selected.

STEP 3: POST PROCESSING

Once the model run is complete, the IHM model will begin the “Model Upload” sequence. This sequence will produce errors because the IHM program was not designed to handle the changes in the .UCI files and model database necessary to run the water quality model. Steps 1-17 provide a work around to reach the end of the Hydrology Post Processing.

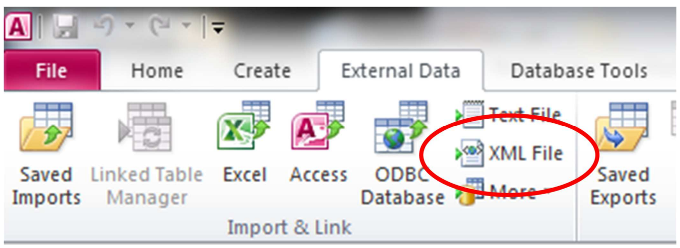
A. DATABASE REVISIONS

1. Navigate to the model run folder (i.e. C:\INTB\ReVerification\ReVerification_YearMonthDayTime)

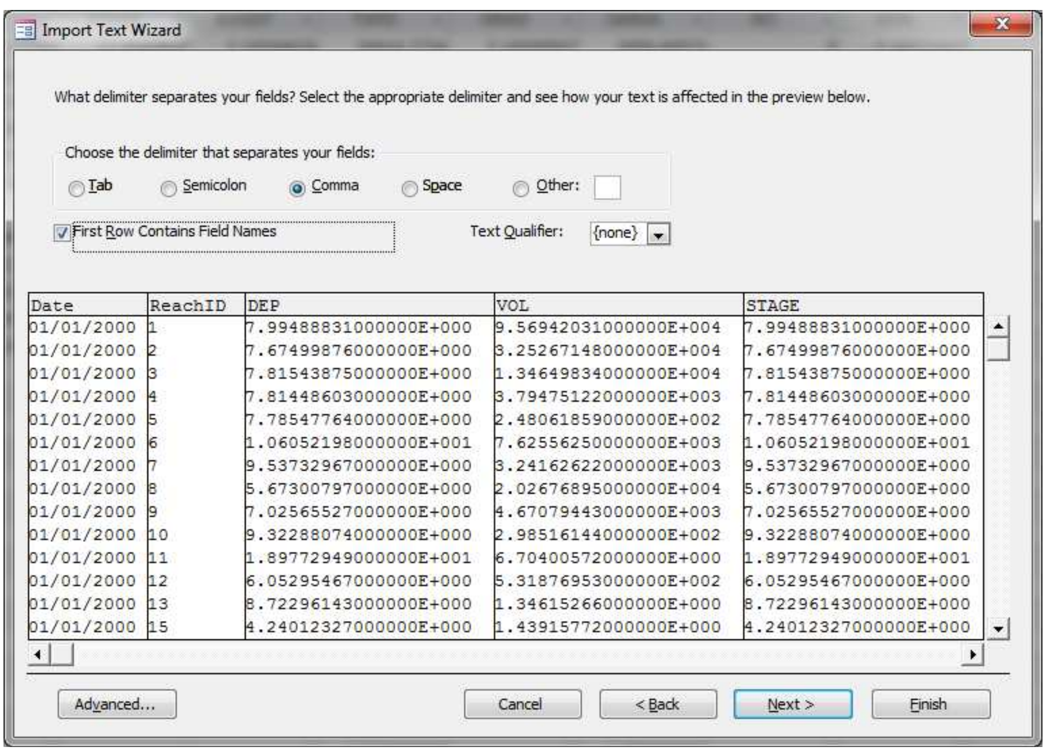
2. Create a new access database

3. Make backup copies of ReachHistory.csv and ReachIntegrationHistory.csv.

4. Import ReachHistory.csv and ReachIntegrationHistory.csv into new access tables.



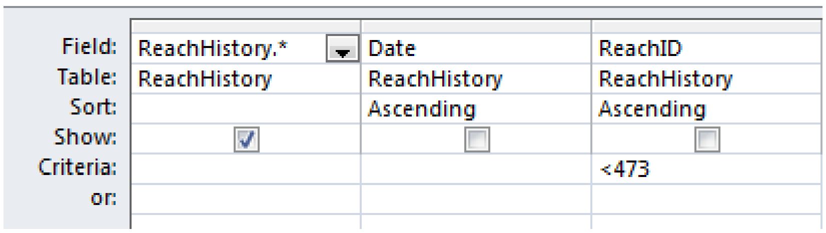
5. In the Import Text Wizard select “First Row Contains Field Names” and “No primary key” then click “Finish”



6. Create a new query in Access

7. Open the query in SQL view and paste the following statement and save.:

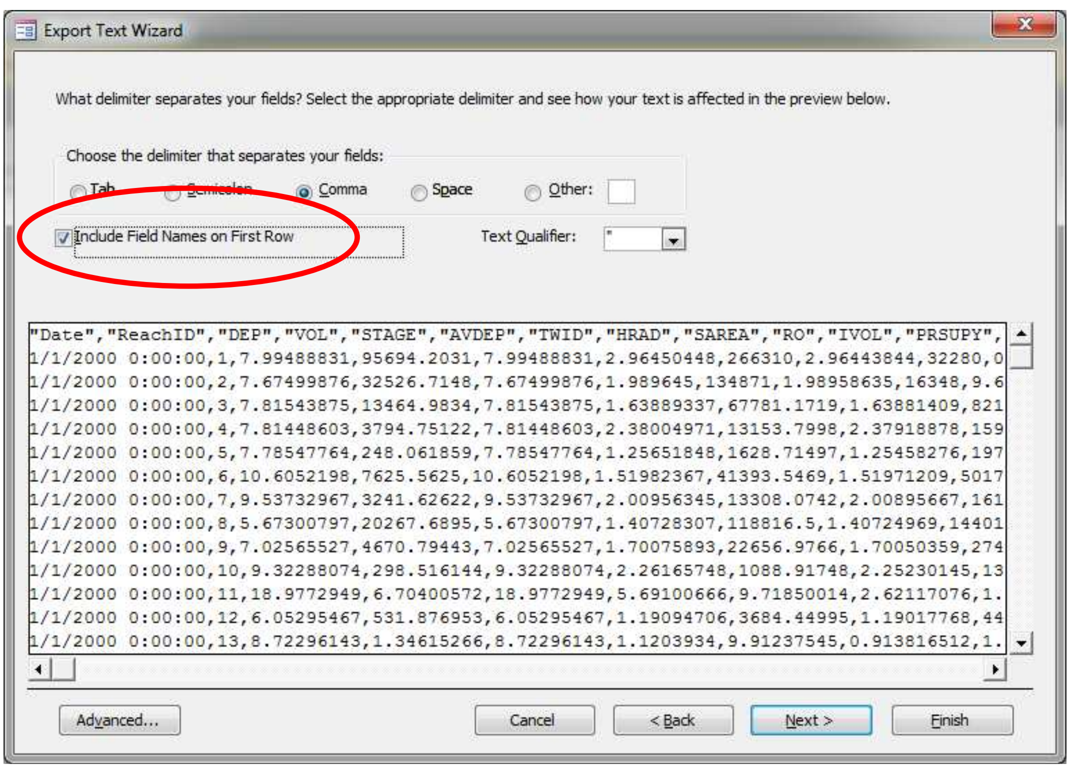
```
SELECT ReachHistory.*
FROM ReachHistory
WHERE (((ReachHistory.ReachID)<473))
ORDER BY ReachHistory.Date, ReachHistory.ReachID;
```

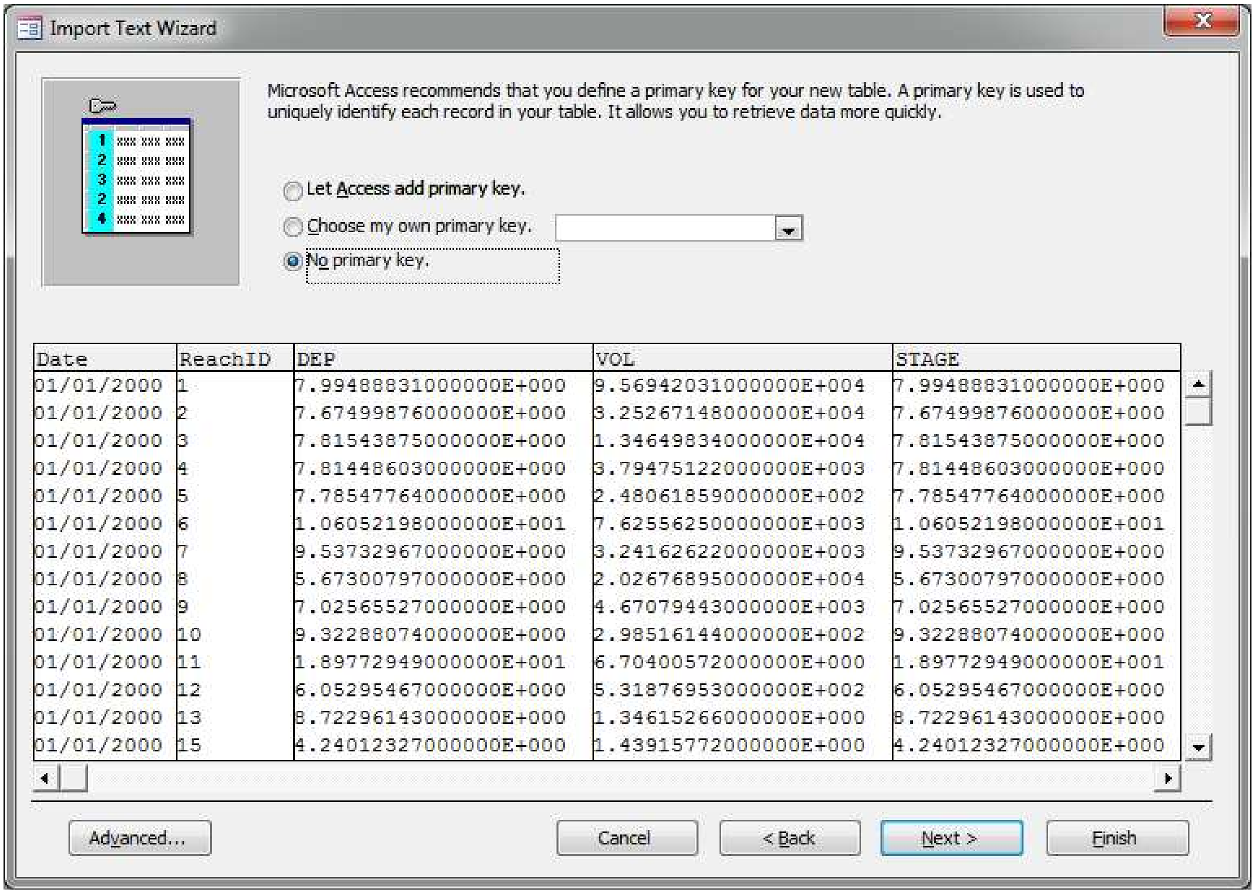


8. Right Click on the query and select export to text.file

9. Browse to and save over the original ReachHistory.csv

10. In the Export Text Wizard, select “Include Field Names on First Row”then click “Finish”.





11. Create a new query in Access

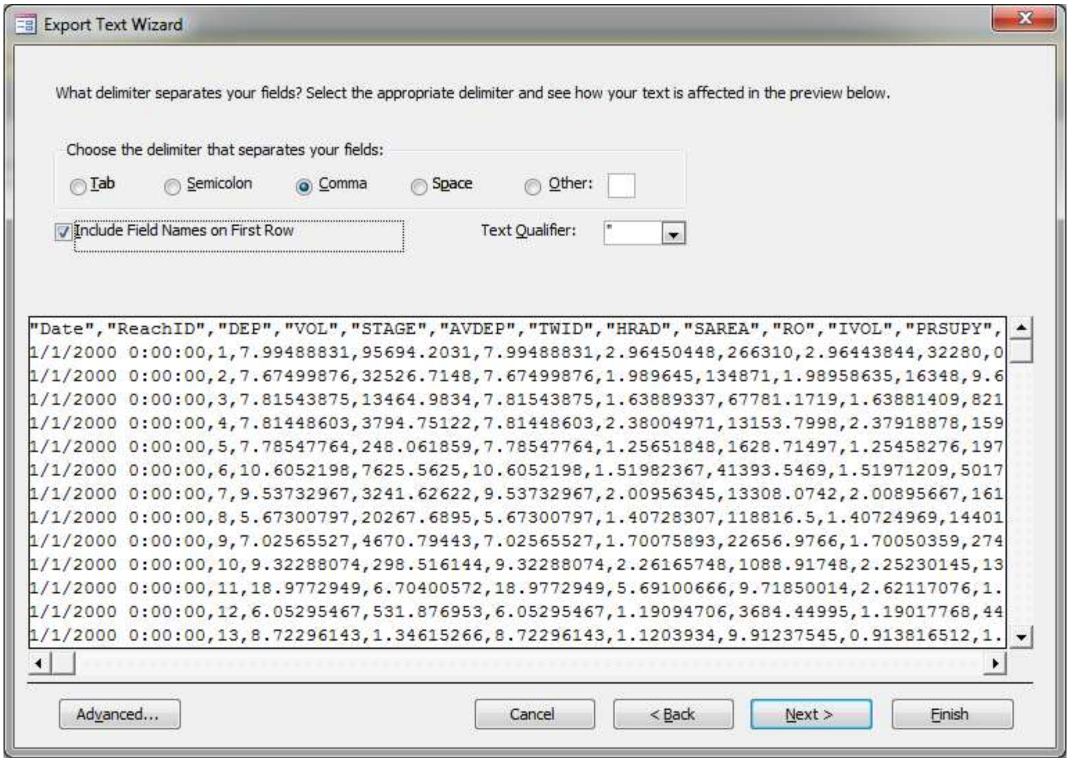
12. Open the query in SQL view and paste the following statement and save.:

SELECT ReachIntegrationHistory.*
FROM ReachIntegrationHistory
WHERE (((ReachIntegrationHistory.ReachID)<473))
ORDER BY ReachIntegrationHistory.Date, ReachIntegrationHistory.ReachID;

13. Right Click on the query and select export to text.file

14. Browse to and save over the original ReachIntegrationHistory.csv

15. In the Export Text Wizard, select “Include Field Names on First Row” then click “Finish”.



16. Close MS Access and navigate back to the model run folder (i.e. C:\INTB\ReVerification\ReVerification_YearMonthDayTime)

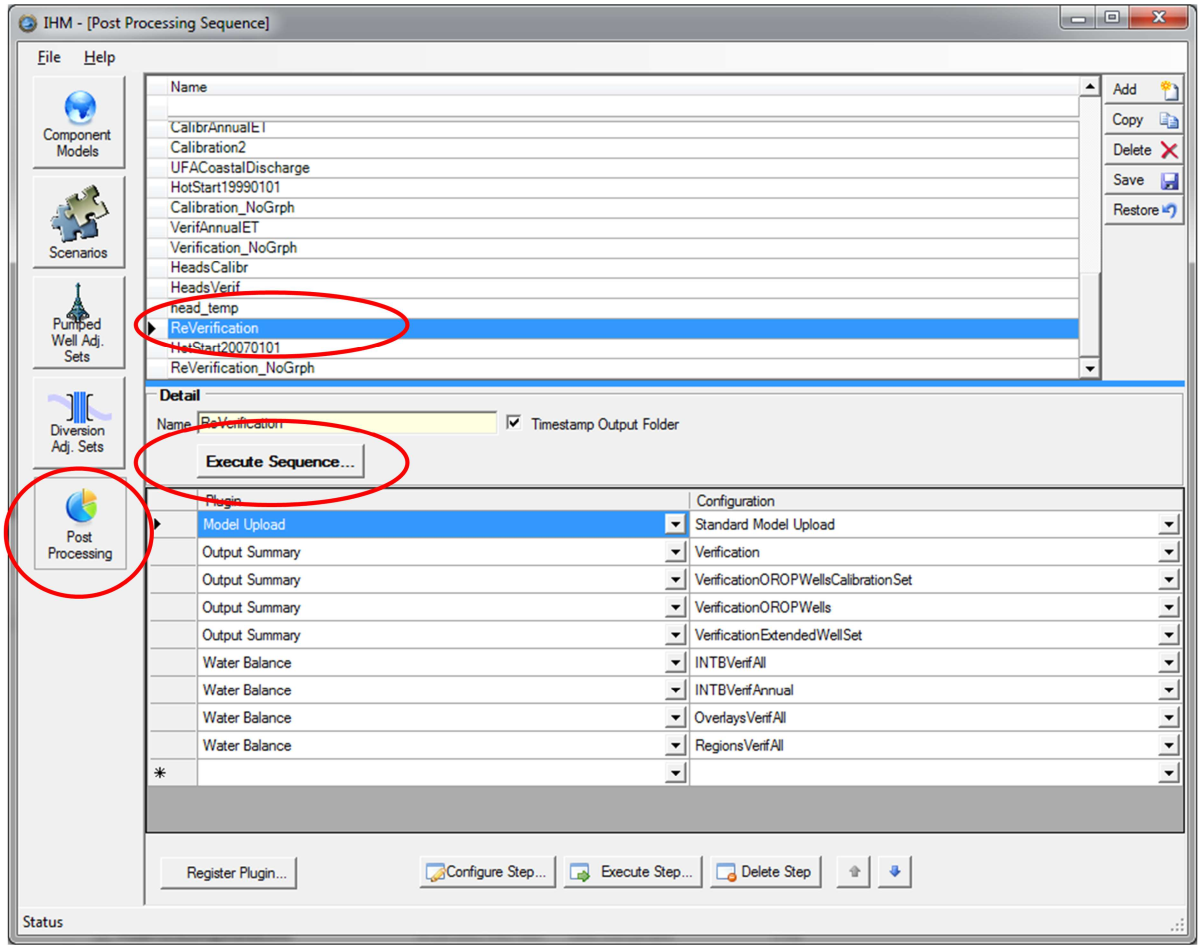
17. Delete the IHMOutput_H.mdb, IHMOutput_M.mdb, and OutputSummary.mdb files.

B. HYDROLOGY

To complete the post processing results for the hydrology model:

1. Navigate to the INTB model scenario working directory.

2. Open the “bin” folder and execute “IHM.exe” to open the IHM interface.



3. At the menu bar, use File\Open to open the model input database (the file "INTB.mdb" within the ReVerification scenario working directory). On the left hand side of the interface click on the "Post Processing" button.

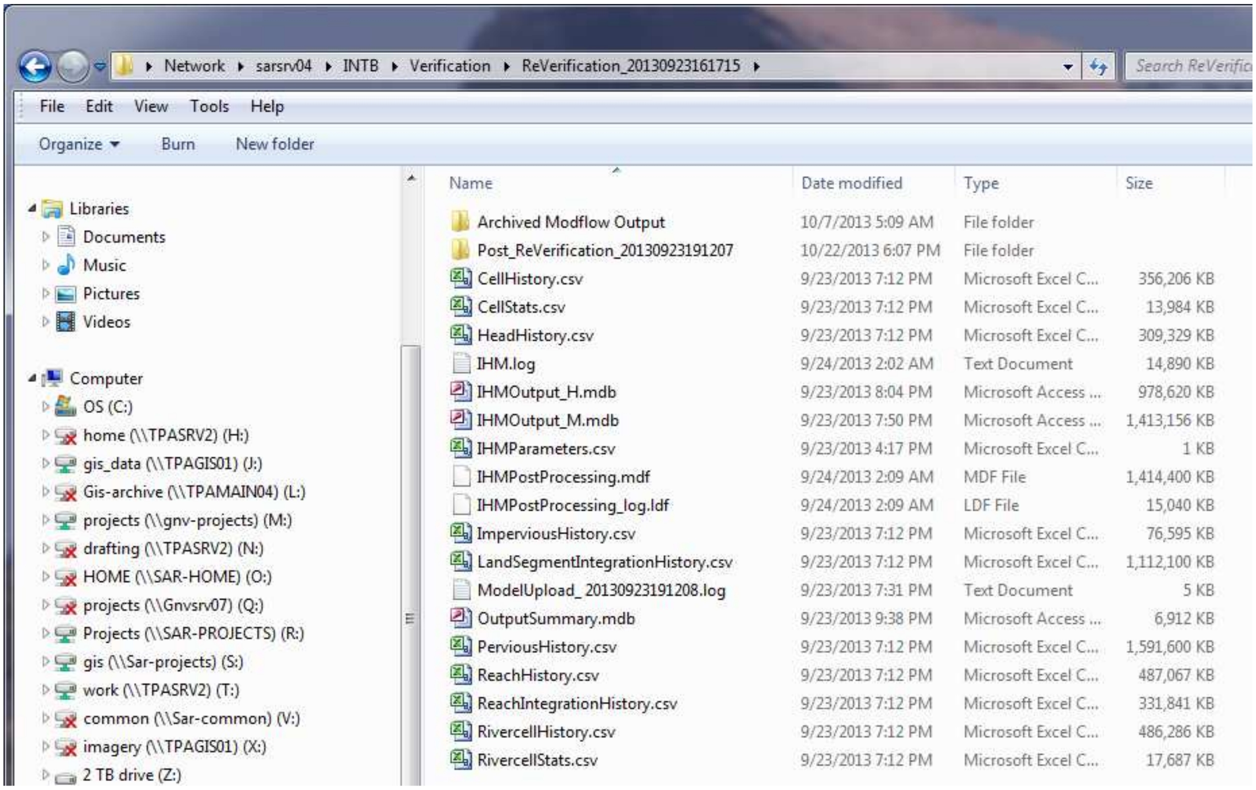
4. Within the box at the top half of the interface, select the name "ReVerification" from the list of possible post processing sequences if you wish to create plots with Grapher. If you are NOT making plots with Grapher, select "ReVerification_NoGrph" from the list.

5. Click "Execute Sequence.."

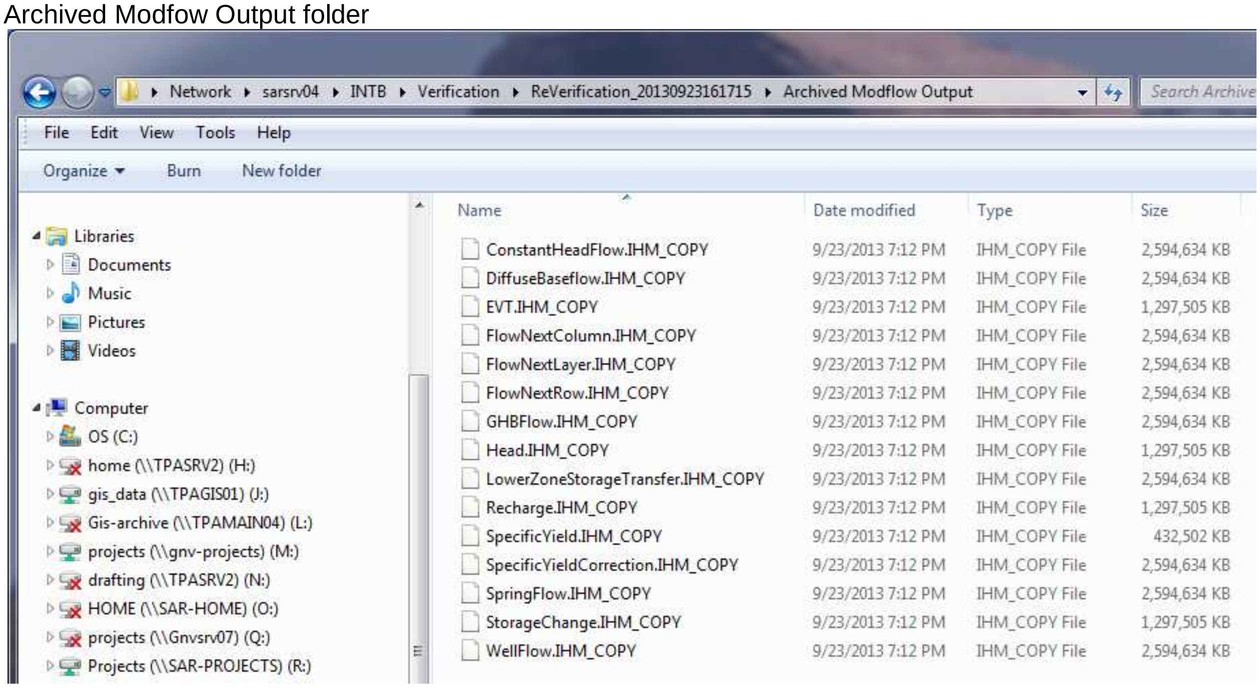
6. Browse to the Simulation Output Directory. The output directory (i.e. C:\INTB\ReVerification\ReVerification_YearMonthDayTime) and then click Start.

7. When the post processing sequence completes there will be a new folder under the simulation output directory that is named "Post_ReVerification_(YearMonthDayTime)" that contains post processed Balances and OutputSummary folders.

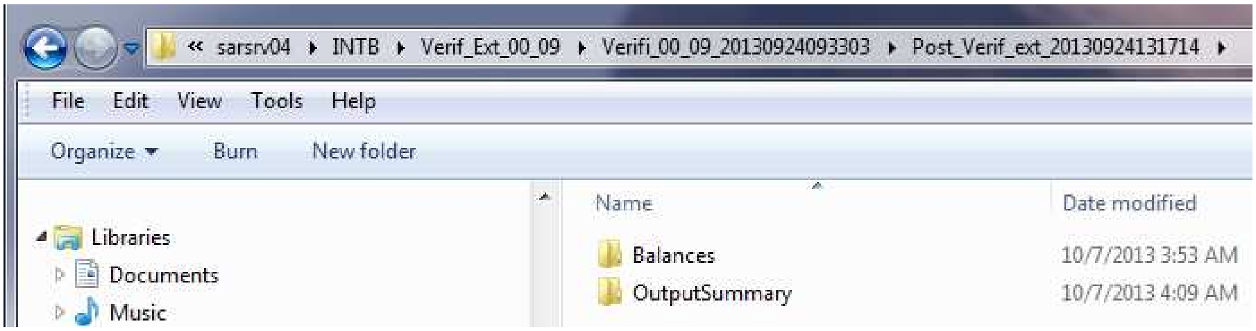
ReVerification folder created during the Post Processing routine.



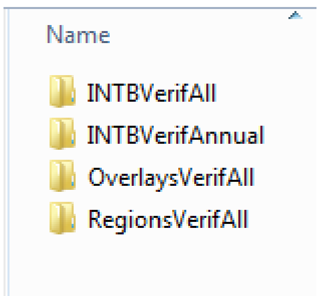
Archived Modfow Output folder



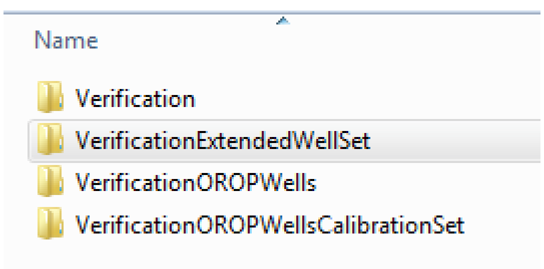
Post Verification Folder



Balances folder



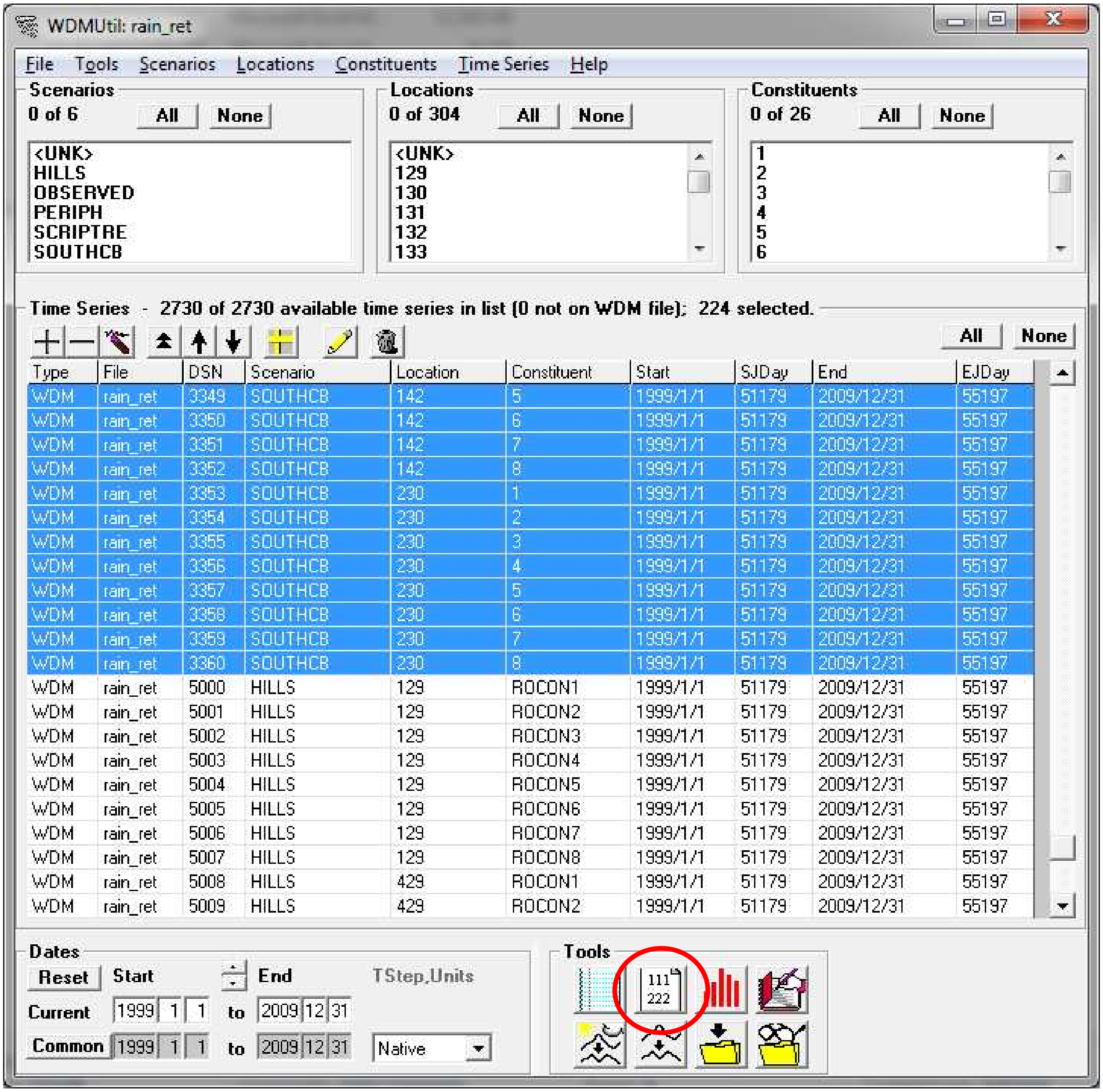
Output Summary folder



Within the Balances and Output Summary folders are the post processing statistics, graphical and tablular, used to reverify water balance, streamflow, ET, springflow, surficial well data, and aquifer water levels.

C. Water Quality

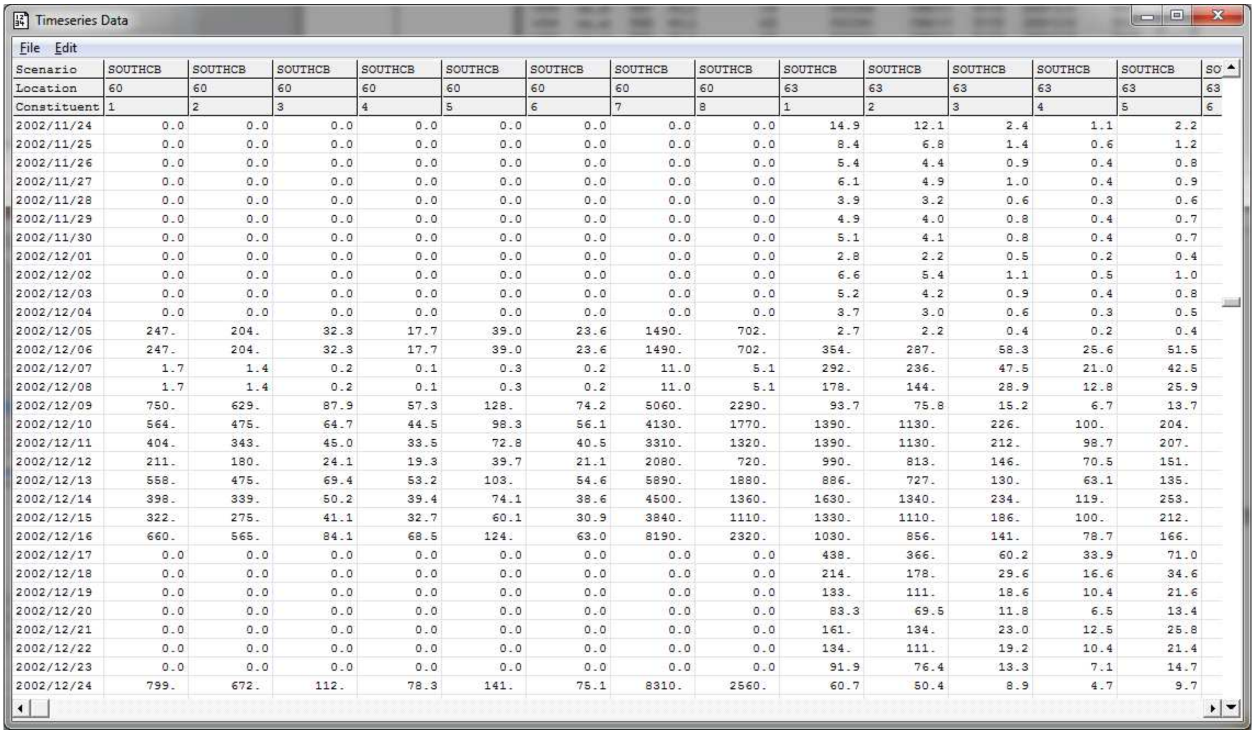
1. Double click the WDMUtil icon on the desktop. From the file drop down, navigate to the rein_ret.wdm for the current simulation and open. Select and highlight the reach or watershed data to export and click on the icon in the Tools box.



Note: In the constituent column are numbers ranging from 1 through 8 each corresponds to a modeled water quality variable as shown in the table below.

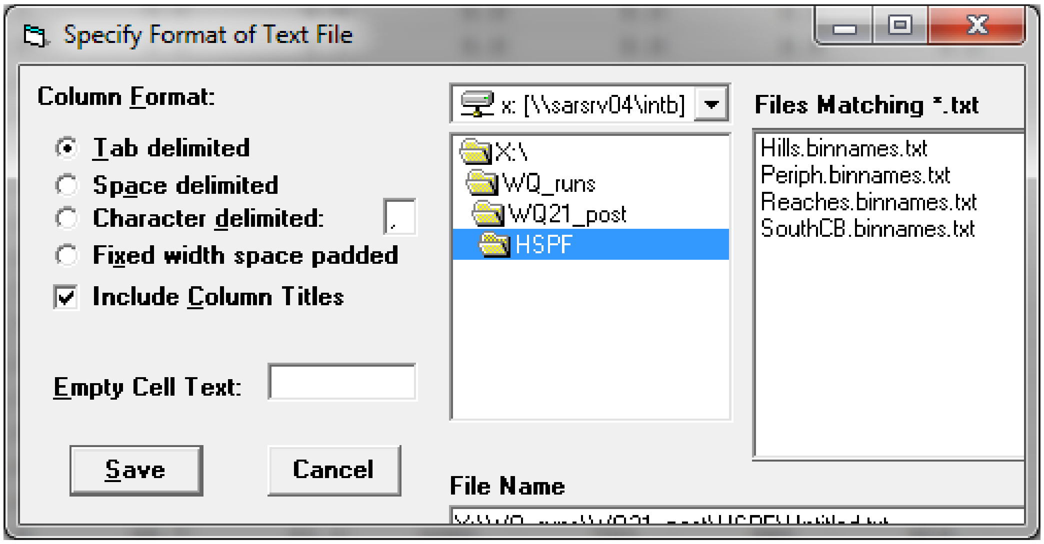
Constituent #	Constituent Name
1	TN
2	TKN
3	NOx
4	NH3
5	TP
6	Ortho-P
7	TSS
8	BOD

2. Within the Timeseries Data table, select File>Save to Text File



Scenario	SOUTHCB	SOUTHCB	SOUTHCB	SOUTHCB	SOUTHCB	SOUTHCB	SOUTHCB	SOUTHCB	SOUTHCB	SOUTHCB	SOUTHCB	SOUTHCB	SOUTHCB	SO
Location	60	60	60	60	60	60	60	60	63	63	63	63	63	63
Constituent	1	2	3	4	5	6	7	8	1	2	3	4	5	6
2002/11/24	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	14.9	12.1	2.4	1.1	2.2	
2002/11/25	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	8.4	6.8	1.4	0.6	1.2	
2002/11/26	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	5.4	4.4	0.9	0.4	0.8	
2002/11/27	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	6.1	4.9	1.0	0.4	0.9	
2002/11/28	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.9	3.2	0.6	0.3	0.6	
2002/11/29	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	4.9	4.0	0.8	0.4	0.7	
2002/11/30	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	5.1	4.1	0.8	0.4	0.7	
2002/12/01	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	2.8	2.2	0.5	0.2	0.4	
2002/12/02	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	6.6	5.4	1.1	0.5	1.0	
2002/12/03	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	5.2	4.2	0.9	0.4	0.8	
2002/12/04	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.7	3.0	0.6	0.3	0.5	
2002/12/05	247.	204.	32.3	17.7	39.0	23.6	1490.	702.	2.7	2.2	0.4	0.2	0.4	
2002/12/06	247.	204.	32.3	17.7	39.0	23.6	1490.	702.	354.	287.	58.3	25.6	51.5	
2002/12/07	1.7	1.4	0.2	0.1	0.3	0.2	11.0	5.1	292.	236.	47.5	21.0	42.5	
2002/12/08	1.7	1.4	0.2	0.1	0.3	0.2	11.0	5.1	178.	144.	28.9	12.8	25.9	
2002/12/09	750.	629.	87.9	57.3	128.	74.2	5060.	2290.	93.7	75.8	15.2	6.7	13.7	
2002/12/10	564.	475.	64.7	44.5	98.3	56.1	4130.	1770.	1390.	1130.	226.	100.	204.	
2002/12/11	404.	343.	45.0	33.5	72.8	40.5	3310.	1320.	1390.	1130.	212.	98.7	207.	
2002/12/12	211.	180.	24.1	19.3	39.7	21.1	2080.	720.	990.	813.	146.	70.5	151.	
2002/12/13	558.	475.	69.4	53.2	103.	54.6	5890.	1880.	886.	727.	130.	63.1	135.	
2002/12/14	398.	339.	50.2	39.4	74.1	38.6	4500.	1360.	1630.	1340.	234.	119.	253.	
2002/12/15	322.	275.	41.1	32.7	60.1	30.9	3840.	1110.	1330.	1110.	186.	100.	212.	
2002/12/16	660.	565.	84.1	68.5	124.	63.0	8190.	2320.	1030.	856.	141.	78.7	166.	
2002/12/17	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	438.	366.	60.2	33.9	71.0	
2002/12/18	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	214.	178.	29.6	16.6	34.6	
2002/12/19	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	133.	111.	18.6	10.4	21.6	
2002/12/20	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	83.3	69.5	11.8	6.5	13.4	
2002/12/21	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	161.	134.	23.0	12.5	25.8	
2002/12/22	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	134.	111.	19.2	10.4	21.4	
2002/12/23	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	91.9	76.4	13.3	7.1	14.7	
2002/12/24	799.	672.	112.	78.3	141.	75.1	8310.	2560.	60.7	50.4	8.9	4.7	9.7	

3. Save the water quality load data to the simulation output directory.



4. Open a new MS Excel file.
5. Under the Data Tab, select Get External Data From Text.
6. Import the water quality load text file and save to .xls format.

R CODE FOR POST PROCESSING

Table 3-1.R

Bridgette Froeschke — *Dec 1, 2013, 7:00 PM*

```
##R Manual for Old Tampa Bay Integreated Model System, Watershed Model:
##Objective:  Calculate geometric means from table 3.1 in the calibration report using an open
resource data analysis package.
##Genearl information: R stores functions as libraries.  We will be using the library doBy for
calculating means. The first
##time you use doBy you must download the library:  1) select the cran mirror (choose USA(CA2), and
then select install package(S),
##choose doBy, all of this is under the Packages drop down menu, afterwards you then need to tell R
to open the library by using the following command.

library(doBy)
```

```
Loading required package: multcomp Loading required package: mvtnorm
Loading required package: survival Loading required package: splines
Loading required package: MASS
```

```
##Steps:

##Data Import:
##The primary functions to read data into R are read.table (for text files) and read.csv (for csv
files).  To read in a data file 1)name the file (if you do not name the file then R will
##open up the data in the workspace), 2) tell R what function to use (read.table or read.csv), 3)
tell R where the file is, 4)how the data are separated, 5)
##and if there are column names (header).
##Note: R is case sensitive
##A skeleton example
x<-read.table("C:/tmpR/example.txt",sep="",header=FALSE)

              ##since the file was named the file is now stored in R

              read.table("C:/tmpR/example.txt",sep="",header=FALSE)
```

```
  V1 V2
1  1  4
2  2  6
3  5  8
4 10 10
```

```
##the above command was not named, therefore, the file will appear in the workspace
##########Note: R is case sensitive#################################

##Below is the code used to calculate the geometric means in Table 3.1 of the calibration report
##Data files used: simulated = model_conc_131029.csv; observed = ambientdata_131022.csv
##Files are csv files with headers


#############################################Simulated##############################
##Data

model_conc<-read.csv("C:/Users/john/Documents/old tampa bay/water optimal model/run
dataset/model_conc_131029.csv",header=TRUE)

head(model_conc)
```

```
       date TN_MOD TKN_MOD NOX_MOD NH3_MOD TP_MOD ORTHOP_MOD TSS_MOD
1 01JAN2000     NA      NA      NA      NA     NA         NA      NA
2 01JAN2000     NA      NA      NA      NA     NA         NA      NA
3 01JAN2000     NA      NA      NA      NA     NA         NA      NA
4 01JAN2000  1.775   1.372  0.4223  0.1518 0.2375     0.1518   7.522
```

```
5 01JAN2000     NA      NA      NA      NA     NA         NA      NA
6 01JAN2000     NA      NA      NA      NA     NA         NA      NA
  BOD_MOD REACH                        name
1      NA    60                Bishop/Mullet
2      NA    63 Alligator/Allens/Long Branch
3      NA    65                    Roosevelt
4   3.213    66                      St Pete
5      NA    68                Double Branch
6      NA    69     Safety Harbor/Mobbly Bay
```

```
dim(model_conc)
```

```
[1] 76713    11
```

```
##Subset data to only include reaches in OTB

model_conc_OTB<-subset(model_conc,model_conc$REACH==60 | model_conc$REACH==63 |
model_conc$REACH==65 | model_conc$REACH==68 | model_conc$REACH==76 |
    model_conc$REACH==77 | model_conc$REACH==227 | model_conc$REACH==229)


##Create County column
##First line creates a column with only Pinellas
model_conc_OTB$County="Pinellas"

##Modify the column to include Hillsborough
model_conc_OTB$County[model_conc_OTB$REACH==68] <- "Hillsborough"
model_conc_OTB$County[model_conc_OTB$REACH==76] <- "Hillsborough"
model_conc_OTB$County[model_conc_OTB$REACH==77] <- "Hillsborough"
model_conc_OTB$County[model_conc_OTB$REACH==227] <- "Hillsborough"
model_conc_OTB$County[model_conc_OTB$REACH==229] <- "Hillsborough"

head(model_conc_OTB)
```

```
       date TN_MOD TKN_MOD NOX_MOD NH3_MOD TP_MOD ORTHOP_MOD TSS_MOD
1 01JAN2000     NA      NA      NA      NA     NA         NA      NA
2 01JAN2000     NA      NA      NA      NA     NA         NA      NA
3 01JAN2000     NA      NA      NA      NA     NA         NA      NA
5 01JAN2000     NA      NA      NA      NA     NA         NA      NA
7 01JAN2000     NA      NA      NA      NA     NA         NA      NA
8 01JAN2000  3.966   3.054  0.9518  0.3569 0.5155     0.3569    16.7
  BOD_MOD REACH                        name       County
1      NA    60                Bishop/Mullet     Pinellas
2      NA    63 Alligator/Allens/Long Branch     Pinellas
3      NA    65                    Roosevelt     Pinellas
5      NA    68                Double Branch Hillsborough
7      NA    76                  Rocky Creek Hillsborough
8   7.178    77             Lower Sweetwater Hillsborough
```

```
##Sort by county
model_conc_county<-data.frame(model_conc_OTB[order(model_conc_OTB$County),])

head(model_conc_county)
```

```
        date TN_MOD TKN_MOD NOX_MOD NH3_MOD TP_MOD ORTHOP_MOD TSS_MOD
5  01JAN2000     NA      NA      NA      NA     NA         NA      NA
7  01JAN2000     NA      NA      NA      NA     NA         NA      NA
8  01JAN2000  3.966   3.054  0.9518  0.3569 0.5155     0.3569  16.695
14 01JAN2000  1.460   1.127  0.3485  0.1262 0.1943     0.1262   6.168
15 01JAN2000  2.478   1.921  0.5926  0.2155 0.3232     0.2155  10.469
26 02JAN2000     NA      NA      NA      NA     NA         NA      NA
   BOD_MOD REACH          name       County
5       NA    68 Double Branch Hillsborough
7       NA    76   Rocky Creek Hillsborough
```

```
8    7.178    77 Lower Sweetwater Hillsborough
14   2.644   227        Channel A Hillsborough
15   4.489   229        Channel G Hillsborough
26      NA    68    Double Branch Hillsborough
```

```
##change 0 to na
model_conc_county[model_conc_county==0]<-NA

##Subset data into seperate parameter data sets, NAs are still present
TKN.na<-data.frame(cbind(model_conc_county$TKN_MOD,model_conc_county$County))
NH3.na<-data.frame(cbind(model_conc_county$NH3_MOD,model_conc_county$County))
NOX.na<-data.frame(cbind(model_conc_county$NOX_MOD,model_conc_county$County))
TN.na<-data.frame(cbind(model_conc_county$TN_MOD,model_conc_county$County))
ORTHOP.na<-data.frame(cbind(model_conc_county$ORTHOP_MOD,model_conc_county$County))
TP.na<-data.frame(cbind(model_conc_county$TP_MOD,model_conc_county$County))
TSS.na<-data.frame(cbind(model_conc_county$TSS_MOD,model_conc_county$County))
BOD.na<-data.frame(cbind(model_conc_county$BOD_MOD,model_conc_county$County))

head(TKN.na)
```

```
      X1           X2
1   <NA> Hillsborough
2   <NA> Hillsborough
3 3.0536 Hillsborough
4 1.1275 Hillsborough
5 1.9214 Hillsborough
6   <NA> Hillsborough
```

```
dim(TKN.na)
```

```
[1] 29224     2
```

```
##Remove NAs from all the data sets
TKN<-na.omit(TKN.na)
NH3<-na.omit(NH3.na)
NOX<-na.omit(NOX.na)
TN<-na.omit(TN.na)
ORTHOP<-na.omit(ORTHOP.na)
TP<-na.omit(TP.na)
TSS<-na.omit(TSS.na)
BOD<-na.omit(BOD.na)

head(TKN)
```

```
       X1           X2
3  3.0536 Hillsborough
4  1.1275 Hillsborough
5  1.9214 Hillsborough
8  1.7354 Hillsborough
9  1.0638 Hillsborough
10 1.5501 Hillsborough
```

```
dim(TKN)
```

```
[1] 18218     2
```

```
##Calculate the natural log as a new column in each dataset
##Have to include as.character to ensure that R recognizes the value as a number and not as a level
of a factor
TKN$ln_TKN_MOD<-log(as.numeric(as.character(TKN[,1])))
NH3$ln_NH3_MOD<-log(as.numeric(as.character(NH3[,1])))
```

```
NOX$ln_NOX_MOD<-log(as.numeric(as.character(NOX[,1])))
```

```
Warning: NaNs produced
```

```
TN$ln_TN_MOD<-log(as.numeric(as.character(TN[,1])))
ORTHOP$ln_ORTHOP_MOD<-log(as.numeric(as.character(ORTHOP[,1])))
TP$ln_TP_MOD<-log(as.numeric(as.character(TP[,1])))
TSS$ln_TSS_MOD<-log(as.numeric(as.character(TSS[,1])))
BOD$ln_BOD_MOD<-log(as.numeric(as.character(BOD[,1])))

head(TKN)
```

```
       X1           X2  ln_TKN_MOD
3  3.0536 Hillsborough    1.11632
4  1.1275 Hillsborough    0.12000
5  1.9214 Hillsborough    0.65305
8  1.7354 Hillsborough    0.55124
9  1.0638 Hillsborough    0.06185
10 1.5501 Hillsborough    0.43832
```

```
##Add column names to each data set
names<-c("P","County","ln")
colnames(TKN)<-names
colnames(NH3)<-names
colnames(NOX)<-names
colnames(TN)<-names
colnames(ORTHOP)<-names
colnames(TP)<-names
colnames(TSS)<-names
colnames(BOD)<-names

##key
  ##  TKN_MOD       = "Modeled Total Kjeldahl Nitrogen (mg/L)
  ##  NH3_MOD       = "Modeled Ammonia (mg/L)
  ##  NOX_MOD       = "Modeled Nitrate Nitrite (mg/L)
  ##  TN_MOD        = "Modeled Total Nitrogen (mg/L)
  ##  ORTHOP_MOD    = "Modeled Ortho Phosphorus (mg/L)
  ##  TP_MOD        = "Modeled Total Phosphorus (mg/L)
  ##  TSS_MOD       = "Modeled Total Suspended Solids (mg/L)
  ##  BOD_MOD       = "Modeled Biological Oxygen Demand 5 (mg/L)
  ##  ln_TKN_MOD    = "Log Transformed Modeled Total Kjeldahl Nitrogen (mg/L)
  ##  ln_NH3_MOD    = "Log Transformed Modeled Ammonia (mg/L)
  ##  ln_NOX_MOD    = "Log Transformed Modeled Nitrate Nitrite (mg/L)
  ##  ln_TN_MOD     = "Log Transformed Modeled Total Nitrogen (mg/L)
  ##  ln_ORTHOP_MOD = "Log Transformed Modeled Ortho Phosphorus (mg/L)
  ##  ln_TP_MOD     = "Log Transformed Modeled Total Phosphorus (mg/L)
  ##  ln_TSS_MOD    = "Log Transformed Modeled Total Suspended Solids (mg/L)
  ##  ln_BOD_MOD    = "Log Transformed Modeled Biological Oxygen Demand 5 (mg/L)
  ##  reach         = "Reach"
  ##  name          = "Name"




##Calculate the means of each variable by County using the function summaryBy in the library doBy

TKN_mean<-summaryBy(P~County, data=TKN, FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
NH3_mean<-summaryBy(P~County, data=NH3,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
NOX_mean<-summaryBy(P~County, data=NOX,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
TN_mean<-summaryBy(P~County,data=TN,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
ORTHOP_mean<-summaryBy(P~County,data=ORTHOP,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
TP_mean<-summaryBy(P~County,data=TP,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
TSS_mean<-summaryBy(P~County,data=TSS,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
BOD_mean<-summaryBy(P~County,data=BOD,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
ln_TKN_mean<-summaryBy( ln~County,data=TKN,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
ln_NH3_mean<-summaryBy( ln~County,data=NH3,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
```

```
ln_NOX_mean<-summaryBy(ln~County,data=NOX,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
ln_TN_mean<-summaryBy(ln~County,data=TN,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
ln_ORTHOP_mean<-summaryBy(ln~County,data=ORTHOP,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
ln_TP_mean<-summaryBy(ln~County,data=TP,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
ln_TSS_mean<-summaryBy(ln~County,data=TSS,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
ln_BOD_mean<-summaryBy(ln~County,data=BOD,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)


##Merge the mean values into one data set using the function cbind
mod2<-cbind(TKN_mean,
            NH3_mean[,2],
            NOX_mean[,2],
            TN_mean[,2],
            ORTHOP_mean[,2],
            TP_mean[,2],
            TSS_mean[,2],
            BOD_mean[,2],
            ln_TKN_mean[,2],
            ln_NH3_mean[,2],
            ln_NOX_mean[,2],
            ln_TN_mean[,2],
            ln_ORTHOP_mean[,2],
            ln_TP_mean[,2],
            ln_TSS_mean[,2],
            ln_BOD_mean[,2])

##Add column names
colnames(mod2) <- c("County",
                    "TKN_mean",
                    "NH3_mean",
                    "NOX_mean",
                    "TN_mean",
                    "ORTHOP_mean",
                    "TP_mean",
                    "TSS_mean",
                    "BOD_mean",
                    "ln_TKN_mean",
                    "ln_NH3_mean",
                    "ln_NOX_mean",
                    "ln_TN_mean",
                    "ln_ORTHOP_mean",
                    "ln_TP_mean",
                    "ln_TSS_mean",
                    "ln_BOD_mean")
head(mod2)
```

```
       County TKN_mean NH3_mean NOX_mean TN_mean ORTHOP_mean TP_mean
1 Hillsborough     4349    767.8     1714    4987       814.1    1410
2     Pinellas     4711    712.8     1510    5299       934.5    1385
  TSS_mean BOD_mean ln_TKN_mean ln_NH3_mean ln_NOX_mean ln_TN_mean
1     7779     6288     -0.3986      -2.688       -1.88    -0.1797
2     8629     7687     -0.3626      -2.820       -2.05    -0.1594
  ln_ORTHOP_mean ln_TP_mean ln_TSS_mean ln_BOD_mean
1         -2.631     -2.026       1.601      0.7167
2         -2.522     -2.077       1.670      0.8296
```

```
##Final calculation using the function exp
  geom_TKN_MOD<-data.frame(County=c("Hillsborough","Pinellas"),geomean=exp(mod2$ln_TKN_mean))
  geom_NH3_MOD<-data.frame(County=c("Hillsborough","Pinellas"),geomean=exp(mod2$ln_NH3_mean))
  geom_NOX_MOD<-data.frame(County=c("Hillsborough","Pinellas"),geomean=exp(mod2$ln_NOX_mean))
  geom_TN_MOD <-data.frame(County=c("Hillsborough","Pinellas"),geomean=exp(mod2$ln_TN_mean))
  geom_ORTHOP_MOD
<-data.frame(County=c("Hillsborough","Pinellas"),geomean=exp(mod2$ln_ORTHOP_mean))
  geom_TP_MOD<-data.frame(County=c("Hillsborough","Pinellas"),geomean=exp(mod2$ln_TP_mean))
  geom_TSS_MOD<-data.frame(County=c("Hillsborough","Pinellas"),geomean=exp(mod2$ln_TSS_mean))
  geom_BOD_MOD<-data.frame(County=c("Hillsborough","Pinellas"),geomean=exp(mod2$ln_BOD_mean))
```

```
head(geom_TKN_MOD)
```

```
        County geomean
1 Hillsborough  0.6713
2     Pinellas  0.6959
```

```
##Merge the geometric mean values into one data set using the function cbind
mod.geomean<-cbind(geom_TKN_MOD,
         geom_NH3_MOD[,2],
            geom_NOX_MOD[,2],
            geom_TN_MOD[,2],
            geom_ORTHOP_MOD[,2],
            geom_TP_MOD[,2],
            geom_TSS_MOD[,2],
            geom_BOD_MOD[,2])


##Add column names
colnames(mod.geomean) <- c("County",
                    "TKN_geomean",
                    "NH3_geomean",
                    "NOX_geomean",
                    "TN_geomean",
                    "ORTHOP_geomean",
                    "TP_geomean",
                    "TSS_geomean",
                    "BOD_geomean")
head(mod.geomean)
```

```
        County TKN_geomean NH3_geomean NOX_geomean TN_geomean
1 Hillsborough      0.6713     0.06799      0.1526     0.8355
2     Pinellas      0.6959     0.05958      0.1287     0.8527
  ORTHOP_geomean TP_geomean TSS_geomean BOD_geomean
1        0.07204     0.1319       4.956       2.048
2        0.08028     0.1253       5.312       2.292
```

```
##Export data
write.csv(mod.geomean,"C:/Users/john/Documents/old tampa bay/water optimal
model/bf/simulated__county_geomean.csv",row.names=FALSE)

##Calculate the means of each varible overall
all.TKN_mean<-mean(na.omit(as.numeric(as.character(TKN$P))))
all.NH3_mean<-mean(na.omit(as.numeric(as.character(NH3$P))))
all.NOX_mean<-mean(na.omit(as.numeric(as.character(NOX$P))))
all.TN_mean<-mean(na.omit(as.numeric(as.character(TN$P))))
all.ORTHOP_mean<-mean(na.omit(as.numeric(as.character(ORTHOP$P))))
all.TP_mean<-mean(na.omit(as.numeric(as.character(TP$P))))
all.TSS_mean<-mean(na.omit(as.numeric(as.character(TSS$P))))
all.BOD_mean<-mean(na.omit(as.numeric(as.character(BOD$P))))
all.ln_TKN_mean<-mean(na.omit(as.numeric(as.character(TKN$ln))))
all.ln_NH3_mean<-mean(na.omit(as.numeric(as.character(NH3$ln))))
all.ln_NOX_mean<-mean(na.omit(as.numeric(as.character(NOX$ln))))
all.ln_TN_mean<-mean(na.omit(as.numeric(as.character(TN$ln))))
all.ln_ORTHOP_mean<-mean(na.omit(as.numeric(as.character(ORTHOP$ln))))
all.ln_TP_mean<-mean(na.omit(as.numeric(as.character(TP$ln))))
all.ln_TSS_mean<-mean(na.omit(as.numeric(as.character(TSS$ln))))
all.ln_BOD_mean<-mean(na.omit(as.numeric(as.character(BOD$ln))))


##Merge the mean values into one data set using the function cbind
all.mod2<-data.frame(cbind(all.TKN_mean,
            all.NH3_mean,
            all.NOX_mean,
            all.TN_mean,
            all.ORTHOP_mean,
            all.TP_mean,
```

```
            all.TSS_mean,
            all.BOD_mean,
            all.ln_TKN_mean,
            all.ln_NH3_mean,
            all.ln_NOX_mean,
            all.ln_TN_mean,
            all.ln_ORTHOP_mean,
            all.ln_TP_mean,
            all.ln_TSS_mean,
            all.ln_BOD_mean))

##Add column names
colnames(all.mod2) <- c("TKN_mean",
                   "NH3_mean",
                   "NOX_mean",
                   "TN_mean",
                   "ORTHOP_mean",
                   "TP_mean",
                   "TSS_mean",
                   "BOD_mean",
                   "ln_TKN_mean",
                   "ln_NH3_mean",
                   "ln_NOX_mean",
                   "ln_TN_mean",
                   "ln_ORTHOP_mean",
                   "ln_TP_mean",
                   "ln_TSS_mean",
                   "ln_BOD_mean")
head(all.mod2)
```

```
  TKN_mean NH3_mean NOX_mean TN_mean ORTHOP_mean TP_mean TSS_mean BOD_mean
1   0.7952  0.07705   0.1755  0.9867      0.0867  0.1512    5.922     2.46
  ln_TKN_mean ln_NH3_mean ln_NOX_mean ln_TN_mean ln_ORTHOP_mean ln_TP_mean
1     -0.3884      -2.725      -1.927     -0.174           -2.6      -2.04
  ln_TSS_mean ln_BOD_mean
1        1.62      0.7485
```

```
##Final calculation using the function exp
  all.geom_TKN_MOD<-data.frame(geomean=exp(all.mod2$ln_TKN_mean))
  all.geom_NH3_MOD<-data.frame(geomean=exp(all.mod2$ln_NH3_mean))
  all.geom_NOX_MOD<-data.frame(geomean=exp(all.mod2$ln_NOX_mean))
  all.geom_TN_MOD <-data.frame(geomean=exp(all.mod2$ln_TN_mean))
  all.geom_ORTHOP_MOD <-data.frame(geomean=exp(all.mod2$ln_ORTHOP_mean))
  all.geom_TP_MOD<-data.frame(geomean=exp(all.mod2$ln_TP_mean))
  all.geom_TSS_MOD<-data.frame(geomean=exp(all.mod2$ln_TSS_mean))
  all.geom_BOD_MOD<-data.frame(geomean=exp(all.mod2$ln_BOD_mean))

head(all.geom_TKN_MOD)
```

```
  geomean
1  0.6781
```

```
##Merge the geometric mean values into one data set using the function cbind
all.mod.geomean<-data.frame(cbind(all.geom_TKN_MOD,
         all.geom_NH3_MOD,
            all.geom_NOX_MOD,
            all.geom_TN_MOD,
            all.geom_ORTHOP_MOD,
            all.geom_TP_MOD,
            all.geom_TSS_MOD,
            all.geom_BOD_MOD))


##Add column names
colnames(all.mod.geomean) <- c("TKN",
                   "NH3",
```

```
                   "NOX",
                   "TN",
                   "ORTHOP",
                   "TP",
                   "TSS",
                   "BOD")
head(all.mod.geomean)
```

```
     TKN     NH3    NOX     TN  ORTHOP   TP   TSS   BOD
1 0.6781 0.06557 0.1456 0.8403 0.07424 0.13 5.054 2.114
```

```
##transpose data set
trans.otb.geomean<-data.frame(Geometric_mean=t(all.mod.geomean))

head(trans.otb.geomean)
```

```
       Geometric_mean
TKN           0.67811
NH3           0.06557
NOX           0.14556
TN            0.84030
ORTHOP        0.07424
TP            0.13000
```

```
##Export data
write.csv(trans.otb.geomean,"C:/Users/john/Documents/old tampa bay/water optimal
model/bf/simulated_otb_geomean.csv",row.names=TRUE)

################################Observed#####################################

##Data

ambient_conc<-read.csv("C:/Users/john/Documents/old tampa bay/water optimal model/run
dataset/ambientdata_131022.csv",header=TRUE)

head(ambient_conc)
```

```
  REACH          name    date    tn  tkn  NOX  NH3   tp ORTHOP tss BOD
1    60 Bishop/Mullet 20JAN00 0.545 0.54 0.01 0.11 0.14  0.085   7 1.5
2    60 Bishop/Mullet 10FEB00 0.460 0.45 0.01 0.05 0.12  0.080   5 1.0
3    60 Bishop/Mullet 16FEB00 0.520 0.50 0.02 0.12 0.10  0.050   1 1.0
4    60 Bishop/Mullet 27MAR00 0.615 0.58 0.04 0.13 0.12  0.060  17 2.0
5    60 Bishop/Mullet 05APR00 0.740 0.70 0.04 0.20 0.37  0.230  15  NA
6    60 Bishop/Mullet 11APR00 0.650 0.64 0.01 0.30 0.14  0.080  13 1.0
       loc
1 Pinellas
2 Pinellas
3 Pinellas
4 Pinellas
5 Pinellas
6 Pinellas
```

```
dim(ambient_conc)
```

```
[1] 1397   12
```

```
##Subset data to only include reaches in OTB

ambient_conc_OTB<-subset(ambient_conc,ambient_conc$REACH==60 | ambient_conc$REACH==63 |
ambient_conc$REACH==65 | ambient_conc$REACH==68 |ambient_conc$REACH==76 |
    ambient_conc$REACH==77 | ambient_conc$REACH==227 | ambient_conc$REACH==229)
```

```
##Create County column
##First line creates a column with only Pinellas
ambient_conc_OTB$County="Pinellas"

##Modify the column to include Hillsborough
ambient_conc_OTB$County[ambient_conc_OTB$REACH==68] <- "Hillsborough"
ambient_conc_OTB$County[ambient_conc_OTB$REACH==76] <- "Hillsborough"
ambient_conc_OTB$County[ambient_conc_OTB$REACH==77] <- "Hillsborough"
ambient_conc_OTB$County[ambient_conc_OTB$REACH==227] <- "Hillsborough"
ambient_conc_OTB$County[ambient_conc_OTB$REACH==229] <- "Hillsborough"


##Sort by county
ambient_conc_county<-data.frame(ambient_conc_OTB[order(ambient_conc_OTB$County),])

head(ambient_conc_county)
```

```
    REACH          name    date    tn  tkn  NOX  NH3   tp ORTHOP tss  BOD
416    68 Double Branch 24JUL01 1.850 1.80 0.05 0.13 0.32   0.21  NA 1.86
417    68 Double Branch 18SEP01 1.260 1.22 0.04 0.04 0.18   0.10  NA 1.73
418    68 Double Branch 23JUL02 1.365 1.31 0.06 0.04 0.14   0.08  NA 2.00
419    68 Double Branch 20AUG02 1.191 1.10 0.09 0.06 0.16   0.07  NA 1.00
420    68 Double Branch 17SEP02 1.305 1.20 0.11 0.08 0.10   0.08  NA 1.00
421    68 Double Branch 10DEC02 1.102 1.02 0.08 0.10 0.14   0.08  NA 3.00
             loc       County
416 Hillsborough Hillsborough
417 Hillsborough Hillsborough
418 Hillsborough Hillsborough
419 Hillsborough Hillsborough
420 Hillsborough Hillsborough
421 Hillsborough Hillsborough
```

```
##change 0 to na
ambient_conc_county[ambient_conc_county==0]<-NA

##Subset data into seperate parameter data sets, NAs are still present
obs.TKN.na<-data.frame(cbind(ambient_conc_county$tkn,ambient_conc_county$County))
obs.NH3.na<-data.frame(cbind(ambient_conc_county$NH3,ambient_conc_county$County))
obs.NOX.na<-data.frame(cbind(ambient_conc_county$NOX,ambient_conc_county$County))
obs.TN.na<-data.frame(cbind(ambient_conc_county$tn,ambient_conc_county$County))
obs.ORTHOP.na<-data.frame(cbind(ambient_conc_county$ORTHOP,ambient_conc_county$County))
obs.TP.na<-data.frame(cbind(ambient_conc_county$tp,ambient_conc_county$County))
obs.TSS.na<-data.frame(cbind(ambient_conc_county$tss,ambient_conc_county$County))
obs.BOD.na<-data.frame(cbind(ambient_conc_county$BOD,ambient_conc_county$County))

head(obs.TKN.na)
```

```
    X1           X2
1  1.8 Hillsborough
2 1.22 Hillsborough
3 1.31 Hillsborough
4  1.1 Hillsborough
5  1.2 Hillsborough
6 1.02 Hillsborough
```

```
dim(obs.TKN.na)
```

```
[1] 683   2
```

```
##Remove NAs from all the data sets
obs.TKN<-na.omit(obs.TKN.na)
obs.NH3<-na.omit(obs.NH3.na)
obs.NOX<-na.omit(obs.NOX.na)
obs.TN<-na.omit(obs.TN.na)
```

```
obs.ORTHOP<-na.omit(obs.ORTHOP.na)
obs.TP<-na.omit(obs.TP.na)
obs.TSS<-na.omit(obs.TSS.na)
obs.BOD<-na.omit(obs.BOD.na)

head(obs.TKN)
```

```
    X1          X2
1  1.8 Hillsborough
2 1.22 Hillsborough
3 1.31 Hillsborough
4  1.1 Hillsborough
5  1.2 Hillsborough
6 1.02 Hillsborough
```

```
dim(obs.TKN)
```

```
[1] 683   2
```

```
##Calculate the natural log as a new column in each dataset
##Have to include as.character to ensure that R recognizes the value as a number and not as a level
of a factor
obs.TKN$ln_TKN_MOD<-log(as.numeric(as.character(obs.TKN[,1])))
obs.NH3$ln_NH3_MOD<-log(as.numeric(as.character(obs.NH3[,1])))
obs.NOX$ln_NOX_MOD<-log(as.numeric(as.character(obs.NOX[,1])))
obs.TN$ln_TN_MOD<-log(as.numeric(as.character(obs.TN[,1])))
obs.ORTHOP$ln_ORTHOP_MOD<-log(as.numeric(as.character(obs.ORTHOP[,1])))
obs.TP$ln_TP_MOD<-log(as.numeric(as.character(obs.TP[,1])))
obs.TSS$ln_TSS_MOD<-log(as.numeric(as.character(obs.TSS[,1])))
obs.BOD$ln_BOD_MOD<-log(as.numeric(as.character(obs.BOD[,1])))

head(obs.TKN)
```

```
    X1          X2 ln_TKN_MOD
1  1.8 Hillsborough    0.58779
2 1.22 Hillsborough    0.19885
3 1.31 Hillsborough    0.27003
4  1.1 Hillsborough    0.09531
5  1.2 Hillsborough    0.18232
6 1.02 Hillsborough    0.01980
```

```
##Add column names to each data set
names<-c("P","County","ln")
colnames(obs.TKN)<-names
colnames(obs.NH3)<-names
colnames(obs.NOX)<-names
colnames(obs.TN)<-names
colnames(obs.ORTHOP)<-names
colnames(obs.TP)<-names
colnames(obs.TSS)<-names
colnames(obs.BOD)<-names


##Calculate the means of each variable by County using the function summaryBy in the library doBy

obs.TKN_mean<-summaryBy(P~County, data=obs.TKN, FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
obs.NH3_mean<-summaryBy(P~County, data=obs.NH3,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
obs.NOX_mean<-summaryBy(P~County, data=obs.NOX,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
obs.TN_mean<-summaryBy(P~County,data=obs.TN,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
obs.ORTHOP_mean<-summaryBy(P~County,data=obs.ORTHOP,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
obs.TP_mean<-summaryBy(P~County,data=obs.TP,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
obs.TSS_mean<-summaryBy(P~County,data=obs.TSS,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
obs.BOD_mean<-summaryBy(P~County,data=obs.BOD,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
obs.ln_TKN_mean<-summaryBy(ln~County,data=obs.TKN,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
```

```
obs.ln_NH3_mean<-summaryBy(ln~County,data=obs.NH3,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
obs.ln_NOX_mean<-summaryBy(ln~County,data=obs.NOX,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
obs.ln_TN_mean<-summaryBy(ln~County,data=obs.TN,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
obs.ln_ORTHOP_mean<-summaryBy(ln~County,data=obs.ORTHOP,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
obs.ln_TP_mean<-summaryBy(ln~County,data=obs.TP,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
obs.ln_TSS_mean<-summaryBy(ln~County,data=obs.TSS,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)
obs.ln_BOD_mean<-summaryBy(ln~County,data=obs.BOD,FUN=c(mean),na.rm=TRUE,keep.names=TRUE)



##Merge the mean values into one data set using the function cbind
obs2<-cbind(obs.TKN_mean,
            obs.NH3_mean[,2],
            obs.NOX_mean[,2],
            obs.TN_mean[,2],
            obs.ORTHOP_mean[,2],
            obs.TP_mean[,2],
            obs.TSS_mean[,2],
            obs.BOD_mean[,2],
            obs.ln_TKN_mean[,2],
            obs.ln_NH3_mean[,2],
            obs.ln_NOX_mean[,2],
            obs.ln_TN_mean[,2],
            obs.ln_ORTHOP_mean[,2],
            obs.ln_TP_mean[,2],
            obs.ln_TSS_mean[,2],
            obs.ln_BOD_mean[,2])

##Add column names
colnames(obs2) <- c("County",
                    "TKN_mean",
                    "NH3_mean",
                    "NOX_mean",
                    "TN_mean",
                    "ORTHOP_mean",
                    "TP_mean",
                    "TSS_mean",
                    "BOD_mean",
                    "ln_TKN_mean",
                    "ln_NH3_mean",
                    "ln_NOX_mean",
                    "ln_TN_mean",
                    "ln_ORTHOP_mean",
                    "ln_TP_mean",
                    "ln_TSS_mean",
                    "ln_BOD_mean")
head(obs2)
```

```
       County TKN_mean NH3_mean NOX_mean TN_mean ORTHOP_mean TP_mean
1 Hillsborough    51.32    10.37    21.79   182.4       46.27   13.21
2     Pinellas    54.21    10.34    18.16   177.3       57.37   15.66
  TSS_mean BOD_mean ln_TKN_mean ln_NH3_mean ln_NOX_mean ln_TN_mean
1    14.88    20.48     -0.1388      -2.544      -2.024    0.06152
2    14.88    34.34     -0.1142      -2.855      -2.250    0.06981
  ln_ORTHOP_mean ln_TP_mean ln_TSS_mean ln_BOD_mean
1         -2.896     -2.195       1.237      0.2935
2         -2.687     -1.991       1.237      0.7557
```

```
##Final calculation using the function exp
  obs.geom_TKN<-data.frame(County=c("Hillsborough","Pinellas"),geomean=exp(obs2$ln_TKN_mean))
  obs.geom_NH3<-data.frame(County=c("Hillsborough","Pinellas"),geomean=exp(obs2$ln_NH3_mean))
  obs.geom_NOX<-data.frame(County=c("Hillsborough","Pinellas"),geomean=exp(obs2$ln_NOX_mean))
  obs.geom_TN <-data.frame(County=c("Hillsborough","Pinellas"),geomean=exp(obs2$ln_TN_mean))
  obs.geom_ORTHOP
<-data.frame(County=c("Hillsborough","Pinellas"),geomean=exp(obs2$ln_ORTHOP_mean))
  obs.geom_TP<-data.frame(County=c("Hillsborough","Pinellas"),geomean=exp(obs2$ln_TP_mean))
  obs.geom_TSS<-data.frame(County=c("Hillsborough","Pinellas"),geomean=exp(obs2$ln_TSS_mean))
```

```
obs.geom_BOD<-data.frame(County=c("Hillsborough","Pinellas"),geomean=exp(obs2$ln_BOD_mean))

obs.geom_TKN
```

```
        County geomean
1 Hillsborough  0.8704
2     Pinellas  0.8920
```

```
##Merge the geometric mean values into one data set using the function cbind
obs.geomean<-cbind(obs.geom_TKN,
         obs.geom_NH3[,2],
            obs.geom_NOX[,2],
            obs.geom_TN[,2],
            obs.geom_ORTHOP[,2],
            obs.geom_TP[,2],
            obs.geom_TSS[,2],
            obs.geom_BOD[,2])


##Add column names
colnames(obs.geomean) <- c("County",
                   "TKN_geomean",
                   "NH3_geomean",
                   "NOX_geomean",
                   "TN_geomean",
                   "ORTHOP_geomean",
                   "TP_geomean",
                   "TSS_geomean",
                   "BOD_geomean")
head(obs.geomean)
```

```
        County TKN_geomean NH3_geomean NOX_geomean TN_geomean
1 Hillsborough      0.8704     0.07858      0.1321      1.063
2     Pinellas      0.8920     0.05753      0.1054      1.072
  ORTHOP_geomean TP_geomean TSS_geomean BOD_geomean
1        0.05523     0.1113       3.445       1.341
2        0.06807     0.1365       3.445       2.129
```

```
##Export data
write.csv(obs.geomean,"C:/Users/john/Documents/old tampa bay/water optimal
model/bf/observed_county_geomean_.csv",row.names=TRUE)

##Calculate the means of each varible overall
obs.all.TKN_mean<-mean(na.omit(as.numeric(as.character(obs.TKN$P))))
obs.all.NH3_mean<-mean(na.omit(as.numeric(as.character(obs.NH3$P))))
obs.all.NOX_mean<-mean(na.omit(as.numeric(as.character(obs.NOX$P))))
obs.all.TN_mean<-mean(na.omit(as.numeric(as.character(obs.TN$P))))
obs.all.ORTHOP_mean<-mean(na.omit(as.numeric(as.character(obs.ORTHOP$P))))
obs.all.TP_mean<-mean(na.omit(as.numeric(as.character(obs.TP$P))))
obs.all.TSS_mean<-mean(na.omit(as.numeric(as.character(obs.TSS$P))))
obs.all.BOD_mean<-mean(na.omit(as.numeric(as.character(obs.BOD$P))))
obs.all.ln_TKN_mean<-mean(na.omit(as.numeric(as.character(obs.TKN$ln))))
obs.all.ln_NH3_mean<-mean(na.omit(as.numeric(as.character(obs.NH3$ln))))
obs.all.ln_NOX_mean<-mean(na.omit(as.numeric(as.character(obs.NOX$ln))))
obs.all.ln_TN_mean<-mean(na.omit(as.numeric(as.character(obs.TN$ln))))
obs.all.ln_ORTHOP_mean<-mean(na.omit(as.numeric(as.character(obs.ORTHOP$ln))))
obs.all.ln_TP_mean<-mean(na.omit(as.numeric(as.character(obs.TP$ln))))
obs.all.ln_TSS_mean<-mean(na.omit(as.numeric(as.character(obs.TSS$ln))))
obs.all.ln_BOD_mean<-mean(na.omit(as.numeric(as.character(obs.BOD$ln))))


##Merge the mean values into one data set using the function cbind
all.obs2<-data.frame(cbind(obs.all.TKN_mean,
            obs.all.NH3_mean,
            obs.all.NOX_mean,
            obs.all.TN_mean,
```

```
            obs.all.ORTHOP_mean,
            obs.all.TP_mean,
            obs.all.TSS_mean,
            obs.all.BOD_mean,
            obs.all.ln_TKN_mean,
            obs.all.ln_NH3_mean,
            obs.all.ln_NOX_mean,
            obs.all.ln_TN_mean,
            obs.all.ln_ORTHOP_mean,
            obs.all.ln_TP_mean,
            obs.all.ln_TSS_mean,
            obs.all.ln_BOD_mean))

##Add column names
colnames(all.obs2) <- c("TKN_mean",
                    "NH3_mean",
                    "NOX_mean",
                    "TN_mean",
                    "ORTHOP_mean",
                    "TP_mean",
                    "TSS_mean",
                    "BOD_mean",
                    "ln_TKN_mean",
                    "ln_NH3_mean",
                    "ln_NOX_mean",
                    "ln_TN_mean",
                    "ln_ORTHOP_mean",
                    "ln_TP_mean",
                    "ln_TSS_mean",
                    "ln_BOD_mean")
head(all.obs2)
```

```
  TKN_mean NH3_mean NOX_mean TN_mean ORTHOP_mean TP_mean TSS_mean BOD_mean
1   0.9643   0.1184   0.2049   1.165     0.08519  0.1505    5.273    1.977
  ln_TKN_mean ln_NH3_mean ln_NOX_mean ln_TN_mean ln_ORTHOP_mean ln_TP_mean
1     -0.1239      -2.735      -2.164    0.06656          -2.77     -2.072
  ln_TSS_mean ln_BOD_mean
1       1.237      0.5126
```

```
##Final calculation using the function exp
  all.geom_TKN_obs<-data.frame(geomean=exp(all.obs2$ln_TKN_mean))
  all.geom_NH3_obs<-data.frame(geomean=exp(all.obs2$ln_NH3_mean))
  all.geom_NOX_obs<-data.frame(geomean=exp(all.obs2$ln_NOX_mean))
  all.geom_TN_obs<-data.frame(geomean=exp(all.obs2$ln_TN_mean))
  all.geom_ORTHOP_obs<-data.frame(geomean=exp(all.obs2$ln_ORTHOP_mean))
  all.geom_TP_obs<-data.frame(geomean=exp(all.obs2$ln_TP_mean))
  all.geom_TSS_obs<-data.frame(geomean=exp(all.obs2$ln_TSS_mean))
  all.geom_BOD_obs<-data.frame(geomean=exp(all.obs2$ln_BOD_mean))

head(all.geom_TKN_obs)
```

```
  geomean
1  0.8835
```

```
##Merge the geometric mean values into one data set using the function cbind
all.obs.geomean<-data.frame(cbind(all.geom_TKN_obs,
         all.geom_NH3_obs,
            all.geom_NOX_obs,
            all.geom_TN_obs,
            all.geom_ORTHOP_obs,
            all.geom_TP_obs,
            all.geom_TSS_obs,
            all.geom_BOD_obs))


##Add column names
```

```
colnames(all.obs.geomean) <- c("TKN",
                   "NH3",
                   "NOX",
                   "TN",
                   "ORTHOP",
                   "TP",
                   "TSS",
                   "BOD")
head(all.obs.geomean)
```

```
     TKN     NH3    NOX    TN  ORTHOP    TP   TSS  BOD
1 0.8835 0.06492 0.1149 1.069 0.06268 0.126 3.445 1.67
```

```
##transpose data set
trans.otb.geomean.obs<-data.frame(Geometric_mean=t(all.obs.geomean))

head(trans.otb.geomean.obs)
```

```
       Geometric_mean
TKN           0.88349
NH3           0.06492
NOX           0.11491
TN            1.06883
ORTHOP        0.06268
TP            0.12599
```

```
##Export data
write.csv(trans.otb.geomean.obs,"C:/Users/john/Documents/old tampa bay/water optimal
model/bf/otb_geomean_observed.csv",row.names=TRUE)
```

Appendix D ts.R

Bridgette Froeschke — *Dec 2, 2013, 10:57 AM*

```
##R Manual for Old Tampa Bay Integreated Model System, Watershed Model:
##Objective:  Construct time series plots from the calibration report using an open resource data
analysis package.

##libraries
library(ggplot2)


##Import data
sim_obs<-read.csv("C:/Users/john/Documents/old tampa bay/water optimal model/bf/5 Appendix D
timeseries predicted & observed/sim_obs_run20_bydate_plot.csv")

head(sim_obs)
```

```
  REACH      date                                 X_LABEL_ parameter
1    60 01JAN2000 Modeled Biological Oxygen Demand 5 (mg/L)       BOD
2    60 01JAN2000                    Modeled Ammonia (mg/L)       NH3
3    60 01JAN2000            Modeled Nitrate Nitrite (mg/L)       NOX
4    60 01JAN2000           Modeled Ortho Phosphorus (mg/L)    ORTHOP
5    60 01JAN2000    Modeled Total Kjeldahl Nitrogen (mg/L)       TKN
6    60 01JAN2000             Modeled Total Nitrogen (mg/L)        TN
  simulated observed OP ABSOP OP2 year month        season
1        NA       NA NA    NA  NA 2000     1 Dry (Nov-May)
2        NA       NA NA    NA  NA 2000     1 Dry (Nov-May)
3        NA       NA NA    NA  NA 2000     1 Dry (Nov-May)
4        NA       NA NA    NA  NA 2000     1 Dry (Nov-May)
5        NA       NA NA    NA  NA 2000     1 Dry (Nov-May)
6        NA       NA NA    NA  NA 2000     1 Dry (Nov-May)
```

```
dim(sim_obs)
```

```
[1] 467584     12
```

```
##Subset data, remove Ln_BOD, Ln_NH3,Ln_NOX,Ln_ORTHOP,Ln_TKN,Ln_TN,Ln_TP,Ln_TSS
all<-subset(sim_obs,sim_obs$parameter=="BOD"|
               sim_obs$parameter=="NH3"|
               sim_obs$parameter=="NOX"|
               sim_obs$parameter=="ORTHOP"|
               sim_obs$parameter=="TKN"|
               sim_obs$parameter=="TN"|
               sim_obs$parameter=="TP"|
               sim_obs$parameter=="TSS")
head(all)
```

```
  REACH      date                                 X_LABEL_ parameter
1    60 01JAN2000 Modeled Biological Oxygen Demand 5 (mg/L)       BOD
2    60 01JAN2000                    Modeled Ammonia (mg/L)       NH3
3    60 01JAN2000            Modeled Nitrate Nitrite (mg/L)       NOX
4    60 01JAN2000           Modeled Ortho Phosphorus (mg/L)    ORTHOP
5    60 01JAN2000    Modeled Total Kjeldahl Nitrogen (mg/L)       TKN
6    60 01JAN2000             Modeled Total Nitrogen (mg/L)        TN
  simulated observed OP ABSOP OP2 year month        season
1        NA       NA NA    NA  NA 2000     1 Dry (Nov-May)
2        NA       NA NA    NA  NA 2000     1 Dry (Nov-May)
3        NA       NA NA    NA  NA 2000     1 Dry (Nov-May)
4        NA       NA NA    NA  NA 2000     1 Dry (Nov-May)
5        NA       NA NA    NA  NA 2000     1 Dry (Nov-May)
6        NA       NA NA    NA  NA 2000     1 Dry (Nov-May)
```

```
##sort by reach, x_label,date
all<-data.frame(all[order(all$REACH,all$X_LABEL_,all$date),])

head(all)
```

```
      REACH      date       X_LABEL_ parameter simulated observed      OP
4626     60 01AUG2001 Ammonia (mg/L)       NH3        NA    0.100      NA
11442    60 01DEC2003 Ammonia (mg/L)       NH3        NA    0.010      NA
19970    60 01NOV2006 Ammonia (mg/L)       NH3        NA    0.100      NA
13642    60 01SEP2004 Ammonia (mg/L)       NH3    0.0147    0.075 -0.0603
7306     60 02JUL2002 Ammonia (mg/L)       NH3    0.0196    0.070 -0.0504
6586     60 03APR2002 Ammonia (mg/L)       NH3        NA    0.560      NA
       ABSOP      OP2 year month        season
4626      NA       NA 2001     8 Wet (Jun-Oct)
11442     NA       NA 2003    12 Dry (Nov-May)
19970     NA       NA 2006    11 Dry (Nov-May)
13642 0.0603 0.003636 2004     9 Wet (Jun-Oct)
7306  0.0504 0.002540 2002     7 Wet (Jun-Oct)
6586      NA       NA 2002     4 Dry (Nov-May)
```

```
##add water body name column
all$name<-"Bishop/Mullet"
all$name[all$REACH==60] <- "Bishop/Mullet"
all$name[all$REACH==63] <- "Alligator/Allens/Long Branch"
all$name[all$REACH==65] <- "Roosevelt"
all$name[all$REACH==68] <- "Double Branch"
all$name[all$REACH==76] <- "Rocky Creek"
all$name[all$REACH==77] <- "Lower Sweetwater Creek"
all$name[all$REACH==227] <- "Channel A"
all$name[all$REACH==229] <- "Channel G"

head(all)
```

```
      REACH      date       X_LABEL_ parameter simulated observed      OP
4626     60 01AUG2001 Ammonia (mg/L)       NH3        NA    0.100      NA
11442    60 01DEC2003 Ammonia (mg/L)       NH3        NA    0.010      NA
19970    60 01NOV2006 Ammonia (mg/L)       NH3        NA    0.100      NA
13642    60 01SEP2004 Ammonia (mg/L)       NH3    0.0147    0.075 -0.0603
7306     60 02JUL2002 Ammonia (mg/L)       NH3    0.0196    0.070 -0.0504
6586     60 03APR2002 Ammonia (mg/L)       NH3        NA    0.560      NA
       ABSOP      OP2 year month        season          name
4626      NA       NA 2001     8 Wet (Jun-Oct) Bishop/Mullet
11442     NA       NA 2003    12 Dry (Nov-May) Bishop/Mullet
19970     NA       NA 2006    11 Dry (Nov-May) Bishop/Mullet
13642 0.0603 0.003636 2004     9 Wet (Jun-Oct) Bishop/Mullet
7306  0.0504 0.002540 2002     7 Wet (Jun-Oct) Bishop/Mullet
6586      NA       NA 2002     4 Dry (Nov-May) Bishop/Mullet
```

```
##sort by name, parameter,date
all<-data.frame(all[order(all$name,all$parameter,all$date),])

head(all)
```

```
      REACH      date                                X_LABEL_ parameter
29953    63 01APR2000 Modeled Biological Oxygen Demand 5 (mg/L)       BOD
32873    63 01APR2001 Modeled Biological Oxygen Demand 5 (mg/L)       BOD
35793    63 01APR2002 Modeled Biological Oxygen Demand 5 (mg/L)       BOD
38713    63 01APR2003         Biological Oxygen Demand 5 (mg/L)       BOD
41641    63 01APR2004 Modeled Biological Oxygen Demand 5 (mg/L)       BOD
44561    63 01APR2005 Modeled Biological Oxygen Demand 5 (mg/L)       BOD
      simulated observed OP ABSOP OP2 year month        season
29953        NA       NA NA    NA  NA 2000     4 Dry (Nov-May)
32873     4.067       NA NA    NA  NA 2001     4 Dry (Nov-May)
```

```
35793        NA       NA NA    NA  NA 2002     4 Dry (Nov-May)
38713     2.828       NA NA    NA  NA 2003     4 Dry (Nov-May)
41641     2.732       NA NA    NA  NA 2004     4 Dry (Nov-May)
44561     3.154       NA NA    NA  NA 2005     4 Dry (Nov-May)
                             name
29953 Alligator/Allens/Long Branch
32873 Alligator/Allens/Long Branch
35793 Alligator/Allens/Long Branch
38713 Alligator/Allens/Long Branch
41641 Alligator/Allens/Long Branch
44561 Alligator/Allens/Long Branch
```

```
##Create new data set for each set (observed and stimulated), bind the two sets together.  This prevents R from deleting an entire row when
##there is a missing value
obs <-data.frame(REACH=all$REACH, parameter=all$parameter,dv=all$observed, method="observed", label=all$X_LABEL_, name=all$name,year=all$year,month=all$month)
head(obs)
```

```
  REACH parameter dv   method                                    label
1    63       BOD NA observed Modeled Biological Oxygen Demand 5 (mg/L)
2    63       BOD NA observed Modeled Biological Oxygen Demand 5 (mg/L)
3    63       BOD NA observed Modeled Biological Oxygen Demand 5 (mg/L)
4    63       BOD NA observed         Biological Oxygen Demand 5 (mg/L)
5    63       BOD NA observed Modeled Biological Oxygen Demand 5 (mg/L)
6    63       BOD NA observed Modeled Biological Oxygen Demand 5 (mg/L)
                          name year month
1 Alligator/Allens/Long Branch 2000     4
2 Alligator/Allens/Long Branch 2001     4
3 Alligator/Allens/Long Branch 2002     4
4 Alligator/Allens/Long Branch 2003     4
5 Alligator/Allens/Long Branch 2004     4
6 Alligator/Allens/Long Branch 2005     4
```

```
sim <-data.frame(REACH=all$REACH, parameter=all$parameter,dv=all$simulated, method="simulated", ts=all$X_LABEL_, name=all$name,year=all$year,month=all$month)
head(sim)
```

```
  REACH parameter    dv    method
1    63       BOD    NA simulated
2    63       BOD 4.067 simulated
3    63       BOD    NA simulated
4    63       BOD 2.828 simulated
5    63       BOD 2.732 simulated
6    63       BOD 3.154 simulated
                                       ts                         name
1 Modeled Biological Oxygen Demand 5 (mg/L) Alligator/Allens/Long Branch
2 Modeled Biological Oxygen Demand 5 (mg/L) Alligator/Allens/Long Branch
3 Modeled Biological Oxygen Demand 5 (mg/L) Alligator/Allens/Long Branch
4         Biological Oxygen Demand 5 (mg/L) Alligator/Allens/Long Branch
5 Modeled Biological Oxygen Demand 5 (mg/L) Alligator/Allens/Long Branch
6 Modeled Biological Oxygen Demand 5 (mg/L) Alligator/Allens/Long Branch
  year month
1 2000     4
2 2001     4
3 2002     4
4 2003     4
5 2004     4
6 2005     4
```

```
##Variables to define, the only part that needs to be changed for the remainder of the code
PARAMETER="TP"
Reach=63
```

```
##Subset by Reach and Parameter
ts.obs.data <- subset(obs, obs$REACH==Reach & obs$parameter==PARAMETER)
ts.sim.data<-subset(sim,sim$REACH==Reach & sim$parameter==PARAMETER)

##Aggregrate and calculate means by year and month
##Observed
aggdata <-suppressWarnings(aggregate(ts.obs.data, by=list(ts.obs.data$year,ts.obs.data$month),
  FUN=mean, na.rm=TRUE))
head(aggdata)
```

```
  Group.1 Group.2 REACH parameter      dv method label name year month
1    2000       1    63        NA 0.10833     NA    NA   NA 2000     1
2    2001       1    63        NA 0.17333     NA    NA   NA 2001     1
3    2002       1    63        NA 0.13250     NA    NA   NA 2002     1
4    2003       1    63        NA 0.08133     NA    NA   NA 2003     1
5    2004       1    63        NA 0.11556     NA    NA   NA 2004     1
6    2005       1    63        NA 0.11444     NA    NA   NA 2005     1
```

```
##Simulated
aggdata.sim <-suppressWarnings(aggregate(ts.sim.data,
by=list(ts.sim.data$year,ts.sim.data$month,ts.sim.data$parameter,ts.sim.data$name),
  FUN=mean, na.rm=TRUE))

head(aggdata.sim)
```

```
  Group.1 Group.2 Group.3                      Group.4 REACH parameter
1    2000       1      TP Alligator/Allens/Long Branch    63        NA
2    2001       1      TP Alligator/Allens/Long Branch    63        NA
3    2002       1      TP Alligator/Allens/Long Branch    63        NA
4    2003       1      TP Alligator/Allens/Long Branch    63        NA
5    2004       1      TP Alligator/Allens/Long Branch    63        NA
6    2005       1      TP Alligator/Allens/Long Branch    63        NA
      dv method ts name year month
1    NaN     NA NA   NA 2000     1
2    NaN     NA NA   NA 2001     1
3 0.1504     NA NA   NA 2002     1
4 0.1256     NA NA   NA 2003     1
5 0.1874     NA NA   NA 2004     1
6 0.1697     NA NA   NA 2005     1
```

```
##Change year to decimal year for plotting
##Observed
aggdata$dec.year <- aggdata$year + ((aggdata$month/12)-1/12)
head(aggdata)
```

```
  Group.1 Group.2 REACH parameter      dv method label name year month
1    2000       1    63        NA 0.10833     NA    NA   NA 2000     1
2    2001       1    63        NA 0.17333     NA    NA   NA 2001     1
3    2002       1    63        NA 0.13250     NA    NA   NA 2002     1
4    2003       1    63        NA 0.08133     NA    NA   NA 2003     1
5    2004       1    63        NA 0.11556     NA    NA   NA 2004     1
6    2005       1    63        NA 0.11444     NA    NA   NA 2005     1
  dec.year
1     2000
2     2001
3     2002
4     2003
5     2004
6     2005
```

```
##Simulated
aggdata.sim$dec.year <- aggdata.sim$year + ((aggdata.sim$month/12)-1/12)
head(aggdata.sim)
```

```
  Group.1 Group.2 Group.3                      Group.4 REACH parameter
1    2000       1      TP Alligator/Allens/Long Branch    63        NA
2    2001       1      TP Alligator/Allens/Long Branch    63        NA
3    2002       1      TP Alligator/Allens/Long Branch    63        NA
4    2003       1      TP Alligator/Allens/Long Branch    63        NA
5    2004       1      TP Alligator/Allens/Long Branch    63        NA
6    2005       1      TP Alligator/Allens/Long Branch    63        NA
      dv method ts name year month dec.year
1    NaN     NA NA   NA 2000     1     2000
2    NaN     NA NA   NA 2001     1     2001
3 0.1504     NA NA   NA 2002     1     2002
4 0.1256     NA NA   NA 2003     1     2003
5 0.1874     NA NA   NA 2004     1     2004
6 0.1697     NA NA   NA 2005     1     2005
```

```
##Sort by decimal year
##Observed
newdata <- aggdata[order(aggdata$dec.year),]
head(newdata)
```

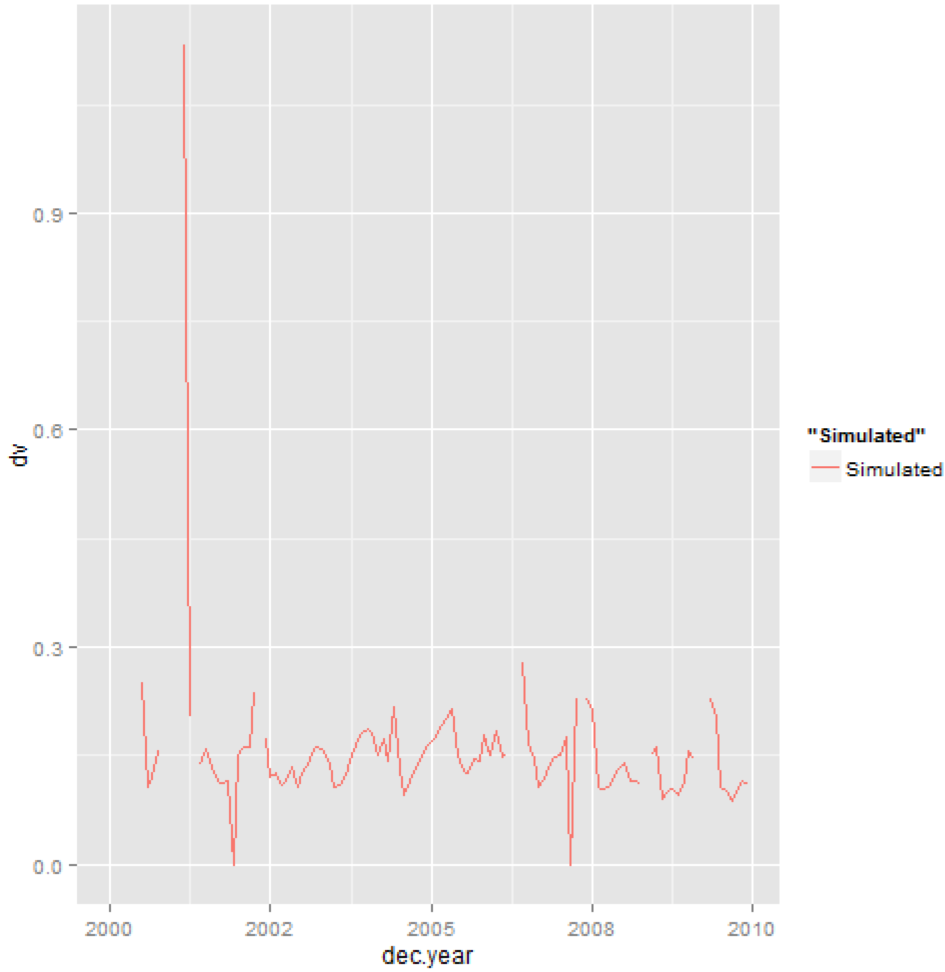
```
   Group.1 Group.2 REACH parameter     dv method label name year month
1     2000       1    63        NA 0.1083     NA    NA   NA 2000     1
11    2000       2    63        NA 0.1720     NA    NA   NA 2000     2
21    2000       3    63        NA 0.2583     NA    NA   NA 2000     3
31    2000       4    63        NA 0.1725     NA    NA   NA 2000     4
41    2000       5    63        NA 0.2250     NA    NA   NA 2000     5
51    2000       6    63        NA 0.3233     NA    NA   NA 2000     6
   dec.year
1      2000
11     2000
21     2000
31     2000
41     2000
51     2000
```

```
##Simulated
newdata.sim <- aggdata.sim[order(aggdata.sim$dec.year),]
head(newdata.sim)
```

```
   Group.1 Group.2 Group.3                      Group.4 REACH parameter
1     2000       1      TP Alligator/Allens/Long Branch    63        NA
11    2000       2      TP Alligator/Allens/Long Branch    63        NA
21    2000       3      TP Alligator/Allens/Long Branch    63        NA
31    2000       4      TP Alligator/Allens/Long Branch    63        NA
41    2000       5      TP Alligator/Allens/Long Branch    63        NA
51    2000       6      TP Alligator/Allens/Long Branch    63        NA
    dv method ts name year month dec.year
1  NaN     NA NA   NA 2000     1     2000
11 NaN     NA NA   NA 2000     2     2000
21 NaN     NA NA   NA 2000     3     2000
31 NaN     NA NA   NA 2000     4     2000
41 NaN     NA NA   NA 2000     5     2000
51 NaN     NA NA   NA 2000     6     2000
```

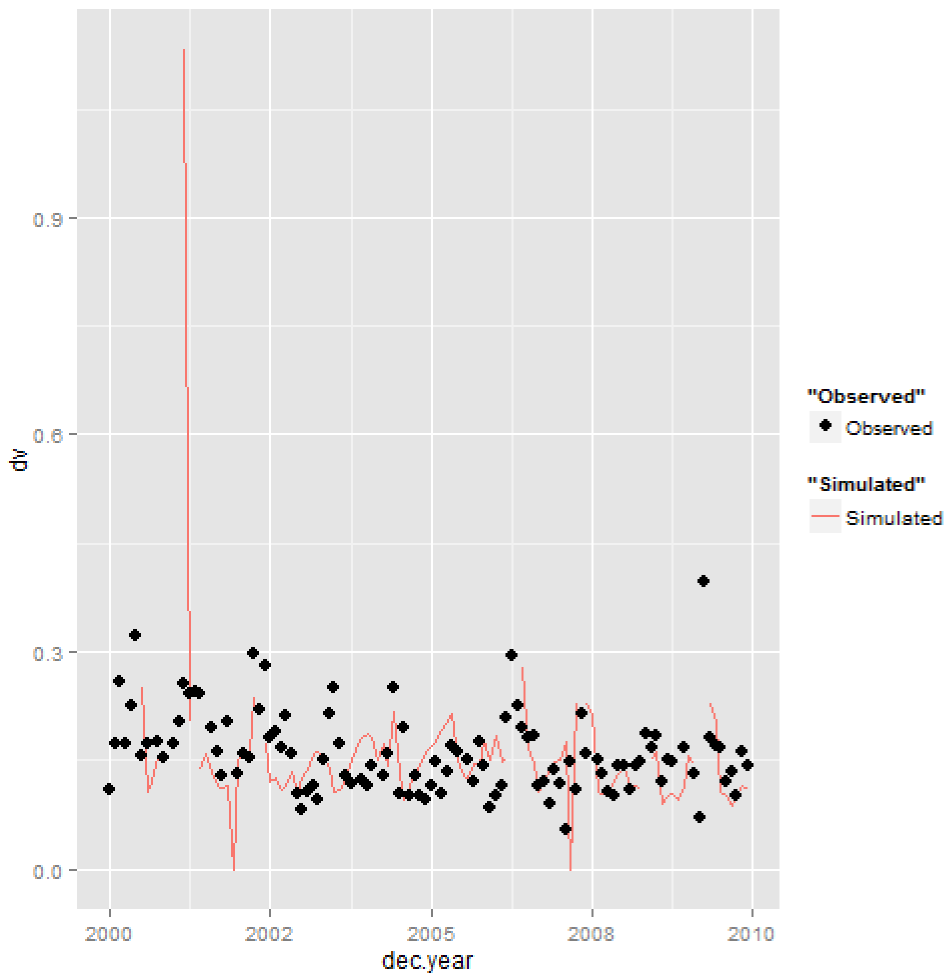
```
##Plot
##Simualted
YLAB=ts.obs.data$parameter[[1]]
ts.p1<-ggplot(newdata.sim, aes(x=dec.year, y=dv,colour="Simulated")) + geom_line(shape="line")
ts.p1
```

```
Warning: Removed 6 rows containing missing values (geom_path).
```



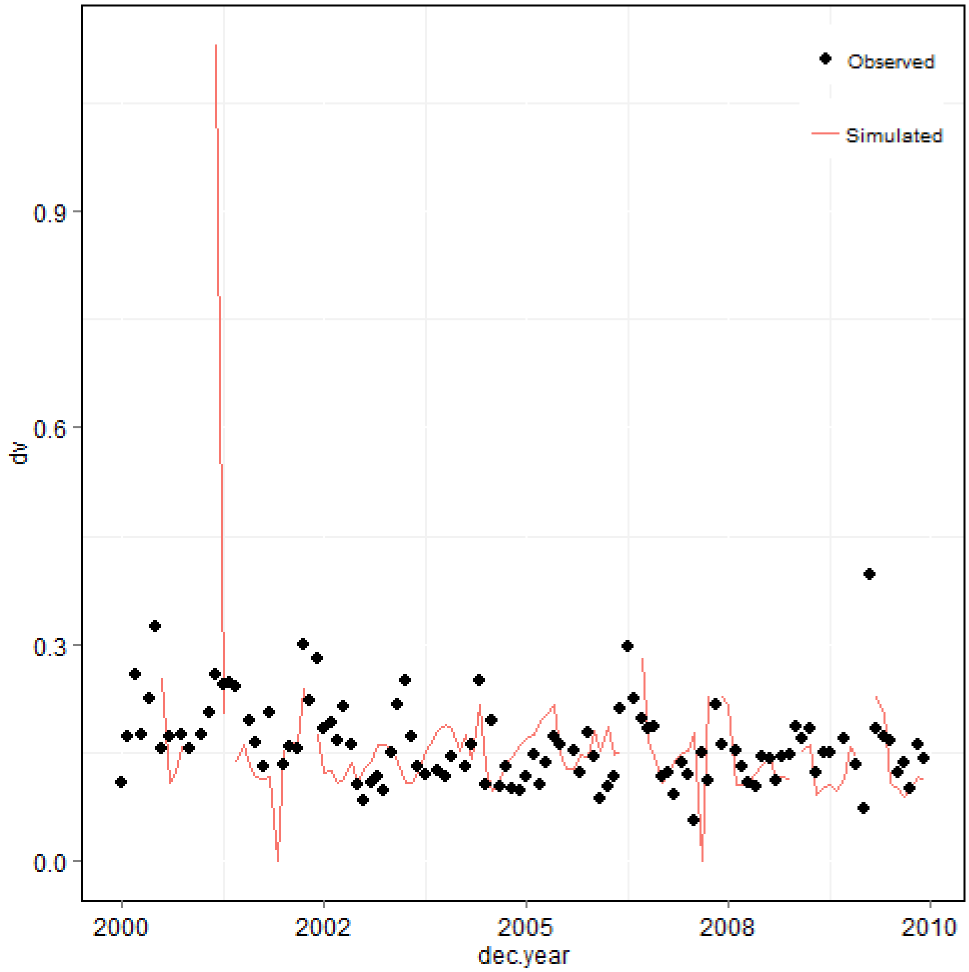

```
##Observed
ts.p2<-ts.p1 + geom_point(aes(x=dec.year, y=dv,
shape="Observed"),data=newdata,size=3,colour="black")
ts.p2
```

```
Warning: Removed 6 rows containing missing values (geom_path). Warning:
Removed 10 rows containing missing values (geom_point).
```



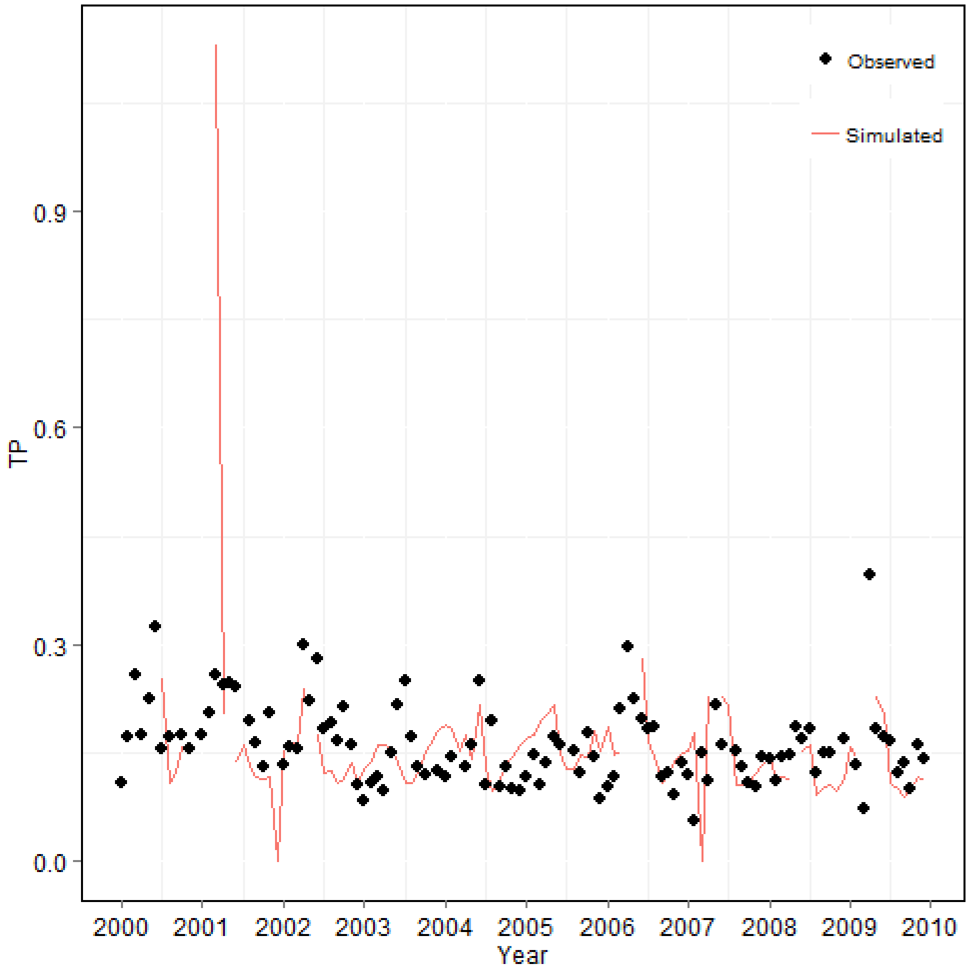

```
##specifiy the theme for the plot, this theme removes the default gray background and adds axes
lines
ts.p3<-ts.p2 + theme(legend.title=element_blank())+ theme(legend.justification=c(1,1),
legend.position=c(1,1))+
  theme(panel.border=element_rect(colour="black", fill=NA))+
  theme(axis.text = element_text(colour = "black",
                                 size = 12))+
  theme(panel.background = element_rect(fill = NA,
                                        colour = "black"))+
  theme(legend.key=element_rect(fill=NA))

ts.p3
```

```
Warning: Removed 6 rows containing missing values (geom_path). Warning:
Removed 10 rows containing missing values (geom_point).
```



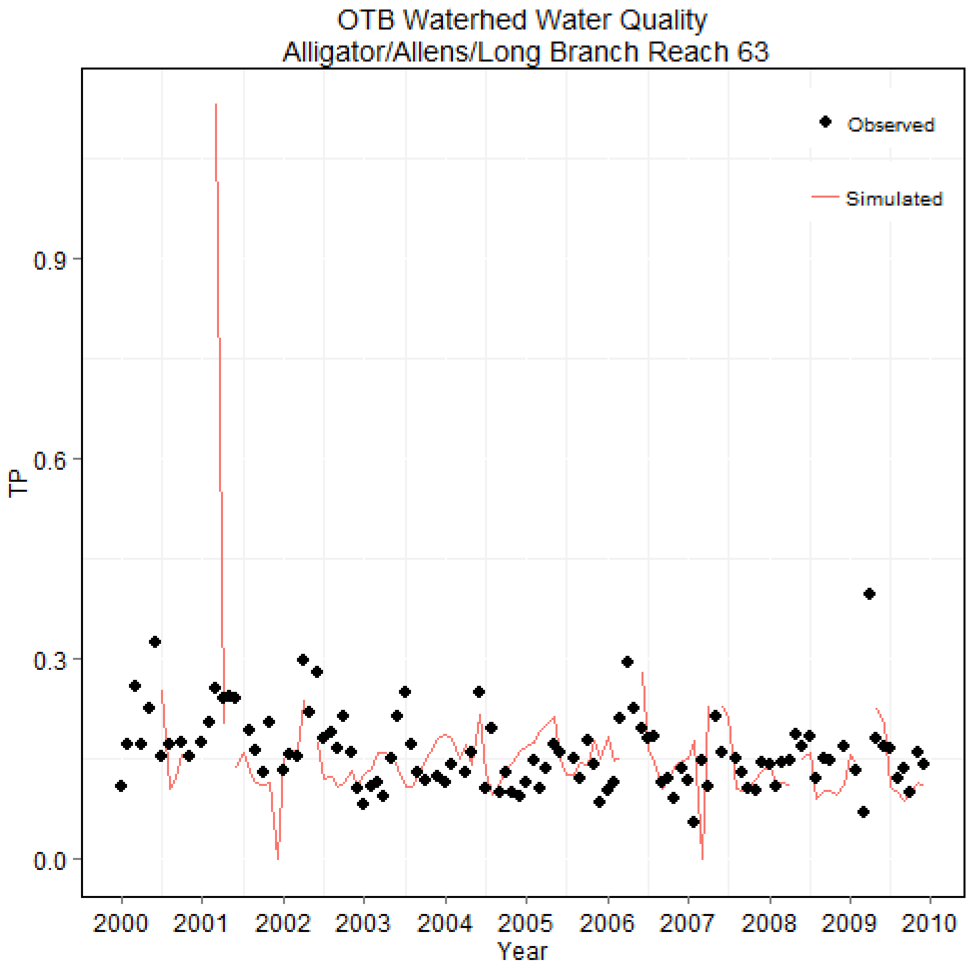

```
##add the x and y labels
ts.p4<-ts.p3 + scale_x_continuous(breaks=c(2000, 2001, 2002, 2003, 2004, 2005,
                  2006, 2007,2008,2009,2010)) + xlab("Year") +ylab(YLAB)
ts.p4
```

```
Warning: Removed 6 rows containing missing values (geom_path). Warning:
Removed 10 rows containing missing values (geom_point).
```



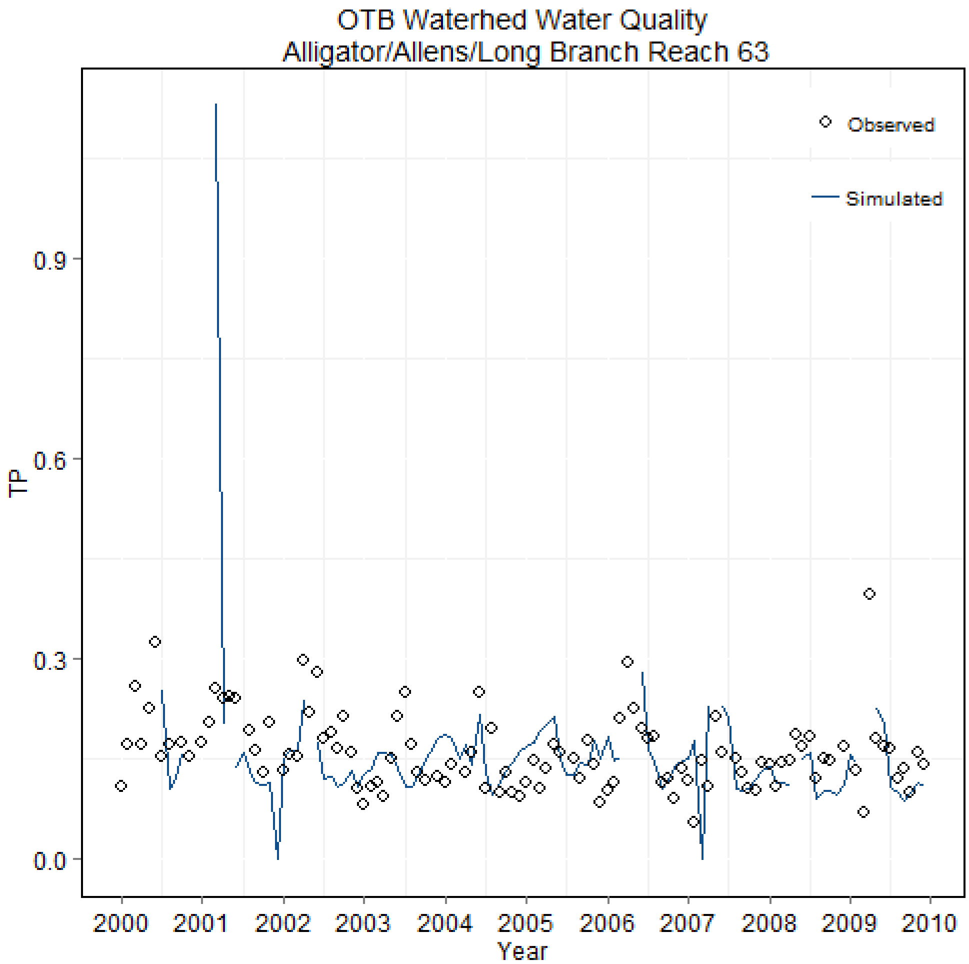

```
##add title
ts.p5 <- ts.p4 + ggtitle(paste("OTB Waterhed Water Quality\n", ts.sim.data$name[1],"Reach",
Reach, sep=" ")  )
ts.p5
```

```
Warning: Removed 6 rows containing missing values (geom_path). Warning:
Removed 10 rows containing missing values (geom_point).
```




```
##clean up legend
ts.p6<-ts.p5+ scale_colour_manual(values=c("dodgerblue4"))+
  scale_shape_manual(values=1)

ts.p6
```

```
Warning: Removed 6 rows containing missing values (geom_path). Warning:
Removed 10 rows containing missing values (geom_point).
```




```
##save and export plot
plot.filename <- paste0("C:/Users/john/Documents/old tampa bay/water optimal model/bf/5 Appendix D
timeseries predicted & observed/", PARAMETER,Reach,".pdf")
ggsave(ts.p6, filename=plot.filename, height=8,width=8)
```

```
Warning: Removed 6 rows containing missing values (geom_path). Warning:
Removed 10 rows containing missing values (geom_point).
```

```
##Redefine REACH and parameter
PARAMETER="TN"
Reach=63


##Subset by Reach and Parameter
ts.obs.data <- subset(obs, obs$REACH==Reach & obs$parameter==PARAMETER)
ts.sim.data<-subset(sim,sim$REACH==Reach & sim$parameter==PARAMETER)

##Aggregate and calculate means by year and month
##Observed
aggdata <-suppressWarnings(aggregate(ts.obs.data, by=list(ts.obs.data$year,ts.obs.data$month),
  FUN=mean, na.rm=TRUE))
head(aggdata)
```

```
  Group.1 Group.2 REACH parameter     dv method label name year month
1    2000       1    63        NA 0.6567     NA    NA   NA 2000     1
2    2001       1    63        NA 1.1067     NA    NA   NA 2001     1
3    2002       1    63        NA 1.0575     NA    NA   NA 2002     1
4    2003       1    63        NA 1.3507     NA    NA   NA 2003     1
```

```
5    2004       1    63         NA 0.9172     NA     NA   NA 2004     1
6    2005       1    63         NA 0.8322     NA     NA   NA 2005     1
```

```
##Simulated
aggdata.sim <-suppressWarnings(aggregate(ts.sim.data,
by=list(ts.sim.data$year,ts.sim.data$month,ts.sim.data$parameter,ts.sim.data$name),
  FUN=mean, na.rm=TRUE))


head(aggdata.sim)
```

```
  Group.1 Group.2 Group.3                      Group.4 REACH parameter
1    2000       1      TN Alligator/Allens/Long Branch    63        NA
2    2001       1      TN Alligator/Allens/Long Branch    63        NA
3    2002       1      TN Alligator/Allens/Long Branch    63        NA
4    2003       1      TN Alligator/Allens/Long Branch    63        NA
5    2004       1      TN Alligator/Allens/Long Branch    63        NA
6    2005       1      TN Alligator/Allens/Long Branch    63        NA
      dv method ts name year month
1    NaN     NA NA   NA 2000     1
2    NaN     NA NA   NA 2001     1
3 1.1109     NA NA   NA 2002     1
4 0.7984     NA NA   NA 2003     1
5 1.3199     NA NA   NA 2004     1
6 1.1442     NA NA   NA 2005     1
```

```
##Change year to decimal year for plotting
##Observed
aggdata$dec.year <- aggdata$year + ((aggdata$month/12)-1/12)
head(aggdata)
```

```
  Group.1 Group.2 REACH parameter     dv method label name year month
1    2000       1    63         NA 0.6567     NA     NA   NA 2000     1
2    2001       1    63         NA 1.1067     NA     NA   NA 2001     1
3    2002       1    63         NA 1.0575     NA     NA   NA 2002     1
4    2003       1    63         NA 1.3507     NA     NA   NA 2003     1
5    2004       1    63         NA 0.9172     NA     NA   NA 2004     1
6    2005       1    63         NA 0.8322     NA     NA   NA 2005     1
  dec.year
1     2000
2     2001
3     2002
4     2003
5     2004
6     2005
```

```
##Simulated
aggdata.sim$dec.year <- aggdata.sim$year + ((aggdata.sim$month/12)-1/12)
head(aggdata.sim)
```

```
  Group.1 Group.2 Group.3                      Group.4 REACH parameter
1    2000       1      TN Alligator/Allens/Long Branch    63        NA
2    2001       1      TN Alligator/Allens/Long Branch    63        NA
3    2002       1      TN Alligator/Allens/Long Branch    63        NA
4    2003       1      TN Alligator/Allens/Long Branch    63        NA
5    2004       1      TN Alligator/Allens/Long Branch    63        NA
6    2005       1      TN Alligator/Allens/Long Branch    63        NA
      dv method ts name year month dec.year
1    NaN     NA NA   NA 2000     1     2000
2    NaN     NA NA   NA 2001     1     2001
3 1.1109     NA NA   NA 2002     1     2002
4 0.7984     NA NA   NA 2003     1     2003
5 1.3199     NA NA   NA 2004     1     2004
6 1.1442     NA NA   NA 2005     1     2005
```

```
##Sort by decimal year
##Observed
newdata <- aggdata[order(aggdata$dec.year),]
head(newdata)
```

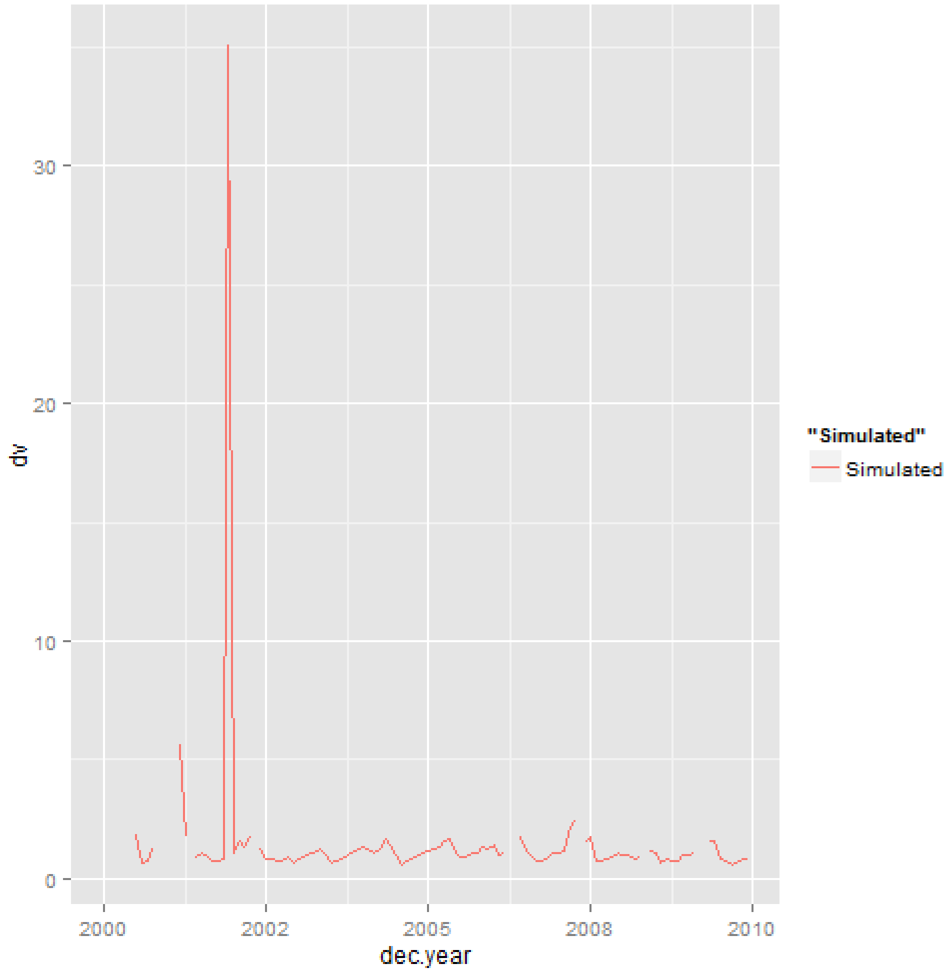
```
   Group.1 Group.2 REACH parameter     dv method label name year month
1     2000       1    63        NA 0.6567     NA    NA   NA 2000     1
11    2000       2    63        NA 0.8720     NA    NA   NA 2000     2
21    2000       3    63        NA 1.6200     NA    NA   NA 2000     3
31    2000       4    63        NA 0.8150     NA    NA   NA 2000     4
41    2000       5    63        NA 1.0775     NA    NA   NA 2000     5
51    2000       6    63        NA 1.5800     NA    NA   NA 2000     6
   dec.year
1      2000
11     2000
21     2000
31     2000
41     2000
51     2000
```

```
##Simulated
newdata.sim <- aggdata.sim[order(aggdata.sim$dec.year),]
head(newdata.sim)
```

```
   Group.1 Group.2 Group.3                     Group.4 REACH parameter
1     2000       1      TN Alligator/Allens/Long Branch    63        NA
11    2000       2      TN Alligator/Allens/Long Branch    63        NA
21    2000       3      TN Alligator/Allens/Long Branch    63        NA
31    2000       4      TN Alligator/Allens/Long Branch    63        NA
41    2000       5      TN Alligator/Allens/Long Branch    63        NA
51    2000       6      TN Alligator/Allens/Long Branch    63        NA
    dv method ts name year month dec.year
1  NaN     NA NA   NA 2000     1     2000
11 NaN     NA NA   NA 2000     2     2000
21 NaN     NA NA   NA 2000     3     2000
31 NaN     NA NA   NA 2000     4     2000
41 NaN     NA NA   NA 2000     5     2000
51 NaN     NA NA   NA 2000     6     2000
```

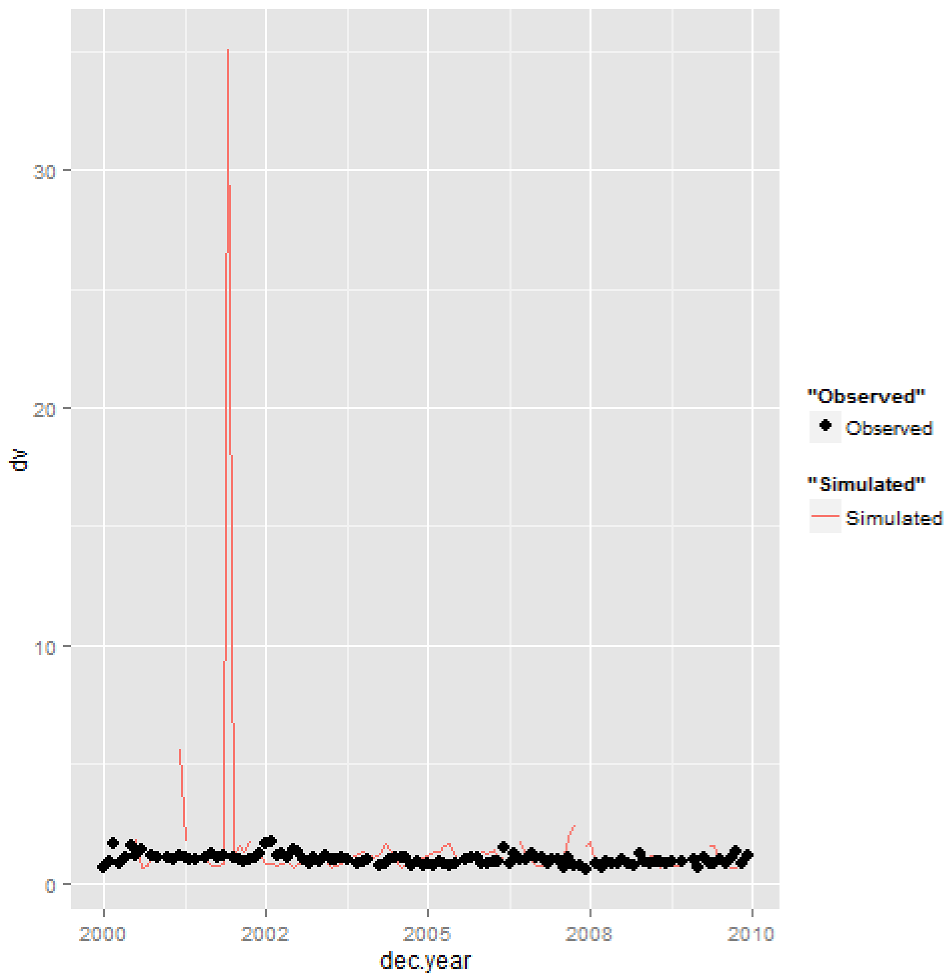
```
##Plot
##Simualted
YLAB=ts.obs.data$parameter[[1]]
ts.p1<-ggplot(newdata.sim, aes(x=dec.year, y=dv,colour="Simulated")) + geom_line(shape="line")
ts.p1
```

```
Warning: Removed 6 rows containing missing values (geom_path).
```



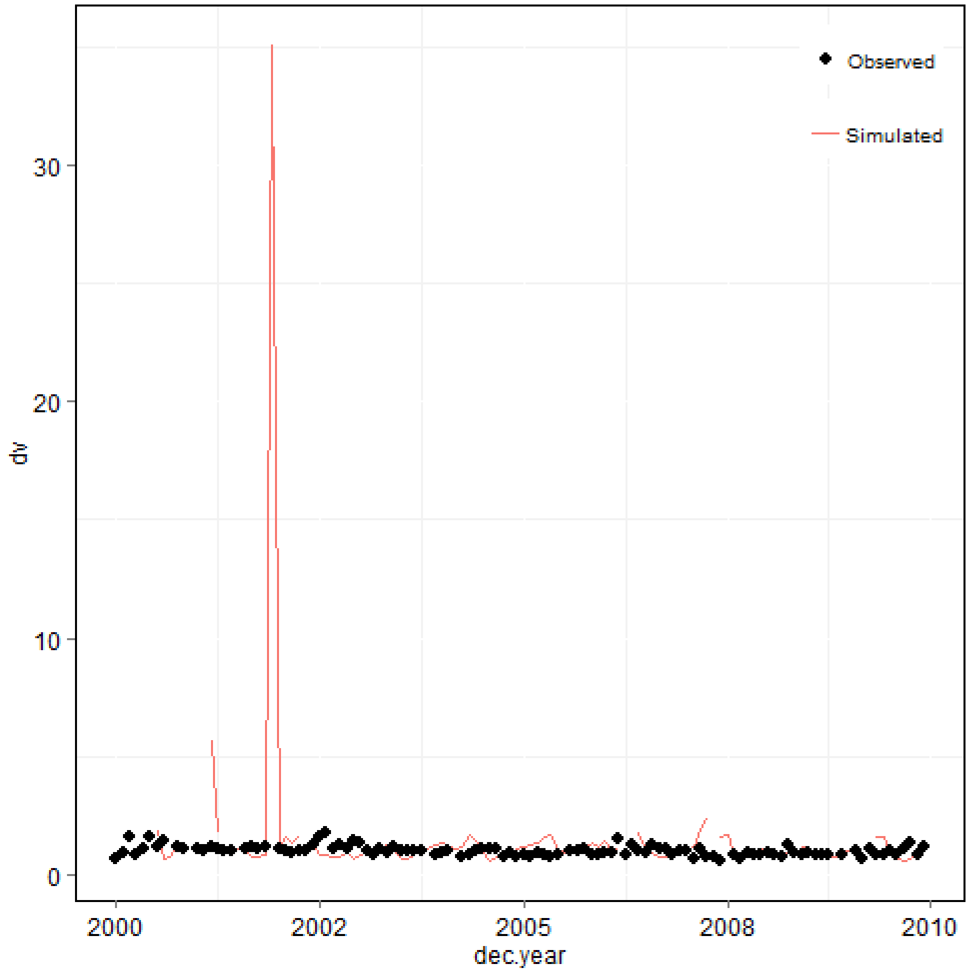

```
##Observed
ts.p2<-ts.p1 + geom_point(aes(x=dec.year, y=dv,
shape="Observed"),data=newdata,size=3,colour="black")
ts.p2
```

```
Warning: Removed 6 rows containing missing values (geom_path). Warning:
Removed 10 rows containing missing values (geom_point).
```



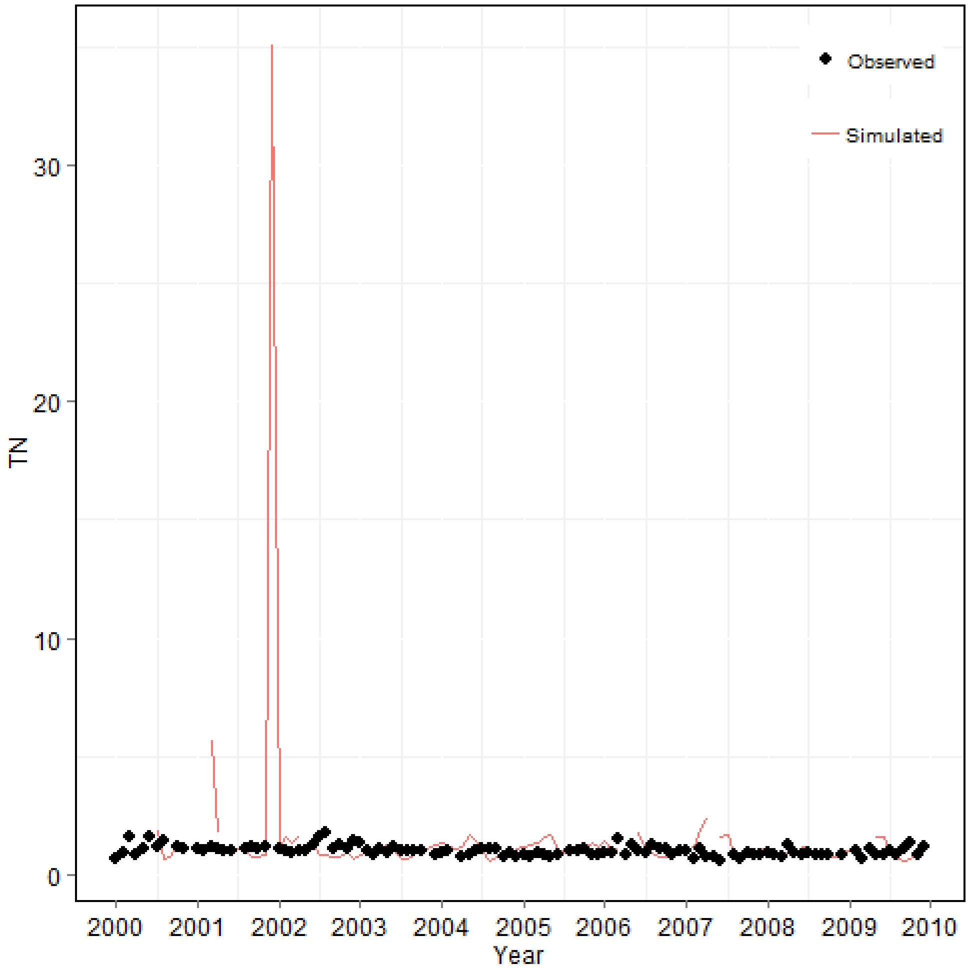

```
##specifiy the theme for the plot, this theme removes the default gray background and adds axes
lines
ts.p3<-ts.p2 + theme(legend.title=element_blank())+ theme(legend.justification=c(1,1),
legend.position=c(1,1))+
  theme(panel.border=element_rect(colour="black", fill=NA))+
  theme(axis.text = element_text(colour = "black",
                                 size = 12))+
  theme(panel.background = element_rect(fill = NA,
                                        colour = "black"))+
  theme(legend.key=element_rect(fill=NA))

ts.p3
```

```
Warning: Removed 6 rows containing missing values (geom_path). Warning:
Removed 10 rows containing missing values (geom_point).
```



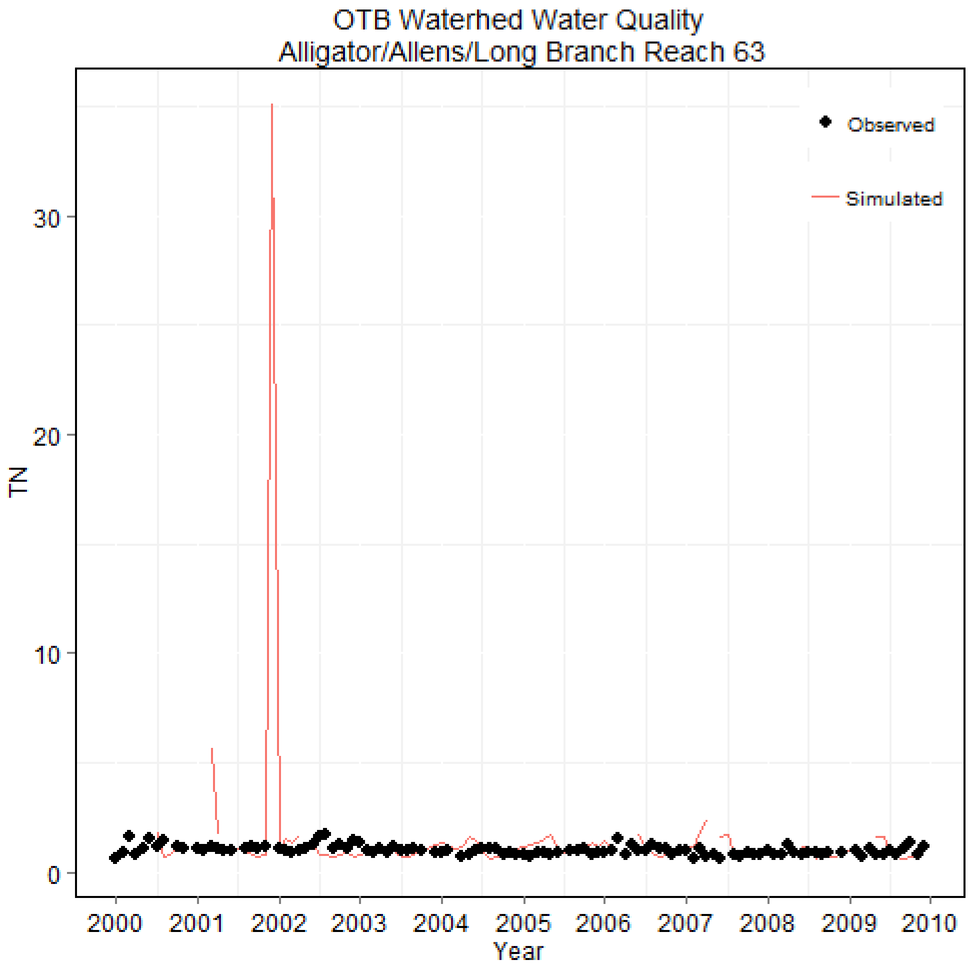

```
##add the x and y labels
ts.p4<-ts.p3 + scale_x_continuous(breaks=c(2000, 2001, 2002, 2003, 2004, 2005,
                  2006, 2007,2008,2009,2010)) + xlab("Year") +ylab(YLAB)
ts.p4
```

```
Warning: Removed 6 rows containing missing values (geom_path). Warning:
Removed 10 rows containing missing values (geom_point).
```



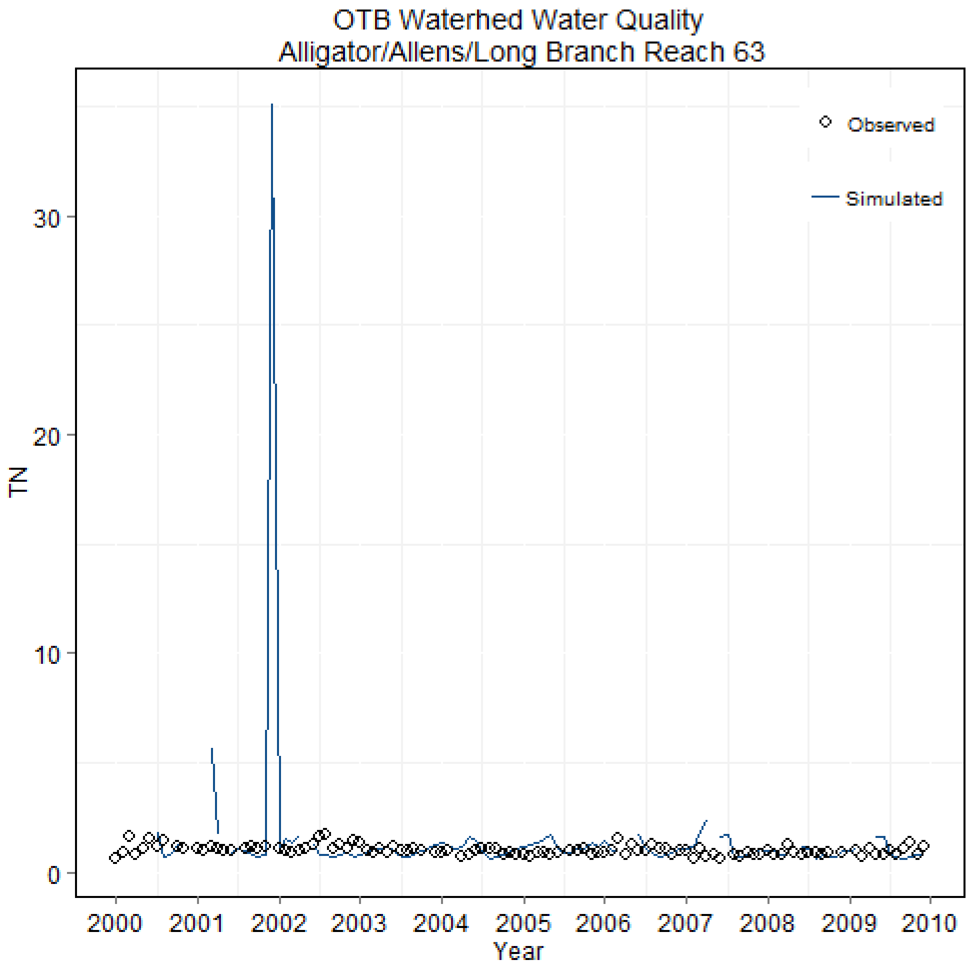

```
##add title
ts.p5 <- ts.p4 + ggtitle(paste("OTB Waterhed Water Quality\n", ts.sim.data$name[1],"Reach",
Reach, sep=" ")  )
ts.p5
```

```
Warning: Removed 6 rows containing missing values (geom_path). Warning:
Removed 10 rows containing missing values (geom_point).
```




```
##clean up legend
ts.p6<-ts.p5+ scale_colour_manual(values=c("dodgerblue4"))+
  scale_shape_manual(values=1)

ts.p6
```

```
Warning: Removed 6 rows containing missing values (geom_path). Warning:
Removed 10 rows containing missing values (geom_point).
```




```
##save and export plot
plot.filename <- paste0("C:/Users/john/Documents/old tampa bay/water optimal model/bf/5 Appendix D timeseries predicted & observed/", PARAMETER,Reach,".pdf")
ggsave(ts.p6, filename=plot.filename, height=8,width=8)
```

```
Warning: Removed 6 rows containing missing values (geom_path). Warning:
Removed 10 rows containing missing values (geom_point).
```

Appendix C cdfs.R

Bridgette Froeschke — *Dec 1, 2013, 7:02 PM*

```
##R Manual for Old Tampa Bay Integreated Model System, Watershed Model:
##Objective:  Construct cdf plots from the calibration report using an open resource data analysis
package.


##Import data
sim_obs<-read.csv("C:/Users/john/Documents/old tampa bay/water optimal model/bf/4 Appendix C
cdfs/CDFs_OTB/sim_obs_run20_bydate_plot.csv")
head(sim_obs)
```

```
  REACH      date                                 X_LABEL_ parameter
1    60 01JAN2000 Modeled Biological Oxygen Demand 5 (mg/L)       BOD
2    60 01JAN2000                   Modeled Ammonia (mg/L)       NH3
3    60 01JAN2000           Modeled Nitrate Nitrite (mg/L)       NOX
4    60 01JAN2000          Modeled Ortho Phosphorus (mg/L)    ORTHOP
5    60 01JAN2000    Modeled Total Kjeldahl Nitrogen (mg/L)       TKN
6    60 01JAN2000            Modeled Total Nitrogen (mg/L)        TN
  simulated observed OP ABSOP OP2 year month        season
1        NA       NA NA    NA  NA 2000     1 Dry (Nov-May)
2        NA       NA NA    NA  NA 2000     1 Dry (Nov-May)
3        NA       NA NA    NA  NA 2000     1 Dry (Nov-May)
4        NA       NA NA    NA  NA 2000     1 Dry (Nov-May)
5        NA       NA NA    NA  NA 2000     1 Dry (Nov-May)
6        NA       NA NA    NA  NA 2000     1 Dry (Nov-May)
```

```
dim(sim_obs)
```

```
[1] 467584     12
```

```


##Subset data, remove Ln_BOD, Ln_NH3,Ln_NOX,Ln_ORTHOP,Ln_TKN,Ln_TN,Ln_TP,Ln_TSS
all<-subset(sim_obs,sim_obs$parameter=="BOD"|
                sim_obs$parameter=="NH3"|
                sim_obs$parameter=="NOX"|
                sim_obs$parameter=="ORTHOP"|
                sim_obs$parameter=="TKN"|
                sim_obs$parameter=="TN"|
                sim_obs$parameter=="TP"|
                sim_obs$parameter=="TSS")
head(all)
```

```
  REACH      date                                 X_LABEL_ parameter
1    60 01JAN2000 Modeled Biological Oxygen Demand 5 (mg/L)       BOD
2    60 01JAN2000                   Modeled Ammonia (mg/L)       NH3
3    60 01JAN2000           Modeled Nitrate Nitrite (mg/L)       NOX
4    60 01JAN2000          Modeled Ortho Phosphorus (mg/L)    ORTHOP
5    60 01JAN2000    Modeled Total Kjeldahl Nitrogen (mg/L)       TKN
6    60 01JAN2000            Modeled Total Nitrogen (mg/L)        TN
  simulated observed OP ABSOP OP2 year month        season
1        NA       NA NA    NA  NA 2000     1 Dry (Nov-May)
2        NA       NA NA    NA  NA 2000     1 Dry (Nov-May)
3        NA       NA NA    NA  NA 2000     1 Dry (Nov-May)
4        NA       NA NA    NA  NA 2000     1 Dry (Nov-May)
5        NA       NA NA    NA  NA 2000     1 Dry (Nov-May)
6        NA       NA NA    NA  NA 2000     1 Dry (Nov-May)
```

```
##sort by reach, x_label,date
all<-data.frame(all[order(all$REACH,all$X_LABEL_,all$date),])
```

```
head(all)
```

```
      REACH      date       X_LABEL_ parameter simulated observed      OP
4626     60 01AUG2001 Ammonia (mg/L)       NH3        NA    0.100      NA
11442    60 01DEC2003 Ammonia (mg/L)       NH3        NA    0.010      NA
19970    60 01NOV2006 Ammonia (mg/L)       NH3        NA    0.100      NA
13642    60 01SEP2004 Ammonia (mg/L)       NH3    0.0147    0.075 -0.0603
7306     60 02JUL2002 Ammonia (mg/L)       NH3    0.0196    0.070 -0.0504
6586     60 03APR2002 Ammonia (mg/L)       NH3        NA    0.560      NA
       ABSOP      OP2 year month        season
4626      NA       NA 2001     8 Wet (Jun-Oct)
11442     NA       NA 2003    12 Dry (Nov-May)
19970     NA       NA 2006    11 Dry (Nov-May)
13642 0.0603 0.003636 2004     9 Wet (Jun-Oct)
7306  0.0504 0.002540 2002     7 Wet (Jun-Oct)
6586      NA       NA 2002     4 Dry (Nov-May)
```

```
##add water body name column
all$name[all$REACH==60] <- "Bishop/Mullet"
all$name[all$REACH==63] <- "Alligator/Allens/Long Branch"
all$name[all$REACH==65] <- "Roosevelt"
all$name[all$REACH==68] <- "Double Branch"
all$name[all$REACH==76] <- "Rocky Creek"
all$name[all$REACH==77] <- "Lower Sweetwater Creek"
all$name[all$REACH==223] <- "Lake Tarpon"
all$name[all$REACH==227] <- "Channel A"
all$name[all$REACH==229] <- "Channel G"

head(all)
```

```
      REACH      date       X_LABEL_ parameter simulated observed      OP
4626     60 01AUG2001 Ammonia (mg/L)       NH3        NA    0.100      NA
11442    60 01DEC2003 Ammonia (mg/L)       NH3        NA    0.010      NA
19970    60 01NOV2006 Ammonia (mg/L)       NH3        NA    0.100      NA
13642    60 01SEP2004 Ammonia (mg/L)       NH3    0.0147    0.075 -0.0603
7306     60 02JUL2002 Ammonia (mg/L)       NH3    0.0196    0.070 -0.0504
6586     60 03APR2002 Ammonia (mg/L)       NH3        NA    0.560      NA
       ABSOP      OP2 year month        season          name
4626      NA       NA 2001     8 Wet (Jun-Oct) Bishop/Mullet
11442     NA       NA 2003    12 Dry (Nov-May) Bishop/Mullet
19970     NA       NA 2006    11 Dry (Nov-May) Bishop/Mullet
13642 0.0603 0.003636 2004     9 Wet (Jun-Oct) Bishop/Mullet
7306  0.0504 0.002540 2002     7 Wet (Jun-Oct) Bishop/Mullet
6586      NA       NA 2002     4 Dry (Nov-May) Bishop/Mullet
```

```
##sort by name, parameter,date
all<-data.frame(all[order(all$name,all$parameter,all$date),])

head(all)
```

```
      REACH      date                                X_LABEL_ parameter
29953    63 01APR2000 Modeled Biological Oxygen Demand 5 (mg/L)       BOD
32873    63 01APR2001 Modeled Biological Oxygen Demand 5 (mg/L)       BOD
35793    63 01APR2002 Modeled Biological Oxygen Demand 5 (mg/L)       BOD
38713    63 01APR2003         Biological Oxygen Demand 5 (mg/L)       BOD
41641    63 01APR2004 Modeled Biological Oxygen Demand 5 (mg/L)       BOD
44561    63 01APR2005 Modeled Biological Oxygen Demand 5 (mg/L)       BOD
      simulated observed OP ABSOP OP2 year month        season
29953        NA       NA NA    NA  NA 2000     4 Dry (Nov-May)
32873     4.067       NA NA    NA  NA 2001     4 Dry (Nov-May)
35793        NA       NA NA    NA  NA 2002     4 Dry (Nov-May)
38713     2.828       NA NA    NA  NA 2003     4 Dry (Nov-May)
41641     2.732       NA NA    NA  NA 2004     4 Dry (Nov-May)
44561     3.154       NA NA    NA  NA 2005     4 Dry (Nov-May)
```

```
                             name
29953 Alligator/Allens/Long Branch
32873 Alligator/Allens/Long Branch
35793 Alligator/Allens/Long Branch
38713 Alligator/Allens/Long Branch
41641 Alligator/Allens/Long Branch
44561 Alligator/Allens/Long Branch
```

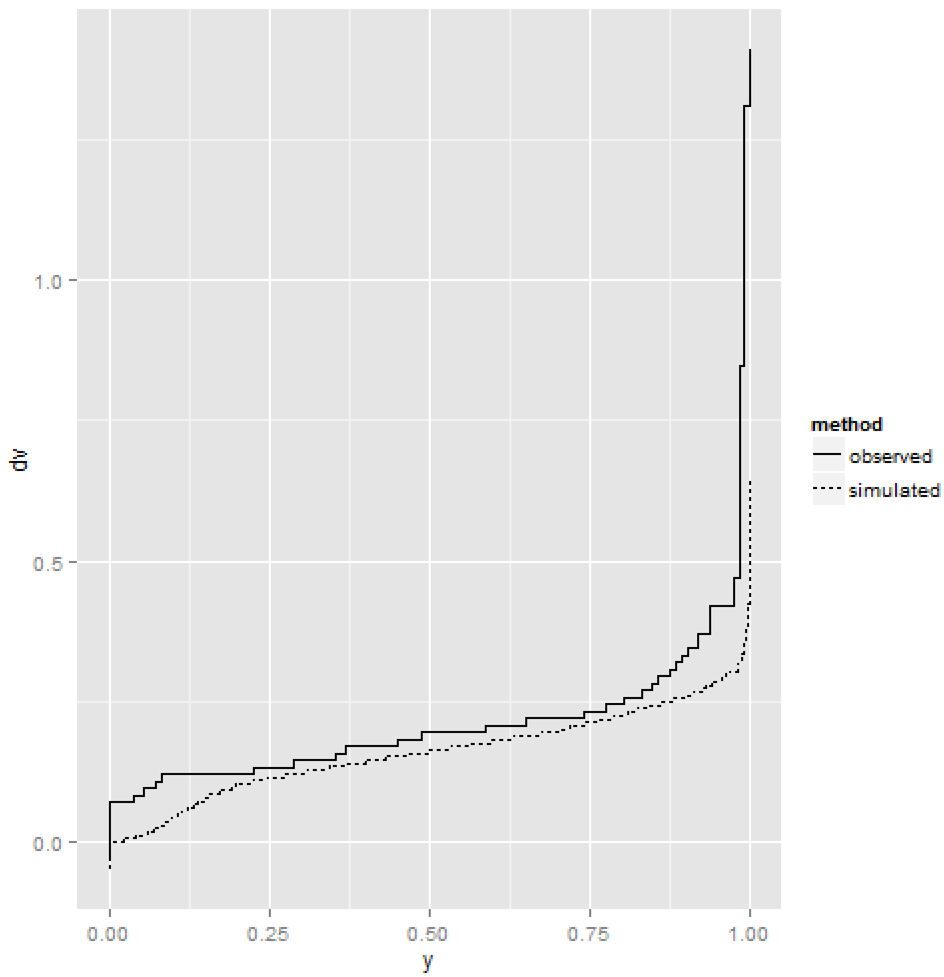
```
##Variables to define
PARAMETER="TP"
Reach=60

##Subset by Reach and Parameter
cdf.data <- subset(all, all$REACH==Reach & all$parameter==PARAMETER)

##Create new data set for each set (observed and stimulated), bind the two sets together.  This
prevents R from deleting an entire row when
##there is a missing value
obs <- na.omit(data.frame(REACH=Reach, parameter=PARAMETER,dv=cdf.data$observed,
method="observed", label=cdf.data$X_LABEL_, name=cdf.data$name))
sim <- na.omit(data.frame(REACH=Reach, parameter=PARAMETER,dv=cdf.data$simulated,
method="simulated", label=cdf.data$X_LABEL_, name=cdf.data$name))
both <- rbind(obs,sim)
head(both)
```

```
    REACH parameter    dv   method                  label          name
12     60        TP 0.240 observed Total Phosphorus (mg/L) Bishop/Mullet
24     60        TP 0.070 observed Total Phosphorus (mg/L) Bishop/Mullet
97     60        TP 0.250 observed Total Phosphorus (mg/L) Bishop/Mullet
115    60        TP 0.200 observed Total Phosphorus (mg/L) Bishop/Mullet
173    60        TP 0.295 observed Total Phosphorus (mg/L) Bishop/Mullet
243    60        TP 1.310 observed Total Phosphorus (mg/L) Bishop/Mullet
```

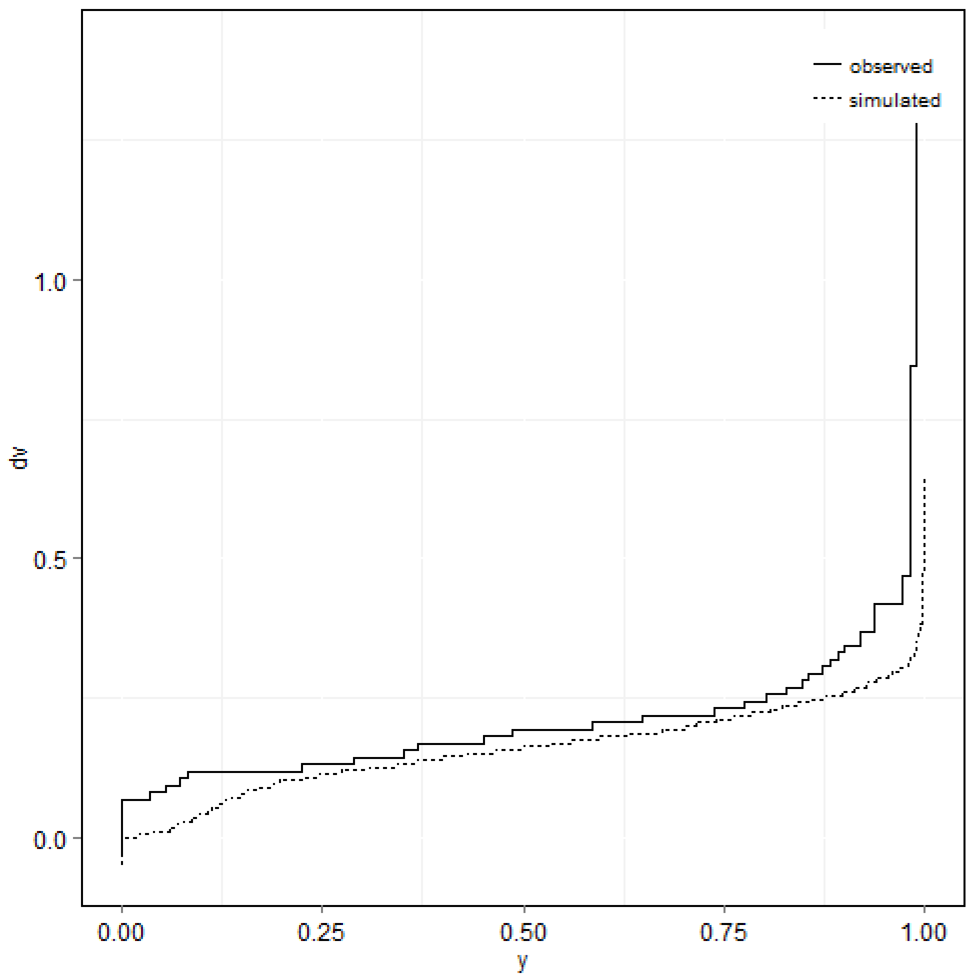
```
tail(both)
```

```
     REACH parameter     dv    method                          label
3621    60        TP 0.1866 simulated Modeled Total Phosphorus (mg/L)
3622    60        TP 0.0829 simulated Modeled Total Phosphorus (mg/L)
3623    60        TP 0.1309 simulated Modeled Total Phosphorus (mg/L)
3639    60        TP 0.2340 simulated Modeled Total Phosphorus (mg/L)
3646    60        TP 0.2148 simulated Modeled Total Phosphorus (mg/L)
3653    60        TP 0.1824 simulated Modeled Total Phosphorus (mg/L)
              name
3621 Bishop/Mullet
3622 Bishop/Mullet
3623 Bishop/Mullet
3639 Bishop/Mullet
3646 Bishop/Mullet
3653 Bishop/Mullet
```

```
##libraries
library(scales)##plots percentages
library(ggplot2)

##plot cdf values for both observed and stimulated using stat_ecdf
p1<-ggplot(both, aes(dv, linetype=method)) + stat_ecdf(n=100) + coord_flip()
p1
```

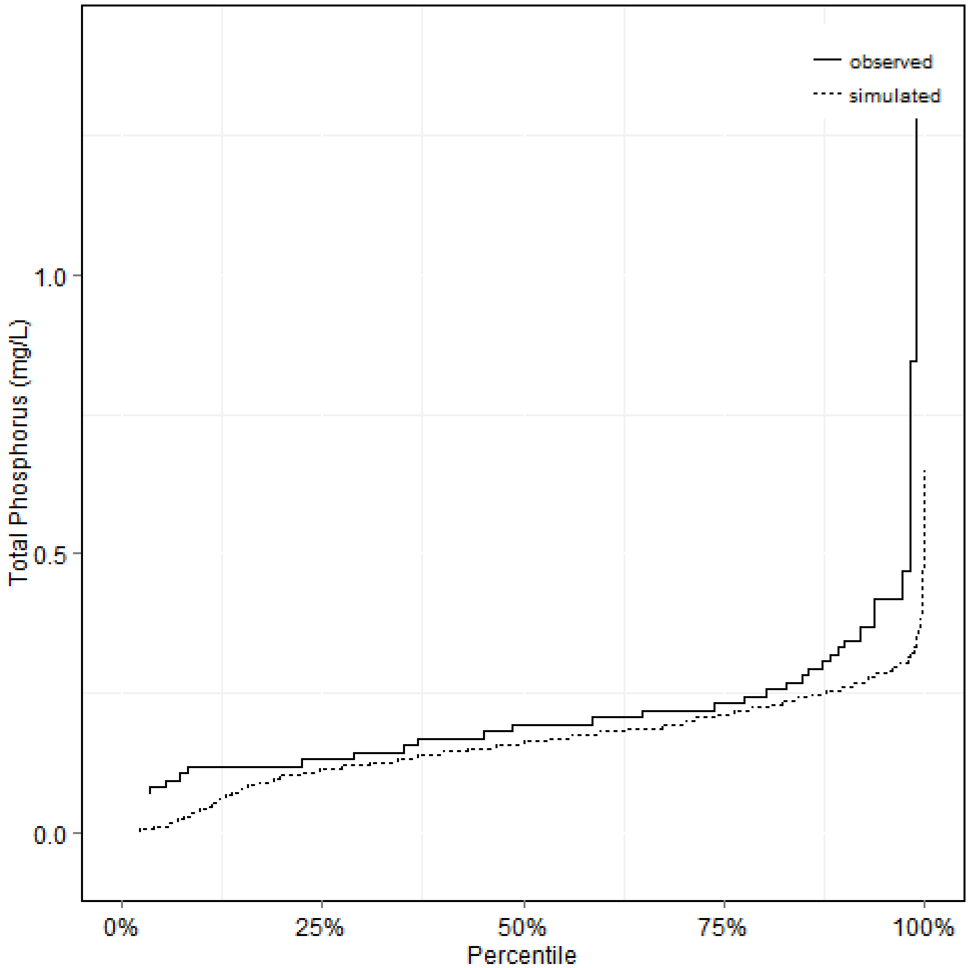



```
##specifiy the theme for the plot, this theme removes the default gray background and adds axes lines
p2<-p1 + theme(legend.title=element_blank())+ theme(legend.justification=c(1,1),
legend.position=c(1,1))+
  theme(panel.border=element_rect(colour="black", fill=NA))+
  theme(axis.text = element_text(colour = "black",
                                 size = 12))+
  theme(panel.background = element_rect(fill = NA,
                                        colour = "black"))+
  theme(legend.key=element_rect(fill=NA))
p2
```



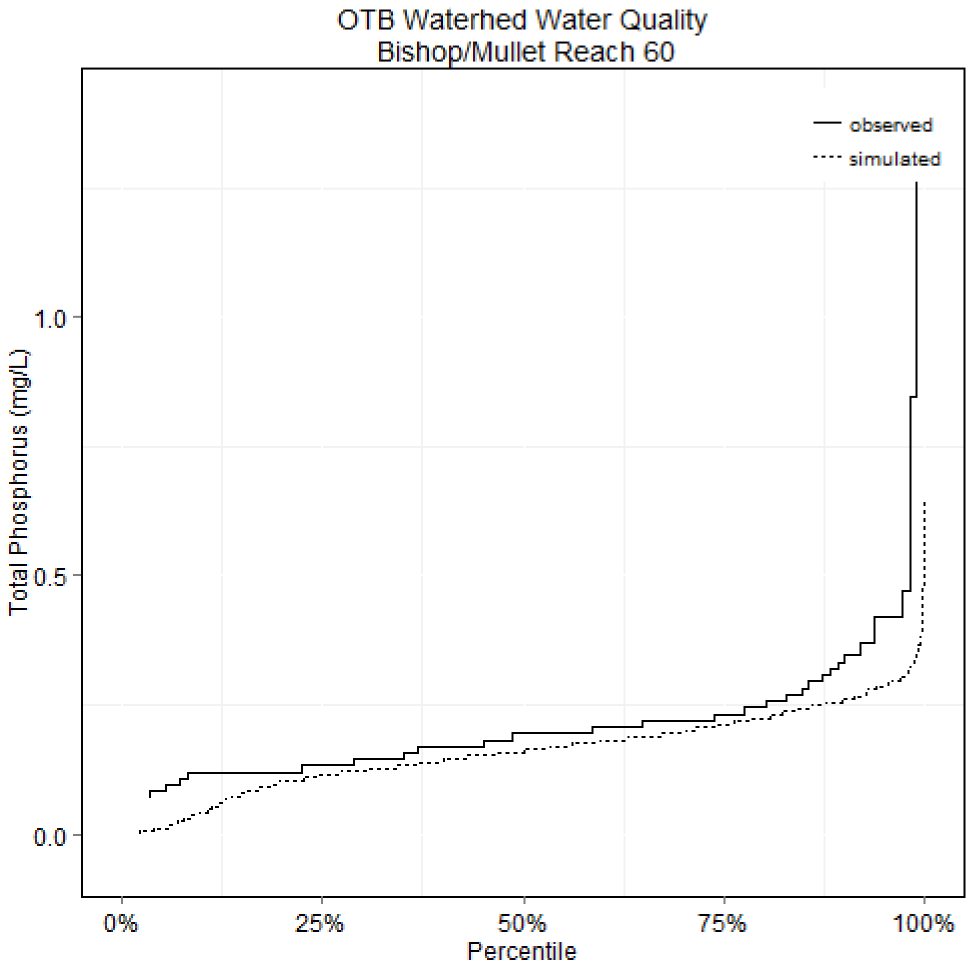

```
##add the x and y labels
p3<-p2 + scale_y_continuous(labels = percent,limits=c(0.00001,1)) +
    ylab("Percentile") + xlab(both$label[1])
p3
```

```
Warning: Removed 2 rows containing missing values (geom_path). Warning:
Removed 2 rows containing missing values (geom_path).
```




```
##add title
p4 <- p3 + ggtitle(paste("OTB Waterhed Water Quality\n", both$name[1],"Reach", Reach, sep=" ")  )
p4
```

```
Warning: Removed 2 rows containing missing values (geom_path). Warning:
Removed 2 rows containing missing values (geom_path).
```




```
##save and export plot
plot.filename <- paste0("C:/Users/john/Documents/old tampa bay/water optimal model/bf/",
PARAMETER,Reach,".pdf")
ggsave(p4, filename=plot.filename, height=8,width=8)
```

```
Warning: Removed 2 rows containing missing values (geom_path). Warning:
Removed 2 rows containing missing values (geom_path).
```

```
##########################Define new variables, the remainder of the code remains the same
##Variables to define
PARAMETER="TN"
Reach=60

##Subset by Reach and Parameter
cdf.data <- subset(all, all$REACH==Reach & all$parameter==PARAMETER)

##Create new data set for each set (observed and stimulated), bind the two sets together.  This
prevents R from deleting an entire row when
##there is a missing value
obs <- na.omit(data.frame(REACH=Reach, parameter=PARAMETER,dv=cdf.data$observed,
method="observed", label=cdf.data$X_LABEL_, name=cdf.data$name))
sim <- na.omit(data.frame(REACH=Reach, parameter=PARAMETER,dv=cdf.data$simulated,
method="simulated", label=cdf.data$X_LABEL_, name=cdf.data$name))
both <- rbind(obs,sim)
head(both)
```

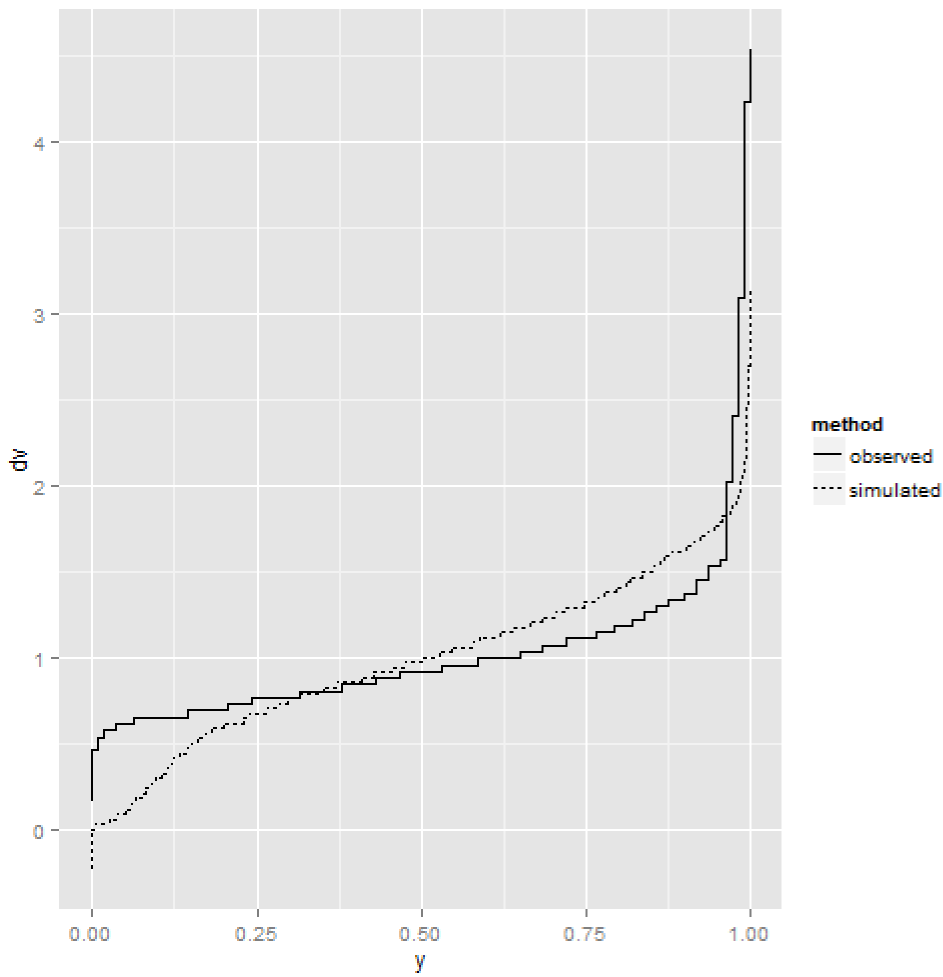
```
    REACH parameter    dv   method                 label          name
12     60        TN 1.000 observed Total Nitrogen (mg/L) Bishop/Mullet
```

```
24      60         TN 0.790 observed Total Nitrogen (mg/L) Bishop/Mullet
97      60         TN 0.800 observed Total Nitrogen (mg/L) Bishop/Mullet
115     60         TN 1.117 observed Total Nitrogen (mg/L) Bishop/Mullet
173     60         TN 1.515 observed Total Nitrogen (mg/L) Bishop/Mullet
243     60         TN 4.230 observed Total Nitrogen (mg/L) Bishop/Mullet
```

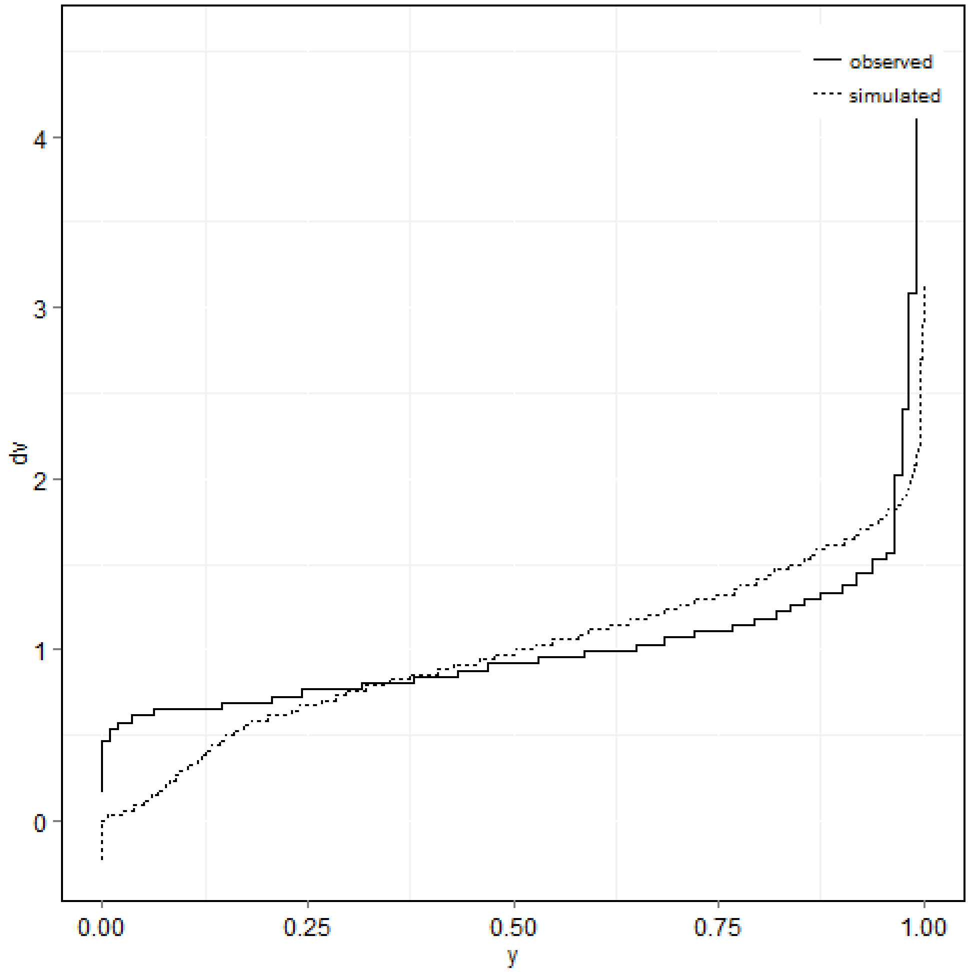
```
tail(both)
```

```
     REACH parameter     dv    method                         label
3621    60        TN 1.3198 simulated Modeled Total Nitrogen (mg/L)
3622    60        TN 0.5142 simulated Modeled Total Nitrogen (mg/L)
3623    60        TN 0.8237 simulated Modeled Total Nitrogen (mg/L)
3639    60        TN 1.5882 simulated Modeled Total Nitrogen (mg/L)
3646    60        TN 1.3110 simulated Modeled Total Nitrogen (mg/L)
3653    60        TN 1.1502 simulated Modeled Total Nitrogen (mg/L)
              name
3621 Bishop/Mullet
3622 Bishop/Mullet
3623 Bishop/Mullet
3639 Bishop/Mullet
3646 Bishop/Mullet
3653 Bishop/Mullet
```

```
##libraries
library(scales)##plots percentages
library(ggplot2)

##plot cdf values for both observed and stimulated using stat_ecdf
p1<-ggplot(both, aes(dv, linetype=method)) + stat_ecdf(n=100) + coord_flip()
p1
```

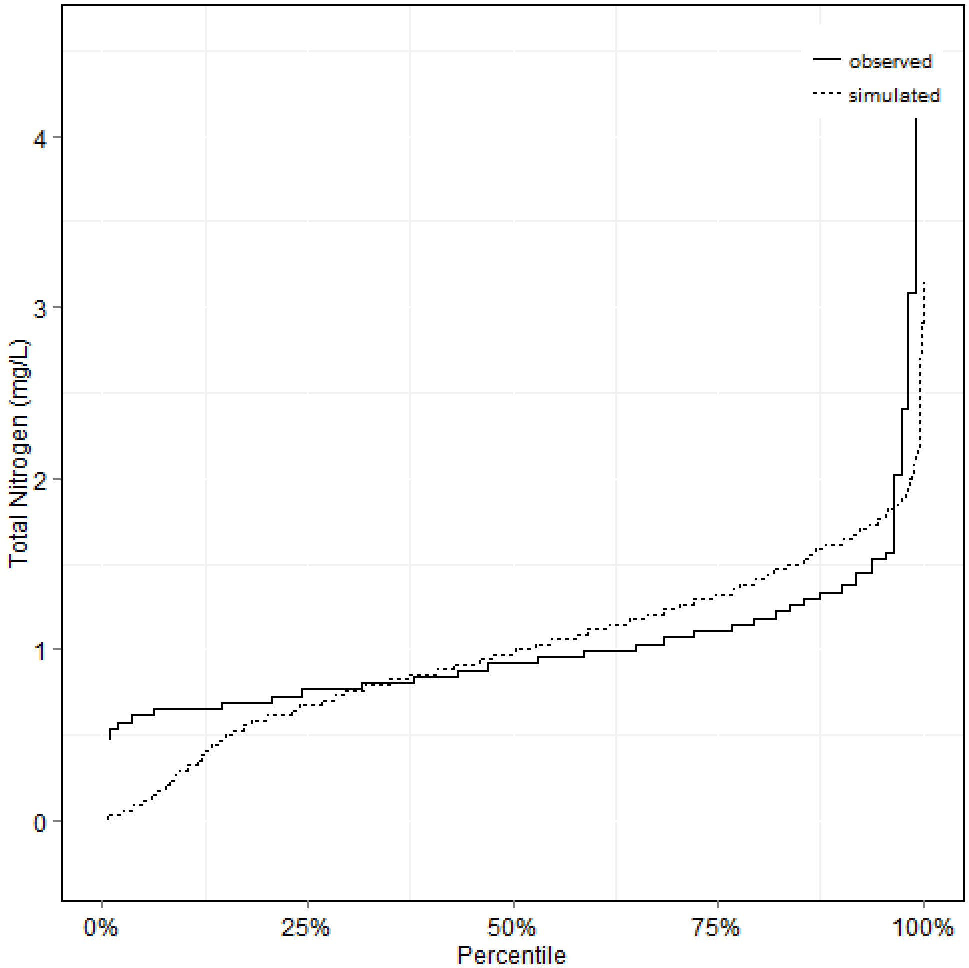



```
##specifiy the theme for the plot, this theme removes the default gray background and adds axes
lines
p2<-p1 + theme(legend.title=element_blank())+ theme(legend.justification=c(1,1),
legend.position=c(1,1))+
  theme(panel.border=element_rect(colour="black", fill=NA))+
  theme(axis.text = element_text(colour = "black",
                                 size = 12))+
  theme(panel.background = element_rect(fill = NA,
                                        colour = "black"))+
  theme(legend.key=element_rect(fill=NA))
p2
```



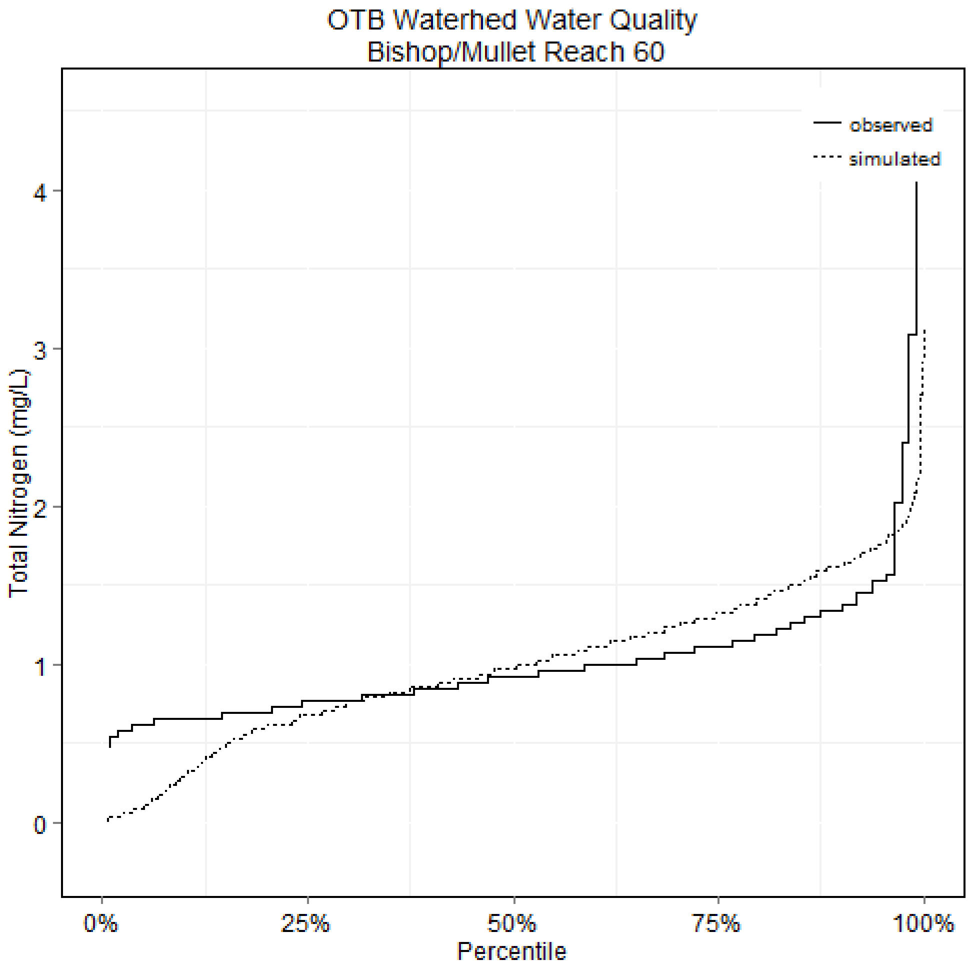

```
##add the x and y labels
p3<-p2 + scale_y_continuous(labels = percent,limits=c(0.00001,1)) +
    ylab("Percentile") + xlab(both$label[1])
p3
```

```
Warning: Removed 2 rows containing missing values (geom_path). Warning:
Removed 2 rows containing missing values (geom_path).
```




```
##add title
p4 <- p3 + ggtitle(paste("OTB Waterhed Water Quality\n", both$name[1],"Reach", Reach, sep=" ")  )
p4
```

```
Warning: Removed 2 rows containing missing values (geom_path). Warning:
Removed 2 rows containing missing values (geom_path).
```




```
##save and export plot
plot.filename <- paste0("C:/Users/john/Documents/old tampa bay/water optimal model/bf/",
PARAMETER,Reach,".pdf")
ggsave(p4, filename=plot.filename, height=8,width=8)
```

```
Warning: Removed 2 rows containing missing values (geom_path). Warning:
Removed 2 rows containing missing values (geom_path).
```