ODM Tools Python: Open Source Software for Managing Hydrologic and Water Quality Time Series Data

Jeffery S. Horsburgh

Stephanie Reeder
James Patton
Amber Spackman-Jones

Utah State University
UWRL
iUTAH
Motivation
Gradients Along Mountain To Urban Transitions (GAMUT) Network

- Ecohydrologic observatory deployed in 3 watersheds: Logan River, Red Butte Creek, Provo River
- Watersheds with similar water source (high elevation snow) but different land use transitions
- Measures aspects of water inputs and outputs and water quality over mountain-to-urban gradient
- Mix of aquatic and terrestrial *in situ* and relocatable sensors
Motivation

- Sediment in sensor cup
- Sensor drift and calibration shift
- Strange anomalies
- Dead battery
Some History

- ODM Tools originally developed as part of the CUAHSI Hydrologic Information System
- Developed in Microsoft Visual Studio .Net
- Limited to Windows Machines
- Only worked with Microsoft SQL Server databases
- Provided editing tools, but did not preserve the history of edits
Observations Data Model
ODM Tools Python: Design Goals

• Muti-platform support (Windows, Linux, Mac)
• Multi-database support (Microsoft SQL Server and MySQL)
• Implement a scripting interface to save the provenance of data edits in QC process
• Modernize the Graphical User Interface (GUI)
Graphical User Interface

- Ribbon
- Toolbar
- Plot Window
- Time Series Selection
Data Visualization

Multiple Plot Types
Plot Display Options
Date Range Restrictions
Dynamic Zooming and Panning
Multiple Time Series Selection
Data Visualization: Plot Types
Query and Export

Build a query

Right click select and export
Data Editing for Quality Control

Common editing tools

Start editing and save edits

Linked plot view

Data selection filters

Tabular data view and selection
Data Editing for Quality Control

Python Code Console

Python Script Editor
How does it work?
Step 1: Select a Time Series for Editing

- **Edit Series**

- **Select Series**

**Drift and calibration shift**
Step 2: Select Data to Edit
Step 3: Linear Drift Correction
Step 4: Interpolate
Step 5: Flag
Step 6: Save Modified Data Series
Recording Edits

Automatically generated Python code with each editing step
Ultimate Objective

• Transform raw sensor data to quality controlled data in a repeatable way
Summary

• ODM Tools Python is a cross platform (Windows, Mac) software for sensor data management

• Visualization capabilities are helpful in screening new data as they arrive

• ODM Tools provides GUI-based and scripting of data quality control edits
Questions?

OPEN SOURCE CODE REPOSITORY:
ODM Tools Python is available in GitHub
https://github.com/UCHIC/ODMToolsPython