Take your Stormwater Modelling to next level

with PySWMM and the Open Water Analytics initiative

Gonzalo Andrés Peña-Castellanos
@goanpeca

PyCon Colombia
EAFIT, Medellín
February 9th, 2018

Content

- About
- The (Urban) Water Cycle
- On Modelling… (How, What, Why, How)
- On Hydrology, Hydraulics and Water Quality

Intermission: on the perils of CFD

- EPA - SWMM

Intermission: networking, lying and coding

- OWA
- SWMM API
- PySWMM API
- Expanded PySWMM API (Under Construction)
- Roadmap and future work

About @goanpeca

Civil Engineer

SIHSA
SISTEMAS HIDRÁULICOS Y SANITARIOS

- MSc Hydroinformatics (2010)
- MSc Sanitary Engineering (2012)
- PhD dropout (2015)

About @goanpeca

Civil Engineer

'Software Engineer'

SIHSA
SISTEMAS HIDRÁULICOS Y SANITARIOS

- MSc Hydroinformatics (2010)
- MSc Sanitary Engineering (2012)
- PhD dropout (2015)

ANACONDA®

- Python developer since 2009

Spyder IDE Core Developer since 2014 http://spyder-ide.org

PyBee Project Area leader since 2017 https://pybee.org

About @goanpeca

Civil Engineer

'Software Engineer'

SIHSA
SISTEMAS HIDRÁULICOS Y SANITARIOS

- MSc Hydroinformatics (2010)
- MSc Sanitary Engineering (2012)
- PhD dropout (2015)

ANACONDA®

- Python developer since 2009

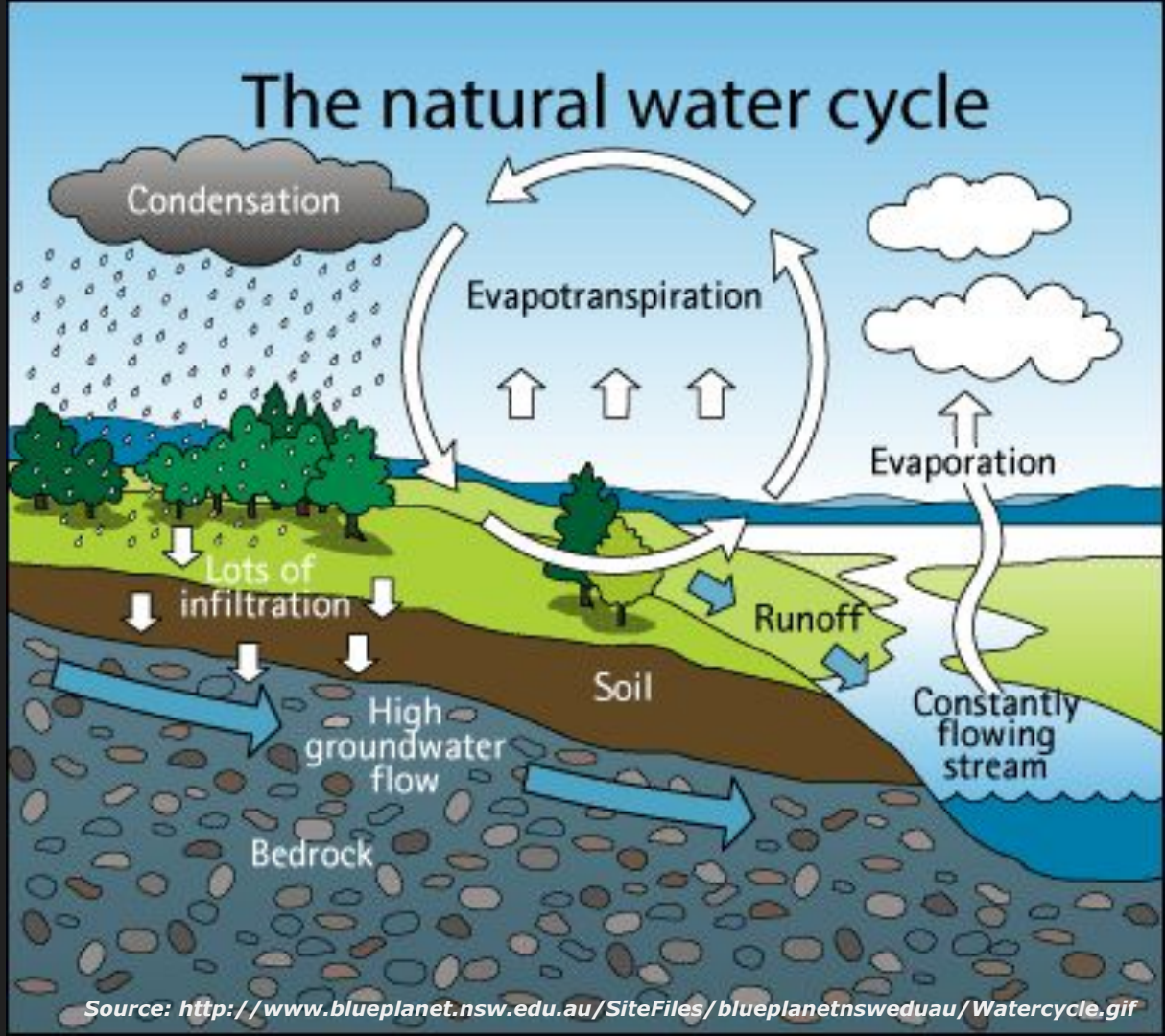
Spyder IDE Core Developer since 2014 http://spyder-ide.org

PyBee Project Area leader since 2017 https://pybee.org

PySWMM core developer since 2017

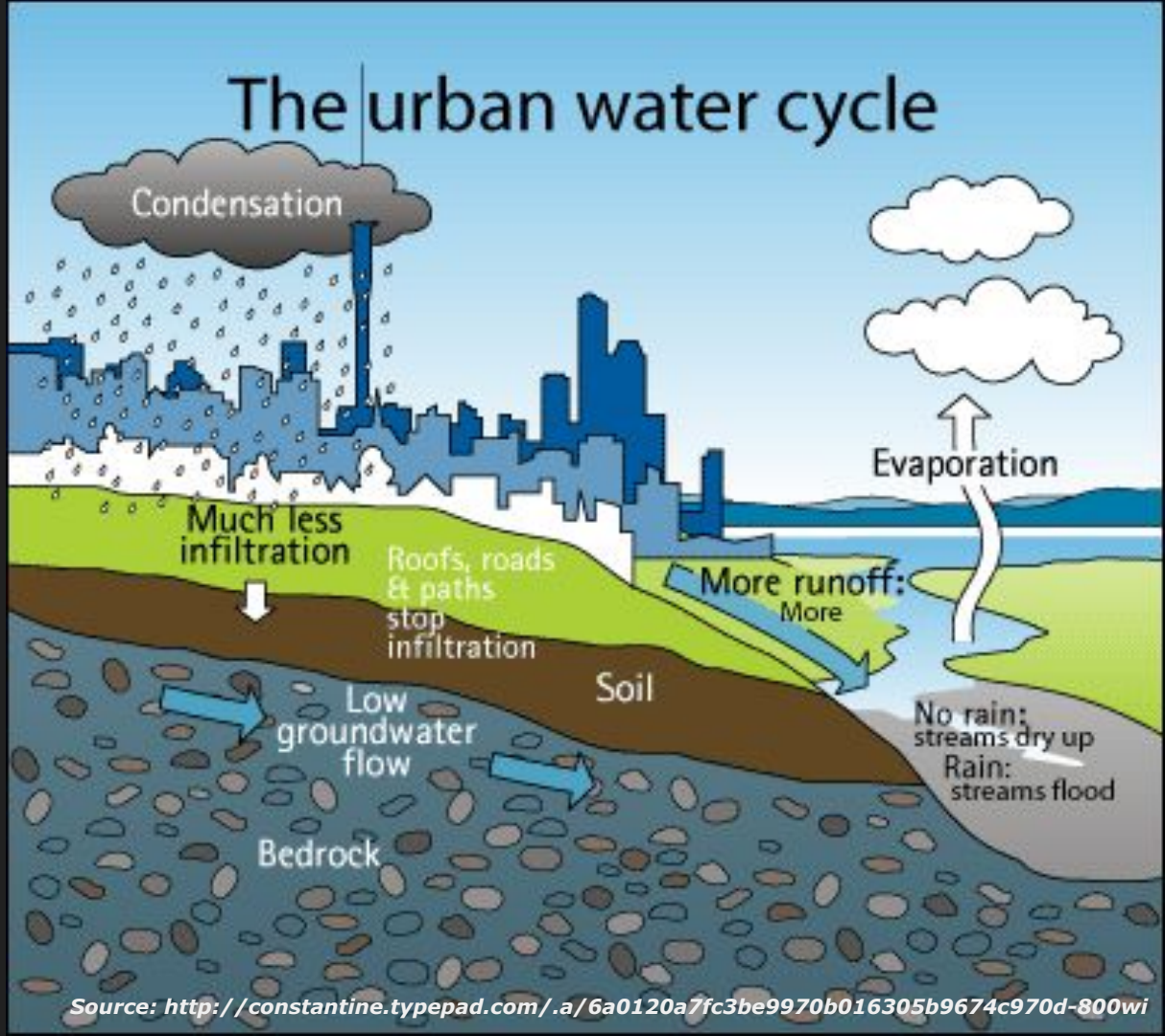
The Water Cycle

- Different processes
- Different flows and interactions



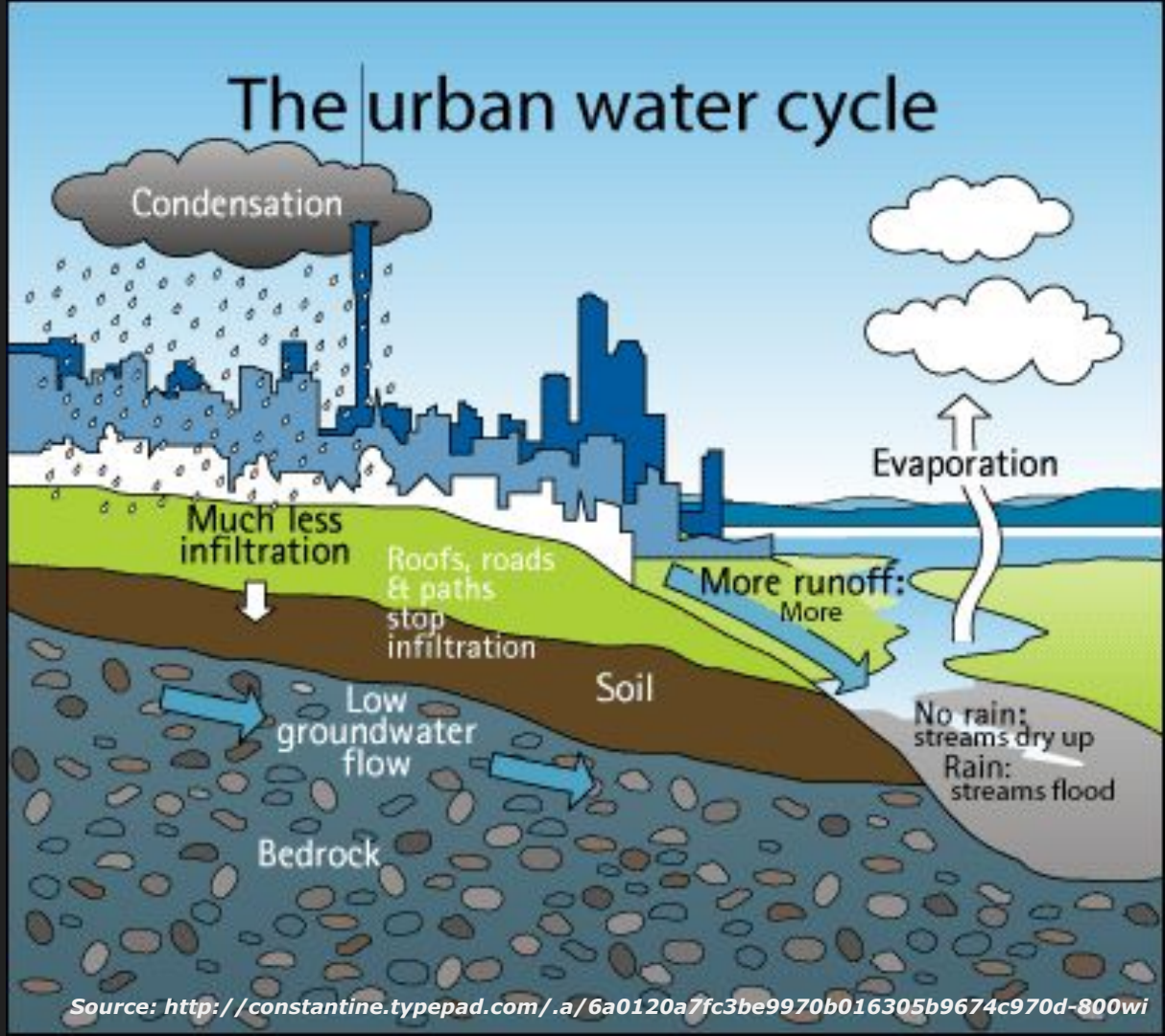
The Urban Water Cycle

Similar as the water cycle we were taught at school, but with humans messing things up with dams, pipes, roads, crops and varying loadings of

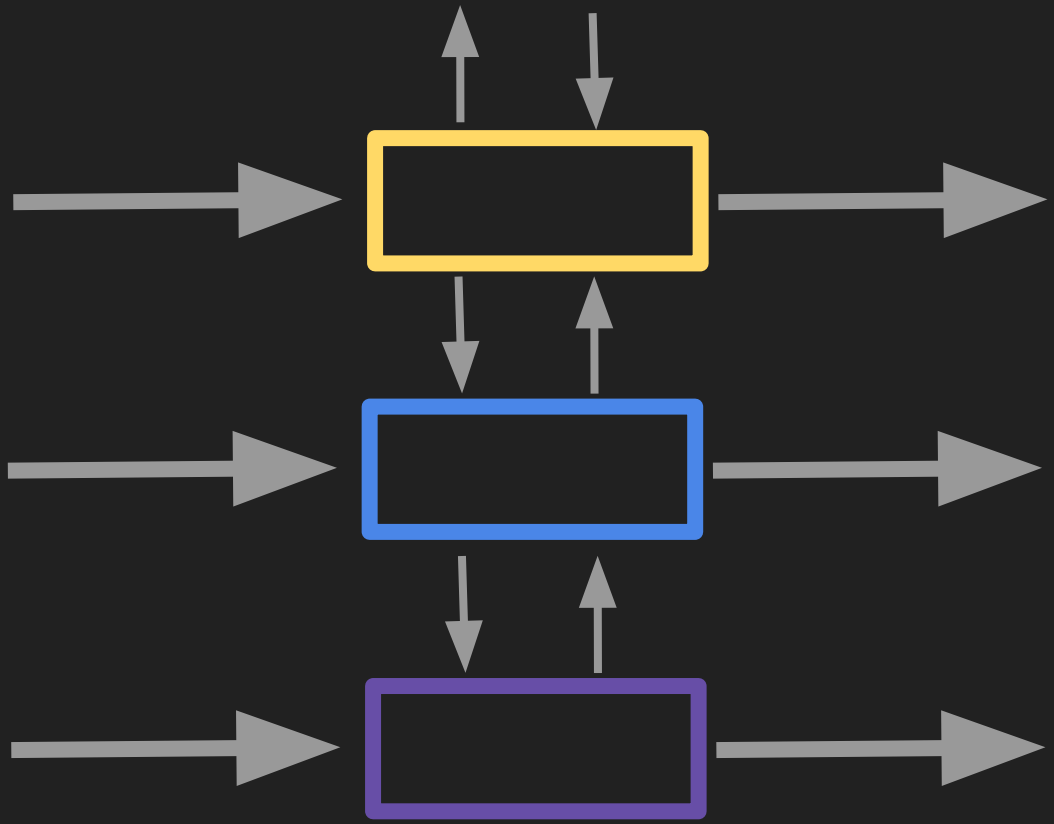


The Urban Water Cycle

Similar as the water cycle we were taught at school, but with humans messing things up with dams, pipes, roads, crops and varying loadings of



How do we model this system?



What is a model?

Is a way to understand, define, quantify, visualize reality by referencing to existing and usually commonly accepted knowledge.

<Insert Pipe Here>

What is a model?

Is a way to understand, define, quantify, visualize reality by referencing to existing and usually commonly accepted knowledge.

What is a model?

Is a way to understand, define, quantify, visualize reality by referencing to existing and usually commonly accepted knowledge.

Source: https://en.wikipedia.org/wiki/File:MagrittePipe.jpg

The Treachery of Images

(This is not a pipe)

René Magritte

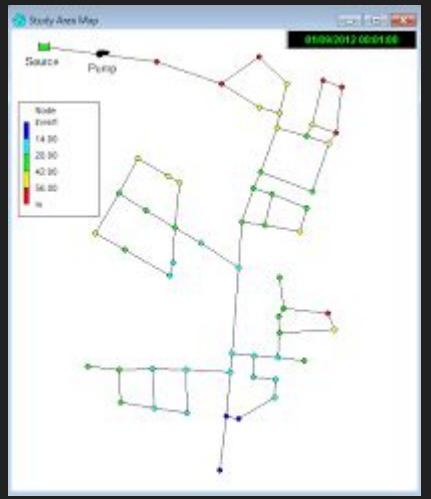
Why model urban water cycle?

- Evaluate influence of climate change

Source: https://www.epa.gov/water-research/storm-water-management-model-swmm
https://news.nationalgeographic.com/content/dam/news/2017/02/12/conservative-climate-change/

Why model urban water cycle?

- Evaluate influence of climate change
- Designing and sizing of drainage system components



Source: https://www.epa.gov/water-research/storm-water-management-model-swmm
https://news.nationalgeographic.com/content/dam/news/2017/02/12/conservative-climate-change/

Why model urban water cycle?

- Evaluate influence of climate change
- Designing and sizing of drainage system components
- Sizing detention facilities

Source: https://www.epa.gov/water-research/storm-water-management-model-swmm
https://news.nationalgeographic.com/content/dam/news/2017/02/12/conservative-climate-change/

Why model urban water cycle?

- Evaluate influence of climate change
- Designing and sizing of drainage system components
- Sizing detention facilities
- Mapping flood plains of natural channel systems

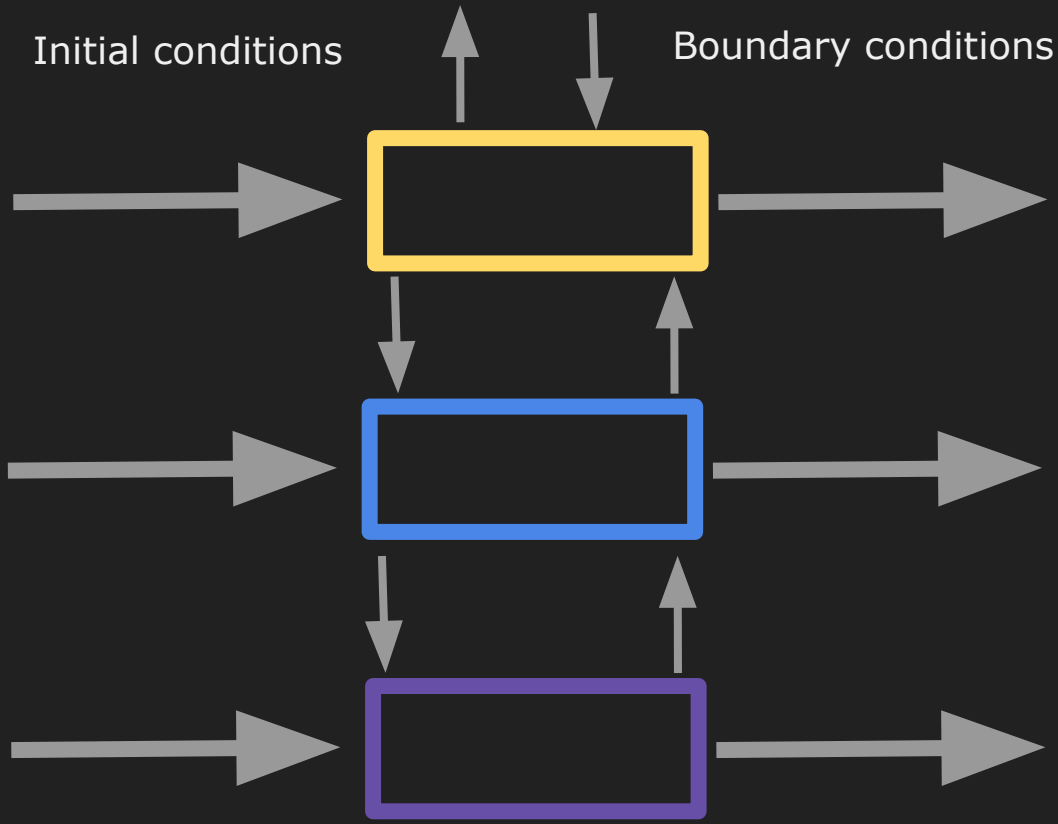
Source: https://www.epa.gov/water-research/storm-water-management-model-swmm
https://news.nationalgeographic.com/content/dam/news/2017/02/12/conservative-climate-change/

Why model urban water cycle?

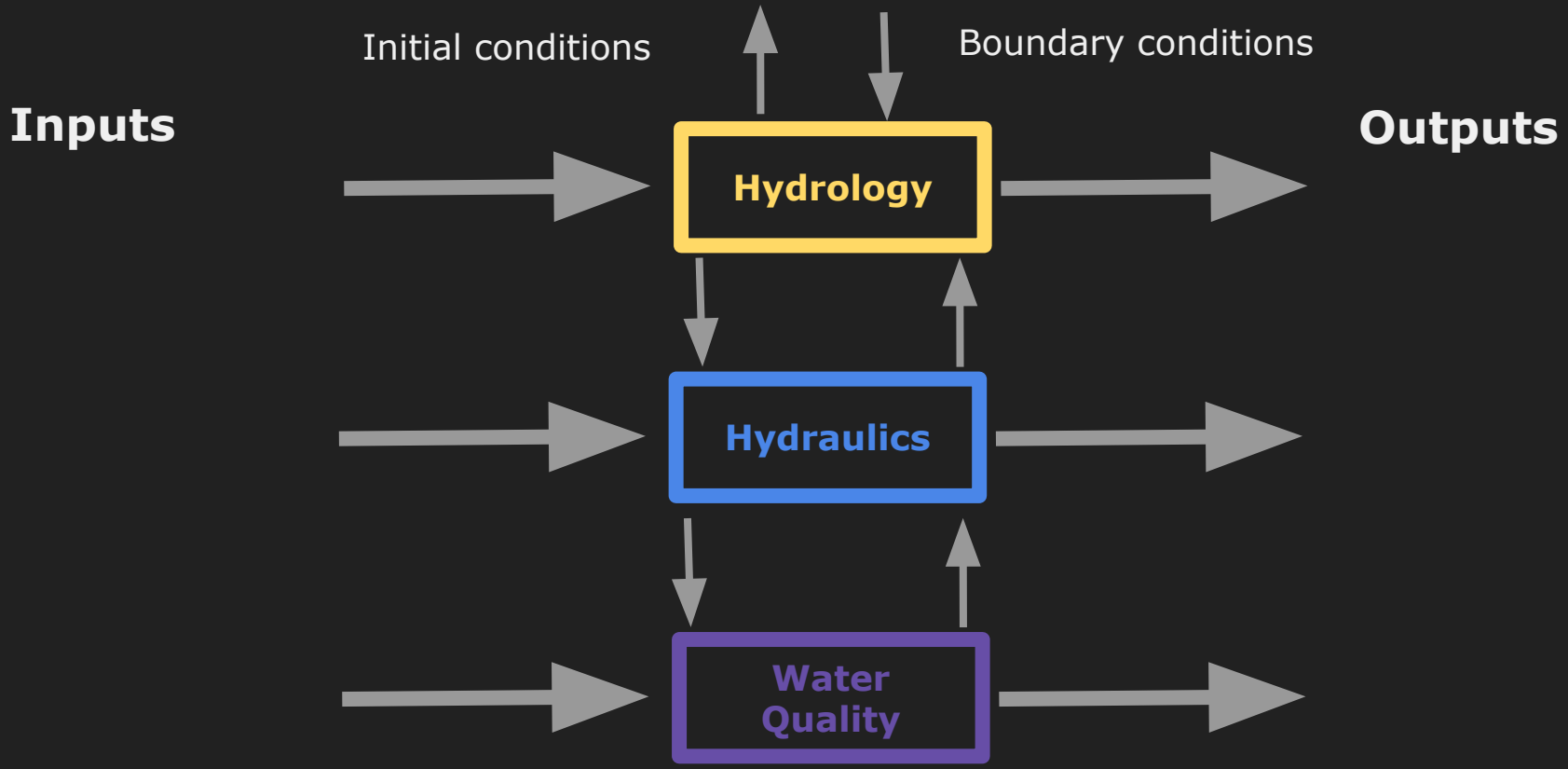
- Evaluate influence of climate change
- Designing and sizing of drainage system components
- Sizing detention facilities
- Mapping flood plains of natural channel systems
- Designing control strategies for minimizing combined sewer overflows.
- Evaluating the impact of inflow and infiltration on sanitary sewer overflows.

Source: https://www.epa.gov/water-research/storm-water-management-model-swmm
https://news.nationalgeographic.com/content/dam/news/2017/02/12/conservative-climate-change/

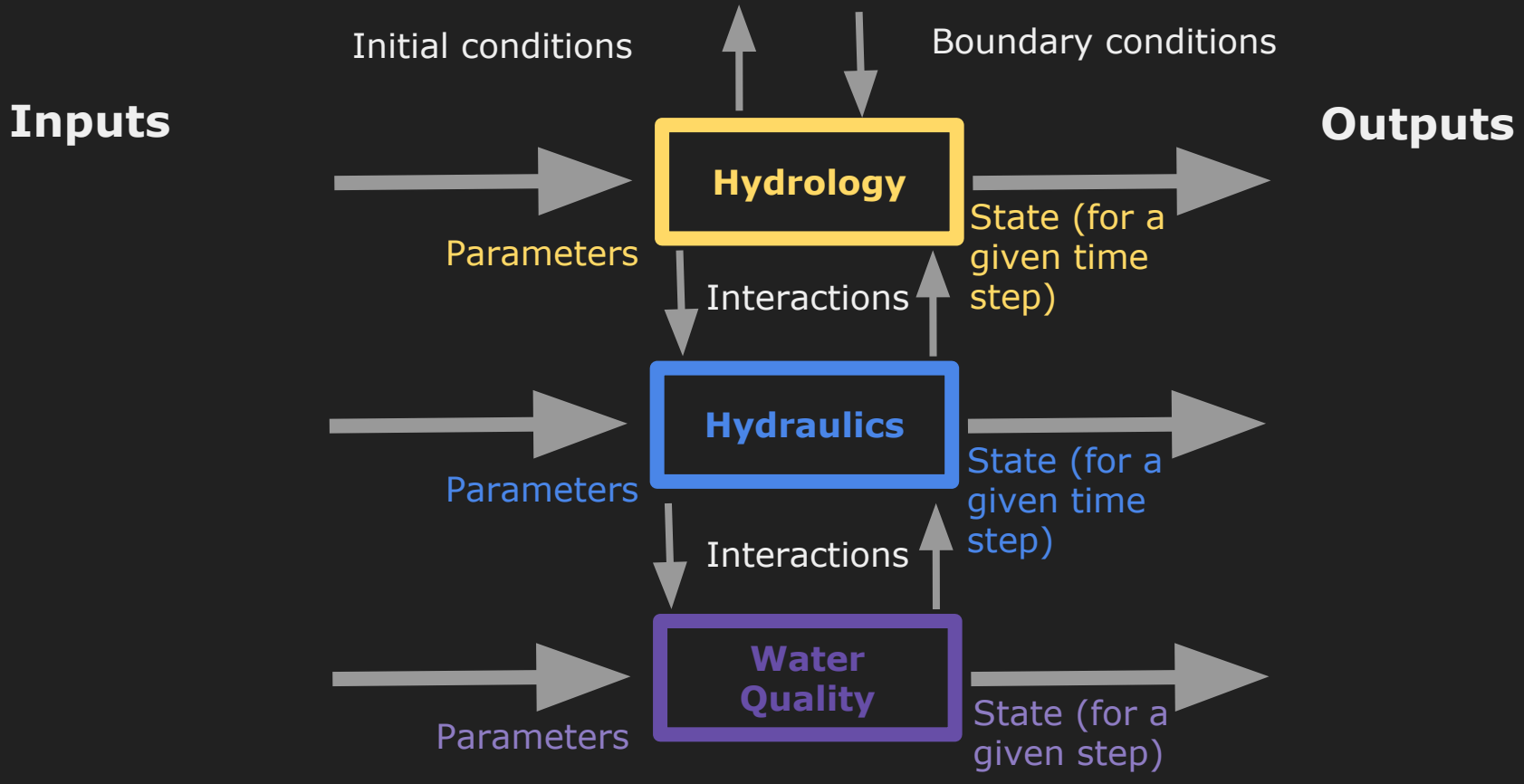
How do we model this system?



How do we model this system?



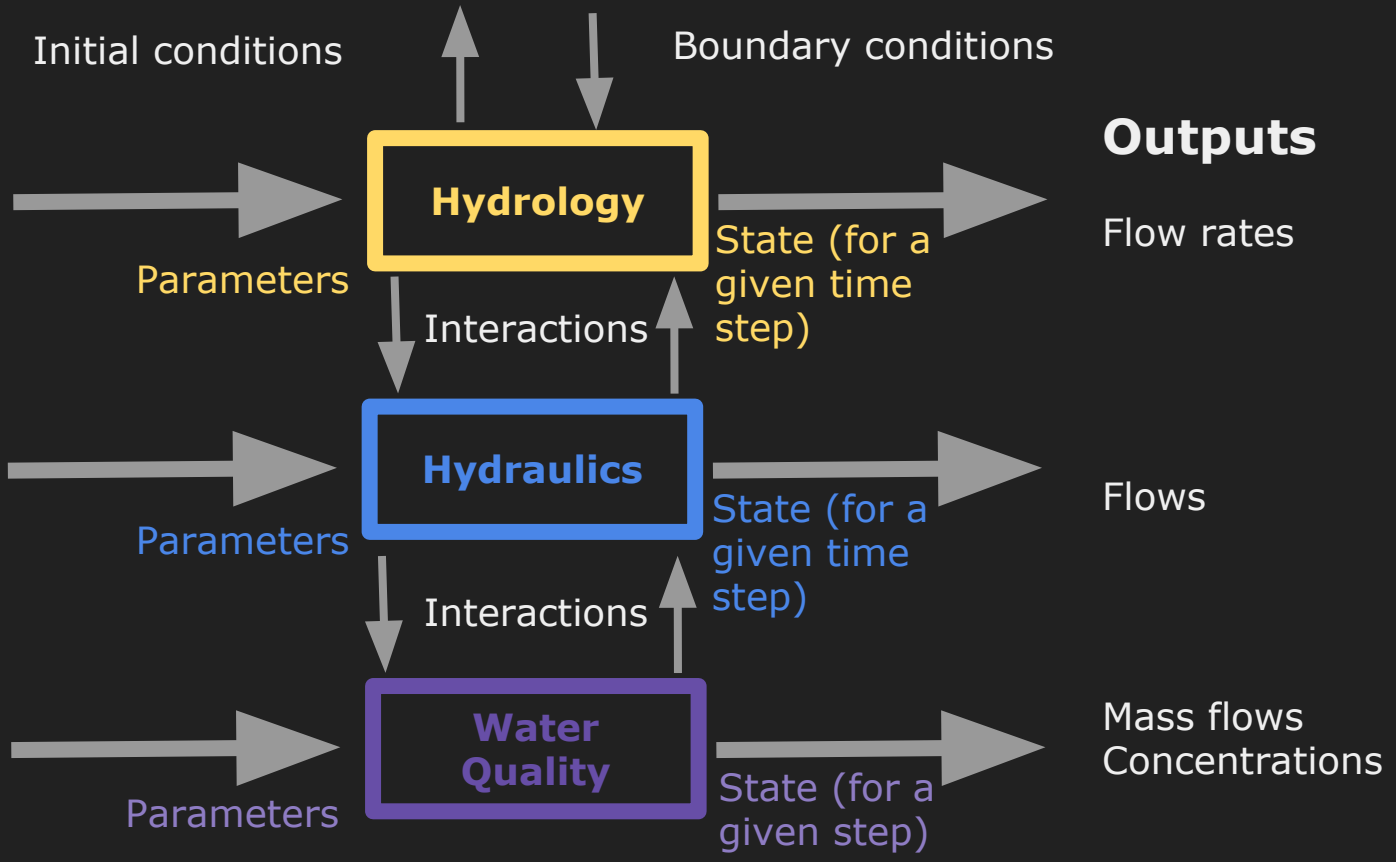
How do we model this system?



How do we model this system?

Inputs

T_0



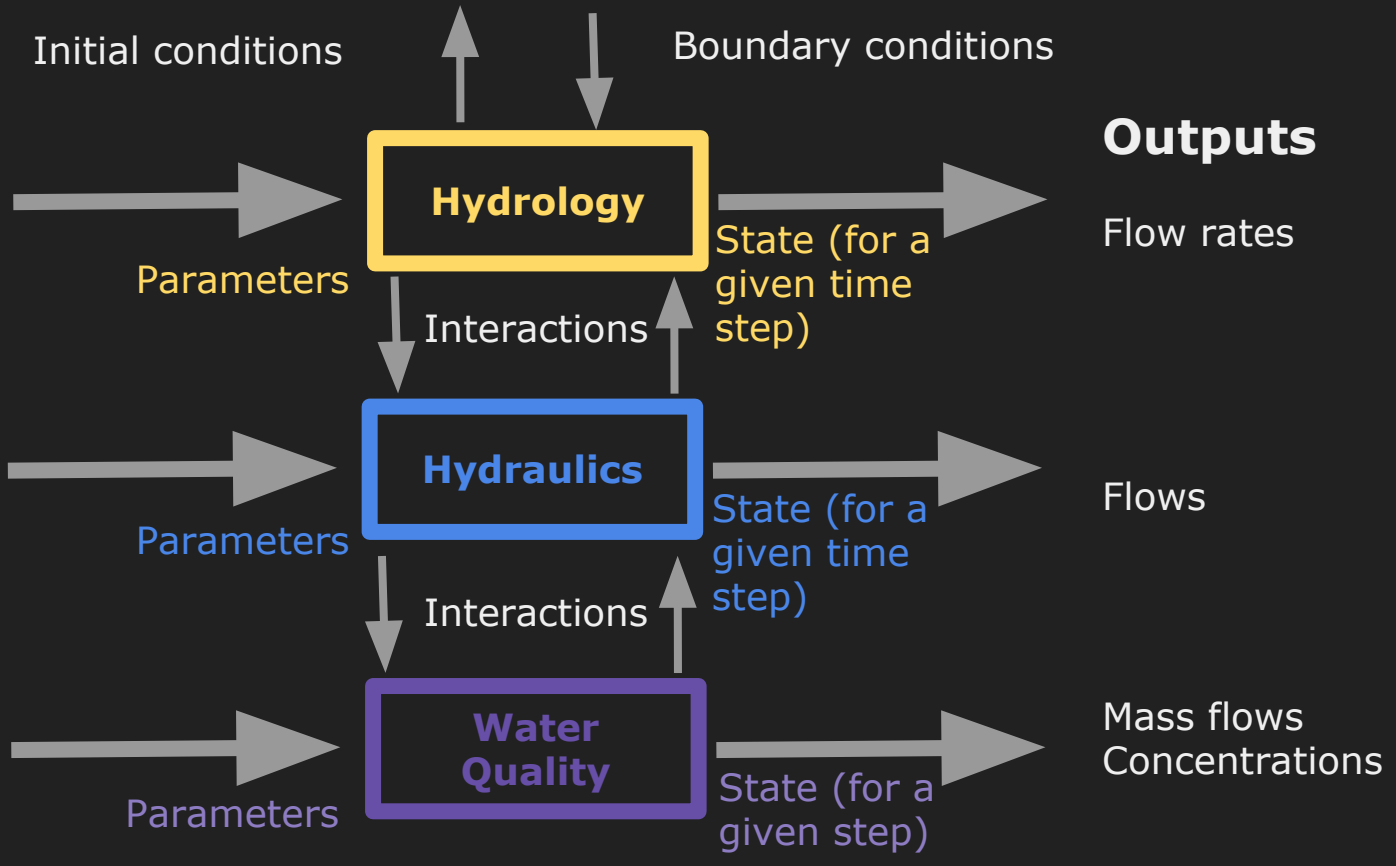
Outputs

Flow rates

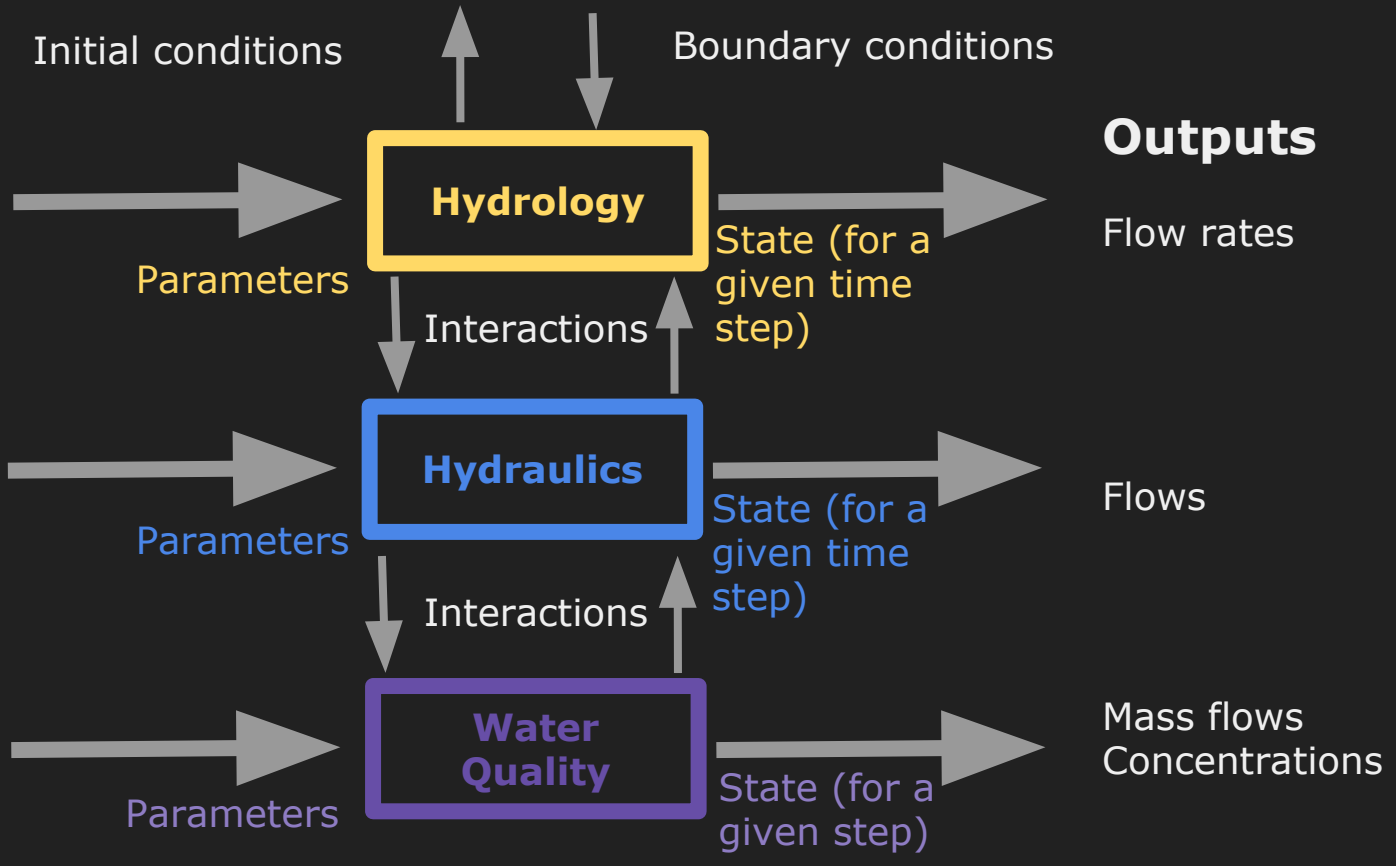
Flows

Mass flows
Concentrations

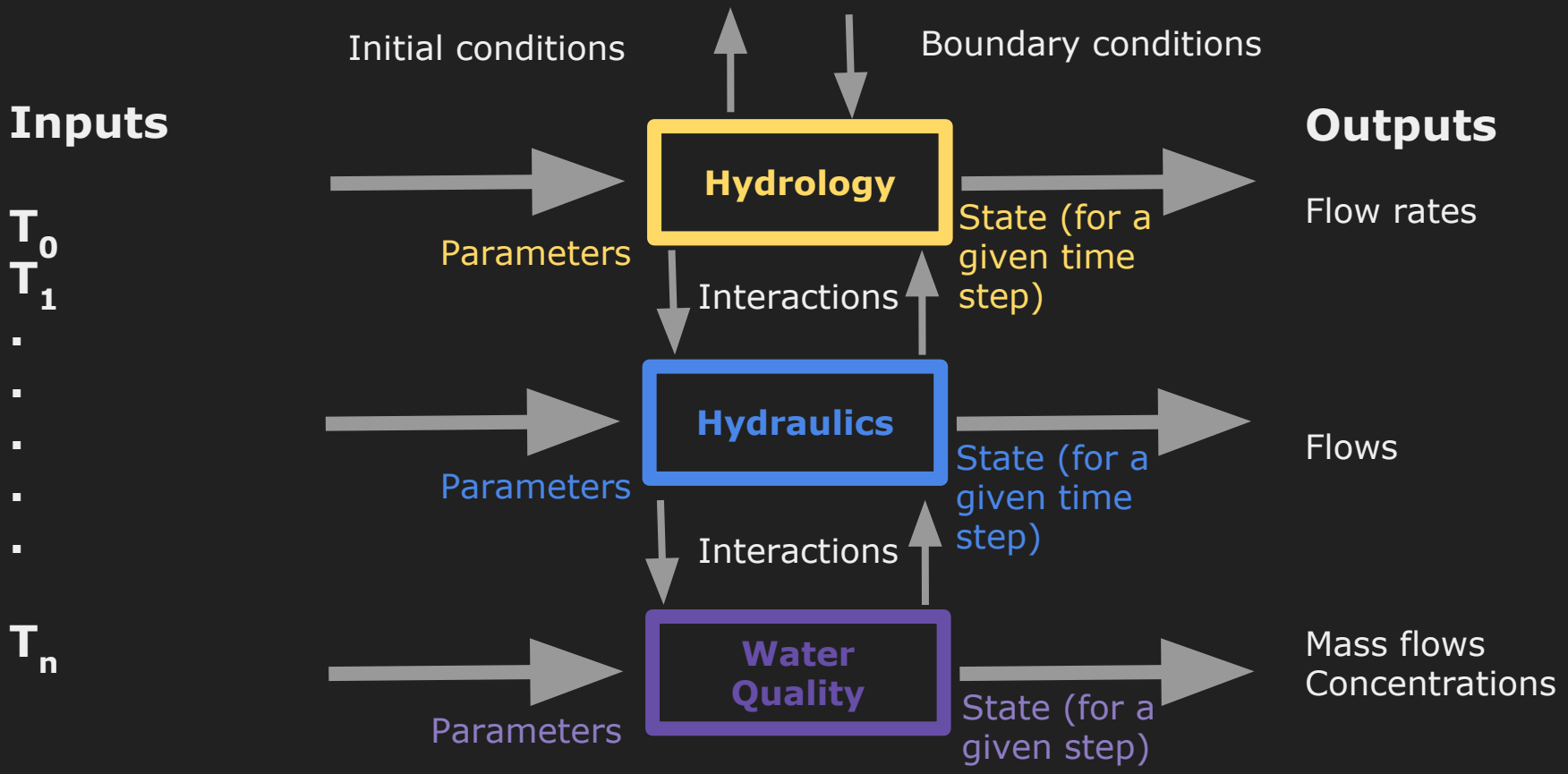
How do we model this system?



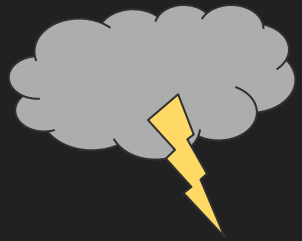
How do we model this system?



How do we model this system?



Hydrological modelling

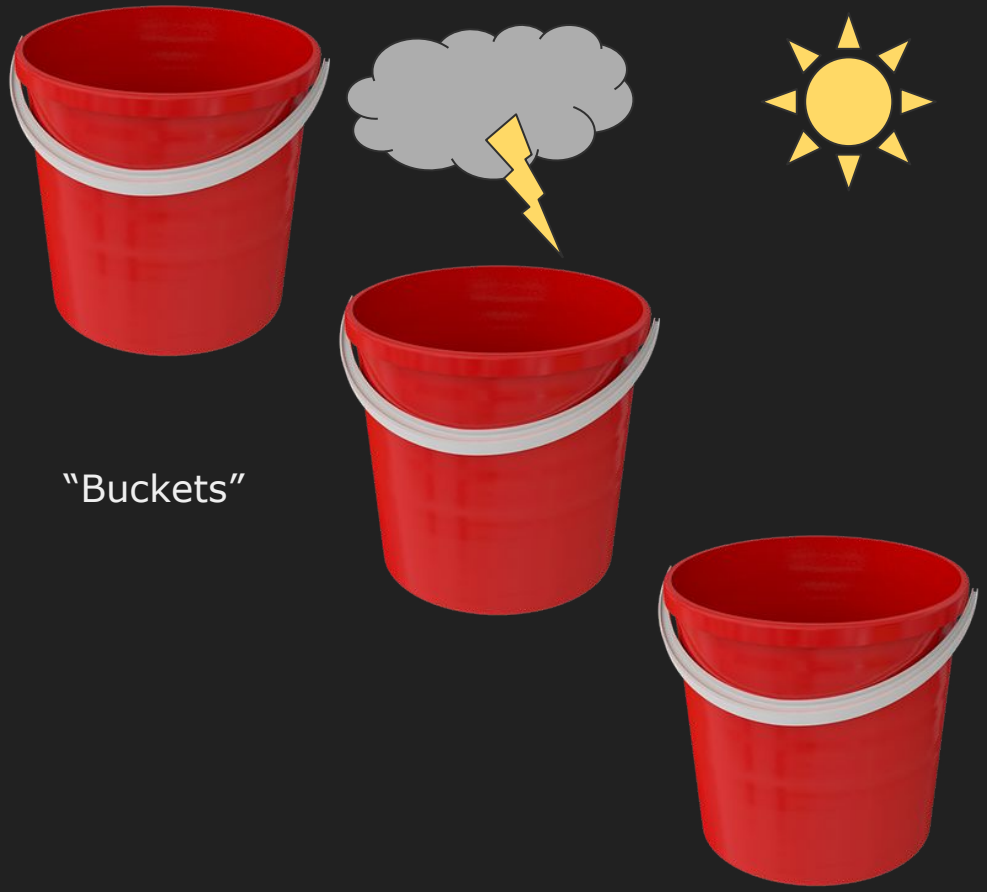


- **Rainfall**
- **Evaporation**
- **Snow** accumulation
- **Storage**
- **Infiltration**
- **Percolation**
- **Interflow**
- **Overland flow**

Source: https://cdn.pixabay.com/photo/2016/09/04/07/02/bucket-1643406_960_720.png
https://www.epa.gov/water-research/storm-water-management-model-swmm

Hydrological modelling

- **Rainfall**
- **Evaporation**
- **Snow** accumulation
- **Storage**
- **Infiltration**
- **Percolation**
- **Interflow**
- **Overland flow**

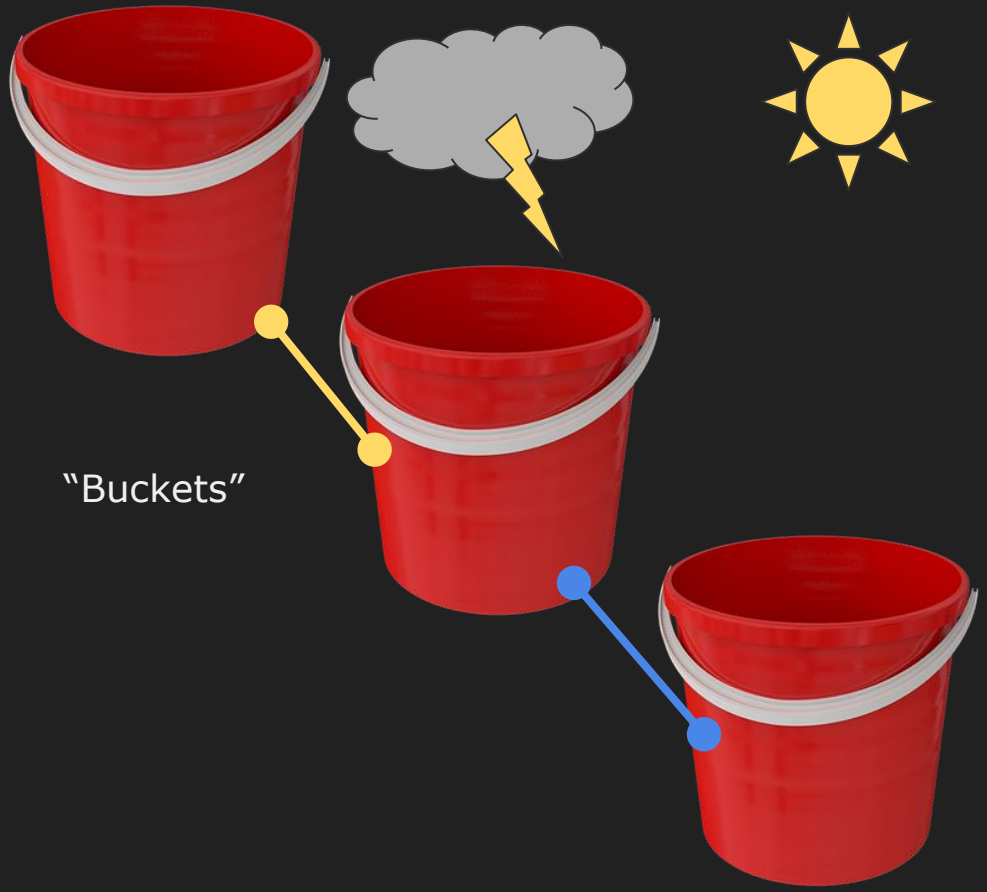


"Buckets"

Source: https://cdn.pixabay.com/photo/2016/09/04/07/02/bucket-1643406_960_720.png
https://www.epa.gov/water-research/storm-water-management-model-swmm

Hydrological modelling

- **Rainfall**
- **Evaporation**
- **Snow** accumulation
- **Storage**
- **Infiltration**
- **Percolation**
- **Interflow**
- **Overland flow**



"Buckets"

Source: https://cdn.pixabay.com/photo/2016/09/04/07/02/bucket-1643406_960_720.png
https://www.epa.gov/water-research/storm-water-management-model-swmm

Hydraulic modelling

- **Drainage** networks
- Natural channel **flows**
- Model special elements, such as:
  - **Storage**
  - **Flow dividers**
  - **Pumps**, **weirs**, and **orifices**.

$$\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} = 0$$

$$\frac{\partial Q}{\partial t} + \frac{\partial (Q^2/A)}{\partial x} + gA\frac{\partial H}{\partial x} + gAS_f = 0$$

The 1D Saint-Venant Equations
(Derived from the Navier Stokes Equations)

Source: https://www.epa.gov/water-research/storm-water-management-model-swmm

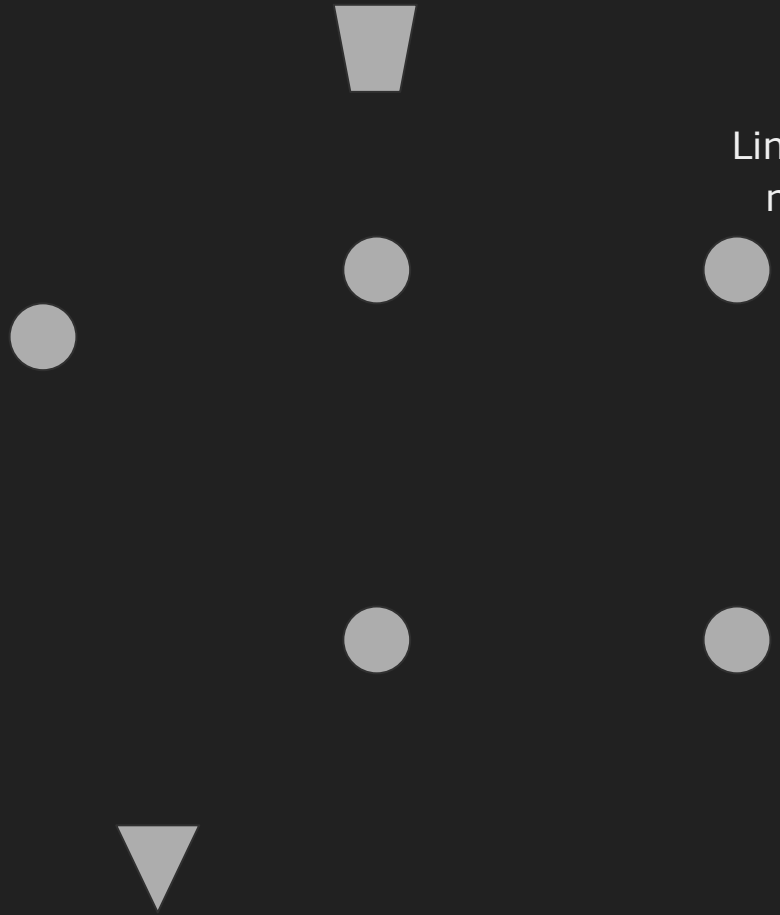
Hydraulic modelling

- **Drainage** networks
- Natural channel **flows**
- Model special elements, such as:
  - **Storage**
  - **Flow dividers**
  - **Pumps**, **weirs**, and **orifices**.

$$\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} = 0$$

$$\frac{\partial Q}{\partial t} + \frac{\partial (Q^2/A)}{\partial x} + gA\frac{\partial H}{\partial x} + gAS_f = 0$$

The 1D Saint-Venant Equations
(Derived from the Navier Stokes Equations)



Source: https://www.epa.gov/water-research/storm-water-management-model-swmm

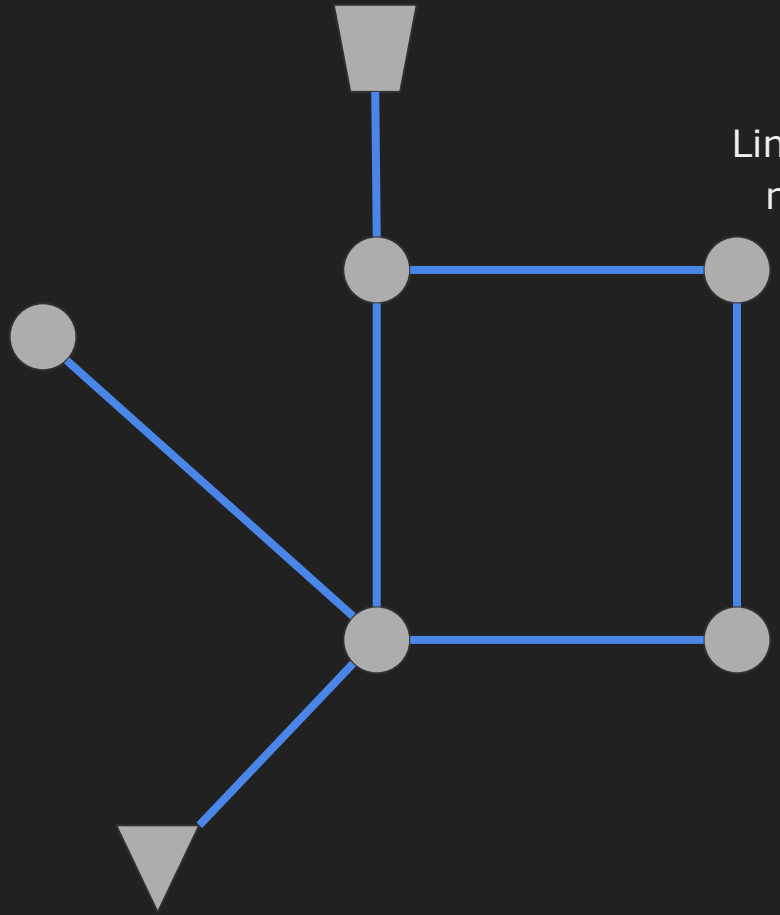
Hydraulic modelling

- **Drainage** networks
- Natural channel **flows**
- Model special elements, such as:
  - **Storage**
  - **Flow dividers**
  - **Pumps**, **weirs**, and **orifices**.

$$\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} = 0$$

$$\frac{\partial Q}{\partial t} + \frac{\partial (Q^2/A)}{\partial x} + gA\frac{\partial H}{\partial x} + gAS_f = 0$$

The 1D Saint-Venant Equations
(Derived from the Navier Stokes Equations)



Source: https://www.epa.gov/water-research/storm-water-management-model-swmm

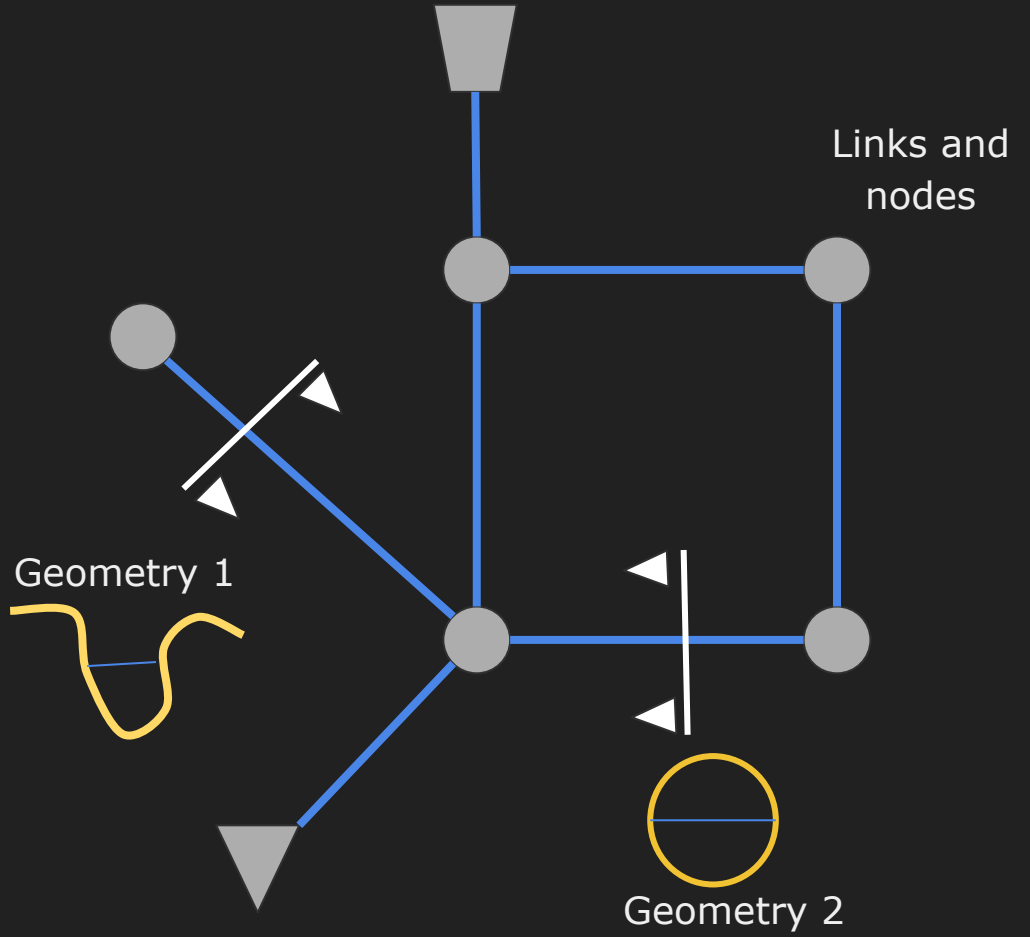
Hydraulic modelling

- **Drainage** networks
- Natural channel **flows**
- Model special elements, such as:
  - **Storage**
  - **Flow dividers**
  - **Pumps**, **weirs**, and **orifices**.

$$\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} = 0$$

$$\frac{\partial Q}{\partial t} + \frac{\partial (Q^2/A)}{\partial x} + gA\frac{\partial H}{\partial x} + gAS_f = 0$$

The 1D Saint-Venant Equations
(Derived from the Navier Stokes Equations)



Source: https://www.epa.gov/water-research/storm-water-management-model-swmm

Water quality modelling

- **Pollutant build up**
- **Pollutant wash-off**
- **Routing** of water quality constituents through the system.
- **Changes in constituent concentration** through treatment in storage units or by natural processes in pipes and channels.

The 1-D Advection Dispersion Equation

$$\frac{\partial c}{\partial t} = -\frac{\partial (uc)}{\partial x} + \frac{\partial}{\partial x}\left(D\frac{\partial c}{dx}\right) + r(c)$$

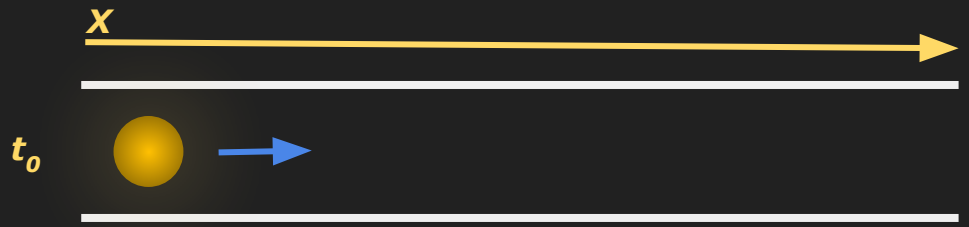
Source: https://www.epa.gov/water-research/storm-water-management-model-swmm

Water quality modelling

- **Pollutant build up**
- **Pollutant wash-off**
- **Routing** of water quality constituents through the system.
- **Changes in constituent concentration** through treatment in storage units or by natural processes in pipes and channels.

The 1-D Advection Dispersion Equation

$$\frac{\partial c}{\partial t} = -\frac{\partial (uc)}{\partial x} + \frac{\partial}{\partial x}\left(D\frac{\partial c}{dx}\right) + r(c)$$



Source: https://www.epa.gov/water-research/storm-water-management-model-swmm

Water quality modelling

- **Pollutant build up**
- **Pollutant wash-off**
- **Routing** of water quality constituents through the system.
- **Changes in constituent concentration** through treatment in storage units or by natural processes in pipes and channels.

The 1-D Advection Dispersion Equation

$$\frac{\partial c}{\partial t} = -\frac{\partial (uc)}{\partial x} + \frac{\partial}{\partial x}\left(D\frac{\partial c}{dx}\right) + r(c)$$

Source: https://www.epa.gov/water-research/storm-water-management-model-swmm

Water quality modelling

- **Pollutant build up**
- **Pollutant wash-off**
- **Routing** of water quality constituents through the system.
- **Changes in constituent concentration** through treatment in storage units or by natural processes in pipes and channels.

The 1-D Advection Dispersion Equation

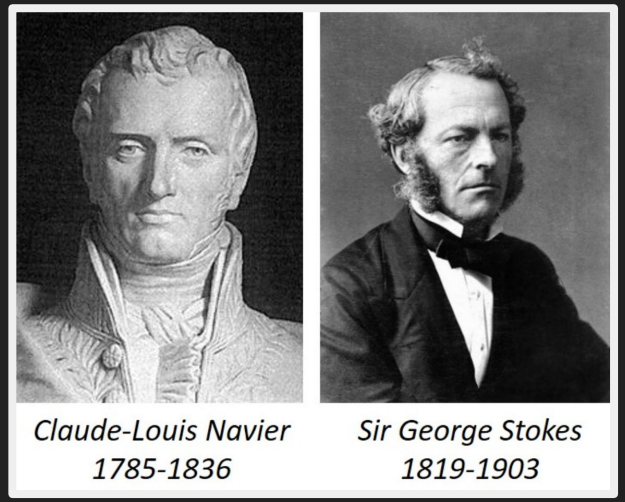
$$\frac{\partial c}{\partial t} = -\frac{\partial (uc)}{\partial x} + \frac{\partial}{\partial x}\left(D\frac{\partial c}{dx}\right) + r(c)$$

Source: https://www.epa.gov/water-research/storm-water-management-model-swmm

<intermission name=”navier-stokes”>

Hydraulic modelling

In Computational Fluid Dynamics (CFD) we have a deity:

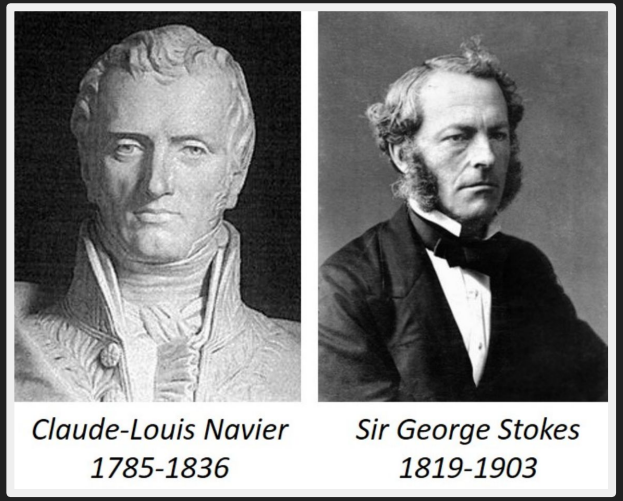


The Navier-Stokes Equations

Source: https://www.simscale.com/docs/_images/2-Newton-Navier-Stokes.jpg
https://www.grc.nasa.gov/www/k-12/airplane/Images/nseqs.gif

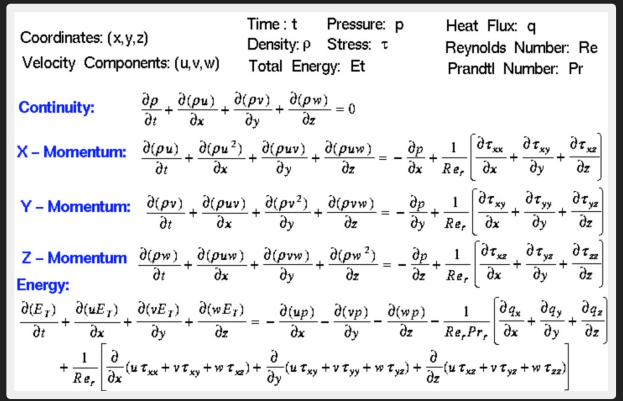
Hydraulic modelling

In Computational Fluid Dynamics (CFD) we have a deity:



The Navier-Stokes Equations

Describe viscous flows →



Source: https://www.simscale.com/docs/_images/2-Newton-Navier-Stokes.jpg
https://www.grc.nasa.gov/www/k-12/airplane/Images/nseqs.gif

Hydraulic modelling

In Computational Fluid Dynamics (CFD) we have a deity:

The Navier-Stokes Equations

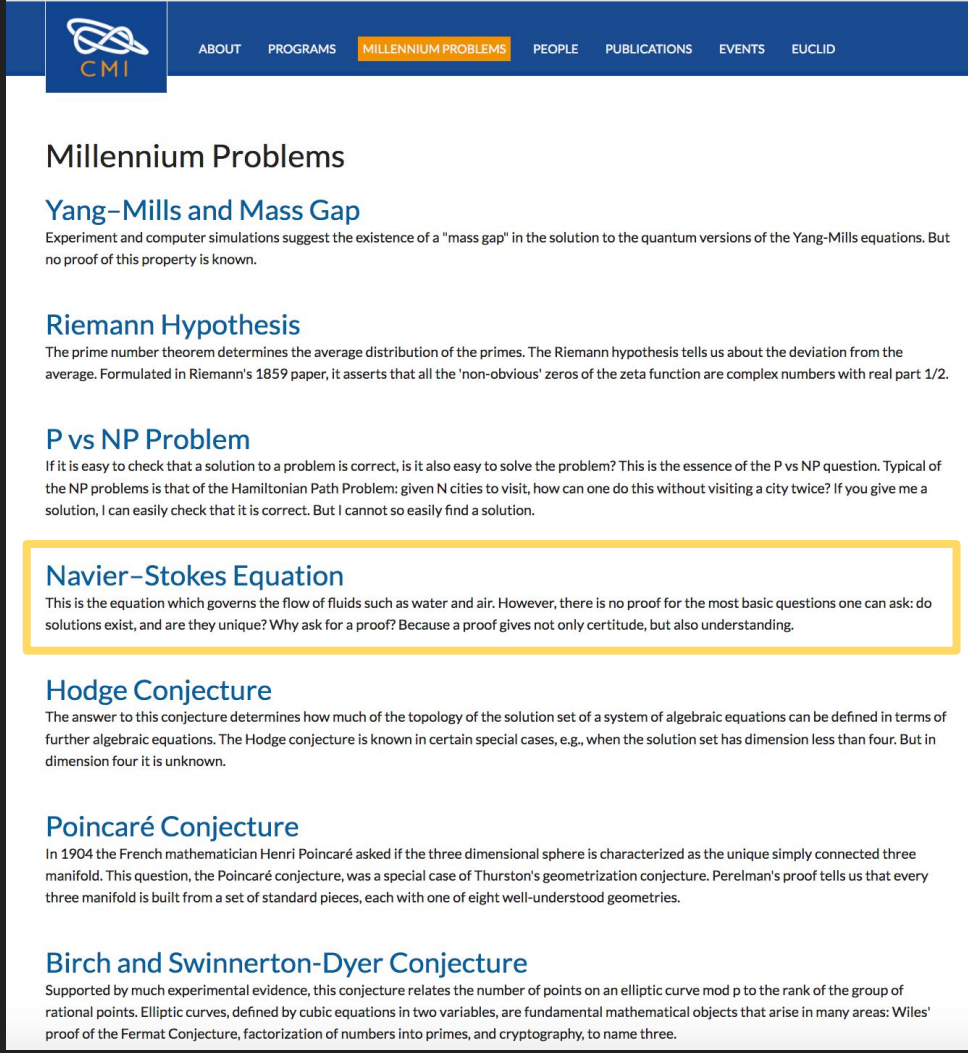
The **Kelvin–Helmholtz instability**

Source: https://www.quantamagazine.org/mathematicians-find-wrinkle-in-famed-fluid-equations-20171221

Hydraulic modelling

In Computational Fluid Dynamics (CFD) we have a deity:

The Navier-Stokes Equations

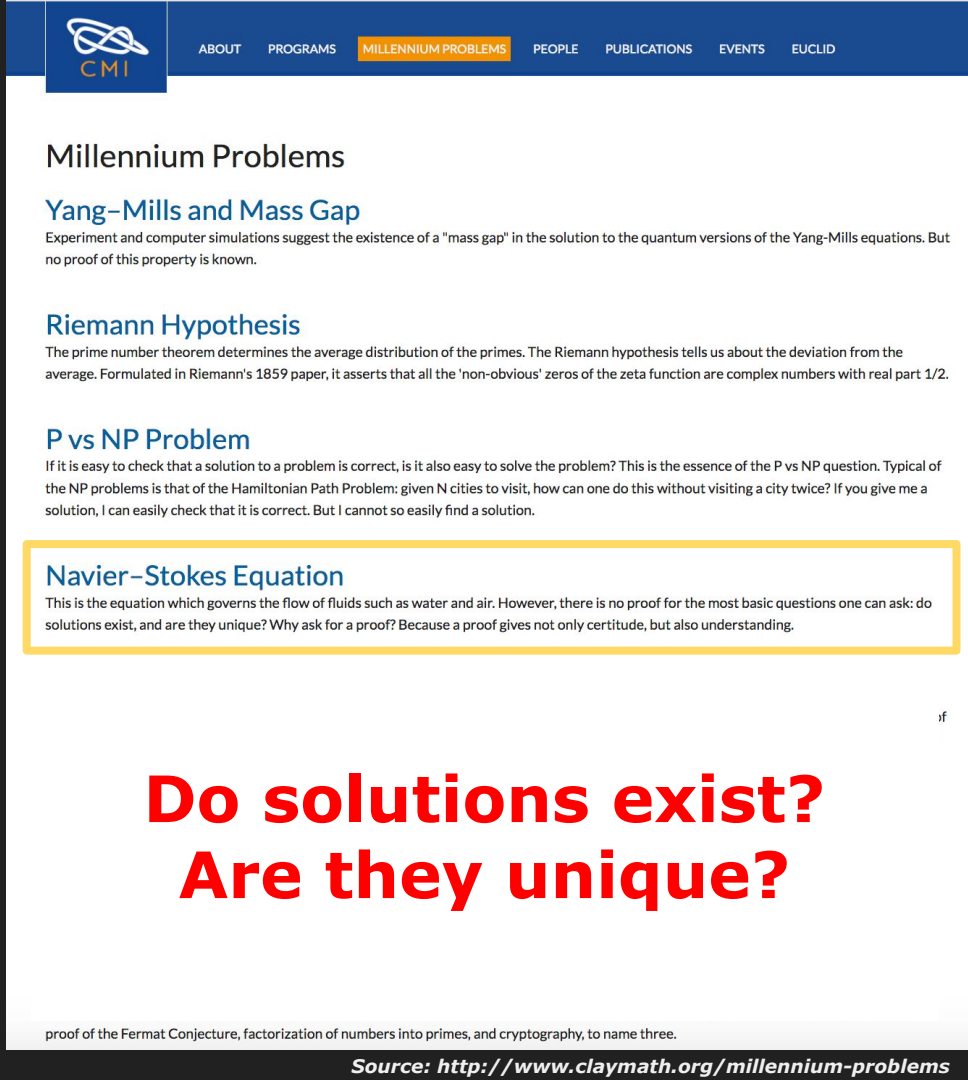


Source: http://www.claymath.org/millennium-problems

Hydraulic modelling

In Computational Fluid Dynamics (CFD) we have a deity:

The Navier-Stokes Equations



Hydraulic modelling

But last September / October

Mathematics > Analysis of PDEs

Nonuniqueness of weak solutions to the Navier-Stokes equation

Tristan Buckmaster, Vlad Vicol

(Submitted on 28 Sep 2017 (v1), last revised 5 Oct 2017 (this version, v2))

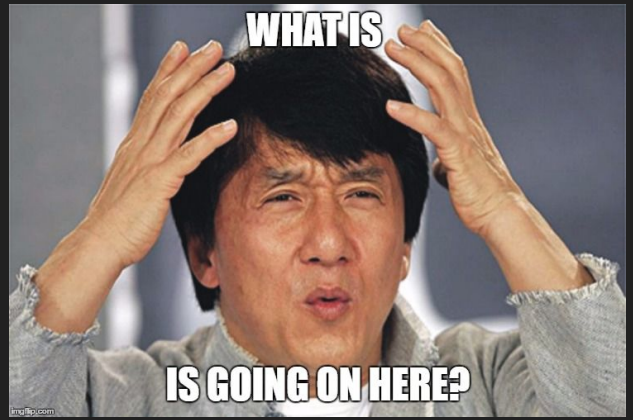
For initial datum of finite kinetic energy, Leray has proven in 1934 that there exists at least one global in time finite energy weak solution of the 3D Navier-Stokes equations. In this paper we prove that weak solutions of the 3D Navier-Stokes equations are not unique in the class of weak solutions with finite kinetic energy. Moreover, we prove that Holder continuous dissipative weak solutions of the 3D Euler equations may be obtained as a strong vanishing viscosity limit of a sequence of finite energy weak solutions of the 3D Navier-Stokes equations.

Comments: 34 pages, added comments regarding Oseen solutions
Subjects: **Analysis of PDEs (math.AP)**; Mathematical Physics (math-ph)
Cite as: arXiv:1709.10033 **[math.AP]**
(or arXiv:1709.10033v2 **[math.AP]** for this version)

Source: https://arxiv.org/abs/1709.10033

Hydraulic modelling

But last September / October



Mathematics > Analysis of PDEs

Nonuniqueness of weak solutions to the Navier-Stokes equation

Tristan Buckmaster, Vlad Vicol

(Submitted on 28 Sep 2017 (v1), last revised 5 Oct 2017 (this version, v2))

For initial datum of finite kinetic energy, Leray has proven in 1934 that there exists at least one global in time finite energy weak solution of the 3D Navier-Stokes equations. In this paper we prove that weak solutions of the 3D Navier-Stokes equations are not unique in the class of weak solutions with finite kinetic energy. Moreover, we prove that Holder continuous dissipative weak solutions of the 3D Euler equations may be obtained as a strong vanishing viscosity limit of a sequence of finite energy weak solutions of the 3D Navier-Stokes equations.

Comments: 34 pages, added comments regarding Oseen solutions
Subjects: **Analysis of PDEs (math.AP)**; Mathematical Physics (math-ph)
Cite as: arXiv:1709.10033 **[math.AP]**
(or arXiv:1709.10033v2 **[math.AP]** for this version)

Source: https://arxiv.org/abs/1709.10033

Hydraulic modelling

Mathematics > Analysis of PDEs

Nonuniqueness of weak solutions to the Navier–Stokes equation

Tristan Buckmaster, Vlad Vicol

(Submitted on 28 Sep 2017 (v1), last revised 5 Oct 2017 (this version, v2))

For initial datum of finite kinetic energy, Leray has proven in 1934 that there exists at least one global in time finite energy weak solution of the 3D Navier–Stokes equations. In this paper we prove that weak solutions of the 3D Navier–Stokes equations are not unique in the class of weak solutions with finite kinetic energy. Moreover, we prove that Holder continuous dissipative weak solutions of the 3D Euler equations may be obtained as a strong vanishing viscosity limit of a sequence of finite energy weak solutions of the 3D Navier–Stokes equations.

Comments: 34 pages, added comments regarding Oseen solutions
Subjects: **Analysis of PDEs (math.AP)**; Mathematical Physics (math-ph)
Cite as: arXiv:1709.10033 **[math.AP]**
(or arXiv:1709.10033v2 **[math.AP]** for this version)

Source: https://arxiv.org/abs/1709.10033

Deobfuscating the title:

FLUID DYNAMICS

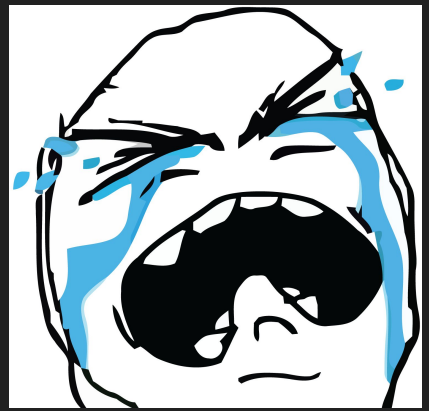
Mathematicians Find Wrinkle in Famed Fluid Equations

34

Two mathematicians prove that under certain extreme conditions, the Navier–Stokes equations output nonsense.

Source: https://www.quantamagazine.org/mathematicians-find-wrinkle-in-famed-fluid-equations-20171221

Hydraulic modelling



Mathematics > Analysis of PDEs

Nonuniqueness of weak solutions to the Navier-Stokes equation

Tristan Buckmaster, Vlad Vicol

(Submitted on 28 Sep 2017 (v1), last revised 5 Oct 2017 (this version, v2))

For initial datum of finite kinetic energy, Leray has proven in 1934 that there exists at least one global in time finite energy weak solution of the 3D Navier-Stokes equations. In this paper we prove that weak solutions of the 3D Navier-Stokes equations are not unique in the class of weak solutions with finite kinetic energy. Moreover, we prove that Holder continuous dissipative weak solutions of the 3D Euler equations may be obtained as a strong vanishing viscosity limit of a sequence of finite energy weak solutions of the 3D Navier-Stokes equations.

Comments: 34 pages, added comments regarding Oseen solutions
Subjects: **Analysis of PDEs (math.AP)**; Mathematical Physics (math-ph)
Cite as: arXiv:1709.10033 [math.AP]
(or arXiv:1709.10033v2 [math.AP] for this version)

Source: https://arxiv.org/abs/1709.10033

Deobfuscating the title:

FLUID DYNAMICS

Mathematicians Find Wrinkle in Famed Fluid Equations

34

Two mathematicians prove that under certain extreme conditions, the Navier-Stokes equations output nonsense.

Output nonsense

Source: https://www.quantamagazine.org/mathematicians-find-wrinkle-in-famed-fluid-equations-20171221

</intermission>

US EPA: United States Environmental Protection Agency

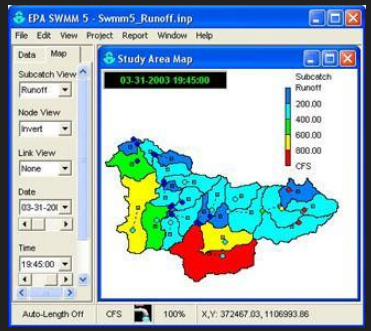
Source: https://www.epa.gov/water-research/storm-water-management-model-swmm
https://www.epa.gov/water-research/epanet

US EPA: United States Environmental Protection Agency

SWMM

(Sewers)

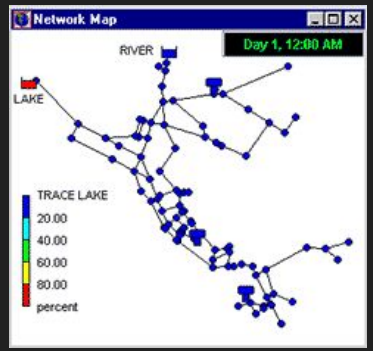
First developed between 1969–1971



EPANET

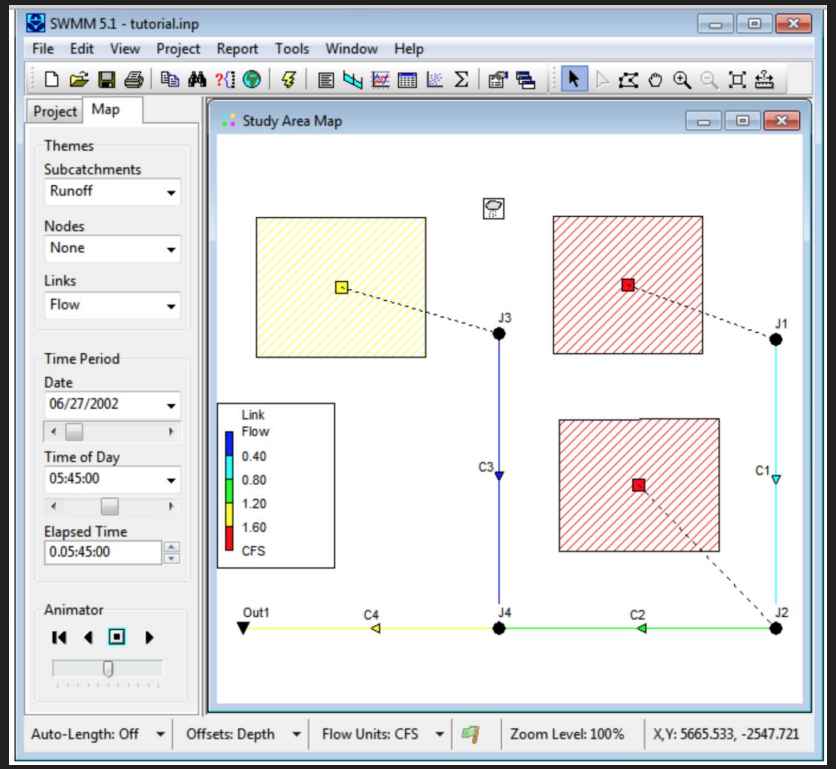
(Water distribution)

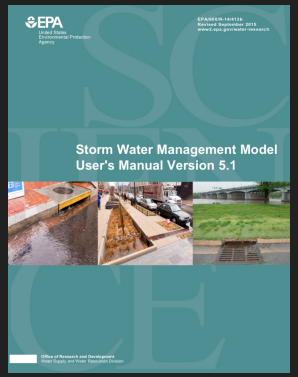
EPANET first appeared in 1993

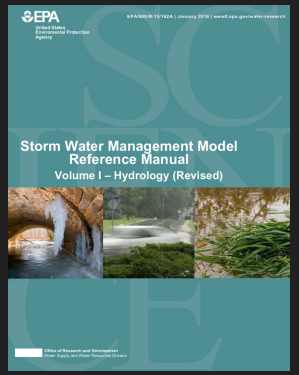


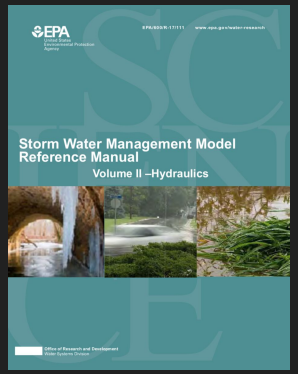
Source: https://www.epa.gov/water-research/storm-water-management-model-swmm
https://www.epa.gov/water-research/epanet

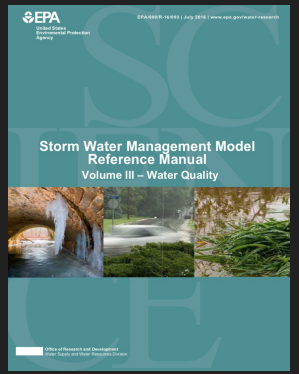
SWMM: StormWater Management Model







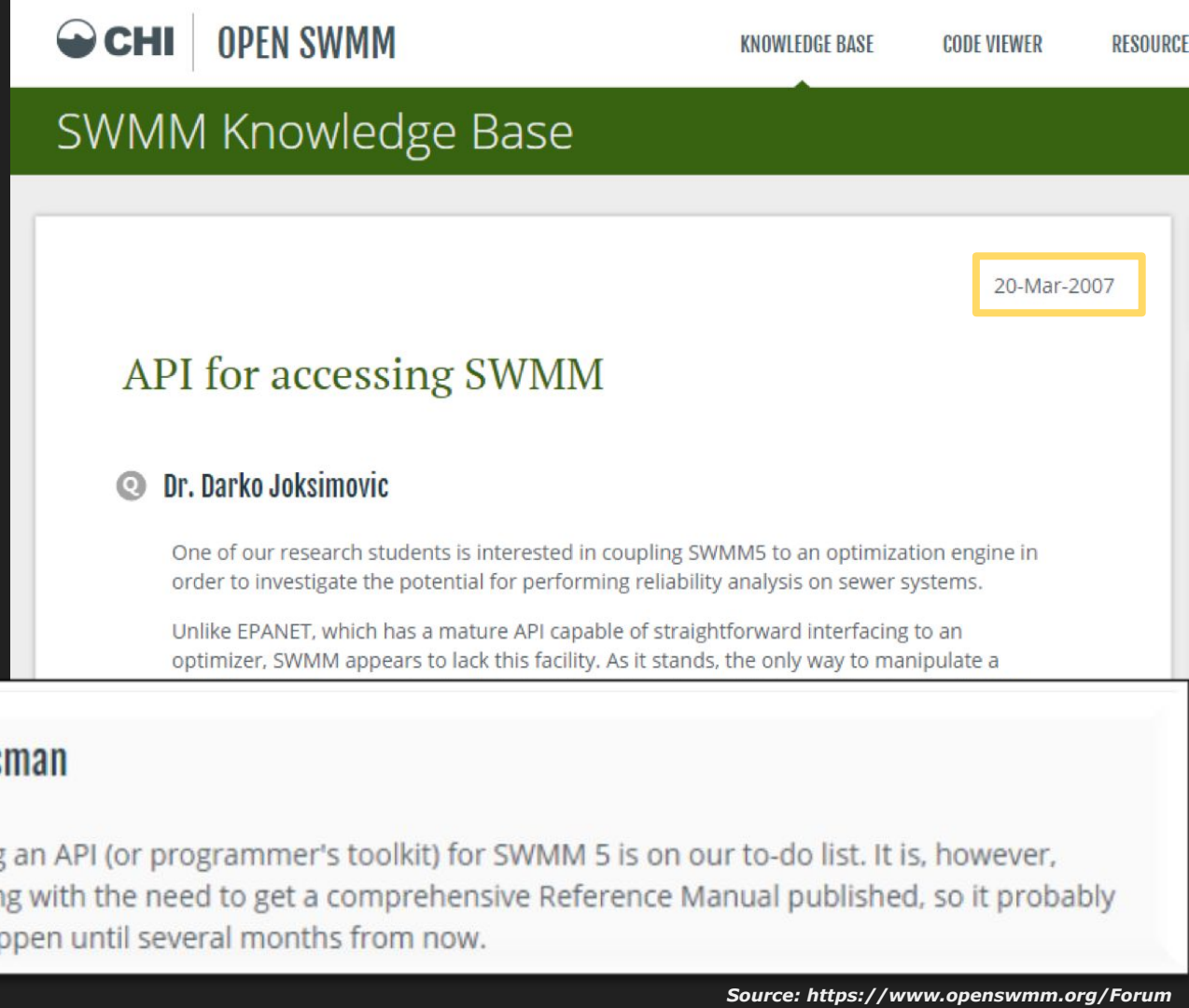




Source: https://www.epa.gov/water-research/storm-water-management-model-swmm

SWMM: So where is the API?

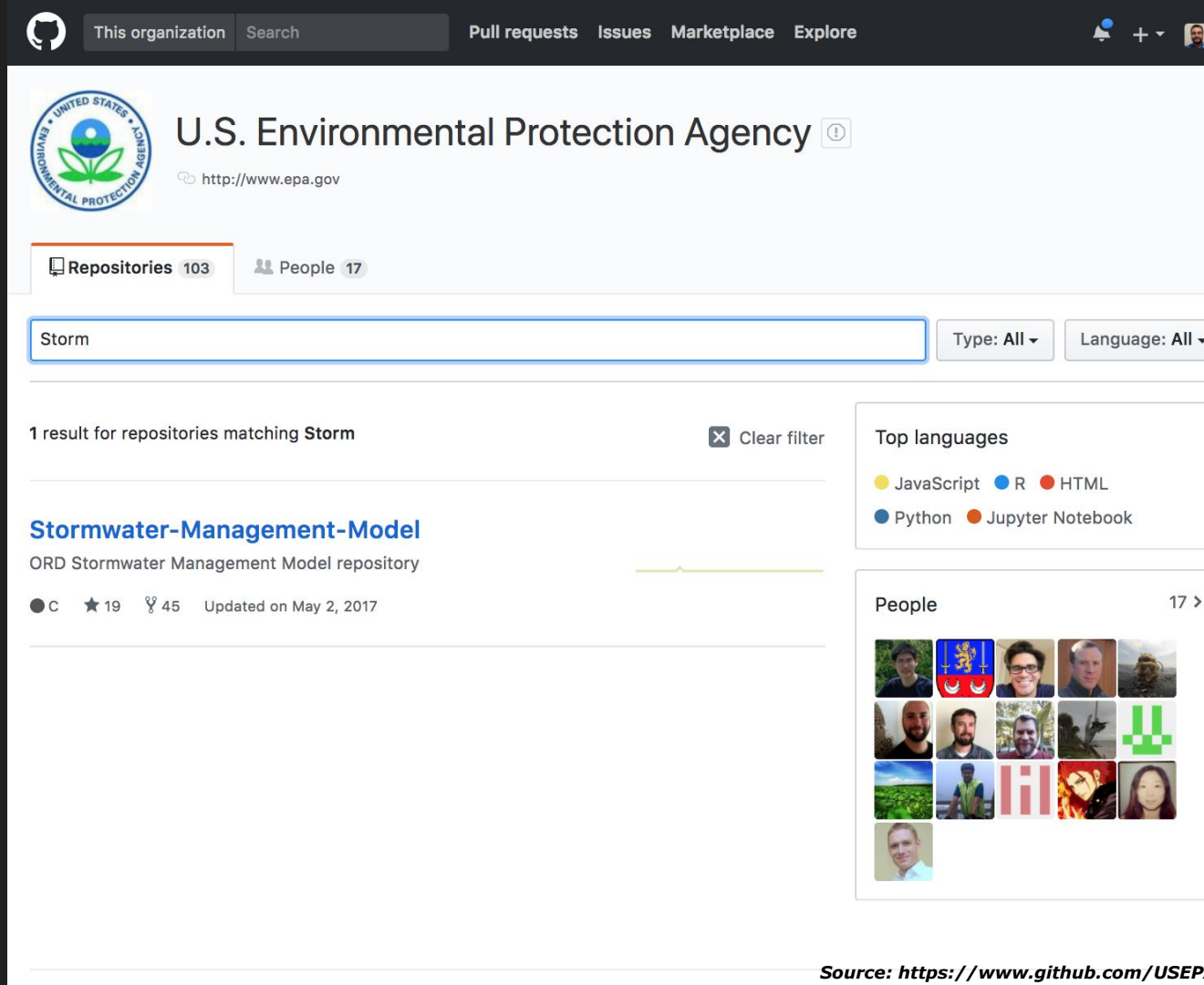
How to automate modeling?



Source: https://www.openswmm.org/Forum

(EPA) SWMM: Is on GitHub

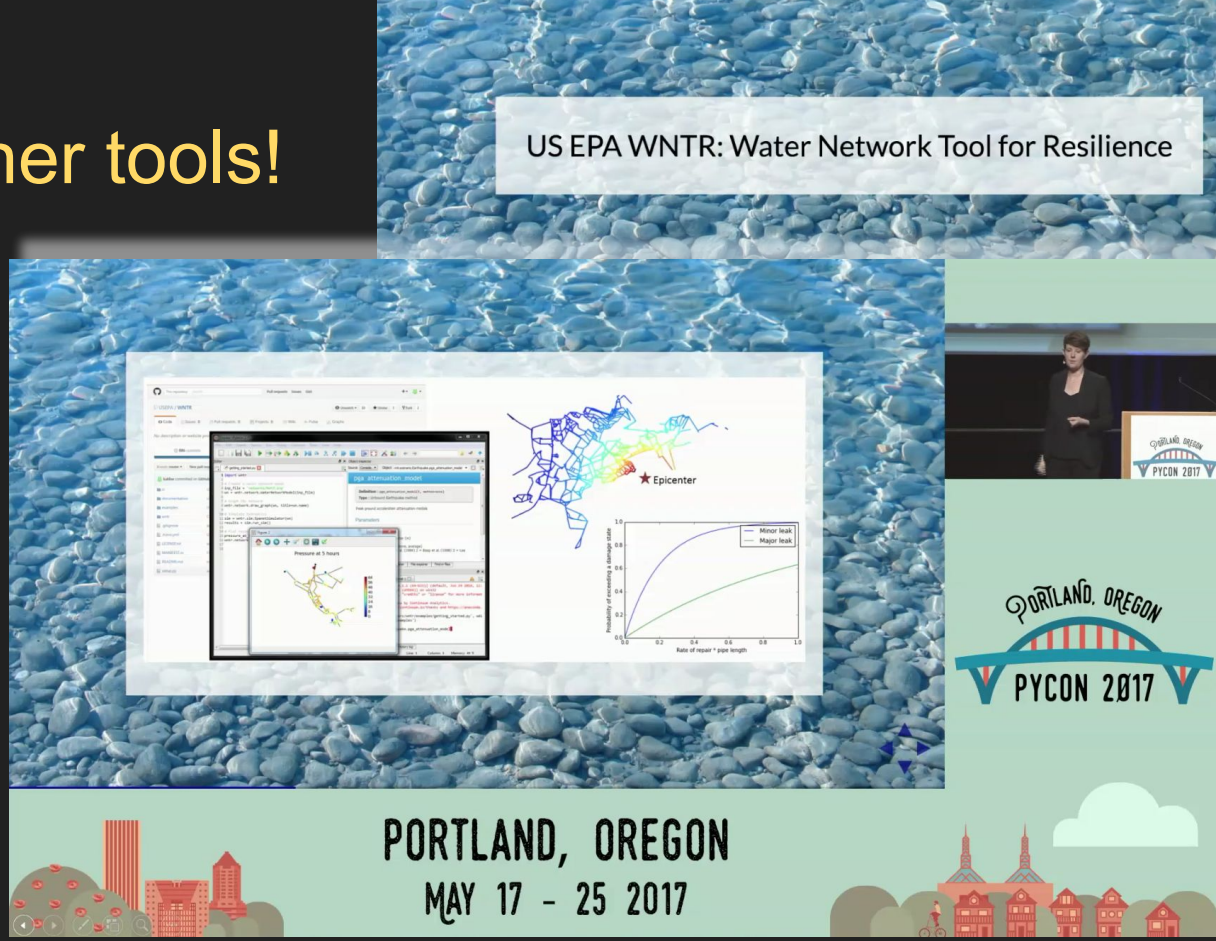
For some years already...



Source: https://www.github.com/USEP

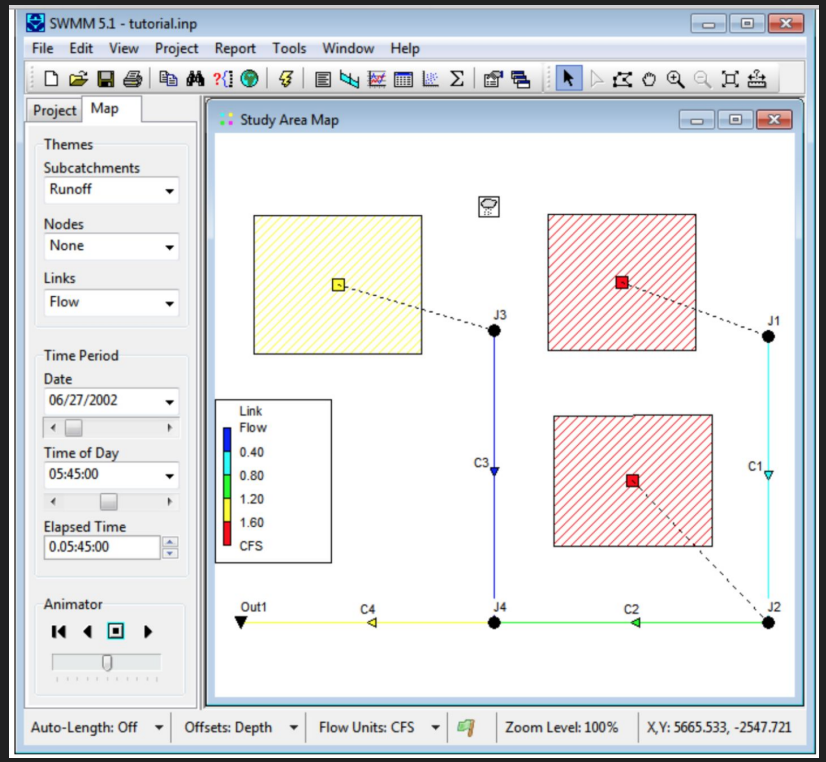
Plus a bunch of other tools!

Go check them out!
https://github.com/usepa



Katy Huff Keynote PyCon US 2017
https://www.youtube.com/watch?v=kaGS4YXwciQ

SWMM: StormWater Management Model



Limitations:

- Originally Windows only
- (until 2015) no API to interact programmatically with models
- No bindings for a higher level language
- This GUI is showing its age…
- Control Language not flexible enough for fast pace iteration

Source: https://www.epa.gov/water-research/storm-water-management-model-swmm

<intermission name=”networking-and-lying”>

Back in 2012

- https://pypi.python.org/pypi/SWMM5 (SWIG)
- https://pypi.python.org/pypi/SWMM5_EA

Assela PATHIRANA
Associate Professor of Integrated Urban Water Cycle Management

Gonzalo
Doing MSc. thesis “Evaluating the impact of climate change on urban scale extreme rainfall events”

Back in 2012

Downscaling

Global climate models

Urban Modelling

Python

- https://pypi.python.org/pypi/SWMM5 (SWIG)
- https://pypi.python.org/pypi/SWMM5_EA

Assela PATHIRANA
Associate Professor of Integrated Urban Water Cycle Management

Gonzalo
Doing MSc. thesis “Evaluating the impact of climate change on urban scale extreme rainfall events”

Back in 2012

Assela PATHIRANA
Associate Professor of Integrated Urban Water Cycle Management

Gonzalo
Doing MSc. thesis “Evaluating the impact of climate change on urban scale extreme rainfall events: Coupling of multiple global circulation models with a stochastic rainfall generator”

Back in 2012

https://pypi.python.org/pypi/PySWMM

Back in 2016: networking...

9/12/2016

Hey Gonzalo, I am looking to push this project to Python Package Index and I ran into a problem:

https://pypi.python.org/pypi/pyswmm/0.1.0

It looks like you have registered the pyswmm project name. Do you have plans to submit a pyswmm to PyPI? If not, would you mind removing it so then this pyswmm project could live there? I really appreciate it!

Thanks

Bryant

10:44

Back in 2016: networking… and lying

9/12/2016

Hey Gonzalo, I am looking to push this project to Python Package Index and I ran into a problem:

https://pypi.python.org/pypi/pyswmm/0.1.0

It looks like you have registered the pyswmm project name. Do you have plans to submit a pyswmm to PyPI? If not, would you mind removing it so then this pyswmm project could live there? I really appreciate it!

Thanks

Bryant

10:44

Hi Bryant, yes I have plans o upload something next year most likely.

Cheers

12:57

It's my preciousssss

Back in 2017: networking… and coding

https://pypi.python.org/pypi/PySWMM

python™

» Package Index > pyswmm > 0.4.7

PACKAGE INDEX
- Browse packages
- List trove classifiers
- RSS (latest 40 updates)
- RSS (newest 40 packages)
- Terms of Service
- PyPI Tutorial
- PyPI Security
- PyPI Support
- PyPI Bug Reports
- PyPI Discussion
- PyPI Developer Info

ABOUT
NEWS
DOCUMENTATION
DOWNLOAD
COMMUNITY
FOUNDATION
CORE DEVELOPMENT

pyswmm 0.4.7

Python Wrapper for SWMM5 API

Python Wrapper for Stormwater Management Model (SWMM5)

Documentation
http://pyswmm.readthedocs.io/en/latest/
Development
https://github.com/OpenWaterAnalytics/pyswmm/
PySWMM Wiki
https://github.com/OpenWaterAnalytics/pyswmm/wiki/

Build status

build passing | circleci passing | build error | Scrutinizer 5.74

Project information

docs passing

YouTube Examples

Stream Results and Adjust Weir Setting

</intermission>

Open Water Analytics (2015)

Bryant McDonnell

Sam Hatchet

emnet

This organization Search Pull requests Issues Marketplace Explore

Open Water Analytics

http://wateranalytics.org

Repositories 19 People 7

Pinned repositories

EPANET
Forked from USEPA/Water-Distribution-Network-Model

The Water Distribution System Hydraulic and Water Quality Analysis Toolkit

C ★ 37 51

Stormwater-Management-Model
Forked from USEPA/Stormwater-Management-Model

Open Water Analytics Stormwater Management Model repository

C ★ 16 25

pyswmm
Python Wrappers for SWMM

Python ★ 34 28

WNTR
Forked from USEPA/WNTR

An EPANET compatible python package to simulate and analyze water distribution networks under disaster scenarios.

Python ★ 6 4

epanet-dev
Development Repository for the Next Version of EPANET

C++ ★ 28 22

PySWMM Type: All Language: All

1 result for repositories matching **PySWMM** Clear filter

pyswmm
Python Wrappers for SWMM

python stormwater hydrology-stormwater-analysis

Python ★ 34 28 Updated 10 days ago

Top languages
Python C Matlab C++ CSS

Most used topics
hydraulics stormwater swmm5

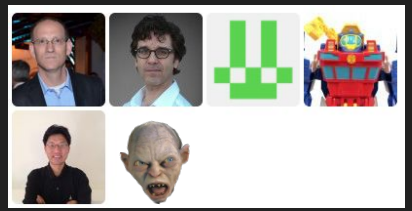
Source: https://github.com/OpenWaterAnalytics/

Open Water Analytics (2015)

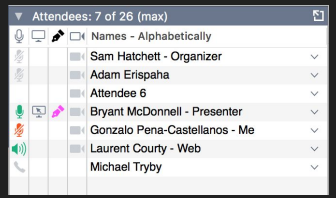
Bryant McDonnell

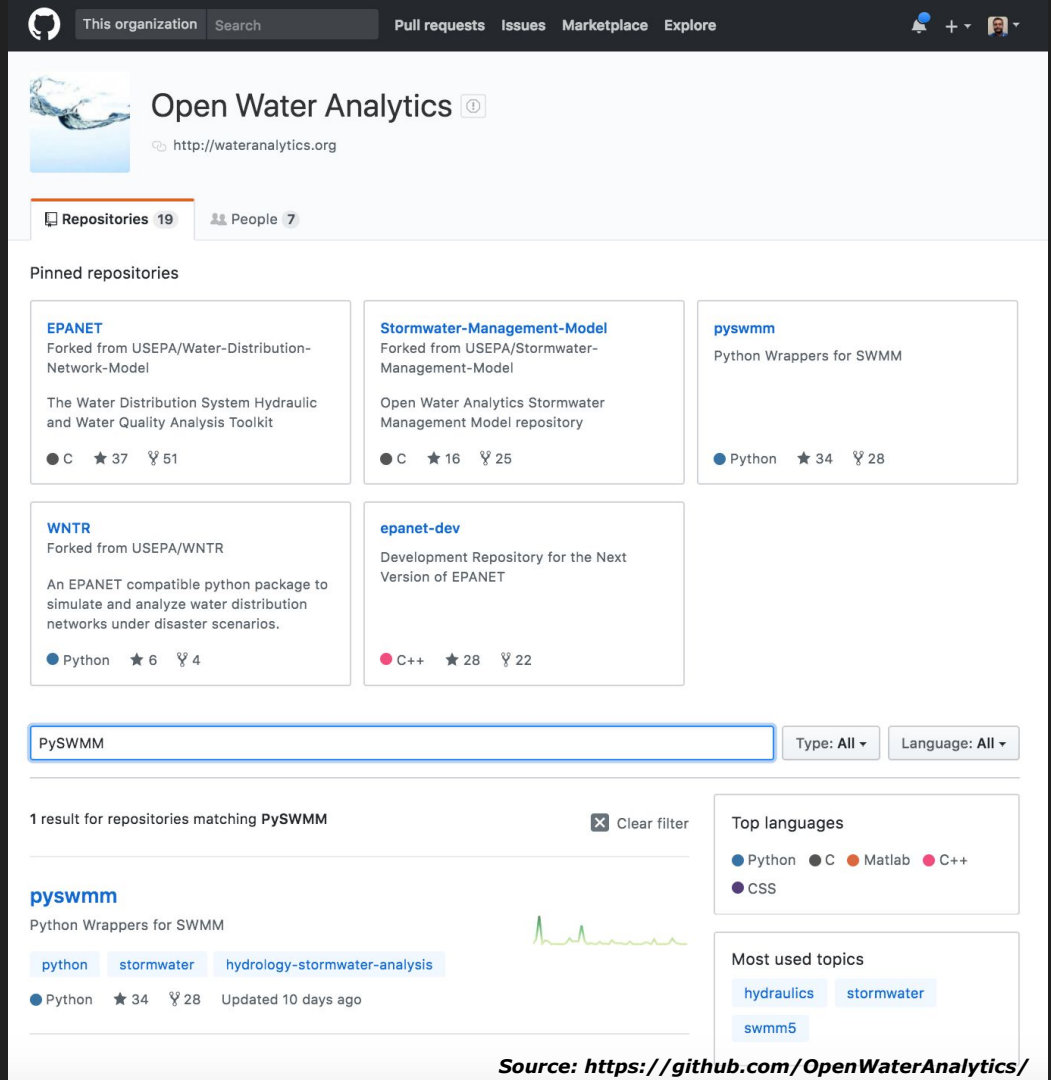
Sam Hatchet

emnet



2018/02/06



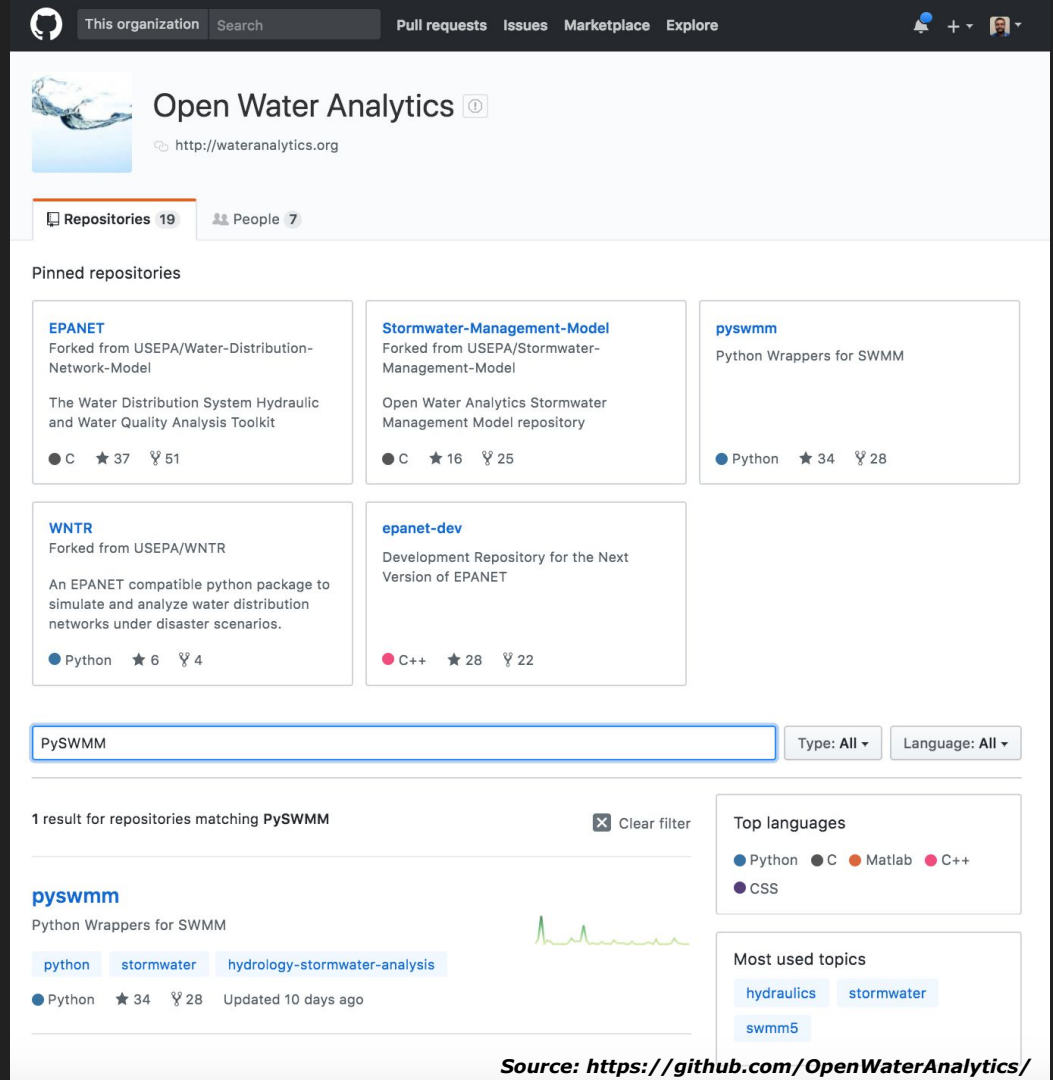


Source: https://github.com/OpenWaterAnalytics/

Open Water Analytics (2015)

Why should we care about Open Source?

- Let's move our community away from a single developer's vision
- Else, Research Projects Grow and Die
- Combine Industry with Academia and Open Source developers

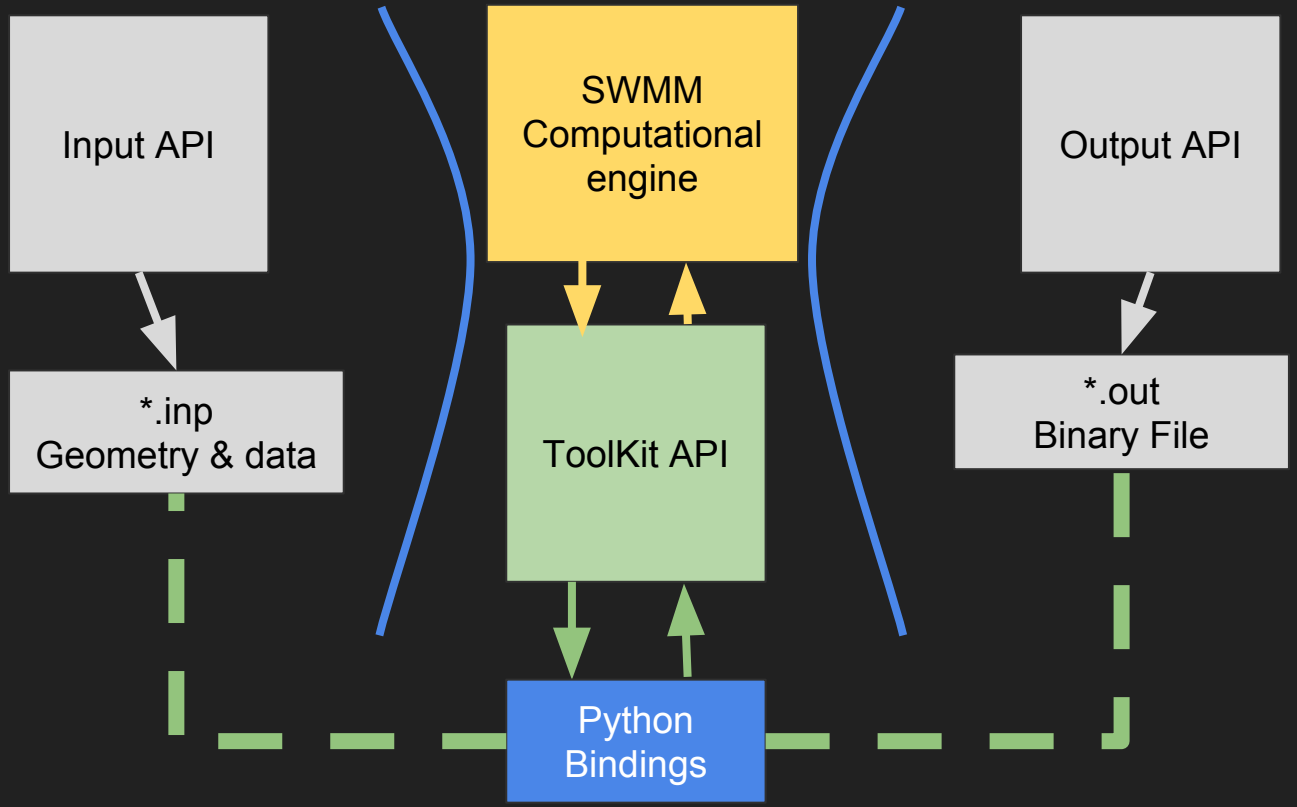


Source: https://github.com/OpenWaterAnalytics/

SWMM API:

- Exposing the Data Model
- Observe Simulated Results During Run

SWMM API: General Framework



PySWMM API:

- Run Simulation
- Link Settings
- Control Rules in Python
- Set Node Inflows

SWMM API: Run a simulation

```
>>> from pyswmm import Simulation
>>>
>>> with Simulation('./testmodel.inp') as sim:
...     for step in sim:
...         pass
...     sim.report()
```

SWMM API: Adjust Link Settings

```
>>> from pyswmm import Simulation, Links
>>>
>>> with Simulation('./testmodel.inp') as sim:
...     link_object = Links(sim)
...
...     #C1:C2 link instantiation
...     c1c2 = link_object["C1:C2"]
...     print(c1c2.flow_limit)
...     print(c1c2.is_conduit())
...
...     #Step through a simulation
...     for step in sim:
...             print(c1c2.flow)
...             if c1c2.flow > 10.0:
...                     c1c2.target_setting = 0.5
...
...     sim.report()
```

SWMM API: Adjust Link Settings

```
>>> from pyswmm import Simulation, Links
>>>
>>> with Simulation('./testmodel.inp') as sim:
...     link_object = Links(sim)
...
...     #C1:C2 link instantiation
...     c1c2 = link_object["C1:C2"]
...     print(c1c2.flow_limit)
...     print(c1c2.is_conduit())
...
...     #Step through a simulation
...     for step in sim:
...             print(c1c2.flow)
...             if c1c2.flow > 10.0:
...                     c1c2.target_setting = 0.5
...
...     sim.report()
```

Query information for every time step

Control based on varying conditions

SWMM API: Build Custom Control Rules

```
>>> from pyswmm import Simulation, Links, Nodes
>>>
>>> def TestDepth(node, node2):
>>>     if node > node2:
>>>             return True
>>>     else:
>>>             return False
>>>
```

```
>>> with Simulation('./testmodel.inp') as sim:
...     link_object = Links(sim)
...
...     #C1:C2 link instantiation
...     c1c2 = link_object["C1:C2"]
...
...     node_object = Nodes(sim)
...     #J1 node instantiation
...     J1 = node_object["J1"]
...     #J2 node instantiation
...     J2 = node_object["J2"]
...
...     #Step through a simulation
...     for step in sim:
...             if TestDepth(J1.depth, J2.depth):
...                     c1c2.target_setting = 0.5
...
...     sim.report()
```

Expanded PySWMM API:

- No API (yet) to construct a network programmatically
- Needs to generate an input file “manually” or using the GUI.

Source: https://www.sciencedirect.com/science/article/pii/S1877705815025941

Expanded PySWMM API:

- No API (yet) to construct a network programmatically
- Needs to generate an input file “manually” or using the GUI.

ELSEVIER

CrossMark

Available online at www.sciencedirect.com

ScienceDirect

Procedia Engineering 119 (2015) 710 – 718

Procedia Engineering

www.elsevier.com/locate/procedia

13th Computer Control for Water Industry Conference, CCWI 2015

OOPNET: An object-oriented EPANET in Python

D. Steffelbauer[a*] and D. Fuchs-Hanusch[a]

[a]*Institute of Urban Water Management and Landscape Water Engineering*
Graz University of Technology, Stremayrgasse 10/I, A-8010 Graz, Austria

Abstract

Several attempts of the past aimed to convert EPANET into a bigger open-source project by rewriting EPANET in an object-oriented way. We introduce a Python based object-oriented EPANET (OOPNET) with the purpose to address water engineers that might be not so familiar with complex programming languages like C++ or Java. EPANET input files are translated into the object oriented structure of OOPNET and manipulated and simulated with EPANET’s command-line interface through Python. The replacement of EPANET by a hydraulic solver in Python is still ongoing and has the purpose to provide a solution completely written in one programing language.

Peer-review under responsibility of the Scientific Committee of CCWI 2015

Keywords: Water Distribution, Parallel Computing, Open Source Project, Simulaton Software, Genetic Algorithms, Optimization

Source: https://www.sciencedirect.com/science/article/pii/S1877705815025941

Example input file *.inp

```
[CONDUITS]
;;               Inlet            Outlet                      Manning    Inlet      Outlet     Init.      Max.
;;Name           Node             Node             Length     N          Offset     Offset     Flow       Flow
;;-------------- ---------------- ---------------- ---------- ---------- ---------- ---------- ---------- ----------
C1               J1               J5               185.00     0.05       0          0          0          0
C2               J2               J11              526.00     0.016      0          4          0          0
C3               J3               J4               109.00     0.016      0          0          0          0
C4               J4               J5               133.00     0.05       0          0          0          0
C5               J5               J6               207.00     0.05       0          0          0          0
C6               J7               J6               140.00     0.05       0          0          0          0
C7               J6               J8               95.00      0.016      0          0          0          0
C8               J8               J9               166.00     0.05       0          0          0          0
C9               J9               J10              320.00     0.05       0          0          0          0
C10              J10              J11              145.00     0.05       0          0          0          0
C11              J11              SU1              89.00      0.016      0          1          0          0
C_out            J_out            O2               100        0.01       0          0          0          0
```

API Proof of Concept

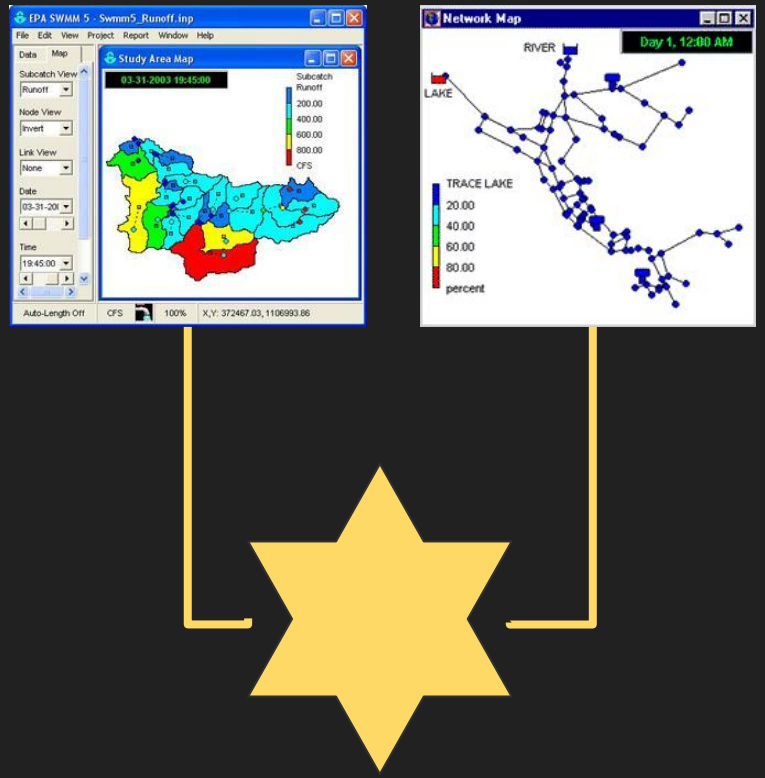
```
# Create Model
model = Model(title='Example 3', description='Detention Pond')

# Create Timeseries (Notice dt_range was passed as a copy of the range, we could do this internally?)
ts_2y = model.create_timeseries(index=dt_range, values=raindata_2y5min, relative=True)
ts_5y = model.create_timeseries(index=dt_range, values=raindata_5y5min, relative=True)
ts_10y = model.create_timeseries(index=dt_range, values=raindata_10y5min, relative=True)

# Create Raingages
rg = model.create_raingage('RainGage', x=-148.485, y=1207.602, timeseries=ts_2y)

# Create Cross Sections/Shapes to be used with conduits
xs_pipe1 = Circular(diameter=2.25)
xs_pipe2 = Circular(diameter=3.5)
xs_pipe3 = Circular(diameter=4.75)
xs_channel1 = Trapezoidal(max_height=3, width=5, left_slope=5, right_slope=5)
xs_channel2 = Trapezoidal(max_height=1, width=0, left_slope=0.0001, right_slope=25)

# Create Junctions/Nodes
j1 = model.create_junction('J1', x=648.532, y=1043.713, invert_elevation=4973)
j2 = model.create_junction('J2', x=648.532, y=1043.713, invert_elevation=4969)
j11 = model.create_junction('J3', x=648.532, y=1043.713, invert_elevation=4973)

# Create Conduits/Pipes
c1 = model.create_conduit('C1', from_node=j1, to_node=j2, length=185, tag='Swale')
c2 = model.create_conduit('C2', from_node=j2, to_node=j11, length=526, tag='Gutter')
```

Why should you care about the API?

- Gives you a window to your model
- Optimization packages
- New Inflow algorithms
- Control Rules in Python
- Get more out of your model
- Watch simulated results while running

- *Something we haven’t thought of yet*

Roadmap and future work - Devs



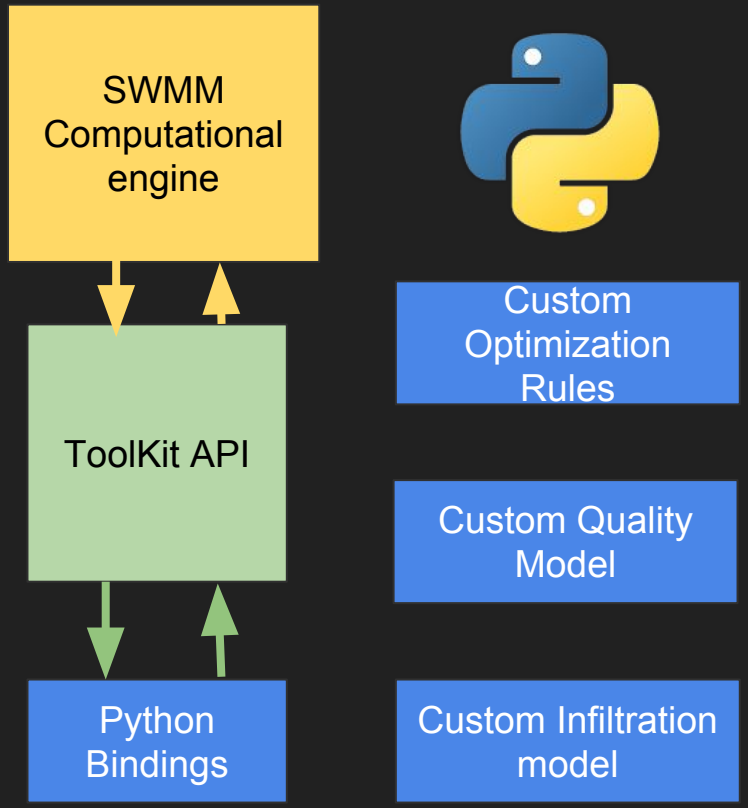
Create a common core for EpaNet and SWMM

C -> C++

Roadmap and future work - Devs

1. **Unit tests**
2. **Create Conda Packages**
3. **Continuous Integration**
4. **Official cross platform support**
    a. **Linux**
    b. **Mac**

Roadmap and future work - Academia



Create a pythonic interface to allow so that new models can be created in Python and plugged to the numerical engine.

We are **NOT** hiring (yet!)

But if you…

- ❏ Like C?
- ❏ Like C.I.?
- ❏ Like C++?
- ❏ Like SWIG?
- ❏ Like Python?
- ❏ Like Bindings?
- ❏ Like Hydrology?
- ❏ Like Hydraulics?
- ❏ Like Unit Testing?
- ❏ Like 💩 Modelling?
- ❏ Like Open Source?
- ❏ Like Water Quality?

… let’s talk!

Questions & (hopefully) Answers

@goanpeca

goanpeca@gmail.com

Source: http://www.revistalaocaloca.com/2017/04/el-poporo-quimbaya