17

Design of the SWMM Data Translation and the Testing of SWMM 5 Engine

Carl Chan, Edward H. Burgess, Trent Schade and Robert E. Dickinson

This chapter presents a graphical user interface (GUI) for the translation of SWMM 4 input files to SWMM 5 input files developed under a Cooperative Research and Development Agreement (CRADA) between Camp Dresser & McKee Inc. (CDM) and the U. S. Environmental Protection Agency (USEPA). This chapter also summaries the design of the SWMM Data Translation program and discusses the SWMM 5 testing procedures. One of the purposes of the CRADA is to provide upward compatibility between SWMM 4 and SWMM 5 by providing a user friendly interface that will translate existing SWMM 3.3 and SWMM 4 RUNOFF, TRANSPORT, and EXTRAN files to a single SWMM 5 input file. The SWMM 5 input file is a quasi-XML file that has link node data in a free formatted unstructured ASCII file. The SWMM 5 input file then can be read by the EPASWMM.EXE graphical user interface or the standalone SWMM 5 engine written in C programming language.

The translator is written in Visual Basic 6. The source code for the SWMM 4 to SWMM 5 translator will be available to the general public in the official release of SWMM 5. The translator parses the SWMM 3.3 and SWMM 4 data sets and saves the SWMM 3.3 and SWMM 4 information to a data structure. The data structure is then sorted and the equivalent information is saved to a SWMM 5 input data file. The translator does more than simply transcribe input data fields. It translates TRANSPORT input data sets without elevation to SWMM 5 hydraulic data sets with elevations and free outfall boundary conditions.

Chan, C.C., E. Burgess, T. Schade and R.E. Dickinson. 2004. "Design of the SWMM Data Translation and the Testing of SWMM 5 Engine." *Journal of Water Management Modeling* R220-17. doi: 10.14796/JWMM.R220-17.
 www.chijournal.org ISSN: 2292-6062 (Formerly in Innovative Modeling of Urban Water Systems. ISBN: 0-9683681-9-0)

An additional purpose of the translator program in the initial release of SWMM 5 is to provide many test data sets for the hydrologic and hydraulic comparison of SWMM 4 and SWMM 5. Over 200 RUNOFF, TRANSPORT, and EXTRAN data sets were translated from SWMM 4 to SWMM 5. The hydrologic and hydraulic output results were then compared between SWMM 4 and SWMM 5 as part of the QA/QC (quality assurance/quality control) task in the SWMM CRADA.

The CDM SWMM 4 data translating process contains three basic components:

- digesting SWMM 4 data sets,
- parsing the SWMM 4 data line and storing data into memory, and
- converting data to SWMM 5 format.

17.1 Digesting the SWMM 4 Input File

The translating procedure begins with opening the SWMM 3/4 input files. The translator reads and processes one data line at a time until the end of the SWMM file. Because SWMM 4.x input is in free-format style, the translator reorganizes the input into a space-delimited format and arranges the data such that it is on the same line as its SWMM line identifier (i.e. no wrap around). To do so, the translator does not process the data immediately after reading a line. The translator saves the line in the temporary memory and continues to read the next line. If the next line is determined to be a comment line, starting with "*," or a data line beginning with a known SWMM line identifier, the translator will start digesting the line which was earlier stored in the temporary memory. The line that was just read will be stored in the temporary memory and the translator will repeat the reading procedure after it finishes the digesting process. If the next line does not started with "*" or a known SWMM line identifier, the translator assumes that the line contains "wrap-around" data from the previous line, and the translator will join this line with the previous one in the temporary memory. Figure 17.1 shows the translator operation in digesting a SWMM 4 data set.

17.2 Parsing SWMM 4 Data Sets

After a data line is translated and a complete SWMM data input line is created, Figure 17.2 shows how the translator parses the SWMM input line. In the parsing process, the translator will first divide the input data line into two parts:

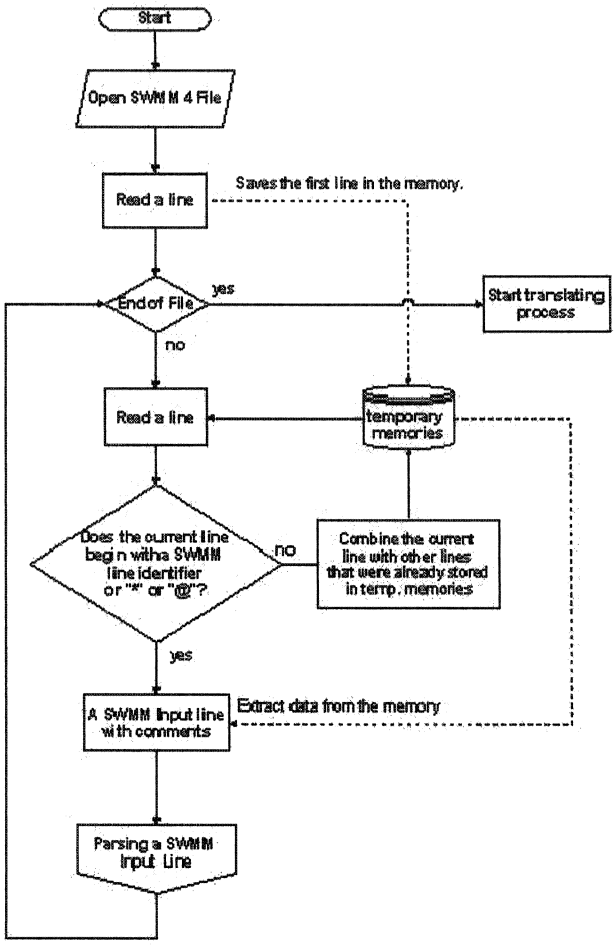


Figure 17.1 Digesting SWMM 4 input file.

SWMM data attributes and comments. The translator carries the comments that are at the end of each SWMM 3.x/4.x input line to the corresponding location in the SWMM 5 file. For the SWMM data attributes, the translator parses the attribute section of the line into tokens by using spaces as the separators.

The translator will next perform a syntax check on the attributes. The first token, which contains the SWMM line identifier, is reviewed first by the translator to determine how to perform the syntax check. For example, if the

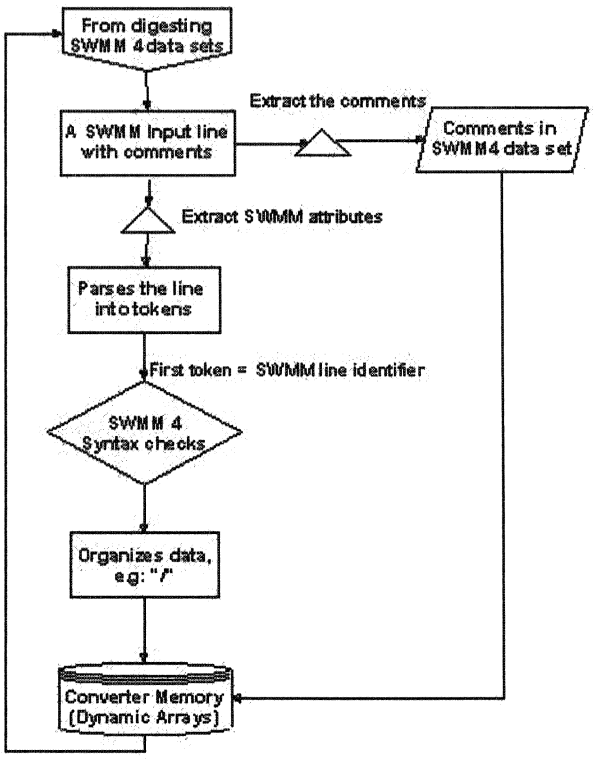


Figure 17.2 Parsing SWMM 4 data sets.

first token is "D1," the translator will check the SWMM data attributes section and examine whether there was at least six entries. If the input fails the syntax check, the translator will continue to process the next line and write an error in the converter log file.

After the data line passes the syntax check, the translator will further organize the data attributes if necessary. For a SWMM input line that ends with "/," the translator will fill in the appropriate number of zeros at the end of the line. The translator will then store all of the data in memory, while waiting to convert to SWMM 5 format. All memory in the translator is dynamically allocated. This means that the number of elements in the translator is unlimited.

17.3 Converting Data to SWMM 5 Format

After parsing an input file and storing all data in memory, the translator will rearrange the data based on the SWMM 5 input format. Before starting conversion, the translator will manipulate the data based on the setting in SWMM 4. For example, the translator will recalculate the data in the H1 line based on the default and ratio lines. Also, the translator will convert the absolute elevation in ZP1 and ZP2 into an offset elevation, if the BB-JELEV control card is set to be larger than zero. The translator will also determine the proper junction invert elevation for TRANSPORT input. (This feature will be discussed in a later section.)

After manipulating and rearranging the data in memory, the translator will convert the data and write to a text file, which the user previously defined. The new format is based on the EPA SWMM 5 input format guide. Meanwhile, the translator also creates a log file that summarizes the translation and records any improperly translated data. Finally, the user can import the translated text file into the SWMM 5 GUI program or use it for solving problems with SWMM5.EXE. Figure 17.3 describes the conversion process.

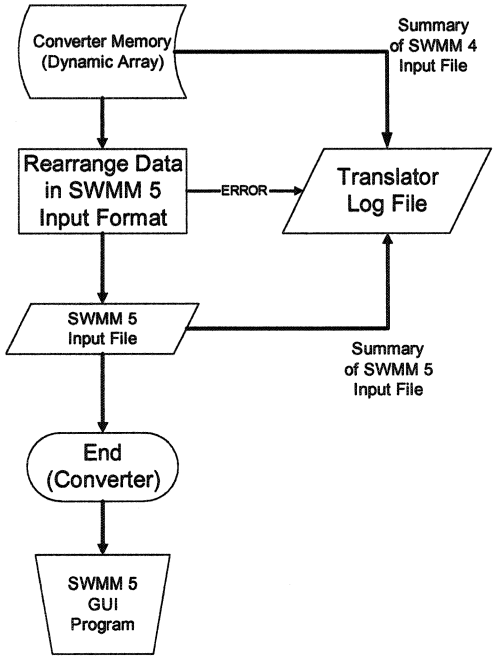


Figure 17.3 Converting data to SWMM 5 format.

17.3.1 Log Files

The translator produces two output files: the translated input file in SWMM 5 format and a log file that records details of the translation. The log file consists of three parts: a summary of the SWMM 3.x/4/x input file; a summary of the converted file; and a log section. The translator summarizes numbers of different elements (i.e. nodes, conduits, weirs, pumps, orifices, etc.) in the SWMM 3.x/SWMM4.x input file. In the summary of the converted file, the translator counts the number of elements that are successfully translated to SWMM 5 format. In the log section, the translator displays assumptions used during the translation. This section also lists errors that have occurred during the translation.

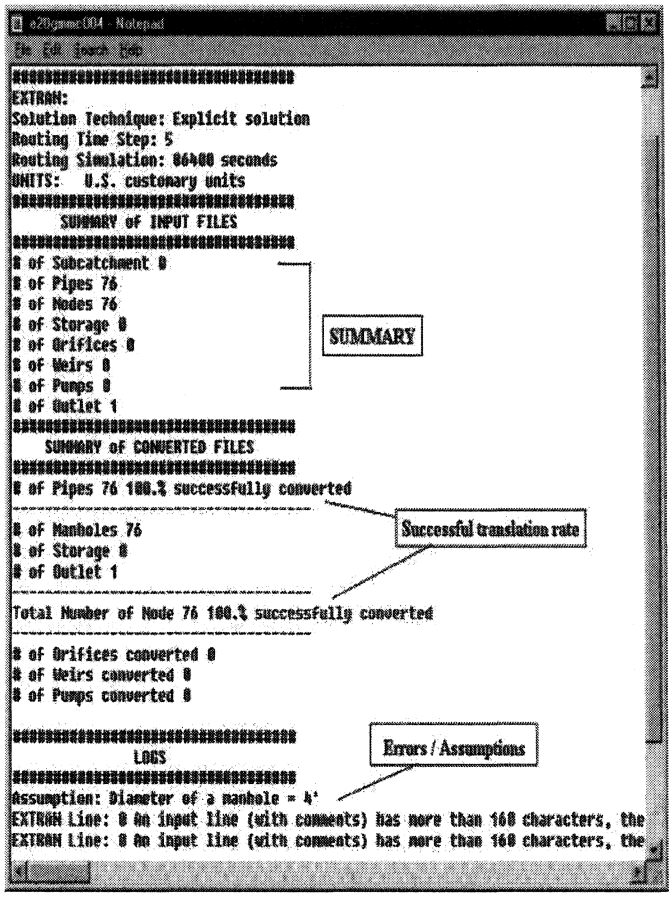


Figure 17.4 Log file.

17.3.2 TRANSPORT

TRANSPORT is kept in the form of a solution in the hydraulic layer of SWMM 5. Because there is no requirement for invert elevations in SWMM 3.x/4.x TRANSPORT, the translator manipulates the data and estimates the invert elevation for each junction based on the length and slope of the conduit elements.

The translator will automatically check the sewer network connectivity. TRANSPORT allows connection between two junction elements without a link. For example, a divider element can connect to a manhole element. In this case, the translator will simply add a dummy conduit in order to keep the network connectivity in SWMM 5.

The translator identifies the outlets in the sewer network and assigns a zero elevation for each outlet. Then it traces the sewer network in an upstream direction only and calculates the invert for each junction using length and slope. The tracing is repeated for each outlet. If a junction was assigned an elevation by another calculation from a different outlet, the translator compares the new calculation with the old elevations. The translator will retain the highest elevation calculation. The translator then retraces the route that was overwritten and recalculates the elevation, both upstream and downstream.

Example: Determining Manhole Inverts for TRANSPORT Input

Assumptions:

1. all links are 100 ft (30 m) in length, and
2. all links are 1% in slope.

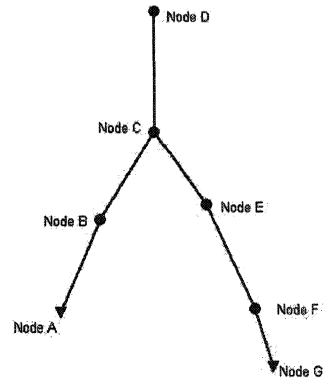


Figure 17.5 TRANSPORT example

Step 1: The translator identifies the outlets in the network and assigns a zero elevation for each outlet.

Node	Type	Invert (ft)
A	Outlet	0
B	Junction	unknown
C	Divider Junction	unknown
D	Junction	unknown
E	Junction	unknown
F	Junction	unknown
G	Outlet	0

Step 2: The translator traces the network in the upstream direction only, and calculates the inverts of each upstream node, based on slope and length - starting with outlet Node A.

Node	Type	Invert (ft)	Description
A	Outlet	0	Outlet: Assume 0
B	Junction	1	(100 * 1% + Invert of Node A)
C	Divider Junction	2	(100 * 1% + Invert of Node B)
D	Junction	3	(100 * 1% + Invert of Node C)
E	Junction	unknown	
F	Junction	unknown	
G	Outlet	0	Outlet: Assume 0

Step 3: The translator traces the network in the upstream direction, starting with outlet node G.

		Invert (ft)			
Node	Type	Step 1	Step 2	Step 3	Description
A	Outlet	0	0	0	
B	Junction	unknown	1	1	
C	Divider Junction	unknown	~~2~~	3	(100 * 1% + Invert of Node E)
D	Junction	unknown	~~3~~	4	(100 * 1% + Invert of Node C)
E	Junction	unknown	unknown	2	(100 * 1% + Invert of Node F)
F	Junction	unknown	unknown	1	(100 * 1% + Invert of Node G)
G	Outlet	0	0	0	Outlet: Assume 0

A junction was previously assigned an invert elevation by another tracing/ calculation from a different outlet (in this case, node C). The translator compares the new calculated invert (3 ft/0.9 m) with the old elevation (2 ft/ 0.6 m), and retains the highest invert elevation (3 ft in this example).

Step 4: The translator retraces the network starting from the node where the invert was overwritten (Node C in this example). All elevations upstream and downstream of Node C are recalculated and overwritten.

		Invert (ft)				
Node	Type	Step 1	Step 2	Step 3	Step 4	Description
A	Outlet	0	0	0	1	Invert of Node B - 100 * 1%
B	Junction	unknown	1	†	2	Invert of Node C - 100 * 1%
C	Divider Junction	unknown	2	3	3	
D	Junction	unknown	3	4	4	
E	Junction	unknown	unknown	2	2	
F	Junction	unknown	unknown	1	1	
G	Outlet	0	0	0	0	

Step 5: Step 3 should be repeated, if there are more outlets in the network.

Figure 17.6 shows the inverts for the manholes after recalculations.

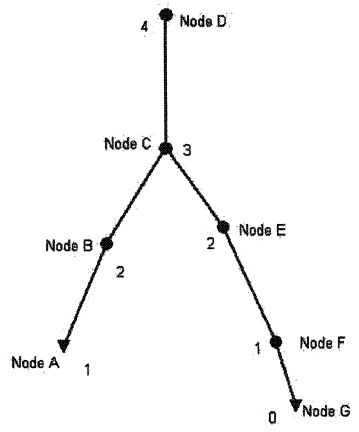


Figure 17.6 Inverts for the manholes after recalculations.

17.3.3 Example of how to Convert a SWMM 4 EXTRAN Formatted Data Set to SWMM 5 Input Data

The tutorial example below (Figure 17.7) demonstrates how to convert a SWMM 4 EXTRAN formatted data set to SWMM 5 input data. This classic EXTRAN Example 1 data file is from the EXTRAN 3 and EXTRAN 4 manuals.

The major line groupings from EXTRAN that will be translated from SWMM4 to SWMM5 are: (i) conduit data from the C1 lines, (ii) manhole data from the D1 lines, (iii) outfall data from I1 line, (iv) inflow time series data from the K3 lines, and (v) EXTRAN job control information from the A and B lines.

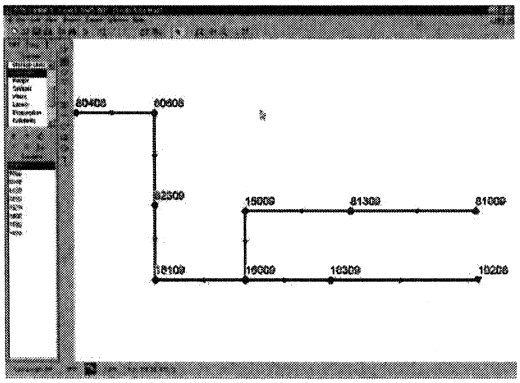


Figure 17.7 EXTRAN User's Manual Example Problem 1.

Step 1: Load the SWMM 4 to SWMM 5 Converter from your Desktop.

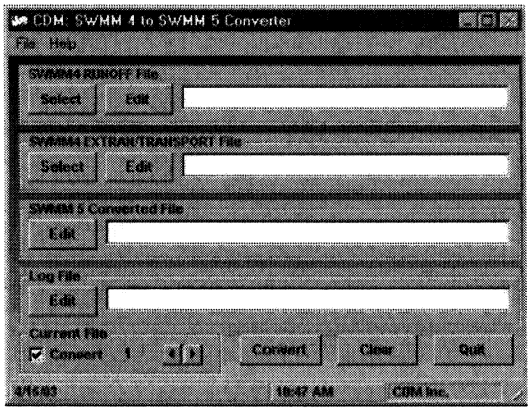


Figure 17.8 CDM SWMM Converter.

Step 2: Select the SWMM 4 Data file to be converted.

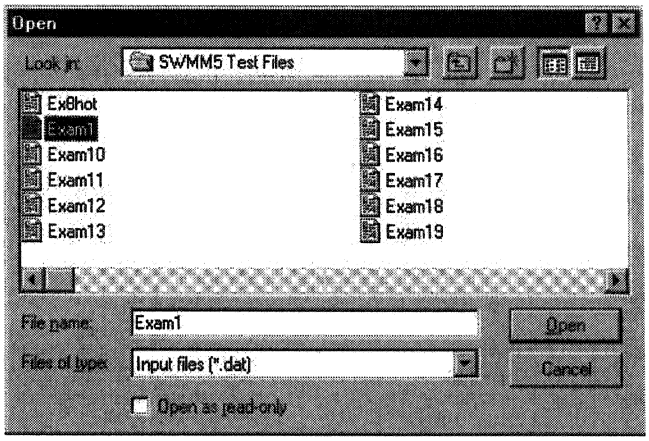


Figure 17.9 CDM SWMM Converter - select SWMM 4 data file.

Step 3: Click on the Convert button to convert the file. The translator program generates the SWMM 5 output file names automatically.

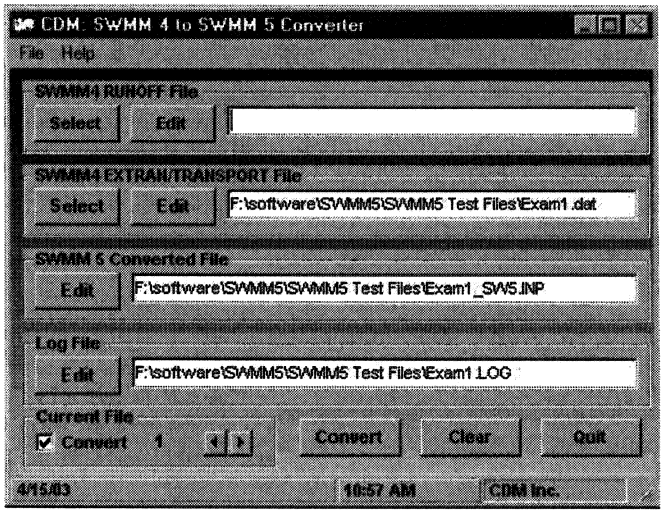


Figure 17.10 CDM SWMM Converter - ready to convert.

Step 4: Click on Edit button under log file to see if all lines have been translated.

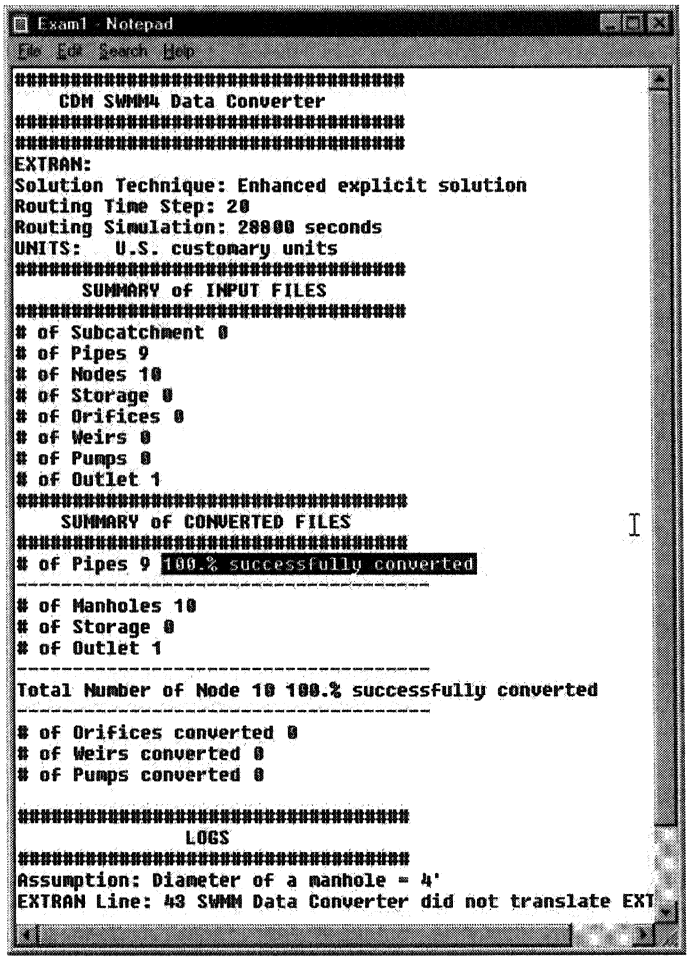


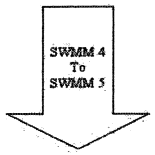
Figure 17.11 CDM SWMM Converter - log file.

Step 5: Edit the SWMM 5 created file to see how the conduit data were translated. The conduit data were translated from C1 lines of WMM 4 to the data categories [CONDUITS], [XSECTIONS], and [LOSSES] in the SWMM5 input data file.

The SWMM 4 Conduit Data from the C1 lines of the file Exam1.dat can be seen in the next table.

SWMM 4 Conduit Data from the C1 lines of the file Exam1.dat

*	NCOND	NJ1	NJ2	QO	NKLASS	AFULL	DEEP	WIDE	LEN	ZP1	ZP2	ROUGH	STHETA	SPHI
C1	8040	80408	80608	0	1	0	4	0	1800	0	0	0.015	0	0
C1	8060	80608	82309	0	1	0	4	0	2075	0	2.2	0.015	0	0
C1	8100	81009	81309	0	1	0	4.5	0	5100	0	0	0.015	0	0
C1	8130	81309	15009	0	1	0	4.5	0	3500	0	0	0.015	0	0
C1	1030	10309	10208	0	6	0	9	0	4500	0	0	0.016	3	3
C1	1570	15009	16009	0	1	0	5.5	0	5000	0	0	0.0154	0	0
C1	1600	16009	16109	0	1	0	6	0	500	0	0	0.015	0	0
C1	1630	16009	10309	0	6	0	9	0	300	0	0	0.015	3	3
C1	1602	82309	16109	0	1	0	5	0	5000	0	0	0.034	0	0



[CONDUITS]

;Name	Node1	Node2	Length	N	Z1	Z2	Q0
8040	80408	80608	1800	0.015	0	0	0
8060	80608	82309	2075	0.015	0	2.2	0
8100	81009	81309	5100	0.015	0	0	0
8130	81309	15009	3500	0.015	0	0	0
1030	10309	10208	4500	0.016	0	0	0
1570	15009	16009	5000	0.0154	0	0	0
1600	16009	16109	500	0.015	0	0	0
1630	16009	10309	300	0.015	0	0	0
1602	82309	16109	5000	0.034	0	0	0

The SWMM 5 Conduit data are split into [CONDUITS]. The data are free formatted on each line without any line identifiers.

The SWMM 5 conduit data are also split into [XSECTIONS] and [LOSSES].

[XSECTIONS]

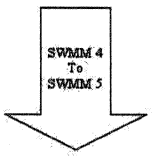
;LINK	SHAPE	GEOM1	GEOM2	GEOM3	GEOM4
8040	CIRCULAR	4	0	0	0
8060	CIRCULAR	4	0	0	0
8100	CIRCULAR	4.5	0	0	0
8130	CIRCULAR	4.5	0	0	0
1030	TRIANGULAR	9	54	0	0
1570	CIRCULAR	5.5	0	0	0
1600	CIRCULAR	6	0	0	0
1630	TRIANGULAR	9	54	0	0
1602	CIRCULAR	5	0	0	0

[LOSSES]

;LINK	Inlet	Outlet	Average	Flapgate
8040	0	0	0	NO
8060	0	0	0	NO
8100	0	0	0	NO
8130	0	0	0	NO
1030	0	0	0	NO
1570	0	0	0	NO
1600	0	0	0	NO
1630	0	0	0	NO
1602	0	0	0	NO

Step 6: SWMM 4 Junction Data from the D1 lines for the input file Exam1.dat.

* JUNCTION DATA						
* JUN	GRELEV	Z	QINST	Y	X-coord	Y-coord
D1 80408	138	124.6	0	0	28.9148	470.882
D1 80608	135	118.3	0	0	66.0218	471.093
D1 81009	137	128.2	0	0	217.7331	424.454
D1 81309	130	117.5	0	0	158.7037	424.454
D1 82309	155	112.3	0	0	66.0218	427.186
D1 10208	100	89.9	0	0	218.6526	392.186
D1 10309	111	101.6	0	0	148.7037	392.186
D1 15009	125	111.5	0	0	108.7037	424.454
D1 16009	120	102	0	0	108.7037	392.186
D1 16109	125	102.8	0	0	66.0218	392.186



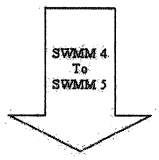
[JUNCTIONS]					
;Name	Invert Elev.	Init.	Max. Depth	Surcharge Depth	Ponded Area
80408	124.6	0	13.4	0	0;
80608	118.3	0	16.7	0	0;
81009	128.2	0	8.8	0	0;
81309	117.5	0	12.5	0	0;
82309	112.3	0	42.7	0	0;
10309	101.6	0	9.4	0	0;
15009	111.5	0	13.5	0	0;
16009	102	0	18	0	0;
16109	102.8	0	22.2	0	0;

[COORDINATES]		
;Node	X-COORD	Y-COORD
80408	28.9148	470.88
80608	66.0218	471.09
81009	217.733	424.45
81309	158.704	424.45
82309	66.0218	427.19
10208	218.653	392.19
10309	148.704	392.19
15009	108.704	424.45
16009	108.704	392.19
16109	66.0218	392.19

The SWMM 5 translated Junction data are split into two sections: [JUNCTIONS] and [COORDINATES]. Each section contains a portion of the D1 line data from SWMM 4.

Step 7: SWMM 4 inflow Data are on the K1, K2, and K3 lines of the input file. The SWMM 5 Translated Inflow Data are in the [INFLOWS] and [TIMESERIES] sections in the created data file.

```
K1      3
K2      82309 80408 81009
K3      0.0   0.0   0.0   0.0
K3      0.25  40.0  45.0  50.0
K3      3.0   40.0  45.0  50.0
K3      3.25  0.0   0.0   0.0
K3      12.0  0.0   0.0   0.0
```

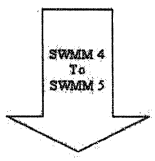


```
[INFLOWS]
;;================================================
82309  FLOW   FLOW@82309
80408  FLOW   FLOW@80408
81009  FLOW   FLOW@81009

[TIMESERIES]
FLOW@82309     01-01-1988    00:00  0.00000
                             00:15  40.00000
                             03:00  40.00000
                             03:15  0.00000
                             12:00  0.00000
FLOW@80408     01-01-1988    00:00  0.00000
                             00:15  45.00000
                             03:00  45.00000
                             03:15  0.00000
                             12:00  0.00000
FLOW@81009     01-01-1988    00:00  0.00000
                             00:15  50.00000
                             03:00  50.00000
                             03:15  0.00000
12:00  0.00000
```

Step 8: The SWMM 4 outfall information is on the I1 or I2 lines of the input file Exam1.dat.

```
I1 10208  1
J1 1
```

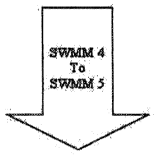



```
[OUTFALLS]
;       Invert Outfall       Stage/Table
;Name   Elev.         Type          Time Series
;;================================================
10208  89.9   FREE   NO
```

The translated SWMM5 outfall information is in the [OUTFALLS] section.

Step 9: The SWMM 5 job control information is translated from the A and B lines of SWMM 4 to the [TITLES] and [OPTIONS] section.

```
$EXTRAN
A1  'EXTRAN USER''S MANUAL EXAMPLE PROBLEM 1'
A1  'BASIC PIPE SYSTEM FROM FIGURE 3-1'
*  ISOL KSUPER
B0  1    0
*   NTCYC DELT  TZERO NSTART INTER JNTER REDO
B1   1440   20.0  0.0    45   -45     45    0
*   METRIC NEQUAL  AMEN ITMAX SURTOL
B2     0     0      0.0   30   0.05
```




```
[TITLE]
EXTRAN USER''S MANUAL EXAMPLE PROBLEM 1
BASIC PIPE SYSTEM FROM FIGURE 3-1

;  CDM SWMM4 Converter


[OPTIONS]
FLOW_UNITS    CFS
START_DATE    01-01-1988
REPORT_START_DATE    01-01-1988
START_TIME    00:00
REPORT_START_TIME    00:00
END_DATE      01-01-1988
END_TIME      08:00
ROUTING_STEP  00:00:20
REPORT_STEP   00:15:00
FLOW_ROUTING  DW
ALLOW_PONDING NO
INERTIAL_DAMPING      1.0
COURANT_FACTOR      0.75
```

17.4 QA/QC Testing of SWMM5

The SWMM QA plan for the SWMM5 CRADA involves testing of two components: the SWMM engine and the SWMM GUI. Both components are being tested extensively in-house. Limited direct beta testing is also being completed before the SWMM 5 beta release to the public. The main thrust of the testing is to confirm that the new SWMM 5 engine produces results consistent with the existing engine. For instances where the two do not meet predetermined numerical agreement criteria, the difference will be explained within the documentation. The GUI has been tested for operational bugs. Before the SWMM 5 beta release to the public, an expert Technical Design Panel will have the opportunity to test and to respond to the new SWMM software. The Technical Design Panel consists of leading SWMM experts who will provide technical input and review throughout the project. More details about the Technical Design Panel are available at http://www.epa.gov/ednnrmrl/swmm/tech_design_pan.htm.

Over 200 RUNOFF, TRANSPORT, and EXTRAN data sets were translated from SWMM 4 to SWMM 5 using the data converter. These data sets, which were CDM's historical test cases for the previous SWMM version, were designed to test all functions in the SWMM engine. Identical test cases were used in the new version (SWMM 5) and the existing version (SWMM 4). The goal was to achieve the following numerical agreement criterion for each point of the outflow and depth hydrograph at all points of the drainage network in each test:

$$| A_t - B_t | / B_t <= 0.005, \text{ for } | B_t | >= 0.005$$
$$| A_t | <= 0.005, \text{ for } | B_t | < 0.005$$

where:

A_t = the depth and flow from the SWMM 5.0 simulation at time t, and

B_t = the depth and flow from SWMM 4.4 simulation at time t.

Roughly speaking, this requires that all numerical results from the two versions of SWMM agree to three significant figures. Prior to the official version release, any unresolved excursion from the criteria will require a full explanation and accounting in the SWMM 5.0 documentation.

Another measure of the performance across all test cases is the system continuity error. The system continuity error is a measure of the volume of

water gained or lost by numerical problems. A small discontinuity (less than 2.0%) is expected. A large discontinuity indicates either a problem with the solution time-step or a problem with the program. Each test case will the compare the system continuity error with the system continuity error from the previous version of SWMM.

17.4.1 Testing Implementation

Figure 17.12 shows a flowchart for the entire testing procedure.

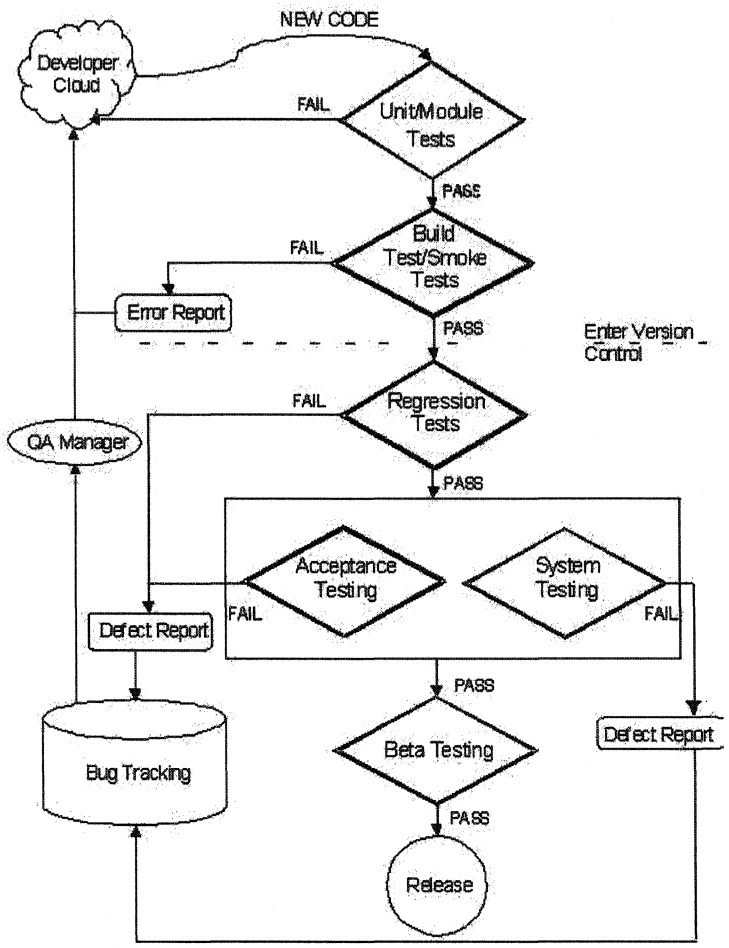


Figure 17.12 SWMM 5 testing procedure.

The testing plan includes six levels of testing:

- unit/module Testing,
- build test/smoke test,
- regression testing,
- acceptance testing,
- system testing, and
- Beta testing.

Unit/Module Testing

Individual developers who are responsible for writing a unit or module may perform Unit/Module Testing at their own discretion. When a developer considers unit or module testing necessary, the developer will include the test program as a utility. This enables future development or redevelopment of the module or unit to utilize the same testing program. The test program should follow all applicable standards for project code and source code control; however, it will not be documented in the Reference Manual or User's Manual. EPA is considering using an open source testing framework such as CppUnit (http://www.xprogramming.com/software.htm).

Build Test/Smoke Test

All developers will submit code to the QA Manager for build testing and smoke testing on a weekly schedule. For the component build, a makefile approach will be used. The build test is intended to confirm that the revisions to any modules allow the source code to compile. This test should also include testing of any installer software. The smoke test runs multiple simple functions several times to make sure there are not basic problems (e.g., memory leakage) with the basic functionality. The QA manager will report the results of each build test/smoke test to the developers as either a pass or a report of errors/exceptions from the compiler or the operating system. EPA is considering using a software tool such as Electric Fence or Purify that will automatically check compiled software for improper use of memory.

Regression Test

To facilitate early testing of the software, the regression testing will be performed on each module as it is completed in the following order:

1. RUNOFF Hydraulics,
2. RUNOFF Water Quality,
3. Kinematic Wave Routing,

4. Kinematic Wave Water Quality,
5. Dynamic Wave (EXTRAN) Routing,
6. Dynamic Wave Water Quality,
7. Rainfall File Reader,
8. Continuous Event Simulation, and
9. Rainfall/Runoff/Routing Statistic.

As modules are completed, the regression test ensures that they work within the full package. The regression test runs a battery of several input files through the computational engine to test individual SWMM functions. The QA manager develops an automated script to run the regression tests and parse the output to perform the comparisons. Also, the QA Manager performs the regression tests and reports results to the development team. The regression test is repeated for each module as it is completed or updated.

In cooperation with CDM and the Technical Panel, the EPA QA staff has prepared a battery of test cases as input files to test the system. The input files were designed to test various computational features of the program. A sample of test files to exercise the hydrologic processes is listed in Table 17.1. The input files for SWMM 5.0 are based on a suite of files used by CDM for testing previous versions of SWMM. The input files were generated from the SWMM data converter. Each regression test file is limited to testing one SWMM function only. The simulation duration includes the entire function exercise. In each test case, the solution of the simulation must be stable in both SWMM 5 and SWMM 4. Also, both results must have low continuity error. The function being tested must exist in both versions of SWMM. A fact sheet is available that describes each test case. The fact sheet includes a schematic of the test system,

Table 17.1 Sample test files and descriptions.

Test File	SWMM Function to be exercised
Impervro.inp	100% impervious
horton_25pct.inp	25% pervious with Horton
horton_50pct.inp	50% pervious with Horton
horton_75pct.inp	75% pervious with Horton
horton.inp	100% pervious with Horton infiltration
horton_regen.inp	Infiltration regeneration with Horton
Greenampt_50pct.inp	50% impervious with Green-Ampt
Greenampt.inp	100% pervious with Green-Ampt infiltration
ga_regen.inp	Infiltration regeneration (redistribution)

the theory behind the system behavior, the parts of the model under review, critical output and non-critical output. The test cases and the associated fact sheets will be released with the User's Manual.

In regression testing, a combination of hydrographs, regression plots, and statistical tables are used to evaluate the results.

Figure 17.13 is a hydrograph comparing the results of SWMM 4.4 and SWMM 5 with the same input data set. This hydrograph demonstrates that the simulation in SWMM 5 is performing the same as in SWMM44. The dots are the results from the SWMM44 output file. The line represents the results from the SWMM5 output file.

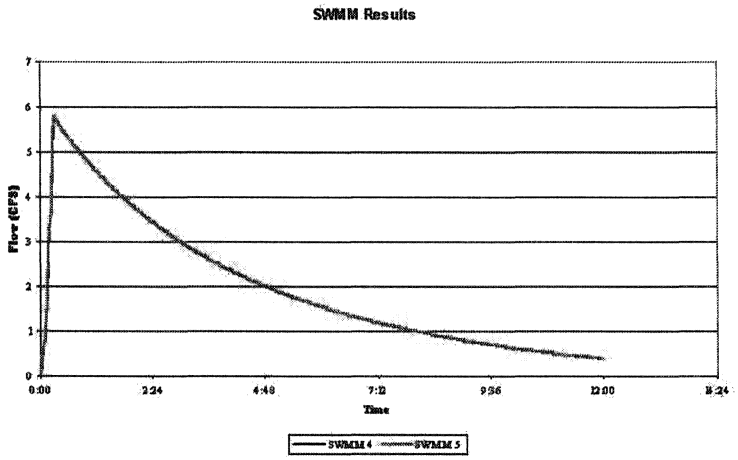


Figure 17.13 SWMM 4.4 and SWMM 5 results comparison.

Figure 17.14 is a regression plot that shows the SWMM 5 simulation shares the same behavior and characteristics as seen with SWMM 4.4.

With paired data, the relative difference between results may be compared each time.

$$\frac{|SWMM\,4 - SWMM\,5|}{SWMM\,4} \le C, \quad for\ SWMM\,4 \ge C$$

$$SWMM\,5 \le C, \ for\ SWMM\,4 < C$$

where:

C = 0.005

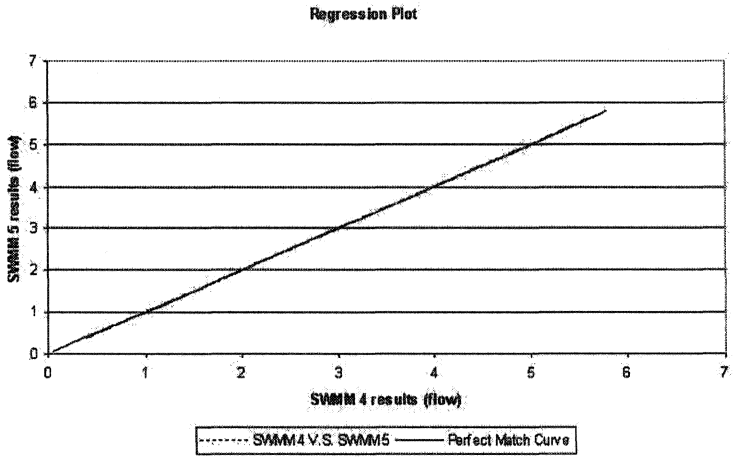


Figure 17.14 Regression test.

If test results are poor, further investigation is necessary. In some cases, the difference between SWMM 5 and the previous SWMM engine is due to a new method/technique introduced in SWMM 5. These differences will be explained in the User's Manual.

Acceptance Test

The acceptance test runs the same battery of test cases as the regression tests through the computational engine. The QA Manager will perform the acceptance tests only after the majority of the units or modules are functional. The results of the acceptance tests are sometimes unusual, but the results will show which modules transmitted values that initiated errors in other modules. The QA Manager will report results to the development team, and the QA manager will document all results in the version control system.

Data generated from the QA testing should present the same information for both SWMM version 4.4 and SWMM version 5.0. Different applications of SWMM have different needs, so it is up to the individual application developers to determine acceptance testing standards.

Data collected for QA will describe performance for a suite of core functions available in SWMM 5.0. The same functions will be tested in SWMM 4.4. Comparison with SWMM 4.4 provides users with information about the relative impact of the upgrade on performance of individual functions.

System Test

To facilitate early detection of inconsistencies or problems with the GUI, the system testing will be ongoing throughout the development. During system testing, each function of the software available through the GUI will be tested. The QA manager will develop the test SOP from the design of the GUI. The QA manager will present any negative results from this test to the development team for review. An acceptable test (no defects found) will move the code from pre-beta into beta. Given that perfect code is not really attainable, minor defects will be documented in the version control system as a "bug fix" list. Minor defects include aesthetic concerns, labeling, formatting, colors, graphics, unexpected, or non-intuitive labeling. Major defects include system crashes, long delays, incorrect labeling, or missing functions.

Beta Testing

Once the system test produces relatively error free code, SWMM 5.0 will move into beta-testing. Beta testing will occur in two phases: the Review Panel Phase and the Community Group Phase. At this point the software will be presented to the members of the Review Panel. These individuals will test the software and report any problems or errors that they encounter.

The second phase of Beta testing will present the software to the SWMM community for testing. This project will utilize an existing community list server with about 400 subscribers created and moderated by Professor William James of the University of Guelph. The community list is available at http://www.chi.on.ca/swmmusers.html. This community has a long history of intense involvement in the development and testing for previous versions of SWMM. When SWMM 5.0 goes into beta testing, EPA will consider making the version control system publicly accessible through a free web-based change request tracking tool. This will make all bug tracking and release testing a community process.

17.5 Conclusions

Prior to the public release of the SWMM software, EPA and CDM will work extensively with the SWMM data converter and testing of the SWMM 5 software. The goal of the data converter is to extend the lives of the historic SWMM models by providing an easy, automatic switch from the old to the new engine. Millions of US dollars have been invested in the SWMM models

throughout the world during the past 30 years. There is neither time nor money to rebuild these historical models to accommodate the new engine. The goal of the testing of SWMM 5 is to ensure the results from the new engine are consistent with the existing engine. Six levels of testing have been applied during the creation of the new engine. Users are invited and encouraged to make improvement in the data converter, test the SWMM 5 engine thoroughly, and to report errors back to EPA. With the long history in the SWMM community of intense involvement in the development and testing for previous versions of SWMM, the SWMM 5 software will continue to improve with the addition of new features.

References

Huber, W. C. and R. E. Dickinson (1988). Storm Water Management Model - Version 4: User's Manual; Cooperative Agreement CR-811607; U.S. EPA; Athens, Georgia.

Roesner, L.A., R.E. Dickinson and J.A.Aldrich (1988) Storm Water Management Model - Version 4: User's Manual - Addendum I EXTRAN; Cooperative Agreement CR-811607; U.S.EPA; Athens, Georgia.

Schade, T. (2002). Quality Assurance Project Plan for SWMM Redevelopment; U.S.EPA Office of Research and Development - Water Supply and Water Resources Division; Cincinnati, Ohio; November 27, 2002.

U.S. EPA (2002). SWMM REDEVELOPMENT PROJECT PLAN - Version 5;
U.S.EPA Office of Research and Development - Water Supply and Water Resources Division; Cincinnati, Ohio; February 19, 2002.

U.S. EPA (2003). SWMM 5 INPUT DATA FORMATS; U.S.EPA Office of Research and Development - Water Supply and Water Resources Division; Cincinnati, Ohio; June 18, 2003.