# Basic conception in SWMM

**Surface runoff:**

The conceptual view of surface runoff used by SWMM is illustrated in Figure 3-7 below. Each subcatchment surface is treated as a **nonlinear reservoir**. Inflow comes from precipitation and any designated upstream subcatchments. There are several outflows, including infiltration, evaporation, and surface runoff. **The capacity of this "reservoir" is the maximum depression storage, which is the maximum surface storage provided by ponding, surface wetting, and interception.** Surface runoff per unit area, **Q, occurs only when the depth of water in the "reservoir" exceeds the maximum depression storage**, ds, in which case the outflow is given by Manning's equation. Depth of water over the subcatchment (d) is continuously updated with time by solving numerically a water balance equation over the subcatchment.

****

**Infiltration:**

Horton, Modified Horton, Green-Ampt, Modified Green-Ampt, Curve Number(SCS) Method

**Groundwater:**

It is a two-zone groundwater model in SWMM. The upper zone is unsaturated with a variable moisture content of θ. The lower zone is fully saturated and therefore its moisture content is fixed at the soil porosity φ.

**Snowmelt:**

It updates the state of the snow packs associated with each subcatchment by accounting for snow accumulation, snow redistribution by areal depletion and removal operations, and snow melt via heat budget accounting.

**Flow routing:**

Flow routing within a conduit (pipes or channels) link in SWMM is governed by the conservation of mass and momentum equations for gradually varied, unsteady flow (i.e., the Saint Venant flow equations).

* Steady Flow Routing
* Kinematic Wave Routing
* Dynamic Wave Routing

Flooding occurs when the water depth at a node exceeds the maximum available depth, and the excess flow is either lost from the system or can pond atop the node and re-enter the drainage system.

Nonlinear reservoir routing of overland flow (codes only reflects he nonlinear reservoir for runoff generation)

# Visual Objects in SWMM

**Subcatchments:**

Subcatchments are hydrologic units of land whose topography and drainage system elements direct surface runoff to a single discharge point. The user is responsible for dividing a study area into an appropriate number of subcatchments, and for identifying the outlet point of each subcatchment. Discharge outlet points can be either nodes of the drainage system or other subcatchments.

**Nodes：**

**Junctions** are drainage system nodes where links join together. Physically they can represent the confluence of natural surface channels, manholes in a sewer system, or pipe connection fittings. External inflows can enter the system at junctions. Excess water at a junction can become partially pressurized while connecting conduits are surcharged and can either be lost from the system or be allowed to pond atop the junction and subsequently drain back into the junction.

**Outfalls,** are terminal nodes of the drainage system used to define final downstream boundaries under Dynamic Wave flow routing. For other types of flow routing they behave as a junction. Only a single link can be connected to an outfall node, and the option exists to have the outfall discharge onto a subcatchment's surface.

**Dividers,** are drainage system nodes that divert inflows to a specific conduit in a prescribed manner. A flow divider can have no more than two conduit links on its discharge side. Flow dividers are only active under Steady Flow and Kinematic Wave routing and are treated as simple junctions under Dynamic Wave routing.

**Storage Units (basin or lake),** are drainage system nodes that provide storage volume. Physically they could represent storage facilities as small as a catch basin or as large as a lake.

**Conduits,** are pipes or channels that move water from one node to another in the conveyance system.

**Pumps,** are links used to lift water to higher elevations. A pump curve describes the relation between a pump's flow rate and conditions at its inlet and outlet nodes. Five different types of pump curves are supported.

**Orifices** , are used to model outlet and diversion structures in drainage systems, which are typically openings in the wall of a manhole, storage facility, or control gate. They are internally represented in SWMM as a link connecting two nodes. An orifice can have either a circular or rectangular shape, be located either at the bottom or along the side of the upstream node, and have a flap gate to prevent backflow.

**Weirs,** like orifices, are used to model outlet and diversion structures in a drainage system. Weirs are typically located in a manhole, along the side of a channel, or within a storage unit. They are internally represented in SWMM as a link connecting two nodes, where the weir itself is placed at the upstream node. A flap gate can be included to prevent backflow.

**Outlets**, are flow control devices that are typically used to control outflows from storage units. They are used to model special head-discharge relationships that cannot be characterized by pumps, orifices, or weirs. Outlets are internally represented in SWMM as a link connecting two nodes. An outlet can also have a flap gate that restricts flow to only one direction.

# Example

Ex1



Ex2



# SWMM code analysis

## Key framework:

swmm\_run->

swmm\_open

swmm\_start //Initialization

do{

swmm\_step (swmm5.c)

execRouting (swmm5.c)

**runoff\_execute (runoff.c)**

**routing\_execute (routing.c)**

} while ()

swmm\_end

swmm\_close

## Routing code structure:

routing\_execute (routing.c)

flowrout\_execute (flowrout.c): routes flow through conveyance network over current time step

**Over land flow**= Volume/tStep (this tStep is from routing\_getRoutingStep in routing.c)

**dynwave\_execute** (dynwave.c): routes flows through drainage network over current time step

findLinkFlows (dynwave.c): find new flow in link

dwflow\_findConduitFlow (dwflow.c): update flow in conduit link

## Overland flow routing code:

**Over land flow**= Volume/tStep=q

How to get tSetp?

**Answer:**

routing\_getRoutingStep (Routing.c) -> flowrout\_getRoutingStep (flowrout.c) -> dynwave\_getRoutingStep (dynawave.c)