THE Water Research FOUNDATION

PROJECT NO.
4954

Integration of High-Frequency Performance Data for Microbial and Chemical Compound Control in Potable Reuse Treatment Systems

THE METROPOLITAN WATER DISTRICT OF SOUTHERN CALIFORNIA

CALIFORNIA Water Boards
STATE WATER RESOURCES CONTROL BOARD
REGIONAL WATER QUALITY CONTROL BOARDS

Integration of High-Frequency Performance Data for Microbial and Chemical Compound Control in Potable Reuse Treatment Systems

Prepared by:

Brian Pecson
Trussell Technologies, Inc.

Aleks Pisarenko
Trussell Technologies, Inc.

Troy Walker
Hazen and Sawyer

Co-sponsored by:
California State Water Resources Control Board
Metropolitan Water District of Southern California

2024

THE
Water
Research
FOUNDATION

The Water Research Foundation (WRF) is the leading research organization advancing the science of all water to meet the evolving needs of its subscribers and the water sector. WRF is a 501(c)(3) nonprofit, educational organization that funds, manages, and publishes research on the technology, operation, and management of drinking water, wastewater, reuse, and stormwater systems—all in pursuit of ensuring water quality and improving water services to the public.

For more information, contact:
The Water Research Foundation

1199 North Fairfax Street, Suite 900	6666 West Quincy Avenue	
Alexandria, VA 22314-1445	Denver, Colorado 80235-3098	www.waterrf.org
P 571-384-2100	P 303-347-6100	info@waterrf.org

©Copyright 2024 by The Water Research Foundation. All rights reserved. Permission to copy must be obtained from The Water Research Foundation.
WRF ISBN: 978-1-60573-672-3
WRF Project Number: 4954

This report was prepared by the organization(s) named below as an account of work sponsored by The Water Research Foundation. Neither The Water Research Foundation, members of The Water Research Foundation, the organization(s) named below, nor any person acting on their behalf: (a) makes any warranty, express or implied, with respect to the use of any information, apparatus, method, or process disclosed in this report or that such use may not infringe on privately owned rights; or (b) assumes any liabilities with respect to the use of, or for damages resulting from the use of, any information, apparatus, method, or process disclosed in this report.

Prepared by Trussell Technologies, Inc. and Hazen and Sawyer

This document was reviewed by a panel of independent experts selected by The Water Research Foundation. Mention of trade names or commercial products or services does not constitute endorsement or recommendations for use. Similarly, omission of products or trade names indicates nothing concerning The Water Research Foundation's positions regarding product effectiveness or applicability.

Acknowledgments

The project team would like to thank the City of San Diego for allowing us access to the North City Pure Water Demonstration Facility for onsite implementation of the event detection system. Additional thanks to the Orange County Water District, Veolia, and Hampton Roads Sanitation District for providing information via interview on the performance monitoring approach at their respective indirect potable reuse facilities. Finally, the project team wants to express our thanks to the Project Advisory Committee and The Water Research Foundation for their support and feedback throughout the duration of this project.

Research Team

Principal Investigators:
Brian Pecson, PhD
Trussell Technologies, Inc.
Aleks Pisarenko, PhD
Trussell Technologies, Inc.
Troy Walker, MIE (Aust)
Hazen and Sawyer

Project Team:
Billy Raseman, PhD, PE
Hazen and Sawyer
Elise Chen, PE
Trussell Technologies, Inc.
Tim Dinh
Trussell Technologies, Inc.
Ben Christensen
Trussell Technologies, Inc.
Bryce McFerrin
Trussell Technologies, Inc.
Rodrigo Tackaert, PE
Trussell Technologies, Inc.
Nolan Townsend
Hazen and Sawyer

WRF Project Subcommittee or Other Contributors

Phil Ackman
Sanitation District of Los Angeles County
Steve Allgeier
U.S. Environmental Protection Agency

Cindy Figueroa
California State Resources Control Board

Mehul Patel
Orange County Water District

Paul Rochelle
The Metropolitan Water District of Southern California

Medi Sinaki, PE
Santa Clara Valley Water District

WRF Staff

John Albert, MPA
Chief Research Officer

Grace Jang, PhD
Research Program Manager

Corina Santos
Project Coordinator

Megan Karklins
Lead Content Manager

Angelina Dinsmore
Communications Coordinator

Contents

Acknowledgments......iii
Tables......vi
Figures......viii
Acronyms and Abbreviations......ix
Executive Summary......xi

Chapter 1: Introduction......1
1.1 Introduction......1
1.2 Summary of Monitoring Advancements in WRF Projects......2
1.2.1 WRF project 1688......3
1.2.2 WRF project 1700......3
1.2.3 WRF project 4759......4
1.2.4 WRF project 4765......4
1.3 Importance of Pathogen Control in DPR......7
1.4 Response Time and Event Detection Systems......8
1.5 Goals and Summary......10

Chapter 2: Literature Review......13
2.1 Utility Surveys......13
2.2 Data Analysis for Engineering Applications......16
2.3 Machine Learning, AI, and Statistical Modeling......18
2.4 Statistical Process Control......18
2.5 Literature Review Summary......18

Chapter 3: Event Detection Framework......21
3.1 Data Storage......24
3.2 Data Screening......24
3.3 Data Flagging......25
3.4 Event Detection......26
3.5 Guidance for Process Engineers......27
3.6 Event Detection......27

Chapter 4: Framework Implementation: City of San Diego......29
4.1 Facility Description and Design Criteria......29
4.2 Software and Data Workflow......34
4.3 Ozone Configuration......37
4.3.1 Ozone Data Selection......37
4.3.2 Ozone Data Screening......43
4.3.3 Ozone Data Flagging and Event Detection......44
4.4 Microfiltration......49
4.4.1 MF Data Selection......49
4.4.2 MF Data Screening......51

4.4.3 MF Data Flagging and Event Detection ... 51
4.5 Reverse Osmosis ... 53
4.5.1 RO Data Selection ... 53
4.5.2 RO Data Screening ... 54
4.5.3 RO Events ... 55
4.6 UV/AOP Configuration ... 59
4.6.1 UV/AOP Data Selection ... 59
4.6.2 UV/AOP Data Screening ... 60
4.6.3 UV/AOP Events ... 61
4.7 Plant wide Configuration ... 62
4.8 Event Notification ... 62
4.9 Framework Implementation Summary ... 66

Chapter 5: Implementation Testing and Validation ... 69
5.1 Testing Methodology ... 69
5.2 Testing Results ... 69
5.2.1 Ozone Event Detection Challenge Testing ... 70
5.2.2 MF Event Detection Challenge Testing ... 76
5.2.3 Reverse Osmosis Event Detection Challenge Testing ... 78
5.2.4 UV/AOP Event Detection Challenge Testing ... 83
5.3 Challenge Testing Summary ... 86

Chapter 6: Project Conclusions ... 89
6.1 Chapter 1 Conclusions—Need for EDS in DPR ... 89
6.2 Chapter 2 Conclusions—Literature Review ... 90
6.3 Chapter 3 Conclusions—Event Detection Framework ... 90
6.4 Chapter 4 Conclusions—Tool Specifications ... 92
6.5 Chapter 5 Conclusions—Proof-of-Concept ... 96
6.6 Next Steps and Additional Considerations ... 98
6.6.1 Event Detection Framework and Implementation ... 98
6.6.2 Event Detection Software Considerations ... 99

Appendix A ... 101
Appendix B ... 127
Appendix C ... 133
Appendix D ... 141

References ... 163

Tables

1-1 Summary of DPR Regulations and Project by State ... 7
2-1 Summary of Findings from Utility Surveys ... 14
4-1 Unit Process Pathogen Removal Summary ... 30
4-2 NCPWDF Ozone System Design Criteria ... 31
4-3 NCPWDF MF System Design Criteria ... 32
4-4 NCPWDF RO System Design Criteria ... 33
4-5 NCPWDF UV/AOP System Design Criteria ... 34
4-6 Python Script Steps for Live and Historical Modes ... 36
4-7 Ozone CCP Monitoring Parameters ... 40
4-8 Ozone Process Calculated Parameters ... 42
4-9 Ozone Data Screening Description ... 43
4-10 Ozone Events Included in Deployed Event Detection Tool ... 44
4-11 MF Process CCP Monitoring Parameters ... 50
4-12 MF Data Screening Tags ... 51
4-13 MF Events ... 52
4-14 RO Process CCP Monitoring Parameters ... 54
4-15 RO Data Screening Description ... 55
4-16 RO Events ... 56
4-17 Operating Range Test Bounds for the RO Process ... 58
4-18 UV/AOP Process CCP Monitoring Parameters ... 59
4-19 UV/AOP Events ... 61

Figures

1-1 Difference in Retention Time versus Response Time for IPR and DPR ... 2
1-2 Key WRF Projects Advancing the Field of Monitoring and Data Analysis ... 3
1-3 Three Layers of Filters Used to Evaluate Process Performance Data in WRF 4765 ... 5
1-4 Response Time Scenarios ... 9
3-1 Factors Impacting the Time Needed to Identify and Respond to an Event ... 22
3-2 Event Detection Framework Steps ... 23
3-3 Logical Tests for CCP Monitoring Parameter Data Flagging ... 26
4-1 NCPWDF Treatment Train Diagram ... 30
4-2 Ozone Pipeline Contactor at NCPWDF ... 31
4-3 Membrane Filtration Skid at NCPWDF ... 32
4-4 RO Skid at NCPWDF ... 33
4-5 UV/AOP Reactor at NCPWDF ... 34
4-6 Software and Data Workflow for the Implemented EDS at NCPWDF ... 35
4-7 Visual Representation of Evaluating CT using the Truncated Extended Integration Method ... 39
4-8 Schematic Overview of the Ozone Process with CCP Monitoring Parameters Detailed in Red ... 42
4-9 Ozone Data Screening Code ... 43
4-10 Historic Data Example of Ozone Residual Meter Drift at OSP4 ... 46
4-11 High Demand – Ozone Water Quality Event Detection Logic ... 48
4-12 Ozone Water Quality Event Detection Code ... 48
4-13 Schematic Overview of an MF Module ... 50
4-14 MF Data Screening Code ... 51
4-15 NCPWDF RO System Online EC and TOC Monitoring Locations ... 53
4-16 UV/AOP CCP Monitoring Parameter Locations Within the UV System ... 60
4-17 Example EDS Dashboard Output ... 63
4-18 Example EDS Event Report Output ... 65
5-1 Partial Ozone Generator Failure Event Detection Report ... 71
5-2 OSP 7 Ozone Monitoring Point Failure Event Report ... 73
5-3 High Ozone Demand Water Quality Event Detection Report ... 75
5-4 Ozone Demand and Crypto LRV During the June 30th Water Quality Event Detection .. 76
5-5 MF High Filtrate Turbidity Process Failure Event Detection Report ... 77
5-6 MF Stagnant Filtrate Turbidity Monitoring Point Failure Event Detection Report ... 78
5-7 RO Potential Membrane Breach Process Failure Event Detection Report ... 79
5-8 RO Feed TOC Meter Drift Monitoring Point Failure Event Detection Report ... 81
5-9 RO Potential Chemical Peak Water Quality Event Detection Report ... 83
5-10 UV/AOP Low UV Dose Process Failure Event Detection Report ... 84
5-11 UV/AOP Stagnant UVI Sensor Monitoring Pint Failure Event Detection Report ... 85
5-12 UV/AOP Low Feed UVT Water Quality Event Detection Report ... 86

Acronyms and Abbreviations

AOP	Advanced oxidation process
AWPF	Advanced water purification facility
BAC	Biological activated carbon
BPL	Ballast power level
BW	Backwash
CCP	Critical control point
CEC	Contaminants of emerging concern
CIP	Clean in place
COP	Critical operating point
CT	Concentration-time
CUSUM	Cumulative sum control charting
DCS	Distributed control system
DDW	Division of Drinking Water
DOH	Department of Health
DPR	Direct potable reuse
DSS	Decision support system
DST	Decision support tool
EC	Electrical conductivity
EDS	Event detection system
EED	Electrical energy dose
EFM	Enhanced flux maintenance
EPA	Environmental Protection Agency
GAC	Granular activated carbon
GPM	Gallons per minute
HACCP	Hazard assessment and critical control point
HDT	Hydraulic detention time
HMI	Human machine interface
HRSD	Hampton Roads Sanitation District
IPR	Indirect potable reuse
ISN	Integrated sensor network
IT	Integrity test
LCL	Lower control limit
LRV	Log-removal value
MBR	Membrane bioreactor
MF	Microfiltration
MGD	Million gallons per day
MIT	Membrane integrity test

NCPWDF	North City Pure Water Demonstration Facility
NTU	Nephelometric turbidity units
OCWD	Orange County Water District
ORP	Oxidation reduction potential
OSP	Ozone sampling point
PLC	Programmable logic controller
PPD	Pounds per day
PPM	Parts per million
PV	Production Value
RO	Reverse osmosis
SCADA	Supervisory control and data acquisition
SP	Set point
TMP	Transmembrane pressure
TOC	Total organic carbon
UCL	Upper control limit
UF	Ultrafiltration
UV	Ultraviolet light
UV/AOP	Ultraviolet advanced oxidation process
UVI	UV intensity
UVT	UV transmittance
V/G/C	Virus/*Giardia*/*Cryptosporidium*
WRF	The Water Research Foundation
WRP	Water reclamation plant

Executive Summary

ES.1 Key Benefits

- Developed an event detection framework promoting proactive and rapid responses to direct potable reuse (DPR) process upsets and errors.
- An event detection system (EDS) based on this framework, implemented at a demonstration DPR facility, increased lead time prior to forced shutdowns and reduced response time.
- Identified 22 parameters that provide a basis for early detection of critical control point (CCP) failures.
- Classified CCP failures under one of three event types (monitoring point, process failure, or water quality), helping operators know which appropriate corrective action(s) should be taken.
- Demonstrated that statistical process control is a viable approach to configuring test bounds and thresholds specific to the site and unit process that accurately identify anomalous data at an acceptable sensitivity.

ES.2 Key Findings

- The four-step framework of data storage, data screening, data flagging, and event detection presented in Figure ES-1 can be implemented for real-time, continuous monitoring of DPR unit processes and provide early detection of possible process errors and upsets. The project team identified 22 out of 8,000 possible CCP monitoring parameters, implemented within the software via supervisory control and data acquisition (SCADA) tags. These SCADA tags enable a variety of parameters (e.g., chemical concentrations, percent removals, percent changes, etc.) to be transferred into a database, which can be read into an event-driven software. Events are categorized into three types: process failure, monitoring point, and water quality. Processes include ozone disinfection, membrane filtration, reverse osmosis (RO) filtration, and ultraviolet advanced oxidation (UV/AOP). The detection of events can be automated, provided the event happens in a predictable and systematic manner.

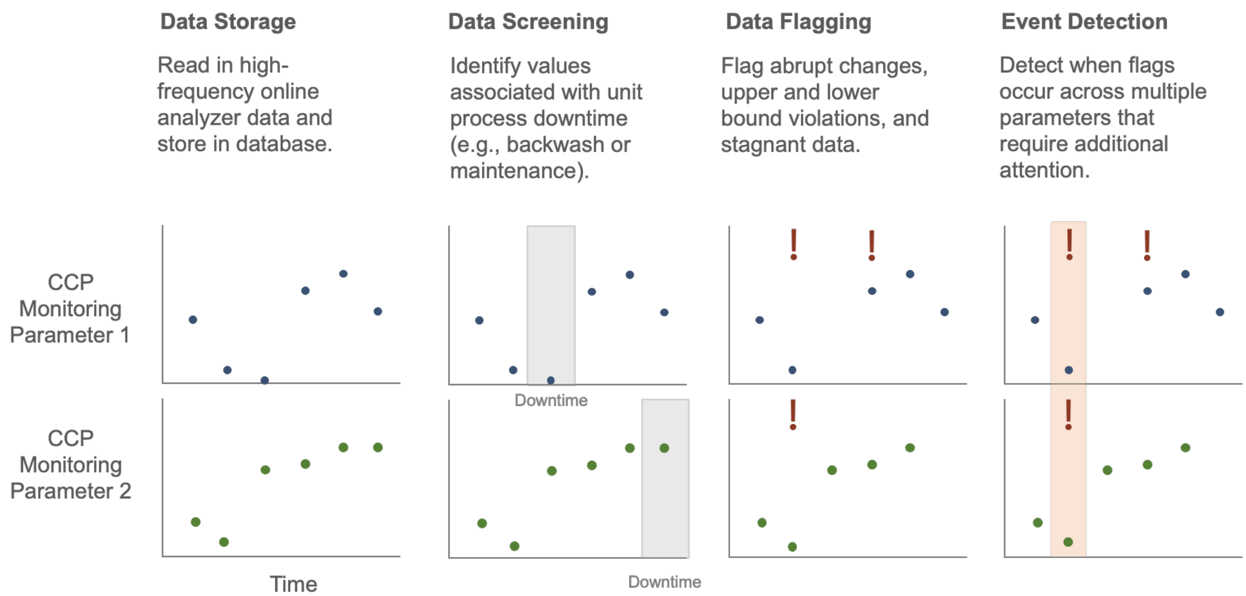


Figure ES-1. Event Detection Framework Steps.

- Event detection logic that provides longer lead times is desirable, but it may be more prone to false positive and nuisance alarms. However, if event detection logic is configured with limited sensitivity, there is a higher likelihood that the event will not be detected in a timely manner. An iterative, site-specific approach is necessary to optimize the event detection system (EDS) and achieve the appropriate level of sensitivity. This can be accomplished by challenge testing the event detection logic using historical data sets containing examples of known events and adjusting the test bounds and minimum consecutive failures based on the results.
- When used on a dataset with high-quality monitoring data, statistical modeling is effective for determining operating range bounds designed to provide early detection of possible unit process errors and upsets.
- A multidisciplinary team consisting of process engineers, operators, programmers, and systems integrators is needed to develop and deploy this type of tool and framework. Knowledge and operational experience with advanced water treatment processes are critical for formulating meaningful limits and logical tests for event detection.

ES.3 Background and Objectives

Protection of public health and a high degree of treatment reliability and performance are critical components of DPR projects. The implications of water quality excursions in the product water are magnified due to the absence of a significant environmental buffer that results in shorter retention time before reaching consumers. DPR projects must be able to demonstrate performance reliability to be protective of public health, and a key component to increasing reliability is responding to emerging performance excursions before they exceed regulatory thresholds. Reliability is centered on the ability of a potable reuse system to protect public health (Pecson et al. 2015a). This can be achieved through redundancy of treatment and monitoring to ensure that treatment objectives are reliably met or more reliably demonstrated.

This project explored a software-based approach to enhance monitoring and ultimately increase reliability of potable reuse treatment trains.

Currently, potable reuse systems are controlled by programmable logic controllers (PLCs) and computerized control systems like distributed control system (DCS) and SCADA systems. Sensors are installed throughout the systems to monitor process performance, and alarms are set so that an alarm is issued at a value close to the regulatory threshold and at the regulatory threshold. While these alarm limits are set to provide lead time so that operators are alerted before the regulatory limit is breached, operators are responsible for assessing if readings are real (i.e., not a false positive), determining what caused the alarm limit to be breached, implementing corrective actions based on troubleshooting and evaluating if the issue has been resolved. Minimizing the response time to go through these steps is important to prevent escalation of the issue and potentially greater consequences. Proactive detection and diagnosis of potential issues within the DPR treatment train were the principal areas of focus when developing the EDS (Figure ES-2).

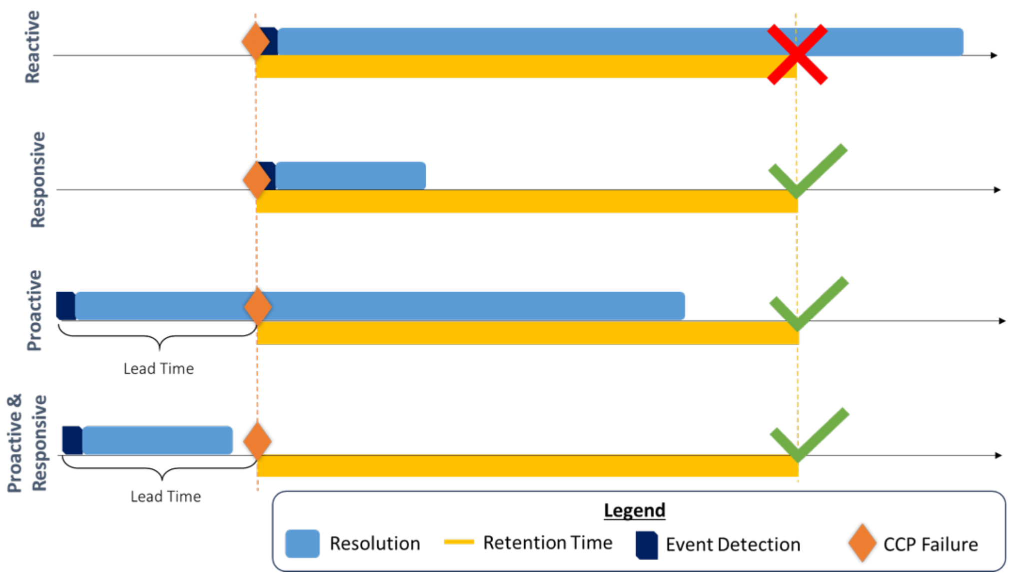


Figure ES-2. Response Time Scenarios.

ES.4 Project Approach

The project objectives and major deliverables were achieved through four primary tasks: Literature Review and Utility Surveys/Case Studies (Chapter 2), Event Detection System Framework Development (Chapter 3), Event Detection System Implementation (Chapter 4), and Challenge Testing and Event Detection System Validation (Chapter 5). The project team considered numerous approaches for developing the automated EDS and included the following: open-source, artificial intelligence (AI), machine learning, and statistical modeling. A review of prior work on these topics evaluated existing open-source data analytic programs (e.g., CANARY and Pecos), Python packages for machine learning and statistical modeling, and timeseries analysis methods for configuring structured event detection logic. Interviews were conducted with Orange County Water District (OCWD), Veolia, and Hampton Roads Sanitation

District (HRSD) to better understand how current monitoring and alarming systems are configured to reduce response time in indirect potable reuse (IPR) systems.

To develop event detection framework for DPR applications, the project team first identified the necessary software capabilities. Next, a curated list of CCP monitoring parameters was developed to focus only on data that would affect the pathogen removal capabilities of each unit process. Lastly, the operational experience and knowledge of the project team was leveraged to identify events that would impact the ability of the process to protect public health. These events consisted of one or more Pecos quality control tests configured with operating range bounds and minimum consecutive failure thresholds for the CCP monitoring parameters.

The EDS prototype was implemented at the City of San Diego's 1 million gallons per day (MGD) North City Pure Water Demonstration Facility (NCPWDF) to test the functionality of the scripts and evaluate the sensitivity of the event detection logic for providing advanced notice of potential issues. An iterative approach was employed when challenge testing the 12 implemented events with either simulated or naturally occurring process upsets. Pecos quality control test configurations within the EDS were adjusted as needed based on the challenge testing results.

ES.5 Results

The literature review and utility case studies confirmed that there is a need within the potable reuse industry for real-time data analytics that can provide advanced notice of potential process upsets and errors. Data-driven modeling (i.e., machine learning/AI) is most ideal for applications with extensive data, therefore it is not currently suitable for DPR facilities where data is limited because true CCP failures are rare. The EDS should be configured so that the results of its analysis are specific and actionable for operations staff. The project team found that this can be accomplished using statistical methods and existing open-source data analysis programs.

The four-step event detection framework developed by the project team can be used as a blueprint for other process engineers to design an EDS specific to their site. The four steps are (1) data storage, (2) data screening, (3) data flagging, and (4) event detection. Data storage involves receiving raw values from the control and monitoring system. Next, data generated while the DPR facility is not in production (i.e., product water is not being distributed) is screened out of the data set that will be analyzed. The tool then analyzed the screened data using Pecos quality control tests configured by the project team for each of the CCP monitoring parameters. The flagged data points represent potential events that warrant further evaluation. In the last step, the flagged data points are evaluated, and if the Pecos quality control test(s) correspond with the event detection logic for one of the three categories (process failure, monitoring point, water quality), then the EDS generates an alert for operations staff to review. For more complex events, multiple CCP monitoring parameters are used as event criteria and evaluated simultaneously so that the appropriate issue can be identified with greater certainty. The framework facilitates site specific configuration of event detection software while providing a well-defined workflow approach to the development of such a tool in DPR applications.

The project team successfully implemented a functional EDS prototype at the NCPWDF, and the associated files can be found in a publicly available Github repository, which can be accessed via the 4954 project page on the WRF website. Figure ES-3 displays the prototype's dashboard screen that was designed to communicate the status of the events configured for each unit process.

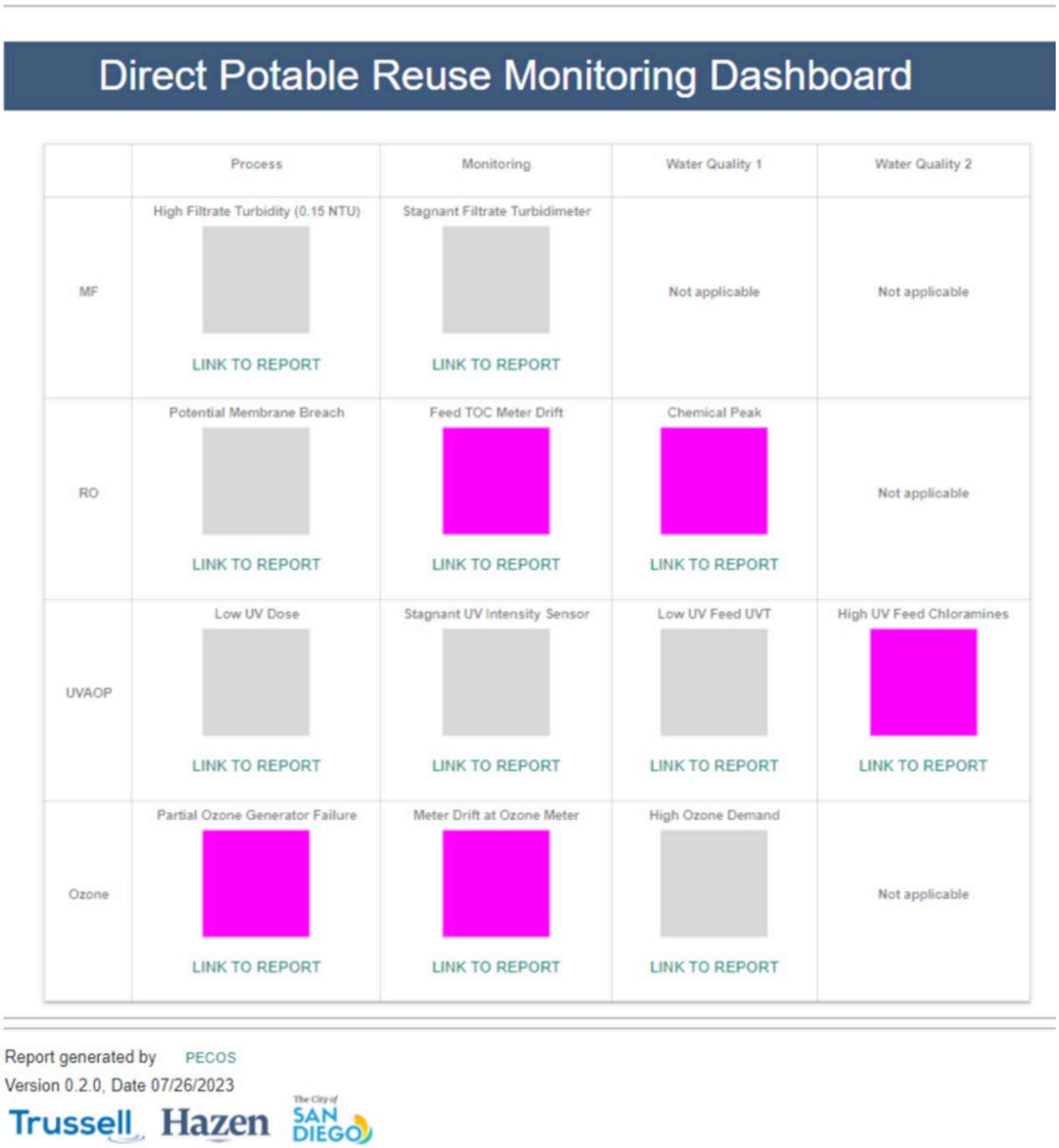
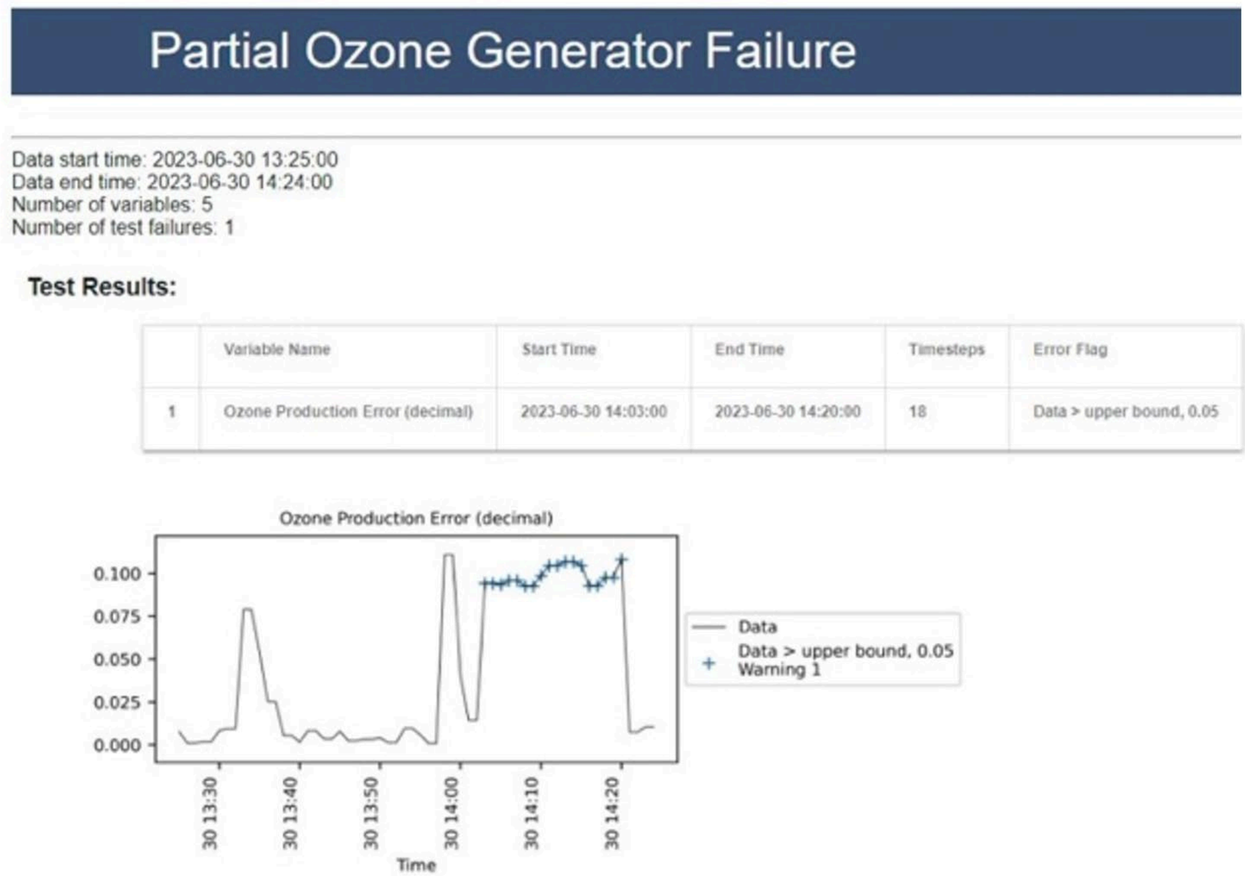


Figure ES-3. EDS Dashboard Output.

Figure ES-4 is an example report generated by the EDS during challenge testing when an ozone process failure event occurred in the timeframe that was being monitored.



Partial Ozone Generator Failure

Data start time: 2023-06-30 13:25:00
Data end time: 2023-06-30 14:24:00
Number of variables: 5
Number of test failures: 1

Test Results:

	Variable Name	Start Time	End Time	Timesteps	Error Flag
1	Ozone Production Error (decimal)	2023-06-30 14:03:00	2023-06-30 14:20:00	18	Data > upper bound, 0.05

Figure ES-4. Partial Ozone Generator Failure Event Detection Report.

Challenge testing using real and simulated process upsets validated the functionality of the EDS and enabled the project team to optimize the event detection logic so that events were accurately identified while nuisance/false alarms were avoided.

ES.6 Related WRF Research

- Monitoring for Reliability and Process Control of Potable Reuse Applications (1688)
- Critical Control Point Assessment to Quantify Robustness and Reliability of Multiple Treatment Barriers of DPR Scheme (1700)
- Integrated Management of Sensor Data for Real Time Decision Making (4759)
- San Diego DPR (4765)

CHAPTER 1

Introduction

1.1 Introduction

The main differentiator between direct potable reuse (DPR) and indirect potable reuse (IPR) is the loss of the environmental buffer. This loss is important because the environment provides many benefits to protect public health. As a result, DPR systems will need to adapt to compensate for these losses. The primary goal of potable reuse projects is to provide reliable protection of public health. Towards that end, the basic approach for IPR has been to both **prevent** issues from occurring and **respond** to those that do. To date, significant research has focused on how to replace these lost benefits in DPR with additional prevention and response. For example, additional treatment [e.g., ozone and biological activated carbon (BAC)] is being used to supplement reverse osmosis (RO)-based treatment in California's draft DPR regulations to compensate for the loss of treatment in the environment. Additional treatment protects public health by further preventing off-spec water from being sent to consumers.

One benefit that cannot be so easily replaced, however, is time. Unlike treatment, there is no easy way to replace the retention time provided by the environment. In all cases, a system's ability to respond to a treatment issue (response time) must be faster than the time the water is retained in the system (retention time) so that systems can identify and respond to failures before the water reaches consumers. California regulations draw a bright line to define IPR: in both groundwater recharge and surface water augmentation, projects must provide at least two months of retention time in the environment. Environmental retention affords IPR projects significant opportunities to diagnose and respond to issues before water is distributed to consumers. The long retention times provided by aquifers and reservoirs offers significant response time to (1) detect treatment or water quality issues and (2) enact a response before off-specification water is consumed. Typically, systems have alarm limits in place to ensure diversion or unit process shutdown if regulatory limits are crossed. Operator response times to proactively prevent breaches of regulatory limits can vary for different types of events but is generally a function of how frequently processes are monitored, how quickly data is analyzed and evaluated to make decisions, how quickly a response is made, and how effective the response was to address the root cause of the issue. The accuracy and quality of the data generated by monitors can also affect how quickly issues can be mitigated.

DPR projects may reduce that retention time from months to as little as hours, representing a thousand-fold or more reduction in the time available to identify and respond to events (Figure 1-1). If retention time cannot be extended, then response time must be shortened. The entire process of integrating and responding to performance data must shift to real-time and proactive. To adapt, DPR projects will need to develop monitoring and control systems that allow them to rapidly process performance data, identify potential issues, and enact corrective responses before water is sent to consumers. Given the short timescales available in DPR, being able to integrate performance data and provide real-time automation of advanced treatment facilities is one of the greatest technical challenges for the implementation of DPR.

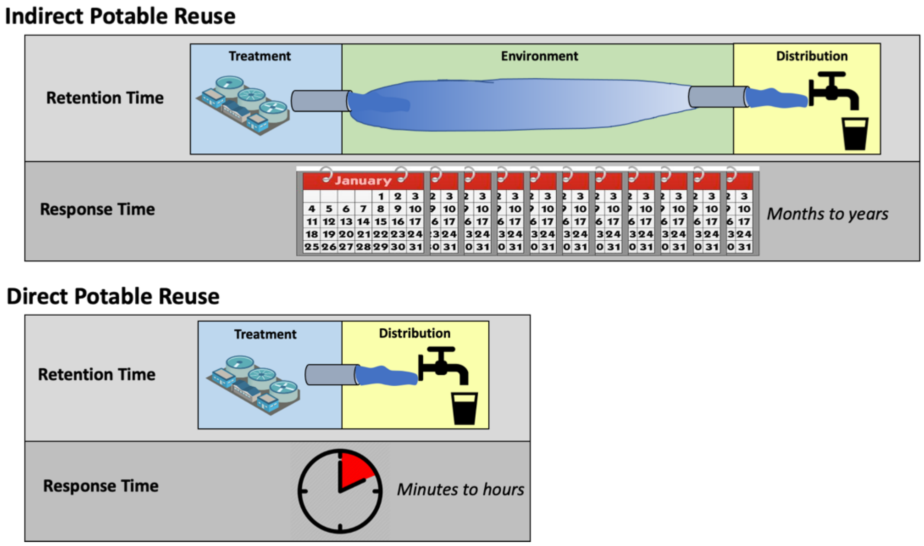


Figure 1-1. Difference in Retention Time versus Response Time for IPR and DPR.

To assess this challenge, existing monitoring and control systems must first be examined to determine the extent to which they are capable of dealing with this significant loss of response time. If they are not currently capable, what other options exist for public health protection in DPR? One option would be to place more emphasis on preventing failures by providing greater redundancy in treatment and monitoring. Nevertheless, there will always be some risk of off-spec water passing through. In light of this, **a response system is critical for public health protection in DPR**. While there is consensus that this response system is needed, is it possible? The goal of WRF project 4954 is to develop a framework for DPR monitoring and control that can rapidly identify and respond to events within the constrained timeframes of DPR.

1.2 Summary of Monitoring Advancements in Water Research Foundation Projects

As the industry develops updated monitoring and control systems for DPR, it can leverage the advancements already made in this field in IPR settings. The need to minimize response time in DPR projects has been recognized as an important topic and several studies funded by the Water Research Foundation (WRF) have been conducted to advance the industry's use of monitoring data. A summary of projects making important advancements in public health protection through treatment, monitoring, operations, and risk assessment is provided in Figure 1-2.

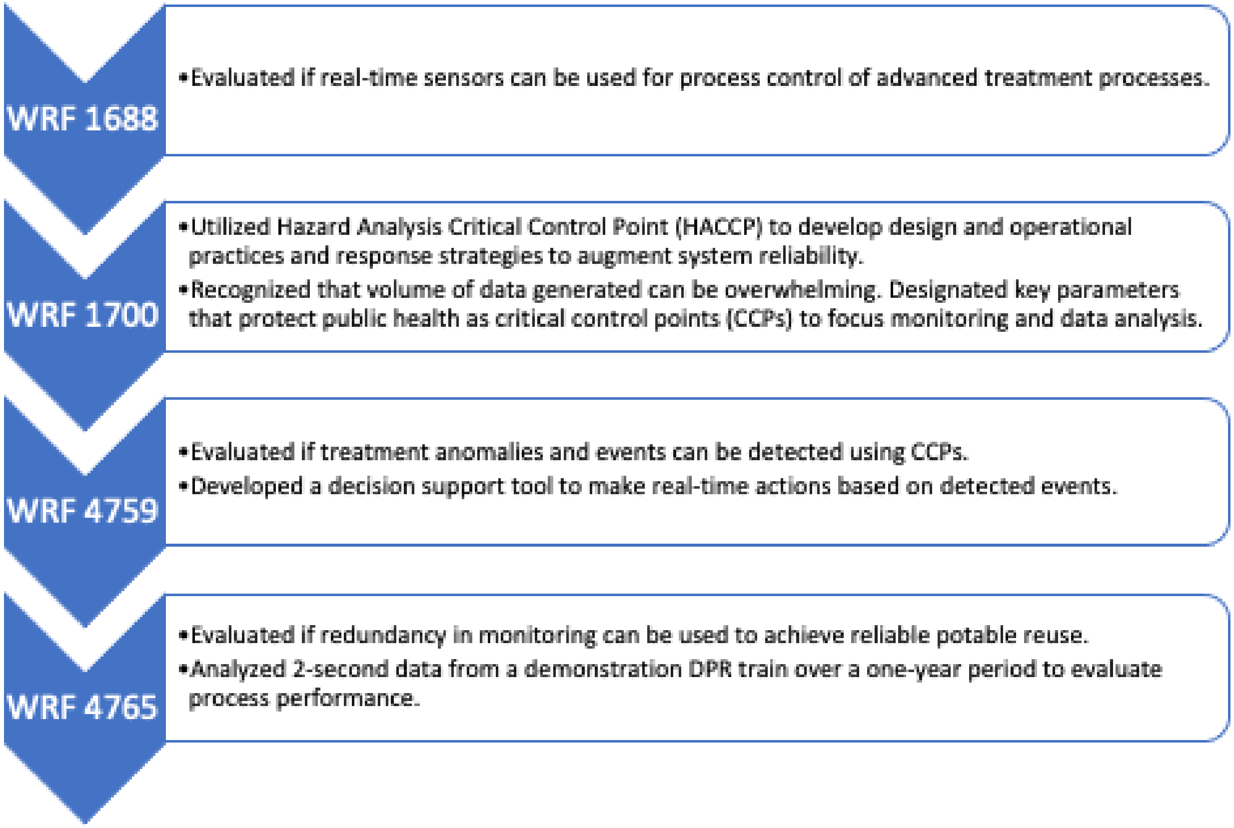


Figure 1-2. Key WRF Projects Advancing the Field of Monitoring and Data Analysis.

1.2.1 WRF 1688

WRF 1688 addressed both IPR and DPR with a focus on the measurement of microbial contaminants (Snyder and Pepper 2016). This project reviewed online instruments in potable reuse—including accuracy, response time, and detection mechanism—and described best practices for these instruments. The study found that online instruments commonly measure chemical contaminants or surrogates but not microbial contaminants. Turbidity, conductivity, and total organic carbon (TOC) are commonly used to detect treatment failure. In addition, online fluorescence sensors and rapid tests for microbial contamination could supplement existing indicators, though the latter were deemed to currently be at an early stage of development.

1.2.2 WRF 1700

WRF 1700 described the process of integrating hazard analysis and critical control point (HACCP) methodologies into DPR including hazard assessment, water quality objectives, identification of CCPs and COPs, CCP/COP monitoring parameters, and CCP/COP response procedures (Walker et al. 2016). This work also developed approaches to understand and communicate the risk associated with a compromised CCP barrier. Some of the key principles of HACCP, as formalized in ISO 22000, evaluate CCPs and evaluate the following questions:

- Is there a hazard at this process step? What are the hazards?
- Do control measures exist for the identified hazard?

- Is the step specifically designed to eliminate or reduce the likely occurrence of the hazard to an acceptable level?
- Could contamination occur at or increase to unacceptable levels?
- Will a subsequent step or action eliminate or reduce the hazard to an acceptable level?

The study also provided commentary on drinking water regulations and how they might set a precedent for reuse regulations. In some countries, surface water treatment plants implement HACCP by monitoring surrogate parameters of pathogens and implement a multi-barrier approach. In the U.S., existing Environmental Protection Agency (EPA) regulations for drinking water treatment use a combination of both CCP monitoring and end-point monitoring. Many advanced treatment processes currently used in potable reuse use online monitoring to verify the performance of the system in real-time such as turbidity monitoring on membrane filters or the use of the CT framework for chemical disinfectants.

1.2.3 WRF 4759

WRF 4759 developed a Decision Support System (DSS) and a Microsoft Excel-based Decision Support Tool (DST) using online sensors that are applicable to potable reuse systems (Neemann et al. 2019). The study evaluated EDS software packages for anomaly detection and discussed the idea of developing an Integrated Sensor Network (ISN) of both water quality and operations and maintenance data to detect failures in treatment. The study also evaluated use of commercially available monitoring sensors to identify failures for a pilot treatment train consisting of ozone and BAC.

1.2.4 WRF 4765

WRF 4765 combined the findings of several DPR research projects to assess how redundancy in treatment and monitoring could be used to promote the reliability of public health protection. One key research question was whether DPR systems could be developed that would provide continuous protection of public health. This uncertainty stemmed largely from the lack of full-scale performance data from actual DPR systems. To help address this data gap, this project evaluated the City of San Diego’s 1 MGD North City Pure Water Demonstration Facility (NCPWDF) to assess the benefits of redundancy and monitoring to achieve reliable potable reuse (Trussell et al. 2017). Yearlong continuous monitoring of the treatment train, consisting of ozone, BAC, membrane filtration, RO, and UV/AOP, provided an extensive dataset to assess process performance. Routine performance monitoring was complemented with multiple challenge tests that assessed the benefits of the enhanced treatment train (Tackaert et al. 2019). The performance data were used in a quantitative microbial risk assessment to demonstrate that a full-scale DPR treatment train could reliably meet performance goals and produce a water that provides public health protection equivalent to, or greater than, conventional drinking water supplies (Pecson et al. 2017). The study also discusses the importance and relevance of timely operating data when operating DPR. This involved filtering and querying of sensor data to verify that the processes and sensors were functioning correctly (Chen et al. 2020; Pecson et al. 2018).

One key takeaway from WRRF 4765 was that processing data from DPR systems—which may produce more than 300 GB of data annually—cannot be done without complex and automated data filtering (Pecson et al., 2017). This requires understanding (a) which CCP monitors are needed to assess the performance of the different unit processes, (b) how to set up filters to focus on unit process performance when the system is producing water (vs. when it is offline or in startup mode), and (c) how to compile and integrate these data to assess systemwide performance. The data processing required removing certain data points from the analysis, such as when the system was offline or shut down. As expected, there was significant deviation in sensor signals when the system was offline or when it was transitioning from an off-line state to steady-state production. If these signals were included in the dataset for analysis, it would appear as if there were many failure events. However, these false positives should be removed from the dataset in order to focus exclusively on the data generated when the system was actually producing and distributing water. Extensive data cleaning/filtering was implemented using an R-script to distinguish the data that was produced when the system was online and operational. The general framework used in WRF 4765 for filtering data is shown in Figure 1-3.

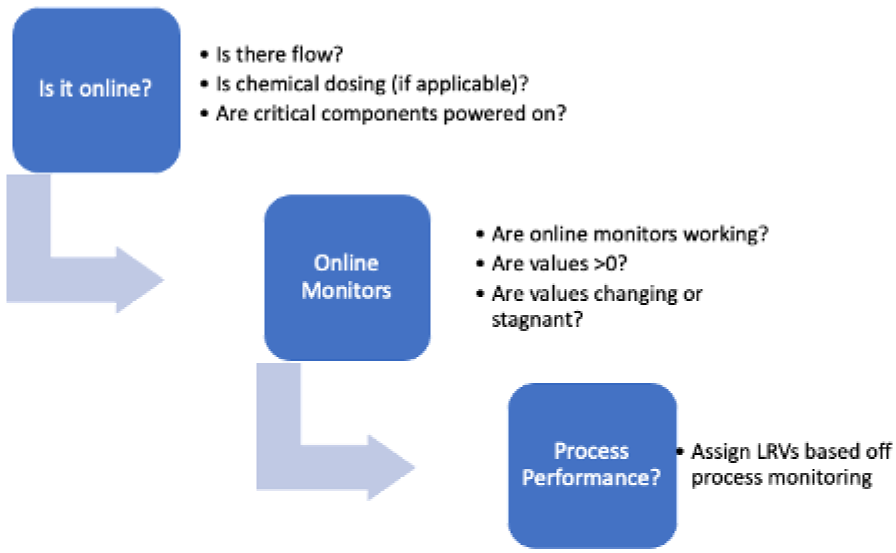


Figure 1-3. Three Layers of Filters Used to Evaluate Process Performance Data in WRF 4765.

The first layer of filtering was used to determine whether the process was actually online. This filter layer helped to remove most of the off-line data by identifying whether basic process functions were being conducted, e.g., that water was flowing through the system, that chemicals were being dosed, and that all critical components were online. Data that passed the first layer went on to the second filter, which provided a rudimentary check to determine whether the meters were online and functional. These types of meter error checks included assessing whether a non-zero reading was being recorded and whether the values were changing. Assuming the first two layers passed, the third layer assigned a log removal value (LRV) based off the process monitoring data. The data filters were capable of removing many data points that were generated during periods when the processes were not in production mode.

Despite the effectiveness of the automated filters, additional manual processing was required to curate the dataset completely. While the filters were helpful for evaluating if processes were offline, there was still a significant number of suspected water quality events. When operators cross-referenced periods of potential failure events with operational event logs and data trends, they found that the apparent "failures" could be attributed to other events occurring at the facility that were not indicative of process performance. These included both upstream and downstream events, such as maintenance events on other unit processes that ceased flow, power outages to the whole facility, or shutdowns to restart other unit processes. Based on the automated filters alone, these could appear to be failure conditions with potential public health implications. Manual evaluation of each event was therefore required to separate the false positives from the data that were truly indicative of process performance. This highlighted the importance of context in assessing the true performance of the system—knowledge of the performance of other unit processes may be necessary to interpret the data for a given process. Of the types of issues that arose that were specific to the process itself, the majority of suspect events could be correlated with the following scenarios:

- Meter error/drift resulting in readings that were not representative of process performance
- Erratic or elevated meter readings during operational changes (particularly flow changes)

Given the lack of automated filters to review the data, manual categorization was performed for all suspected water quality events. Because the two scenarios identified above occurred routinely, a systematic routine for categorizing and confirming these scenarios was developed. With the aid of this routine, the time and effort to manually categorize these events was reduced; though it still remained tedious and time-consuming. Certain parameters were affected by these types of scenarios more than others, which generated a large number of trends that needed manual review. For example, the membrane filtration systems underwent a backwash (BW) every 30 minutes and the filtrate turbidity always spiked after a BW due to sample flow changes to the filtrate turbidimeters. At this frequency of flow changes, many "suspect water quality events" were generated which had to undergo manual review for categorization. The testing site for WRF 4765 only had two membrane filtration trains making the manual evaluation of the data relatively feasible; such an approach would be infeasible with the number of trains typically present at a full-scale facility. This effort highlighted that the data filtering developed for WRF 4765 needed improvement and further automation to better distinguish the true water quality events and reduce the need for manual filtering.

WRF 4765 demonstrated that creating effective filters for CCP performance requires significant effort. Further refinement of the filters developed in WRF 4765 would be needed to reduce the number of false positives and eliminate (or greatly reduce) the need for the manual categorization of suspected events. The filters should be able to identify a number of scenarios both within a given unit process (e.g., meter drift and operational changes) as well as contextual information (e.g., the impact of failures of upstream and downstream systems). Refinement of filters can be difficult because it is important that the developed filters do not mistakenly categorize true water quality events as false positives. Development of new filters is an iterative approach and requires manual review to confirm if categorizations made by filters are accurate.

Based on this study, the team made several conclusions about the challenges surrounding monitoring and control systems for DPR: 1) DPR systems will produce large quantities of performance data, 2) it will not be possible to analyze the data and evaluate system performance manually, and 3) automated systems will be needed for DPR implementation. At the time, the project team concluded that the functionality needed for DPR monitoring and control did not exist but would be a key technical requirement for future implementation. In identifying these needs, WRF 4765 was an important springboard for the current project.

1.3 Importance of Pathogen Control in DPR

Reliable control of pathogens is the most important goal for potable reuse systems given that even brief periods of inadequate treatment can lead to infection and illness within the community. For this reason, the main focus of this project is on the unit processes—or critical control points (CCPs)—related to the control of pathogens. DPR systems must ensure that CCPs are either properly functioning (on-spec) or that any off-spec water is diverted prior to distribution. The treatment required or recommended for DPR varies across states. Some states are prescriptive about treatment while others are not, but all require reliable control of pathogens. As seen in Table 1-1, treatment trains are always composed of multiple unit processes. This multiple-barrier approach has been a fundamental aspect of public health protection for decades and provides a diversity of barriers to control against the full spectrum of pathogens (including virus, protozoa, bacteria).

In light of this, DPR projects will need systems to evaluate the performance of multiple unit processes in real-time and provide reliable detection of events or potential events. Typically, potable reuse trains use many similar unit processes for pathogen control including physical barriers (microfiltration, ultrafiltration, reverse osmosis), chemical disinfectants (ozone, chlorine), and UV disinfection. As a result, DPR monitoring and control systems should be largely adaptable across locations even if the treatment requirements differ from one location to the next.

Table 1-1. Summary of DPR Regulations and Project by State.

State	DPR Regulations Exist?*	Required Treatment Train	DPR Projects	Project Name/Location
Arizona	Draft in progress (expected Dec 2023)	Draft in progress	No	Scottsdale (has a permit but no project)
California	Draft	UV light disinfection	Yes (projects in plan)	LA: Headworks DPR Demonstration Project
		Ozone		
		BAC - follows Ozone unless exempt		
		RO		
		AOP		
Colorado	Yes	None	Yes (demonstration facilities)	Colorado Pure Water Mobile Demonstration Project
Florida	Draft	RO and high-level disinfection - UV (can propose alternative)	Yes, (pilots and demonstration designs)	Polk County DPR Pilot
				Design only: Dania Beach DPR Project

State	DPR Regulations Exist?*	Required Treatment Train	DPR Projects	Project Name/Location
				Daytona Beach Demonstration Testing System
New Mexico	Draft in progress	Draft in progress	Yes	Cloudcroft
Oregon	Yes, case by case basis	RO and/or other advanced treatment systems	No	n/a
Texas	Yes, case by case basis	RO and UV with Advanced Oxidation**	Yes	Big Spring
				Wichita Falls
Washington	Yes, case by case basis	None	None	n/a

*As of March 2023

** Or a treatment unit that removes a similar wide range of chemical contaminants

1.4 Response Time and Event Detection Systems

Response time depends on event detection capability and the time it takes to resolve the issue (resolution time). The event must first be detected before corrective actions can be implemented to resolve the issue. Response to events can be reactive, responsive, proactive, or both proactive and responsive as shown in Figure 1-4. A reactive response is detection of the event when a CCP in the treatment process meant to protect public health has failed. In the worst-case reactive scenario, the resolution time exceeds the retention time and product water with compromised water quality is distributed to the public. In the responsive scenario, the event is still detected at the time of CCP failure but since the response time is shorter than the retention time, the treatment system remains protective of public health. Ideally, the response falls under the proactive or proactive and responsive scenarios. In both scenarios, the event is detected *prior to* a CCP failure. The time between event detection and CCP failure can be characterized as the “lead time” and ideally the issue is resolved prior to an actual CCP failure. Even if the issue is not resolved within the lead time, the early detection grants additional time to resolve the issue so that resolution time does not exceed the retention time. The goal for future event detection systems will therefore be to 1) increase the lead time and 2) decrease the resolution time.

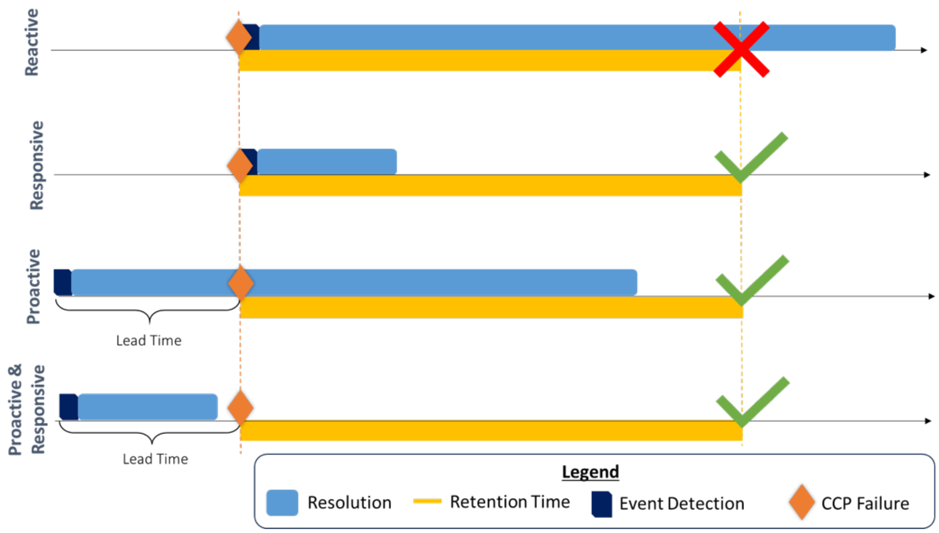


Figure 1-4. Response Time Scenarios.

In defining failure events, multiple types of events may occur. Based on the project team's operational experience and events that occurred during WRF 4765, the events were distilled into three event categories: 1) treatment, 2) water quality, and 3) monitoring. The first type of failure is when a treatment process itself is not functioning properly, so the expected pathogen removal is either partially or completely impacted. Next, the upstream water quality may degrade beyond what the treatment process is designed to treat. In that case, despite a properly functioning treatment barrier, the finished water may still be of unacceptable quality. The third failure mode relates to a monitoring failure. In the first two failure modes, it is assumed that the data collected by the online analyzers reflects the true state of the system. However, several conditions may impact analyzer accuracy, e.g., online analyzers may drift, require calibration, or may fail to collect data entirely. In the case of a monitoring failure, it is not clear whether or not the CCP is working properly because the data does not reflect reality.

Therefore, the project team defined three categories of events in this work: 1) treatment process, 2) water quality, and 3) monitoring point failures. A process failure is a measure of whether treatment equipment is functioning properly. An ozone generator failure, a membrane breach in membrane filtration, and a RO seal failure are all examples of potential process failures. A water quality failure is a measure of whether there are upstream water quality changes that require attention. Degraded water quality might be due to a disruption in upstream wastewater treatment processes resulting in changes to the ozone demand or the introduction of a chemical peak. A monitoring point failure identifies online analyzers reporting potentially poor-quality data. For instance, a stagnant reading on a sensor or values drifting beyond typical ranges.

While advanced water treatment technologies are generally reliable (Pecson et al. 2017), these types of events can still occur based on the project team's operational experience and findings from WRF 4765. An example of a monitoring failure that occurred at an actual reuse facility was when a meter value was left in hold while performing meter maintenance and not returned to monitoring mode. In this example, the control system will see a constant value. This prevents the control system from being able to adjust operations and issue alarms since monitoring is not reflective of actual process performance. In addition, since the held value is not representative of process performance, reporting during this period is also inaccurate since the held value is recorded in historian, not the raw meter readings.

Water quality excursions may also occur due to external factors outside of operator control, like chemical peaks. There are certain chemicals that are difficult to remove even with advanced water treatment technologies like formaldehyde and acetone. Illicit discharges to the sewershed of these types of chemicals have occurred at IPR projects (Debroux 2021). In these types of scenarios, if the volume of chemical discharge is large enough, the advanced water treatment train will not be able to remove these chemicals to acceptable levels and the only option is to divert the product water to protect public health. Diversion would likely be automated to occur based on a high TOC level setpoint for these types of events. To diagnose the cause of the event and resolve the event after diversion has occurred, the operator must determine if a chemical peak truly occurred or if elevated TOC was due to TOC meter issue and RO membrane integrity breach. After identifying the cause, the operator can respond accordingly to address the root issue and resolve the matter before resuming production of purified water.

Process failures can also occur if a critical component has failed. For example, UV lamps are expected to have a useful life of approximately one year before they fail, though failure can occur before one year has elapsed. If enough UV lamps fail in a reactor, the desired treatment objectives might not be met and would be considered a process failure. Operators rely on the control system to inform them that an UV lamp has failed since there is no other way of checking while UV system is in operation. UV lamps must be replaced before the critical number of failed lamps is reached. From these examples, it is clear that monitoring and response is dependent on the monitoring system and operator response.

If an event detection system for DPR can differentiate and identify the different types of failure—treatment, water quality, or monitoring—it can more rapidly point operators to the cause of the issue. Knowing both **that there is an issue,** and **the cause of the issue** can help to reduce the resolution time.

1.5 Goals and Summary

The goal of this project was to develop a software-based event detection system to integrate high-frequency data from DPR facilities and to further inform reliable operations. The event detection system is intended to complement industrial control systems logic, alerts, and alarms implemented by systems integrators.

The event detection system seeks to address the short response times available in DPR by providing advanced warning of potential issues and data-based insight into the cause of the issues. By giving operators additional lead time and more rapid resolution times, the event detection system will help ensure that response times are faster than the short retention times of DPR. The framework developed in this project was applied to ozone, membrane filtration, reverse osmosis (RO), and ultraviolet advanced oxidation process (UV/AOP). While the configuration of treatment trains may differ between potable reuse projects, the framework can be applied to any process in any potable reuse train. This generalizable framework is intended to inform reuse stakeholders about the needs of DPR, including utilities, technology vendors, supervisory control and data acquisition (SCADA) integrators, and consulting firms

.

CHAPTER 2

Utility Interviews and Literature Review

The project team evaluated the monitoring and alarming systems currently in use and software tools available for event detection. Interviews were conducted with utility staff working in IPR Projects to investigate current industry practices. The results of these interviews informed a literature review of existing software tools and best practices for proactive monitoring and event detection that could be applied to DPR.

The literature review draws on the following disciplines: 1) event detection for engineering applications, 2) data science, and 3) manufacturing and industrial process control. To increase the lasting usefulness of this research, the project team limited the scope of the review to open-source solutions.

2.1 Utility Interviews

To better understand how current monitoring and alarming systems and advancements in monitoring and alarm systems are configured to reduce response time, utility interviews were conducted for staff currently working with IPR systems. Interviews were conducted with the following groups: OCWD, Veolia, and Hampton Roads Sanitation District (HRSD).

An overview of the findings from the interviews is provided in Table 2-1. All facilities utilize a control framework that involves the monitoring of CCP performance with online meters. Control limits are defined for all CCPs, and most limits are fixed, static values that may be periodically updated. None of the facilities dynamically set control limits. To minimize the impact of any single monitor on plant performance, all facilities have redundancy for one or more of the critical instruments. While the systems collect a high amount of performance data, much of the processing of the data is done by the operators rather than through automated processes. For example, operators are responsible for:

- Identifying abnormal meter behavior and initiating follow up investigations
- Spotting false positive readings from monitors
- Creating strategies (e.g., alarm delays) to minimize the impact of erroneous readings
- Developing compliance reports (though some steps in the process may be automated)

Table 2-1. Summary of Findings from Utility Surveys.

Survey Topic	OCWD	Veolia	HRSD
Reuse system description	Train: MF–RO–UV–UV/H_2O_2–stabilization Capacity: 100 MGD. 70 MGD to groundwater recharge, 30 MGD to seawater intrusion barrier	Train: MF–RO–UV–UV/H2O2–stabilization Capacity: 3 Advanced water purification facilities with 61 MGD capacity (48 MGD available). Only one facility running at the time of the interview	Train: Floc/Sed-O3-BAC-GAC-UV-Cl2. Capacity: 1 MGD research center. Up to 100 MGD with implementation of full-scale facilities.
Key control parameters	• MF: turbidity, membrane integrity test (MIT) • RO: TOC, electrical conductivity (EC) • UV/AOP: UV transmittance (UVT), electrical energy dose (EED)	• Water standards. Currently only distributing for industrial users, yet meets drinking	• Ozone: concentration-time (CT) • Granular activated carbon (GAC): turbidity • UV: dose • Cl_2: CT
Operating Criteria	• MF: filtrate turbidity < 0.15 nephelometric turbidity unit (NTU) • MF: MIT > 0.2 psi/minute triggers work order • RO: Permeate TOC < 0.1 mg/L • UV: Feed UVT > 95% • UV: Dose > 101 mJ/cm2 • TOC: limit based on control chart statistical analysis of historical operational data	• Facilities use COPs and CCPs. These are related to ISO 22000 in Australia for Drinking Water. • Several alarms for CCPs based off of rate of change (e.g., ammonia). These are periodically reviewed and manually adjusted based on statistics.	• Ozone: virus LRV > 3.5 • GAC: < 0.15 NTU • UV: dose > 186 mJ/cm2 • Influent: TOC < 15 mg/L • Influent: EC < 2,000 µS/cm • Influent: turbidity < 5 NTU • Influent: total nitrogen < 5 mg/L • Effluent: TOC < 4 mg/L
Redundancy in critical analyzers	Redundant TOC analyzers for RO permeate	Redundant oxidation reduction potential (ORP) probes ahead of RO	Redundant TOC analyzers for GAC effluent
Frequency of validation/ calibration of instruments	Regular calibration/ validation done in-house for most meters/probes. Service contract for TOC analyzers.	Daily/weekly/monthly calibration/verification of all online instrumentation. Performed by in-house staff and external consultants/ vendors for more sophisticated analyzers.	Weekly/monthly (as-needed) calibration/verification of all online instrumentation. Performed by in-house staff.

Survey Topic	OCWD	Veolia	HRSD
Procedure for documenting instrument service	Tracked in Maximo	Tracked in Maximo	
Operational response to abnormal instrument behavior	Operators trained to spot anomalies and troubleshoot as part of SOP for responding to alarms. If sustained, can trigger sampling events.	Operators make decisions based on trends. Can trigger sampling events. SCADA does list responses/action items to fix issues based on alarms/warnings.	Operator to investigate. Confirmation with bench top readings. Instrumentation staff are available 24/7. Must confirm issue is fixed.
Compliance reporting procedures and purpose	• Monthly reports to DDW • Max and average values for each day in reporting period • Some automation from excel macro, but manual analysis required	• Reporting is primarily made to SNP Water, which is made available to Department of Health (DOH) • Exceedances are identified to the DOH on an annual basis and analytes are also made available to DOH • Internal reports are developed on a daily basis (process reports) for operational purposes and rolled up to monthly reporting • Some automation, but require manual to finalize	• Quarterly regulatory reporting for the EPA's Underground Injection Control Program • Identifies operations and reasons for process being offline when applicable • Describes CCPs compliance for reporting period
Predictive analytics	Not implemented. There is a cybersecurity concern for cloud-based systems. Such analytics would need to be done locally.	Not implemented.	Not implemented.
Dynamic or static alarm levels	Mostly static. Some parameters, like TOC, are statistically informed using prior historical data	Dynamic: rate of change metrics that are specifically re-evaluated with some frequency	Mostly static. Correlations developed based on site-specific influent water quality and influence on product water quality (e.g., influent conductivity influent on bromate formation).
Frequent false positives or nuisance alarms	Operators are trained to spot false positives.	Operators are trained to spot false positives. Daily process info is	Alarms triggered after two consecutive readings to diminish

Survey Topic	OCWD	Veolia	HRSD
		reviewed by plant operators and management.	false positives. Lessons have been learned on how to diminish false positives.

Responses from the utilities surveyed revealed certain aspects of existing monitoring/reporting systems that may need additional advancement for DPR applications:

- False positive water quality events are common. The following strategies are utilized to identify and minimize their impact:
  1. Train operators to spot false positives
  2. Use alarm delays to minimize false positives
- All utilities surveyed responded that generating compliance reports is mostly automated, but manual analysis is needed to finish reports.
- Static versus dynamic alarms: two utilities surveyed use static limits exclusively for generating alarms. The third also employ some dynamic, rate-of-change metrics to trigger alarms.

2.2 Event Detection Systems for Engineering Applications

In the wake of the September 11th terrorist attacks, the US federal government focused on identifying potential vulnerabilities in the nation's infrastructure, including drinking water. In an attempt to identify water contamination events in the distribution system, Sandia National Labs and the EPA jointly developed an open-source EDS called CANARY (U.S. EPA 2012). After the release of CANARY, Sandia National Labs built another EDS called Pecos designed to monitor solar photovoltaic systems (Klise and Stein 2016). Although Pecos was not designed for the water sector, it is an EDS tool that shares many features in common with CANARY. Among open-source products for EDS within engineering, CANARY and Pecos fit many of the requirements for an EDS for DPR. They both support real-time data analysis, are well documented, tested, and maintained, and are published in peer-reviewed literature. Both CANARY and Pecos were designed to analyze high volumes of sensor data, identify anomalies in the data, create alerts for operators, and improve data quality.

CANARY (U.S. EPA 2012) uses three water quality event detection algorithms described in the literature: timeseries increment, linear filter, and multivariate nearest neighbor (Klise and McKenna 2006a; b). These algorithms are combined with what is known as a binomial event discriminator (McKenna et al. 2007) to identify suspicious data points. Moreover, the user can choose to run the event detection analysis on water quality sensors individually or as a group. The probability of a true event is evaluated by CANARY by evaluating the series of suspicious data points. If the probability of a true event exceeds a user-defined threshold, CANARY issues an alarm that an event is occurring. To provide additional flexibility, CANARY enables users to develop their own event detection algorithms using MATLAB (U.S. EPA 2012) and more recently, Java (Hall et al. 2017).

There are some known applications of CANARY in industry. For example, CANARY has been integrated into distribution system models by hydraulic modeling vendors for water quality event detection (Hall et al. 2017). One study was published describing the use of CANARY to identify events during normal operations of a decentralized membrane bioreactor (MBR) system and during simulated failure events (Leow et al. 2017). Failure was simulated by performing sludge bypass events to simulate membrane integrity failure by pumping mixed liquor into the effluent lines of the MBR system. Simulated events were detected by CANARY and were correctly detected as process failures. Alarms detecting process failure were also generated during normal operating conditions and it was found that 89% of the alarms were false positives. Retroactive review (i.e., not real-time and performed manually by a human) of the data by the researchers found that the false positives could be attributed to normal MBR operations (backflushes, membrane cleaning, etc.), sensor maintenance and calibration, and change in feed water quality. While sensor signals may have deviated from the normal baseline, all of these events occur routinely, and deviated readings do not represent process failure or unacceptable water quality. In addition, there were also 23 alarms that were generated during the normal operation period in which the causes are unknown. Of these 23 alarms, 13 of the events had trends that were similar with trends of known events, suggesting detection of true events. More recently, CANARY has also been used to identify spill events related to natural gas production (Wickline and Hopkinson 2020). This study however found that the EDS capabilities of CANARY were not suitable for detecting the simulated spills due to size of the spill relative to the watershed size, sensor location, and type of contaminant. This suggests that outliers must exceed a minimum threshold for event detection and may not be sensitive to capture all events.

Originally developed for the operation of photovoltaic cells, Pecos is industry-agnostic and has been applied to water science and engineering applications, including marine hydrokinetics (Klise and Stein 2016). The primary purpose of Pecos is to analyze the quality of real-time data streams and generate visualizations and reports to communicate that information. Data can be flagged by Pecos using a suite of “quality control tests” to identify missing, duplicate, or corrupt data, data outside a user-defined range, or abrupt change. The tests are designed to be simple yet highly flexible. For example, Pecos provides EDS developers with the ability to specify the minimum number of failures before flagging the data. If a single data point is outside a user-defined range, the developers may not want that to be reported as a failure. A single value may be the result of brief issue in sensor data quality that does not require attention. Instead, the developers may determine that five consecutive out-of-range values is preferred to avoid false positives.

To further minimize false positives, Pecos provides “time filters” to ignore data associated with system downtime. For instance, for photovoltaic data, Klise and Stein (2016) apply a time filter that is applied whenever the sun elevation is less than 10 degrees. This time filter pauses quality control tests when the sun sets and implements the tests when the sun rises. For water treatment applications, similar time filters can be applied during maintenance activities, such as a membrane backwash.

Pecos and CANARY have similar functionalities, and either could be applied to high-frequency data analysis in water reuse. However, there are key differences between these two approaches. CANARY is an application with a user interface and was developed using Java and

MATLAB. MATLAB is a proprietary programming language and requires a license to use. Pecos, on the other hand, is a Python software package which provides programmers with an EDS toolkit. Python is now the de facto programming language in many industries for data analysis, including environmental engineering. Moreover, its popularity has consistently trended upward in recent years (The Economist 2018). Pecos tools can be used alongside other Python-based tools, such as those used for machine learning, AI, and statistical modeling.

2.3 Machine Learning, AI, and Statistical Modeling

There is a wide range of machine learning and statistical modeling Python packages that can be used to analyze and forecast water reuse data. Popular packages for these applications include scikit-learn (Pedregosa et al. 2011), TensorFlow (Abadi et al. 2015), and SciPy (Virtanen et al. 2020). For forecasting time series data, Facebook's Core Data Science team openly released Prophet, which is used across many of Facebook's applications (Taylor and Letham 2017). In addition to predicting future performance, machine learning may provide insights about measurement errors, such as sensor drift.

A Kalman filter is an algorithm used in control systems to correct for sensor measurement errors. Water sector applications include modeling distribution system water quality and urban drainage systems (Bartos and Kerkez 2021; Rajakumar et al. 2019) and estimating nutrient composition in pilot wastewater treatment plants (Nair et al. 2019). FilterPy (Labbe Jr. 2015a) is a popular Python package focused on Kalman filters and similar techniques. These tools can be used to evaluate individual sensor signals and also be supplemented with custom Python code to cross-reference other parameters for confirmation of true anomalies (i.e., events) as opposed to other causes like normal process changes or meter errors.

2.4 Statistical Process Control

Another option to identify events or anomalies is through the use of statistical process control techniques from manufacturing such as control charts (Kaelin et al. 2008; Nilsson et al. 2007). A study by Nilsson et al. (2007) identified two methods for identifying changes in process performance including the Shewart method and cumulative sum control charting (CUSUM). Both rely on statistical process control methods that identify outliers based on the comparison of current performance to the mean (or other similar statistic) and a control limit boundary. The control limit boundary can either be set as a fixed value or statistically estimated based on historical performance. For example, the probability of a process falling outside of a control limit that is three standard deviations from the mean is only 0.3%. Thresholds could be set to determine when a system is no longer in control, e.g., nine consecutive readings outside of the range. The benefit of the statistical approach is that it provides system-specific thresholds that may allow deviations to be identified and minimizing false positive readings.

2.5 Software Selection

In reviewing open-source software, the project team identified promising tools across multiple sectors, including engineering, data science, and industrial process control. Considering the requirements of DPR data analysis and event detection, the following factors were identified as high priority for software selection:

- Software must be built using a modern programming language that is free to use.
- Software must provide flexibility to allow adaptation to different treatment trains and regulatory standards.
- EDS must provide transparency about how events are detected to improve quality control and encourage trust.
- Few CCP failures ever occur, so detecting failures is inherently a data limited problem. The EDS must not require a high volume of training data (i.e., failure events) to perform well.

By comparing these requirements with the available software, Pecos was identified as the tool best suited for developing an EDS for this project. Pecos was developed to monitor and generate reports based on real-time data streams, such as industrial control systems. This Python-based tool is modern, flexible, and provides a series of logic-based tests that are clearly structured and understood. For example, concepts from statistical process control can be implemented using Pecos tests. Because CCP failures are rare, data-driven approaches like machine learning/AI were deemed unlikely to perform well for this project. Moreover, those approaches would not provide a clear rationale to the operator or engineer about why an event was detected.

CHAPTER 3

Event Detection Framework

Due to the lack of environmental buffers in DPR, treated water from these facilities can reach customers in minutes or hours, which can be defined as the retention time. Therefore, CCP failures must be detected and resolved in less than the retention time to avoid sending out off specification water to customers. The event detection framework aims to identify software-based solutions to support proactive and rapid responses to DPR failures. To do so, the framework is designed to increase lead time and decrease time for event detection and resolution (Figure 3-1).

Lead time is the duration between when an issue is detected and when a failure occurs. If the event is detected in advance, operators can safely resolve an issue before it becomes a threat. **Event detection** is the amount of time for software to identify an emerging or confirmed event. This step is dependent on several factors:

- Speed of data storage: time required to store data
- Frequency of analysis: how often event detection logic is run (e.g., 1-minute, 5-minute)
- Data screening: time required to identify data associated with process downtime
- Detection logic: time required to identify issues in CCP monitoring parameters, to detect events, alert operators, and create visuals

Event resolution is the amount of time needed to correct the problem after identifying an event. Ideally, the CCP failure would be resolved at the root cause, such as correcting the applied ozone dose, but it could also be resolved by diverting the product water before it reaches customers.

There are several factors that contribute to lead time, event detection, and event resolution (Figure 3-1). Lead time can be increased by identifying early indicators of failure; however, it may be difficult to determine these indicators without a first-hand experience of near misses at the facility. Event detection time, on the other hand, is within the control of the staff implementing the logic. Detection time can be decreased by running analyses more frequently and optimizing code to run faster. Compared to lead time and event detection, the timing of event resolution is perhaps the most uncertain.

If a CCP failure is detected and the root cause is not clear, operators may divert water to waste to avoid the risk of sending off specification water to customers. This approach will minimize time for event resolution but will result in waste, process downtime, and impacts on distribution system operations. If the problem can be identified quickly, these adverse effects can be minimized. Event resolution time can be decreased by providing actionable information to an operator. For well-characterized failures, control logic can be added to automate the response.

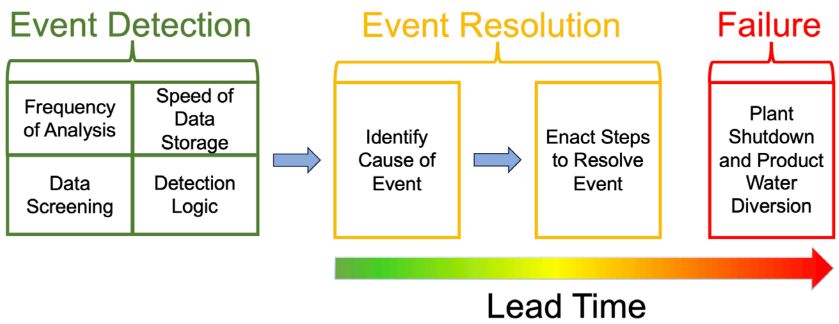


Figure 3-1. Factors Impacting the Time Needed to Identify and Respond to an Event.

A multidisciplinary team is required to implement event detection software, including process engineers, operators, IT staff, systems integrators, data engineers, and software developers. Consulting each group throughout development is recommended to avoid unexpected barriers and costs and to keep the user experience in mind:

- **Process engineers** and **operators** provide context for how the software should be configured based on DPR regulations, engineering principles, and ease of use for operators. Engineers may also participate in acceptance testing to determine whether software meets required specifications.
- **IT staff** must be consulted about purchasing software licenses, cybersecurity practices, database configuration, and network architecture. Databases are preferred to alternatives like Excel spreadsheets for critical business operations because they store data more efficiently, have security measures in place, and allow for multiple people to access data at once. Network architecture is how devices like sensors, databases, servers, and computers are connected to one another. Utility networks are often divided into two main categories: 1) a SCADA network for control of treatment infrastructure and 2) a business network for routine tasks like sending emails and storing documents.
- **Systems integrators** are responsible for programming the critical monitoring and control systems at a facility. Integrators can add new calculated values or online analyzers in SCADA system, refine control logic, and update alerts and alarms.
- **Data engineers** clean, transform, and move data to where it needs to be.
- **Software developers** can implement event detection logic and develop data visualizations and user interfaces for end users, like operators.

The event detection framework developed in this project includes the following four-step process: data storage, data screening, data flagging, and event detection (Figure 3-2). For data storage, the team must identify which SCADA tags are needed to detect and resolve events and the frequency to store and analyze SCADA data (i.e., 1-minute, 5-minute). They also must identify where and how the data is stored within the network architecture and determine the storage requirements for the database.

Technical conversations about networking and data storage will require input from the entire team. Process engineers and operators should clearly articulate what they want to be able to do. For instance, “As an operator, I want to get alerts about potential CCP failures on my phone.” The IT staff may respond that text alerts are not possible with the current network architecture because there is nothing in place to securely export data from the SCADA network. As an alternative, IT may recommend sending alerts to the HMI at the facility. When discussing databases, engineers and operators should provide an inventory of the tags that need to be monitored, the frequency data should be pulled, and who will need access to the data. With this information, IT staff can determine the database that is optimized for that task.

After storing the data, it must be screened to avoid detecting events when treatment processes are not in production. Data screening is critical to focus operators on real events and avoid nuisance alarms. Once the data is cleaned to avoid false alarms, logic is developed to label data that may indicate poor or deteriorating performance at a CCP, known as data flagging. Lastly, event detection logic is developed to identify emerging or confirmed CCP failures based on one or more data flags. Operators are notified of detections through alerts sent to HMI, texts, or emails.

Events are defined by one or more flagged CCP monitoring parameters to strike a balance between lead time, false positives, and actionable response to an issue. The more flags, the more specific the event notification can be. A single flag may indicate *that* there is an issue but not *why* it is happening. Is the problem due to upstream changes in water quality, a process failure, or a monitoring error? By defining events based on a combination of flags—those related to indicators of CCP failures and the root causes—operators are notified *that* the system is approaching a regulatory threshold and *why* it may be occurring.

In the next sections, the four-step process—data storage, screening, flagging, and event detection—is described in more detail.

Figure 3-2. Event Detection Framework Steps.

3.1 Data Storage

At a DPR demonstration or full-scale facility, there may be thousands of SCADA tags logging data in the historian. For an event detection workflow, only a small fraction of those tags must be stored and analyzed. These most important tags for this study include 1) CCP monitoring parameters (i.e., online analyzers that track whether a CCP is performing properly) to demonstrate compliance and 2) operational status indicators that describe whether processes are in or out of production. Although not included in this study, online analyzers often record information about data transmission, power usage, and sensor maintenance events. This data could also be stored and used to inform an EDS. Less commonly, sensors can perform self diagnostics to detect anomalous values, drift, or other unexpected behavior.

After selecting the SCADA tags, the team must determine the appropriate frequency for storing the data (e.g., 1-min, 5-min). The choice of frequency represents a tradeoff between speed of detection and storage costs and capacity. Because data completeness and rapid response is essential, 1-minute timesteps may be considered as a good starting point for DPR applications to provide frequent data logging and keeping data storage capacity needs reasonable.

For reliability and performance, databases are recommended for the storage of event detection data. IT staff can guide the selection and implementation of databases, but they will need input about what data is stored and who needs access. Data engineers should provide input on the format of the data, such as the column names and data types. How large the database needs to be will depend on the type of data, how frequently it is collected, and how long it needs to be stored. Different types of data take up different amounts of storage. For instance, integers require less storage than values that require decimal-level precision. The greater the number of decimals needed, the greater the storage requirements will be.

To identify what database is appropriate, consult IT staff at the utility about existing licenses, storage costs, preferred database configurations, and access controls (e.g., passwords, read/write privileges) for the database. Furthermore, IT staff will need to provide input about whether the database should be on-premises or in the cloud, such as Microsoft Azure, Amazon Web Services, Google Cloud. If on-premises, the database may be located within the SCADA or business network. The SCADA network is reserved for critical systems connected to the treatment process, so cybersecurity controls are more stringent than the business network, which is used for common business operations like shared file storage and email.

To facilitate conversations about network architecture, consider all inputs and outputs of the event detection framework. For instance, to integrate alerts and alarms directly into the human machine interface (HMI) the operators use to monitor treatment processes, the database may need to be within the SCADA network. Alternatively, if located in the cloud, alerts and alarms could be sent to operators via text or email.

3.2 Data Screening

When a treatment process is down for maintenance, the event detection logic should be suspended until that process comes back online. To track downtime, the status of pumps and unit processes are recorded using operational status codes. For instance, consider a membrane process that cycles between different states: offline, in production, backwashing, membrane integrity testing, clean in place (CIP), etc. These states are often logged as integers (e.g. 0, 1, 2, 3, …) as the membrane status tag. When the membrane is backwashing, the status would record 2 until the BW sequence is complete. Because the public will only be exposed to water when the system is “in production,” the data screening step can be used to limit downstream analyses to periods when the membrane system is in state “1” status.

These status codes generally align well with treatment operations; however, some situations require correction. For instance, if data has been aggregated—for instance, by taking the 1-minute average of 5-second observations—status codes (integers) may be converted to a decimal, such as 0.5 if the process switched between 0 and 1 halfway through that minute. This approach distorts status data making it more difficult to identify and remove the appropriate values. As an alternative, 1-minute instantaneous values, either the first or last 5-second observation recorded during the minute could be stored.

Hydraulic lags are another common issue that status codes may not account for. For example, in the first few minutes after membranes return to production, anomalous values may be recorded by online analyzers. Consider a turbidimeter that is connected to membrane piping to measure effluent turbidity. That effluent pipe may also convey water for BW or chemical cleans. After completing a BW and returning to production, residual BW water needs to be flushed out of the turbidimeter sample line. Until that occurs, the turbidity readings may not be representative of effluent water quality. In such cases, time lags might need to be incorporated into status codes to avoid false positives in event detection. Adding or modifying codes will require assistance from a systems integrator. A well-designed data screening step will identify only the relevant performance data for use in downstream steps.

3.3 Data Flagging

After screening data for periods of downtime, the data is ready for analysis. The data flagging step labels values that may indicate poor or deteriorating performance for CCP monitoring parameters. Individual flags may not require action; however, they provide insights into the types of issues that are evolving. If an event is detected, the flags provide details to operators about what is causing the CCP to fail.

To flag CCP monitoring parameters, three types of logical tests were considered: range (above or below bounds), stagnant data, or abrupt change (Figure 3-3). A range test identifies when parameters go beyond specified threshold. Range tests are useful because they may provide operators with advanced warning of an issue (i.e., before a parameter reaches either a regulatory limit or another defined limit, such as one based on historic records of performance). Range tests are also the simplest tests—they are understandable and easy to verify that tests are working as expected.

Stagnant data test checks whether values are repeating for an extended period. For continuous values, like conductivity or pH, stagnant data could result from many causes, such as data transmission failure, a meter placed on hold during maintenance, or a meter malfunction. These types of scenarios compromise the accuracy of CCP monitoring. It is important to note, however, that repeated values are expected and acceptable for some data tags. For example, operational setpoints or status codes (e.g., membrane filter status discussed previously) would be expected to maintain a constant integer value.

Lastly, abrupt change tests check whether values are rapidly increasing or decreasing over time. These tests could indicate an issue such as the sudden loss of a disinfectant residual or the entrance of a peak of chemicals, both of which may require operator intervention. Compared to range and stagnant data tests, the abrupt change test is perhaps the most difficult to define. When considering abrupt changes in data, one must define the rate-of-change threshold and the time window to consider. If a single data point changes rapidly, should that be flagged, or should the logic require multiple values to avoid false positives? What if there is a missing value or process downtime? How should the abrupt change logic consider missing or screened values?

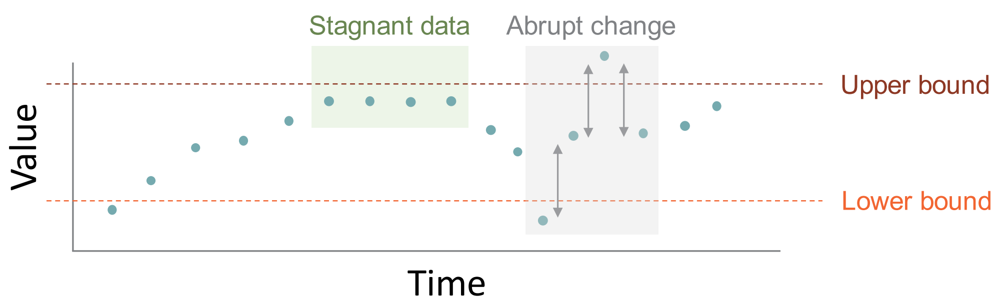


Figure 3-3. Logical Tests for CCP Monitoring Parameter Data Flagging.

3.4 Event Detection

Flags serve as the building blocks for event logic in the framework, in which events are defined by one or more flags simultaneously occurring. For example, for an ozone CCP, there may be a regulatory limit for microbial pathogen LRV. To provide advanced warning about that limit, a range test may be implemented when LRV approaches the limit or falls outside of a typical operating range. If the LRV data is flagged by a range test, it indicates *that* LRV has dropped below a threshold but not *why* that occurred. Ozone LRV is calculated based on several parameters, including temperature, hydraulics, and ozone application. Is the downward trend in LRV due to upstream changes in water quality, changes in retention time, ozone dosing, or a monitoring error? By defining an event based on a combination of flags—those related to indicators of CCP failures (pathogen LRV threshold) and the root causes (water quality, process, or monitoring failure)—operators can address the root cause of the failure.

In the ozone LRV example, the lead time is associated with how close to the regulatory limit the LRV data is flagged. By adding a larger safety factor, lead time will increase but nuisance alarms (false positives) will also be more frequent. Response times can decrease if events describe what the problem is and why it is occurring so operators can quickly resolve the issue. If the root cause of the event is well understood, response times can be decreased even further if the detection and control logic can be programmed directly into SCADA (via a systems integrator).

Although many events are characterized by multiple CCP monitoring parameters, some events require only one parameter. Consider UV disinfection in a reuse treatment train. UV dose is a regulated parameter, so if the UV dose drops, it could result in a CCP failure. In this case, the indicator of failure (UV dose) and the root cause of the failure (UV dose) are the same. The operator should be notified that there is an issue with the UV equipment or the dosing setpoint to resolve the problem.

3.5 Guidance for Process Engineers

To guide the development of an EDS, the project team has developed the following brainstorming exercises for process engineers and their teams to track which CCP failures need to be detected and what data is required.

- **Initial evaluation:** exercise to identify CCP failures and related data
    1. CCP: list unit processes
    2. Critical limits: list limits related to each CCP
    3. CCP monitoring parameters: list parameters that must be monitored to confirm CCP performance
    4. CCP status indicators: list SCADA tags that indicate whether unit process is in production or out of service
    5. Monitor failure: describe most likely scenarios that would result in a CCP failure due to a metering issue
    6. Water quality failure: describe most likely scenarios that would result in a CCP failure due to upstream water quality degradation
    7. Process failure: describe most likely scenarios that would result in a CCP failure due to equipment malfunction or damage
- **Prioritization and gap analysis:** exercise to prioritize which CCP failures should be included in event detection and whether the required data is available
    1. Prioritize events: identify CCP failure events that provide the most value for operators
    2. Data gaps: identify whether existing SCADA tags are sufficient to implement prioritized events
    3. Systems integrator collaboration: if possible, add tags to fill data gaps

3.6 Event Detection Framework Summary

Key takeaways from Chapter 3:

- The event detection framework is defined as a four-step process: data storage, data screening, data flagging, and event detection.
- The aim of the framework is to promote a proactive and rapid response to emerging CCP failures through software.
- Implementing event detection for potable reuse requires a multidisciplinary team, including process engineers, operators, IT staff, systems integrators, data engineers, and software developers.
- Representatives from each discipline on the team should be consulted throughout the development of the EDS to clearly define goals and expectations.

- Data storage lays the foundation for event detection and relies heavily on coordination with IT staff, systems integrators, and data engineers.
- Process engineers and operators should communicate their understanding of the regulatory requirements and system operations to inform the development of data screening, data flagging, and event detection logic.
- Events should be defined such that, when detected, operators are notified that a CCP failure may be approaching and the likely cause of the failure.

The following section, Chapter 4, describes the implementation of the event detection framework at a DPR demonstration facility in California. The discussion includes a detailed description of the hardware and software specifications, the rationale behind tag selection, data screening logic, data flagging and event configuration, and the user interface and visuals developed for operators.

CHAPTER 4

Framework Implementation: City of San Diego

The developed framework discussed in the preceding chapter was used to build an EDS for deployment at the City of San Diego's (City's) 1 MGD NCPWDF. While the NCPWDF was originally intended to demonstrate the effectiveness of the City of San Diego's surface water augmentation IPR project, the site was an ideal testing ground to implement the prototype due to the similarity of the process train with California's default DPR treatment train. The deployed EDS was intended to validate the framework detailed in Chapter 3.

EDS development benefited from access to historic operational and water quality data from the NCPWDF. The historical data was used to inform the data flagging test parameters for event detection. In the absence of historic data availability, test parameters would be set based on regulatory or operational limits. Not all data sources benefitted from historic data due to seasonal or other fluctuations preventing the test inputs for some data sources from being easily determined. For example, ozone residual concentrations at a given monitoring location differ greatly based on the setpoint dose of ozone to achieve 1-log removal of *Cryptosporidium*, as well as variations in water quality that impact target ozone CT for disinfection. On the other hand, parameters such as RO permeate TOC are more stable and better suited to apply historic data analysis to inform the flagging thresholds.

In the case of (1) having historic data or (2) needing to use existing limits as a known reference, the task of optimizing data flagging for event detection is an iterative process which uses data from a known or simulated event occurrence to tailor the given event detection sensitivity. The EDS was configured to analyze plant data in real-time while proactively identifying and categorizing events within each of the unit processes. Achieving proactive and targeted event detection should reduce troubleshooting and provide operators with greater time to respond and resolve the event.

In addition to summarizing the facility's characteristics, the discussion below details the development of the deployed event detection criteria for each unit process at the NCPWDF using the four-step framework of (1) data source selection, (2) data screening, and (3) data flagging, and (4) event detection.

4.1 Facility Description and Design Criteria

The NCPWDF's treatment train consists of ozonation and BAC (added in 2014) upstream of MF and UF skids operating in parallel. The combined filtrate from the MF and UF processes is the feed to the RO process that has two parallel 0.5 MGD trains—Train A and Train B. RO permeate is the feed water to the UV disinfection and AOP process. Figure 4-1 provides an overview of the NCPWDF treatment train.

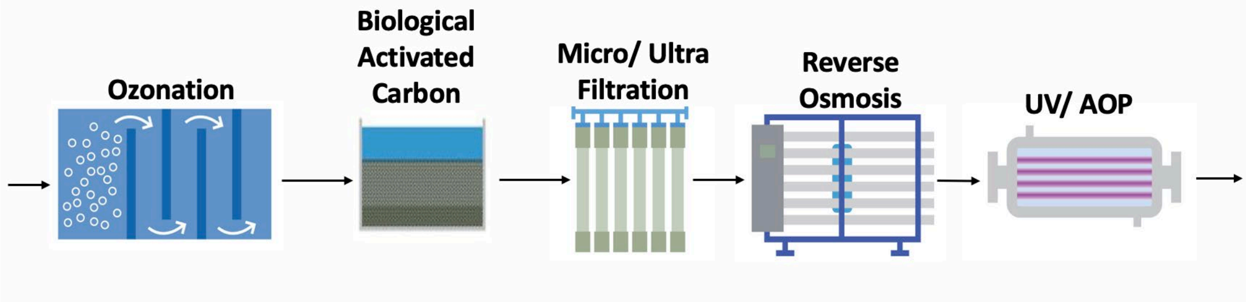


Figure 4-1. NCPWDF Treatment Train Diagram.

Each individual treatment process contributes to the removal and/or inactivation of pathogens from the water. The 4 critical control points (CCPs) for pathogen removal are ozone, MF, RO, and UV/AOP. To protect public health and meet the overall pathogen removal goals in the product water, each CCP must individually meet its treatment goals. The following discussion outlines the design of each CCP in the treatment train. Table 4-1 summarizes how each unit process CCP achieves pathogen credit as a log-removal value.

Table 4-1. Unit Process Pathogen Removal Summary.

Unit Process	Mechanism of Pathogen Treatment	LRV Crediting
Ozone	Oxidative disinfection	Concentration-Time based disinfection calculation.
Membrane Microfiltration	Physical barrier	Membrane integrity directly tested daily via pressure decay across the membranes. Filtrate turbidity continuously monitored to indirectly assess removal.
Reverse Osmosis Membrane Filtration	Physical barrier	TOC and EC removals used as surrogates to determine pathogen removal.
UV/AOP	Oxidative inactivation	UVT and UV Dose thresholds must be met to achieve any pathogen removal credit.

Pathogen removal achieved by the ozone process is quantified with pathogen-specific LRV. In California, per the draft DPR regulations, the maximum LRV that can be granted for a pathogen through a single unit process is 6-logs (State Water Board 2021). Of the three indicator pathogens—virus, *Giardia*, and *Cryptosporidium* (V/G/C)—*Cryptosporidium* is the most resistant to inactivation via ozonation. Therefore, *Cryptosporidium* LRV is the key metric used to assess the ozone system's performance and dictates operational dosage. The ozone dose is set to meet a target *Cryptosporidium* LRV of at least 1-log. If the ozone system meets the target LRV of at least 1-log *Cryptosporidium* reduction, then the maximum 6-log reduction of virus and *Giardia* are also obtained since the ozone CT requirement for 1-log *Cryptosporidium* is higher than for 6-log virus and *Giardia* (U.S. EPA 2010).

The major components of the ozone process are the ozone generator, the injection system, and the pipeline contactor. Ozone is generated onsite at the NCPWDF and is then injected into the process stream to inactivate pathogens and oxidize chemicals present in the water. The ozone generator has a capacity of 190 pounds per day (ppd) at 10% ozone by weight. After ozone is injected into the main process flow, it travels for approximately 100 ft in an 8-inch pipe, after which it enters the pipeline contactor, made of 24-inch diameter PVC pipeline with 360 feet in total length (Figure 4-2). At 1,100 gallons per minute (gpm), the contact time provided is approximately 7 minutes. Flow straighteners are used at each turn to maintain plug flow hydraulics by redistributing the flow and minimizing short-circuiting. The design criteria for the NCPWDF ozone process are listed in Table 4-2.

Table 4-2. NCPWDF Ozone System Design Criteria.

Parameter	Units	Value
Ozone Generation and Injection System		
Design Flow	MGD	1.6
Generator Capacity	PPD	190 at 10% wt.
Maximum Applied Dose	mg/L	14.3
Manufacturer		Wedeco
Ozone Contactor		
Pipeline Diameter	Inches	24
Pipeline Length	feet	6 x 60 ft segments
Total Volume	gallons	7800
Contact Time	minutes	7.5 (at 1.5 MGD)

Figure 4-2. Ozone Pipeline Contactor at NCPWDF.

After the ozone product water is filtered via BAC filtration, the water is sent to MF and UF skids operating in parallel.

The NCPWDF uses pressure-driven membrane modules (Figure 4-3) which contain thousands of hollow membrane fibers. A feed pump supplies the driving pressure needed to pass influent water through the pores in the hollow fibers. Treated water is referred to as filtrate.

Membrane filtration systems physically remove suspended solids and select pathogens from the process water through size exclusion. Pathogen LRV can be obtained based on the pore size of the membranes, which is 0.1 µm for MF and 0.01 µm for UF. Although the NCPWDF treatment train includes both MF and UF systems that operate in parallel, MF was the focus of event detection framework configuration and implementation. The framework implementation process described herein for MF is also applicable to UF due to the similarities shared between the two processes.

Membrane filtration systems can demonstrate greater than 4-log removal of *Cryptosporidium* and *Giardia* through membrane integrity tests (MIT) and continuous monitoring of filtrate turbidity. Although viruses can be removed by the membranes, United States regulatory agencies typically do not grant virus LRV credit. Table 4-3 summarizes the design criteria for the MF process at the NCPWDF.

Table 4-3. NCPWDF MF System Design Criteria.

Parameter	Units	Value
Net Product Flow	MGD	.625
Nominal Pore Size	µm	.1
Number of Modules		50
Area per Module	Square feet	538
Instantaneous Flux	GFD	29
Recovery	%	93
Chemical Cleaning Frequency		>3 months
Manufacturer		Pall Corporation
Membrane Material		Polyvinylidene fluoride

Figure 4-3. Membrane Filtration Skid at NCPWDF.

MF filtrate is sent to the RO system at the NCPWDF. The RO system consists of two trains that both receive combined membrane filtration filtrate as feed water. Train A is a 2-stage system and Train B is a 3-stage system.

RO membrane treatment separates dissolved solutes such as salts, contaminants, and pathogens from water through a highly selective semi-permeable membrane. Pressure is applied on the feed water side of the membranes to overcome osmotic pressure and force water across the membrane to generate a permeate stream. The portion of water that remains on the feed side is called concentrate. Table 4-4 summarizes the design criteria for the NCPWDF RO process and Figure 4-4 displays a photo of the RO skid.

Table 4-4. NCPWDF RO System Design Criteria.

Parameter	Units	Train A	Train B
Net Product Flow	MGD	0.5	0.5
Membrane Manufacturer and Type		Toray TMG20D-400	Toray TMG20D-400
Number of Elements		105	108
Element Area	ft^2	400	395
Elements per Vessel		7	6
Number of Vessels		10:5 (Stage 1:Stage 2)	10:5:3 (Stage 1:Stage 2: Stage 3)
Instantaneous Flux	GFD	12	12
Recovery	%	75-80	75-80
Chemical Cleaning Frequency		>3 months	>3 months
System Manufacturer		EnAqua	EnAqua

Figure 4-4. RO Skid at NCPWDF.

The UV/AOP system is the final process step in the advanced treatment train. During the UV/AOP process, sodium hypochlorite is dosed as a chemical oxidant in the presence of UV light to produce hydroxyl radicals that break down recalcitrant organic compounds. The UV dose also inactivates pathogens and is credited with 6.0 logs of V/G/C removal if operational criteria specified by regulatory requirements are met. The system has been designed to remove 1.2 logs of NDMA to comply with discharge permit requirements and 0.5 logs of 1,4-dioxane, a treatment objective for AOP, using a sodium hypochlorite dose of 1 mg/L and a minimum UV dose of 850 mJ/cm^2. The design criteria for the UV/AOP system are summarized in Table 4-5 below and a photograph of the reactor at the facility is shown in Figure 4-5.

Table 4-5: NCPWDF UV/AOP System Design Criteria.

Parameter	Units	Value
Design Flow	MGD	1.0
Reactor Manufacturer		Trojan UV
Number of Lamps		72
Watts per Lamp	W	240
Total Power	kW	17.3
Design UV Transmittance		>95%
Electrical Energy per Order for NDMA	kWh/kgal	0.18
Electrical Energy per Order for 1,4-dioxane	kWh/kgal	0.46
AOP Oxidant		Sodium Hypochlorite
Oxidant dose	mg/L	1.0
Design UV dose for 6-log pathogen inactivation	mJ/cm^2	278
Design UV dose for AOP	mJ/cm^2	850

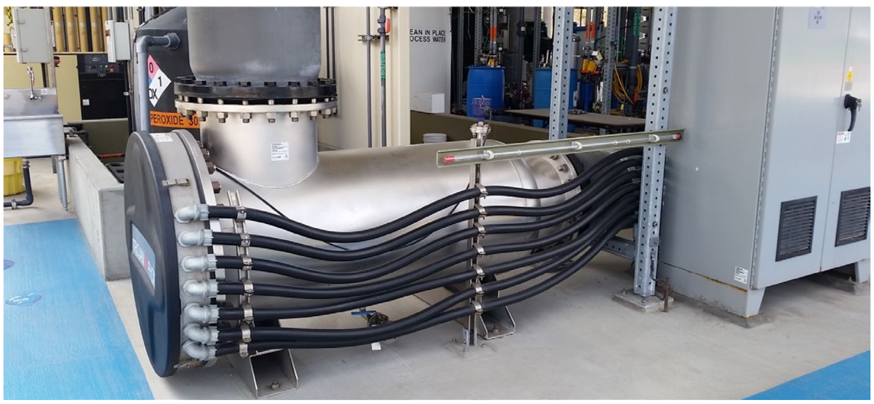


Figure 4-5. UV/AOP Reactor at NCPWDF.

The UV/AOP system is fully automated and controlled by a programmable logic controller (PLC) and power distribution center. The target UV dose is calculated based on the measured process water flow, UV transmittance (UVT), UV lamp status, and UV intensity (UVI). Provided that these parameters remain within the bounds of the system's design, the applied UV dose will continuously be at or near the target setpoint regardless of fluctuations in feed water quality because the process controls will adjust the power delivered to compensate for changes.

Establishing software and data handling practices was necessary to detect potential failure events preemptively across each CCP using a custom software tool at the NCPWDF, additional software and data handling practices were necessary.

4.2 Software and Data Workflow

The project team chose to implement the event detection system on-premises, as opposed to the cloud, based on the City's IT preference to keep NCPWDF data within the SCADA network. The workflow is presented in Figure 4-6 with a description of each step provided in Table 4-6 below. The NCPWDF PLCs transmit data to a Windows computer that displays real-time facility information to an HMI screen. On that computer, data management software, called

KepServerEX (with the KepWare Data Logger add-on) connects with the PLCs and records the data to a database (Microsoft SQL Server Express). The data is logged to the database every minute for each SCADA tag.

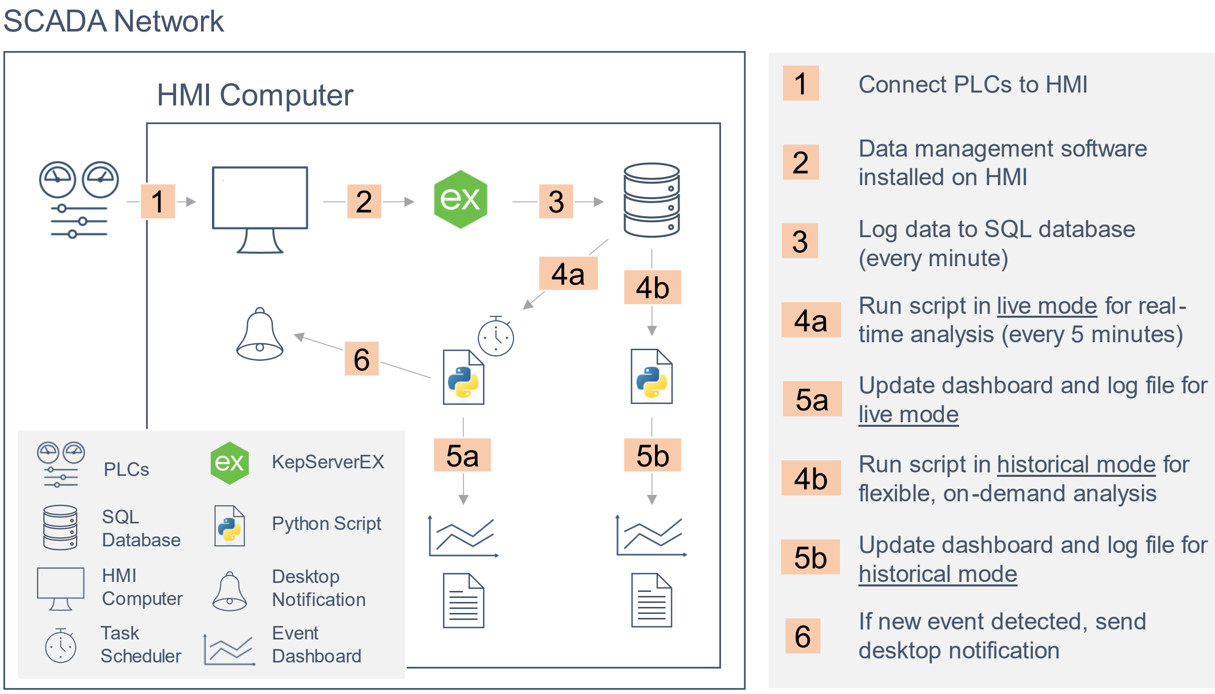


Figure 4-6. Software and Data Workflow for the Implemented EDS at NCPWDF.

The Python scripts, data, and additional documentation are freely available in a Github repository, which can be accessed via the 4954 project page on the WRF website. There are two modes for the Python scripts that perform event detection: 1) live mode and 2) historical mode. The live mode script performs event detection analysis in near real-time (configured to run every 5 minutes using Task Scheduler). This analysis displays data from the past hour of operations. After the script is run, the event detection dashboard (HTML/CSS) and log files are updated, and if an event was detected, an alert is sent to the computer as a desktop notification. An alert is only triggered if there is a new event—one that was not occurring the last time the script was run—to avoid nuisance notifications. Historical mode allows the user to specify the date and time the event detection analysis begins and ends. While live mode supports the tool's primary function, rapid event detection, historical mode allows users to investigate past data and test new event logic on known event occurrences in past data.

The organization of the Python code is detailed in Table 4-6 and aligns with the event detection framework outlined in Chapter 3, including data storage, data screening, data flagging, and event detection. Additional features were included in the EDS to log information about the system performance, visualize monitoring data for operators to review, and notify operators if events are detected. Table 4-6 describes the primary Python package used to implement each step. Python packages are coding toolkits that can be easily shared and reused by other programmers. One of the most popular packages is Pandas, which simplifies the process of reading, cleaning, and transforming data. The scripts rely on more than a dozen packages, but the core features are implemented using Loguru, Pandas, Pecos, and Plyer. Pecos is central to the data screening, flagging, and event detection aspects of the EDS.

Table 4-6. Python Script Steps for Live and Historical Modes.

Step	Description	Primary Python Package
1. Initialize logging file	Create a log file to record the script's progress to provide a comprehensive record. The file tracks detected events, database connectivity, data quality, and errors in the code. Throughout the script, four types of messages are recorded: info, warning, error, and critical.	Loguru
2. Import and clean data	Read in data from database and tag information. Then clean and transform data by checking for missing data, calculating absolute value of error tags and/or difference calculations, and create calculated columns (i.e., primary event tags) for events associated with multiple tags.	Pandas
3. Screen data	Using treatment status data, screen data associated with downtime. If status tags are not available, assume the process is online.	Pecos
4. Flag data and detect events	Implement quality control tests to flag data. Detect events based on CCP monitoring parameters that have been flagged.	Pecos
5. Update event detection dashboard	Create dashboard using images and HTML for data visualization. A CSS file is used to customize the look and feel of the dashboard.	Pecos
6. Alert if new events are detected	Send desktop notification if any new events have occurred since the last time script was run.	Plyer

Pecos was developed to provide insights into the quality of data collected by sensor technologies. The package includes several features, which are defined by its developers as "quality control tests" and "time filters." Quality control tests provide a structured approach to setting limits on acceptable performance, such as stagnant values, abrupt changes, and out-of-range measurements. For NCPWDF implementation, data flags and event detection logic focused on what the project team defined as "range tests" and "stagnant data tests." A range test is violated when a value is above or below a threshold. A stagnant data test is violated when a value remains constant when variation is expected. To screen the data during downtime, time filters are used to indicate when quality control tests should be suppressed.

Quality control tests are used during data flagging and event detection. Tests were developed for all CCP monitoring parameters to identify unusual or unacceptable behavior at monitoring locations. When that behavior occurs, those values are flagged using Pecos and visualized using automated reporting tools. An event is defined by one or more CCP monitoring parameters being flagged at the same time. To implement event detection logic using quality control tests, an "event tag" is created if more than one monitoring parameter is needed to identify an event. If the event tag is flagged, an event is detected. An event tag is an aggregation of all the conditions that must be met for an event to occur. For examples of event tags, refer to subsequent sections on data flagging and event detection for ozone, MF, RO, and UV/AOP.

To visualize events and associated CCP monitoring parameters, a dashboard was developed to investigate flagged values. The event detection dashboard created by the Python scripts is a web application composed of HTML, CSS, and image files. Typically, web applications are hosted and viewed in a web browser over the Internet. Because the EDS created for NCPWDF was deployed within the SCADA network, no internet connection is available. However, the dashboard is still accessible on the computer without internet access. To access the dashboard offline, users must navigate to the folder that contains their Python scripts. If the scripts have been run with default settings, a file named *dashboard.html* will be in the same folder. By opening this file in a web browser (i.e., double-clicking the file), the dashboard will appear and will function like a website. Whenever the script has been run, the browser must be refreshed to show the latest results. If this project were deployed in the cloud, the dashboard could be hosted on a website and refreshed in real-time.

The events that were configured in the script and the SCADA tags that support the event detection logic are described below for each CCP unit process and plant-wide.

4.3 Ozone Configuration

4.3.1 Ozone Data Selection

Pathogen log-removal values are the metric by which the ozone process is assessed. Thus, detecting CCP failure events is done by monitoring pathogen LRV. To give context to the relevant data sources for the EDS, the calculation of pathogen LRV is described below.

The measurements required for the calculations to determine LRV are process water flow rate, dissolved ozone residual concentrations at a minimum of three sampling locations—referred to as ozone sampling points (OSPs), and water temperature. These three measurements and the associated intermediate calculations used to determine *Cryptosporidium* LRV are considered CCP monitoring parameters. The NCPWDF ozone contactor has 14 total OSPs numbered in order of increasing distance from the injection point. Meters are installed only at OSP 4, OSP 7 and OSP 10 for continuous monitoring of the dissolved ozone residual.

The general steps for calculating pathogen LRVs achieved by ozone are:

1. Calculate hydraulic detention time (HDT) using the water flow rate.
    - Between the injection point and OSP 4.
    - Between OSP 4 (1st monitoring location) and OSP 7 (2nd monitoring location).
    - Between OSP 4 and OSP 10 (3rd monitoring location).
    - Between OSP 4 and the end of the contactor.
2. Calculate and select the ozone decay rate constant.
3. Calculate Concentration-Time (CT).
4. Calculate pathogen LRVs using CT and the inactivation rate constant specific to each pathogen.

The following discussion details step-by-step the continuous calculation of pathogen LRVs carried out by the plant's control system.

First, HDT between the various points within the contactor listed above are calculated using the Equation 4-1 below. This value is a simplifying assumption that is later corrected with a baffling factor as seen in Equations 4-5 and 4-6 below.

$$HDT\ (\mathrm{min}) = \frac{\text{Volume of pipe section (gal)}}{\text{Water flow (gpm)}} \qquad \text{(Equation 4-1)}$$

Next, the ozone decay rate constants are calculated. Ozone decay follows a first-order reaction after the initial rapid decay reactions caused by ozone demand present in the feed water have occurred. A first-order reaction assumes that the rate of residual decay is proportional to a constant. The ozone decay rate constant is commonly referred to as the ozone decay coefficient. To determine the decay coefficient, the ozone residuals at two locations within the contactor and the HDT between these two points must be known. The concentration at non-monitored points can then be predicted using the decay coefficient.

The equations for calculating the ozone decay coefficient between OSP 4 and OSP 7 ($k_{1,2}$) and OSP 4 and OSP 10 ($k_{1,3}$) are presented below in Equation 4-2 and Equation 4-3, respectively. The subscripts indicate the monitoring points between which the decay is evaluated. For example, the subscript "1,2" indicates that the ozone decay coefficient was evaluated using the first and second sampling points (OSP 4 and OSP 7).

$$k_{1,2} = \frac{\ln\left(\frac{C_1}{C_2}\right)}{HDT_{1-2}} \qquad \text{(Equation 4-2)}$$

$$k_{1,3} = \frac{\ln\left(\frac{C_1}{C_3}\right)}{HDT_{1-3}} \qquad \text{(Equation 4-3)}$$

where:
C_1= Ozone residual at OSP 4
C_2=Ozone residual at OSP 7
C_3=Ozone residual at OSP 10
HDT_{1-2}= Hydraulic detention time between OSP 4 and OSP 7
HDT_{1-3}=Hydraulic detention time between OSP 4 and OSP 10

The following first order decay equation is then used to predict the concentration of any location downstream of a sampling point:

$$\mathrm{C} = \mathrm{C_0}\mathrm{e}^{-\mathrm{kt}} \qquad \text{(Equation 4-4)}$$

where:
C = Concentration at time t downstream of C_0
C_0 = Upstream ozone concentration
k = Ozone decay coefficient
t = Process water travel time between monitoring locations of C_0 and C

Third, the ozone CT is evaluated. The truncated extended-integration method is used to evaluate the CT through the ozone contactor. Typically, CT is not granted in the dissolution and reactive zones of the ozone system where much of the applied ozone dose is consumed by the rapid-rate reactions of the process water. At the NCPWDF, the dissolution zone starts at the point of ozone injection and ends at the off-gas point. The reactive zone starts at the off-gas point and ends where OSP 4 is located. CT in the dissolution and reactive zones is conservatively determined by assuming that the ozone concentration throughout the dissolution and reactive zone (injection site to OSP 4) is equal to the ozone residual measured at the first sampling location (OSP 4). Calculation of CT for the remainder of the contactor is performed using integration.

A visual representation of CT calculated using the truncated extended integration method is shown in Figure 4-7 below. The CT_{rect} corresponds to the CT granted in the initial dissolution and reactive zone. CT_{curve} corresponds to the CT granted using the continuously calculated residual throughout the system using the ozone decay coefficient.

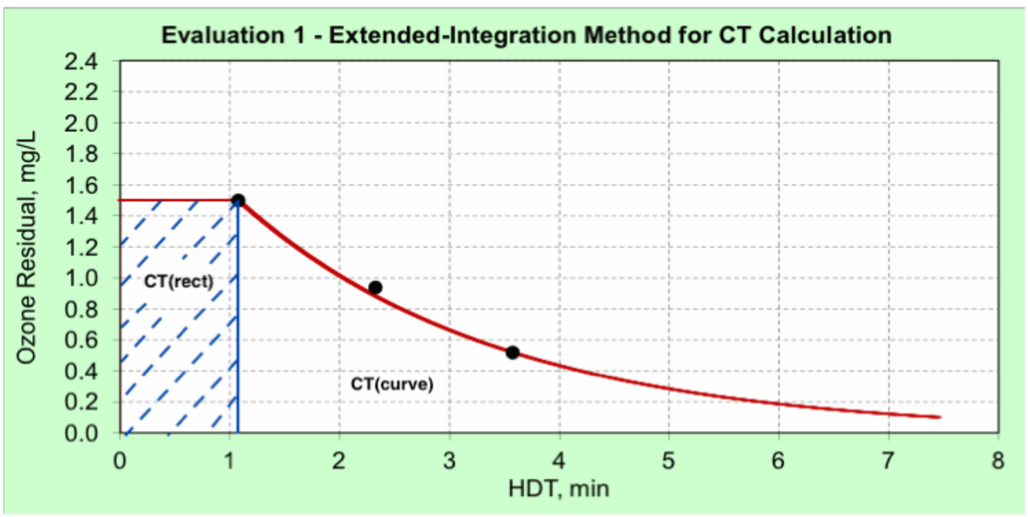


Figure 4-7. Visual Representation of Evaluating CT using the Truncated Extended Integration Method.

The total CT for the ozone system is the sum of $CT_{rectangle}$ and CT_{curve}. Equations 4-5 and 4-6 for evaluating the CT of the rectangle and curve portion are presented below:

$$CT_{rectangle} = \left(\frac{T_0}{T}\right)_{0-1} \times C_1 \times HDT_{0-1} \quad \text{(Equation 4-5)}$$

$$CT_{curve} = \left(\frac{T_0}{T}\right)_{1-end} \times \left(\frac{C_1}{-k^*}\right) \times \left(\left(e^{-k^* \times HDT_{1-end}}\right) - 1\right) \quad \text{(Equation 4-6)}$$

Where: C_1 = ozone residual at OSP 4 (the first monitoring location after injection)
$\left(\frac{T_0}{T}\right)_{0-1}$ = Baffling factor from injection point to OSP 4 (the first monitoring location).
$\left(\frac{T_0}{T}\right)_{0-end}$ = Baffling factor from OSP 4 to the end of the contactor
k^* = Maximum between $k_{1,2}$ and $k_{1,3}$ (see above)

HDT_{0-1} = Hydraulic retention time between the injection point and OSP 4.

HDT_{1-end}=Hydraulic retention time between OSP 4 and the end of the contactor

Lastly, once the total CT is known the pathogen LRVs are calculated by multiplying the CT by a temperature-dependent inactivation rate constant specific to each pathogen. The larger an inactivation rate constant, the more effective a given ozone dose will be at inactivating the pathogen. The relationship used to find the inactivation rate constants of each pathogen at a given temperature are shown below in Equations 4-7, 4-8, 4-9.

$$k_c = 0.0397 \times 1.09757^T \quad \text{(Equation 4-7)}$$
$$k_g = 1.038 \times 1.0741^T \quad \text{(Equation 4-8)}$$
$$k_v = 2.1744 \times 1.0726^T \quad \text{(Equation 4-9)}$$

where: k_c= *Cryptosporidium* inactivation rate constant
k_g = *Giardia* inactivation rate constant
k_v = Virus inactivation rate constant
T = Water temperature, °C

To evaluate the LRV, the CT is multiplied by the inactivation rate constant—as shown below.

$$\mathit{Cryptosporidium}\ \mathrm{LRV} = k_c \times \mathrm{CT} \quad \text{(Equation 4-10)}$$
$$\mathit{Giardia}\ \mathrm{LRV} = k_g \times \mathrm{CT} \quad \text{(Equation 4-11)}$$
$$\mathrm{Virus\ LRV} = k_v \times \mathrm{CT} \quad \text{(Equation 4-12)}$$

Ozone system monitoring is composed of hundreds of measured and calculated sources of data (i.e., tags). However, only a handful of tags are necessary for quickly and accurately assessing the ozone process performance. These tags are referred to as CCP monitoring parameters in this report. Identifying the essential tags for inclusion within the EDS was a crucial step in development. Although each of the hundreds of tags provides insight and information about the ozone system's operation, the selected tags shown in Table 4-7 are those which relate to the calculation of *Cryptosporidium* LRV as detailed above—and thus directly correspond to process performance and ensuring adequate public health protection.

Table 4-7. Ozone CCP Monitoring Parameters.

Parameter	Units	Importance to System Monitoring
Dissolved ozone residual at OSP 4, 7, and 10	PPM	• Used to assess meter functionality and process water quality • Used to calculate ozone demand, ozone decay coefficient, CT, and LRV
Ozone generator production	PPD	• Used to assess process functionality • Used in the ozone demand calculation

Temperature	deg F	• Used to calculate pathogen inactivation rate constant for LRV calculation
Water flow	gpm	• Used to calculate HRTs for CT calculation
Ozone demand	PPM	• Calculated using ozone generator production and dissolved ozone residual at OSP 4 • Used to monitor water quality changes of the feed water
Ozone decay coefficient	min^{-1}	• Calculated using the dissolved ozone residual at OSP 4, 7, and 10 • Used in CT calculation
CT	PPM*min	• Calculated using HDT, OSP 4 ozone residual, and ozone decay coefficients with the EPA truncated extended-integration method
Pathogen Removal	LRV	• Quantifies treatment of the system • Calculated using CT and a pathogen-specific inactivation rate constant

Although not directly included in the multi-step calculation of *Cryptosporidium* LRV, ozone demand is a tag that was determined to be beneficial for assessing process health and treatment within the event detection framework. The ozone demand corresponds to the initial rapid-rate reactions of the process water and is correlated to changes in LRV. As detailed in Equation 4-13, the calculation of demand requires fewer measured inputs and intermediate calculations than LRV, which facilitates an ease of demand verification within the event detection logic discussed below.

$$\text{Ozone Demand}\left(\frac{mg}{L}\right) = \text{Applied Ozone Dose} - \text{OSP 4 Ozone Residual} \quad \text{(Equation 4-13)}$$

Figure 4-8 depicts the location of each of the CCP monitoring parameter used by the EDS with the ozone system.

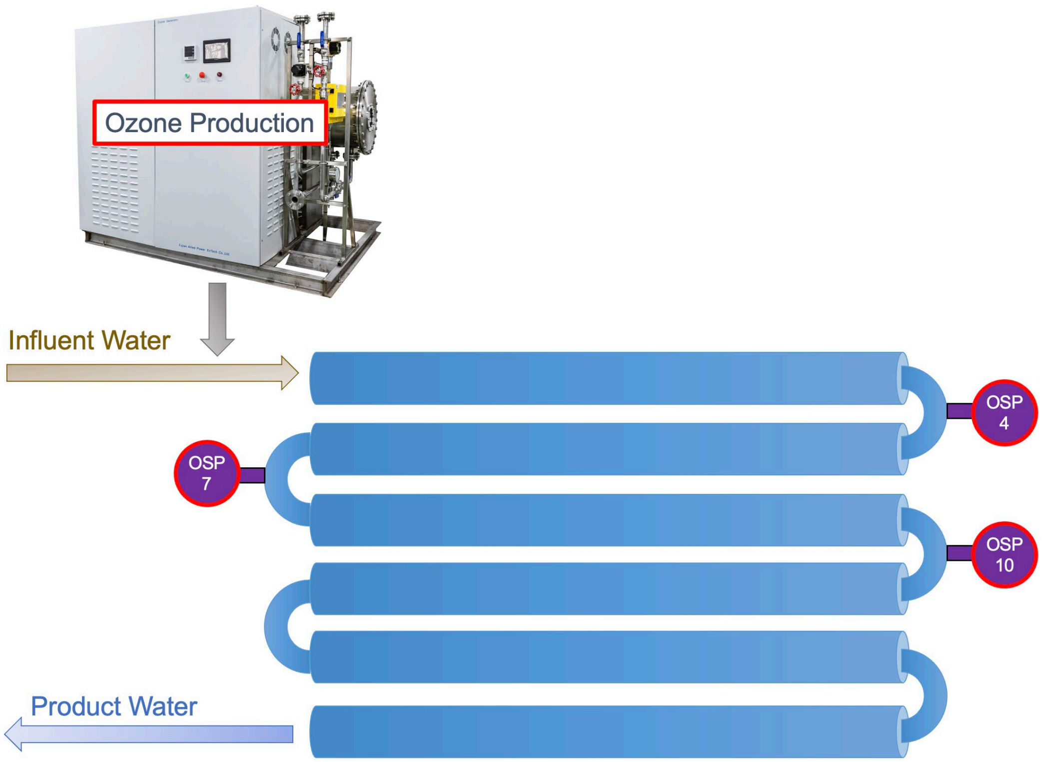


Figure 4-8. Schematic Overview of the Ozone Process with CCP Monitoring Parameters Detailed in Red.

The introduction of calculated tags into the EDS enabled multiple CCP monitoring parameters to be compared at once. The calculated percent differences listed in Table 4-8 were incorporated to monitor ozone generation and evaluate the accuracy of ozone residual meter readings. Each of the process parameters uses a reference measurement or setpoint to evaluate the accuracy and/or proper functioning of the online measurement. The development of these additional data tags is a crucial aspect of the event detection framework. These tags provide context to the operations of the ozone system and verify the veracity of the CCP monitoring parameters. The event criteria outlined in Section 4.3.3 incorporates the calculated parameters in Table 4-8.

Table 4-8. Ozone Process Calculated Parameters.

Parameter	Units	Description
Ozone Generator Production Difference	unitless	• OSP Meter Rolling Average
OSP Meter Difference	unitless	• Decimal error difference between the primary and the reference ozone residual meter at OSP 4 and at OSP 7 • Used to assess meter functionality and drift
OSP Meter Rolling Average	PPM	• Arithmetic mean of the previous 120 minutes of ozone residual measurements. • Used to detect meter drift when redundant meter is non-operable

4.3.2 Ozone Data Screening

To detect events within the ozone system, data collected during periods of non-operation or abnormal production must first be removed from analysis. During BAC filter BW, the process water flow decreases through the ozone contactor. The decrease in flow temporarily increases the HDT, and thus the CT and subsequent LRV values also increase. The known temporary increase to LRV is removed from event detection analysis so that the tool does not incorrectly identify anticipated operations as an event. Thus, in addition to screening data collected during periods when ozone is shut down, ozone data is screened from event detection analysis while BAC filter(s) are in BW. Data screening is performed with the following lines of code shown in Figure 4-9.

```
elif event_process == 'Ozone':

    time_filter_process1 = pm.data[tag_bac1_status] == 1
    pm.add_time_filter(time_filter_process1)
    time_filter_process2 = pm.data[tag_bac2_status] == 1
    pm.add_time_filter(time_filter_process2)
```

Figure 4-9. Ozone Data Screening Code.

The code above elucidates the simplicity of configuration and application that the Pecos package provides—only 4 lines of code are required to execute the data screening of BAC BW data. A complete copy of the Python code that was implemented at the NCPWDF can be found in Appendix D.

Table 4-9 details the integer value assigned to the various statuses of the ozone system via the status tag. Ozone system data logged during periods when ozone is in any status other than “2” is screened and thus not included in analysis by the EDS.

Table 4-9. Ozone Data Screening Description.

	Description
Ozone Process Status	Integer value used to indicate the ozone process status: • 0 = Off • 1 = Local On • 2 = Remote On • 3 = Standby • 4 = Manual Purge

When the ozone system is called to run, the generator ramps up production to meet an operator set dose in PPD. After the start-up sequence the ozone system will operate to maintain a constant ozone dose. When the ozone operations mode is switched from “OFF” to “REMOTE ON”, there is a brief period in which there is process water flow, but ozone is not being dosed. This does not constitute a process failure since it is an expected part of the start-up process of the ozone system and is accounted for in the plant-wide status data screening delay timer discussed in Chapter 4.7.

4.3.3 Ozone Data Flagging and Event Detection

For each of the unit processes, the project team identified the common failures (i.e., events) that could occur. These events were prioritized for onsite implementation at the NCPWDF because they are expected to occur at a higher frequency than other issues based on historical operations and expertise. Table 4-10 below summarizes the event criteria specific to the ozone process designed to detect potential failures of the process, monitoring points, and pathogen removal goals. Additional ozone event detection logic for further development and optimization of the EDS is outlined in Appendix C.1.

Table 4-10. Ozone Events Included in Deployed Event Detection Tool.

<table>
<tr><th>Event</th><th>Tags</th><th>Description</th></tr>
<tr><td colspan="3">Process Failure Events</td></tr>
<tr><td>Ozone Generator Failure</td><td>• Ozone generator production
• Ozone production set point</td><td>The percent difference between the ozone generator’s production SP and actual PV is greater than a maximum threshold indicating failure of generator to meet the required ozone gas production.</td></tr>
<tr><td colspan="3">Monitoring Point Events</td></tr>
<tr><td>Ozone Meter Drift at OSP 4</td><td>• Dissolved ozone measured at OSP 4 primary meter
• Dissolved ozone measured at OSP 4 redundant meter</td><td rowspan="2">The primary online meter is known to drift below the actual ozone residual if not calibrated regularly, while the redundant meter at the same location exhibits more stable readings. Percent difference between the meters is used to characterize a failure event indicating that the primary meter needs maintenance or calibration.</td></tr>
<tr><td>Ozone Meter Drift at OSP 7</td><td>• Dissolved ozone measured at OSP 7 primary meter
• Dissolved ozone measured at OSP 7 redundant meter</td></tr>
<tr><td colspan="3">Water Quality Event</td></tr>
<tr><td>High Ozone Demand</td><td>• Ozone demand
• OSP 4 primary meter
• OSP 4 redundant meter[1]
• Ozone generator production
• Ozone production SP</td><td>This event detects an increase in the ozone demand caused by changing water quality. OSP 4 ozone residual monitoring and ozone generator production are first verified as reporting within expected operating range. Thus, the probable cause of the increase in ozone demand can be attributed to a change in feed water quality, and not the result of process or monitoring point failures.</td></tr>
</table>

[1]While redundant meter is non-operational, the calculated rolling average is used.

4.3.3.1 Ozone Process Failure

An ozone process failure results from the ozone generation system failing to meet the ozone dose specified by the operator. Although the ozone generator and its affiliated components can be compromised or need maintenance for a variety of reasons, the event detection framework is focused on events with the potential to compromise pathogen removal goals. When ozone dose is inadequate, the corresponding CT will be too low to achieve the ozone process LRV goals. For this reason, detecting a declining trend in ozone production before absolute failure occurs will enhance DPR system reliability. To detect this failure, a calculated tag was created within the SCADA and PLC of the NCPWDF that continuously determined the percent difference between the ozone production set point (SP) and production value (PV). The difference between the ozone generator PV and SP was observed to be consistently less than 3% during normal operations based on historic data. The event criterion for an ozone process failure was configured using a Pecos range test. If the percent difference between SP and PV increases above 5% for longer than 15 minutes (15 consecutive data points in the SQL logger), the EDS generates an alert for an ozone process failure. While the occurrence of this is unlikely, this event criteria was included in the deployed EDS to validate the detection of all three types of failure within each unit process of the DPR train. The results of the challenge test for this event are found in Section 5.2.1.1 below.

4.3.3.2 Ozone Monitoring Point Failure

Data that is being inaccurately reported by problematic meters within the ozone system will compromise the ability of the operations staff to evaluate the treatment performance of the ozone system, and specifically the pathogen LRVs. Therefore, early detection of a loss in meter accuracy is an important feature of the event detection framework that provides increased response time for meter calibrations or other service actions. This ensures that continuous treatment performance is maintained.

The main source of meter error is the gradual loss of meter sensitivity due to fouling of the inline probe sensor. This is especially true in DPR treatment trains where ozone is the first unit process in the advanced treatment train, and thus is receiving a constantly changing wastewater matrix. When dissolved ozone meters lose sensitivity, the meter will read lower ozone residuals resulting in incorrectly low LRV values. Low meter values are not always indicative of false meter readings since water quality changes can also cause low meter values. When low meter values are due to water quality changes, a dose adjustment can increase meter values to meet the LRV treatment objective. If meter fouling has occurred, dose adjustment may not rectify the low meter values. Thus, the event detection criteria were designed so that a loss of meter sensitivity within the ozone system is detected from a comparison between the primary and the reference meter at the same OSP instead of basing the event solely on low ozone residual measurements. The comparison is calculated using the percent difference calculation shown below (Equation 4-14), with the more stable reference meter as the value that the primary meter is compared against.

$$\%\ \text{Difference between OSP meters} = \left|\frac{(\text{primary meter} - \text{redundant meter})}{\text{redundant meter}}\right| \quad \text{(Equation 4-14)}$$

Monitoring point failure is detected by applying a range test with an upper bound of 15% to the calculated percent difference values at OSP 4 and 7. In full scale ozone system monitoring, each OSP will be equipped with a reference meter and thus this approach should be applied to all metered OSPs. The threshold for detection is specific to the monitoring location and instrument, since the propensity for a given meter to drift can vary between manufacturers and is influenced by the process being measured. At the NCPWDF, operator experience and analysis of historical data found that the primary and reference meter operate within 10% of each other if each meter is calibrated and maintained adequately. For this reason, a 15% threshold for 15 minutes was used as an early indicator that the primary meter is drifting from the reference meter. This test configuration provides early event detection without generating false alarms. The reason the reference meter is not used as the primary meter is because it measures ozone residual every 2.5 minutes via colorimetric analysis, whereas the primary meter is an inline amperometric probe which measures the dissolved ozone concentration every second. The EDS processes data points every 60 seconds, and thus the inline probe is the preferred data source for LRV monitoring. Once the meters differ by 20%, the operations staff must bring the meter back to accuracy immediately because this would be considered a CCP failure where the primary meter's readings are no longer accurate.

A historical example used to inform the ozone monitoring point event configuration occurred on June 3rd, 2015, when the primary meter at OSP 4 drifted out of calibration. Figure 4-10 demonstrates the efficacy of the 15% bound for the early detection of meter drift using this historic example. The green line at 6:20pm indicates when the EDS would have alerted operations of the drift exceeding 15%. The red line at 8:22PM is where the drift begins to exceed 20%. The data series of both the primary and the reference meter is included (shown in orange and blue respectively), as well as the calculated percent difference between the two (shown in grey).

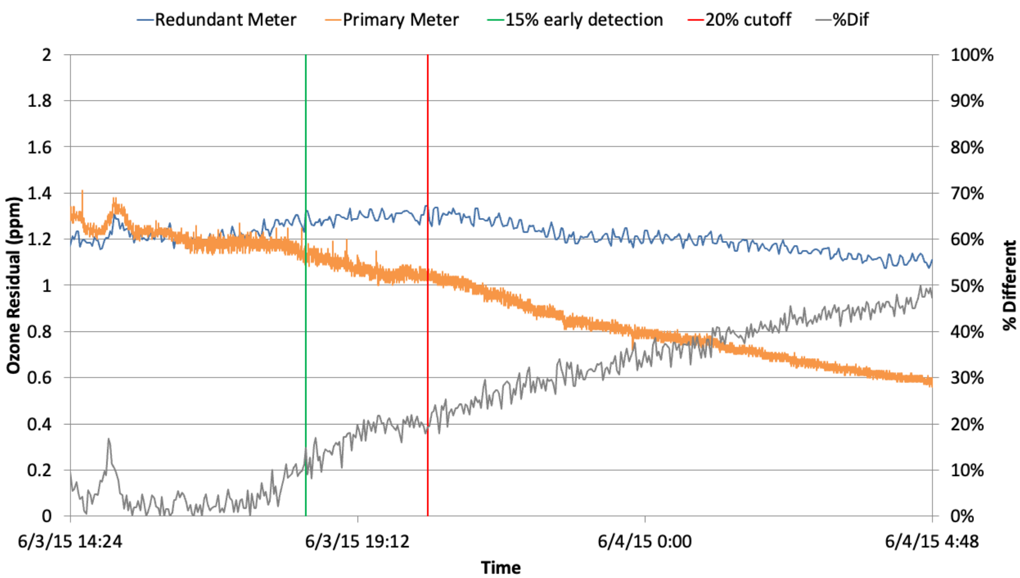


Figure 4-10. Historic Data Example of Ozone Residual Meter Drift at OSP4.

The EDS would have provided operations staff with approximately 2-hours of lead time in this historical example. Because a full-scale DPR facility's HRT is on the order of several hours, the addition of 2 hours of response time is beneficial to preventing off specification product water and/or loss of production. The precise amount of lead time provided will vary between failure occurrences based on the rate of primary meter drift, but the benefit of early detection remains the same.

4.3.3.3 Ozone Water Quality Event

Although upstream WRP processes provide pretreatment for the AWPF, the water quality is constantly changing as it exits the WRP and enters the ozonation process. Water quality alterations can lead to the ozone system's operations not meeting treatment needs, in which case, the ozone dose should be increased to meet the increased ozone demand in the water. As detailed above, *Cryptosporidium* LRV is the primary CCP monitoring parameter of the ozone system. The system must always provide at least 1-log removal regardless of changes in feed water quality. The EDS identifies water quality changes that could potentially compromise ozone treatment goals before the 1-log removal minimum is crossed.

The ozone demand was selected as the indicator of potentially compromising water quality changes. Ozone demand requires fewer intermediate calculations than the LRV calculation making it the preferred CCP monitoring parameter because there are less inputs that influence the measurement. To preemptively detect a water quality event, the ozone demand value is flagged when outside of an upper threshold using a range test. The upper threshold decided upon for preemptive detection of water quality events in the prototype deployment at NCPWDF was 6.5 ppm at a 15-minute minimum failure duration (15 consecutive data points in the database logger). This upper bound was informed by historic statistics and iteratively adapted using operational observations. Ozone demand is calculated (as detailed above in section 4.3.4.2) as the difference in ozone applied dose and the concentration at the first monitoring location (OSP 4), so the calculated demand value is expected to differ due to water quality changes primarily. Therefore, the statistical method provides useful insight for the event detection bound since the influencing variables that could be prone to errors are minimized.

This logic for ozone water quality event detection may be consistently applied across all DPR facilities, yet site specific bound determination is needed for full-scale deployment as each feed water will exhibit varying ozone demand.

To confidently detect a water quality event with increases in demand, the values being used to determine demand must first be checked for their accuracy. The water quality event detection logic first confirms that the ozone generator is producing within 5% of the setpoint dose. The OSP 4 primary meter is also checked to ensure it is reporting within 15% of the reference meter. Verification of meter accuracy and adequate process operation performance is accomplished with range tests on the OSP 4 meter and the ozone generator percent difference tags. OSP 4 is the only ozone residual monitoring location that needs to be checked for accuracy since it is directly included in the calculation of demand. These range tests will flag continuously when the system is operating as expected so the presence of these flags represents normal operations.

After the EDS confirms that there are no process failures or monitoring point events actively occurring, an increase in demand above the threshold of 6.5 ppm for greater than 15 consecutive minutes can be identified as a water quality event. Similar conditional logic for detecting water quality changes, by first checking that monitoring and process measurements are within expected ranges, can be applied to other CCP monitoring parameters such as *Cryptosporidium* LRV.

The conditional, multi-parameter framework for detecting changes in ozone water quality is visually depicted in Figure 4-11. The event detection logic was designed this way to provide guidance for the necessary corrective action and avoid false alarms in the event that high ozone demand is being caused by issues unrelated to feed water quality.

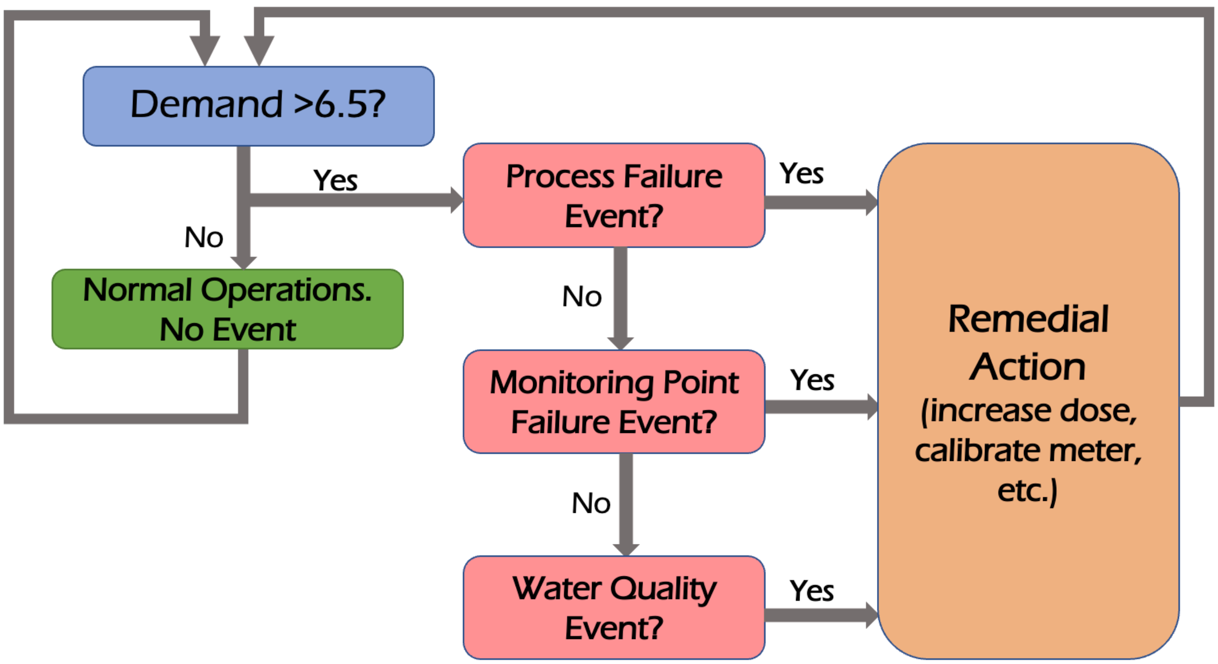


Figure 4-11. High Demand – Ozone Water Quality Event Detection Logic.

The above referenced range test thresholds are configurable and are subject to change based on the ozone feed water quality at each DPR facility. The key takeaway is the logical framework that achieves early detection and provides operations staff with insight can reduce response time.

The lines of code shown in Figure 4-12 contain the three range tests that the event is configured with, as well as the range test for the event tag. The code shown gives an example of the syntax of a multi-condition event detection framework, and the resultant event tag.

```
        elif eventid == 15:

            pm.check_range(key=name_ozone_wq1, bound=[None, 0], min_failures=15)
            pm.check_range(key=tag_dict[91], bound=[0.15, None], min_failures=2)
            pm.check_range(key=tag_dict[92], bound=[0.15, None], min_failures=2)
            pm.check_range(key=tag_dict[86], bound=[0.05, None], min_failures=2)
            pm.check_range(key=tag_dict[59], bound=[None, 6.5], min_failures=15)
```

Figure 4-12. Ozone Water Quality Event Detection Code.

4.4 Microfiltration Configuration

4.4.1 MF Data Selection

The following discussion outlines how pathogen removal is determined through the microfiltration (MF) process and thus what CCP monitoring parameters are necessary to include in the EDS.

A membrane integrity test (MIT) is conducted daily to directly assess membrane integrity. During a MIT, the MF system is first drained and then pressurized with air to a pressure specified in the MIT procedure. The pressure drop is measured over a 5-minute period and compared to the baseline pressure drop for the intact skid to evaluate if there are breaches or damaged membrane fibers. The baseline pressure drop for MF membranes is product-specific and requires demonstration testing prior to operation.

Pathogen LRV is calculated using the product-specific correlation between pressure drop and pathogen LRV. At least 4-log removal of *Cryptosporidium* and *Giardia* should be demonstrated per treatment objectives from the State and Federal Long Term 2 Enhanced Surface Water Treatment Rule (U.S. EPA 2010).

California's Division of Drinking Water (DDW) requires that integrity testing of the MF system at the NCPWDF demonstrate the ability to detect a 3-µm hole in the membranes using daily MITs. The pressure decay rate and water temperature are measured during the daily MIT and are used to calculate a pathogen LRV using the Equation 4-15 below:

$$\mathrm{LRV} = \log\left(\frac{\mathrm{Q_p \times ALCR \times P_{atm}}}{\mathrm{\Delta P_{test} \times V_{sys} \times VCF}}\right) \qquad \text{(Equation 4-15)}$$

where: Q_p = design capacity filtrate flowrate (gpm)
ALCR = air-liquid conversion ratio
P_{atm} = atmospheric pressure (psia)
ΔP_{test} = manufacturer-provided rate of pressure decay associated with a 3-µm membrane breach (psi/min)
V_{sys} = volume of system pressurized with air (gal)
VCF = volumetric concentration factor

During operation, several terms in the formula above can be estimated using continuous measurements of other parameters. The ALCR can be estimated by measuring water temperature and transmembrane pressure (TMP). This means a continuous LRV can be calculated to indirectly monitor for indications of membrane integrity issues, but a daily MIT must still be performed to directly validate membrane integrity.

Direct integrity testing using pressure decay must result in at least 4-log reduction or the membrane is considered to have failed the MIT and the required pathogen control is not being met. To meet the expected minimum LRV at this facility, the ΔP_{test} term can be back calculated using the system-specific variables to determine the upper control limit (UCL) for the pressure decay rate.

Unlike the ozone process where pathogen LRV is continuously measured for regulatory compliance, the MF process reports only a daily pathogen LRV using the pressure decay rate measured by the MIT. Due to the shortened retention time for DPR projects, a daily assessment of MF process performance is too infrequent to detect sudden changes in pathogen removal before inadequately treated water leaves the facility. Filtrate turbidity is a continuously measured parameter used as an indirect measurement of membrane integrity. For this reason, it is considered the primary and only CCP monitoring parameter for the MF process as outlined in Table 4-11 and shown in the schematic below (Figure 4-13).

Table 4-11. MF Process CCP Monitoring Parameters.

Parameter	Units	Description
MF Filtrate Turbidity	NTU	• Used to assess process and meter functionality

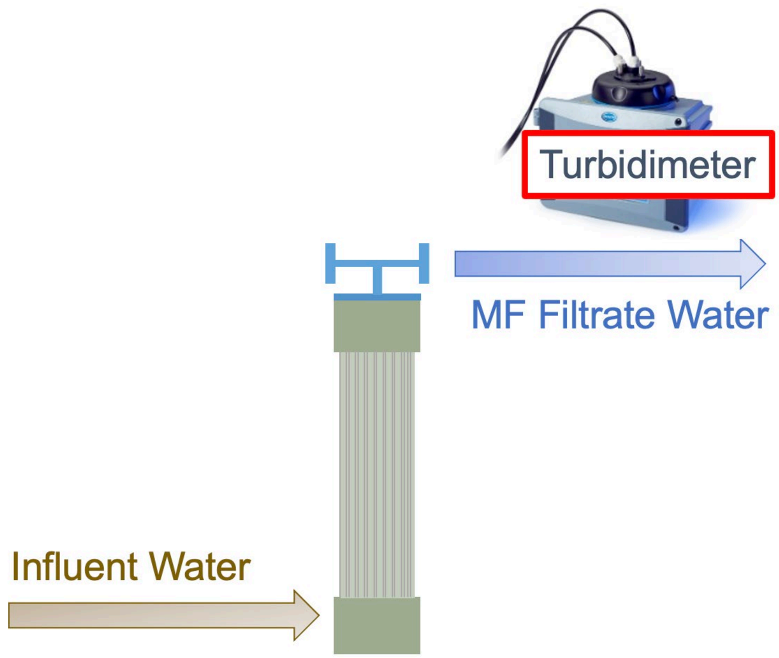


Figure 4-13. Schematic Overview of an MF Module.

Another reason that MF filtrate turbidity was considered the only CCP monitoring parameter necessary for inclusion in the EDS is that it is the primary continuously monitored MF parameter that is subject to regulatory requirements. The California State Water Resources Control Board's Regulations Related to Recycled Water for membrane filtration processes requires that filtrate turbidity not exceed 0.2 nephelometric turbidity units (NTU) for more than 5% of the time in a 24-hour period and always be below 0.5 NTU (SWRCB, 2018). For a 24-hour period, 5% of the time equates to 72 minutes, which means filtrate turbidity cannot exceed 0.2 NTU for more than 72 minutes on any single day of continuous operation. The Membrane Filtration Guidance Manual requires a direct integrity test (i.e., MIT) to be performed if filtrate turbidity exceeds 0.15 NTU for more than 15 minutes (U.S. EPA 2005). This will be a typical requirement for full-scale California potable reuse projects.

4.4.2 MF Data Screening

Once the MF system is called to run, a production cycle is initiated. A production cycle consists of a period of production where filtrate is generated and is followed by a BW to remove the accumulated particles that have been rejected by the filter fibers. The duration of each production cycle is approximately 30 minutes. Data from periods where the system is backwashing, performing integrity tests, or undergoing chemical cleans (i.e., CIPs) are not representative of system performance. The EDS does not consider data recorded during these times.

The MF system records the process status as integer values corresponding to manufacturer-defined descriptions. The recorded process status data is used to identify periods of time when the data should be ignored or removed. For MF, the only time the data should be analyzed is when the process status is 'Forward Flow' (or status = 1). Table 4-12 summarizes the MF process status tags.

Table 4-12. MF Data Screening Tags.

Status Tag	Description
MF Process Status	Integer value used to indicate the state of MF: • 0 = Off • 1 = Forward Flow • 2 = Reverse Filtration • 3 = Air Scrub • 4 = Strainer BW • 5 = MIT • 6 = CIP • 7 = Flush • 8 = Enhanced Flux Maintenance (EFM) • 9 = Fill • 10 = Paused

The Python script for the EDS was configured with a time filter for the MF process so that data is analyzed only from when the status tag value equals 1 (production mode). The lines of code below (Figure 4-14) provide details as to how this was implemented at the NCPWDF.

```
if event_process == 'MF':
        time_filter_process1 = pm.data[tag_mf_status] == 1
        pm.add_time_filter(time_filter_process1)
```

Figure 4-14. MF Data Screening Code.

4.4.3 MF Data Flagging and Event Detection

The pathogen LRV for the MF process is directly monitored once daily using a MIT. The MIT results are easy to interpret and trends in pressure decay that approach regulatory thresholds are likely to be noticed and corrected by plant operations. However, daily MIT results are too infrequent to provide advanced notice of compromised treatment performance. Filtrate turbidity is an online measurement that can be used as an indirect monitor for membrane integrity. Although filtrate turbidity monitoring is generally not considered by regulators to be sufficient for verifying pathogen LRV on its own, it is an ideal CCP monitoring parameter for real-time event detection when used alongside daily MITs because it can be monitored continuously. This enables changes in process performance to be tracked on the scale of minutes and hours, rather than days, which is essential for DPR monitoring. Table 4-13 below summarizes events defined for the MF process.

Table 4-13. MF Events.

Event	Tags	Description
Process Failure Events		
Sustained High Filtrate Turbidity	MF Filtrate Turbidity	Turbidity exceeds threshold of 0.15 NTU for at least 15 consecutive minutes (per U.S. EPA 2005)
Monitoring Point Events		
Stagnant Data	MF Filtrate Turbidity	MF Filtrate Turbidity is monitored for erroneous readings or if the turbidimeter is accidentally left in hold

Since the goal of this monitoring system is to detect events prior to exceeding regulatory thresholds for shutdown or diversion, the first MF process event employs the Membrane Filtration Guidance Manual limit of 0.15 NTU for more than 15 minutes as the basis for filtrate turbidity monitoring (U.S. EPA 2005). With this approach, the EDS notifies operations in real time that a direct integrity test is needed and provides advanced notice that MF filtrate is trending toward an exceedance of the 0.2 NTU regulatory threshold that should not occur more than 5% of the time. The 0.5 NTU regulatory threshold already has a high-high (HH) alarm programmed into the NCPWDF SCADA system to trigger a shutdown if exceeded, so it is not the focus for the design of this monitoring system.

The second event for detecting a stagnant MF filtrate turbidimeter is a common example of a monitoring point error where the measurement being reported may not be representative of the current conditions. When turbidimeters undergo maintenance, it is standard practice for the operator to hold the measurement output at a constant value while performing the maintenance to avoid triggering any alarms related to turbidity. Although some turbidimeter models contain a “time-out” feature that will return the meter to measurement mode if it is left in hold for a specified time. The detection logic of this EDS is relevant for meter models that do not have this functionality. Furthermore, turbidimeters may report stagnant values due to other unforeseen reasons, including communication losses for example. If the turbidimeter is not returned to the measurement mode once maintenance is completed, the static filtrate turbidity value being reported could mislead operators into assuming that system performance is normal. The stagnant data test is applied to MF filtrate turbidity. The parameters for the stagnant data test are the stagnancy threshold and minimum number of consecutive failures. The stagnancy threshold was determined based on the typical variation in historical data, and the minimum number of consecutive failures was specified based on how often variation typically occurred in the historical data. Alternatively, redundant turbidimeters could be implemented with a stagnant data test monitoring the calculated meter difference between a primary and reference meter. This approach would be similar to the detection of an ozone monitoring point failure described in Section 4.3.3.2 using Equation 4-14.

A water quality event for the MF process was not configured since it was reasoned that water quality changes with the potential to impact the MF process would be detected upstream in the ozone process first. The MF process acts as a physical barrier to particulates and pathogens, and as discussed, its proper functioning is assessed using MITs and filtrate turbidity measurements. Water quality changes would not impact the filtrate turbidity or MIT results. Thus, water quality changes would likely not compromise the membranes' acute treatment goals, and are rather related to process optimization (i.e., flux) and cleaning requirements. For these reasons, in the absence of an ozone water quality event, it is expected that the most probable source of declining trends in MF process performance would be due to a process failure or monitoring point error.

4.5 Reverse Osmosis Configuration

4.5.1 RO Data Selection

Pathogen LRVs for the RO system are determined based on the LRV of surrogate parameters such as TOC or electrical conductivity (EC). TOC and EC indirectly monitor system integrity via online analyzers located on the RO feed and permeate lines provide continuous measurements of each parameter (Figure 4-15).

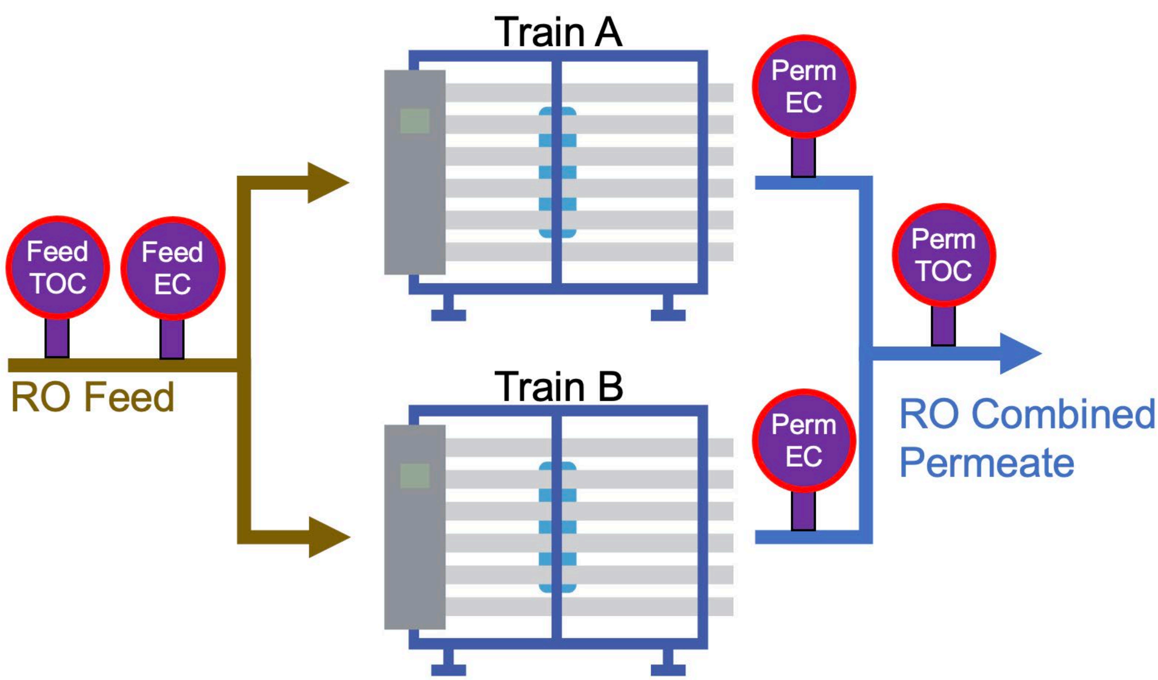


Figure 4-15. NCPWDF RO System Online EC and TOC Monitoring Locations.

Permeate TOC is measured at a combined line from Train A and B whereas EC analyzers are located on the individual Train A and B permeate lines. When implementing the EDS, only Train A was considered for simplicity, but the same event detection logic is also applicable to Train B. Generally, RO achieves greater than 1.5-log removal of TOC and greater than 1-log removal of EC, so the preferred surrogate for pathogen log reduction is TOC. If TOC monitoring data is unavailable, EC is used as the surrogate for pathogen LRV. The LRV of surrogate parameters used for determining pathogen removal through RO is calculated using the formula below.

$$\mathrm{LRV} = \log\left(\frac{C_{feed}}{C_{permeate}}\right) \qquad \text{(Equation 4-16)}$$

Where: C_{feed} = feed concentration of surrogate parameter
$C_{permeate}$ = permeate concentration of surrogate parameter

The feed and permeate concentrations for TOC and EC were the selected CCP monitoring parameters for the RO process because they directly link to pathogen LRV performance and are influenced by process upsets such as chemical spikes and membrane breaches. Table 4-14 describes how these parameters are used to assess RO membrane performance.

Table 4-14. RO Process CCP Monitoring Parameters.

Parameter	Units	Importance to System Monitoring
RO Feed TOC	µg/L	Analytical measurement used to calculate RO TOC removal
RO Combined Permeate TOC	µg/L	Analytical measurement used to: • Calculate RO TOC removal • Monitor permeate water quality to meet permeate TOC regulatory requirements • Detect events such as a membrane breach or organic chemical spike
RO TOC Removal	LRV	Calculated log-reduction value: • From feed and combined permeate TOC measurements • Used as the primary surrogate for pathogen LRV
RO Feed EC	µS/cm	Analytical measurement used to calculate RO EC removal
RO Train A permeate EC	µS/cm	Analytical measurement used to: • Calculate RO EC removal • Monitor permeate water quality to detect a membrane breach
RO EC removal	LRV	Calculated log-reduction value: • From feed and combined permeate EC measurements • Used as the secondary surrogate for pathogen LRV

4.5.2 RO Data Screening

When RO is called to run, the feed pumps first ramp up to a fixed initial speed for a duration specified by the operator. After the duration at the initial speed is met, pumps will continue to ramp up to meet product water flow setpoints. Once product water setpoints are met, the trains are considered to be “in production” which is represented by status number 4 “Normal PID mode” in Table 4-15.

Full-scale facilities usually have a “permeate to waste” step to confirm water quality before permeate is allowed to continue on to the next downstream process. At the NCPWDF, the permeate produced during this start-up period proceeds directly to the downstream UV process and there is no filter to waste diversion. Thus, there is a period of time in which water quality meters are recording data during start-up and water quality is not confirmed before permeate travels to the next downstream process. A "start-up" counter is employed to ignore data measured during the typical duration of the start-up process. This is accounted for by the plant wide status data tag discussed in Chapter 4.7.

Table 4-15. RO Data Screening Description.

	Description
RO Process Status	Integer value used to indicate the state of RO: • 0 = Sequence is reset • 1 = Standby • 2 = Inlet Valves Opening • 3 = Pump in Steady Ramp Mode • 4 = Normal PID Mode • 5 = Ramp Down Mode • 6 = Inlet Valves Closing • 7 = Flush Valves Opening • 8 = Permeate Flush Running • 9 = Flush Pump Stopping • 10 = Flush Valves Closing • 11 = Sequence Resetting

4.5.3 RO Events

Pathogen LRVs for the RO system are determined based on the LRV of TOC or EC. Since the LRVs of TOC and EC are calculated from measurements taken in the RO feed and permeate, there are several operating scenarios that can impact the calculated LRV. These scenarios include membrane failure, meter error, or a change in water quality. Each of these scenarios is summarized in Table 4-16 below and discussed in more detail in the following subsections.

Table 4-16. RO Events.

Event	Tags	Description
Process Failure Events		
Membrane Breach	• Permeate TOC • Train A (or B) Permeate EC	A membrane breach is a treatment failure caused by O-ring failures, glue line leaks, or physical membrane damage. A breach is indicated by increased EC and TOC in the RO permeate above historical levels. Train A and Train B are tested separately for membrane breaches.
Monitoring Point Events		
Potential Feed TOC Meter Drift (High)	• Feed TOC • RO TOC Removal	For typical water quality, the TOC removal (LRV) is expected to remain relatively constant. Thus, if the feed TOC increases, it is expected the permeate TOC would also increase such that the TOC LRV remains relatively constant. If an increase in feed TOC above historical levels is observed while the TOC removal (LRV) also increases above historical levels, then it is possible that the feed TOC meter is drifting high. This event would prompt an operator to check the feed TOC meter calibration.
Water Quality Events		
Organic Chemical Peak	• Permeate TOC • Train A (or B) Permeate EC	The presence of elevated levels of small, low molecular weight organic chemicals (e.g., acetone) in the feed water jeopardizes an RO train's ability to meet permeate TOC requirements due to the ability of these chemicals to more readily pass through RO membranes. This event can be detected by monitoring combined permeate TOC and permeate EC. If high concentrations of TOC are observed in the permeate while permeate EC is observed to be at normal levels, the cause is potentially an organic chemical peak. The normal permeate EC rules out the high TOC being caused by a membrane breach.

4.5.3.1 RO Process Failure Event

A process failure for RO is defined as a failure of the RO membranes to perform their designed function. The function of the RO membranes is to reject salts and other impurities to produce a permeate stream with low concentrations of ions and TOC. Thus, permeate TOC and EC are measured continuously to monitor RO performance.

A membrane failure can be caused by O-ring failures, glue line leaks, or damage to the membrane surface. In all these cases, feed water bypasses the RO membrane and mixes with the permeate stream. Both TOC and EC in the permeate stream will increase since feed water bypassing the membranes will have very high concentrations of TOC and ions relative to the permeate water. The event detection criteria for the permeate TOC and EC were based on historical concentrations recorded during operator rounds from the online TOC analyzers over a one-year period. Operator rounds are performed when the plant has ready stable operations and can be understood as representative. The normal operating ranges for permeate TOC and EC were defined using data collected in 2021. The feed TOC data consists of 125 recorded values spanning 7 months of operation. The permeate TOC data consists of 141 recorded values spanning 12 months of operation.

A standard deviation approach was used for determining the feed TOC range test bounds since the variation in the data is not expected to be influenced by startup operations. A combined percentile and quartile approach was used for permeate TOC because the data set contained a significant number of outliers that are believed to be caused by sampling after system startup when the RO system was not yet stable. The combined percentile and interquartile range approach was established in DRPT-4991 for detection of high permeate TOC above baseline concentrations (Debroux 2021). The 5th percentile from the data set was calculated to be 14 µg/L, which was used as the lower bound for the normal operating range. The upper bound (UB) for the normal operating range was calculated to be 50 µg/L using an interquartile range approach, which is detailed in Equation 4-17 below.

$$\mathrm{UB} = \mathrm{Q3} + (1.5 \times \mathrm{IQR}) \qquad \text{(Equation 4-17)}$$

Where: Q3 = 3rd quartile (or 75th percentile)
IQR = interquartile range = Q3 – Q1
Q1 = 1st quartile (or 25th percentile)

The quartile approach is commonly used to exclude the impact of outliers on the determination of an acceptable range of data. Since the demonstration facility lacks a “filter to waste” step to recirculate permeate during start-up, it was expected that the permeate TOC data will have outliers present during start-up. If a standard deviation approach were used, any significant outliers would impact the normal operating range. Using Equation 4-17 with historical data spanning approximately 2 years of operation, the upper bound for the normal operating range was calculated to be 50 µg/L.

The normal operating range for TOC removal was calculated using the same data set of recorded values, but only when feed and permeate values were recorded in pairs. The sample set consisted of 98 sample pairs spanning 7 months of operation. The upper and lower bounds were calculated using the 95th and 5th percentiles, respectively.

The typical permeate EC is defined by comparing the permeate EC at the beginning and end of life for a set of RO membranes that was used at the NCPWDF for 3 years of operation. Over the life of the membranes, the permeate EC gradually increased from 25 µS/cm up to 125 µS/cm while the feed EC was relatively constant in the range of 1300-1600 µS/cm. The gradual increase in permeate EC over the life of the membrane is expected due to slow oxidation of the RO membranes by the residual disinfectant that is maintained to minimize biological growth. In addition, loss in salt rejection can also be attributed to repeated CIP events over time. The Python scripts used in this project for the EDS were designed to be configurable through minor edits in the code, therefore the operating range test bounds can be narrowed and adjusted as the RO membranes age to provide more precise detection of anomalous EC measurements. Table 4-17 summarizes the operating range test bounds that were determined for detecting events in the RO process.

Table 4-17. Operating Range Test Bounds for the RO Process.

	Median	(Min, Max)	Basis for Min/Max
RO Feed TOC (µg/L)	4800	(3800, 5800)	± 2 standard deviations
RO Permeate TOC (µg/L)	29	(14, 50)	Max = Q3 + 1.5*IQR Min = 5th percentile
RO Train A and B Permeate EC	65	(25, 125)	Max = Maximum EC observed at end of membrane life (after 3 years of operation) Min = Minimum EC observed following new membrane installation
TOC Removal (LRV)	2.2	(2.1,2.6)	Max = 95th percentile Min = 5th percentile

4.5.3.2 RO Monitoring Point Events

A monitoring point failure for RO is defined as a failure of the instrumentation to report real or accurate data. Instrument failures can include times when the meters report stagnant values, values outside of the analytical range of the instrument, or when meters exhibit symptoms of meter drift. Symptoms of meter drift can be site-specific and depend on the monitoring system design. Feed TOC meter drift was selected to be the proof-of-concept example for an RO monitoring point event because the variable water quality it receives in the feed was observed to result in higher maintenance frequency at the NCPWDF.

For TOC meter drift, the symptoms are defined using the measured TOC concentrations and the calculated TOC removal values. While the TOC concentrations in the RO feed are expected to vary, the TOC removal (or rejection) across the RO membranes is expected to stay relatively constant under normal water quality conditions. If an increase in both the feed TOC and LRV above historical levels is observed while the permeate TOC remains within the expected operating range, then it is possible that the feed TOC meter is drifting high. An alert for this event by the EDS would prompt an operator to check the feed TOC meter calibration.

4.5.3.3 RO Water Quality Event

Detecting a water quality event in the RO process requires monitoring permeate TOC and EC. Chemical peaks as discussed in section 1.3 are cause for concern in DPR treatment trains because they have been found to be poorly rejected (Debroux 2021). RO permeate TOC is the monitoring parameter that would best detect the presence of a chemical peak; therefore, it is the focus of the RO water quality event detection logic. The RO permeate TOC and EC upper operating bounds outlined in Table 4-16 of the RO process failure section are used to determine if these CCP monitoring parameters are abnormally high. To distinguish between whether a chemical peak or membrane breach is causing the increase in RO permeate TOC, permeate EC must be within its normal operating range for the EDS to generate an alert for a water quality event.

Based on real-life occurrences, it is expected that the RO permeate TOC timeseries data would resemble a bell curve distribution in the event of a chemical peak. The abrupt change test was a type of Pecos quality control test initially considered for detecting chemical peaks by analyzing the slope of the RO permeate TOC, which is typically very flat (i.e., stable). The hypothesis was that a sharp spike in RO permeate TOC would be flagged by the abrupt change test which can be configured to have a maximum threshold for the difference between two data points within a moving window. However, this approach was not implemented in the final version of the EDS because it was more difficult to configure for the desired sensitivity than a range test. Chemical peak events tend to last on the order of hours or days, so detecting 15 minutes of consecutive values above 50 µg/L through a range test is very likely to provide enough time to respond to a potential chemical peak event before the regulatory limit of 500 µg/L is exceeded.

4.6 UV/AOP Configuration

4.6.1 UV/AOP Data Selection

The influent stream to the UV reactors is RO permeate that has been dosed with sodium hypochlorite. Free chlorine is measured to confirm adequate sodium hypochlorite dosing to meet the AOP criteria. For redundancy, influent free chlorine is measured in two ways: by an individual free chlorine meter and the calculated difference between two total chlorine analyzers located before and after the sodium hypochlorite injection point. The free chlorine meter is the primary meter for monitoring oxidant dosing because it measures more frequently than the total chlorine analyzers. The total chlorine concentration prior to the addition of sodium hypochlorite is measured in the RO permeate to monitor for downstream effects on UVT in the UV/AOP system.

The CCP monitoring parameters included for analysis by the EDS are essential for ensuring compliance with the regulatory requirements for obtaining 6 logs of pathogen LRV credit. UVT and UV dose are regulated parameters measured inside the UV reactors to monitor water quality and ensure that pathogen inactivation and AOP performance criteria are achieved. A minimum oxidant dose for AOP is a permit requirement for reuse facilities in California (State Water Board 2021). This facility's permit stipulates a minimum 1.0 mg/L sodium hypochlorite dose. Table 4-18 describes each of the CCP monitoring parameters used to assess UV/AOP process performance.

Table 4-18. UV/AOP Process CCP Monitoring Parameters.

Parameter	Units	Importance to System Monitoring
UV Dose	mJ/cm^2	• Calculated parameter used to assess process functionality and determine if pathogen inactivation and AOP requirements are met
UV Intensity (UVI)	mW/cm^2	• Used to assess monitoring point functionality • Used to calculate UV dose

Parameter	Units	Importance to System Monitoring
Feed UV Transmittance (UVT)	%	• Used to assess UV/AOP feed water quality • Indicates amount of UV light that is available for disinfection/photolysis/AOP
RO Permeate Total Chlorine	mg/L	• Total chlorine measured prior to the oxidant injection point • Used to assess water quality and oxidant dosing
UV Feed Total Chlorine	mg/L	• Total chlorine measured following the oxidant injection point • Used to assess oxidant dosing
UV Feed Free Chlorine	mg/L	• Free chlorine measured following the oxidant injection point • Used to assess oxidant dosing
Pathogen Removal	LRV	• UV/AOP receives 6.0 LRV for viruses, *Giardia*, *Cryptosporidium* if UV dose exceeds minimum regulatory requirement and treatment conditions are within operating envelope

Figure 4-16 provides an overview of where the UV/AOP CCP monitoring parameters are measured within the system.

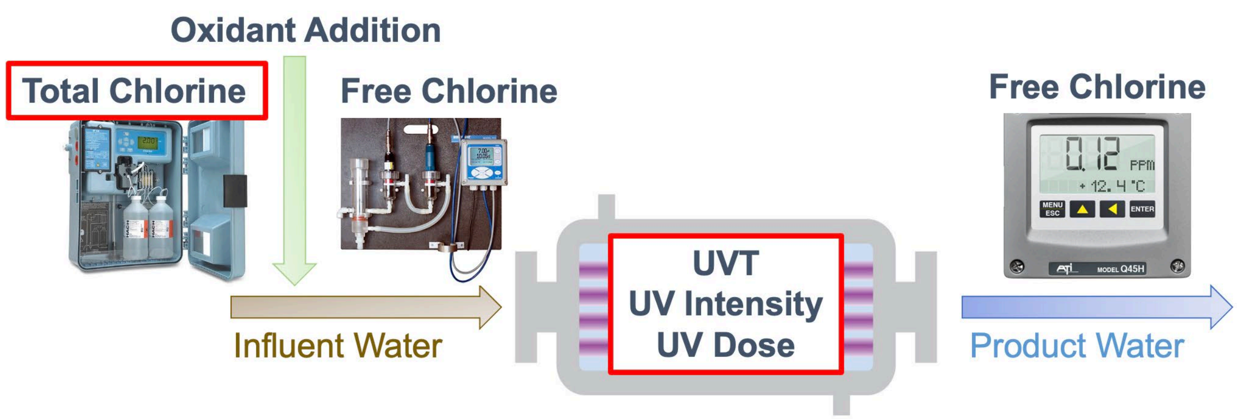


Figure 4-16. UV/AOP CCP Monitoring Parameter Locations Within the UV System.

4.6.2 UV/AOP Data Screening

When UV/AOP system is restarted following a shutdown, the UV lamps are turned on and the ballast power level (BPL) runs at 100% for a warming period of 5 minutes. Once the warm-up period is over, the manufacturer's control algorithm adjusts the BPL to control UV dose and achieve 0.5 log removal of 1,4-dioxane. Similar to the RO system, the UV effluent produced during the UV system's warming period is not used for product water and instead routed to the NCPWDF waste stream. The monitoring system accounts for the warming period by filtering out any data generated during this time. This prevents nuisance alarms from being generated by only analyzing data when UV-treated water is being sent to downstream processes.

4.6.3 UV/AOP Events

The criteria for detecting the UV/AOP process events listed in Table 4-19 was based on the CCP monitoring parameters identified in Table 4-18 above. An event is categorized as a process failure, monitoring point issue, or water quality event depending on the measurement(s) being flagged.

Table 4-19. UV/AOP Events.

Event	Tags	Description
Process Failure Events		
Low UV Dose	UV Dose	UV dose is approaching minimum required dose for pathogen credit. Operator should check the status of the UV lamps in the reactor and replace failed lamps if needed.
Monitoring Point Events		
Stagnant Data	UV Intensity (UVI) Sensor	UVI data is monitored for erroneous reporting of stagnant values. Operator should perform maintenance on the UVI sensor if needed.
Water Quality Events		
Low UV Feed UVT	UV Feed UVT	UV Feed UVT is approaching the minimum regulatory requirement for UVT. Potential causes could be an upset in an upstream process, the oxidant dosing system is not working properly, or the UV lamp sleeves need to be cleaned.
High UV Feed Chloramines	RO Permeate Total Chlorine	RO Permeate Total Chlorine exceeds the typical operating threshold. Operator should check RO chloramine dosing system.

4.6.3.1 UV/AOP Process Failure Event

The primary indicator of a UV/AOP process failure event is declining UV dose. Regulatory requirements for UV dose differ based on the treatment objective. The minimum UV dose to meet AOP criteria (greater than 0.5 log removal of 1,4-dioxane) is 850 mJ/cm^2 while the minimum dose to obtain 6.0 logs of pathogen inactivation is 278 mJ/cm^2. A process failure event alert is generated if the UV dose approaches the minimum dose for pathogen inactivation because the process is at risk of losing pathogen removal credits and being out of regulatory compliance. A UV dose value of 300 mJ/cm^2 was determined to be an appropriate threshold for providing advanced warning of the UV/AOP process trending toward a process failure.

4.6.3.2 UV/AOP Monitoring Point Event

Stagnant readings from an instrument indicate the occurrence of a monitoring point error where process performance measurements are potentially inaccurate. One example of this event type within UV/AOP is the UVI sensor. Since UVI is used by the system PLC to calculate the UV dose, loss of this measurement could lead to inadequate UV dosing. If the UVI reading does not change by more than 0.01 for 30 consecutive minutes, then the stagnant UVI sensor alert is generated. The minimum change value of 0.01 was determined iteratively after observing that values less than 0.01 were too sensitive thereby generating false alarms while values greater than 0.01 were not sensitive enough.

4.6.3.3 UV/AOP Water Quality Event

Worsening UV/AOP influent water quality caused by issues upstream would primarily be reflected by a decrease in UVT. Low UVT in UV/AOP feed water results in reduced treatment performance due to increased interference of UV light from being used for disinfection, photolysis, and AOP. For AOP and pathogen inactivation criteria to be met, the UVT must be at least 95.0%. A lower bound of 96.0% was implemented to prompt an alert when UVT is decreasing towards the regulatory minimum of 95.0%.

The presence of high chloramine concentrations in the UV feed will absorb UV radiation and interfere with UV/AOP treatment efficacy. An upstream issue with the chloramine dosing system prior to MF that is used for RO membrane fouling control would be indicated by high chlorine levels in the RO permeate (i.e., UV feed). If the RO permeate total chlorine is measured to be greater than the target chloramine concentration of 1.0 mg/L, a water quality event alert for UV/AOP is generated.

4.7 Plant Wide Configuration

In addition to process specific screening discussed in the section above, process data across the entire train is also screened from event detection analysis when the plant wide status is not "plant stable." The conditions for "plant stable" are as follows:

- NCPWDF "start" has been triggered
- Ozone generator enters ON status = "2"
- MF (or UF) enters ON status = integer other than "0"
- RO Train A or Train B enters "ON" status = "4"
- UV running status = "1"

Once the "plant stable" conditions are satisfied, the delay timer of the data screening function of the EDS starts and counts down an operator set time period (0-60 minutes) before beginning to process operations data for event detection purposes. This offset timer allows the processes to reach stable and characteristic operating conditions to avoid false event detection during the startup of unit processes.

4.8 Event Notification

A key component of the EDS is an output mechanism that notifies operations staff of potential events and provides visualization of the data. The implemented monitoring system at the NCPWDF presents the results of the 12 configured events detailed below (Figure 4-17) on a dashboard to display whether the events occurred across the facility's unit processes during the specified time period. The dashboard was developed using the Pecos library's built-in visualization features and is generated as an .html file on the HMI computer where the event detection script is installed. High performance graphics to notify operations staff of any event occurrences using a magenta color to indicate a potential event while grey indicates that the event is not occurring. The live mode of the EDS allows for continuous analysis of the system and automatically updates the dashboard every five minutes. When using the historic mode of the system, the dashboard will reflect the analysis results for the manually set timeframe.

Figure 4-17 provides a dashboard example from the EDS run in historical mode over an 80-minute period on 12/20/22 as it would appear to the operations staff. The dashboard contains a matrix where each row corresponds to one of the four CCPs at the facility, and each column corresponds to one of the three event types. The example dashboard pictured indicates the detection of five potential events during the monitoring period including ozone and RO monitoring point failures, water quality events at the RO and UV/AOP processes, and an ozone process failure event detected during challenge testing, which is described in Chapter 5. The date that the dashboard was generated in the bottom left corner is helpful in differentiating the results between the detection system's live and historic modes.

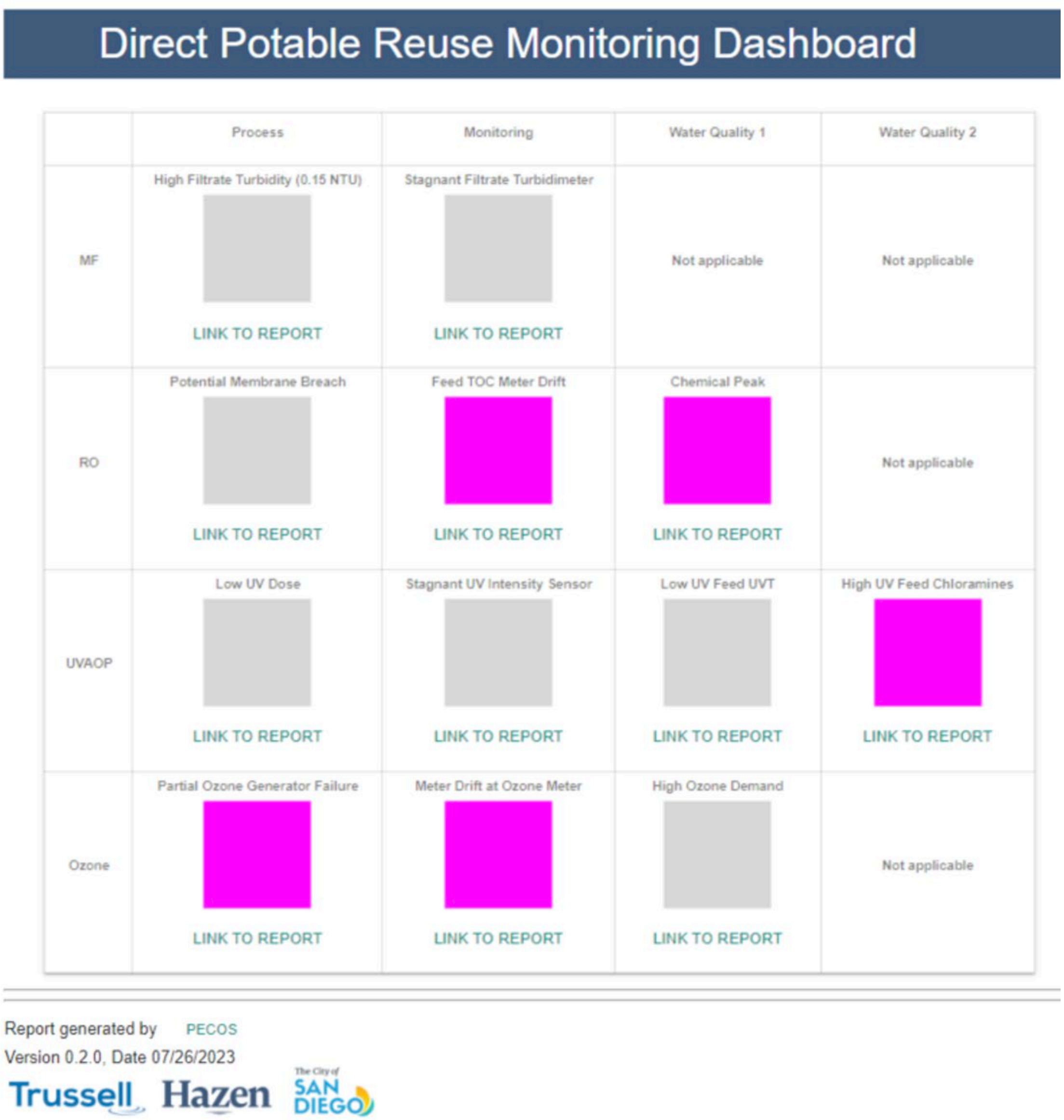


Figure 4-17. Example EDS Dashboard Output.

In addition to being notified of event occurrences, the dashboard provides operators access to visuals of the process data from a potential event and details about the data analysis performed by the EDS. Data visualization is necessary to provide insight that can be used to develop a targeted response to any potential events. Operations staff may click on the “LINK TO REPORT" below an individual event’s alert square to access the event’s report.

A complete report generated when an event has been detected include the following components as labeled in Figure 4-18:

a. The event title
b. The start and end time of the monitoring period
c. Time series data plot(s) of the CCP monitoring parameter(s) included in the event criteria
d. A tabulated list of the times during the monitoring period that the event occurred
e. Time series data plot(s) of the CCP monitoring parameter(s) included in the event criteria with screened data labeled with gray columns and flagged data marked with blue crosses

Figure 4-18. Example EDS Event Report Output.

The example presented above is for the RO feed TOC meter drift event. The purpose of this output is to convey the contents of an event report and further details on challenge testing results are discussed in Chapter 5. The test results table lists the tags that were flagged during the reporting period, the flagging duration, and the minimum or maximum threshold configured for each respective Pecos quality control test. The legend for each flagged data plot includes the Pecos quality control test type and threshold value that triggered the flag.

The report for events containing logic that requires multiple criteria to be true for the event to be detected includes data time series of each individual criterion. In Figure 4-18, the plots for RO feed TOC and LRV represent the individual event criteria. The event detection compiles the results from these criteria into a single integer tag representing the event (e.g., RO monitoring event) known as the "event tag." When the event has occurred for a duration longer than the minimum number of consecutive failures value, the event tag will flag, and the detection system will identify the event. When an event only requires flagging a single tag, an event tag is not required since aggregating multiple test results into a single tag is not necessary (see section 5.2.1.1 for an example).

A report will be generated regardless of an event occurrence. If no event is occurring, only the raw time series data will be displayed and the test results table and data plot with test flags will not be included in the report.

4.9 Framework Implementation Summary

The implementation of the four-step framework—data storage, data screening, data flagging, and event detection—across each CCP of the DPR treatment train at NCPWDF required using an iterative workflow to:

- Identify the CCP monitoring parameters relevant to early detection of potential losses in treatment for pathogen removal, including the addition of calculated parameters
- Develop site-specific Pecos quality control test inputs (e.g., operating range bounds and minimum number of consecutive failures for detection) for range and stagnant data tests
- Configure the event detection logic for process failure, monitoring point, and water quality events using the Pecos quality control tests as criteria
- Design and develop monitoring reports which provide useful insight and alerts regarding event detection

The project team selected 21 CCP monitoring parameters from the thousands of available data sources to target likely failure scenarios, while narrowing the data storage and analysis required. The site-specific Pecos quality control test inputs for these CCP monitoring parameters were determined through statistical methods, operational knowledge, and iterative trial and error (detailed further in Chapter 5) depending on the unit process and variability. Since each CCP has unique failure mechanisms based on the method of treatment used to remove pathogens (i.e., chemical oxidation vs. membrane filtration), the event detection criteria for each event varied across CCPs within the treatment train. Ultimately, the workflow yielded an EDS with site-specific event detection abilities using broadly applicable event detection logic.

In Chapter 5, the functionality of the implemented software, event detection logic, and Python script configuration described in this chapter are challenge tested using either natural or artificial data sets at the NCPWDF.

CHAPTER 5

Implementation Testing and Validation

In addition to the development of a conceptual framework for event detection, the project team deployed and tested the EDS to continuously monitor process data at the NCPWDF. Once the EDS was implemented for early event detection across the NCPWDF treatment train, the 12 strategically selected events detailed in Chapter 4 were either simulated or an actual occurrence was captured onsite to confirm that the event detection logic and system functioned as expected. Most of the events necessitated a simulated failure to test detection capabilities because the treatment train is robust, and failures are rarely observed during operations. By capturing each mechanism of failure across each of the NCPWDF unit processes, a holistic assessment of the event detection system's ability to detect events of high consequence and/or importance across the DPR treatment train was conducted. This approach was needed for challenge testing because the application of the Pecos test(s) to each process, and each type of failure mode within that process, are unique.

5.1 Testing Methodology

Since naturally occurring CCP failure events at the NCPWDF rarely occur, it was necessary to simulate potential failures to validate the functionality of the EDS. Simulated challenge tests were conducted for most events, but actual events that took place while the event detection system was implemented are also included and discussed in this Chapter when available.

Many iterations of the EDS Python script existed over the course of development. The challenge testing discussed below began when a working version of the Python script was implemented onsite at the NCPWDF in December 2022 and continued until July 2023. During this time period, iterative adjustments were made to various aspects of the EDS such as the configuration of event detection logic, Pecos test parameters, and data visualization based on the challenge testing results.The adjustments aimed to facilitate detection of failure events by the EDS as early as possible while avoiding a nuisance amount of false detections.

5.2 Testing Results

For each CCP at the NCPWDF, a representative event for each of the three potential event types (process failure, monitoring point failure, and water quality event) was either simulated by the project team or captured in real plant data. The presented results include context surrounding the event's occurrence, the visual outputs generated by the EDS at the time of the event, and a discussion of the extent to which the event was successfully identified by the event detection system.

The dashboard displays a magenta square when an event has occurred during the specified timeframe that the EDS has been configured to analyze. When the active event report link is selected, the report will appear containing the timeframe of analysis. The event report is included for each challenge test discussed below and raw data plots are omitted for more concise communication of the results.

5.2.1 Ozone Event Detection Challenge Testing

Due to its location at the beginning of the NCPWDF advanced treatment train, the ozone process receives variable water quality that has the highest concentration of contaminants. Variations in ozone demand occur frequently due to fluctuating concentrations of readily oxidizable compounds in the ozone feed water. Therefore, ozone generation requirements are in a constant state of flux to meet the varied demand. Furthermore, accurately measuring the dissolved ozone residual proves to be challenging in a tertiary filtered matrix. For these reasons, examples of monitoring point and water quality events specific to the ozone process took place during normal operations and were used to confirm that the EDS functioned as expected. Individual event criteria were subsequently configured appropriately. An ozone process failure did not occur at the NCPWDF during the testing period, so an artificial challenge test was conducted to simulate an ozone generator failure. Included below is discussion of each of the challenge tests and the outcomes.

5.2.1.1 Process Failure

The simulation of an ozone generator failure required manipulating the ozone generator setpoint to report a static value in the PLC of 100 ppd while simultaneously reducing the actual ozone production to 90 ppd. These actions caused the calculated tag monitoring the percent difference between ozone production setpoint and actual ozone production to increase past the maximum threshold of 5%. For a timespan of 18 minutes, the calculated difference remained greater than 5% causing the EDS to generate an alert for an ozone generator process failure. In addition to the sustained exceedance, there are two instances leading up to the challenge test when the calculated ozone production difference increased above 5% but recovered below the maximum threshold soon after—encircled in Figure 5-1. These increases occur following production SP adjustments when the PV is changing to meet the new SP. The EDS is configured to generate an event alert only if the ozone production difference is greater than 5% for 15 consecutive minutes to avoid false positives when adjusting ozone production. Therefore, the challenge test was successful in proving that the EDS was accurately applying the event criteria parameters for identifying an ozone process failure with the desired sensitivity. The output report generated by Pecos for this event on June 30, 2023, is shown below in Figure 5-1 where the event lasting 18 minutes can be observed as flagged and detected.

Partial Ozone Generator Failure

Data start time: 2023-06-30 13:25:00
Data end time: 2023-06-30 14:24:00
Number of variables: 5
Number of test failures: 1

Test Results:

	Variable Name	Start Time	End Time	Timesteps	Error Flag
1	Ozone Production Error (decimal)	2023-06-30 14:03:00	2023-06-30 14:20:00	18	Data > upper bound, 0.05

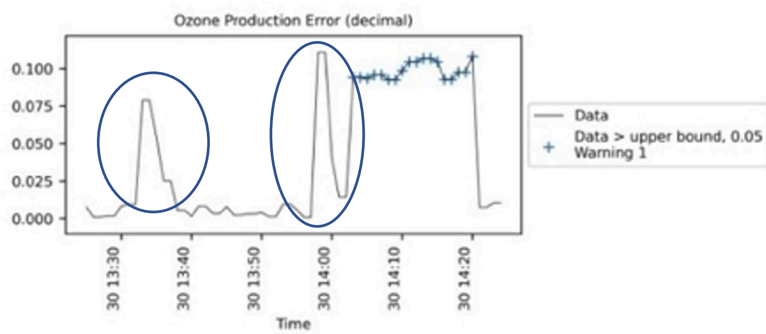


Figure 5-1. Partial Ozone Generator Failure Event Detection Report.

5.2.1.2 Monitoring Point

As described in section 4.3.3.2, ozone residual meters within a wastewater matrix have a high propensity to drift and lose calibration. Thus, the EDS was configured to detect a meter drift monitoring point failure in the ozone process. Preemptive detection of meter drift in an ozone system is paramount for DPR facilities to ensure that the LRV calculation is consistently accurate, so detecting measurement inaccuracies before they become critical is the goal of the EDS. To reiterate, ozone monitoring locations have a primary and a reference meter; should the difference (as a percentage) between the reference and primary meter's values read greater than the threshold for early event detection (15%), the data will be flagged by the EDS. After 15 minutes of consecutive flags, the event will be detected, and operations will receive an alert.

Validation of the detection system's logic did not need synthetic challenge testing because ozone meter drift events occur frequently if the primary meter at OSP 4 and/or OSP 7 is not regularly calibrated every 1-3 days. To ensure quality data for testing, the primary and reference meters were both calibrated at the same time on the first date of challenge testing. Each proceeding day, only the redundant meter was calibrated if necessary while the primary meter was left to drift from calibration. Based on operational experience, it is expected that the meter would drift within a few days to meet the criteria for detecting this event.

In the 10-hour period displayed by the output shown in Figures 5-2, ozone monitoring point failures at OSP 4 and 7 were detected. The data from August 31, 2023, indicates that the difference between the primary and reference ozone meters at OSP 4 and 7 became consistently greater than 15%, signaling that calibration is needed for the primary meter. The first event occurred at 1:00 PM at OSP 7 followed by drift at OSP 4 at 2:33 PM as indicated by the green and red circles. The circles correspond to the boxes highlighted in the test results table. The test results table gives the exact start and end times of detection. This test data is a clear example of how the early detection framework can provide added lead time for operators to respond to failures. The gradual loss of ozone meter accuracy is observable in Figure 5-2 as the percent difference value between the reference and primary meters increases at both sampling points, eventually crossing the critical threshold of 20%. In this case, four hours of lead time would be provided between the 15% detection threshold and the 20% failure threshold to address the meter issue at OSP 7, and three hours of lead time would be provided at OSP 4. Furthermore, when action needs to be taken, this event detection output gives clear direction to the operations staff that both OSP 4 and 7 are the monitoring locations in need of calibration, further reducing the response time necessary to address the event.

Meter Drift at Ozone Meter

Data start time: 2023-08-31 11:00:00
Data end time: 2023-08-31 18:59:00
Number of variables: 7
Number of test failures: 6

Test Results:

	Variable Name	Start Time	End Time	Timesteps	Error Flag
1	Ozone Monitoring Event	2023-08-31 13:00:00	2023-08-31 18:59:00	360	Data > upper bound, 0
2	Ozone Normalized Difference (Across Meters) OSP 4 (decimal)	2023-08-31 14:33:00	2023-08-31 15:14:00	42	Data > upper bound, 0.15
3	Ozone Normalized Difference (Across Meters) OSP 4 (decimal)	2023-08-31 15:17:00	2023-08-31 17:24:00	128	Data > upper bound, 0.15
4	Ozone Normalized Difference (Across Meters) OSP 4 (decimal)	2023-08-31 17:27:00	2023-08-31 18:59:00	93	Data > upper bound, 0.15
5	Ozone Normalized Difference (Across Meters) OSP 7 (decimal)	2023-08-31 13:00:00	2023-08-31 15:42:00	163	Data > upper bound, 0.15
6	Ozone Normalized Difference (Across Meters) OSP 7 (decimal)	2023-08-31 16:57:00	2023-08-31 18:59:00	123	Data > upper bound, 0.15

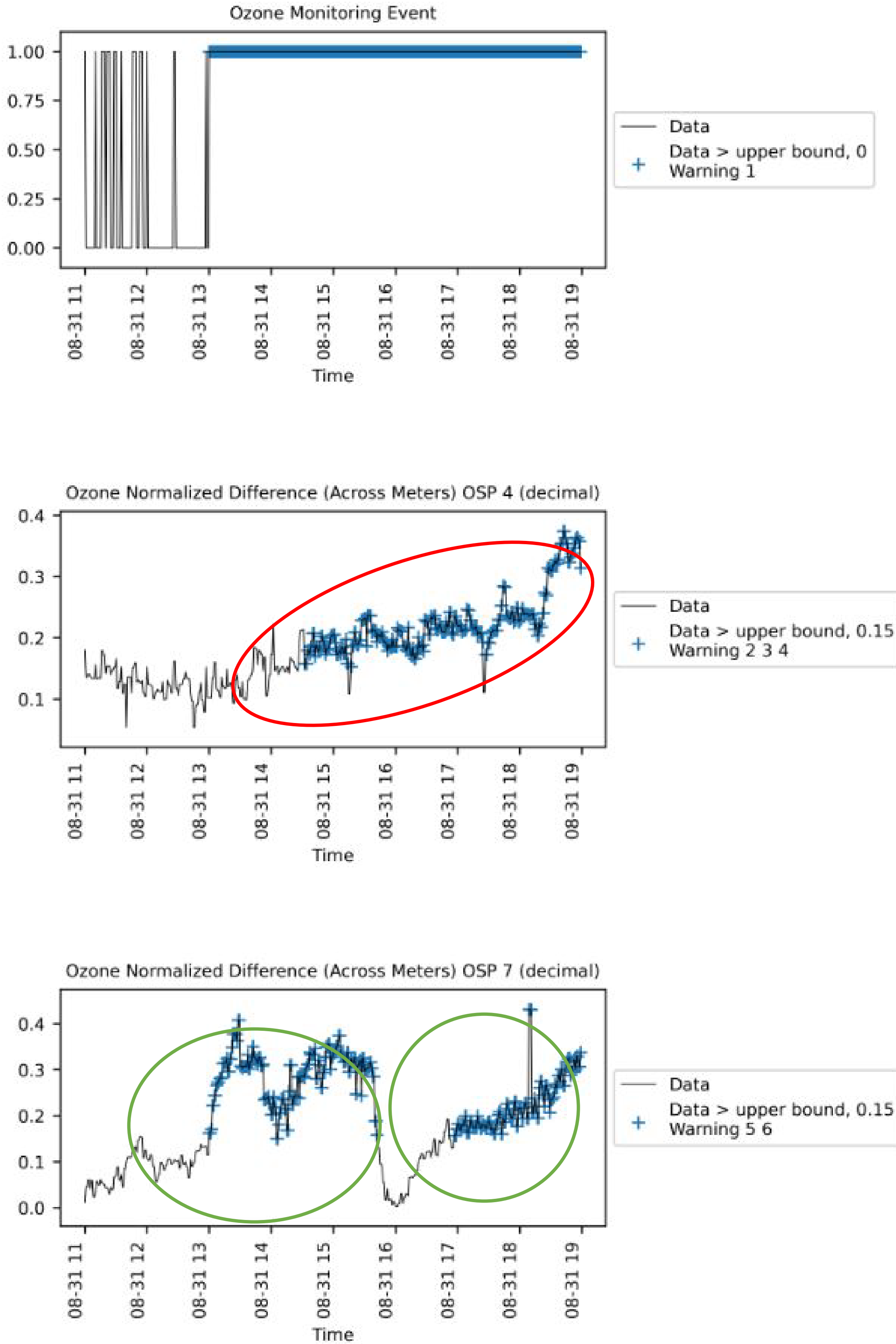


Figure 5-2. OSP 4 and 7 Ozone Monitoring Point Failures Event Report.
Data flagging boxed in green and red in the test results table respectively correspond to the green and red circles on the data plot.

5.2.1.3 Water Quality

A CCP failure event at the ozone system caused by water quality changes can be detected early only if the CCP monitoring parameters that are monitoring the water quality changes are first verified as accurate. See section 4.3.3.3 above for detailed logic as to how reduction of the LRV due to water quality changes can be detected. The event configuration included in the prototype EDS uses ozone demand as the CCP monitoring parameter that indicates LRV changes. This event logic was chosen for implementation to the detection system and for challenge testing because it contains multiple tags that all must be flagged simultaneously for the event to be detected. This challenge test validated the script's architecture of 'and' logic between data tags for event detection using an event tag.

Similar to the monitoring point event, this challenge data is not synthetic but came from real operations. Due to upstream construction at the NCWRP, flows were changed within unit processes at the WRP, and an upset occurred in the secondary biological process. The upset led to alterations in the water quality of the feed water to the ozone process and increased the ozone demand. The demand increased beyond the 6.5 mg/L threshold on the evening of June 30th, 2023. The event was successfully detected since the monitoring point and ozone generator were both validated as operating properly. During this testing period, the reference meter at OSP 4 was out of service, and thus a rolling average of the previous 120 minutes of data from the remaining in-service meter was used for the % difference comparison in place of the second meter. Both approaches operate in the context of detecting changes to verify that the meter is reading properly. If the meter were to drift, the instantaneous value would begin to differ from the rolling average, indicating a meter failure.

The output report from the elevated demand event is included in Figure 5-3 below. Although the time series plots for the generator production error data and the monitoring point error data both appear to indicate failure, when noting the axis scale it becomes clear that the data indicates no process or monitoring point failures. The report shows that the generator was producing within 5% of the setpoint and that the primary meter at OSP 4 was reading within 15% of the reference value. Therefore, the increase in demand is verified as driven by changes in water quality. The ozone water quality event was correctly not detected via the event tag until the upper demand threshold was crossed, although monitoring point and process checks are flagging continuously. This verifies that the conditional "and" logic for detection was functioning as expected.

High Ozone Demand

Data start time: 2023-06-30 15:30:00
Data end time: 2023-07-01 13:59:00
Number of variables: 8
Number of test failures: 6

Test Results:

	Variable Name	Start Time	End Time	Timesteps	Error Flag
1	Ozone Demand Hach Meter (mg/L)	2023-06-30 22:45:00	2023-06-30 23:11:00	27	Data > upper bound, 6.5
2	Ozone Demand Hach Meter (mg/L)	2023-06-30 23:14:00	2023-07-01 13:59:00	885	Data > upper bound, 6.5
3	Ozone Normalized Difference (Inst vs Rolling) OSP 4 Hach (decimal)	2023-06-30 15:30:00	2023-07-01 13:59:00	1349	Data < lower bound, 0.15
4	Ozone Production Error (decimal)	2023-06-30 15:30:00	2023-07-01 13:59:00	1349	Data < lower bound, 0.05
5	Ozone Water Quality Event	2023-06-30 22:45:00	2023-06-30 23:11:00	27	Data > upper bound, 0
6	Ozone Water Quality Event	2023-06-30 23:14:00	2023-07-01 13:59:00	885	Data > upper bound, 0

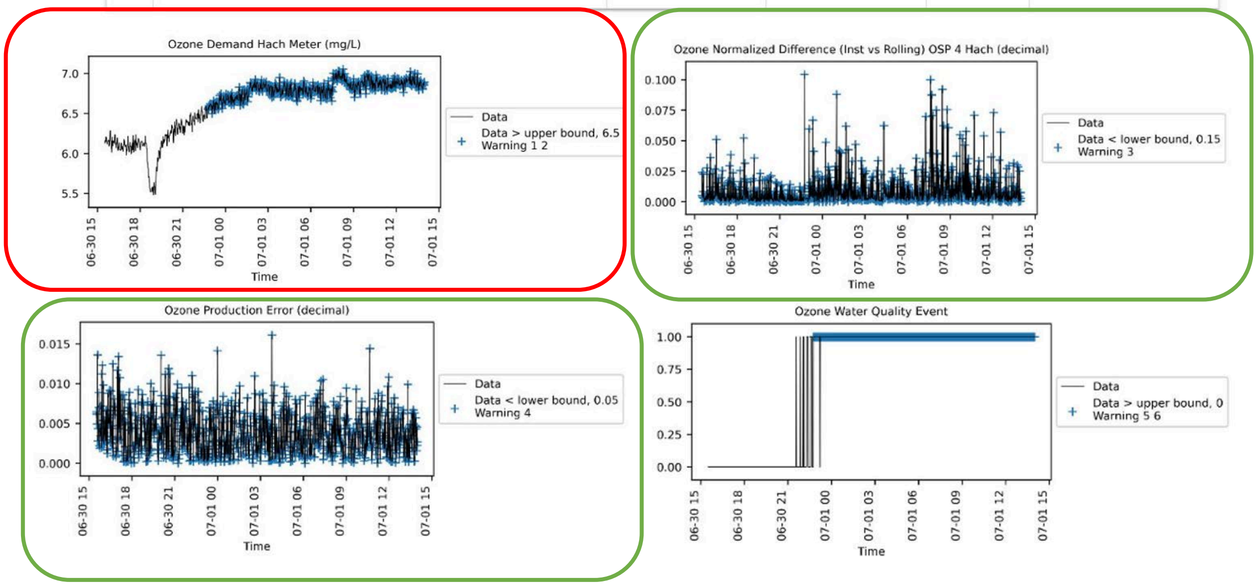


Figure 5-3. High Ozone Demand Water Quality Event Detection Report.
The Event Tag shown in bottom right; the process and monitoring point checks are boxed in green; the CCP failure shown by the increase in ozone demand is boxed in red.

The event detection logic of using CCP monitoring parameters to preemptively detect changes to the LRV, while simultaneously using the calculated percent difference tags to verify accurate system monitoring is sound. When properly configured, this approach will not only provide lead time to prevent CCP failure from occurring, but also reduce response time due to the specificity of information given about the event.

Included in Figure 5-4 is the *Crypto* LRV during this failure event. It can be noted LRV decreases to less than 1.0 during this testing period's data—shown with the red "X". Yet this CCP failure occurred before the demand increased above 6.5 mg/L to trigger event detection—shown with a yellow "X". Consequently, this event's detection configuration needs further refinement.

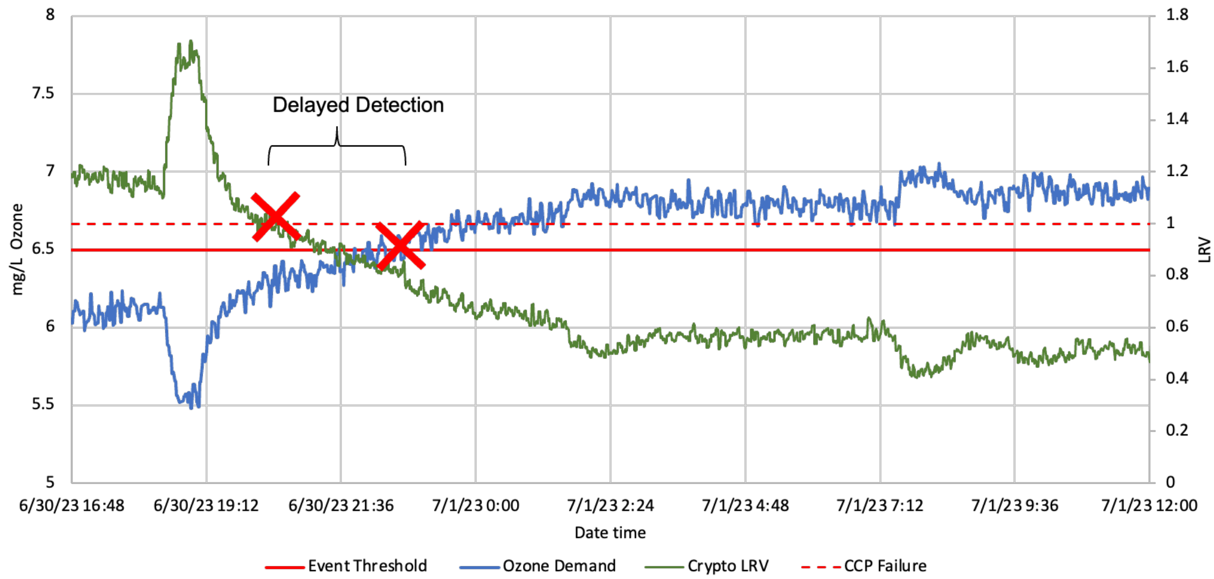


Figure 5-4. Ozone Demand and Crypto LRV During the June 30th Water Quality Event Detection.
The current configuration requires additional refinement to adequately detect CCP failure preemptively as noted by the delay in event detection.

To detect this event preemptively, the demand threshold would need to be reduced to 6.0PPM. However, based on historic data, the median of demand is equal to 6.05PPM and the standard deviation is equal to 0.7. Thus, reducing the threshold to detect the event in time to prevent CCP failure would result in many nuisance alarms for operations staff. So, although the correlation between LRV and demand is clear, in this particular event the LRV response to demand changes were not directly proportional, so additional CCP monitoring parameters should be investigated for ozone water quality event detection. Further developments of the event detection logic should incorporate direct LRV monitoring as described in Appendix C, Chapter C.2.

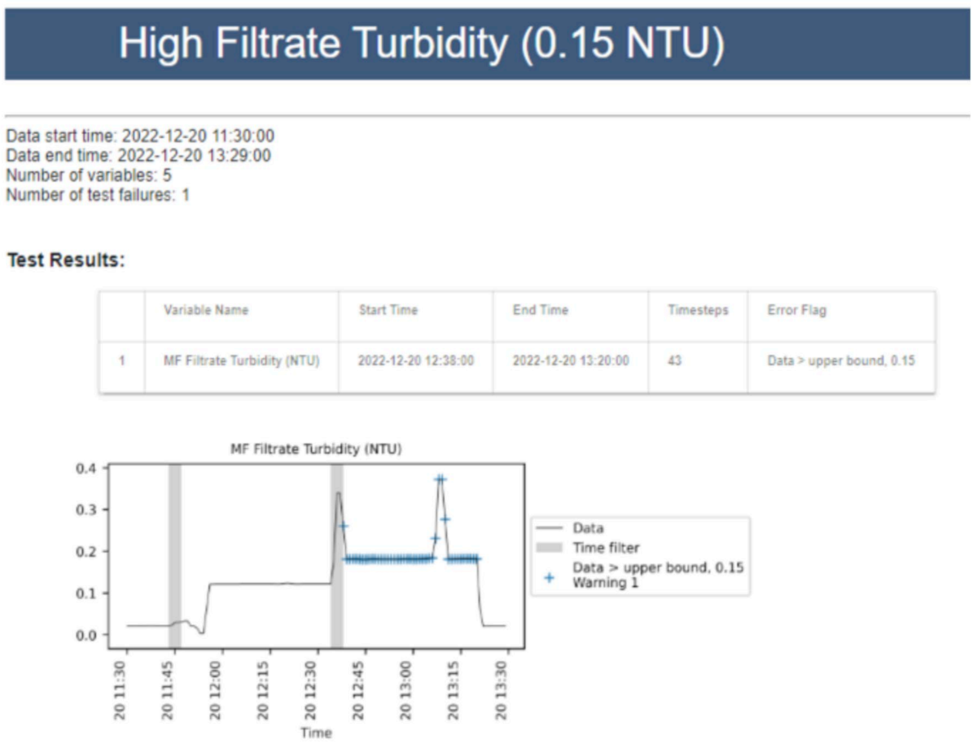
5.2.2 MF Event Detection Challenge Testing

Out of the various CCP unit processes in the NCPWDF treatment train, MF required the simplest monitoring configuration for understanding process performance. Filtrate turbidity is the only parameter used by the EDS to identify events because it serves as an indirect measurement of the MF membranes' integrity and is required in regulatory compliance for claiming pathogen removal credits. During the timeframe that the EDS was being developed and implemented, the MF process did not experience broken membrane fibers or turbidimeter errors. To challenge test the EDS filtrate turbidity readings were manually manipulated to simulate potential MF events. During normal operations a serious membrane breach would lead to changes in the filtrate turbidity.

5.2.2.1 Process Failure

One of the primary indicators that the integrity of the MF membranes has been compromised is a sustained high filtrate turbidity reading. To simulate this occurrence, the MF filtrate turbidity was increased to a constant value of 0.18 for 30 continuous minutes within the data logger. This value exceeds the 0.15 NTU regulatory limit established by the Membrane Filtration Guidance Manual (U.S. EPA 2005).

The process maintained adequate treatment throughout the testing period, but the value being recorded by the plant data logger was synthetically elevated. The resulting monitoring report output shown in Figure 5-5 confirms that the EDS was successful in identifying the change in MF filtrate turbidity as an MF process failure event.


High Filtrate Turbidity (0.15 NTU)

Data start time: 2022-12-20 11:30:00
Data end time: 2022-12-20 13:29:00
Number of variables: 5
Number of test failures: 1

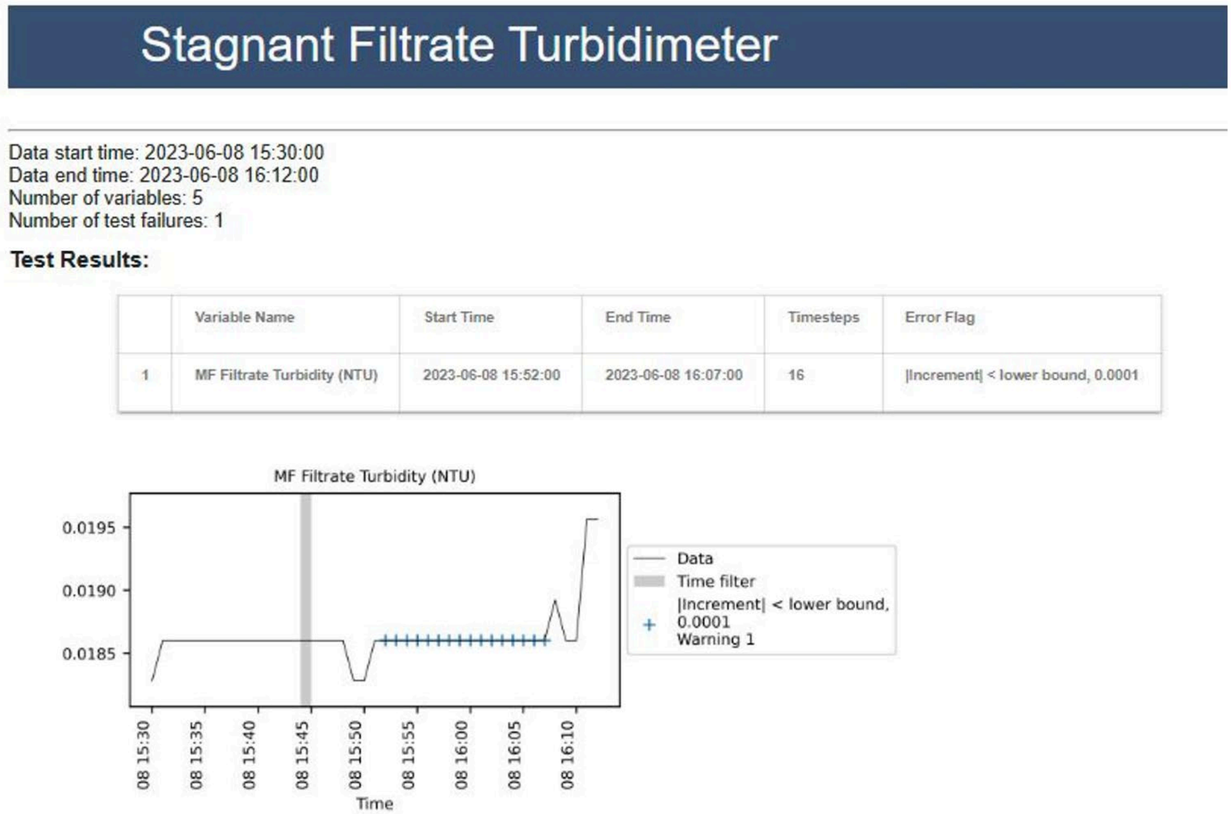
Test Results:

	Variable Name	Start Time	End Time	Timesteps	Error Flag
1	MF Filtrate Turbidity (NTU)	2022-12-20 12:38:00	2022-12-20 13:20:00	43	Data > upper bound, 0.15

Figure 5-5. MF High Filtrate Turbidity Process Failure Event Detection Report.

5.2.2.2 Monitoring Point

To test the detection system's ability to detect a turbidimeter being left in a "hold" status, the MF filtrate turbidimeter at NCPWDF was configured to report the same value for 16 minutes. The resulting MF monitoring point event outputs confirmed that the EDS was properly configured for identifying stagnant data, as demonstrated by the blue crosses in Figure 5-6 during the timeframe detailed in the test results table.



Stagnant Filtrate Turbidimeter

Data start time: 2023-06-08 15:30:00
Data end time: 2023-06-08 16:12:00
Number of variables: 5
Number of test failures: 1

Test Results:

	Variable Name	Start Time	End Time	Timesteps	Error Flag
1	MF Filtrate Turbidity (NTU)	2023-06-08 15:52:00	2023-06-08 16:07:00	16	|Increment| < lower bound, 0.0001

Figure 5-6. MF Stagnant Filtrate Turbidity Monitoring Point Failure Event Detection Report.

The gray bar on the data plot indicates the time period that data was screened out due to the MF process being inactive (i.e., backwash). The number of consecutive stagnant data points restarts following each time filter which is why the EDS does not flag the initial stagnant stretch of data. Initial attempts to identify stagnant MF filtrate turbidity data found that a minimum failure duration of less than 15 minutes would cause excessive false alarms due to the stable nature of the parameter. Extending the minimum failure duration requires more consecutive data values to be the same before the Pecos stagnant data test is flagged. In addition, the stagnancy threshold used at the NCPWDF went through multiple iterations to determine that 0.0001 was the appropriate sensitivity for detecting a turbidimeter that has been set to hold a value while avoiding the generation of false alarms as seen in Figure 5-6 above.

5.2.3 Reverse Osmosis Event Detection Challenge Testing

The RO system at the NCPWDF maintained stable operations throughout the duration of the project, therefore naturally occurring examples of process failure, monitoring point failure, and water quality events did not present themselves. In lieu of real event occurrences, the manually generated challenge tests discussed in the sections below confirmed the efficacy of the EDS for identifying potential events in the RO process.

5.2.3.1 Process Failure

Process failure of the RO CCP such as an O-ring compromise would result in poor permeate water quality, as indicated by high measurements of both EC and TOC. Thus, process failure in the RO system was replicated by simultaneously increasing RO permeate TOC and EC for a sustained duration of 44 minutes within the EDS's data logger. The blue crosses in Figure 5-7 represent flags generated by the EDS for RO permeate TOC values greater than the maximum threshold of 50 µg/L and Train A EC values greater than 125 µS/cm. The Test Results table within the output report displays the timeframe of the event and confirms that the challenge test was appropriately classified as a process failure.

Potential Membrane Breach

Data start time: 2022-12-20 08:00:00
Data end time: 2022-12-20 10:09:00
Number of variables: 8
Number of test failures: 4

Test Results:

	Variable Name	Start Time	End Time	Timesteps	Error Flag
1	RO Combined Permeate TOC (ug/L)	2022-12-20 09:21:00	2022-12-20 10:05:00	45	Data > upper bound, 50
2	RO Process Event	2022-12-20 09:22:00	2022-12-20 10:05:00	44	Data > upper bound, 0
3	RO Train A Permeate Conductivity (uS/cm)	2022-12-20 09:22:00	2022-12-20 10:09:00	48	Data > upper bound, 125
4	RO Train B Permeate Conductivity (uS/cm)	2022-12-20 09:22:00	2022-12-20 10:09:00	48	Data > upper bound, 125

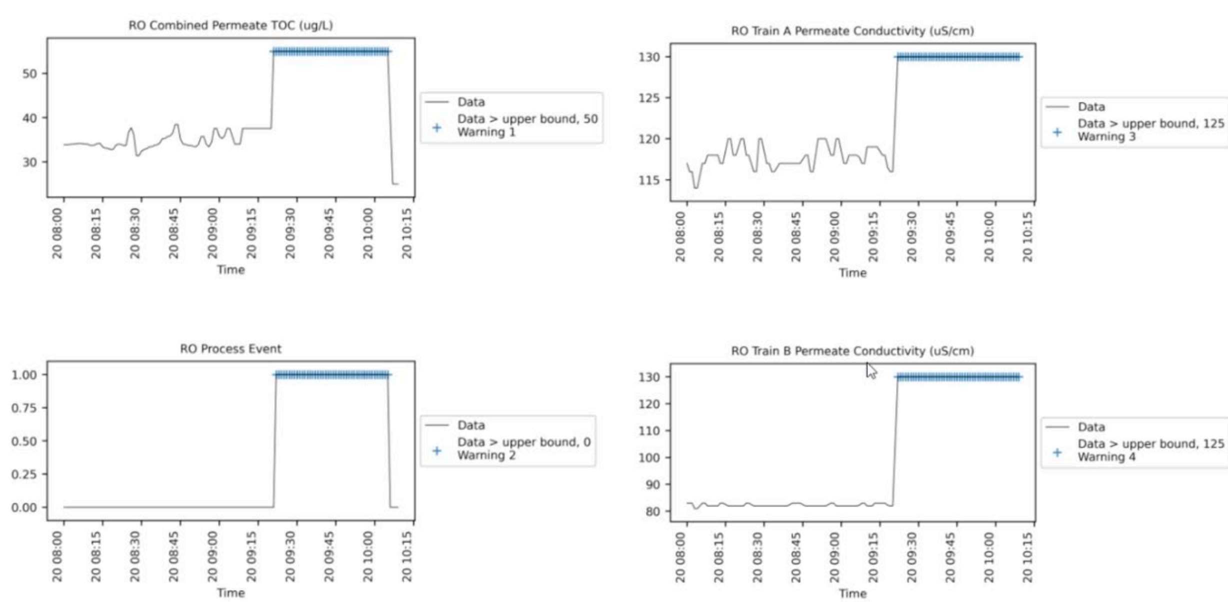


Figure 5-7. RO Potential Membrane Breach Process Failure Event Detection Report.

5.2.3.2 Monitoring Point

Since the analyzers measuring TOC and EC in the RO feed and permeate do not have reference meters, measurement accuracy must be assessed using an alternative strategy. When analyzing normalized data, RO membranes typically reject contaminants at a fixed percent removal rate, meaning that an increase or decrease in the feed concentration of a constituent would correspond with a similar increase or decrease in the permeate concentration. Therefore, the LRV for TOC and EC would be expected to remain within a narrow range. Data that deviates from these expected trends in TOC and EC would indicate that measurements from one of the feed or permeate analyzers is drifting from the true value and in need of maintenance. The event example simulated a feed TOC meter that has drifted downwards as indicated in Figure 5-8 by a drop in feed TOC concentration below the lower operational bound threshold of 3,780 µg/L. Since the permeate TOC concentration remained stable due to a properly functioning analyzer in that location, relative decrease in TOC across the membrane was perceived as less than typical and the TOC LRV declined below the lower operational bound of 2.1 as shown in Figure 5-8. The combination of low feed TOC concentration and low TOC LRV prompted the EDS to identify a RO monitoring point event from 11:12 to 11:44 as demonstrated in the test results table of the event report (Figure 5-8).

Feed TOC Meter Drift

Data start time: 2022-12-20 10:22:00
Data end time: 2022-12-20 11:41:00
Number of variables: 7
Number of test failures: 3

Test Results:

	Variable Name	Start Time	End Time	Timesteps	Error Flag
1	RO Feed TOC (ug/L)	2022-12-20 11:12:00	2022-12-20 11:41:00	30	Data < lower bound, 3780
2	RO LRV via TOC (LRV)	2022-12-20 11:10:00	2022-12-20 11:41:00	32	Data < lower bound, 2.1
3	RO Monitoring Event	2022-12-20 11:12:00	2022-12-20 11:41:00	30	Data > upper bound, 0

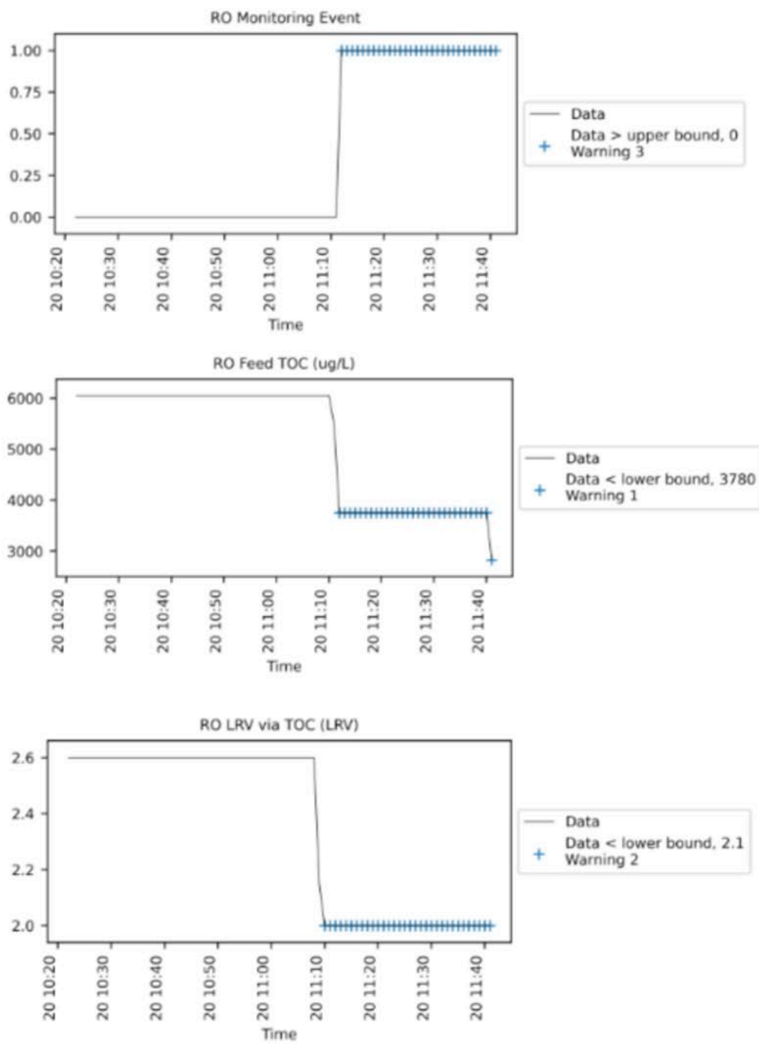


Figure 5-8. RO Feed TOC Meter Drift Monitoring Point Failure Event Detection Report.

5.2.3.3 Water Quality

As discussed in section 4.5.3.3, one example of a water quality event in the RO process is referred to as a "chemical peak." The rejection of low molecular weight organic compounds through the RO process can vary across a wide range. For example, acetone is a compound that was found in bench scale experiments to have lower removal rates than higher molecular weight organics at around 50% (Breitner 2017). The presence of elevated concentrations of these types of compounds in the RO feed would result in a spike in RO permeate TOC readings but would not increase permeate EC because this type of water quality change would not impact the membranes' ability to reject dissolved salts in the water. The ability to distinguish between an RO membrane breach and a chemical peak water quality event is based on this concept. A simulated chemical peak was achieved by sending RO feed through the permeate TOC analyzer to increase the permeate TOC readings above the upper operational bound of 50 μg/L for 25 minutes. There are two apparent spikes displayed in Figure 5-9. It is important to note that the RO water quality event alert was not triggered by the first peak because the duration was less than the specified minimum number of consecutive failures (15 minutes). Since the second peak caused RO permeate TOC to remain above 50 μg/L for more than 15 minutes, the EDS correctly identified the RO water quality event as demonstrated in report's test results table. Throughout the timeframe of the challenge test, RO permeate EC from both trains remained within its typical operational range as illustrated by the blue crosses in Figure 5-9, ensuring that an RO process failure event alert was not generated.

Chemical Peak

Data start time: 2023-06-08 15:30:00
Data end time: 2023-06-08 17:00:00
Number of variables: 8
Number of test failures: 6

Test Results:

	Variable Name	Start Time	End Time	Timesteps	Error Flag
1	RO Combined Permeate TOC (ug/L)	2023-06-08 16:21:00	2023-06-08 16:45:00	25	Data > upper bound, 50
2	RO Train A Permeate Conductivity (uS/cm)	2023-06-08 15:30:00	2023-06-08 16:59:00	90	Data < lower bound, 125
3	RO Train B Permeate Conductivity (uS/cm)	2023-06-08 15:30:00	2023-06-08 16:59:00	90	Data < lower bound, 125
4	RO Water Quality Event	2023-06-08 16:21:00	2023-06-08 16:45:00	25	Data > upper bound, 0

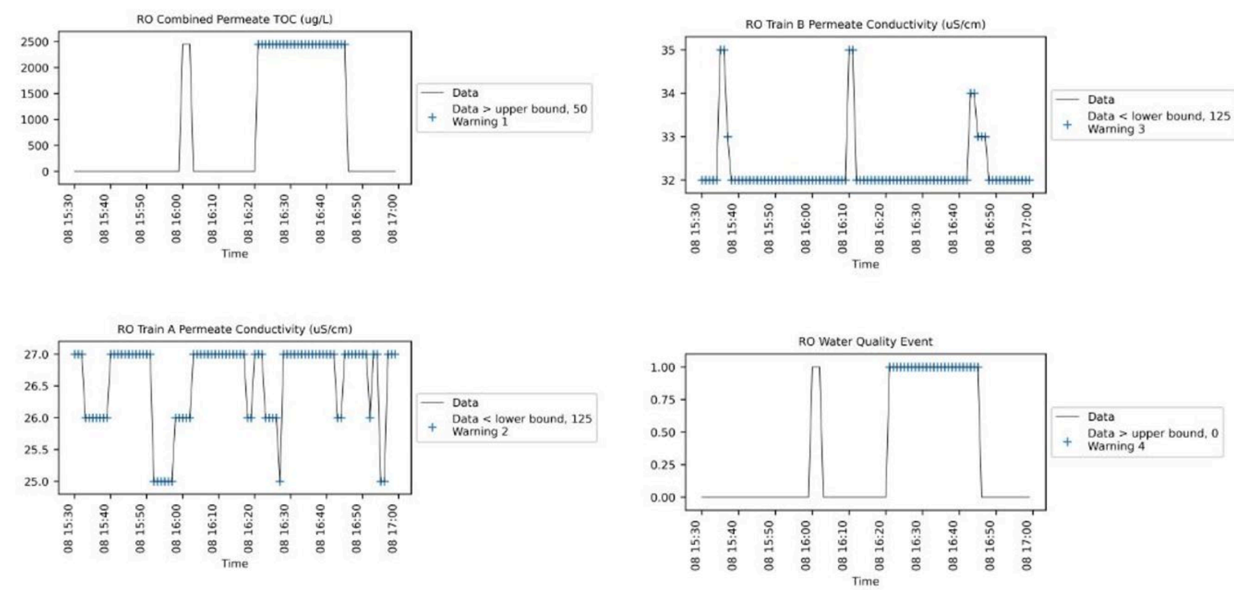


Figure 5-9. RO Potential Chemical Peak Water Quality Event Detection Report.

5.2.4 UV/AOP Event Detection Challenge Testing

Due to consistent performance by the NCPWDF UV/AOP system over the duration of the project, it was rare that process failure, monitoring point, or water quality events occurred organically. This necessitated the use of artificial challenge tests to confirm the functionality of the EDS. A challenge test for each of the three event categories was conducted on March 23rd, 2023, and the results are described below.

5.2.4.1 Process Failure Challenge Testing

To simulate a UV/AOP process failure from a sudden decline in UV dose, the UV dose was lowered to 295 mJ/cm^2 for almost 30 minutes to test if an UV/AOP process failure event would be detected by the EDS, and if the system would provide an alert. The blue crosses in Figure 5-

10 represent flags generated by the EDS for UV dose values less than the minimum threshold of 300 mJ/cm^2. The test results table in the output report displays the timeframe of the event and confirms that the challenge test was appropriately classified as a process failure.

Low UV Dose

Data start time: 2023-03-24 09:00:00
Data end time: 2023-03-24 11:59:00
Number of variables: 5
Number of test failures: 1

Test Results:

	Variable Name	Start Time	End Time	Timesteps	Error Flag
1	UV Dose (mJ/cm2)	2023-03-24 10:32:00	2023-03-24 11:00:00	29	Data < lower bound, 300

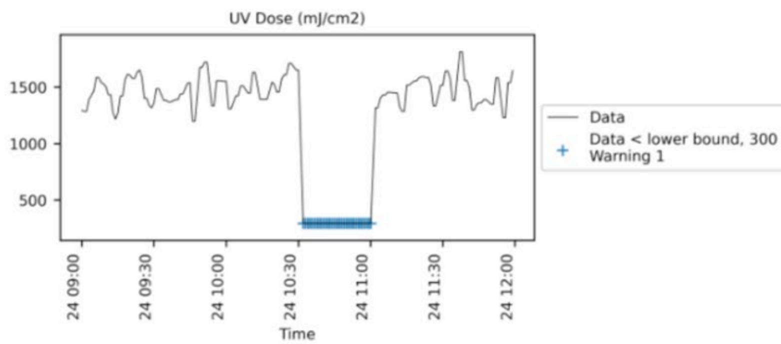
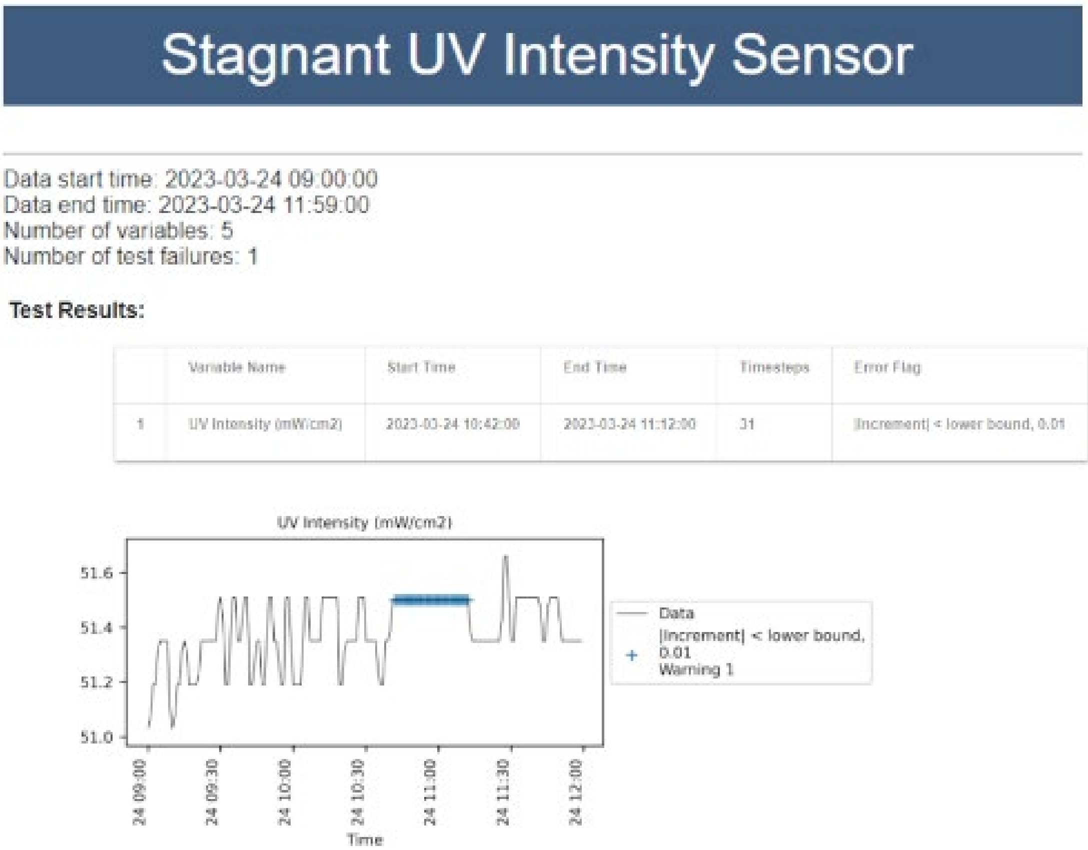


Figure 5-10. UV/AOP Low UV Dose Process Failure Event Detection Report.

5.2.4.2 Monitoring Point Challenge Testing

The goal of the UV/AOP monitoring point challenge test was to provide proof of concept for identifying stagnant data readings throughout the NCPWDF treatment train. The UVI sensor output value was manually fixed at 51.5 mW/cm^2 for approximately 30 minutes. The results presented in Figure 5-11 indicate that the EDS successfully identified the UVI sensor reading as stagnant and generated the appropriate alert for a UV/AOP monitoring point event.


Stagnant UV Intensity Sensor

Data start time: 2023-03-24 09:00:00
Data end time: 2023-03-24 11:59:00
Number of variables: 5
Number of test failures: 1

Test Results:

	Variable Name	Start Time	End Time	Timesteps	Error Flag
1	UV Intensity (mW/cm2)	2023-03-24 10:42:00	2023-03-24 11:12:00	31	|Increment| < lower bound, 0.01

Figure 5-11. UV/AOP Stagnant UVI Sensor Monitoring Pint Failure Event Detection Report.

5.2.4.3 Water Quality Challenge Testing

A change in water quality that could impact UV/AOP performance would likely be captured by a change in UV feed UVT. Therefore, the EDS was tested to see if it would identify the trend of decreasing UVT. This was accomplished by artificially setting the UVT reading to a low value of 95.5% for 35 minutes. Figure 5-12 displays the system's outputs demonstrating that the drop in UVT was recognized by the system to be below the minimum threshold of 96% and the event was appropriately detected.

Low UV Feed UVT

Data start time: 2023-03-24 09:00:00
Data end time: 2023-03-24 11:59:00
Number of variables: 5
Number of test failures: 1

Test Results:

	Variable Name	Start Time	End Time	Timesteps	Error Flag
1	UV Feed UVT (%)	2023-03-24 10:19:00	2023-03-24 10:53:00	35	Data < lower bound, 96

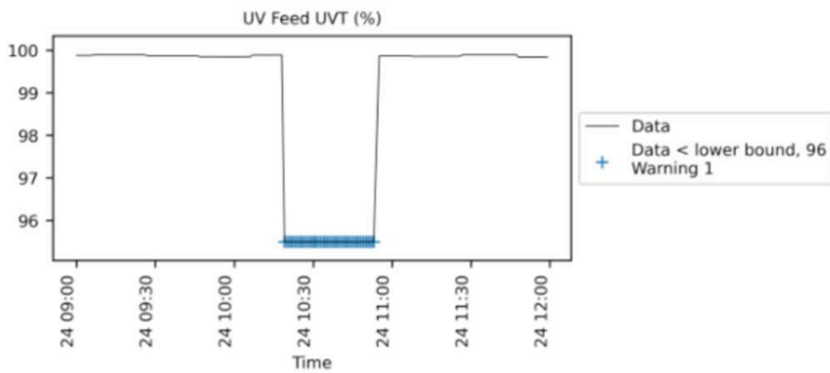


Figure 5-12. UV/AOP Low Feed UVT Water Quality Event Detection Report.

5.3 Challenge Testing Summary

The goals of challenge testing the deployed event detection system at the NCPWDF were to:

- Validate the four-step event detection framework.
- Assess the software infrastructure, data workflow, and event detection logic.
- Optimize the Pecos quality control test input values for each CCP monitoring parameter.

The EDS output reports shown above demonstrate that the logic-based framework is a valuable tool for helping ensure that public health is protected when receiving product water from a DPR facility. Although the test input values used in the system will vary across facilities (and even at the same facility over time), the framework described in Chapter 3 provides guidance for developing these tests and the challenge testing conducted in this chapter validates the framework developed in Chapter 4. Throughout challenge testing, multiple iterations of the event detection logic for certain process events were required to achieve the appropriate detection sensitivity. Logic tests that were poorly configured often led to false positives and/or nuisance alarms while logic that required too many conditions to detect the event (i.e., overly specific) was susceptible to reporting false negatives. Further details on the events that needed adjustments in the event detection logic can be found in Appendix B. These test results are evidence that an EDS using Python packages (pandas, loguru, Pecos, and plyer) can successfully monitor for and detect potential CCP failures within a DPR facility in real-time.

The following section, Chapter 6, includes a comprehensive discussion of the findings from the project team's work. Also included are recommendations for the continued development of the EDS at the NCPWDF and future DPR facilities using the established framework.

CHAPTER 6

Project Conclusions

Development of the tool included the following aspects and was discussed in the following Chapters:

- Chapter 1: Need for EDS in DPR Systems
  1. Describes challenges to implementing DPR and research focus of Project
  2. Present previous WRF projects that have advanced data monitoring and analysis
  3. Importance of pathogen control in a DPR setting and increasing lead time before CCP failures
- Chapter 2: Literature Review
  1. Description of current monitoring and control systems based on interviews with operations staff from IPR facilities
  2. Compare different software tools that could be used to accomplish objectives of this Project
- Chapter 3: Develop an event detection framework to respond to CCP failures
  1. Collecting raw data
  2. Screening offline data
  3. Data flagging
  4. Event classification
- Chapter 4: Develop tool specifications
  1. Selection of tags for specific CCP monitors
  2. Identifying event categories and common events that may indicate CCP failures in potable reuse treatment trains
- Chapter 5: Assess proof of concept
  1. Deploy the framework and software tool to run in real-time at a potable reuse demonstration facility
  2. Perform challenge testing to assess EDS's ability to diagnose emerging events before regulatory limit is crossed

6.1 Chapter 1 Conclusions—Need for EDS in DPR

One of the major challenges for DPR implementation is ensuring reliable protection of public health at all times in the absence of an environmental buffer. This project sought to increase the reliability of DPR systems by developing a software-based event detection system to augment existing control and alarm systems by automating event detection and providing initial assessment of root cause. The following goals and for creating an event detection system framework are summarized below:

- Monitoring and alarms in DPR projects must be proactive and responsive to prevent release of off-specification product water.
- Giving operators additional lead time and more rapid resolution times will help ensure that response times are faster than the short retention times of DPR.

- Develop and apply a real-time event detection system framework to ozone, membrane filtration, RO, and UVAOP.
- Develop a generalizable blueprint for reuse stakeholders to apply to own reuse project.

6.2 Chapter 2 Conclusions—Utility Interviews and Literature Review

In the utility interviews conducted, it was found that utilities currently use static limits to alert operators to take action as an event detection system. Different alarm levels can be configured such that early notification can be provided for low and high values and compliance limits can be set for low and/or high values to represent a more severe alarm level. Once an alarm was issued, operators were responsible for determining if the alarm is real and proceed to troubleshoot and resolve issue through process of elimination if the alarm were real. The utility surveys demonstrated that current industry practices for event detection consist of notification once static limits are violated and actions thereafter are mostly manual and rely heavily on operators to troubleshoot and resolve potential events before critical regulatory limits are crossed.

In reviewing open-source software, the project team identified promising tools across multiple sectors, including engineering, data science, and industrial process control. Considering the requirements of DPR data analysis and event detection, the following factors were identified as high priority for software selection:

- Software must be built using a modern programming language that is free to use.
- Software must provide flexibility to allow adaptation to different treatment trains and regulatory standards.
- EDS must provide transparency about how events are detected to improve quality control and encourage trust.
- Few CCP failures ever occur, so detecting failures is inherently a data limited problem. The EDS must not require a high volume of training data (i.e., failure events) to perform well.

By comparing these requirements with the available software, Pecos was identified as the tool best suited for developing an EDS for this project. Pecos was developed to monitor and generate reports based on real-time data streams, such as industrial control systems. This Python-based tool is modern, flexible, and provides a series of logic-based tests that are clearly structured and understood. For example, concepts from statistical process control can be implemented using Pecos tests. Because CCP failures are rare, data-driven approaches like machine learning/AI were deemed unlikely to perform well for this project. Moreover, those approaches would not provide a clear rationale to the operator or engineer about why an event was detected.

6.3 Chapter 3 Conclusions—Event Detection Framework

A four-step event detection framework was developed by the project team and can be used as a blueprint for other process engineers to design an EDS specific to their site. The four steps include (1) data storage, (2) data screening, (3) data flagging, and (4) event detection.

First, the project team had to decide which data to store and use for further analysis. The amount of data tags and frequency of monitoring data can impact the runtime of the event detection program script. Selecting a high recording frequency will slow down the script but selecting a low recording frequency will increase the lead time to detecting an event. The project team tested a few different recording frequencies and found that a 1-minute recording frequency was optimum for decreasing lead time without overburdening the program script. Next, the project team took the data screening lessons from WRRF 4765 and utilized integer status codes instead of using a series of logical tests (e.g., check flow rate, pump speeds, etc).

For data flagging, many different quality control tests (e.g., abrupt change, range, etc.) could be used but ultimately range tests were applied to flag data for further evaluation. Bounds were assigned based off Team's operational knowledge and experience and using historical data from the deployment site. Bounds set with this methodology ground the data flagging step in real-world experience. While new facilities will not initially be able to base bounds off of statistical analysis of historical data, the operational experience and knowledge considerations shared in this Report can be used when enough historical data is amassed. As noted in this report, process data that is highly variable is not suited for applying statistical analysis of historical data since fluctuations are too variable to create narrow bounds. This is where an iterative approach becomes necessary and operator knowledge must be leveraged.

Lastly, event classification was accomplished through the use of CCP monitoring parameters and a combination of "if/then" and "and/or" statements. For example, "IF ozone generator is operating properly AND primary residual meters are within acceptable variance from redundant meters, THEN high ozone demand can be attributed to a water quality upset." It was found that automation of event detection is a balance between creating too many or too little logical test conditions. Logic tests that have too few arguments lead to more false positive and nuisance alarms. Conversely, if logic tests are too specific to limit false positives, there is a higher likelihood that an event will not be detected by the logic tests because the logical test arguments are too restrictive. The project team continued to modify the logical tests throughout the proof-of-concept phase to balance generating too many event detection notifications and too little. The logical tests presented in Chapter 4 represent one year of optimization and can be used as the starting point for further optimization for other groups interested in implementing an EDS. The following guidelines and conclusions were gathered from this phase of the Project:

- Development of an EDS relies on the collective expertise of process engineers, operators, IT staff, systems integrators, data engineers, and software developers.
- Data storage lays the foundation for event detection and data must be transferrable between PLCs, SCADA, and a database and must be accessible by the event detection system. Ensuring data transferability relies heavily on coordination with IT staff, systems integrators, and data engineers.

- Process engineers and operators should communicate their understanding of the regulatory requirements and system operations to ground the data screening, data flagging, and event detection logic in real-world applications and practices.
- One-minute timesteps may be considered as a good starting point for DPR applications to provide frequent data logging and keeping data storage capacity needs and script speed reasonable.
- Automation of early event detection is feasible for events that develop/occur in a systematic and predictable manner.
- Events should be defined such that, when detected, operators are notified that a potential CCP failure may occur if left unaddressed and what the potential cause may be.
- Events cannot be classified with a high degree of certainty based on a single monitor. Comparison with other monitoring points corroborates and increases confidence that the identified root cause for the event is an accurate assessment.
- Quality control tests can be configured as positive or negative indicators. Combining positive and negative logic tests allows for more complex analysis of a unit process making event detection more robust through process of elimination and confirmation.
- When designing logic tests for event detection, utilizing less logic tests will increase the number of detected potential events whereas utilizing too many logic tests will run the risk of missing actual events.

6.4 Chapter 4 Conclusions—Tool Specifications

The first step in development was specifying the CCP monitoring parameters that would undergo data analysis and the types of events that the tool would be designed to detect. The Team identified events that would impact a CCP's ability to protect public health and would likely occur in potable reuse projects. Several members of the project team had substantial years of experience operating potable reuse trains and could leverage this knowledge and experience to define what events are most likely to occur as discussed in Chapter 3. The types of events that were ultimately implemented underwent several rounds of refinement since not every type of event is suitable for automated detection. Some events occur too sporadically and do not manifest in a systematic way each time they occur. To automate event detection, the event must be detectable using the same monitoring parameters and monitoring parameters must exhibit the same behavior each time the event occurs.

After the events were identified, an initial list of 22 out of 8,000 available tags was developed for ozone, membrane filtration, RO, and UVAOP and identified as critical monitoring parameters that would affect the CCP's ability to protect public health. These tags are summarized in Table 6-1.

Table 6-1. Selected Critical Monitoring Points

Process	Parameter	Units	Description
Ozone	Dissolved ozone residual at OSP 4, 7, and 10	PPM	• Used to assess meter functionality and process water quality Used to calculate ozone demand, ozone decay coefficient, CT, and LRV
	Ozone generator production	PPD	• Used to assess process functionality Used in the ozone demand calculation
	Temperature	deg F	Used to calculate pathogen inactivation rate constant for LRV calculation
	Water flow	gpm	Used to calculate HRTs for CT calculation
	Ozone demand	PPM	• Calculated using ozone generator production and dissolved ozone residual at OSP 4 Used to monitor water quality changes of the feed water
	Ozone decay coefficient	min^{-1}	• Calculated using the dissolved ozone residual at OSP 4, 7, and 10 Used in CT calculation
	CT	PPM*min	Calculated using HDT, OSP 4 ozone residual, and ozone decay coefficients with the EPA truncated extended-integration method
	Pathogen Removal	LRV	• Quantifies treatment of the system Calculated using CT and a pathogen-specific inactivation rate constant
MF	**MF Filtrate Turbidity**	NTU	Used to assess process and meter functionality
RO	RO Feed TOC	µg/L	Analytical measurement used to calculate RO TOC removal
	RO Combined Permeate TOC	µg/L	Analytical measurement used to: • Calculate RO TOC removal • Monitor permeate water quality to meet permeate TOC regulatory requirements Detect events such as a membrane breach or organic chemical spike
	RO TOC Removal	LRV	Calculated log-reduction value:

Process	Parameter	Units	Description
			• From feed and combined permeate TOC measurements Used as the primary surrogate for pathogen LRV
	RO Feed EC	µS/cm	Analytical measurement used to calculate RO EC removal
	RO Train A permeate EC	µS/cm	Analytical measurement used to: • Calculate RO EC removal Monitor permeate water quality to detect a membrane breach
	RO EC removal	LRV	Calculated log-reduction value: • From feed and combined permeate EC measurements Used as the secondary surrogate for pathogen LRV
UVAOP	UV Dose	mJ/cm^2	• Calculated parameter used to assess process functionality and determine if pathogen inactivation and AOP requirements are met Accounts for UV lamp status
	UV Intensity (UVI)	mW/cm^2	• Used to assess monitoring point functionality Used to calculate UV dose
	Feed UV Transmittance (UVT)	%	• Used to assess UV/AOP feed water quality Indicates amount of UV light that is available for disinfection/photolysis/AOP
	RO Permeate Total Chlorine	mg/L	• Total chlorine measured prior to the oxidant injection point Used to assess water quality and oxidant dosing
	UV Feed Total Chlorine	mg/L	• Total chlorine measured following the oxidant injection point Used to assess oxidant dosing

Process	Parameter	Units	Description
	UV Feed Free Chlorine	mg/L	• Free chlorine measured following the oxidant injection point Used to assess oxidant dosing
	Pathogen Removal	LRV	UV/AOP receives 6.0 LRV for viruses, *Giardia*, *Cryptosporidium* if UV dose exceeds minimum regulatory requirement and treatment conditions are within operating envelope

Curation of the tag list was an iterative process and the project team continued to modify the tag list throughout the development of EDS framework and proof of concept phases. New calculated tags were even implemented by the system integrator in SCADA and PLCs to further augment the reliability of data flagging and event detection. Types of calculations included percent difference between raw sensor values and rolling average and percent difference between redundant monitors.

The following takeaways and lessons learned were encountered during this aspect of the project:

- The curated list at the conclusion of this Project represents one year of optimization and can be used as the starting point for further optimization for other groups interested in implementing an EDS.
- Test limits and configuration of logical tests for event detection must be site-specific due to the unique water quality and operational considerations at individual facilities.
- CCP monitoring parameters and event detection criteria developed for monitoring vary between CCPs within the facility due to the unique pathogen removal mechanisms that are employed by each CCP.
- The event detection system developed during this Project is site-specific in terms of quality control test input parameters. The event detection logic, though, can be broadly applied for the failure events identified and developed during this Project.
- Utilization of raw values from monitoring points is not enough to build an EDS and implementing calculations on SCADA/PLCs is another useful way to compare raw values from two different data sources. These types of calculations represent comparisons between meter readings and expected readings and augment the reliability of the EDS.

6.5 Chapter 5 Conclusions—Proof-of-Concept

Once the EDS was developed and tested offsite using historical data from the deployment site, the tool was installed at the NCPWDF to evaluate if real-time monitoring and detection of events was possible. The following components were configured at the NCPWDF to deploy the EDS.

- Hardware
  1. HMI / PC Workstation, Windows 11 Pro.
- Software
  1. Kepware, Python packages, SQL server

The event detection script was automated to run every 5 minutes. In addition, the script could be initiated by a user to run for a user-defined period of time (e.g., over previous hour, several hours, or days) and scan for events. The EDS is an additional monitoring system that supplements the existing control and alarm system. While there is no flexibility to adjust control setpoints and alarm limits subject to regulatory thresholds, the EDS can be adjusted freely. Even if changes to bounds or logical tests lead to undesirable alarms (e.g., nuisance false positive alarms or decreased sensitivity to detecting events), the existing control and alarm system will react per automated control strategy to shut down affected unit or divert. Event detection on the other hand should alert the operator of any emerging or past events in advance of the shutdown conditions and by providing quicker detection of events allow for a faster response time.

Results generated during deployment were reviewed by the project team to evaluate if detected events were accurate. The tag list and logical tests were adjusted throughout deployment as more events were detected. With each new event detection, the project team found ways to optimize the event detection script to minimize false positives and increase lead times. Even with advanced testing with historical data prior to deployment, further adjustments were needed. This highlights the complexity of automating event detection and the more data that is available, the better the tests. In addition, for events that did not occur organically (e.g., due to some minor process upset upstream or some mechanical failure), the project team configured and carried out simulated failures for each event category. The following conclusions were drawn from this testing period:

- The data screening method using status tags implemented in this Project eliminated many of the false positives observed in WRF 4765 and only passes on data while treatment train is in production for further analysis. The project team recommends utilizing process status integers in logical tests to assess whether data should be screened out.
- Downstream processes (MF, RO, UV/AOP) are easier to fine-tune for accuracy and sensitivity than upstream processes (ozone) due to reduced dependence on water quality and/or process mechanics.
    1. Ozone/BAC pretreats feed water that is most concentrated with contaminants, which reduces organic load to downstream membrane system resulting in more stable and enhanced performance of said process.
- The bounds used for the logical tests were tailored to the NCPWDF processes and may not be directly applicable to other facilities. However, the approach used to define and optimize the bounds can be used as a practical example.
- The event tool is a redundant monitoring tool that can be used to enhance reliability of the monitoring and alarm capabilities of the DPR system and functions separately from SCADA and historized data sets.
- An accurate and functional EDS package would be difficult to implement as a plug and play tool without site specific knowledge to optimizing Pecos quality control test parameters such as:
    1. Workflow, logic tests, and tag list for ozone, membrane filtration, RO, UV/AOP developed in this project should be applicable to other projects, but some adjustments to the logical test arguments may be needed to tailor the system to specific treatment train.
- The Python library Pecos can be used to develop a software system that continuously monitors DPR process data for key events identified by the project team:
    1. Implemented 12 events, each looking at unique water quality, process, or monitoring location failure as a proof of concept and the same logic and code can be expanded to comprehensively cover a larger full-scale facility.
    2. Earlier detection of events extends lead time for operators to address potential issues before a shutdown/diversion becomes necessary.

6.6 Next Steps and Additional Considerations

6.6.1 Event Detection Framework and Implementation

Although the event detection system implemented at the NCPWDF contained event detection logic for each of the facility's CCP unit processes, it focused on 12 events and therefore did not provide a fully comprehensive monitoring of all the possible issues that may occur. The project team spent considerable time optimizing logic for detection of the 12 events including running into some approaches that were not as successful. Additional information surrounding the development of the event detection logic and criteria, including approaches that were tested but not recommended for implementation are summarized in Appendix B.

Further development of the system and functionalities would require development of logic for additional events. The details of which are described in Appendix C. Beyond the finalized Python script used for event detection during this project and provided in Appendix D, the efforts of the project team to develop the EDS resulted in considerations that should be useful for future applications. Specifically, Appendix C outlined some additional ideas for implementation that may be considered for implementation to further improve the event detection system developed during this project.

For example, stagnant data and analytical range events should be configured system-wide for all monitoring points (i.e., meters, instruments, analyzers). Ozone disinfection monitoring should include an event for declining LRV and a TOC-specific water quality event. Additional meter drift events should be incorporated for the TOC and EC analyzers used in the RO process. Event detection logic that confirms the oxidant dose and measurement accuracy for the UV/AOP process should also be included. Event configuration should follow the same approach as Chapter 4 where Pecos quality control tests are set to run on CCP monitoring parameters and flag data that reads outside of typical operating ranges. Pecos test inputs should continue to be iteratively adjusted by operations staff to account for seasonal variations (e.g., water temperature) in process data or gradual shifts in the normal operating range over time (e.g., aging RO membranes). The expansion of the event detection system would also require bolstering the dashboard to accommodate these additional events.

In addition, the project team started to explore implementing calculations comparing analyzers and reference values on SCADA and PLCs. These calculated parameters were incorporated for quality control tests and event detection logical tests towards the end of the project. For example, CCP monitoring parameters with reference analyzers (e.g., primary and backup analyzer to serve as reference values), the percent difference between the two analyzers were calculated and if the percent difference exceeded a predetermined limit (e.g., 20%), it is likely that one of the meters has drifted. While there was not sufficient time to fully implement and test event detection logical tests for all of these calculated parameters across all CCPs, initial testing showed that this was a powerful method to increase confidence in the event detection hits.

Lastly, further testing of the event detection system tool should continue over longer periods of time to evaluate if quality control test limits should be revised periodically and at what frequency. Certain water quality parameters and process performance may fluctuate seasonally, and/or over longer runtime. Although this was not fully explored during this project since the deployment period was less than one year, updating the test bounds to account for seasonal variability or equipment age may improve the sensitivity of event detection and further increase the lead time.

6.6.2 Event Detection Software Considerations

The event detection framework is designed to increase lead time and decrease time for event detection and resolution. An implementation of the framework was developed for the City of San Diego's NCPWDF as a proof-of-concept. That implementation was built using open-source code, largely with Python, and was deployed on-premises without connectivity to the internet. That code successfully implemented the core principles of the framework, such as data storage, screening, flagging, and event detection.

While building the software, the project team identified a list of desired features that could not be implemented within the scope of the project. These features are presented in this section to inform future development of event detection for reuse:

- Accessibility: data visualizations and alerts are accessible remotely by authorized staff.
- Configurability: data screening, flagging, and event detection can be modified from a user interface without the need to write code.
- Auditing and approval: whenever screening, flagging, or event logic is modified, what changed and who changed it is logged and traceable. Modifications to critical logic requires approval according to a user hierarchy.
- Alarm management: alerts and alarms are well organized and include troubleshooting instructions and likely causes.
- Maintainability: developers can access the latest version of the software and roll out updates and bug fixes remotely.
- Support: to assist operations staff, remote experts are on-call to provide troubleshooting help.
- Reporting: data screening logic can be reused for regular reports sent to regulators. This would automate the process of removing data during process downtimes, performing any statistical calculations, and outputting data into standardized report forms.

Many of these features require a cloud-based approach in which the treatment data is accessible via the internet. Exporting the data from the SCADA network can be done securely but it needs to be done according to best practices in cybersecurity to avoid creating vulnerabilities. The SCADA network must be highly secure because treatment processes can be controlled from within that network. Therefore, no utility should risk exposing the SCADA network to a cyberattack. Using hardware, such as a data diode, data can be securely exported out of the network. With this one-way transfer, treatment data can be connected to the internet so staff can monitor operations remotely.

Although the scope of this project was to develop open-source tools for potable reuse, it is important to note that the wish list features described are currently supported by software vendors. Such companies include Aquatic Informatics, Aveva, Autodesk, Canary Labs, Idrica, IOSight, Pani, Rockwell Automation, Royal HaskoningDHV, Veolia, and Xylem. These vendors are developing commercial-off-the-shelf software for industrial control systems, including water and wastewater treatment. When considering software licenses, systems integrators and utility IT staff will need to be closely involved in decision making. These professionals can provide guidance about whether SCADA data can be exported from the system, and if not, what investments would be required to do so. They will also need to be consulted about PLC programming, SCADA tags, and database configuration.

In addition, the industry generally lacks a standard approach to designing an EDS with specific applications such as DPR. The functions and features developed during this project and some additional developments described in this report may provide a preview of the level of detailed and goal-oriented planning that perhaps should be considered early on during project planning and implementation to be included in the design and delivery of specialized control systems for full-scale facilities.

APPENDIX A

Literature Review - Updated September 15, 2021

A.1 Introduction

The key feature distinguishing direct potable reuse (DPR) from indirect potable reuse (IPR) is the loss of an environmental buffer. By reducing or eliminating the environmental buffer, DPR creates a closer connection between the treatment and consumption of purified water. This is an important differentiator given that existing indirect potable reuse (IPR) paradigms rely on the environment to further improve the quality of the water and to provide additional time to respond to treatment and water quality issues. As a result, DPR introduces a collection of challenges above and beyond what is encountered in IPR schemes. Replacing the *treatment* provided by the environment is arguably more straightforward. Recently, California's Division of Drinking Water (DDW) developed anticipated criteria for DPR that includes significant additional treatment compared to the IPR requirements. The standard RO-based treatment train must be supplemented with additional robustness in the form of ozone and biological activated carbon (BAC) pre-treatment, both of which help replace the environmental attenuation that occurs in IPR. Unlike treatment, there is no easy way to replace the *retention time* provided by the environment. In all cases, a system's ability to respond to a treatment issue (response time) must be faster than the time the water is retained in the system (retention time) so that systems can identify and respond to failures before the water reaches consumers. If retention time cannot be extended, then response time must be shortened. The requirement for fast response times in direct potable reuse (DPR) systems imposes unprecedented new challenges to rapidly evaluate treatment performance.

Direct potable reuse (DPR) presents challenges in process monitoring that the water industry has not dealt with before in indirect potable reuse (IPR). While individual unit processes may be monitored on a real-time basis, the calculation of treatment train performance frequently occurs post hoc. Such performance analyses may only occur once per day or week (or even month) since the regulations may only require a monthly log removal value (LRV) report. This type of reactive analysis is acceptable in IPR scenarios because response times (approximately one month) are still significantly shorter than retention times in environmental barriers (multiple months to years). The response time situation differs radically for DPR: instead of months to years of retention time there may only be minutes to hours, i.e., a 1000- to 10,000-fold decrease in time. This imposes unprecedented new requirements for DPR. The entire process of integrating and responding to performance data must shift to real-time and proactive (DPR). Consequently, being able to integrate performance data and provide real-time automation of advanced treatment facilities is one of the biggest hurdles to DPR viability. Adapting to this is perhaps the biggest technical challenge for potable reuse in the absence of an environmental buffer.

This need has been recognized by the industry and there have been several studies funded by the Water Research Foundation to advance the industry's use of monitoring data. A summary of advancements in industry's use of monitoring data is provided in Figure A-1.

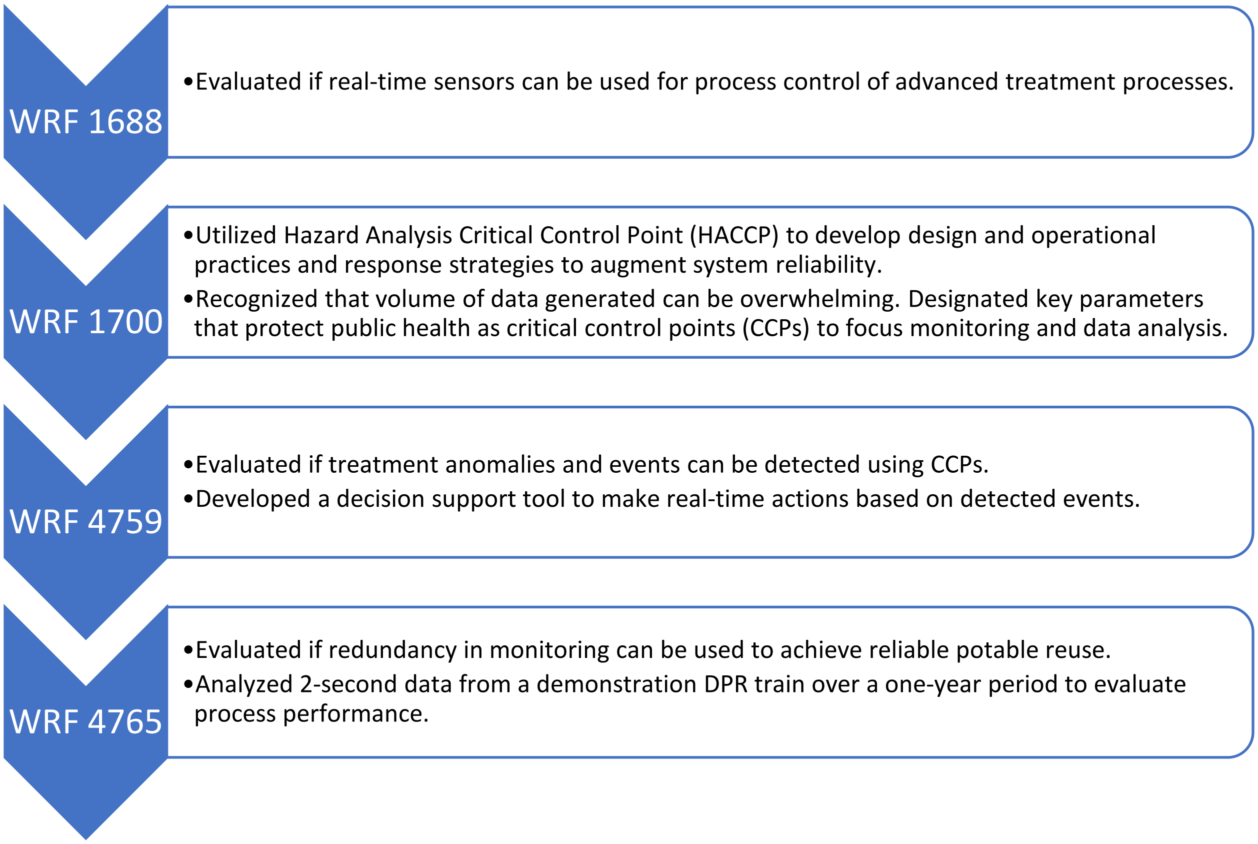


Figure A-1. Key Water Research Foundation Projects Advancing the Field of Monitoring and Data Analysis.

WRF 1688 (previously WRRF 11-01): addressed both IPR and DPR with a focus on the measurement of microbial contaminants (Snyder and Pepper 2016). This project reviewed online instruments in reuse—including accuracy, response time, and detection mechanism—and described best practices for using these instruments. The study found that online instruments commonly measure chemical contaminants or surrogates but not microbial contaminants. Turbidity, conductivity, and total organic carbon are commonly used to detect treatment failure. In addition, online fluorescence sensors and rapid tests for microbial contamination could supplement existing indicators, though the latter are still at an early stage of development.

WRF 1700 (previously WRRF 13-03): described the process of integrating hazard assessment and critical control point (HACCP) methodologies for DPR including hazard assessment, water quality objectives, identification of critical control points (CCPs) and critical operating points (COPs), CCP/COP monitoring parameters, and CCP/COP response procedures (Walker et al. 2016). This work also developed approaches to understand and communicate the risk associated with a compromised CCP barrier. Some of the key principles of HACCP, as formalized in ISO 22000, evaluate CCPS and evaluate the following questions:

1. Is there a hazard at this process step? What are the hazards?
2. Do control measures exist for the identified hazard?

3. Is the step specifically designed to eliminate or reduce the likely occurrence of the hazard to an acceptable level?
4. Could contamination occur at or increase to unacceptable levels?
5. Will a subsequent step or action eliminate or reduce the hazard to an acceptable level?

The study also provided commentary on drinking water regulations and how they might set a precedent for reuse regulations. In some countries, surface water treatment plants implement HACCP by monitoring surrogate parameters of pathogens and implement a multi-barrier approach. In the U.S., existing EPA regulations for drinking water treatment use a combination of both CCP monitoring and end-point monitoring. Many advanced treatment processes currently used in potable reuse use online monitoring to verify the performance of the system in real-time, e.g., turbidity monitoring on membrane filters or the use of the CT framework for chemical disinfectants.

WRF 4759 (previously WRRF 14-01): developed a Decision Support System (DSS) and an excel-based Decision Support Tool (DST) using online sensors that are applicable to potable reuse systems (Neemann et al. 2019). The study evaluated Event Detection System (EDSs) software packages for anomaly detection and discussed the idea of developing an Integrated Sensor Network (ISN) of both water quality and operations and maintenance data to detect failures in treatment. The study also evaluated use of commercially available monitoring sensors to identify failures for a pilot treatment train consisting of ozone and biological activated carbon.

WRF 4765 (previously WRRF 14-12): Direct potable reuse (DPR) has the potential to greatly expand the scope of water reuse worldwide, though questions remain about its ability to continuously protect public health. This uncertainty stems largely from the lack of full-scale performance data from actual DPR systems. Under legislative mandate, California was recently required to evaluate the feasibility of developing regulations for DPR. To help address this industry data gap, this project evaluated a 1.0-mgd demonstration facility at the City of San Diego’s North City Water Reclamation Plant to assess the benefits of redundancy and monitoring to achieve reliable potable reuse (Trussell et al. 2017). Yearlong continuous monitoring of the treatment train – consisting of ozone, biological activated carbon, membrane filtration, reverse osmosis, and UV with advanced oxidation – provided an extensive dataset to assess process performance. Routine performance monitoring was complemented with multiple challenge tests that assessed the benefits of the enhanced treatment train (Tackaert et al. 2019). The performance data were used in a quantitative microbial risk assessment to demonstrate that a full-scale DPR treatment train could reliably meet performance goals and produce a water that provides public health protection equivalent to, or greater than, conventional drinking water supplies (Pecson et al. 2017). The study also discusses the importance and relevance of timely operating data when operating DPR. This involved filtering and querying of sensor data to verify that the processes and sensors were functioning correctly (Chen et al. 2020; Pecson et al. 2018).

These studies demonstrated that existing monitoring capabilities can detect changes in process performance but identified a number of deficiencies in the current capabilities of real-time detection systems. The following sources were reviewed to determine the gaps that need to be addressed to enhance processing and analysis of monitoring data:

- Surveys with staff that are currently operating IPR facilities
- Lessons learned from WRF 4765 project on data management and analysis
- Literature studies on the use of data analysis tools in the water industry

The findings from these sources are summarized in the following sections below.

A.2 Review of Current Practices in Data Monitoring and Analyses

A.2.1 Utilities Surveys

To gain an understanding of the current practices in data monitoring and analysis, interviews were conducted with staff involved in the operations, monitoring, and control of facilities currently engaged in IPR: Orange County Water District (OCWD), Veolia, and Hampton Roads Sanitation District (HRSD). The focus of the interviews was to understand the current "state of the industry" for potable reuse systems, in particular existing monitoring and control systems, and data management strategies for regulatory and permit compliance. The WRF 4954 project team conducted interviews with the following staff:

- Mehul Patel (OCWD) on August 31, 2020
- Scott Murphy and Grahame Simpson (Veolia) on September 28, 2020
- Germano Salazar-Benites (HRSD) on December 9, 2020

An overview of the findings from the interviews is provided in Table A-1. All facilities utilize a control framework that involves the monitoring of CCP performance with online meters. Control limits are defined for all CCPs, and most limits are fixed, static values that may be periodically updated. None of the facilities dynamically set control limits. While the systems collect a high amount of performance data, much of the processing of the data is done by the operators rather than through automated processes. For example, operators are responsible for:

- Identifying abnormal meter behavior and initiating follow up investigations
- Spotting false-positive readings from monitors
- Creating strategies (e.g., alarm delays) to minimize the impact of transient blips in readings
- Developing compliance reports (though some steps in the process may be automated)

To minimize the impact of any single monitor on plant performance, all facilities have redundancy for one or more of the critical instruments.

Table A-1. Summary of Findings from Utility Surveys.

Survey Topic	OCWD	Veolia	HRSD
Reuse system description	Train: MF–RO–UV–UV/H2O2–stabilization Capacity: 100 MGD. 70 MGD to groundwater recharge, 30 MGD to seawater intrusion barrier	Train: MF–RO–UV–UV/H2O2–stabilization Capacity: 3 AWTF with 61 MGD capacity (48 MGD available). Only one facility running at the moment.	Train: Floc/Sed-O_3-BAC-GAC-UV-Cl_2. Capacity: 1 MGD research center. Up to 100 MGD with implementation of full-scale facilities.
Key control parameters	• MF: turbidity, MIT • RO: TOC, EC • UV/AOP: UVT, EED	• Currently only distributing for industrial users, yet meets drinking water standards.	• Ozone: CT • GAC: turbidity • UV: dose • Cl2: CT
Operating Criteria	• MF: filtrate turbidity < 0.15 NTU • MF: MIT > 0.2 psi/minute triggers work order • RO: Permeate TOC < 0.1 mg/L • UV: Feed UVT > 95% • UV: Dose > 101 mJ/cm2 • TOC: limit based on control chart statistical analysis of historical operational data	• Facilities use Critical Operating Points (COPs) and Critical Control Points (CCPs). These are related to ISO 22000 in Australia for Drinking Water. • Several alarms for CCPs based off of rate of change (e.g., ammonia). These are periodically reviewed and manually adjusted based on statistics. •	• Ozone: virus LRV > 3.5 • GAC: < 0.15 NTU • UV: dose > 186 mJ/cm2 • Influent: TOC < 15 mg/L • Influent: EC < 2,000 µS/cm • Influent: turbidity < 5 NTU • Influent: total nitrogen < 5 mg/L • Effluent: TOC < 4 mg/L
Redundancy in critical analyzers	Redundant TOC analyzers for RO permeate	Redundant ORP probes ahead of RO	Redundant TOC analyzers for GAC effluent
Frequency of validation/ calibration of instruments	Regular calibration/ validation done in-house for most meters/probes. Service contract for TOC analyzers.	Daily/weekly/monthly calibration/verification of all online instrumentation. Performed by in-house staff and external consultants/ vendors for more sophisticated analyzers.	Weekly/monthly (as-needed) calibration/verification of all online instrumentation. Performed by in-house staff.
Procedure for documenting instrument service	Tracked in Maximo	Tracked in Maximo	
Operational response to abnormal instrument behavior	Operators trained to spot anomalies and troubleshoot as part of SOP for responding to alarms. If sustained, can trigger sampling events.	Operators make decisions based on trends. Can trigger sampling events.	Operator to investigate. Confirmation with bench top readings. Instrumentation staff are available 24/7. Must confirm issue is fixed.

Survey Topic	OCWD	Veolia	HRSD
Compliance reporting procedures and purpose	• Monthly reports to DDW • Max and average values for each day in reporting period • Some automation from excel macro, but manual analysis required	• Reporting is primarily made to SNP Water, which is made available to Dept. of Health (DOH) • Exceedances are identified to the DOH on an annual basis and analytes are also made available to DOH • Internal reports are developed on a daily basis (process reports) for operational purposes and rolled up to monthly reporting • Some automation, but require manual to finalize	• Quarterly regulatory reporting for EPA's Underground Injection Control Program • Identifies operations and reasons for process being offline when applicable • Describes CCPs compliance for reporting period
Predictive analytics	None. There is a cybersecurity concern for cloud-based systems. Such analytics would need to be done locally.	None. Though SCADA does list responses/action items to fix issues based on alarms/warnings.	None.
Dynamic or static alarm levels	Mostly static. Some parameters, like TOC, are statistically informed using prior historical data	Dynamic: rate of change metrics that are specifically re-evaluated with some frequency	Mostly static. Correlations developed based on site-specific influent water quality and influence on product water quality (e.g., influent conductivity influent on bromate formation).
Frequent false positives or nuisance alarms	Operators are trained to spot false positives.	Operators are trained to spot false positives. Daily process info is reviewed by plant operators and management.	Alarms triggered after two consecutive readings to diminish false positives. Lessons have been learned on how to diminish false positives.

A.2.2 Lessons learned from WRF 4765

One key takeaway from WRF 4765 was that processing data from DPR systems—which may produce more than 300 GB of data annually—cannot be done without complex and automated data filtering (Pecson et al., 2017). This requires understanding (a) which critical control point monitors are needed to assess the performance of the different unit processes, (b) how to set up filters to focus on unit process performance when the system is producing water (vs. when it is offline or in startup mode), and (c) how to compile and integrate these data to assess systemwide performance. The data processing required removing certain data points from the analysis, such as when the system was offline or shut down. As expected, there was significant deviation in sensor signals when the system was offline or when it was transitioning from an off-line state to steady-state production. If these signals were included in the dataset for analysis, it would appear as if there were many failure events. However, these false-positives should be removed from the dataset in order to focus exclusively on the data generated when the system was actually producing and distributing water. Extensive data cleaning/filtering was implemented using an R-script to distinguish the data that were produced when the system was online and operational. The general framework used in WRF 4765 for filtering data is shown in Figure A-2.

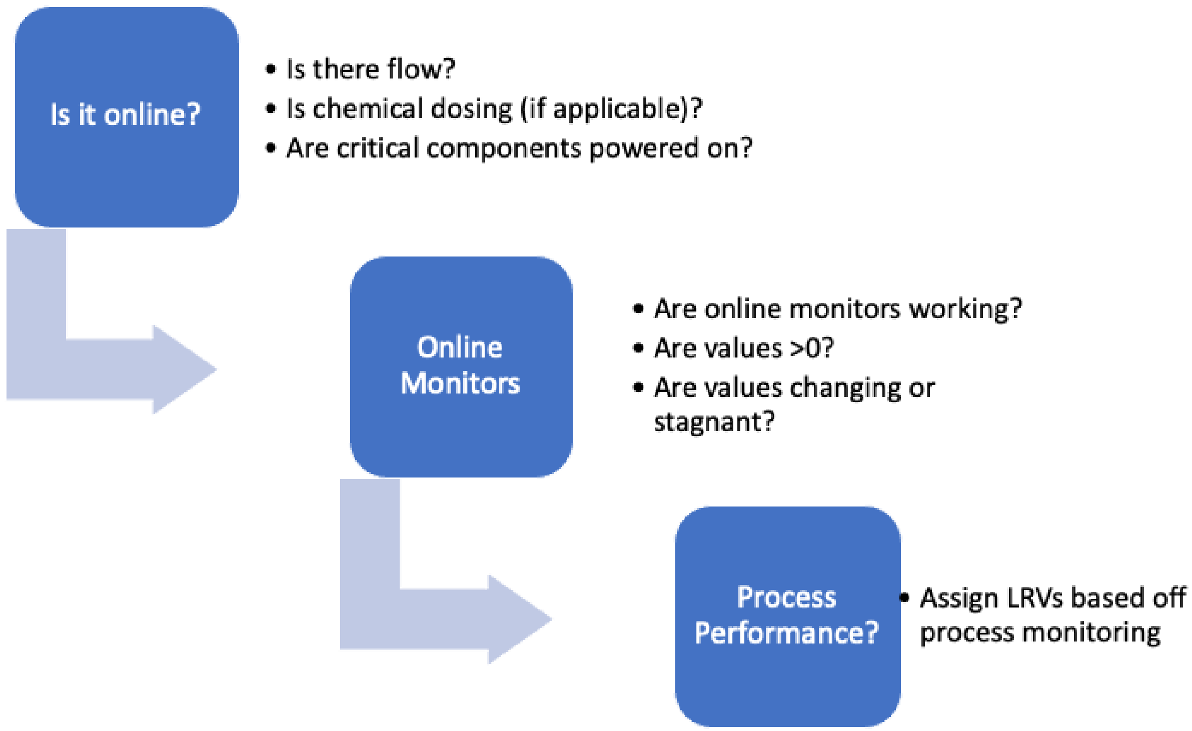


Figure A-2. Three Layers of Filters Used to Evaluate Process Performance Data in WRF 4765.

The first layer of filter was used to determine whether the process was actually online (i.e., "is it online?"). This filter layer helped to remove most of the off-line data by identifying whether basic process functions were being conducted, e.g., that water was flowing through the system, that chemicals were being dosed, and that all critical components were online. Data that passed the first layer went onto the second filter, which provided a rudimentary check to determine whether the meters were online and functional. These types of meter error checks included assessing whether a non-zero reading was being recorded and whether the values were changing. Assuming the first two layers passed, the third layer assigned a LRV based off the process monitoring data. The data filters were capable of removing many data points that were generated during periods when the processes were not in production mode.

Despite the effectiveness of the automated filters, additional manual processing was required to curate the dataset completely. This can be illustrated by looking at the results of the automated MF data filtering (Figure A-3). While the filters were helpful for evaluating if processes were offline, there was still a significant number of suspected water quality events. Any time periods that did not pass the filters were identified with a different colored bar denoting potential failures. As is clear from the figure, there were over a dozen events over a 3-week period that appeared to be potential failure conditions.

However, when operators cross-referenced the time periods with operational event logs and data trends, they found that the apparent "failures" could be attributed to other events occurring at the facility that were not indicative of MF process performance. These included both upstream and downstream events, such as maintenance events on other unit processes that ceased flow to the MF, power outages to the whole facility, or shutdowns to restart other unit processes. Based on the automated filters alone, these could appear to be MF failure conditions with potential public health implications. Manual evaluation of each event was therefore required to separate the false-positives from the data that were truly indicative of MF performance. This example illustrates the importance of **context** in assessing the true performance of the system—knowledge of the performance of other unit processes may be necessary to interpret the data for a given process.

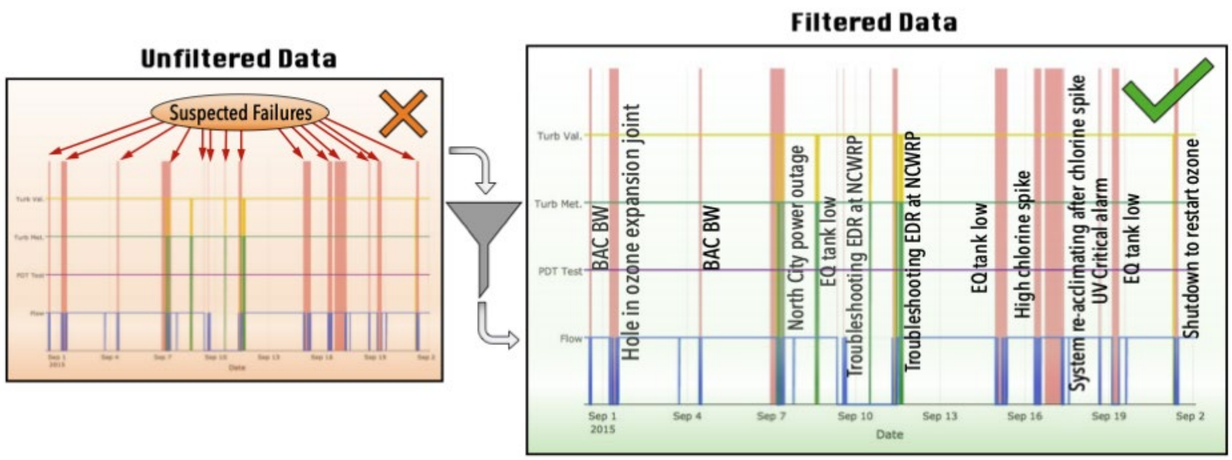


Figure A-3. Results of the Data Filtering from WRF 4765.

Not all of the apparent MF failures, however, were due to issues with other unit processes. Of the types of issues that arose that were specific to the process itself, the majority of suspect events could be correlated with the following scenarios:

1. Meter error/drift resulting in readings that were not representative of process performance
2. Erratic or elevated meter readings during operational changes (particularly flow changes)

Due to the use of redundant meters, the project team was able to evaluate when suspected deviations in process performance were due to meter error. The first scenario was most pronounced in the ozone system, which included two redundant ozone analyzers installed at each of three monitoring locations. Measurement of ozone residual in a wastewater application is challenging and more frequent meter maintenance is required to mitigate against meter error and drift (Chen et al. 2020). Meter readings tended to drift downwards with time due to the buildup of materials from the wastewater matrix on the analyzers. The downward drift of the meters meant lower ozone residuals and resulted in lower pathogen LRVs. However, trending of redundant monitors at the different monitoring locations were used to retroactively determine if the ozone residual truly decreased or if the decrease in reading was due to meter error. This differentiation was done manually by the operations staff. Similar downward drift was observed in the TOC meters used at OCWD in 2017 (Patel and Dadakis 2018).

A number of strategies can be employed to identify meter drift and reduce its impact. The use of redundant meters at a given location can provide evidence of drift when the meters begin to show deviations from each other. It is possible, however, that the use of identical meters at a given location will both drift at the same rate and show minimal variation between each other. To address this, meters with different measurement methodologies can be used since the rate of drift will likely not be equivalent across different meters. For example, the selection of dissolved ozone meters could vary between those using colorimetric methods (such as diethyl-p-phenlyenediamine), amperometric readings, and redox sensors. Another strategy to identify drift is to offset the maintenance schedule for the redundant analyzers, where the more recently maintained analyzer can serve as a back-check against the other meter.

The second scenario (operational changes) caused false-positives to occur frequently because (1) flow changes occur frequently as part of normal operations and (2) system startup occurred frequently due to the operational limitations of the demonstration facility where testing occurred. For example, the reverse osmosis (RO) permeate total organic carbon (TOC) readings are shown in Figure A-4. Abnormally *low* values were observed during shutdowns whereas brief 15-minute period with *elevated* readings were observed during RO start-up. The low readings during RO shut down were the result of a lack of sample flow to the analyzer; the analysis is actually performed on a “dry sample” that is not indicative of the actual RO permeate water quality. The high readings observed during start up are an artifact of the stagnant water that was present in the sample line during the shutdown; it must be flushed out before the RO permeate produced by the RO system is analyzed by the meter. This creates a scenario in which the analyzed sample is not representative of the concentrations in the actual RO permeate being produced. These types of scenarios were confirmed by project engineers who cross-referenced these spikes with RO flow data to evaluate if these spikes coincided with RO system start-up. This trend was observed systematically during every start-up of the RO system.

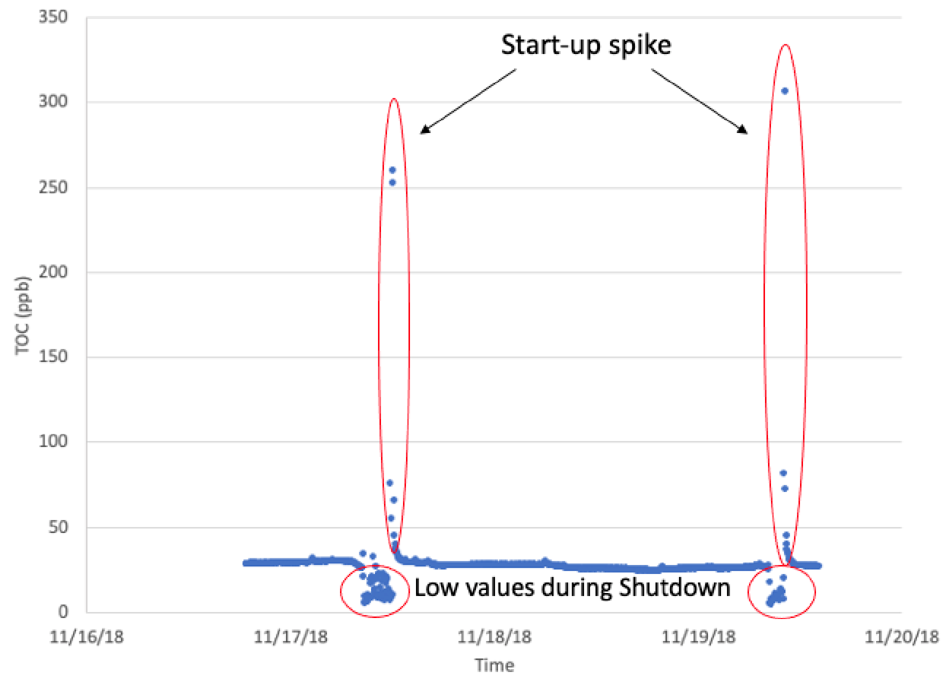


Figure A-4. Example of Deviations in Meter Readings due to Changes in Process Flow.

This manual categorization was performed for all suspected water quality events. Because the two scenarios identified above occurred routinely, a systematic routine for categorizing and confirming these scenarios was developed. With the aid of this routine, the time and effort to manually categorize these events was reduced, though still remained tedious and time-consuming. Certain parameters were affected by these types of scenarios more than others, which generated a large number of trends that needed manual review. For example, the membrane filtration (MF and UF) systems undergo a backwash every 30 minutes and the filtrate turbidity always spiked after a backwash due to sample flow changes to the filtrate turbidimeters[1]. At this frequency of flow changes, many “suspect water quality events” were generated which had to undergo manual review for categorization. The testing site for WRF 4765 only had two membrane filtration trains making the manual evaluation of the data relatively feasible; such an approach would be infeasible with the number of trains typically present at a full-scale facility. This effort highlighted that the data filtering developed for WRF 4765 needs to be improved and further automated to better distinguish the true water quality events and reduce the need for manual filtering.

[1] Typical turbidimeters (e.g., Hach 1720E and Filtertrak 660 sc) measure turbidity by directing a strong beam of collimated light from the sensor head assembly down into the sample in a turbidimeter body. Over time, buildup can form in the turbidimeter body. A sudden change in flow can cause an increase in the turbidity reading for a couple of reasons. A pause in flow followed by a subsequent increase in flow (e.g., after MF comes out of a backwash) can cause sufficient turbulence in the turbidmeter body that it dislodges buildup that interferes with the method and causes an erroneous reading. Air bubbles are also another culprit for erroneous turbidity readings. These can be mitigated by using more recent models (e.g., Hach TU5300sc/TU5400sc) that use a smaller online sample volume and adjusting flow and/or adding bubble trap to prevent bubbles. Such turbidity spikes are identified as “suspect water quality events” and their origin must be understood in order to filter and flag the data.

A.2.3 Implementation of Data Analytics in Industry

Real-time detection of treatment failures and water quality anomalies and automation of system response has been identified as a need to increase the reliability of potable reuse systems. One of the goals of WRF 4954 is to evaluate if there are programs that can be used to detect water quality events. Data analytics can be performed using either proprietary tools or open-source solutions. Recent reviews on the digitalization of the water industry stress the potential danger of "lock-in," meaning that utilities will purchase proprietary systems that cannot communicate freely between platforms, essentially locking them into a single platform (Affleck et al. 2015). To avoid lock-in and increase the transparency and lasting usefulness of this research, the project team limited the scope of the review to open-source solutions. Although open-source tools were explored in this review, the most important component of the data analytics review is not the tools themselves, rather, it is the discussion of the methods that the tools represent. These methods are represented in proprietary software solutions and will continue to be advanced by the private sector and adopted by utilities.

Among open-source products, the project team explored tools for real-time data analysis with sufficient documentation, clear protocols for software testing and maintenance, a wide user base, and peer-reviewed literature associated with the tool. There are two tools created by Sandia National Labs, CANARY and Pecos, with the potential to perform real-time data analytics.

Both CANARY and Pecos were designed to analyze high volumes of sensor data, identify anomalies in the data, create alerts for operators, and improve data quality. Released in 2009, CANARY was developed by the Sandia National Labs in collaboration with the USEPA to detect contamination events in drinking water distribution systems. CANARY (U.S. EPA 2012) uses three water quality event detection algorithms described in the literature: timeseries increment, linear filter, and multivariate nearest neighbor (Klise and McKenna 2006a; b). These algorithms are combined with what is known as a binomial event discriminator (McKenna et al. 2007) to identify suspicious data points. Moreover, the user can choose to run the event detection analysis on water quality sensors individually or as a group. The probability of a true event is evaluated by CANARY by evaluating the series of suspicious data points. If the probability of a true event exceeds a user-defined threshold, CANARY issues an alarm that an event is occurring. To provide additional flexibility, CANARY enables users to develop their own event detection algorithms using MATLAB (U.S. EPA 2012) and more recently, Java (Hall et al. 2017).

There are some known applications of CANARY in industry. For example, CANARY has been integrated into distribution system models by hydraulic modeling vendors for water quality event detection (Hall et al. 2017). One study was published describing the use of CANARY to identify events during normal operations of a decentralized membrane bioreactor (MBR) system and also during simulated failure events (Leow et al. 2017). Failure was simulated by performing sludge bypass events to simulate membrane integrity failure by pumping mixed liquor into the effluent lines of the MBR system. Simulated events were detected by CANARY and were correctly detected as process failures. Alarms detecting process failure were also generated during normal operating conditions and it was found that 89% of the alarms were false positives. Retroactive review (i.e., not real-time and performed manually by a human) of the data by the project team found that the false positives could be attributed to normal MBR operations (backflushes, membrane cleaning, etc.), sensor maintenance and calibration, and change in feed water quality. While sensor signals may have deviated from the normal baseline, all of these events occur routinely, and deviated readings do not represent process failure or unacceptable water quality. In addition, there were also 23 alarms that were generated during the normal operation period in which the causes are unknown. Of these 23 alarms, 13 of the events had trends that were similar with trends of known events, suggesting detection of true events.

More recently CANARY has also been used to identify spill events related to natural gas production (Wickline and Hopkinson 2020). This study however found that the EDS capabilities of CANARY was not suitable for detecting the simulated spills due to size of the spill relative to the watershed size, sensor location, and type of contaminant. This suggests that outliers must exceed a minimum threshold for event detection and may not be sensitive to capture all events.

Released in 2016, Pecos is an open source software for automated performance monitoring of timeseries data (Klise 2018). Pecos was developed by the core team at Sandia National Labs that created CANARY so there is considerable overlap in functionality. The primary purpose of Pecos is to analyze the quality of real-time data streams and generate visualizations and reports to communicate that information. With these reports, users can identify missing, duplicate, or corrupt data, data outside a user-defined range, or abrupt change. Originally developed for the operation of photovoltaic cells, Pecos is industry agnostic and has been incorporated into marine hydrokinetics software (Klise 2018). However, there is currently no peer-reviewed literature that uses it for water quality or treatment process data.

Pecos and CANARY have similar functionalities, and either could be applied for high-frequency data analysis in water reuse. However, there are key differences between these two approaches. CANARY is a user interface application, like Microsoft Word or PowerPoint, which has a static set of features. Pecos, on the other hand, is a software package which provides programmers with a set of tools to choose from. The importance of the flexibility to use Pecos alongside other Python-based tools alongside cannot be understated. High quality, open-source Python packages for data analysis are constantly being developed and distributed freely by academics, nonprofits, and companies, like Facebook and Microsoft.

Python is among the most popular programming languages in the world for engineering applications, software development, and data science. Python is the *de facto* programming language in many industries for data analysis, including environmental engineering. Moreover, its popularity has consistently trended upward in recent years (The Economist 2018). In reviewing open-source Python packages, the project team found several that support anomaly detection, data filtering, machine learning, and statistical modeling for this project that could complement Pecos.

A.2.4 Major Findings of the Review of Current Practices

High-frequency monitoring is needed to assess and confirm the performance of DPR treatment facilities. Due to the massive volume of data produced by AWPFs, however, it is not feasible to manually analyze every data point. This review of current industry practices confirmed that most AWPFs currently in operation utilize a CCP (or HACCP) framework for assessing process performance. Most utilities focus primarily on the monitoring of CCPs (vs. COPs) because they directly impact public health and are often required by regulatory agencies for compliance reporting. Responses from the utilities surveyed revealed certain aspects of existing monitoring/reporting systems that may need additional advancement for DPR applications:

- False-positive water quality events are common. The following strategies are utilized to identify and minimize their impact:
    - Train operators to spot false positives
    - Use alarm delays to minimize false positives
- All utilities surveyed responded that generating compliance reports is mostly automated, but manual analysis is needed to finish reports.
- Static vs. dynamic alarms: two utilities surveyed use static limits exclusively for generating alarms. The third also employ some dynamic, rate-of-change metrics to trigger alarms.

WRF 4765 demonstrated that creating effective filters for CCP performance requires significant effort. Further refinement of the filters developed in WRF 4765 is needed to reduce the number of false positives and eliminate (or greatly reduce) the need for the manual categorization of suspected events. This need for data filtering cannot be overcome simply through the use of a packaged algorithm (like CANARY), since the literature studies reported that these packages ran into similar issues with false positives. The filters should be able to identify a number of scenarios both within a given unit process (e.g., meter drift and operational changes) as well as contextual information (e.g., the impact of failures of upstream and downstream systems). Refinement of filters can be tricky because it is important that the developed filters do not mistakenly categorize true water quality events as false positives. Development of new filters is an iterative approach and requires human review to confirm if categorizations made by filters are accurate.

Based on the review, an enhanced data analytics system would need the following characteristics to address key gaps:

- Filters to address monitoring issues
    - Identify stagnant data
    - Identify missing data
- Filters to address process performance issues
    - Identify data associated with operational changes (e.g., transient spikes in turbidity or TOC during MF and RO start-up, respectively)
- Ability to contextualize the data within a unit process
    - Comparison of one meter to a redundant meter at the same location
    - Comparison of a meter to other meters used to monitor the process
    - Correlations between multiple process variables

- Ability to contextualize the data across the treatment train
- Ability to identify static limits (e.g., fixed regulatory thresholds like effluent TOC) and dynamic limits (e.g., limits based on rate of change)
- Use of statistical process control to assess current performance based on past performance
- Ability to calculate performance based on a hierarchy of surrogates (e.g., assessing RO performance using both TOC and electroconductivity measurements)
- Ability to distinguish actual performance of monitoring events from meter noise
- Ability to overlay instrument calibration data for visualization purposes

The following section provides an overview of the full suite of data processing needs, from the collection and storage of data to analysis, visualization, response, and reporting.

A.3 Key Steps in Data Processing

The overall workflow for data processing is summarized in Figure A-5. After the data is generated, it is stored. A data analytics layer then evaluates the data and refines the data set. After data refinement, the data can be analyzed. Once data has passed through the data analytics layer, the data can be used for display, triggering responses/corrective actions, and output for reporting. Development of the data analytics layer is the main focus for the project and subsequent display and output options will also be evaluated as part of this project. The following sections describe the data analytics layer, display options, and report options for this project.

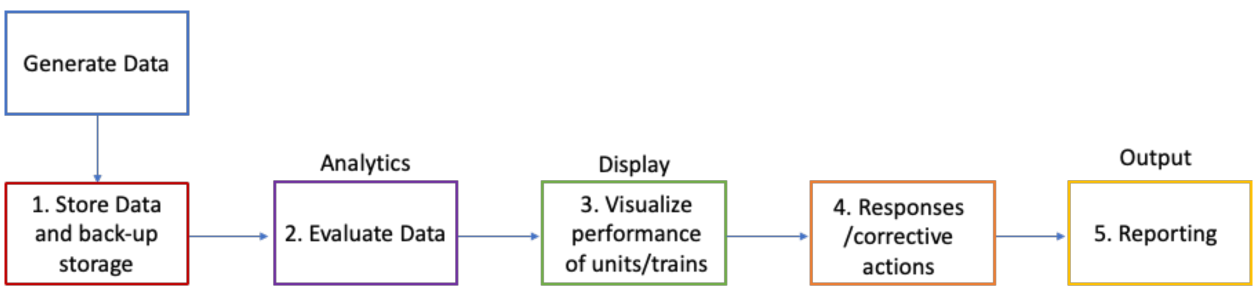


Figure A-5. Overview of Entire Workflow for Data Processing Needs.

A.3.1 Data Analytics Layer

Key issues identified in the review of current industry practices included process-specific data querying, timeliness of warning and response, handling high volumes of data in real-time, improving anomaly/event detection, and evaluating data quality of meters. To address these issues, four types of functionalities needed for the data analytics layer were identified:

- Data quality monitoring and cleaning
- Data filtering
- Anomaly detection / Event detection
- Data point labelling
- Machine learning and statistical modeling (future work)

The data analytics layer, in general, performs evaluation of the raw data set by (1) cleaning the data set, (2) filtering downtime out of the dataset, (3) detecting potential anomalies and water quality events, and (4) triaging the potential anomalies events into respective categories. The workflow for the data analytics layer is presented in Figure A-6.

Figure A-6. Workflow of the Data Analytics Layer.

The following subsections describe the evaluations performed within each category of the analytics layer.

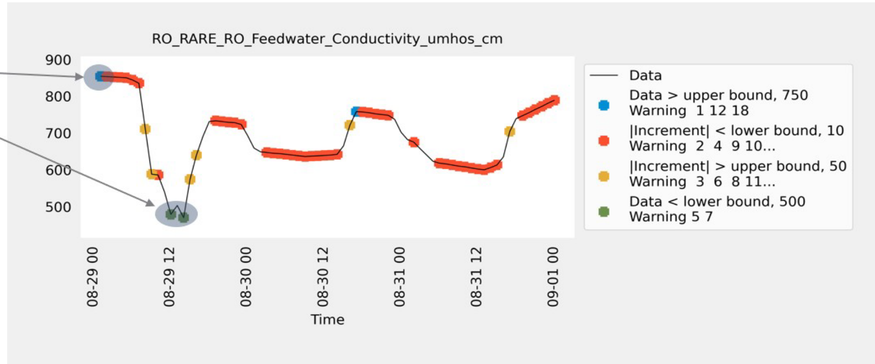
A.3.2 Data Cleaning and Quality Monitoring

The first step, data quality monitoring, serves as the foundation for data analysis because it is used to improve the quality of real-time data streams and identify potential issues with the data. Potential issues with data include stagnant data, meter signals exceeding scaling ranges, data falling outside of historical expected ranges, and abnormal noise in data. Pecos, described above, is an example of a data quality monitoring toolkit. To demonstrate the features of Pecos and how they could be used for high-frequency reuse data, the project team analyzed an illustrative dataset for a reverse osmosis treatment train (shown in Figure A-7). This highlights the package's ability to:

- Check for data that is outside an expected range.
- Check for stagnation in the data.
- Check for abrupt changes between consecutive time steps.

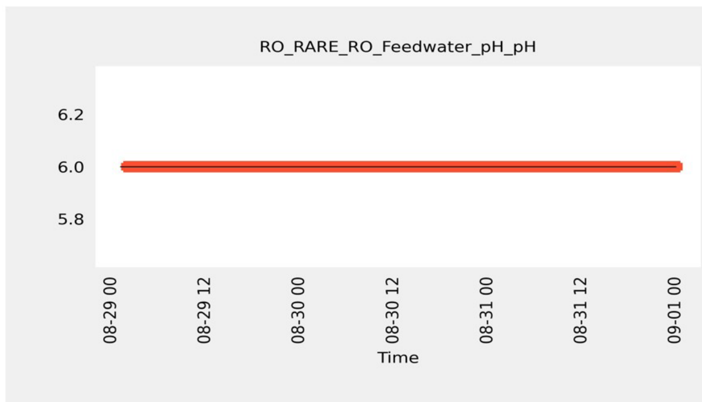
Function 1: pm.check_range([lower, upper], "sensor_tag")
Check for data that is outside expected range

Parameter	Value
Upper conductivity	750
Lower conductivity	500



Function 2: pm.check_delta([Increment lower, Increment upper], duration, "sensor_tag")
Check for stagnant data within a

Parameter	Value
Increment lower	0.0001
Increment upper	0.0001
duration	2 hours



Function 3: pm.check_increment([Increment lower, Increment upper], "sensor_tag")
Check data increments using the difference between values

Parameter	Value
Increment lower	10
Increment upper	50

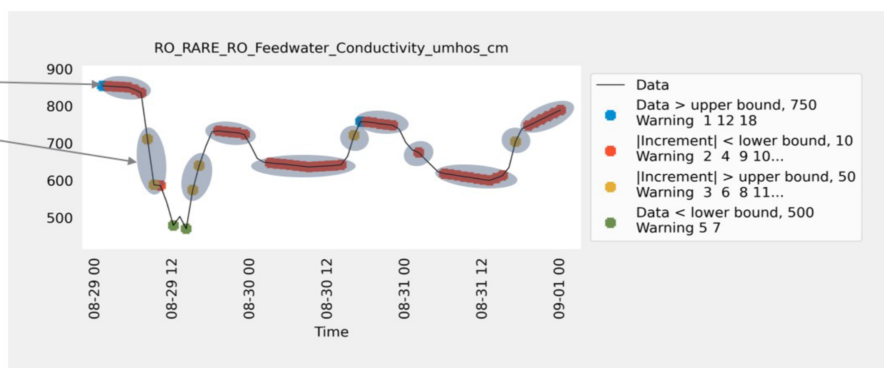


Figure A-7. Demonstration of Three Functions from the PECOS Python Package Using RO Process Data.
(Top) Detection of values outside of a user-specified range of conductivity; (middle) detection of stagnant values from RO pH meter; (bottom) use of rate of change metrics to analyze data stream

The packaged algorithms in Pecos can be augmented with operator experience and knowledge to establish the bounds of expected signal readings. Because some signals are dynamic whereas others are more static, the same bounds cannot be applied to evaluate every signal. For example, the performance of the ozone process is based off many parameters: some—like measured ozone residual—change often whereas others—like ozone gas flow—are relatively stable with little variation. The measured ozone residual signals will need wider bounds to account for real variations whereas the ozone gas flow will need narrower bounds to avoid categorization as “bad quality data” with stagnant values. Each CCP will be evaluated in this way with operator input so that packaged algorithm inputs are informed by operator experience and knowledge.

A.3.3 Data Filtering

After evaluating the quality of the raw data and removing poor quality data, the data should be filtered to remove any points that were generated while the system was offline. Filters were developed in WRF 4765 to remove offline data and could be utilized for this project. Improvements in the monitoring system at the WRF 4954 demonstration test site (North City Demonstration Pure Water Facility) have been made, including the use of status indicators. These indicators signify whether the system is online or offline, eliminating the need for some of the rudimentary filters used in WRF 4765. These improvements will be used in tandem with additional filters from WRF 4765 to filter out off-line data and identify when the system has reached steady-state after starting up.

A.3.4 Event Detection

To improve detection and response to process failures in water reuse, anomaly detection algorithms can be applied. Two functions previously illustrated in **Error! Reference source not found.** include out-of-range and rate-of-change alarms. While useful at identifying many events, these more basic alarms may fail to identify systemwide failures or anomalies. To provide this additional functionality, Python-based multivariate anomaly detection algorithms, like those used in CANARY, can consider groups of sensors for real-time data analysis. Most of these techniques are industry agnostic and were developed for applications such as detecting credit card fraud, cybersecurity breaches, and identifying malfunctions in industrial control systems (Bartos et al. 2019). Available Python packages include *rrcf* (Robust Random Cut Forest) algorithm for anomaly detection on high volume streaming data (Bartos et al. 2019; Guha et al. 2016) and *PyOD*, a library for outlier detection and anomaly detection in multivariate data.

Another option to identify events or anomalies is through the use of timeseries analysis methods such as control charts (Kaelin et al. 2008; Nilsson et al. 2007). A study by Nilsson et al. (2007) identified two methods for identifying changes in process performance including the Shewart method and cumulative sum control charting (CUSUM). Both rely on statistical process control methods that identify outliers based on the comparison of current performance to the mean (or other similar statistic) and a control limit boundary. The control limit boundary can either be set as a fixed value or statistically estimated based on historical performance. For example, the probability of a process falling outside of a control limit that is three standard deviations from the mean is only 0.3%. Thresholds could be set to determine when a system is no longer in control, e.g., nine consecutive readings outside of the range. OCWD proposed criteria for control of the RO process based on fifteen consecutive about the upper control limit (UCL) (Dadakis 2014). The use of multiple consecutive readings helped them to avoid responding to isolated outliers. The benefit of the statistical approach is that it provides system-specific thresholds that may allow deviations to be identified while not triggering an excessive amount of false-positive readings. It also can provide advanced warning of an upset that occurs in advance of the process crossing a regulatory threshold.

Nilsson et al. (2007) noted that the effectiveness of this approach relies on having high-quality monitoring data. It was easier to identify potential hazardous events if the performance data were of high quality, meaning that staff could rule out the possibility that the event was simply a meter issue rather than a true performance issue. Through the process, they identified both hazardous and non-hazardous events (e.g., cleanings) using the statistical process control methods, and were able to identify the frequency, duration, and magnitude of a number of hazardous events. Nevertheless, the researchers concluded that the manual process of evaluating SCADA data was time-consuming, and that future efforts should seek to automate the maximum extent of the data analysis as possible.

A.3.5 Data Point Labelling

One method that will be used by the WRF 4954 project team to facilitate the data filtering process is the use of unique signifiers to identify *how* points have been categorized using the filters. This “flagging” or “labeling” of each data point will allow project team members to more easily evaluate the accuracy of the refined filters. In addition to filter development, labelled data points can be a useful tool for reporting. For example, when performing statistics on a dataset, all nonrepresentative data should be removed to prevent skewing results. Current practice relies on the manual identification of outliers and the determination of whether the outliers are representative of process performance. If the outliers are not characteristic of process performance, their causes must be identified. With the use of data labels, the categorization process is automated, and the end user can rapidly perform statistics on a data set excluding nonrepresentative data points. Data labelling also satisfies the need for transparency in that it can help ensure that the evaluated dataset has not selectively “cherry-picked” only the favorable datapoints. Lastly, true water quality events are also labelled. Because they have already been identified, trending can be performed on these datasets to evaluate the best course of action for responding to these events and also preparing prevention plans. The three main categories used for data labelling are: (1) confirmation of anomalies/events, (2) changes in water quality due to process changes but not process failure, and (3) meter error that is nonrepresentative of process performance.

A.3.6 Future Functionalities (Machine Learning and Statistical Modeling)

Beyond the tools described above, there is a wide range of machine learning and statistical modeling Python packages that can be used to analyze water reuse data, including regression, classification, clustering, and general statistics. Popular packages for these applications include *scikit-learn* (Pedregosa et al. 2011), *TensorFlow* (Abadi et al. 2015), and *SciPy* (Virtanen et al. 2020). For forecasting time series data, Facebook’s Core Data Science team openly released *Prophet*, which is used across many of Facebook’s applications (Taylor and Letham 2017). Other methods include merger of instrumentation operational controls developed with Fuzzy Logic/Neural Network Principles and Numerical Method/Statistical Modeling.

The above techniques could be used in conjunction with Pecos or as standalone analyses to support the development of a data analytics layer for water reuse. Based on the review of data monitoring and analyses currently in use in industry, certain bridges need to be addressed before moving into machine learning and statistical modeling. Machine learning and statistical modeling would be better implemented in future projects after refining how meter signal deviations are categorized (i.e., water quality anomaly vs. meter error and flow changes).

Normal process changes, sensor drift, and other data quality issues can cause failure of treatment system that goes undetected. These measurement errors may be difficult to identify based on a single sensor or even a group of sensors without additional analytics to verify the validity of the measurement. Data filters, such as Kalman filters, are a promising method to address this problem. Water sector applications include modeling distribution system water quality and urban drainage systems (Bartos and Kerkez 2020; Rajakumar et al. 2019) and estimating nutrient composition in pilot wastewater treatment plants (Nair et al. 2019). FilterPy (Labbe Jr. 2015a) is a popular Python package for implementing data filtering techniques. The package is well documented and has a companion book that is also free and open source, "Kalman and Bayesian Filters in Python" (Labbe Jr. 2015b). These tools can be used to evaluate individual sensor signals and also be supplemented with custom Python code to cross-reference other parameters for confirmation of true anomalies/events vs other causes like normal process changes and meter errors.

A.4 Visualization of Process Data and Alarms

For best practices in data visualization and alarms for SCADA, the project team reviewed the standards from ANSI/ISA-101 (Human Machine Interfaces for Process Automation Systems), also known as high-performance HMI (Hollifield et al. 2008). The basic principles are as follows:

- Show information on the screen, not raw data.
- Show clearly if the process is running well or not.
- Use depiction methods that make abnormal conditions visible and alarms stand out.
- Organize the graphics in hierarchical layers to progressively show more detail and information as needed.
- Use color consistently and effectively.
- Display current data with desired operating range and alarm thresholds.
- Use of low contrast colors unless to show an alarm.
- Display hierarchy to drill into different levels of detail.

To implement these principles, novel visualizations can be used to show multiple attributes and enable operators to rapidly view system-wide "health." Visuals should represent results of complex data analytics and distill large data sets into visuals that can be interpreted quickly and easily. There are two levels for evaluating performance of a reuse facility: the individual process level and the overall treatment train level. The "status" of each of these levels will need to be presented visually on screens. The following types of visuals may be utilized for representing statuses of the two levels:

- Gauges

- Radar Chart
- Dashboards

The first style of visuals are gauges. Visualization on process/train "status" can be both quantitative and qualitative. Regions of the gauge can be split into three regions: low risk, medium risk, and high risk and signified by colors. An example of gauge representations is shown in Figure A-8. A needle/slider is used to indicate current status of system, updates dynamically, and provides context of how close system is to sliding into higher risk ranges.

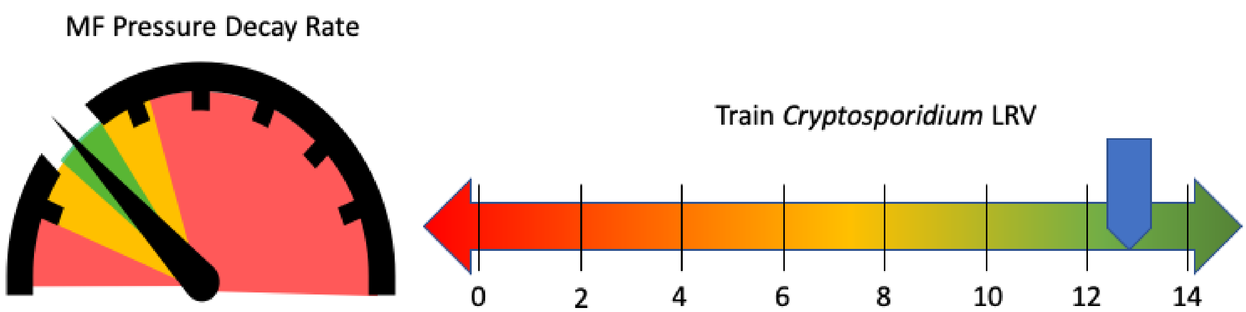


Figure A-8. Example of Gauges to Represent Unit Process and Treatment Train Status.

Radar charts are a graphical method for displaying multiple variables on a single visual and can be used to visualize risk. Radar charts can represent both quantitative and qualitative data. Radar charts show the risk for each parameter and the status of each parameter may have different level of risks. The overall risk is considered by balancing the individual risk from each parameter and is represented with colors to signify low risk, medium risk, and high risk. Examples of how radar charts can be used to visualize risk for different parameters at a reuse facility is shown in Figure A-9. In the figure below, the visualization on the left depicts parameters in an ozone system and shows "no risk" for the UVT removal and generator status parameters, "medium risk" for the residual ozone parameter, and "high risk" for nitrite and meter maintenance. The overall aggregate risk for the ozone system is considered medium after balancing the risk for all the parameters shown.

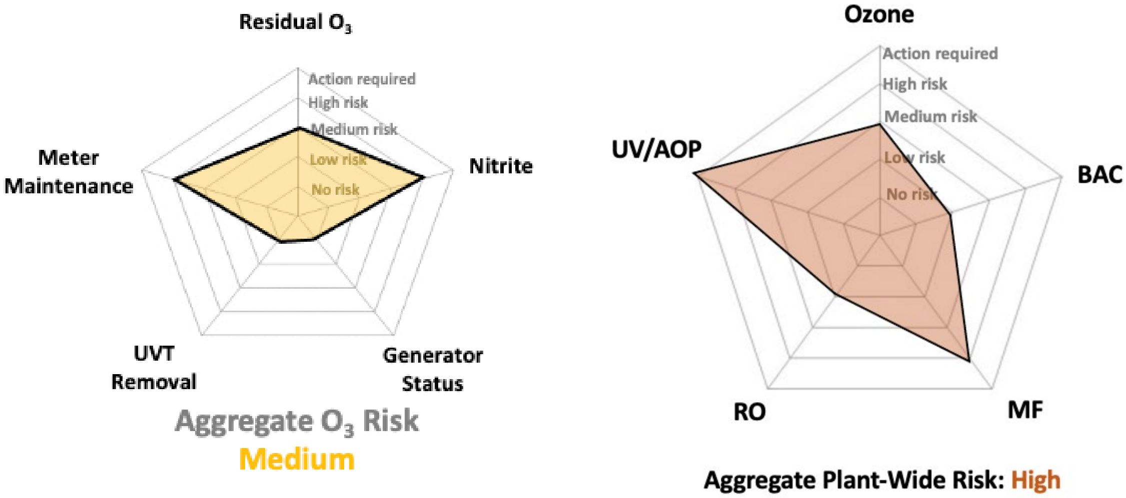
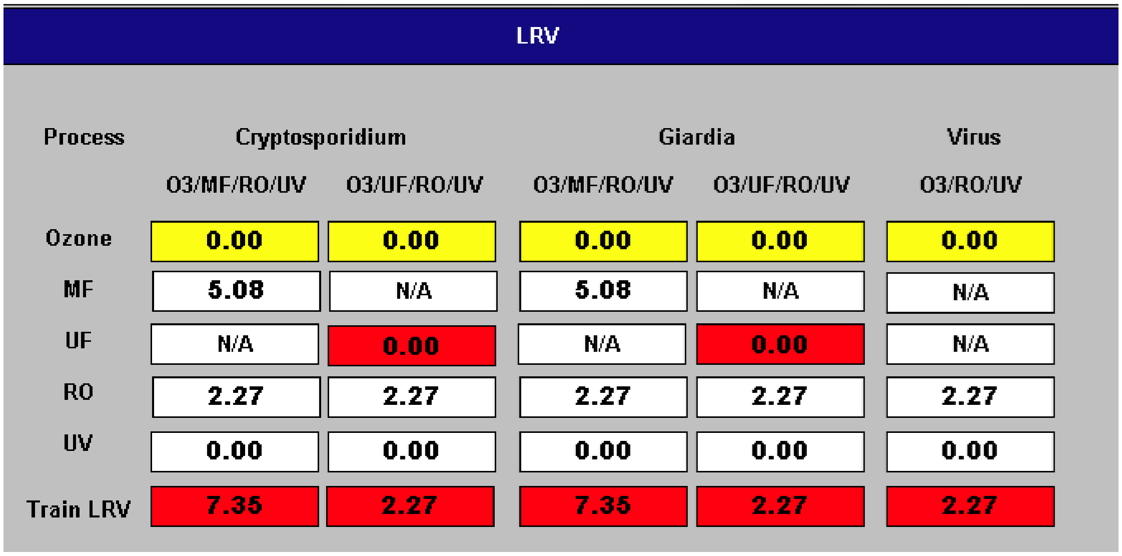


Figure A-9. Examples of Radar Charts for Visualizing Performance Parameters for A Unit Process (Left) and A Treatment Train (Right) in A Reuse Facility.

The last form of visuals considered for this project is dashboards. Dashboards can incorporate both quantitative and qualitative data. Dashboards can be used to visualize the status of several parameters using colors to signify low risk, medium risk, and critical risk. Examples of different types of dashboards are presented in Figure A-10 and Figure A-11. Figure A-10 shows various areas of the plant and an overall status for that area is represented. In this example, “Operational Integrity” is at a critical risk and the “Event” detection is at medium risk. All other aspects are low risk. This would help operators visualize which aspect of the plant requires attention. Figure A-11 shows the pathogen LRV achieved by each individual unit process and the cumulative train LRVs for *Cryptosporidium*, *Giardia*, and virus. If a LRV has medium or high risk, the parameter is highlighted in yellow and red, respectively.

Figure A-1. Example of Qualitative Dashboard Depicting Status of Different Areas of Plant Monitoring.



LRV					
Process	Cryptosporidium		Giardia		Virus
	O3/MF/RO/UV	O3/UF/RO/UV	O3/MF/RO/UV	O3/UF/RO/UV	O3/RO/UV
Ozone	0.00	0.00	0.00	0.00	0.00
MF	5.08	N/A	5.08	N/A	N/A
UF	N/A	0.00	N/A	0.00	N/A
RO	2.27	2.27	2.27	2.27	2.27
UV	0.00	0.00	0.00	0.00	0.00
Train LRV	7.35	2.27	7.35	2.27	2.27

Figure A-2. Example of Quantitative Dashboard Depicting Pathogen Log Removal Values.

The project will explore which visualization options are most effective for assessing the status of potable reuse facilities.

A.5 Responses and Corrective Actions

System responses and corrective actions are based on results from the analytics layer. Before implementing real-time control changes, it is important to first fully vet the accuracy of the analytics layer. Another restriction for this project is that the testing site is a demonstration facility with less flexibility to make operational changes. Nevertheless, the system will be designed so that messages will be generated with a description of what actions would be taken if the system were fully automated. This allows the project team to evaluate if generated responses are appropriate and to iterate on the control system before implementation at an actual full-scale facility.

For alarm priorities, both water reuse-specific practices described in WRF 1700 (Response Procedures section) and ANSI standards will be considered, such as ANSI/ISA-18.2 (Management of Alarm Systems for the Process Industries) (ANSI/ISA, 2009; 2015).

A.6 Automated Compliance Reporting

Another potential challenge for direct potable reuse projects is implementation of automated compliance reporting. The reporting requirements are subject to regulator approvals and operating permits can vary based on the facilities actual treatment process train and integration with the drinking water treatment and distribution system. For indirect potable reuse, compliance reporting is typically filed monthly. Based on the utility surveys, this turnaround time frame does not require as much automation to query the reporting data and compile it into specific templates. This task is typically performed manually by staff rather than automated. Tools such as Microsoft Excel are used for compiling the daily max and average values for each day for reporting period. Data is reviewed by a person experienced with the treatment process control to spot the anomalous spikes and decide whether a redundant data source is available or if the anomaly was due to downtime or maintenance activities. This is similar to the function of data filtering and event detection as discussed in previous sections and is somewhat of an indirect and reactive approach. Similar to how the real-time control must respond quickly enough with corrective actions, the compliance reporting also needs to be assessed more frequently, including daily, weekly, and monthly reports that can demonstrate compliance quickly and with desired time frames. There are some commercial platforms that provide customization of compliance reporting such as the Dream Report® by Ocean Data Systems (Ocean Data Systems 2017). Pecos can also be used for reporting and together with the data analytics provide the necessary data preparation for the purposes of compliance reporting. The processed data by Pecos, can be queried over custom time frame and range of process specific CCPs. Custom reporting templates would need to be created. Dream Report® essentially provides a platform that bridges that gap by offering user-friendly customizable report templates and interface with the SCADA data.

A.7 Conclusions and Next Steps

DPR systems need the ability to collect and process high volumes of performance data in real-time. This requirement stems from the need to identify and respond to potential public health issues in the short period of time between the treatment and distribution of purified waters. The overall workflow for processing data involves many steps: 1) collection and storage of data, 2) data analysis, 3) visualization of performance, 4) response actions, and 5) reporting. Through the literature review, the key step in the process that requires the most significant advancement for DPR is the data analytics layer.

Through the literature review, a number of functionalities were identified for the enhanced data analytics layer. Existing open-source tools like CANARY and Pecos were reviewed and identified as potential options for many of these functions. Other site-specific tools may be better served through programming in the Python core package. Table A-2 includes a list of the desired data analytics functionalities along with an evaluation of the platforms that could be leveraged to achieve these goals. While no single platform is best suited to perform all of the functions needed for the layer, the use of multiple tools in conjunction can achieve these needs. The integration of these tools into a single data analytics layer will be the main goal of Task 2 of WRF 4954.

Table A-2. Desired Functionality of Data Analytics Layer and Options Identified through Literature Review.

Functionality	Python Core Package	Pecos	PyOD/rrcf
Identify stagnant data	X	X	
Identify missing data	X	X	
Identify non-representative data during flow changes	X		
Comparison of redundant meters		X	X
Integration of multiple performance meters		X	
Contextualization across treatment train		X	X
Use of static and dynamic limits	X	X	
Statistical process control	X	X	
Event detection		X	X

Appendix A References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jozefowicz, R., Jia, Y., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Schuster, M., Monga, R., Moore, S., Murray, D., Olah, C., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y. and Zheng, X. 2015. *TensorFlow: Large-scale machine learning on heterogeneous systems.* USENIX Association, 12th USENIX Symposium on Operating Systems Design and Implementation 265-278.

Affleck, M., Allen, G., Bozalongo, C.B., Brown, H., Gasson, C., Gasson, J., González-Manchón, C.,

Hudecova, M., McFie, A., Shuttleworth, H., Tan, M., Underwood, B., Uzelac, J., Virgili, F. and Walker., C. 2015. *Water's Digital Future*. Global Water intelligence.

ANSI/ISA 2009. Management of Alarm Systems for the Process Industry (ANSI/ISA-18.2).

ANSI/ISA 2015. Human Machine Interfaces for Process Automation Systems (ANSI/ISA-101.01).

Bartos, M. and Kerkez, B. 2020. Pipedream: an interactive digital twin model for urban drainage networks. EarthArXiv.

Bartos, M., Mullapudi, A. and Troutman, S.C. 2019. "Implementation of the Robust Random Cut Forest algorithm for anomaly detection on streams." *Journal of Open Source Software*, 4(35): 1336.

Chen, E.C., Pisarenko, A.N., Kolakovsky, A., Howe, E.W., Trussell, R.S. and Trussell, R.R. 2020.

"Evaluation of Four Dissolved Ozone Residual Meters' Performance and Disinfection Credits in Potable Reuse Applications". *Ozone-Science & Engineering*, 42(3): 213-229.

Dadakis, J. 2014. *Demonstration of GWRS process performance*, San Diego, CA. Expert Panel Presentation, December 9, 2014.

Guha, S., Mishra, N., Roy, G. and Schrijvers, O. 2016 Robust Random Cut Forest Based Anomaly Detection On Streams. *33rd International Conference on Machine Learning.* New York, NY.

Hall, J., Panguluri, S., Murray, R. and Burkhardt, J. 2017. "CANARY Eases Water Quality Event

Detection". *Opflow*, 43(4): 30-32.

Hollifield, B., Habibi, E., Oliver, D., Nimmo, I. and Renner, K. 2008. The High Performance HMI Handbook: A Comprehensive Guide to Designing, Implementing, and Maintaining Effective HMIs for Industrial Plant Operations.

Kaelin, D., Rieger, L., Eugster, J., Rottermann, K., Banninger, C. and Siegrist, H. 2008. "Potential of in-situ sensors with ion-selective electrodes for aeration control at wastewater treatment plants". *Water Science and Technology*, 58(3): 629-637.

Klise, K.A. 2018 *Pecos - Open Source Software for PV System Monitoring, Sandia National Lab.* (SNL-NM), Albuquerque, NM (United States).

Klise, K.A. and McKenna, S.A. 2006a. Multivariate applications for detecting anomalous water quality. *8th Annual Water Distribution Systems Analysis (WDSA) Symposium*. Cincinnati, OH.: ASCE

Klise, K.A. and McKenna, S.A. 2006b. Water quality change detection: multivariate algorithms.

SPIE Defense and Security Symposium. Orlando, FL.: SPIE

Labbe Jr., R.R. 2015a. *FilterPy*, GitHub.

Labbe Jr., R.R. 2015b *Kalman and Bayesian Filters in Python*, GitHub.

Leow, A., Burkhardt, J., Platten, W.E., Zimmerman, B., Brinkman, N.E., Turner, A., Murray, R., Sorial, G. and Garland, J. 2017. "Application of the CANARY event detection software for real-time performance monitoring of decentralized water reuse systems." *Environmental Science-Water Research & Technology,* 3(2): 224-234.

McKenna, S.A., Hart, D.B., Klise, K.A., Cruz, V.A. and Wilson, M.P. 2007. Event Detection from Water Quality Time Series. *ASCE World Environmental and Water Resources Congress*. Tampa, FL.: ASCE

Nair, A.M., Fanta, A., Haugen, F.A. and Ratnaweera, H. 2019. "Implementing an Extended Kalman Filter for estimating nutrient composition in a sequential batch MBBR pilot plant". *Water Science and Technology*, 80(2): 317-328.

Neemann, J., DeCarolis, J., ten Bosch, D., Snyder, S.A., Pepper, I.L. and Minkyu, P. 2019. *Integrating Management of Sensor Data for a Real Time Decision Making and Response System.* Water Environment & Reuse Foundation.

Nilsson, P., Roser, D., Thorwaldsdotter, R., Petterson, S., Davies, C., Signor, R., Bergstedt, O. and Ashbolt, N. 2007. "SCADA data and the quantification of hazardous events for QMRA". *J. Water Health*, 5: 99-105.

Ocean Data Systems. 2017. Ocean Data Systems: The Art of Industrial Intelligence. Accessed February 13, 2024. https://dreamreport.net.

Patel, M. and Dadakis, J. 2018. *Practical operational approaches to meet pathogen LRV requirements for GWRS*, Monterey, CA.

Pecson, B.M., Triolo, S.C., Olivieri, S., Chen, E.C., Pisarenko, A.N., Yang, C.C., Olivieri, A., Haas, C.N., Trussell, R.S. and Trussell, R.R. 2017. "Reliability of pathogen control in direct potable reuse: Performance evaluation and QMRA of a full-scale 1 MGD advanced treatment train". *Water Research*,

122: 258-268.

Pecson, B.M., Chen, E.C., Triolo, S.C., Pisarenko, A.N., Olivieri, S., Idica, E., Kolakovsky, A., Trussell, R.S. and Trussell, R.R. 2018. "Mechanical Reliability in Potable Reuse: Evaluation of an Advanced Water Purification Facility". *Journal - American Water Works Association*, 110(4): E19-E28.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M. and Duchesnay, É. 2011. "Scikit-learn: Machine Learning in Python". *Journal of Machine Learning Research*,

12: 2825-2830.

Rajakumar, A.G., Kumar, M.S.M., Amrutur, B. and Kapelan, Z. 2019. "Real-Time Water Quality Modeling with Ensemble Kalman Filter for State and Parameter Estimation in Water Distribution Networks". *J. Water Resour. Plan. Manage.-ASCE*, 145(11).

Snyder, S.A. and Pepper, I.L. 2016. *Monitoring for Reliability and Process Control of Potable Reuse Applications.* Water Environment & Reuse Foundation.

Tackaert, R.A., Pisarenko, A.N., Chen, E.C., Kolakovsky, A., Pecson, B.M., Drewes, J.E., Trussell, R.R. and Trussell, R.S. 2019. "Demonstrating process robustness of potable reuse trains during challenge testing with elevated levels of acetone, formaldehyde, NDMA, and 1,4-dioxane". *Journal of Water Supply: Research and Technology – AQUA*, 68(5): 313-324.

Taylor, S.J. and Letham, B. 2017. *Forecasting at scale*. PeerJ Preprints 5.

The Economist. 2018. *Python is becoming the world's most popular coding language.*

Trussell, R.S., Pecson, B.M., Pisarenko, A.N., Idica, E.Y., Howe, E.W. and Trussell, R. 2017 Demonstrating redundancy and monitoring to achieve reliable potable reuse: Final Report for WE&RF Project 14-12. WE&RF (ed), Alexandria, VA.

U.S. EPA 2012. *CANARY User's Manual version 4.3.2*. EPA/600/R-08/040B.

Virtanen, P., Gommers, R., Oliphant, T.E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., Walt, S.J.v.d., Brett, M., Wilson, J., Millman, K.J., Mayorov, N., Nelson, A.R.J., Jones, E., Kern, R., Larson, E., Carey, C., Polat, İ., Feng, Y., Moore, E.W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E.A., Harris, C.R., Archibald, A.M., Ribeiro, A.H., Pedregosa, F., Mulbregt, P.v. and Contributors, S. 2020. "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python". *Nature Methods*, 17(3): 261-272.

Walker, T., Stanford, B.D., Khan, S., Robillot, C., Snyder, S.A., Valerdi, R., Dwivedi, S. and Vickers, J. 2016. *Critical Control Point Assessment to Quantify Robustness and Reliability of Multiple Treatment Barriers of a DPR Scheme.* Water Environment & Reuse Foundation.

Wickline, J. and Hopkinson, L. 2020. "Detection of spills related to natural gas production". *Water Environ. Res.*, 92(8): 1104-1110.

APPENDIX B

Additional Configuration Considerations at NCPWDF

This section provides additional information surrounding the development of the event detection logic and criteria, including approaches that were tested but not recommended for implementation. Beyond the finalized Python script used for event detection, the efforts of the project team to develop the EDS resulted in considerations that could be useful for future applications.

B.1. Additional Considerations for Ozone Event Detection

B.1.1 Water Quality Event Detection Logic

The preliminary event detection logic for an ozone water quality event was defined by low residual readings at both OSP 4 and 7 coinciding with a decrease in *Cryptosporidium* LRV. The logic being that if these meters are reading low and the LRV is declining, the consistent trend across monitoring locations can rule out a monitoring point failure therefore the drop in LRV is indicative of a water quality event.

To develop the ozone residual and LRV operating range bounds, statistical analysis was performed on historic data. Ozone dose varied historically in response to fluctuating water quality which caused there to be a lot of noise in the ozone residual data. The consequence of noisy data is that the operating range bounds determined using 2 standard deviations from the mean were too wide to accurately evaluate abnormal ozone residual data.

Figure B-1 shows data for a known water quality event at the NCPWDF from September 2022. The event detection was delayed using this initial detection logic that relied on the relationship between ozone residual and *Cryptosporidium* LRV. The dashed red line indicates when the *Cryptosporidium* LRV decreased below the early detection threshold of 1.1, yet the event was not detected until the ozone residual meters decreased below their low operating range thresholds. The ozone residual at OSP 7 and *Cryptosporidium* LRV were both flagged for a low value around a similar time, but the ozone water quality event was configured to only be detected if the ozone residual at OSP 4 was also below its minimum threshold. This did not occur until approximately 2 hours later leading the lag in event detection. This delay would not be useful for an online monitoring system and did not meet the goals for the tool to increase lead time and reduce response time.

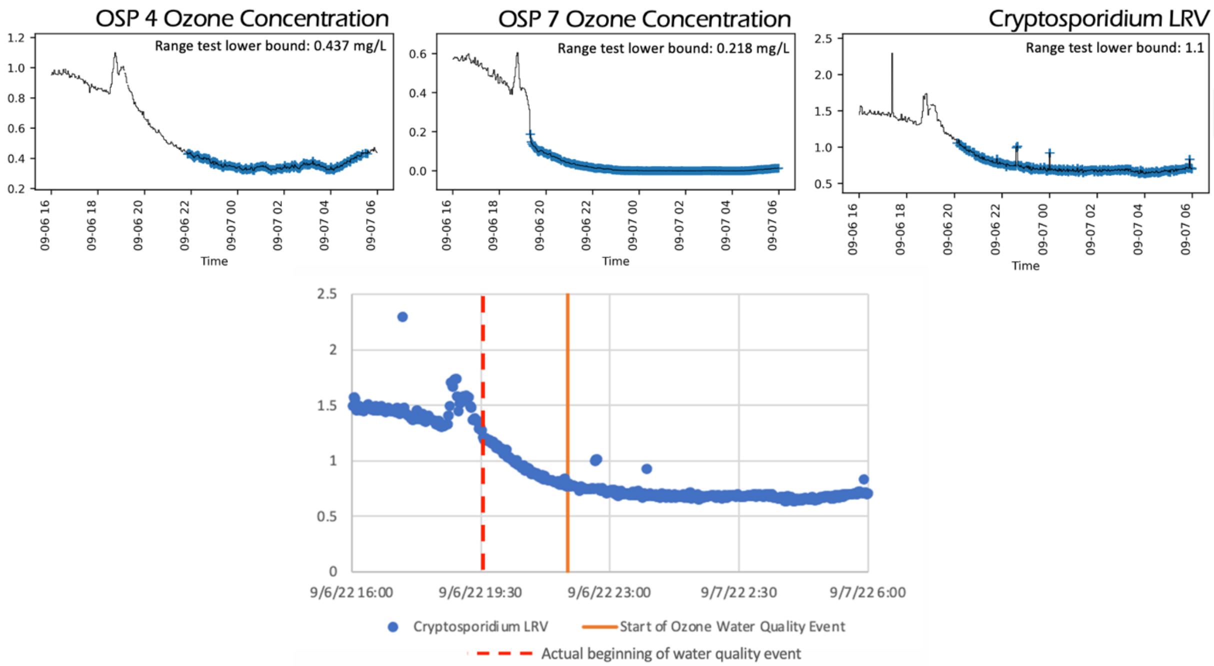


Figure B-1. September 6, 2022, Water Quality Event Detected Late Using Positive Identification Logic.

The approach of identifying water quality changes based on a decrease in ozone concentration throughout the contactor was determined to be too targeted based on the late detection described above. The implemented logic defined in section 4.3.3.3 focuses on ozone demand instead of ozone residuals. It was found to be more accurate and better at early detection of ozone water quality events because it first checks for indications of process and monitoring point failures before identifying the event as water quality related.

B.1.2 Monitoring Point Event Detection Parameters

Monitoring error event detection is configured to detect events based on the percent difference between two meters at a location exceeding 15%. As residual concentrations decrease, it becomes more likely that the percent threshold will be exceeded due to fluctuations in ozone residual measurements. When changes in water quality increase ozone demand, the ozone residual measurements at OSP 7 can approach zero which may be falsely detected as a monitoring point failure.

For example, if a chemical peak entering the contactor causes a steady decline in ozone residual, the two meters at OSP 7 could read on the order of 0.03 and 0.01 resulting in a calculated % difference equal to 67%, and a monitoring point failure will be falsely detected. Therefore, as the tool is currently developed, operations should take care when considering a situation where both a water quality event and a meter error event are occurring simultaneously. Ozone dose would be a conservative corrective action to take while determining which of the events is truly occurring because if the meter error returns below 15% soon after, the event was driven by water quality.

B.2 Additional Considerations for MF Event Detection

B.2.1 Monitoring Point Event Detection Threshold Adjustments

During framework development, certain qualities specific to the MF process presented challenges that are outlined in this section. Under normal operations, MF filtrate turbidity readings only fluctuate in small increments. This inherent lack of variability in the data initially led to false alarms for stagnant data (Figure B-2) when in actuality it was a prolonged stretch of stable readings as illustrated by the example in Figure B-3 when the stagnant threshold was set at 0.001.

Stagnant Filtrate Turbidimeter

Data start time: 2023-07-06 15:30:00
Data end time: 2023-07-06 16:59:00
Number of variables: 5
Number of test failures: 3

Test Results:

	Variable Name	Start Time	End Time	Timesteps	Error Flag
1	MF Filtrate Turbidity (NTU)	2023-07-06 15:36:00	2023-07-06 16:04:00	29	|Increment| < lower bound, 0.001
2	MF Filtrate Turbidity (NTU)	2023-07-06 16:11:00	2023-07-06 16:39:00	29	|Increment| < lower bound, 0.001
3	MF Filtrate Turbidity (NTU)	2023-07-06 16:45:00	2023-07-06 16:59:00	15	|Increment| < lower bound, 0.001

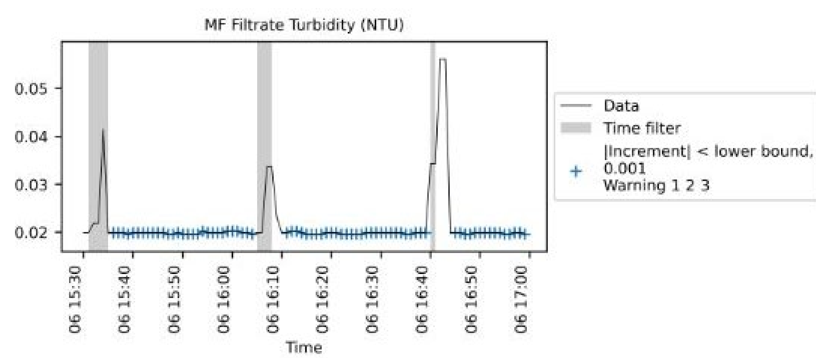


Figure B-2. False Identification of a MF Monitoring Point Event Due to an Overly Sensitive Detection Threshold.

The Pecos quality control test parameters that can be adjusted to achieve the desired sensitivity are the stagnant data threshold and the minimum number of consecutive failures requirement. To address the oversensitive nature of the stagnant filtrate turbidity test, the stagnant threshold was decreased from 0.001 to 0.0001 while the minimum number of consecutive failures remained at 15 minutes.

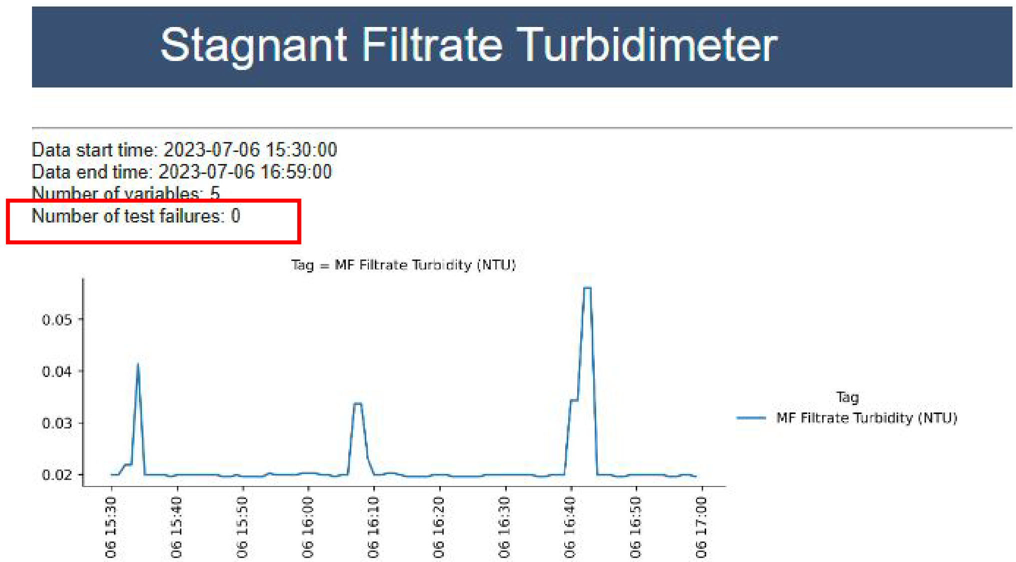


Figure B-3. Correct Assessment of MF Filtrate Turbidimeter Data Due to Appropriate Detection Threshold Sensitivity.

B.2.2 MF BW Considerations

Another finding from MF filtrate turbidity data analysis was that temporary spikes were observed during the first 2 to 3 minutes following each BW (Figure B-4). Regardless of whether or not these spikes went above the high filtrate turbidity threshold of 0.15 NTU, the EDS is designed to not generate an alert because the duration of the spike is less than the minimum number of consecutive failures (15 minutes). This ensures that only instances of sustained high filtrate turbidity are identified as an MF process failure event.

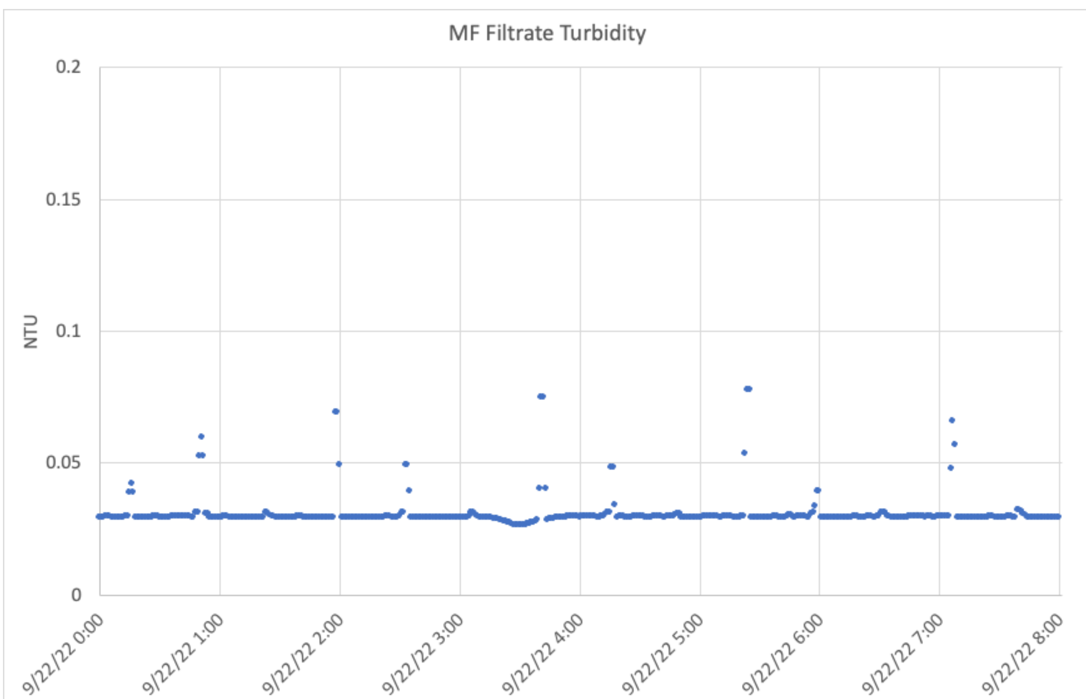


Figure B-4. MF Filtrate Data Showing Stability and Impact of Backwashes.

B.3 Additional Considerations for RO Event Detection

B.3.1 RO CCP Monitoring Parameter Test Bounds

RO systems typically exhibit stable performance without major fluctuations in the permeate water quality. However, as the membranes age over time, gradual increases in permeate TOC and EC can be observed which reduce the LRV achieved by the system. The upper and lower bounds for the Pecos range tests used in event detection can account for this expected decline in performance by drawing from historical data that captures these expected trends to inform the test bounds.

B.3.2 RO Monitoring Point Event Detection Logic

Another consideration is the approach for assessing meter accuracy without a redundant meter present for comparison. The event configuration for RO monitoring point events utilized multiple Pecos tests on both measured and calculated tags. The feed and permeate concentrations of TOC and EC are used to calculate respective LRVs. Since these parameters are expected to remain relatively constant during operation, a low LRV reading combined with a TOC or EC measurement outside of the normal operating range at only one of the feeds and permeate monitoring locations would indicate that the meter has potentially lost accuracy (i.e., “drifted”) and is in need of maintenance.

B.3.3 RO Water Quality Event Detection Logic

RO feed TOC was not included in the water quality event detection logic because of the possibility that a spike in certain chemical contaminants would not be noticeable in feed TOC concentrations. Neutral, low-molecular weight organic compounds can pass through RO membranes more easily than the other types of TOC typically found in wastewater, translating to a lower rejection (or LRV) that would be observed. Since only a small amount of poorly rejected organic compounds would be needed to see a significant increase in permeate TOC, using feed TOC to monitor for chemical peaks that result in exceeding a regulatory threshold was not found to be a reliable method.

B.4 Additional Considerations for UV/AOP Event Detection

B.4.1 RO Water Quality Event Detection Logic

Overall, the UV/AOP performance is very consistent and measurements of CCP monitoring parameters tend to remain stable for sustained periods. This proved difficult when trying to identify threshold values for detecting stagnant outputs from instruments such as the UVI sensor and UVT meter. The detection sensitivity can be adjusted by increasing or decreasing the threshold of minimum change that must occur, the time period for the minimum number of consecutive failures, or both. An iterative approach was used to identify site-specific values for these two test parameters that provided the appropriate sensitivity.

APPENDIX C

Future Development

The EDS implemented at the NCPWDF provided proof of concept examples for each of the three event categories within the individual unit processes. There are improvements to the EDS interface and additional events that should be incorporated if applying this framework to a full-scale DPR facility to ensure that all potential issues are captured. The purpose of this section is to express areas for improvement in the EDS and expand upon the implemented events.

C.1 Future Development of Event Detection System Visualization

A recommended improvement to the architecture of the Pecos output dashboard is to implement a static y-axis for all tags that exhibit large fluctuations in values. The plots generated in the event report use a y-axis set by the maximum and minimum values in the time series being investigated. Due to signal variability and other communication artifacts, there are often single data points reported as extremely high values. Although this does not impact the tool's ability to detect and alert operators of events, it leads to plots that are un-informative and difficult to read. Figure C-1 demonstrates this phenomenon where single data points disrupt the time series' typical range, resulting in a plot that has limited utility because actual fluctuations can't be observed.

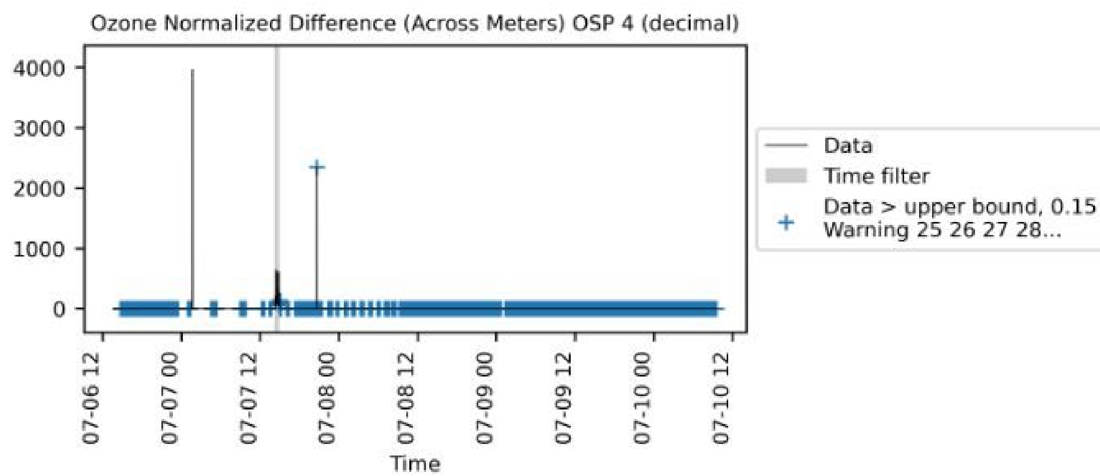


Figure C-1. Example of the Limitations of PECOS Reporting Plot.

This issue can be resolved by adding additional processing to the Pecos outputs and holding the y-axis maximum to a known upper limit of 1.0 that is representative of typical data fluctuations. The percent difference value between meters should remain within at least 100% when the meters are consistently maintained, so the plot will be far more useful if the y-axis is narrowed.

C.2 Future Development of Ozone Event Detection

Future work on detecting events within the ozone process should include the implementation of the events listed in Table C-1. The tool deployed at NCPWDF contained one event from each category to confirm that the event detection logic was functional. The addition of more events would make the tool more robust and comprehensive. The tags necessary to interpret data and detect these additional events already exist, and the logic behind event detection has been outlined below.

Table C-1. Ozone Events for Future Implementation.

Event	Tags	Description
Monitoring Point Events		
Stagnant Data	All analyzers and instrumentation	All data sources are monitored for erroneous reporting of stagnant data
Outside of Analytical Range	All analyzers and instrumentation	All data sourced from physical instrumentation is monitored for erroneous reporting of data outside of the analytical range of the instrument.
Ozone decay coefficient error – delta check	All dissolved ozone residual primary meters	$k_{1,2}$ and $k_{1,3}$ are compared to each other for consistency
Ozone decay coefficient error	All dissolved ozone residual primary meters	$k_{1,2}$ and $k_{1,3}$ are checked for a positive value which indicates downstream ozone concentration is incorrectly being reported as higher than upstream
Water Quality Event		
Low Crypto LRV	• Ozone Crypto LRV • OSP 4 primary meter • OSP 4 redundant meter • OSP 7 primary meter • OSP 7 redundant meter • Ozone generator production • Ozone production SP	This event detects a failure of the system to meet treatment goals. OSP 4 and OSP 7 ozone residual monitoring and ozone generator production are first verified as reporting properly. Thus, the decrease in Crypto LRV can be trusted as indicative of alterations to feed water quality, and not the result of process or monitoring point failures.
Low Ozone: TOC ratio	• Applied ozone dose • Feed TOC • Ozone generator production • Ozone production SP	Feed TOC is an indicator of water quality entering the ozone system. This event aims to detect alterations in the feed

		water quality preemptively, before the ratio threshold mandated by CA draft DPR regulations is crossed.[1] This event also gives context and can inform the cause of treatment compromises as elevated TOC. OSP 4 and OSP 7 ozone residual monitoring and ozone generator production are first verified as reporting properly.

[1]Based on the treatment goal of 1-log *Cryptosporidium* removal, and typical feed TOC values observed at NCPWDF, the regulatory ratio threshold of 1.0 will be exceeded.

C.2.1 Monitoring Point Events for Future Development

In addition to loss of meter sensitivity (as is currently detected by the tool), erroneously stagnant values reported by any of the instrumentation within the ozone system will also compromise the operations staff's ability to accurately assess the pathogen removal of the system. Stagnant data can be caused from improper meter maintenance, physical blockages in the instrumentation, or faulty data logging. Therefore, any future EDSs should include the testing of all online ozone instrumentation for the reporting of stagnant data. Stagnant data can be detected using a Pecos increment test configured with a stagnancy bound that is dependent on the precision of the data source.

Erroneous data that is outside of the meter's analytical range should be detected with a Pecos range test set to that analytical range. For example, the primary meter at OSP 4 has an analytical range of 0-5 ppm. If the meter is reading greater than 5ppm, it will be detected as monitoring point failure and alert operations to the need for calibration or maintenance of the meter. Water flow and temperature are also important parameters to ensure are reporting within analytical range because these parameters impact LRV.

Additional, yet less common ozone monitoring point failure events can be detected using the following logic applied to the continuously logged values for the ozone decay coefficients calculated between OSP4 and OSP 7, and OSP4 and OSP 10:

- Evaluate if meter readings decrease along the contactor (i.e., OSP4 residual>OSP 7 residual>OSP 10 residual). If an upstream meter is reading lower than the downstream meter and the downstream meter is reading within its historical average, there is likely an issue at the upstream meter and the data of the upstream meter will be flagged by a Pecos quality control test. To detect this failure, the EDS uses a Pecos range test to detect this failure event. A range test using a lower threshold of 0 to each of the ozone decay coefficients will detect when a downstream residual is reading lower than that of an upstream residual.

- The ozone decay between OSP4 and OSP 7, and OSP4 and OSP 10 should be similar. If both decay rates are negative but decay coefficients have an absolute difference of 0.3, there is likely an error with either Meters 2 or 3. Comparison to historical average decay rates and historical average of residuals at each location can be done to evaluate which meter should be flagged.

Event detection via ozone decay coefficients is redundant relative to the meter drift detection approach discussed in Chapter 4 using the percent difference between primary and reference meters at OSP 4 and 7. This is because if the meters at each OSP are not being flagged for drift from the redundant meter at that location, the ozone decay coefficients being calculated will be accurate, and the event will also not be occurring. However, the logic behind this method of detection is sound and can bolster a full-scale EDS, and thus is included in this discussion for future development as a measure of increased reliability.

C.2.2 Water Quality Events for Future Development

The logic and approach to the high ozone demand event detection described in Chapter 4 should be applied to directly monitoring *Cryptosporidium* LRV to ensure that alerts are generated when the ozone process is at risk of losing pathogen LRV credit. When detecting water quality changes, the process and monitoring instruments must first be verified as accurate. Since LRV can be calculated using residual values from all three OSPs, a meter accuracy check of each monitoring location (not just OSP 4 as configured in the high demand event) should be included. An early detection threshold of 1.25 should be implemented to flag when *Cryptosporidium* LRV is approaching 1.0.

An additional water quality event at the ozone system that can be included in the EDS uses feed TOC to evaluate the ozone:TOC ratio. Increased TOC concentration will consume ozone and potentially reduce CT below what is required for meeting treatment goals. Note that this event is more targeted than the high ozone demand and low LRV events because it only detects water quality changes related to TOC. Only the generator needs to be checked for proper functioning because data from the ozone residual meters are not involved in the calculations for this event.

C.3 Future Development of MF Event Detection

Future work on detecting events within the MF process should include the implementation of the event listed in Table C-2 which uses the analytical limits of the filtrate turbidimeter as the upper and lower bounds to ensure that the measurement outputs are within a sensible range. The minimum number of consecutive failures should be configured for a shorter time period (e.g., 5 minutes) than the high filtrate turbidimeter event because values that exceed the analytical measurement range should be detected prior to high filtrate turbidity caused by a process failure.

Table C-2. MF Events for Future Implementation.

Event	Tags	Description
Monitoring Point Events		
Analytical Range	MF Filtrate Turbidity	MF Filtrate Turbidity is monitored to ensure recorded values fall within the analytical range of the turbidimeter

C.4 Future Development for RO Event Detection

Additional RO events that would be useful for full-scale implementation of an EDS are listed in Table C-3 below. The recommended events for future development focus on monitoring the accuracy of each of the feed and permeate EC and TOC analyzers. A stagnant data event like the MF monitoring point event (see Chapter 4 for details) should be configured for each of the feed and permeate EC and TOC analyzers. As described in the future development sections for the ozone and MF processes, a Pecos range test configured with bounds for the analytical limits of the EC and TOC analyzers should be incorporated. To identify the occurrence of potential meter drift, the logic applied to the RO monitoring point event described in Chapter 4 should be implemented for the RO feed and permeate TOC and EC analyzers. If the RO feed TOC or EC meter was to drift down to measurements that are less than the true value, the calculated TOC/EC LRV would also decline because the permeate TOC/EC would continue reading a stable, true value. In contrast, if the RO permeate TOC meter was to drift down to values lower than the true value, the calculated TOC removal would drift high because the feed TOC would continue reading a stable, true value. By using the known relationship between feed and permeate TOC/EC measurements and the corresponding LRV, it is possible to detect when a meter is reporting abnormal values that do not following the expected trend.

Table C-3. RO Events for Future Implementation.

Event	Tags	Description
Monitoring Point Events		
Outside of analytical range	All TOC and EC meters	Meter data is tested to ensure the values reported remain within the range of physically possible values. The analytical range is meter-specific.
Potential Feed TOC Meter Drift (Low)	• Feed TOC • RO TOC Removal	For typical water quality, the TOC LRV is expected to remain relatively constant. Thus, if the feed TOC decreases, it is expected that the permeate TOC would also decrease such that the TOC LRV remains relatively constant. If a decrease in feed TOC below historical levels is observed while the TOC LRV also decreases below the historical operating range, then it is possible that the feed TOC meter is drifting low. This event would prompt an operator to check the feed TOC meter calibration.
Potential Permeate TOC Meter Drift (Low)	• Permeate TOC • RO TOC Removal	For typical water quality, the TOC LRV is expected to remain relatively constant. Thus, if the permeate TOC decreases, it is expected to be caused by a decrease in feed TOC. If a

		decrease in permeate TOC below historical levels is observed while the TOC LRV also increases above historical levels, then it is possible that the permeate TOC meter is drifting low. This event would prompt an operator to check the permeate TOC meter calibration.
Potential Feed EC Meter Drift (Low)	• Feed EC • RO EC Removal	For typical water quality, the EC LRV is expected to remain relatively constant. Thus, if the feed EC decreases, it is expected that the permeate EC would also decrease such that the EC LRV remains relatively constant. If a decrease in feed EC below historical levels is observed while the EC LRV also decreases below historical levels, then it is possible that the feed EC meter is drifting low. This event would prompt an operator to check the feed EC meter calibration.
Potential Feed EC Meter Drift (High)	• Feed EC • RO EC Removal	For typical water quality, the EC LRV is expected to remain relatively constant. Thus, if the feed EC increases, it is expected that the permeate EC would also increase such that the EC LRV remains relatively constant. If an increase in feed EC above historical levels is observed while the EC LRV also increases above historical levels, then it is possible that the feed EC meter is drifting high. This event would prompt an operator to check the feed EC meter calibration.
Potential Permeate EC Meter Drift (Low)	• Permeate EC • RO EC Removal	For typical water quality, the EC LRV is expected to remain relatively constant. Thus, if the permeate EC decreases, it is expected that the feed EC would also decrease such that the EC LRV remains relatively constant. If a decrease in permeate EC below historical levels is observed while the EC LRV increases above the historical operating range, then it is possible that the permeate EC meter is drifting low. This event would prompt an operator to check the permeate EC meter calibration.
Potential Permeate EC Meter Drift (High)	• Permeate EC • RO EC Removal	For typical water quality, the EC LRV is expected to remain relatively constant. Thus, if the permeate EC increases, it is expected that the feed EC would also increase such that the EC LRV remains relatively constant. If an increase in permeate EC above historical levels is observed while the EC LRV decreases below the historical operating range, then it is possible that the permeate EC meter is drifting high. This event would prompt an operator to check the permeate EC meter calibration.

C.5 Future Development for UV/AOP Event Detection

Future work on event detection within the UV/AOP process should include the implementation of the events listed in Table C-4. The UVT meter, flow meter, and chlorine analyzers are all instruments with measurement ranges that the EDS should monitor with analytical range event logic.

Oxidant dose is a key parameter for ensuring that AOP criteria are met and the design sodium hypochlorite dose at the NCPWDF is 1.0 mg/L. Oxidant dosing is monitored by measurements taken by a free chlorine analyzer and the difference between total chlorine measurements before and after the sodium hypochlorite injection point. If either of these free chlorine readings are below 1.0 mg/L, a process failure of the chemical dosing system is potentially occurring.

The calculated difference between these sources for free chlorine measurements allows for confirmation of the accuracy of the primary free chlorine meter. If the difference becomes larger than a maximum threshold of 20%, an error in one of the meters may have occurred and an alert should be generated by the EDS to check the calibration of each meter.

Table C-4. UV/AOP Events for Future Implementation.

Event	Tags	Description
Process Failure Events		
Low Oxidant Concentration	• RO Permeate Total Chlorine • UV Feed Total Chlorine • UV Feed Free Chlorine	The measured oxidant concentration from the free chlorine analyzer and difference between the total chlorine analyzers is lower than the operating bound indicating that there is a potential issue with the chemical dosing system.
Monitoring Point Events		
Outside of Analytical Range	All analyzers and instrumentation	Measured data is monitored for erroneous reporting outside of the analytical range specified by the manufacturer.
Oxidant Dose Monitoring Point Failure	• RO Permeate Total Chlorine • UV Feed Total Chlorine • UV Feed Free Chlorine	The difference between free chlorine readings from primary and reference chlorine analyzers exceeds the acceptable threshold indicating that at least one of the meters needs maintenance or requires calibrating.

APPENDIX D

DPR Early Event Detection Python Script Deployed at NCPWDF

The code repository for the EDS implemented at NCPWDF can be accessed via the WRF 4954 project page on waterrf.org. The two Python scripts developed for the analysis are also included in Figures D-1 and D-2 below.

```
"""
Create dashboard to identify events in direct potable reuse (DPR) systems.
Author(s): Billy Raseman and Nolan Townsend

Description: script to be deployed at the City of San Diego Pure Water North City DPR Demonstration
Facility as part of Water Research Foundation (WRF) Project 4954. This script uses data from the
demonstration facility and open source Python packages to create a dashboard that identifies events in the
DPR system. The dashboard is intended to be used by operators to identify events and to help inform the
development of an automated event detection system.

Acknowledgements: this project is a collaboration between Hazen & Sawyer and Trussell Technologies with
support from the City of San Diego and WRF.
"""

# Import functions from other scripts
from datetime import datetime, timedelta
from library import *

# Common Python packages
from loguru import logger
import pandas as pd
import matplotlib.pyplot as plt
import matplotlib as mpl
import numpy as np
import os
import pecos
import json
from plyer import notification

# Pecos is a python package for performing automated quality control of time series data.
# It is designed to work with data stored in pandas DataFrames and Series.
# Pecos contains tools for:
# 1. Automatically detecting and flagging anomalies in time series data
# 2. Calculating performance metrics for time series data
```

```
# 3. Generating summary statistics and plots for time series data
# 4. Generating HTML reports for time series data
# 5. Performing quality control on time series data

############################################# USER INPUTS #############################################
## SQL database connection information
# NOTE REMOVED DATABASE CONNECTION INFO
server =
database =
username =
password =
driver =
table =
datetime_col =

## User defined run mode and outputs
save_plots = True  # save plots as png files
notify = True  # send notifications
# run_now = False  # run script in live mode (True) or historical mode (False)

## User defined paths
path_config = r"config.xlsx"
results_directory = 'example_1'  # name of results directory
path_working = os.getcwd()
path_events_json = os.path.join(path_working, 'events.json')

# ## User defined parameters
# ## Note: most data beings the week of September 12, 2022.
# ##  Additional tags for ozone and RO were added the week of March 20, 2023 and in June 2023.
# user_datetime_start = "5/24/2023 12:00:00"  # start datetime for historical mode
# user_datetime_end = "5/24/2023 14:00:00"  # end datetime for historical mode

############################################# END USER INPUTS #########################################

def main ():

    # # Use current datetime if run_now == True, otherwise use user input
    # if run_now == True:
    #     datetime_start_init = datetime.now() - timedelta(days=0.5)  # start datetime for live mode
    #     datetime_start_round = datetime_start_init.replace(second=0, microsecond=0)
    #     datetime_start = datetime.strftime(datetime_start_round, '%Y-%m-%d %H:%M:%S')

    #     datetime_end_init = datetime.now()  # end datetime for live mode
    #     datetime_end_round = datetime_end_init.replace(second=0, microsecond=0)
    #     datetime_end = datetime.strftime(datetime_end_round, '%Y-%m-%d %H:%M:%S')
```

```
# else:  # user input for historical mode
datetime_start = user_datetime_start
datetime_end = user_datetime_end

# Check that datetime_start and datetime_end are in the correct format
datetime_start = check_datetime_format(datetime_start)
datetime_end = check_datetime_format(datetime_end)

# Check that datetime_start is before datetime_end
check_datetime_order(datetime_start, datetime_end)

# Read in configuration file data
logger.info('Reading data from configuration file.')

## Read in Tags and Events tables from config file
df_tags = pd.read_excel(path_config, sheet_name='Tags')
check_tags_table(df_tags)
df_events = pd.read_excel(path_config, sheet_name='Events')
check_events_table(df_events)

## Create engine to connect to project's SQL database
logger.info('Creating engine for SQL database.')
engine = create_engine(driver, server, database, username, password)

## Check for TagIDs in config file that are not in the database
check_tagids_missing_from_sql(engine, df_tags, table, datetime_col, datetime_start, datetime_end)

## Convert Tag and TagID columns to a dictionary
tag_dict = dict(zip(df_tags['TagID'], df_tags['Tag (Units)']))

## Convert Event and EventID columns to a dictionary
event_dict = dict(zip(df_events['EventID'], df_events['TagIDs']))
event_dict = convert_items_to_list_of_ints(event_dict) # convert items to list of ints
event_flags = dict.fromkeys(range(1, len(event_dict) + 1), False) # initialize event flags to False

# Read in data from SQL database

## Only include datetime range that is specified
tagid_set = set(tag_dict.keys())  # only include TagIDs that are in the config file
logger.info(f'Querying data from SQL database between {datetime_start} and {datetime_end}.')
df = create_df_from_sql(engine, table, datetime_start, datetime_end, tagid_set, datetime_col)

# Add Tag to SQL data based on TagID using the tag_dict dictionary
df['Tag'] = df['TagID'].map(tag_dict)
```

```
    df1 = df[['Datetime', 'Tag', 'Value']]  # only keep Datetime, Tag, and Value columns

    # Pivot dataframe to wide format (required for Pecos logic)
    ## Wide format: each Tag is a column (e.g., 'RO Feed TOC', 'UV Dose', etc.)
    df_wide = df1.pivot(index='Datetime', columns='Tag', values='Value').reset_index()
    df_wide = df_wide.set_index('Datetime')

    # Check for missing columns in dataframe

    ## Begin with production status tags
    logger.info('Checking for missing production status tags in dataframe.')

    tag_plant_status = tag_dict[43]
    tag_mf_status = tag_dict[64]
    tag_bac1_status = tag_dict[48]
    tag_bac2_status = tag_dict[49]

    status_tags = [tag_plant_status, tag_mf_status, tag_bac1_status, tag_bac2_status]

    # Check for missing columns for each production status tag
    check_for_missing_status_tag(df_wide, tag_plant_status)  # Plantwide status
    check_for_missing_status_tag(df_wide, tag_mf_status)  # MF status
    check_for_missing_status_tag(df_wide, tag_bac1_status)  # BAC1 status
    check_for_missing_status_tag(df_wide, tag_bac2_status)  # BAC2 status

    ## Next check value tags
    logger.info('Checking for missing value tags in dataframe.')

    # Check for missing columns for each value tag. If all tags are missing, then set event flag to True.
    for eventid in event_dict:
        tagids = event_dict[eventid]
        for tagid in tagids:
            check_for_missing_value_tag(df_wide, tag_dict, tagid)
        event_flags[eventid] = check_all_tags_missing_for_event(df_wide, tag_dict, event_dict[eventid])

    # For tags related to error or difference calculations, modify the data to be the absolute value of
error/difference.
    # This is to ensure that the data is always positive to avoid errors in the Pecos logic.
    tags_err_diff = [tag_dict[86], tag_dict[67], tag_dict[91], tag_dict[92]]
    logger.info(f'Modifying data for error/difference tags. Calculating absolute value for
{tags_err_diff}.')

    for tag in tags_err_diff:
        df_wide[tag] = df_wide[tag].abs()
```

```
    # RO events: create calculated columns for RO events
    logger.info('Creating calculated columns for RO events.')
    name_ro_process = 'RO Process Event'  # column name for new tag
    name_ro_wq1 = 'RO Water Quality Event'
    name_ro_monitoring = 'RO Monitoring Event'

    # Create new column for RO Process event:
    # If RO Combined Permeate TOC > 50 ppb & RO Train A Permeate Conductivity > 125 ppb & RO Train B
Permeate Conductivity > 125 ppb,
    # then RO Process = 1, else 0.
    cond1 = (df_wide[tag_dict[15]] > 50) | (df_wide[tag_dict[15]].isna())
    cond2 = (df_wide[tag_dict[29]] > 125) | (df_wide[tag_dict[29]].isna())
    cond3 = (df_wide[tag_dict[31]] > 125) | (df_wide[tag_dict[31]].isna())
    df_wide[name_ro_process] = np.where(cond1 & cond2 & cond3, 1, 0)
    del cond1, cond2, cond3

    # Create new column for RO Monitoring event:
    # If RO Feed TOC < 3780 ppb & RO LRV via TOC < 2.1, then RO Monitoring = 1, else 0.
    cond1 = (df_wide[tag_dict[1]] < 3780) | (df_wide[tag_dict[1]].isna())
    cond2 = (df_wide[tag_dict[17]] < 2.1) | (df_wide[tag_dict[17]].isna())
    df_wide[name_ro_monitoring] = np.where(cond1 & cond2, 1, 0)
    del cond1, cond2

    # Create new column for RO Water Quality event:
    # If RO Combined Permeate TOC > 50 ppb & RO Train A Permeate Conductivity < 125 ppb & RO Train B
Permeate Conductivity < 125 ppb,
    # then RO Water Quality = 1, else 0.
    cond1 = (df_wide[tag_dict[15]] > 50) | (df_wide[tag_dict[15]].isna())
    cond2 = (df_wide[tag_dict[29]] < 125) | (df_wide[tag_dict[29]].isna())
    cond3 = (df_wide[tag_dict[31]] < 125) | (df_wide[tag_dict[31]].isna())
    df_wide[name_ro_wq1] = np.where(cond1 & cond2 & cond3, 1, 0)
    del cond1, cond2, cond3

    # Ozone events: create calculated columns for Ozone events
    logger.info('Creating calculated columns for Ozone events.')
    name_ozone_wq1 = 'Ozone Water Quality Event'

    # Create new column for Ozone Water Quality event:

    # If OSP 4 Hach Meter and rolling average difference is LESS than 10%,
    # and OSP 7 Hach Meter and rolling average difference is LESS than 10%,
    # and ozone production error is LESS than 10%,
    # and ozone demand is greater than 6.0 mg/L,
    # then Ozone Water Quality = 1, else 0.
    # If any values are empty, assume they violate the condition.
```

```
    cond1 = (df_wide[tag_dict[91]] < 0.1)  # don't apply missing value logic here
    cond2 = (df_wide[tag_dict[92]] < 0.1)  # don't apply missing value logic here
    cond3 = (df_wide[tag_dict[86]] < 0.1)  # don't apply missing value logic here
    cond4 = (df_wide[tag_dict[59]] > 6.0) | (df_wide[tag_dict[59]].isna())
    df_wide[name_ozone_wq1] = np.where(cond1 & cond2 & cond3 & cond4, 1, 0)
    del cond1, cond2, cond3, cond4

    # Create dashboard based on Pecos results
    dashboard_content = {} # Initialize the dashboard content dictionary
    logger.info('Implementing Pecos tests.')

    # Create dictionary of primary event tags (the ones that are used to determine if an event is
occurring)
    primary_event_tags = create_primary_event_tags_dict(event_dict, tag_dict, name_ro_process,
name_ro_monitoring, name_ro_wq1, name_ozone_wq1)

    # Loop through the events in df_events
    list_detected_events = [] # Initialize list of detected events
    num_events = df_events.shape[0]
    for i in range(0, num_events):

        ## Get values for this EventID from df_events
        eventid = df_events['EventID'][i]
        event_text = df_events['Event Text'][i]
        event_process = df_events['Event Process'][i]
        event_type = df_events['Event Type'][i]

        ## Get tags for this event from dictionary
        event_tagids = event_dict[eventid]
        if len(event_tagids) == 0:
            is_empty = True
        else:
            is_empty = False
        event_tags = [tag_dict[tagid] for tagid in event_tagids]

        ## For events with calculated columns, add the calculated column name to the list of tags (in
front)
        if eventid == 5:
            event_tags.insert(0, name_ro_process)
        elif eventid == 6:
            event_tags.insert(0, name_ro_monitoring)
        elif eventid == 7:
            event_tags.insert(0, name_ro_wq1)
        elif eventid == 15:
            event_tags.insert(0, name_ozone_wq1)
```

```
        ## Initialize event flag
        if is_empty == True:
            # If event is not applicable just store placeholder text
            content = { 'text': event_text }
            dashboard_content[(event_process, event_type)] = content
        else:
            # Create new Pecos PerformanceMonitoring object
            pm = pecos.monitoring.PerformanceMonitoring()
            event_and_status_tags = event_tags + status_tags  # only keep value tags needed for the event
plus all status tags
            df_wide_event = df_wide[event_and_status_tags]  # subset data to only include tags for this
event
            pm.add_dataframe(df_wide_event)  # add data to Pecos object

            ## Timefilter
            time_filter_system = pm.data[tag_plant_status] == 1
            pm.add_time_filter(time_filter_system)  # add time filter when plantwide status is not 1
(normal)

            ## Check missing data
            pm = pecos_check_missing(pm, event_tags)

            # Apply process-specific time filters
            if event_process == 'MF':
                # Filter based on MF process status (if not in production)
                time_filter_process1 = pm.data[tag_mf_status] == 1
                pm.add_time_filter(time_filter_process1)
            elif event_process == 'Ozone':
                # Filter based on BAC process status (if not in production)
                time_filter_process1 = pm.data[tag_bac1_status] == 1
                pm.add_time_filter(time_filter_process1)
                time_filter_process2 = pm.data[tag_bac2_status] == 1
                pm.add_time_filter(time_filter_process2)

            # Apply Pecos tests for each EventID
            if eventid == 1:
                # MF Process
                pm.check_range(key=tag_dict[10], bound=[None, 0.15], min_failures=15)

            elif eventid == 2:
                # MF Monitoring
                pm.check_increment(key=tag_dict[10], bound=[0.0001, None], min_failures=30)

            elif eventid == 5:
```

```
            # RO Process event
            pm.check_range(key=name_ro_process, bound=[None, 0], min_failures=30)
            pm.check_range(key=tag_dict[15], bound=[None, 50], min_failures=30)
            pm.check_range(key=tag_dict[29], bound=[None, 125], min_failures=15)
            pm.check_range(key=tag_dict[31], bound=[None, 125], min_failures=15)

        elif eventid == 6:
            # RO Monitoring event
            pm.check_range(key=name_ro_monitoring, bound=[None, 0], min_failures=30)
            pm.check_range(key=tag_dict[1], bound=[3780, None], min_failures=30)
            pm.check_range(key=tag_dict[17], bound=[2.1, None], min_failures=30)

        elif eventid == 7:
            # RO Water Quality 1 event
            pm.check_range(key=name_ro_wq1, bound=[None, 0], min_failures=30)
            pm.check_range(key=tag_dict[15], bound=[None, 50], min_failures=30)
            pm.check_range(key=tag_dict[29], bound=[125, None], min_failures=5)
            pm.check_range(key=tag_dict[31], bound=[125, None], min_failures=5)

        elif eventid == 9:
            # UVAOP Process
            pm.check_range(key=tag_dict[40], bound=[300, None], min_failures=5)

        elif eventid == 10:
            # UVAOP Monitoring
            # pm.check_increment(key=tag_dict[65], bound=[0.01, None], min_failures=15)  / in
documentation
            pm.check_increment(key=tag_dict[65], bound=[0.01, None], min_failures=30)

        elif eventid == 11:
            # UVAOP Water Quality 1
            pm.check_range(key=tag_dict[32], bound=[96, None], min_failures=15)

        elif eventid == 12:
            # UVAOP Water Quality 2
            pm.check_range(key=tag_dict[83], bound=[None, 1], min_failures=15)

        elif eventid == 13:
            # Ozone Process
            pm.check_range(key=tag_dict[86], bound=[None, 0.05], min_failures=15)

        elif eventid == 14:
            # Ozone Monitoring
            pm.check_range(key=tag_dict[67], bound=[None, 0.1], min_failures=15)
```

```
        elif eventid == 15:
            # Ozone Water Quality 1
            pm.check_range(key=name_ozone_wq1, bound=[None, 0], min_failures=15)
            pm.check_range(key=tag_dict[91], bound=[0.1, None], min_failures=2)
            pm.check_range(key=tag_dict[92], bound=[0.1, None], min_failures=2)
            pm.check_range(key=tag_dict[86], bound=[0.1, None], min_failures=2)
            pm.check_range(key=tag_dict[59], bound=[None, 6.0], min_failures=15)
        else:
            logger.warning(f'Incorrect logic for EventID {eventid}')

        # Compute metrics
        mask = pm.mask[event_tags]
        QCI = pecos.metrics.qci(mask, pm.tfilter)

        # If QCI is less than 1, then flag the event
        if QCI[primary_event_tags[eventid]] < 1:
            event_flags[eventid] = True

        # Define output files and subdirectories for this event
        results_subdirectory = os.path.join(results_directory, event_process+'_'+event_type)
        reset_directory(results_subdirectory)  # reset directory if it already exists. If it doesn't
exist, create it.
        graphics_file_rootname = os.path.join(results_subdirectory, 'test_results')
        custom_graphics_file = os.path.abspath(os.path.join(results_subdirectory, 'custom.png'))
        test_results_file = os.path.join(results_subdirectory, 'test_results.csv')
        colorblock_graphics_file = os.path.abspath(os.path.join(results_subdirectory,
'colorblock.png'))
        report_file =  os.path.join(results_subdirectory, 'monitoring_report.html')

        # Create plots
        test_results_graphics = pecos.graphics.plot_test_results(pm.data, pm.test_results,
                                    pm.tfilter, filename_root=graphics_file_rootname)

        # Log event and modify colorblock if event is occurring
        if event_flags[eventid]:
            logger.critical(f'Alert: {event_text} event is occurring, take action!')
            list_detected_events.append(f'{event_text}')
            color = 0  # fill in colorblock if event is occurring
        else:
            color = 1  # otherwise, keep colorblock gray

        # Create colorblock plot
        pecos.graphics.plot_heatmap(pd.Series(color), vmin=0.9999, vmax=1,
cmap=mpl.colors.ListedColormap(['magenta','lightgray']))  # colorblock (magenta or gray)
        plt.savefig(colorblock_graphics_file, dpi=90, bbox_inches='tight', pad_inches = 0.1)
```

```
            # Create timeseries plot each tag in the event to show the raw data
            if save_plots:
                df_plot = df1[df1['Tag'].isin(event_tags)]

                # If df_plot is empty, create a dummy plot
                if df_plot.empty:  # If no data, create an empty png
                    fig, ax = plt.subplots()
                    ax.axis('off')
                    ax.axis('tight')
                    ax.text(0.5, 0.5, 'No data to display', horizontalalignment='center',
verticalalignment='center', transform=ax.transAxes)
                else:  # Create a facet plot of the data
                    facetplot_by_tag(df_plot)

                plt.savefig(custom_graphics_file, format='png', dpi=250, bbox_inches='tight')

            # Write test results and report files
            pecos.io.write_test_results(pm.test_results, test_results_file)
            pecos.io.write_monitoring_report(data=pm.data,
                                             test_results=pm.test_results,
                                             test_results_graphics=test_results_graphics,
                                             custom_graphics=[custom_graphics_file],
                                             metrics=QCI,
                                             title=event_text,
                                             filename=report_file)

            # Close plots
            plt.close('all')

            ## Write CSS to report file
            with open(report_file, 'a') as f:
                css_file = os.path.join(path_working, 'style2.css')
                html_content = f'<link rel="stylesheet" type="text/css" href="{css_file}">\n'
                logo_file = os.path.join(path_working, 'logo_all.png')
                html_content += f'<div style="height: 50px; display: flex; flex-direction: row; align-
items: center; justify-content: flex-start;"> <img style="height: 50px; width: 355px !important;"
src="{logo_file}" alt="logo"></div>\n'
                f.write(html_content)

            # Store content to be displayed in the dashboard
            content = { 'text': event_text,
                        'graphics': [colorblock_graphics_file],
                    'link': {'Link to Report': os.path.abspath(report_file)}}
            dashboard_content[(event_process, event_type)] = content
```

```
    # Create/update dashboard
    logger.info('Writing Pecos results to dashboard.')
    events = ['Process', 'Monitoring', 'Water Quality 1', 'Water Quality 2']
    processes = ['MF', 'RO', 'UVAOP', 'Ozone']
    dashboard_file = f'dashboard_{results_directory}.html'

    pecos.io.write_dashboard(column_names=events, row_names=processes,
                             content=dashboard_content, title='Direct Potable Reuse Monitoring Dashboard',
                            filename=dashboard_file)

    ## Write CSS to dashboard file
    path_dashboard_file = os.path.join(path_working, dashboard_file)
    with open(path_dashboard_file, 'a') as f:
        css_file = os.path.join(path_working, 'style1.css')
        html_content = f'<link rel="stylesheet" type="text/css" href="{css_file}">\n'
        logo_file = os.path.join(path_working, 'logo_all.png')
        html_content += f'<div style="height: 50px; display: flex; flex-direction: row; align-items:
center; justify-content: flex-start;"> <img style="height: 50px; width: 355px !important;"
src="{logo_file}" alt="logo"></div>\n'
        f.write(html_content)

    # Send notification if any new events have occurred since last time script was run
    logger.info('Checking event flags and sending notifications, if necessary.')
    if notify == True:  # only send notifications if script is running live
        # Read in events from previous run
        if os.path.exists(path_events_json):
            with open(path_events_json, 'r') as f:
                events_json = json.load(f)
        else:
            events_json = {'events': []}

        # Check if any new events have occurred
        list_new_events = []
        for event in list_detected_events:
            if event not in events_json['events']:
                list_new_events.append(event)

        # If new events have occurred, send notification
        if len(list_new_events) > 0:
            logger.info('New event(s) detected. Sending notification.')
            message = f"New event(s) detected. Refresh the dashboard to review the events:
{list_new_events}"
            notification.notify(title="Alert!", message=message)
            events_json['events'] = list_detected_events  # update events_json
```

```python
            with open(path_events_json, 'w') as f:
                json.dump(events_json, f)
        else:
            logger.info('No new events have occurred since last time script was run.')

if __name__ == '__main__':

    # Initialize logger in current working directory
    path_log = os.path.join(path_working, 'dashboard.log')
    logger.add(path_log, retention='365 days', level='INFO', rotation='30 days', compression='zip')

    # Run the main function for 24 hour time periods
    # This is to test the script for a full month of data.
    # After running the script each time, pause for user input to continue to the next time period.

    start_datetime = datetime(2023, 5, 18, 0, 0, 0)
    end_datetime = datetime(2023, 5, 18, 23, 59, 59)

    # For loop to run the main function for each 24 hour time period
    for i in range(31):

        # Convert datetime to "YYYY-MM-DD HH:MM:SS" format
        user_datetime_start = start_datetime.strftime("%Y-%m-%d %H:%M:%S")
        user_datetime_end = end_datetime.strftime("%Y-%m-%d %H:%M:%S")

        # Run main function
        main()

        # Pause for user input to continue to the next time period
        input("Press Enter to continue to next time period...")

        # Increment start and end datetimes by 24 hours
        start_datetime += timedelta(days=1)
        end_datetime += timedelta(days=1)
```

Figure D-1. Python Script “main.py” for EDS Dashboard.

```
"""
Library of functions for the dashboard.
"""

# Import libraries
import os
import sys
import numpy as np
import sqlalchemy as sa
import urllib
import pandas as pd
import seaborn as sns
from loguru import logger

# User defined functions
def check_datetime_format(datetime):
    """Check that datetime is in the correct format."""
    try:
        datetime = pd.to_datetime(datetime)
    except:
        logger.error(f'Datetime {datetime} is not in the correct format. Please enter a datetime in the
format "mm/dd/yyyy hh:mm:ss".')
        raise
    return datetime

def check_datetime_order(datetime_start, datetime_end):
    """Check that datetime_start is before datetime_end."""
    if datetime_start > datetime_end:
        logger.error(f'Datetime {datetime_start} is after {datetime_end}. Please enter a datetime_start
that is before datetime_end.')
        raise

def check_tags_table(df):
    """
    Check that tags table has been loaded correctly from the configuration file.

    Args:
    df (pandas.DataFrame): Dataframe of tags table.

    If tags table is not loaded correctly, log a warning.
    """

    # Check that df is a dataframe
    if not isinstance(df, pd.DataFrame):
        logger.warning('check_tags_table(): df is not a dataframe.')
```

```
    # Check that there are rows in the dataframe
    if df.shape[0] == 0:
        logger.warning('check_tags_table(): df is empty.')

    # Check that columns include TagID, Tag, Process, and Units
    if 'TagID' not in df.columns:
        logger.warning('check_tags_table(): TagID column not found in tags table.')
    if 'Tag' not in df.columns:
        logger.warning('check_tags_table(): Tag column not found in tags table.')
    if 'Process' not in df.columns:
        logger.warning('check_tags_table(): Process column not found in tags table.')
    if 'Units' not in df.columns:
        logger.warning('check_tags_table(): Units column not found in tags table.')

    # Check that TagID column is type int
    if df['TagID'].dtype != 'int64':
        logger.warning('check_tags_table(): TagID column is not type int64.')

    # Check that Tag column is type str
    if df['Tag'].dtype != 'object':
        logger.warning('check_tags_table(): Tag column is not type object.')

    # Check that Process column is type str
    if df['Process'].dtype != 'object':
        logger.warning('check_tags_table(): Process column is not type object.')

    # Check that Units column is type str
    if df['Units'].dtype != 'object':
        logger.warning('check_tags_table(): Units column is not type object.')

    # Check that TagID column is unique
    if not df['TagID'].is_unique:
        logger.warning('check_tags_table(): TagID column is not unique.')

    # Check that Tag column is unique
    if not df['Tag'].is_unique:
        logger.warning('check_tags_table(): Tag column is not unique.')

    return

def check_events_table(df):
    """
    Check that events table has been loaded correctly from the configuration file.
```

```
    Args:
        df (pandas.DataFrame): Dataframe of events table.

    If events table is not loaded correctly, log a warning.
    """

    # Check that df is a dataframe
    if not isinstance(df, pd.DataFrame):
        logger.warning('check_events_table(): df is not a dataframe.')

    # Check that there are rows in the dataframe
    if df.shape[0] == 0:
        logger.warning('check_events_table(): df is empty.')

    # Check that columns include Empty, EventID, Event Text, Event Process, and Event Type
    if 'EventID' not in df.columns:
        logger.warning('check_events_table(): EventID column not found in events table.')
    if 'Event Text' not in df.columns:
        logger.warning('check_events_table(): Event Text column not found in events table.')
    if 'Event Process' not in df.columns:
        logger.warning('check_events_table(): Event Process column not found in events table.')
    if 'Event Type' not in df.columns:
        logger.warning('check_events_table(): Event Type column not found in events table.')

    # Check that EventID column is type int
    if df['EventID'].dtype != 'int64':
        logger.warning('check_events_table(): EventID column is not type int64.')

    # Check that Event Text column is type str
    if df['Event Text'].dtype != 'object':
        logger.warning('check_events_table(): Event Text column is not type object.')

    # Check that Event Process column is type str
    if df['Event Process'].dtype != 'object':
        logger.warning('check_events_table(): Event Process column is not type object.')

    # Check that Event Type column is type str
    if df['Event Type'].dtype != 'object':
        logger.warning('check_events_table(): Event Type column is not type object.')

    # Check that EventID column is unique
    if not df['EventID'].is_unique:
        logger.warning('check_events_table(): EventID column is not unique.')

    return
```

```python
def convert_items_to_list_of_ints(event_dict):
    """
    Convert event dictionary items into a list of integers.
    If an integer already, convert to list of integers.
    If a string, convert to list of integers.
    If empty, convert to empty list.
    Otherwise, log an warning.

    Args:
        event_dict (dictionary): dictionary of event items to convert to a list of integers.

    Returns:
        event_dict (dictionary): dictionary of event items converted to a list of integers.

    """
    # Check that parameter is a dictionary
    if type(event_dict) != dict:
        logger.error(f'Input argument is not a dictionary, it is {type(event_dict)}.')

    # Convert dictionary items into a list of integers
    for k, v in event_dict.items():
        if type(v) == int:
            event_dict[k] = [v]
        elif type(v) == str:
            event_dict[k] = [int(i) for i in v.split(',')]
        elif np.isnan(v):
            event_dict[k] = []
        elif v == '':
            event_dict[k] = []
        else:
            logger.warning(f'Issue converting TagIDs {k} in Events table to a list of integers.')

    return event_dict

def create_engine(driver, server, database, username, password):
    """
    Create engine with the project's SQL database.

    Args:
        driver (str): SQL driver name.
        server (str): SQL server name.
        database (str): SQL database name.
        username (str): SQL username.
        password (str): SQL password.
```

```
    Returns:
        sqlalchemy.engine: Engine object for the project's SQL database.
    """

    # Create SQL engine to read/write table data
    cnxn_url=
urllib.parse.quote_plus(f'DRIVER={driver};SERVER={server};DATABASE={database};UID={username};PWD={password
}')
    engine = sa.create_engine(f'mssql+pyodbc:///?odbc_connect={cnxn_url}')

    # Check the connection
    try:
        conn = engine.connect()
        logger.info('Database connection test successful.')
        conn.close()
    except Exception as e:
        logger.error(f'Database connection test failed: {e}')

    return engine

def check_tagids_missing_from_sql(engine, df, table, datetime_col, datetime_start, datetime_end):
    """
    Compare the TagIDs in the SQL database to the TagIDs in the configuration file.
    If there are TagIDs in the configuration file that are not found in the SQL database,
    log a warning with the TagIDs that are missing from the database.

    Args:
        engine (sqlalchemy.engine.base.Engine): SQLAlchemy connection engine.
        df (pandas.DataFrame): Dataframe of tags table.

    Returns:
        list_missing_tagids (list): List of TagIDs that are missing from the SQL database.
    """

    ## Get a set of TagIDs from SQL database for datetime range
    logger.info('compare_sql_config_tags(): querying TagIDs from SQL database.')
    query1 = f'SELECT DISTINCT TagID FROM {table} WHERE {datetime_col} >= \'{datetime_start}\' AND
{datetime_col} < \'{datetime_end}\''
    df_sql = pd.read_sql(query1, engine)
    sql_tagids = set(df_sql['TagID'])  # convert from dataframe to a set

    ## Get a set of TagIDs from config file
    config_tagids = set(df['TagID'])
```

```python
    ## Compare the two sets of TagIDs. If there are TagIDs in the config file that aren't found in the database, log a warning with the TagIDs that are missing from the database
    if config_tagids.issubset(sql_tagids) == False:
        missing_tagids = config_tagids - sql_tagids
        logger.warning(f'The following TagIDs are missing from the database: {missing_tagids}')
        list_missing_tagids = df[df['TagID'].isin(missing_tagids)]
    else:
        logger.info('All TagIDs in the config file are found in the database.')
        list_missing_tagids = []

    return list_missing_tagids


def create_df_from_sql(engine, table, datetime_start, datetime_end, tagid_set, datetime_col="[DateTime]"):
    """
    Create a dataframe from a subset of the project's SQL database.

    Args:
        engine (sqlalchemy.engine): SQL engine to read/write table data.
        table (str): SQL table name.
        datetime_start (datetime): Start datetime for query.
        datetime_end (datetime): End datetime for query.
        tagid_set (set): Set of tagids to query.
        datetime_col (str): Datetime column name in SQL database.

    Returns:
        pandas.DataFrame: Dataframe of SQL query results.

    Warns:
        UserWarning: If tagid_set cannot be converted to type set.
    """

    # Convert tagid_set to type set, if cannot convert, log a warning
    try:
        tagid_set = set(tagid_set)
    except:
        logger.warning('create_df_from_sql(): tagid_set cannot be converted to a set.')

    # Convert set to a string to pass to SQL query
    tagid_str = '(' + ', '.join(str(e) for e in tagid_set) + ')'
    print(tagid_str)
    # Set up query
    query= f'SELECT TagID, DateTime, Value FROM {table} WHERE {datetime_col} >= \'{datetime_start}\' AND {datetime_col} < \'{datetime_end}\' AND TagID IN {tagid_str}'

    # Use pandas to query the database and return a pandas dataframe
```

```
    sql_df = pd.read_sql(query, engine)

    if datetime_col == '[DateTime]':
        sql_df = sql_df.rename(columns={'DateTime':'Datetime'})

    # Truncate datetimes with seconds on the minute containing the seconds
    sql_df['Datetime'] = pd.to_datetime(sql_df['Datetime'])
    sql_df['Datetime'] = sql_df['Datetime'].dt.floor('min')

    # Check if dataframe is empty. If so, log an error and exit the program.
    if sql_df.empty:
        logger.error(f'create_df_from_sql(): Error! No data found in {table} between {datetime_start} and
{datetime_end}. Exiting program.')
        sys.exit()

    return sql_df

def check_for_missing_status_tag(df_wide, tag):
    """
    Check for missing columns in dataframe (df_wide) for each production status tag (tagid).
    If a column is missing, log a warning and create a new column with all values set to 1.0.

    Args:
        df_wide (pandas.DataFrame): Dataframe with wide format.
        tag (str): Tag name of production status tag.

    Returns:
        None
    """
    if tag not in df_wide.columns:
        logger.warning(f'The following column is missing from the dataframe: {tag}. Creating new column
and setting all values to 1.0.')
        df_wide[tag] = 1.0  # 1.0 = in production (assume in production for entire datetime range if
column is missing)

def check_for_missing_value_tag(df_wide, tag_dict, tagid):
    """
    Check for missing columns in dataframe (df_wide) for each value tag (tagid).
    If a column is missing, log a warning and create a new column with all values set to NaN.

    Args:
        df_wide (pandas.DataFrame): Dataframe with wide format.
        tag_dict (dict): Dictionary of tagid:tagname.
        tagid (int): TagID of value tag.
```

```
    Returns:
        None
    """
    tag = tag_dict[tagid]
    if tag not in df_wide.columns:
        logger.warning(f'The following column is missing from the dataframe: {tag}. Creating new column 
and setting all values to NaN.')
        df_wide[tag] = np.nan  # create new column and set all values to NaN

def check_all_tags_missing_for_event(df_wide, tag_dict, tagids):
    """
    Check if all tags listed are missing. If so, then set event flag to True.

    Args:
        df_wide (pandas.DataFrame): Dataframe with wide format.
        tag_dict (dict): Dictionary of tagid:tagname.
        tagids (list): List of TagIDs.

    Returns:
        bool: True if all tags are missing, False if at least one tag is not missing.
    """
    tags = [tag_dict[tagid] for tagid in tagids]

    if not tags:  # if tags is empty return False
        return False
    elif all(tag not in df_wide.columns for tag in tags):  # if tags are not empty and all tags are 
missing, return True
        logger.warning(f'All of the following columns are missing from the dataframe: {tags}. Setting 
event flag to True.')
        return True
    else:
        return False

def create_primary_event_tags_dict(event_dict, tag_dict, name_ro_process, name_ro_monitoring, name_ro_wq1, 
name_ozone_wq1):
    """
    Create a dictionary of eventid:primary event tag. The primary event tag is the tag that is used to 
determine if the event is occurring or not.

    Args:
        event_dict (dict): Dictionary of eventid:tagids.
        tag_dict (dict): Dictionary of tagid:tagname.
        name_ro_process (str): Tag name of RO Process.
        name_ro_monitoring (str): Tag name of RO Monitoring.
        name_ro_wq1 (str): Tag name of RO WQ1.
```

```
        name_ozone_wq1 (str): Tag name of Ozone WQ1.

    Returns:
        dict: Dictionary of eventid:primary event tag.
    """
    primary_event_tags = {}
    for eventid in event_dict.keys():
        # If an event only has one tag, then that tag is the primary event tag
        if len(event_dict[eventid]) == 1:
            primary_event_tags[eventid] = tag_dict[event_dict[eventid][0]]
        # If an event has multiple tags, set the primary event tag to the name of the calculated tag.
        elif len(event_dict[eventid]) > 1:
            if eventid == 5:
                primary_event_tags[eventid] = name_ro_process
            elif eventid == 6:
                primary_event_tags[eventid] = name_ro_monitoring
            elif eventid == 7:
                primary_event_tags[eventid] = name_ro_wq1
            elif eventid == 15:
                primary_event_tags[eventid] = name_ozone_wq1
            else:
                logger.warning(f'create_primary_event_tags_dict(): Event {eventid} has multiple tags but
no primary event tag has been defined.')
        else:
            primary_event_tags[eventid] = None

    return primary_event_tags

def pecos_check_missing(pm, event_tags):
    """
    Apply Pecos check for missing data to the event tags.

    Args:
        pm (pecos.monitoring.Monitor): Pecos monitoring object.
        event_tags (list): List of event tags.

    Returns:
        pm (pecos.monitoring.Monitor): Pecos monitoring object.
    """
    for tag in event_tags:
        pm.check_missing(tag)

    return pm

def facetplot_by_tag(df):
```

```
    """
    Create a multi-panel plot (facet plot) of the data.

    Args:
        df (pandas.DataFrame): Dataframe in tall format with columns: Datetime, Tag, Value.

    Returns:
        None.
    """

    # Create the FacetGrid
    g = sns.FacetGrid(data=df, col='Tag', hue='Tag', col_wrap=1, aspect=2.5, sharey=False)
    g.map_dataframe(sns.lineplot, x='Datetime', y='Value')
    g.set_axis_labels("", "")
    g.add_legend()

    # Rotate x-axis labels in FacetGrid
    for ax in g.axes.flat:
        for label in ax.get_xticklabels():
            label.set_rotation(90)

# Folder management functions
def reset_directory(folder_path):
    """
    Deletes all files and subfolders in the specified folder.
    If the folder does not exist, creates one.
    """
    if not os.path.exists(folder_path):
        os.makedirs(folder_path)
    else:
        for file_name in os.listdir(folder_path):
            file_path = os.path.join(folder_path, file_name)
            try:
                if os.path.isfile(file_path):
                    os.remove(file_path)
                elif os.path.isdir(file_path):
                    reset_directory(file_path)
                    os.rmdir(file_path)
            except Exception as e:
                logger.warning(f"Error deleting {file_path}: {e}")
```

Figure D-2. Python Script "library.py" with Library of Functions for the EDS Dashboard.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jozefowicz, R., Jia, Y., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Schuster, M., Monga, R., Moore, S., Murray, D., Olah, C., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y. and Zheng, X. 2015. *TensorFlow: Large-scale machine learning on heterogeneous systems.* USENIX Association, 12th USENIX Symposium on Operating Systems Design and Implementation 265-278.

Affleck, M., Allen, G., Bozalongo, C.B., Brown, H., Gasson, C., Gasson, J., González-Manchón, C., Hudecova, M., McFie, A., Shuttleworth, H., Tan, M., Underwood, B., Uzelac, J., Virgili, F. and Walker., C. 2015. *Water's Digital Future*. Global Water intelligence.

Bartos, M., Mullapudi, A. and Troutman, S.C. 2019. "Implementation of the Robust Random Cut Forest algorithm for anomaly detection on streams." *Journal of Open Source Software*, 4(35): 1336.

Bartos, M. and Kerkez, B. 2021. "Observability-Based Sensor Placement Improves Contaminant Tracing In River Networks." *Water Resources Research,* 57(7).

Breitner, L. 2017. *Rejection of low molecular weight neutral organics by reverse osmosis membranes for potable reuse.* University of New Mexico.

Chen, E.C., Pisarenko, A.N., Kolakovsky, A., Howe, E.W., Trussell, R.S. and Trussell, R.R. 2020. "Evaluation of Four Dissolved Ozone Residual Meters' Performance and Disinfection Credits in Potable Reuse Applications". *Ozone-Science & Engineering*, 42(3): 213-229.

Dadakis, J. 2014. *Demonstration of GWRS process performance*, San Diego, CA.

Debroux, J. 2021. *Defining Potential Chemical Peaks and Management Options*. Water Research Foundation, Project No. 4991. https://www.waterrf.org/research/projects/defining-potential chemical-peaks-and-management-options

Hall, J., Panguluri, S., Murray, R. and Burkhardt, J. 2017. "CANARY Eases Water Quality Event Detection". *Opflow*, 43(4): 30-32.

Kaelin, D., Rieger, L., Eugster, J., Rottermann, K., Banninger, C. and Siegrist, H. 2008. "Potential of in-situ sensors with ion-selective electrodes for aeration control at wastewater treatment plants". *Water Science and Technology*, 58(3): 629-637.

Klise, K.A., Stein, J.S. 2016. Automated Performance Monitoring for PV Systems using Pecos, 43th Photovoltaic Specialists Conference (PVSC), Portland, OR, June 5-10.

Klise, K.A. and McKenna, S.A. 2006a. Multivariate applications for detecting anomalous water quality. *8th Annual Water Distribution Systems Analysis (WDSA) Symposium*. Cincinnati, OH.: ASCE

Klise, K.A. and McKenna, S.A. 2006b. Water quality change detection: multivariate algorithms. *SPIE Defense and Security Symposium*. Orlando, FL.: SPIE

Labbe Jr., R.R. 2015a. *FilterPy*, GitHub.

Labbe Jr., R.R. 2015b *Kalman and Bayesian Filters in Python*, GitHub.

Leow, A., Burkhardt, J., Platten, W.E., Zimmerman, B., Brinkman, N.E., Turner, A., Murray, R., Sorial, G. and Garland, J. 2017. "Application of the CANARY event detection software for real-time performance monitoring of decentralized water reuse systems." *Environmental Science-Water Research &Technology,* 3(2): 224-234.

McKenna, S.A., Hart, D.B., Klise, K.A., Cruz, V.A. and Wilson, M.P. 2007. Event Detection from Water Quality Time Series. *ASCE World Environmental and Water Resources Congress*. Tampa, FL.: ASCE

Nair, A.M., Fanta, A., Haugen, F.A. and Ratnaweera, H. 2019. "Implementing an Extended Kalman Filter for estimating nutrient composition in a sequential batch MBBR pilot plant". *Water Science and Technology*, 80(2): 317-328.

Neemann, J., DeCarolis, J., ten Bosch, D., Snyder, S.A., Pepper, I.L. and Minkyu, P. 2019. *Integrating Management of Sensor Data for a Real Time Decision Making and Response System.* Water Environment & Reuse Foundation.

Nilsson, P., Roser, D., Thorwaldsdotter, R., Petterson, S., Davies, C., Signor, R., Bergstedt, O. and Ashbolt, N. 2007. "SCADA data and the quantification of hazardous events for QMRA". *J. Water Health*, 5: 99-105.

Patel, M. and Dadakis, J. 2018. *Practical operational approaches to meet pathogen LRV requirements for GWRS*, Monterey, CA.

Pecson, B.M., Trussell, R.S., Pisarenko, A.N. and Trussell, R.R. 2015a. "Achieving Reliability in Potable Reuse: The Four Rs"*. Journal - American Water Works Association.*

Pecson, B.M., Trussell, R.S., Pisarenko, A.N., Howe, E., Idica, E. and Triolo, S. 2015b. *Draft Experimental Plan for Reliable Potable Reuse Demonstration.*

Pecson, B.M., Triolo, S.C., Olivieri, S., Chen, E.C., Pisarenko, A.N., Yang, C.C., Olivieri, A., Haas, C.N., Trussell, R.S. and Trussell, R.R. 2017. "Reliability of pathogen control in direct potable reuse: Performance evaluation and QMRA of a full-scale 1 MGD advanced treatment train". *Water Research*, 122: 258-268.

Pecson, B.M., Chen, E.C., Triolo, S.C., Pisarenko, A.N., Olivieri, S., Idica, E., Kolakovsky, A., Trussell, R.S. and Trussell, R.R. 2018. "Mechanical Reliability in Potable Reuse: Evaluation of an Advanced Water Purification Facility". *Journal - American Water Works Association*, 110(4): E19-E28.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M. and Duchesnay, É. 2011. "Scikit-learn: Machine Learning in Python". *Journal of Machine Learning Research*, 12: 2825-2830.

Rajakumar, A.G., Kumar, M.S.M., Amrutur, B. and Kapelan, Z. 2019. "Real-Time Water Quality Modeling with Ensemble Kalman Filter for State and Parameter Estimation in Water Distribution Networks". *J. Water Resour. Plan. Manage.-ASCE*, 145(11).

Snyder, S.A. and Pepper, I.L. 2016. *Monitoring for Reliability and Process Control of Potable Reuse Applications.* Water Environment & Reuse Foundation.

SWRCB, 2018. *Regulations Related to Recycled Water*. California Division of Drinking Water.

State Water Board, 2021. *DPR Framework 2nd edition Addendum – Early Draft of Anticipated Criteria for Direct Potable Reuse.*

Tackaert, R.A., Pisarenko, A.N., Chen, E.C., Kolakovsky, A., Pecson, B.M., Drewes, J.E., Trussell, R.R. and Trussell, R.S. 2019. "Demonstrating process robustness of potable reuse trains during challenge testing with elevated levels of acetone, formaldehyde, NDMA, and 1,4-dioxane". *Journal of Water Supply: Research and Technology – AQUA*, 68(5): 313-324.

Taylor, S.J. and Letham, B. 2017. *Forecasting at scale*. PeerJ Preprints 5.

The Economist. 2018. *Python is becoming the world's most popular coding language.*

Trussell, R.S., Pecson, B.M., Pisarenko, A.N., Idica, E.Y., Howe, E.W. and Trussell, R. 2017. *Demonstrating redundancy and monitoring to achieve reliable potable reuse: Final Report for WE&RF Project 14-12 / WRF Project No. 4765. https://www.waterrf.org/research/projects/san-diego-dpr*

U.S. EPA. 2005. *Membrane Filtration Guidance Manual.*

U.S. EPA. 2010. *Long Term 2 Enhanced Surface Water Treatment Rule Toolbox Guidance Manual.*

U.S. EPA 2012. *CANARY User's Manual version 4.3.2*. EPA/600/R-08/040B.

Virtanen, P., Gommers, R., Oliphant, T.E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., Walt, S.J.v.d., Brett, M., Wilson, J., Millman, K.J., Mayorov, N., Nelson, A.R.J., Jones, E., Kern, R., Larson, E., Carey, C., Polat, İ., Feng, Y., Moore, E.W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E.A., Harris, C.R., Archibald, A.M., Ribeiro, A.H., Pedregosa, F., Mulbregt, P.v. and Contributors, S. 2020. "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python". *Nature Methods*, 17(3): 261-272.

Walker, T., Stanford, B.D., Khan, S., Robillot, C., Snyder, S.A., Valerdi, R., Dwivedi, S. and Vickers, J. 2016. *Critical Control Point Assessment to Quantify Robustness and Reliability of Multiple Treatment Barriers of a DPR Scheme.* Water Environment & Reuse Foundation.

Wickline, J. and Hopkinson, L. 2020. “Detection of spills related to natural gas production”. *Water Environ. Res.*, 92(8): 1104-1110.