

# GitHub Actions to Improve Software QA/QC: Open Source SWMM

By Michael E. Tryby Ph.D.

Office of Research and Development  
Center for Environmental Solutions and Emergency Response

# Disclaimer

The views expressed in this presentation are those of the author and do not necessarily reflect the views of policies of the US Environmental Protection Agency. It has been subjected to review by the Office of Research & Development and approved for presentation. Mention of trade names or commercial products does not constitute endorsement or recommendation for use.

# SWMM Solver

- Dynamic simulation of stormwater systems
- *De facto* international standard
- Typical scientific and engineering application
- 40,000 lines, C library, delivered as dll
- QA/QC informal and *ad hoc*

# Motivation

- Open source development model
- Volunteer development team
- Adopt best practices
- Make QA/QC more systematic and transparent

# What is CI?

- Continuous Integration (CI) is a software development best practice
- Automates build, test, packaging, and deployment workflows
- We use CI to help improve project QA/QC
- CI workflows run in the cloud
- Requires a service linked to project repository

# What is Actions?

- Actions is a “CI” service integrated directly with GitHub
- No third-party integrations required
- Similar to Travis, Appveyor, and Circle but avoids licensing issues
- Actions are triggered by Git push and pull events
- Every change to the repository gets built and tested

# Basic CI Workflow Steps

```
build-and-test.yml
# build-and-test.yml - GitHub Actions CI for swmm-solver
#
# Created: May 19, 2020
# Updated: May 31, 2020
#
# Author: Michael E. Tryby
#        US EPA - ORD/CESER
#
name: Build and Test
on:
  push:
    branches: [ develop ]
  pull_request:
    branches: [ develop ]
jobs:
  win_build:
    runs-on: windows-2016
    defaults:
      run:
        shell: cmd
    env:
      OMP_NUM_THREADS: 1
      PROJECT: swmm
      BUILD_HOME: build
      TEST_HOME: nrtests
    steps:
      - name: Checkout
        uses: actions/checkout@v2
      - name: Setup python
        uses: actions/setup-python@v2
        with:
          python-version: '3.7'
      - name: Install requirements
        run: |
          python -m pip install --upgrade pip
          python -m pip install -r tools/requirements-win.txt
      - name: Build and unit test
        run: tools/make.cmd /t
      - name: Build for reg test
        run: tools/make.cmd
      - name: Before reg test
        run: tools/before-nrtest.cmd
      - name: Run reg test
        run: tools/run-nrtests.cmd %GITHUB_RUN_ID%_%GITHUB_RUN_NUMBER%
      - name: Upload artifacts
        if: ${{ always() }}
        uses: actions/upload-artifact@v2
        with:
          name: nrtest-artifacts
          path: upload/
```

- Prepare Worker
- Checkout
- Install
- Prepare Build
- Build
- Prepare Test
- Test
- Upload Artifacts

# Actions Demo

# SWMM CI Dev Stack

- Visual Studio: 32 - 64 Bit Windows builds
- cmake, ctest: build and unit test automation
- Boost: unit test framework
- Custom Library: SWMM output file reader
- Python: wrappers for SWMM libraries
- Custom nrttest Plugin: Python package for numerical regression testing
- Custom Scripts: build worker automation
- Tests and Benchmarks: separate repo for configuration management

# Advantages

- Essential for team development
- Encourages frequent small updates to code
- Detects bugs sooner
- Improves developer productivity
- Automation makes QA actionable and consistent

# Disadvantages

- Requires considerable investment to setup
- Configuration can be difficult
- Requires buy-in from developers
- Requires ongoing maintenance
- Too effort intensive for small projects

# One More Thing

- Open Source EPANET: Drinking Water Distribution Model