

SWMM (EPA) adapter

Contents

- [Contents](#)
 - [SWMM pre-adapter](#)
 - [Properties](#)
 - [Notes for users](#)
 - [PI XML run file example](#)
 - [SWMM post-adapter](#)
 - [Properties](#)
 - [Notes for users](#)
 - [System requirements](#)
 - [GeneralAdapterRun Example Configuration](#)
 - [General](#)
 - [Start-up activities](#)
 - [Export activities](#)
 - [Execute activities](#)
 - [Import activities](#)
 - [PI XML Run file example](#)

This page documents the Delft-FEWS EPA-SWMM adapter required for running SWMM (Storm Water Management Model) models from Delft-FEWS. The adapter supports SWMM model version 5.1. For information about the SWMM model see <https://www.epa.gov/water-research/storm-water-management-model-swmm>

The adapter consists of separate pre- and post adapters that can be used to import and export timeseries data to/from the SWMM model and control the simulation period. The FEWS general adapter is used to configure both the pre- and post adapter and execute the SWMM simulation using the SWMM5 command line executable. This model adapter is written in java. Note there is an open-source python based adapter in development as well, developed by Matrix Solutions inc. More information about that adapter will be provided [here](#).

SWMM pre-adapter

The SWMM pre-adapter converts the pi-xml files from Delft-FEWS to model native input files.

Usage: SWMMPreAdapter <PI XML run file pathname relative to current working directory>

Class name: nl.deltares.fews.swmm.SwmmPreAdapter

Properties

SWMM_input_file	(required)	Pathname of SWMM input file (*.inp) to update. This should be either an absolute path or a path relative to the workDir specified in the netcdf run file. The input file will be updated to modify only the START_DATE, START_TIME, END_DATE, END_TIME, REPORT_START_DATE and REPORT_START_TIME fields, the rest of the file is not modified in any way.
------------------------	------------	---

Notes for users

- For all files that are written by this adapter, if the file to be written already exists, then it will be overwritten.
- This program can process timeseries input data per catchment, node or link model entity, and requires that each model entity has a unique ID, regardless of its entity type.
- This program assumes that the SWMM model configuration uses the same time zone as given in the PI XML run file.
- The working directory (workDir) defined in the PI XML run file is the directory where the SWMM model will be run, the SWMM input files will be stored in this directory.
- This program writes log messages to a log file called "swmm_pre_adapter_log.txt" in the workDir specified in the PI XML run file.
- This program uses the information in the specified PI XML run file as input and uses this information to perform the following actions:
 1. Update the start and end time for the simulation.
The SWMM input file (*.inp) file specified in the property "SWMM_input_file" will be updated to reflect the start and end date / time for the simulation as specified in the PI XML run file. The adapter will only modify the START_DATE, START_TIME, END_DATE, END_TIME, REPORT_START_DATE and REPORT_START_TIME fields where the report start date/time will be the same as the start of the simulation.
 2. Convert input PI XML timeseries files to SWMM timeseries input file format files:
For each of the input timeseries files listed in the PI XML run file, one or more SWMM compatible timeseries file will be written to the working directory as follows:
 - For each of the timeseries present in the PI XML input file, the location ID is looked up in the [TIMESERIES] section of the SWMM model input file (*.INP) to get the FILE name listed there.
 - A compatible timeseries file (see the SWMM5 user manual section 11.6) is written for each location in the working directory.

PI XML run file example

runinfo.xml

```
<startUpActivities>
  <purgeActivity>
    <filter>%ROOT_DIR%/input/*. *</filter>
  </purgeActivity>
  <purgeActivity>
    <filter>%ROOT_DIR%/output/*. *</filter>
  </purgeActivity>
  <purgeActivity>
    <filter>%ROOT_DIR%/model/*. *</filter>
  </purgeActivity>
</startUpActivities>
```

SWMM post-adapter

The SWMM post-adapter converts the native model output to pi-xml files to be imported by Delft-FEWS.

Usage: SWMMPostAdapter <PI XML run file pathname relative to current working directory>

Class name: nl.deltares.fews.swmm.SwmmPostAdapter

Properties

SWMM_report_file	(required)	Pathname of SWMM report file (*.rpt) to update. This should be either an absolute path or a path relative to the workDir specified in the netcdf run file.
-------------------------	------------	--

Notes for users

- For all files that are written by this adapter, if the file to be written already exists, then it will be overwritten.
- This program can process timeseries output data per catchment, node or link model entity, and requires that each model entity has a unique ID, regardless of its entity type.
- The General adapter must be configured using the **exportPlaceholderFile** option, to provide a template file for each timeseries output file listed in the PI XML run file. These files will be overwritten with actual model output after each model run.
- The SWMM output report must be properly configured to contain the required output time series (see the SWMM5 user manual section 8.2)
- This program assumes that the SWMM model configuration uses the same time zone as given in the PI XML run file.
- The working directory (workDir) defined in the PI XML run file is the directory where the SWMM model will be run, the SWMM report files is expected to be found in this directory.
- This program writes log messages to a log file called "swmm_post_adapter_log.txt" in the workDir specified in the PI XML run file.
- This program uses the information in the specified PI XML run file as input and uses this information to perform the following actions:
 1. Scan the report file for log messages (Error or Warning) and transfer these to the post-adapter log file in FEWS format.
 2. Convert output scalar time series per configured location and parameter as follows:
 - For each of the timeseries headers in the configured 'placeholder' PI XML file listed in the run file, the location ID and parameter ID are used
 - The report file is scanned to find the timeseries output for the requested location, and the column for the requested parameter to parse the timeserie output
 - The 'template' PI XML file generated by the General adapter is overwritten by the timeseries output data.

System requirements

- This program needs Java version 11 or higher.
- This program needs the following Java libraries:
 - commons-httpclient-3.0.1.jar
 - Delft_FEWS.jar
 - Delft_Util.jar
 - log4j-1.2.14.jar
 - netcdf-4.2.jar
 - slf4j-api-1.5.6.jar
 - slf4j-log4j12-1.5.6.jar
 - TimeSeriesImport.jar

GeneralAdapterRun Example Configuration

The following gives an example of how to set up the GeneralAdapterRun file for SWMM in FEWS using the SWMM model pre and post-adapters. The GeneralAdapterRun file follows the general structure as described [here](#).

General

In this section general information regarding the module such as version number, file directories, missing values, and time zone information can be specified.

general

```
<general>
  <rootDir>$REGION_HOME$/Modules/swmm/$MINE$/rootDir>
  <workDir>%ROOT_DIR%/model</workDir>
  <exportDir>%ROOT_DIR%/input</exportDir>
  <exportDataSetDir>%ROOT_DIR%/model</exportDataSetDir>
  <exportIdMap>IdExportSWMM</exportIdMap>
  <importDir>%ROOT_DIR%/output</importDir>
  <importIdMap>IdImportSWMM</importIdMap>
  <dumpFileDir>$GA_DUMPFILEDIR$/dumpFileDir>
  <dumpDir>%ROOT_DIR%/dumpDir>
  <diagnosticFile>%ROOT_DIR%/piDiagnostic.xml</diagnosticFile>
  <missVal>-99</missVal>
  <convertDatum>>true</convertDatum>
  <timeZone>
    <timeZoneOffset>+10:00</timeZoneOffset>
  </timeZone>
</general>
```

Start-up activities

It may be useful to clear the model working directory of any previous runs before starting a new run.

startUpActivities

```
<startUpActivities>
  <purgeActivity>
    <filter>%ROOT_DIR%/input/*. *</filter>
  </purgeActivity>
  <purgeActivity>
    <filter>%ROOT_DIR%/output/*. *</filter>
  </purgeActivity>
  <purgeActivity>
    <filter>%ROOT_DIR%/model/*. *</filter>
  </purgeActivity>
</startUpActivities>
```

Export activities

In this section the data to be exported from FEWS as input to the module is specified. Data to export to SWMM generally includes:

- Input timeseries data (i.e. rainfall, flow, water levels, evaporation)
- Run file (i.e. start and end time for simulation)

The run file contains information regarding the input file names, start and stop times, and time step. Additional properties can be passed using the run file as listed above under Properties.

exportActivities

```
<exportActivities>
  <exportTimeSeriesActivity>
    <exportFile>rain.xml</exportFile>
    <timeSeriesSets>
      <timeSeriesSet>
        <moduleInstanceId>$SWMM_MODEL$SWMMProcess</moduleInstanceId>
        <valueType>scalar</valueType>
        <parameterId>P.obs</parameterId>
        <locationSetId>stations.P.$MINE$/locationSetId>
        <timeSeriesType>simulated historical</timeSeriesType>
        <timeStep id="day_AEST"/>
        <relativeViewPeriod unit="day" start="-366" end="0" startOverrutable="true"/>
        <readWriteMode>add originals</readWriteMode>
      </timeSeriesSet>
    </timeSeriesSets>
  </exportTimeSeriesActivity>
</exportActivities>
```

```

        </timeSeriesSet>
    </timeSeriesSets>
</exportTimeSeriesActivity>
<exportTimeSeriesActivity>
    <exportFile>evaporation.xml</exportFile>
    <timeSeriesSets>
        <timeSeriesSet>
            <moduleInstanceId>$$SWMM_MODEL$SWMMProcess</moduleInstanceId>
            <valueType>scalar</valueType>
            <parameterId>E.pot</parameterId>
            <locationSetId>stations.P.$MINE$</locationSetId>
            <timeSeriesType>simulated historical</timeSeriesType>
            <timeStep id="day_AEST"/>
            <relativeViewPeriod unit="day" start="-366" end="0" startOverrulable="true"/>
            <readWriteMode>add originals</readWriteMode>
        </timeSeriesSet>
    </timeSeriesSets>
</exportTimeSeriesActivity>
<exportTimeSeriesActivity>
    <exportFile>level.xml</exportFile>
    <timeSeriesSets>
        <timeSeriesSet>
            <moduleInstanceId>$$SWMM_MODEL$SWMMProcess</moduleInstanceId>
            <valueType>scalar</valueType>
            <parameterId>H.obs</parameterId>
            <locationSetId>stations.ml.$MINE$.H</locationSetId>
            <timeSeriesType>simulated historical</timeSeriesType>
            <timeStep id="month_AEST"/>
            <relativeViewPeriod unit="day" start="-366" end="0" startOverrulable="true"/>
            <readWriteMode>add originals</readWriteMode>
        </timeSeriesSet>
    </timeSeriesSets>
</exportTimeSeriesActivity>
<exportTimeSeriesActivity>
    <exportFile>flow.xml</exportFile>
    <timeSeriesSets>
        <timeSeriesSet>
            <moduleInstanceId>$$SWMM_MODEL$SWMMProcess</moduleInstanceId>
            <valueType>scalar</valueType>
            <parameterId>Q.ml</parameterId>
            <locationSetId>stations.ml.$MINE$.Q</locationSetId>
            <timeSeriesType>simulated historical</timeSeriesType>
            <timeStep id="day_AEST"/>
            <relativeViewPeriod unit="day" start="-366" end="0" startOverrulable="true"/>
            <readWriteMode>add originals</readWriteMode>
        </timeSeriesSet>
    </timeSeriesSets>
</exportTimeSeriesActivity>
<exportDataSetActivity>
    <moduleInstanceId>$$MODULE_INSTANCE_ID$</moduleInstanceId>
</exportDataSetActivity>
<exportRunFileActivity>
    <exportFile>runinfo.xml</exportFile>
    <properties>
        <string key="swmm_input_file" value="%ROOT_DIR%/model/$SWMM_INP_FILE$"/>
        <string key="swmm_report_file" value="%ROOT_DIR%/model/$SWMM_RPT_FILE$"/>
    </properties>
</exportRunFileActivity>
</exportActivities>

```

Execute activities

This section calls the SWMM pre and post-adapters as well as the SWMM executable. Note: the run file must be passed as an argument to the SWMM pre and post-adapters.

executeActivities

```
<executeActivities>
  <executeActivitys>
    <description>Run SWMM pre</description>
    <command>
      <className>nl.deltares.fews.swmm.SwmmPreAdapter</className>
      <binDir>${SWMM_BIN}</binDir>
      <moduleDataSetName>UpdateSoftwareSWMM</moduleDataSetName>
    </command>
    <arguments>
      <argument>%ROOT_DIR%/input/runinfo.xml</argument>
    </arguments>
    <logFile>
      <file>%ROOT_DIR%/input/swmm_pre_adapter_log.txt</file>
      <infoLinePattern>* INFO *</infoLinePattern>
      <warningLinePattern>* WARNING *</warningLinePattern>
      <errorLinePattern>* ERROR *</errorLinePattern>
    </logFile>
    <timeOut>3600000</timeOut>
    <ignoreDiagnostics>true</ignoreDiagnostics>
  </executeActivity>
  <executeActivity>
    <description>Run SWMM</description>
    <command>
      <executable>${SWMM_BIN}/swmm5.exe</executable>
    </command>
    <arguments>
      <argument>%ROOT_DIR%/model/${SWMM_INP_FILE}</argument>
      <argument>%ROOT_DIR%/model/${SWMM_RPT_FILE}</argument>
    </arguments>
    <timeOut>3600000</timeOut>
    <ignoreDiagnostics>true</ignoreDiagnostics>
  </executeActivity>
  <executeActivity>
    <description>Run SWMM post</description>
    <command>
      <className>nl.deltares.fews.swmm.SwmmPostAdapter</className>
      <binDir>${SWMM_BIN}</binDir>
    </command>
    <arguments>
      <argument>%ROOT_DIR%/input/runinfo.xml</argument>
    </arguments>
    <logFile>
      <file>%ROOT_DIR%/input/swmm_post_adapter_log.txt</file>
      <infoLinePattern>* INFO *</infoLinePattern>
      <warningLinePattern>* WARNING *</warningLinePattern>
      <errorLinePattern>* ERROR *</errorLinePattern>
    </logFile>
    <timeOut>3600000</timeOut>
    <ignoreDiagnostics>true</ignoreDiagnostics>
  </executeActivity>
</executeActivities>
```

Import activities

In this section the data to be imported into FEWS as output from the module is specified. Data to import from SWMM generally includes:

- Output timeseries data (water levels, flows)

importActivities

```
<importActivities>
  <importTimeSeriesActivity>
    <exportPlaceholderFile>true</exportPlaceholderFile>
    <importFile>outputswmm_links.xml</importFile>
    <timeSeriesSets>
      <timeSeriesSet>
        <moduleInstanceId>$MODULE_INSTANCE_ID$</moduleInstanceId>
        <valueType>scalar</valueType>
        <parameterId>Q.gw</parameterId>
        <locationSetId>stations.ml.$MINE$.H</locationSetId>
        <timeSeriesType>simulated historical</timeSeriesType>
        <timeStep id="day_AEST"/>
        <readWriteMode>add originals</readWriteMode>
      </timeSeriesSet>
    </timeSeriesSets>
  </importTimeSeriesActivity>
</importActivities>
```

PI XML Run file example

Typically, the general adapter needs to export only one PI XML run file for use by both adapters, as in the below example:

runinfo.xml

```
<Run xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://www.wldelft.nl/fews/PI" xsi:
schemaLocation="http://www.wldelft.nl/fews/PI http://fews.wldelft.nl/schemas/version1.0/pi-schemas/pi_run.xsd"
version="1.5">
  <timeZone>10.0</timeZone>
  <startDateTime date="2019-05-01" time="00:00:00"/>
  <endDateTime date="2020-05-01" time="00:00:00"/>
  <time0 date="2020-05-01" time="00:00:00"/>
  <lastObservationDateTime date="2020-05-01" time="00:00:00"/>
  <workDir>D:\FEWS\REGIONHOME\Modules\swmm\model</workDir>
  <inputTimeSeriesFile>D:\FEWS\REGIONHOME\Modules\swmm\input\rain.xml</inputTimeSeriesFile>
  <inputTimeSeriesFile>D:\FEWS\REGIONHOME\Modules\swmm\input\evaporation.xml</inputTimeSeriesFile>
  <inputTimeSeriesFile>D:\FEWS\REGIONHOME\Modules\swmm\input\level.xml</inputTimeSeriesFile>
  <inputTimeSeriesFile>D:\FEWS\REGIONHOME\Modules\swmm\input\flow.xml</inputTimeSeriesFile>
  <outputDiagnosticFile>D:\FEWS\REGIONHOME\Modules\swmm\piDiagnostic.xml</outputDiagnosticFile>
  <outputTimeSeriesFile>D:\FEWS\REGIONHOME\Modules\swmm\output\outputswmm_links.xml</outputTimeSeriesFile>
  <properties>
    <string key="swmm_input_file" value="$WORKDIR$/model/SWC.inp"/>
    <string key="swmm_report_file" value="$WORKDIR$/model/SWC.rpt"/>
  </properties>
</Run></startUpActivities>
```

Please note that the <outputDiagnosticFile> element is required, even though it is not actually used in this configuration.