SAND2018-6123C

Open source software packages developed at Sandia National Laboratories and their use in water distribution system analysis

Katherine A. Klise[1], Carl D. Laird[1], David B. Hart[1], Bethany Nicholson[1], Michael Bynum[1], Regan Murray[2]

1 Sandia National Laboratories

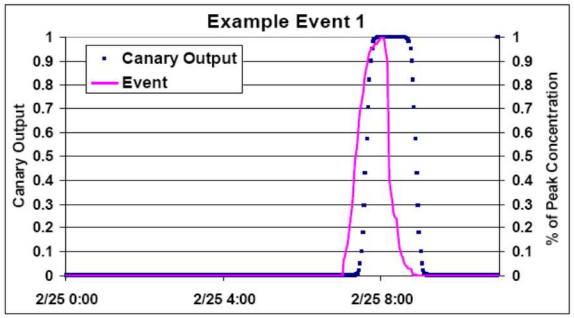
2 United States Environmental Protection Agency

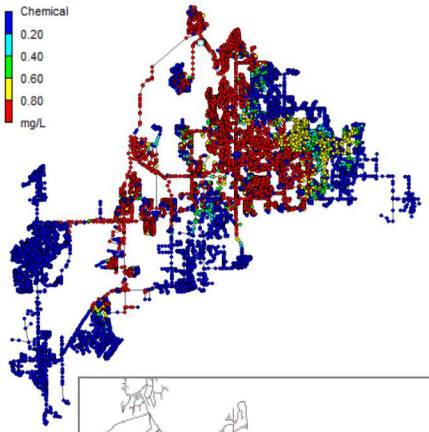
Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

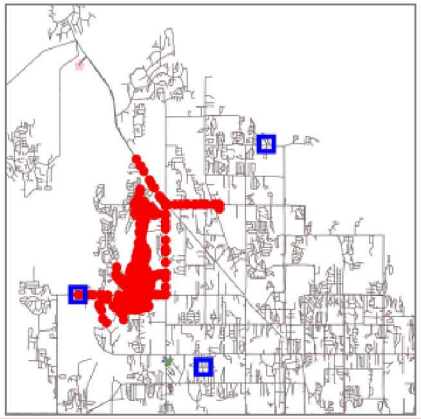
Background

Sandia and the EPA have a long history of developing open source software packages for water distribution systems analysis.

- CANARY (Matlab, Java)
  - Event detection algorithms for online sensors, alerts water utilities if water quality becomes anomalous
  - First released 2009
- TEVA-SPOT (C and Python)
  - Simulate contaminant scenarios and optimize sensor placement to minimize impact
  - First released 2009
- WST (C and Python)
  - Optimize response action plans to a contamination event
  - First released 2012
- WNTR (Python)
  - Simulate and analyze disaster and recovery scenarios
  - First released 2016



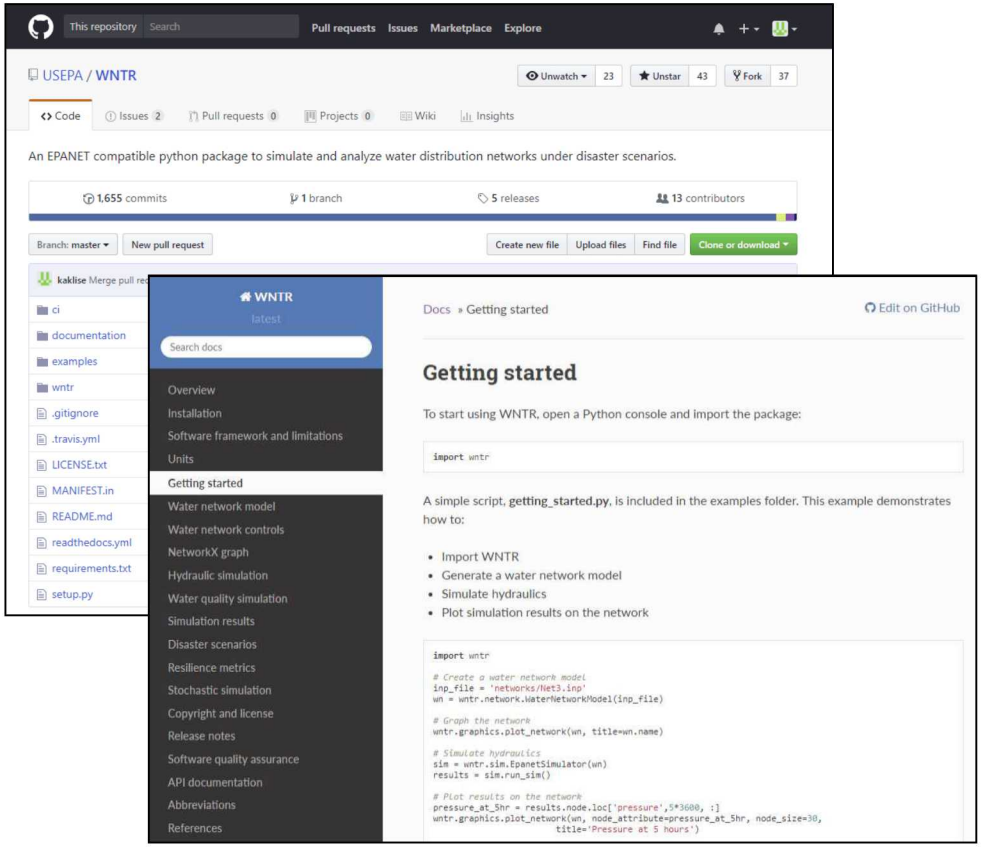




Python Software Framework

Sandia National Laboratories

Advances in open source Python packages and software development tools have driven recent development in WNTR



ANACONDA

NumPy

SciPy

NetworkX

pandas

$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$

matplotlib

plotly

GitHub

Travis CI

Read the Docs

New Software Packages

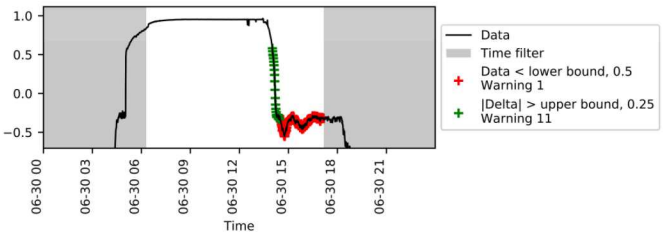
Sandia recently released two open source software packages that build on capabilities in Canary and TEVA-SPOT

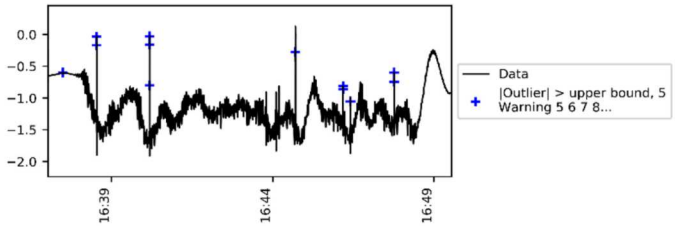
- Pecos (Python)
  - Monitor sensor data, perform quality control checks, generate reports, and send alerts if sensor readings indicate anomalous conditions
  - First released 2016
- Chama (Python)
  - Optimize sensor locations and technologies to maximize monitoring effectiveness
  - Mixed integer programming formulations using Pyomo
  - First released 2017

Pecos

Pecos was designed to:

- Analyze large amounts of time series data across multiple systems and locations
- Run quality control tests
- Alert system operators when the system has changed
- Generate reports and dashboards
- Collect performance statistics to track long term system health





Current use cases

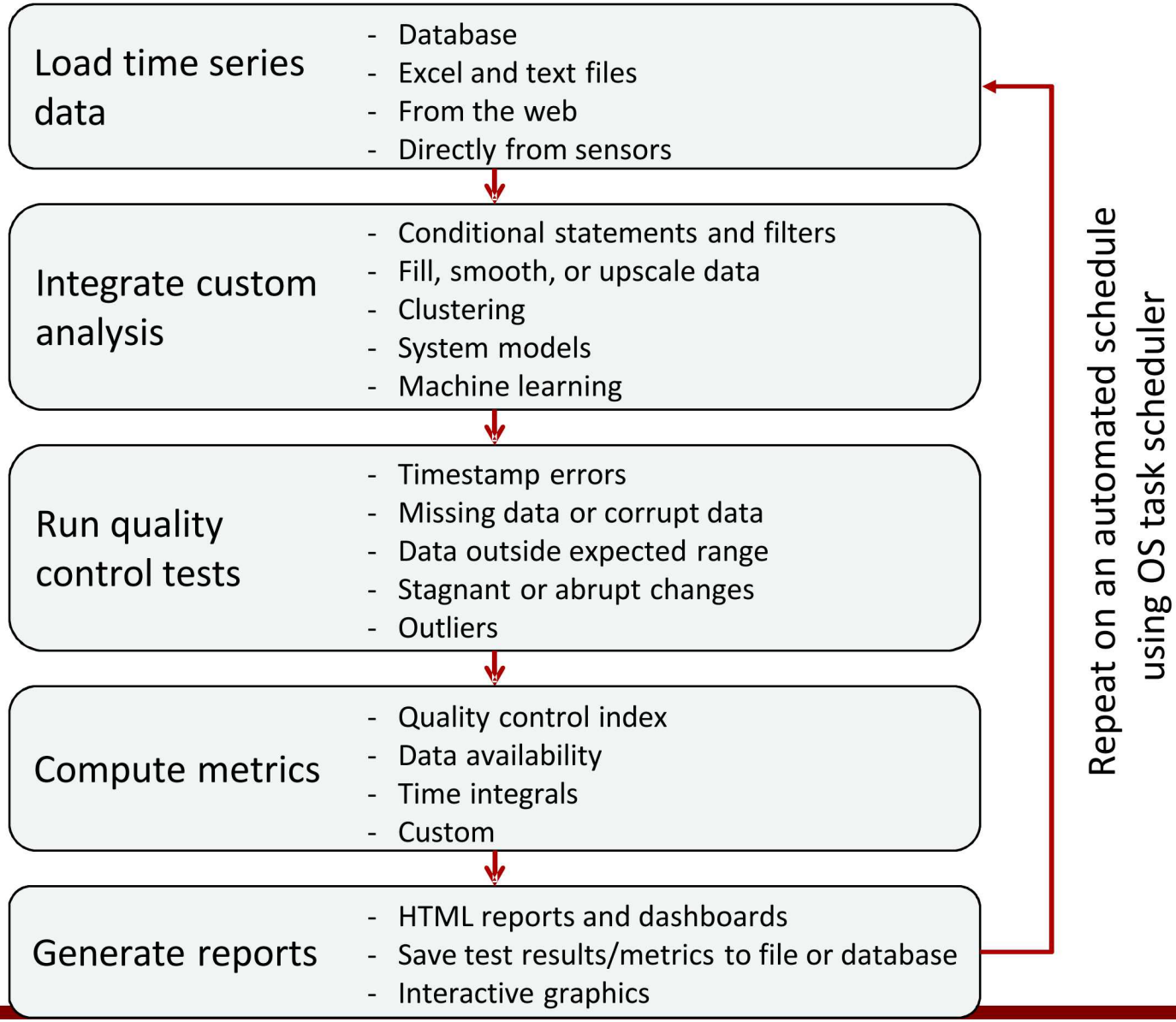
- Solar photovoltaics
- Weather stations
- Marine hydrokinetics
- Surface water quality

	Variable Name	Start Date	End Date	Timesteps	Error Flag
1		2015-01-01 05:00:00	2015-01-01 05:00:00	1	Missing timestamp
2		2015-01-01 17:00:00	2015-01-01 17:00:00	1	Duplicate timestamp
3		2015-01-01 19:30:00	2015-01-01 19:30:00	1	Nonmonotonic timestamp
4	A	2015-01-01 12:15:00	2015-01-01 14:30:00	10	|Increment| < lower bound, 0.0001
5	B	2015-01-01 06:30:00	2015-01-01 06:30:00	1	Data < lower bound, 0
6	B	2015-01-01 15:30:00	2015-01-01 15:30:00	1	Data > upper bound, 1
7	C	2015-01-01 07:30:00	2015-01-01 09:30:00	9	Corrupt data

Pecos Dashboard

	system1	system2	system3	system4	system5
location1	Link to Report	Link to Report	Link to Report	Link to Report	Link to Report
location2	Link to Report	Link to Report	Link to Report	Link to Report	Link to Report
location3	Link to Report	Link to Report	Link to Report	Link to Report	Link to Report

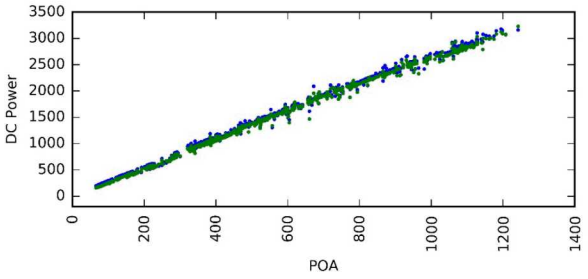
Pecos Flowchart



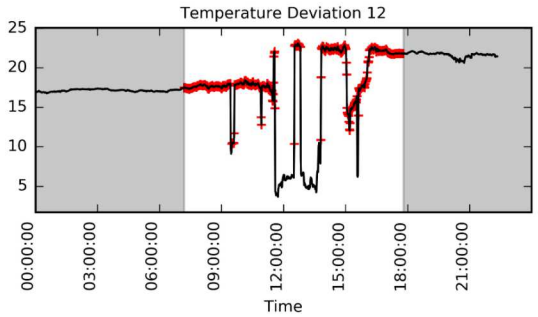
Pecos PV Application

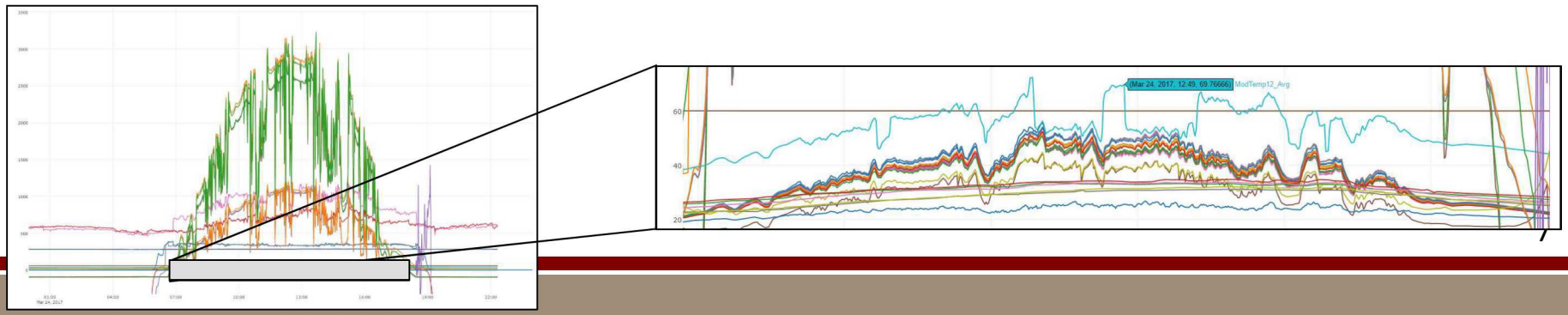
Pecos was developed to monitor photovoltaic and weather data collected at DOE Regional Test Centers (29 systems at 5 sites across the US)

- Analyzing 2 million data points per day
- Track measurement to model error and sensor drift
- Filter data based on sun position
- Dashboards are sent by email with links to detailed reports and interactive graphics
- Color-coded score given to each data type



	New Mexico	Florida	Vermont	Nevada
Baseline	Irradiance 1.00 Temperature 1.00 Current 1.00 Voltage 1.00 Power 0.64 Detailed Report Interactive Plot	Irradiance 1.00 Temperature 0.19 Current 1.00 Voltage 1.00 Power 0.93 Detailed Report Interactive Plot	Irradiance 1.00 Temperature 1.00 Current 0.27 Voltage 1.00 Power 0.24 Detailed Report Interactive Plot	Irradiance 1.00 Temperature 1.00 Current 1.00 Voltage 1.00 Power 1.00 Detailed Report Interactive Plot

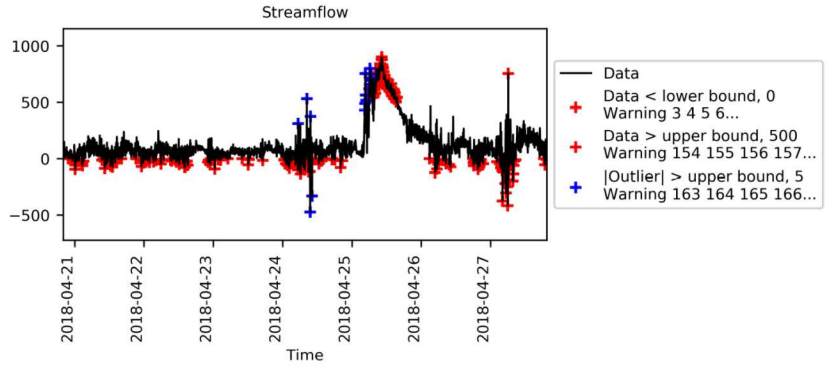


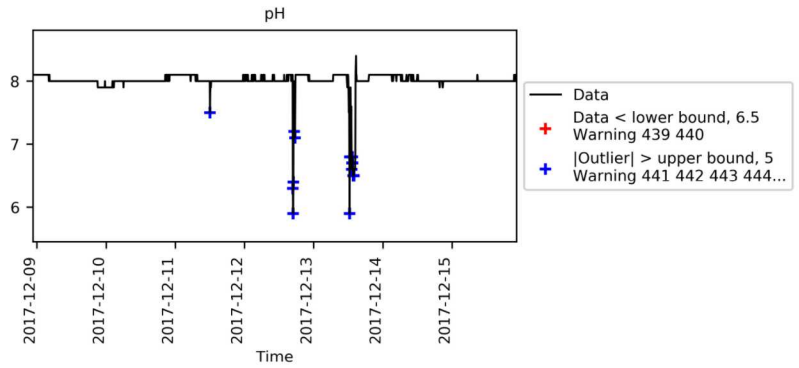


Pecos Village Blue Application

- EPA and USGS project to monitor water quality in the Baltimore Harbor
- Pecos was used to run QC tests on data every 7 days. Data retrieved from https://waterservices.usgs.gov
- Simple quality control tests include
  - Check to make sure data is collected at least every 15 minutes
  - Check for missing data
  - Check for corrupt data
  - Check to see if data is with the expected ranges
  - Check for outliers
- Results were gathered in an HTML report
- An interactive plot of the raw data was created
- Analysis is run on an automated schedule

	Min	Max
Dissolved oxygen (mg/L)	5	50
Gage height (ft above ref)	-2	2
Nitrate plus nitrite (mg/L)	0	10
Specific conductance (µS/cm at 25C)	0	5000
Streamflow (ft3/s)	0	500
Temperature (C)	0	21
Turbidity (FNU)	0	10
pH (-)	6.5	9





Pecos Conclusions

Sandia National Laboratories

- General purpose time series performance monitoring and alert package
- Run quality control tests on historical or real time data
- Facilitates very flexible analysis, graphics, and reporting options
- Pecos does not currently include the same algorithms as Canary
- Plans to add more streaming algorithms and history/training window

Pecos ✓
Time series data monitoring and alert

https://github.com/sandialabs/pecos

Chama

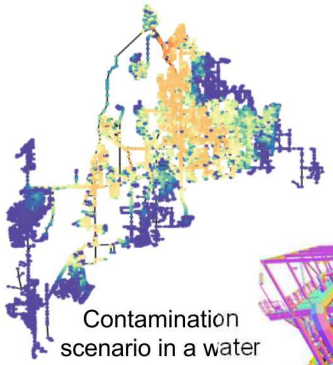
Chama was developed to be a general purpose sensor placement optimization software tool

- Integrate simulations from third party software
- Integrate a wide range of sensor technologies
- P-median formulation from TEVA-SPOT and WST used to minimize impact
- Additional coverage formulation used to maximize scenario or geographic coverage
- Flexible data and graphics options

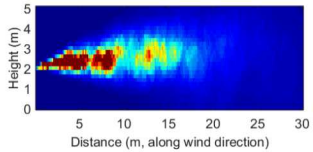
Current use cases

- Methane emissions
- Fire detection
- Water quality

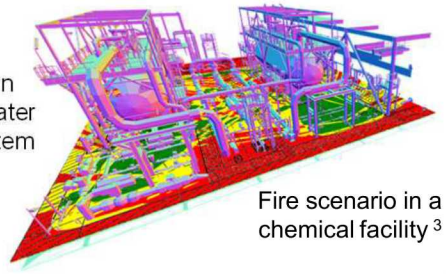
Integrate simulations from third party software



Contamination scenario in a water distribution system [1]



Gas leak from a production well pad [2]

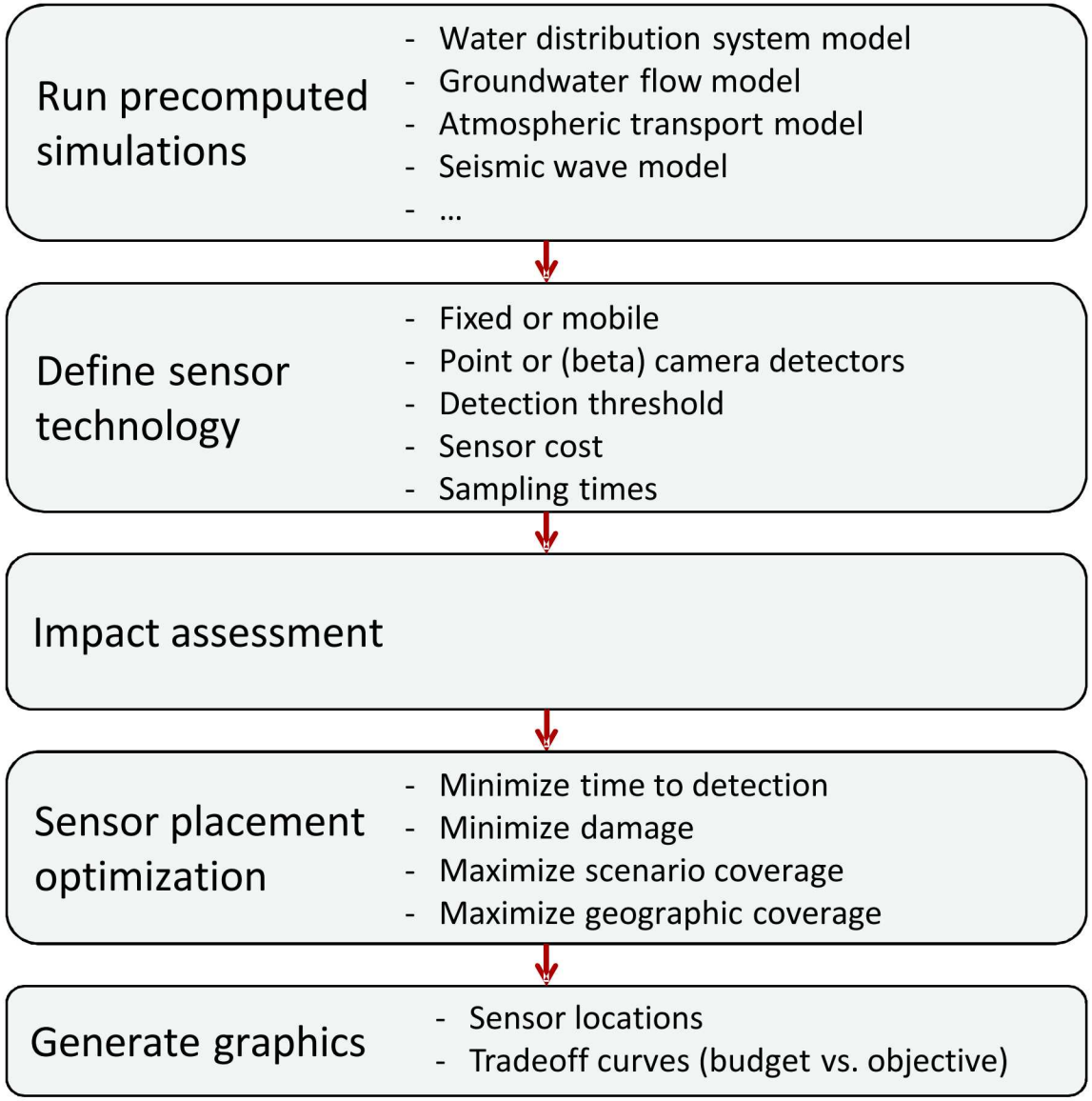


Fire scenario in a chemical facility [3]

Integrate a wide range of sensor technologies

[1] Water Network Tool for Resilience, http://wntr.readthedocs.io
[2] Modified Gaussian plume simulation
[3] Kenexis Effigy, https://www.kenexis.com/software/effigy/

Chama Flowchart

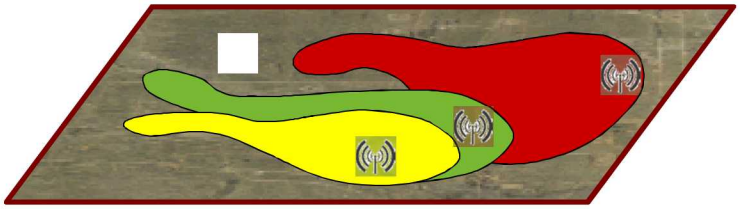


Input data for optimization

Scenario	Sensor	Detection times
S1	A	[5, 8, 12]
S1	B	[3, 5, 7]
S2	A	[]
S2	B	[1,6, 11]
…		

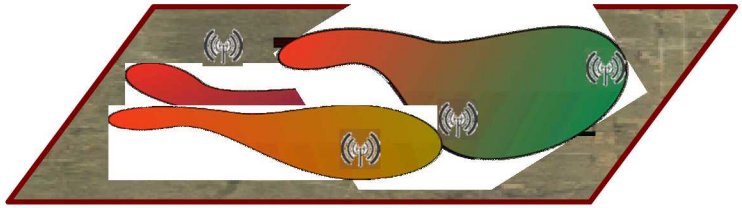
Sensor	Scenario coverage
A	[S1]
B	[S1, S2]
…	

Maximize scenario coverage



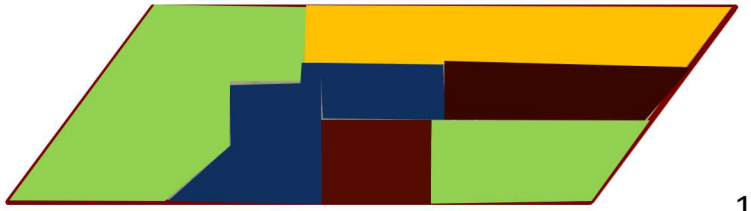
Scenario	Sensor	Minimum detection time	Damage metric
S1	A	5	100
S1	B	3	50
S2	A	NA	NA
S2	B	1	15
…			

Minimize time to detection or damage



Sensor	Geographic coverage
A	[School, Zone1]
B	[School, Hospital, Zone2]
…	

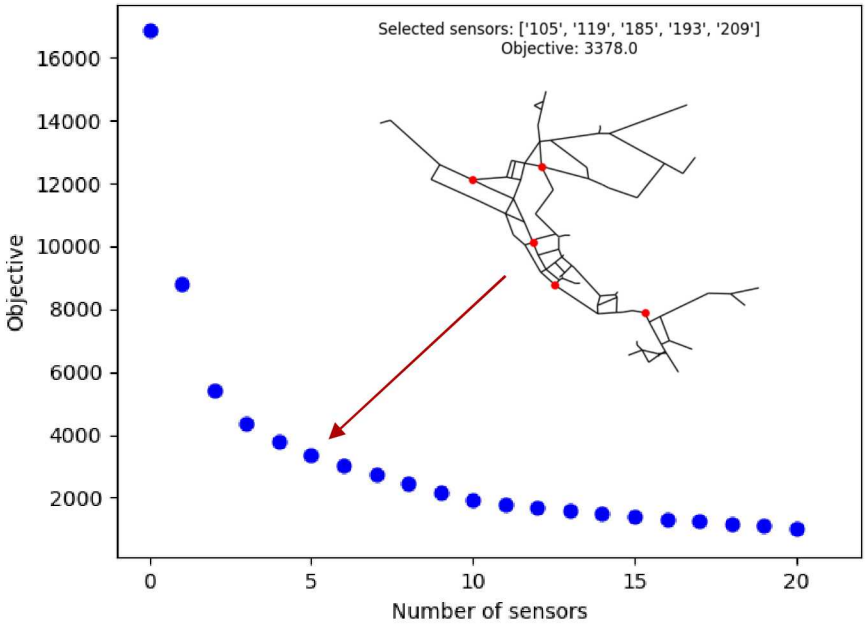
Maximize geographic coverage



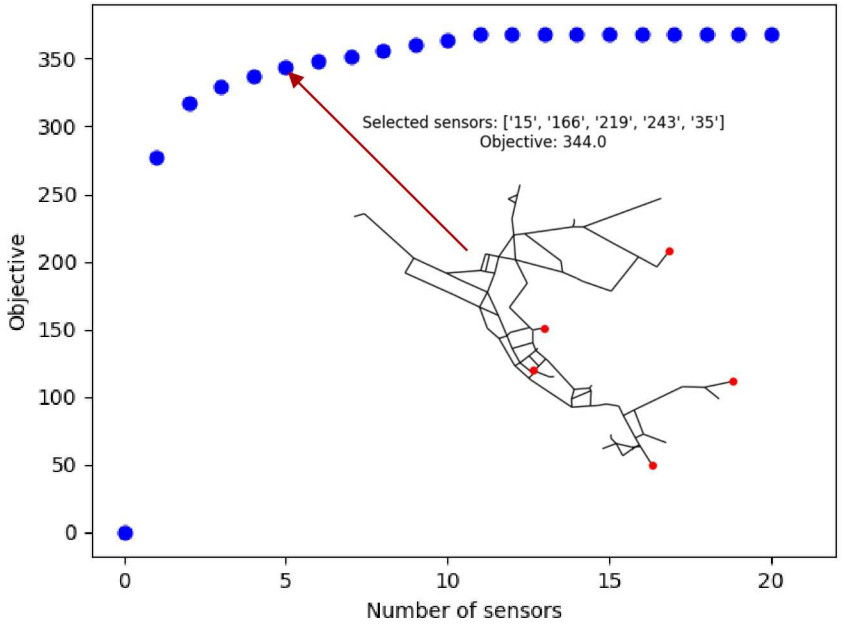
Chama WDS Application

- Integrate WNTR with Chama
- Precompute water contamination or other type of scenarios
- Define sensors with different costs and detection thresholds
- Given a sensor budget ($ amount or number of sensors):
  - Minimize time to detection, extent of contamination, or other metrics
  - Maximize coverage of scenarios or scenario-time pairs

Minimize Extent of Contamination



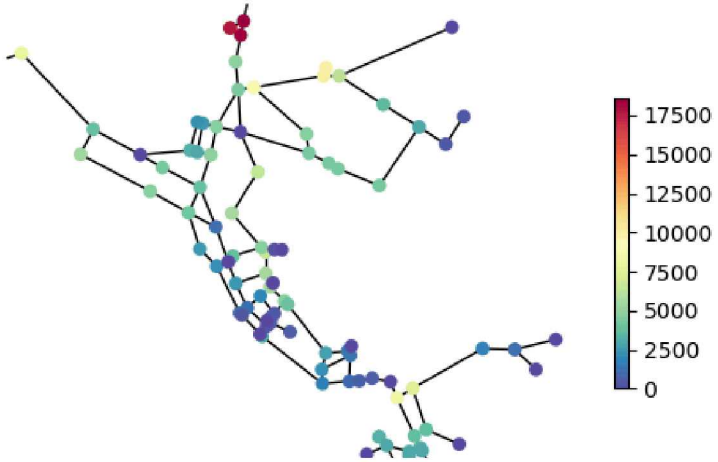
Maximize scenario coverage



* 368 contamination scenarios, 97 potential sensor locations

Chama WDS Application

Results include an evaluation of the sensor placement

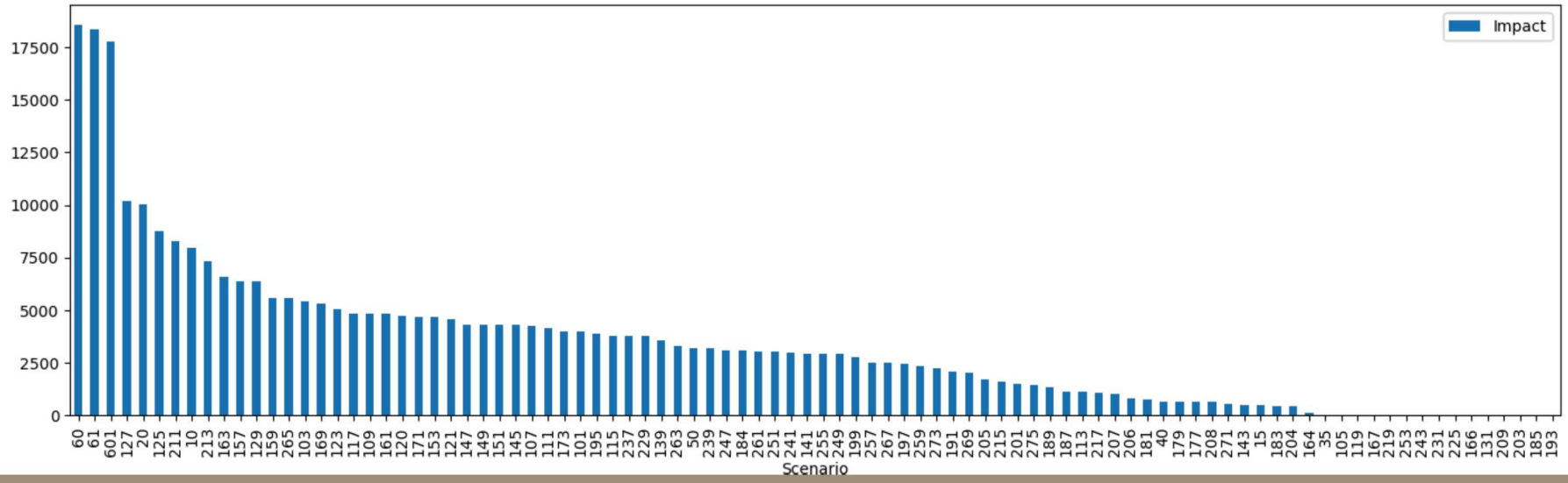


Scenario detection

```
S1: []
S2: [219, 243, 35]
S3: [166, 219, 35]
```

Sensor detection

```
15: []
166: [S3]
219: [S2, S3]
243: [S2]
35: [S2, S3]
```



Chama Conclusions

- General purpose sensor placement optimization package
- Facilitates very flexible analysis and graphics options
- Chama includes the same P-median formulation as TEVA-SPOT and WST
- Chama includes coverage formulations and can be extended to include additional optimization methods
- Plans to add additional optimization methods and sensor models

Chama
SENSOR PLACEMENT OPTIMIZATION

https://github.com/sandialabs/chama

WNTR

WNTR was designed to analyze water distribution networks considering disaster scenarios and recovery actions

- Quantify resilience for a wide range of hazards
  - Pipe breaks
  - Power outages
  - Contamination incidents
  - Earthquakes
  - Landslides
  - Hurricanes
  - Cyber attacks
- Evaluate and prioritize resilience-enhancing actions
  - Isolate and repair pipe breaks
  - Change valve and tank operation to maintain water service
  - Install backup generation
  - Plan flushing or water conservation mandates
  - Evaluate fire fighting capacity

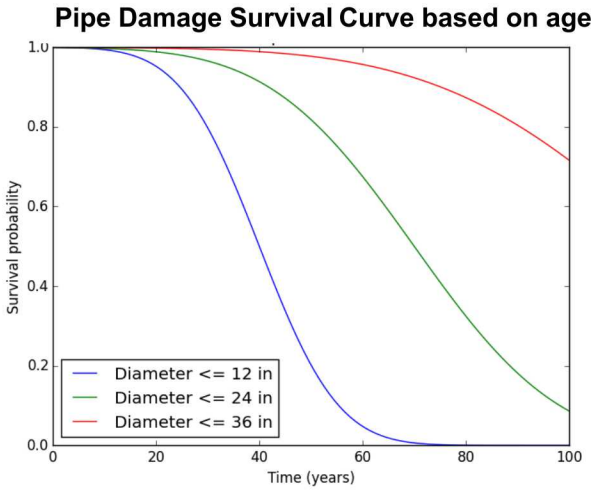
WNTR Capabilities

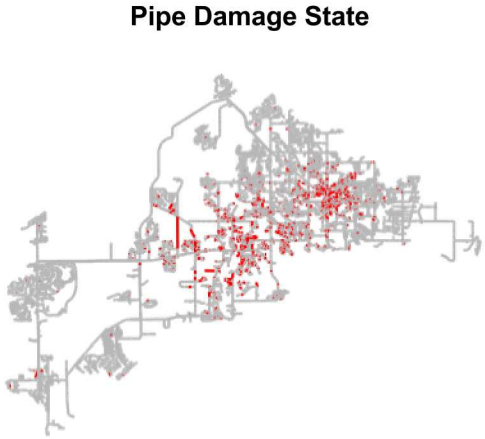
Features include

- Demand driven and pressure dependent demand hydraulics
- Pipe leak model
- Fragility/survival curves
- Resilience metrics, including hydraulic, topographic, water quality, and cost metrics
- Network graphics, time series plots, animations

Example use cases

- Generate network model from an EPANET INP file to analyze topography
- Run component criticality analysis
- Perform deterministic or stochastic analysis of disaster scenarios and recovery actions

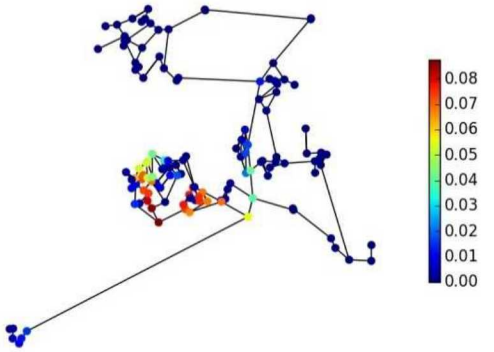




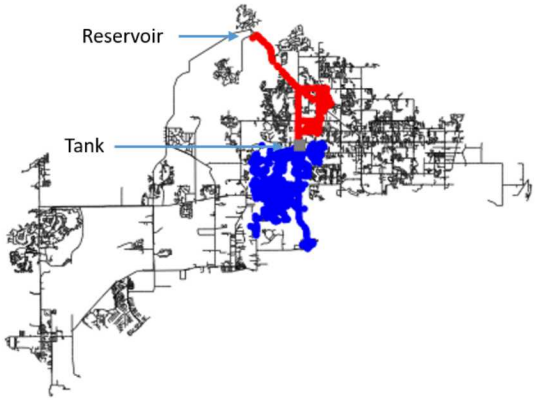
WNTR Applications

Sandia National Laboratories

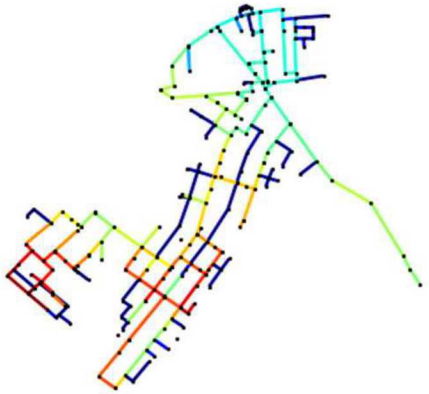
Betweenness Centrality



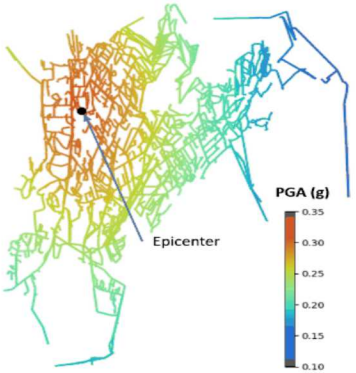
Hydraulic connectivity analysis

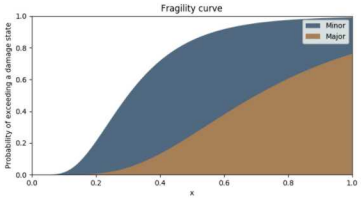


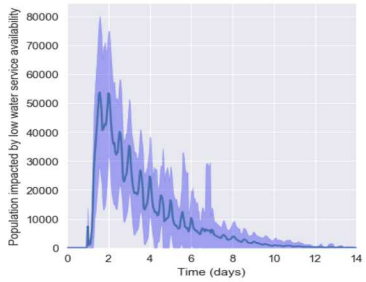
Pipe criticality analysis



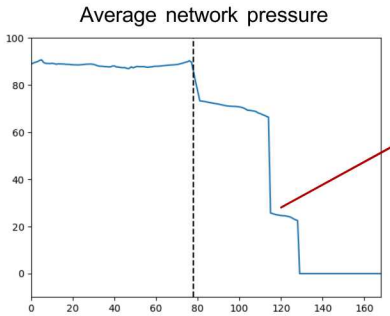
Earthquake disaster and recovery analysis

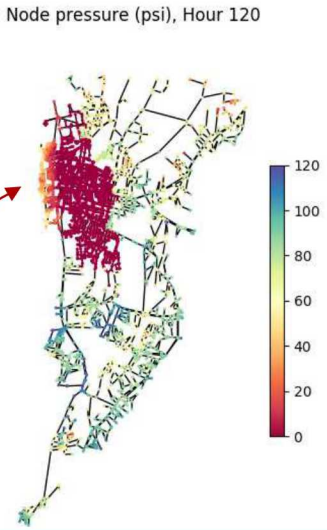






Power outage or compromised source water analysis





Conclusions

- WNTR, Chama, and Pecos are built using a similar set of Python packages, making the software very flexible and extensible
- The software can be used together to create early warning detection systems for water distribution systems
- Chama and Pecos can also be applied to a wide range of applications for general purpose sensor placement and event detection
- Feature requests and code contributions are encouraged!

Pecos
Time series data monitoring and alert
https://github.com/sandialabs/pecos

Chama
SENSOR PLACEMENT OPTIMIZATION
https://github.com/sandialabs/chama

WNTR
Water Network Tool for Resilience
https://github.com/USEPA/WNTR