

# Running the South Florida Water Management Model SFWMM

Luis G. Cadavid, Ph.D., P.E.

Ken Tarboton, Ph.D., P.E.

August 28, 2002

# Running the SFWMM

- Make sure `/vol/hsm/bin/solaris` is in your path
- On the command line type
  - `cd` to input directory (where `altwmm` file is located)
  - 2: `sfwmm.scr < runinput_file_name`
  - 3: `sfwmm.scr` and answer to prompts

# Running the SFWMM

- Contents of the runinput file

(portions in bold face are definitions)

/vol/hsm/src/models/sfwmm/V4.4r10/wmm.exe ALTWMM\_v4.4\_aug02PA\_3 **path to SFWMM executable altwmm\_file**

/vol/hsm2/data/sfwmm/OP\_PLN/O\_AUG02\_PA\_3 **model run output directory**

stdout **standard output file name**

Y **delete existing files in output directory ?**

cadavid rsantee ktarbot jobey **users to mail to about model run (Unix user's name)**

econ\_post.cf

ea\_econ.cf

wmmwbud.cf

lok2est\_bud.def

hpdiff\_nsm-wmm.cf

pondiff\_nsm-wmm.cf

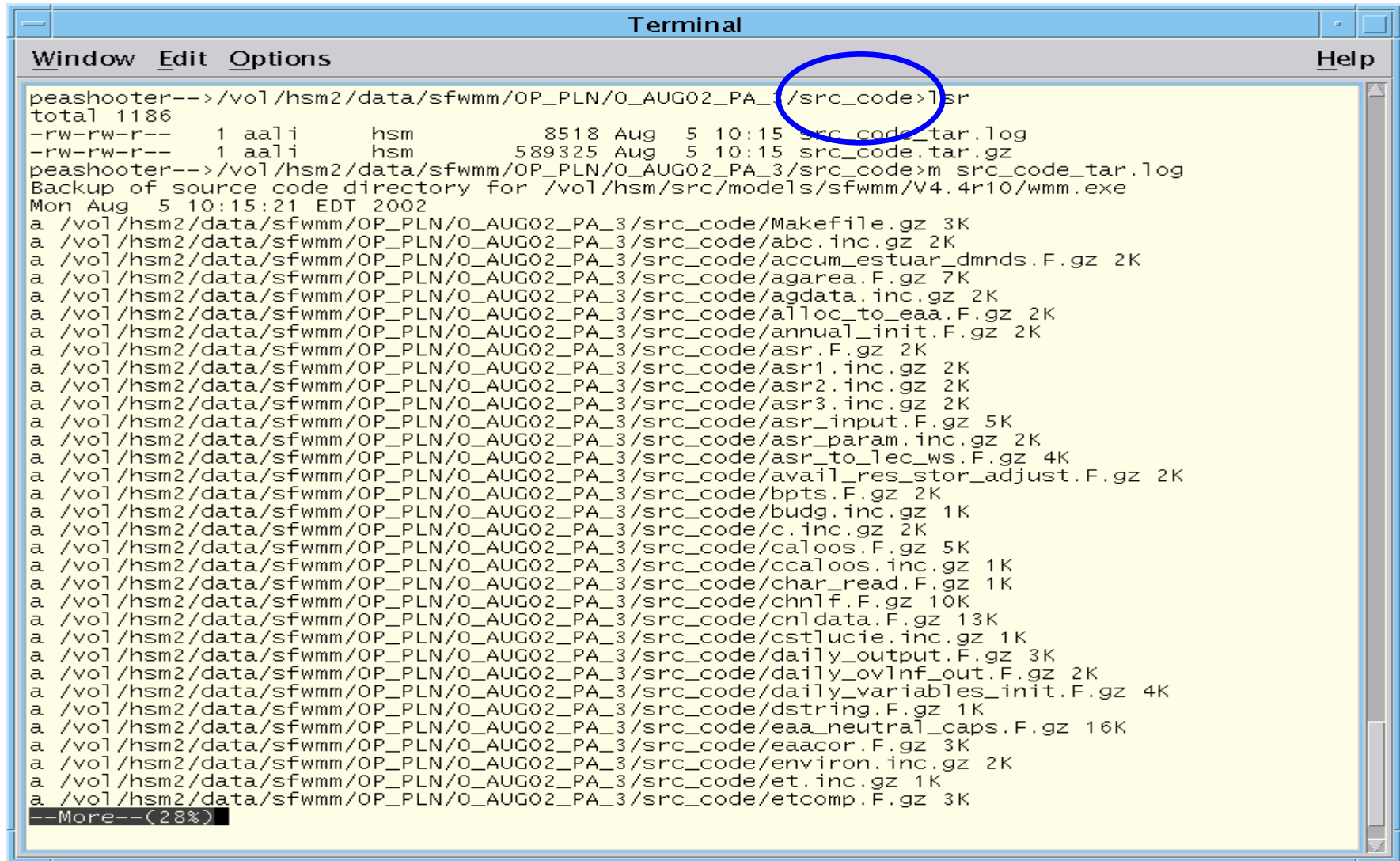
hpimp\_wrt\_nosta34.cf

**Post processing definition and configuration files**

# Running the SFWMM

- What does `sfwmm.scr` do ?
  - It is a c-shell script
  - Set up environment variables
  - Get input from the user
  - Check for existence of files and directories (e.g. SFWMM executable and ALTWMM file)
  - Archive source code used for the simulation under the simulation output directory (**src\_code**)

# Source Code Archive



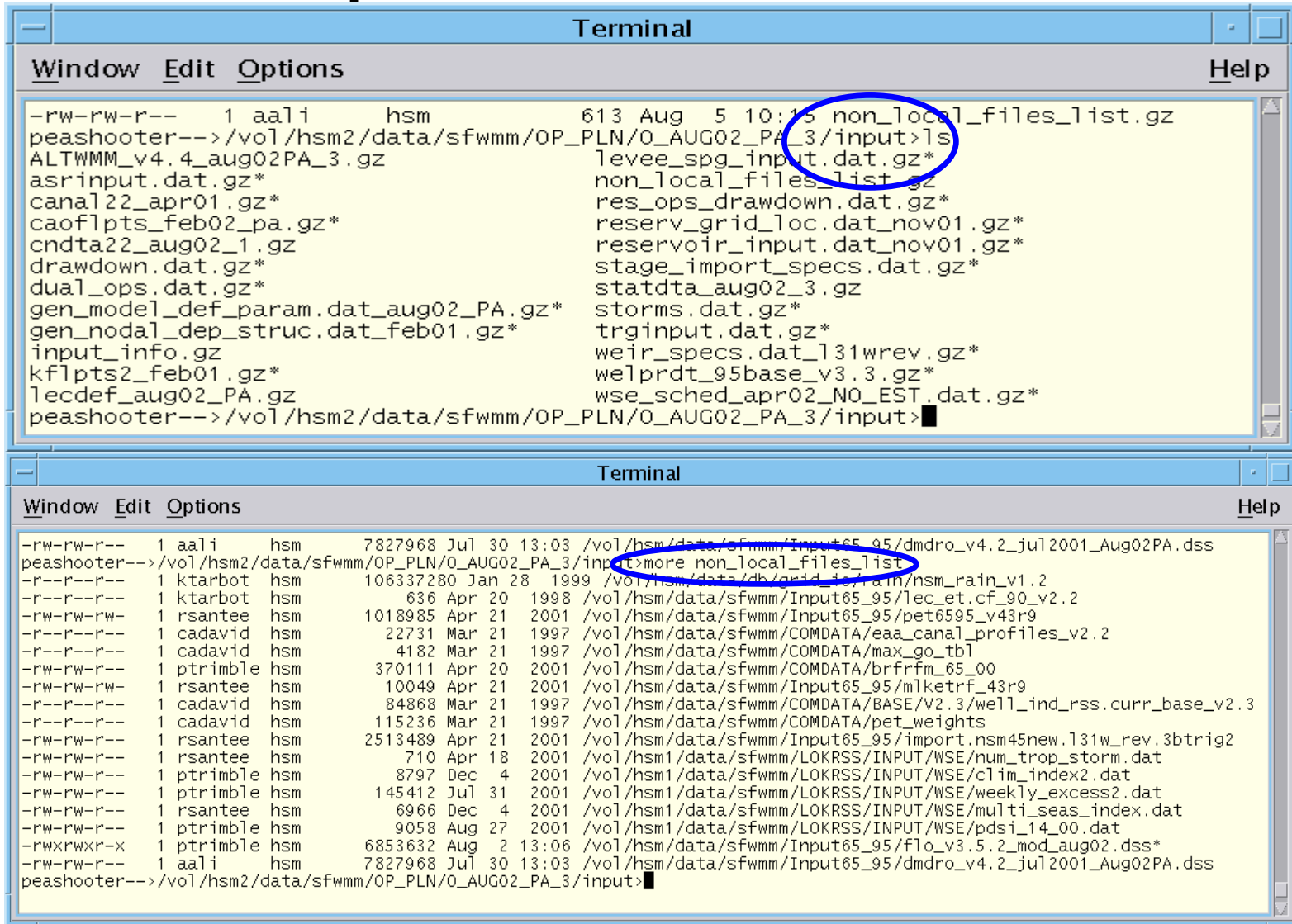
The image shows a terminal window titled "Terminal" with a menu bar containing "Window", "Edit", "Options", and "Help". The terminal output shows a user named "peashooter" navigating to a directory and listing files. A blue circle highlights the command "ls" in the first line. The listing shows various files and directories, including "src\_code.tar.log" and "src\_code.tar.gz".

```
peashooter-->/vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code>ls
total 1186
-rw-rw-r-- 1 aali hsm 8518 Aug 5 10:15 src_code.tar.log
-rw-rw-r-- 1 aali hsm 589325 Aug 5 10:15 src_code.tar.gz
peashooter-->/vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code>m src_code.tar.log
Backup of source code directory for /vol/hsm/src/models/sfwmm/V4.4r10/wmm.exe
Mon Aug 5 10:15:21 EDT 2002
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/Makefile.gz 3K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/abc.inc.gz 2K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/accum_estuar_dmnds.F.gz 2K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/agarea.F.gz 7K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/agdata.inc.gz 2K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/alloc_to_eaa.F.gz 2K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/annual_init.F.gz 2K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/asr.F.gz 2K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/asr1.inc.gz 2K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/asr2.inc.gz 2K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/asr3.inc.gz 2K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/asr_input.F.gz 5K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/asr_param.inc.gz 2K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/asr_to lec_ws.F.gz 4K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/avail_res_stor_adjust.F.gz 2K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/bpts.F.gz 2K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/budg.inc.gz 1K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/c.inc.gz 2K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/caloos.F.gz 5K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/ccaloos.inc.gz 1K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/char_read.F.gz 1K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/chn1f.F.gz 10K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/chn1data.F.gz 13K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/cstlucie.inc.gz 1K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/daily_output.F.gz 3K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/daily_ovlnf_out.F.gz 2K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/daily_variables_init.F.gz 4K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/dstring.F.gz 1K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/ea_neutral_caps.F.gz 16K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/eaacor.F.gz 3K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/environ.inc.gz 2K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/et.inc.gz 1K
a /vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/src_code/etcomp.F.gz 3K
--More--(28%)
```

# Running the SFWMM

- What does `sfwmm.scr` do ?
  - Archive input directory for the simulation under the simulation output directory (**input**)
    - Copy of local files
    - Long list of non-local input files
  - Remove existing files from output directory, including the **rundone** file
  - Mail users on run starting:
    - Post processing definition/configuration file existence
  - Run SFWMM and time the execution

# Input Files Archive



```
Terminal
Window Edit Options Help
-rw-rw-r-- 1 aali hsm 613 Aug 5 10:15 non_local_files_list.gz
peashooter-->/vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/input>ls
ALTWMM_v4.4_aug02PA_3.gz levee_spg_input.dat.gz*
asrinput.dat.gz* non_local_files_list.gz
canal22_apr01.gz* res_ops_drawdown.dat.gz*
caoflpts_feb02_pa.gz* reserv_grid_loc.dat_nov01.gz*
cndta22_aug02_1.gz reservoir_input.dat_nov01.gz*
drawdown.dat.gz* stage_import_specs.dat.gz*
dual_ops.dat.gz* statdta_aug02_3.gz
gen_model_def_param.dat_aug02_PA.gz* storms.dat.gz*
gen_nodal_dep_struct.dat_feb01.gz* trginput.dat.gz*
input_info.gz weir_specs.dat_131wrev.gz*
kflpts2_feb01.gz* welprdt_95base_v3.3.gz*
lecdef_aug02_PA.gz wse_sched_apr02_NO_EST.dat.gz*
peashooter-->/vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/input>

Terminal
Window Edit Options Help
-rw-rw-r-- 1 aali hsm 7827968 Jul 30 13:03 /vol/hsm/data/sfwmm/Input65_95/dmdro_v4.2_jul2001_Aug02PA.dss
peashooter-->/vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/input>more non_local_files_list
-r--r--r-- 1 ktarbot hsm 106337280 Jan 28 1999 /vol/hsm/data/db/psid_307/am/hsm_rain_v1.2
-r--r--r-- 1 ktarbot hsm 636 Apr 20 1998 /vol/hsm/data/sfwmm/Input65_95/lec_et.cf_90_v2.2
-rw-rw-rw- 1 rsantee hsm 1018985 Apr 21 2001 /vol/hsm/data/sfwmm/Input65_95/pet6595_v43r9
-r--r--r-- 1 cadavid hsm 22731 Mar 21 1997 /vol/hsm/data/sfwmm/COMDATA/eaa_canal_profiles_v2.2
-r--r--r-- 1 cadavid hsm 4182 Mar 21 1997 /vol/hsm/data/sfwmm/COMDATA/max_go_tbt1
-rw-rw-rw- 1 ptrimble hsm 370111 Apr 20 2001 /vol/hsm/data/sfwmm/COMDATA/brfrfm_65_00
-rw-rw-rw- 1 rsantee hsm 10049 Apr 21 2001 /vol/hsm/data/sfwmm/Input65_95/m1ketr43r9
-r--r--r-- 1 cadavid hsm 84868 Mar 21 1997 /vol/hsm/data/sfwmm/COMDATA/BASE/V2.3/well_ind_rss.curr_base_v2.3
-r--r--r-- 1 cadavid hsm 115236 Mar 21 1997 /vol/hsm/data/sfwmm/COMDATA/pet_weights
-rw-rw-rw- 1 rsantee hsm 2513489 Apr 21 2001 /vol/hsm/data/sfwmm/Input65_95/import.nsm45new.131w_rev.3btrig2
-rw-rw-rw- 1 rsantee hsm 710 Apr 18 2001 /vol/hsm1/data/sfwmm/LOKRSS/INPUT/WSE/num_trop_storm.dat
-rw-rw-rw- 1 ptrimble hsm 8797 Dec 4 2001 /vol/hsm1/data/sfwmm/LOKRSS/INPUT/WSE/clim_index2.dat
-rw-rw-rw- 1 ptrimble hsm 145412 Jul 31 2001 /vol/hsm1/data/sfwmm/LOKRSS/INPUT/WSE/weekly_excess2.dat
-rw-rw-rw- 1 rsantee hsm 6966 Dec 4 2001 /vol/hsm1/data/sfwmm/LOKRSS/INPUT/WSE/multi_seas_index.dat
-rw-rw-rw- 1 ptrimble hsm 9058 Aug 27 2001 /vol/hsm1/data/sfwmm/LOKRSS/INPUT/WSE/pdsi_14_00.dat
-rwxrwxr-x 1 ptrimble hsm 6853632 Aug 2 13:06 /vol/hsm/data/sfwmm/Input65_95/flo_v3.5.2_mod_aug02.dss*
-rw-rw-rw- 1 aali hsm 7827968 Jul 30 13:03 /vol/hsm/data/sfwmm/Input65_95/dmdro_v4.2_jul2001_Aug02PA.dss
peashooter-->/vol/hsm2/data/sfwmm/OP_PLN/O_AUG02_PA_3/input>
```

# Running the SFWMM

- What does sfwmm.scr do ?
  - Mail users on run completion/status
  - Creates file with simple listing/definition of SFWMM input/output files, in the simulation output directory  
(**sfwmmV4.4\_file\_definitions**)



# Running the SFWMM

- What does sfwmm.scr do ?
  - Post-processing and maps production
    - LEC Urban Areas
    - EAA
    - LOSA Report (Calendar and Water Year)
    - Water Budgets (Annual, Water Year, Seasonal and Monthly)
    - Overland Flow
    - LEC trigger maps
    - Hydroperiod and Hydroperiod improvement maps
    - Ponding and Ponding difference maps

# Running the SFWMM

- What is the altwmm file ?
  - Input data locator file, giving the full pathname for all the input files required to run the SFWMM
  - ALTWMM changes with the version fo the model

# Running the SFWMM

- What is the altwmm file ?(V4.4R10)

(portions in bold face are definitions; see SFWMM documentation)

SFWMM v4.4r10 - AUG02 PA LOK 14.53 **run title**

36 **number of units to open**

112 gen\_model\_def\_param.dat\_aug02\_PA **unit number and file name**

2 lecdef\_aug02\_PA

7 /vol/hsm/data/db/grid\_io/rain/nsm\_rain\_v1.2

20 /vol/hsm/data/sfwmm/Input65\_95/lec\_et.cf\_90\_v2.2

94 trginput.dat

11 cndta22\_aug02\_1 **physical parameters for each canal**

23 canal22\_apr01 **cell location of each canal**

22 kflpts2\_feb01 **flow point definition file**

60 statdta\_aug02\_3 **model static data defined for each cell**

18 welprdt\_95base\_v3.3

28 /vol/hsm/data/sfwmm/Input65\_95/pet6595\_v43r9

# Running the SFWMM

- What is the altwmm file ?(V4.4R10)

13 /vol/hsm/data/sfwmm/COMDATA/eea\_canal\_profiles\_v2.2

15 /vol/hsm/data/sfwmm/COMDATA/max\_go\_tbl

59 /vol/hsm/data/sfwmm/COMDATA/brfrfm\_65\_00 **WCA-3A rainfall and ET for ENP rainfall formula**

21 caoflpts\_feb02\_pa

36 /vol/hsm/data/sfwmm/Input65\_95/mlketrf\_43r9

37 /vol/hsm/data/sfwmm/COMDATA/BASE/V2.3/well\_ind\_rss.curr\_base\_v2.3

17 asrinput.dat

12 /vol/hsm/data/sfwmm/COMDATA/pet\_weights

102 wse\_sched\_apr02\_NO\_EST.dat

104 levee\_spg\_input.dat **specification file for computation of localized levee seepage**

103 reservoir\_input.dat\_nov01

101 gen\_nodal\_dep\_struc.dat\_feb01

105 stage\_import\_specs.dat

133 weir\_specs.dat\_l31wrev **properties of fixed-crest passive weirs**

# Running the SFWMM

- What is the altwmm file ?(V4.4R10)

83 /vol/hsm/data/sfwmm/Input65\_95/import.nsm45new.l31w\_rev.3btrig2

108 /vol/hsm1/data/sfwmm/LOKRSS/INPUT/WSE/num\_trop\_storm.dat

109 /vol/hsm1/data/sfwmm/LOKRSS/INPUT/WSE/clim\_index2.dat

134 /vol/hsm1/data/sfwmm/LOKRSS/INPUT/WSE/weekly\_excess2.dat

135 /vol/hsm1/data/sfwmm/LOKRSS/INPUT/WSE/multi\_seas\_index.dat

148 /vol/hsm1/data/sfwmm/LOKRSS/INPUT/WSE/pdsi\_14\_00.dat

136 drawdown.dat

147 res\_ops\_drawdown.dat

137 storms.dat

145 reserv\_grid\_loc.dat\_nov01

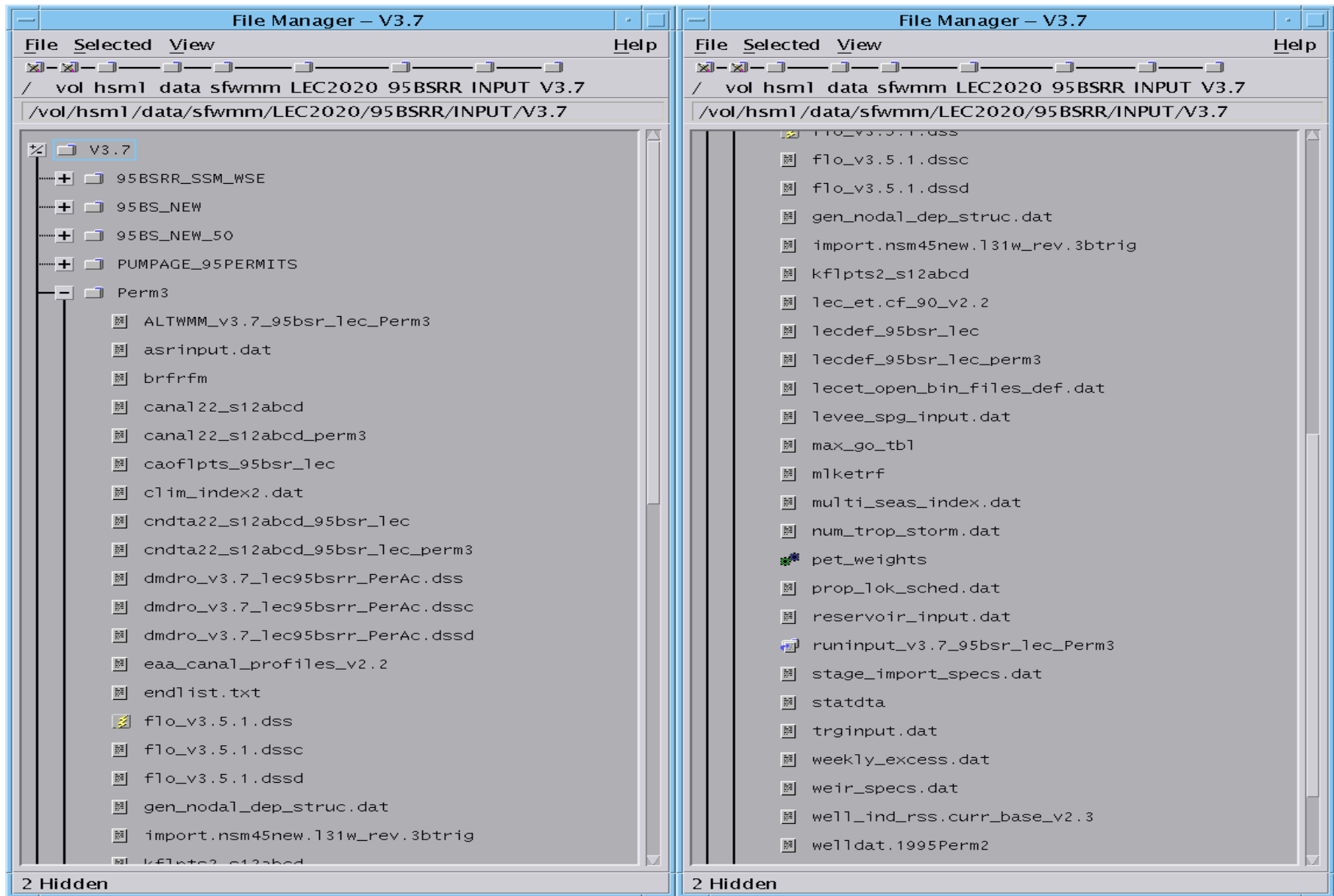
157 dual\_ops.dat

/vol/hsm/data/sfwmm/Input65\_95/flo\_v3.5.2\_mod\_aug02.dss **known (historical) flows**

/vol/hsm/data/sfwmm/Input65\_95/dmdro\_v4.2\_jul2001\_Aug02PA.dss **daily runoff and demand time series**

# Running the SFWMM

- Input file structure
  - Keep projects in a group
  - Have a different input directory for each simulation. Facilitates documentation and quality assurance
  - Duplicate input directory from similar run and modify input files
  - Some input files are documented.
  - Additional documentation is on the works

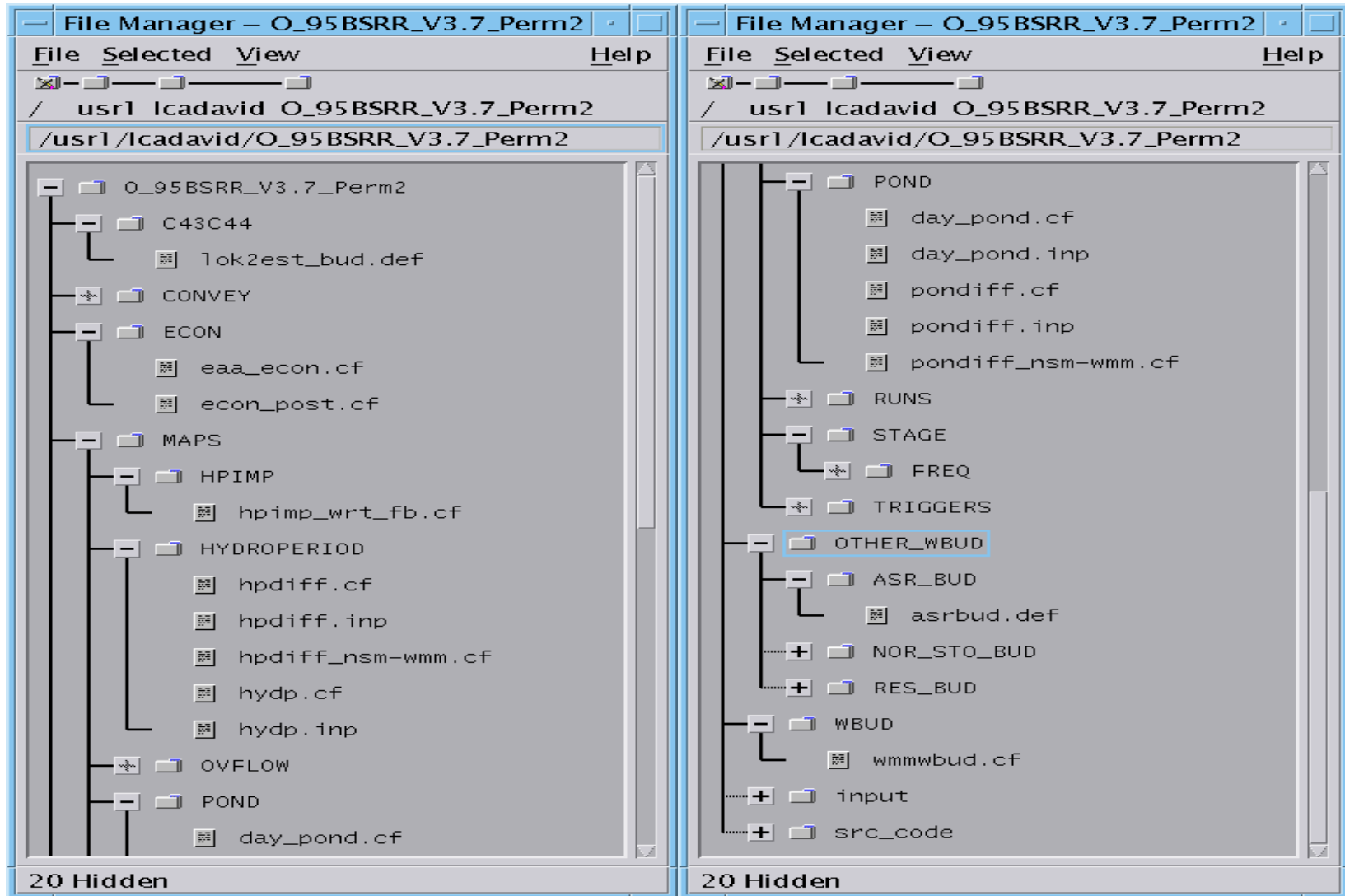


# Running the SFWMM

- Output directory structure
  - Use `sfwmm_mkdirs.scr` to create simulation output directory structure
  - **Modify titles in \*.cf, \*.def and \*.inp files**



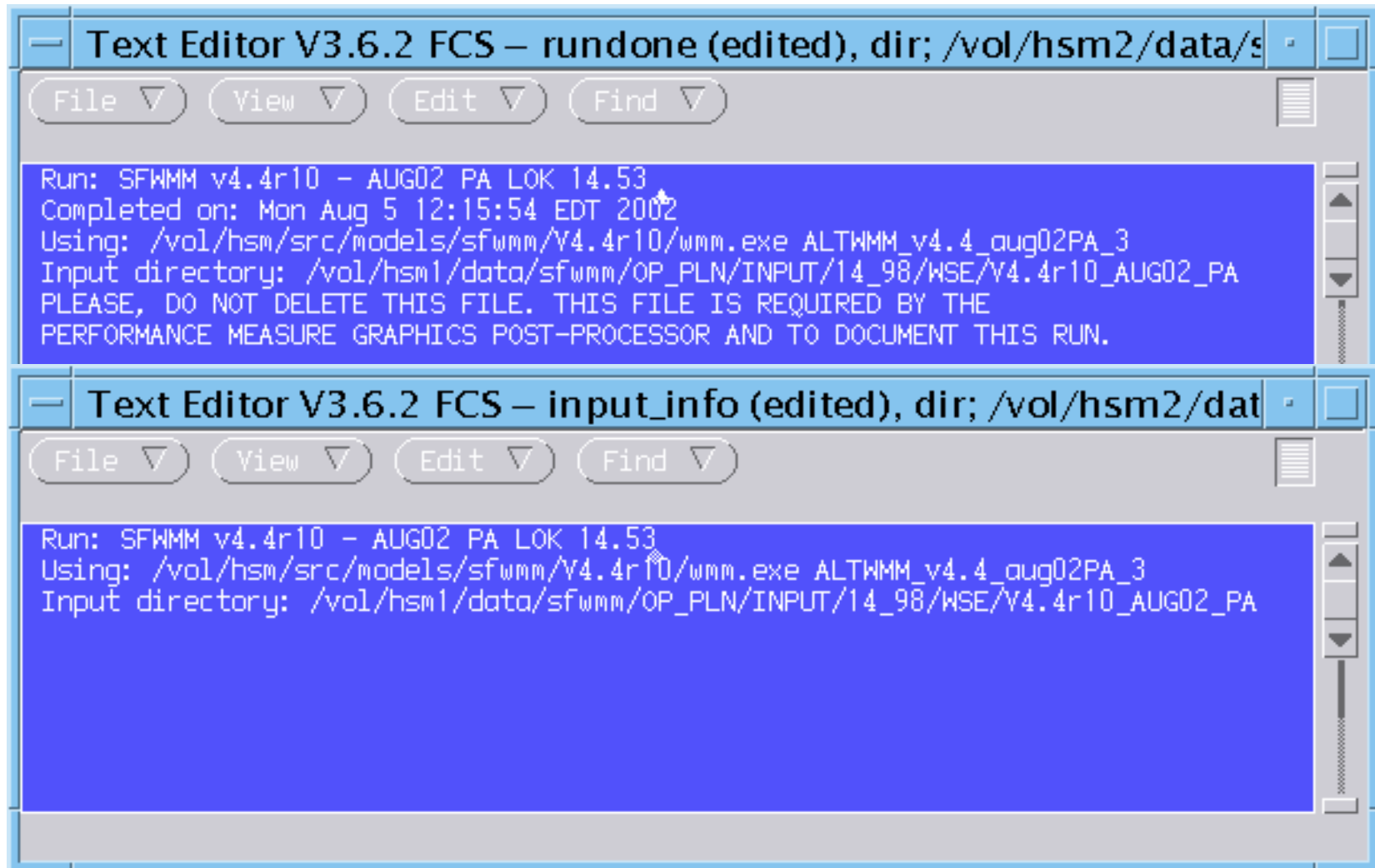
# Output Directory Structure



# Running the SFWMM

- When simulation completes
  - Useful information in the **rundone** file
  - Existence of this file signals PM graphics scripts can start execution.
  - It is created after Water Budget, LEC and EAA post-processor have been completed successfully
  - Similar information is stored before run starts in **input/input\_info**

# rundone and input\_info files



# Checking the Run

- Look at the residuals in the monthly water budget result file
  - cd to simulation\_output\_directory/WMMBUD and edit or view monbud
  - Use the chk\_bud.scr utility
- Use other utilities (grid\_io, dsstool) to explore results

# Checking Residuals

Terminal

Window Edit Options Help

1965 WATER BUDGET SUMMARY FOR LAKE\_OKEECHOBEE SUBBASIN AREA (square miles) = 728.  
RESIDUAL (All values in thousand acre-feet)

YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC	YEAR
1965	-1.5	0.0	0.0	0.7	1.6	1.3	0.1	-0.1	-0.3	-0.4	0.5	0.1	2.1
1966	-0.1	-0.1	0.6	0.4	1.1	0.5	0.2	-0.1	-0.2	-0.1	0.8	0.6	3.6
1967	0.2	0.5	1.2	2.2	2.3	1.6	0.9	0.0	-0.3	-0.3	1.2	0.1	9.6
1968	0.7	0.6	0.6	0.9	0.9	1.1	0.1	0.5	0.3	-0.2	-0.2	0.7	6.0
1969	0.1	0.3	-0.1	0.6	0.4	0.6	0.1	-0.2	-0.2	-0.4	-0.1	-0.2	1.1
1970	-0.1	0.0	-0.2	0.4	1.4	0.9	0.1	-0.2	-0.2	-0.4	0.4	0.7	2.8
1971	0.8	0.6	1.5	2.1	2.1	1.8	1.0	-0.1	-0.3	-0.1	0.0	0.6	10.0
1972	0.7	0.2	0.4	0.8	0.4	1.4	0.1	0.0	0.8	0.3	0.1	0.0	5.1
1973	-0.1	0.0	0.2	0.3	0.3	0.2	0.1	0.0	-0.1	-0.2	1.2	0.1	2.1
1974	0.7	0.4	0.5	0.6	0.3	1.6	0.8	0.0	-0.2	0.2	0.5	-0.2	5.1
1975	0.7	0.2	1.1	1.4	1.1	0.6	1.0	0.1	-0.1	-0.4	0.7	0.7	7.2
1976	0.7	0.7	0.7	1.3	0.4	0.7	1.1	0.4	-0.3	-0.2	-0.2	0.5	5.8
1977	-0.1	0.6	0.5	1.5	0.6	0.7	0.0	0.5	-0.2	0.1	-0.2	-0.2	3.8
1978	-0.1	0.0	0.1	0.7	0.0	0.4	0.1	-0.2	-0.3	-0.2	0.0	0.1	0.6
1979	0.0	0.3	0.6	0.9	0.4	1.2	0.3	0.0	0.6	-0.3	0.0	0.0	4.2
1980	0.5	0.0	0.6	-0.2	0.8	1.1	0.3	0.1	-0.2	0.0	0.4	0.2	3.5
1981	0.7	0.1	1.0	1.5	0.6	0.2	0.0	1.1	0.2	0.4	0.1	0.4	6.2
1982	0.3	0.1	0.1	-0.3	1.0	0.9	0.1	-0.2	-0.1	-0.4	0.7	0.6	2.9
1983	0.4	-0.2	-0.5	0.1	2.0	0.6	0.3	-0.1	-0.2	-0.2	-0.1	0.0	2.2
1984	0.4	0.0	-0.4	-0.2	0.9	0.4	0.1	-0.2	0.0	0.7	0.7	0.6	3.0
1985	0.8	0.7	0.9	0.3	1.1	1.6	0.8	-0.2	-0.2	0.4	0.4	-0.1	6.8
1986	0.1	0.7	0.3	1.4	1.4	1.9	0.3	-0.2	0.1	0.5	0.5	0.1	7.0
1987	0.0	0.0	0.0	0.7	1.3	1.1	0.1	1.1	0.2	-0.6	0.1	-0.1	4.1
1988	-0.1	-0.1	-0.1	1.3	1.1	0.5	0.7	0.1	0.2	1.2	0.4	0.6	5.7
1989	0.6	0.8	0.5	0.4	1.5	0.6	1.2	0.1	-0.2	-0.3	0.3	0.4	6.0
1990	0.4	0.2	0.4	0.5	0.1	0.2	1.3	0.0	0.4	-0.2	0.3	0.7	4.2
1991	0.4	0.4	0.2	-0.2	0.8	0.4	0.2	-0.2	-0.2	-0.1	0.5	0.2	2.5
1992	0.5	-0.2	0.1	0.1	1.5	1.2	-0.3	0.2	-0.1	0.5	0.2	0.7	4.5
1993	0.2	0.0	-0.1	0.8	1.9	0.7	0.3	0.9	1.1	0.0	0.1	0.5	6.5
1994	-0.1	-0.1	-0.1	0.8	1.2	0.8	0.3	-0.2	-0.2	-0.4	-0.2	-0.1	1.9
1995	-0.2	-0.2	-0.3	0.2	0.6	0.3	0.2	-0.1	-0.2	-0.2	0.4	0.6	1.1

Continue (y, Y or Return): █

# Running the SFWMM

- Another way to run model
  - setenv SFWMMDAT “output\_directory\_path”
  - On the command line, type:
    - sfwmm\_executable\_path altwmm\_file\_name
  - This will only run the model: no pre or post processing, no mail to the user

/vol/hsm/bin/solaris/chk\_bud.scr

```
#!/bin/csh -f
#
# Script to glance at SFWMM monthly budget components on the screen
# Written by Luis G. Cadavid, HSM Department, April 2001
# SCCSID = "@(#)chk_bud.scr 1.3 07/05/01 SFWMD HSM Department, Water Supply
Division"
# Find number of areas
#
set in_yr = 1965
set nyears = 31
set kw = "RESIDUAL"
clear
echo " "
echo "Program to glance at SFWMM monthly budget results on the screen"
echo "monbud file needs to be in the local directory"
echo " "
if !(-e monbud)then
    echo "monbud file was not found. Please try again \!"
    echo " "
    exit
endif
echo -n " Enter keyword to search for (RESIDUAL is the default): "
set wd = $<
if !( $wd == '' ) then
    set kw = $wd
endif
set lin1 = $kw' (All values in thousand acre-feet)'
set lin2 = '          JAN      FEB      MAR      APR      MAY      JUN      JUL
AUG      SEP      OCT      NOV      DEC      YEAR'
set nareas = `grep "SUMMARY" annbud | wc -l`
#
# Find area names
#
grep "SUMMARY" monbud > /tmp/areas.$$
grep $kw monbud > /tmp/resid.$$
@ i=1
while ( $i <= $nareas )
    set title = `gawk '{if (NR == (i-1)*ny+1) {print $0;exit}}' i=$i
ny=$nyears /tmp/areas.$$`
    clear
    echo " "
    echo " "
    echo $title
    echo $lin1 | gawk '{printf("%s \n", $0)}'
    echo " "
    echo $lin2 | gawk '{printf("YEAR      %s", $1); for(j=2;j<=13;j++)
printf("      %s",$(j)); printf("      %s\n", $14)}'
    gawk 'BEGIN {yr=0} { if(NR >= (i-1)*ny+1 && NR <= i*ny ) {printf("%4d
%8.1f", inyr+yr,$2);for(j=3;j<=13;j++) printf("%8.1f",$(j));
printf("%9.1f\n", $14);yr++;}}' i=$i inyr=$in_yr ny=$nyears /tmp/resid.$$
    echo " "
    echo -n "Continue (y, Y or Return): "
    set ans = $<
```

```
    if !($ans == 'y' || $ans == 'Y' || $ans == '')then
        /bin/rm /tmp/*.$$
        exit
    endif
    @ i++
end
/bin/rm /tmp/*.$$
exit
```



/vol/hsm/scripts/solaris/sfwmm\_mkdirs.scr

```
#!/bin/csh -f
# script to create the SFWMM output dir structure and control files.
# set file/directory permissions to rwxrwxr-x (chmod 775 or umask 002)
umask 002
echo "Enter the path to an existing run with similar output structure and
control files"
set from_path = $<
if (! -e $from_path) then
    echo "$from_path is not found"
    exit -1
endif

echo "Enter the path to the new SFWMM run output dir"
set to_path = $<
if (-e $to_path) then
    echo "This directory already exists: what do you want to do?"
    echo "Stop, re-select a name, and then re-run the script, type: [s] "
    echo "Keep the dir. name, (make sure it has the same subdir structure),
but completely overwrite files, type: [c]"
    echo "Keep the dir. name, (make sure it has the same subdir structure),
just partially overwrite files, type: [p]"
    set reply = $<
    switch ($reply)
        case [sS] :
            echo "STOP: choose different dir location"
            exit -1
        case [cC] :
            cd $to_path
            umask 002
            /bin/cp $from_path/C43C44/lok2est_bud.def C43C44/
            /bin/cp $from_path/ECON/econ_post.cf ECON/
            /bin/cp $from_path/ECON/ea_econ.cf ECON/
            /bin/cp $from_path/MAPS/HPIMP/hpimp_wrt_fb.cf MAPS/HPIMP/
            /bin/cp $from_path/MAPS/HPIMP/hpimp.inp MAPS/HPIMP/
            /bin/cp $from_path/MAPS/HYDROPERIOD/hpdiff.inp
MAPS/HYDROPERIOD/
            /bin/cp $from_path/MAPS/HYDROPERIOD/hydp.inp MAPS/HYDROPERIOD/
            /bin/cp $from_path/MAPS/HYDROPERIOD/hpdiff_nsm-wmm.cf
MAPS/HYDROPERIOD/
            /bin/cp $from_path/MAPS/POND/day_pond.inp MAPS/POND/
            /bin/cp $from_path/MAPS/POND/pondiff.inp MAPS/POND/
            /bin/cp $from_path/MAPS/POND/pondiff_nsm-wmm.cf MAPS/POND/
            /bin/cp $from_path/OTHER_WBUD/ASR_BUD/asrbud.def
OTHER_WBUD/ASR_BUD/
            /bin/cp $from_path/OTHER_WBUD/NOR_STO_BUD/noresbud.def
OTHER_WBUD/NOR_STO_BUD/
            /bin/cp $from_path/WBUD/wmmwbud.cf WBUD/
            breaksw
        case [pP] :
            cd $to_path
            umask 002
            cp -i $from_path/C43C44/lok2est_bud.def C43C44/
            cp -i $from_path/ECON/econ_post.cf ECON/
            cp -i $from_path/ECON/ea_econ.cf ECON/
```

```

        cp -i $from_path/MAPS/HPIMP/hpimp_wrt_fb.cf MAPS/HPIMP/
        cp -i $from_path/MAPS/HPIMP/hpimp.inp MAPS/HPIMP/
        cp -i $from_path/MAPS/HYDROPERIOD/hpdiff.inp MAPS/HYDROPERIOD/
        cp -i $from_path/MAPS/HYDROPERIOD/hydp.inp MAPS/HYDROPERIOD/
        cp -i $from_path/MAPS/HYDROPERIOD/hpdiff_nsm-wmm.cf
MAPS/HYDROPERIOD/
        cp -i $from_path/MAPS/POND/day_pond.inp MAPS/POND/
        cp -i $from_path/MAPS/POND/pondiff.inp MAPS/POND/
        cp -i $from_path/MAPS/POND/pondiff_nsm-wmm.cf MAPS/POND/
        cp -i $from_path/OTHER_WBUD/ASR_BUD/asrbud.def
OTHER_WBUD/ASR_BUD/
        cp -i $from_path/OTHER_WBUD/NOR_STO_BUD/noresbud.def
OTHER_WBUD/NOR_STO_BUD/
        cp -i $from_path/WBUD/wmmwbud.cf WBUD/
        breaksw
        default :
            echo "Error- no such option. Only select s, c, or p"
            exit -1
        breaksw
    endsw
else
    umask 002
    mkdir $to_path
    cd $to_path
    mkdir C43C44 CONVEY ECON MAPS OTHER_WBUD WBUD
    cd MAPS
    mkdir HPIMP HYDROPERIOD OVFLOW POND RUNS STAGE TRIGGERS
    mkdir STAGE/FREQ
    cd ../OTHER_WBUD
    mkdir ASR_BUD NOR_STO_BUD RES_BUD
    cd ..
    cp $from_path/C43C44/lok2est_bud.def C43C44/
    cp $from_path/ECON/econ_post.cf ECON/
    cp $from_path/ECON/ea_econ.cf ECON/
    cp $from_path/MAPS/HPIMP/hpimp_wrt_fb.cf MAPS/HPIMP/
    cp $from_path/MAPS/HPIMP/hpimp.inp MAPS/HPIMP/
    cp $from_path/MAPS/HYDROPERIOD/hpdiff.inp MAPS/HYDROPERIOD/
    cp $from_path/MAPS/HYDROPERIOD/hydp.inp MAPS/HYDROPERIOD/
    cp $from_path/MAPS/HYDROPERIOD/hpdiff_nsm-wmm.cf MAPS/HYDROPERIOD/
    cp $from_path/MAPS/POND/day_pond.inp MAPS/POND/
    cp $from_path/MAPS/POND/pondiff.inp MAPS/POND/
    cp $from_path/MAPS/POND/pondiff_nsm-wmm.cf MAPS/POND/
    cp $from_path/OTHER_WBUD/ASR_BUD/asrbud.def OTHER_WBUD/ASR_BUD/
    cp $from_path/OTHER_WBUD/NOR_STO_BUD/noresbud.def OTHER_WBUD/NOR_STO_BUD/
    cp $from_path/WBUD/wmmwbud.cf WBUD/
endif

```

wmmwbud.cf

```
*****
* WATER BUDGET DEFINITION DATA FILE FOR THE SOUTH FLORIDA WATER *
* MANAGEMENT MODEL (SFWMM) / NATURAL SYSTEM MODEL (NSM) *
* WATER BUDGET POST-PROCESSING PROGRAM *
*****
"95BSRR with Perm PWS & Caloo&Istokpoga acreages & 83BASE & ENP MinDelivery
Sched" = Run Title
1965 = Simulation Start Year
1995 = Simulation End Year
10560. = Grid Cell Size in x (E-W) direction(ft)
10560. = Grid Cell Size in y (N-S) direction(ft)
19 = No. of Sub-areas to Summarize (including entire area)
SFWMM = Model that budget is desired for (NSM or SFWMM)
*** Input Files Pertinent to SFWMM or NSM ***
"./pumpage.bin" = Input Filename for monthly well pumpage
"./mthly_levee_spg.dat" = Input Filename for monthly levee seepage
"./lkrfetsto" = Input Filename for monthly lake rf, et & eom storage
"./unsatdph.bin" = Input Filename for month-end depth of storage in unsat zone
"/vol/hsm/data/sfwmm/Input65_95/flo_v3.5.1.dss" = DSS Input Filename
for historical structure flows
"./str2x2.dss" = DSS Output Filename for simulated structure flows
1 = flag for performing unsaturated zone budgets for the SFWMM
*** Input Files Pertinent to the SFWMM for Unsaturated Zone Water Budgets
"./infiltr_perc.bin" = Input Filename for monthly infiltration & percolation
"./et_components.bin" = Input Filename for monthly et components
"./supply.bin" = Input Filename for monthly net irrigation supplies
"./et_unsat_unacct.bin" = Input Filename unaccounted for unsaturated ET
*** Input Files Common to Both Models ***
"./rainfall.bin" = Input Filename for monthly rainfall
"./et_total.bin" = Input Filename for monthly evapotranspiration
"./stage.bin" = Input Filename for month-end stages
"./ponding.bin" = Input Filename for month-end ponding
"./surface_flow.bin" = Input Filename for monthly overland flow
"./gw_flow.bin" = Input Filename for monthly groundwater flow
"/vol/hsm1/data/sfwmm/LEC2020/95BSRR/INPUT/V3.7/Perm3/statdta" = Input
Filename for storage coefficients
*** Output Files ***
"monbud" = Output Filename for Monthly Water Budget Summary
"annbud" = Output Filename for Annual Budget Summary
"wetbud" = Output Filename for Wet Season Budget Summary
"WET SEASON(JUN-OCT)" = String for Wet Season Output Title
6 = First Month of Wet Season
10 = Last Month of Wet Season
"drybud" = Output Filename for Dry Season Budget Summary
"DRY SEASON(NOV-MAY)" = String for Dry Season Output Title
11 = First Month of Dry Season
5 = Last Month of Dry Season
"wyrbud" "plotfile" = Output Filename for Water Year Budget Summary
"WATER-YEAR(OCT-SEP)" = String for Water Year Output Title
10 = First Month of Water Year
9 = Last Month of Water Year
0 = flag for printing monthly columnar data to subarea named files
*****
```

ENTIRE\_SFWMMArea = 1st Sub-area Name (must be entire model area - no budget printed)

\*\*\*\*\*

\* SUBAREA BOUNDARY

1 = Southernmost Row No.

65 = Northernmost Row No.

\* Subarea Definition (Row #'s in descending order)

\* ROW# MIN COL# MAX COL#

65	23	39
64	23	39
63	23	39
62	23	40
61	23	40
60	22	40
59	21	40
58	21	41
57	21	41
56	21	41
55	20	41
54	14	41
53	13	41
52	13	41
51	13	41
50	13	41
49	13	41
48	13	40
47	13	40
46	13	40
45	13	40
44	13	40
43	14	40
42	14	40
41	1	40
40	1	40
39	1	40
38	1	40
37	1	39
36	1	39
35	1	39
34	1	39
33	1	39
32	1	39
31	1	39
30	1	38
29	1	38
28	1	38
27	1	38
26	1	37
25	3	37
24	5	37
23	6	37
22	6	36
21	7	35
20	8	35
19	10	34
18	10	34
17	10	33

16	11	33
15	11	33
14	12	32
13	13	32
12	12	32
11	12	32
10	12	32
9	13	32
8	13	32
7	13	31
6	13	30
5	13	29
4	13	27
3	15	25
2	15	21
1	15	21

\*

\* LEVEE SEEPAGE

0 = No. of Levee Seepage Segments

\*

\* STRUCTURE INFLOWS AND OUTFLOWS

10 = No. of Structure Inflows to Subbasin

\* STRUCTURE NAME

"/SFWMM/L8CP/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S352/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S351/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S354/FLOW//1DAY/SIMULATED/"  
"/SFWMM/AGQ/FLOW//1DAY/SIMULATED/"  
"/SFWMM/G136/FLOW//1DAY/BASE/"  
"/SFWMM/G88/FLOW//1DAY/BASE/"  
"/SFWMM/G89/FLOW//1DAY/BASE/"  
"/SFWMM/G155/FLOW//1DAY/BASE/"  
"/SFWMM/S190/FLOW//1DAY/HISTORICAL/"

28 = No. of Structure Outflows from Subbasin

\* STRUCTURE NAME

"/SFWMM/S2PMP/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S3PMP/FLOW//1DAY/SIMULATED/"  
"/SFWMM/LOXRQ/FLOW//1DAY/SIMULATED/"  
"/SFWMM/PBDRQ/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S155/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S40/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S41/FLOW//1DAY/SIMULATED/"  
"/SFWMM/RVBDRQ/FLOW//1DAY/SIMULATED/"  
"/SFWMM/C10ABK/FLOW//1DAY/SIMULATED/"  
"/SFWMM/HLSBEQ/FLOW//1DAY/SIMULATED/"  
"/SFWMM/G57DNQ/FLOW//1DAY/SIMULATED/"  
"/SFWMM/QC13E/FLOW//1DAY/SIMULATED/"  
"/SFWMM/Q2C57/FLOW//1DAY/SIMULATED/"  
"/SFWMM/C10Q/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S29DNQ/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S28/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S27/FLOW//1DAY/SIMULATED/"  
"/SFWMM/C6EQ/FLOW//1DAY/SIMULATED/"  
"/SFWMM/G97/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S22/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S123/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S21/FLOW//1DAY/SIMULATED/"

```

"/SFWMM/S21A/FLOW//1DAY/SIMULATED/"
"/SFWMM/S20F/FLOW//1DAY/SIMULATED/"
"/SFWMM/S20G/FLOW//1DAY/SIMULATED/"
"/SFWMM/S20/FLOW//1DAY/SIMULATED/"
"/SFWMM/S197/FLOW//1DAY/SIMULATED/"
"/SFWMM/HW29Q/FLOW//1DAY/SIMULATED/"
*****
LAKE_OKEECHOBEE = Sub-area Name
*****
466000. = Lake Surface Area within Levees (acres)
* STRUCTURE INFLOWS AND OUTFLOWS
10 = No. of Structure Inflows to Subbasin
* STRUCTURE NAME
"/SFWMM/MDSPOS/FLOW//1DAY/EST_HIST_V3.7/"
"/SFWMM/S2PMP/FLOW//1DAY/SIMULATED/"
"/SFWMM/S3PMP/FLOW//1DAY/SIMULATED/"
"/SFWMM/S65E/FLOW//1DAY/HISTORICAL/"
"/SFWMM/TCNSQ/FLOW//1DAY/EST_HIST_V3.4/"
"/SFWMM/C10ABK/FLOW//1DAY/SIMULATED/"
"/SFWMM/S236RO/FLOW//1DAY/SIMULATED/"
"/SFWMM/AGQRF/FLOW//1DAY/SIMULATED/"
"/SFWMM/S77BK/FLOW//1DAY/SIMULATED/"
"/SFWMM/S308BK/FLOW//1DAY/SIMULATED/"
13 = No. of Structure Outflows from Subbasin
* STRUCTURE NAME
"/SFWMM/MDSNEG/FLOW//1DAY/EST_HIST_V3.7/"
"/SFWMM/S77OUT/FLOW//1DAY/SIMULATED/"
"/SFWMM/S308OT/FLOW//1DAY/SIMULATED/"
"/SFWMM/S352/FLOW//1DAY/SIMULATED/"
"/SFWMM/S351/FLOW//1DAY/SIMULATED/"
"/SFWMM/S354/FLOW//1DAY/SIMULATED/"
"/SFWMM/L8CP/FLOW//1DAY/SIMULATED/"
"/SFWMM/AGQWS/FLOW//1DAY/SIMULATED/"
"/SFWMM/S4DMD/FLOW//1DAY/SIMULATED/"
"/SFWMM/S236WS/FLOW//1DAY/SIMULATED/"
"/SFWMM/S235TC/FLOW//1DAY/SIMULATED/"
"/SFWMM/DMDSEM/FLOW//1DAY/SIMULATED/"
"/SFWMM/LKTFPL/FLOW//1DAY/SIMULATED/"
*****
EAA+HOL+ROT+298 = Sub-area Name
*****
* SUBAREA BOUNDARY
42 = Southernmost Row No.
61 = Northernmost Row No.
* SUBAREA DEFINITION (Row #'s in descending order)
* ROW# MIN COL# MAX COL#
61 23 24
60 22 25
59 21 26
58 21 27
57 21 28
56 21 29
55 20 29
54 14 29
53 13 29
52 13 28
51 13 28

```

50	13	27
49	13	27
48	13	27
47	13	27
46	13	27
45	13	26
44	13	26
43	14	25
42	14	25

\*

\* LEVEE SEEPAGE

0 = No. of Levee Seepage Segments

\*

\* STRUCTURE INFLOWS AND OUTFLOWS

7 = No. of Structure Inflows to Subbasin

\* STRUCTURE NAME

"/SFWMM/S354/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S351/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S352/FLOW//1DAY/SIMULATED/"  
"/SFWMM/AGQWS/FLOW//1DAY/SIMULATED/"  
"/SFWMM/G136/FLOW//1DAY/BASE/"  
"/SFWMM/G88/FLOW//1DAY/BASE/"  
"/SFWMM/SUGDMD/FLOW//1DAY/SIMULATED/"

13 = No. of Structure Outflows from Subbasin

\* STRUCTURE NAME

"/SFWMM/S2PMP/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S3PMP/FLOW//1DAY/SIMULATED/"  
"/SFWMM/SUGRF/FLOW//1DAY/SIMULATED/"  
"/SFWMM/AGQRF/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S5A1/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S6/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S7/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S150/FLOW//1DAY/SIMULATED/"  
"/SFWMM/LKTSEM/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S8/FLOW//1DAY/SIMULATED/"  
"/SFWMM/G204/FLOW//1DAY/SIMULATED/"  
"/SFWMM/G205/FLOW//1DAY/SIMULATED/"  
"/SFWMM/G206/FLOW//1DAY/SIMULATED/"

\*\*\*\*\*

HOLEY\_LAND = Sub-area Name

\*\*\*\*\*

\* SUBAREA BOUNDARY

42 = Southernmost Row No.

45 = Northernmost Row No.

\* SUBAREA DEFINITION (Row #'s in descending order)

ROW#	MIN COL#	MAX COL#
45	17	20
44	18	20
43	18	20
42	18	21

\*

\* LEVEE SEEPAGE

0 = No. of Levee Seepage Segments

\*

\* STRUCTURE INFLOWS AND OUTFLOWS

1 = No. of Structure Inflows to Subbasin

\* STRUCTURE NAME

```

    "/SFWMM/HLYQIN/FLOW//1DAY/SIMULATED/"
3 = No. of Structure Outflows from Subbasin
* STRUCTURE NAME
    "/SFWMM/G204/FLOW//1DAY/SIMULATED/"
    "/SFWMM/G205/FLOW//1DAY/SIMULATED/"
    "/SFWMM/G206/FLOW//1DAY/SIMULATED/"
*****
ROTEN_TRACT = Sub-area Name
*****
* SUBAREA BOUNDARY
42 = Southernmost Row No.
46 = Northernmost Row No.
* SUBAREA DEFINITION (Row #'s in descending order)
* ROW#  MIN COL#  MAX COL#
    46      15      16
    45      15      16
    44      15      17
    43      15      17
    42      15      17
*
* LEVEE SEEPAGE
0 = No. of Levee Seepage Segments
*
* STRUCTURE INFLOWS AND OUTFLOWS
0 = No. of Structure Inflows to Subbasin
2 = No. of Structure Outflows from Subbasin
* STRUCTURE NAME
    "/SFWMM/RTECV1/FLOW//1DAY/SIMULATED/"
    "/SFWMM/RTECV2/FLOW//1DAY/SIMULATED/"
*****
S5A_COMPLEX = Sub-area Name
*****
* SUBAREA BOUNDARY
54 = Southernmost Row No.
54 = Northernmost Row No.
* SUBAREA DEFINITION (Row #'s in descending order)
* ROW#  MIN COL#  MAX COL#
    54      30      30
*
* LEVEE SEEPAGE
0 = No. of Levee Seepage Segments
*
* STRUCTURE INFLOWS AND OUTFLOWS
4 = No. of Structure Inflows to Subbasin
* STRUCTURE NAME
    "/SFWMM/S5A1/FLOW//1DAY/SIMULATED/"
    "/SFWMM/S5A3SO/FLOW//1DAY/SIMULATED/"
    "/SFWMM/S5A4W/FLOW//1DAY/SIMULATED/"
    "/SFWMM/S5A2NO/FLOW//1DAY/SIMULATED/"
3 = No. of Structure Outflows from Subbasin
* STRUCTURE NAME
    "/SFWMM/S5A2SO/FLOW//1DAY/SIMULATED/"
    "/SFWMM/S5A4E/FLOW//1DAY/SIMULATED/"
    "/SFWMM/S5A3NO/FLOW//1DAY/SIMULATED/"
*****
WATER_CONSERVATION_AREA-1 = Sub-area Name
*****

```



```

* SUBAREA BOUNDARY
43 = Southernmost Row No.
53 = Northernmost Row No.
* SUBAREA DEFINITION (Row #'s in descending order)
* ROW#  MIN COL#  MAX COL#
      53      30      30
      52      29      31
      51      29      33
      50      28      34
      49      28      34
      48      28      34
      47      28      34
      46      29      34
      45      29      34
      44      30      34
      43      31      33
*
* LEVEE SEEPAGE
1 = No. of Levee Seepage Segments
L-40 = Name of Levee Seepage Segment #1
11 = No. of Levee Seepage Cells in X-direction for Segment #1
*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY
  30,53    1  (1=cell inside subbasin boundary,0=outside)
  31,52    1
  33,51    1
  34,50    1
  34,49    1
  34,48    1
  34,47    1
  34,46    1
  34,45    1
  34,44    1
  33,43    1
1 = No. of Levee Seepage Cells in Y-direction for Segment #1
*COORDINATES OF CELL IMMEDIATELY NORTH OF LEVEE SEEPAGE BOUNDARY
  32,52    0  (1=cell inside subbasin boundary,0=outside)
*
* STRUCTURE INFLOWS AND OUTFLOWS
4 = No. of Structure Inflows to Subbasin
* STRUCTURE NAME
  "/SFWMM/S5A2SO/FLOW//1DAY/SIMULATED/"
  "/SFWMM/S6/FLOW//1DAY/SIMULATED/"
  "/SFWMM/L8TCA1/FLOW//1DAY/SIMULATED/"
  "/SFWMM/ACMERF/FLOW//1DAY/SIMULATED/"
6 = No. of Structure Outflows from Subbasin
* STRUCTURE NAME
  "/SFWMM/S5A2NO/FLOW//1DAY/SIMULATED/"
  "/SFWMM/S10/FLOW//1DAY/SIMULATED/"
  "/SFWMM/S39/FLOW//1DAY/SIMULATED/"
  "/SFWMM/LWDD/FLOW//1DAY/SIMULATED/"
  "/SFWMM/ACMEWS/FLOW//1DAY/SIMULATED/"
  "/SFWMM/S10E/FLOW//1DAY/SIMULATED/"
*****
WATER_CONSERVATION_AREA-2A = Sub-area Name
*****
* SUBAREA BOUNDARY
37 = Southernmost Row No.

```

46 = Northernmost Row No.  
 \* SUBAREA DEFINITION (Row #'s in descending order)  
 \* ROW# MIN COL# MAX COL#

46	28	28
45	27	28
44	27	29
43	26	30
42	26	32
41	26	32
40	27	32
39	27	32
38	28	30
37	28	28

\*  
 \* LEVEE SEEPAGE  
 2 = No. of Levee Seepage Segments  
 L-36.N = Name of Levee Seepage Segment #1  
 4 = No. of Levee Seepage Cells in X-direction for Segment #1  
 \*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY

32,42	1 (1=cell inside subbasin boundary,0=outside)
32,41	1
32,40	1
32,39	1

0 = No. of Levee Seepage Cells in Y-direction for Segment #1  
 L-35B = Name of Levee Seepage Segment #2  
 1 = No. of Levee Seepage Cells in X-direction for Segment #2  
 \*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY

28,37	1 (1=cell inside subbasin boundary,0=outside)
-------	---

3 = No. of Levee Seepage Cells in Y-direction for Segment #2  
 \*COORDINATES OF CELL IMMEDIATELY NORTH OF LEVEE SEEPAGE BOUNDARY

29,38	1 (1=cell inside subbasin boundary,0=outside)
30,38	1
31,39	1

\*  
 \* STRUCTURE INFLOWS AND OUTFLOWS  
 3 = No. of Structure Inflows to Subbasin  
 \* STRUCTURE NAME  
 "/SFWMM/S7/FLOW//1DAY/SIMULATED/"  
 "/SFWMM/S10/FLOW//1DAY/SIMULATED/"  
 "/SFWMM/S10E/FLOW//1DAY/SIMULATED/"  
 6 = No. of Structure Outflows from Subbasin  
 \* STRUCTURE NAME  
 "/SFWMM/S11/FLOW//1DAY/SIMULATED/"  
 "/SFWMM/S38/FLOW//1DAY/SIMULATED/"  
 "/SFWMM/S143/FLOW//1DAY/SIMULATED/"  
 "/SFWMM/S144/FLOW//1DAY/SIMULATED/"  
 "/SFWMM/S145/FLOW//1DAY/SIMULATED/"  
 "/SFWMM/S146/FLOW//1DAY/SIMULATED/"  
 \*\*\*\*\*  
 WATER\_CONSERVATION\_AREA-3A = Sub-area Name  
 \*\*\*\*\*  
 \* SUBAREA BOUNDARY  
 23 = Southernmost Row No.  
 41 = Northernmost Row No.  
 \* SUBAREA DEFINITION (Row #'s in descending order)  
 \* ROW# MIN COL# MAX COL#

41	16	25
----	----	----

40	16	26
39	16	26
38	16	27
37	16	27
36	16	27
35	16	27
34	16	27
33	16	27
32	16	27
31	16	26
30	15	25
29	16	25
28	16	24
27	16	24
26	16	23
25	16	22
24	16	22
23	16	21

\*

\* LEVEE SEEPAGE

2 = No. of Levee Seepage Segments

L-68A = Name of Levee Seepage Segment #1

4 = No. of Levee Seepage Cells in X-direction for Segment #1

\*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY

27,35	1	(1=cell inside subbasin boundary,0=outside)
27,34	1	
27,33	1	
27,32	1	

0 = No. of Levee Seepage Cells in Y-direction for Segment #1

L-67 = Name of Levee Seepage Segment #2

8 = No. of Levee Seepage Cells in X-direction for Segment #2

\*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY

26,31	1	(1=cell inside subbasin boundary,0=outside)
25,30	1	
25,29	1	
24,28	1	
24,27	1	
23,26	1	
22,25	1	
22,24	1	

0 = No. of Levee Seepage Cells in Y-direction for Segment #2

\*

\* STRUCTURE INFLOWS AND OUTFLOWS

11 = No. of Structure Inflows to Subbasin

\* STRUCTURE NAME

"/SFMMM/S8/FLOW//1DAY/SIMULATED/"
"/SFMMM/S150/FLOW//1DAY/SIMULATED/"
"/SFMMM/S11/FLOW//1DAY/SIMULATED/"
"/SFMMM/S9/FLOW//1DAY/SIMULATED/"
"/SFMMM/L28WQ/FLOW//1DAY/SIMULATED/"
"/SFMMM/S140A/FLOW//1DAY/SIMULATED/"
"/SFMMM/G155/FLOW//1DAY/BASE/"
"/SFMMM/G204/FLOW//1DAY/SIMULATED/"
"/SFMMM/G205/FLOW//1DAY/SIMULATED/"
"/SFMMM/G206/FLOW//1DAY/SIMULATED/"
"/SFMMM/S142W/FLOW//1DAY/SIMULATED/"

10 = No. of Structure Outflows from Subbasin

```

* STRUCTURE NAME
"/SFWMM/S151RG/FLOW//1DAY/SIMULATED/"
"/SFWMM/S151WS/FLOW//1DAY/SIMULATED/"
"/SFWMM/S333/FLOW//1DAY/SIMULATED/"
"/SFWMM/S12A/FLOW//1DAY/SIMULATED/"
"/SFWMM/S12B/FLOW//1DAY/SIMULATED/"
"/SFWMM/S12C/FLOW//1DAY/SIMULATED/"
"/SFWMM/S12D/FLOW//1DAY/SIMULATED/"
"/SFWMM/S343/FLOW//1DAY/SIMULATED/"
"/SFWMM/S344/FLOW//1DAY/SIMULATED/"
"/SFWMM/S142E/FLOW//1DAY/SIMULATED/"
*****
WATER_CONSERVATION_AREA-2B = Sub-area Name
*****
* SUBAREA BOUNDARY
35 = Southernmost Row No.
38 = Northernmost Row No.
* SUBAREA DEFINITION (Row #'s in descending order)
* ROW#   MIN COL#   MAX COL#
      38      31      32
      37      29      32
      36      28      31
      35      30      30
*
* LEVEE SEEPAGE
3 = No. of Levee Seepage Segments
L-35B = Name of Levee Seepage Segment #1
1 = No. of Levee Seepage Cells in X-direction for Segment #1
*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY
  28,37    0 (1=cell inside subbasin boundary,0=outside)
3 = No. of Levee Seepage Cells in Y-direction for Segment #1
*COORDINATES OF CELL IMMEDIATELY NORTH OF LEVEE SEEPAGE BOUNDARY
  29,38    0 (1=cell inside subbasin boundary,0=outside)
  30,38    0
  31,39    0
L-36(S) = Name of Levee Seepage Segment #2
2 = No. of Levee Seepage Cells in X-direction for Segment #2
*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY
  32,38    1 (1=cell inside subbasin boundary,0=outside)
  32,37    1
0 = No. of Levee Seepage Cells in Y-direction for Segment #2
L35&35A = Name of Levee Seepage Segment #3
1 = No. of Levee Seepage Cells in X-direction for Segment #3
*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY
  31,36    1 (1=cell inside subbasin boundary,0=outside)
2 = No. of Levee Seepage Cells in Y-direction for Segment #3
*COORDINATES OF CELL IMMEDIATELY NORTH OF LEVEE SEEPAGE BOUNDARY
  29,36    1 (1=cell inside subbasin boundary,0=outside)
  30,35    1
*
* STRUCTURE INFLOWS AND OUTFLOWS
6 = No. of Structure Inflows to Subbasin
* STRUCTURE NAME
"/SFWMM/S143/FLOW//1DAY/SIMULATED/"
"/SFWMM/S144/FLOW//1DAY/SIMULATED/"
"/SFWMM/S145/FLOW//1DAY/SIMULATED/"
"/SFWMM/S146/FLOW//1DAY/SIMULATED/"

```

```

"/SFWMM/S142E/FLOW//1DAY/SIMULATED/"
"/SFWMM/G123/FLOW//1DAY/SIMULATED/"
2 = No. of Structure Outflows from Subbasin
* STRUCTURE NAME
"/SFWMM/S34/FLOW//1DAY/SIMULATED/"
"/SFWMM/S142W/FLOW//1DAY/SIMULATED/"
*****
WATER_CONSERVATION_AREA-3B = Sub-area Name
*****
* SUBAREA BOUNDARY
23 = Southernmost Row No.
31 = Northernmost Row No.
* SUBAREA DEFINITION (Row #'s in descending order)
* ROW# MIN COL# MAX COL#
    31      27      27
    30      26      27
    29      26      27
    28      25      27
    27      25      27
    26      24      26
    25      23      26
    24      23      26
    23      22      26
*
* LEVEE SEEPAGE
3 = No. of Levee Seepage Segments
L-67 = Name of Levee Seepage Segment #1
8 = No. of Levee Seepage Cells in X-direction for Segment #1
*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY
    26,31      0 (1=cell inside subbasin boundary,0=outside)
    25,30      0
    25,29      0
    24,28      0
    24,27      0
    23,26      0
    22,25      0
    22,24      0
0 = No. of Levee Seepage Cells in Y-direction for Segment #1
L30&33 = Name of Levee Seepage Segment #2
8 = No. of Levee Seepage Cells in X-direction for Segment #2
*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY
    27,31      1 (1=cell inside subbasin boundary,0=outside)
    27,30      1
    27,28      1
    27,27      1
    26,26      1
    26,25      1
    26,24      1
    26,23      1
0 = No. of Levee Seepage Cells in Y-direction for Segment #2
L-29E = Name of Levee Seepage Segment #3
0 = No. of Levee Seepage Cells in X-direction for Segment #3
5 = No. of Levee Seepage Cells in Y-direction for Segment #3
*COORDINATES OF CELL IMMEDIATELY NORTH OF LEVEE SEEPAGE BOUNDARY
    22,23      1 (1=cell inside subbasin boundary,0=outside)
    23,23      1
    24,23      1

```

```

25,23    1
26,23    1
*
*  STRUCTURE INFLOWS AND OUTFLOWS
2  = No. of Structure Inflows to Subbasin
*  STRUCTURE NAME
    "/SFWMM/S151RG/FLOW//1DAY/SIMULATED/"
    "/SFWMM/S151WS/FLOW//1DAY/SIMULATED/"
5  = No. of Structure Outflows from Subbasin
*  STRUCTURE NAME
    "/SFWMM/S31RG/FLOW//1DAY/SIMULATED/"
    "/SFWMM/S31/FLOW//1DAY/SIMULATED/"
    "/SFWMM/S337/FLOW//1DAY/SIMULATED/"
    "/SFWMM/LSPL33/FLOW//1DAY/SIMULATED/"
    "/SFWMM/LSPC6/FLOW//1DAY/SIMULATED/"
*****
EVERGLADES_NATIONAL_PARK_EAST = Sub-area Name
*****
*  Subarea Boundary
3  = Southernmost Row No.
22 = Northernmost Row No.
*  Subarea Definition (Row #'s in descending order)
*  ROW#  MIN COL#  MAX COL#
    22    16    26
    21    15    26
    20    15    26
    19    15    26
    18    15    26
    17    15    26
    16    15    25
    15    15    24
    14    15    24
    13    15    24
    12    15    24
    11    15    23
    10    15    24
     9    15    24
     8    16    24
     7    16    24
     6    16    24
     5    16    27
     4    16    25
     3    16    21
*
*  LEVEE SEEPAGE
3  = No. of Levee Seepage Segments
L-29E = Name of Levee Seepage Segment #1
0  = No. of Levee Seepage Cells in X-direction for Segment #1
5  = No. of Levee Seepage Cells in Y-direction for Segment #1
*COORDINATES OF CELL IMMEDIATELY NORTH OF LEVEE SEEPAGE BOUNDARY
    22,23    0 (1=cell inside subbasin boundary,0=outside)
    23,23    0
    24,23    0
    25,23    0
    26,23    0
L31N.N = Name of Levee Seepage Segment #2
6  = No. of Levee Seepage Cells in X-direction for Segment #2

```

\*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY

26,22 1  
26,21 1 (1=cell inside subbasin boundary,0=outside)  
26,20 1  
26,19 1  
26,18 1  
26,17 1

0 = No. of Levee Seepage Cells in Y-direction for Segment #2

L-31N(S) = Name of Levee Seepage Segment #3

4 = No. of Levee Seepage Cells in X-direction for Segment #3

\*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY

25,16 1 (1=cell inside subbasin boundary,0=outside)  
24,15 1  
24,14 1  
24,13 1

0 = No. of Levee Seepage Cells in Y-direction for Segment #3

\*

\* STRUCTURE INFLOWS AND OUTFLOWS

6 = No. of Structure Inflows to Subbasin

\* STRUCTURE NAME

"/SFWMM/S12A/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S12B/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S12C/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S12D/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S333/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S174/FLOW//1DAY/SIMULATED/"

1 = No. of Structure Outflows from Subbasin

\* STRUCTURE NAME

"/SFWMM/S334/FLOW//1DAY/SIMULATED/"

\*\*\*\*\*

INDIAN\_TRAILS\_RESERVOIR = Sub-area Name

\*\*\*\*\*

\* SUBAREA BOUNDARY

59 = Southernmost Row No.

59 = Northernmost Row No.

\* SUBAREA DEFINITION (Row #'s in descending order)

\* ROW# MIN COL# MAX COL#  
59 29 29

\*

\* LEVEE SEEPAGE

0 = No. of Levee Seepage Segments

\*

\* STRUCTURE INFLOWS AND OUTFLOWS

1 = No. of Structure Inflows to Subbasin

\* STRUCTURE NAME

"/SFWMM/SITWCD/FLOW//1DAY/SIMULATED/"

2 = No. of Structure Outflows from Subbasin

\* STRUCTURE NAME

"/SFWMM/RESTL8/FLOW//1DAY/SIMULATED/"  
"/SFWMM/RESL80/FLOW//1DAY/SIMULATED/"

\*\*\*\*\*

L-8\_BASIN = Sub-area Name

\*\*\*\*\*

\* SUBAREA BOUNDARY

55 = Southernmost Row No.

63 = Northernmost Row No.

\* SUBAREA DEFINITION (Row #'s in descending order)

ROW#	MIN COL#	MAX COL#
63	23	23
62	23	24
61	25	25
60	26	26
59	27	28
58	28	29
57	29	29
56	30	30
55	30	30

\*

\* LEVEE SEEPAGE

0 = No. of Levee Seepage Segments

\*

\* STRUCTURE INFLOWS AND OUTFLOWS

8 = No. of Structure Inflows to Subbasin

\* STRUCTURE NAME

"/SFWMM/L8CP/FLOW//1DAY/SIMULATED/"  
"/SFWMM/RESL80/FLOW//1DAY/SIMULATED/"  
"/SFWMM/RESTL8/FLOW//1DAY/SIMULATED/"  
"/SFWMM/BKMCL8/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S5A3NO/FLOW//1DAY/SIMULATED/"  
"/SFWMM/DPRESO/FLOW//1DAY/SIMULATED/"  
"/SFWMM/CORBT1/FLOW//1DAY/SIMULATED/"  
"/SFWMM/CORBT2/FLOW//1DAY/SIMULATED/"

5 = No. of Structure Outflows from Subbasin

\* STRUCTURE NAME

"/SFWMM/C10ABK/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S2TMCL/FLOW//1DAY/SIMULATED/"  
"/SFWMM/L8C51W/FLOW//1DAY/SIMULATED/"  
"/SFWMM/L8TCA1/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S5A3SO/FLOW//1DAY/SIMULATED/"

\*\*\*\*\*

EASTERN PALM BEACH COUNTY = Sub-area Name

\*\*\*\*\*

\* SUBAREA BOUNDARY

42 = Southernmost Row No.

65 = Northernmost Row No.

\* SUBAREA DEFINITION (Row #'s in descending order)

ROW#	MIN COL#	MAX COL#
65	36	38
64	36	39
63	24	39
62	25	39
61	26	39
60	27	40
59	29	40
58	30	40
57	30	40
56	31	40
55	31	40
54	30	40
53	31	40
52	32	40
51	34	40
50	35	40
49	35	40



48	35	40
47	35	40
46	35	39
45	35	39
44	35	39
43	34	39
42	33	39

\*

\* LEVEE SEEPAGE

2 = No. of Levee Seepage Segments

L-40 = Name of Levee Seepage Segment #1

11 = No. of Levee Seepage Cells in X-direction for Segment #1

\*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY

30,53	0	(1=cell inside subbasin boundary,0=outside)
31,52	0	
33,51	0	
34,50	0	
34,49	0	
34,48	0	
34,47	0	
34,46	0	
34,45	0	
34,44	0	
33,43	0	

1 = No. of Levee Seepage Cells in Y-direction for Segment #1

\*COORDINATES OF CELL IMMEDIATELY NORTH OF LEVEE SEEPAGE BOUNDARY

32,52	1	(1=cell inside subbasin boundary,0=outside)
-------	---	---

L36.N1 = Name of Levee Seepage Segment #2

1 = No. of Levee Seepage Cells in X-direction for Segment #2

\*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY

32,42	0	
-------	---	--

0 = No. of Levee Seepage Cells in Y-direction for Segment #2

\*

\* STRUCTURE INFLOWS AND OUTFLOWS

8 = No. of Structure Inflows to Subbasin

\* STRUCTURE NAME

"/SFWMM/S5A1/FLOW//1DAY/SIMULATED/"  
"/SFWMM/LWDD/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S39/FLOW//1DAY/SIMULATED/"  
"/SFWMM/ACMEWS/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S2TMCL/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S5A3SO/FLOW//1DAY/SIMULATED/"  
"/SFWMM/L8C51W/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S5A2NO/FLOW//1DAY/SIMULATED/"

17 = No. of Structure Outflows from Subbasin

\* STRUCTURE NAME

"/SFWMM/S5A2SO/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S5A3NO/FLOW//1DAY/SIMULATED/"  
"/SFWMM/LOXRVQ/FLOW//1DAY/SIMULATED/"  
"/SFWMM/PBDRQ/FLOW//1DAY/SIMULATED/"  
"/SFWMM/ACMERF/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S155/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S41/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S40/FLOW//1DAY/SIMULATED/"  
"/SFWMM/RVBDRQ/FLOW//1DAY/SIMULATED/"  
"/SFWMM/HLSBEQ/FLOW//1DAY/SIMULATED/"  
"/SFWMM/HLSBR/FLOW//1DAY/SIMULATED/"

```

"/SFWMM/RESTL8/FLOW//1DAY/SIMULATED/"
"/SFWMM/BKMCL8/FLOW//1DAY/SIMULATED/"
"/SFWMM/RESL80/FLOW//1DAY/SIMULATED/"
"/SFWMM/DPRESO/FLOW//1DAY/SIMULATED/"
"/SFWMM/CORBT1/FLOW//1DAY/SIMULATED/"
"/SFWMM/CORBT2/FLOW//1DAY/SIMULATED/"
*****
EASTERN_BROWARD_COUNTY = Sub-area Name
*****
* SUBAREA BOUNDARY
27 = Southernmost Row No.
41 = Northernmost Row No.
* SUBAREA DEFINITION (Row #'s in descending order)
* ROW# MIN COL# MAX COL#
    41      33      39
    40      33      39
    39      33      39
    38      33      39
    37      33      38
    36      32      38
    35      28      38
    34      28      38
    33      28      38
    32      28      38
    31      28      38
    30      28      38
    29      28      38
    28      36      38
    27      37      37
*
* LEVEE SEEPAGE
5 = No. of Levee Seepage Segments
L36.N2 = Name of Levee Seepage Segment #1
3 = No. of Levee Seepage Cells in X-direction for Segment #1
*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY
    32,41    0 (1=cell inside subbasin boundary,0=outside)
    32,40    0
    32,39    0
0 = No. of Levee Seepage Cells in Y-direction for Segment #1
L-36(S) = Name of Levee Seepage Segment #2
2 = No. of Levee Seepage Cells in X-direction for Segment #2
*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY
    32,38    0 (1=cell inside subbasin boundary,0=outside)
    32,37    0
0 = No. of Levee Seepage Cells in Y-direction for Segment #2
L35&35A = Name of Levee Seepage Segment #3
1 = No. of Levee Seepage Cells in X-direction for Segment #3
*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY
    31,36    0 (1=cell inside subbasin boundary,0=outside)
1 = No. of Levee Seepage Cells in Y-direction for Segment #3
*COORDINATES OF CELL IMMEDIATELY NORTH OF LEVEE SEEPAGE BOUNDARY
    29,36    0 (1=cell inside subbasin boundary,0=outside)
L-68A = Name of Levee Seepage Segment #4
4 = No. of Levee Seepage Cells in X-direction for Segment #4
*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY
    27,35    0 (1=cell inside subbasin boundary,0=outside)
    27,34    0

```

```

    27,33    0
    27,32    0
0 = No. of Levee Seepage Cells in Y-direction for Segment #4
L-33 = Name of Levee Seepage Segment #5
2 = No. of Levee Seepage Cells in X-direction for Segment #5
*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY
    27,31    0 (1=cell inside subbasin boundary,0=outside)
    27,30    0
0 = No. of Levee Seepage Cells in Y-direction for Segment #5
*
* STRUCTURE INFLOWS AND OUTFLOWS
6 = No. of Structure Inflows to Subbasin
* STRUCTURE NAME
"/SFWMM/S34/FLOW//1DAY/SIMULATED/"
"/SFWMM/HLSBR/FLOW//1DAY/SIMULATED/"
"/SFWMM/S38/FLOW//1DAY/SIMULATED/"
"/SFWMM/C9DRSQ/FLOW//1DAY/SIMULATED/"
"/SFWMM/LSPL33/FLOW//1DAY/SIMULATED/"
"/SFWMM/C9DESQ/FLOW//1DAY/SIMULATED/"
8 = No. of Structure Outflows from Subbasin
* STRUCTURE NAME
"/SFWMM/G123/FLOW//1DAY/SIMULATED/"
"/SFWMM/S9/FLOW//1DAY/SIMULATED/"
"/SFWMM/G57DNQ/FLOW//1DAY/SIMULATED/"
"/SFWMM/QC13E/FLOW//1DAY/SIMULATED/"
"/SFWMM/Q2C57/FLOW//1DAY/SIMULATED/"
"/SFWMM/C10Q/FLOW//1DAY/SIMULATED/"
"/SFWMM/S29DNQ/FLOW//1DAY/SIMULATED/"
"/SFWMM/S32/FLOW//1DAY/SIMULATED/"
*****
EASTERN_DADE_COUNTY = Sub-area Name
*****
* SUBAREA BOUNDARY
5 = Southernmost Row No.
28 = Northernmost Row No.
* SUBAREA DEFINITION (Row #'s in descending order)
* ROW#  MIN COL#  MAX COL#
    28      28      35
    27      28      36
    26      27      36
    25      27      36
    24      27      36
    23      27      36
    22      27      35
    21      27      34
    20      27      34
    19      27      33
    18      27      33
    17      27      32
    16      26      32
    15      25      32
    14      25      31
    13      25      31
    12      25      31
    11      24      31
    10      25      31
     9      25      31

```

8	25	31
7	25	30
6	25	29
5	28	28

\*

\* LEVEE SEEPAGE

3 = No. of Levee Seepage Segments

L-30 = Name of Levee Seepage Segment #1

6 = No. of Levee Seepage Cells in X-direction for Segment #1

\*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY

27,28 0 (1=cell inside subbasin boundary,0=outside)  
 27,27 0  
 26,26 0  
 26,25 0  
 26,24 0  
 26,23 0

0 = No. of Levee Seepage Cells in Y-direction for Segment #1

L31N.N = Name of Levee Seepage Segment #2

6 = No. of Levee Seepage Cells in X-direction for Segment #2

\*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY

26,22 0  
 26,21 0 (1=cell inside subbasin boundary,0=outside)  
 26,20 0  
 26,19 0  
 26,18 0  
 26,17 0

0 = No. of Levee Seepage Cells in Y-direction for Segment #2

L31N.S = Name of Levee Seepage Segment #3

4 = No. of Levee Seepage Cells in X-direction for Segment #3

\*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY

25,16 0 (1=cell inside subbasin boundary,0=outside)  
 24,15 0  
 24,14 0  
 24,13 0

0 = No. of Levee Seepage Cells in Y-direction for Segment #3

\*

\* STRUCTURE INFLOWS AND OUTFLOWS

6 = No. of Structure Inflows to Subbasin

\* STRUCTURE NAME

"/SFWMM/S32/FLOW//1DAY/SIMULATED/"  
 "/SFWMM/S31RG/FLOW//1DAY/SIMULATED/"  
 "/SFWMM/S31/FLOW//1DAY/SIMULATED/"  
 "/SFWMM/S337/FLOW//1DAY/SIMULATED/"  
 "/SFWMM/S334/FLOW//1DAY/SIMULATED/"  
 "/SFWMM/LSPC6/FLOW//1DAY/SIMULATED/"

15 = No. of Structure Outflows from Subbasin

\* STRUCTURE NAME

"/SFWMM/S28/FLOW//1DAY/SIMULATED/"  
 "/SFWMM/S27/FLOW//1DAY/SIMULATED/"  
 "/SFWMM/C6EQ/FLOW//1DAY/SIMULATED/"  
 "/SFWMM/G97/FLOW//1DAY/SIMULATED/"  
 "/SFWMM/S22/FLOW//1DAY/SIMULATED/"  
 "/SFWMM/S123/FLOW//1DAY/SIMULATED/"  
 "/SFWMM/S21/FLOW//1DAY/SIMULATED/"  
 "/SFWMM/S21A/FLOW//1DAY/SIMULATED/"  
 "/SFWMM/S20F/FLOW//1DAY/SIMULATED/"  
 "/SFWMM/S20G/FLOW//1DAY/SIMULATED/"

```

"/SFWMM/S20/FLOW//1DAY/SIMULATED/"
"/SFWMM/S174/FLOW//1DAY/SIMULATED/"
"/SFWMM/S197/FLOW//1DAY/SIMULATED/"
"/SFWMM/C9DRSQ/FLOW//1DAY/SIMULATED/"
"/SFWMM/C9DESQ/FLOW//1DAY/SIMULATED/"
*****
WCA_SYSTEM = Sub-area Name
*****
* SUBAREA BOUNDARY
23 = Southernmost Row No.
53 = Northernmost Row No.
* SUBAREA DEFINITION (Row #'s in descending order)
* ROW# MIN COL# MAX COL#
  53     30     30
  52     29     31
  51     29     33
  50     28     34
  49     28     34
  48     28     34
  47     28     34
  46     28     34
  45     27     34
  44     27     34
  43     26     33
  42     26     32
  41     16     32
  40     16     32
  39     16     32
  38     16     32
  37     16     32
  36     16     31
  35     16     27
  34     16     27
  33     16     27
  32     16     27
  31     16     27
  30     15     27
  29     16     27
  28     16     27
  27     16     27
  26     16     26
  25     16     26
  24     16     26
  23     16     26
*
* LEVEE SEEPAGE
7 = No. of Levee Seepage Segments
L-40 = Name of Levee Seepage Segment #1
11 = No. of Levee Seepage Cells in X-direction for Segment #1
*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY
  30,53     1 (1=cell inside subbasin boundary,0=outside)
  31,52     1
  33,51     1
  34,50     1
  34,49     1
  34,48     1
  34,47     1

```

34,46 1  
 34,45 1  
 34,44 1  
 33,43 1  
 1 = No. of Levee Seepage Cells in Y-direction for Segment #1  
 \*COORDINATES OF CELL IMMEDIATELY NORTH OF LEVEE SEEPAGE BOUNDARY  
 32,52 0 (1=cell inside subbasin boundary,0=outside)  
 L-36.N = Name of Levee Seepage Segment #2  
 4 = No. of Levee Seepage Cells in X-direction for Segment #2  
 \*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY  
 32,42 1 (1=cell inside subbasin boundary,0=outside)  
 32,41 1  
 32,40 1  
 32,39 1  
 0 = No. of Levee Seepage Cells in Y-direction for Segment #2  
 L-68A = Name of Levee Seepage Segment #3  
 4 = No. of Levee Seepage Cells in X-direction for Segment #3  
 \*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY  
 27,35 1 (1=cell inside subbasin boundary,0=outside)  
 27,34 1  
 27,33 1  
 27,32 1  
 0 = No. of Levee Seepage Cells in Y-direction for Segment #3  
 L-36(S) = Name of Levee Seepage Segment #4  
 2 = No. of Levee Seepage Cells in X-direction for Segment #4  
 \*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY  
 32,38 1 (1=cell inside subbasin boundary,0=outside)  
 32,37 1  
 0 = No. of Levee Seepage Cells in Y-direction for Segment #4  
 L35&35A = Name of Levee Seepage Segment #5  
 1 = No. of Levee Seepage Cells in X-direction for Segment #5  
 \*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY  
 31,36 1 (1=cell inside subbasin boundary,0=outside)  
 1 = No. of Levee Seepage Cells in Y-direction for Segment #5  
 \*COORDINATES OF CELL IMMEDIATELY NORTH OF LEVEE SEEPAGE BOUNDARY  
 29,36 1 (1=cell inside subbasin boundary,0=outside)  
 L30&33 = Name of Levee Seepage Segment #6  
 8 = No. of Levee Seepage Cells in X-direction for Segment #6  
 \*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY  
 27,31 1 (1=cell inside subbasin boundary,0=outside)  
 27,30 1  
 27,28 1  
 27,27 1  
 26,26 1  
 26,25 1  
 26,24 1  
 26,23 1  
 0 = No. of Levee Seepage Cells in Y-direction for Segment #6  
 L-29E = Name of Levee Seepage Segment #7  
 0 = No. of Levee Seepage Cells in X-direction for Segment #7  
 5 = No. of Levee Seepage Cells in Y-direction for Segment #7  
 \*COORDINATES OF CELL IMMEDIATELY NORTH OF LEVEE SEEPAGE BOUNDARY  
 22,23 1 (1=cell inside subbasin boundary,0=outside)  
 23,23 1  
 24,23 1  
 25,23 1  
 26,23 1

```

*
* STRUCTURE INFLOWS AND OUTFLOWS
15 = No. of Structure Inflows to Subbasin
* STRUCTURE NAME      INDEX
  "/SFWMM/S5A2SO/FLOW//1DAY/SIMULATED/"
  "/SFWMM/S6/FLOW//1DAY/SIMULATED/"
  "/SFWMM/L8TCA1/FLOW//1DAY/SIMULATED/"
  "/SFWMM/S7/FLOW//1DAY/SIMULATED/"
  "/SFWMM/S8/FLOW//1DAY/SIMULATED/"
  "/SFWMM/S150/FLOW//1DAY/SIMULATED/"
  "/SFWMM/S9/FLOW//1DAY/SIMULATED/"
  "/SFWMM/ACMERF/FLOW//1DAY/SIMULATED/"
  "/SFWMM/L28WQ/FLOW//1DAY/SIMULATED/"
  "/SFWMM/S140A/FLOW//1DAY/SIMULATED/"
  "/SFWMM/G155/FLOW//1DAY/BASE/"
  "/SFWMM/G204/FLOW//1DAY/SIMULATED/"
  "/SFWMM/G205/FLOW//1DAY/SIMULATED/"
  "/SFWMM/G206/FLOW//1DAY/SIMULATED/"
  "/SFWMM/G123/FLOW//1DAY/SIMULATED/"

```

```

18 = No. of Structure Outflows from Subbasin
* STRUCTURE NAME      INDEX
  "/SFWMM/S5A2NO/FLOW//1DAY/SIMULATED/"
  "/SFWMM/S39/FLOW//1DAY/SIMULATED/"
  "/SFWMM/LSPL33/FLOW//1DAY/SIMULATED/"
  "/SFWMM/LSPC6/FLOW//1DAY/SIMULATED/"
  "/SFWMM/LWDD/FLOW//1DAY/SIMULATED/"
  "/SFWMM/S38/FLOW//1DAY/SIMULATED/"
  "/SFWMM/S34/FLOW//1DAY/SIMULATED/"
  "/SFWMM/S31RG/FLOW//1DAY/SIMULATED/"
  "/SFWMM/S31/FLOW//1DAY/SIMULATED/"
  "/SFWMM/S337/FLOW//1DAY/SIMULATED/"
  "/SFWMM/S333/FLOW//1DAY/SIMULATED/"
  "/SFWMM/ACMEWS/FLOW//1DAY/SIMULATED/"
  "/SFWMM/S12A/FLOW//1DAY/SIMULATED/"
  "/SFWMM/S12B/FLOW//1DAY/SIMULATED/"
  "/SFWMM/S12C/FLOW//1DAY/SIMULATED/"
  "/SFWMM/S12D/FLOW//1DAY/SIMULATED/"
  "/SFWMM/S343/FLOW//1DAY/SIMULATED/"
  "/SFWMM/S344/FLOW//1DAY/SIMULATED/"

```

\*\*\*\*\*

LEC\_DEVELOPED\_AREA = Sub-area Name

\*\*\*\*\*

\* SUBAREA BOUNDARY

5 = Southernmost Row No.

65 = Northernmost Row No.

\* SUBAREA DEFINITION (Row #'s in descending order)

ROW#	MIN COL#	MAX COL#
65	36	38
64	36	39
63	24	39
62	25	39
61	26	39
60	27	40
59	29	40
58	30	40
57	30	40
56	31	40

55	31	40
54	30	40
53	31	40
52	32	40
51	34	40
50	35	40
49	35	40
48	35	40
47	35	40
46	35	39
45	35	39
44	35	39
43	34	39
42	33	39
41	33	39
40	33	39
39	33	39
38	33	39
37	33	38
36	32	38
35	28	38
34	28	38
33	28	38
32	28	38
31	28	38
30	28	38
29	28	38
28	28	38
27	28	37
26	27	36
25	27	36
24	27	36
23	27	36
22	27	35
21	27	34
20	27	34
19	27	33
18	27	33
17	27	32
16	26	32
15	25	32
14	25	31
13	25	31
12	25	31
11	24	31
10	25	31
9	25	31
8	25	31
7	25	30
6	25	29
5	28	28

\*

\* LEVEE SEEPAGE

10 = No. of Levee Seepage Segments

L-40 = Name of Levee Seepage Segment #1

11 = No. of Levee Seepage Cells in X-direction for Segment #1

\*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY



30,53 0 (1=cell inside subbasin boundary,0=outside)  
 31,52 0  
 33,51 0  
 34,50 0  
 34,49 0  
 34,48 0  
 34,47 0  
 34,46 0  
 34,45 0  
 34,44 0  
 33,43 0

1 = No. of Levee Seepage Cells in Y-direction for Segment #1  
 \*COORDINATES OF CELL IMMEDIATELY NORTH OF LEVEE SEEPAGE BOUNDARY  
 32,52 1 (1=cell inside subbasin boundary,0=outside)

L36.N1 = Name of Levee Seepage Segment #2  
 1 = No. of Levee Seepage Cells in X-direction for Segment #2  
 \*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY  
 32,42 0

0 = No. of Levee Seepage Cells in Y-direction for Segment #2  
 L36.N2 = Name of Levee Seepage Segment #3  
 3 = No. of Levee Seepage Cells in X-direction for Segment #3  
 \*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY  
 32,41 0 (1=cell inside subbasin boundary,0=outside)  
 32,40 0  
 32,39 0

0 = No. of Levee Seepage Cells in Y-direction for Segment #3  
 L-36(S) = Name of Levee Seepage Segment #4  
 2 = No. of Levee Seepage Cells in X-direction for Segment #4  
 \*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY  
 32,38 0 (1=cell inside subbasin boundary,0=outside)  
 32,37 0

0 = No. of Levee Seepage Cells in Y-direction for Segment #4  
 L35&35A = Name of Levee Seepage Segment # 5  
 1 = No. of Levee Seepage Cells in X-direction for Segment #5  
 \*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY  
 31,36 0 (1=cell inside subbasin boundary,0=outside)

1 = No. of Levee Seepage Cells in Y-direction for Segment #5  
 \*COORDINATES OF CELL IMMEDIATELY NORTH OF LEVEE SEEPAGE BOUNDARY  
 29,36 0 (1=cell inside subbasin boundary,0=outside)

L-68A = Name of Levee Seepage Segment #6  
 4 = No. of Levee Seepage Cells in X-direction for Segment #6  
 \*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY  
 27,35 0 (1=cell inside subbasin boundary,0=outside)  
 27,34 0  
 27,33 0  
 27,32 0

0 = No. of Levee Seepage Cells in Y-direction for Segment #6  
 L-33 = Name of Levee Seepage Segment # 7  
 2 = No. of Levee Seepage Cells in X-direction for Segment #7  
 \*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY  
 27,31 0 (1=cell inside subbasin boundary,0=outside)  
 27,30 0

0 = No. of Levee Seepage Cells in Y-direction for Segment #7  
 L-30 = Name of Levee Seepage Segment #8  
 6 = No. of Levee Seepage Cells in X-direction for Segment #8  
 \*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY  
 27,28 0 (1=cell inside subbasin boundary,0=outside)

```

27,27    0
26,26    0
26,25    0
26,24    0
26,23    0
0 = No. of Levee Seepage Cells in Y-direction for Segment #8
L31N.N = Name of Levee Seepage Segment #9
6 = No. of Levee Seepage Cells in X-direction for Segment #9
*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY
26,22    0
26,21    0 (1=cell inside subbasin boundary,0=outside)
26,20    0
26,19    0
26,18    0
26,17    0
0 = No. of Levee Seepage Cells in Y-direction for Segment #9
L31N.S = Name of Levee Seepage Segment #10
4 = No. of Levee Seepage Cells in X-direction for Segment #10
*COORDINATES OF CELL IMMEDIATELY WEST OF LEVEE SEEPAGE BOUNDARY
25,16    0 (1=cell inside subbasin boundary,0=outside)
24,15    0
24,14    0
24,13    0
0 = No. of Levee Seepage Cells in Y-direction for Segment #10
*
* STRUCTURE INFLOWS AND OUTFLOWS
16 = No. of Structure Inflows to Subbasin
* STRUCTURE NAME
"/SFWMM/S5A1/FLOW//1DAY/SIMULATED/"
"/SFWMM/L8C51W/FLOW//1DAY/SIMULATED/"
"/SFWMM/LWDD/FLOW//1DAY/SIMULATED/"
"/SFWMM/LSPL33/FLOW//1DAY/SIMULATED/"
"/SFWMM/LSPC6/FLOW//1DAY/SIMULATED/"
"/SFWMM/S39/FLOW//1DAY/SIMULATED/"
"/SFWMM/S38/FLOW//1DAY/SIMULATED/"
"/SFWMM/S34/FLOW//1DAY/SIMULATED/"
"/SFWMM/S31RG/FLOW//1DAY/SIMULATED/"
"/SFWMM/S31/FLOW//1DAY/SIMULATED/"
"/SFWMM/S337/FLOW//1DAY/SIMULATED/"
"/SFWMM/S334/FLOW//1DAY/SIMULATED/"
"/SFWMM/ACMEWS/FLOW//1DAY/SIMULATED/"
"/SFWMM/S2TMCL/FLOW//1DAY/SIMULATED/"
"/SFWMM/S5A3SO/FLOW//1DAY/SIMULATED/"
"/SFWMM/S5A2NO/FLOW//1DAY/SIMULATED/"
36 = No. of Structure Outflows from Subbasin
* STRUCTURE NAME      INDEX
"/SFWMM/S5A2SO/FLOW//1DAY/SIMULATED/"
"/SFWMM/S5A3NO/FLOW//1DAY/SIMULATED/"
"/SFWMM/ACMERF/FLOW//1DAY/SIMULATED/"
"/SFWMM/LOXRQ/FLOW//1DAY/SIMULATED/"
"/SFWMM/S155/FLOW//1DAY/SIMULATED/"
"/SFWMM/PBDRQ/FLOW//1DAY/SIMULATED/"
"/SFWMM/S41/FLOW//1DAY/SIMULATED/"
"/SFWMM/S40/FLOW//1DAY/SIMULATED/"
"/SFWMM/RVBDRQ/FLOW//1DAY/SIMULATED/"
"/SFWMM/RESTL8/FLOW//1DAY/SIMULATED/"
"/SFWMM/RESL80/FLOW//1DAY/SIMULATED/"

```

"/SFWMM/BKMCL8/FLOW//1DAY/SIMULATED/"  
"/SFWMM/HLSBEQ/FLOW//1DAY/SIMULATED/"  
"/SFWMM/DPRESO/FLOW//1DAY/SIMULATED/"  
"/SFWMM/CORBT1/FLOW//1DAY/SIMULATED/"  
"/SFWMM/CORBT2/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S9/FLOW//1DAY/SIMULATED/"  
"/SFWMM/G123/FLOW//1DAY/SIMULATED/"  
"/SFWMM/G57DNQ/FLOW//1DAY/SIMULATED/"  
"/SFWMM/QC13E/FLOW//1DAY/SIMULATED/"  
"/SFWMM/Q2C57/FLOW//1DAY/SIMULATED/"  
"/SFWMM/C10Q/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S29DNQ/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S28/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S27/FLOW//1DAY/SIMULATED/"  
"/SFWMM/C6EQ/FLOW//1DAY/SIMULATED/"  
"/SFWMM/G97/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S22/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S123/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S21/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S21A/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S20F/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S20G/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S20/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S174/FLOW//1DAY/SIMULATED/"  
"/SFWMM/S197/FLOW//1DAY/SIMULATED/"

/vol/hsm/bin/solaris/sfwmm.scr

```
#!/bin/csh -f

# SCCSID = "@(#)sfwmm.scr      1.19 03/18/02 SFWMD, HSM, Planning Department"

# c-shell script for RUNNING SFWMM & POST PROCESSING IT'S RUNS
# Author: Kris Krishnan, HSM Division, Planning Dept

# Usage: sfwmm.scr [< response file]

#####
# DEFINE ALL VARIABLES USED IN THIS SCRIPT
# THAT MAY NEED TO BE CHANGED
#####

# Set the display.  Needed for arc/info
setenv DISPLAY `who am i | cut -d'(' -f2 | cut -d')' -f1 -s`:0

# set the hsm binary dir
set hsmBin = /vol/hsm/bin/solaris

# set the hsm gis directory
if ( $?HSMGIS == 0) setenv HSMGIS '/vol/hsm_data2/hsmgis'

# set the right NSM Output directory
set nsmRun = /vol/hsm/data/nsm/output/nsm45

# set the start and end dates for MFL post-processing
# Note: date has to be in yyyyymmdd format
set SFWMM_START_YR = 1965      # needs to be yyyy format
set SFWMM_START_MONTH = 01    # needs to be mm format
set SFWMM_START_DAY = 01      # needs to be dd format
set SFWMM_END_YR = 1995       # needs to be yyyy format
set SFWMM_END_MONTH = 12     # needs to be mm format
set SFWMM_END_DAY = 31       # needs to be dd format
set input_path = $cwd

# end define variables
#####

clear
echo "C-Shell script for RUNNING SFWMM & POST PROCESSING IT'S RUNS"
echo "Usage: sfwmm.scr [ < response_filename ]"
echo "Response file, used in a batch run, contains all the responses."
echo "For a sample response file, look at file"
echo "sample_response_for_sfwmm.scr"

echo ""
echo "The following directories should be created before running this in batch
mode:"
echo ""
echo "****" "'ECON'", "'WBUD'", "'CONVEY'", "'C43C44'"
echo "****" "'MAPS/OVFLOW'", "'MAPS/HYDROPERIOD'", "'MAPS/POND'",
"'MAPS/RUNS'", "'MAPS/HPIMP'"

```

```

echo "****" "'MAPS/STAGE'", "'MAPS/TRIGGERS'", "'OTHER_WBUD/NOR_STO_BUD'"
and "'OTHER_WBUD/ASR_BUD'"
echo "**** These are sub-directories in the filespace where the run's"
echo "**** output is going to be written. Also, create the control files for,"
echo "**** ECON" '(LEC & EAA)', WBUD , HYDROPERIOD, PONDING, HPIMP, C43C44
NOR_STO_BUD and ASR_BUD"
echo "**** in their respective directories."
echo ""
sleep 4

# prompt name of SFWMM executable
sfwmmmin:
echo "Type the COMPLETE path name of the SFWMM executable with the config file
as argument."
echo "(default config file:" "'ALTWMM'" "in the CWD).)"
echo "Eg. /home/harney/rsantee/SFWMM/SRC_REV/wmm.exe
/home/harney/rsantee/SFWMM/DATA/ALTWMM"
set sfwm = $<
set sfwmm = ($sfwm)

echo ""
#chk for the existence of exe file just typed
if (! -e $sfwmm[1]) then
    echo $sfwmm[1] executable not found. Type again.
    goto sfwmmmin
endif

# check if there is argument with sfwmm executable or default ALTWMM file
if( $#sfwmm == 1 && ! -e ALTWMM) then
    echo ""
    echo "Neither a config file was specified as an argument to $sfwmm[1]"
    echo nor does file "'ALTWMM'" exist in the CWD. Try again.
    echo "***** ABORTING...."
    echo ""
    exit -1
endif

# check for the existence of the argument (in a roundabout way
# 'coz of problems with $sfwmm[2])
if ($#sfwmm == 2 ) then
    goto test
else
    goto skiptest
endif

test:
if(! -e $sfwmm[2]) then
    echo ""
    echo "SFWMM config file $sfwmm[2] does not exist. Try again."
    echo "***** ABORTING...."
    echo ""
    exit -1
endif

skiptest:
#get the RUN NAME from the argument
if ($#sfwmm == 2) then

```

```

    set runtitt = `head -1 $sfwmm[2]`
# echo $runtitt
else
#from the default ALTWMM file
    set runtitt = `head -1 ALTWMM`
endif

#get the RUN Filespace where run's OUTPUT is to be written
runout:
echo ""
echo Type the COMPLETE Path to the directory where output of this \
    run is to be written. Eg. /home/kissimnee/nkrishna/tmp
set runpath = $<
echo ""
#chk for the existence of filespace just typed
if (! -e $runpath) then
    echo Directory $runpath doesnt exist. Type again.
    goto runout
endif

# Create subdir runpath/input to store all input files used for the run.
if (-e $runpath/input) /bin/rm -R $runpath/input
mkdir $runpath/input

# Create a file that contains the basic information usually contained in the
rundone file just in case the rundone file is not created due to post-processing
failure.
echo Run: $runtitt >! $runpath/input/input_info
echo Using: $sfwmm >> $runpath/input/input_info
echo Input directory: $cwd >> $runpath/input/input_info

# tar and zip the source code components
if (-e $runpath/src_code) /bin/rm -R $runpath/src_code
mkdir $runpath/src_code
foreach file (F f C c inc)
/bin/cp -p $sfwmm[1]:h/*. $file $runpath/src_code
end
/bin/cp -p $sfwmm[1]:h/Makefile $runpath/src_code
/vol/bin/gzip $runpath/src_code/*
echo "Backup of source code directory for" $sfwmm[1] >
$runpath/src_code/src_code_tar.log
date >> $runpath/src_code/src_code_tar.log
tar -cvf $runpath/src_code/src_code.tar $runpath/src_code/*.gz >>
$runpath/src_code/src_code_tar.log
/bin/rm $runpath/src_code/*.gz
/vol/bin/gzip $runpath/src_code/src_code.tar

# Copy the ALTWMM file
/bin/cp -p $sfwmm[2] $runpath/input

# scan the input file names in ALTWMM file
if (-e $runpath/input/non_local_files_list) /bin/rm
$runpath/input/non_local_files_list
set allfile = `nawk '{if (NR > 2) print $NF}' $sfwmm[2]`
@ i=1
foreach infile ($allfile)

```

```

if(-e $infile:t) then
# the file is local, then copy it to $runpath/input
/bin/cp -p $infile $runpath/input/
else
# the file is not local, then long list it in file non_local_files_list
ls -ltr -CF $infile >> $runpath/input/non_local_files_list
endif
@ i++
end
/vol/bin/gzip $runpath/input/*

# prompt name of std output file
echo "Type JUST the file name where you want standard output to be saved."
set stdout = $<
echo ""

#chk for the existence of the file just typed
if (-e $stdout) then
    /bin/rm $stdout
endif

# prompt name of error output file
#echo "Type the file name where you want diagnostic output, if any, to be
saved."
#set errout = $<
#echo ""

# ask if Existing *.bin & tape* *.dat & *.dss* files are to be deleted before
echo "Do you want all the EXISTING *.bin, tape* and str2x2.dss output files"
echo "from the previous run to be deleted before starting this Run?"
echo "Type Y for Yes and N for No"
set yes_no = $<

if ($yes_no == "Y" || $yes_no == "y") then
    /bin/rm $runpath/*.bin
    /bin/rm $runpath/tape*
    /bin/rm $runpath/daily_asr_bud.dat $runpath/testc43.out $runpath/testc44.out
    /bin/rm $runpath/asrc43.bud $runpath/asrc44.bud
    /bin/rm $runpath/*.dat
    /bin/rm $runpath/*.dss*
    /bin/rm $runpath/CONVEY/convey.dss*
endif

#delete the rundone file
if (-e $runpath/rundone) then
    /bin/rm $runpath/rundone
endif

# Prompt for the Mailing list
echo ""
echo "Type the user names of people besides y'self you want to send email to."
echo "Example: nkrishna cal lbrion rsantee ptrimble"
set ml = $<
set mlist = ( $user $ml )

@ i=1

```

```

while ( $i <= $#mlist )
    set mlist[$i] = $mlist[$i]"@sfwmd.gov"
    @ i++
end

# get the control filename for the ECON post processor program
echo ""
echo "Type the control filename for the LEC ECON post processor program."
set econin = $<

# get the control filename for the EAA ECON post processor program
echo ""
echo "Type the control filename for the EAA econ post processor program."
set eaain = $<

# get the control filename for the water budget program
echo ""
echo "Type the control filename for the water budget program."
set wbudin = $<

# get the control filename for the c43c44 budget program
echo ""
echo "Type the control filename for the C43C44 budget program."
set c43c44in = $<

# get the control filename for the HYDROPERIOD post processor program
echo ""
echo "Type the control filename for the HYDROPERIOD post processor program."
set hpin = $<

# get the control filename for the Ponding post processor program
echo ""
echo "Type the control filename for the PONDING post processor program."
set pondin = $<

# get the control filename for the HPIMP post processor program
echo ""
echo "Type the control filename for the HPIMP post processor program."
set hpimpin = $<

# get the control filename for the North Storage Water Budget post processor
program
echo ""
echo "If North Storage exists for this run, Type the control filename for the
noresbud budget post processor program."
set norstin = $<

# get the control filename for the ASR Bubble Budget post processor program
echo ""
echo "If ASR exists for this run, Type the control filename for the ASR Bubble
budget post processor program."
set asrin = $<

mvetbin:
# Prompt for the name of the directory where the file est_etiu_unrestr.bin will
be moved.

```



```

# This file is optionally created by the SFWMM and is used by the LEC Economics
post-processor.
# No entry signals the file is not being created.
echo "Type full path directory in which file est_etiu_unrestr.bin will be saved.
No entry signals the file is not being created."
set mvetbin_dir = $<

if ( $mvetbin_dir != "" )then
  if !(-e $mvetbin_dir) then
    echo "Directory \"$mvetbin_dir\" does not exist. Try again"
    goto mvetbin
  endif
endif

#check for the existence of all the files just typed
if (! -e $runpath/ECON/$econin) echo "ECON/$econin file doesnt exist." >
/tmp/addmail.$$
if (! -e $runpath/ECON/$eaain) echo "ECON/$eaain file doesnt exist." >>
/tmp/addmail.$$
if (! -e $runpath/WBUD/$wbudin) echo "WBUD/$wbudin file doesnt exist." >>
/tmp/addmail.$$
if (! -e $runpath/C43C44/$c43c44in) echo "C43C44/$c43c44in file doesnt exist."
>> /tmp/addmail.$$
if (! -e $runpath/MAPS/HYDROPERIOD/$hpin) echo "MAPS/HYDROPERIOD/$hpin file
doesnt exist." >> /tmp/addmail.$$
if (! -e $runpath/MAPS/POND/$pondin) echo "MAPS/POND/$pondin file doesnt
exist." >> /tmp/addmail.$$
if (! -e $runpath/MAPS/HPIMP/$hpimpin) echo "MAPS/HPIMP/$hpimpin file doesnt
exist." >> /tmp/addmail.$$
if (! -e $runpath/MAPS/STAGE) echo "MAPS/STAGE directory doesnt exist." >>
/tmp/addmail.$$
if (! -e $runpath/MAPS/TRIGGERS) echo "MAPS/TRIGGERS directory doesnt exist."
>> /tmp/addmail.$$
if (! -e $runpath/MAPS/OVFLOW) echo "MAPS/OVFLOW directory doesnt exist." >>
/tmp/addmail.$$
if(($norstin != "") && (! -e $runpath/OTHER_WBUD/NOR_STO_BUD/$norstin)) then
  echo "OTHER_WBUD/NOR_STO_BUD/$norstin file doesnt exist." >>
/tmp/addmail.$$
endif
if(($asrin != "") && (! -e $runpath/OTHER_WBUD/ASR_BUD/$asrin)) then
  echo "OTHER_WBUD/ASR_BUD/$asrin file doesnt exist." >> /tmp/addmail.$$
endif
set wcount = `wc /tmp/addmail.$$ | awk '{print $1}' `
if ( $wcount != 0 ) echo "PLEASE create these dir/files before the WMM run
completes." >> /tmp/addmail.$$

#send mail to all in the mailing list that sfwmm program has started
echo '$runtit' run has been initiated on\
`date` in the filespace\
'$runpath'. The CPU being used is `hostname`. `cat /tmp/addmail.$$`\
| mailx -r $user"@sfwmd.gov" -s "$runtit run initiated on `hostname`" $mlist
#
# Input directory
#
set inp_dir = `pwd`
#

```

```

# Number of files to be open simulateneously by the programs
#
limit descriptors 256
#run the program, redirecting std output to stdout & error output to errout
#($sfwmm >! $stdout) >&! $errout
#run the program, redirecting std output & eeroutput to stdout
# first setenv SFWMMDAT to the runpath
setenv SFWMMDAT $runpath
# echo $SFWMMDAT
# $sfwmm >&! $runpath/$stdout
# pipe the std out & err to a file & screen
# get the execution and elapsed time
clear
echo "Executing sfwmm program....."
/usr/bin/time -p $sfwmm |& tee $runpath/$stdout

echo ""
echo "Finished executing sfwmm program."
echo ""

#grep for the string "real user sys" in the stdout and pick up elapsed time
#and execution (users process CPU) time. then convert to hours and min
#set time_stg = `grep "real" $runpath/$stdout | grep "user" | grep "sys" | awk
'`{print $1,$3}`
set time_stg_real = `grep "real" $runpath/$stdout | awk '{print $2}`
set time_stg_user = `grep "user" $runpath/$stdout | awk '{print $2}`
set elapse = `echo $time_stg_real | awk '{hr=sprintf("%d", $1/3600);
min=( $1/3600-hr)*60; printf("%d %.0f\n", hr,min)}' `
set exe = `echo $time_stg_user | awk '{hr=sprintf("%d", $1/3600); min=( $1/3600-
hr)*60; printf("%d %.0f\n", hr,min)}' `
echo "Execution time [Hr:min]" $exe[1]':'$exe[2]
echo "Elapsed time [Hr:min] " $elapse[1]':'$elapse[2]

#grep for the string "terminated correctly" in the stdout
set string = `grep -i "terminated correctly" $runpath/$stdout`

#Error messages, if any, has been saved in the file '$errout'`.

#send mail to all in the mailing list that sfwmm program completed
#also delete the stdout file
if ($#string == 2) then
echo '$runtit' run terminated correctly on\
`date`. The output is in the filespace\
'$runpath'. The CPU used `hostname`.
Execution time '[Hr:min] =' $exe[1]':'$exe[2].
Elapsed time '[Hr:min] =' $elapse[1]':'$elapse[2].
Post Processing "(ECON, WBUD, CONVEY, HYDROPERIOD, PONDING, TRIGGER, C43C44,
NOR_STOR & ASR)" has just been initiated.
| mailx -r $user"@sfwmd.gov" -s "SUCCESSFUL - $runtit run on `hostname`"
$mlist
/bin/rm $runpath/$stdout
else
echo '$runtit' run terminated INCORRECTLY on\
`date`. The "(incomplete?)" output is in the filespace\
'$runpath'. The CPU used `hostname`.
Execution time '[Hr:min] =' $exe[1]':'$exe[2].
Elapsed time '[Hr:min] =' $elapse[1]':'$elapse[2].

```

```

    All standard output along with 'error' messages have been saved in file
    '$stdout'.'.
    No POST PROCESSING HAS BEEN INITIATED.\
    | mailx -r $user"@sfwmd.gov" -s "UNSUCCESSFUL - $runtit run on `hostname`"
$mlist
endif

#
# If required, move file required for LEC ECONOMICS post processing
(est_etiu_unrestr.bin)
#
if ($mvetbin_dir != "") then
    if !(-e $runpath/est_etiu_unrestr.bin) then
        echo "File est_etiu_unrestr.bin was requested to be moved to directory
"$mvetbin_dir\
        ". However, the file was not created by the run." > /tmp/mvetbin.$$
    else
        /bin/mv $runpath/est_etiu_unrestr.bin $mvetbin_dir/est_etiu_unrestr.bin
        echo "File est_etiu_unrestr.bin was moved to directory "$mvetbin_dir". " >
/tmp/mvetbin.$$
    endif
else
    if (-e $runpath/est_etiu_unrestr.bin) then
        echo "File est_etiu_unrestr.bin was created by the run but it will not be
moved." > /tmp/mvetbin.$$
    else
        echo "No file est_etiu_unrestr.bin was created by this run." >
/tmp/mvetbin.$$
    endif
endif
/bin/cp /vol/hsm/scripts/solaris/sfwmm-pm-
graphics/control_files/con_files/sfwmmV4.4_file_definitions
$runpath/sfwmmV4.4_file_definitions
#start post processing
if ($#string == 2) then
    chdir $runpath/ECON
# echo $PWD
#first the LEC ECON post processor
# save the screen echos
echo "Performing LEC econ_post processing..."
echo $econin | $hsmBin/econ_post >&! econ_post.echo
#grep for the string "terminated correctly" in the echo file
set econstg = `grep -i "terminated correctly" econ_post.echo`
#Now the EAA ECON post processor
echo "Performing EAA econ_post processing..."
echo $eaain | $hsmBin/ea_econ >&! eaa_econ.echo
#grep for the string "terminated correctly" in the echo file
set eaastg = `grep -i "terminated correctly" eaa_econ.echo`
#Run LOSA SSM post processor
if(-e losassm.cf)/bin/rm losassm.cf
echo '01' > losassm.cf
echo '"LOSA"' >> losassm.cf
echo '01' >> losassm.cf
echo '$runtit' >> losassm.cf
echo '"../losa_dmnd_runff_supply.dat"' >> losassm.cf
if(-e file.inp)/bin/rm file.inp

```

```

    echo 'losassm.cf' > file.inp
    echo "Performing LOSA SSM post processing..."
    $hsmBin/losassm < file.inp >&! losassm.echo
    /bin/rm file.inp
#grep for the string "terminated correctly" in the echo file
set losastg = `grep -i "terminated correctly" losassm.echo`
#next the Water Budget post processor
chdir $runpath/WBUD
# echo $PWD
# NOT ABLE TO SAVE ERROR MESSAGES, if any - BOMBS OUT
echo "Performing WBUD post-processing..."
echo $wbudin | $hsmBin/wmmwbud >! wbud.echo
#grep for the string "terminated correctly" in the echo file
set wbudstg = `grep -i "terminated correctly" wbud.echo`

#create a dss catalog for str2x2.dss file in the output directory
chdir $runpath
echo Creating dss catalog for str2x2.dss file
$hsmBin/catDSS -q str2x2.dss
echo Finished dss catalog for str2x2.dss file

#run gr_summary for daily stages
chdir $runpath
if !(-e ./MAPS) mkdir ./MAPS
chdir MAPS
if !(-e ./STAGE) mkdir ./STAGE
chdir STAGE
echo "Running gr_summary for daily stages..."
$hsmBin/gr_summary -a -o ./dsml $runpath/daily_stg_minus_lsel.bin
chdir $runpath

#create a rundone file in the output directory if wbud & econ post processing
was completed.
#this file will be used by the perf.measure post processor
#echo $#wbudstg $#c43c44stg $#econstg
if ($#wbudstg == 2 && $#econstg == 2 && $#eaastg == 2) then
    chdir $runpath
    echo Run: $runtit >! rundone
    echo Completed on: `date` >> rundone
    echo Using: $sfwmm >> rundone
    echo Input directory: $inp_dir >> rundone
    echo "PLEASE, DO NOT DELETE THIS FILE. THIS FILE IS REQUIRED BY THE\
PERFORMANCE MEASURE GRAPHICS POST-PROCESSOR AND TO DOCUMENT THIS RUN." >>
rundone
    echo rundone file HAS been created. >! /tmp/rundone.$$
else
    echo rundone file was NOT created. >! /tmp/rundone.$$
endif

#run gr_summary for overland flow
chdir $runpath
chdir MAPS
if !(-e ./OVFLOW) mkdir ./OVFLOW
chdir OVFLOW
echo "Running gr_summary for overland flow..."
$hsmBin/gr_summary -a -d2 -m -o ovflow $runpath/surface_flow.bin
chdir $runpath

```

```

#next the HYDROPERIOD post processor
#chk for the existence of MAPS/HYDROPERIOD dir
if (-e $runpath/MAPS/HYDROPERIOD && -e $runpath/daily_stg_minus_lsel.bin) then
  chdir $runpath/MAPS/HYDROPERIOD
  echo $PWD
  echo "Performing HYDROPERIOD post-processing..."
  # first create hydroperiod bin file from the daily bin file
  # its assumed that lake is part of the daily bin file
  $hsmBin/hydroperiod -o hp_z0.001.bin $runpath/daily_stg_minus_lsel.bin

  # create a soft link to the nsm hydroperiod.bin file
  ln -s $nsmRun/MAPS/HYDROPERIOD/hyp_nsm45_cut4_wmm_z0.001.bin
nsm45hp_z0.001.bin

  # run grid_math to get the difference between nsm45hp_z0.001.bin and
hp_z0.001.bin
  # use the control file in this directory
  $hsmBin/grid_math < $hpin > hpstdout.$$

  # finally run the gridsumalt2 to get nsm-this_run hydroperiod difference
statistics
  # get the hp diff file name from line 5 of control file
  set hpdiff_bin = `sed -n 5p $hpin | sed -e s/\"// | sed -e s/\"//`
  $hsmBin/gridsumalt2 -o hpdiff_z0.001_stats.bin $hpdiff_bin

  # run the gridsumalt2 to get this WMM run hydroperiod statistics
  $hsmBin/gridsumalt2 -o hp_z0.001_stats.bin hp_z0.001.bin

  #create mean WMM hydroperiod roco file
#   $hsmBin/grid_shot -n6 -d2 hp_z0.001_stats.bin | $hsmBin/g2roco >!
hp_z0.001_stats_mean.roco
  $hsmBin/grid_shot -n6 -d2 hp_z0.001_stats.bin | $hsmBin/g2roco >!
hp_z0.001_annave.roco

  #create mean NSM-WMM hydroperiod difference roco file
#   $hsmBin/grid_shot -n6 -d2 hpdiff_z0.001_stats.bin | $hsmBin/g2roco >!
hpdiff_z0.001_stats_mean.roco
  $hsmBin/grid_shot -n6 -d2 hpdiff_z0.001_stats.bin | $hsmBin/g2roco >!
hpdiff_z0.001_annave.roco

  #create WMM hydroperiod roco file for 1989
  $hsmBin/grid_shot -n25 -d2 hp_z0.001.bin | $hsmBin/g2roco >!
hp_z0.001_1989.roco

  #create WMM hydroperiod roco file for 1991
  $hsmBin/grid_shot -n27 -d2 hp_z0.001.bin | $hsmBin/g2roco >!
hp_z0.001_1991.roco

  #create WMM hydroperiod roco file for 1995
  $hsmBin/grid_shot -n31 -d2 hp_z0.001.bin | $hsmBin/g2roco >!
hp_z0.001_1995.roco

  #create NSM-WMM hydroperiod difference roco file for 1989
  $hsmBin/grid_shot -n25 -d2 $hpdiff_bin | $hsmBin/g2roco >!
hpdiff_z0.001_1989.roco

```

```

#create NSM-WMM hydroperiod difference roco file for 1991
$hsmBin/grid_shot -n27 -d2 $hpdiff_bin | $hsmBin/g2roco >!
hpdiff_z0.001_1991.roco

#create NSM-WMM hydroperiod difference roco file for 1995
$hsmBin/grid_shot -n31 -d2 $hpdiff_bin | $hsmBin/g2roco >!
hpdiff_z0.001_1995.roco

/bin/rm nsm45hp_z0.001.bin
endif

#see if HYDROP completed successfully
set hpstg = `grep -i "completed successfully" hpstdout.$$`
/bin/rm hpstdout.$$

# next the PONDING post processor
#chk for the existence of MAPS/POND dir
if (-e $runpath/MAPS/POND && -e $runpath/daily_stg_minus_lsel.bin) then
  chdir $runpath/MAPS/POND
  #echo $PWD
  echo "Performing PONDING post-processing..."
  # Using gr_summary create daily_ponding all statistics bin files using only
  # positive values from the WMM daily bin file
  # its assumed that lake is part of the daily bin file
  $hsmBin/gr_summary -a -z -o day_ponding_noneg
$runpath/daily_stg_minus_lsel.bin
  # create a soft link to the nsm ponding.bin file
  ln -s $nsmRun/MAPS/POND/day_ponding_noneg_cut4wmm.annave
nsm45_day_ponding_noneg_cut4wmm.annave
  # run gird_math to get the difference between
nsm45_day_ponding_noneg_cut4wmm.annave &
  # day_ponding_noneg.annave files (creates pondiff.annave file)
  $hsmBin/grid_math < $pondin > pondout.$$
  /bin/rm nsm45_day_ponding_noneg_cut4wmm.annave

  # create roco file of WMM annave ponding
  $hsmBin/grid_shot -d2 day_ponding_noneg.annave | $hsmBin/g2roco >!
day_ponding_noneg_annave.roco

  # create roco file of WMM average May ponding
  $hsmBin/grid_shot -n5 -d2 day_ponding_noneg.moave | $hsmBin/g2roco >!
day_ponding_noneg_mayave.roco

  # create roco file of WMM average October ponding
  $hsmBin/grid_shot -n10 -d2 day_ponding_noneg.moave | $hsmBin/g2roco >!
day_ponding_noneg_octave.roco

  # create roco file of NSM-WMM annave ponding difference
  $hsmBin/grid_shot -d2 pondiff.annave | $hsmBin/g2roco >!
pondiff_annave.roco

  echo "PONDING post-processing done."
endif

#see if POND file was completed successfully
set pondstg = `grep -i "completed successfully" pondout.$$`

```

```

/bin/rm pondout.$$

#next the RUNS/Min flows and Levels post processor
#chk for the existence of MAPS/RUNS dir
if (! -e $runpath/MAPS/RUNS) then
    chdir $runpath
    mkdir MAPS/RUNS
endif
if (-e $runpath/daily_stg_minus_lsel.bin) then
    chdir $runpath/MAPS/RUNS
    echo $PWD
    echo "Performing MFL RUNS post-processing..."
    # first create the three mfl bin files (for 30,75 and 150 days duration,
from the daily bin file
    # its assumed that lake is part of the daily bin file
    $hsmBin/hydroperiod -o mfl_r151_n1_B-3.bin -b0.0 -r151,9999 -
y${SFWMM_START_YR}${SFWMM_START_MONTH}${SFWMM_START_DAY},${SFWMM_END_YR}${SFWMM_
END_MONTH}${SFWMM_END_DAY} -n1 -B-3.0 $runpath/daily_stg_minus_lsel.bin
    $hsmBin/hydroperiod -o mfl_r76_n1_B-2.bin -b0.0 -r76,9999 -
y${SFWMM_START_YR}${SFWMM_START_MONTH}${SFWMM_START_DAY},${SFWMM_END_YR}${SFWMM_
END_MONTH}${SFWMM_END_DAY} -n1 -B-2.0 $runpath/daily_stg_minus_lsel.bin
    $hsmBin/hydroperiod -o mfl_r31_n1_B-1.bin -b0.0 -r31,9999 -
y${SFWMM_START_YR}${SFWMM_START_MONTH}${SFWMM_START_DAY},${SFWMM_END_YR}${SFWMM_
END_MONTH}${SFWMM_END_DAY} -n1 -B-1.0 $runpath/daily_stg_minus_lsel.bin

    #create grid_shot files for each
    $hsmBin/grid_shot -n2 -d2 mfl_r151_n1_B-3.bin > /tmp/mfl130gs.$$
    $hsmBin/grid_shot -n2 -d2 mfl_r76_n1_B-2.bin > /tmp/mfl175gs.$$
    $hsmBin/grid_shot -n2 -d2 mfl_r31_n1_B-1.bin > /tmp/mfl1150gs.$$

    #need to create a composite roco file using above three & some sort of mask
    #this is needed because any one cell had ONE certain soil type and ONE
corresponding criteria
    paste /vol/hsm/scripts/solaris/sfwmm-pm-graphics/env/exec/sfmbudsoils.lst
/tmp/mfl130gs.$$ /tmp/mfl175gs.$$ /tmp/mfl1150gs.$$ > /tmp/allgs.$$
    awk '{if($2==0) printf(" %2d %2d %d\n", $3,$4,-99);\
        if($2==1) printf(" %2d %2d %d\n", $3,$4,$5);\
        if($2==2) printf(" %2d %2d %d\n", $6,$7,$8);\
        if($2==3) printf(" %2d %2d %d\n", $9,$10,$11)}' /tmp/allgs.$$ >
/tmp/mfl_composite_gs.$$

    #create composite roco file
    cat /tmp/mfl_composite_gs.$$ | $hsmBin/g2roco >! mfl_composite.roco

    #create a WMM composite roco file - NSM45 composite file
    paste $nsmRun/MAPS/RUNS/mfl_composite.roco mfl_composite.roco | gawk
' {printf(" %s %5.2f\n", $1,$2-$4)}' >! mfl_diff_nsm45-wmm_composite.roco

    /bin/rm /tmp/allgs.$$ /tmp/mfl_composite_gs.$$ /tmp/mfl130gs.$$
/tmp/mfl175gs.$$ /tmp/mfl1150gs.$$
    echo "MFL post-processing done."
endif

#next the MAPS/HPIMP hydroperiod improvement post processing
#chk for the existence of MAPS/HPIMP dir
if (-e $runpath/MAPS/HPIMP) then

```

```

chdir $runpath/MAPS/HPIMP
echo $PWD
echo "Performing HPIMP post-processing..."

# run grid_hpimp to get the hydroperiod improvement
$hsmBin/grid_hpimp < $hpimpin > hpimpstdout.$$

#see if HPIMP completed successfully
set hpimpstg = `grep -i "completed successfully" hpimpstdout.$$`
/bin/rm hpimpstdout.$$

# get the hpimp.bin diff file name from line 5 of control file
set hpimp_bin = `sed -n 5p $hpimpin | sed -e s/\`"/ | sed -e s/\`"/ | cut -
d. -f1`

#create mean SFWMM hpimp roco file
$hsmBin/grid_shot -n1 -d2 $hpimp_bin.bin | $hsmBin/gs2roco >!
$hpimp_bin.roco

echo "HPIMP post-processing done."
endif

#next the c43c44 Budget summaries post processor
chdir $runpath/C43C44
echo "Performing C43C44 post-processing..."
$hsmBin/lok2est_bud2 $c43c44in >&! c43c44.echo
#grep for the string "terminated correctly" in the echo file
set c43c44stg = `grep -i "terminated correctly" c43c44.echo`

#next the ASR Bubble Budget summaries post processor
set norststg = ""
if($norstin != "") then
chdir $runpath/OTHER_WBUD/NOR_STO_BUD
echo "Performing North Storage Water Budget post-processing..."
$hsmBin/noresbud $norstin >&! norstbud.echo
#grep for the string "terminated correctly" in the echo file
set norststg = `grep -i "terminated correctly" norstbud.echo`
endif

#next the ASR Bubble Budget summaries post processor
set asrstg = ""
if($asrin != "") then
chdir $runpath/OTHER_WBUD/ASR_BUD
echo "Performing ASR Bubble Budget post-processing..."
$hsmBin/asrbud $asrin >&! asrbud.echo
#grep for the string "terminated correctly" in the echo file
set asrstg = `grep -i "terminated correctly" asrbud.echo`
endif

#next the CONVEYANCE post processor
if( ! -e $runpath/CONVEY) mkdir $runpath/CONVEY
chdir $runpath/CONVEY
echo "Performing CONVEYANCE post-processing USING sfwmm2dss_v2.4..."
if( -e tape62) /bin/rm tape62
ln -s $runpath/tape62 tape62
$hsmBin/sfwmm2dss_convey_v2.4 tape62

```



```

# store the ascii data files into convey.dss
foreach i ( avail_sto design hw non_thru tw cal_flow ds_needs max_capacity )
    echo Storing $i data file into convey.dss
    echo convey.dss | $hsmBin/stoDSS ${i}
end
#if convey.dss has been created, delete all the ascii data files
if( -e convey.dss ) /bin/rm avail_sto hw tw cal_flow ds_needs max_capacity
tape62
if( -e convey.dss ) /bin/rm design non_thru

#end all post processing
endif

#chk if post processing terminated right
if ($#econstg == 2) then
    echo "LEC ECON post processing was terminated CORRECTLY." > /tmp/econ.$$
    /bin/rm $runpath/ECON/econ_post.echo
else
    echo "LEC ECON post processing was terminated INCORRECTLY. For screen\
echos, see the file econ_post.echo." > /tmp/econ.$$
endif
if ($#eaastg == 2) then
    echo "EAA ECON post processing was terminated CORRECTLY." > /tmp/ea.$$
    /bin/rm $runpath/ECON/ea_econ.echo
else
    echo "EAA ECON post processing was terminated INCORRECTLY. For screen\
echos, see the file ea_econ.echo." > /tmp/ea.$$
endif
if ($#losastg == 2) then
    echo "LOSA SSM post processing was terminated CORRECTLY." > /tmp/losa.$$
    /bin/rm $runpath/ECON/losassm.echo
else
    echo "LOSA SSM post processing was terminated INCORRECTLY. For screen\
echos, see the file losassm.echo." > /tmp/losa.$$
endif
if ($#wbudstg == 2) then
    echo "WBUD post processing was terminated CORRECTLY." > /tmp/wbud.$$
    /bin/rm $runpath/WBUD/wbud.echo
else
    echo "WBUD post processing was terminated INCORRECTLY. For screen\
echos, see the file wbud.echo." > /tmp/wbud.$$
endif
if ($#c43c44stg == 2) then
    echo "C43C44 post processing was terminated CORRECTLY." > /tmp/c43c44.$$
    /bin/rm $runpath/C43C44/c43c44.echo
else
    echo "C43C44 post processing was terminated INCORRECTLY. For screen\
echos, see the file c43c44.echo." > /tmp/c43c44.$$
endif
if ($#hpstg == 6) then
    echo "HYDROPERIOD post processing was done." > /tmp/hp.$$
else
    echo "HYDROPERIOD post processing was not done because
daily_stg_minus_lsel.bin file\
didnt exist or MAPS/HYDROPERIOD dir didnt exist." > /tmp/hp.$$
endif
if ($#pondstg == 6) then

```

```

        echo "POND post processing was done." > /tmp/pond.$$
    else
        echo "POND post processing was not done because daily_stg_minus_lsel.bin
file\
didnt exist or MAPS/POND dir didnt exist." > /tmp/pond.$$
    endif
#   if ($#sttrigmap == 9) then
#       echo "Trigger maps were done." > /tmp/trig.$$
#       /bin/rm $runpath/trigmap.log
#   else
#       echo "Trigger maps were not done because the program failed or \
#MAPS/TRIGGERS dir didnt exist.  If errors, see file "$runpath"/trigmap.log" >
/tmp/trig.$$
#   endif
    if ($#norststg == 2) then
        echo "North Storage water budget post processing was terminated
CORRECTLY." > /tmp/norstbud.$$
        /bin/rm $runpath/OTHER_WBUD/NOR_STO_BUD/norstbud.echo
        else if ($norstin != "" && $#norststg != 2) then
            echo "North Storage water budget post processing was terminated
INCORRECTLY. For screen\
echos, see the file norstbud.echo." > /tmp/norstbudbud.$$
            else if ($norstin == "") then
                set norststg = (0 1)
                echo "No North Storage for this run." > /tmp/norstbud.$$
            endif
            if ($#asrstg == 2) then
                echo "ASR Bubble budget post processing was terminated CORRECTLY." >
/tmp/asrbud.$$
                /bin/rm $runpath/OTHER_WBUD/ASR_BUD/asrbud.echo
                else if ($asrin != "" && $#asrstg != 2) then
                    echo "ASR Bubble budget post processing was terminated INCORRECTLY. For
screen\
echos, see the file asrbud.echo." > /tmp/asrbud.$$
                    else if ($asrin == "") then
                        set asrstg = (0 1)
                        echo "NO ASR for this run." > /tmp/asrbud.$$
                    endif
                endif

            if ($#hpimpstg == 6) then
                echo "HPIMP post processing was done." > /tmp/hpimp.$$
            else
                echo "HPIMP post processing was not done because MAPS/HPIMP dir or
control file didnt exist." > /tmp/hpimp.$$
            endif

#send mail to all in the mailing list that post processing completed
if ($#string == 2 && $#econstg == 2 && $#eaastg == 2 && $#wbudstg == 2 &&
#c43c44stg == 2 && $#norststg == 2 && $#asrstg == 2 ) then
    echo ALL Post Processing "(ECON, WBUD, CONVEY, OVFLOW, HYDROPERIOD, POND,
HPIMP, TRIGGERS, C43C44, NOR_STOR & ASR)" for the run\
    '$runpath' has been completed on `date` in the directory '$runpath'.
Maps will be produced next: See files with\
extension .log for results.\
    `cat /tmp/econ.$$` `cat /tmp/eaastg.$$` `cat /tmp/losa.$$` `cat /tmp/wbud.$$`
`cat /tmp/c43c44.$$` `cat /tmp/hpimp.$$`\

```

```

    `cat /tmp/pond.$$` `cat /tmp/hpimp.$$` `cat /tmp/norstbud.$$` `cat
/tmp/asrbud.$$` `cat /tmp/mvetbin.$$`\
    `cat /tmp/rundone.$$` | mailx -r $user"@sfwmd.gov" -s "SUCCESSFUL - $runtit
Post-Processing" $mlist
else if ($#string == 2) then
    echo ALL Post Processing "(ECON, WBUD, CONVEY, HYDROPERIOD, POND, HPIMP,
C43C44, NOR_STOR & ASR)" for the run\
    '$runtit' has been completed on `date` in the directory '$runpath'.
Maps will be produced next: See files\
with extension .log for results.\
    `cat /tmp/econ.$$` `cat /tmp/eea.$$` `cat /tmp/losa.$$` `cat /tmp/wbud.$$`
`cat /tmp/c43c44.$$` `cat /tmp/hpimp.$$`\
    `cat /tmp/pond.$$` `cat /tmp/hpimp.$$` `cat /tmp/norstbud.$$` `cat
/tmp/asrbud.$$` `cat /tmp/mvetbin.$$`\
    `cat /tmp/rundone.$$` | mailx -r $user"@sfwmd.gov" -s "UNSUCCESSFUL - $runtit
Post-Processing" $mlist
endif
#
# Produce trigger maps only if the directory MAPS/TRIGGERS does exist
#
    chdir $runpath
    echo $PWD
    echo "Performing TRIGGER map production..."
#
#
# if (-e $runpath/MAPS/TRIGGERS)then
#
#     Remove existing map
#
#     /bin/rm $runpath/MAPS/TRIGGERS/trig_map.ps
#
#     Creating short name for the run out of the SFWMM output directory path.
#
#     set short2 = $runpath:h
#     set short1 = $short2:t
#
#     The gawk command in the following line, puts an "x" in front of short
name, in case it starts
#     with a number (arc-info limitation in creating maps)
#
#     set short = `echo $short1 | /bin/tr '[A-Z]' '[a-z]' | gawk '{num=0;
for(i=0;i <= 9;i++) {if(substr($1,1,1) == i) num=1}; if (num == 1)
{printf("x%s",$1); exit} ; if(num == 0) {printf("%s",$1);exit}}`
#     echo $short
#
#
#     Create the trigmap.cf, with the data to post process trigoutp file
#
#     if (-e trigmap.cf)/bin/rm trigmap.cf
#
#     echo '$short' '$runpath'/trigoutp' > trigmap.cf
#     /bin/cp trigmap.cf $runpath/MAPS/TRIGGERS/trigmap.cf
#     /bin/mv trigmap.cf $HSMGIS/programs/triggers/trigmap.cf
#
#     Pass run title to the trigmaps.scr script
#
#     echo $runtit > sfwmm_title.inp

```

```

#
# Run the process
#
# $hsmBin/trigmaps.scr >&! trigmap.log
#
# Set the status of the process
#
# set sttrigmap = grep "COMPLETED" trigmap.log
# /bin/rm sfwmm_title.inp
else
# set sttrigmap = 0
# echo "Directory "$runpath"/MAPS/TRIGGERS does not exist. Trigger maps
were not created" > trigmap.log
endif
#
# Create hydroperiod maps
#
if (-e $runpath/MAPS/HYDROPERIOD ) then
# chdir $runpath/MAPS/HYDROPERIOD
# echo $PWD
# echo "Performing HYDROPERIOD map production..."
#
# Remove existing maps
#
# /bin/rm *.ps
#
#Create file needed by script to create maps (sfwmm.inp)
# echo $runpath > sfwmm.inp
# echo $runtit >> sfwmm.inp
#
#Produce hydroperiod maps (Only if file hydp.inp does exist)
# if (-e hydp.inp)then
# $hsmBin/crt_maps.scr < hydp.inp >&! hydpmap.log
# else
# echo "Hydroperiod maps were not produced because file hydp.inp was not
found." >&! hydpmap.log
# endif
#
#Produce hydroperiod difference maps (Only if file hpdiff.inp does exist)
# if (-e hpdiff.inp)then
# $hsmBin/crt_maps.scr < hpdiff.inp >&! hpdiffmap.log
# else
# echo "Hydroperiod difference maps were not produced because file
hpdiff.inp was not found." >&! hpdiffmap.log
# endif
# /bin/rm sfwmm.inp
endif
#
# Create hydroperiod improvement maps
#
#
if (-e $runpath/MAPS/HPIMP) then
# chdir $runpath/MAPS/HPIMP
# echo $PWD
#
# Remove existing maps
#
# /bin/rm *.ps

```

```

#Create file needed by script to create maps (sfwmm.inp)
echo $runpath > sfwmm.inp
echo $runtit >> sfwmm.inp

echo "Performing HPIMP map production..."
#Produce hydroperiod difference maps (Only if file hpimp.inp does exist)
if (-e hpimp.inp)then
    $hsmBin/crt_maps.scr < hpimp.inp >&! hpimpmap.log
else
    echo "Hydroperiod improvement map was not produced because file
hpimp.inp was not found." >&! hpimpmap.log
endif
/bin/rm sfwmm.inp
endif
#
# Create daily ponding maps
#
if (-e $runpath/MAPS/POND) then
    chdir $runpath/MAPS/POND
    echo $PWD
#
# Remove existing maps
#
/bin/rm *.ps

echo "Performing PONDING map production..."

#Create file needed by script to create maps (sfwmm.inp)
echo $runpath > sfwmm.inp
echo $runtit >> sfwmm.inp

#Produce ponding maps (Only if file day_pond.inp does exist)
if (-e day_pond.inp)then
    $hsmBin/crt_maps.scr < day_pond.inp >&! pondmap.log
else
    echo "Ponding map was not produced because file day_pond.inp was not
found." >&! pondpmap.log
endif

#Produce ponding difference maps (Only if file pondiff.inp does exist)
if (-e pondiff.inp)then
    $hsmBin/crt_maps.scr < pondiff.inp >&! pondiffmap.log
else
    echo "Ponding difference map was not produced because file pondiff.inp
was not found." >&! pondiffmap.log
endif
endif
#remove temp files
/bin/rm /tmp/*$$*

```